E9 241 DIP - Assignment 01

August 21, 2021

Due Date: September 05, 2021 **Total Marks:** 90 + 10

Instructions:

- For all the questions, write your own functions. Use library functions for comparison only.
- Your function should take the specified parameters as inputs and output the specified results.
- Also provide the wrapper/demo code to run your functions. Your code should be self contained i.e. one should be able to run your code as-is without any modifications.
- For python, if you use any libraries other than numpy, scipy, scikit-image, opencv, pillow, matplotlib, pandas and default modules, please specify the library that needs to be installed to run your code.
- Along with your code, also submit a PDF with all the results and answers to subjective questions, if any.
- Put all your files into a single zip file and submit the zip file. Name the zip file with your name.

Q1. **Histogram Computation:** Compute the histogram of the image `coins.png`. Verify your result using the MATLAB built-in function hist (or the corresponding function in python if you are using python).
Function inputs: grayscale image, number of bins
Function outputs: bin centers, corresponding frequencies (from both your function and MATLAB/Python function) (**10M**)

Q2. **Otsu's Binarization:** In the class, we showed that $\sigma_w^2(t) + \sigma_b^2(t) = \sigma_T^2(t)$, where t is the threshold for binarization. Compute the threshold t for the image `coins.png` by:

(a) Minimizing the within class variance σ_w^2.

(b) Maximizing the between class variance σ_b^2.

Verify that both methods are equivalent. Compare the time taken by each of the approach and also compare with the library function.
Function inputs: grayscale image
Function outputs: thresholds from both approaches, time taken by both approaches, binarized image and threshold from the library function. (**20M**)

Q3. **Foreground Extraction:** For the image `SingleColorText_Gray.png`, separate the foreground text from the background using otsu binarization. Display the text in red color on the green background in `GrassBackground.png`.
Function inputs: text and background images
Function outputs: an image with the text in red color superimposed on the background. (**15M**)

Q4. **Connected Components:** Binarize the image `PiNumbers.png` and count the number of digits $(0-9)$ using connected component analysis. Also compute the number of occurrences of the digit 1.
Function inputs: `PiNumbers.png` image
Function outputs: total number of digits, number of occurences of digit 1. (**30M**)

Q5. **Binary Morphology:** Binarize the image `NoisyImage.png` and apply binary morphological operations to remove the noise in the image.
Function inputs: noisy image
Function outputs: cleaned image. (**15M**)

Q6. **Optional Bonus Question - MSER:** Maximally Stable Extremal Regions (MSER) correspond to regions of connected components which when thresholded around a certain threshold are stable in terms of the size of the component. This allows an adaptive thresholding method where different regions can be thresholded using different thresholds automatically. Determine the binarized image for `DoubleColorText\_Gray.png` based on MSER using the following steps:

(a) Sweep over all thresholds.

(b) For each threshold, determine connected components in the image.

(c) A connected component is termed an MSER if the size of the component does not change much (within δ) for a small perturbation ϵ in the choice of the threshold. Note that both δ and ϵ are parameters that need to be chosen. Determine the stable threshold for each connected component.

(d) Ignore extremely large or extremely small connected components in the above analysis.

Compare the performance of Otsu's binarization with the above method on `DoubleColorText_Gray.png`. Think about why Otsu will not be able to extract both the words in this case.
Function inputs: the `DoubleColorText_Gray.png` image
Function outputs: MSER binary image, Otsu binary image, number of connected components detected by MSER, number of connected components when applied on Otsu binarized image. (**10M**)