Worksheet 1

Fundamentals of Image Processing
CMPE 403

October 11, 2022
Deadline: October 18, 2022

Complete the exercises below using `Python`. These exercises are aimed to teach you the basics of `Python` and `Numpy`. You can see some hints for exercises in the footnotes.

You should display the matrices in between every step in order to see the changes.[1]

1 Exercise

1. Create a 128×128 matrix named M_1 .
2. Set all values to 0 in every row which has an even index.[2]
3. Set all values to 120 in every row which has an odd index.
4. Rotate the matrix $90°$.[3]
5. Create a 128×128 matrix named M_2.
6. Set all values to 0 in every row which has an even index.
7. Set all values to 120 in every row which has an odd index.
8. Calculate sum of the matrices M_1 and M_2, assign the result to variable M_3.
9. Set all values of M_3 which are greater than 200 to 255, and lower than 200 to 0.
10. **Extra:** Create an RGB image from M_3 with a red and blue grid.[4]

2 Exercise

1. Read the image `einstein.jpg` as an array, assign to variable `img`.
2. Apply binarization (thresholding) to `img` and assign the result to variable `img_bin`.[5].
3. **Extra:** Implement simple thresholding only using `Numpy`, show and compare your result with `OpenCV`.[6]

[1] You may use `Matplotlib` or `OpenCV`.
[2] Check indexing operations from `Numpy` documentation.
[3] You do not need to use OpenCV to rotate the image, think about what $90°$ rotation corresponds to in a *square matrix*.
[4] OpenCV uses BGR format by default. In order for an array to be percieved as colored, it has to have three dimensions.
[5] See different types of thresholding methods in `OpenCV` docs
[6] This is a one liner.