AI Lab Pratical1 : Digital Image and Geometric Transformation

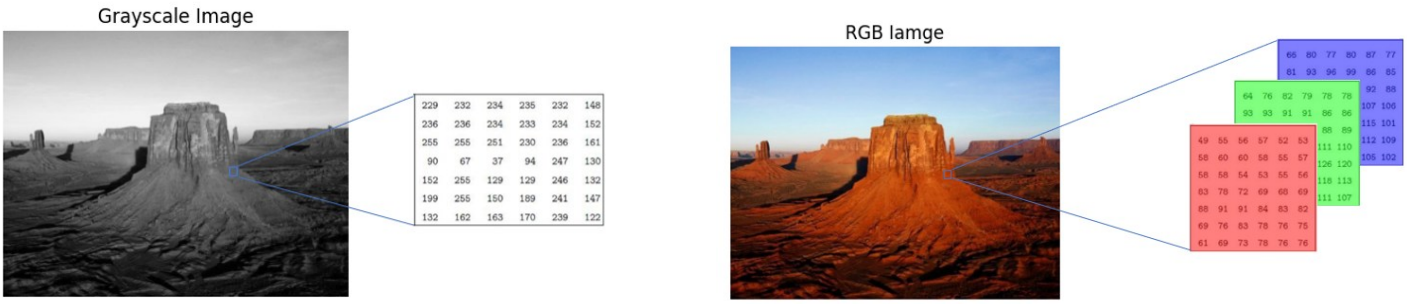
CHHEANG Vinha

October 5, 2025

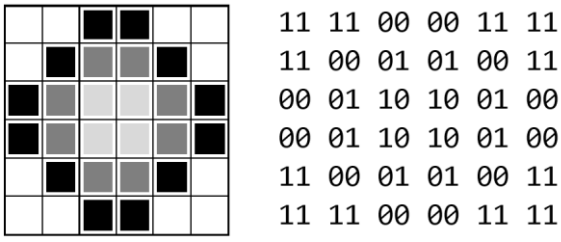
1.Introduction : Digital Image

A digital image is a numerical representation of a visual object, composed of picture elements called pixels, each having an intensity value ranging from 0 to 255. In this section, we will focus on two types of images:

- Grayscale Image: Each pixel has a single value [0,1,…225], where 0 represents black and 255 represents white.
- Color Image (RGB): Each pixel is represented by three values Rad, Green, and Blue with intensities ranging from 0 to 255.



Question 1: Given a grayscale image and matrix of pixels value bellow:



1.1. What are the dimensions and pixel values of this image? Create an array from the given data and use Matplotlib to visualize the image in Python. Does the output image look the same as the original image? Why or why not? If it does not look the same, explain how you can transform it to match the original image.

1.2. Now we want reduce the size of original pixel to 4x4 using function `resize(m,n)` in OpenCV, Show result by plot this image after resize and check the pixels value again. Any different from the original?

1.3. Generate another two-color channel Green and Blue your self by keeping the Red channel has the same data give in order to create a color image. What do you observe when you switch the color order from (R, G, B) to (B, G, R) using a subplot layout of 1×2 (1 row and 2 columns)?

Question 2: Given that color (R-G-B) image `vo.png` from data folder name `image_test_1` you can use your picture. In this section we will work on real properties image and relation between color (R-G_B) to grayscale. For loading this image and setting name `'im_cv'` by using function `cv2.imread("vo.png")` in OpenCV, `'im_plt'`

by using function `plt.imread("vo.png")` in matplotlib . Let an array name `'arr_image'` to be keep all intensity pixels (R_o, G_0, B_o).

2.1. Display this image using library OpenCV and matplotlib. Any different from both result and give remark?

2.2. Now let convert from color to grayscale by setting an array name '`arr_gray1`' is defined by $\frac{1}{3}(R_0 + G_0 + B_0)$ and '`arr_gray2`' is defined by the linear combination $0,2126\mathrm{R}_0 + 0,7152\mathrm{G}_0 + 0,0722\mathrm{B}_0$. Plot these images following two arrays above and finally using function directly `imread("image_name", cv2.IMREAD_GRAYSCALE)`

2.3. Now loading the image name `a.png` from folder name `image_test_1`, checking the image properties it is a grayscale or color (R-G-B)? why? It not grayscale so transform this image to grayscale.

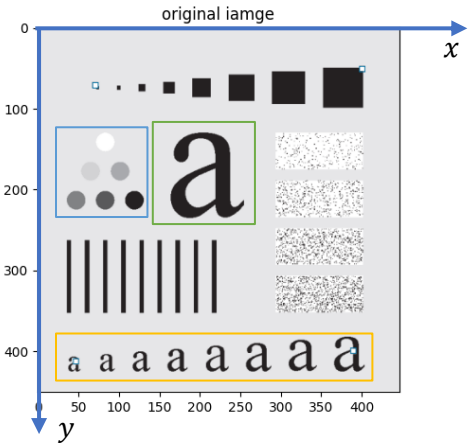
2. Cropping an image

In objective to extract a specific region of interest (ROI) from an image using slicing operations. Concept of cropping an image means selecting a rectangular portion from the original image based on pixel coordinates:

$$cropped = image[y_{start}, y_{end}, x_{start}, x\_end]$$

- y is represent vertical axis row from top to bottom
- x is represent horizontal axis column from left to right
- *image* loading from your image dataset

Given an image name `a.png` from folder name `image_test_1`, as shown in picture below:



- *The vertical coordinate* $y \in [0, 451]$
- *The horizontal coordinate* $x \in [0, 447]$

Question1: Based on the original image, crop the regions defined by the bounding boxes corresponding to each color green, blue, and orange.

Question2: Based on bounding boxes corresponding to each color green, blue, and orange, try to crop with another shape of bounding box such as circle.

3.Geometric Transformation

Geometric operation of digital image consists of two basic operations spatial (plan) for transformation coordinate and intensity interpolation for assigned intensity values to spatially transform pixels. In this part we introduce geometric transformation operation:

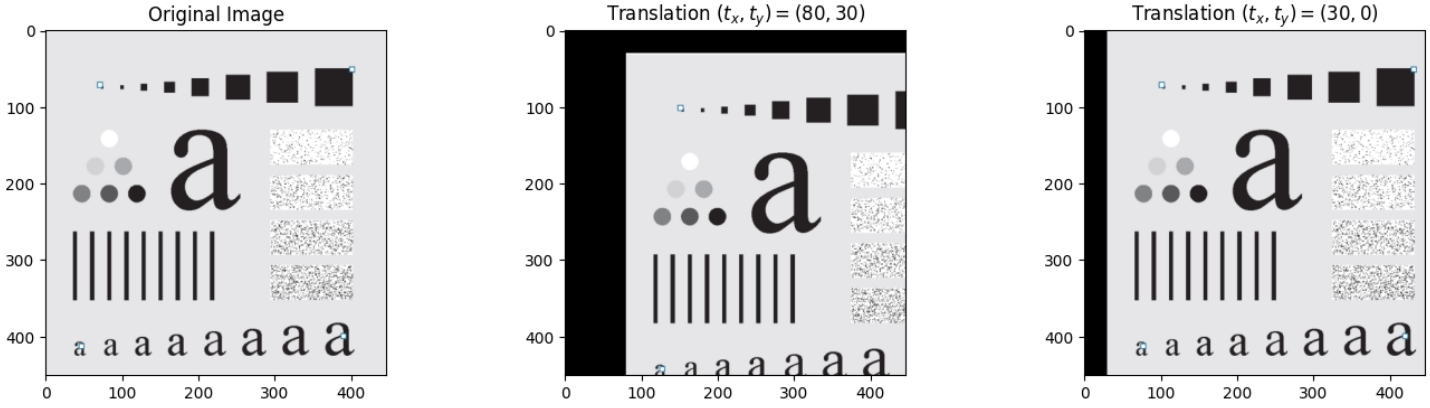
$$\begin{bmatrix} x' \\ y' \\ 1 \end{bmatrix} = A \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} = \begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}$$

For the image has provided name `a.png` from folder name `image_test_1`, we will work on many transformation methods according to the matrix A was given.

Question 1:

Given that a matrix $A = \begin{bmatrix} 1 & 0 & t_x \\ 0 & 1 & t_y \\ 0 & 0 & 1 \end{bmatrix}$ for image translation then the new coordinates are defined as: $\begin{cases} x' = x + t_x \\ y' = y + t_y \end{cases}$

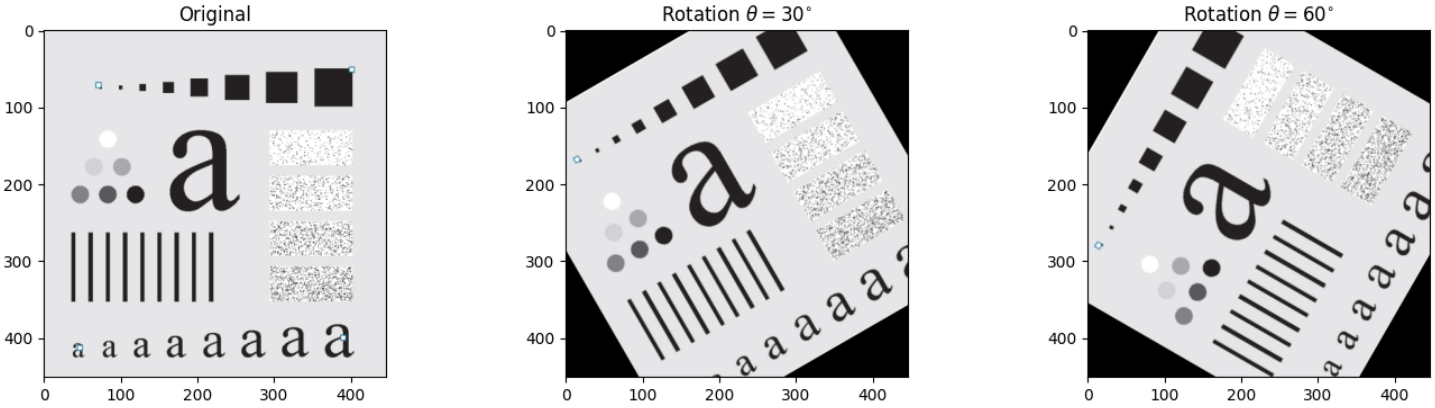
Testing the any values of t_x and t_y and then plot this image using matplotlib. For example, by replacing $(t_x, t_y) =$ (80,30) and (30,0).



Question 2:

Given that a matrix $A = \begin{bmatrix} \cos(\theta) & -\sin(\theta) & 0 \\ \sin(\theta) & \cos(\theta) & 0 \\ 0 & 0 & 1 \end{bmatrix}$ for image rotation then the new coordinates are defined as: $\begin{cases} x' = x\cos(\theta) - y\sin(\theta) \\ y' = x\sin(\theta) + y\cos(\theta) \end{cases}$

2.1. Testing the any values of $\theta = [30^0, 45^0, 60^0]$ and then plot this image using matplotlib as picture bellow:



2.2. Combine both technique rotation and translation at the same time by testing the parameter value

$(\theta, t_x, t_y) = [(30^0, 30, 60), (45^0, 0, 80)]$ and then using a subplot layout of 1×3 (1 row and 3 columns) of the result original image, and another two receive from the transformation.

Question 3:

Given that two matrix $A_v = \begin{bmatrix} 1 & s_v & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$ and $A_h = \begin{bmatrix} 1 & 0 & 0 \\ s_h & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$ represent a shear vertical and horizontal respectively. The new coordinates are defined as:

$$\begin{cases} x'_v = x + s_v y \\ y'_v = y \end{cases} \quad \text{and} \quad \begin{cases} x'_h = x \\ y'_h = s_h x + y \end{cases}$$

Testing the value of $s_v = 0.5$ and $s_h = 0.5$ then plot subplot layout 1×3 (1 row and 3 columns) as picture given: