Vanishing Point Estimation in Uncalibrated Images with Prior Gravity Direction

Rémi Pautrat[1]   Shaohui Liu[1]   Petr Hruby[1]   Marc Pollefeys[1,2]   Daniel Barath[1]

[1] Department of Computer Science, ETH Zurich   [2] Microsoft Mixed Reality and AI Zurich lab

(a) 4-line solver [43]
Num inliers: **390**, Rotation error: **12.05°**

(b) Our 2-line solver with IMU gravity
Num inliers: **455**, Rotation error: **2.54°**

(c) Our hybrid RANSAC
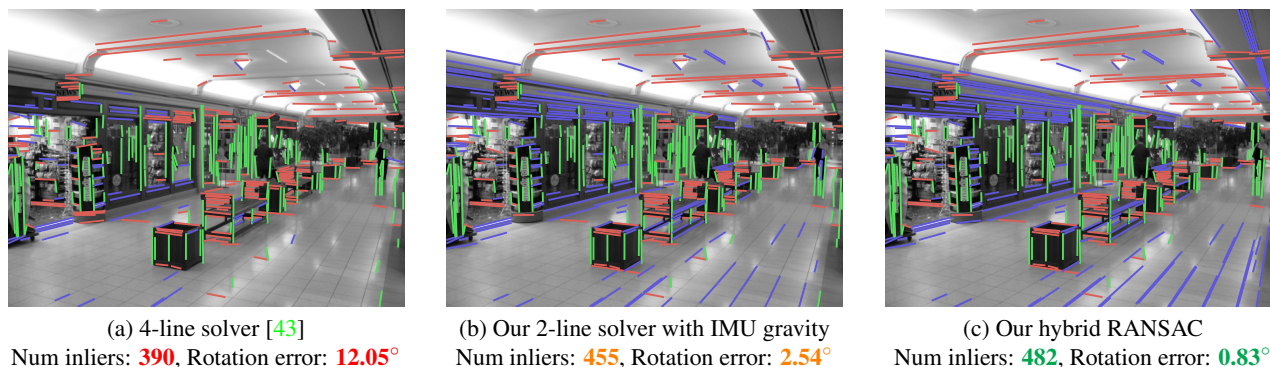Num inliers: **482**, Rotation error: **0.83°**

Figure 1: **Vanishing point (VP) estimation with gravity prior.** We propose two new 2-line solvers for VP estimation with uncalibrated cameras leveraging a known direction. Inlier lines of RANSAC are depicted with one color per assigned VP. Compared to previous work (a), our new solvers (b) and their combination in a hybrid RANSAC (c) are more accurate.

Abstract

We tackle the problem of estimating a Manhattan frame, i.e. three orthogonal vanishing points, and the unknown focal length of the camera, leveraging a prior vertical direction. The direction can come from an Inertial Measurement Unit that is a standard component of recent consumer devices, e.g., smartphones. We provide an exhaustive analysis of minimal line configurations and derive two new 2-line solvers, one of which does not suffer from singularities affecting existing solvers. Additionally, we design a new non-minimal method, running on an arbitrary number of lines, to boost the performance in local optimization. Combining all solvers in a hybrid robust estimator, our method achieves increased accuracy even with a rough prior. Experiments on synthetic and real-world datasets demonstrate the superior accuracy of our method compared to the state of the art, while having comparable runtimes. We further demonstrate the applicability of our solvers for relative rotation estimation. The code is available at https://github.com/cvg/VP-Estimation-with-Prior-Gravity.

1. Introduction

Projecting parallel lines of the world into a perspective image reveals a common intersection point: their vanishing point (VP). Estimating VPs from a single image is a long-standing task in vision, as they provide critical information about the 3D structure of the scene [39]. Vanishing points are relevant in many vision applications, such as camera rotation estimation [11, 13, 18], calibration [10], single-view reconstruction [22], wireframe parsing [35, 48].

Vanishing point estimation methods can be categorized into two main paradigms: unconstrained VP detection discovering an undefined number of VPs [5, 34, 45, 24], and approaches that make the Manhattan world assumption [11, 7, 46, 28], *i.e.*, there are three dominant and orthogonal directions in the world. We opt in this paper for the second option, assuming a Manhattan world. This scenario is particularly relevant in human-made environments, where most applications leveraging VPs take place. By making this assumption, we can directly estimate the three orthogonal directions, significantly reducing the search space and simplifying the robust estimation problem. Although it is possible to detect an unconstrained number of VPs and later orthogonalize them [13], directly looking for a set of 3 orthogonal VPs is more efficient and accurate in practice [7].

Vanishing point detection has been extensively studied in the past. We distinguish five main directions of research: exhaustive search methods [32, 36, 6, 31, 33], expectation-maximization-based [3, 13], optimization-based [8, 41], deep learning-based [4, 47, 29, 40] and RANSAC-based ones [2, 44, 43, 7, 46]. Due to their robustness to outliers, we adopt the strand of research based on RANSAC in this

paper [19]. To make RANSAC fast and accurate, minimal solvers are recommended in the model estimation step. For the Manhattan world assumption, the minimum number of lines to estimate the 3 VPs from a calibrated image is 3 [7]. In the uncalibrated case with unknown focal lengths and known principal point, 4 lines are needed to recover the 3 VPs and the focal length [43, 46]. Assuming the principal point to be in the center of the image is a common practice [43, 46, 33], which leads to accurate focal lengths [23].

In this paper, we focus on estimating three orthogonal VPs in an uncalibrated image, given a known direction. Specifically, we consider the case of a vertical prior (*i.e.*, gravity direction) but the same theory applies to other common directions, such as the horizon line. Nowadays, most commercial devices, including smartphones and AR glasses, come with built-in Inertial Measurement Units (IMU) measuring the gravity direction with good precision. Moreover, people tend to take pictures upright, providing a prior that vertical lines should roughly align with the gravity direction. This known direction is a rich signal for VP detection, as it gives one VP in the calibrated setting [7].

Assuming unknown focal lengths and known vertical direction is a practical problem with numerous real-world applications. For example, it is crucial for camera systems that are re-calibrating on-the-fly or that allow zooming in and out. An application for this task is autonomous driving, where the camera always stays upright, but its calibration may drift after several months of use, requiring an online calibration. Furthermore, tourist pictures found on social media are often taken roughly upright, without any available camera calibration. Finally, vanishing points provide strong constraints on the 3D rotation when calculating the relative pose of an image pair. This is particularly useful when the cameras undergo pure rotation, where traditional fundamental matrix-based solutions fail [20].

We propose two new minimal solvers that address line configurations not discussed in prior work [30]. The solvers estimate the camera rotation and focal length from a pair of lines. Importantly, one of the proposed solvers is not affected by a singularity that makes other methods unstable when dealing with perfectly upright images. To enable estimation from any number of lines, we also present a new non-minimal solver that can be used for local optimization in RANSAC. Moreover, we combine all the proposed and existing solvers in a hybrid RANSAC [9] framework, illustrated in Figure 1, that adaptively decides when to use the gravity prior. In summary, our contributions are as follows:

- We provide *all line configurations leading to minimal problems* for VP estimation given known gravity but unknown focal length.
- We derive *solutions for configurations not previously discussed in the literature*, one of which overcomes a common singularity of previous solvers.

- We propose *a new non-minimal solver* to boost performance in the local optimization of RANSAC.
- We combine all the existing configurations into a hybrid RANSAC framework to *increase robustness to situations with only a rough gravity prior.*

The proposed methods are evaluated on large-scale and real-world datasets, demonstrating superior accuracy compared to the state of the art when the gravity direction is known. When we are only given a rough prior, the hybrid solution efficiently uses all solvers together to obtain results more accurately than by previous solvers. Furthermore, we demonstrate that such an approach is highly beneficial for relative rotation estimation in a pair of cameras.

2. Uncalibrated Vanishing Point Estimation

In this work, we make the Manhattan world assumption. We are looking for 3 orthogonal vanishing points from a single image with a pinhole camera model. While the principal point is assumed to be in the center of the image, we consider the focal length unknown, but the gravity direction to be known (*e.g.*, from an IMU) or we have a rough prior on it (*e.g.*, the image is upright). In this section, we first discuss existing work [30] (2.1) and introduce two new solvers (2.2 and 2.3). Then, we describe a new non-minimal solver in Section 2.4. Finally, a hybrid RANSAC approach is designed to combine all solvers in Section 2.5.

Theoretical Background. A vanishing point is an intersection of 2D projections of parallel 3D lines. The homogeneous coordinates of the VP in the camera are

$$\mathbf{v}_i = \mathbf{K}\mathbf{R}\mathbf{d}_i, \tag{1}$$

where $\mathbf{K} \in \mathbb{R}^3$ is the matrix of camera intrinsics, $\mathbf{R} \in$ SO(3) is the rotation of the camera, and $\mathbf{d}_i \in \mathbb{R}^3$ is the direction of the line in 3D in the world coordinate system.

In this paper, we assume known principal point and square pixels. The intrinsic matrix can be simplified as

$$\mathbf{K} = \mathrm{diag}(f, f, 1) \in \mathbb{R}^{3 \times 3}, \tag{2}$$

where $f \in \mathbb{R}$ is the focal length. We assume that the directions $\mathbf{d}_1$, $\mathbf{d}_2$, $\mathbf{d}_3$ are mutually orthogonal, *i.e.*, we are in a Manhattan world [11]. Directions $\mathbf{d}'_1$, $\mathbf{d}'_2$, $\mathbf{d}'_3$ in the camera coordinate system are given by $\mathbf{d}'_i = \mathbf{R}\mathbf{d}_i$, $i \in \{1, 2, 3\}$.

If we set the basis of the world coordinate system to be equal to directions $\mathbf{d}_1$, $\mathbf{d}_2$, $\mathbf{d}_3$ (*i.e.*, $[\mathbf{d}_1\ \mathbf{d}_2\ \mathbf{d}_3] = I$), then the rotation $\mathbf{R}$ between the world and the camera coordinate system equals to $\mathbf{R} = [\mathbf{d}'_1\ \mathbf{d}'_2\ \mathbf{d}'_3]$.

2.1. Uncalibrated 2-0-0$_g$ Solver [30]

In this section, we recap the minimal solver from [30], estimating the three orthogonal vanishing points from two lines $\mathbf{l}_1$, $\mathbf{l}_2$ both being associated with the same VP (third

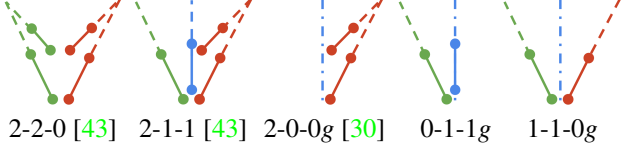| 2-2-0 [43] | 2-1-1 [43] | 2-0-0g [30] | 0-1-1g | 1-1-0g |

Figure 2: **Minimal solvers in the uncalibrated case.** The first two drawings display the line configuration of the 4-line solvers [43], where colors correspond to VPs. The last three show the existing [30] and proposed 2-line solvers with known gravity (in dashed blue). The notation $x_1-x_2-x_3$ means x_i lines are associated with VP i.

configuration in Figure 2). In this case, lines $\mathbf{l}_1$, $\mathbf{l}_2$ are projections of parallel lines in direction $\mathbf{d}'_2$. We further assume that the vertical direction $\mathbf{d}'_1$ is known. We calculate vanishing point $\mathbf{v}_2$ as the cross product of the two lines as $\mathbf{v}_2 = \mathbf{l}_1 \times \mathbf{l}_2$. We can express the second direction as $\mathbf{d}'_2 = \mathbf{K}^{-1}\mathbf{v}_2$ and use the constraint that $\mathbf{d}'_1$ and $\mathbf{d}'_2$ are orthogonal as follows:

$$\mathbf{d}'_1{}^{\mathsf{T}}\mathbf{d}'_2 = \mathbf{d}'_1{}^{\mathsf{T}}\mathbf{K}^{-1}\mathbf{v}_2 = 0. \tag{3}$$

Matrix $\mathbf{K}^{-1}$ can be easily computed, yielding:

$$(\mathbf{d}'_1(0)\mathbf{v}_2(0) + \mathbf{d}'_1(1)\mathbf{v}_2(1))\frac{1}{f} + \mathbf{d}'_1(2)\mathbf{v}_2(2) = 0. \tag{4}$$

Finally, the focal length f is obtained as:

$$f = -\frac{\mathbf{d}'_1(0)\mathbf{v}_2(0) + \mathbf{d}'_1(1)\mathbf{v}_2(1)}{\mathbf{d}'_1(2)\mathbf{v}_2(2)}. \tag{5}$$

This equation implies two potential singularities when the denominator equals 0. The first one happens when $\mathbf{v}_2(2) = 0$, *i.e.* lines $\mathbf{l}_1$ and $\mathbf{l}_2$ are parallel. This can be easily avoided by ignoring samples of parallel lines (at least one of the 2 remaining VPs will have lines that are not parallel). The second singularity happens if $\mathbf{d}'_1(2) = 0$, *i.e.* the gravity has no z component, which happens in datasets where the vertical direction is along the y-axis, for example.

After the focal length f is found, we build the camera matrix $\mathbf{K}$, compute $\mathbf{d}'_2 = \mathbf{K}^{-1}\mathbf{v}_2$, and $\mathbf{d}'_3 = \mathbf{d}'_1 \times \mathbf{d}'_2$. Then, we build the rotation matrix as $\mathbf{R} = [\mathbf{d}'_1 \; \mathbf{d}'_2 \; \mathbf{d}'_3]$.

2.2. Uncalibrated 0-1-1g Solver

In this section, we derive a minimal solver to estimate the three orthogonal vanishing points and the focal length from a line $\mathbf{l}_1$ stemming from the known vertical direction $\mathbf{d}'_1$ and a line $\mathbf{l}_2$ stemming from a horizontal direction $\mathbf{d}'_2$. This is the fourth configuration in Figure 2.

Given (1) and line $\mathbf{l}_1$ stemming from the vertical direction, we have $\mathbf{l}_1^{\mathsf{T}}\mathbf{v}_1 = \mathbf{l}_1^{\mathsf{T}}\mathbf{K}\mathbf{d}'_1 = 0$ that can be written as:

$$(\mathbf{l}_1(0)\mathbf{d}'_1(0) + \mathbf{l}_1(1)\mathbf{d}'_1(1))f + \mathbf{l}_1(2)\mathbf{d}'_1(2) = 0. \tag{6}$$

The focal length f is then obtained as:

$$f = -\frac{\mathbf{l}_1(2)\mathbf{d}'_1(2)}{\mathbf{l}_1(0)\mathbf{d}'_1(0) + \mathbf{l}_1(1)\mathbf{d}'_1(1)}. \tag{7}$$

This solver also suffers from a singularity if the denominator is 0. After finding the focal length f, we build the camera matrix $\mathbf{K}$. The horizontal VP $\mathbf{v}_2$ equals to $\mathbf{v}_2 = \mathbf{K}\mathbf{d}'_2$. As $\mathbf{l}_2$ stems from $\mathbf{d}'_2$, $\mathbf{l}_2^{\mathsf{T}}\mathbf{v}_2 = \mathbf{l}_2^{\mathsf{T}}\mathbf{K}\mathbf{d}'_2 = 0$ holds. There also holds $\mathbf{d}'_1{}^{\mathsf{T}}\mathbf{d}'_2 = 0$. Thus, we calculate $\mathbf{d}'_2 = \mathbf{d}'_1 \times (\mathbf{K}^{\mathsf{T}}\mathbf{l}_2)$. Then, we calculate $\mathbf{d}'_3 = \mathbf{d}'_1 \times \mathbf{d}'_2$, normalize all 3 directions, and build the rotation matrix $\mathbf{R} = [\mathbf{d}'_1 \; \mathbf{d}'_2 \; \mathbf{d}'_3]$.

2.3. Uncalibrated 1-1-0g Solver

In this section, we derive a minimal solver to estimate the three orthogonal vanishing points and focal length from two lines $\mathbf{l}_1$, $\mathbf{l}_2$, stemming from two mutually orthogonal horizontal directions $\mathbf{d}'_2$ and $\mathbf{d}'_3$, assuming a known vertical direction $\mathbf{d}'_1$. This is the fifth configuration in Figure 2.

Since $\mathbf{d}'_2 \perp \mathbf{d}'_1$, $\mathbf{d}'_3 \perp \mathbf{d}'_1$, directions $\mathbf{d}'_2$, $\mathbf{d}'_3$ live in a 2D subspace of $\mathbb{R}^3$ that is orthogonal to $\mathbf{d}'_1$. This subspace has an orthogonal basis $\{\mathbf{b}_1, \mathbf{b}_2\}$ such that $\mathbf{b}_1^{\mathsf{T}}\mathbf{b}_1 = \mathbf{b}_2^{\mathsf{T}}\mathbf{b}_2 = 1, \mathbf{b}_1^{\mathsf{T}}\mathbf{b}_2 = 0$. Let $\mathbf{u} \in \mathbb{R}^3$ be an arbitrary generic vector. Then, this basis can be estimated from $\mathbf{d}'_1$ as follows:

$$\mathbf{b}'_1 = \mathbf{d}'_1 \times \mathbf{u}; \; \mathbf{b}_1 = \frac{1}{||\mathbf{b}'_1||}\mathbf{b}'_1,$$
$$\mathbf{b}'_2 = \mathbf{d}'_1 \times \mathbf{b}'_1; \; \mathbf{b}_2 = \frac{1}{||\mathbf{b}'_2||}\mathbf{b}'_2. \tag{8}$$

Since $\mathbf{d}'_2 \perp \mathbf{d}'_3$, the directions $\mathbf{d}'_2, \mathbf{d}'_3$ can be obtained by rotating the basis elements $\mathbf{b}_1, \mathbf{b}_2$ by an unknown angle φ. They can be expressed as follows:

$$\mathbf{d}'_2 = \cos\varphi\mathbf{b}_1 - \sin\varphi\mathbf{b}_2, \; \mathbf{d}'_3 = \sin\varphi\mathbf{b}_1 + \cos\varphi\mathbf{b}_2. \tag{9}$$

Then, we can get the vanishing points as $\mathbf{v}_2 = \mathbf{K}\mathbf{d}'_2$ and $\mathbf{v}_3 = \mathbf{K}\mathbf{d}'_3$. The fact that $\mathbf{v}_2$ lies on $\mathbf{l}_1$ and $\mathbf{v}_3$ lies on $\mathbf{l}_2$ imposes the following two constraints on f and φ as

$$\mathbf{l}_1^{\mathsf{T}}\mathbf{K}\mathbf{d}'_2 = \cos\varphi\mathbf{l}_1^{\mathsf{T}}\mathbf{K}\mathbf{b}_1 - \sin\varphi\mathbf{l}_1^{\mathsf{T}}\mathbf{K}\mathbf{b}_2 = 0,$$
$$\mathbf{l}_2^{\mathsf{T}}\mathbf{K}\mathbf{d}'_3 = \sin\varphi\mathbf{l}_2^{\mathsf{T}}\mathbf{K}\mathbf{b}_1 + \cos\varphi\mathbf{l}_2^{\mathsf{T}}\mathbf{K}\mathbf{b}_2 = 0. \tag{10}$$

We replace the goniometric functions $\cos\varphi, \sin\varphi$ by $\frac{1-t^2}{1+t^2}$, $\frac{2t}{1+t^2}$, and multiply both equations by $1 + t^2$ to obtain an equivalent system of polynomial equations as

$$(1 - t^2)\mathbf{l}_1^{\mathsf{T}}\mathbf{K}\mathbf{b}_1 - 2t\mathbf{l}_1^{\mathsf{T}}\mathbf{K}\mathbf{b}_2 = 0,$$
$$2t\mathbf{l}_2^{\mathsf{T}}\mathbf{K}\mathbf{b}_1 + (1 - t^2)\mathbf{l}_2^{\mathsf{T}}\mathbf{K}\mathbf{b}_2 = 0. \tag{11}$$

System (11) is linear in f and quadratic in t. This gives us two ways to eliminate variables. In the proposed solver, we first eliminate t, and then solve for f, which is shown in the following paragraph. An alternative approach is to

eliminate the equations in reverse order, which we derive in the supplementary material. The first option performed better in our experiments, so we only use this one in the following. After we obtain f and t, we build $\mathbf{K}$ according to (2), $\cos\varphi$ and $\sin\varphi$ as $\frac{1-t^2}{1+t^2}$, $\frac{2t}{1+t^2}$. Finally, we compute $\mathbf{d}_2'$, $\mathbf{d}_3'$ according to (9) and $\mathbf{R}$ as $[\mathbf{d}_1'\ \mathbf{d}_2'\ \mathbf{d}_3']$.

Variable Elimination. In this section, we discuss how to solve (11) by first eliminating t and by solving for f. The equations (11) are quadratic in t. Let us use notations

$$
\begin{aligned}
\delta_1 &= \mathbf{l}_1(0)\mathbf{b}_1(0) + \mathbf{l}_1(1)\mathbf{b}_1(1), \ \delta_2 = \mathbf{l}_1(2)\mathbf{b}_1(2), \\
\delta_3 &= \mathbf{l}_1(0)\mathbf{b}_2(0) + \mathbf{l}_1(1)\mathbf{b}_2(1), \ \delta_4 = \mathbf{l}_1(2)\mathbf{b}_2(2), \\
\delta_5 &= \mathbf{l}_2(0)\mathbf{b}_2(0) + \mathbf{l}_2(1)\mathbf{b}_2(1), \ \delta_6 = \mathbf{l}_2(2)\mathbf{b}_2(2), \\
\delta_7 &= \mathbf{l}_2(0)\mathbf{b}_1(0) + \mathbf{l}_2(1)\mathbf{b}_1(1), \ \delta_8 = \mathbf{l}_2(2)\mathbf{b}_1(2).
\end{aligned}
\tag{12}
$$

Then, the equations (11) can be written as

$$
\begin{aligned}
(1-t^2)(f\delta_1 + \delta_2) - 2t(f\delta_3 + \delta_4) &= 0, \\
(1-t^2)(f\delta_5 + \delta_6) + 2t(f\delta_7 + \delta_8) &= 0.
\end{aligned}
\tag{13}
$$

Multiplying the first equation by $f\delta_5 + \delta_6$ and the second one by $f\delta_1 + \delta_2$, then subtracting them, we obtain

$$
2t[(f\delta_1 + \delta_2)(f\delta_7 + \delta_8) + (f\delta_3 + \delta_4)(f\delta_5 + \delta_6)] = 0. \tag{14}
$$

If $t \neq 0$, then the second term must be 0. If $t = 0$, (13) yields $f\delta_1 + \delta_2 = f\delta_5 + \delta_6 = 0$, and so the second term is also 0. Thus, we have in both cases:

$$
(f\delta_1 + \delta_2)(f\delta_7 + \delta_8) + (f\delta_3 + \delta_4)(f\delta_5 + \delta_6) = 0. \tag{15}
$$

This is a univariate quadratic equation, so we obtain its roots by a closed-form formula. We only keep the positive values of f and finally obtain t as common roots of (11).

2.4. Non-Minimal Solver

Here, we are going to discuss how to compute vanishing points corresponding to 3 directions from a larger-than-minimal sample. This step is essential to obtain accurate results, *e.g.*, when refitting on all inliers after RANSAC finishes or in the local optimization procedure.

Suppose that we are given three vanishing points $\mathbf{v}_1$, $\mathbf{v}_2$, $\mathbf{v}_3$, either from the previously introduced minimal solvers, or from another minimal solver that does not require known gravity. Let $\mathcal{L}_1, \mathcal{L}_2, \mathcal{L}_3$ be three sets of lines, where each set $\mathcal{L}_i, i \in \{1,2,3\}$ contains n_i inliers of vanishing point $\mathbf{v}_i$. First, we 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 (we give the exact derivation in the supplementary material).

Next, we estimate the focal length f using these updated vanishing points. For every pair of vanishing points $\mathbf{v}_i, \mathbf{v}_j$, and (1) and (3), $\mathbf{d}_i^\mathsf{T}\mathbf{d}_j = \mathbf{v}_i^\mathsf{T}(\mathbf{K}^{-1})^\mathsf{T}\mathbf{K}^{-1}\mathbf{v}_j = 0$ holds. This constraint can be rewritten as:

$$
-\mathbf{v}_i(2)\mathbf{v}_j(2)f^2 = \mathbf{v}_i(0)\mathbf{v}_j(0) + \mathbf{v}_i(1)\mathbf{v}_j(1), \tag{16}
$$

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. We use the LSQ method to find f^2 (see the supplementary material for the full derivation). Since f can not be negative, we take $f = +\sqrt{f^2}$.

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 $\mathbf{D} = \mathbf{U}\mathbf{S}\mathbf{V}^\mathsf{T}$. We take the rows of matrix $\mathbf{R} = \mathbf{U}\mathbf{V}^\mathsf{T}$ as the refined calibrated vanishing points. Note that the VPs are further refined with a least square optimization described in Sec. 3.3.

2.5. Hybrid Robust Estimation

At this point, we have five minimal solvers for uncalibrated vanishing point estimation: two 4-line solvers without prior [43] and three solvers with known gravity direction. As the best solver for each problem depends on the accuracy of the prior gravity, we employ all the solvers in a hybrid RANSAC framework to combine the advantages of both worlds. At each iteration of RANSAC, we first sample a minimal solver with respect to a probability distribution, which is computed with the prior distribution and the inlier ratio ϵ. Specifically, we multiply the corresponding prior probability by ϵ^2 for the 2-line solvers with gravity direction and ϵ^4 for the 4-line solvers without prior, followed by normalization with the sum of the five probabilities. Then, similarly as in RANSAC, we run the solver on a randomly sampled minimal set, score the models, and apply our proposed local optimization when a new best model is found. The termination criterion is adaptively determined for each solver as in [9], depending on the inlier ratio and the predefined confidence parameter. In our experiment, we set the prior probability to be uniform across the five solvers.

3. Experiments

3.1. Synthetic Experiments

In order to test our solvers in a fully controlled synthetic environment, we generate instances of the problem we wish to solve, together with the ground truth solution.

Solver times. The theoretical run-times of the solvers inside RANSAC are plotted as a function of the outlier ratio in Fig. 3. It is calculated as the actual time of each solver, implemented in C++, multiplied by the theoretical number of iterations RANSAC has to perform on a particular outlier level to achieve 0.99 confidence. The average wall-clock times of the solvers are $0.09\mu s$ (2-2-0), $0.69\mu s$ (2-1-1), $0.09\mu s$ (2-0-0g), $0.08\mu s$ (0-1-1g), and $0.17\mu s$ (1-1-0g). The proposed solvers are significantly faster inside RANSAC, mainly because they require fewer iterations.

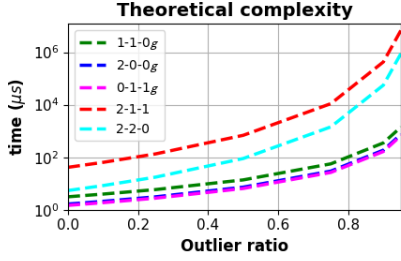**Numerical stability.** We generate a random rotation matrix

Figure 3: **Theoretical RANSAC run-time**, in μs, for solvers 2-2-0, 2-1-1 [43], 2-0-0g [30], and the proposed 1-1-0g, and 0-1-1g, calculated by multiplying the solver time by the theoretical number of RANSAC iterations at 0.99 conf.
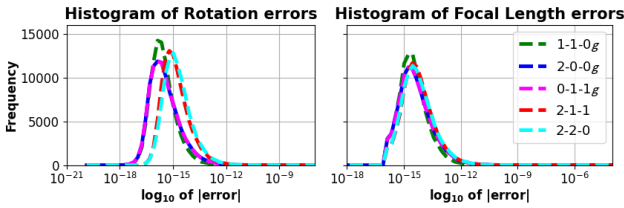

Figure 4: **Stability study.** The frequencies (100k runs) of the $\log_{10}$ errors in rotation and focal length estimated by solvers 2-2-0, 2-1-1 [43], 2-0-0g [30], 1-1-0g, and 0-1-1g.

$\mathbf{R}_{gt} = [\mathbf{d}_1', \mathbf{d}_2', \mathbf{d}_3']$, and a random focal length f_{gt} from a uniform distribution between 100 and 2000 pixels. We take the vanishing points $\mathbf{v}_1$, $\mathbf{v}_2$, $\mathbf{v}_3$ as the columns of matrix $\mathbf{K}_{gt}\mathbf{R}_{gt}$. The vertical direction is the first column $\mathbf{d}_1'$ of $\mathbf{R}_{gt}$. To generate a 3D line in direction i, we sample the first endpoint $\mathbf{X}_A$ from the Gaussian distribution with mean $[0\ 0\ 5]^{\mathsf{T}}$ and standard deviation set to 1. We sample a parameter $\lambda \in \mathbb{R}$ from the normalized Gaussian distribution, and take the second endpoint as $\mathbf{X}_B = \mathbf{X}_A + \lambda \mathbf{d}_i'$. Then, we project both endpoints as $\mathbf{x}_A = \mathbf{K}_{gt}\mathbf{X}_A$, $\mathbf{x}_B = \mathbf{K}_{gt}\mathbf{X}_B$, and obtain the homogeneous coordinate of the line as $\mathbf{l} = \mathbf{x}_A \times \mathbf{x}_B$.

Let $\mathbf{R}_{est}$, f_{est} be the rotation and focal length estimated by the minimal solver. We compute the rotation error as the angle of the rotation represented as $\mathbf{R}_{est}^{\mathsf{T}}\mathbf{R}_{gt}$, and the focal length error as $|f_{est} - f_{gt}|$. We generated $n = 100000$ random problem instances and ran the solvers on the noiseless samples. Figure 4 shows histograms of $\log_{10}$ rotation and focal length errors. The plots show that all our solvers are very stable – there is no peak close to 10^0.

Tests with noise. To evaluate the robustness w.r.t. the noise in the input, we generate minimal problems similarly as in the previous section. We perturb the gravity direction and the endpoints using Gaussian noise with standard deviation σ_i To perturb the gravity direction $\mathbf{d}_1'$, we sample axis $\mathbf{a}_g$ uniformly from a unit sphere, and an angle α_g from a Gaussian distribution with zero mean and standard deviation σ_g.
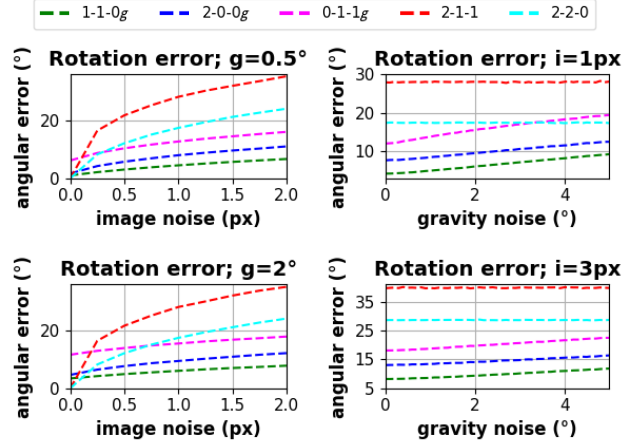

Figure 5: **Noise study.** The avg. (100k runs) rotation error of solvers 2-1-1 [43], 2-2-0 [43], 2-0-0g [30], 1-1-0g, and 0-1-1g as a function of the image (left) and gravity noise (right). The noise std. is in the title (g – gravity, i – image).

We build a rotation matrix $\mathbf{R}_g$ with angle α_g and axis $\mathbf{a}_g$, and get the noisy gravity direction as $\mathbf{R}_g \mathbf{d}_1'$.

Figure 5 shows the average rotation and focal length errors on different levels of σ_i and σ_g. In the plots on the left, the proposed solvers are significantly less sensitive to the image noise than the 4-line ones. In the right plots, they are more accurate than the 4-line solvers, even with some noise on the gravity prior. Note that accelerometers used in cars and modern smartphones have noise levels around $0.06°$ (or expensive "good" ones have less than $0.02°$) [16]. Additional experiments are shown in the supp. material.

3.2. Real-World Experiments Setup

Datasets. We evaluate our methods on two real-world datasets. The first one is the York Urban DB dataset [14], a classical benchmark for vanishing point estimation. It consists of 102 images of indoor and outdoor scenes, annotated with 3 orthogonal VPs. We split it into 51 images for tuning hyper-parameters and 51 for testing. When using our solvers with known gravity, we consider two cases: the gravity is given by the ground truth vertical VP and is, consequently, very accurate (we label it as *IMU*), or we only use a prior gravity assuming that the images are upright (denoted as *Prior*). In the latter case, the average gravity error w.r.t. the ground truth is of 3.9 degrees, showing that most pictures are indeed close to being upright.

The second dataset that we consider is ScanNet [12], a large-scale indoor RGB-D dataset. We follow the evaluation of NeurVPS [47], by finding the three main orthogonal directions aligning with the most surface normals in each image. For each of the 100 scenes of the official ScanNet validation set, we randomly sample 10 images for testing

and one for validation, resulting in a test set of 1000 images and a validation set of 100 images to tune the hyperparameters. We again choose the *IMU* gravity to be the VP closest to the vertical direction, and use the prior that images should be upright. In this case, the average error of the prior w.r.t. the GT gravity is much larger, with 24.8° error.

Metrics. Since the 3 detected VPs are orthogonal, we can directly derive the corresponding camera rotation. Thus, we report several metrics for rotation estimation and vanishing point detection. The rotation error is the distance between the quaternions of the GT rotation q_{gt} and the predicted quaternions q in degrees: $R_{err} = \arccos(2(q \cdot q_{gt})^2 - 1)$. The rotation AUC measures the area under the recall curve at 5 / 10 / 20 degrees. The VP error is the average of the angular errors in degrees between the predicted VPs and the GT ones. Finally, we regularly sample 20 thresholds between 0 and 10 degrees, plot the curve of the percentage of images successfully recovering VPs that are close enough to the GT VPs up to these thresholds, and compute the Area Under the Curve to get the VP AUC, as in [24].

Baselines. For all following experiments, we apply the widely used Line Segment Detector [42] to extract line segments from images. All RANSAC methods are run with a minimum of 1000 iterations and we report the median results after 30 runs on York Urban and 10 runs on ScanNet. We compare our minimal solvers with the two previously existing 4-line solvers for uncalibrated cameras [43]: 2-2-0 and 2-1-1 (first two configurations in Figure 2), and the 2-line solver 2-0-0g of [30]. We show the individual results of our two new 2-line solvers 1-1-0g and 0-1-1g, as well as of our hybrid RANSAC combining all five solvers. Figure 6 illustrates the extracted vanishing points of different methods on the challenging ScanNet dataset [12].

3.3. Impact of Local Optimization

We start by evaluating the performance gain from our proposed non-minimal solver and the resulting local optimization (LO) applied in RANSAC whenever a new best model is found. We follow the LO-RANSAC implementation of [25], by first extracting inliers of the best model returned by one of the considered minimal solvers and then running 100 iterations of LO on random subsets of these inliers. Each LO step generates a new model from the non-minimal solver, refines it with a least squares (LS) optimization fitting the VP to the inlier lines, and stores the new model if it improves the previous best model. The LS optimization minimizes the sum of squared distances between each VP and its corresponding inlier lines, and is implemented in Ceres [1] with the Levenberg–Marquardt algorithm [26]. We compare our LO with the Iter method proposed in [46] using the exact same pipeline as ours, but replacing the model of our non minimal solver with the best model returned by the minimal solvers instead.

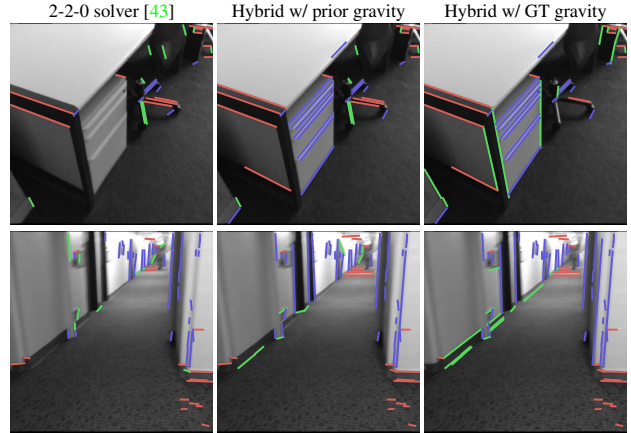| 2-2-0 solver [43] | Hybrid w/ prior gravity | Hybrid w/ GT gravity |

Figure 6: **Visual examples.** We provide examples of VP estimation on ScanNet [12] by the 2-2-0 solver [43], our hybrid RANSAC with prior gravity, and with GT gravity. Lines are colored according to their associated VP. Even with a rough prior, our hybrid solver obtains more accurate results than the baseline.

We show the results on York Urban in Table 1 without LO, with Iter, and our proposed LO. We only show the results with prior gravity, as it is where the LO brings the most significant contribution. It can be seen from the table that not using any LO yields very poor results, thus highlighting the importance of LO. While Iter brings a large improvement, our proposed LO obtains the best results in all metrics by a very large margin. This overall large boost of performance of our LO comes at the price of an increased computation time – still close to real-time. It is possible to reduce the run-time by decreasing the number of LO iterations, and trading performance for efficiency.

3.4. Comparison to Previous Works

We compare our new solvers to the baselines on York Urban DB [14] in Table 2 and on ScanNet [12] in Table 3, where our LO has been applied to all methods. Several conclusions can be drawn from these two tables. First, our two new solvers significantly outperform the baselines when IMU information is available, with, for example, an improvement of up to 27% in median rotation error over the best baselines on York Urban, and up to 40% on ScanNet.

Second, our solvers using the prior gravity obtain similar results as the baselines without gravity information, showing that even with a rough estimate of the gravity, it is not detrimental to use our solvers. This is especially true for ScanNet, where the prior gravity is particularly noisy with 24.8° of average error w.r.t. to the GT gravity. The better performance of the 1-1-0g solver is mostly due to the fact that it is the only gravity-based solver that does not suffer from a singularity when the vector $[0, -1, 0]$ is used as grav-

Solver	LO	Rotation estimation		VP estimation		Time (ms)
		Err ↓	AUC ↑	Err ↓	AUC ↑	
2-2-0 [43] No gravity	No	6.04	9.7 / 36.0 / 63.8	4.96	5.35	**65**
	Iter [46]	1.77	58.3 / 76.6 / 87.9	1.87	7.85	78
	Ours	**1.51**	**67.6 / 83.6 / 91.8**	**1.34**	**8.93**	119
2-1-1 [43] No gravity	No	6.87	7.9 / 32.1 / 58.7	5.23	5.14	**40**
	Iter [46]	2.07	53.0 / 74.9 / 87.3	1.99	7.57	57
	Ours	**1.52**	**68.3 / 83.2 / 90.6**	**1.35**	**8.88**	103
2-0-0g [30] + Prior	No	25.09	2.3 / 5.0 / 12.0	7.63	2.47	**68**
	Iter [46]	25.08	2.3 / 5.0 / 12.0	7.61	2.48	72
	Ours	**1.25**	**64.3 / 78.0 / 86.3**	**1.79**	**8.43**	121
0-1-1g + Prior	No	25.18	2.3 / 5.1 / 14.1	7.60	2.68	**40**
	Iter [46]	24.95	5.0 / 9.6 / 16.3	7.35	2.76	47
	Ours	**1.37**	**68.4 / 84.1 / 92.1**	**1.28**	**8.80**	142
1-1-0g + Prior	No	4.48	26.0 / 44.2 / 59.2	4.79	5.35	**44**
	Iter [46]	1.63	54.5 / 69.3 / 77.6	2.60	7.62	52
	Ours	**1.20**	**70.1 / 84.1 / 91.1**	**1.29**	**8.87**	84
Hybrid + Prior	No	4.59	26.4 / 46.1 / 64.9	4.30	5.95	**43**
	Iter [46]	1.51	61.7 / 79.0 / 88.5	1.69	8.19	52
	Ours	**1.13**	**71.5 / 85.8 / 92.9**	**1.16**	**8.97**	100

Table 1: **Local optimization on the YorkUrban dataset [14].** We report median errors in degrees and AUCs over 30 runs. The rotation AUC is given 5° / 10° / 20°.

ity vector. To prevent the division by zero in Equations 5 and 7, we perturb the prior with small random noise.

Last but not least, the hybrid approach is always at least as good as all previous solvers, and it obtains the best results when used with the prior gravity, with for instance an improvement of 12% in rotation AUC on ScanNet. This can be explained by the fact that whenever the prior is close enough to the GT gravity, our proposed solvers will yield their best performance, while when the prior becomes too noisy, the previous 2-2-0 and 2-1-1 solvers can help improving the performance.

In brief, the proposed solvers lead to significantly better accuracy when the gravity is accurate, and are not worse than the baselines when it is inaccurate. Hybrid RANSAC leads to the best results when having only a rough gravity prior, *i.e.*, the image is upright.

3.5. Focal Length Estimation

Similarly as for the previous uncalibrated solvers [43], our method provides an estimate of the focal length in addition to the predicted VPs. We study here the quality of the prediction, by computing the relative error between the predicted and GT focal lengths. The results are reported in Table 4 for the case with known gravity from IMU without LO, with Iter, and our LO. Our solvers and their hybrid version bring a significant improvement over the baselines without known gravity on both York Urban and ScanNet. Using our LO instead of Iter also generally improves the quality of the predicted focal length, in particular on York Urban. Thus, this experiment shows that the focal length can be extracted from VPs, and the knowledge of the grav-

Solver	Gravity	Rotation estimation		VP estimation		Time (ms)
		Err ↓	AUC ↑	Err ↓	AUC ↑	
2-2-0 [43]	No	1.51	67.6 / 83.6 / 91.8	1.34	8.93	119
2-1-1 [43]		1.52	68.3 / 83.2 / 90.6	1.35	8.88	103
2-0-0g [30]	Prior	1.25	64.3 / 78.0 / 86.3	1.79	8.43	121
0-1-1g		1.37	68.4 / 84.1 / 92.1	1.28	8.80	142
1-1-0g		1.20	70.1 / 84.1 / 91.1	1.29	8.87	84
Hybrid		1.13	71.5 / 85.8 / 92.9	1.16	8.97	100
2-0-0g [30]	IMU	1.13	71.1 / 84.7 / 92.5	1.22	8.99	99
0-1-1g		1.13	71.4 / 84.6 / 92.4	**1.07**	9.04	112
1-1-0g		**1.10**	**73.2 / 86.6 / 93.3**	1.08	**9.12**	**66**
Hybrid		1.12	72.5 / 86.3 / 93.2	1.11	9.08	79

Table 2: **Rotation and vanishing points estimation on the YorkUrban DB dataset [14] in the uncalibrated case.** We report median errors in degrees and AUCs over 30 runs. The rotation AUC is given at error thresholds 5° / 10° / 20°. We consider using either no gravity information, a prior on the vertical direction (*e.g.* the y axis should be vertical), or the ground truth vertical direction (*e.g.* obtained through IMU). Our proposed LO is applied to all methods.

Solver	Gravity	Rotation estimation		VP estimation		Time (ms)
		Err ↓	AUC ↑	Err ↓	AUC ↑	
2-2-0 [43]	No	43.47	6.8 / 13.8 / 21.5	6.50	3.67	130
2-1-1 [43]		42.07	7.0 / 14.3 / 22.0	6.42	3.75	**112**
2-0-0g [30]	Prior	43.99	6.5 / 13.1 / 20.2	6.60	3.55	131
0-1-1g		42.59	5.8 / 12.2 / 19.9	6.74	3.49	247
1-1-0g		43.22	6.4 / 12.9 / 20.8	6.64	3.49	273
Hybrid		42.72	7.9 / 16.2 / 24.8	5.89	4.21	160
2-0-0g [30]	IMU	25.79	21.0 / 29.7 / 37.3	2.94	7.18	200
0-1-1g		26.67	19.0 / 27.7 / 35.5	3.23	6.88	229
1-1-0g		**24.99**	**21.4 / 30.3 / 38.1**	**2.87**	**7.25**	168
Hybrid		25.83	20.3 / 28.9 / 36.6	3.09	6.94	190

Table 3: **Rotation and vanishing point estimation on the ScanNet dataset [12].** We report median errors in degrees and AUCs over 10 runs. The rotation AUC is given at thresholds 5° / 10° / 20°. Our proposed LO is applied to all methods. Note the higher performance of the hybrid RANSAC with prior gravity.

ity direction also helps with the camera calibration. Note that the most accurate focal lengths on ScanNet are obtained by the Iter method, but that it uses our proposed 1-1-0g solver. Moreover, the difference between Iter and our LO is marginal, *i.e.*, only 0.6%.

3.6. Application: Relative Rotation with IMU

To evaluate the solvers on a downstream application, we propose to leverage the predicted VPs and focal length to estimate the relative rotation in two uncalibrated images. The LaMAR dataset [38] is a real-world dataset for benchmarking localization and mapping. We considered the sequences of images coming from the validation query images of the CAB building, consisting of 4 sequences of Hololens data (images, GT poses and IMU gravity), alternating between

Solver	YorkUrban DB [14]			ScanNet [12]		
	No LO	Iter [46]	Ours	No LO	Iter [46]	Ours
2-2-0 [43]	0.262	0.104	0.039	0.527	0.405	0.252
2-1-1 [43]	0.176	0.216	0.041	0.501	0.394	0.265
2-0-0g [30]	0.075	0.062	0.041	0.184	0.126	0.136
0-1-1g	0.059	0.051	0.059	0.204	0.201	0.182
1-1-0g	0.045	0.049	**0.031**	0.167	**0.123**	0.129
Hybrid	0.065	0.058	**0.031**	0.147	0.132	0.134

Table 4: **Focal length estimation on the YorkUrban DB [14] and ScanNet [12] datasets.** We report the median relative distance to the ground truth focal length with and without local optimization.

Solver	Gravity	Rotation estimation		f estimation	Time
		Err ↓	AUC ↑	Err ↓	(ms)
F from 7 pts [20, 21]		5.19	26.9 / 43.4 / 61.3	0.253	**11**
E+f from 6 pts [27]	No	93.65	3.7 / 9.2 / 17.3	1.000	459
2-2-0 [43]		12.02	2.4 / 13.6 / 36.7	0.287	28
2-1-1 [43]		10.06	3.7 / 18.7 / 42.8	0.221	38
R+f from 4 pts [17]		22.53	16.0 / 24.3 / 33.9	0.869	834
2-0-0g [30]	IMU	3.01	36.4 / 54.3 / 68.6	0.107	56
Hybrid (Ours)		**2.89**	**38.1 / 56.8 / 71.6**	**0.093**	52

Table 5: **Relative rotation and focal length estimation on the LaMAR dataset [38].** We report the median error in degrees and AUC at error thresholds 5° / 10° / 20° for the rotation, as well as relative focal length error, over 5 runs. Our LO is applied to all VP solvers, and the gravity-based solvers leverage the IMU of the Hololens.

indoors and outdoors. Taking each pair of successive images provides a reasonable overlap for relative rotation estimation, with a total of 1415 pairs. The Hololens is an AR headset and the images are often rotated, making the information of gravity – provided by the IMU sensor of the headset – essential for accurate pose estimation. While the calibration of the Hololens is provided, a device used over several months is subject to drift and online recalibration is necessary to keep an accurate output. Thus, we did not use the GT calibration of the dataset, and only compared methods for uncalibrated images, *i.e.*, assuming an unknown focal length and principal point in the center of the image.

We leverage the same solvers as in previous sections to obtain sets of 3VPs and the focal length for pairs of images, independently. We then match the VPs by matching lines between the two images, checking for every pair of VPs the number of matched lines in common, and solving the optimal assignment between VPs. Each set of VPs can be concatenated in a matrix to form a camera rotation matrix, and the two matrices are combined to obtain the relative rotation between the images. We introduce here additional baselines based on keypoints matching. The first one is the 7-point algorithm for fundamental matrix estimation [20], from which a focal length can be extracted [21], allowing to convert the matrix to an essential matrix and to extract the corresponding relative rotation. We use the OpenCV implementation. The second one is a 6-point algorithm retrieving the essential matrix and a shared focal length between both images [27]. Finally, the 4-point algorithm retrieving the relative pose given known gravity and unknown focal length [17] is also compared. We use the authors' code for the last two methods. Points are detected and matched with SuperPoint [15] + SuperGlue [37], and lines are matched by associating line endpoints with SuperGlue. We report the median rotation error, rotation AUC at a 5° / 10° / 20° error, and the relative focal length error.

Results can be found in Table 5. The 7-point algorithm is the de facto method used to obtain a relative pose from point correspondences from uncalibrated cameras, and scores well in this task. The 6-point solver is on the contrary

more unstable and yields poorer results. Adding the information about gravity significantly boost the performance of the VP solvers, as can be seen in the last two lines. Our hybrid solver obtains the strongest performance overall, by leveraging the IMU information in three of its five solvers. Known gravity is a strong asset here, especially in such a dataset, where pairs of images have large pure rotations. On the contrary, point-based methods such as the 7-point algorithm [20] are degenerate for pure rotations.

4. Conclusion

In summary, we propose two new minimal solvers for VP estimation using 2 lines and a known gravity direction for an uncalibrated camera. Our solvers obtain better accuracy in VP, rotation and focal length estimations than previous works. Furthermore, even a coarse prior on the gravity direction is enough to get comparable or better results to previous baselines. In order to automatically leverage the most stable solver in a data-dependent manner, we combine our two 2-line solvers with three previously existing solvers into a single hybrid robust estimator. We additionally introduce a new non-minimal solver significantly increasing the overall performance of all methods.

While the gravity is the most common direction that is easily accessible, our method could also be applied with other known directions, for instance the horizon line. There are several possible improvements that can further increase the accuracy, such as cleverly sampling the pairs of lines directly into a 2-0-0g, 1-1-0g, or 0-1-1g configuration, extending the methods to unknown principal point, or developing a Bayesian reasoning on the noise of the prior gravity to better select solvers in our hybrid RANSAC.

References

[1] Sameer Agarwal and Keir Mierle. Ceres solver. http://ceres-solver.org. 6

[2] D. G. Aguileraa, J. Gómez Lahoza, and J. Finat Codesb. A new method for vanishing points detection in 3d reconstruction from a single view. In *Journ. of the Inter. Soc. for Photog. and Remote Sens. (ISPRS)*, 2005. 1

[3] M.E. Antone and S. Teller. Automatic recovery of relative camera rotations for urban scenes. In *IEEE Conf. Comput. Vis. Pattern Recog. (CVPR)*, volume 2, 2000. 1

[4] Michel Antunes, Joao P. Barreto, Djamila Aouada, and Bjorn Ottersten. Unsupervised vanishing point detection and camera calibration from a single manhattan image with radial distortion. In *IEEE Conf. Comput. Vis. Pattern Recog. (CVPR)*, 2017. 1

[5] Stephen T. Barnard. Interpreting perspective images. *Artificial Intelligence*, 21(4), 1983. 1

[6] Jean Charles Bazin, Cédric Demonceaux, Pascal Vasseur, and In-So Kweon. Rotation estimation and vanishing point extraction by omnidirectional vision in urban environment. *The International Journal of Robotics Research*, 31, 2012. 1

[7] Jean-Charles Bazin and Marc Pollefeys. 3-line ransac for orthogonal vanishing point detection. In *International Conference on Intelligent Robots and Systems (IROS)*, 2012. 1, 2

[8] Jean-Charles Bazin, Yongduek Seo, Cédric Demonceaux, Pascal Vasseur, Katsushi Ikeuchi, Inso Kweon, and Marc Pollefeys. Globally optimal line clustering and vanishing point estimation in manhattan world. In *IEEE Conf. Comput. Vis. Pattern Recog. (CVPR)*, 2012. 1

[9] Federico Camposeco, Andrea Cohen, Marc Pollefeys, and Torsten Sattler. Hybrid Camera Pose Estimation. In *IEEE Conf. Comput. Vis. Pattern Recog. (CVPR)*, 2018. 2, 4

[10] Bruno Caprile and Vincent Torre. Using vanishing points for camera calibration. *Int. J. Comput. Vis. (IJCV)*, 4, 2004. 1

[11] J.M. Coughlan and A.L. Yuille. Manhattan world: compass direction from a single image by bayesian inference. In *Int. Conf. Comput. Vis. (ICCV)*, 1999. 1, 2

[12] 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. 5, 6, 7, 8

[13] 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. 1

[14] 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, 6, 7, 8

[15] Daniel DeTone, Tomasz Malisiewicz, and Andrew Rabinovich. Superpoint: Self-supervised interest point detection and description. In *IEEE Conf. Comput. Vis. Pattern Recog. Worksh. (CVPRW)*, 2018. 8

[16] Yaqing Ding, Daniel Barath, Jian Yang, Hui Kong, and Zuzana Kukelova. Globally optimal relative pose estimation with gravity prior. In *IEEE Conf. Comput. Vis. Pattern Recog. (CVPR)*, pages 394–403, 2021. 5

[17] Yaqing Ding, Jian Yang, Jean Ponce, and Hui Kong. Minimal solutions to relative pose estimation from two views sharing a common direction with unknown focal length. In *IEEE Conf. Comput. Vis. Pattern Recog. (CVPR)*, 2020. 8

[18] Mirza Faraz and Stergios Roumeliotis. Optimal estimation of vanishing points in a manhattan world. In *Int. Conf. Comput. Vis. (ICCV)*, 2011. 1

[19] Martin A. Fischler and Robert C. Bolles. Random sample consensus: A paradigm for model fitting with applications to image analysis and automated cartography. In *Readings in Computer Vision*, 1987. 2

[20] Richard Hartley and Andrew Zisserman. *Multiple View Geometry in Computer Vision*. Cambridge University Press, 2004. 2, 8

[21] Richard I. Hartley. Extraction of focal lengths from the fundamental matrix, 2001. 8

[22] Varsha Hedau, Derek Hoiem, and David Forsyth. Recovering the spatial layout of cluttered rooms. In *Int. Conf. Comput. Vis. (ICCV)*, 2009. 1

[23] Kenichi Kanatani and Yasuyuki Sugaya. Statistical optimization for 3-d reconstruction from a single view. In *IAPR International Workshop on Machine Vision Applications*, 2005. 2

[24] Florian Kluger, Eric Brachmann, Hanno Ackermann, Carsten Rother, Michael Ying Yang, and Bodo Rosenhahn. CONSAC: Robust multi-model fitting by conditional sample consensus. In *IEEE Conf. Comput. Vis. Pattern Recog. (CVPR)*, 2020. 1, 6

[25] Karel Lebeda, Jiri Matas, and Ondrej Chum. Fixing the Locally Optimized RANSAC. In *Brit. Mach. Vis. Conf. (BMVC)*, 2012. 6

[26] Kenneth Levenberg. A method for the solution of certain non-linear problems in least squares. *Quarterly of Applied Mathematics*, 2(2), 1944. 6

[27] Hongdong Li. A simple solution to the six-point two-view focal-length problem. In *Eur. Conf. Comput. Vis. (ECCV)*, 2006. 8

[28] Haoang Li, Ji Zhao, Jean-Charles Bazin, Wen Chen, Zhe Liu, and Yun-Hui Liu. Quasi-globally optimal and efficient vanishing point estimation in manhattan world. In *Int. Conf. Comput. Vis. (ICCV)*, 2019. 1

[29] Shichen Liu, Yichao Zhou, and Yajie Zhao. Vapid: A rapid vanishing point detector via learned optimizers. In *Int. Conf. Comput. Vis. (ICCV)*, 2021. 1

[30] 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. 2, 3, 5, 6, 7, 8

[31] Xiaohu Lu, Jian Yaoy, Haoang Li, Yahui Liu, and Xiaofeng Zhang. 2-line exhaustive searching for real-time vanishing point estimation in manhattan world. In *2017 IEEE Winter Conference on Applications of Computer Vision (WACV)*, 2017. 1

[32] M.J Magee and J.K Aggarwal. Determining vanishing points from perspective images. *Computer Vision, Graphics, and Image Processing*, 26(2), 1984. 1

[33] Yiming Qian and James H. Elder. A reliable online method for joint estimation of focal length and camera rotation. In *Eur. Conf. Comput. Vis. (ECCV)*, 2022. 1, 2

[34] Long Quan and Roger Mohr. Determining perspective structures using hierarchical hough transform. *Pattern Recognition*, 9, 1989. 1

[35] Siddhant Ranade and Srikumar Ramalingam. Novel single view constraints for manhattan 3d line reconstruction. In *IEEE International Conference on 3D Vision (3DV)*, 2018. 1

[36] Carsten Rother. A new approach to vanishing point detection in architectural environments. *Image and Vision Computing*, 20(9):647–655, 2002. 1

[37] Paul-Edouard Sarlin, Daniel DeTone, Tomasz Malisiewicz, and Andrew Rabinovich. SuperGlue: Learning feature matching with graph neural networks. In *IEEE Conf. Comput. Vis. Pattern Recog. (CVPR)*, 2020. 8

[38] 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. 7, 8

[39] Julian Straub, Oren Freifeld, Guy Rosman, John J. Leonard, and John W. Fisher. The manhattan frame model—manhattan world inference in the space of surface normals. *IEEE Trans. Pattern Anal. Mach. Intell. (PAMI)*, 40, 2018. 1

[40] Xin Tong, Xianghua Ying, Yongjie Shi, Ruibin Wang, and Jinfa Yang. Transformer based line segment classifier with image context for real-time vanishing point detection in manhattan world. In *IEEE Conf. Comput. Vis. Pattern Recog. (CVPR)*, 2022. 1

[41] Elena Tretyak, Olga Barinova, Pushmeet Kohli, and Victor S. Lempitsky. Geometric image parsing in man-made environments. *Int. J. Comput. Vis. (IJCV)*, 97, 2011. 1

[42] Rafael Grompone Von Gioi, Jeremie Jakubowicz, Jean-Michel Morel, and Gregory Randall. LSD: A fast line segment detector with a false detection control. *IEEE Trans. Pattern Anal. Mach. Intell. (PAMI)*, 32(4), 2008. 6

[43] Horst Wildenauer and Allan Hanbury. Robust camera self-calibration from monocular images of manhattan worlds. In *IEEE Conf. Comput. Vis. Pattern Recog. (CVPR)*, 2012. 1, 2, 3, 4, 5, 6, 7, 8

[44] Horst Wildenauer and Markus Vincze. Vanishing point detection in complex man-made worlds. *International Conference on Image Analysis and Processing (ICIAP)*, 2007. 1

[45] Menghua Zhai, Scott Workman, and Nathan Jacobs. Detecting vanishing points using global image context in a non-manhattanworld. In *IEEE Conf. Comput. Vis. Pattern Recog. (CVPR)*, 2016. 1

[46] 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. 1, 2, 6, 7, 8

[47] Yichao Zhou, Haozhi Qi, Jingwei Huang, and Yi Ma. NeurVPS: Neural vanishing point scanning via conic convolution. In *Adv. Neural Inform. Process. Syst. (NeurIPS)*, 2019. 1, 5

[48] Yichao Zhou, Haozhi Qi, Yuexiang Zhai, Qi Sun, Zhili Chen, Li-Yi Wei, and Yi Ma. Learning to reconstruct 3d manhattan wireframes from a single image. In *Int. Conf. Comput. Vis. (ICCV)*, 2019. 1