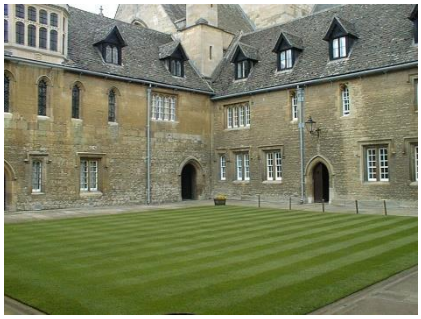
CORK INSTITUTE OF TECHNOLOGY
INSTITIÚID TEICNEOLAÍOCHTA CHORCAÍ

Computing@CIT

MACHINE VISION



Assignment 1:
Feature points and image matching

DUE DATE

This assignment should be submitted to Canvas before 11:59pm on **Friday 20/03/2020**.

Please submit a single ZIP file with your student number and name in the filename. Your submission should contain:

- A detailed documentation of all code you developed, including the tests and evaluations you carried out. Please make sure that you include a document with every result image you produce referencing the exact subtask and lines of code.

- All Python code you developed in a single .py file that can be executed and that generates the outputs you are referring to in your evaluation. Please make sure that you clearly indicate in your comments the exact subtask every piece of code is referring to.

You can achieve a total of 50 points as indicated in the tasks.

TASK 1 (Feature points, *33 points*)

In this task you will implement a scale and rotation invariant point feature extraction algorithm inspired by SIFT to identify a set of interest points in an image together with their respective scales and rotations.

A. Download the input image file **Assignment_MV_01_image_1.jpg** from Canvas. Load the file and convert it into a single channel grey value image [*2 points*]. Make sure the data type is float32 to avoid any rounding errors [*1 point*]. Determine the size of the image and resize the image to double its size [*2 points*].
B. Create twelve Gaussian smoothing kernels with increasing $\sigma = 2^{k/2}, k = 0, \dots, 11$, and plot each of these kernels as image [*4 points*]. Make sure that the window size is large enough to sufficiently capture the characteristic of the Gaussian. Apply these kernels to the resized input image from subtask A to create a scale-space representation and display all resulting scale-space images [*2 points*].
C. Use the scale-space representation from subtask B to calculate difference of Gaussian images at all scales. Display all resulting DoG images [*3 points*].

D. Find key-points by thresholding all DoGs from subtask C using a threshold of $T = 10$. Suppress non- maxima in scale-space by making sure that the key-points have no neighbours, both in space as well as in scale, with higher values [*3 points*]. The resulting key-points should comprise three coordinates (x, y, σ), two spatial and the scale at which they were detected.

E. Calculate derivatives of all scale-space images from subtask B using the kernels $d_x = (1 \quad 0 \quad -1)$ and $d_y = (1 \quad 0 \quad -1)^T$. Display the resulting derivative images at all scales [*4 points*].

F. Calculate the gradient length m_{qr} and gradient direction θ_{qr} for the 7×7 grid of points $(q, r) \in \left\{x + \frac{3}{2}k\sigma \mid k = -3,..,3\right\} \times \left\{y + \frac{3}{2}k\sigma \mid k = -3,..,3\right\}$ sampled around each key-point (x, y) and using the appropriate scale σ determined in subtask D and the correct gradient images from subtask E [*4 points*]. Also calculate a Gaussian weighting function $w_{qr} = e^{-(q^2+r^2)/(9\sigma^2/2)}/(9\pi\sigma^2/2)$ for each of the grid points [*1 point*]. Now create a 36-bin orientation histogram vector h and accumulate the weighted gradient lengths $w_{qr}m_{qr}$ for each grid point (q, r) where the gradient direction θ_{qr} falls into this particular bin [*3 points*]. Use the maximum of this orientation histogram to determine the orientation of the key-point [*1 point*].

G. Draw all the key-points into the input image using a circle with 3σ radius and a line from the key-point centre to the circle radius to indicate the orientation (see example for a single key-point on the right). Display the resulting output image with all the key-points [*3 points*].

TASK 2 (Image matching, *17 points*)

In this task you will implement a correlation-based area matching algorithm to find a patch extracted from one image in a second image.

A. Download the input image files **Assignment_MV_01_image_1.jpg** (the same as in the previous task) and **Assignment_MV_01_image_2.jpg** from Canvas. Load both files and convert them into a single channel grey value image [*2 points*]. Make sure the data type is float32 to avoid any rounding errors [*1 point*].

B. The window on the 1st floor above the arch on the left wing is in a rectangle with the image coordinates $((360,210), (430,300))$ in the first input image. Draw a rectangle around this window in the input image and display it [*1 point*]. Now cut out the image patch only containing the window and display it as image [*2 points*].

C. Calculate the mean and standard deviation of the cut-out patch from subtask B [*2 points*]. Go through all positions in the second input image and cut out a patch of equal size [*2 points*]. Also calculate mean and standard deviation and from this the cross-correlation between the two patches [*3 points*]. Create and display an image of all cross-correlations for all potential positions in the second image [*2 points*]. Find the position with maximum cross-correlation and draw a rectangle around this position in the second input image. Display the result [*2 points*].