COMP 0137 Machine Vision: Homework #1
Due 22nd November 2022 at 16:00 (UK time)
Worth 10% of your overall grade
Submit online, through Moodle

For this homework, we'll revisit the practical from the 3rd week: Mixtures of Gaussians. There are two parts, so please read the instructions carefully. Everything you turn in must be YOUR OWN WORK, with one exception (in this case): the new images and their ground truth mask images. See below for more details, but as always, list names/references for anything you're submitting that is not your own work.

Late Policy: We must follow the official UCL late-policy, and this gets applied *after* your coursework is marked on Moodle, based on the Moodle timestamps. The instructor/TA's have no control over this – at all:
https://www.ucl.ac.uk/academic-manual/chapters/chapter-4-assessment-framework-taught-programmes/section-3-module-assessment#3.12

What to turn in (all inside one folder, zip'ed as a single .zip file):
- Three jupyter notebooks: for practicalMixGaussA.ipynb; also for B and C, but not D.
- Your jupyter notebook for practicalMixGauss_Apples.ipynb
- One folder containing the photos of apples
- One folder containing image masks for the apples

Code/Hints: All code must be python, submitted as jupyter python notebook files (.ipynb). If you're using VS Code, you can export to a notebook. Include your explanations interspersed as markdown within the notebooks. **Do not use other libraries beyond**: os, time, sys, numpy, matplotlip, scipy.io, glob, and pillow and/or opencv if you need it. You can use functions like randn(), but **not more advanced built-in normals-related functions**.

Details

Part I: A), B), C)
Do all the TO DO's in parts A, B, and C of the Mixtures of Gaussians practical. Some of the TO DO's are tagged a-j.

For every figure or plot that is generated in the code, write 1-3 sentences (maximum) explaining what the figure shows or pros/cons of what is happening, good or bad.

It IS NOT SUFFICIENT to just say things like "update the variable." Ok, some things are deterministic, but explain, to demonstrate you understand why those steps are happening. Examples of things to talk about, in the 1-3 sentences:
- Give some analysis about code working/not working out, especially when stochastic.
- Discuss where/which results don't match your expectations. Why? Be specific.
- Describe what **EACH** figure or plot is showing, what it would look like ideally.
- Discuss ways some step or experiment could be better / more robust. Validating your process means being thorough, and what would you need to be more thorough?

You do NOT need to do the 4th part, practicalMixGaussD.

(continued on next page…)

Part II Make a new file, practicalMixGauss_Apples.ipynb.
A) Download and unzip the file apples.zip. Notice that for every color photo containing apples, there is a corresponding binary image mask. In a mask image, white pixels indicate locations where the corresponding photo is an apple. In floating point, you may need to threshold to get binary values. Note that *these* mask images are inexact! While a perfect ground-truth mask image's black pixels should correspond to non-apples, these masks were painted in a hurry, so the white areas were painted conservatively.

```
# You may want to use this or similar example code, for
loading in your jpg's and png's:
import glob
import numpy as np
import matplotlib.pyplot as plt

files = glob.glob("apples/*.jpg")

ColorImgs = []
for myFile in files:
    im = plt.imread(myFile)
    ColorImgs.append(im)
```

B) Use mixtures of Gaussians to distinguish apple vs. non-apple pixels. Use red, green, and blue as your dimensions for now. Make any other decisions you need to, and document them in your .ipynb notebook.

C) Download the file testApples.zip. Generate figures for your notebook, showing each pixel's posterior probability of being "apple." Comment on the outcomes.

D) For the test image with a ground-truth mask, quantify and report your result. Hint: consider applying a range of thresholds to the posterior to produce sets of {True Positives (TP), True Negatives (TN), False Positives (FP), and False Negatives (FN)} and using an ROC curve. Learn about ROC on Wikipedia or see Peter Flach's chapter on the subject.

E) Download or two non-copyrighted photos with apples (maybe http://search.creativecommons.org/ or some other source of images that are not copyrighted). Make good ground-truth masks for them. You can use Windows Paint, or more sophisticated programs like Gimp (free). Use these as extra test-images. Report your qualitative and quantitative results.

F) We should really be using three separate sets of files: a training set, a validation set, and a test set! Explain why.

Optional extras (no points awarded): Put these at the end of the practicalMixGauss_Apples.ipynb notebook

- Consider manipulating the photographs' colors to improve the classification
- Consider running 2D Gabor filters on the photos to get additional channels of data, in place or in addition to red, green, and blue.
- Consider using an alternate model to mixture of Gaussians, and compare to MoG.