Vanishing Point Estimation in Uncalibrated Images with Prior Gravity Direction
Supplementary Material

In the following, we provide additional details about our approach. Section A gives the complete derivation of our proposed non minimal solver. Section B offers additional details and derivations that were not covered in the main paper. Section C displays more synthetic experiments with our proposed solvers. Section D provides an ablation of our proposed Local Optimization (LO) on ScanNet [4] and with different numbers of iterations. Section E offers insights about the generalization of our methods to other RANSAC strategies. Finally, Section F shows multiple visualizations of vanishing point estimation.

A. Complete Derivations of the Non Minimal Solver

Section 2.4 of the main paper introduces our non minimal solver to estimate the orthogonal vanishing points and unknown focal length from an existing set of three vanishing points and their inlier lines. We describe here in more details the two least square methods that are used in this solver.

Re-using the same notations, we are given three vanishing points $\mathbf{v}_1$, $\mathbf{v}_2$, $\mathbf{v}_3$, and three sets of lines $\mathcal{L}_1$, $\mathcal{L}_2$, $\mathcal{L}_3$, where each set $\mathcal{L}_i, i \in \{1, 2, 3\}$ contains n_i inliers of vanishing point $\mathbf{v}_i$. The first step is to re-estimate each vanishing point $\mathbf{v}_i, i \in \{1, 2, 3\}$ from its inliers $\mathcal{L}_i$ using the least squares (LSQ) method. For this, we write the sum of distances between each inlier line $\mathbf{l}_j \in \mathcal{L}_i$ and the corresponding VP $\mathbf{v}_i$, using homogeneous coordinates:

$$\sum_{\mathbf{l}_j \in \mathcal{L}_i} d(\mathbf{l}_j, \mathbf{v}_i) = \sum_{\mathbf{l}_j \in \mathcal{L}_i} \frac{|\mathbf{l}_j^T \mathbf{v}_i|}{\sqrt{l_j(0)^2 + l_j(1)^2}}. \quad (1)$$

Introducing the $n_i \times 3$ matrix M_i, defined by its rows $M_i(j)$:

$$M_i(j) = \frac{\text{sign}(\mathbf{l}_j^T \mathbf{v}_i)}{\sqrt{l_j(0)^2 + l_j(1)^2}} \mathbf{l}_j^T, \quad (2)$$

One can re-write the previous objective as $M\mathbf{v}_i = 0$, and the solution is obtained by computing the right null space of the SVD of matrix M. This solution becomes our refined vanishing point.

Next, we compute the unknown focal length. From equations (1) and (3) of the main paper and re-using the same notations, we have, for every pair of vanishing points $\mathbf{v}_i$, $\mathbf{v}_j$: $\mathbf{d}_i^T \mathbf{d}_j = \mathbf{v}_i^T (\mathbf{K}^{-1})^T \mathbf{K}^{-1} \mathbf{v}_j = 0$. This constraint can be rewritten as:

$$-\mathbf{v}_i(2)\mathbf{v}_j(2)f^2 = \mathbf{v}_i(0)\mathbf{v}_j(0) + \mathbf{v}_i(1)\mathbf{v}_j(1), \quad (3)$$

where the numbers in the parentheses refer to the x, y, and w coordinates of the homogeneous points. Taking $(i, j) \in \{(1, 2), (1, 3), (2, 3)\}$ gives three independent constraints that all are linear in f^2:

$$\begin{pmatrix} -\mathbf{v}_1(2)\mathbf{v}_2(2) \\ -\mathbf{v}_1(2)\mathbf{v}_3(2) \\ -\mathbf{v}_2(2)\mathbf{v}_3(2) \end{pmatrix} f^2 = \begin{pmatrix} \mathbf{v}_1(0)\mathbf{v}_2(0) + \mathbf{v}_1(1)\mathbf{v}_2(1) \\ \mathbf{v}_1(0)\mathbf{v}_3(0) + \mathbf{v}_1(1)\mathbf{v}_3(1) \\ \mathbf{v}_2(0)\mathbf{v}_3(0) + \mathbf{v}_2(1)\mathbf{v}_3(1) \end{pmatrix} \quad (4)$$

We solve it via QR decomposition, and since f can not be negative, we finally obtain $f = +\sqrt{f^2}$.

Finally, we correct the vanishing points to be orthogonal. We compute the calibrated vanishing points $\mathbf{d}_i, i \in \{1, 2, 3\}$ as $\mathbf{d}_i = \mathbf{K}^{-1}\mathbf{v}_i$, build a matrix $\mathbf{D} = [\mathbf{d}_1 \ \mathbf{d}_2 \ \mathbf{d}_3]$, and decompose as $\mathbf{D} = \mathbf{U}\mathbf{S}\mathbf{V}^T$. We take the rows of matrix $\mathbf{R} = \mathbf{U}\mathbf{V}^T$ as the refined calibrated vanishing points.

B. Alternative Solvers

B.1. Different Elimination Order for 1-1-0g Solver

Here, we give an alternative elimination order for our 1-1-0g solver, introduced in Section 2.3 of the main paper. We reuse here the same notations. Estimating the orthogonal vanishing points from projections of two mutually orthogonal horizontal lines, and a known vertical direction (referred to as 1-1-0g), leads to two equations of unknown t and f:

$$\begin{aligned} (1 - t^2)\mathbf{l}_1^T \mathbf{K}\mathbf{b}_1 - 2t\mathbf{l}_1^T \mathbf{K}\mathbf{b}_2 = 0, \\ 2t\mathbf{l}_2^T \mathbf{K}\mathbf{b}_1 + (1 - t^2)\mathbf{l}_2^T \mathbf{K}\mathbf{b}_2 = 0. \end{aligned} \quad (5)$$

In the main paper, we propose to solve these equations by first eliminating t, and solving for f, leading to a degree 2 polynomial. Here, we give an alternative approach, where the equations are solved by eliminating f and solving for t afterwards.

Both equations are linear in f. Reusing the notation of the main paper defined in (12), these two equations become:

$$
\begin{aligned}
(1 - t^2)(f\delta_1 + \delta_2) - 2t(f\delta_3 + \delta_4) &= 0, \\
(1 - t^2)(f\delta_5 + \delta_6) + 2t(f\delta_7 + \delta_8) &= 0.
\end{aligned} \tag{6}
$$

We can easily express the focal length from the first equation of (6) as a function of t and substitute it into the second equation to get the following constraint:

$$
\begin{aligned}
0 =\,& 4t^2(\delta_4\delta_7 - \delta_3\delta_8) \\
& + 2t(1 - t^2)(\delta_4\delta_5 + \delta_1\delta_8 - \delta_3\delta_6 - \delta_2\delta_7) \\
& + (1 - t^2)(\delta_1\delta_6 - \delta_2\delta_5).
\end{aligned} \tag{7}
$$

This is a univariate quartic polynomial equation. We use the hidden variable approach to solve for t, which gives us 4 solutions. For every solution, we find f from (6) and only keep (t, f) pairs where the focal length f is positive. Then, we use t to calculate the rotation matrix $\mathbf{R}$. However, this approach is slower than the one proposed in the main paper: it needs $1.64\mu s$ to solve one instance of the 1-1-0g problem, while the proposed one only needs $0.17\mu s$.

B.2. Non Minimal Solver with Linearized Rotation

In the main paper, we use a non-minimal solver that estimates each vanishing point separately, and then corrects the vanishing points to be orthogonal. Here, we propose an alternative non-minimal solver, which is iterative and uses a first-order approximation of the matrix $\mathbf{KR}$ to estimate all 3 vanishing points simultaneously.

We use the same notation as in the main paper: $\mathbf{v}_i, i \in \{1, 2, 3\}$ denotes a vanishing point in direction $\mathbf{d}_i$. $\mathcal{L}_i = \{\mathbf{l}_{i,j}, j \in \{1, ..., n_i\}\}, i \in \{1, 2, 3\}$ is a set of n_i lines consistent with vanishing point $\mathbf{v}_i$.

This non-minimal solver uses a linearized model of matrix $[\mathbf{v}_1\ \mathbf{v}_2\ \mathbf{v}_3] = \mathbf{KR}$ to simultaneously minimize the sum of squared errors:

$$
\sum_{i=1}^{3} \sum_{j=1}^{n_i} \left(\frac{\mathbf{l}_{i,j}^{\mathrm{T}} \mathbf{v}_i}{\|\mathbf{l}_{i,j}\|} \right)^2. \tag{8}
$$

Let $\mathbf{K}_0\mathbf{R}_0$ be the initial estimate of $\mathbf{KR}$, obtained by the minimal solver. The first-order Taylor polynomial of $\mathbf{KR}$ can be obtained as:

$$
\mathbf{KR} \approx \mathbf{K}_0\mathbf{R}_0 + \delta\mathbf{K}\mathbf{R}_0 - \mathbf{K}\delta\mathbf{R}\mathbf{R}_0 \tag{9}
$$

with derivatives $\delta\mathbf{K}$ and $\delta\mathbf{R}$ defined as:

$$
\delta\mathbf{K} = \begin{bmatrix} \delta f & 0 & 0 \\ 0 & \delta f & 0 \\ 0 & 0 & 0 \end{bmatrix}, \delta\mathbf{R} = \begin{bmatrix} 0 & -\delta v_3 & \delta v_2 \\ \delta v_3 & 0 & -\delta v_1 \\ -\delta v_2 & \delta v_1 & 0 \end{bmatrix}. \tag{10}
$$

Now, we can approximate every vanishing point $\mathbf{v}_i$ as:

$$
\mathbf{v}_i \approx \mathbf{B}_i\delta\mathbf{x} + \mathbf{c}_i, \tag{11}
$$

where

$$
\begin{aligned}
\mathbf{B}_i &= \begin{bmatrix} 0 & -f_0 r_{3,i} & f_0 r_{2,i} & r_{1,i} \\ f_0 r_{3,i} & 0 & -f_0 r_{1,i} & r_{2,i} \\ -r_{2,i} & r_{1,i} & 0 & 0 \end{bmatrix}, \\
\mathbf{c}_i &= \begin{bmatrix} f_0 r_{1,i} \\ f_0 r_{2,i} \\ r_{3,i} \end{bmatrix}, \delta\mathbf{x} = \begin{bmatrix} \delta v_1 & \delta v_2 & \delta v_3 & \delta f \end{bmatrix}^{\mathrm{T}}.
\end{aligned} \tag{12}
$$

We use matrices $\mathbf{A}_i, i \in \{1, 2, 3\}$ defined in Section 2.4 of the main paper, and build matrices:

$$
\mathbf{A} = \begin{bmatrix} \mathbf{A}_1 & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{A}_2 & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{A}_2 \end{bmatrix}, \mathbf{B} = \begin{bmatrix} \mathbf{B}_1 \\ \mathbf{B}_2 \\ \mathbf{B}_3 \end{bmatrix}, \mathbf{C} = \begin{bmatrix} \mathbf{c}_1 \\ \mathbf{c}_2 \\ \mathbf{c}_3 \end{bmatrix}. \tag{13}
$$

Then, we re-write the minimization problem of (8) as:

$$
\min\|\mathbf{AB}\delta\mathbf{x} + \mathbf{AC}\|^2, \tag{14}
$$

and we find the update $\delta\mathbf{x}$ with the least-squares method. Then, we find the vanishing points by (11). To find the focal length f and the orthogonal calibrated vanishing points $\mathbf{d}_1, \mathbf{d}_2, \mathbf{d}_3$ from the uncalibrated vanishing points $\mathbf{v}_1, \mathbf{v}_2, \mathbf{v}_3$, we use the procedure proposed in the main paper in Section 2.4. For the next iteration, we set $f_0 := f$, and $\mathbf{R}_0 := [\mathbf{d}_1, \mathbf{d}_2, \mathbf{d}_3]$, and find the update in the same way.

The comparison of this non-minimal solver with the one proposed in the main paper is shown in the following section. We observed a lower performance for the solver with linearized rotation in our real-world experiments, and thus only presented the non orthogonal one in the main paper. However, the former could still be used in cases with small rotations with only a few iterations, and could become an efficient alternative to the non orthogonal solver.

C. Additional Synthetic Tests

In order to further evaluate the solvers in various scenarios, we have performed additional synthetic tests, presented in this section.

Non minimal solver tests. To evaluate the non-minimal solvers, we generated the minimal problems exactly as in the main paper. Figure 1 shows the average rotation and focal length errors of every proposed minimal solvers refined by the linearized non-minimal solver with 10 iterations. The 1-1-0g solver leads to the most accurate solutions on almost all noise configurations. Figure 2 shows the average rotation and focal length errors of the results of different non-minimal solvers, as a function of the number of lines used within the solver. The error does not change significantly

after adding more than 20 lines per direction. While the linearized non-minimal solver gives a better estimation of the rotation, using the nonorthogonal non-minimal solver leads to lower focal length errors. Figure 3 shows the errors of different non-minimal solvers as a function of the input noise. Again, the linearized non-minimal solver gives more accurate rotations, while the nonorthogonal non-minimal solver gives more accurate focal lengths. Figure 4 shows the evaluation of the running time of different non-minimal solvers. The runtime increases with the increasing number of lines used within the non-minimal solver, and this growth is roughly linear.

Principal point tests. To evaluate the robustness of our solvers to the incorrectly estimated principal point, we generate minimal problems similarly to the main paper, and we perturb the principal point with Gaussian noise with standard deviation σ_p. Figure 5 shows the average rotation and focal length errors on different levels of σ_p. Solvers 1-1-0_g and 2-2-0 lead to the more accurate solutions. Figure 6 shows the average rotation and focal length errors on different levels of σ_p refined by the non-minimal solver. It can be seen that all the proposed non-minimal solvers are very robust to small noise on the principal point, and that their estimate of the focal length is significantly better than without any local optimization.

D. Additional Ablations on the LO

Evaluation on ScanNet [4]. Table 1 shows a similar ablation study for our proposed Local Optimization (LO) as in the main paper, but for the ScanNet dataset [4]. Note that the lack of improvement from Iter for our solvers stems from the very noisy gravity prior in this dataset, that makes the initial model too noisy for the optimization to converge to an accurate solution.

Number of Iterations. The results in the main paper correspond to the best numbers that could be obtained, assuming that time is not a limitation. In resource-constrained scenarios when speed matters, it is also possible to reduce the number of LO iterations to significantly speed up the vanishing point detection, for a minor drop of performance. As shown in Table 2, running only 10 iterations of LO is already enough to achieve a high performance, at a negligible overhead time.

E. Generalization to other RANSAC

While our proposed approach leverages existing RANSAC frameworks such as LO-RANSAC [7] and hybrid RANSAC [3], it can also be applied to more recent RANSAC strategies. We adapt here our approach to MAGSAC [1, 2], one of the state-of-the-art RANSAC currently existing. We replace our scoring method with the one proposed in MAGSAC, and obtain the results of
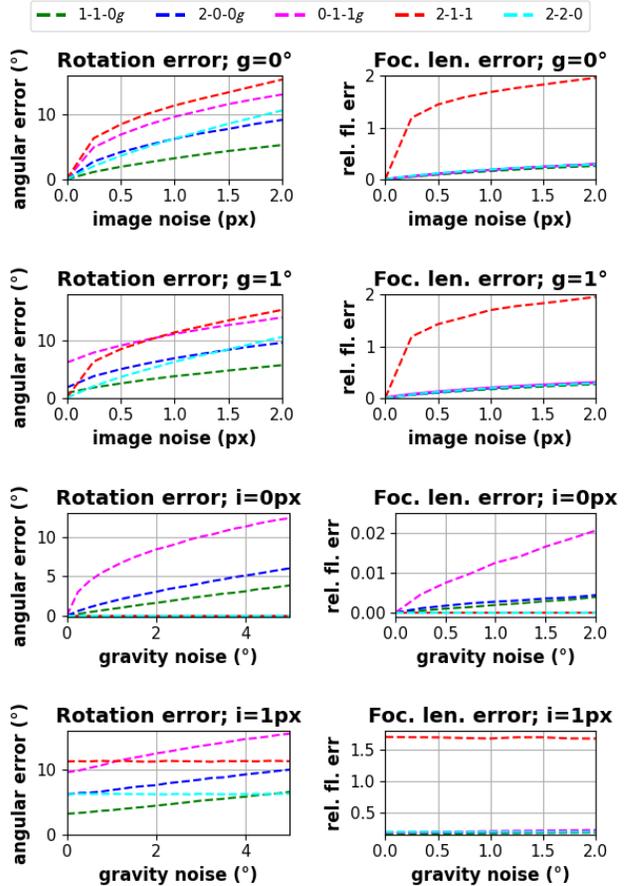

Figure 1: **Noise study of the linearized non-minimal solver.**: The average (over 100000 runs) rotation (left), and relative focal length error (right) of five solvers as a function of the image and gravity noise. The results are refined with a linearized non-minimal solver with 10 iterations and 20 lines per direction. The fixed noise std. is in the title (g - gravity, i - image).

Table 3. MAGSAC scoring can slightly boost the performance, showing that future improvements on RANSAC can further benefit our approach.

F. Visualizations of VPs and their Applications

We display several visualizations of the inlier lines for each vanishing point in Figure 7, with one color per VP. Our hybrid RANSAC with prior gravity is able to find more inliers and better rotations than the previous best solver for uncalibrated images [10]. When using the ground truth gravity, the results are even further improved.

Note that in the last two rows, we display images of the ScanNet dataset [4], which has some unstructured scenes and can be quite challenging for vanishing point estima-
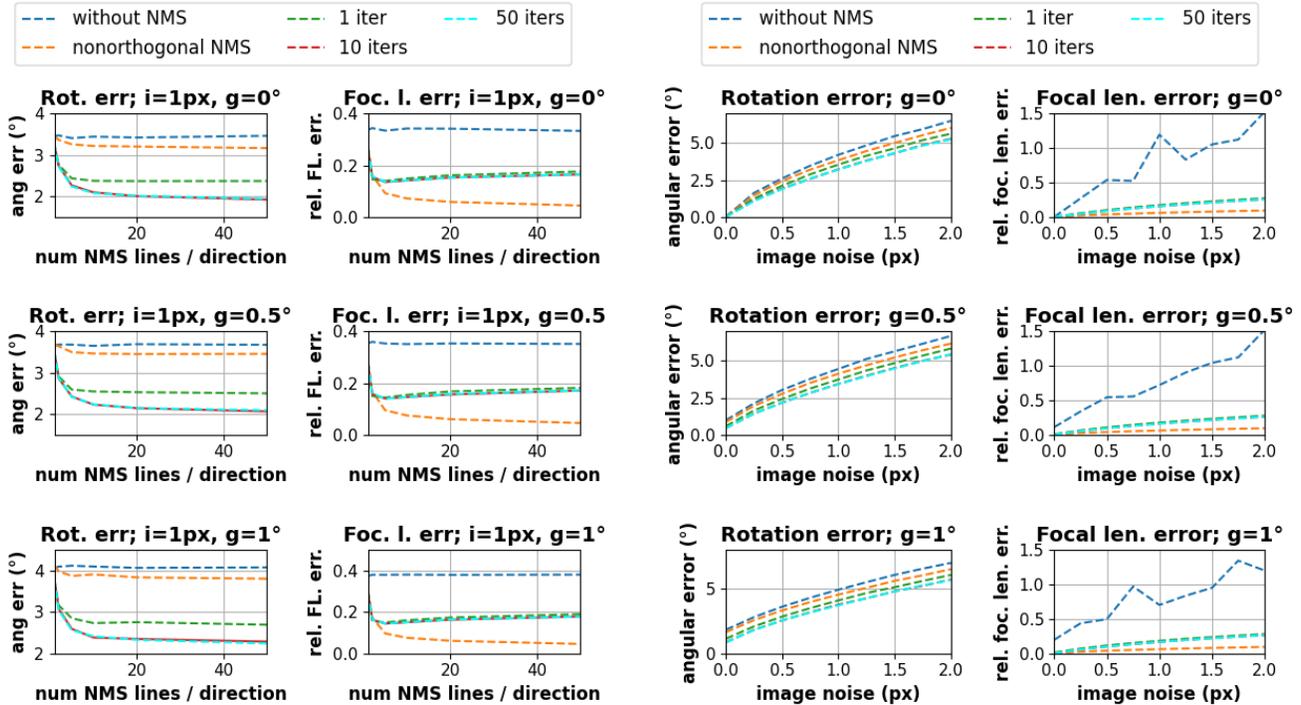
Figure 2: **Effect of the number of lines on non-minimal solvers (NMS).**: Average (over 100000 samples) rotation (left) and relative focal length error (right) as a function of the number of lines used in the NMS. We considered the following approaches: baseline (without NMS) , non-orthogonal NMS presented in the main paper, and linearized rotation with n iterations (n iters). The 1-1-0g solver was used for the initialization. The fixed noise std. is in the title (g - gravity, i - image).

tion. For example, in the last row, the 2-1-1 solver [10] is misleaded by the many red lines of the dish dryer, which are actually inconsistent with the real Manhattan directions. On the contrary, our approach with prior gravity finds a better vertical direction and consequently better recognizes the other two vanishing directions. Finally, the hybrid RANSAC leveraging ground truth gravity is able to detect correctly all the Manhattan directions and obtains a much lower rotation error.

Additionally, to better highlight the application scenarios of our method, we show in Figure 8 two cases where a prior on the gravity is available. The first one (first three rows) is for autonomous driving, where the cameras are usually always upright and the gravity can be assumed to be vertical. In the second example (last two rows), augmented and mixed reality (AR/MR) devices have usually an onboard IMU providing the gravity. In both cases, cars and AR/MR headsets are devices used over long periods of time, and the calibration of theirs cameras are subject to drift, and may
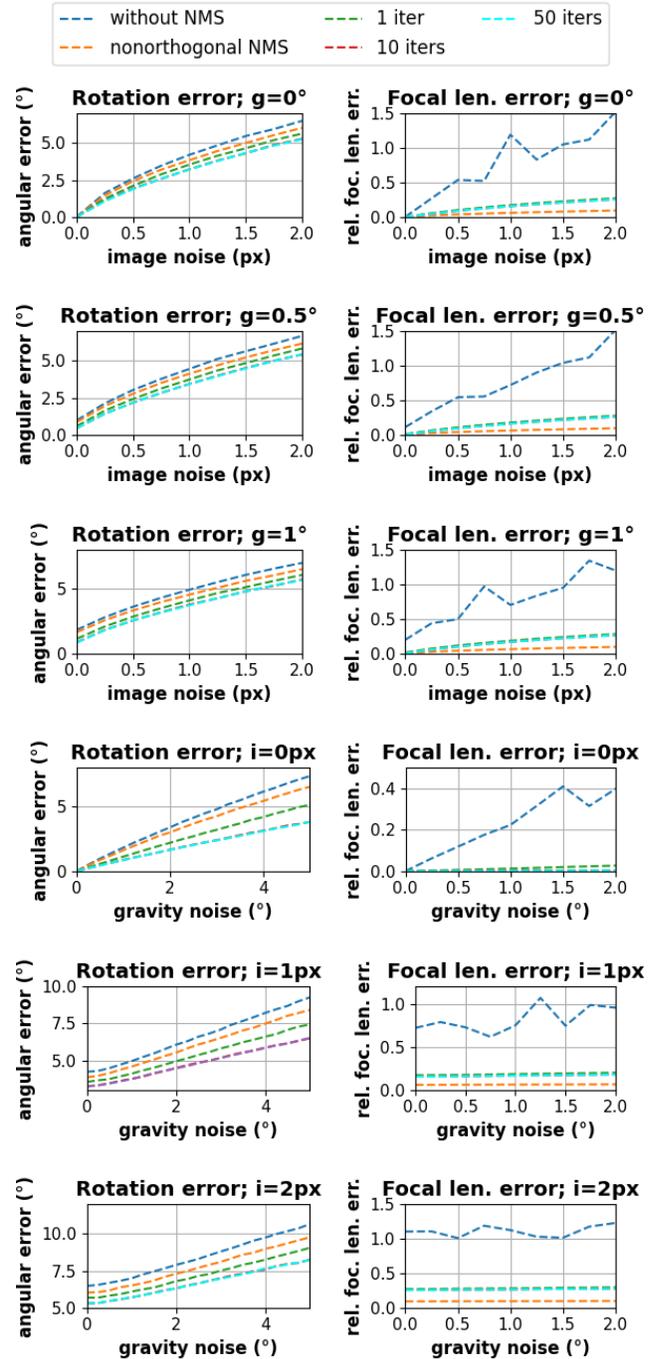
Figure 3: **Effect of noise on our non-minimal solvers (NMS).**: The plots show the relation between the noise of the input (endpoints of the lines in px, gravity vector in degrees) and the rotation and relative focal length error. We consider the following approaches: baseline (without NMS) , non orthogonal NMS of the mai paper, linearized rotation with n iterations (n iters). The 1-1-0g solver was used for the initialization. The fixed noise std. is in the title (g - gravity, i - image).

Figure 4: **Run-time of our non-minimal solvers** as a function of number of lines used in the solver.

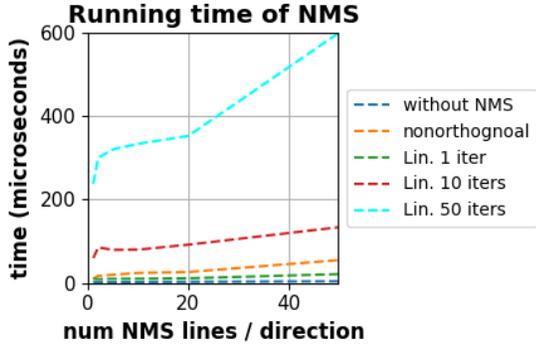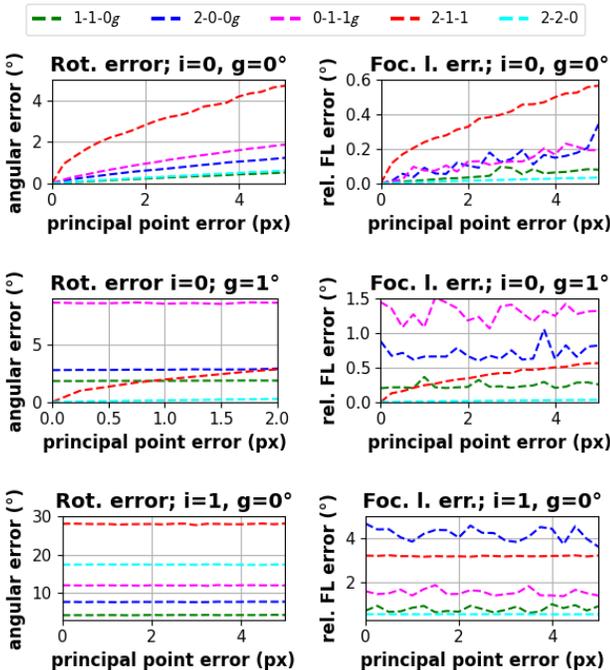Figure 5: **Effect of noise on the principal point.** The average (over 100000 runs) rotation (left), and relative focal length error (right) of the proposed solvers as a function of the principal point noise σ_p. The fixed noise std. is in the title (g - gravity, i - image).

need to be recalibrated on-the-fly. Thus, our method can benefit from the prior gravity available in both situations, while providing an estimate of the focal length at test time.

References

[1] Daniel Barath, Jiri Matas, and Jana Noskova. MAGSAC: marginalizing sample consensus. In *Conference on Computer Vision and Pattern Recognition*, 2019. 3

Solver	LO	Rotation estimation		VP estimation		Time (ms)
		Err ↓	AUC ↑	Err ↓	AUC ↑	
2-2-0 [10] No gravity	No	42.98	0.7 / 3.6 / 11.6	8.45	1.94	**22**
	Iter [11]	**42.96**	3.5 / 8.7 / 16.5	7.55	2.73	26
	Ours	43.47	**6.8 / 13.8 / 21.5**	**6.50**	**3.67**	130
2-1-1 [10] No gravity	No	42.21	1.0 / 4.9 / 14.3	8.09	2.26	**28**
	Iter [11]	42.69	4.0 / 10.2 / 18.1	7.18	2.99	33
	Ours	**42.07**	**7.0 / 14.3 / 22.0**	**6.42**	**3.75**	112
2-0-0g [8] + Prior	No	50.07	0.2 / 0.7 / 2.8	9.40	0.64	**24**
	Iter [11]	50.46	0.2 / 0.7 / 2.9	9.39	0.64	30
	Ours	**43.99**	**6.5 / 13.1 / 20.2**	**6.60**	**3.55**	131
0-1-1g + Prior	No	49.87	0.2 / 0.7 / 2.8	9.40	0.64	**14**
	Iter [11]	50.01	0.3 / 1.0 / 3.2	9.32	0.71	18
	Ours	**42.59**	**5.8 / 12.2 / 19.9**	**6.74**	**3.49**	247
1-1-0g + Prior	No	50.49	0.6 / 1.7 / 4.4	9.30	0.73	**9**
	Iter [11]	50.39	1.2 / 2.9 / 6.7	8.99	1.12	16
	Ours	**43.22**	**6.4 / 12.9 / 20.8**	**6.64**	**3.49**	273
Hybrid + Prior	No	43.19	0.8 / 2.8 / 9.2	8.66	1.80	**8**
	Iter [11]	42.85	2.6 / 7.3 / 15.2	7.64	2.73	17
	Ours	**42.72**	**7.9 / 16.2 / 24.8**	**5.89**	**4.21**	160

Table 1: **Local optimization on the ScanNet dataset [4].** We report median errors in degrees and AUCs over 10 runs. The rotation AUC is given at thresholds 5° / 10° / 20°.

# LO iterations	Rotation estimation		VP estimation		Time (ms)
	Err ↓	AUC ↑	Err ↓	AUC ↑	
0	4.59	26.4 / 46.1 / 64.9	4.30	5.95	**43**
10	1.16	69.3 / 83.7 / 90.9	1.31	8.75	46
20	1.18	70.0 / 84.0 / 91.0	1.29	8.79	55
100	**1.13**	**71.5 / 85.8 / 92.9**	**1.16**	**8.97**	100

Table 2: **Study on the number of LO iterations.** We report median errors in degrees and AUCs over 30 runs on the YorkUrban dataset [5] for the hybrid solver with prior gravity. The rotation AUC is given 5° / 10° / 20°.

	RANSAC scoring	Rotation estimation		VP estimation		Time (ms)
		Err ↓	AUC ↑	Err ↓	AUC ↑	
YorkUrban [5]	RANSAC	1.12	72.5 / 86.3 / 93.2	1.11	**9.08**	**79**
	MAGSAC	**1.11**	**72.9 / 86.5 / 93.3**	**1.10**	9.06	81
ScanNet [4]	RANSAC	25.83	20.3 / 28.9 / **36.6**	3.09	6.94	190
	MAGSAC	**25.75**	**20.6 / 29.0** / 36.5	**3.02**	**7.00**	**178**

Table 3: **RANSAC vs MAGSAC.** We report the rotation estimation AUC at 5°/10°/20° and VP estimation metrics on the YorkUrban [5] and ScanNet [4] datasets for the hybrid solver with IMU gravity.

[2] Daniel Barath, Jana Noskova, Maksym Ivashechkin, and Jiri Matas. MAGSAC++, a fast, reliable and accurate robust estimator. In *Conference on Computer Vision and Pattern Recognition*, 2020. 3

[3] Federico Camposeco, Andrea Cohen, Marc Pollefeys, and Torsten Sattler. Hybrid Camera Pose Estimation. In *IEEE Conf. Comput. Vis. Pattern Recog. (CVPR)*, 2018. 3

[4] Angela Dai, Angel X. Chang, Manolis Savva, Maciej Halber, Thomas Funkhouser, and Matthias Nießner. Scannet:

Richly-annotated 3d reconstructions of indoor scenes. In *IEEE Conf. Comput. Vis. Pattern Recog. (CVPR)*, 2017. 1, 3, 5, 7

[5] Patrick Denis, James H Elder, and Francisco J Estrada. Efficient edge-based methods for estimating manhattan frames in urban imagery. In *Eur. Conf. Comput. Vis. (ECCV)*, 2008. 5, 7

[6] Andreas Geiger, Philip Lenz, and Raquel Urtasun. Are we ready for autonomous driving? the kitti vision benchmark suite. In *IEEE Conf. Comput. Vis. Pattern Recog. (CVPR)*, 2012. 8

[7] Karel Lebeda, Jiri Matas, and Ondrej Chum. Fixing the Locally Optimized RANSAC. In *Brit. Mach. Vis. Conf. (BMVC)*, 2012. 3

[8] J. Lobo and J. Dias. Vision and inertial sensor cooperation using gravity as a vertical reference. *IEEE Trans. Pattern Anal. Mach. Intell. (PAMI)*, 25(12), 2003. 5

[9] Paul-Edouard Sarlin, Mihai Dusmanu, Johannes L. Schönberger, Pablo Speciale, Lukas Gruber, Viktor Larsson, Ondrej Miksik, and Marc Pollefeys. LaMAR: Benchmarking Localization and Mapping for Augmented Reality. In *Eur. Conf. Comput. Vis. (ECCV)*, 2022. 8

[10] Horst Wildenauer and Allan Hanbury. Robust camera self-calibration from monocular images of manhattan worlds. In *IEEE Conf. Comput. Vis. Pattern Recog. (CVPR)*, 2012. 3, 4, 5, 7, 8

[11] Lilian Zhang, Huimin Lu, Xiaoping Hu, and Reinhard Koch. Vanishing point estimation and line classification in a manhattan world with a unifying camera model. *Int. J. Comput. Vis. (IJCV)*, 117, 2015. 5
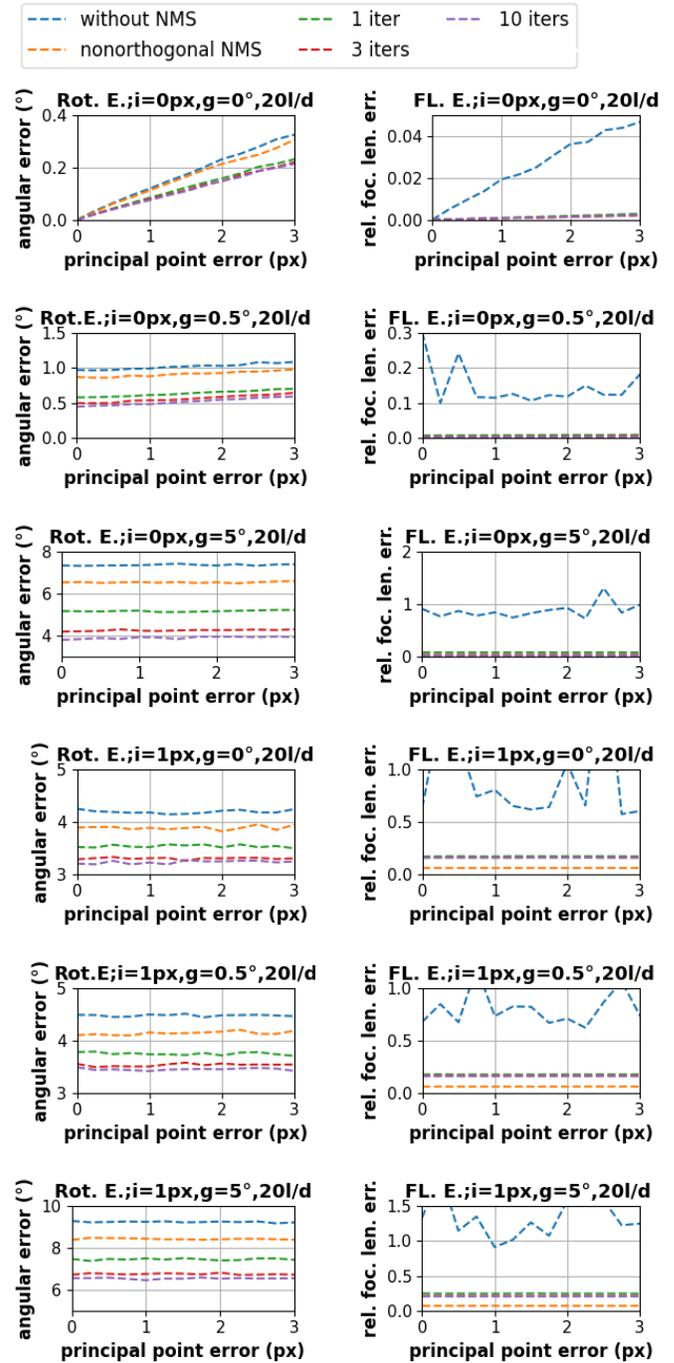
Figure 6: **Relation between the principal point noise and our non-minimal solvers.** The plots show the relation between the error of the principal point and the rotation and relative focal length error. We considered the following approaches: baseline (without NMS) , non orthogonal NMS as in the main paper, and linearized rotation with n iterations (n iters). The 1-1-0g solver was used for the initialization. The fixed noise std. is in the title (g - gravity, i - image).

| **2-1-1 solver [10]** | **Hybrid - Prior gravity** | **Hybrid - Ground truth gravity** |

Num inliers = 292 ; Rot error = 8.92   Num inliers = 353 ; Rot error = 5.38   Num inliers = 402 ; Rot error = 0.48

Num inliers = 138 ; Rot error = 6.81   Num inliers = 154 ; Rot error = 2.05   Num inliers = 160 ; Rot error = 1.43

Num inliers = 324 ; Rot error = 8.55   Num inliers = 488 ; Rot error = 0.48   Num inliers = 487 ; Rot error = 0.37

Num inliers = 24 ; Rot error = 32.68   Num inliers = 35 ; Rot error = 22.03   Num inliers = 62 ; Rot error = 2.67

Num inliers = 58 ; Rot error = 40.45   Num inliers = 51 ; Rot error = 33.76   Num inliers = 69 ; Rot error = 1.29
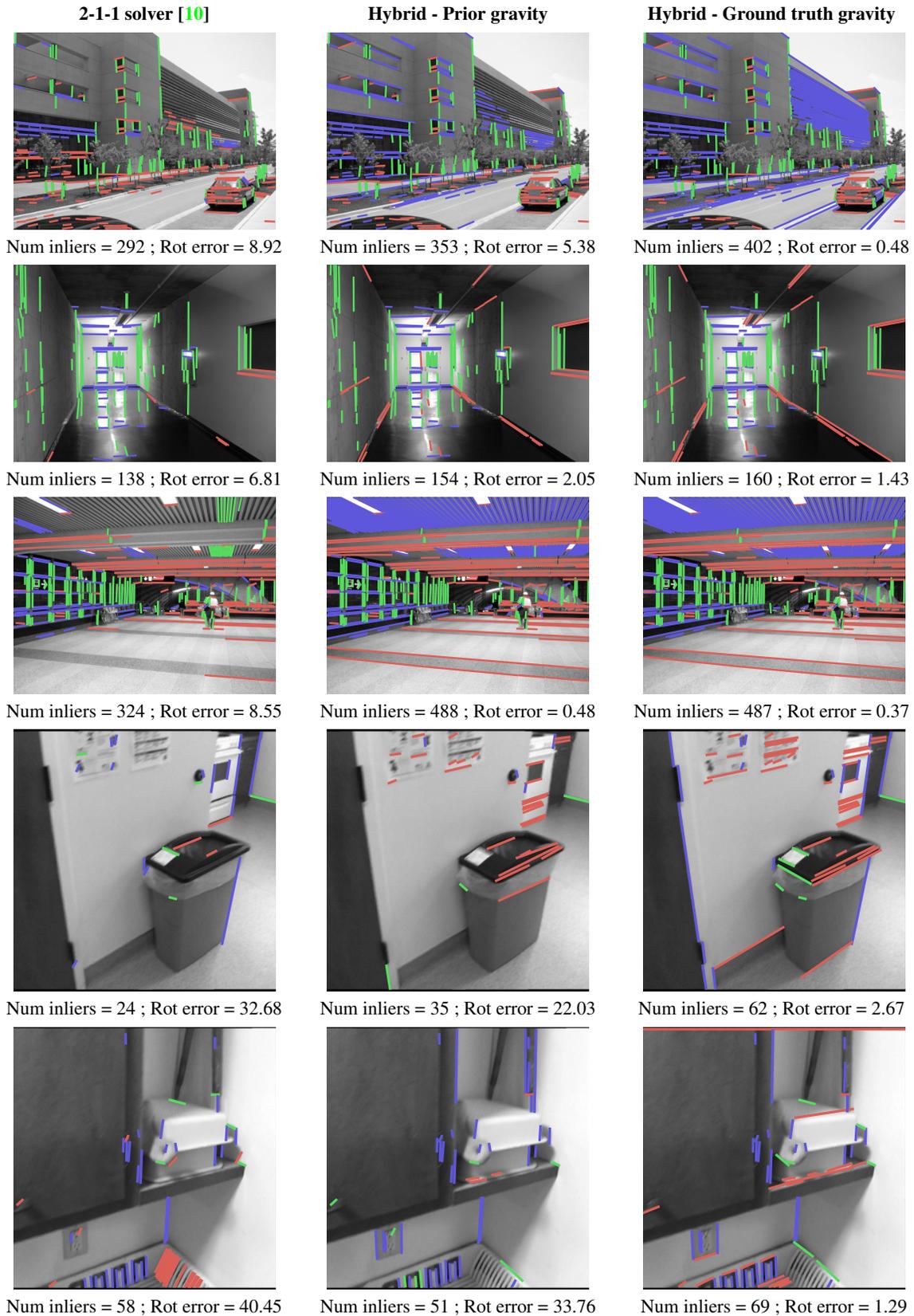
Figure 7: **Visualization of vanishing points.** We display the inlier lines with one color per vanishing point, for the 2-1-1 solver of [10], and our hybrid solver with either prior or ground truth gravity. The first three rows are from YorkUrban [5], and last two rows from ScanNet [4].

2-1-1 solver [10]   **Hybrid solver**

Rot error = 30.58° ; f error = 0.514   Rot error = 6.88° ; f error = 0.290

Rot error = 12.13° ; f error = 0.309   Rot error = 4.05° ; f error = 0.182

Rot error = 9.59° ; f error = 0.436   Rot error = 2.74° ; f error = 0.071

Rot error = 17.98° ; f error = 0.647   Rot error = 7.34° ; f error = 0.569

Rot error = 27.82° ; f error = 1.294   Rot error = 10.82° ; f error = 0.362
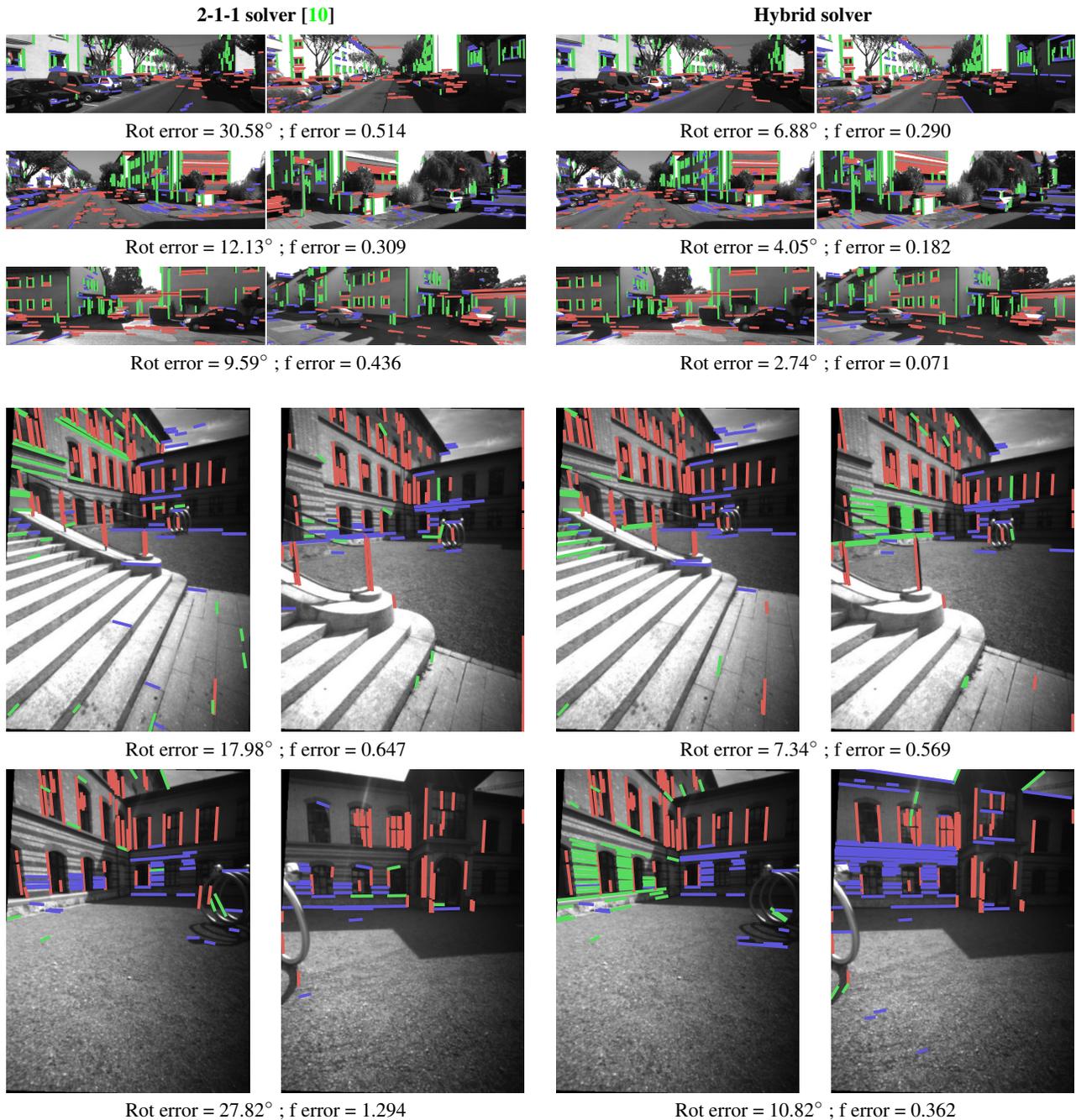
Figure 8: **Visualization of applications.** We display the inlier lines with one color per vanishing point, for the 2-1-1 solver of [10] (left), and our hybrid solver (right). The first three rows display pairs of images from an autonomous driving scenario on the KITTI dataset [6]. Assuming that the gravity is vertical already gives a very good prior to our solver and boost the performance on relative rotation estimation. The last two rows are from an augmented reality setup on the LaMAR dataset [9]. Head movements are often purely rotational, and the IMU information is crucial to obtain accurate pose estimates.