

Image processing

```

import numpy as np
import cv2
from google.colab.patches import cv2_imshow

arr = np.arange(10)
print(arr)

[0  1  2  3  4  5  6  7  8  9]

arr_A = np.array([[3,8,2,4,6],
                  [1,5,9,3,7],
                  [6,2,7,4,9],
                  [4,3,1,6,8],
                  [9,7,5,2,1]])
arr_B = np.array([[5,2,6,1,3],
                  [9,4,7,2,8],
                  [2,6,3,9,1],
                  [7,5,8,3,6],
                  [1,3,9,4,2]])

#sum
arr_C = arr_A + arr_B
#multiply
arr_D = arr_A.dot(arr_B) # or (arr_A @ arr_B) or np.matmul(arr_A,arr_B)
print(arr_C)
print(arr_D)

[[ 8 10  8  5  9]
 [10  9 16  5 15]
 [ 8  8 10 13 10]
 [11  8  9  9 14]
 [10 10 14  6  3]]
[[125  88 166  73 111]
 [ 96 112 155 129  84]
 [ 99 109 184 121  83]
 [ 99  80 168  69  89]
 [133  89 143  78 102]]

# All 0s matrix
np.zeros((2,3))

array([[0., 0., 0.],
       [0., 0., 0.]])

# All 1s matrix
np.ones((4,2,2), dtype='int32')

array([[[1, 1],
        [1, 1]],

       [[1, 1],
        [1, 1]],

       [[1, 1],
        [1, 1]],

       [[1, 1],
        [1, 1]]], dtype=int32)

# Any other number
np.full((2,2), 99)

array([[99, 99],
       [99, 99]])

# Random decimal numbers
np.random.rand(4,2)

array([[0.30426152, 0.2355099 ],
       [0.48708633, 0.54312186],
       [0.75933678, 0.78488253],
       [0.12775178, 0.34734094]])

```

```

# Random Integer values
np.random.randint(-8,8, size=(3,3))

array([[ -6,  -1,   4],
       [  0,   0,   0],
       [  4,  -3,   1]])

# The identity matrix
np.identity(5)

array([[1.,  0.,  0.,  0.,  0.],
       [0.,  1.,  0.,  0.,  0.],
       [0.,  0.,  1.,  0.,  0.],
       [0.,  0.,  0.,  1.,  0.],
       [0.,  0.,  0.,  0.,  1.]])

# Repeat an array
arr = np.array([[1,2,3]])
repeat_arr = np.repeat(arr,3, axis=0)
print(repeat_arr)

[[1 2 3]
 [1 2 3]
 [1 2 3]]

#reshape matrix
before = np.array([[1,2,3,4],[5,6,7,8]])
print(before)

after = before.reshape(4,2)
print(after)

[[1 2 3 4]
 [5 6 7 8]]
[[1 2]
 [3 4]
 [5 6]
 [7 8]]

# Vertically stacking vectors
v1 = np.array([1,2,3,4])
v2 = np.array([5,6,7,8])

np.vstack([v1,v2])

array([[1, 2, 3, 4],
       [5, 6, 7, 8]])

# Horizontal stack
h1 = np.ones((2,4))
h2 = np.zeros((2,2))

np.hstack((h1,h2))

array([[1., 1., 1., 1., 0., 0.],
       [1., 1., 1., 1., 0., 0.]])

#version of opencv
cv2.__version__

'4.7.0'

#read and show image
img = cv2.imread('messi.jpg')
cv2.imshow(img)

```



img

```
array([[175, 83, 10],
       [175, 83, 10],
       [175, 83, 10],
       ...,
       [177, 86, 11],
       [177, 86, 11],
       [177, 86, 11]],

       [[179, 85, 12],
       [179, 85, 12],
       [179, 85, 12],
       ...,
       [176, 88, 12],
       [176, 88, 12],
       [176, 88, 12]],

       [[178, 84, 11],
       [178, 84, 11],
       [178, 84, 11],
       ...,
       [174, 88, 12],
       [174, 88, 12],
       [174, 88, 12]],

       ...,

       [[151, 210, 182],
       [151, 210, 182],
       [151, 210, 182],
       ...,
       [151, 210, 182],
       [151, 210, 182],
       [151, 210, 182]],

       [[151, 210, 182],
       [151, 210, 182],
       [151, 210, 182],
       ...,
       [151, 210, 182],
       [151, 210, 182],
       [151, 210, 182]]]
```

```
[[151, 210, 182],
 [151, 210, 182],
 [151, 210, 182],
 ...,
 [151, 210, 182],
 [151, 210, 182],
 [151, 210, 182]], dtype=uint8)
```

```
img.shape
```

```
(670, 740, 3)
```

```
#convert color img to gray img
'''
fomula to convert
def convert_rgb_to_gray(img):
    red   = 0.299 * img[:, :, 0]
    green = 0.587 * img[:, :, 1]
    blue  = 0.114 * img[:, :, 2]
    '''
gray_img = cv2.cvtColor(img, cv2.COLOR_BGR2GRAY)
cv2.imshow(gray_img)
```



```
gray_img.shape
```

```
(670, 740)
```

Write and Save the gray image

```
#go to the folder
%cd /content/drive/MyDrive/image

/content/drive/MyDrive/image
```

```
#write image
cv2.imwrite("gray_img.jpg", gray_img)
```

True

```
#Get (R,G,B) at random pixel (300,400)
(R,G,B) = img[300,400]
print(f"R = {R}, G = {G}, B = {B}")

R = 230, G = 239, B = 248
```

```
#Slice image
slice_img = img[100:350,400:600]
cv2_imshow(slice_img)
```



```
#Resize image
resized_img = cv2.resize(img,(300,300))
cv2_imshow(resized_img)
```

