E9 241 Digital Image Processing

Assignment 01

Due Date: September 3, 2023 - 11:59 pm **Total Marks:** 75 + 15

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** (images or numbers) and **inferences** (very important: you may not be explicitly asked to give inferences in each question. You should always include your inferences from what you have observed). Include answers to subjective questions, if any.
- Put all your files (code files and a report PDF) into a single zip file and submit the zip file. Name the zip file with your name.

1. **Histogram Computation:** Compute the histogram of the image `coins.png`, by finding the frequency of pixels for each intensity level $\{0, 1, \ldots, 255\}$. Show the histogram by plotting frequencies w.r.t. intensity levels. Comment on what you observe. Also, find the average intensity of the image using this histogram. Verify the result with the actual average intensity.

Function	Histogram
Input	Grayscale image
Output	Frequencies at each intensity level ((a list/vector of size 256))

(10 Marks)

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

   (a) Minimizing the within class variance $\sigma_w^2(t)$ over t.

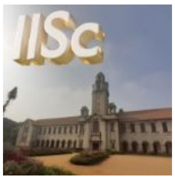
   (b) Maximizing the between class variance $\sigma_b^2(t)$ over t.

   Verify that both methods are equivalent. Compare the time taken by each of the approaches.

Function	Within class variance	**Function**	Between class variance
Input	Grayscale image, threshold	Input	Grayscale image, threshold
Output	Within class variance	Output	Between class variance

(20 Marks)

3. **Depth based Extraction:** The image `IIScTextDepth.png` is an inverse depth map of `IIScText.png`. A depth map indicates the depth of an object from the camera for each pixel. Particularly, an inverse depth map has a higher value when the object is nearer to the camera and a lower value when it is farther apart. Binarize the inverse depth map `IIScTextDepth.png` and use that information to extract the text in `IIScText.png` and display it over the background image `IIScMainBuilding.png`. The expected image is shown below.

Function	Image superimpose	
Input	Text image, depth image, background image	
Output	An image with the text superimposed on the background image	

(15 Marks)

4. **Connected Components:** Binarize the image `quote.png` and count the total number of characters excluding punctuations using connected component analysis.

Function	Count characters
Input	Image
Output	Number of characters

(30 Marks)

5. **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. Determine the number of characters in `Characters.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.

   Think about why finding connected components over an Otsu binarized image will not work well in this scenario.

Function	MSER
Input	Image
Output	Number of characters

(15 Marks)