Coursework on Texture Segmentation

Deadline

10am 27th February 2019.

Submission

Submit work electronically via Minerva. This coursework is worth 20% of the credit on the module.

Time allocation

An indicative duration for this coursework is a total of around 12 hours.

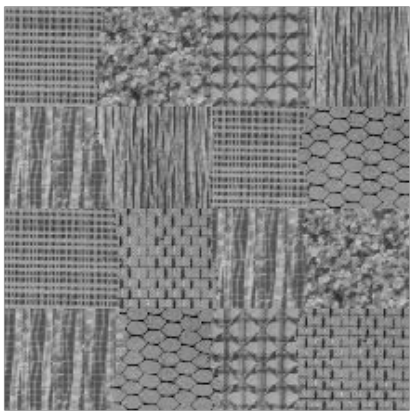
Motivation

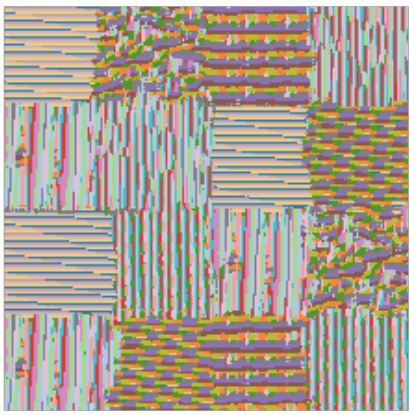
Through this coursework, you will:

- Learn about a standard problem in 'early' visual processing;
- Reinforce your understanding of some of the methods from the first weeks of the module and become more experienced in the use of Python;
- Gain experience in the evaluation and presentation of results.

Task

The assignment is to design and implement a Python program for segmenting images composed of a patchwork of textures. Four images like that shown on the left below are provided for testing:





All the images are composed of textures from a small number of classes, sometimes appearing more than once in the patchwork.

For each image, start out by finding a codebook of NxN patches, learnt either by clustering a large sample of patches from the image, or simply by clustering all possible patches. Use this codebook to label all NxN patches, producing an array of labels that can be visualised like the image on the right above.

The code for everything so far is provided in the Image Features Jupyter notebook on Minerva.

The next step is to generate histograms of the labels within a second fixed-sized MxM window that moves over the image (e.g. 25x25) (use np.bincount). Cluster the histograms using k-means, where K is the number of distinct textures observed in the source image.

Visualise the resulting array of labels as above, except that you are now doing this at a second layer of abstraction. The visualisations should look something like this, in which the different texture areas appear.

Notice that areas with the same label (colour) correspond to the same repeated texture.

Experiment by varying the window sizes at both levels (N and M), and the number of prototypes K at the first level. You may be able to improve on the result above, which was obtained using N=12, M=20, and K=20 at the first level.

Deliverable and Assessment

You should submit a report of **not more than 6 pages** (excluding Python code) in which you:

1. Describe the methods you have used to (1) learn features and produce an array of labels from these features, and (2) generate and then cluster histograms to produce an array of labels at the second level. (max. 30 marks).

2. Present the results of experimentation on the four images provided and comment on these results. The figures produced in Python can be copied into your report. (max. 40 marks).

3. Suggest one way in which you might improve the method given more time. (max. 10 marks)

4. Include the Python code for your program as an annex (max. 20 marks).