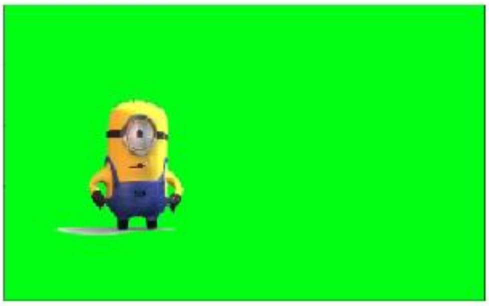
Programming Assignment-1 on course CSL442_IVP

Solve five questions out of six. Each question reserved 3 marks. Evaluation date: 15 Feb 2024

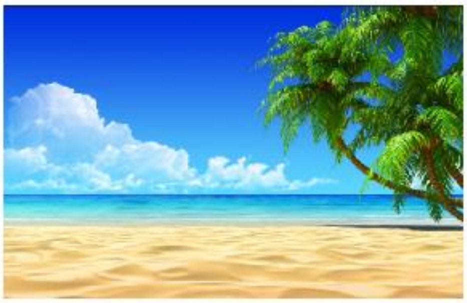
Q1.

(a) Create a function capable of converting a color image to gray scale and determining the predominant intensity value by finding the one that occurs most frequently in the image.(use ‘tq.jpg’ image)

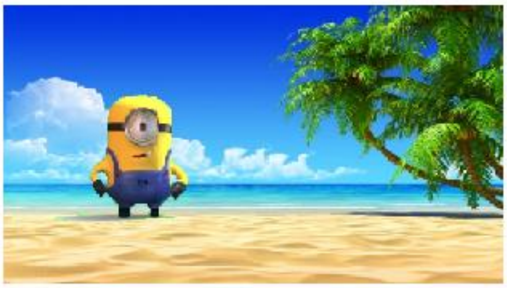
(b) Create a function to merge the images **foreground.jpg** and **background.jpg** as per the result given below.



Foreground



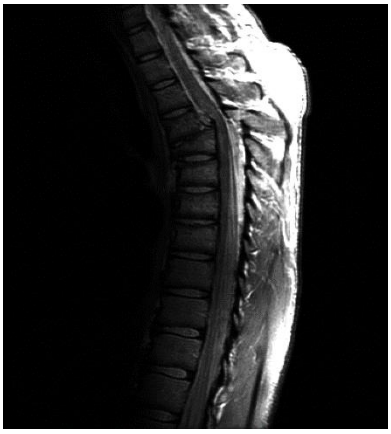
Background



Merged

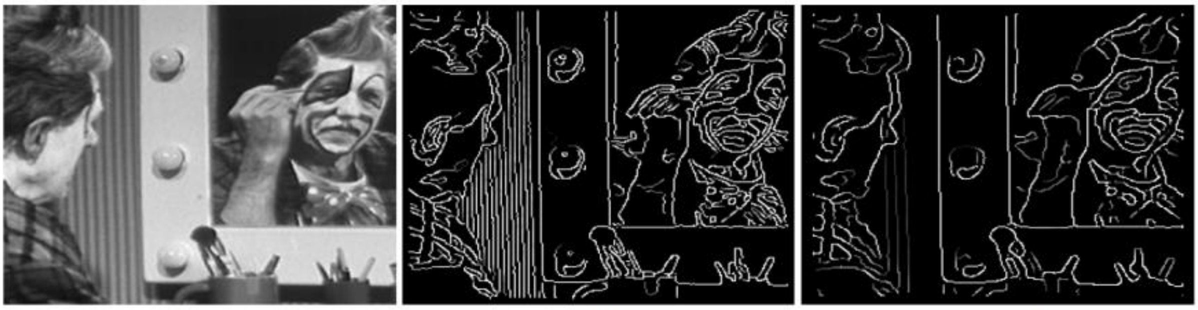
Q2. Create a function called **ImageBitQuantizer** that accepts an 8-bit image (**im**) and an integer **k** representing the desired number of bits for image quantization. The function should return the image quantized to k bits. Demonstrate the results of the function using the image named "pixel.jpg."

Q3. For the attached image ‘spine.tiff’, enhance it using (a) The log transformation and (b) A power-law transformation. In (a) the only free parameter is c, but in (b) there are two parameters, c and r for which values have to be selected. By experimentation, obtain the best visual enhancement possible with the methods in (a) and (b). Once (according to your judgment) you have the best visual result for each transformation, explain the reasons for the major differences between them. (Use spine.tiff image)



Q4. Write a program to implement spatial domain averaging filter, weighted averaging filter, median filter of size 3X3, 5X5 and observe its blurring effect on the given noise.tiff image, also find the optimum size of the kernel for median filter for which the features in the image ‘noise.tif’ are the clearest with minimum noise and deformation of the features. Do not use inbuilt spatial filtering function.

Q5. Write an edge detection program by using the inbuilt canny edge detection function and see the results with different parameters like threshold, smoothing variance etc., and note your observations. Also implement the Laplacian filter and Sobel filter and observe the anomalies in their respective outputs. (Output image for canny edge detection shown below) (Use clown.png image)



original Canny with $\sigma = 1$ Canny with $\sigma =2$

Q6. Write a program to find the largest correlation spot in the given image (hills.jpg) using linear filtering-based template matching technique. Draw rectangular bounding box at the detected template (template.png) matched locations. (Use hills.jpeg, template.png images)