

Non-iterative Camera Calibration Procedure Using A Virtual Camera

Tobias Elbrandt, Ralf Dragon, Jörn Ostermann

Institut für Informationsverarbeitung
Leibniz Universität Hannover, Appelstr. 9A, 30167 Hannover, Germany
{elbrandt/dragon/ostermann}@tnt.uni-hannover.de

Abstract

This article presents a method to improve camera calibration by separating determination of the non-linear calibration parameters from that of the linear ones. We measure correspondences between the distorted image on a camera target and a flexible undistorted calibration pattern displayed on a TFT monitor. Using these correspondences the image of the camera may be projected onto the plane of the calibration pattern to remove the distortion. We prove that this projection follows the principles of a pinhole camera and call it virtual camera. Using the undistorted images of the virtual camera for traditional camera calibration methods, linear camera parameters can be calculated with a 7.5% to 39.5% higher precision compared to Zhang/Heikkilä. The described procedure allows a camera calibration process in a non-iterative way.

1 Introduction

Every time a camera is used for geometric measurement, which is determination of the position or size of an object or its distance to another object, it is necessary to know the properties of the optical system of the camera.

Determining these properties is a classic research topic in computer vision [5][10][12] so the reader is referred to established literature (e.g. [4], chapter 6) for description of the parameters and how to work with them. The main parameters needed for measurement are the linear extrinsic and intrinsic camera parameters. Extrinsic parameters describe the position \bar{C}_c (translation to the origin of the coordinate system¹) and the orientation \bar{R}_c (rotation) of the camera. Linear intrinsic camera parameters

describe the linear properties of the optical system within the camera – i.e. the focal length f , aspect ratio of the pixels s_x/s_y , shear r , and displacement of the principal point (h_x, h_y) . The linear parameters may be conveniently combined into the so called camera matrix M_c :

$$M_c = K_c R_c \quad (1)$$

with

$$R_c = \begin{bmatrix} \bar{R}_c & \mathbf{0} \\ \mathbf{0} & 1 \end{bmatrix} \begin{bmatrix} \mathbf{1} & -\bar{C}_c \\ \mathbf{0} & 1 \end{bmatrix} = \begin{bmatrix} \bar{R}_c & -\bar{R}_c \bar{C}_c \\ \mathbf{0} & 1 \end{bmatrix} \quad (2)$$

$$K_c = \begin{bmatrix} f s_x & r & h_x & 0 \\ 0 & f s_y & h_y & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \quad (3)$$

Non-linear parameters describing radial, tangential, or prism distortion are also considered as intrinsic parameters (see [13] for their description).

Most calibration methods for single cameras (Tsai [10], direct linear transformation [4], Zhang [12], Heikkilä [5]) derive the parameters of the camera model on the basis of corresponding points in the world coordinate system and on the camera target. For this procedure, they need one or more camera images of a planar calibration pattern or calibration object with marks on the surface with known size or distance to each other. For each visible mark the position on the camera target is determined giving a list of correspondences between points P_i in world coordinates and points p_i in camera target coordinates, i.e. pixel. By using Equation (4) of the mapping of a pinhole camera, these correspondences are also modeled by algebraic means. The projective linear factor λ is calculated such that the third component of p equals 1.

$$p = \lambda M_c P \quad (4)$$

¹Cartesian coordinates are marked here with a crossbar over the letter whereas there is no additional tag for homogeneous coordinates.

Traditional calibration methods determine the coefficients of the camera matrix M_c minimizing the sum of the errors d done by mapping the known world points to the measured camera points:

$$\hat{M}_c = \operatorname{argmin}_{M_c} \sum_i d(\lambda M_c P_i, p_i) \quad (5)$$

Prerequisite for this approach is that the correspondences between points in the world and the target can be described by the pinhole camera model. The image of a scene taken by a pinhole camera is *undistorted*. A real camera uses a system of different lenses and an aperture within the course of the light rays to bundle the light projecting a scene onto the camera target, so the image of the scene becomes *distorted*. The distortion by the optical system is represented here shortly with the function $l : (p_c \rightarrow p)$, $p_c, p \in (\mathbb{R}, \mathbb{R}, 1)^T$. With this function, mapping a point onto the camera target may be written in contrast to Equation (4) as:

$$p = l(\lambda M_c P) \quad (6)$$

To be able to determine the camera parameters although the mapping is distorted, many authors typically describe the distortion l using a non-linear mathematical model. The radial distortion describes the major part of the distortion. It is estimated by most of the established calibration methods like [12]. Some approaches additionally model tangential distortion [5] and prism distortion [13]. By naming the distortion model l_θ with the respective parameter set θ , the camera parameters are again determined by minimizing the sum of errors:

$$(\hat{M}_c, \hat{\theta}) = \operatorname{argmin}_{M_c, \theta} \sum_i d(l_\theta(\lambda M_c P_i), p_i) \quad (7)$$

Multi camera calibration methods work quite similar but they use corresponding points on camera images from different views. Silhouettes [8] or corresponding feature points [9], on the recorded scene may be used.

An analytical solution of Equation (7) cannot be calculated in general. For this reason, an initial set of distortion parameters is presumed to be able to estimate the camera matrix M_c . Then the distortion parameters are adjusted to minimize the projection error. Afterwards, the camera matrix is estimated again. This is done iteratively until the projection

error is small enough or there are no improvements by adjusting the distortion parameters.

The basic premise for this approach is that the distortion of a real lens system can be completely described using algebraic formulas and that their degrees of freedom can be calculated. Both assumptions are usually not valid: The lenses are different to each other and normally have unknown characteristics. There are manufacturing tolerances in the fabrication process of both the lenses and the lens system; the lenses are not perfectly symmetrical, they are not exactly concentric and orthogonal to the optical axis, etc. In addition, the global optimum of Equation (7) cannot be calculated.

In this article we compensate the non-linear distortion l of the optical system prior to the camera calibration. The distortion is measured by associating every point of the camera target with a corresponding point on a known image plane. If it is known for every point of the camera target which point of the image plane it is the mapping of, then every camera image can be undistorted by reprojecting it onto that image plane.

A similar approach was presented in [7] where correspondences between camera target and image plane were measured using structured light shown on a plasma display panel and then used to undistort camera images. In contrast to that paper, we quantify exactly the projection error at each point of the camera target, whereas the other paper just fits in a line for visual control of its results. Also, we proceed using the undistorted images for camera calibration.

In photogrammetry, distortions are sometimes calculated in a comparable manner using a *réseau* technique [2]. A small list of correspondences is build by measuring the position of grid points projected onto a film at exposure. The undistorted positions for the points of the image are then interpolated using these measured control points. Due to the control points being too sparsely distributed, this approach lacks accuracy.

We prove algebraically that by undistorting camera images with the presented method the real camera target is substituted by a *virtual target* with known measurements and without distortions. The usual iterative two-step camera calibration procedure is substituted by a three-step non-iterative procedure. The first step of the procedure results in an undistorted image of the scene. A second step ad-

justs the virtual target to an upright position. This step can be skipped if the undistorted images are used for photogrammetry only. After both steps, there is a complete mapping from the real camera to a virtual camera which is without distortions and which has an optical axis identical to that of the real camera.

In a third step, the images undistorted with this mapping are used for traditional camera calibration, skipping the estimation of the non-linear parameters, to obtain the linear camera parameters of the virtual camera. The big advantage is that the undistorted image of a calibration pattern follows the linear pinhole camera model, so the calibration procedure is much more stable and precise since estimation of the non-linear parameters can be skipped.

The following Section 2 describes the compensation method in detail. Subsection 2.1 explains the first step, the measurement and compensation of the distortions, and Subsection 2.2 explains the second step, the adjustment to an upright position. The last Subsection 2.3 discusses practical considerations of the proposed method. In the following Section 3, measurements of the accuracy of the method and their results are shown. The last Section 4 summarizes the paper.

2 Measurement and compensation of non-linear distortion

We assume that it is practically impossible to completely describe the properties of the distortion of an optical system with a parametrized algebraic model. Therefore, we determine for every point p on the camera target the exact coordinates of the corresponding point P_V on a real image plane V whose image is p . Having corresponding points for all points on the camera target we are able to reproject the camera image onto the plane V .

Figure 1 clarifies this approach. The object space is projected onto the camera target through the focus. Due to the camera lens system, the projected image is distorted which is indicated by the non-rectangular camera target. The light ray that projects the world point P_W onto the point p on the camera target also goes through point P_V of the virtual target. As the image plane is undistorted by definition, the camera image becomes undistorted

by reprojecting it onto V , the virtual target².

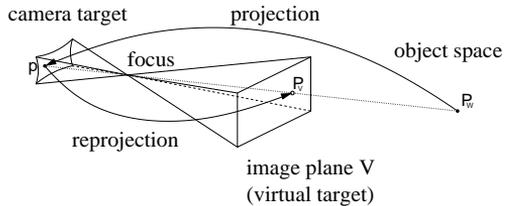


Figure 1: Mapping onto the virtual target

A set of points on the image plane is arranged on a rectangular grid. Within the coordinate system of the image plane, the position in homogeneous coordinates is $p_v = (u, v, 1)^T$. The image plane is at position \bar{C}_I in world coordinates and is rotated in space with the rotation matrix \bar{R}_I . Both matrices may be combined to $R_I = \bar{R}_I [I - \bar{C}_I]$. Naming the horizontal distance and the vertical distance between the points on the grid on the image plane with d_x and d_y , respectively, one can calculate the world coordinates P_V of the point p_v with $S_I(u, v, 1)^T = (d_x u, d_y v, 0, 1)^T$:

$$P_V = R_I S_I p_v \quad (8)$$

$$\text{with } R_I = \begin{bmatrix} \bar{R}_I & -\bar{R}_I \bar{C}_I \\ \mathbf{0} & 1 \end{bmatrix}$$

$$\text{and } S_I = \begin{bmatrix} d_x & 0 & 0 \\ 0 & d_y & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

The function f maps all points $(u, v, 1)^T$ of the image plane onto the camera target:

$$p = l(\lambda_I M_c P_V) \quad (9)$$

$$= l(\lambda_I M_c R_I S_I (u, v, 1)^T) \\ = f((u, v, 1)^T) \quad (10)$$

By determining the correspondences Φ between points $(u, v)^T$ on the image plane and points $(x, y)^T$ on the camera target $f : (u, v, 1) \rightarrow (x, y, 1)$ can be ascertained. Thus, by building the inverse $f^{-1} : (x, y, 1) \rightarrow (u, v, 1)$ one gets the mapping (provided that l is bijective):

$$f^{-1}(p) = S_I^{-1} R_I^{-1} M_c^{-1} \lambda_I^{-1} l^{-1}(p) \quad (11)$$

²In the following text, the term *image plane* is used when talking about the source of the calibration points. The term *virtual target* is used when the target of the reprojection is meant. However, these two planes are the same.

By reprojecting the point P_W onto the virtual target one receives

$$\begin{aligned} & f^{-1}(l(\lambda_W M_c P_W)) \\ &= S_I^{-1} R_I^{-1} M_c^{-1} \lambda_I^{-1} l^{-1}(l(\lambda_W M_c P_W)) \\ &= \frac{\lambda_W}{\lambda_I} S_I^{-1} R_I^{-1} P_W \end{aligned} \quad (12)$$

Since S_I is not a square matrix, S_I^{-1} can be calculated using the pseudo inverse. Please note that the term $S_I^{-1} R_I^{-1}$ has size 3×4 like a homogeneous camera matrix; therefore it maps the point P_W onto the virtual target with the camera matrix $M_V = S_I^{-1} R_I^{-1}$ with $\lambda_V = \lambda_W / \lambda_I$. Hence,

$$p_v = \lambda_V M_V P_W \quad (13)$$

This is the reason why we call the reprojection onto the image plane a *virtual camera*. In contrast to Equation 6, this mapping is calculated without the distortion by function l .

If the image plane is orientated exactly orthogonal to the optical axis of the camera and the horizontal and vertical axes of the image plane and the camera target are parallel to each other, then the reprojected image is free of any distortion, no matter what physical origin it has. If the image plane was not adjusted exactly, then the reprojected image shows a certain slanted position. Subsection 2.2 describes how to compensate this slant.

2.1 Undistorted Virtual Camera

The first step of the compensation method is to determine the correspondences Φ between points $(u_i, v_i, 1)^T$ on the image plane V and points $(x_i, y_i, 1)^T$ on the camera target. The exact position and orientation of the plane in the world is relatively irrelevant for this step given that only points within the image plane are mapped onto the camera target and the camera image is sharp enough to determine the position of projected points on the camera target.

In order to determine the correspondences, points are displayed on the visible part of the image plane. The method used for that does not matter in general as far as the position is exact and well-known. A conventional TFT-monitor is used here (Subsection 2.3) because a single white pixel on a black background has the effect of a point light source. For every point $(u_i, v_i, 1)^T$, $u_i, v_i \in \mathbb{N}$ displayed on

the image plane, which lies in the field of view of the camera, the position of its image on the camera target is determined with sub-pixel accuracy by calculating its centroid.

The list of correspondences $\Phi : (u_i, v_i, 1)^T \rightarrow (x_i, y_i, 1)^T$ of all observable points of the image plane describes the mapping function f with very fine granularity. We use f for two purposes:

- Determine the inverse function f^{-1} in order to convert camera target coordinates (2.1.1)
- Resample camera images onto the virtual target using a scanline-algorithm (2.1.2)

2.1.1 Building the Inverse Mapping Function

The points $(u_i, v_i, 1)^T$ on the image plane V build a dense grid at discrete equidistant coordinates, i.e. $u_i, v_i \in \mathbb{N}$. The corresponding points $(x_i, y_i, 1)^T$ on the camera target are distributed densely, too, but not necessarily at equidistant locations, i.e. $x_i, y_i \in \mathbb{Q}$. Inverting these correspondences directly would result in an inverse function $f^{-1} : \mathbb{Q}^2 \rightarrow \mathbb{N}^2$ of the optical mapping function f .

To be able to easily reproject coordinates measured on the camera target onto the virtual target, the inverse function f^{-1} should have a domain in \mathbb{N}^2 and its codomain in \mathbb{Q}^2 . This means, for every point $(x_j, y_j, 1)^T$, $x_j, y_j \in \mathbb{N}$, on the camera target a point $(u_j, v_j, 1)^T$, $u_j, v_j \in \mathbb{Q}$ on the image plane has to be determined by interpolation.

This is done here with an image warping algorithm [11]. When calculating the inverse function it is assumed, that the mapping function is locally linear, which is reasonable as the points are very close to each other.

The resulting inverse function f^{-1} can then be used to convert camera target coordinates into undistorted coordinates of the virtual target.

2.1.2 Undistortion Using a Scanline Algorithm

The list of correspondences may also be used directly, i.e. without inverting, to undistort camera images.

It was determined for every point P_i of the visible image plane onto what coordinates p_i on the camera target it maps. Therefore, it is possible to reproject the camera image onto the image plane by taking the pixel value at position p_i for the point P_i on the image plane. As the coordinates of the points p_i are not discrete in general, the color/grey values have

to be interpolated, e.g. using a Sinc-function with a Blackman-Harris window[3]. The filters for the horizontal and vertical direction are separable, and their coefficients have to be calculated only once for a correspondence list. Therefore the calculation expense for the undistortion is within bounds for this method. However, a bilinear interpolation would be less expensive, but would result in images of lower quality.

2.2 Deskewed Undistorted Virtual Camera

If the image plane is not orthogonal to the optical axis of the real camera, the reprojection adds a slant to the image. To completely undistort reprojected images, this slant has to be compensated. As it can be seen in Figure 1, the real camera target and the virtual target have the same focus. This fact does not change if the virtual target is tilted. Two mappings that have the same focus are connected by a homography, i.e. one mapping can be transferred into the other. The mapping tilting the virtual target into the correct orientation is given by

$$p'_v = M'_V M_V^{-1} p_v \quad (14)$$

Each point p_v is reprojected into space using the inverse of the camera matrix M_V of the virtual camera to be projected onto the upright virtual camera using the camera matrix M'_V . The term $M'_V M_V^{-1}$ is denoted as homography.

The camera matrix M'_V is a short form for $K'_V R'_V$ and is unknown from the beginning. By using a traditional camera calibration method, the matrices K_V and R_V of the virtual camera are determined using undistorted camera images (see 2.1.2). The intrinsic camera matrix K'_V should have the same focal length and the same image resolution as the virtual target, but the principal point of the upright virtual target is moved to P' , the center of the image.

To move the principal point to the image center, the optical axis of the virtual target has to be rotated. Figure 2 demonstrates this: On the left side, there is the real camera target with the optical axis A_c defined by R_c orthogonal on its center. The tilted virtual target is shown on the right side of the diagram. It is clear, that tilting the image plane may move the principal point outside of the actual image region. Now the principal point P is moved to

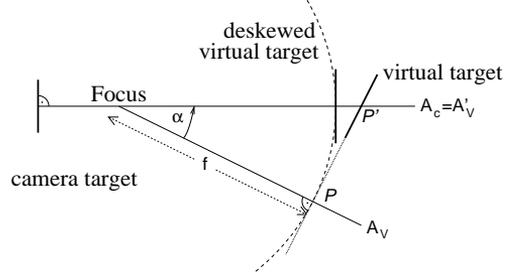


Figure 2: The virtual target is tilted into an upright position such that its principal point P moves to the center P' of the image.

the center P' of the image, rotating the optical axis of the virtual target to that of the real target. The angles α and β needed for that rotation can be calculated directly from the difference of P and P' . The optical axis A_V defined by R_V has to be rotated around these angles to obtain A'_V defined by R'_V , i.e. $R'_V = R_y(\beta)R_x(\alpha)R_V$ with $R_x(\alpha)$ and $R_y(\beta)$ being the rotation matrices around the x - and y -axis.

Finally, the homography $M'_V M_V^{-1}$ required to tilt the virtual target into an upright position according to Equation (14), is given by:

$$\begin{aligned} M'_V M_V^{-1} &= K'_V R'_V (K_V R_V)^{-1} \\ &= K'_V R_y(\beta) R_x(\alpha) R_V R_V^{-1} K_V^{-1} \\ &= K'_V R_y(\beta) R_x(\alpha) K_V^{-1} \end{aligned} \quad (15)$$

By applying this homography onto the camera matrix M_V of the virtual camera, the virtual camera gets an optical axis that is exactly orthogonal to the virtual target and which intersects it in its center. The deskewed virtual camera is completely described.

2.3 Practical Considerations

This subsection describes the technique to determine the correspondences required for the undistortion.

First tests were done using a conventional TFT monitor. It is reasonable to assume that its pixels are equidistant in both directions, their distance can be measured and the horizontal and vertical axes are orthogonal to each other. The prerequisites for using it as a image plane are therefore given.

To measure the correspondences the camera is orientated towards the TFT-monitor such that the camera image only shows pixels of the monitor. To prevent reverberations from the border of the monitor, a certain distance to it has to be kept. A white pixel on the monitor illuminates some sensor elements (\hat{x}_j, \hat{y}_j) of the camera target with the intensities $I(\hat{x}_j, \hat{y}_j)$. The center of gravity of the intensities is given by the sum of the positions of the elements each weighted with its intensity [1, Eq. 3.291]. For this method to work, the images of the points on the monitor must be relatively sharp.

To normalize the intensity values, a black and a white screen is shown and captured by the camera at the beginning. The brightness is adjusted to the best intensity range while preventing over- and underexposure. Next, the center, the direction of the axes, and the horizontal and vertical distances of the points on the camera target are determined by displaying three points in the center of the screen.

Now the actual measurement of correspondences is started. A pattern consisting of single points arranged uniformly with horizontal and vertical distances r_x and r_y , respectively, is shown full screen. The shorter the distance between the points, the more points are displayed at the same time and therefore, the less images have to be captured. On the other hand, the distance between the points has to be large enough such that the unsharp images of two pixels do not overlap. The tests for this article were done with a raster distance of $(r_x, r_y) = (32, 32)$.

Each point pattern is captured four times to reduce the effect of camera noise. Then the point pattern is displayed at a new position. After all $32 * 32 = 1024$ point patterns, the correspondence for every visible pixel on the monitor is measured; the correspondence list Φ is completely determined.

3 Experimental Results

In this section, results of our camera calibration method are presented and compared with a traditional calibration method. We used an IEEE1394 Prosilica EC 1380C single-CCD chip camera with a mounted Schneider-Kreuznach Cinegon 1.4/8-0512 industrial lens. The full-color RGB images used for the experiments were calculated using the high-quality AHD demosaicing algorithm [6]. The non-linear distortions were determined using a NEC

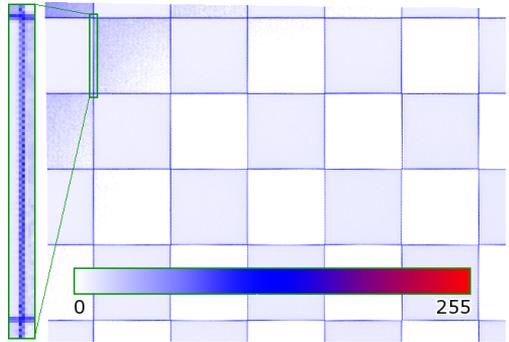


Figure 3: Upper left quarter of the difference between screen-shot and undistorted camera image of about 600×450 pixel size. White means no difference, blue medium difference and red means big difference. To be able to assess the colors an additional color wedge is added. On the left side, a zoom on the image shows that the line is perfectly undistorted and has constant width.

MultiSync LCD1860NX TFT monitor.

In the first step, the correspondence list was determined. Then, a checkerboard was displayed on the monitor without touching camera and monitor. The image captured by the camera was then undistorted using the scanline algorithm of Subsection 2.1.2. Now the difference between the undistorted camera image and the screen-shot was calculated and converted to a colored gradient to be able to assess the difference values. No difference is displayed as white, medium difference as blue and big difference as red. The upper left quarter of the resulting image is shown in Figure 3.

There are three main observations: First of all, one can recognize the checkerboard pattern. The reason for this is that the black of the screen-shot is really black (grey value 0) whereas the black as seen by the camera is only dark grey (grey value 10). The difference between black and dark grey can be observed. Secondly, the difference increases near the upper left corner. This can be traced to the fact that the pixel's luminosity on the TFT monitor lowers with a greater viewing angle. Both errors are due to an insufficient normalization of the intensity signal but have only a minor impact on the undistortion method.

The third observation is that one can clearly see the edges between the chessboards, due to a slightly

unsharp camera image. It is important that the edges are absolutely straight and have constant width. This means that the image was undistorted accurately. Otherwise, the width of the edges would differ.

A second test examines the accuracy of the undistortion more precisely. Camera images of the captured point patterns used for determining the correspondence list Φ are undistorted using the scan-line algorithm from Subsection 2.1.2. These undistorted images are again used as input for determining a correspondence list. If the determination of the position of the points as well as the compensation of the optical mapping errors based on the first correspondence list was perfect, every point in the second correspondence list would be mapped to its original location.

Measuring the distributions of the reprojection errors in horizontal and vertical direction indicate that both are approximately Gaussian distributed. The standard deviation of the error in horizontal direction is 0.0344 pixel and in vertical direction 0.0305 pixel. More than 99% of the errors are lower than 0.09 pixel and 0.078 pixel, respectively.

The last experiment shows that by preprocessing camera images with the presented undistortion method, the accuracy of traditional camera calibration methods is substantially improved.

To estimate the intrinsic camera parameters of a traditional calibration method, we use the camera calibration tool of Jean-Yves Bouguet³ which implements Zhang/Heikkilä [12][5]. Optionally, some non-linear distortion parameters, i.e. radial distortion of first to third order (κ_1 - κ_3) and tangential distortion of first and second order (t_1 , t_2) can be estimated. Pictures of a flat calibration pattern in 10 different positions and orientations are captured for this purpose.

The standard deviations of the parameters in Table 1 are given in pixel. In the first column the method used to undistort the camera images is listed. The second to fifth column give the standard deviations of the intrinsic camera parameters⁴. Finally, the reprojection error for the calibration points is given in the last two columns.

³Freely available under http://www.vision.caltech.edu/bouguet_j/calib_doc.

⁴The used calibration toolbox outputs three times the parameters' standard deviation instead because it aims to show the uncertainties.

The first row shows the standard deviation of the linear intrinsic camera parameters without prior estimation of the non-linear distortion parameters. In the following five rows, the stabilities of the intrinsic parameters are listed if the non-linear distortion parameters are estimated using the toolbox. The second to last row shows the results of the estimated parameters if the camera images are first undistorted using our method and then are used for estimating the linear camera parameters. The last row lists the results when applying both the traditional and our method for camera calibration.

As can be seen from the standard deviations in the table, the accuracy of the estimation of the intrinsic linear camera parameters is substantially improved if the images are undistorted before with the presented method, while decreasing the reprojection error. Since different camera parameters have different stability in the reference method, only the most stable results are used here for comparison: It can be calculated, that the accuracy of the focal lengths $f s_x$ and $f s_y$ is improved by 20% and 7.5%, respectively⁵, whereas the stability of the estimation of the principal point (h_x , h_y) is yet improved by 37.5% and 40%, respectively. When trying to estimate any distortion parameters on the undistorted images, the reprojection error is slightly improved but the accuracy of the camera parameters drops.

It has to be pointed out that these are results for a *normal* lens, i.e. no wide-angled or fisheye lens. This is especially remarkable because the lens is of high industrial quality, which normally means it was manufactured with high precision.

4 Conclusions

This article introduces a non-iterative camera calibration procedure. The method uses a flexible calibration pattern displayed by a TFT monitor. By capturing a high number of different calibration patterns from the monitor, a correspondence list between pixels on the camera target and points on the calibration patterns is calculated. Using this correspondence list, subsequent images may be undistorted precisely by projecting them onto a virtual target. The intrinsic linear camera parameters of this virtual target are then determined using a traditional calibration method. These can be adjusted

⁵Since the pixel width and height need not be equal the focal lengths may be different.

Undistortion method	Intrinsic parameters				Reprojection error		
	$\sigma[f s_x]$	$\sigma[f s_y]$	$\sigma[h_x]$	$\sigma[h_y]$	$\sigma[e_x]$	$\sigma[e_y]$	
no undistortion	1.6964	1.7701	0.9062	1.1979	0.9673	0.7906	
reference	undistortion using κ_1	0.5728	0.5877	0.4156	0.4982	0.2534	0.3144
	undistortion using κ_1, κ_2	0.4846	0.4980	0.3547	0.4228	0.1905	0.2812
	undistortion using κ_1, κ_2, t_1	0.4700	0.4836	0.3430	0.6514	0.1899	0.2684
	undistortion using $\kappa_1, \kappa_2, t_1, t_2$	0.4697	0.4833	0.8893	0.6513	0.1898	0.2681
	undistortion using $\kappa_1-\kappa_3, t_1, t_2$	0.4833	0.4976	0.8898	0.6509	0.1896	0.2681
new	undistortion using virtual camera	0.3749	0.4479	0.2144	0.2555	0.1599	0.2316
	undistortion using virtual camera and additionally $\kappa_1-\kappa_3, t_1, t_2$	0.3785	0.4335	0.7552	0.5151	0.1433	0.2236

Table 1: Impact of the virtual camera on camera calibration. The standard deviations of the parameters are given in pixel.

to rotate the optical axis of the virtual camera onto that of the real one.

Examination of the accuracy of the undistortion method shows a very low reprojection error. Undistorting camera images with our method greatly improves the accuracy of traditional calibration methods by 7.5% to 39.5%. Since the undistortion process is independent from the actual camera calibration process, it is very suitable to be used as a pre-processing method.

In particular, multi camera calibration methods, which are very complicated for distorted camera images, should greatly benefit by using this method.

References

- [1] I. N. Bronstein and K. A. Semendjajew. *Taschenbuch der Mathematik*. Verlag Harri Deutsch, Thun und Frankfurt/Main, 6th edition, 2005.
- [2] K. Hanke and P. Grussenmeyer. Architectural photogrammetry: Basic theory, procedures, tools. *ISPRS Commission 5 tutorial*, September 2002.
- [3] F. J. Harris. On the use of windows for harmonic analysis with the discrete fourier transform. *Proceedings of the IEEE*, 66(1):51–83, January 1978.
- [4] R. Hartley and A. Zisserman. *Multiple View Geometry in Computer Vision*. Cambridge University Press, second edition, 2003.
- [5] J. Heikkilä. Geometric camera calibration using circular control points. *IEEE Trans. Pattern Anal. Mach. Intell.*, 22(10):1066–1077, 2000.
- [6] K. Hirakawa and T. W. Parks. Adaptive homogeneity-directed demosaicing algorithm. *IEEE Transactions on Image Processing*, 14(3), March 2005.
- [7] R. Sagawa, M. Takatsuji, T. Echigo, and Y. Yagi. Calibration of lens distortion by structured-light scanning. In *IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 832–837, 2005.
- [8] S. N. Sinha, M. Pollefeys, and L. McMillan. Camera network calibration from dynamic silhouettes. *Proceedings of the 2004 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 1:195–202, July 2004.
- [9] P. Sturm and B. Triggs. A factorization based algorithm for multi-image projective structure and motion. In *ECCV (2)*, pages 709–720, 1996.
- [10] R. Y. Tsai. A versatile camera calibration technique for high-accuracy 3d machine vision metrology using off-the-shelf tv cameras and lenses. *IEEE Journal of Robotics and Automation*, RA-3, No.4, August 1987.
- [11] G. Wolberg. *Digital Image Warping*. IEEE Computer Society Press, Los Alamitos, CA, USA, 1994.
- [12] Z. Zhang. A flexible new technique for camera calibration, August 2002.
- [13] H. Zollner and R. Sablatnig. A method for determining geometric distortion of off-the-shelf wide-angle cameras. In *27th DAGM Symposium on Pattern Recognition, Vienna*, 2005.