

COMP9517: Computer Vision

2020 T2 Lab 3 Specification

Maximum Marks Achievable: 2.5

This lab is **worth 2.5% of the total course marks.**

The lab files should be submitted online.
Instructions for submission will be posted closer to the deadline.
Deadline for submission is Week 5, Wednesday 1 July 2020, 23:59:59.

Objective: This lab revisits important concepts covered in the lectures of Week 4 and aims to make you familiar with implementing specific algorithms.

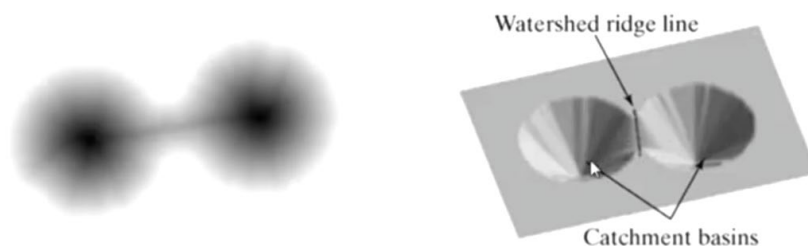
Materials: The sample images and support code to be used in the tasks of this lab are available in WebCMS3. You are required to use OpenCV 3+ with Python 3+. Jupyter notebook files are preferred for submitting your code.

Submission: The tasks are assessable **after the lab**. Submit your code and results in a zip file via WebCMS3 by the above deadline.

Image Segmentation

MeanShift is a clustering algorithm that assigns pixels to clusters by iteratively shifting points towards the modes in the feature space, where a mode is a position with the locally highest number of data points (highest density). A visualisation can be seen [here](#).

Watershed is a transformation that aims to segment the regions of interest in a grayscale image. This method is particularly useful when two regions of interest are close to each other; that is, their edges touch. It treats the image as a topographic map, with the intensity of each pixel representing the height. For instance, dark areas are considered to be ‘lower’, and act as troughs. Bright areas are ‘higher’, acting as hills or as a mountain ridge.



Visualising the Watershed: The left image can be topographically represented as the image on the right. Adopted from [Agarwal 2015](#).

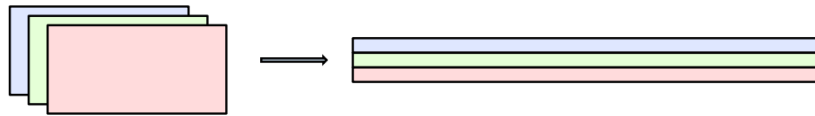
Image segmentation can be thought of as labelling pixels in an image. It is an important research topic in computer vision and comes in many different flavours: interactive segmentation, semantic segmentation, instance segmentation, and many more. In this lab the MeanShift clustering algorithm and the Watershed algorithm will be used to solve unsupervised image segmentation.

Task 1 (1 mark): Use the MeanShift algorithm to segment images.

Images to be used for this task: **shapes.png** and **strawberry.png**

Hint: Use [MeanShift clustering](#) from scikit-learn.

Step 1. Once you have read the images into numpy arrays, extract each colour channel (R, G, B) so you can use each as a variable for classification. To do this you will need to convert the colour matrices into a flattened vector as depicted in the image below.



Step 2. Then you can use the new flattened colour sample matrix (10,000 x 3 if your original image was 100 x 100) as your variable for classification.

Step 3. Use the MeanShift **fit_predict** function to perform a clustering and save the cluster labels, which we want to observe.

Submit the segmented images.

Task 2 (1 mark): Use Watershed transformation to segment grayscale versions of the images.

Images to be used for this task: **shapes.png** and **strawberry.png**

Hint: Use [Watershed segmentation](#) from scikit-learn.

Step 1. Convert the image to grayscale.

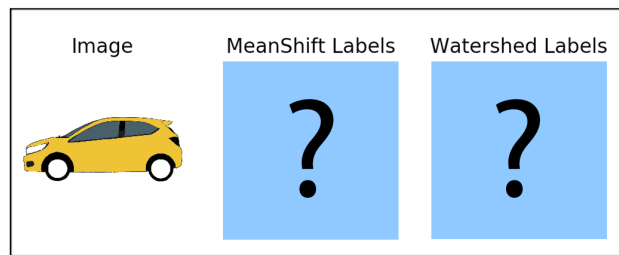
Step 2. Calculate the distance transform of the image. Note: this is a vital step of the Watershed algorithm. Visualising this step may help you understand how the algorithm works! Plot the result of the distance transform to see what is happening under the hood.

Step 3. Generate the Watershed markers as the ‘clusters’ furthest away from the background. This can be syntactically confusing so make sure to check the example code in the link above.

Step 4. Perform Watershed on the image. This is the part where the image is ‘flooded’ and the water sinks to the ‘catchment basins’ based on the markers found in Step 3.

Submit the segmented images.

Note: You might notice that MeanShift performs better for one image, while Watershed performs better for the other. A plotting template has been provided so you can compare the results of the two algorithms side by side. This will look something like this:



Task 3 (0.5 mark): Improve Watershed segmentation results.

Images to be used for this task: **coins.png** and **kiwi.png**

Step 1. Use the MeanShift and Watershed algorithms on these images as in the previous tasks.

Step 2. Notice that Watershed does not work very well for either image. Do some image manipulation to improve the Watershed segmentation.

Submit the final segmented images.

References

[Segmentation using Watershed Algorithm in Matlab](#)

[Meanshift Algorithm for the Rest of Us \(Python\)](#)

Copyright: UNSW CSE COMP9517 Team

Released: 25 June 2020