

Assignment 1

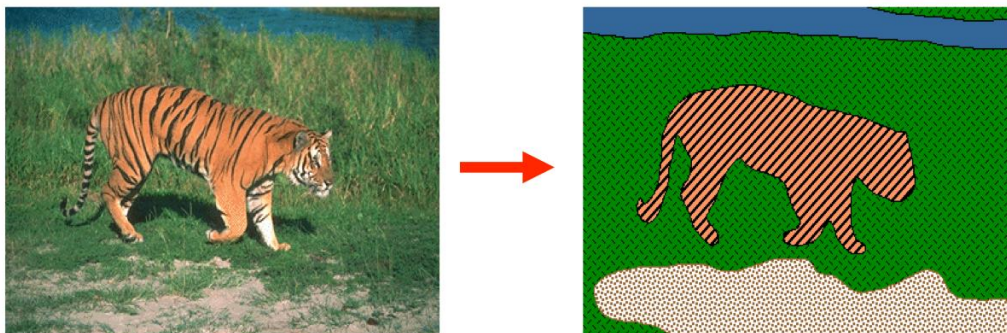
Dated: Mar 2, 2024, Submission due: to be announced

Q1. Write a Python program that takes any one of the RGB image from the given database as input, converts it to a quantized image by using k-Means image quantization at varying 'k' values i.e. 1, 2, 4, 8, 16. Also, calculates the Mean Squared Error (MSE).

Your program should include the following steps:

- Read an RGB image as input.
- Implement a function to perform k-Means image quantization.
- Convert the input RGB image to a quantized image using the k-Means algorithm by varying 'k'.
- Calculate the Mean Squared Error (MSE) between the original RGB image and quantized image.
- Analyze and discuss the effectiveness and limitations of k-Means image quantization in the context of the calculated MSE.

Q2. Write a code to find the connected components present in an image as shown in the given example. Utilize the provided image in the database 'Connected component' to identify the components, emphasizing accuracy.



Q3. From the attached database of standard test images, select the gray scale 'lena_gray_512.tif' image of dimension pixels. Perform the following operations:

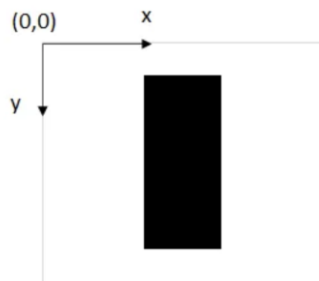
- Reduce the image size by down sampling to 256x256, name this image as 'reduce_lena_256.tif'.
- Compare the reduced image 'reduced_lena_256.tif' with the given image in the database 'lena_gray_256.tif' by finding the PSNR values between them. Also plot the image showing MSE difference for the same.
- Now again resize the computed image 'reduced_lena_256.tif' to original dimension pixel using these methods:
 - Nearest Neighbour Interpolation
 - Bilinear Interpolation
 - Bicubic Interpolation

Q4. Develop codes in python to obtain the filtered imaged from the noisy image given to you as input 'Noisy image' in the database. Show the steps involved to be applied for filtering and explain why?



Q5. Take any one image from the given database and compute:

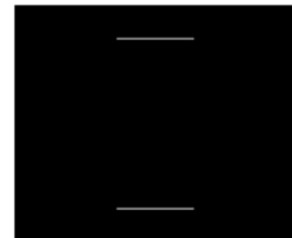
- First order derivative along 'x'
- First order derivative along 'y'
- Gradient image magnitude and followed by thresholding to output edge map.
- Edge map computed using second order derivative
- Edge map using LoG and Canny edge detector



(a)



(b)



(c)

Q6. Apply the bit plane slicing of the following image size 3x3, for an 8-bit image “0” is encoded in 00000000 and “255” is encoded in 11111111

167	133	111
144	140	135
159	154	148

Implement the bit-plane slicing process using a programming language of your choice (e.g., Python with OpenCV). Provide the code snippet and explain each step in the implementation.