Invertibility and Estimation of Two-Parameter Polynomial and Division Lens Distortion Models

7 authors, including:

D. Santana-Cedrés
Universidad de Las Palmas de Gran Canaria
25 PUBLICATIONS   204 CITATIONS

SEE PROFILE

Luis Gomez Deniz
Universidad de Las Palmas de Gran Canaria
89 PUBLICATIONS   471 CITATIONS

SEE PROFILE

Miguel Alemán-Flores
Universidad de Las Palmas de Gran Canaria
53 PUBLICATIONS   338 CITATIONS

SEE PROFILE

Julio Esclarín Monreal
Universidad de Las Palmas de Gran Canaria
27 PUBLICATIONS   205 CITATIONS

SEE PROFILE

Some of the authors of this publication are also working on these related projects:

SAR/PolSAR View project

Sistema de procesado de imágenes para el diagnóstico, la planificación, la simulación y el seguimiento de operaciones de cirugía vascular (SIMVA) (TIN2009-10770) View project

Invertibility and estimation of two-parameter polynomial and division lens distortion models

Daniel Santana-Cedrés[†]   Luis Gomez[‡]   Miguel Alemán-Flores[†]   Agustín Salgado[†]

Julio Esclarín[†]   Luis Mazorra[†]   Luis Alvarez[†]

Abstract

In this paper, we study lens distortion for still images considering two well-known distortion models: the two-parameter polynomial model and the two-parameter division model. We study the invertibility of these models and we mathematically characterize the conditions for the distortion parameters under which the distortion model defines a one-to-one transformation. This ensures that the inverse transformation is well-defined and the distortion-free image can be properly computed, which provides robustness to the distortion models. A new automatic method to correct the radial distortion is proposed and a comparative analysis for this method is extensively performed using the polynomial and the division models. With the aim of obtaining an accurate estimation of the model, we propose an optimization scheme which iteratively improves the parameters to achieve a better matching between the distorted lines and the edge points. The proposed method estimates two-parameter radial distortion models by detecting the longest distorted lines within the image. This is done by applying the Hough transform extended with a radial distortion parameter. Next, a two-parameter model is estimated using an iterative non-linear optimization scheme. This scheme aims at minimizing the distance from the edge points to their associated lines by adjusting the two distortion parameters as well as the coordinates of the center of distortion. We present some experiments on real images with significant distortion to show the ability of the proposed approach to correct the radial distortion. A visual and quantitative comparison between both automatic two-parameter model estimations indicates that the division model is more efficient for those images showing strong distortion.

1 Introduction

[26] Radial distortion is a quite frequent type of distortion which causes barrel distortion at short focal lengths and pincushion distortion at long focal lengths. This phenomenon is mainly due to the imperfection of the lens and the misalignment of the optical system, and is embedded in the well-known pinhole camera model by means of a distortion model [11]. In [13], the authors review the most used radial distortion models and illustrate how important a precise correction is in order to assist the automated diagnosis of celiac disease from endoscopic images with strong barrel distortion. In [29], a more comprehensive review of state-of-the-art radial distortion models is discussed. An important contribution of this extended review lies on the proposal of a self-consistency and universality validation of radial distortion correction in order to allow a fair (unbiased) comparison among models. Self-consistency is evaluated by the residual error when the distortion generated with a certain model is corrected by the best parameters for such model. Universality is measured by the residual error when a model is used to correct the distortions generated by a family of other models.

The general equation of a lens distortion model is given by the equation

$$\begin{pmatrix} \hat{x} - x_c \\ \hat{y} - y_c \end{pmatrix} = L(r) \begin{pmatrix} x - x_c \\ y - y_c \end{pmatrix}, \tag{1}$$

where (x_c, y_c) represents the distortion center, (x, y) is a point in the image domain, $(\hat{x}, \hat{y})$ is the transformed point, $r = \|(x, y) - (x_c, y_c)\|$, and $L(r)$ represents the shape of the distortion model. Two types of radial lens distortion models are the most frequently applied in computer vision due to their excellent trade-off between accuracy and

---
[†]CTIM (Centro de Tecnologías de la Imagen), Departamento de Informática y Sistemas. Universidad de Las Palmas de Gran Canaria, Spain.
dsantana@ctim.es, maleman@ctim.es, asalgado@ctim.es, jesclarin@ctim.es, lmazorra@ctim.es, lalvarez@ctim.es
[‡]CTIM (Centro de Tecnologías de la Imagen), Departamento de Ingeniería Electrónica y Automática. Universidad de Las Palmas de Gran Canaria, Spain. lgomez@ctim.es

easy calculation: the polynomial model and the division model. The polynomial model (or simple radial distortion model [18]), using two parameters, is formulated as

$$L(r) = 1 + k_1 r^2 + k_2 r^4. \tag{2}$$

Two-parameter models have been used in the literature due to their simplicity and accuracy ([3], [20]). In [3], an algebraic method is proposed which is suitable for correcting significant radial distortion as well as highly efficient in terms of computational cost. An on-line demo of the implementation of this algebraic method can be found in [4].

The division model was initially proposed in [23], but has received special attention after the research by Fitzgibbon [12]. In the case of two parameters, it is formulated as

$$L(r) = \frac{1}{1 + k_1 r^2 + k_2 r^4}. \tag{3}$$

The main advantage of the division model is the requirement of fewer terms than the polynomial model to cope with images showing severe distortion. Therefore, the division model seems to be more adequate for wide-angle lenses (see a recent review on distortion models for wide-angle lenses in [19]).

For both models, $L(r)$ can be estimated by considering that 3D lines in the image must be projected onto 2D straight lines, and minimizing the distortion error, which is given by the sum of the squares of the distances from the points to the lines [10]. In [5], [6] and [28], the user must identify some known straight lines, which makes these methods robust, independent of camera parameters, and no calibration pattern is required. However, these methods are slow for the case of dealing with a large set of images.

New automatic methods without user intervention have recently emerged. In [7], an automatic method to estimate radial distortion models working on a single image without human intervention is discussed. This approach applies Fitzgibbon's one-parameter division model to estimate the distortion from a set of automatically detected non-overlapping circular arcs within the image.

In [16], a new automatic approach to correct lens distortion is discussed, which differs considerably from the standard approach, thus being a truly new methodology, not only for lens distortion but also for camera calibration. The proposed method is non-parametric, non-iterative and model-free. Even the calibration pattern used is completely different from standard models: it consists of a highly textured planar pattern obtained by printing a textured image on a flat object. Then, by taking two photographs of this pattern in a fixed lens configuration, a *tracking* of the distorted points is obtained and the points are classified applying the SIFT (Scale-invariant feature transform) method. Wrong SIFT matches (outliers) are disregarded through an iterative algorithm which matches points using retroprojection of the points by estimating a homography by RANSAC algorithm and rejecting all points not compatible with this homography. From that, a distortion field is built and then, its inverse is applied for image correction. Results are compared with the *classical* results from [21] achieving a precision, in terms of the RMS (Root Mean Squared) error, slightly outperforming the state of the art global calibration methods obtained by Lavest (RMS of 0.08 pixels).

In [17], a similar approach (a *model-free* method) is proposed and compared with [16] to show that using a higher order polynomial (degree 7 to 11) to model the distortion provides better results. An energy function which measures the distance from the undistorted points to their associated lines similar to the one in [3] is defined and optimized using the Levenberg-Marquardt algorithm in an iterative way. To refine the solution, the algorithm first estimates a 3-order polynomial, this solution is used to initialize a 4-order polynomial, and this process continues until minimizing an 11-order polynomial. Additionally, it is worth mentioning that the method uses a real plumb line pattern calibration: a harp built with good quality strings which have been tightly stretched to ensure straightness. A detailed study about calibration harp reliability in the context of high-precision camera distortion measurements has been presented in [27].

The extension of the *classical* Hough transform including a distortion model has become a usual approach for automatic methods. These methods enrich the standard Hough space by including a set of distortion parameters, which allows a better recognition of the existing straight lines. Basically, the difference between the models lies in the implementation or in the radial distortion model which is applied.

In [8], the authors propose an automatic method for radial lens distortion correction using an adapted Hough transform including the radial distortion parameter to automatically detect straight lines within the image. The one-parameter radial model is used and an exhaustive search is performed testing each distortion parameter in a discretized set within the range (k_{min}, k_{max}) and selecting the one that provides the best fitting (i.e., which maximizes the *straightness* of the candidate lines obtained from the corrected points). The method works automatically but better results are obtained using a semi-automatic version which includes the manual selection of a ROI (region of interest) containing some known straight lines.

A fully automatic distortion correction method, which also embeds the radial distortion parameter into the Hough transform to better detect straight lines, is presented in Lee et al. [22]. It is applied to endoscopic images

captured with a wide-angle *zoomable* lens. In this case, the division model with one parameter is used to deal with the strong distortion caused by the wide-angle lens. The method is adapted to also include the effects related to the variation of the focal length due to zooming operations. The method is intended for real time applications once mapped to a GPU computing platform. The distortion parameter is estimated by optimizing the Hough entropy in the image gradient space.

Based on [1, 2], we propose a new automatic method to correct the radial distortion in still images. As main differences with [1, 2], in this paper we consider both polynomial and division models, and estimate two distortion parameters, instead of just one-parameter division models. Furthermore, a new efficient iterative algorithm is used to get optimal distortion parameters as well as the center of distortion (in [2], the center of distortion is assumed to be the center of the image). We show that, by using the optimization scheme, more lines (and longer) are extracted, which implies a better final undistorted image.

Another contribution of this work is that we mathematically characterize the conditions for the distortion parameters under which the distortion model defines a one-to-one transformation. This ensures that the inverse transformation required to undistort the image is well-defined for the two-parameter models, and the distortion-free image can properly be computed which provides robustness to the distortion models.

Although we only discuss two-parameter models, the proposed optimization method can be applied to estimate models with more distortion parameters. However, in the case of models using more that two distortion parameters, the mathematical study of the invertibility of the distortion should be explored. In this paper, the conditions to guarantee the inversion of the models have been obtained *only* for the case of dealing with two parameters.

This paper is organized as follows: In Section 2, we study the invertibility of the lens distortion model. In Section 3, we present the iterative optimization method proposed to obtain the two distortion parameters and the center of distortion. In Section 4, we present a large variety of experiments to illustrate the performance of the proposed method, including a comparison between division and polynomial models. Finally, in Section 5, we present our main conclusions.

2   Invertibility of lens distortion models

A two-parameter radial lens distortion model defines a transformation on the image domain. In this section, we study the invertibility of such transformation according to the values of the distortion parameters k_1, k_2. Since we deal with radial transformations, the invertibility of lens distortion models is equivalent to the invertibility of the associated radial transformation

$$\hat{r}(r, k_1, k_2) = r \cdot L(r, k_1, k_2), \tag{4}$$

which determines the way the distance r from a point to the distortion center is modified when the lens distortion model is applied. The domain of definition of $\hat{r}(r, k_1, k_2)$ with respect to r is the interval $[0, r_1]$, where r_1 is defined by

$$r_1 = max\{\|(x, y) - (x_c, y_c)\| : (x, y) \in \text{Image Domain}\}. \tag{5}$$

In the next theorems, we present some results concerning the invertibitiy of $\hat{r}(r, k_1, k_2)$ according to the distortion parameters k_1, k_2 for the polynomial and division model.

Theorem 1 (Invertibility of polynomial distortion models) *For any $k_1, k_2 \in R$, the function*

$$\hat{r}(r, k_1, k_2) = r(1 + k_1 r^2 + k_2 r^4) \tag{6}$$

is a one-to-one function with respect to r in the interval $[0, r_1]$ if and only if k_1, k_2 satisfy

$$\begin{cases} 9r_1^4 k_1^2 - 20r_1^4 k_2 < 0 & if \quad r_1^2 k_1 < -\frac{2}{3} \\ 5r_1^4 k_2 + 3r_1^2 k_1 + 1 > 0 & if \quad r_1^2 k_1 \geq -\frac{2}{3}. \end{cases} \tag{7}$$

Moreover, if k_1, k_2 satisfy (7), for any $r' \in (0, \hat{r}(r_1, k_1, k_2)]$, there exists a unique $r \in (0, r_1]$ such that $\hat{r}(r, k_1, k_2) = r'$, given by a root of the polynomial

$$P(r) = k_2 r^5 + k_1 r^3 + r - r' = 0. \tag{8}$$

Proof: The derivative of $\hat{r}(r, k_1, k_2)$ with respect to r is given by

$$\frac{\partial \hat{r}(r, k_1, k_2)}{\partial r} = 1 + 3k_1 r^2 + 5k_2 r^4. \tag{9}$$

We observe that the above derivative is positive in $r = 0$. Thus, the function $\hat{r}(r, k_1, k_2)$ will be one-to-one in the interval $[0, r_1]$ iff $\hat{r}(r, k_1, k_2)$ is an increasing function with respect to r in the interval $[0, r_1]$. This is equivalent to the fact that the above derivative is positive in the interval $[0, r_1]$ and, therefore, its roots must be outside the interval $[0, r_1]$. Formally, the potential positive real roots of the polynomial in (9) are given by

$$
\begin{cases}
\sqrt{\frac{-3k_1 \pm \sqrt{9k_1^2 - 20k_2}}{10k_2}} & if \quad k_2 \neq 0, \ 9k_1^2 - 20k_2 \geq 0 \ and \ \frac{-3k_1 \pm \sqrt{9k_1^2 - 20k_2}}{10k_2} \geq 0 \\
\sqrt{-\frac{1}{3k_1}} & if \quad\quad\quad\quad\quad k_2 = 0 \ and \ k_1 < 0.
\end{cases}
\tag{10}
$$

Next, we consider the case $r_1 = 1$ and we study the regions in the k_1, k_2 parameter space where all the real roots of (9) are outside the interval $[0, 1]$. To find such regions we will consider the following level curves in the k_1, k_2 space

$$
\begin{cases}
k_2 = 0 \\
9k_1^2 - 20k_2 = 0 \\
\frac{-3k_1 \pm \sqrt{9k_1^2 - 20k_2}}{10k_2} = 1.
\end{cases}
\tag{11}
$$

We observe that the last equation is equivalent to

$$
9k_1^2 - 20k_2 = (10k_2 + 3k_1)^2,
$$

and a straightforward computation yields that this equation is equivalent to

$$
5k_2 + 3k_1 + 1 = 0.
$$

In figure 1, we illustrate these level curves. The pink area represents the regions where all real roots of the polynomial (9) are outside the interval $[0, 1]$ and, therefore, the function $r \to \hat{r}(r, k_1, k_2)$ is one-to-one in the interval $[0, 1]$. We observe that, in case $r_1 = 1$, this area corresponds to the one given by the expression (7), which proves the statement of the theorem for the case $r_1 = 1$. For the general case $r_1 \neq 1$, we first observe that the function $r \to \hat{r}(r, k_1, k_2)$ is one-to-one in the interval $[0, r_1]$ if and only if the function $\tilde{r} \to \hat{r}(r_1\tilde{r}, k_1, k_2)$ is one-to-one in the interval $[0, 1]$. On the other hand, taking into account (6), we obtain

$$
\hat{r}(r_1\tilde{r}, k_1, k_2) = r_1 \cdot \hat{r}(\tilde{r}, k_1 r_1^2, k_2 r_1^4),
$$

and, therefore, by applying the result obtained for the case $r_1 = 1$ to the function $\tilde{r} \to \hat{r}(\tilde{r}, k_1 r_1^2, k_2 r_1^4)$, we obtain (7). Moreover, if the function $r \to \hat{r}(r, k_1, k_2)$ is one-to-one, using (6) we obtain that the inverse function can be computed using the roots of the polynomial $P(r)$ given by (8), which concludes the proof of the theorem.

Theorem 2 (Invertibility of division distortion models) *For any $k_1, k_2 \in R$, the function*

$$
\hat{r}(r, k_1, k_2) = \frac{r}{1 + k_1 r^2 + k_2 r^4}
\tag{12}
$$

is a one-to-one function with respect to r in the interval $[0, r_1]$ if and only if $k_1 r_1^2 > -2$ and

$$
\begin{cases}
-1 - r_1^2 k_1 < r_1^4 k_2 < \frac{1 - r_1^2 k_1}{3} & if \quad -2 < r_1^2 k_1 < 2 \\
-1 - r_1^2 k_1 < r_1^4 k_2 < -\frac{r_1^4 k_1^2}{12} & if \quad\quad r_1^2 k_1 \geq 2.
\end{cases}
\tag{13}
$$

Moreover, if k_1, k_2 satisfy (13), for any $r' \in (0, \hat{r}(r_1, k_1, k_2)]$, there exists a unique $r \in (0, r_1]$ such that $\hat{r}(r, k_1, k_2) = r'$ and it is given by a root of the polynomial

$$
P(r) = r' k_2 r^4 + r' k_1 r^2 - r + r' = 0.
\tag{14}
$$

Proof: We follow the same technique used in theorem 2 adapted to the division model. The derivative of $\hat{r}(r, k_1, k_2)$ with respect to r is given by

$$
\frac{\partial \hat{r}(r, k_1, k_2)}{\partial r} = \frac{1 - k_1 r^2 - 3k_2 r^4}{(1 + k_1 r^2 + k_2 r^4)^2}.
\tag{15}
$$

In this case, we have to consider the roots of the numerator and denominator of the above expression and we obtain
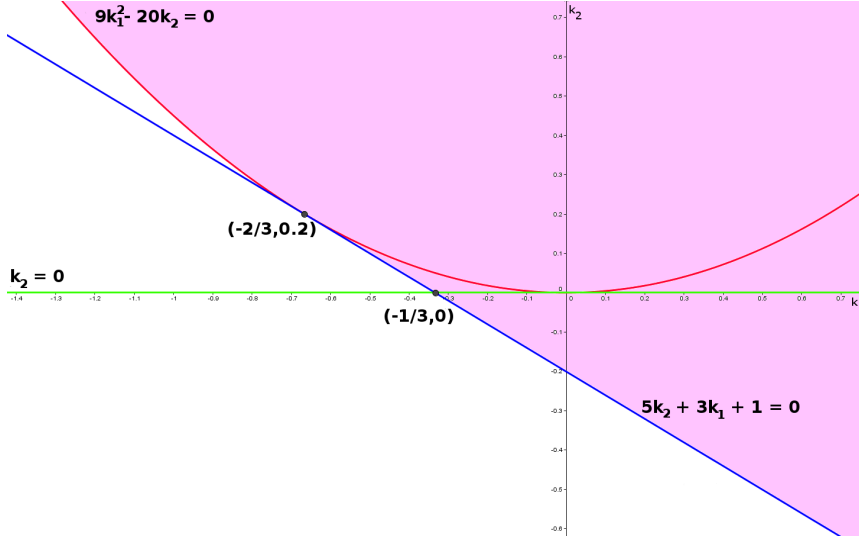
4

Figure 1: Polynomial model: level curves used to study the regions where all roots of the derivative function (9) are outside the interval $[0, 1]$. The pink area represents the regions where all roots are outside $[0, 1]$.

$$\begin{cases} \sqrt{\dfrac{-k_1 \pm \sqrt{k_1^2 - 4k_2}}{2k_2}} & if \quad k_2 \neq 0, \ k_1^2 - 4k_2 \geq 0 \ and \ \dfrac{-k_1 \pm \sqrt{k_1^2 - 4k_2}}{2k_2} \geq 0 \\[3mm] \sqrt{\dfrac{-k_1 \pm \sqrt{k_1^2 + 12k_2}}{6k_2}} & if \quad k_2 \neq 0, \ k_1^2 + 12k_2 \geq 0 \ and \ \dfrac{-k_1 \pm \sqrt{k_1^2 + 12k_2}}{6k_2} \geq 0 \\[3mm] \sqrt{\pm \dfrac{1}{k_1}} & if \qquad\qquad k_2 = 0 \ \ and \ \ k_1 \neq 0. \end{cases} \tag{16}$$

Next, we initially assume that $r_1 = 1$ and we consider the following level curves in the k_1, k_2 space separating the different regions with respect to the roots

$$\begin{cases} k_2 = 0 \\[2mm] k_1^2 - 4k_2 = 0 \\[2mm] \dfrac{-k_1 \pm \sqrt{k_1^2 - 4k_2}}{2k_2} = 1 \quad \rightarrow \quad k_1 + k_2 + 1 = 0 \\[2mm] k_1^2 + 12k_2 = 0 \\[2mm] \dfrac{-k_1 \pm \sqrt{k_1^2 + 12k_2}}{6k_2} = 1 \quad \rightarrow \quad k_1 + 3k_2 - 1 = 0. \end{cases} \tag{17}$$

In figure 2, we show these level curves. The pink area represents the regions where all real roots of the polynomials in (15) are outside the interval $[0, 1]$ and, therefore, the function $r \to \hat{r}(r, k_1, k_2)$ is one-to-one in the interval $[0, 1]$. For the general case $r_1 \neq 1$, as in the previous theorem, we prove (13) by reducing the problem to the case $r_1 = 1$ using a change of variable. Using (12), we obtain that the inverse function can be computed using the roots of the polynomial $P(r)$ given by (8), which concludes the proof of the theorem.

The results showed in the above theorems allow us to easily check if a given parameter configuration k_1, k_2 defines a one-to-one lens distortion transformation and the inverse transformation is well-defined. Otherwise, the lens distortion transformation generates singularities and we cannot properly compute the distortion-free image because the inverse of the lens distortion transformation is required.

5

k_2

$k_1^2 - 4k_2 = 0$

$(-2,1)$

$k_2 = 0$

$(1,0)$

$(2,-1/3)$

$k_1 + 3k_2 - 1 = 0$

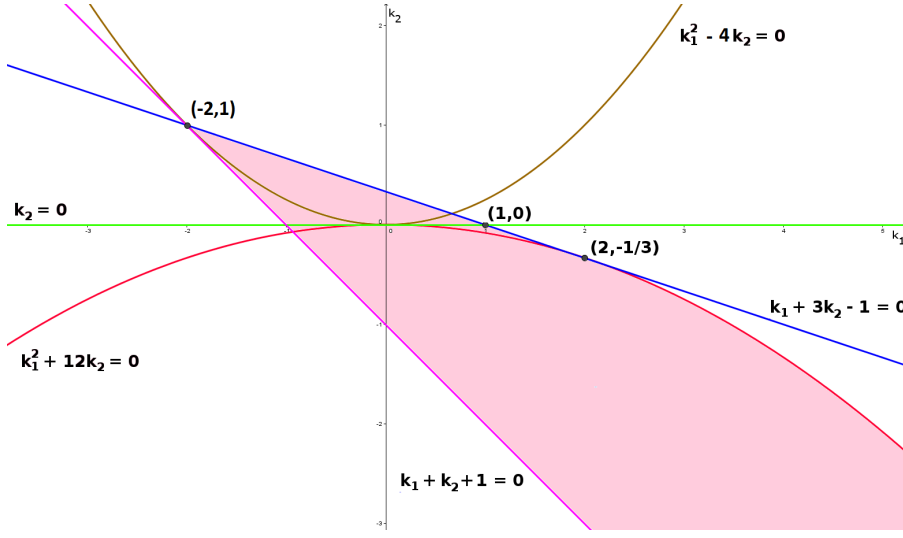$k_1^2 + 12k_2 = 0$

k_1

$k_1 + k_2 + 1 = 0$

Figure 2: Division model : level curves used to study the regions where all roots of the polynomials in (15) are outside the interval $[0, 1]$. The pink area represents the regions where all roots are outside $[0, 1]$.

3   Lens distortion estimation and correction

In this section, we present the approach we use to estimate the lens distortion model and correct the distortion. First, we extract an initial collection of distorted lines using a one-parameter model. Afterward, with the same collection of primitives, a two-parameter model is estimated by minimizing an energy function given by the distance from the corrected points to their associated lines. In this stage, we perform a normalization of the distortion parameters so that the method is independent of the image resolution. Then, an iterative optimization method is applied to obtain a better collection of lines and, therefore, improve the two-parameter model. Finally, the distortion model which has been calculated is used to correct the distortion of the images. Figure 3 illustrates the different stages for the estimation and correction of the distortion. Next, we will present a more detailed description of the different steps of the proposed method.

3.1   Initial estimation of the collection of lines and a single parameter division model

Different strategies to find aligned structures in an image are possible. In [9], a parameterless method to detect segments which guarantees that no more than one segment on the average will be wrongly detected (*false positive*) is discussed. This work is based on the Helmholtz principle, which states that any observed geometric structure (*in an image*) is perceptually meaningful if the expectation of its occurrences is small in a random image. This approach has been extended in [15, 14], where the authors introduce the LSD (Line Segment Detector) method. This method detects segments in an image in a very fast and effective way. In this paper, as we deal with distorted lines, we use an extension of the Hough transform which allows extracting aligned structures distorted by a one-parameter distortion model. The traditional Hough transform consists in searching for certain shapes which can be expressed in a parameterized way and determining which values of the parameters are the most reliable ones, according to a voting scheme given by the number of edge points which fit into the shapes. The higher the number of votes, the more significant the shape. In the case of straight lines, this can easily be implemented by considering only two parameters, namely the orientation and the distance to the origin. When perfect straight lines are present in the scene, the votes are agglutinated around the right values. However, the lens distortion curves the projected lines, so that long lines cannot be extracted by this method because they are not projected on the camera plane as straight lines. This can be tackled by introducing a new parameter in the Hough searching space. This new parameter is the radial distortion parameter. Therefore, a line in this space is described by a triplet (d, α, k_1) representing the distance to the origin, line orientation and radial distortion.

An extension of the Hough transform which includes a single radial distortion parameter to extract lines in

images with radial distortion has been applied in several works with the purpose of correcting the distortion. In the work by Alemán-Flores et al. [2], this is first done and then a polynomial model with two parameters is used. The same approach is applied in [1], but, in this case, a one-parameter division model is applied to correct the distortion. We refer the reader to these papers for details. At this stage, it is assumed that the distortion which affects all lines within the image is given by a one-parameter model. Hence, a single value for the distortion parameter is estimated for the whole image. In this sense, due to the fact that a single parameter is considered in the first step of the proposed method, [1] can be regarded as the initial step, but applying the corresponding model (polynomial or division).

We also assume that the distortion center is the image center, which is usually a good approximation. The selection of the longest lines is performed through the usual voting scheme related to the standard Hough transform: first, we extract the edge points and estimate the edge point orientations. The proposed technique would work with any method which provides this information, but in our case, we are using our own implementation of the Canny edge detector. Secondly, for each value of the distortion parameter k_1 in the Hough space, we correct the edge point coordinates and orientation using the lens distortion model associated to k_1. Each edge point votes for the set of lines in the Hough space with a similar orientation to the edge point orientation. Moreover, the vote of a point for a line depends on how close they are, and is given by $v = 1/(1 + d_{k_1})$, where d_{k_1} is the distance from the point to the line. Next, from the voting score matrix, we select the n most voted lines for each possible value of the radial distortion parameter.

In order to select a value for the distortion parameter, we search for the value which best corrects the lines, i.e. which provides the longest lines after correction. To compare the possible values, we consider the n longest lines which can be extracted using each value and we add the scores of these lines (as explained above, this score depends on the number of points associated to that line and the distance from the points to the line). This measure for the reliability of each value of the distortion parameter favors the values for which the longest lines have been detected. The reliability measure v_i for the ith value of the first distortion parameter is

$$v_i = \sum_{j=1}^{n} s_j^i, \tag{18}$$

where s_j^i is the total score of the jth line of the ith value.

As a summary, this stage provides a first approximation of a one-parameter distortion model, the collection of the longest lines within the image associated to such model (we denote by Nl the number of lines) and, for each line j, we associate the collection of edge points $\{\bar{x}_{ji} = (x_{ji}, y_{ji})\}_{i=1,..,N(j)}$ which vote for the line. It is important to remark that this simple model is not optimized as our interest lies only in extracting an initial set of lines. In the following sections, from this set of lines, we use a two-parameter model, and an optimization for the distortion parameters and the center of distortion is discussed.

In figure 4, we show a graphic example of applying this step to an image of a calibration pattern. On the left, the image used for the voting process is depicted. On the center column, we show the voting matrix for each value of k_1, with a highlighted region, which is enlarged on the right. The votes are shown using a gray scale inside each matrix. The darker the color, the higher the score. As observed, for the value of k_1 which reaches the highest voting score, the votes are concentrated around the values corresponding to the lines instead of being dispersed.

3.2 Parameter normalization

In order to normalize the parameters k_1 and k_2, so that the method is independent of the image resolution, and avoid working with too small values, we introduce the parameters p_1 and p_2 instead. The value of p_1 represents the percentage of correction of the furthest point in the image from the center of distortion, and p_2 represents the same percentage of correction, but for the midpoint between the center of distortion and the furthest point. This way, the parameters are easier to interpret and do not depend on the image resolution. In what follows, we will denote as r_1 the distance from the center of distortion to the furthest point in the image domain given by (5), and as r_2 half of this distance.

Polynomial model

According to the equations for the polynomial distortion model, the values described above are related as follows

$$
\begin{aligned}
(1 + p_1)r_1 &= r_1(1 + k_1 r_1^2 + k_2 r_1^4), \\
(1 + p_2)r_2 &= r_2(1 + k_1 r_2^2 + k_2 r_2^4),
\end{aligned}
\tag{19}
$$

and simplifying

$$p_1 = k_1 r_1^2 + k_2 r_1^4,$$
$$p_2 = k_1 r_2^2 + k_2 r_2^4. \tag{20}$$

Using the relation $r_1 = 2r_2$, we can rewrite the equations as

$$p_1 = k_1 4r_2^2 + k_2 16r_2^4,$$
$$p_2 = k_1 r_2^2 + k_2 r_2^4. \tag{21}$$

According to this, we have the system

$$\begin{bmatrix} 4r_2^2 & 16r_2^4 \\ r_2^2 & r_2^4 \end{bmatrix} \begin{bmatrix} k_1 \\ k_2 \end{bmatrix} = \begin{bmatrix} p_1 \\ p_2 \end{bmatrix}, \tag{22}$$

which provides the following values for k_1 and k_2

$$k_1 = \frac{p_1 - 16p_2}{-12r_2^2}, \qquad k_2 = \frac{4p_2 - p_1}{-12r_2^4}. \tag{23}$$

Therefore, using the expressions above, we can replace k_1, k_2 by p_1, p_2 as distortion parameters.

Division model

In the case of the two-parameter division distortion model, the relation between the parameters changes according to the division model:

$$(1 + p_1)r_1 = \frac{r_1}{1 + k_1 r_1^2 + k_2 r_1^4}, \tag{24}$$
$$(1 + p_2)r_2 = \frac{r_2}{1 + k_1 r_2^2 + k_2 r_2^4},$$

and, therefore

$$p_1 = \frac{1}{1 + k_1 r_1^2 + k_2 r_1^4} - 1, \tag{25}$$
$$p_2 = \frac{1}{1 + k_1 r_2^2 + k_2 r_2^4} - 1.$$

For computing k_1 and k_2, we have a similar system as above

$$\begin{bmatrix} r_1^2 & r_1^4 \\ r_2^2 & r_2^4 \end{bmatrix} \begin{bmatrix} k_1 \\ k_2 \end{bmatrix} = \begin{bmatrix} \frac{-p_1}{1+p_1} \\ \frac{-p_2}{1+p_2} \end{bmatrix}. \tag{26}$$

Solving this system, and considering the relation $r_1 = 2r_2$, we obtain

$$k_1 = \frac{\frac{-p_1}{1+p_1} + \frac{16p_2}{1+p_2}}{-12r_2^2}, \qquad k_2 = \frac{\frac{-4p_2}{1+p_2} + \frac{p_1}{1+p_1}}{-12r_2^4}. \tag{27}$$

3.3   Two-parameter lens distortion estimation for a given collection of distorted lines

We denote by $\mathbf{u}$ the tuple (p_1, p_2, x_c, y_c), which defines the distortion model, and by $\hat{\bar{x}}_{ji}^{\mathbf{u}} = (\hat{x}_{ji}^{\mathbf{u}}, \hat{y}_{ji}^{\mathbf{u}})$ the distortion corrected points obtained using equation (1) with the model given by $\mathbf{u}$. For a given $\mathbf{u}$ and a given line j, the equation of the associated straight line given by $cos(\alpha_j^{\mathbf{u}})x + sin(\alpha_j^{\mathbf{u}})y + d_j^{\mathbf{u}} = 0$ is defined by the pair $(\alpha_j^{\mathbf{u}}, d_j^{\mathbf{u}})$ obtained by minimizing

$$D(\alpha, d) = \sum_{i=1}^{N(j)} \left(cos(\alpha)\hat{x}_{ji}^{\mathbf{u}} + sin(\alpha)\hat{y}_{ji}^{\mathbf{u}} + d \right)^2. \tag{28}$$

This well-known minimization problem has a simple close-form solution (see for instance [11] for more details).

The optimization of the lens distortion model is performed by minimizing the average of the square distance from the corrected primitive points to their associated lines, so that, using the notation introduced above, this optimization consists in minimizing the following energy

$$E(\mathbf{u}) = \frac{\sum_{j=1}^{Nl} \sum_{i=1}^{N(j)} \left(cos(\alpha_j^{\mathbf{u}}) \hat{x}_{ji}^{\mathbf{u}} + sin(\alpha_j^{\mathbf{u}}) \hat{y}_{ji}^{\mathbf{u}} + d_j^{\mathbf{u}} \right)^2}{\sum_{j=1}^{Nl} N(j)}. \tag{29}$$

The initial value for p_1 is given by the final value obtained in the previous stage, whereas the initial value for p_2 is 0. The center of distortion is initialized at the geometric center of the image. To minimize energy (29), we propose an iterative scheme which is a combination of the gradient descent and Newton-Raphson methods. The gradient descent method is based on the following iterative scheme

$$\mathbf{u}_{n+1} = \mathbf{u}_n - \lambda \nabla E(\mathbf{u}_n), \tag{30}$$

where $\lambda > 0$. It is well-known that if $\nabla E(\mathbf{u}_n) \neq 0$ and λ is small enough, then $E(\mathbf{u}_n) < E(\mathbf{u}_{n+1})$. Newton-Raphson method is based on the Taylor expansion of $E(\mathbf{u})$ given by

$$E(\mathbf{u}) = E(\mathbf{u}_n) + (\nabla E(\mathbf{u}_n))^T (\mathbf{u} - \mathbf{u}_n) + (\mathbf{u} - \mathbf{u}_n)^T \nabla^2 E(\mathbf{u}_n)(\mathbf{u} - \mathbf{u}_n) + \dots \tag{31}$$

Since we want to minimize this energy, we derive the previous expression and make it equal to 0

$$\nabla E(\mathbf{u}) \approx \nabla E(\mathbf{u}_n) + \nabla^2 E(\mathbf{u}_n)(\mathbf{u} - \mathbf{u}_n) = 0. \tag{32}$$

Therefore,

$$\nabla^2 E(\mathbf{u}_n)(\mathbf{u} - \mathbf{u}_n) = -\nabla E(\mathbf{u}_n), \tag{33}$$

and the Newton-Raphson minimization scheme is given by

$$\mathbf{u}_{n+1} = \mathbf{u}_n - \nabla^2 E(\mathbf{u}_n)^{-1} \nabla E(\mathbf{u}_n). \tag{34}$$

Usually, Newton-Raphson is faster than the gradient descent method when the initial solution is not far from a local minimum of the energy, but in case the initial solution is far from a local minimum energy, the method can diverge and, moreover, there is no way to ensure that the energy decreases across the iterations. To avoid this limitation of Newton-Raphson scheme, in 1944, Levenberg [24] introduced the idea of using a damping parameter to combine gradient descent and Newton-Raphson schemes in the context of the well-known Levenberg-Marquardt optimization algorithm. Using Levenberg's approach we propose to use the following minimization scheme:

$$\mathbf{u}_{n+1} = \mathbf{u_n} - (\nabla^2 E(\mathbf{u}_n) + \gamma Id)^{-1} \nabla E(\mathbf{u}_n), \tag{35}$$

where γ is the damping parameter used to control the convergence of the minimization as follows: γ is updated in each iteration to ensure that $E(\mathbf{u}_{n+1}) < E(\mathbf{u}_n)$. Usually, its value is higher when we are far from the solution and decreases when we approach it. Hence, the damping parameter helps to avoid Newton-Raphson instabilities when the initial solution is far from a local minimum of the energy.

The damping parameter γ is initialized to 10, and its evolution depends on the value obtained for the optimized energy in each iteration. In order to ensure that the new computed energy is less than the previous one, the value is multiplied/divided by 10, according to the increase/decrease of the energy. As the optimization process consists in optimizing the tuple constituted by (p_1, p_2, x_c, y_c), the stop condition is that the difference between all the optimized parameters and their previous values is less than a tolerance. In order to avoid a large number of iterations, we include an extra condition, by establishing a limit of 100 iterations. In the worst case, when no iteration minimizes the energy, the model will be equal to the input one, and the center of distortion will be the geometric center of the image.

We remark that we look for local minima of the energy $E(\mathbf{u})$ because, in general, the global minima of this energy can be attained in singular configurations (especially when very few primitive points are used). To avoid such singular solutions, we check whether the lens distortion transformation is one-to-one using the results of theorems 1 and 2 and, otherwise, we reject the proposed solution. We point out that, since we deal with an iterative scheme, we cannot ensure that the method always converges to the best solution. However, as shown in the experiments, the method behaves quite well in most situations.

3.4 Iterative optimization of the two-parameter distortion model and the collection of lines

In the previous section, we explained how the parameters for the polynomial and division models can be optimized to minimize the distance from the edge points to the lines. During this process, we considered the same points which were originally detected as line points. Only the parameters varied in order to obtain a better fitting between the points and the line equations.

However, once the two-parameter model and the center of distortion have been optimized using the current collection of lines and their associated points, the new model can be used to improve the lines using the Hough voting strategy. The underlying idea is that the larger the collection of points associated to the primitive lines, the better the model estimation. Indeed, by integrating the optimized distortion model into the Hough process, the edge points vote for the lines after having been corrected with the optimized model, in such a way that those points which were not initially associated to the lines because the matching was not satisfactory, can now be associated. As a result of the voting process, we obtain an improved collection of lines with a greater number of associated points. Then, since the collection of lines has been updated, we can optimize the two-parameter model again using this new set of primitive lines. This optimization is carried out by minimizing the distance from the corrected points to the new lines computed in each iteration. This iterative procedure stops if the global amount of points associated to lines does not increase when we integrate the optimized model into the Hough process.

We use some images of Figure 5 to illustrate this optimization procedure. Figures 6(a), 7 and 8 show the lines detected on a calibration pattern in images with different degrees of distortion. We have tested the polynomial and the division models and, in both cases, we have first used a single distortion parameter to search within the Hough space (Figures 6(a) and 7(a) correspond to the polynomial model, whereas Figure 8(a) correspond to the division model). Afterward, we have introduced the second distortion parameter in the process and we have refined the results by iteratively optimizing the energy $E(\mathbf{u})$ and recalculating the set of points which are considered as line points (see Figure 7(b) for the polynomial model with iterative optimization, and Figure 8(b) for the division model with iterative optimization).

As observed, especially when the distortion is significant, the introduction of the iterative scheme to recalculate the line points from the optimized parameters allows detecting new lines and significantly increases the number of points of the detected lines. For instance, the four squares at the corners, which were not initially detected using the polynomial model, are now detected with the same model (see Figure 7(b)). In the case of the division model, the lines are longer, so that more reliable information is available to correct the distortion (see Figure 8(b)).

Some other results are shown in Figs. 9, 10, 11 and 12. As observed in Figs. 9 and 10, the division model is able to extract longer lines, which are split when the polynomial model is applied (see, for instance, the top horizontal lines). Furthermore, the iterative optimization allows extracting even very close parallel lines. In Figs. 11 and 12, the introduction of the iterative optimization detects more line points and the use of the division model provides longer lines, which are not split into different segments (see, for instance, the stairs at the bottom).

We point out that we stop the iterations when the total amount of points associated to the collection of distorted lines does not increase through the iterations. In practice, only a few iterations are required to attain convergence.

3.5 Image distortion correction

First, we point out that the final lens distortion model obtained by the proposed method satisfies the conditions of theorems 1 or 2 because we check such conditions in the different steps of the algorithm and any lens distortion model which does not satisfy such conditions is rejected. Therefore, the obtained lens distortion model is a one-to-one transformation and the inverse transformation can be computed (as explained in theorems 1 and 2). Using such inverse transformation, we obtain the distortion-free image by computing the inverse distortion in the image points to get the RGB color channels from the corresponding point in the original image.

4 Results

A detailed description of the algorithm is presented in the IPOL companion paper [25], including an online demo[1]. In this paper, we will focus on the presentation and discussion of the experimental results. We have performed experiments in a variety of real images presenting different amount of distortion (Figure 5). Images in figure 6 have been taken with a NIKON D90 camera with a NIKKOR DX 17-200 ZOOM lens presenting a small amount of distortion. For the rest of the experiments, the images have been taken with a NIKON D90 camera with a TOKINA DX 10-17 ZOOM lens presenting a significant amount of distortion. Images have been taken using different settings of the lens focal distance (by changing the lens ZOOM), which provides a variety of degrees of distortion.

---

[1]http://dev.ipol.im/~asalgado/ipol_demo/workshop_ldm_pq2p_io/

Figures 13–17 show the corrected images using the one-parameter polynomial and division models (first row) as well as the iterative optimization technique with two-parameter models (second row). Figures 13 and 14 illustrate the results of correcting the distortion with the different models on the calibration pattern shown above. Although the results seem similar for 14(c) and 14(d), if we analyze the lines detected in 7(b) and 8(b), we can see that the information provided by the division model is more complete. This is mainly noticeable in the left and top lines belonging to the pattern frame. For the top line, although the two parts of the line are in the opposite sides of the image, it is detected as a single line. Similarly, the process has been applied to the images described in the previous section. The results are shown in Figures 15 and 16.

In Figure 15, we can observe that the introduction of the iterative optimization in the division model (15(d)) allows correcting the bottom line on the floor. The polynomial model cannot correct this distortion, even if the iterative optimization is considered. This is due to the detected primitives. Therefore, if we look at figure 10(b), the provided lines are significantly longer than in the other cases, especially at the top and bottom of the building.

In Figure 16, the benefits of introducing the iterative optimization are especially noticeable if we look at the stairs. The one and two-parameter models without optimization are not capable to cope with this strong distortion and the images are still distorted after correction. In this case, the division model provides much better results when the iterative optimization is applied (16(d)). Considering the lines depicted on figure 12(b), we can observe that the lines provided by the two-parameter division model with the iterative optimization are longer than with the other configurations. This fact is especially remarkable on the lines of the stairs at the bottom of the image, as well as those that belong to buildings at both sides of the chapel.
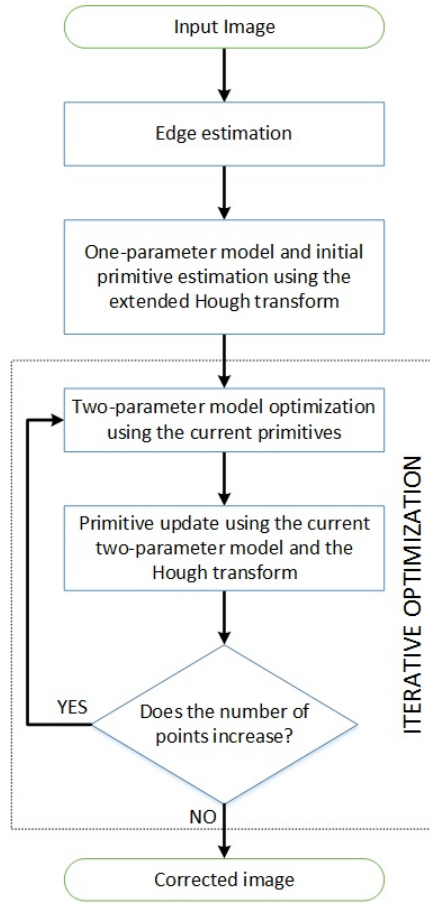
Figure 3: Flowchart of the distortion correction process. In this case, primitives are given by the collection of detected distorted lines and their corresponding associated points. The algorithm aims at maximizing the total number of points associated to the collection of detected lines.
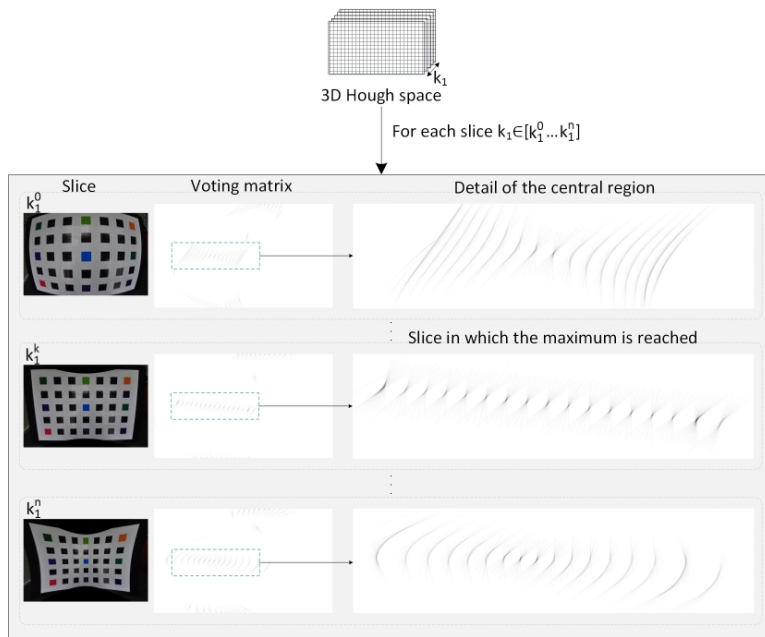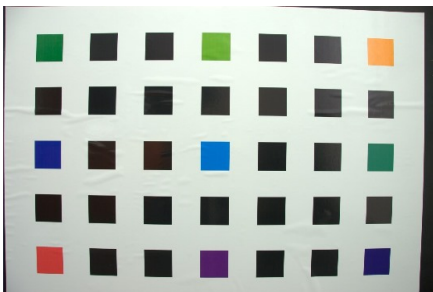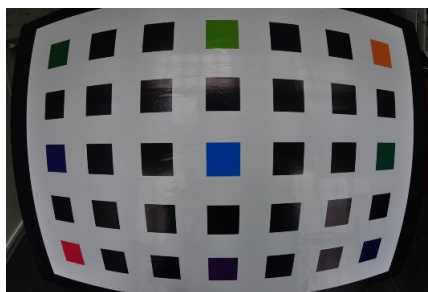
3D Hough space

For each slice $k_1 \in [k_1^0 \dots k_1^n]$

Slice   Voting matrix   Detail of the central region

k_1^0

Slice in which the maximum is reached

k_1^k

k_1^n

Figure 4: Detail of the three-dimensional Hough space used for the voting process.

(a)

(b)

(c)

(d)

(e)

Figure 5: Original images used in the experiments.

(a)

(b)
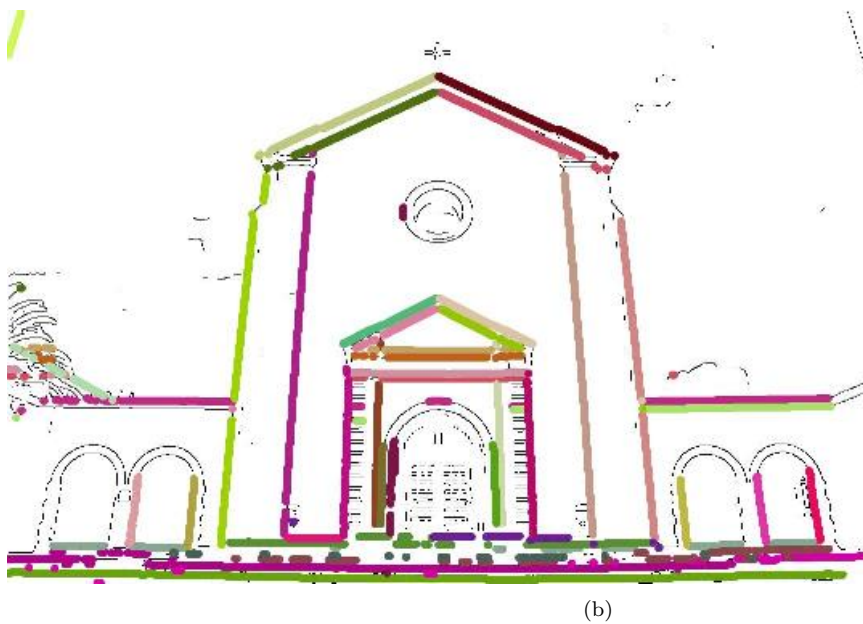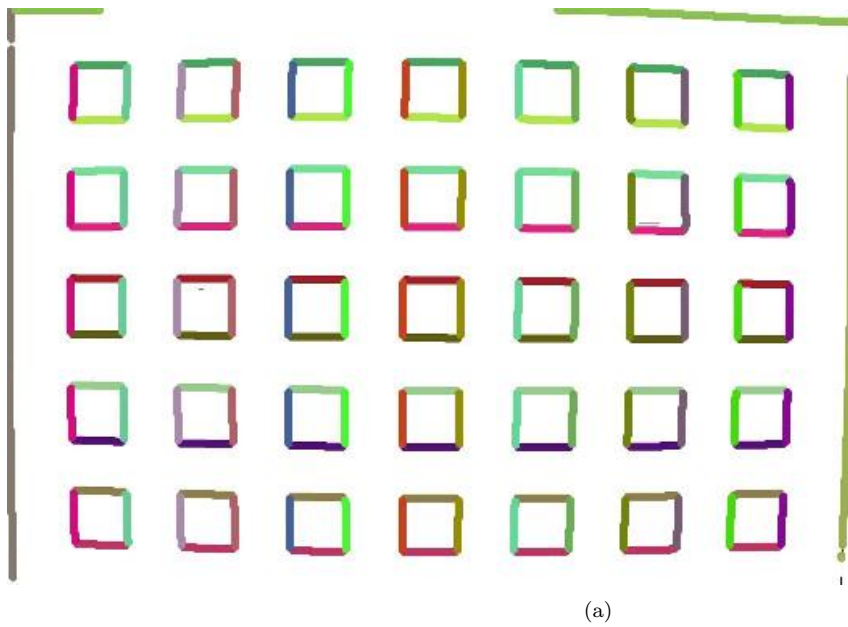
Figure 6: Initial collection of lines detected with the improved Hough transform using the polynomial model for (a) a calibration pattern and (b) a building.
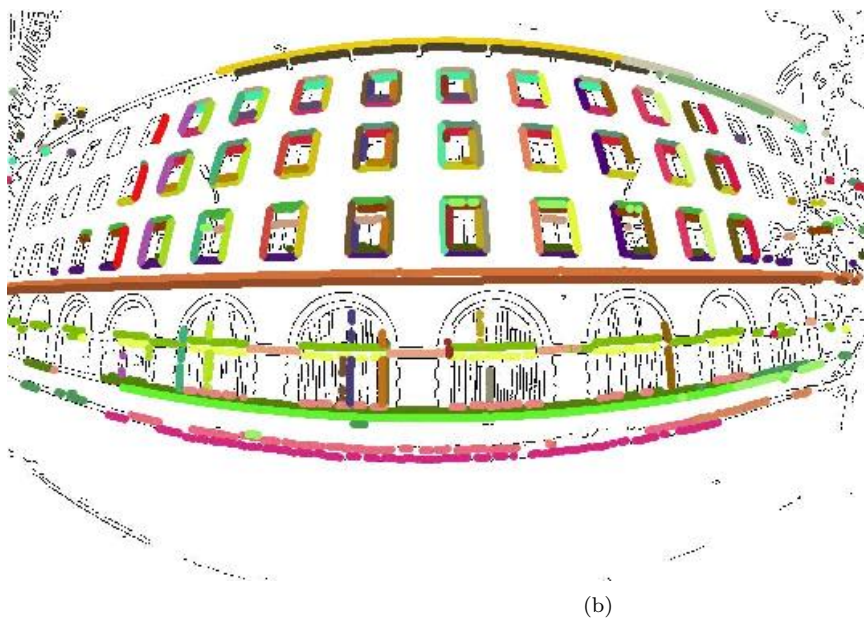
(a)

(b)
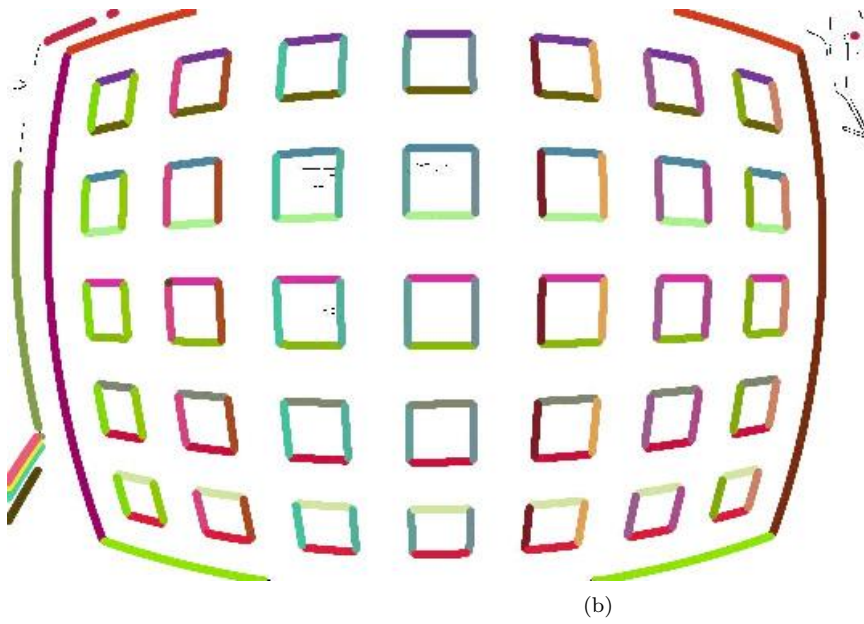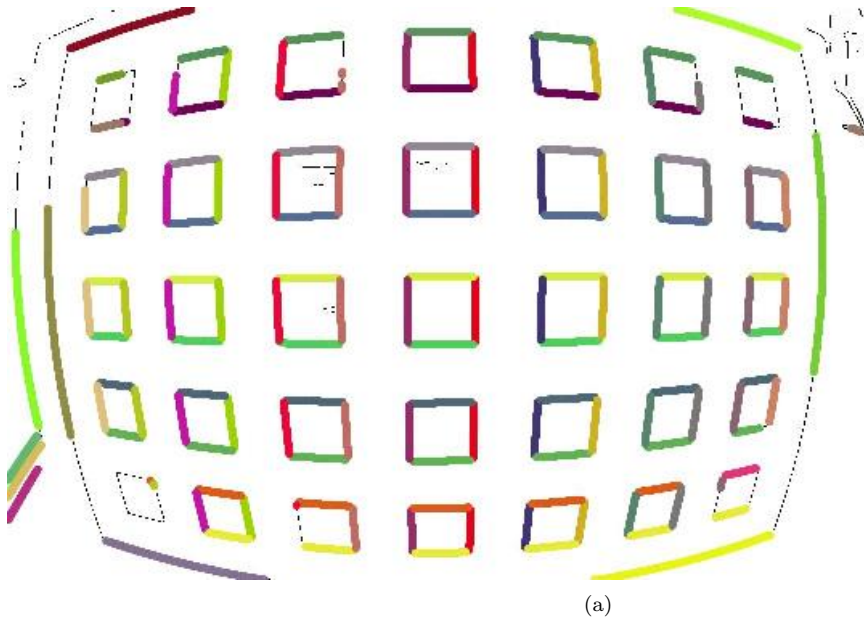
Figure 7: Collection of lines detected with the Hough transform using (a) the polynomial model and (b) the polynomial model with iterative optimization.

(a)

(b)
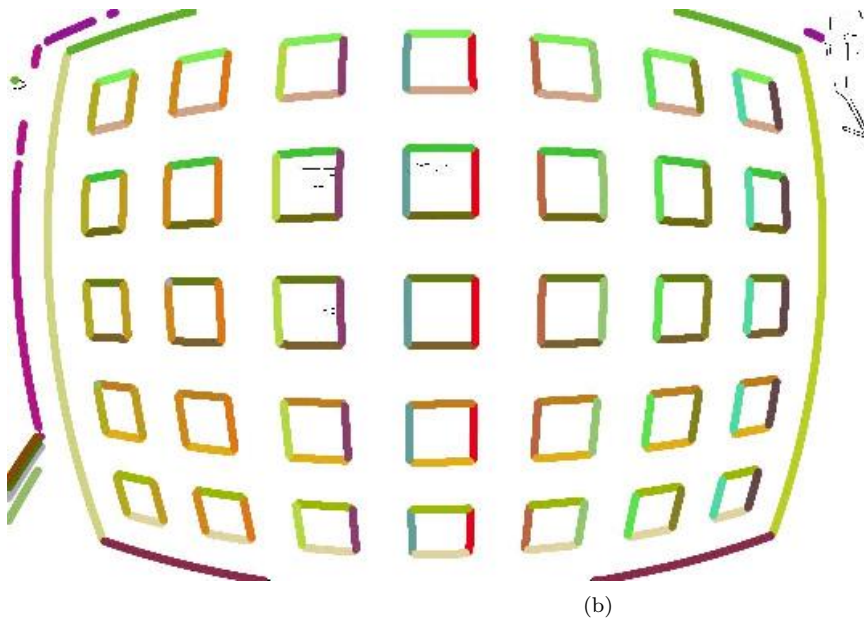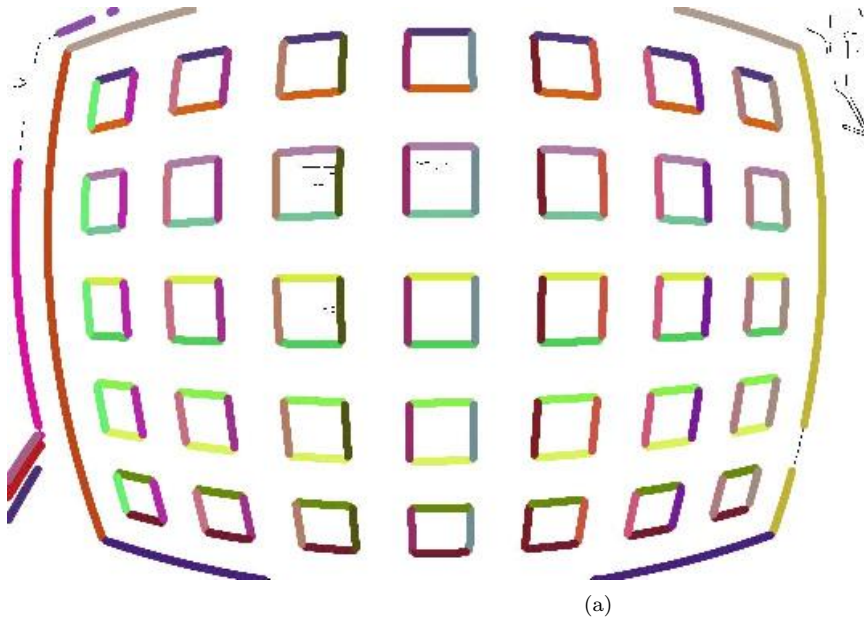
Figure 8: Collection of lines detected with the Hough transform using (a) the division model and (b) the division model with iterative optimization.
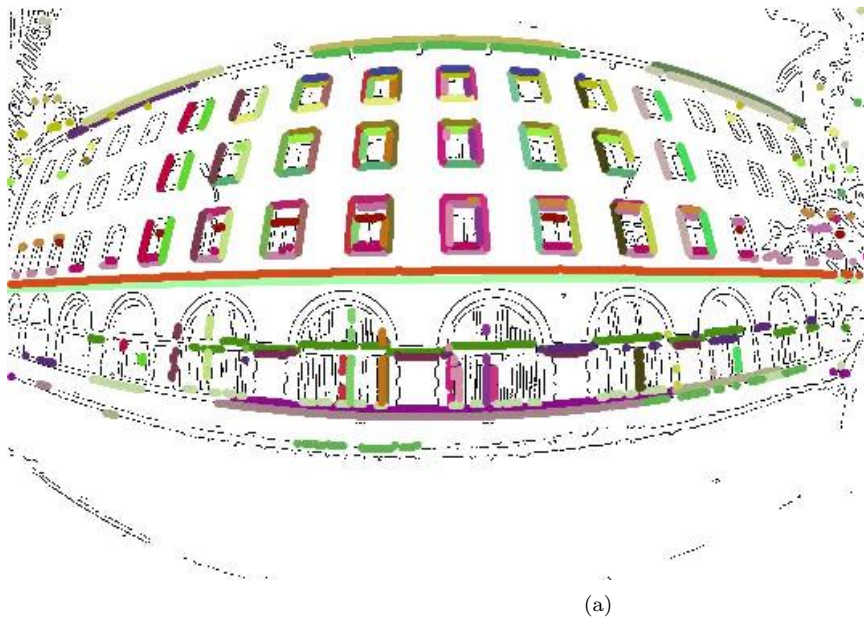
(a)

(b)

Figure 9: Collection of lines detected with the Hough transform using (a) the polynomial model and (b) the polynomial model with iterative optimization.
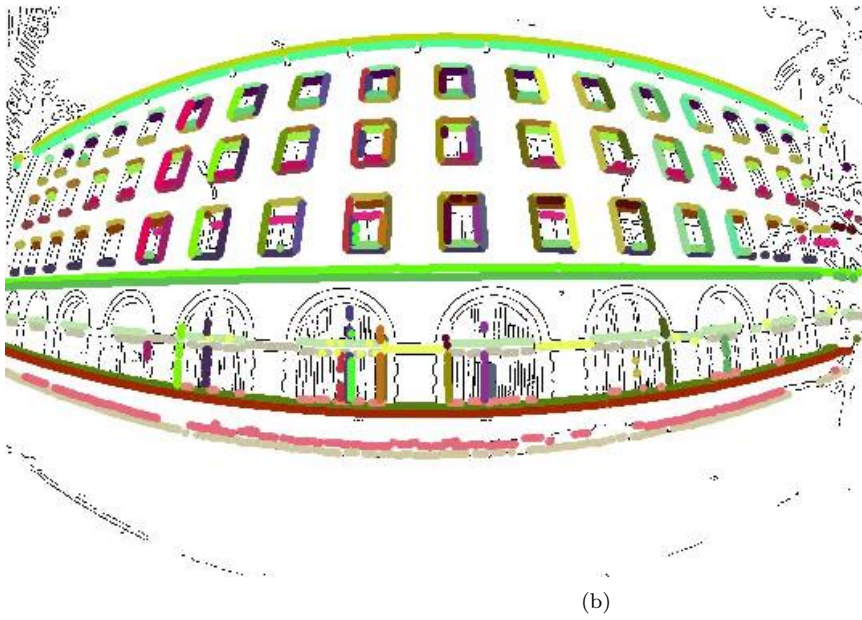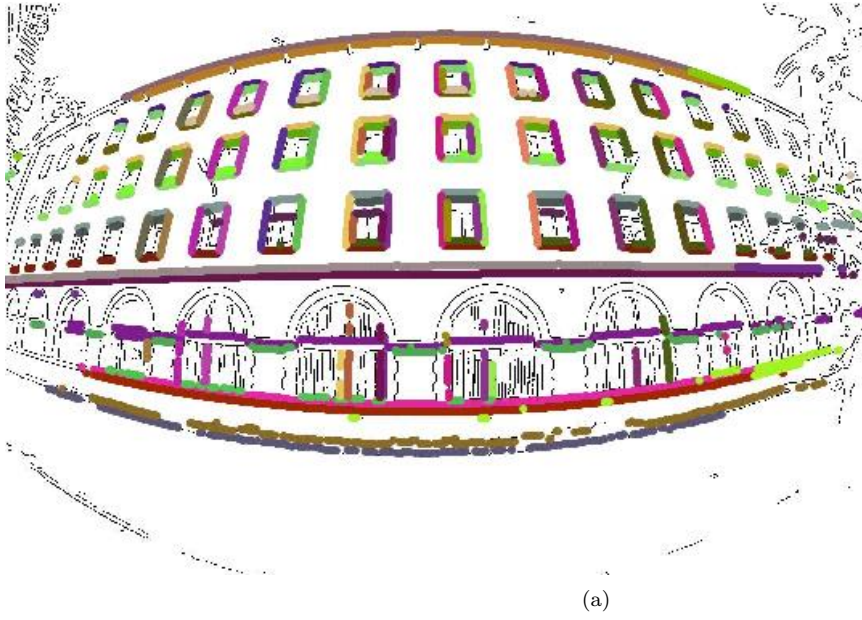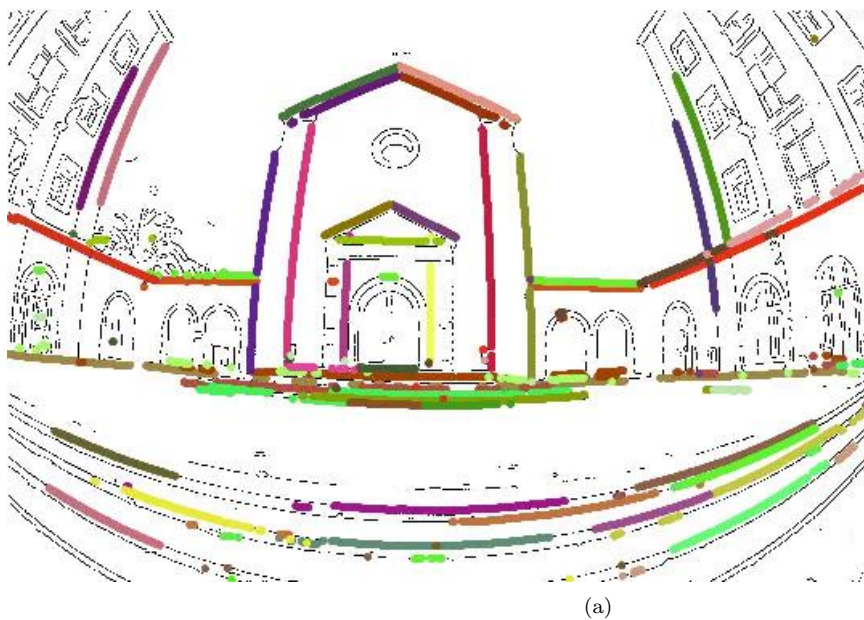
(a)

(b)

Figure 10: Collection of lines detected with the Hough transform using (a) the division model and (b) the division model with iterative optimization.

(a)

(b)

Figure 11: Collection of lines detected with the Hough transform using (a) the polynomial model and (b) the polynomial model with iterative optimization.
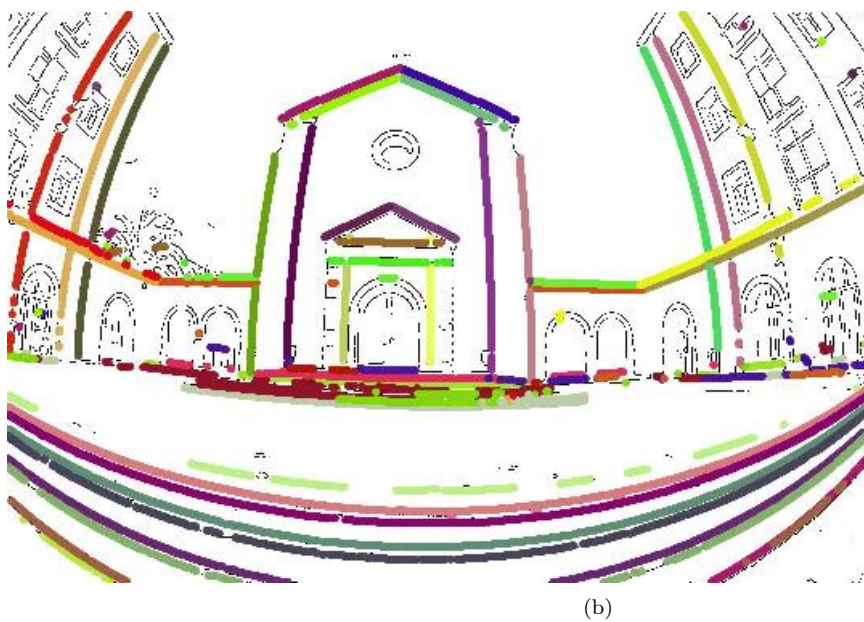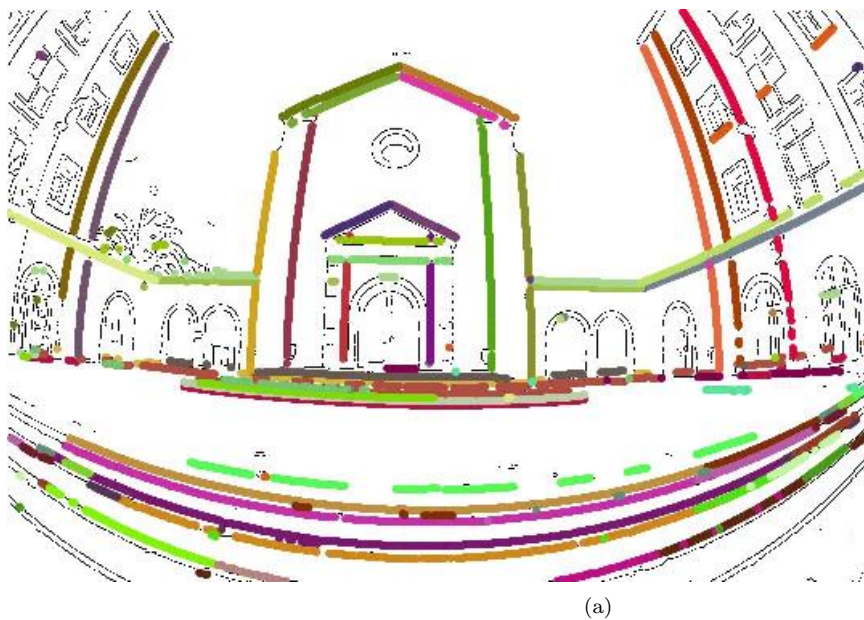
Figure 12: Collection of lines detected with the Hough transform using (a) the division model and (b) the division model with iterative optimization.

Model	N. Lines	N. Points	p_1	p_2	center	E
Pol1p	27	10504	6.33%		(536,357.5)	0.494157
Pol2pIO	27	10632	6.96%	2.79%	(544.6,337.5)	0.101998
Div1p	30	10455	7.00%		(536,357.5)	0.916431
Div2pIO	27	10634	6.86%	2.72%	(544,337.6)	0.098361

Table 1: Number of lines and points, distortion parameters, distortion center and energy for the calibration pattern in Figs. 6 and 13 using the one-parameter polynomial and division models (Pol1p and Div1p) and the two-parameter models with iterative optimization (Pol2pIO and Div2pIO).
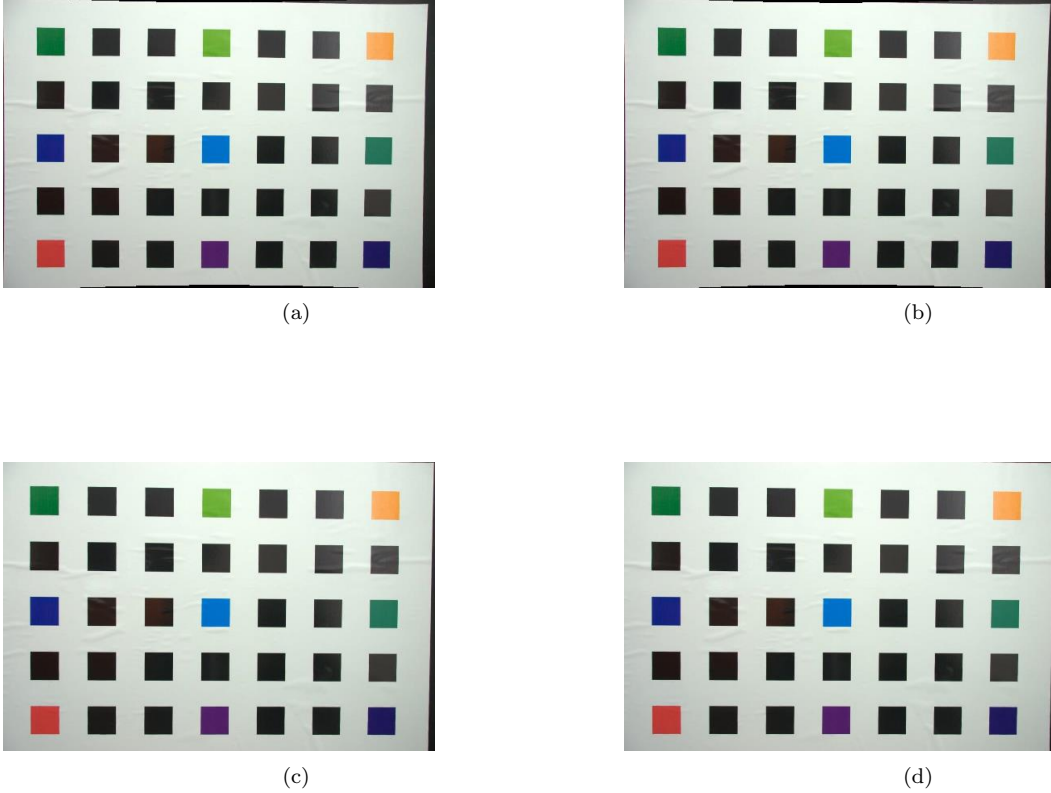

(a)

(b)

(c)

(d)

Figure 13: Corrected images using the following approaches: (a) one-parameter polynomial model, (b) one-parameter division model, (c) two-parameter polynomial model with iterative optimization and (d) two-parameter division model with iterative optimization.

Tables 1–5 show the number of lines and points, the parameters of the models, the distortion center, and the final energy using the different approaches. The rows "Pol1p" and "Div1p", represent the results obtained using [2] and [1]. On the other hand, the rows "Pol2p" and "Div2p", show the results using the proposed method. As observed, when the iterative optimization is introduced, the number of detected points increases significantly in both cases, the polynomial and the division models. It is also remarkable that the introduction of the proposed technique allows minimizing the energy more than in the other configurations. As a consequence, this reduction of the mean error means that the corrected points are closer to the straight lines than in the cases in which we use a one-parameter model.

Figure 18 illustrates the variation of the number of points through the iterative optimization process using the division model. The first value corresponds to the number of points obtained by means of the improved Hough transform. As observed, from the first iteration of the optimization, we obtain a significantly greater number of points. This is due to the minimization of the energy function, that provides an optimized two-parameter model. This model allows obtaining points that were not considered when we used a one-parameter model.

In figure 19, we show the ability of the proposed method using a wide variety of images. We observe that the method is able to deal with a strong distortion in different scenarios, obtaining satisfactory results.

Model	N. Lines	N. Points	p_1	p_2	center	E
Pol1p	41	9822	4.09%		(536,356.5)	2.48706
Pol2pIO	44	10293	3.12%	0.82%	(536,356.5)	2.03084
Div1p	42	9619	-0.73%		(536,356.5)	3.97142
Div2pIO	42	10344	3.17%	1.17%	(536,356.5)	2.42632

Table 2: Number of lines and points, distortion parameters, distortion center and energy for the image in Figs. 6 and 17 using the one-parameter polynomial and division models (Pol1p and Div1p) and the two-parameter models with iterative optimization (Pol2pIO and Div2pIO).

Model	N. Lines	N. Points	p_1	p_2	center	E
Pol1p	38	9536	65.01%		(536,356)	1.4673
Pol2pIO	34	11051	95.45%	11.38%	(525,361.9)	0.532611
Div1p	34	10846	98.72%		(536,356)	1.53463
Div2pIO	34	11137	118.04%	13.22%	(525.1,362.3)	0.261851

Table 3: Number of lines and points, distortion parameters, distortion center and energy for the calibration pattern in Figs. 7, 8 and 14 using the one-parameter polynomial and division models (Pol1p and Div1p) and the two-parameter models with iterative optimization (Pol2pIO and Div2pIO).

Model	N. Lines	N. Points	p_1	p_2	center	E
Pol1p	48	10405	73.24%		(536,356)	2.63275
Pol2pIO	54	12577	121.02%	24.41%	(528.1,361.9)	1.64061
Div1p	47	12990	262.80%		(536,356)	2.99372
Div2pIO	48	14632	555.61%	21.41%	(533,362.02)	1.00159

Table 4: Number of lines and points, distortion parameters, distortion center and energy for the image in Figs. 9, 10 and 15 using the one-parameter polynomial and division models (Pol1p and Div1p) and the two-parameter models with iterative optimization (Pol2pIO and Div2pIO).

Model	N. Lines	N. Points	p_1	p_2	center	E
Pol1p	47	9550	52.62%		(536,356)	2.53074
Pol2pIO	47	10212	64.28%	10.98%	(571.6,363.2)	2.67954
Div1p	53	14209	321.66%		(536,356)	3.98304
Div2pIO	45	15874	1732.70%	22.25%	(510.2,353.6)	2.58055

Table 5: Number of lines and points, distortion parameters, distortion center and energy for the image in Figs. 11, 12 and 16 using the one-parameter polynomial and division models (Pol1p and Div1p) and the two-parameter models with iterative optimization (Pol2pIO and Div2pIO).
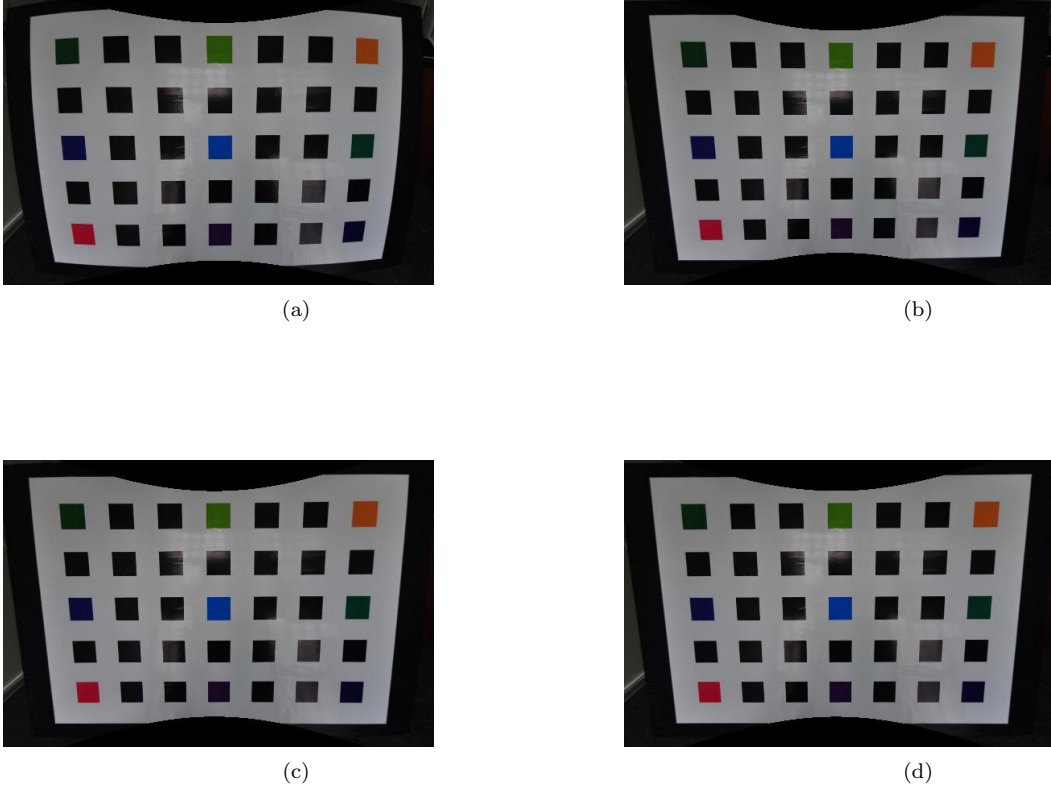
Figure 14: Corrected images using the following approaches: (a) one-parameter polynomial model, (b) one-parameter division model, (c) two-parameter polynomial model with iterative optimization and (d) two-parameter division model with iterative optimization.
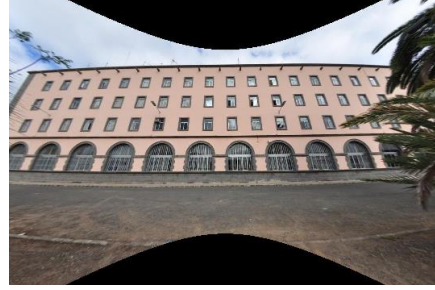
5   Conclusion

In this paper we study in detail two-parameter radial models of camera lenses using polynomial and division models and we propose an automatic method to estimate two-parameter radial distortion models. First we show new mathematical results which characterize the conditions on the lens distortion parameters in order to ensure that the lens distortion transformation is one-to-one and, therefore, the inverse transformation is well-defined and we can properly compute the distortion-free image. Secondly, we present a new automatic method to correct radial distortion based on an iterative optimization which aims at improving the collection of detected lines used to estimate the lens distortion model. The method has been extensively validated on real images with different degrees of distortion and the results are very promising. According to the experiments, we can conclude that the proposed method is very robust and provides good results in a large variety of images. In the case of strong distortion, a significant improvement in the results is obtained when we use two distortion parameters (instead of just one) and, moreover, the division model performs better than the polynomial model. The main limitations of the proposed method are, first, that the image must contain the projection of visible 3D straight lines, and second, the distortion model must be invertible in the whole image. Therefore, it would not work on non-invertible lens distortion model scenarios such as fish-eye images, where the image is included in a circle surrounded by a black background. In the future, we intend to apply the proposed method to camera calibration, specially in scenarios showing a significant lens distortion. We have built a lens distortion image database freely available at `http://www.ctim.es/WideLensImageDatabase/`.

References

[1] M. ALEMÁN-FLORES, L. ALVAREZ, L. GOMEZ, AND D. SANTANA-CEDRÉS, *Wide-angle lens distortion correction using division models*, LNCS: Progress in Pattern Recognition, Image Analysis, Computer Vision, and

Figure 15: Corrected images using the following approaches: (a) one-parameter polynomial model, (b) one-parameter division model, (c) two-parameter polynomial model with iterative optimization and (d) two-parameter division model with iterative optimization.

Applications Lecture Notes in Computer Science, 8258 (2013), pp. 415–422.

[2] M. Alemán-Flores, L. Alvarez, L. Gomez, and D. Santana-Cedrés, *Line detection in images showing significant lens distortion and application to distortion correction*, Pattern Recognition Letters, 36 (2014), pp. 261–271.

[3] L. Alvarez, L. Gomez, and R. Sendra, *An algebraic approach to lens distortion by line rectification*, Journal of Mathematical Imaging and Vision, 39 (2008), pp. 36–50.

[4] L. Alvarez, L. Gomez, and R. Sendra, *Algebraic lens distortion model estimation*, Image Processing On Line, 1 (2010).

[5] L. Alvarez, L. Gomez, and R. Sendra, *Accurate depth dependent lens distortion models: An application to planar view scenarios*, Journal of Mathematical Imaging and Vision, 39 (2011), pp. 75–85.

[6] D.C. Brown, *Close-range camera calibration*, Photogrammetric Engineering, 37 (1971), pp. 855–866.

[7] F. Bukhari and M. Dailey, *Automatic radial distortion estimation from a single image*, Journal of Mathematical Imaging and Vision, 45 (2012), pp. 31–45.

[8] R. Cucchiara, C. Grana, A. Prati, and R. Vezzani, *A Hough transform-based method for radial lens distortion correction*, In: Proc. of the 12th International Conference on Image Analysis and Processing (ICIAP03), (2003), pp. 182–187.

[9] A. Desolneux, L. Moisan, and J.-M. Morel, *Meaningful alignments*, Int. J. Comput. Vis., 40 (2000), pp. 7–23.

[10] F. Devernay and O. Faugeras, *Straight lines have to be straight*, Machine Vision and Applications, 13 (2001), pp. 14–24.

Figure 16: Corrected images using the following approaches: (a) one-parameter polynomial model, (b) one-parameter division model, (c) two-parameter polynomial model with iterative optimization and (d) two-parameter division model with iterative optimization.

[11] O. FAUGERAS, *Three-dimensional computer vision*, MIT Press, 1993.

[12] A. W. FITZGIBBON, *Simultaneous linear estimation of multiple view geometry and lens distortion*, In: Proc. IEEE International Conference on Computer Vision and Pattern Recognition, (2001), pp. 125–132.

[13] M. GADEMAYR, LIEDLGRUBER M., AND A. VCSEI, *Evaluation of different distortion correction methods and interpolation techniques for an automated classification of celiac disease*, Computer Methods and Programs in Biomedicine, 112 (2013), pp. 694–712.

[14] R. GROMPONE, J. JAKUBOWICZ, J.M. MOREL, AND G. RANDALL, *LSD: a line segment detector*, IPOL : Image Processing On Line, 2 (2012), pp. 35–55.

[15] R. GROMPONE, J. JAKUBOWICZ, J. M. MOREL, AND G. RANDALL, *On straight line segment detector*, Journal of Mathematical Imaging and Vision, 32 (2008), pp. 313–347.

[16] R. GROMPONE, P. MONASSE, J. M. MOREL, AND Z. TANG, *Towards high-precision lens distortion correction*, In: Proc. of the 17th IEEE International Conference on Image Processing (ICP), (2010), pp. 4237–4240.

[17] R. GROMPONE, P. MONASSE, J. M. MOREL, AND Z. TANG, *Lens distortion correction with a calibration harp*, In: Proc. of the 18th IEEE International Conference on Image Processing (ICP), (2011), pp. 617–620.

[18] R. I. HARTLEY AND A. ZISSERMAN, *Multiple view geometry in computer vision*, Cambridge University Press, 2004.

[19] CIARAN HUGHES, M. GLAVIN, E. JONES, AND P. DENNY, *Review of geometric distortion compensation in fish-eye cameras*, in Signals and Systems Conference, 208. (ISSC 2008). IET Irish, Galway, Ireland, 2008, pp. 162–167.

Figure 17: Corrected images using the following approaches: (a) one-parameter polynomial model, (b) one-parameter division model, (c) two-parameter polynomial model with iterative optimization and (d) two-parameter division model with iterative optimization.

[20] S. Kang, *Radial distortion snakes*, In: Proc. IEICE Transactions on Information and Systems, (2000), pp. 1603–1611.

[21] J. M. Lavest, M. Viala, and M. Dhome, *Do we really need an accurate calibration pattern to achieve a reliable camera calibration?*, In: Proc. of the 5th European Conference on Computer Vision (ECCV'98), I (1998), pp. 158–174.

[22] T. Lee, T. Chang, C. Wei, S. Lai, K. Liu, and H. Wu, *Automatic distortion correction of endoscopic images captured with wide-angle zoom lens*, IEEE Transactions on Biomedical Engineering, (2013), pp. 2603–2613.

[23] Reimar Lenz, *Linsenfehlerkorrigierte Eichung von Halbleiterkameras mit Standardobjektiven für hochgenaue 3D - Messungen in Echtzeit*, in Mustererkennung 1987, Erwin Paulus, ed., vol. 149 of Informatik-Fachberichte, Springer Berlin Heidelberg, 1987, pp. 212–216.

[24] K. Levenverg, *A method for the solution of certain non-linear problems in least-squares*, Quarterly of Applied Mathematics, 2 (1944), pp. 164–168.

[25] Daniel Santana-Cedrés, Luis Gomez, Miguel Alemán-Flores, Agustín Salgado, Julio Esclarín, Luis Mazorra, and Luis Alvarez, *An iterative optimization algorithm for lens distortion correction using two-parameter models (companion paper)*, IPOL Image Processing On Line.

[26] Daniel Santana-Cedrés, Luis Gomez, Miguel Alemán-Flores, Agustín Salgado, Julio Esclarín, Luis Mazorra, and Luis Alvarez, *Invertibility and estimation of two-parameter polynomial and division lens distortion models*, SIAM Journal on Imaging Sciences, 8 (2015), pp. 1574–1606.

[27] Zhongwei Tang, Rafael Grompone von Gioi, Pascal Monasse, and Jean-Michel Morel, *High-precision camera distortion measurements with a calibration harp*, JOSA A, 29 (2012), pp. 2134–2143.
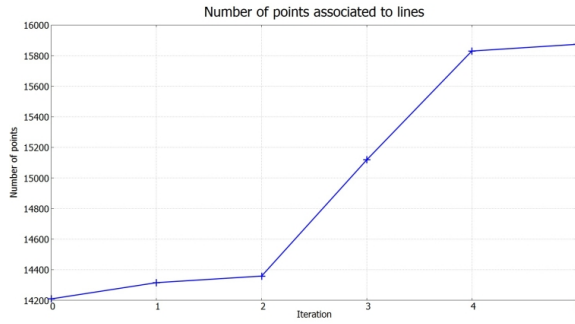
Figure 18: Variation of the number of points for the image shown in Fig. 16(d) using the iterative scheme.

[28] A. Wang, T. Qiu, and L. Shao, *A simple method to radial distortion correction with centre of distortion estimation*, Journal of Mathematical Imaging and Vision, 35 (2009), pp. 165–172.

[29] T. Zhongwei, R. Grompone, P. Monasse, and J. M. Morel, *Self-consistency and universality of camera distortion models*, In hal-00739516, version 1, (2012), pp. 1–8.
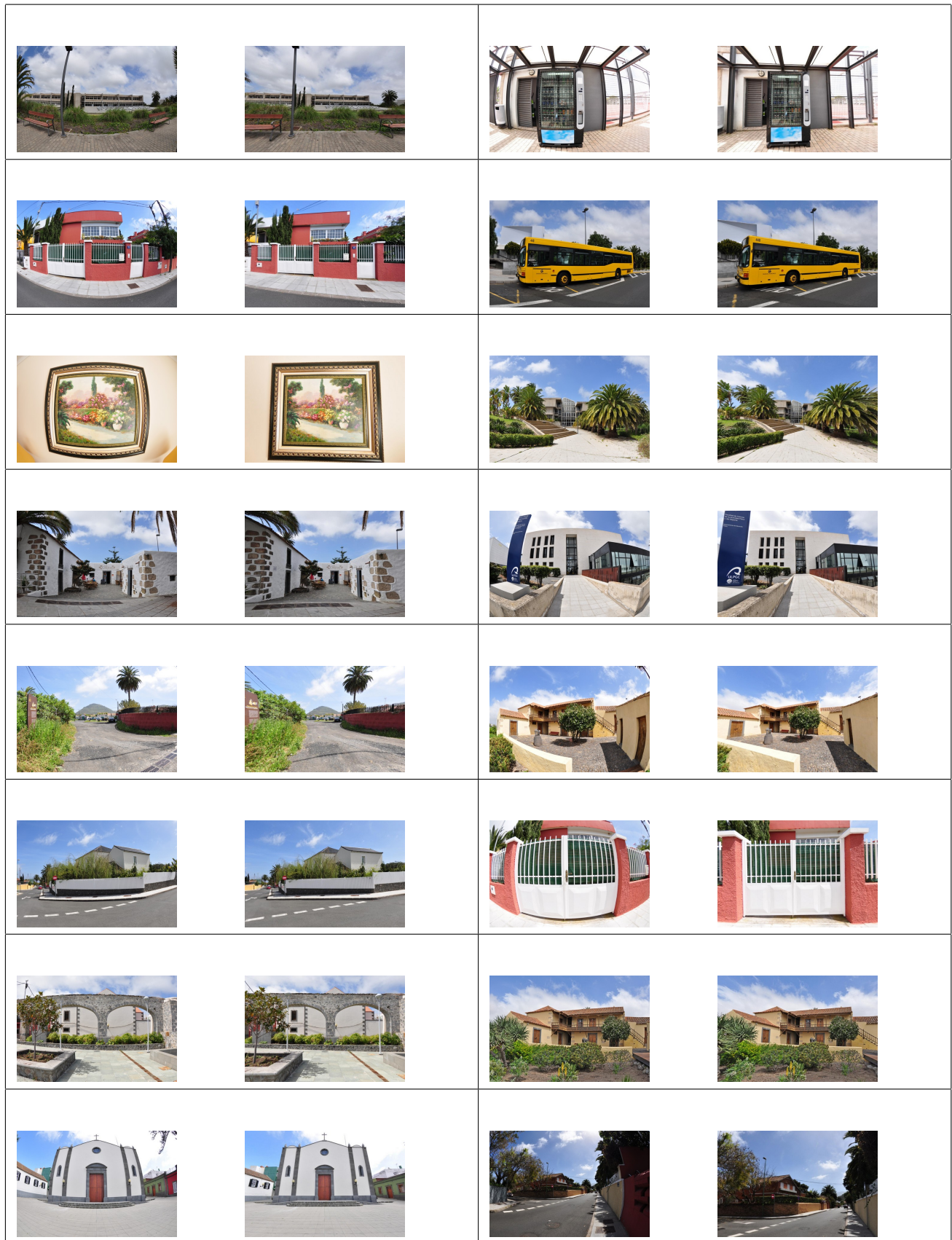
Figure 19: Some examples of image distortion corrections using the proposed method. Additional results and access to the original images are provided in http://www.ctim.es/WideLensImageDatabase/