| | 1 | 2 | Σ |
|---|---|---|---|
| César Díaz Blanco<br>Long Nguyen | | | |

# Exercise Nr. 1

## 1 3D Transformations

### a) Preserving properties of 3D transformations

#### i) Rigid transformations preserve lengths

Recall that $\mathbf{x} \in \mathbb{R}^3$ is transformed by rigid transformation $\mathbf{T}$ with rotation matrix $\mathbf{R}$ and translation $\mathbf{t}$ as follows:

$$\mathbf{T}\begin{bmatrix}\mathbf{x}\\1\end{bmatrix} = \begin{bmatrix}\mathbf{R} & \mathbf{t}\\0 & 1\end{bmatrix}\begin{bmatrix}\mathbf{x}\\1\end{bmatrix} = \begin{bmatrix}\mathbf{Rx}+\mathbf{t}\\1\end{bmatrix}$$

Let $\mathbf{x}, \mathbf{y} \in \mathbb{R}^3$ be two arbitrary 3D points. We will prove that rigid transformations preserve lengths by showing that the distance between $\mathbf{x}$ and $\mathbf{y}$ remains the same after applying a rigid transformation to $\mathbf{x}$ and $\mathbf{y}$ and measuring the distance between them.

$$||(\mathbf{Rx}+\mathbf{t}) - (\mathbf{Ry}+\mathbf{t})|| = ||\mathbf{Rx} - \mathbf{Ry}|| = ||\mathbf{R}(\mathbf{x}-\mathbf{y})||$$

Since $\mathbf{R} \in \mathrm{SO}(3)$, norms are preserved: $||\mathbf{R}(\mathbf{x}-\mathbf{y})|| = ||\mathbf{x}-\mathbf{y}||$

#### ii) Similarity transformations preserve angles

Recall that a vector $\mathbf{v}$ with start point $\mathbf{x_1} \in \mathbb{R}^3$ and endpoint $\mathbf{x_2}$ is transformed by a similarity transformation $\mathbf{T}$ with scale $s$, rotation matrix $\mathbf{R}$, and translation $\mathbf{t}$ as follows:

$$\mathbf{x_1} \to s\mathbf{Rx_1} \text{ and } \mathbf{x_2} \to s\mathbf{Rx_2}, \text{ then } \mathbf{v} \to s\mathbf{Rx_2} - s\mathbf{Rx_1} = s\mathbf{Rv}$$

We will prove that similarity transformations preserve angles by showing that the cosine of angle $\theta$ between vectors $\mathbf{u}, \mathbf{v}$ remains the same before and after transforming the vectors with a similarity transformation $\mathbf{T}$ with scale, $s$ rotation matrix $\mathbf{R}$, and translation $\mathbf{t}$.

$$\mathbf{u}' = sR\mathbf{u}, \quad \mathbf{v}' = sR\mathbf{v} \Rightarrow \mathbf{u}'^\top\mathbf{v}' = s^2\mathbf{u}^\top R^\top R\mathbf{v} = s^2\mathbf{u}^\top\mathbf{v}$$

Also:

$$\|\mathbf{u}'\| = s\|\mathbf{u}\|, \quad \|\mathbf{v}'\| = s\|\mathbf{v}\|$$

Then, the cosine of angle $\theta$ between vectors $\mathbf{u}, \mathbf{v}$ remains the same:

$$\cos\theta' = \frac{\mathbf{u}'^\top\mathbf{v}'}{\|\mathbf{u}'\|\|\mathbf{v}'\|} = \frac{s^2\mathbf{u}^\top\mathbf{v}}{s^2\|\mathbf{u}\|\|\mathbf{v}\|} = \frac{\mathbf{u}^\top\mathbf{v}}{\|\mathbf{u}\|\|\mathbf{v}\|} = \cos\theta$$

#### iii) Affine transformations preserve parallelism

We will prove that affine transformations preserve parallelism by showing that given two parallel vectors $\mathbf{u}, \mathbf{v}$ such that $\mathbf{u} = \lambda\mathbf{v}$, they can still be written as a scaled version of each other after transforming them with affine transformation $\mathbf{A}$:

$$\mathbf{u}' = \mathbf{Au}, \mathbf{v}' = \mathbf{Av} \quad \to \quad \mathbf{u}' = \mathbf{Au} = \lambda\mathbf{Av} = \lambda\mathbf{v}' \quad \Rightarrow \quad \mathbf{u}' \parallel \mathbf{v}'$$

## b) Affine transformation matrices properties

### i) Length ratio is preserved for parallel line segments or vectors

Given two parallel vectors $\mathbf{u}, \mathbf{v}$ such that $\mathbf{u} = \lambda\mathbf{v}$, we will prove that an affine transformation $\mathbf{A}$ preserves their length ratio by showing that after transforming them and getting $\mathbf{u}', \mathbf{v}'$ they can still be written as $\mathbf{u}' = \lambda\mathbf{v}'$ thus preserving their length ratio.

$$\mathbf{u}' = \mathbf{A}\mathbf{u}, \mathbf{v}' = \mathbf{A}\mathbf{v} \quad \rightarrow \quad \mathbf{u}' = \mathbf{A}\mathbf{u} = \lambda\mathbf{A}\mathbf{v} = \lambda\mathbf{v}' \quad \Rightarrow \quad \mathbf{u}' = \lambda\mathbf{v}'$$

### ii) Length ratio is not preserved for non-parallel line segments or vectors

We present a counterexample. Let vectors $\mathbf{u} = [1 \quad 0 \quad 0]^\top$ and $\mathbf{v} = [0 \quad 1 \quad 0]^\top$. These are not parallel as there is no possible way to write $\mathbf{u}$ as a scaled $\mathbf{v}$ or viceversa. Their length ratio is 1: $||\mathbf{u}||/||\mathbf{v}|| = 1$. Now we transform the vectors with an affine transformation $\mathbf{T}$:

$$\mathbf{T}\begin{bmatrix} \mathbf{u} \\ 1 \end{bmatrix} = \begin{bmatrix} 2 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 \\ 0 \\ 0 \\ 1 \end{bmatrix} = \begin{bmatrix} 2 \\ 0 \\ 0 \\ 1 \end{bmatrix} = \begin{bmatrix} \mathbf{u}' \\ 1 \end{bmatrix}$$

$$\mathbf{T}\begin{bmatrix} \mathbf{v} \\ 1 \end{bmatrix} = \begin{bmatrix} 2 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 0 \\ 1 \\ 0 \\ 1 \end{bmatrix} = \begin{bmatrix} 0 \\ 1 \\ 0 \\ 1 \end{bmatrix} = \begin{bmatrix} \mathbf{v}' \\ 1 \end{bmatrix}$$

Then their length ratio is: $||\mathbf{u}'||/||\mathbf{v}'|| = 2$.

## c) Degrees of freedom for 3D transformations

A 4 by 4 matrix would have 16 degrees of freedom but our transformations have requirements which decrease this.

### i) Rigid

Rotations have 3 degrees of freedom as any rotation can be determined by three numbers. An example is rotational vectors which point along the axis rotation and whose magnitude encodes the angle of rotation.
Translations have 3 degrees of freedom.
In total rigid transformations have 6 degrees of freedom.

### ii) Similarity

There are at least as many degrees of freedom as for the rigid transformation. The scaling $s$ gives an extra degree of freedom.
In total rigid transformations have 7 degrees of freedom.

### iii) Affine

Now we need to think about how many degrees of freedom does a 3 by 3 invertible matrix have. There is only one constraint which is for the determinant to be different than 0. This constraint doesn't lead to a single equation which could constrain the values. So there are 9 degrees of freedom.
Translations have 3 degrees of freedom.
In total affine transformations have 12 degrees of freedom.

### iv) Projective

Now we need to think about how many degrees of freedom does a 4 by 4 invertible matrix have. This is 16 degrees of freedom as there is only one constraint which is for the determinant to be different than 0 which is not enough to lower the count. However, guided by the hint, one could multiply the entire matrix by any nonzero number and recover the same result in homogeneous coordinates. This lowers the count by 1. In total projective transformations have 15 degrees of freedom.

## 2 Estimating Camera Pose from 3D-2D Correspondences

### a) Uncalibrated image points to calibrated

For a general camera with intrinsic matrix $\mathbf{K}$, the projection equation is:

$$\begin{bmatrix} u_i \\ v_i \\ 1 \end{bmatrix} = \mathbf{K} \begin{bmatrix} \mathbf{R} & \mathbf{t} \end{bmatrix} \begin{bmatrix} \mathbf{x}_i \\ 1 \end{bmatrix}$$

Assume invertible $\mathbf{K}$ then

$$K^{-1} \begin{bmatrix} u_i \\ v_i \\ 1 \end{bmatrix} = K^c \begin{bmatrix} \mathbf{R} & \mathbf{t} \end{bmatrix} \begin{bmatrix} \mathbf{x}_i \\ 1 \end{bmatrix}$$

### b) Linear system with camera parameters as unknowns

For a single correspondence pair $\mathbf{x} \in \mathbb{R}^3$ and $\mathbf{u} \in \mathbb{R}^2$, let's denote $\mathbf{x} = [x, y, z]^T$ and $\mathbf{u} = [u, v]^T$.

The projection gives us:

$$\begin{bmatrix} u \\ v \\ 1 \end{bmatrix} \propto \begin{bmatrix} r_{11} & r_{12} & r_{13} & t_1 \\ r_{21} & r_{22} & r_{23} & t_2 \\ r_{31} & r_{32} & r_{33} & t_3 \end{bmatrix} \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix}$$

This yields:

$$\begin{cases} u \propto r_{11}x + r_{12}y + r_{13}z + t_1 \\ v \propto r_{21}x + r_{22}y + r_{23}z + t_2 \\ 1 \propto r_{31}x + r_{32}y + r_{33}z + t_3 \end{cases}$$

Since the proportions are constant, we can write:

$$\begin{cases} u = \frac{r_{11}x + r_{12}y + r_{13}z + t_1}{r_{31}x + r_{32}y + r_{33}z + t_3} \\ v = \frac{r_{21}x + r_{22}y + r_{23}z + t_2}{r_{31}x + r_{32}y + r_{33}z + t_3} \end{cases}$$

Remove the denominators:

$$\begin{cases} u(r_{31}x + r_{32}y + r_{33}z + t_3) = r_{11}x + r_{12}y + r_{13}z + t_1 \\ v(r_{31}x + r_{32}y + r_{33}z + t_3) = r_{21}x + r_{22}y + r_{23}z + t_2 \end{cases}$$

Rearranging to get equations in the form $\mathbf{v}^T\boldsymbol{\theta} = 0$:

$$\begin{cases} r_{11}x + r_{12}y + r_{13}z + t_1 - ur_{31}x - ur_{32}y - ur_{33}z - ut_3 = 0 \\ r_{21}x + r_{22}y + r_{23}z + t_2 - vr_{31}x - vr_{32}y - vr_{33}z - vt_3 = 0 \end{cases}$$

Given $\boldsymbol{\theta}^T = [r_{11}, r_{12}, r_{13}, t_1, r_{21}, r_{22}, r_{23}, t_2, r_{31}, r_{32}, r_{33}, t_3]$, we have:

$$\mathbf{v}_1^T = [x, y, z, 1, 0, 0, 0, 0, -ux, -uy, -uz, -u]$$
$$\mathbf{v}_2^T = [0, 0, 0, 0, x, y, z, 1, -vx, -vy, -vz, -v]$$

## c) Full linear system

Each correspondence pair gives us two linear equations of the form $\mathbf{v}^T\boldsymbol{\theta} = 0$. Given 16 unknowns, we need 8 pairs.

## d) Steps to get the least squares solution

We perform SVD to get $M = U\Sigma V^T$. The last column $v_t$ of $V$ which corresponds to the smallest singular value $\sigma_t$ is the solution since

$$Mv_t = (U\Sigma V^T)v_t = U\Sigma(V^T v_t) = U\Sigma e_t = u_t\sigma_t$$

Since $||v_t|| = 1$ we satisfied the unitary constraint.

## e) Recovering the rotation and translation from $\theta$

Rearrange elements of $\boldsymbol{\theta}^*$ to form a preliminary $3 \times 3$ matrix $\tilde{\mathbf{R}}$. Due to noise, $\tilde{\mathbf{R}}$ may not satisfy the properties of a valid rotation matrix:

- $\mathbf{R}^T\mathbf{R} = \mathbf{I}$ (orthogonality)
- $\det(\mathbf{R}) = 1$ (proper rotation)

To find the closest valid rotation matrix to $\tilde{\mathbf{R}}$:

1. Compute the SVD of $\tilde{\mathbf{R}}$:
$$\tilde{\mathbf{R}} = \mathbf{U\Sigma V}^T \tag{1}$$
2. The closest rotation matrix is:
$$\mathbf{R} = \mathbf{UV}^T \tag{2}$$
3. Check determinant: if $\det(\mathbf{R}) = -1$, we have a reflection. To ensure proper rotation:
$$\mathbf{R} = -\mathbf{UV}^T \tag{3}$$

Since $R$ should be a rotation matrix $\Sigma$ is the diagonal matrix with the eigen-values of $R^TR$ which is a identity matrix, hence $\Sigma$ can be ignored in case of valid rotation matrix.

## f) Camera-to-world transformation matrix and camera position in world frame

Camera to world: $R^{-1} = R^T$.

Camera position: $t$.