German University in Cairo
Faculty of Media Engineering and Technology
Winter 2021

DMET 901 – Computer Vision

Assignment #1

(Due on November 8 at midnight – Submit to dmet901.w21@gmail.com)

The main aim of this assignment is to generate the indexed image of a colored image which includes the index map (Data Matrix) and the color map. Moreover, apply different quantization levels to the colored image using the indexed image to diffuse some of the colors. Therefore, the assignment is structured as follows:

1. Task 1: Indexed Image
    a. Index Map
    b. Color Map
2. Task 2: Applying Different Quantization Levels

Task 1: Indexed Image:

In this task, you are asked to implement one function as follows:

1. CalculateColorMap:
    - Input: Image
    - Output: 2D array representing the Index map and a dictionary representing the color map.
    - Description: Generates the indexed image by returning both the index map and the color map (color map is dictionary having the keys as (Red value, Green value, blue value) and the values as indices)

Task 2: Applying Different Quantization Levels:

As discussed in class, reducing the number of quantization levels results in false contours in the image. Generating a colored image with a reduced number of quantization levels can be achieved by first representing the colors as 3-dimensional vectors, where the 3 values correspond to the Red, Green and Blue components. Second, vectors that are similar could be then considered to represent a single quantization level in the reduced quantization levels image. Similarity could be determined based on differences between the vectors in a difference range of +/- some constant value to be added or subtracted to the Red, Green and Blue components.

Note: The run time differs according to the different numbers of colors. Also, by increasing the difference range to be added or subtracted, the run time also increases.

DMET 901 – Computer Vision

Assignment #1

(Due on November 8 at midnight – Submit to dmet901.w21@gmail.com)

In this task, you are asked to implement four functions as follows:

Required Functions

1. QuantizationLevels:
    - Input: dictionary representing the color map and integer representing the difference range to be added or subtracted from the color values.
    - Output: A pair of dictionaries one representing the color map after quantization and the other one represents the removed colors that have been combined with others during the re-quantization procedure.
    - Description: The first dictionary stores the colors after re-quantization with the values of their indices, and the second dictionary represents the deleted colors together with their indices and this is done after checking the range to be added or subtracted from the 3-dimensional color vector (Red, Green, and Blue).
2. AdjustIndex:
    - Input: 2D array representing the index map and dictionary representing the removed colors.
    - Output: 2D array representing the new index map after adjusting the indices of the removed colors.
    - Description: Fixing the indices of the removed colors to map the index kept in the new color map after quantization in order to take the same color which is quantized to it.
3. ReverseColorMap:
    - Input: Dictionary representing the new color map after quantization.
    - Output: Dictionary representing the new color map after reversing.
    - Description: reverse the keys and the values of the color map dictionary to be (key: index and the value: color value which is red, green, blue).
4. ColorMapToImage:
    - Input: 2D array representing the index map and dictionary representing the color map.
    - Output: Image constructed using the index map and the color map.
    - Description: construct image using the index map and the color map by putting the pixel value referred to the index in its place.

German University in Cairo
Faculty of Media Engineering and Technology
Winter 2021

DMET 901 – Computer Vision

Assignment #1

(Due on November 8 at midnight – Submit to dmet901.w21@gmail.com)

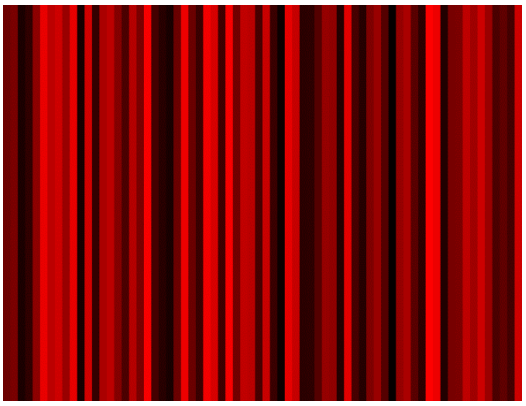
Deliverables:

- Your Python code. You can use the sample images discussed below (files provided with the assignment) to test your code.
- The output of Task 1 and Task 2 when applied to the image "Test Image.jpg". Show the output of Task 2 for difference ranges of 10, 20 and 30. Save the outputs as .jpg files.
- Comment on the results of each image before and after applying different quantization levels.

- **Send your solution as one zip file to dmet901.w21@gmail.com**
- **This assignment can be done in groups of maximum 2 students. Both students must be from the tutorial groups of the same TA.**
- **In the subject of your e-mail, write the student ID of both students.**

Here are samples of the required output:

First Sample Image:

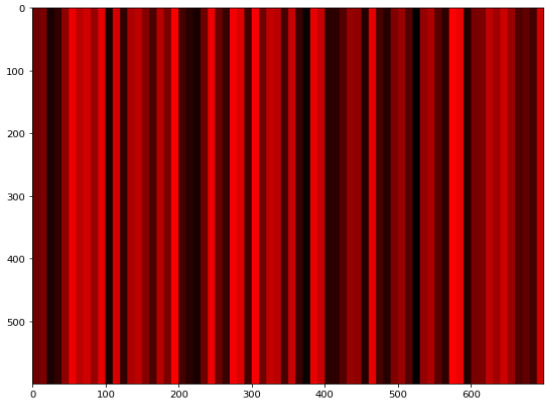


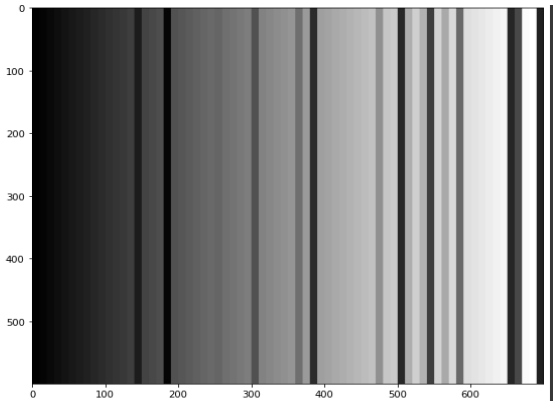
DMET 901 – Computer Vision

Assignment #1

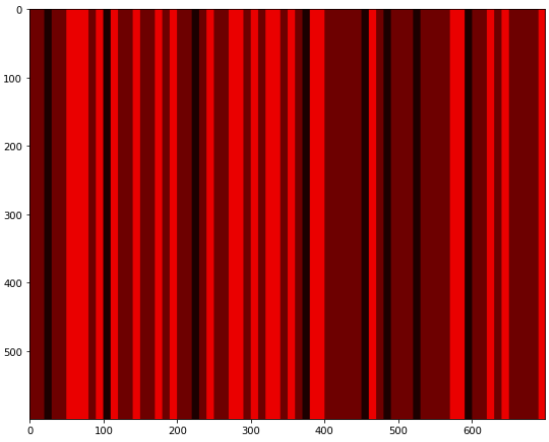
(Due on November 8 at midnight – Submit to dmet901.w21@gmail.com)

Here is the reconstructed image form the indexed image on the left and the corresponding index map on the right. **Note that the image on the right is not a gray scale representation of the colored image. It represents the index map.**





Here is the output of the indexed image after quantization by a difference range of 70 and reconstructed image form the indexed image on the left.

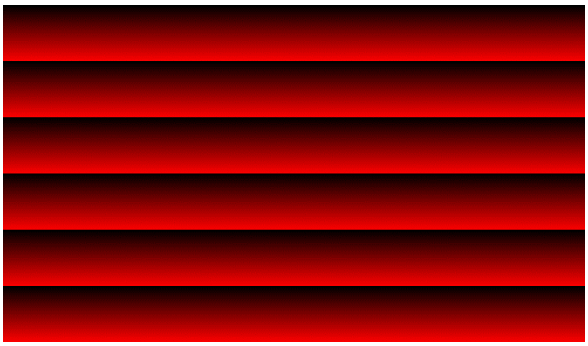


DMET 901 – Computer Vision

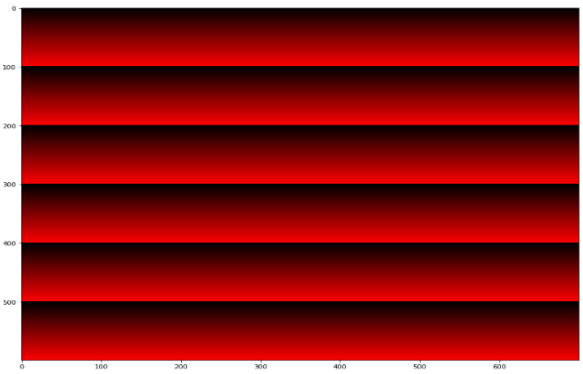
Assignment #1

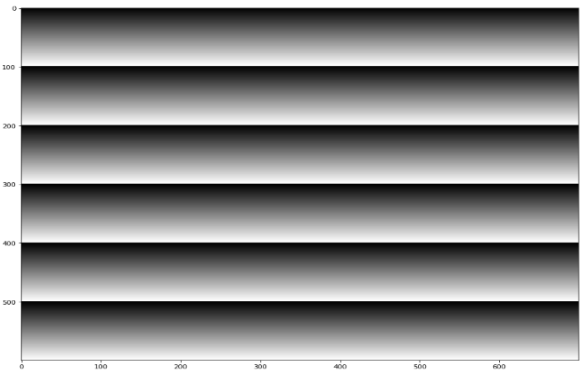
(Due on November 8 at midnight – Submit to dmet901.w21@gmail.com)

Second Sample Image:

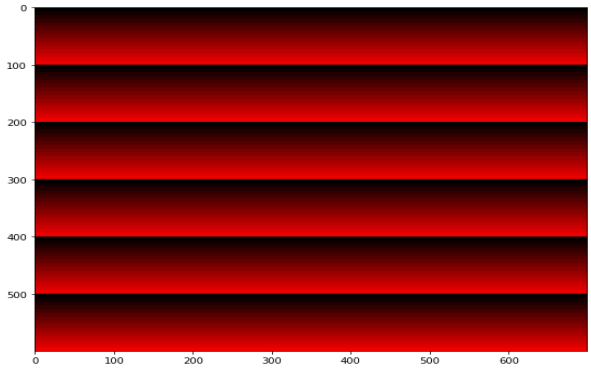


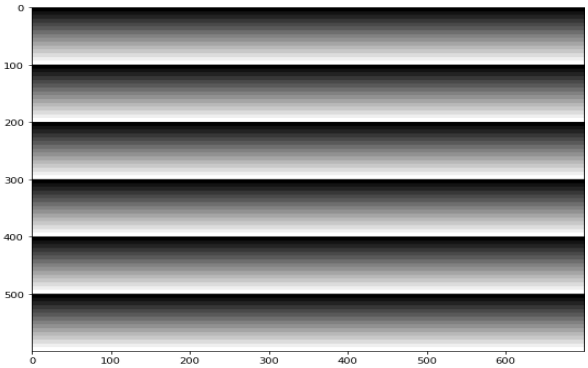
Here is the output of the indexed image before quantization and reconstructed image form the indexed image on the left.





Here is the output of the indexed image after quantization by a difference range of 15 and reconstructed image form the indexed image on the left.



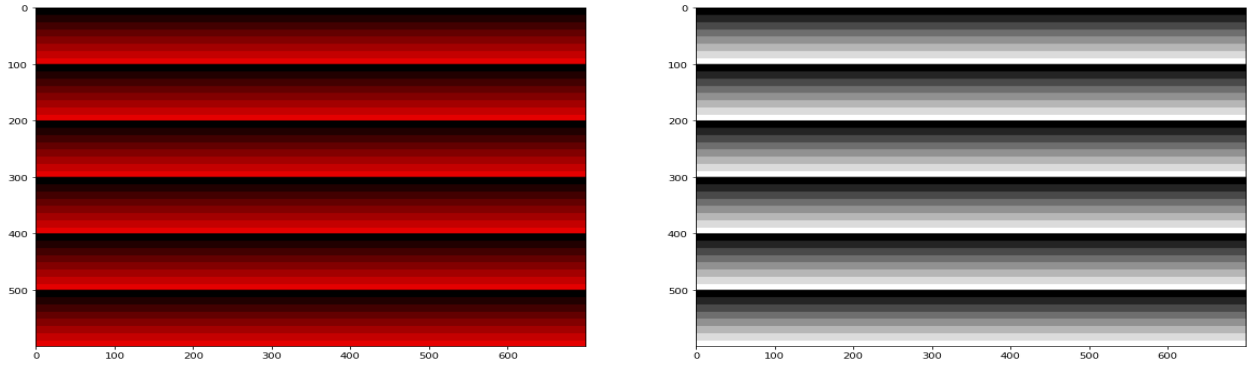


DMET 901 – Computer Vision

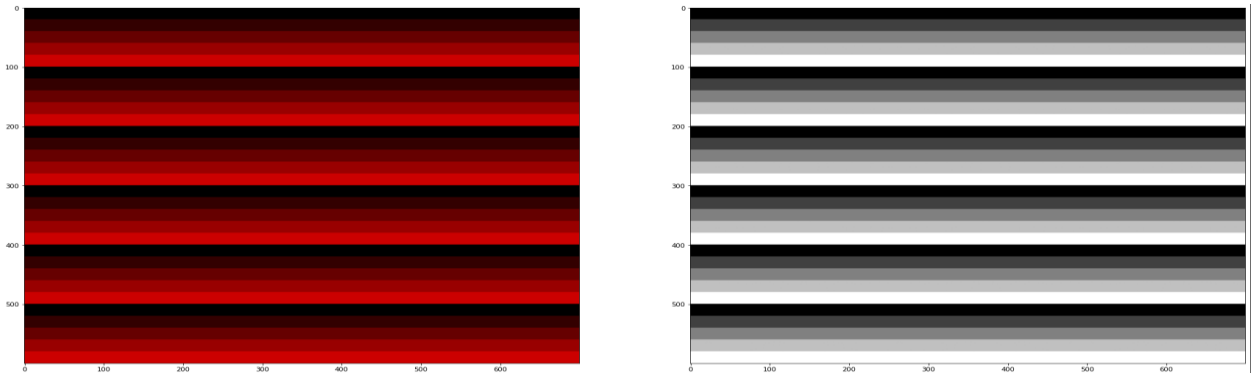
Assignment #1

(Due on November 8 at midnight – Submit to dmet901.w21@gmail.com)

Here is the output of the indexed image after quantization by a difference range of 30 and reconstructed image form the indexed image on the left.



Here is the output of the indexed image after quantization by a difference range of 50 and reconstructed image form the indexed image on the left.



Third Sample image:

DMET 901 – Computer Vision

Assignment #1

(Due on November 8 at midnight – Submit to dmet901.w21@gmail.com)

Here is the output of the indexed image before quantization and reconstructed image form the indexed image on the left.


Here is the output of the indexed image after quantization by a difference range of 10 and reconstructed image form the indexed image on the left.


Here is the output of the indexed image after quantization by a difference range of 20 and reconstructed image form the indexed image on the left.


DMET 901 – Computer Vision

Assignment #1

(Due on November 8 at midnight – Submit to dmet901.w21@gmail.com)

Notes:

- In the provided outputs for the sample images, we were looping over the indexed image on columns first from left to right.

Example of the dictionary data structure in python:

```
col = {(200,231,55):1}
print(col[(200,231,55)])
print((200,231,155) in col)
col[(200,231,155)] = 0
print((200,231,155) in col)
del col[(200,231,155)]
print((200,231,155) in col)
```

Starter Code:

```
import numpy as np
from PIL import Image
from matplotlib import pyplot as plt
```

Getting pixel value:

```
r,g,b = img.getpixel((0,0))
```

Inserting pixel value:

```
img.putpixel((0,0),(r,g,b))
```

Plotting the reconstructed image together with its indexed image:

```
plt.subplots(nrows=2, ncols=2, figsize=(20, 20))
```

DMET 901 – Computer Vision

Assignment #1

(Due on November 8 at midnight – Submit to dmet901.w21@gmail.com)

```
#original image reconstructed from indexed image
plt.subplot(2,2,1)
plt.imshow(imgR)

#indexed image of the original image
plt.subplot(2,2,2)
plt.imshow(im, cmap = 'gray')

#image reconstructed after quantization by certain range
plt.subplot(2,2,3)
plt.imshow(imgN)

#indexed image of the quantized image by certain range
plt.subplot(2,2,4)
plt.imshow(indxMapN, cmap = 'gray')
```