Table 5. Distribution of the number of unique axes (%)

Dataset[1]	Number of unique axes							
	0	1	2	3	4	5	6	7
Train	0.0	1.3	13.5	25.7	24.8	18.8	10.9	5.1
Test	0.0	1.4	12.8	25.7	25.6	19.6	10.2	4.6

[1] SL-MH, SL-PB, SP360, and HoliCity all have the same distribution of the number of unique axes shown in this table

Ignoring back labels. After removing label ambiguity, we ignored back labels because the training and test sets had only 0.1% and 0.3% back labels, respectively. Therefore, the VP estimator detected 13 points, that is, the five VPs (front, left, right, top, and bottom) and eight ADPs in Table 2. If all VP/ADPs are successfully detected, our method can estimate a unique rotation for over 98% images with two or more unique axes from the VP/ADPs in Table 5.

4.3. Parameter settings

Following [53], we fixed the image sensor height to 24 mm, $d_u = d_v$, used the principal points (c_u, c_v) as the image center, and the translation vectors $\mathbf{t}$ as the zero vectors. Therefore, in our method, we focused on the estimation of five camera parameters, that is, focal length f, distortion coefficient k_1, pan angle ϕ, tilt angle θ, and roll angle ψ in a Manhattan world. We independently trained the VP estimator and distortion estimator using a mini-batch size of 32. We optimized the VP estimator, which was pretrained on ImageNet [42], using the Adam optimizer [17] and Random Erasing augmentation [62] with $(p, s_l, s_h, r_1, r_2) = (0.5, 0.02, 0.33, 0.3, 3.3)$. The initial learning rate was set to 1×10^{-4} and was multiplied by 0.1 at the 100th epoch. We also trained the distortion estimator pretrained on Wakai *et al.*'s original network [53] using the RAdam optimizer [31] with a learning rate of 1×10^{-5}.

4.4. Experimental results

We implemented the comparison methods according to the corresponding papers but trained them on SL-MH, SL-PB, SP360, and HoliCity.

4.4.1 Vanishing point estimation

To demonstrate the validity and effectiveness of the proposed VP estimator, we used pose estimation metrics and the distance error between the detected and ground-truth VP/ADPs. Following [7], we used the standard keypoint metrics of the average precision (AP) and average recall (AR) [47] with a 5.6-pixel sk_i object keypoint similarity [36] as well as the percentage of correct keypoints (PCK) [2] with a 5.6-pixel distance threshold, which corresponds to $1/10$th of the heatmap height.

Overall, the VP estimator detected the VP/ADPs, although the performance in the cross-domain evaluation decreased, as Table 6 reveals. The AP, AR, and PCK results suggest that VP/ADP estimation is an easier task than human pose estimation because the VP/ADPs have the implicit constraints of the geometric coordinates shown in Figure 3. Table 6 also reveals that ADP detection is more difficult than VP detection because VPs generally have specific appearances at infinity. In terms of distance errors, the VP estimator addressed the various directions of VP/ADPs. In addition, considering the qualitative results in Figure 5, the VP estimator stably detected VP/ADPs in the entire image. Therefore, the VP estimator can precisely detect VP/ADPs from a fisheye image.

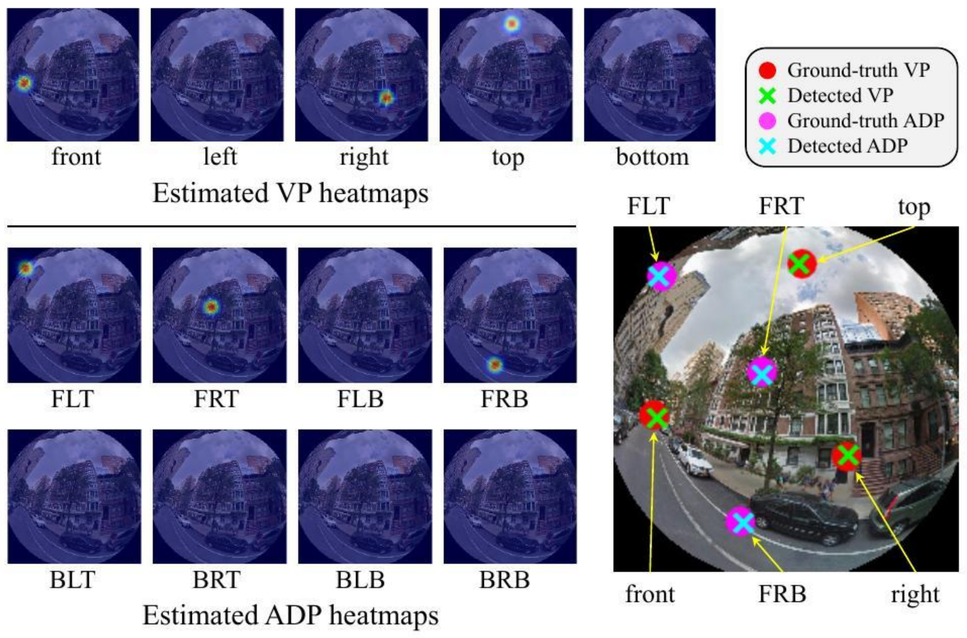

Figure 5. Qualitative results of VP/ADP detection using the proposed VP estimator on the SL-MH test set. The VP estimator estimated five VP and eight ADP heatmaps for each VP/ADP.

4.4.2 Parameter and reprojection errors

To validate the accuracy of the camera parameters, we compared our method with conventional methods that estimate both rotation and distortion. Following [53], we evaluated the mean absolute error and reprojection error (REPE). Our method achieved the lowest mean absolute angle error and REPE of the methods listed in Table 7. The pan-angle errors in our method using HRNet-W32 for the VP estimator are substantially smaller than those of Lochman *et al.*'s method [32] by 20.16° on the SL-MH test set. Our method achieved 3.15° and 3.00° errors for the tilt and roll angles, respectively, outperforming the other methods. We evaluated our method using HRNet-W48 (a larger backbone) and obtained a slight RMSE improvement of 0.16 pixels with respect to the RMSE obtained using HRNet-W32.

The Pritts *et al.*'s [41] and Lochman *et al.*'s [32] methods could not perform calibration for some images because of a lack of arcs. In particular, Lochman *et al.*'s method [32] had a 59.1% executable rate, that is, the number of successful executions divided by the number of all images. Note that we calculated errors using only these successful executions. By contrast, our learning-based method can ad-