

# Digital Image Processing

## Week 02 Assignment Report

Umidjon Gofurov

## Overview

This assignment explored fundamental digital image processing techniques using Python libraries such as OpenCV and NumPy. The main objective was to understand how images are represented numerically and how mathematical operations influence their visual properties.

## Digital Image Representation

A digital color image can be described as a three-dimensional array. The first two dimensions correspond to the spatial coordinates (height and width), while the third dimension represents the color channels. In the RGB color model, each pixel contains three intensity values representing red, green, and blue components. When stored using the `uint8` data type, pixel values range from 0 to 255.

Region-of-interest (ROI) extraction was performed using array slicing. Since images are stored in row-major order, indexing follows this structure:

```
img[y1:y2, x1:x2]
```

where the row index (y-coordinate) is specified first, followed by the column index (x-coordinate).

## Thresholding and Segmentation

Two thresholding techniques were applied in this assignment. Manual thresholding uses a predefined intensity value to separate foreground and background pixels. In contrast, Otsu's method automatically determines an optimal threshold based on the histogram of the image. The automatic approach typically provides improved segmentation when illumination conditions vary across the image.

## Brightness and Contrast Adjustments

Image brightness and contrast were modified using arithmetic operations. Adding a constant value increases overall brightness, while subtracting decreases it. Multiplying pixel intensities changes contrast and scales the image values. When working with `uint8` images, it is necessary to clip pixel values to the valid range [0, 255] to prevent overflow or underflow.

## Conclusion

This assignment improved my understanding of how digital images are structured and manipulated. Through cropping, thresholding, and intensity transformations, I gained practical experience with pixel-level operations. Additionally, I learned the importance of handling data types and color channel ordering correctly when using OpenCV. These concepts form a strong foundation for more advanced image processing tasks.