

OpenCV Version control

Project

- Bilateral\_filter.py
- cv.py
- Edge\_detection.py
- flipping\_image.py
- Gaussain\_blur.py
- morplogical\_op.py
- new\_car.jpg
- Output.mp4
- Reading\_image.py
- Reading\_video.py
- resize\_img.py
- resizing\_basedOn\_ratio.py
- rotate\_image.py
- shape\_and\_text.py
- Shifting\_image.py
- Writing\_image.py
- Writing\_video.py

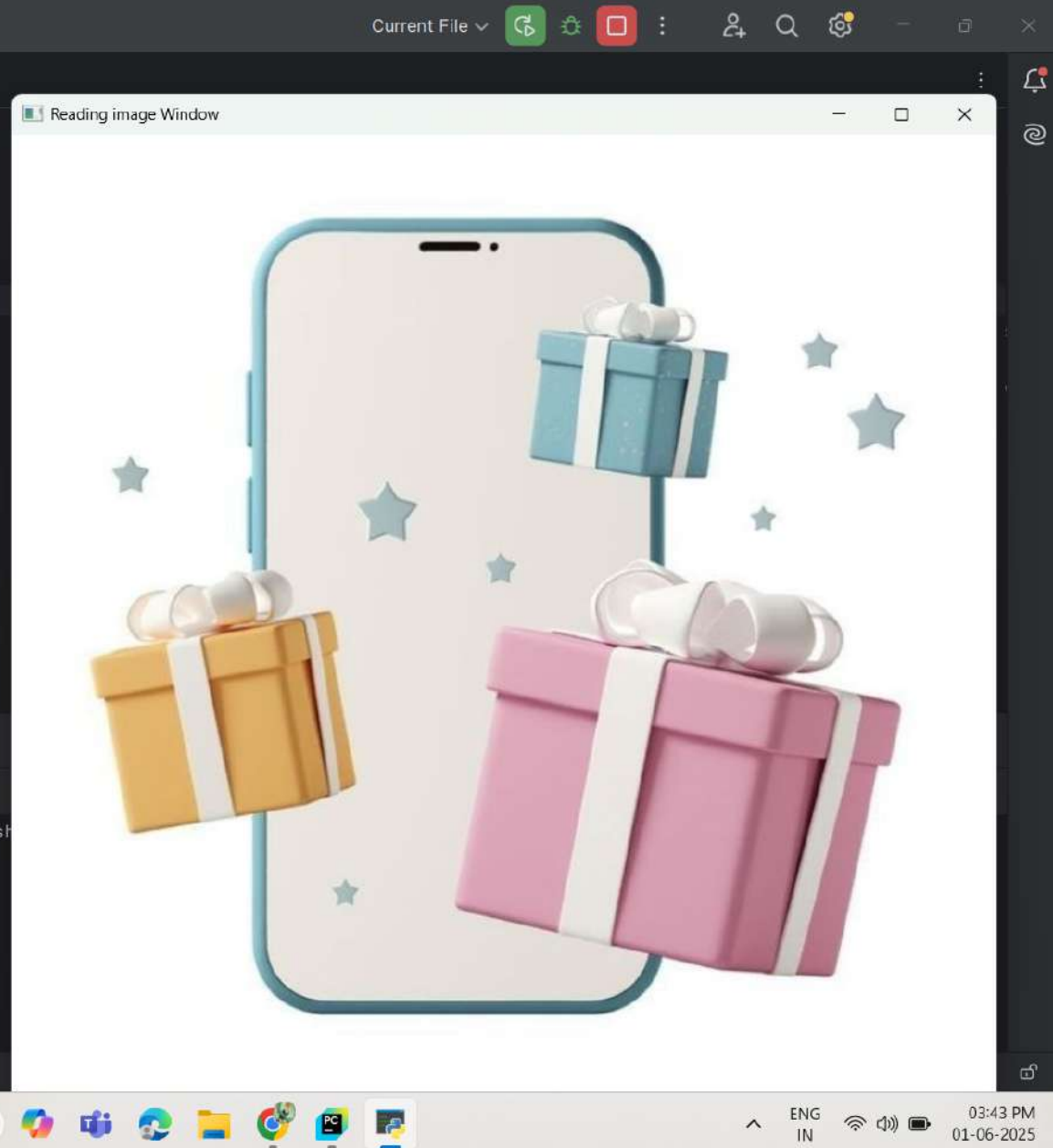
External Libraries

Python 3.12 (OpenCV)

Run Reading\_image

```
1 import cv2
2
3 img_path="C:/Users/BIKASH/OneDrive/Pictures/"
4 img_name="mobile.jpeg"
5 img = cv2.imread(img_path+img_name,1)
6
7 cv2.imshow( winname: 'Reading image Window',img)
8 cv2.waitKey(0)
9 cv2.destroyAllWindows()
```

"D:\Bikash learning\TuteDude\_Assignment\OpenCV\.venv\Scripts\python.exe" "D:\Bikash learning\TuteDude\_Assignment\OpenCV\Scripts\python.exe" "D:\Bikash learning\TuteDude\_Assignment\OpenCV\Scripts\python.exe"



OpenCV Version control

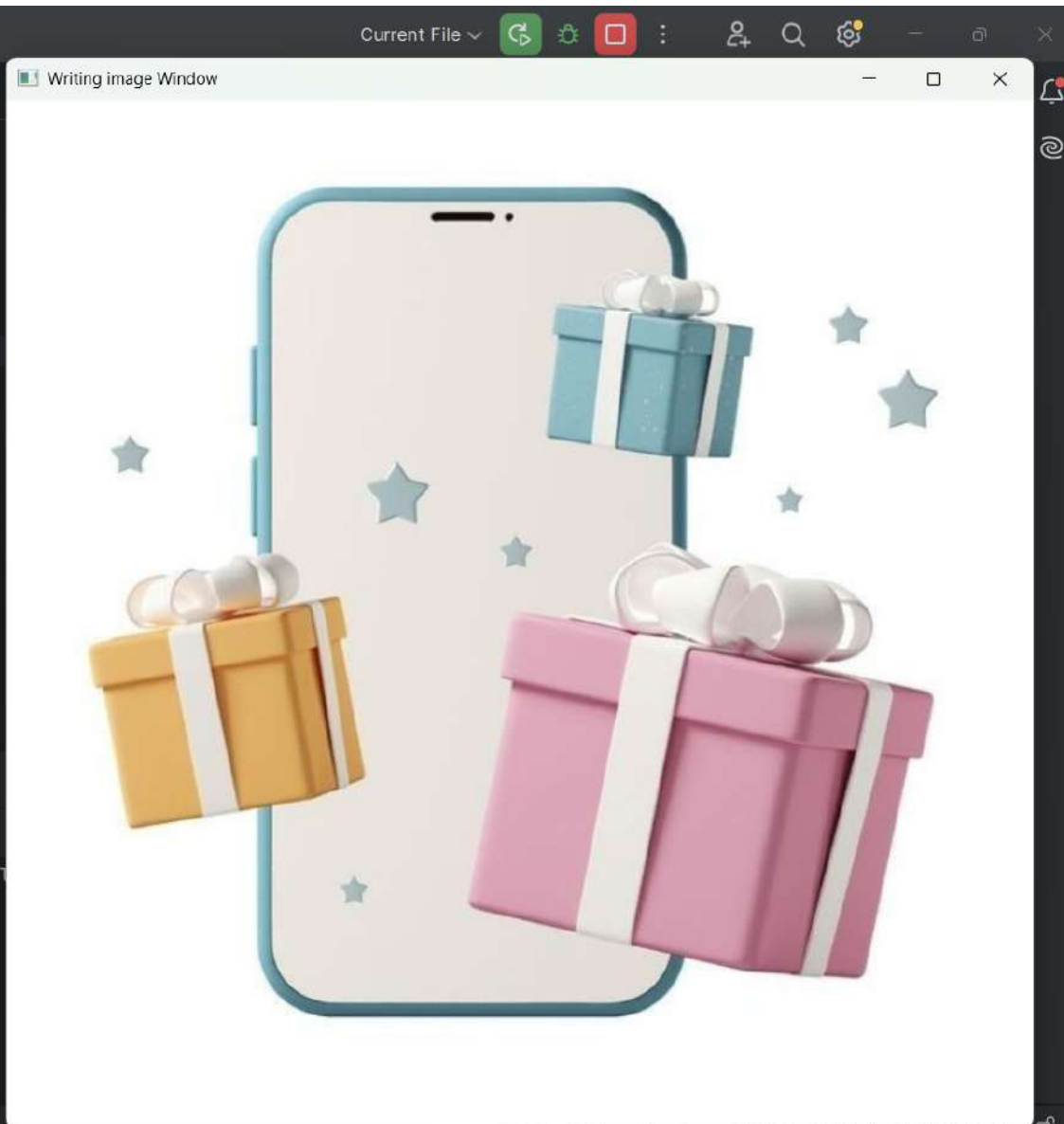
Project Writing\_image.py

```
1 import cv2
2
3 img_path="C:/Users/BIKASH/OneDrive/Pictures/"
4 img_name="mobile.jpeg"
5 img = cv2.imread(img_path+img_name,1)
6
7 cv2.imshow( winname: 'Writing image Window' ,img)
8 cv2.imwrite(img_path + 'smart_logo.jpeg' ,img)
9
10 cv2.waitKey(0)
11 cv2.destroyAllWindows()
```

Run Writing\_image

"D:\Bikash learning\TuteDude\_Assignment\OpenCV\.venv\Scripts\python.exe" "D:\Bikash T

OpenCV Writing\_Image.py



OpenCV Version control

Project

Bilateral\_filter.py

cv.py

Edge\_detection.py

flipping\_image.py

Gaussian\_blur.py

morphological\_op.py

new\_car.jpg

Output.mp4

Reading\_image.py

Reading\_video.py

resize\_img.py

resizing\_basedOn\_ratio.py

rotate\_image.py

shape\_and\_text.py

Shifting\_image.py

Writing\_image.py

Writing\_video.py

External Libraries

Python 3.12 (OpenCV)

Run

resize\_img

"D:\Bikash Learning\TuteDude\_Assignment\OpenCV\.venv\Scripts\python.exe" "D:\Bikash Learning\TuteDude\_Assignment\OpenCV\resize\_img.py"

resize\_img.py

```
1 import cv2
2
3 img_path="C:/Users/BIKASH/OneDrive/Pictures/"
4 img_name="car.jpg"
5 img = cv2.imread(img_path+img_name,0)
6
7 width = 800
8 height = 500
9 dim = (width,height)
10 img_resize = cv2.resize(img,dim)
11 cv2.imshow( winname: 'resize window',img_resize)
12 cv2.waitKey(0)
13 cv2.destroyAllWindows()
14
15
16
17
18
```

resize window

13:24 CRLF UTF-8 4 spaces Python 3.12 (OpenCV)

36°C Haze

Search

ENG IN 03:46 PM 01-06-2025



OpenCV Version control

resize\_img.py morpological\_op.py

```
1 img_path="C:/Users/BIKASH/OneDrive/Pictures
2 img_name="car.jpg"
3 img = cv2.imread(img_path+img_name,0)
4
5
6 width = 300
7 height = 200
8 dim = (width,height)
9 img_resize = cv2.resize(img,dim)
10
11
12 kernel = np.ones((5,5),dtype = 'uint8')
13 erosion = cv2.erode(img_resize,kernal,iterations=1)
14 dilation = cv2.dilate(img_resize,kernal,iterations=1)
15
16
17 opening = cv2.morphologyEx(img_resize,cv2.MORPH_OPEN,kernal)
18 closing = cv2.morphologyEx(img_resize,cv2.MORPH_CLOSE,kernal)
19 gradient = cv2.morphologyEx(img_resize,cv2.MORPH_GRADIENT,kernal)
20 top_hat = cv2.morphologyEx(img_resize,cv2.MORPH_TOPHAT,kernal)
21 black_hat = cv2.morphologyEx(img_resize,cv2.MORPH_TOPHAT,kernal)
22
23
24
25
26 cv2.imshow( winname: 'original window',img_resize)
27 cv2.imshow( winname: 'Erosion window',erosion)
28 cv2.imshow( winname: 'Dilation window',dilation)
29
30 cv2.imshow( winname: 'Opening window',opening)
31 cv2.imshow( winname: 'closing window',closing)
32 cv2.imshow( winname: 'gradient window',gradient)
33 cv2.imshow( winname: 'top_hat window',top_hat)
34 cv2.imshow( winname: 'black_hat window',black_hat)
35
36 cv2.waitKey(0)
```

original window

Erosion window

closing window

Dilation window

gradient window

black\_hat window

Opening window

top\_hat window

Run morpological\_op.py

OpenCV morpological\_op.py

16:53 CRLF UTF-8 4 spaces Python 3.12 (OpenCV)

36°C Haze

Search

ENG IN

03:52 PM 01-06-2025

OpenCV Version control

flipping\_image.py

flipping\_image.py

1

2

3

4

5

6

7

8

9

10

11

12

13

14

15

16

17

18

19

20

21

22

23

24

25

26

27

28

```
import cv2

img_path="C:/Users/BIKASH/OneDrive/Pictures/"
img_name="car.jpg"
img = cv2.imread(img_path+img_name,0)

width = 400
height = 250
dim = (width,height)

img_resize = cv2.resize(img,dim)
print('Original size of image: ',img.size)
print('Size of image after resize: ',img_resize.size)

flip_hor = cv2.flip(img_resize, flipCode: 1) # horizontal flip code
flip_ver = cv2.flip(img_resize, flipCode: 0) # vertical flip code
flip_hor_ver = cv2.flip(img_resize,-1) # horizontal flip code

cv2.imshow( winname: 'Original Window',img_resize)
cv2.imshow( winname: 'Horizontal flip Window',flip_hor)
cv2.imshow( winname: 'vertical flip window',flip_ver)
cv2.imshow( winname: 'Horizontal & vertical flip window',flip_hor_ver)

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

Original Window

vertical flip window

Horizontal & vertical flip window

Horizontal flip Window

Run flipping\_image

OpenCV > flipping\_image.py

9:11 CRLF UTF-8 4 spaces Python 3.12 (OpenCV)

36°C Haze

Search

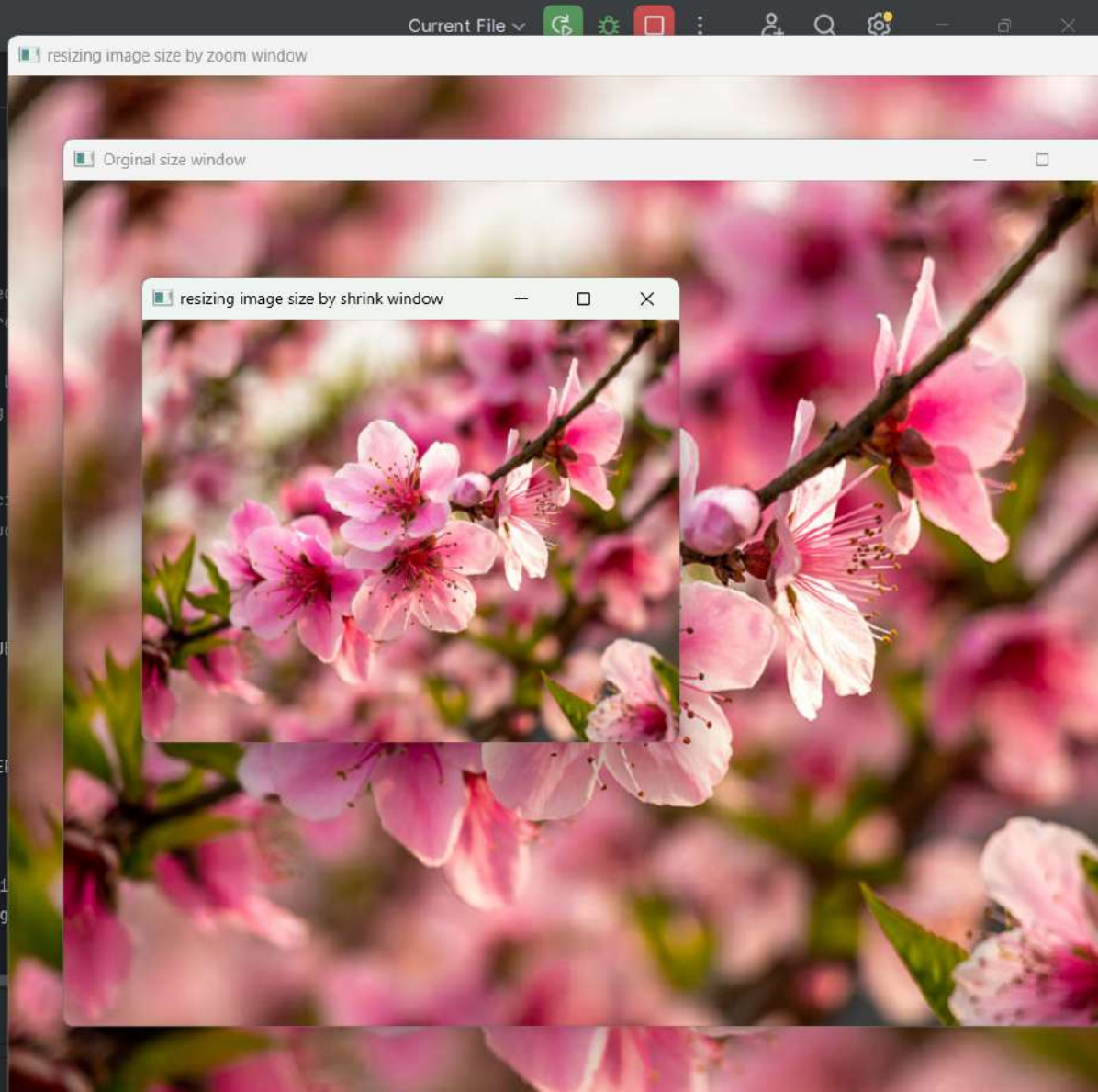
ENG IN 03:55 PM 01-06-2025



```
OpenCV Version control
flipping_image.py resizing_basedOn_ratio.py
4 img_path="C:/Users/BIKASH/OneDrive/Pictures/"
5 img_name="peach_flower.jpg"
6 img = cv2.imread(img_path+img_name,1)
7
8 print('original size of the image: ',img.shape)
9
10 scale_up = 150 # if the scale 100 then its original size, if we re
11 scale_down = 50 # if the scale 100 then its original size, if we re
12
13 width_up = int(img.shape[1] * scale_up / 100) # increase/reducing
14 height_up = int(img.shape[0] * scale_up / 100) # increase/reducing
15 dim_up = (width_up,height_up)
16
17 width_down = int(img.shape[1] * scale_down / 100) # increase/reduc
18 height_down = int(img.shape[0] * scale_down / 100) # increase/reduc
19 dim_down = (width_down,height_down)
20
21 # for scale Up use either:-INTER_CUBIC , INTER_LINEAR
22 resizing_img_up = cv2.resize(img,dim_up,interpolation=cv2.INTER_CUBIC)
23 print('size of the image resizing : ',resizing_img_up.shape)
24
25 # for scale Down use either :-INTER_AREA , INTER_NEAREST
26 resizing_img_down = cv2.resize(img,dim_down,interpolation=cv2.INTER_AREA)
27 print('size of the image resizing : ',resizing_img_down.shape)
28
29 cv2.imshow( winname: 'Original size window',img)
30 cv2.imshow( winname: 'resizing image size by zoom window',resizing_img_up)
31 cv2.imshow( winname: 'resizing image size by shrink window',resizing_img_down)
32
33 cv2.waitKey(0)
```

Run resizing\_basedOn\_ratio

OpenCV > resizing\_basedOn\_ratio.py



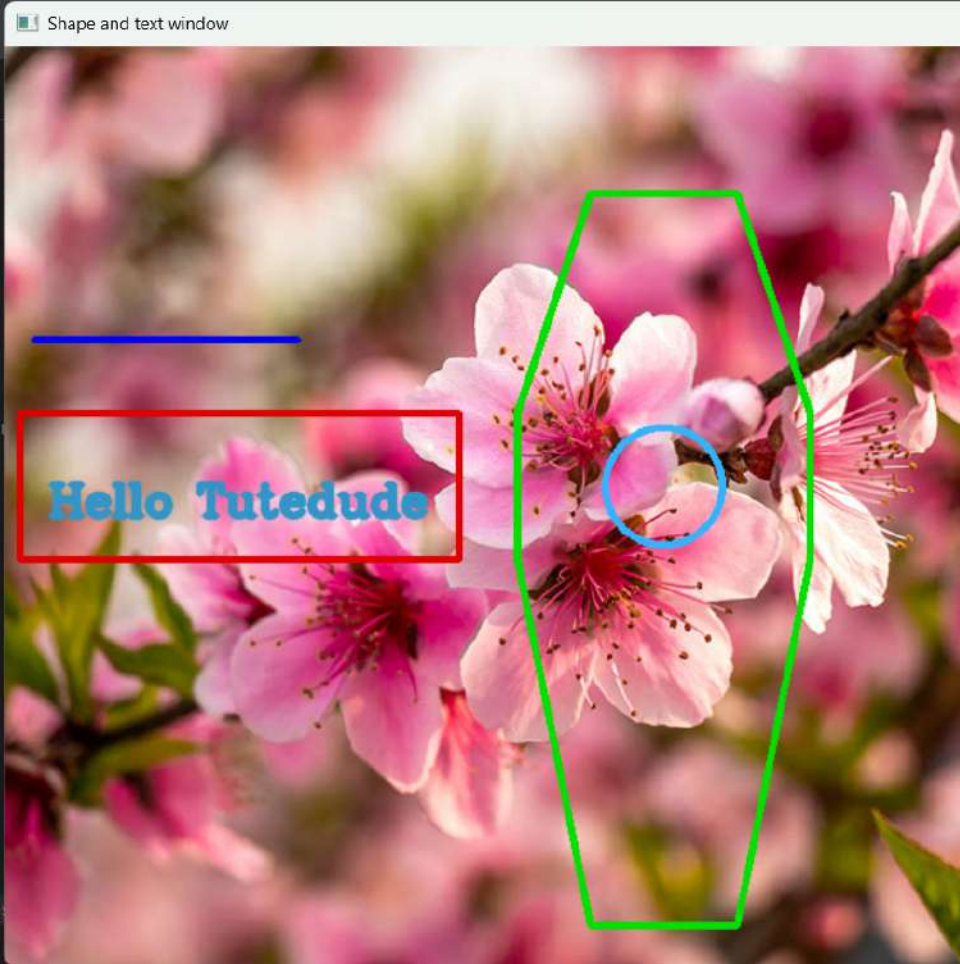
OpenCV Version control

shape\_and\_text.py

```
1 from pickletools import uint8
2
3 import cv2
4 import numpy as np
5
6 img_path="C:/Users/BIKASH/OneDrive/Pictures/"
7 img_name="peach_flower.jpg"
8 img = cv2.imread(img_path+img_name,1)
9
10 #img = np.zeros(shape=[650,600,3],dtype='uint8') # This line of code helps us to make the wi
11 #img.fill(255) # this helps to change the background colour
12
13
14 cv2.line(img, pt1: (20,200), pt2: (200,200), color: (255,0,0), thickness: 4) # it draw a line
15 cv2.rectangle(img, (10,250), (310,350), (0,0,225),3)
16 cv2.circle(img, center: (450,300), radius: 40, color: (255,180,50), thickness: 3)
17
18 polygon_point = np.array(
19     object [[400,100],[500,100],[550,250],[550,350],[500,600],[400,600],[350,350],[350,250]]
20     ,np.int32)
21 cv2.polylines(img, pts: [polygon_point], isClosed: True, color: (0,225,0), thickness: 3)
22
23
24 font = cv2.FONT_HERSHEY_COMPLEX
25 cv2.putText(img, text: "Hello Tutedude", org: (30,320), font, fontScale: 1, color: (200,150,50), thicknes
26
27 cv2.imshow( winname: 'Shape and text window', img)
28
29 cv2.waitKey(0)
30 cv2.destroyAllWindows()
```

Run shape\_and\_text

Shape and text window



The image displays a Windows desktop environment. On the left, a code editor window titled 'shape\_and\_text.py' contains Python code using OpenCV and NumPy. The code reads an image of pink flowers, sets a white background, and draws several shapes: a blue horizontal line, a red rectangle containing the text 'Hello Tutedude', a blue circle, and a green closed polygon. The code also includes comments and imports for OpenCV and NumPy. At the bottom of the code editor, there is a 'Run' button and a tab for 'shape\_and\_text'. On the right, a window titled 'Shape and text window' displays the result of the script. It shows the same pink flower image with the drawn shapes and text overlaid. The text 'Hello Tutedude' is in blue, and the background is white. The shapes are drawn in their respective colors: blue line, red rectangle, blue circle, and green polygon.

34°C Haze

Search

25:64 CRLF UTF-8 4 spaces Python 3.12 (OpenCV)

ENG IN 04:22 PM 01-06-2025



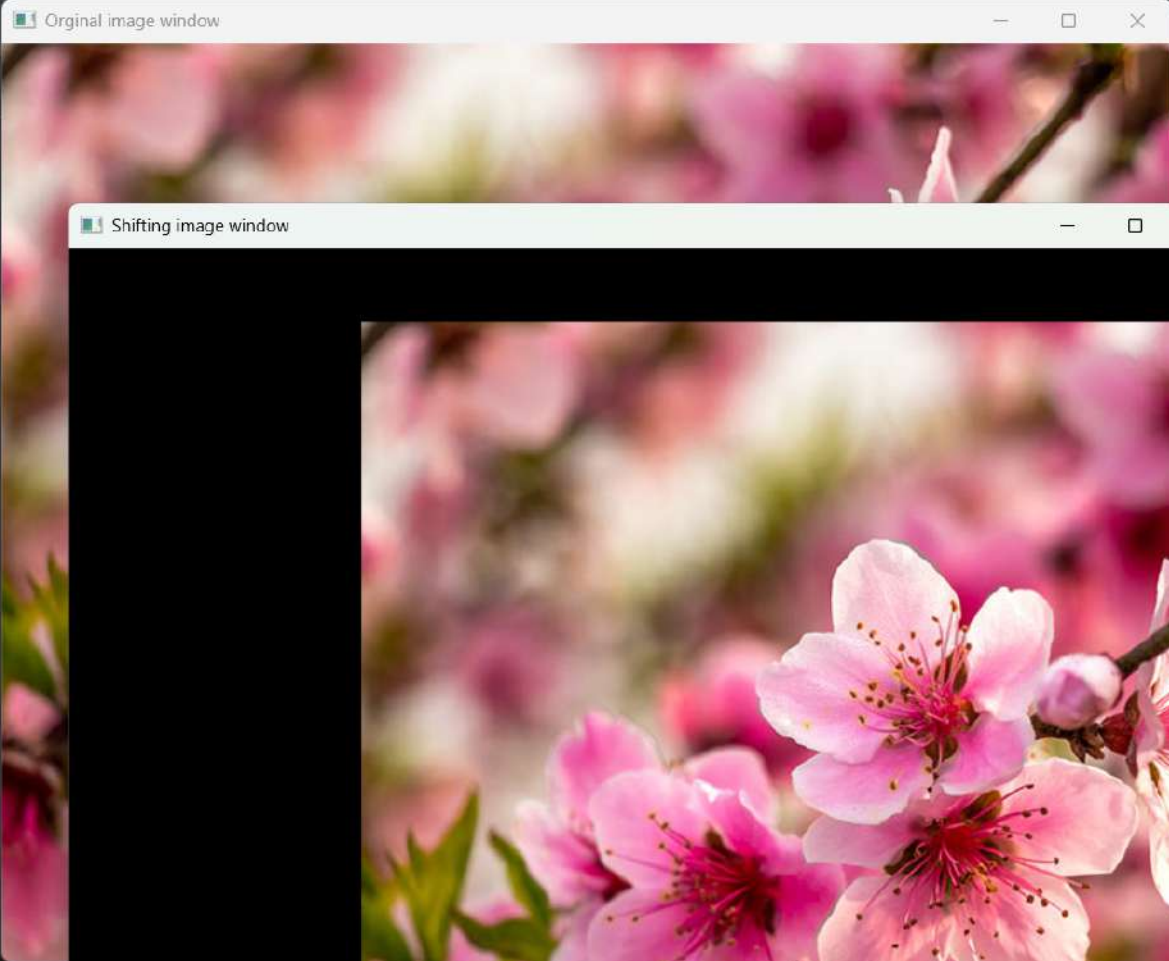
OpenCV Version control

Shifting\_image.py

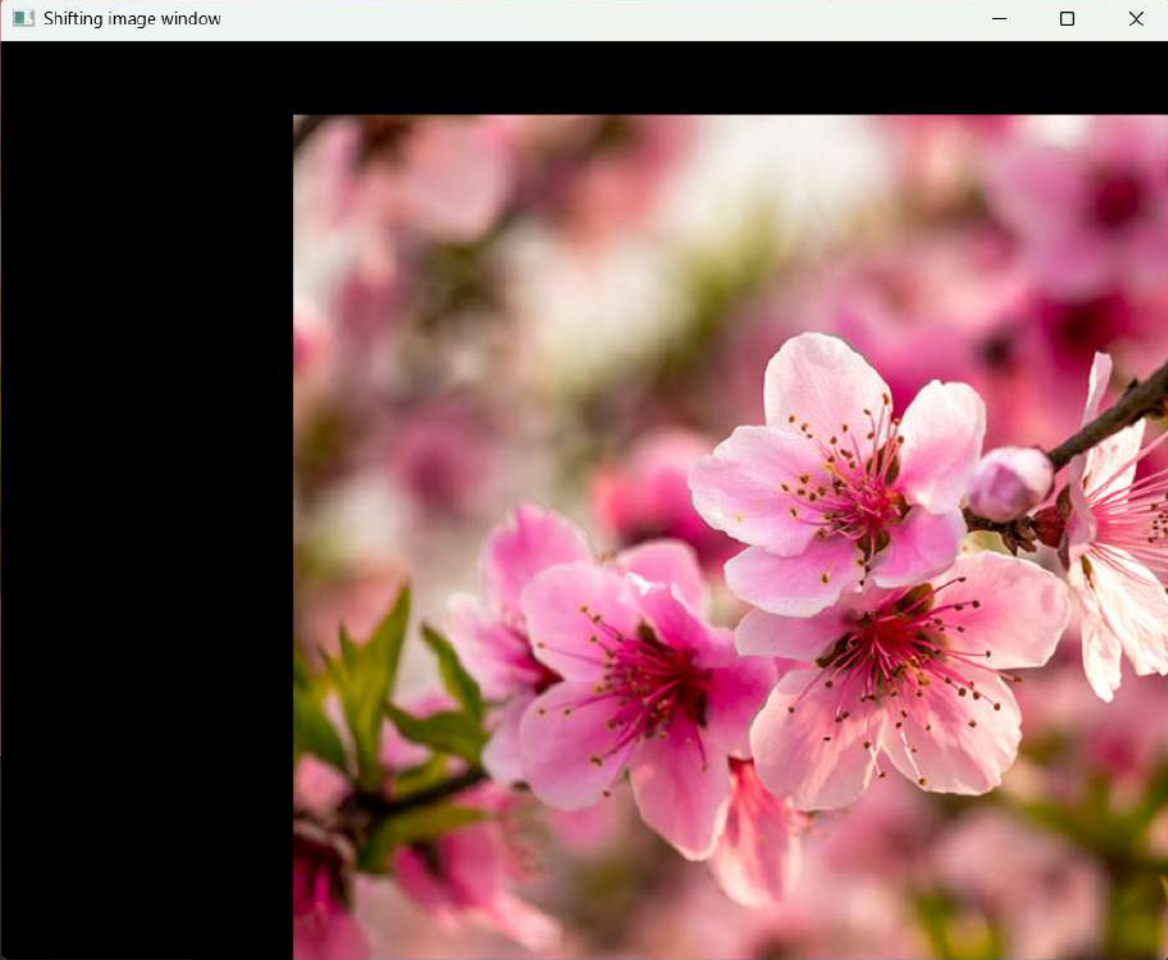
```
1 > import ...
3
4 img_path="C:/Users/BIKASH/OneDrive/Pictures/"
5 img_name="peach_flower.jpg"
6 img = cv2.imread(img_path+img_name,1)
7
8 column = img.shape[1] # it is a width
9 row = img.shape[0] # it is a height
10
11
12 shift_to_right = (1,0,200)
13 shift_to_down = (0,1,50)
14
15 shift = np.float32([shift_to_right,shift_to_down])
16
17 shifted = cv2.warpAffine(img,shift, dsize: (column,row))
18
19 cv2.imshow( winname: 'Original image window',img)
20 cv2.imshow( winname: 'Shifting image window',shifted)
21
22 cv2.waitKey(0)
23 cv2.destroyAllWindows()
24
```

Run Shifting\_image

Original image window




Shifting image window



34°C Haze

Search



ENG IN

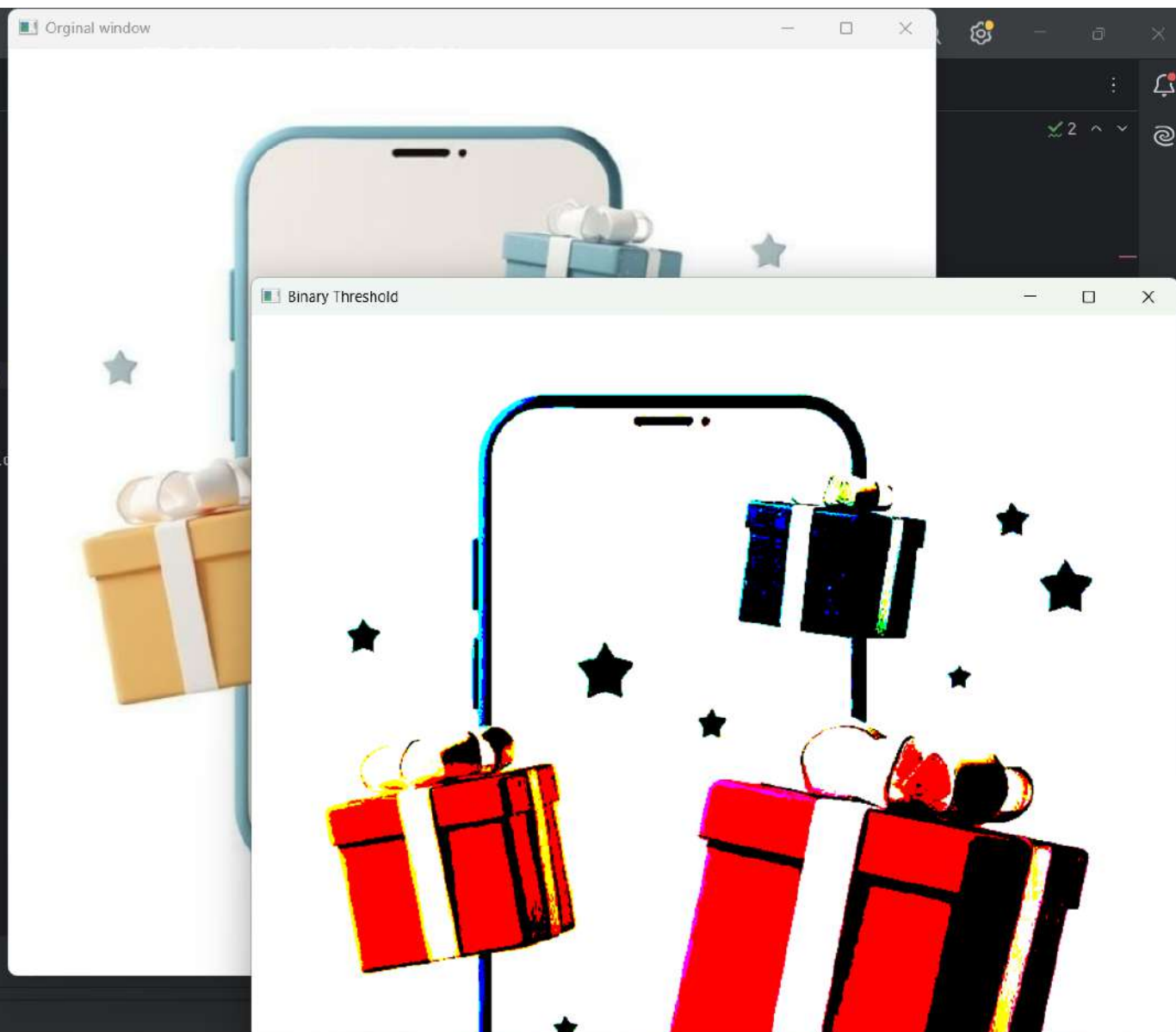
04:24 PM 01-06-2025



```
OpenCV Version control
rotate_image.py
1 import cv2
2
3 img_path="C:/Users/BIKASH/OneDrive/Pictures/"
4 img_name="mobile.jpeg"
5 img = cv2.imread(img_path+img_name,1)
6
7
8
9 column = img.shape[1] # it is a width
10 row = img.shape[0] # it is a height
11
12 center = (column/2 , row/2) # to get the center of image
13 angle = 90
14
15 # if the scale is 1 it keep the original size,
16 # if the scale is less than '1' it 'shrink the image'
17 # if greater than 1 it zoom the image
18 r = cv2.getRotationMatrix2D(center,angle, scale= 1)
19
20 rotate = cv2.warpAffine(img,r, dsize=(column,row))
21
22 cv2.imshow( winname: 'Original Image Window',img)
23 cv2.imshow( winname: 'Rotating image : ',rotate)
24 cv2.waitKey(0)
25 cv2.destroyAllWindows()
26
27
28
29
30
Run rotate_image
```



```
OpenCV Version control
Gaussain_blur.py Threshold.py
1 import cv2
2
3 img_path="C:/Users/BIKASH/OneDrive/Pictures/"
4 img_name="mobile.jpeg"
5 img = cv2.imread(img_path+img_name,1)
6
7 threshold_value = 200
8
9 _,binary_threshold = cv2.threshold(
10 img,threshold_value, maxval: 255,cv2.THRESH_BINARY)
11
12 cv2.imshow( winname: 'Original window',img)
13 cv2.imshow( winname: 'Binary Threshold',binary_threshold)
14
15 cv2.waitKey(0)
16 cv2.destroyAllWindows()
```





```
OpenCV Version control
Gaussain_blur.py
1 import cv2
2
3 img_path="C:/Users/BIKASH/OneDrive/Pictures/"
4 img_name="mobile.jpeg"
5 img = cv2.imread(img_path+img_name,1)
6
7 # kernel size should always be (+ ve and odd)
8 width = 7
9 height = 5
10 kernal_size = (width,height)
11
12 sigmaX = 0
13 sigmaY = 0
14
15 blur = cv2.GaussianBlur(img,kernal_size,sigmaX)
16
17 cv2.imshow( winname: 'Orginal image window ',img)
18 cv2.imshow( winname: 'Blur image window ',blur)
19
20
21 cv2.waitKey(0)
22 cv2.destroyAllWindows()
```



OpenCV > Gaussain\_blur.py

OpenCV Version control

median\_blur.py

```
1 import cv2
2
3 img_path="C:/Users/BIKASH/OneDrive/Pictures/"
4 img_name="eagle.jpeg"
5 img = cv2.imread(img_path+img_name,1)
6
7 resize = cv2.resize(img, dsize: (550,550))
8 kernal_size = 3
9
10 blur = cv2.medianBlur(resize,kernal_size)
11
12 cv2.imshow( winname: 'Original Window',img)
13 cv2.imshow( winname: 'Median Blur',blur)
14
15 cv2.waitKey(0)
16 cv2.destroyAllWindows()
```

Original Window

Median Blur

Run median\_

OpenCV > median\_

34°C Haze

Search

ENG IN

04:49 PM 01-06-2025




OpenCV Version control

Bilateral\_filter.py


```
1 import cv2
2
3 ## parameters of bilateral filter
4 # src---->input
5 # d ---->Diameter of a pixel
6 # sigma color
7 # sigma space
8
9 img_path="C:/Users/BIKASH/OneDrive/Pictures/"
10 img_name="eagle.jpeg"
11 img = cv2.imread(img_path+img_name,1)
12
13 resize = cv2.resize(img, dsize: (750,750)) # src
14 d = 5 # diameter
15 sigmaColor = 250
16 sigmaSpace = 200
17
18 b_filter = cv2.bilateralFilter(img,d,sigmaColor,sigmaSpace)
19
20 cv2.imshow( winname: 'input image Window',resize)
21 cv2.imshow( winname: 'Output image Window',b_filter)
22
23 cv2.waitKey(0)
24 cv2.destroyAllWindows()
```

Run Bilateral\_filter

input image Window



Output image Window



34°C Haze

Search

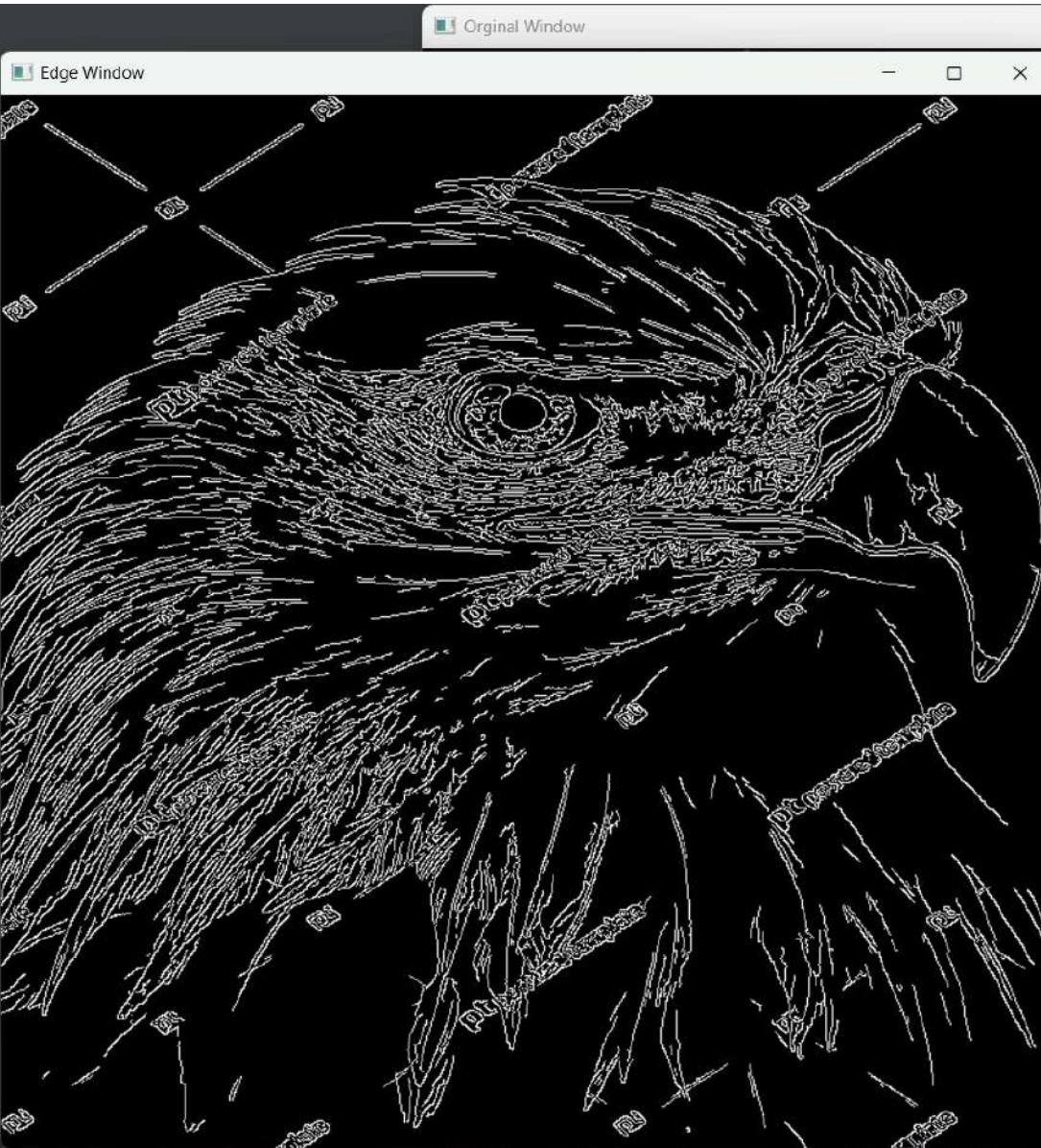
ENG IN

04:52 PM 01-06-2025

```
OpenCV Version control
Edge_detection.py
1 import cv2
2
3 # canny edge detection
4 # Noise reduction
5 # Intensity of the gradient image
6 # Non maximum suppression
7 # Thresholding
8
9 img_path="C:/Users/BIKASH/OneDrive/Pictures/"
10 img_name="eagle.jpeg"
11 img = cv2.imread(img_path+img_name,1)
12
13 resize_img = cv2.resize(img, dsize: (750,750)) # s
14 min_thresh = 100
15 max_thresh = 200
16
17 edge =cv2.Canny(resize_img,min_thresh,max_thresh)
18
19 cv2.imshow( winname: 'Original Window',img)
20 cv2.imshow( winname: 'Edge Window',edge)
21
22 cv2.waitKey(0)
23 cv2.destroyAllWindows()
```

Run Edge\_detection

OpenCV > Edge\_detection.py





OpenCV Version control


Current File

Reading\_video.py

```
1 import cv2
2
3 video_path = "C:/Users/BIKASH/OneDrive/Pictures/"
4 video_name = "ice.mp4"
5 video = cv2.VideoCapture(video_path + video_name)
6
7 width = 950
8 height = 550
9 dim = (width,height)
10
11 while video.isOpened():
12
13     _,frame = video.read()
14     frame = cv2.resize(frame,dim)
15     cv2.imshow( winname: 'Video Window',frame)
16
17     if cv2.waitKey(1) & 0xFF== ord('q'): # 0xFF--
18         break
19
20 cv2.destroyAllWindows()
```

Run Reading\_video

Video Window



OpenCV > Reading\_video.py

17:69 CRLF UTF-8 4 spaces Python 3.12 (OpenCV)

34°C Haze Search

04:55 PM 01-06-2025




OpenCV Version control

Writing\_video.py

```
1 import cv2
2
3 video_path = "C:/Users/BIKASH/OneDrive/Pictures/"
4 video_name = "forest.mp4"
5 video = cv2.VideoCapture(video_path + video_name)
6
7 width = 1280
8 height = 720
9 dim = (width,height)
10 FPS = 23.98
11
12 fourcc = cv2.VideoWriter_fourcc(*'mp4v')
13 output = cv2.VideoWriter('Output.mp4',fourcc,FPS,dim)
14
15 while video.isOpened():
16     ret,frame = video.read()
17     if ret:
18         output.write(frame)
19         cv2.imshow( winname: "Output Video Window",frame)
20         if cv2.waitKey(30) == ord('s'):
21             break
22     else:
23         break
24
25 cv2.destroyAllWindows()
```

Run Writing\_video

Output Video Window




OpenCV > Writing\_video.py

16:29 CRLF UTF-8 4 spaces Python 3.12 (OpenCV)

34°C Haze

Search



ENG IN 04:56 PM 01-06-2025



OpenCV Version control

Current File

Access\_camera.py

```
5 width = 720
6 height = 520
7 dim = (width,height)
8 FPS = 20
9
10 forcc = cv2.VideoWriter_fourcc(*'mp4v')
11 output = cv2.VideoWriter('live_cam.mp4',forcc,FPS,dim) #
12 while camera.isOpened():
13     ret,frame = camera.read()
14     if ret:
15         frame = cv2.resize(frame, dim) # Resize to match
16         output.write(frame)
17         cv2.imshow( winname: "Live Camera",frame)
18
19         key = cv2.waitKey(10) &0xFF
20         if key == ord(' '):
21             break
22     else:
23         break
24
25 camera.release()
26 output.release()
27 cv2.destroyAllWindows()
28
29
```

Run Access\_camera

Live Camera

OpenCV > Gaussain\_blur.py

11:104 CRLF UTF-8 4 spaces Python 3.12 (OpenCV)

34°C Haze

Search

ENG IN

04:58 PM 01-06-2025