Assignment 4

Digital Image Processing - M25

Deadline: 18th November 11:59PM

General Instructions

1. Make sure you run your Jupyter notebook before submitting to save all outputs.

2. If python files are used, make sure to save all the plots in a seperate directory for submission.

3. Answer any questions asked in a markdown cell in your notebook.

4. Proofs can be typed or written.

5. Keep pushing your code as you progress, not just at the end. That is the purpose of using git instead of Moodle.

6. the functions in Q5, Q6 and Q7 need to be implemented from scratch. You are free to use cv2 and any numpy functions for the other questions.

7. you are free to try multiple methods for each question, analyzing pros and cons of each.

1 Question (10 Marks)

Harsha is facing a frustrating issue: due to his poor internet connection, images on his machine are being rendered in low resolution, making it hard for him to enjoy a clear view. As a solution, you are tasked with writing a code that can enhance the resolution of `google.jpeg` and restore its clarity.

2 Question (10 Marks)

Harsha recently took a picture of some coins scattered on his desk but, being a bit lazy, he doesn't feel like counting the coins manually. Your task is to help Harsha by identifying and counting the number of coins in `coin.jpg` using morphological operations. The goal

is to automate the process, ensuring that Harsha never has to manually count his coins again!

3 Question (10 Marks)

The human skeletal system is a fascinating and complex structure made up of numerous bones, but have you ever wondered if it's the most efficient design? To explore this concept further, you are tasked with performing skeletonization on the `human.png`.

4 Question (10 Marks)

Harsha is trying to mark his attendance. However, because his hands weren't clean, the scanned fingerprint image contains various artifacts and noise, making it difficult for the system to process his attendance. Your task is to help Harsha by cleaning the `fingerprint.png` and detecting the key edges needed for accurate recognition.

5 Question (10 Marks)

Load `thresholding.png` and implement the following thresholding methods to segment the image:

- Binary Thresholding

- Adaptive Thresholding

- Otsu's Thresholding

6 Question (15 Marks)

Implement a function to detect circles in an image using the **Hough Circle Transform**. Use suitable preprocessing steps (e.g., smoothing) to improve detection accuracy. Apply your function to `hough.jpg`. Display the original image and the processed image with detected circles overlaid.

7 Question (15 Marks)

Create a function to detect and mark corners in an image using the **Harris Corner Detection** method. Use an appropriate method to mark the detected corners on the original image. Apply your function to `harris.jpg`. Display the original and processed images with corners marked.

8 Question (20 Marks)

You are provided with a scanned OMR answer sheet image (`omr_sheet.jpeg`). Your task is to design a system that can automatically evaluate the marks of any student based on their OMR sheet.

The system should be able to:

- Detect the roll number from the roll-number bubble area.

- Identify the filled answer bubbles for all questions.

- Compute the total marks obtained by the student using the provided `answerKey.json`.

The exam contains **150 single-option-correct questions.**

Marking scheme:

- +4 marks for each correct answer

- -1 mark for each incorrect answer

- -0.5 marks for each unattempted question (no bubble filled)

Constraints & Assumptions:

- You may assume the OMR layout and image resolution are identical for all students (same bubble grid locations and same image size).

- The code should be robust to edge cases, such as skipped questions or partially filled bubbles.

- Do not use OCR libraries or hard-coded answers. Solve the problem using only image-processing techniques — thresholding, morphological operations, contour detection, connected components, blob analysis, template matching, etc.

Bonus - 20 Marks (only counts towards any marks lost in the assignments):

Attempt to make your system robust to varying image resolutions and small rotations or translations.