

Calibration of Lens Distortion by Structured-Light Scanning

Ryusuke Sagawa, Masaya Takatsuji, Tomio Echigo and Yasushi Yagi

The Institute of Scientific and Industrial Research

Osaka University

8-1 Mihogaoka, Ibaraki, Osaka, 567-0047, JAPAN

{sagawa,takatuji,echigo,yagi}@am.sanken.osaka-u.ac.jp

Abstract— This paper describes a new method to automatically calibrate lens distortion of wide-angle lenses. We project structured-light patterns using a flat display to generate a map between the display and the image coordinate systems. This approach has two advantages. First, it is easier to take correspondences of image and marker (display) coordinates around the edge of a camera image than using a usual marker, e.g. a checker board. Second, since we can easily construct a dense map, a simple linear interpolation is enough to create an undistorted image. Our method is not restricted by the distortion parameters because it directly generates the map. We have evaluated the accuracy of our method and the error becomes smaller than results by parameter fitting.

Index Terms— Calibration of lens distortion, structured-light scanning, non-parametric

I. INTRODUCTION

A wide field of view is often required for tasks such as surveillance and robot navigation that involve observing the environment. There are several approaches to acquire the necessary wide field of view. For example, using wide-angle lenses [1], omnidirectional mirrors [2], [3], a moving camera [4] or a multiple camera system [5].

Using a wide-angle lens is a simple solution for such applications. However, images through a wide-angle lens, such as a fish-eye lens, are greatly distorted. Since such distortion is not desirable for image processing and needs to be removed, the projection model is handled like a pin-hole camera model.

Several approaches have been proposed to calibrate lens distortion. Most usually assume distortion models that contain radial and tangential distortions. The latter effect is also called *decentering distortion* [6], [7]. If (x_d, y_d) is a point of a distorted image and (x_u, y_u) is the corresponding point of the undistorted image,

$$\begin{aligned} x_u &= \left(1 + \sum_{i=1}^{\infty} K_i r^{2i}\right) x_d + \\ &\quad (2P_1 \bar{x}\bar{y} + P_2(r^2 + 2\bar{x}^2)) \left(1 + \sum_{i=1}^{\infty} P_{i+2} r^{2i}\right) \\ y_u &= \left(1 + \sum_{i=1}^{\infty} K_i r^{2i}\right) y_d + \\ &\quad (P_1(r^2 + 2\bar{y}^2) + 2P_2 \bar{x}\bar{y}) \left(1 + \sum_{i=1}^{\infty} P_{i+2} r^{2i}\right), \end{aligned} \quad (1)$$

where $\bar{x} = x_d - C_x$, $\bar{y} = y_d - C_y$, $r^2 = \bar{y}^2 + \bar{x}^2$. C_x and C_y are the optical center. K_i and P_i are the parameters of the radial and tangential distortions, respectively. Actually, the higher order terms on the right-hand of (1) are ignored. For example, (1) is approximated as follows [1], [7]:

$$\begin{aligned} x_u &= (1 + K_1 r^2 + K_2 r^4) x_d + (2P_1 \bar{x}\bar{y} + P_2(r^2 + 2\bar{x}^2)) \\ y_u &= (1 + K_1 r^2 + K_2 r^4) y_d + (P_1(r^2 + 2\bar{y}^2) + 2P_2 \bar{x}\bar{y}), \end{aligned} \quad (2)$$

Some studies [8], [9], [10] have used known calibration markers to estimate distortion parameters. In contrast, *nonmetric* methods [1], [11], [12], [13] have also been proposed. These methods estimate the distortion parameters by fitting straight lines in the scene and their corresponding curved lines in the image, which then must be projected as straight lines in the perspective projection.

The above methods that estimate the distortion parameters are restricted by the definition of their parameters. If a method only estimates radial distortion, it cannot recover tangential distortion. Even if all the parameters in (2) are estimated, the error becomes large at the points that are far from the optical center. On the other hand, some other *non-parametric* approaches do not use these parameters. Green et. al. [14] applied local affine transformations to map grid elements in the distorted image to grid elements in the undistorted image. However, since the grid patterns are sparse and the transformations are linear, the sampling is not smooth. Goshtasby [15] modeled the distortion parameters using Bezier patches. Though the sampling becomes smooth, the error still remains since the control vertices are sparse. Ying and Hu [16] used markers attached on a hemisphere and generated a non-parametric imaging model for distortion correction.

In this paper, we propose a novel method for calibrating lens distortion by projecting dense markers. Our method obtains the relationships between distorted and undistorted points for all pixels. Therefore, we can straightforwardly generate an undistorted image without needing to estimate the distortion parameters. The undistorted image becomes the one generated by the perspective projection. To generate dense markers, we utilize a flat display, such as a liquid crystal display (LCD) and a plasma display panel (PDP). Since these displays are planar and have very high resolution, we can generate dense mapping of the scene and image. This approach has two advantages. First, if we use

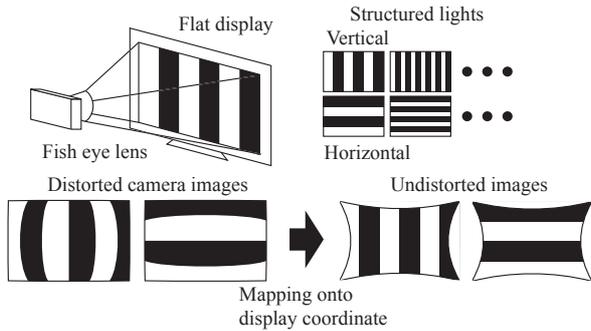


Fig. 1. Calibration of distortion by structured-light scanning

a marker as a world coordinate, e.g. a checker board, it is not easy to find the correspondence of image and marker coordinates around the edge of a camera image because of lens distortion. However, this method easily takes correspondences of image and marker (display) coordinates even around the edge of a camera image. Second, because of the dense sampling of markers, a simple linear interpolation is enough to generate a smooth image.

In Section II, we describe a method of acquiring dense markers using a flat display and of calibration using the acquired markers. We experiment with our proposed method and estimate its accuracy in Section III. Finally we summarize this paper in Section IV.

II. CALIBRATION BY STRUCTURED-LIGHT SCANNING

Instead of estimating the parameters of distortion, Our method creates a map between the distorted and undistorted points for all pixels. As it is necessary to observe many markers to create such a dense map, we use a structured-light technique to generate dense markers. This technique is used by structured-light range finders [17], [18], [19]. For range measurement, the structured lights are projected onto a target object. In this paper, the camera to be calibrated directly observes the structured lights.

Fig. 1 shows the basic outline of our proposed method. We use a flat display such as an LCD to project the light patterns. The camera observes two series of light patterns, that are parallel to the x - or y -axis of the display coordinate system. Since each pixel is coded by time-multiplexing, the position of the pixel in the display coordinate system is computed. By mapping each pixel onto the corresponding position in the display coordinate system, the undistorted image can be obtained.

A. Spatial Coding

In our implementation, we use Gray codes [20] as the structured-light patterns. The advantage of using Gray code is that the number of images required is small. Fig. 2 shows encoding by a Gray code. By decoding the acquired light pattern, we compute the coordinate along the axis that is parallel to the stripe of the pattern. Since we use two series of patterns that are perpendicular to each other, we obtain the mapping function f from the display coordinate system

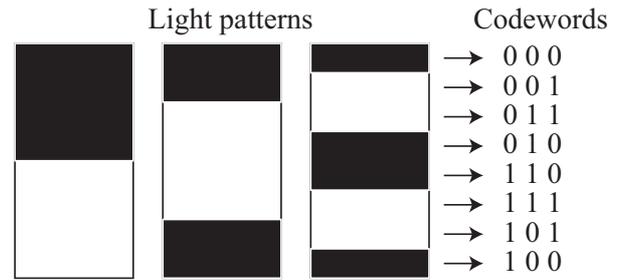


Fig. 2. Spatial coding by a Gray code

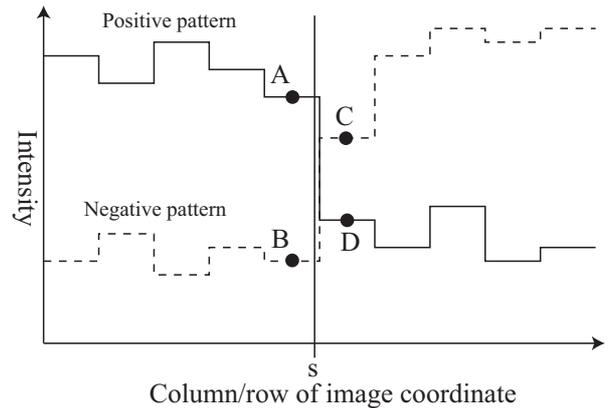


Fig. 3. Sub-pixel matching of stripe patterns

to the image coordinate system of the camera:

$$(x, y) = f(i, j), \quad (3)$$

where (i, j) is the point in the display and it is projected onto (x, y) in the camera image.

To estimate the mapping function, our method consists of the following steps:

- 1) Compute with subpixel accuracy the intersecting points of stripes along the column and row axis.
- 2) Find the crossing points of perpendicular stripes.
- 3) Interpolate values to determine the projected points for all (i, j) .

1) *Compute Intersecting Points with Subpixel Accuracy:* To improve the accuracy of the matching between the display and image coordinate systems, we measure the edges between adjacent stripes using sub-pixel coordinates. Thus, the display coordinate (i, j) is assigned to the edges of the stripes. Fig. 3 shows the detection of an edge with sub-pixel accuracy. First, we project the positive and negative patterns. The interpolated edge exists at the intersecting position s of the stripe profiles. s is computed as follows [18]:

$$s = \frac{A - B}{(A - B) + (C - D)}, \quad (4)$$

where A, B, C and D are the intensities of pixels. We find the intersecting positions along both the column and row axes of the image coordinate.

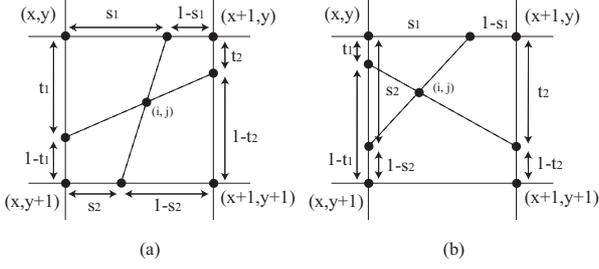


Fig. 4. Compute the crossing point of the edges of the perpendicular stripes.

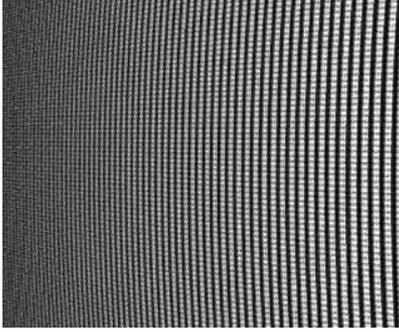


Fig. 5. Stripes cannot be recognized in the left part.

2) *Find Crossing Points of Stripes:* Next, we find the crossing point (x, y) of the perpendicular stripes in the image coordinate, since it is the projected point of (i, j) in the display coordinate. Fig. 4 shows two examples of crossing stripes. The edges of the stripes are indicated by dotted lines. Since we have computed the intersecting points along the column and row axes, we compute the crossing point by linear interpolation of the intersecting points.

We compute the four subpixel coordinates, s_1, s_2, t_1 and t_2 , by (4). s_1 and s_2 are the coordinates for the stripes along i -axis of the display coordinate, and t_1 and t_2 are those along j -axis of the display coordinate. In Fig. 4(a), the crossing point $(x_{(i,j)}, y_{(i,j)}) = f(i, j)$ is computed by solving the following equation for the parameters a and b :

$$\mathbf{p}_1 + a(\mathbf{p}_2 - \mathbf{p}_1) = \mathbf{p}_3 + b(\mathbf{p}_4 - \mathbf{p}_3), \quad (5)$$

where $\mathbf{p}_1, \mathbf{p}_2, \mathbf{p}_3$ and \mathbf{p}_4 are the position of the intersecting points. Though there are some cases of the crossing points such as Fig. 4(b), they are also computed in similar manner.

3) *Determine Projected Points for All Display Points:* If the width of a stripe is very narrow, the edge cannot be detected as shown in Fig. 5. In such a case, it is impossible to compute $f(i, j)$ for all (i, j) . Thus, we interpolate it by using the neighborhood. In Fig. 6, The corresponding positions x_1 and x_3 for i_1 and i_3 are found. However, x_2 for i_2 is not computed because the edge of i_2 is not detected. Then, we interpolate x_2 by using x_1 and x_3 linearly.

B. Normalization of Coordinate

To generate an undistorted image, it is necessary to normalize the mapping function $f(i, j)$. The reasons are:

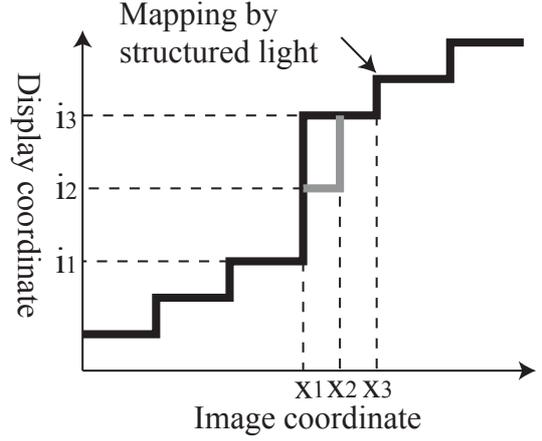


Fig. 6. The corresponding coordinate x_2 for i_2 is interpolated by using x_1 and x_3 .

- 1) the resolutions of image and display are different, and
- 2) the display may not be perpendicular to the optical axis of the camera.

In the neighborhood of the image center, we can assume that the distortion is quite small and so is negligible. Thus, a 3×3 matrix H exists, which satisfies

$$a \begin{pmatrix} x \\ y \\ 1 \end{pmatrix} = H \begin{pmatrix} f_i^{-1}(x, y) \\ f_j^{-1}(x, y) \\ 1 \end{pmatrix}, \quad (6)$$

where $(f_i^{-1}(x, y), f_j^{-1}(x, y)) = f^{-1}(x, y)$ and a is an arbitrary scale factor. H can be solved if more than three points are given in the neighborhood of the image center.

Now, we define a new mapping function $g(u, v)$ as follows:

$$(x, y) = f(H^{-1}(u, v, 1)^T) = f(h(u, v)) = g(u, v), \quad (7)$$

where $h(u, v) = (i, j)$ which satisfies $a(i, j, 1)^T = H^{-1}(u, v, 1)^T$. Because of the normalization, the resolution around the image center does not change after applying $g(u, v)$.

C. Mapping of Undistorted and Distorted Images

Since we only measure the correspondences of the display and image coordinates on the edges of the stripes, they are not defined for all pixels of the undistorted image. However, we measure them densely enough to interpolate them by simple linear interpolation.

If a point (u, v) of the undistorted image is transformed as

$$(i + \Delta i, j + \Delta j) = h(u, v), \quad (8)$$

where i and j are integers and $0 \leq \Delta i < 1, 0 \leq \Delta j < 1$, the corresponding point of the camera image is computed by bilinear interpolation as follows:

$$\begin{aligned} (x, y) &= (1 - \Delta i)(1 - \Delta j)f(i, j) \\ &+ \Delta i(1 - \Delta j)f(i + 1, j) \\ &+ (1 - \Delta i)\Delta j f(i, j + 1) \\ &+ \Delta i\Delta j f(i + 1, j + 1). \end{aligned} \quad (9)$$

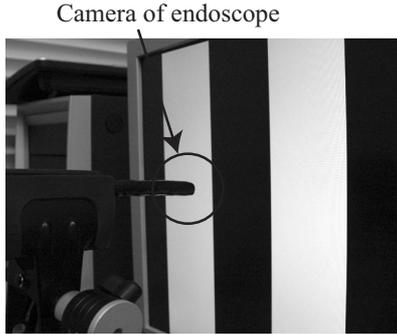


Fig. 7. Projecting structured-light pattern to a camera of an endoscope.

Then, $g(u, v)$ for all (u, v) can be computed by (8) and (9).

III. EXPERIMENTS

We now evaluate the accuracy of our method. To project structured-light patterns, we used a 20.1-inch flat panel LCD with 1600×1200 pixels and 0.255mm pixel pitch. The maximum view angle is $\pm 88^\circ$. The number of images in each series is 11. Thus, the size of display coordinate system is $2^{11} \times 2^{11}$.

We first experimented with an endoscopic camera. The image angle is 140° (diagonal). The experimental set-up is shown in Fig. 7. Since the resolution of the camera image (676×672 pixels) is lower than the display, the lower bits of Gray codes are not recognized. Thus, we interpolate the corresponding points as described in Section II-A. Fig. 8 shows the generation of the undistorted image from a camera image of the endoscope. Each correspondence of coordinates are computed by $f(i, j)$ and $h(u, v)$. The left image is an input camera image and the middle image is the corresponding area in the display coordinate. The right image is the generated undistorted image by normalization, the size of which is (1736×1736) pixels. It successfully takes correspondences of image and display coordinates, even around the edge of a camera image.

To evaluate the accuracy, we estimate error by fitting a line in the undistorted image, because a straight line in the scene should be straight in the undistorted images. In Fig. 9, we compare three images: (a) a camera image, (b) the undistorted image by our method, and (c) the undistorted image by parameter fitting. The stripe pattern in Fig. 9 is not used to compute the correspondence of coordinates. We use the camera calibration function of OpenCV [21] for parameter fitting and the parameters are defined as (2). By fitting dotted lines in Fig. 9 to straight lines, the RMS errors become (a) 16.6 pixel, (b) 0.83 pixel, and (c) 1.18 pixel. Thus, the results from our method are better than the results of parameter fitting.

Next, we used a digital still camera, a Nikon D70, and a wide-angle lens, an AF DX Fisheye Nikkor ED 10.5mm F2.8G. The size of acquired image is 3008×2000 pixels and the image angle is 180° (diagonal). The experimental set-up and the acquired images of the patterns are shown

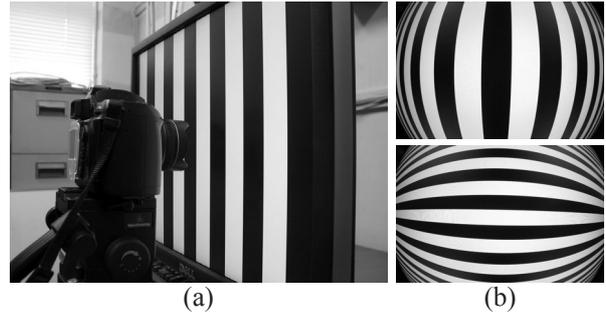


Fig. 10. (a) Experimental set-up for projecting structured-light patterns with D70, and (b) acquired images of structured-light patterns.

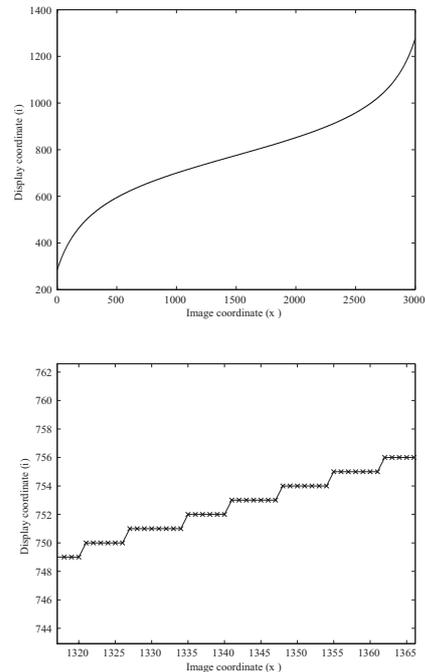


Fig. 12. The mapping result $i = f_i^{-1}(x, y)$ by structured-light patterns without subpixel estimation. Bottom: a zoom-up of the central area.

in Fig. 10. The distance between the camera and display is about 3cm.

Fig. 11 shows examples of camera images and undistorted results. Fig. 11(a) is the result of the lower image of Fig. 10(b). The size of the undistorted image is 5328×4288 pixels. Fig. 11(b) is an example of the camera image and Fig. 11(c) is the undistorted result. Our method successfully generates undistorted images.

Fig. 12 shows an example of the relationship in $(i, j) = f^{-1}(x, y)$ without subpixel estimation. Since the image resolution is higher than the display, it is a stepwise function around the image center. However, the function $f(i, j)$ becomes smooth since it is computed with subpixel accuracy by (4) and (5) (see Fig. 13).

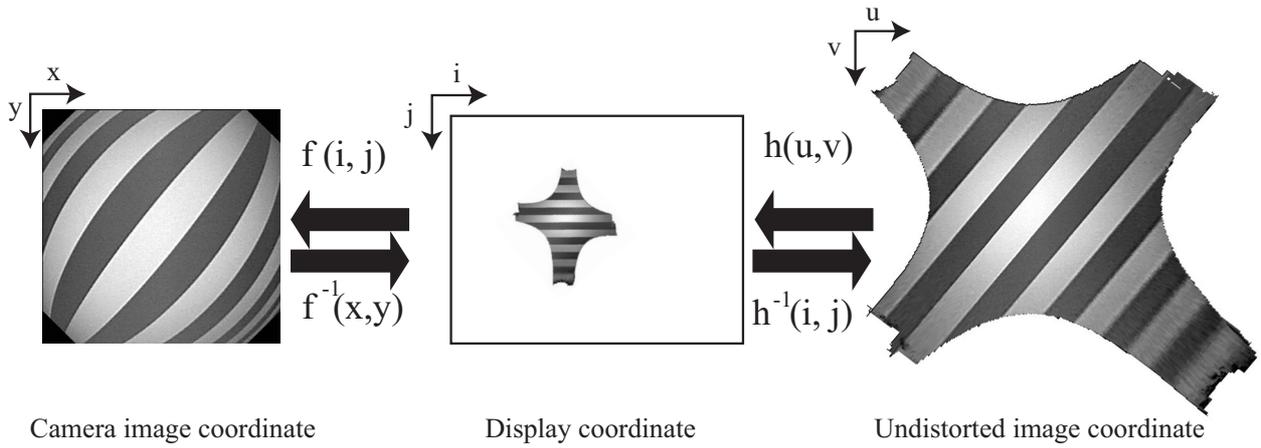


Fig. 8. Generating the undistorted image from a camera image of an endoscope and the correspondence of coordinates.

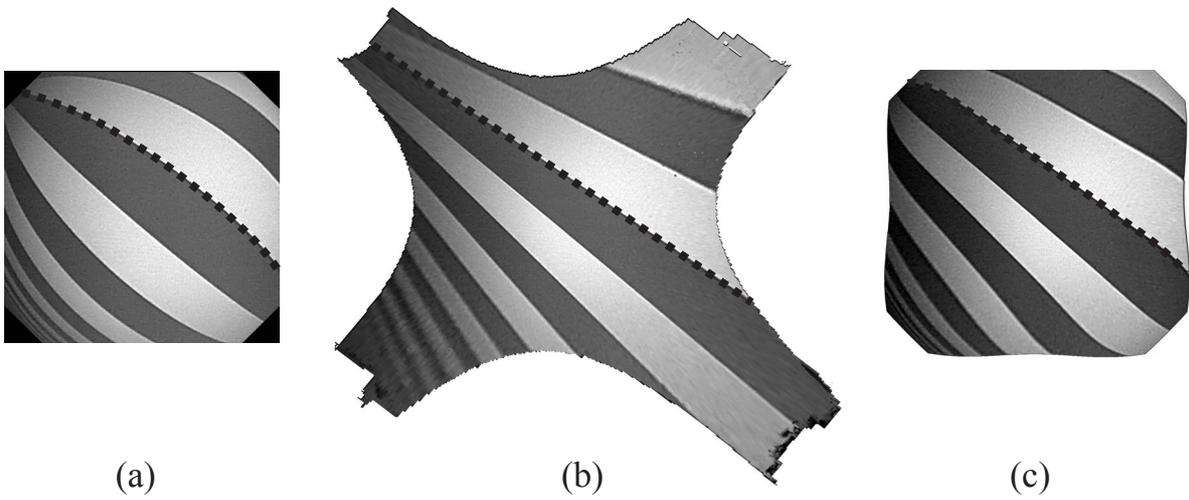


Fig. 9. Comparison of distorted and undistorted images: (a) a camera image, (b) the undistorted image by our method, and (c) the undistorted image by parameter fitting.

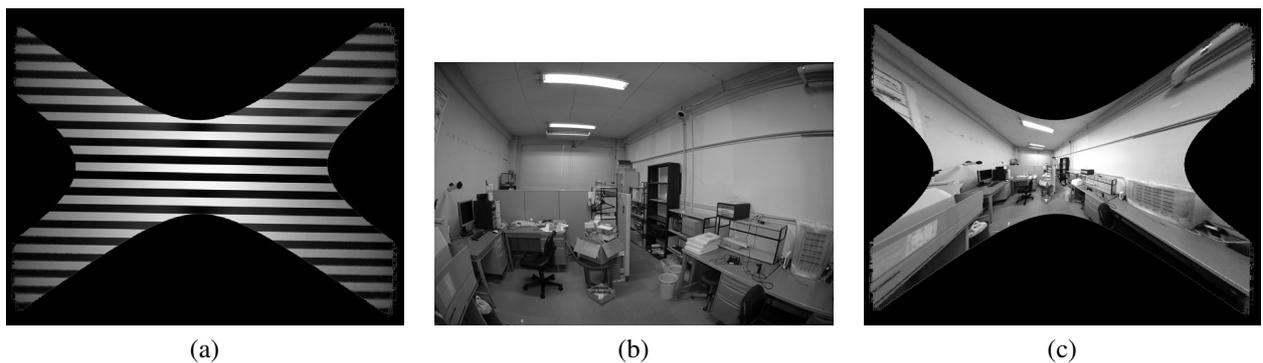


Fig. 11. Distorted and Undistorted images of D70: (a) the undistorted image of Fig. 10(b), (b) a distorted camera image and (c) the undistorted image.

IV. CONCLUSION

This paper described a new method to automatically calibrate lens distortion of wide-angle lenses. Calibration of lens distortion is necessary for some geometrical analyses, for example, image mosaicing and three dimensional

modeling. We project structured-light patterns using a flat display to generate a map between the display coordinate system and the image coordinate system. This approach has two advantages. First, since it is easy to take correspondences of image and display coordinates even around the

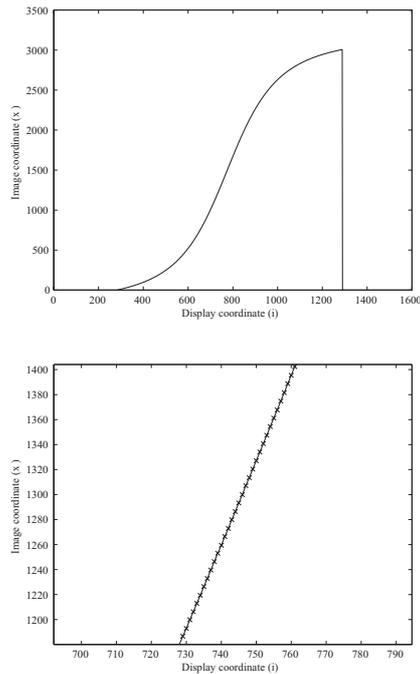


Fig. 13. The interpolated function $x = f_x(i, j)$ by structured-light patterns. Bottom: a zoom-up of the central area.

edge of a camera image, our method successfully generates an undistorted image from a camera image with large distortion. Second, since we can easily construct a dense map, a simple linear interpolation is enough to create an undistorted image. Our method is not restricted by the distortion parameters because it directly generates the map. We have evaluated the accuracy of our method and the error is smaller than results from a parameter fitting method.

REFERENCES

- [1] R. Swaminathan and S. Nayar, "Nonmetric calibration of wide-angle lenses and polycameras," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 22, no. 10, pp. 1172–1178, 2000.
- [2] K. Yamazawa, Y. Yagi, and M. Yachida, "Omnidirectional imaging with hyperboloidal projection," in *Proceedings IEEE/RSJ International Conference on Intelligent Robots and Systems*, vol. 2, July 1993, pp. 1029–1034.
- [3] S. K. Nayar, "Catadioptric omnidirectional camera," in *Proc. of IEEE Conference on Computer Vision and Pattern Recognition*, Puerto Rico, June 1997.
- [4] S. Peleg and M. Ben-Ezra, "Stereo panorama with a single camera," in *Proc. IEEE Conference on Computer Vision and Pattern Recognition*, Ft. Collins, Colorado, June 1999, pp. 395–401.
- [5] S. Shimizu, K. Yamamoto, C. Wang, Y. Satoh, H. Tanahashi, and Y. Niwa, "Moving object detection using depth information obtained by mobile stereo omni-directional system (sos)," in *Proc. Asia Conference on Computer Vision (ACCV2004)*, vol. 1, 2004, pp. 336–341.
- [6] A. Conrady, "Decentering lens systems," *Monthly Notices of the Royal Astronomical Society*, vol. 79, pp. 384–390, 1919.
- [7] D. Brown, "Decentering distortion of lenses," *Photogrammetric Eng.*, vol. 32, no. 3, pp. 444–462, 1966.
- [8] R. 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*, vol. 3, no. 4, pp. 323–344, 1987.
- [9] J. Weng, P. Cohen, and M. Herniou, "Camera calibration with distortion models and accuracy evaluation," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 14, no. 10, pp. 965–980, 1992.
- [10] Z. Zhang, "A flexible new technique for camera calibration," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 22, no. 11, pp. 1330–1334, 2000.
- [11] S. B. Kang, "Radial distortion snakes," in *IAPR Workshop on Machine Vision Applications*, Tokyo, 2000, pp. 603–606.
- [12] S. Becker and V. B. Bove, "Semiautomatic 3-d model extraction from uncalibrated 2-d camera views," in *SPIE Visual Data Exploration and Analysis II*, vol. 2410, Feb 1995, pp. 447–461.
- [13] G. Stein, "Lens distortion calibration using point correspondences," in *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, 1997, pp. 602–608.
- [14] W. Green, P. Jespen, J. Kreznar, R. Ruiz, A. Schwartz, and J. Seidman, "Removal of instrument signature from mariner 9 television images of mars," *Applied Optics*, vol. 14, no. 1, pp. 105–114, 1975.
- [15] A. Goshtasby, "Correction of image deformation from lens distortion using bezier patches," *Computer Vision, Graphics, and Image Processing*, vol. 47, no. 3, pp. 385–394, September 1989.
- [16] X. Ying and Z. Hu, "Distortion correction of fisheye lenses using non-parametric imaging model," in *Proc. Asian Conference on Computer Vision*, Jeju, Korea, 2004, pp. 527–532.
- [17] J. Posdamer and M. Altschuler, "Surface measurement by space-encoded projected beam systems," *Computer Graphics and Image Processing*, vol. 18, no. 1, pp. 1–17, 1982.
- [18] K. Sato and S. Inokuchi, "Range-imaging system utilizing nematic liquid crystal mask," in *Proc. International Conference on Computer Vision*, 1987, pp. 657–661.
- [19] J. Gühring, "Dense 3-d surface acquisition by structured light using off-the-shelf components," in *Videometrics and Optical Methods for 3D Shape Measurement*, vol. 4309, 2001, pp. 220–231.
- [20] F. Gray, "Pulse code communication," US Patent, March 17 1953, 2,632,058.
- [21] Open Source Computer Vision Library, <http://www.intel.com/research/mrl/research/opencv/>.