

# Computer vision task 1

The image displays two screenshots of a JupyterLab environment running on a local host (localhost:8888). The top screenshot shows a notebook named 'cv' with three code cells. The first cell imports cv2 and numpy, reads an image 'k1.jpg', and displays it in a window titled 'anime'. The second cell reads the same image in 'IMREAD\_UNCHANGED' mode and displays it in a window titled 'anime1'. The third cell reads the image in 'IMREAD\_GRAYSCALE' mode and displays it in a window titled 'anime2'. The bottom screenshot shows a notebook named 'Untitled5' with three code cells. The first cell reads an image 'VIRAT.jpeg' and displays it in a window titled 'virat'. The second cell reads the same image in 'IMREAD\_UNCHANGED' mode and displays it in a window titled 'virat1'. The third cell reads the image in 'IMREAD\_GRAYSCALE' mode and displays it in a window titled 'virat2'. The image outputs are visible as windows on the desktop, showing a portrait of a man in a blue shirt and cap.

**Top Screenshot: JupyterLab 'cv' notebook**

```
[12]: #reading the image
import cv2
import numpy as np

#read the image
image=cv2.imread(r"C:\Users\Admin\OneDrive\Desktop\task\Deep Learning\images\k1.jpg")

cv2.imshow('anime',image)# display the image
cv2.waitKey(0)# it delays the closing of output window till any key is pressed
cv2.destroyAllWindows()#close all windows in one short using a single instance

[18]: #UNCHANGED
image=cv2.imread(r"C:\Users\Admin\OneDrive\Desktop\task\Deep Learning\images\k1.jpg",cv2.IMREAD_UNCHANGED)
cv2.imshow('anime1',image)
cv2.waitKey(0)
cv2.destroyAllWindows()

[19]: #gray scale
image=cv2.imread(r"C:\Users\Admin\OneDrive\Desktop\task\Deep Learning\images\k1.jpg",cv2.IMREAD_GRAYSCALE)
cv2.imshow('anime2',image)
cv2.waitKey(0)
cv2.destroyAllWindows()

[21]: #getting dimension of the image
```

**Bottom Screenshot: JupyterLab 'Untitled5' notebook**

```
[*]: #Reading the image
import cv2
import numpy as np

#read the image
image=cv2.imread(r"C:\Users\ANGELIN\Downloads\VIRAT.jpeg")

cv2.imshow('virat',image)
cv2.waitKey(0)
cv2.destroyAllWindows()

[ ]: #UNCHANGED
image=cv2.imread(r"C:\Users\ANGELIN\Downloads\VIRAT.jpeg",cv2.IMREAD_UNCHANGED)
cv2.imshow('virat1',image)
cv2.waitKey(0)
cv2.destroyAllWindows()

[ ]: #GRAY SCALE
image=cv2.imread(r"C:\Users\ANGELIN\Downloads\VIRAT.jpeg",cv2.IMREAD_GRAYSCALE)
cv2.imshow('virat2',image)
cv2.waitKey(0)
cv2.destroyAllWindows()
```

# Computer vision task 1

The image displays two sequential screenshots of a JupyterLab environment running Python code for computer vision tasks. The interface includes a browser window at the top showing the URL `localhost:8888/notebooks/Downloads/Untitled5.ipynb` and a JupyterLab header with a 'Trusted' status.

**Top Screenshot:** The code cell shows three steps:

- #UNCHANGED:** Reads the image `VIRAT.jpeg` and displays it in a window titled `virat3`.
- #GRAY\_SCALE:** Converts the image to grayscale and displays it in a window titled `virat2`.
- #GETTING DIMENSION OF THE IMAGE:** Prints the image dimensions, showing `(231, 218, 3)`.

**Bottom Screenshot:** The code cell continues with:

- #EDGE DETECTION:** Applies the Canny edge detection algorithm to the grayscale image and displays the result in a window titled `edge detection`.

The Windows taskbar at the bottom shows the system time as 07:13 on 11-07-2024.

# Computer vision task 1

The image displays two screenshots of a JupyterLab interface, showing the execution of computer vision tasks using OpenCV (cv2) in Python.

**Top Screenshot:** The JupyterLab interface shows a code cell with the following Python code:

```
image=cv2.imread(r"C:\Users\ANGELIN\Downloads\VIRAT.jpeg",cv2.IMREAD_UNCHANGED)
cv2.imshow('virat3',image)
dim=image.shape
print(dim)
cv2.waitKey(0)
cv2.destroyAllWindows()

(231, 218, 3)

[*]: #EDGE DETECTION
image=cv2.imread(r"C:\Users\ANGELIN\Downloads\VIRAT.jpeg",cv2.IMREAD_UNCHANGED)
edge=cv2.Canny(image,100,120)
cv2.imshow('edge detection',edge)
cv2.waitKey(0)
cv2.destroyAllWindows()
```

The output of the code is a small window titled "edge d..." displaying the edge detection result of the input image, showing the facial features in white on a black background.

**Bottom Screenshot:** The JupyterLab interface shows a code cell with the following Python code:

```
(231, 218, 3)

[11]: #EDGE DETECTION
image=cv2.imread(r"C:\Users\ANGELIN\Downloads\VIRAT.jpeg",cv2.IMREAD_UNCHANGED)
edge=cv2.Canny(image,100,120)
cv2.imshow('edge detection',edge)
cv2.waitKey(0)
cv2.destroyAllWindows()

[17]: #VERTICAL CONCATINATION
img1=cv2.imread(r"C:\Users\ANGELIN\Downloads\LEBIN coat add.jpg")
img2=cv2.imread(r"C:\Users\ANGELIN\Downloads\LEBIN coat add.jpg")
new=cv2.vconcat([img1,img2])
cv2.imwrite(r"C:\Users\ANGELIN\Downloads\LEBIN coat add vconcat.jpg",new)
cv2.waitKey(0)
cv2.destroyAllWindows()
```

The output of the code is a small window titled "LEBIN coat ad..." displaying the result of vertical concatenation of two images, showing two identical images of a man in a suit and red tie stacked vertically.

# Computer vision task 1

Search results - lebinbright@gr x Home x Untitled5 x +

localhost:8888/notebooks/Downloads/Untitled5.ipynb

Jupyter Untitled5 Last Checkpoint: 55 minutes ago

File Edit View Run Kernel Settings Help

cv2.destroyAllWindows()

```
[17]: #VERTICAL CONCATINATION
img1=cv2.imread(r"C:\Users\ANGELIN\Downloads\LEBIN coat add.jpg")
img2=cv2.imread(r"C:\Users\ANGELIN\Downloads\LEBIN coat add.jpg")

new=cv2.vconcat([img1,img2])
cv2.imwrite(r"C:\Users\ANGELIN\Downloads\LEBIN coat add vconcat.jpg",new)

cv2.waitKey(0)
cv2.destroyAllWindows()
```


```
[19]: #HORIZONTAL CONCATINATION
img1=cv2.imread(r"C:\Users\ANGELIN\Downloads\LEBIN coat add.jpg")
img2=cv2.imread(r"C:\Users\ANGELIN\Downloads\LEBIN coat add.jpg")

new=cv2.hconcat([img1,img2])
cv2.imwrite(r"C:\Users\ANGELIN\Downloads\LEBIN coat add hconcat.jpg",new)

cv2.waitKey(0)
cv2.destroyAllWindows()
```

[ ]:

LEBIN coat ad...



110%

SOFT COPY - lebinbright@gr x Home x Untitled5 x +

localhost:8888/notebooks/Downloads/Untitled5.ipynb

Jupyter Untitled5 Last Checkpoint: 1 hour ago

File Edit View Run Kernel Settings Help

```
[19]: #HORIZONTAL CONCATINATION
img1=cv2.imread(r"C:\Users\ANGELIN\Downloads\LEBIN coat add.jpg")
img2=cv2.imread(r"C:\Users\ANGELIN\Downloads\LEBIN coat add.jpg")

new=cv2.hconcat([img1,img2])
cv2.imwrite(r"C:\Users\ANGELIN\Downloads\LEBIN coat add hconcat.jpg",new)


cv2.waitKey(0)
cv2.destroyAllWindows()
```

```
[24]: #TILE
img1=cv2.imread(r"C:\Users\ANGELIN\Downloads\LEBIN coat add (1).jpg")
new=np.tile(img1,(5,4,1))
cv2.imwrite(r"C:\Users\ANGELIN\Downloads\LEBIN coat add (1) tile.jpg",new)

cv2.waitKey(0)
cv2.destroyAllWindows()
```

[ ]:

LEBIN coat ad...



23%

# Computer vision task 1

The image displays two screenshots of a JupyterLab interface, showing the execution of computer vision tasks using OpenCV and NumPy.

**Top Screenshot:**

- Code:**

```
cv2.waitKey(0)
cv2.destroyAllWindows()

[24]: #TILE
img1=cv2.imread(r"C:\Users\ANGELIN\Downloads\LEBIN coat add (1).jpg")
new=np.tile(img1,(5,4,1))
cv2.imwrite(r"C:\Users\ANGELIN\Downloads\LEBIN coat add (1) tile.jpg",new)

cv2.waitKey(0)
cv2.destroyAllWindows()

[*]: #FLIPPING AN IMAGE
#HORIZONTAL FLIP
img=cv2.imread(r"C:\Users\ANGELIN\Downloads\LEBIN coat add (1).jpg")
new=cv2.flip(image,0)
cv2.imshow('h flip',new)
cv2.waitKey(0)
cv2.destroyAllWindows()
```
- Output:** A window titled "h flip" displays the original image of a cricket player (Virat Kohli) flipped horizontally.

**Bottom Screenshot:**

- Code:**

```
cv2.waitKey(0)
cv2.destroyAllWindows()

[25]: #FLIPPING AN IMAGE
#HORIZONTAL FLIP
img=cv2.imread(r"C:\Users\ANGELIN\Downloads\LEBIN coat add (1).jpg")
new=cv2.flip(image,0)
cv2.imshow('h flip',new)
cv2.waitKey(0)
cv2.destroyAllWindows()

[*]: #FLIPPING AN IMAGE
#VERTICAL FLIP
img=cv2.imread(r"C:\Users\ANGELIN\Downloads\LEBIN coat add (1).jpg")
new=cv2.flip(image,1)
cv2.imshow('V flip',new)
cv2.waitKey(0)
cv2.destroyAllWindows()
```
- Output:** A window titled "V flip" displays the original image of a cricket player (Virat Kohli) flipped vertically.