Task 1.1: Let $\mathbf{H}$ be the homography that maps $(X, Y) \leftrightarrow (x, y)$, and let $\mathbf{H}_k = k\mathbf{H}$, $k \neq 0$. Then,

$$\mathbf{H}_k = \begin{bmatrix} kr_{11} & kr_{12} & kt_x \\ kr_{21} & kr_{22} & kt_y \\ kr_{31} & kr_{32} & kt_z \end{bmatrix}$$

Applying this matrix to a point $(X, Y, 1)$ and dividing by the third row yields the calibrated image coordinates

$$x_k = \frac{k(r_{11}X + r_{12}Y + t_x)}{k(r_{31}X + r_{32}Y + t_z)}$$
$$y_k = \frac{k(r_{21}X + r_{22}Y + t_y)}{k(r_{31}X + r_{32}Y + t_z)}$$

The scaling factor k cancels out, and we are left with $x_k = x$ and $y_k = y$, which shows that any scalar multiple of $\mathbf{H}$ maps (X, Y) to the same coordinates (x, y).

Task 1.2: There are additional constraints because the first two columns must satisfy the properties of the columns of a rotation matrix (orthogonal and of unit length).

The assignments and projects in this course are copyrighted by Simen Haugo and may not be copied, redistributed, reused or repurposed without written permission.

Task 3.3: A matrix $\mathbf{R}$ is a rotation matrix if and only if it is an orthogonal matrix (i.e. the columns are orthogonal and of unit length) and the determinant is equal to 1. The first property is equivalent to

$$\mathbf{R}^T\mathbf{R} = \mathbf{I} \tag{1}$$

which is the same constraint that Zhang uses in Eq. (15) in his paper. We may check how well this is satisfied using the Frobenius norm of $\mathbf{I} - \mathbf{R}^T\mathbf{R}$, which can be computed by `numpy.linalg.norm` (Python) and `norm` (Matlab). We might also want to check that the determinant is equal to 1, but this by itself is not sufficient to have a valid rotation matrix. Note also that the determinant of an orthogonal matrix is always $+1$ or -1.

The assignments and projects in this course are copyrighted by Simen Haugo and may not be copied, redistributed, reused or repurposed without written permission.

Task 4.1: Consider a single correspondence $\mathbf{u} \leftrightarrow \mathbf{X}$. Let $\tilde{\mathbf{x}} = \mathbf{K}^{-1}\tilde{\mathbf{u}}$ and

$$\mathbf{t} = \begin{bmatrix} t_x \\ t_y \\ t_z \end{bmatrix} \quad \text{and} \quad \mathbf{R} = \begin{bmatrix} \mathbf{r}_1^T \\ \mathbf{r}_2^T \\ \mathbf{r}_3^T \end{bmatrix}.$$

Then the inhomogeneous form of $\tilde{\mathbf{x}}$ is

$$x = \frac{\mathbf{X}^T\mathbf{r}_1 + t_x}{\mathbf{X}^T\mathbf{r}_3 + t_z},$$
$$y = \frac{\mathbf{X}^T\mathbf{r}_2 + t_y}{\mathbf{X}^T\mathbf{r}_3 + t_z}.$$

Following the DLT strategy, we multiply by the denominator and rearrange terms to get two equations of the desired system:

$$\begin{bmatrix} -\mathbf{X}^T & \mathbf{0} & x\mathbf{X}^T \\ \mathbf{0} & -\mathbf{X}^T & y\mathbf{X}^T \end{bmatrix} \mathbf{m} = \begin{bmatrix} t_x - xt_z \\ t_y - yt_z \end{bmatrix}$$

where $\mathbf{m} = \begin{bmatrix} \mathbf{r}_1^T & \mathbf{r}_2^T & \mathbf{r}_3^T \end{bmatrix}^T$ is a vector concatenating the rotation matrix rows. The complete linear system is built by stacking n such pairs of equations.

Task 4.2: Because $\mathbf{b}$ is not necessarily equal to zero, the system is in general inhomogeneous.

Task 4.3: With $n \geq 5$ correspondences, we get an over-determined system. This can be solved with linear least squares methods, e.g. using the pseudo-inverse. As the system is inhomogeneous, the solution scale is not ambiguous. Thus, the rotation matrix is recovered by simply rearranging the 9×1 vector $\mathbf{m}$ into a 3×3 matrix.

The assignments and projects in this course are copyrighted by Simen Haugo and may not be copied, redistributed, reused or repurposed without written permission.