Hybrid Camera Pose Estimation
Supplementary Material Document

Federico Camposeco[1] Andrea Cohen[1] Marc Pollefeys[1,2] Torsten Sattler[1]
[1]Department of Computer Science, ETH Zürich [2]Microsoft
{fede, acohen, pomarc, tsattler}@inf.ethz.ch

Abstract

In this supplementary document, we offer additional derivation details for the pre-processing transformations used to derive our minimal solvers introduced in Sec. 4 of the paper. Sec. 1.1 of this document specifies how to obtain the rotation that aligns the camera with the Z axis as required by the upright cases of both **PROBLEM 4-DOF** *and* **PROBLEM 5-DOF** *as presented in the main paper. In Sec. 1.2, we give details on how to eliminate the two first components of the camera translation, a process that is needed by all the solvers presented in the paper that use only one 2D-3D correspondence. Finally, we offer an additional experiment showing the effect of our Hybrid RANSAC scheme on different types of images. There we show that, depending on the difficulties present on a particular instance of the problem, our method will adapt accordingly.*

1. Details on Solvers Derivation

1.1. Aligning the Camera with a Known Vertical

In Sec. 4 of the paper, we claim that one can always represent the camera rotation by using (3) from the main paper, if one knows the direction of gravity in the camera frame $\{C\}$. In this section we show that one can always obtain a rotation that allows the camera and global Z axis to be aligned. Doing this allows the rotation between the camera and the global frames to be reduced to the simple form presented in (3).

Let ${}^{\mathrm{C}}\mathbf{g}$ be the direction in the camera frame of the gravity vector ${}^{\mathrm{G}}\mathbf{g} = [0\ 0\ 1]^\intercal$ in the global frame $\{G\}$. We aim to find the rotation $\mathtt{R}_\mathbf{g}$ to rotate the frame $\{C\}$ into $\{C'\}$, s.t.

$$\mathtt{R}_\mathbf{g}{}^{\mathrm{C}}\mathbf{g} = \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} \quad . \tag{1.1}$$

The rotation that aligns two vectors can be found by a simple procedure outlined next. Let $\mathbf{a}, \mathbf{b} \in \mathbb{R}^3$ be unit length vectors, and $\mathbf{v} = \mathbf{a} \times \mathbf{b}$ be the cross product between them. One may write the rotation $\mathtt{R}$ that aligns $\mathbf{a}$ with $\mathbf{b}$ in terms of the axis of rotation $\mathbf{v}$ and its angle θ using the *Euler-Rodrigues* formula:

$$\mathtt{R} = \mathbf{I} + \sin\theta\lfloor\mathbf{v}\rfloor_\times + (1-\cos\theta)\lfloor\mathbf{v}\rfloor_\times^2 \ , \tag{1.2}$$

where $\sin\theta = |\mathbf{v}|$ and $\cos\theta = \mathbf{a}\cdot\mathbf{b}$. For our particular case, finding the rotation that aligns ${}^{\mathrm{C}}\mathbf{g}$ and $[0\ 0\ 1]^\intercal$ is particularly simple, since ${}^{\mathrm{C}}\mathbf{g}\cdot[0\ 0\ 1]^\intercal = {}^{\mathrm{C}}g_z$ and ${}^{\mathrm{C}}\mathbf{g}\times[0\ 0\ 1]^\intercal = [{}^{\mathrm{C}}g_y\ \ -{}^{\mathrm{C}}g_x\ \ 0]^\intercal$.

Thus, the rotation to align ${}^{\mathrm{C}}\mathbf{g}$ to the Z-axis is given by

$$\mathtt{R}_\mathbf{g} = \begin{bmatrix} -{}^{\mathrm{C}}g_x^2\kappa + 1 & -{}^{\mathrm{C}}g_x{}^{\mathrm{C}}g_y\kappa & -{}^{\mathrm{C}}g_x \\ -{}^{\mathrm{C}}g_x{}^{\mathrm{C}}g_y\kappa & -{}^{\mathrm{C}}g_y^2\kappa + 1 & -{}^{\mathrm{C}}g_y \\ {}^{\mathrm{C}}g_x & {}^{\mathrm{C}}g_y & 1 - {}^{\mathrm{C}}g_x^2 + {}^{\mathrm{C}}g_y^2\kappa \end{bmatrix} \ , \tag{1.3}$$

with $\kappa = 1/({}^{\mathrm{C}}g_z + 1)$. Given $\mathtt{R}_\mathbf{g}$, one can then pre-rotate all measurements and camera centers in $\{C\}$ to $\{C'\}$, an auxiliary frame of reference where the local Z-axis is aligned to the Z-axis of the global frame of reference $\{G\}$. Hence, the rotation of the camera pose is merely a rotation about Z. Consequently, one may use (3) from the main paper to parameterize this rotation.

1.2. Elimination of Translation Variables

For minimal problems where we have only *one* 2D-3D correspondence (*i.e.*, **H41**, **uH21**, **H51+s** and **uH31+s**), we proceed to eliminate the first two unknown elements of the camera translation using this correspondence (*c.f.* (8) in the main paper).

Let $\mathbf{v}$ be a camera ray, $\mathbf{c}$ its center of projection (set to zero for pinhole cameras), α its depth and s the scale of the generalized camera. Then, the algebraic constrain given by the 2D-3D correspondence $\{\mathbf{v} \leftrightarrow \mathbf{p}\}$ can be written as

$$\alpha\mathbf{v} + s\,\mathbf{c} = \mathtt{R}\mathbf{p} + \mathtt{t}\,, \tag{1.4}$$

where $\mathbf{p}$ is the matched 3D point and $\mathtt{R}, \mathtt{t}$ is the camera pose. As explained in the main paper in Sec. 4, if we can rotate and translate the camera frame $\{C\}$ to some intermediate frame $\{C'\}$ where $\mathbf{c} = \mathbf{0}$ and $\mathbf{v} = [0\ k_1\ k_2]^\intercal$, then

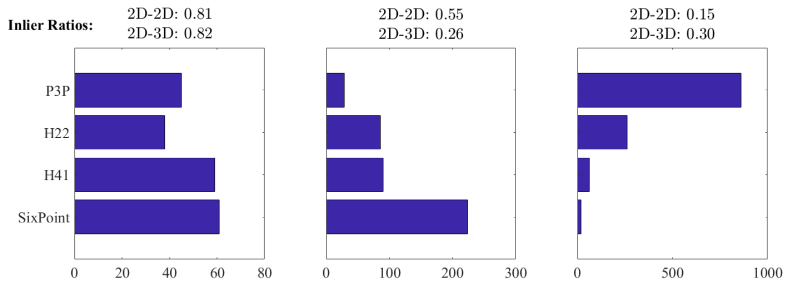

Figure 1. Number of times each solver is chosen by our hybrid method for 3 different images with different inlier ratios.

we may express $\mathtt{t}$ in terms of $\mathtt{t}_z$, *i.e.*, the last element of translation (*c.f.* (8) in the paper). Here we will explain a simple procedure to transform $\{C\}$ to $\{C'\}$, with the desirable characteristic that the obtained transformation will rotate *only* around the Z-axis. This is desirable since this procedure can be used in combination with the change of frames detailed in the previous section (*i.e.*, the transformation for upright problems).

Let $\mathtt{t}_p$ be the translation that transforms elements from $\{C\}$ to $\{C'\}$. From our requirements of $\mathbf{c} = \mathbf{0}$, it is easy to see that

$$\mathtt{t}_p = -\mathbf{c} \ . \tag{1.5}$$

The rotation $\mathtt{R}_p$ from $\{C\}$ to $\{C'\}$ must be such that the image ray has zero x component, *i.e.*,

$$\mathtt{R}_p \mathbf{v} = \begin{bmatrix} 0 \\ v'_y \\ v'_z \end{bmatrix} \ . \tag{1.6}$$

In contrast to (1.1), there is not a unique rotation that aligns the two vectors in play, rather we are interested in the family of rotations where the first element of the multiplication by $\mathbf{v}$ is zero. Thus, in order to be compatible with previous transformations for vertical alignment, we choose to define $\mathtt{R}_p$ as a rotation around the Z-axis. So, similarly to (3) in the paper, $\mathtt{R}_p = \mathtt{R}_p(a, b)$, where a and b are the only two unknown elements of the rotation. We can then solve a very small linear system in a and b given $\mathbf{v}$. This gives the following rotation

$$\mathtt{R}_p = \begin{bmatrix} v_y\tau & v_x\tau & 0 \\ v_x\tau & v_y\tau & 0 \\ 1 & 2 & 1 \end{bmatrix} \quad \text{with} \quad \tau = \frac{1}{\sqrt{v_x^2 + v_y^2}} \ , \tag{1.7}$$

which one can use to pre-rotate the cameras and measurements in $\{C\}$ in order to be able to use (8).

2. Hybrid RANSAC Adaptability

The motivation behind the proposed RANSAC extension, is twofold. On one hand, we wish to have a procedure that rejects outliers coming from different types of matches. On the other hand, we want RANSAC to focus on matches which, up to the current iteration, are expected to yield better inlier counts, while at the same time exploring different types of matches gradually. To this end, it is informative to know how our Hybrid RANSAC can adapt to different types of queries. As it can be seen in Fig. 1, for three different images (selected from the Dubrovnik dataset), we see very different usages of minimal solvers by our RANSAC variant. Given the inlier ratios for 2D-2D and 2D-3D matches of each image (shown on top of each plot in Fig. 1), the distribution of selected solvers is expected and justified.