Radial Distortion Refinement by Inverse Mapping-Based Extrapolation

Ho Gi Jung, Yun Hee Lee, Pal Joo Yoon
MANDO Corporation, Korea
E-mail: hgjung@mando.com

Jaihie Kim
Yonsei University, Korea
E-mail: jhkim@yonsei.ac.kr

Abstract

“Caltech Calibration Toolbox” by Jean-Yves Bouguet is one of the most famous open-source calibration tools [1]. Although it works well for general camera lenses, it reveals problems in case of wide-angle lens. This paper analyzes the problem and shows that the insufficient calibration of radial distortion is caused by least-square based polynomial fitting. To overcome the problem, this paper proposes novel refinement method using inverse mapping-based extrapolation. The proposed method is expected to be an efficient supplement and is opened to the public in the appendix as a short Matlab script.

1. Introduction

Camera calibration is the first step towards computational computer vision [2]. Both intrinsic and extrinsic parameters are essential to the precise 3D reconstruction of real world object from projected 2D images. Although, for a long time, various calibration methods are proposed [2], computer vision researchers who are focusing on application topics prefer to use already proved open-source calibration tools rather than implement methods explained in papers [3]. “Caltech Calibration Toolbox” by Jean-Yves Bouguet is one of the most famous open-source calibration tools.

Although Caltech Calibration Toolbox is able to successfully calibrate general kinds of lenses, it fails in case of wide-angle lens, sometimes called fisheye lens. Fig. 1 is an input image captured with typical wide-angle lens, of which FOV (Field Of View) is about 150°. Fig. 2 shows the result of correction with distortion parameters estimated by Caltech Calibration Toolbox. It can be observed that the central portion of the resultant image is successfully undistorted but the outer portion still has severe distortion. Recently, many people pay attention to the application of non-classical camera such as wide-angle lens and omni-directional vision sensor [4]. Therefore, the improvement of Caltech Calibration Toolbox to overcome the wide-

angle lens related problem is supposed to be interesting to vision application engineers.

Figure 1. Input image captured with 150° FOV Lens (size: 1024x768)

Figure 2. Undistorted image by Caltech Calibration Toolbox (size: 4096x3072)

First, this paper analyzes the insufficient correction of radial distortion by Caltech Calibration Toolbox. General calibration pattern cannot cover full range of wide-angle lens' FOV. Furthermore, radial distortion curve should have positive slope and negative slope changing rate, which rapidly deviates just after data supported region. Second, this paper proposes a novel refinement method, which can be fulfilled after the execution of Caltech Calibration Toolbox. It is based on the fact that the inverse mapping of radial distortion curve is suitable for polynomial fitting. The proposed

method can be implemented by several dozens of Matlab command lines.

2. Analysis of calibration error

2.1. Radial distortion model

It is generally accepted that dominant lens distortion is radial distortion, sometimes called barrel distortion, and it can be modeled as a 5th order polynomial like Eq. (1) [5]. Where, r_d denotes the radius from distortion center in distorted image and r_u denotes the radius from image center in undistorted image. This paper assumes that Caltech Calibration Toolbox can successfully estimate effective focal lengths F_x, F_y , and optical center (O_x, O_y) . Although Caltech Calibration Toolbox uses both radial distortion and tangential distortion [1], only radial terms are considered herein. Therefore, Caltech Calibration Toolbox is thought to estimate only a_1, a_3 like Eq. (2). Where, kc is the estimate of distortion parameters.

$$r_d = a_1 r_u^5 + a_2 r_u^4 + a_3 r_u^3 + a_4 r_u^2 + a_5 r_u + a_6 \quad (1)$$

$$\begin{aligned} x_d &= x_u + x_u(kc(1)r_u^2 + kc(2)r_u^4) \\ &= \cos \theta(kc(2)r_u^5 + kc(1)r_u^3 + 1) \\ y_d &= y_u + y_u(kc(1)r_u^2 + kc(2)r_u^4) \\ &= \sin \theta(kc(2)r_u^5 + kc(1)r_u^3 + 1) \end{aligned} \quad (2)$$

$$\text{where, } \theta = \tan^{-1} \frac{y_u}{x_u}$$

$$\begin{aligned} x_d &= (X_d - O_x) / F_x, \quad y_d = (Y_d - O_y) / F_y \\ x_u &= (X_u - O_x) / F_x, \quad y_u = (Y_u - O_y) / F_y \end{aligned}$$

2.2. Radial distortion curve

It is supposed that radial distortion curve (i.e. r_u - r_d curve) derived from radial distortion parameters by Caltech Calibration Toolbox has positive a_1 and three inflection points. Furthermore, the curve portion in data supported range, where r_d and r_u are measured by corner detection and predefined calibration pattern geometry, should have positive slope and negative slope changing rate. If (s,c) denotes the slope and slope changing rate of a curve portion respectively, the data supported curve portion can be denoted by (+,-).

Because there are three inflection points, total radial distortion curve has four curve portions with different

shape. Fig. 3 shows the radial distortion curve estimated with Fig. 1 image. It can be observed that the radial distortion curve has four curve portions and the data supported range is curve portion III. Because a_1 is positive, curve portion I and IV should have (+,-) and (+,+) shape respectively. Naturally, curve portion II should have (+,+) shape to smoothly connect neighboring curve portions.

Figure 3. Radial distortion curve. Three inflection points and four curve portions.

Because curve portion IV, which is expected to define r_u - r_d relation in the outer portion of distorted image, has positive slope and positive slope changing rate, it rapidly derives from the ideal r_u - r_d curve depicted in Fig. 3. In wide-angle lens, as the radius of a pixel increases, corresponding image compression ratio drastically increases [6]. That is the reason why wide-angle lens is selected to cover wide FOV. Therefore, although the undistorted portion looks small, it is critical for the use of wide-angle lens.

2.3. Drawback of least square based curve fitting with limited data

It is found that the incorrect estimation of radial distortion parameters is the inevitable result of least square based curve fitting with limited data. As mentioned previously, in practical situation, r_u - r_d data covering whole lens FOV is inaccessible, because plane calibration pattern cannot be extended unlimitedly.

If the 2nd derivative of 5th order polynomial is defined like Eq. (3), then Eq. (1) can be rewritten like Eq. (4) by integrating Eq. (3) twice.

$$r_d'' = k(r_u - \alpha)(r_u - \beta)(r_u - \gamma) \quad (3)$$

$$\begin{bmatrix} b_1 r_u^5 + b_2 r_u^4 + b_3 r_u^3 + b_4 r_u^2 & r_u & 1 \end{bmatrix} \begin{bmatrix} k \\ a_5 \\ a_6 \end{bmatrix} = r_d \quad (4)$$

where, $b_1 = \frac{1}{20}$

$$b_2 = -\frac{1}{12}\alpha - \frac{1}{12}\beta - \frac{1}{12}\gamma$$

$$b_3 = \frac{1}{6}\alpha\beta + \frac{1}{6}(\alpha + \beta)\gamma$$

$$b_4 = -\frac{1}{6}\alpha\beta\gamma$$

One of three inflection points, γ , is supposed to be zero for the sake of smooth shape around the origin. By replacing $\alpha(>0)$ with $\alpha+\delta$ and $\beta(<0)$ with $\beta-\delta$ respectively, we can investigate the effect of inflection points located farther than initially estimated values. With modified α , β and predefined r_u data, matrix X is constructed like Eq. (5-1). With measured r_d data, matrix Y is constructed like Eq. (5-2). And, variable matrix A is defined like Eq. (5-3). Therefore, A can be estimated by pseudo-inverse like Eq. (6).

$$X = \begin{bmatrix} b_1 r_u^5 + b_2 r_u^4 + b_3 r_u^3 + b_4 r_u^2 & r_u & 1 \\ \vdots & \vdots & \vdots \end{bmatrix} \quad (5-1)$$

$$Y = \begin{bmatrix} r_d \\ \vdots \end{bmatrix} \quad (5-2) \quad A = \begin{bmatrix} k \\ a_5 \\ a_6 \end{bmatrix} \quad (5-3)$$

$$\hat{A} = (X^T X)^{-1} X^T Y \quad (6)$$

Figure 4. Squared error of r_d with respect to δ

With estimated $\hat{A}$, all coefficients of Eq. (1) can be determined. By comparing r_d data calculated with estimated $\hat{A}$ and measured r_d data, squared error with respect to δ is evaluated as depicted in Fig. 4. It is noticeable that initial inflection points are optimal in the sense of least squared error. In other words, r_u - r_d data given in the limited range definitely leads to nearby inflection points and consequently incorrect distortion estimation in outer image portion.

3. Refinement of distortion parameters

Because limited r_u - r_d range used for least square based curve fitting is the primary cause of incorrect estimation, by securing wide range of r_u - r_d data, distortion parameters can be refined. Compared with r_u - r_d graph, r_d - r_u graph, i.e. inverse mapping, has suitable characteristics for least square based curve fitting.

Extrapolation by inverse mapping consists of five steps. First, reliable r_u - r_d data is generated by evaluating Eq. (1) with the radial distortion parameters by Caltech Calibration Toolbox from zero to the positive inflection point α . Second, reliable r_d - r_u data can be obtained just by changing the axes: $(r_u, r_d) \rightarrow (r_d, r_u)$. Third, applying least square based curve fitting to the r_d - r_u data estimates the polynomial coefficients of the inverse mapping. Fig. 5 shows the 5th order polynomial graph estimated with the reliable r_d - r_u data, which is depicted by bold line. It is noticeable that the reliable r_d - r_u data has (+,+) shape without positive inflection point. Fourth, evaluating the inverse mapping polynomial in an extended r_d range can generate extended r_d - r_u data. Finally, extended r_u - r_d data can be acquired by changing axes again.

Figure 5. Extended data by inverse mapping

Figure 6. Refined radial distortion curve

Consecutively, refined radial distortion parameters can be estimated by applying least square based curve

fitting to the extended r_u-r_d data. Fig. 6 shows the resultant r_u-r_d graph estimated with the extended r_u-r_d data, which is depicted by bold line. It can be observed that r_u-r_d graph corresponding to the outer portion of distorted image becomes similar to the ideal r_u-r_d curve.

4. Experiment results and Conclusion

Fig. 7 shows the undistorted image of Fig. 1 with refined radial distortion parameters. It is noticeable that the outer portion can be successfully undistorted. Fig. 8 is another resultant image undistorted by the proposed method. ru_theta is the unique tuning parameter adjusting the extend r_d range.

Proposed method can be executed after Caltech Calibration Toolbox to refine the radial distortion parameter of wide-angle lens. Matlab source code is attached in the appendix for convenient use.

Figure 7. Undistorted image of Fig. 1 (size: 4096x3072, $ru_theta=5$)

Figure 8. Another undistorted image. (size: 1280x960, Input image: 640x480 with 120° FOV lens, $ru_theta=2.5$)

5. References

- [1] J. Y. Bouguet, "Camera Calibration Toolbox for Matlab", http://www.vision.caltech.edu/bouguetj/calib_doc/index.html.
- [2] J. Salvi, X. Armanu, and J. Batlle, "A comparative review of camera calibrating methods with accuracy evaluation", *Pattern Recognition* 35 (2002) pp. 1617-1635.
- [3] S. K. Gehrig, "Large-field-of-view stereo for automotive applications", The 6th Workshop on Omnidirectional Vision, Camera Networks and Non-classical cameras (OMNIVIS 2005), Oct. 21, 2005, Beijing, China.
- [4] The 6th Workshop on Omnidirectional Vision, Camera Networks and Non-classical cameras (OMNIVIS 2005), <http://www.eecs.berkeley.edu/~cgeyer/OMNIVIS05/>.
- [5] R. Swaminathan, and S. K. Nayar, "Nonmetric calibration of wide-angle lenses and polycameras", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 22, No. 10, Oct. 2000, pp. 1172-1178.
- [6] K. Cornelis, M. Pollefeys, and L. V. Gool, "Lens distortion recovery for accurate sequential structure and motion recovery", *ECCV 2002, LNCS 2351*, pp. 186-200.

Appendix. RefineRadialDistortion.m

```
ok=[kc(2) 0 kc(1) 0 1 0]; % Caltech Toolbox result
dd_ok=polyder(polyder(ok));
dd_ok_roots=roots(dd_ok);
ThirdInflection=-inf;
for(i=1:3)
    if(imag(dd_ok_roots(i))==0 && real(dd_ok_roots(i))>0
    && dd_ok_roots(i)>ThirdInflection)
        ThirdInflection=dd_ok_roots(i);
    end
end
ru_reliable=ThirdInflection; % Positive inflection point
ru=0:0.001:ru_reliable;
rd=polyval(ok,ru); % Reliable ru-rd data
ik=polyfit(rd,ru,5); % Curve fitting of inverse mapping
ru_required= $ru\_theta$ *ru_reliable; % Extended ru range
problem=ik; problem(6)=problem(6)-ru_required;
rd_roots=roots(problem);
rd_required=rd_roots(1);
for(i=2:5)
    if(imag(rd_roots(i))==0 && real(rd_roots(i))>0 &&
rd_roots(i)<rd_required)
        rd_required=rd_roots(i);
    end
end
rd_ex=0:0.001:rd_required; % Extended rd range
ru_ex=polyval(ik,rd_ex); % Extrapolation
rk=polyfit(ru_ex,rd_ex,5); % Refining parameters
rd_org=polyval(ok,ru_ex);
rd_result=polyval(rk,ru_ex);
figure; plot(ru_ex,rd_result,'b');
axis([0 ru_required 0 1.5*rd_required]);
title('Refined ru-rd curve'); xlabel('ru'); ylabel('rd');
hold on; plot(ru_ex,rd_org,'r');
```