Istanbul Technical University
Computer Engineering Department
BLG 453E - Computer Vision - Fall 25/26

Due: up.to.youu | Fall 25/26, Semester | Assignment 1

Notes

- You should do your own work! . Cheating is highly discouraged.
- You can **not** use built-in Python functions, Numpy, and OpenCV functions until otherwise stated.
- You should write your codes in Python.
- For your questions, do not hesitate to reach Res. Asst. Tuğçe Temel (temel21@itu.edu.tr).

Q1) PDF of an Image from Scratch

Problem. Consider a small grayscale image whose pixel intensity histogram (binned) is given below. The image has 100 pixels total.

Intensity bin	Pixel count
0	10
64	20
128	40
192	20
255	10

Answer the following questions (show your work):

1. Compute the **Probability Density Function (PDF)** for each bin (i.e. $p(i)$ for $i \in \{0, 64, 128, 192, 255\}$).
2. Compute the **Cumulative Distribution Function (CDF)** values for each bin.
3. Compute the **expected (mean) intensity** $E[I]$ of the image (treat each bin value as the representative intensity).
4. Compute the **variance** $\mathrm{Var}(I)$ (use the population variance definition).
5. What is the probability that a randomly chosen pixel has intensity greater than or equal to 128?

Notes: treat the given bin intensities as exact intensity values (i.e., assume each pixel in the bin has that intensity).

Q2) O Gama, Gama, wherefore art thou Gama 🎭 ?

In digital image processing, gamma correction is a nonlinear operation used to adjust the brightness of an image. Human vision does not perceive brightness linearly, and display devices (such as monitors and cameras) also apply gamma encoding to represent intensity values more efficiently.

This assignment focuses on enhancing a low or high light image using gamma correction, demonstrating how different gamma values affect visual perception and detail visibility. The test image contains strong contrast: a bright sun with dark foreground elements where details are not clearly visible.

Problem Description

You are given the image in Figure 1. The task is to enhance the image using gamma correction and analyze the effect of different gamma values.

Gamma correction is defined as:

$$I_{\text{output}} = \left(\frac{I_{\text{input}}}{255}\right)^{\gamma} \times 255 \tag{1}$$

Where:

- $\gamma < 1$ brightens the image (useful for dark scenes).
- $\gamma > 1$ darkens the image (useful for overly bright images).
- I_{input} and I_{output} are pixel values in the range [0, 255].

Figure 1: Original Low-Light Image

1. Load the provided image in Python using OpenCV or similar tools.
2. Apply gamma correction using at least three different gamma values:
   - $\gamma = 0.4$
   - $\gamma = 1.0$
   - $\gamma = 2.2$
3. Display and compare all results.
4. Extract and display the transformation curve for each gamma value.
5. Answer the following questions:
   (a) Which gamma value provides the best detail enhancement?
   (b) Why is gamma correction more effective than simply increasing pixel intensity?

Q3) Extract color by masking them

In digital image processing, different color spaces are used to manipulate, analyze, or enhance images more effectively. While the RGB (Red-Green-Blue) color model represents color based on light emission levels, the HSV (Hue-Saturation-Value) color space describes colors in a way that is more aligned with human perception — separating color tone, vividness, and brightness. In this assignment, you are required to convert an image from the RGB color space into the HSV color space and analyze the components.

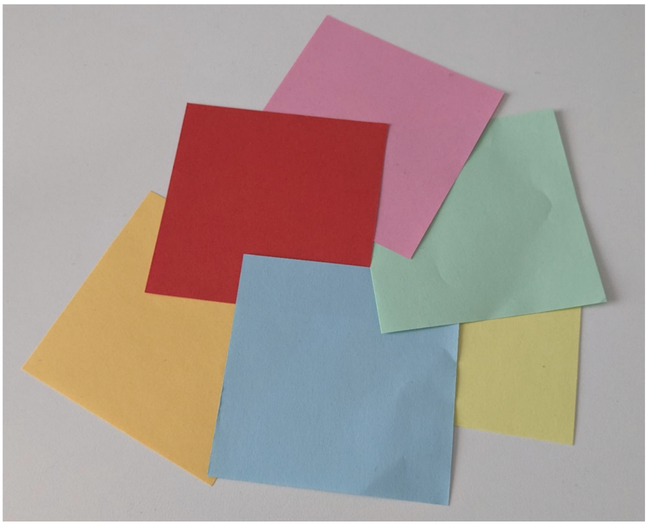

Figure 2: Can you separate us without scissors ?

You can follow these instructions to complete the parts, respectively:

- Read provided RGB image (Figure 2) .

- Convert the image from RGB to HSV color space
- Segment image using the special colors ranges.
- Explain briefly the following question with3–4 sentences:
  *Why color space conversion is useful in image processing
  *What H, S, and V values represent
  *One real-world application example of HSV using area (e.g., skin detection, object tracking, segmentation)

Q4-) 👓 Looking is with the eyes, but seeing is with the mind!🧠

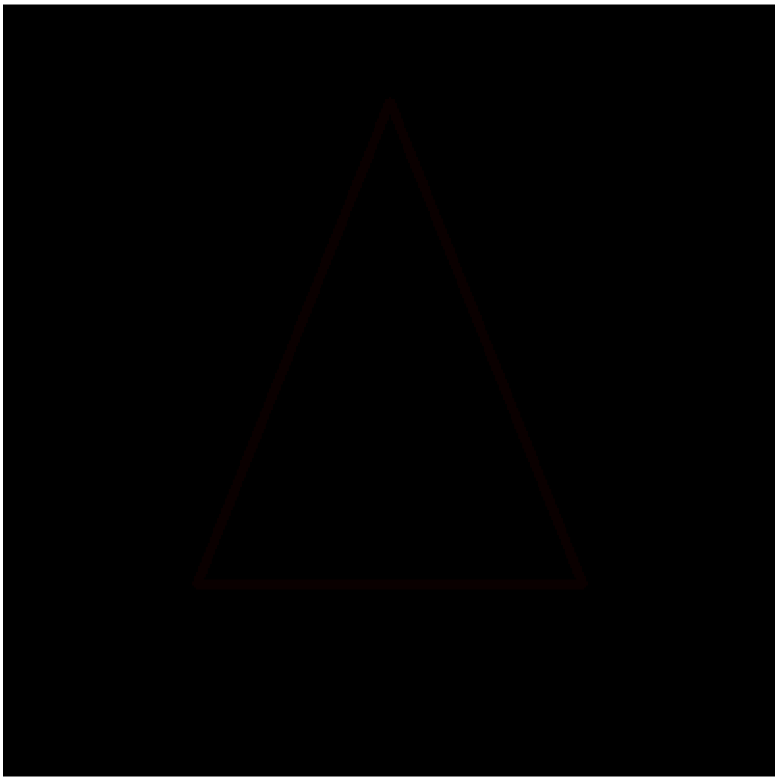

Figure 3: I am hiding something, can you uncover it 🙃 ?

In digital image processing, colors are represented using multiple channels such as Red (R), Green (G), and Blue (B). Although an image may appear normal at first, a secret is hidden within the colors, which only becomes visible when remapping techniques are applied. The purpose of this homework is to demonstrate how 'looking' at an image (simply viewing it in RGB form) is not the same as 'seeing' hidden information encoded within its channels.

These shapes remain invisible in the normal RGB view of the image (called findme.png, which is provided by us). However, when specific channel remapping rules are applied, you can find the secret.

If you are as curious as I, just follow my instructions to uncover the secret of the image.

- Read the provided image, which is Figure 3, with cv2 (Hint: Be aware of the color space when using the cv2)
- Show the image
- Remap the color image as R= -R -G -B
- Show the remapped image

That is all the steps:)