DATA.ML.300 Computer Vision
Exercise Round 5

For these exercises you will need Python and a webcam. Webcams are present in exercise classrooms during sessions. Return all your answers and output figures in a pdf-file. For pen & paper task(s), the submitted pdf-file should include a screenshot of hand written task, or text converted from Word or Latex format. In addition, submit runnable .py files, where you have filled in your codes to the template files. Do not include all the code from the .py files in the pdf. Exercise points will be granted after a teaching assistant has checked your answers. Returns done before the deadline will result in maximum of 4 points, whereas returns after the deadline will result in maximum of 1 point.

Make sure you have *OpenCV* library for Python installed.

```
pip install --user --upgrade opencv-python
```

Task 1. Similarity transformation from two point correspondences. (pen & paper) (1 point)
A similarity transformation consists of rotation, scaling and translation and is defined in two dimensions as follows:

$$\mathbf{x}' = s\mathbf{R}\mathbf{x} + \mathbf{t} \quad \Leftrightarrow \quad \begin{pmatrix} x' \\ y' \end{pmatrix} = s \begin{pmatrix} \cos(\theta) & -\sin(\theta) \\ \sin(\theta) & \cos(\theta) \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix} + \begin{pmatrix} t_x \\ t_y \end{pmatrix} \tag{1}$$

Describe a method for solving the parameters s, θ, t_x, t_y of a similarity transformation from two point correspondences $\{\mathbf{x}_1 \rightarrow \mathbf{x}'_1\}$, $\{\mathbf{x}_2 \rightarrow \mathbf{x}'_2\}$ using the following stages. Remember to include the equations used and all intermediate steps, the end results are not enough.

a) Compute the correspondence between vectors $\mathbf{v}' = \mathbf{x}'_2 - \mathbf{x}'_1$ and $\mathbf{v} = \mathbf{x}_2 - \mathbf{x}_1$ using the similarity transform above. Use corresponding unit vectors to solve the scale factor s from this correspondence. *Hint: There should be no scaling in a transformation between two unit vectors*

b) Solve also the rotation angle θ from this correspondence.

c) After solving s and θ compute $\mathbf{t}$ using equation (1) and either one of the two point correspondences.

d) Use the procedure to compute the transformation from the following point correspondences: $\{(\frac{1}{2}, 0) \rightarrow (0, 0)\}$, $\{(0, \frac{1}{2}) \rightarrow (-1, -1)\}$.
(Hint: Drawing the point correspondences on a grid paper may help you to check your answer.)

Task 2. Homography using SIFT (Programming exercise) (1 point)
Look up the code in **homography.py** and complete the missing parts. **Include all the output plots to your pdf-file, and return also your runnable version of homography.py.** Feel free to try your own images albeit not required.

Task 3. Real-time face point tracking (Programming exercise) (2 points)
We'll be using KLT-tracker to track points detected from a face. You will need a web-cam. Webcams can be found in computer classroom during exercise sessions. Open **face_tracking.py** and follow the instructions written in the comments. Answer the following questions a) and b). **Include your answers and two plots to your pdf-file, and return also your runnable version of face_tracking.py.**

a) How does this program work, i.e. what are its main parts? List 4 separate steps in the tracking process.

b) Do you notice any problems with the tracking? How do you think these could be avoided?