

Exercises in 3D Computer Vision

No exercises on this Tuesday due to the "Studentische Vollversammlung"!

Exercise 5 (H) Euclidian Transformations

For the following exercises we take the point set

```
p = [0 0;1 0;1 1;0 1;0.5 1.6;1 1;0 0;0 1;1 0]'
```

These can be changed to homogeneous coordinates by `p_h=[p; ones(1,9)]`

Both can be plotted by `plot(p(1,:),p(2,:)); axis equal`

Note that this is just possible because the w -value of a point $[xyw]^T$ is normalized to 1.

Just to remind you, if \mathbf{T} is transformation matrix and \mathbf{P} is a matrix that was generated by concatenated vectors $\mathbf{T} \cdot \mathbf{P}$ does the same as transforming the points first and concatenating them after.

All of the exercises must be calculated in homogeneous coordinates.

- Create a transformation matrix \mathbf{T}_1 that translates the points by $\mathbf{t} = [3, -4]^T$
- Create a transformation matrix \mathbf{T}_2 that rotates the points by 20 degrees. Mind that the function `sin()` just accepts radian values.
- Create a transformation matrix \mathbf{T}_3 by multiplication that applies the rotation first and the translation after.
- Create a transformation matrix \mathbf{T}_4 by multiplication that applies the translation first and the rotation after.
- Express \mathbf{T}_3 and \mathbf{T}_4 by the rotation matrix \mathbf{R} , translation vector \mathbf{t} . Do this calculation manually.

Exercise 6 (H) Affine Transformations

- Apply the matrix $\begin{bmatrix} 1 & 2 & 0 \\ 0.5 & 3 & 0 \\ 0 & 0 & 1 \end{bmatrix}$ to the set of points.

- How would the matrix look like if the bottom right point of the house would be projected to $[2, -0.5]^T$ and the left top point to $[-1, 1]^T$?

Exercise 7 (H) Projective Transformations

- Apply the matrix $\begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0.3 & 0.5 & 1 \end{bmatrix}$ and plot it.

b) Write the function `plotHomo(p)` that normalizes each point before showing and plots the points after.

c) Apply the matrix $\begin{bmatrix} 4 & 0 & 0 \\ 0 & 4 & 0 \\ 0 & 0 & 4 \end{bmatrix}$ and plot it. Can you explain what you see?