Rethinking Generic Camera Models for Deep Single Image Camera Calibration to Recover Rotation and Fisheye Distortion Supplementary

Nobuhiko Wakai[1], Satoshi Sato[1], Yasunori Ishii[1], and Takayoshi Yamashita[2]

[1] Panasonic Holdings {*lastname.firstname*}@jp.panasonic.com
[2] Chubu University takayoshi@isc.chubu.ac.jp

S1 Proposed camera model

We describe details of our proposed camera model. For this supplementary, we again show our generic camera model as

$$\gamma = f(\eta + k_1\eta^3), \tag{S1}$$

where γ is distortion, f is focal length, η is an incident angle, and k_1 is a distortion coefficient.

S1.1 Incident angle limitations

We limit incident angles to consider the physical size of cameras, that is, outer image circles. Incident angle limitations in our generic camera model in Equation (S1) are shown in

$$\gamma = \begin{cases} f(\eta + k_1\eta^3) & \text{if} \quad \eta < \eta_{\max} \\ \text{invalid} & \text{otherwise} \end{cases}, \tag{S2}$$

where $\eta_{\max}$ is the maximum incident angle and "invalid" means that the camera projection is not determined in Equation (S1). The pixel values are regarded as outer image circles in the invalid case. Note that our loss function does not encounter the invalid case because sampling points on a unit sphere are generated within valid incident angles.

S1.2 Uniqueness of closed-form solution

We demonstrate that our closed-form solution in back-projection has a unique solution. For the back-projection in Equation (S1), converting image coordinates to incident angles, we solve the cubic equation against η using a closed-form expression called Cardano's formula in Section S3. In general, the solution consists

of three complex numbers. However, we always select an optimal real number for η in our generic camera model.

Through theoretical analysis, we found that the solution is categorized into two types: one real number and three real numbers for the solution. This categorization depends on the sign of discriminant D of the cubic equation in Equation (S11) described later. In the case of three real numbers, we select the optimal solution in the middle of the three real numbers because the minimum and maximum numbers are pseudo incident angles. The minimum real number is a negative value that does not satisfy the condition of $\eta > 0$. Based on our theoretical investigation, the maximum real number is out of the incident angle range, that is, $0 < \eta < \eta_{\max}$. Therefore, we can calculate the back-projection without ambiguity.

S1.3 Camera parameter ranges

We must determine camera parameter ranges because our network predicts normalized camera parameters from 0 to 1. Unlike conventional learning-based methods, we found that our camera parameter ranges can be determined. Trigonometric function models for fisheye lenses [6,11] and pinhole camera projection are shown in

$$\gamma = \begin{cases} f \sin\eta & \text{(i)} \\ 2f \sin(\eta/2) & \text{(ii)} \\ f\eta & \text{(iii)} \\ 2f \tan(\eta/2) & \text{(iv)} \\ f \tan(\eta) & \text{(v)} \end{cases}, \tag{S3}$$

where (i) is orthogonal projection, (ii) is equisolid angle projection, (iii) is equidistance projection, (iv) is stereographic projection, and (v) is pinhole camera projection. Note that the projection from (i) to (iv) is the standard fisheye projection of off-the-shelf cameras.

Trigonometric function models in Equation (S3) can be rewritten using the Taylor series expansion based on a third-order polynomial function as

$$\gamma = \begin{cases} f(\eta - 1/6 \cdot \eta^3) & \text{(i)} \\ f(\eta - 1/24 \cdot \eta^3) & \text{(ii)} \\ f\eta & \text{(iii)} \\ f(\eta + 1/12 \cdot \eta^3) & \text{(iv)} \\ f(\eta + 1/3 \cdot \eta^3) & \text{(v)} \end{cases}. \tag{S4}$$

Our proposed generic camera model in Equation (S1) satisfies all projection in Equation (S4) by adjusting k_1. We, thus, define that the range of k_1 is from $-1/6$ to $1/3$. The range of focal length f from 6 to 15 [mm] was determined on the basis of off-the-shelf specifications in Table S1 because our generic camera model has explicit focal length.

Table S1: Details of off-the-shelf fisheye cameras with experimental IDs

ID	Camera body	Camera lens	Projection	Focal length [mm]	Maximum incident angle [deg]
1	Canon EOS 6D	Canon EF8-15mm F4L Fisheye USM	Equisolid angle[1]	8–15	90
2	Canon EOS 6D	Canon EF15mm F2.8 Fisheye	Equisolid angle[1]	15	90
3	Panasonic LUMIX GM1	Panasonic LUMIX G FISHEYE 8mm F3.5	Equisolid angle	14.8[2]	90
4	FLIR BFLY-U3-23S6C	FIT FI-40	Orthogonal	13.9[2]	84
5	FLIR FL3-U3-88S2	FUJIFILM FE185C057HA-1	Equidistance	6.1[2]	92.5
6	KanDao QooCam8K	Bulit-in	Stereographic[1]	6.0[1,2]	90[1]

[1] Our estimation
[2] Conversion focal length using a full-size image sensor

Table S2: Comparison of mean SSIM on the test set of trigonometric function models

Method	StreetLearn					SP360				
	Stereo-graphic	Equi-distance	Equisolid angle	Ortho-gonal	All	Stereo-graphic	Equi-distance	Equisolid angle	Ortho-gonal	All
Alemán-Flores [1]	0.308	0.282	0.267	0.208	0.266	0.356	0.322	0.302	0.223	0.301
Santana-Cedrés [12]	0.341	0.304	0.285	0.224	0.288	0.387	0.343	0.322	0.253	0.326
Liao [8]	0.357	0.365	0.372	0.406	0.375	0.400	0.406	0.412	0.447	0.416
Yin [17]	0.344	0.351	0.357	0.396	0.362	0.385	0.390	0.396	0.434	0.401
Chao [5]	0.396	0.384	0.383	0.406	0.392	0.436	0.427	0.427	0.454	0.436
Bogdan [3]	0.356	0.357	0.359	0.366	0.359	0.487	0.462	0.454	0.441	0.461
Li (GeoNetS-$\mathcal{B}$) [7]	0.439	0.416	0.411	0.423	0.422	0.481	0.459	0.453	0.469	0.466
López-Antequera [9]	0.494	0.462	0.450	0.432	0.460	0.458	0.438	0.434	0.458	0.447
Wakai [14]	0.658	0.688	0.655	0.501	0.626	0.695	0.690	0.650	0.537	0.643
Ours w/o HNGBL[1]	0.780	0.831	0.813	**0.690**	0.779	0.773	0.839	0.828	**0.719**	0.790
Ours	**0.791**	**0.857**	**0.838**	0.686	**0.793**	**0.784**	**0.856**	**0.841**	**0.719**	**0.800**

[1] "Ours w/o HNGBL" refers to replacing HNGBL with non-grid bearing loss [14]

S1.4 Evaluating our camera model

We describe details of evaluating our camera model in Table 1 (main paper). Equation (S1) shows the relation between an incident angle η and distortion γ representing the distance from the principal point at an image sensor. To convert image-sensor coordinates of γ to pixel coordinates, we assumed a full-size image sensor and 224-pixel image height. This image height corresponded to the input image height for our network. This assumption was used for the readability that we can easily understand pixel coordinates rather than image-sensor coordinates. Note that the image-sensor size and the image height are scale factors.

S1.5 Camera parameter distribution for train set

To generate our train images, we used random distribution based on the procedure [14] for rotation angles and aspect ratio, and we determined the range of $\eta_{\max}$ as follows. The maximum incident angle tends to be 90° in off-the-shelf fisheye cameras because of the physical size of cameras. Considering this tendency,

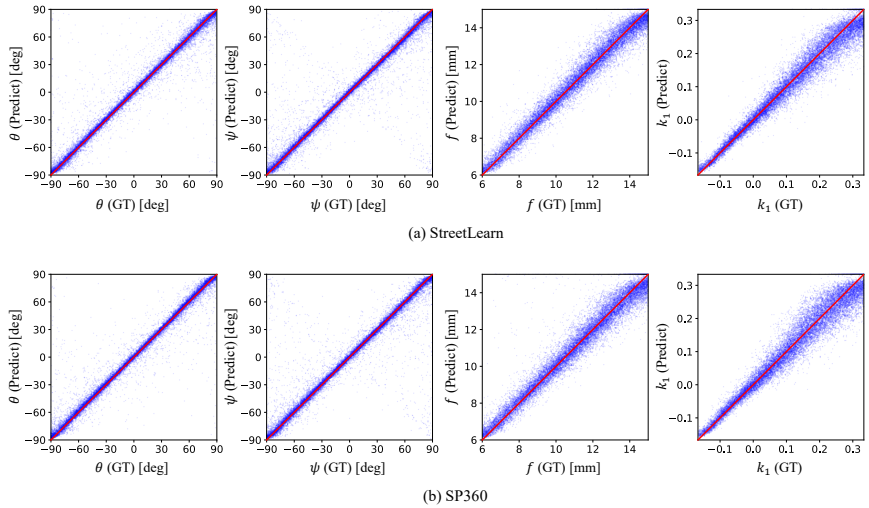


Fig. S1: Error distribution of our method on the test set for our generic camera model using the StreetLearn [10] and SP360 [4] datasets shown in (a) and (b), respectively. The horizontal and vertical axes show ground-truth (GT) and predicted camera parameters, respectively. The diagonal red lines indicate perfect prediction, and each blue point indicates a predicted result.

we used 90° for the center of uniform distribution of $\eta_{\max}$. As shown in Table S1, the ID 4 camera has the most difference of $\eta_{\max}$ from 90°, and this difference is 6°. We can determine the range of $\eta_{\max}$ with 84–96°; that is, the lower bound is $84° = 90° - 6°$ and the upper bound is $96° = 90° + 6°$.

S2 Experiments

To validate the adaptiveness of our method to various types of fisheye cameras, we demonstrate supplemental results.

S2.1 Comparison using SSIM

To demonstrate validity and effectiveness in images, we use the structural similarity (SSIM) [15] for intrinsic parameters; that is, extrinsic parameters are arbitrary. Table S2 shows the performance on the trigonometric function models in the StreetLearn [10] and SP360 [4] datasets. Similar to the result of the peak signal-to-noise ratio (PSNR) in the main paper, our method had the highest SSIM in all cases. This suggests that our generic camera model can behave like a trigonometric function model.

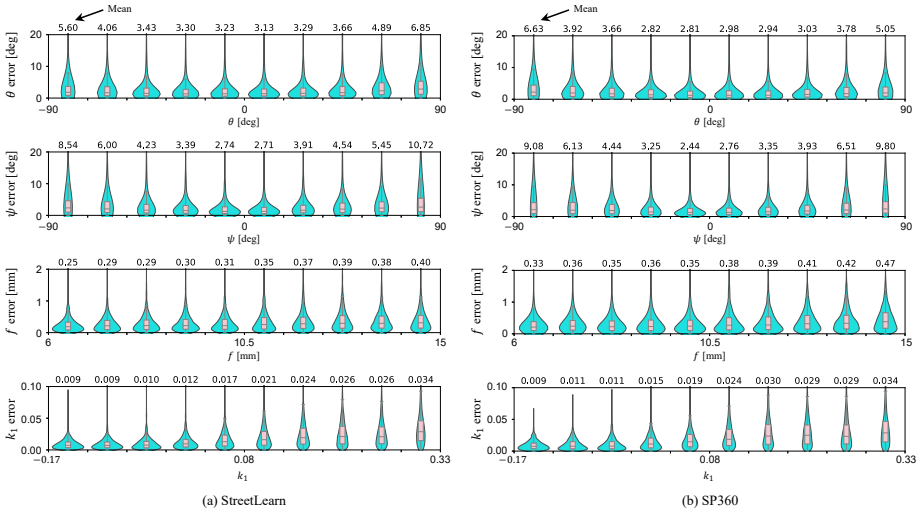


Fig. S2: Error frequency distribution of our method on the test set for our generic camera model using the StreetLearn [10] and SP360 [4] datasets shown in (a) and (b), respectively. This frequency distribution is illustrated using box plots and violin plots. The horizontal and vertical axes show camera parameter ranges and the mean absolute errors of camera parameters, respectively. The mean values are indicated at the top of the plots. Each camera parameter range is divided into 10 classes at equal intervals.

S2.2 Error distribution of camera parameters

To evaluate error distribution, we analyzed the results of predicted camera parameters. Figure S1 shows the error distribution between ground-truth and predicted camera parameters using our method. Most of the predicted camera parameters are close to the ground-truth values; that is, points of the predicted values are close to the perfect prediction indicating diagonal lines in Figure S1.

Moreover, we examined detailed error distribution in that we divided the range of camera parameters at equal intervals. For example, the divided 10 ranges of tilt angles θ were $[-90°, -72°]$, $[-72°, -54°]$, $\ldots$, $[72°, 90°]$. We visualized the error distribution of each divided range using box plots and violin plots[1] in Figure S2. The box plots show that all the boxes had slight errors, although errors increased at the edge of the ranges, for example, $[-90°, -72°]$ in tilt angles θ. Furthermore, each violin plot shows that the error distribution is a single peak close to the median. This single peak suggests that our network was well optimized because optimization failure may cause multiple peaks. Overall, our predicted results had slight errors throughout the camera parameter ranges.

[1] A violin plot is one of the distribution plots similar to a box plot. Unlike box plots, the violin plot can indicate the probability density of the distribution using shapes like a violin instead of a box. We can easily distinguish a single peak and multiple peaks on the basis of this probability density.

Table S3: Feature summarization of our method and conventional methods including additional methods

Method	DL[1]	Rot[1]	Dist[1]	>180° FOV[1]	Projection	Network
Alemán-Flores [1]			✓		Perspective	–
Santana-Cedrés [12]			✓		Perspective	–
Liao [8]	✓		✓		Perspective	Regressor
Yin [17]	✓		✓	✓	Generic camera [6]	Regressor
Chao [5]	✓		✓	–	–	Generator (GAN)
Bogdan [3]	✓		✓	✓	Unified spherical model [2]	Regressor
Yang [16]	✓		✓	–	–	Generator
Li (GeoNetS-$\mathcal{B}$) [7]	✓		✓	–	–	Generator
Li (GeoNetS-$\mathcal{R}, \mathcal{P}$) [7]	✓	✓		–	–	Generator
López-Antequera [9]	✓	✓	✓		Perspective	Regressor
Wakai [14]	✓	✓	✓	✓	Equisolid angle	Regressor
Ours	✓	✓	✓	✓	Proposed generic camera	Regressor

[1] DL denotes learning-based method; Rot denotes rotation; Dist denotes distortion; ">180° FOV" denotes supporting over 180° FOV

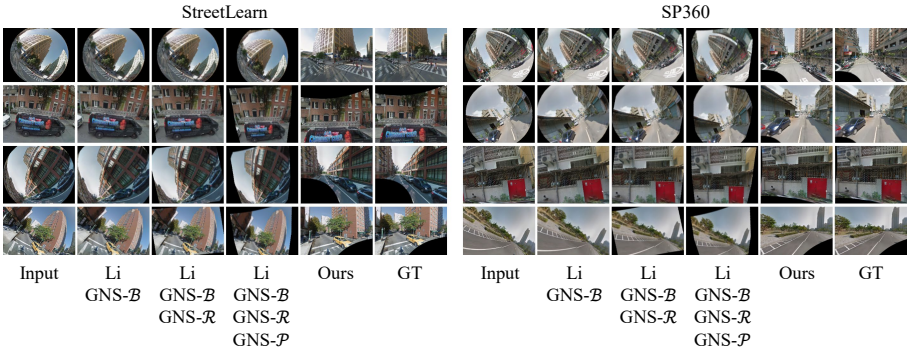


Fig. S3: Qualitative results of fully recovering rotation and fisheye distortion for our generic camera model shown in the input image, results of Li (i), Li (ii), Li (iii), our method, and the ground-truth image from left to right for each image. The details of Li's methods [7] are as follows: Li (i) is results using GNS-$\mathcal{B}$; Li (ii) is results using GNS-$\mathcal{B}$ and GNS-$\mathcal{R}$; Li (iii) is results using GNS-$\mathcal{B}$, GNS-$\mathcal{R}$, and GNS-$\mathcal{P}$. Note that we employed the transformation in the order of attached networks from top to bottom.

S2.3 Comparison using cascaded transformation

To validate the effectiveness of full recovery, we conducted cascaded transformation using Li's methods [7]. Table S3 shows the feature summarization of our method and conventional methods including additional methods mentioned in this supplementary. Li *et al.* [7] proposed various image transformation networks for rotation and distortion. We found that Li's methods [7] can obtain fully recovered images when we employ image transformation as follows. First, we removed fisheye distortion using GeoNetS-$\mathcal{B}$ (GNS-$\mathcal{B}$), which is the single-model distortion network to remove barrel distortion. Second, we rotated the undistorted image using GeoNetS-$\mathcal{R}$ (GNS-$\mathcal{R}$), which is the single-model distortion network to remove image rotation. Finally, we transformed the rotated image to

Table S4: Additional comparison of mean PSNR and SSIM on the test set for our generic camera model

Method	StreetLearn						SP360					
	PSNR ↑			SSIM ↑			PSNR ↑			SSIM ↑		
	Diag[1]	Circ[1]	All	Diag	Circ	All	Diag	Circ	All	Diag	Circ	All
Yang [16] w/o p.p.[2]	12.98	10.92	11.96	0.287	0.239	0.263	12.09	10.04	11.08	0.319	0.260	0.290
Yang [16] w/ p.p.[3]	22.12	19.50	20.82	0.599	0.497	0.548	21.22	18.93	20.09	0.617	0.531	0.574
Ours	**28.39**	**29.63**	**29.01**	**0.828**	**0.847**	**0.838**	**27.19**	**29.03**	**28.10**	**0.819**	**0.852**	**0.835**

[1] Diag denotes evaluation using only diagonal fisheye images; Circ denotes evaluation using only circumferential fisheye images
[2] Yang [16] without the postprocess of image scaling
[3] Yang [16] with the postprocess of image scaling
[4] "Ours w/o HNGBL" refers to replacing HNGBL with non-grid bearing loss [14]

Table S5: Additional comparison of mean PSNR on the test set for the trigonometric function models

Method	StreetLearn					SP360				
	Stereo-graphic	Equi-distance	Equisolid angle	Ortho-gonal	All	Stereo-graphic	Equi-distance	Equisolid angle	Ortho-gonal	All
Yang [16] w/o p.p.[1]	12.10	11.02	10.48	8.86	10.61	11.27	10.17	9.57	7.75	9.69
Yang [16] w/ p.p.[2]	20.96	20.08	19.63	17.54	19.55	20.15	19.44	19.06	17.12	18.94
Ours w/o HNGBL[3]	26.49	29.08	28.56	**23.97**	27.02	25.35	28.53	28.26	23.85	26.50
Ours	**26.84**	**30.10**	**29.69**	23.70	**27.58**	**25.74**	**29.28**	**28.95**	**23.93**	**26.98**

[1] Yang [16] without the postprocess of image scaling
[2] Yang [16] with the postprocess of image scaling
[3] "Ours w/o HNGBL" refers to replacing HNGBL with non-grid bearing loss [14]

the fully recovered image using GeoNetS-$\mathcal{P}$ (GNS-$\mathcal{P}$), which is the single-model distortion network for the projective transformation.

Figure S3 shows the qualitative results of Li's methods [7] with the cascaded transformation. Note that we trained Li's methods using the StreetLearn [10] or SP360 [4] datasets. In diagonal fisheye images, these methods can recover rotation and distortion. However, the results for the methods had distortion and/or rotation errors. Our method outperformed Li's methods [7] on both diagonal and circumferential fisheye images.

S2.4 Comparison using Yang's method

To validate the performance of undistortion, we compared an additional method. Yang *et al.* [16] proposed a learning-based method that uses a generator to remove fisheye distortion. We trained Yang's method using StreetLearn [10] or SP360 [4] for our evaluation. Although we used Yang's implementation, which is publicly available, the undistortion results had low PSNR on the test set for our generic camera model. We found that Yang's implementation generated pairs of distorted and undistorted images with different image scales between the pair of images. In this generation, undistorted images were transformed using resizing after cropping. Thus, the image scale of the distorted image did not correspond to the undistorted image. Note that the magnitude of the scale difference depended

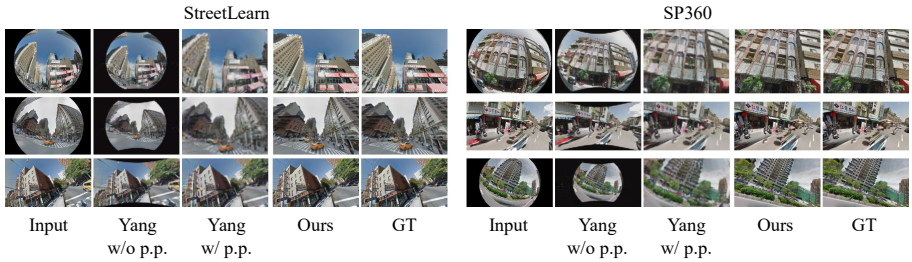


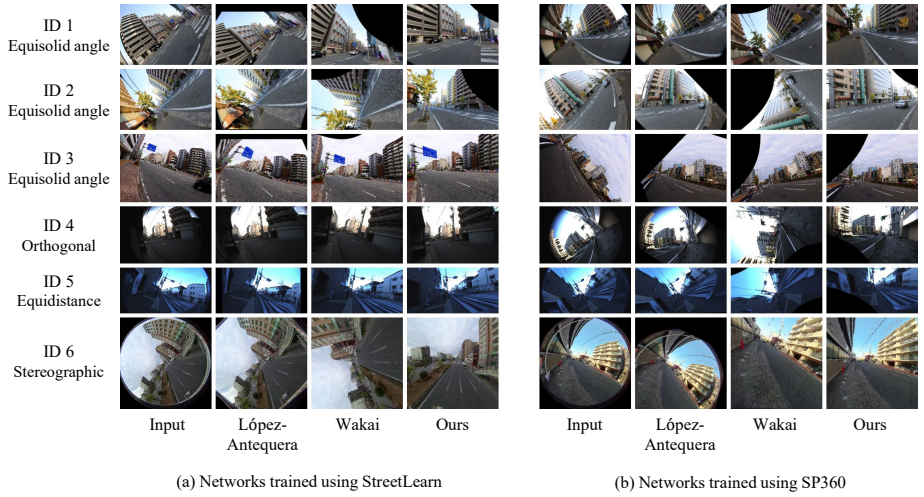
Fig. S4: Qualitative results of fisheye undistortion without recovering rotation for our generic camera model shown in the input image, results of Yang's method [16] without the postprocess, Yang's method [16] with the postprocess, our method, and the ground-truth image from left to right for each image.

on individual images because the distortion coefficients determined the cropping areas.

To evaluate Yang's method [16], we scaled the predicted images to maximize PSNR by searching the image scale from 1 to 3 with an interval of 0.1. Note that we scaled images and then center-cropped them to obtain the original image resolution. We regarded the postprocessed results as an auxiliary evaluation because the postprocess degraded the image quality due to scaling. Table S4 shows the results of Yang's method [16] with or without the postprocess of image scaling on the test set for our generic camera model, and we omitted other results shown in Table 6 (main paper). Similarly, Table S5 shows the results of Yang's method [16] with or without the postprocess of image scaling on the test set for the trigonometric camera models. We omitted other results shown in Table 7 (main paper). The postprocess increased PSNR and SSIM on the test of our generic camera model and the trigonometric function models. Figure S4 shows the qualitative results of Yang's method [16] with or without the postprocess. The result of Yang's method [16] had image distortion compared with our method. Considering the difference between our method and Yang's method [16] in PSNR, our method seems to outperform Yang's method [16]. As described above, it is impossible for us to evaluate the method with complete fairness.

S2.5 Qualitative evaluation using off-the-shelf cameras

We validated the performance of undistortion and full recovery for off-the-shelf cameras to describe the image quality after calibration with supplementary results. Figure S5 shows qualitative results of fully recovering rotation and fisheye distortion. Learning-based calibration methods were trained using the StreetLearn [10] or SP360 [4] datasets. Similar to Figure 4 (main paper), the results for López-Antequera's method [9] had rotation and distortion errors. Although Wakai's method [14] removed distortion, the results had rotation errors.



(a) Networks trained using StreetLearn (b) Networks trained using SP360

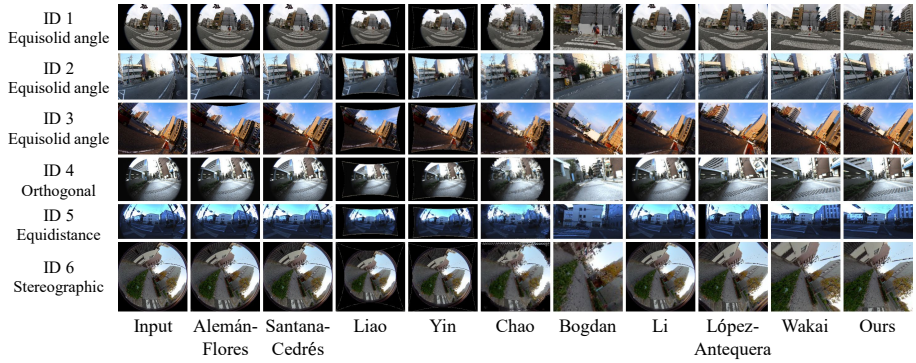
Fig. S5: Qualitative results of fully recovering rotation and fisheye distortion for the off-the-shelf cameras shown in the input image, results of the compared methods (López-Antequera [9] and Wakai [14]), and our method from left to right for each image. The IDs correspond to IDs in Table S1, and the projection names are attached to the IDs from specifications (ID: 3–5) and our estimation (ID: 1, 2, and 6). Qualitative results of the methods trained using StreetLearn [10] and SP360 [4] as shown in (a) and (b), respectively.

Our fully recovered images demonstrated the effectiveness of off-the-shelf fisheye cameras with various types of projection.

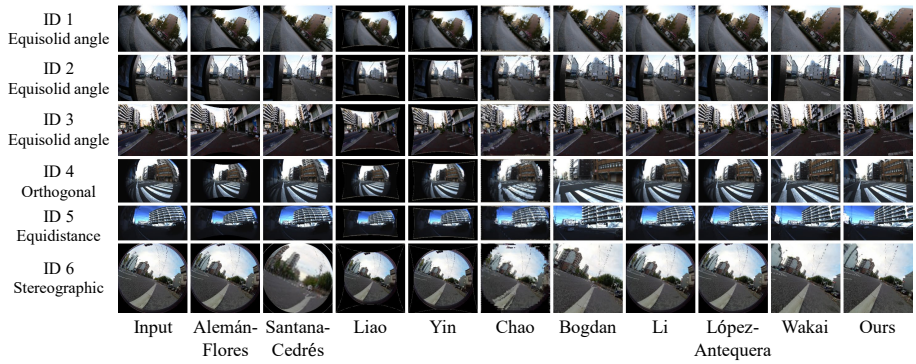
We also evaluated the performance of undistortion using off-the-shelf cameras. Figure S6 shows qualitative results of undistortion without recovering rotation. Our method and Wakai's method [14] removed fisheye distortion. By contrast, the results had distortion errors in Alemán-Flores's [1], Santana-Cedrés's [12], Liao's [8], Yin's [17], Chao's [5], Li's (GeoNetS-$\mathcal{B}$) [7], and López-Antequera's [9] method. In particular, circumferential fisheye images led to large distortion errors in these conventional methods. Overall, our method outperformed all the conventional methods considering both rotation and distortion errors.

S2.6 Sample off-the-shelf images

We attached sample high-resolution images captured using our off-the-shelf cameras. These images were used for Figure 4 (main paper). For our experiment, we resized the off-the-shelf images with 224-pixel image height (H_{img}) and image width ($W_{img} = H_{img} \cdot A$), where A is the image aspect ratio.



(a) Networks trained using StreetLearn



(b) Networks trained using SP360

Fig. S6: Qualitative results of fisheye undistortion without recovering rotation for the off-the-shelf cameras shown in the input image, results of the compared methods (Alemán-Flores [1], Santana-Cedrés [12], Liao [8], Yin [17], Chao [5], Bogdan [3], Li (GeoNetS-$\mathcal{B}$) [7], López-Antequera [9], and Wakai [14]), and our method from left to right for each image. The IDs correspond to IDs in Table S1, and the projection names are attached to the IDs from specifications (ID: 3–5) and our estimation (ID: 1, 2, and 6). Qualitative results of the methods trained using StreetLearn [10] and SP360 [4] as shown in (a) and (b), respectively. Because of geometric-based calibration, Alemán-Flores's [1] and Santana-Cedrés's methods [12] need not train.

S3 Appendix of Cardano's formula

We briefly describe Cardano's formula to solve back-projection. Cardano's formula addresses a general cubic equation against x as

$$a_3x^3 + a_2x^2 + a_1x + a_0 = 0, \quad a_3 \neq 0, \tag{S5}$$

where a_3, a_2, a_1, and a_0 are coefficients of the cubic equation. We rewrite Equation (S5) by dividing a_3 as

$$\begin{aligned} x^3 + \frac{a_2}{a_3}x^2 + \frac{a_1}{a_3}x + \frac{a_0}{a_3} &= 0, \\ x^3 + b_2x^2 + b_1x + b_0 &= 0, \end{aligned} \tag{S6}$$

where b_2, b_1, and b_0 are coefficients of the cubic equation. To remove the second-order term, we also rewrite Equation (S6) as follows:

$$\begin{aligned} y^3 + py + q &= 0, \\ y &= x + \frac{b_2}{3}, \end{aligned} \tag{S7}$$

where

$$\begin{aligned} p &= b_1 - \frac{1}{3}b_2^2, \\ q &= b_0 - \frac{1}{3}b_1b_2 + \frac{2}{27}b_2^3. \end{aligned}$$

Here, we use $y = u + v$ to Equation (S7) as

$$u^3 + v^3 + q + (3uv + p)(u + v) = 0. \tag{S8}$$

Equation (S8) can be converted to simultaneous equations as follows:

$$\begin{cases} u^3 + v^3 + q = 0 \\ (3uv + p)(u + v) = 0 \end{cases}. \tag{S9}$$

This conversion is acceptable because the fundamental theorem of algebra [13] ensures that cubic equations have exact three solutions including complex numbers. Therefore, we only find the three solutions using any conversion. By removing v in Equation (S9), we can obtain an equation using $(u^3)^2$ and u^3 as

$$(u^3)^2 + qu^3 - \left(\frac{p}{3}\right)^3 = 0. \tag{S10}$$

We solve Equation (S10) as a quadratic equation against u^3, given by

$$\begin{aligned} u^3 &= -\frac{q}{2} \pm \sqrt{\left(\frac{q}{2}\right)^2 + \left(\frac{p}{3}\right)^3} \\ &= -\frac{q}{2} \pm \sqrt{D}, \end{aligned} \tag{S11}$$

where D is the discriminant of this cubic equation. We obtain three real numbers and one real number of the solution on the condition that D is positive and negative, respectively. Note that we do not encounter $D = 0$ in our generic camera model without the second-order term, although $D = 0$ leads to multiple roots of the solution. Therefore, we obtain u using the cube root of 1 as

$$u = \omega \sqrt[3]{-\frac{q}{2} \pm \sqrt{D}}, \tag{S12}$$

where $\omega = 1, (-1 \pm \sqrt{3}\ i)\ /\ 2$. Similar to u, we can calculate v as

$$v = \omega^2 \sqrt[3]{-\frac{q}{2} \mp \sqrt{D}}. \tag{S13}$$

Finally, the solution is calculated using $y = u + v$ and $y = x + b_2/3$ as

$$x = -\frac{b_2}{3} + \omega \sqrt[3]{-\frac{q}{2} \pm \sqrt{D}} + \omega^2 \sqrt[3]{-\frac{q}{2} \mp \sqrt{D}}. \tag{S14}$$

In our generic camera model, the equation of back-projection is written as follows:

$$k_1 \eta^3 + \eta - \frac{\gamma}{f} = 0, \quad f > 0. \tag{S15}$$

As described above, we can solve Equation (S15) using Cardano's formula. We need not use Cardano's formula if k_1 is 0, that is, a linear equation. Note that, for back-propagation, we can calculate Equation (S14) through an argument of the complex plane instead of the imaginary unit i.

References

1. Alemán-Flores, M., Alvarez, L., Gomez, L., Santana-Cedrés, D.: Automatic lens distortion correction using one-parameter division models. Image Processing On Line (IPOL) **4**, 327–343 (2014). https://doi.org/10.5201/ipol.2014.106
2. Barreto, J.: A unifying geometric representation for central projection systems. Computer Vision and Image Understanding (CVIU) **103**(3), 208–217 (2006). https://doi.org/10.1016/j.cviu.2006.06.003
3. Bogdan, O., Eckstein, V., Rameau, F., Bazin, J.C.: DeepCalib: A deep learning approach for automatic intrinsic calibration of wide field-of-view cameras. In: Proceedings of SIGGRAPH European Conference on Visual Media Production (CVMP) (2018). https://doi.org/10.1145/3278471.3278479
4. Chang, S., Chiu, C., Chang, C., Chen, K., Yao, C., Lee, R., Chu, H.: Generating 360 outdoor panorama dataset with reliable sun position estimation. In: Proceedings of SIGGRAPH Asia. pp. 1–2 (2018). https://doi.org/10.1145/3283289.3283348
5. Chao, C., Hsu, P., Lee, H., Wang, Y.: Self-supervised deep learning for fisheye image rectification. In: Proceedings of IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP). pp. 2248–2252 (2020). https://doi.org/10.1109/ICASSP40776.2020.9054191

6. Kannala, J., Brandt, S.S.: A generic camera model and calibration method for conventional, wide-angle, and fish-eye lenses. IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI) **28**(8), 1335–1340 (2006). https://doi.org/10.1109/TPAMI.2006.153
7. Li, X., Zhang, B., Sander, P.V., Liao, J.: Blind geometric distortion correction on images through deep learning. In: Proceedings of IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). pp. 4855–4864 (2019). https://doi.org/10.1109/CVPR.2019.00499
8. Liao, K., Lin, C., Zhao, Y.: A deep ordinal distortion estimation approach for distortion rectification. IEEE Transactions on Image Processing (TIP) **30**, 3362–3375 (2021). https://doi.org/10.1109/TIP.2021.3061283
9. López-Antequera, M., Marí, R., Gargallo, P., Kuang, Y., Gonzalez-Jimenez, J., Haro, G.: Deep single image camera calibration with radial distortion. In: Proceedings of IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). pp. 11809–11817 (2019). https://doi.org/10.1109/CVPR.2019.01209
10. Mirowski, P., Banki-Horvath, A., Anderson, K., Teplyashin, D., Hermann, K.M., Malinowski, M., Grimes, M.K., Simonyan, K., Kavukcuoglu, K., Zisserman, A., Hadsell, R.: The StreetLearn environment and dataset. arXiv preprint arXiv:1903.01292 (2019). https://doi.org/10.48550/arXiv.1903.01292
11. Ray, F.S.: Applied photographic optics. Focal Press Oxford (1994)
12. Santana-Cedrés, D., Gomez, L., Alemán-Flores, M., Salgado, A., Esclarín, J., Mazorra, L., Alvarez, L.: An iterative optimization algorithm for lens distortion correction using two-parameter models. Image Processing On Line (IPOL) **6**, 326–364 (2016). https://doi.org/10.5201/ipol.2016.130
13. Velleman, D.: The fundamental theorem of algebra: A visual approach. The Mathematical Intelligencer **37**, 12–21 (2015). https://doi.org/10.1007/s00283-015-9572-7
14. Wakai, N., Yamashita, T.: Deep single fisheye image camera calibration for over 180-degree projection of field of view. In: Proceedings of IEEE/CVF International Conference on Computer Vision Workshops (ICCVW). pp. 1174–1183 (2021). https://doi.org/10.1109/ICCVW54120.2021.00137
15. Wang, Z., Bovik, A.C.: Image quality assessment: From error visibility to structural similarity. IEEE Transactions on Image Processing (TIP) **13**(4), 600–612 (2004). https://doi.org/10.1109/TIP.2003.819861
16. Yang, S., Lin, C., Liao, K., Zhang, C., Zhao, Y.: Progressively complementary network for fisheye image rectification using appearance flow. In: Proceedings of IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). pp. 6344–6353 (2021). https://doi.org/10.1109/CVPR46437.2021.00628
17. Yin, X., Wang, X., Yu, J., Zhang, M., Fua, P., Tao, D.: FishEyeRecNet: A multi-context collaborative deep network for fisheye image rectification. In: Proceedings of European Conference on Computer Vision (ECCV). pp. 475–490 (2018). https://doi.org/10.48550/arXiv.1804.04784