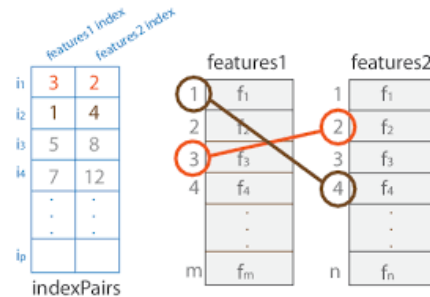# _Computer Vision and Pattern Recognition_

Coursework on computer vision [50% mark]

The course work requires Matlab or python programming. In all questions, you can use any existing toolbox/code, unless '_implement'_ is specified. You can compare results of your implementation to a standard toolbox.

Submission instructions:

One joint report by each pair

Page limit: 3 (three) A4 pages per report. List of references and appendix do not count for this page limit. Use report template from Overleaf.
https://www.overleaf.com/read/pymdnrcccwbf

At master level of this course, general principles for writing technical report are expected to be known and adhered to. Similarly, for practices in conducting experiments, some are as listed below:

a) Select relevant results that support the points you want to make rather than everything that matlab gives.

b) The important results should be in the report, not just in the appendix.

c) Use clear and tidy presentation style, consistent across the report e.g. figures, tables.

d) The experiments should be described such that there is no ambiguity in the settings, protocol and metrics used.

e) The main points are made clear, identifying the best and the worst-case results or other important observations.

f) Do not: copy standard formulas from lecture notes, explain algorithms in detail, or copy figures from other sources. References to lecture slides or publications/webpages are

enough in such cases, however short explanations of new terms or parameters referred to are needed.

The report should present different steps of the experiment, briefly discuss: the main operations, the challenges, justify the choices you made. Quality and completeness of discussions within the page limit will be marked. Include formulas (if different from lecture slides or other online available resources otherwise reference is sufficient), and results presented in figures and their discussion.

You can structure the report into 5 seactions corresponding to the CW taks.

In order to show your results you can use an existing code or implement methods to display, in pictures, points and lines using their parameters as inputs.

Code required for the experiments can be taken from any public library, otherwise implemented if necessary. Source code, is not required, however if needed it can go to appendices, which do not count for the page limit.

Submit the report in **pdf** through the Blackboard system. Write your **full names, logins and CID numbers on the first page. Use both logins in the submitted filename e.g. login1_login2.pdf.** The latest submission before the deadline will be assessed.

If you have questions, please post it on **Blackboard Discussion Forum**

==

Task 1. [10 marks] Collect Data
Build a callibration grid e.g. from a cardbox. Note that the grid depends on the callibration - method/implentation you use.

Choose a small textured object, e.g. toy, sculpture, tool.

Collect a sequence of 5-10 pictures (we call it **FD**) with and without the object in the grid. Change the camera position between pictures.

Collect a similar sequence (we call it **HG**) by changing the zoom (e.g. factor 1.5) and slightly rotating the camera (e.g. 10-20 degree) but keeping exactly the same location of the camera. (2 planar by using 2 books)

Show the whole data in the appendix.

Task 2. [10 marks] Keypoint orrespondences between images

1) Compare quality/quantity of correspondences found by two methods
   a) Manual (clicking on corresponding points)

b) Automatic (detecting keypoint and matching descriptors)

Task 3. [10 marks] Camera calibration

1) Find and report camera parameters.

2) Can you estimate or illustrate distortions of your camera?

Task 4. [10 marks] Transformation estimation

1) Estimate a homography matrix between a pair of images from HG.

a) Show the keypoints and their correspondences projected from the other image.

2) Estimate fundamental matrix between a pair of images from FD.

a) Show the keypoints and their corresponding epipolar lines in the other image.

b) Show epipoles, vanishing points and horizon in your images.

3) find out how many outliers the estimation method tolerates

Task 5. [10 marks] 3D geometry

1) Show a stereo rectified pair of your images with epipolar lines.

2) Calculate and display depth map of your object estimated from different views.