

# A Reliable Online Method for Joint Estimation of Focal Length and Camera Rotation: Supplemental Material

Anonymous ECCV submission

Paper ID 1737

## 1 Evaluating deviation measures

In Section 5.1 of the main paper we reported the performance of five candidate deviation measures (Fig. 2 in the paper) on the PanoContext- $f\mathbf{R}$  training dataset, including modeling of the likelihood distributions. Tables 1 and 1 below shows performance on both the YorkUrbanDB and PanoContex- $f\mathbf{R}$  training sets, with and without probabilistic modeling.

Results are consistent for the two datasets: Deviation measure **b** yields the lowest errors for both focal length and camera rotation. We found that for each of the deviation measures, errors were far lower if the measures were modeled probabilistically, rather than just minimizing the sum of log deviations. Tables 1 and 2 show the performance of each deviation measure, with and without probabilistic modeling, on the York Urban DB and PanoContex- $f\mathbf{R}$  datasets, respectively.

Table 1: Evaluation of deviation measures on the YorkUrbanDB training set. Numbers are mean $\pm$ standard error.

Deviation measure	Frame error (deg)	Focal length MAE (%)
<b>Without probabilistic modeling</b>		
a	24.1 $\pm$ 1.52	40.9 $\pm$ 0.93
b	3.19 $\pm$ 0.29	11.5 $\pm$ 1.43
c	2.94 $\pm$ 0.33	11.4 $\pm$ 1.18
d	6.13 $\pm$ 1.13	14.4 $\pm$ 1.61
e	4.94 $\pm$ 0.99	13.4 $\pm$ 1.37
<b>With probabilistic modeling</b>		
a	8.13 $\pm$ 1.20	19.5 $\pm$ 2.39
<b>b (<math>f\mathbf{R}</math>)</b>	<b>1.54<math>\pm</math> 0.16</b>	<b>4.6<math>\pm</math>0.78</b>
c	1.62 $\pm$ 0.17	5.4 $\pm$ 1.07
d	2.53 $\pm$ 0.59	6.6 $\pm$ 1.29
e	2.40 $\pm$ 0.70	8.1 $\pm$ 1.18

We also noted in this section that deviation measure **e** based on the angle of deviation between the interpretation plane normal for a line segment and the

Table 2: Evaluation of deviation measures on the PanoContext- $f\mathbf{R}$  training set. Numbers are mean $\pm$ standard error.

Dev. measure	Roll MAE (deg)	Tilt MAE (deg)	Focal length MAE (%)
<b>Without probabilistic modeling</b>			
a	5.28 $\pm$ 0.053	16.4 $\pm$ 0.17	42.9 $\pm$ 0.43
b	1.72 $\pm$ 0.017	5.91 $\pm$ 0.058	36.3 $\pm$ 0.36
c	2.48 $\pm$ 0.025	8.94 $\pm$ 0.090	44.2 $\pm$ 0.44
d	3.88 $\pm$ 0.038	9.95 $\pm$ 0.097	34.0 $\pm$ 0.33
e	1.77 $\pm$ 0.017	5.64 $\pm$ 0.055	48.7 $\pm$ 0.48
<b>With probabilistic modeling</b>			
a	4.53 $\pm$ 0.045	14.2 $\pm$ 0.14	35.6 $\pm$ 0.35
<b>b (<math>f\mathbf{R}</math>)</b>	<b>0.79<math>\pm</math>0.008</b>	<b>1.65<math>\pm</math>0.016</b>	<b>8.8<math>\pm</math>0.09</b>
c	0.90 $\pm$ 0.009	2.26 $\pm$ 0.022	14.1 $\pm$ 0.14
d	0.81 $\pm$ 0.008	1.67 $\pm$ 0.015	10.1 $\pm$ 0.08
e	0.89 $\pm$ 0.009	2.13 $\pm$ 0.021	19.5 $\pm$ 0.19

plane orthogonal to the vanishing direction (Fig. 2 in paper), yields on average reasonable estimates of camera rotation but relatively poor estimates of focal length. We argued there that this failure likely arises due to a degeneracy: As focal length tends to infinity, interpretation plane normals collapse to a great circle parallel to the image plane, so that a vanishing point at the principal point generates a maximal likelihood. This predicts that a system based on this deviation measure will be biased to estimate higher focal lengths (smaller FOVs).

Fig. 1 confirms this prediction. Note that each peak is biased to smaller FOV (higher focal length) relative to ground truth, and there is a spike in FOV at the lower bound of our search region (50 deg), suggesting that, for a significant number of images, the system is headed for a degenerate solution with infinite focal length.

## 2 PanoContext- $uf\mathbf{R}$ dataset

We designed the PanoContext- $f\mathbf{R}$  dataset to sample five ground truth FOVs in order to clearly visualize whether algorithms are able to estimate focal length while simultaneously estimating camera rotation. In order to verify that this discrete sampling did not somehow distort our results, we have also created a PanoContext- $uf\mathbf{R}$  dataset of 5,295 images that samples rotation parameters as for PanoContext- $f\mathbf{R}$ , but also samples FOV randomly and uniformly over the continuous interval from 60 to 120 deg.

Fig. 2 shows the distribution of ground truth and estimated parameters for this new PanoContext- $uf\mathbf{R}$  dataset together with those for the York Urban DB and PanoContext- $f\mathbf{R}$  datasets used in the paper. We observe that for all three datasets, the geometric methods are fairly well-tuned to the ground truth FOVs, while the deep learning methods are not. Note in particular that while our  $f\mathbf{R}$  system adapts to the new distribution of focal lengths in the PanoContext- $uf\mathbf{R}$

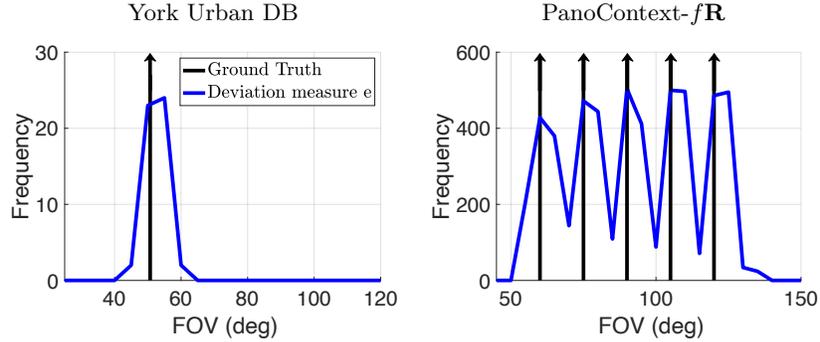


Fig. 1: Distribution of estimated FOVs relative to ground truth for method based upon deviation measure  $e$ , on York UrbanDB and PanoContext- $f\mathbf{R}$  test partitions.

Table 3: Performance comparison with SOA on the PanoContext- $u/\mathbf{R}$  test set. Numbers are mean  $\pm$  standard error.

Methods	Roll MAE (deg)	Tilt MAE (deg)	Focal length MAE (%)
Lee[4]	$2.38 \pm 0.04$	$3.80 \pm 0.06$	$24.7 \pm 0.37$
Simon[7]	$1.49 \pm 0.02$	$3.22 \pm 0.05$	$47.9 \pm 0.48$
Hold-Geoffroy[3]	$1.45 \pm 0.01$	$3.43 \pm 0.03$	$36.1 \pm 0.35$
CTRL-C[5] S360	$1.01 \pm 0.01$	$1.95 \pm 0.02$	$11.2 \pm 0.11$
CTRL-C [5] GSV	$2.17 \pm 0.02$	$8.93 \pm 0.09$	$26.94 \pm 0.26$
$f\mathbf{R}$ (ours)	<b><math>0.74 \pm 0.02</math></b>	<b><math>1.31 \pm 0.04</math></b>	<b><math>8.2 \pm 0.23</math></b>

dataset, the deep learning systems do not, revealing an insensitivity to focal length. This confirms that results reported in the paper were not a consequence of the discrete sampling of FOVs in the PanoContext- $f\mathbf{R}$  dataset.

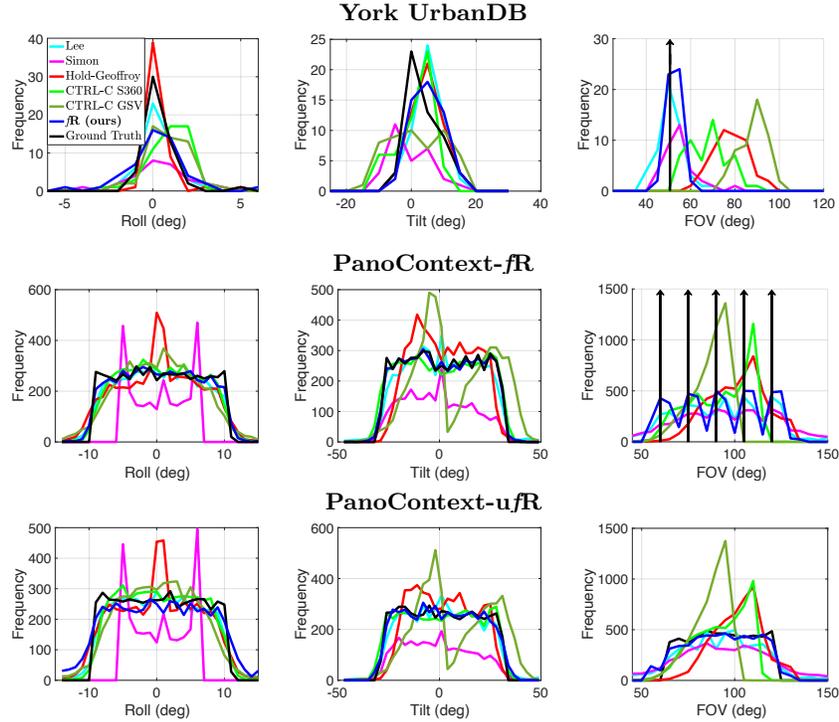


Fig. 2: Distribution of ground truth and estimated camera parameters for the test partition of the PanoContext- $uf\mathbf{R}$  dataset.

### 3 $f\mathbf{R}$ design choices

#### 3.1 Principal point

Our  $f\mathbf{R}$  system estimates focal length and camera rotation but assumes a central principal point. Since the three vanishing points of a Manhattan world impose a total of 6 constraints, it is possible in principal to estimate principal point jointly with focal length and camera rotation. To explore this idea, we created a new version of our  $f\mathbf{R}$  system that assumes a central principal point in Stage 1 of our search (Section 3.2 of the paper) but then includes the principal point as one of the target parameters in the second stage nonlinear search, constraining it to lie within the central 5% of the image.

Table 4 compares the performance of this approach with our standard approach that assumes a central principal point, and an oracle approach that uses the ground truth principal point. The results are clear: while perfect knowledge of the principal point does not significantly improve accuracy of focal length or camera rotation estimates, attempting to jointly estimate the principal point tends to increase error, presumably because increasing the dimension of the search space increases the probability that noise will distort the objective function and also increases the number of local extrema.

Table 4: Here we assess how alternative methods for handling the principal point affect the accuracy of our  $f\mathbf{R}$  system on the YorkUrbanDB training set. We consider a) using the known (ground truth) value of the principal point, b) assuming a central principal point and c) estimating the principal point jointly with focal length and rotation. Numbers reported are mean and standard error.

Principle point	Frame angle error (deg)	Focal length MAE (%)
Ground truth	1.28± 0.20	4.40±0.94
<b>Central</b>	<b>1.45± 0.14</b>	<b>5.2±1.10</b>
Estimated	1.90± 0.22	6.9±1.37

### 3.2 Weighting by line segment length

The  $f\mathbf{R}$  objective function (Eqn. 2 in the main paper) weights the likelihood of each line segment by its length. Table 5 shows that this results in a slight improvement in accuracy on the York Urban DB.

Table 5: Here we assess the affect of weighting the log likelihoods of each line segment by its length in the  $f\mathbf{R}$  objective function (Eqn. 2 in the main paper). Numbers reported are mean and standard error of the frame error and mean of the absolute focal length error.

Line segment weighting	Frame angle error (deg)	Focal length MAE (%)
Uniform	1.54± 0.16	5.6±0.95
<b>By length</b>	<b>1.45± 0.14</b>	<b>5.2±1.10</b>

## 4 Likelihood models

The  $f\mathbf{R}$  objective function (Eqn. 2 in the main paper) is based upon a likelihood measure (Eqn. 1 in the main paper) that measures the likelihood of a deviation between a hypothesized vanishing point and a segment generated by that vanishing point. We evaluated five possible deviation measures **a** - **e**. We employed exponential likelihood models for all measures except **d**, for which we used the Gaussian measure employed by Xu et al. [9]. The parameters employed for these likelihood models are shown in Table 1 of the main paper.

The likelihood models for deviation measures **a** - **c** were fit to the YorkUrbanDB training partition - example fits for horizontal vanishing points are shown in Fig. 3 below. The model parameters for deviation measures **d** and **e** were taken from Xu et al [9] and Tal & Elder [8], respectively.

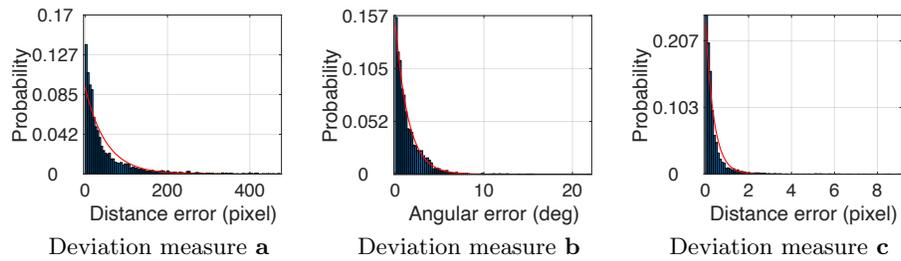


Fig. 3: Maximum likelihood fits of exponential likelihood models for deviation measures **a-c** for horizontal vanishing points on the York Urban DB training partition.

## 5 Evaluating line segment detectors

Table 4 in the main paper evaluates the performance of our  $f\mathbf{R}$  system with three different line segment detectors, on the PanoContext- $f\mathbf{R}$  test set. Table 6 below provides the same evaluation on the York Urban DB test set. As for PanoContext- $f\mathbf{R}$ , the MCMLSD detector yields the highest accuracy.

Table 6: Evaluating the choice of line segment detector on the YorkUrbanDB test set. Numbers are mean  $\pm$  standard error.

Detector	Frame err (deg)	Focal len. MAE (%)
MCMLSD [1]	1.54 $\pm$ 0.16	4.6 $\pm$ 0.78
LSD [2]	1.76 $\pm$ 0.29	5.6 $\pm$ 1.09
HT-LCNN [6]	1.58 $\pm$ 0.16	5.2 $\pm$ 0.99

## 6 Effects of domain shift on focal length estimation

Performance of deep learning methods on FOV/focal length estimation might derive in part from limitations in the range of FOVs in their training datasets. While we evaluated performance over the range of horizontal FOVs found in typical consumer cameras (60 - 120 deg), CTRL-C was trained on images ranging only over 40-80 deg. To assess the impact of this domain shift, we employed our PanoContext-u/ $f\mathbf{R}$  dataset that samples FOVs uniformly over 60 - 120 deg, and evaluated CTRL-C and  $f\mathbf{R}$  systems on the subset of images with FOVs in the 60 - 80 deg range, within the CTRL-C training range. We found that while focal length error declined slightly for CTRL-C (12.0% for S360 and 18.8% for GSV),  $f\mathbf{R}$  was still substantially more accurate (9.4%). Thus domain shift cannot entirely explain the poorer performance of CTRL-C on focal length estimation.

## 7 Pan Estimation Results

One advantage of geometric methods over deep learning methods trained on curated planar projections from panoramic datasets is that the geometric methods can estimate the full rotation matrix, including the pan angle. Table 7 below compares pan error for the three geometric systems considered here, evaluated on the York Urban DB. Our  $f\mathbf{R}$  system performs substantially better than the other two geometric systems, reducing average pan error relative to the next best system (Lee et al. [4]) from 3.91 deg to 2.60 deg, a 34% improvement.

Table 7: Evaluation of pan angle estimation for geometric methods, on the York Urban DB test partition.

Methods	Pan MAE (deg)
Lee[4]	3.91
Simon[7]	12.40
$f\mathbf{R}$ (ours)	<b>2.60</b>

## 8 Distribution of Errors

Fig. 4 shows the distribution of errors for the systems evaluated here, on the York Urban DB test partition. Note that our  $f\mathbf{R}$  system has a much lighter positive tail than competing methods, indicating more robust performance.

## 9 Predicting Reliability

We employ three global cues to predict camera parameter estimation error: 1) The minimum number of segments over the three Manhattan directions, 2) Entropy over our parameter grid search and 3) mean log likelihood of the final parameter estimate. To assess each of the cues visually, we compute the log error in

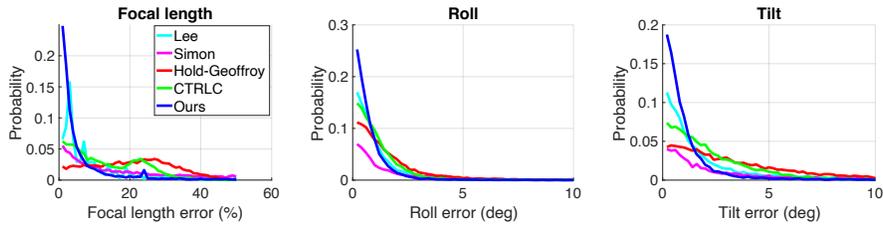


Fig. 4: Distribution of errors on the York Urban DB test partition.

the camera parameters estimated for each training image in the PanoContext- $f\mathbf{R}$  training set as a function of the cue, using KNN regression to smooth the data, and using five-fold cross-validation to identify the optimal K (Fig. 5). We can see from these plots that each of the three cues is predictive of error in focal length and camera rotation parameters.

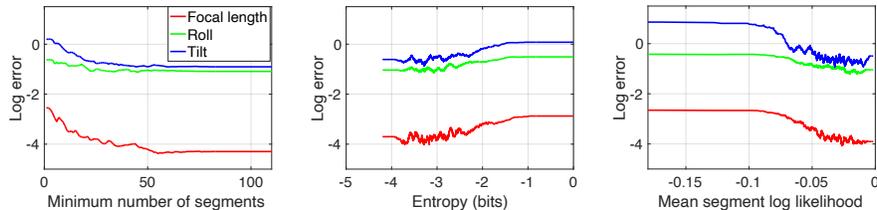
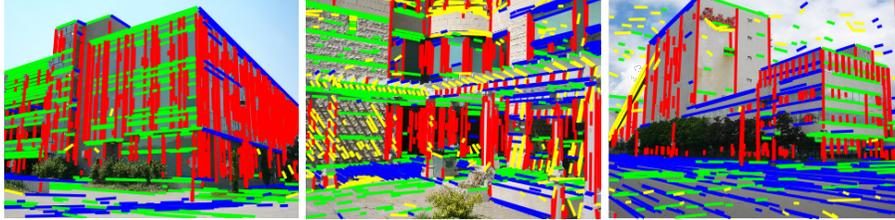


Fig. 5: KNN predictions of parameter error as a function of three cues.

## 10 Qualitative Results

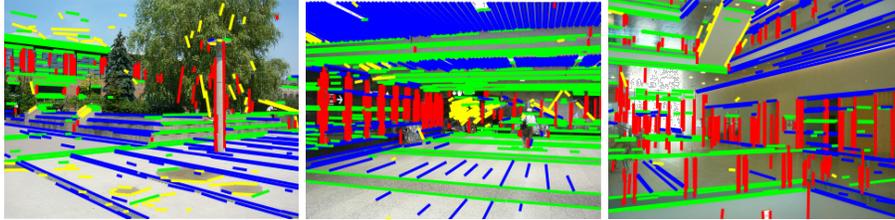
Figs. 6 and 7 show the YorkUrbanDB and PanoContext- $f\mathbf{R}$  test images on which our  $f\mathbf{R}$  system generates the three best, three median and three worst focal length errors. Examination of the worst cases reveal three main failure modes: 1) Interference from salient non-Manhattan segments (e.g., YorkUrbanDB worst examples 1 and 3). Note that even when we correctly label these segments in the images as background (yellow), they still influence the estimation of the camera parameters, since we do not collapse the mixture model to these modes; 2) Limited or poor quality Manhattan segments in one or more of the 3 Manhattan directions (e.g., YorkUrbanDB worst example 2, PanoContext- $f\mathbf{R}$  worst examples 2-3; 3) Pan angles near 0 deg, i.e., non-generic views (e.g., YorkUrbanDB worst example 3, PanoContext- $f\mathbf{R}$  worst examples 1-3).

### Lowest focal length error



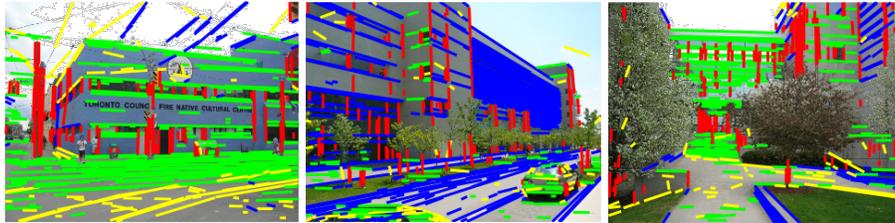
Frame error: 1.17 deg    Focal length error: 0.02%    Frame error: 0.35 deg    Focal length error: 0.07%    Frame error: 0.23 deg    Focal length error: 0.25%

### Median focal length error



Frame error: 0.98 deg    Focal length error: 2.86%    Frame error: 0.89 deg    Focal length error: 2.87%    Frame error: 1.63 deg    Focal length error: 2.87%

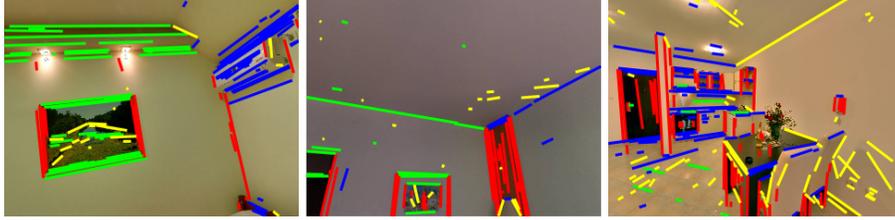
### Highest focal length error



Frame error: 3.0 deg    Focal length error: 11.4%    Frame error: 7.3 deg    Focal length error: 30.2%    Frame error: 3.1 deg    Focal length error: 30.3%

Fig. 6: YorkUrbanDB test images on which our  $f\mathbf{R}$  system generates the three best, three median and three worst focal length errors. Detected MCMLSD line segments  $\mathbf{l}_i$  are coloured according to the maximum likelihood generating process  $m_i \in M$ : red for vertical, blue and green for horizontal, and yellow for background.

### Lowest focal length error

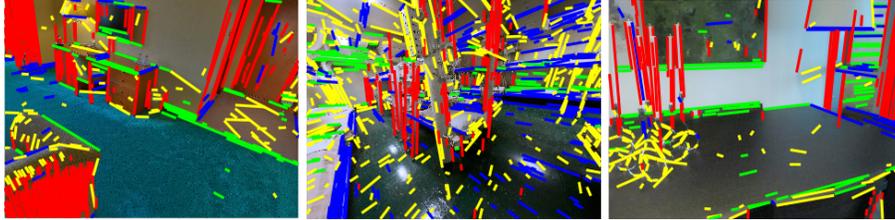


Roll error: 1.39 deg  
Tilt error: 0.27 deg  
Focal length error: 0%

Roll error: 0.65 deg  
Tilt error: 0.56 deg  
Focal length error: 0%

Roll error: 0.04 deg  
Tilt error: 0.02 deg  
Focal length error: 0%

### Median focal length error

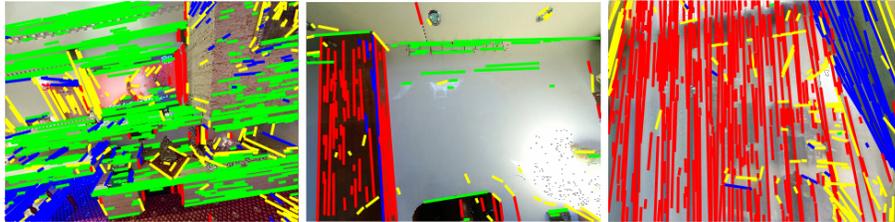


Roll error: 0.04 deg  
Tilt error: 0.71 deg  
Focal length error: 2.53%

Roll error: 0.42 deg  
Tilt error: 0.54 deg  
Focal length error: 2.53%

Roll error: 0.8 deg  
Tilt error: 0.51 deg  
Focal length error: 2.54%

### Highest focal length error



Roll error: 1.48 deg  
Tilt error: 52.04 deg  
Focal length error: 271.4%

Roll error: 3.48 deg  
Tilt error: 17.46 deg  
Focal length error: 271.4%

Roll error: 1.12 deg  
Tilt error: 15.09 deg  
Focal length error: 271.4%

Fig. 7: PanoContext- $f\mathbf{R}$  test images on which our  $f\mathbf{R}$  system generates the three best, three median and three worst focal length errors. Detected MCMLSD line segments  $\mathbf{l}_i$  are coloured according to the maximum likelihood generating process  $m_i \in M$ : red for vertical, blue and green for horizontal, and yellow for background.

## References

1. Almazan, E.J., Tal, R., Qian, Y., Elder, J.H.: MCMLSD: A dynamic programming approach to line segment detection. In: IEEE Conference on Computer Vision and Pattern Recognition. pp. 2031–2039 (2017) [6](#)
2. von Gioi, R.G., Jakubowicz, J., Morel, J.M., Randall, G.: Lsd: A fast line segment detector with a false detection control. *IEEE Transactions on Pattern Analysis & Machine Intelligence* **32**(4), 722–732 (2008) [6](#)
3. Hold-Geoffroy, Y., Sunkavalli, K., Eisenmann, J., Fisher, M., Gambaretto, E., Hadap, S., Lalonde, J.F.: A perceptual measure for deep single image camera calibration. In: IEEE Conference on Computer Vision and Pattern Recognition. pp. 2354–2363 (2018) [3](#)
4. Lee, D.C., Hebert, M., Kanade, T.: Geometric reasoning for single image structure recovery. In: IEEE Conference on Computer Vision and Pattern Recognition. pp. 2136–2143. IEEE (2009) [3](#), [7](#)
5. Lee, J., Go, H., Lee, H., Cho, S., Sung, M., Kim, J.: CTRL-C: Camera calibration transformer with line-classification. In: International Conference on Computer Vision. pp. 16228–16237 (2021) [3](#)
6. Lin, Y., Pintea, S.L., van Gemert, J.C.: Deep hough-transform line priors. In: European Conference on Computer Vision. pp. 323–340. Springer (2020) [6](#)
7. Simon, G., Fond, A., Berger, M.O.: A Simple and Effective Method to Detect Orthogonal Vanishing Points in Uncalibrated Images of Man-Made Environments. In: Bashford-Rogers, T., Santos, L.P. (eds.) EuroGraphics - Short Papers. The Eurographics Association (2016). <https://doi.org/10.2312/egsh.20161008> [3](#), [7](#)
8. Tal, R., Elder, J.H.: An accurate method for line detection and Manhattan frame estimation. In: Park, J.I., Kim, J. (eds.) Asian Conference on Computer Vision-Workshops. pp. 580–593. Springer Berlin Heidelberg, Berlin, Heidelberg (2012) [6](#)
9. Xu, Y., Oh, S., Hoogs, A.: A minimum error vanishing point detection approach for uncalibrated monocular images of man-made environments. In: IEEE Conference on Computer Vision and Pattern Recognition. pp. 1376–1383 (2013) [6](#)