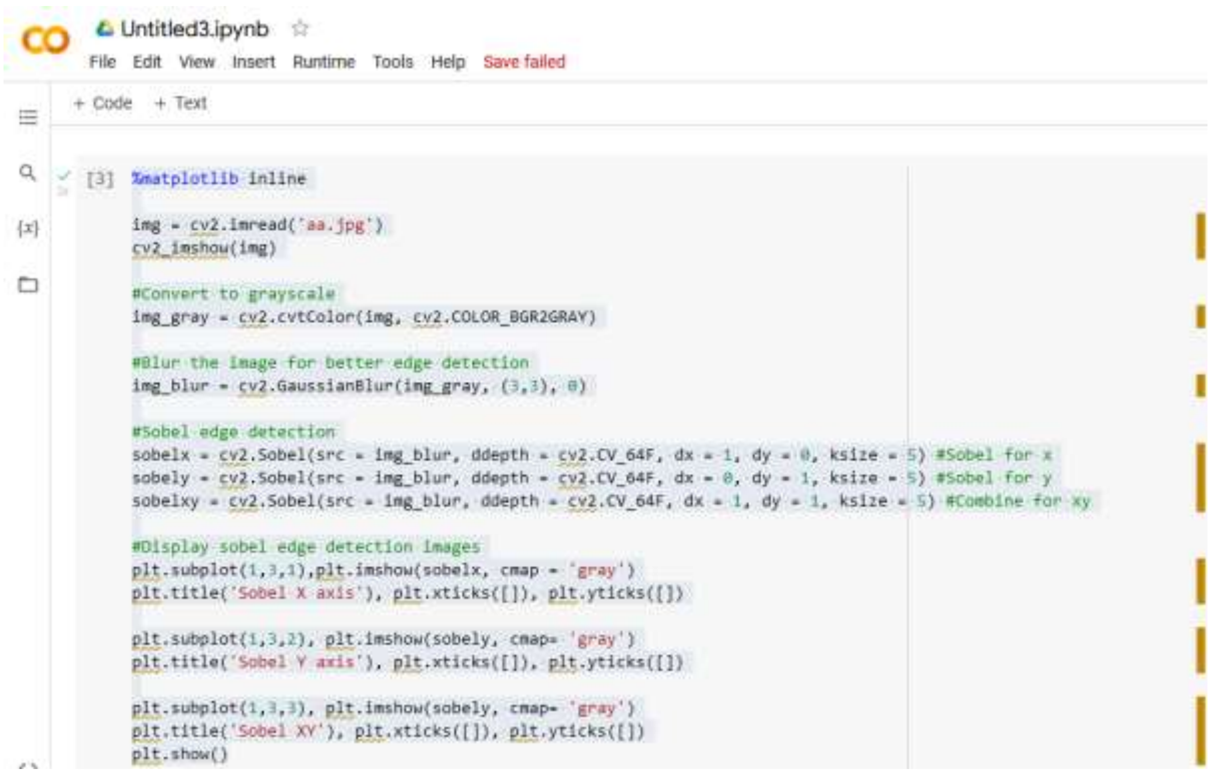
02_GAURAV BANE

```
%matplotlib inline

img = cv2.imread('aa.jpg')
cv2_imshow(img)

#Convert to grayscale
img_gray = cv2.cvtColor(img, cv2.COLOR_BGR2GRAY)

#Blur the image for better edge detection
img_blur = cv2.GaussianBlur(img_gray, (3,3), 0)

#Sobel edge detection
sobelx = cv2.Sobel(src = img_blur, ddepth = cv2.CV_64F, dx = 1, dy = 0,
 ksize = 5) #Sobel for x
sobely = cv2.Sobel(src = img_blur, ddepth = cv2.CV_64F, dx = 0, dy = 1,
 ksize = 5) #Sobel for y
sobelxy = cv2.Sobel(src = img_blur, ddepth = cv2.CV_64F, dx = 1, dy = 1
, ksize = 5) #Combine for xy

#Display sobel edge detection images
plt.subplot(1,3,1),plt.imshow(sobelx, cmap = 'gray')
plt.title('Sobel X axis'), plt.xticks([]), plt.yticks([])

plt.subplot(1,3,2), plt.imshow(sobely, cmap= 'gray')
plt.title('Sobel Y axis'), plt.xticks([]), plt.yticks([])

plt.subplot(1,3,3), plt.imshow(sobely, cmap= 'gray')
plt.title('Sobel XY'), plt.xticks([]), plt.yticks([])
plt.show()
```

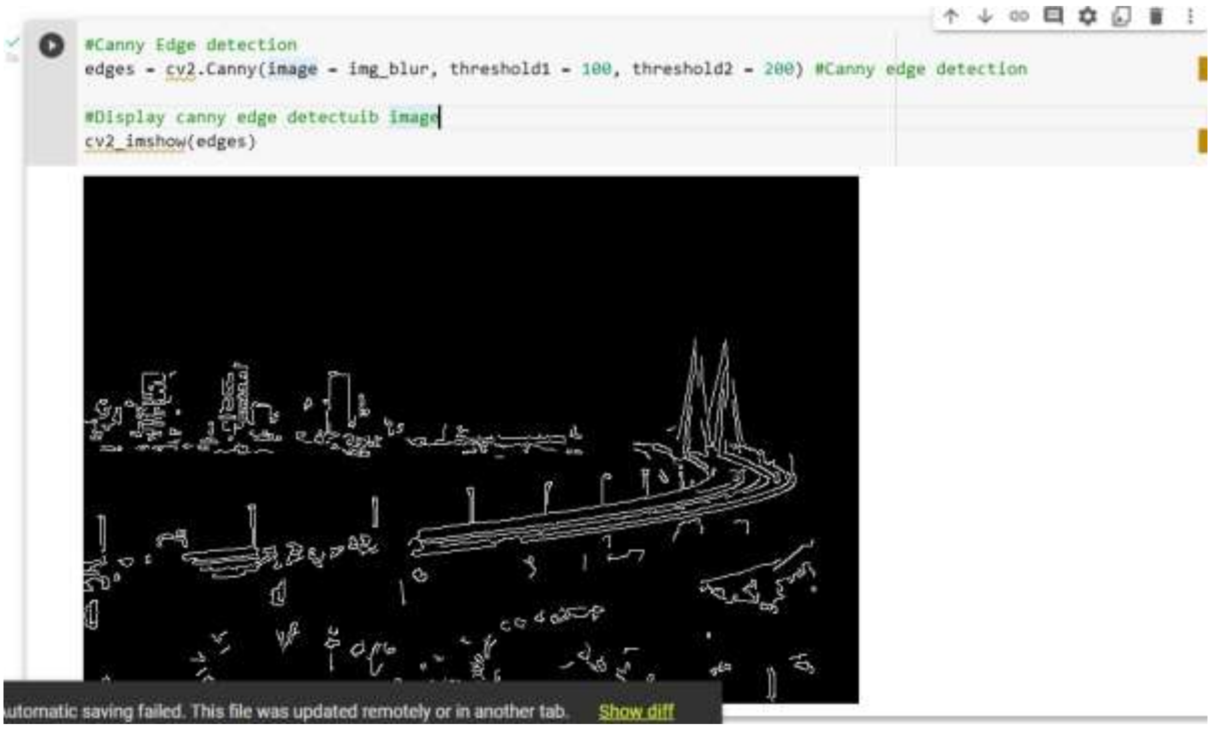
02_GAURAV BANE



```
%matplotlib inline

img = cv2.imread('aa.jpg')
cv2_imshow(img)

#Convert to grayscale
img_gray = cv2.cvtColor(img, cv2.COLOR_BGR2GRAY)

#Blur the image for better edge detection
img_blur = cv2.GaussianBlur(img_gray, (3,3), 0)

#Sobel edge detection
sobelx = cv2.Sobel(src = img_blur, ddepth = cv2.CV_64F, dx = 1, dy = 0, ksize = 5) #Sobel for x
sobely = cv2.Sobel(src = img_blur, ddepth = cv2.CV_64F, dx = 0, dy = 1, ksize = 5) #Sobel for y
sobelxy = cv2.Sobel(src = img_blur, ddepth = cv2.CV_64F, dx = 1, dy = 1, ksize = 5) #Combine for xy

#Display sobel edge detection images
plt.subplot(1,3,1),plt.imshow(sobelx, cmap = 'gray')
plt.title('Sobel X axis'), plt.xticks([]), plt.yticks([])

plt.subplot(1,3,2), plt.imshow(sobely, cmap= 'gray')
plt.title('Sobel Y axis'), plt.xticks([]), plt.yticks([])

plt.subplot(1,3,3), plt.imshow(sobely, cmap= 'gray')
plt.title('Sobel XY'), plt.xticks([]), plt.yticks([])
plt.show()
```

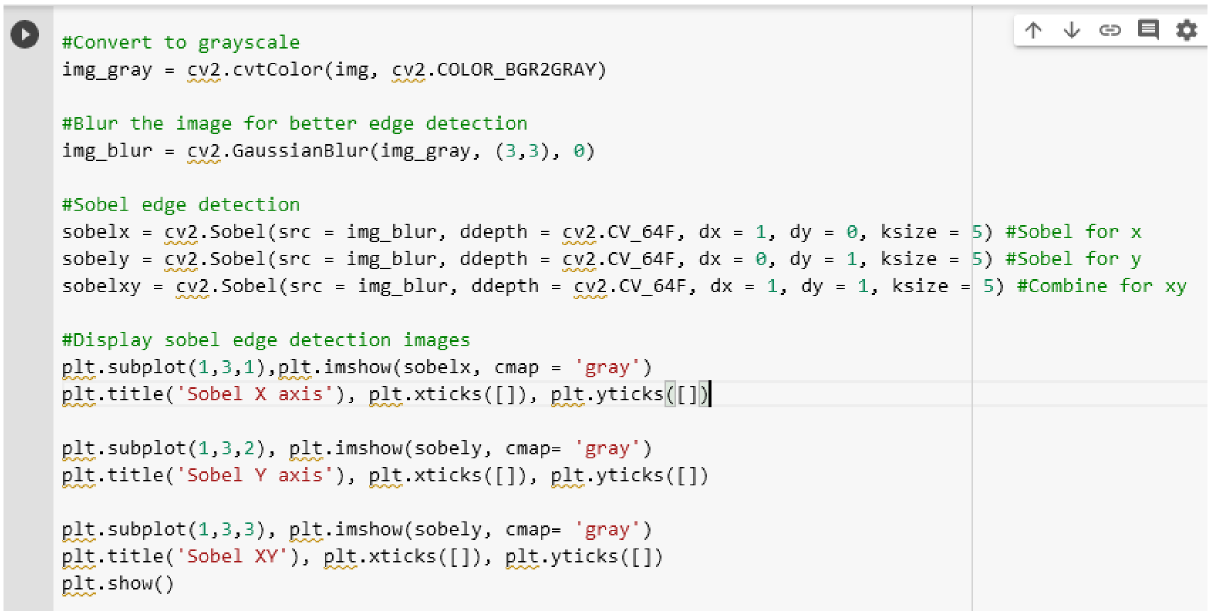
```
#Canny Edge detection
edges = cv2.Canny(image = img_blur, threshold1 = 100, threshold2 = 200)
 #Canny edge detection

#Display canny edge detectuib image
cv2_imshow(edges)
```

02_GAURAV BANE



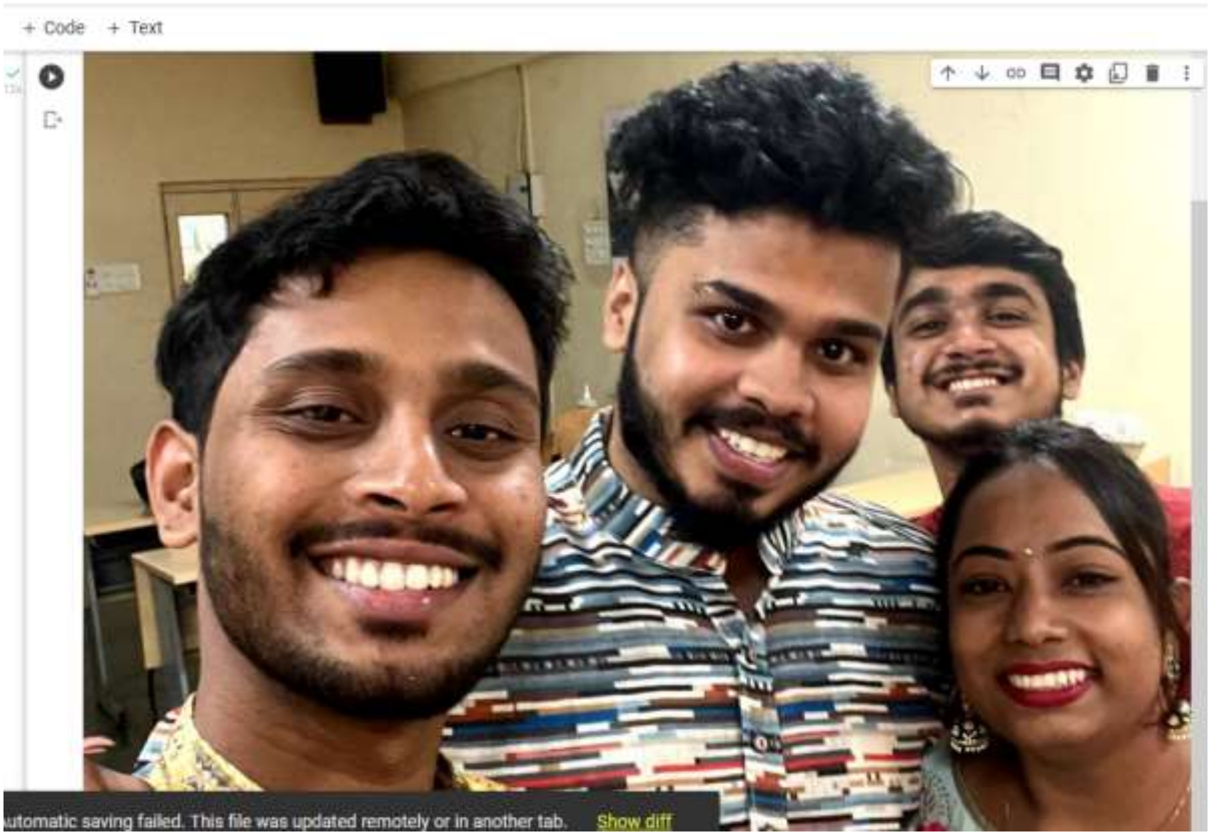
```
#Canny Edge detection
edges = cv2.Canny(image = img_blur, threshold1 = 100, threshold2 = 200) #Canny edge detection

#Display canny edge detectuib image
cv2_imshow(edges)
```

utomatic saving failed. This file was updated remotely or in another tab. Show diff

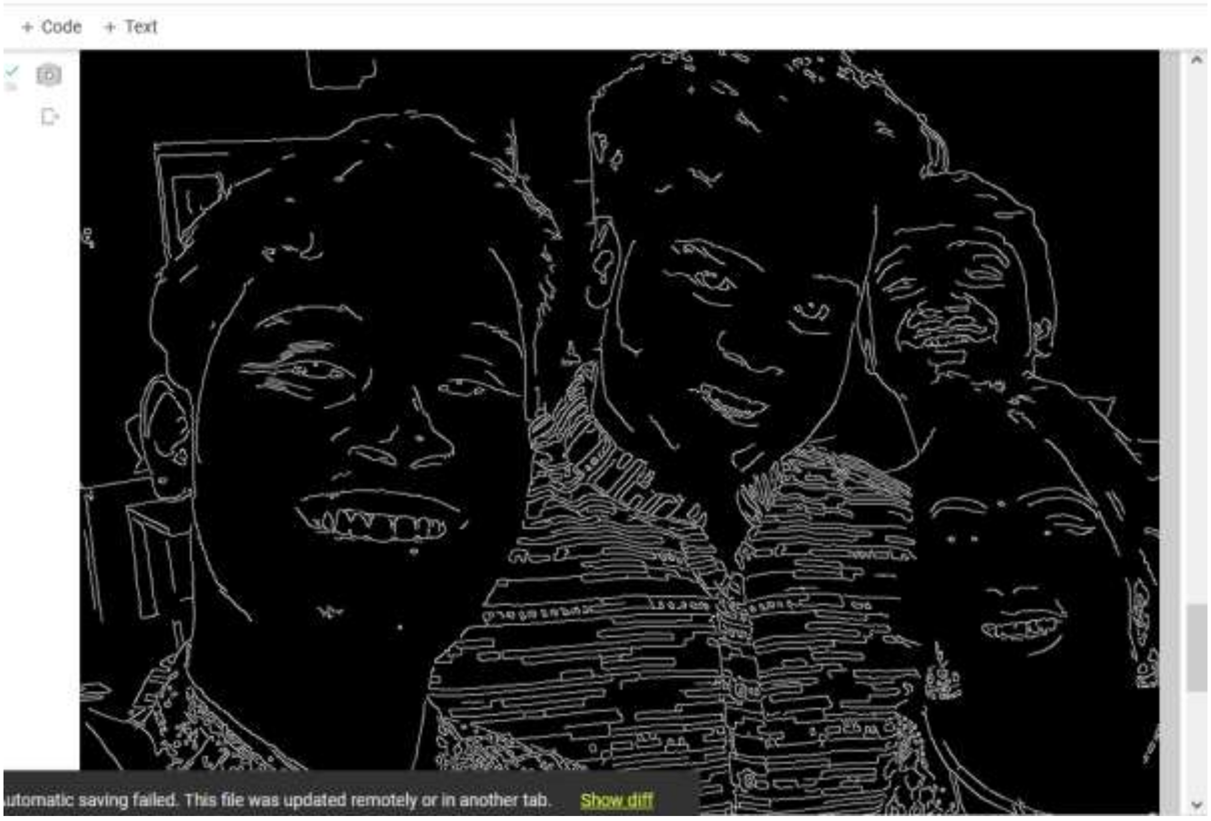


```
#Convert to grayscale
img_gray = cv2.cvtColor(img, cv2.COLOR_BGR2GRAY)

#Blur the image for better edge detection
img_blur = cv2.GaussianBlur(img_gray, (3,3), 0)

#Sobel edge detection
sobelx = cv2.Sobel(src = img_blur, ddepth = cv2.CV_64F, dx = 1, dy = 0, ksize = 5) #Sobel for x
sobely = cv2.Sobel(src = img_blur, ddepth = cv2.CV_64F, dx = 0, dy = 1, ksize = 5) #Sobel for y
sobelxy = cv2.Sobel(src = img_blur, ddepth = cv2.CV_64F, dx = 1, dy = 1, ksize = 5) #Combine for xy

#Display sobel edge detection images
plt.subplot(1,3,1),plt.imshow(sobelx, cmap = 'gray')
plt.title('Sobel X axis'), plt.xticks([]), plt.yticks([])

plt.subplot(1,3,2), plt.imshow(sobely, cmap= 'gray')
plt.title('Sobel Y axis'), plt.xticks([]), plt.yticks([])

plt.subplot(1,3,3), plt.imshow(sobely, cmap= 'gray')
plt.title('Sobel XY'), plt.xticks([]), plt.yticks([])
plt.show()
```

02_GAURAV BANE

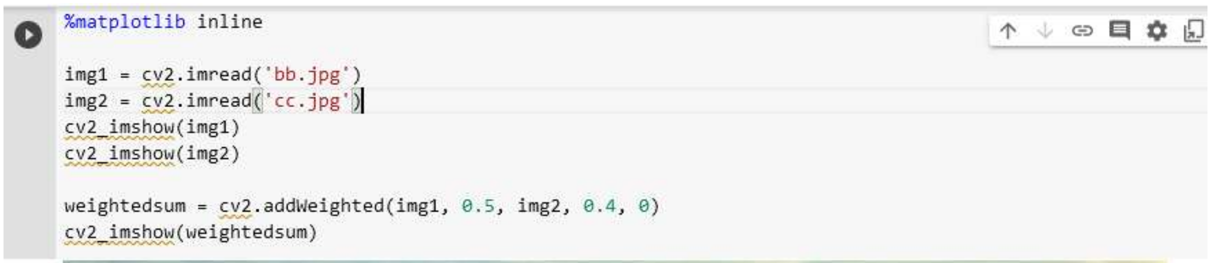


utomatic saving failed. This file was updated remotely or in another tab. Show diff

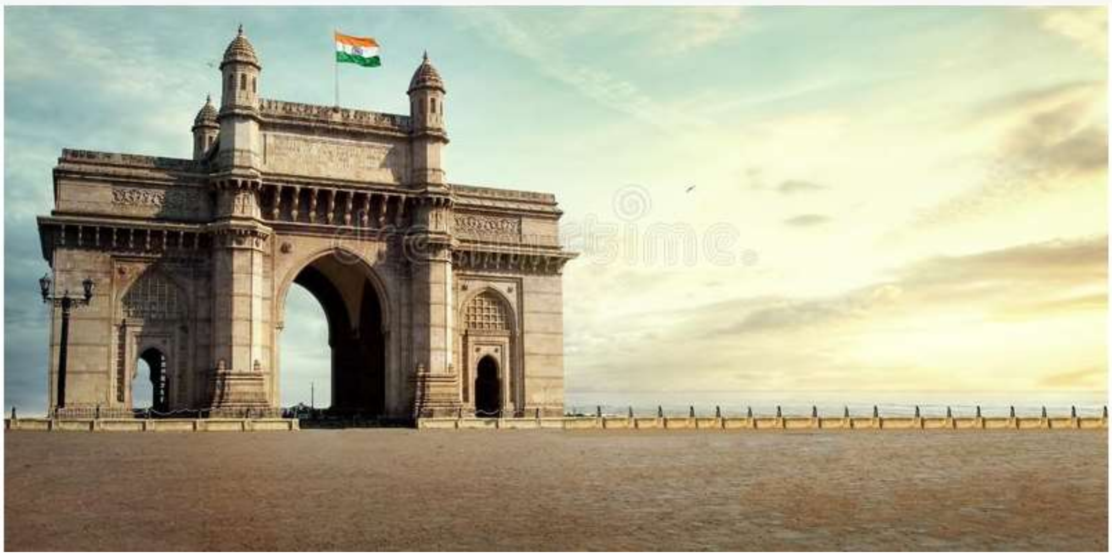


utomatic saving failed. This file was updated remotely or in another tab. Show diff

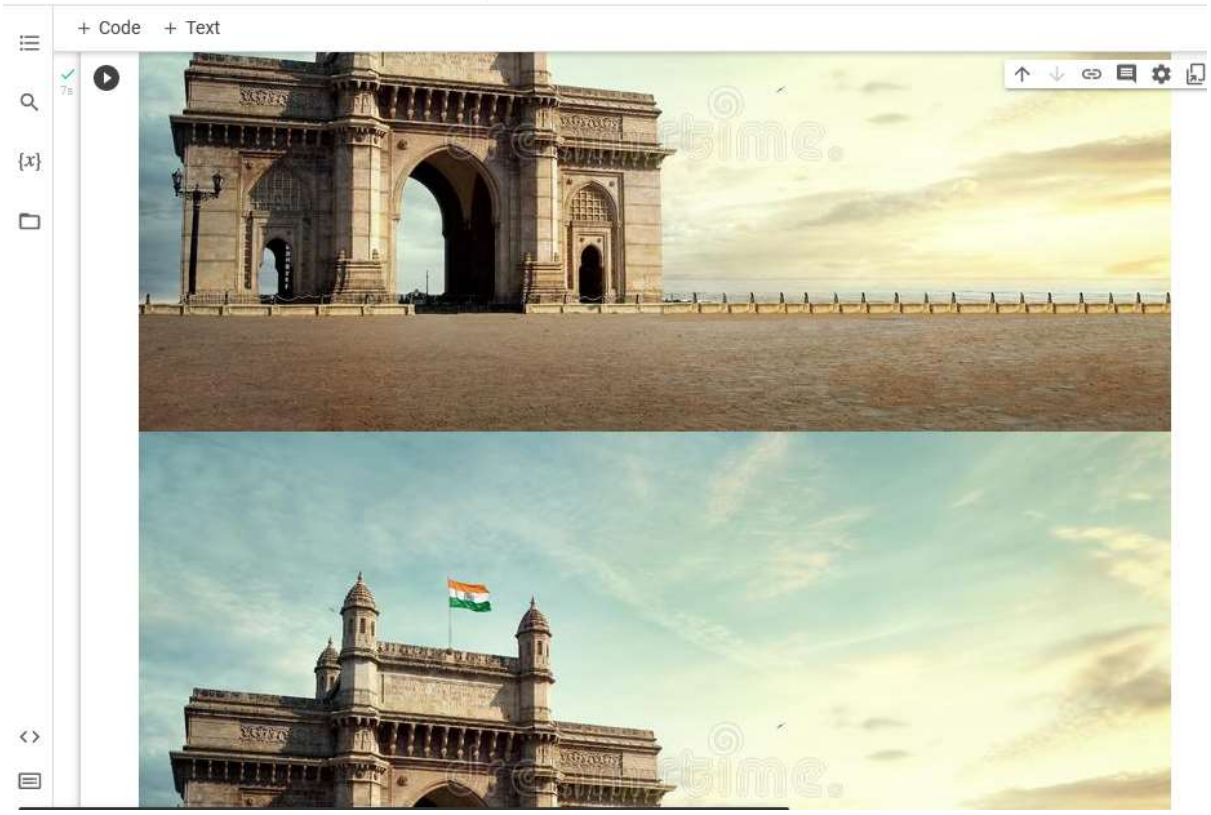
```
%matplotlib inline

image1 = np.zeros((400,400), dtype = "uint8")

img1 = cv2.imread('bb.jpg')
img2 = cv2.imread('cc.jpg')
cv2_imshow(img1)
cv2_imshow(img2)

weightedsum = cv2.addWeighted(img1, 0.5, img2, 0.4, 0)
cv2_imshow(weightedsum)
```



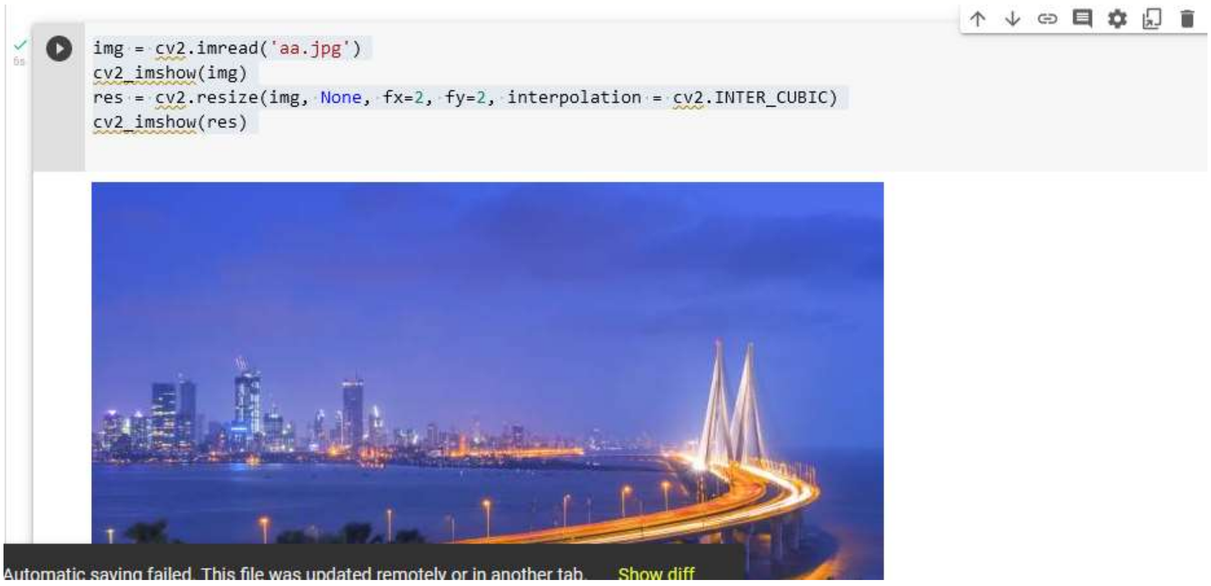
```
%matplotlib inline

img1 = cv2.imread('bb.jpg')
img2 = cv2.imread('cc.jpg')
cv2_imshow(img1)
cv2_imshow(img2)

weightedsum = cv2.addWeighted(img1, 0.5, img2, 0.4, 0)
cv2_imshow(weightedsum)
```



02_GAURAV BANE



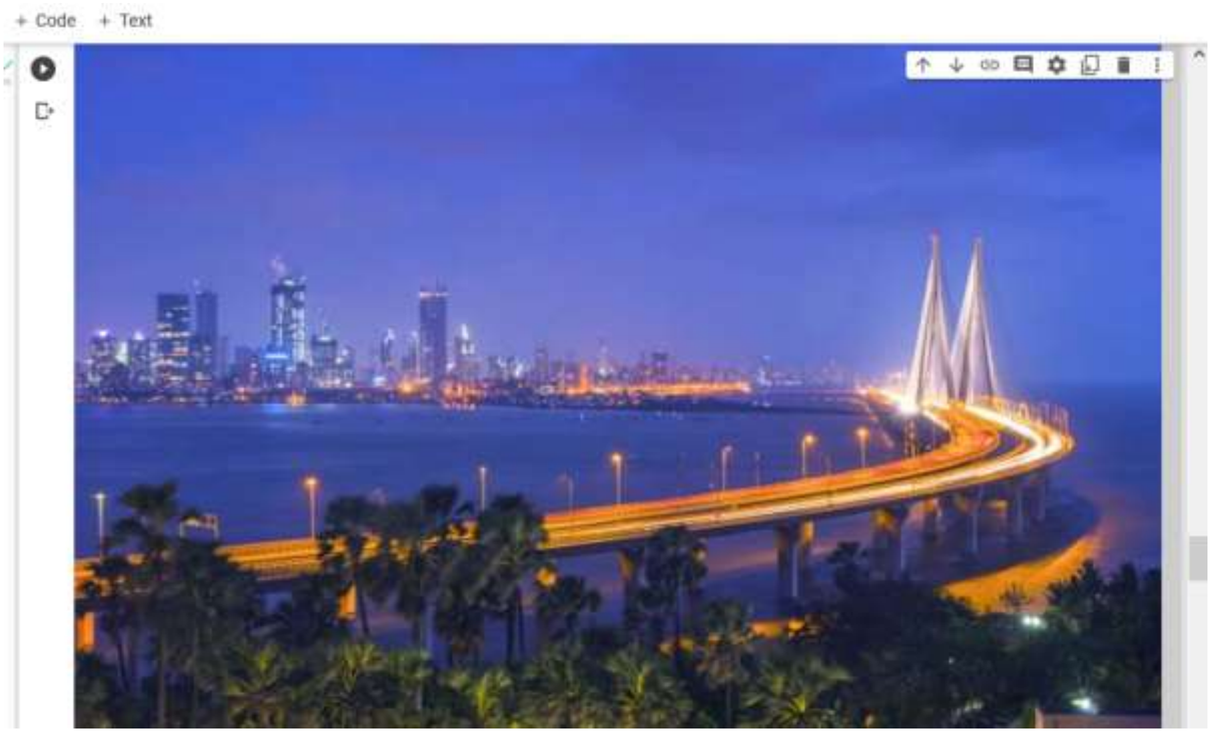
```
img = cv2.imread('aa.jpg')
cv2_imshow(img)
res = cv2.resize(img, None, fx=2, fy=2, interpolation = cv2.INTER_CUBIC
)
cv2_imshow(res)
```



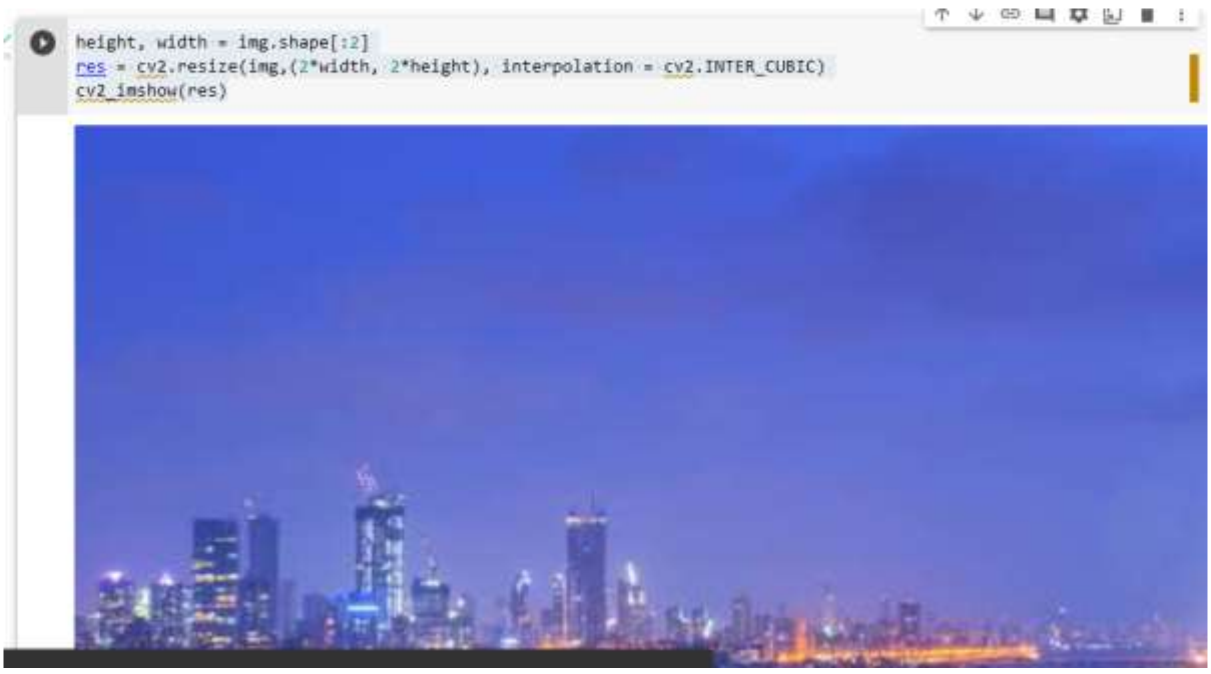
```
img = cv2.imread('aa.jpg')
cv2_imshow(img)
res = cv2.resize(img, None, fx=2, fy=2, interpolation = cv2.INTER_CUBIC)
cv2_imshow(res)
```

Automatic saving failed. This file was updated remotely or in another tab. Show diff

02_GAURAV BANE



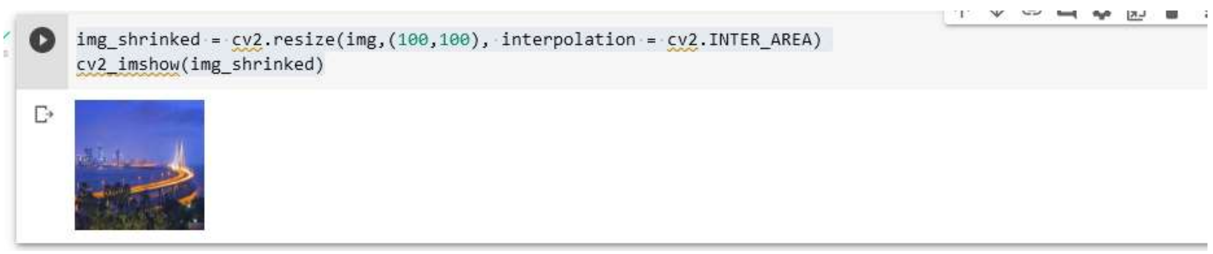
```
height, width = img.shape[:2]
res = cv2.resize(img,(2*width, 2*height), interpolation = cv2.INTER_CUB
IC)
cv2_imshow(res)
```



```
height, width = img.shape[:2]
res = cv2.resize(img,(2*width, 2*height), interpolation = cv2.INTER_CUBIC)
cv2_imshow(res)
```

02_GAURAV BANE

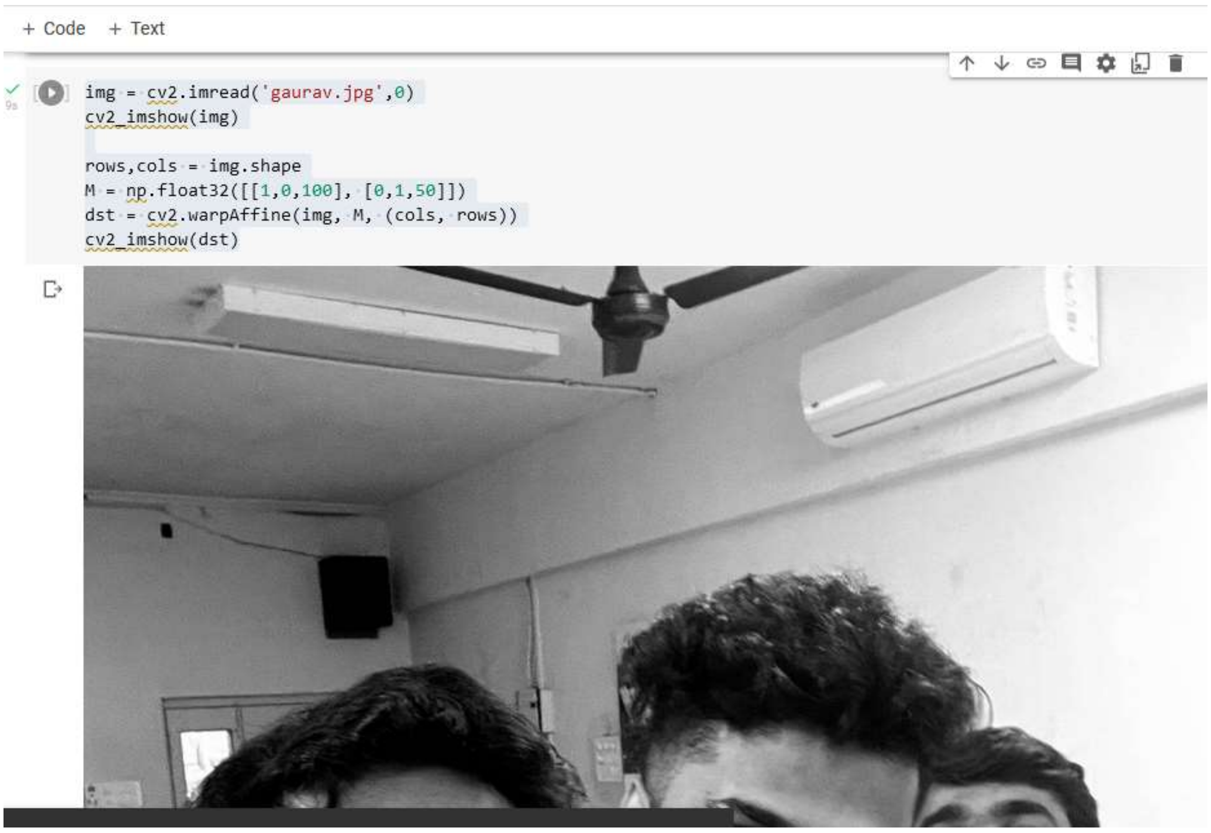
```
img_shrinked = cv2.resize(img,(100,100), interpolation = cv2.INTER_AREA
)
cv2_imshow(img_shrinked)
```



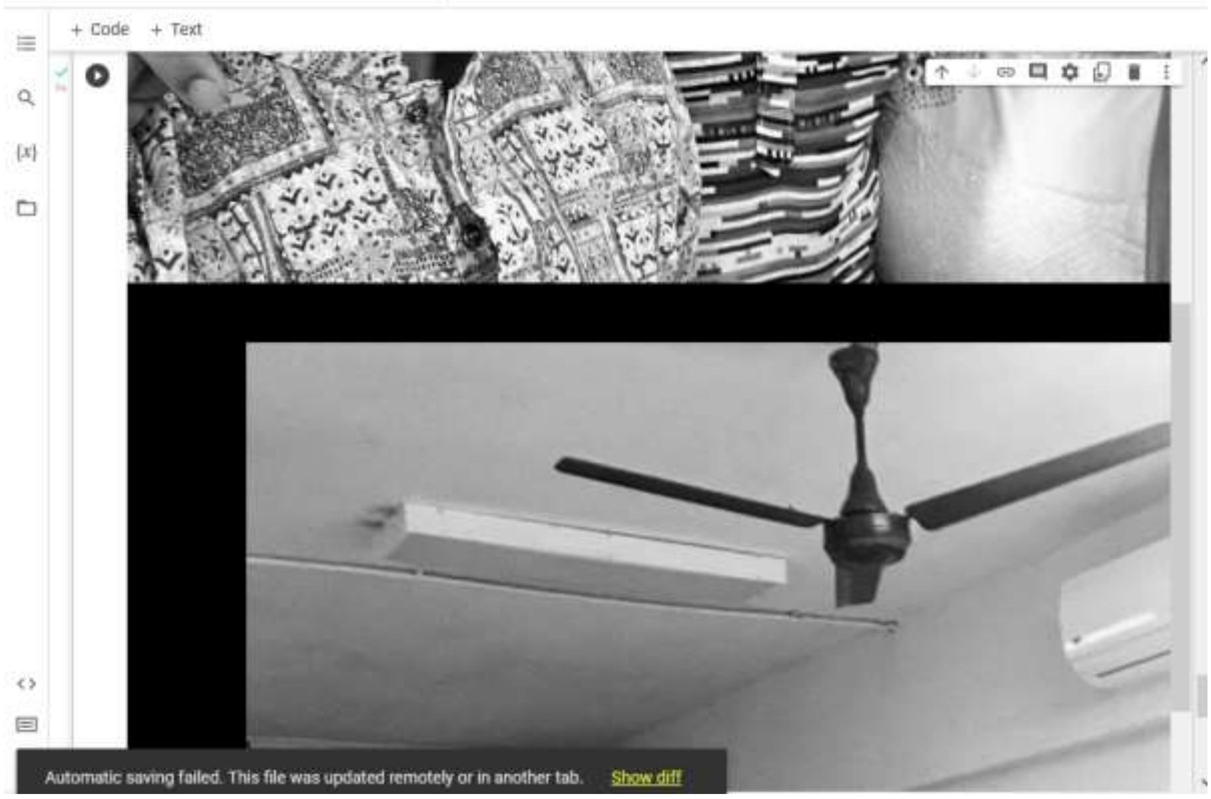
```
img_shrinked = cv2.resize(img,(100,100), interpolation = cv2.INTER_AREA)
cv2_imshow(img_shrinked)
```

```
img = cv2.imread('gaurav.jpg',0)
cv2_imshow(img)

rows,cols = img.shape
M = np.float32([[1,0,100], [0,1,50]])
dst = cv2.warpAffine(img, M, (cols, rows))
cv2_imshow(dst)
```

02_GAURAV BANE



```
img = cv2.imread('gaurav.jpg',0)
cv2_imshow(img)

rows,cols = img.shape
M = np.float32([[1,0,100], [0,1,50]])
dst = cv2.warpAffine(img, M, (cols, rows))
cv2_imshow(dst)
```

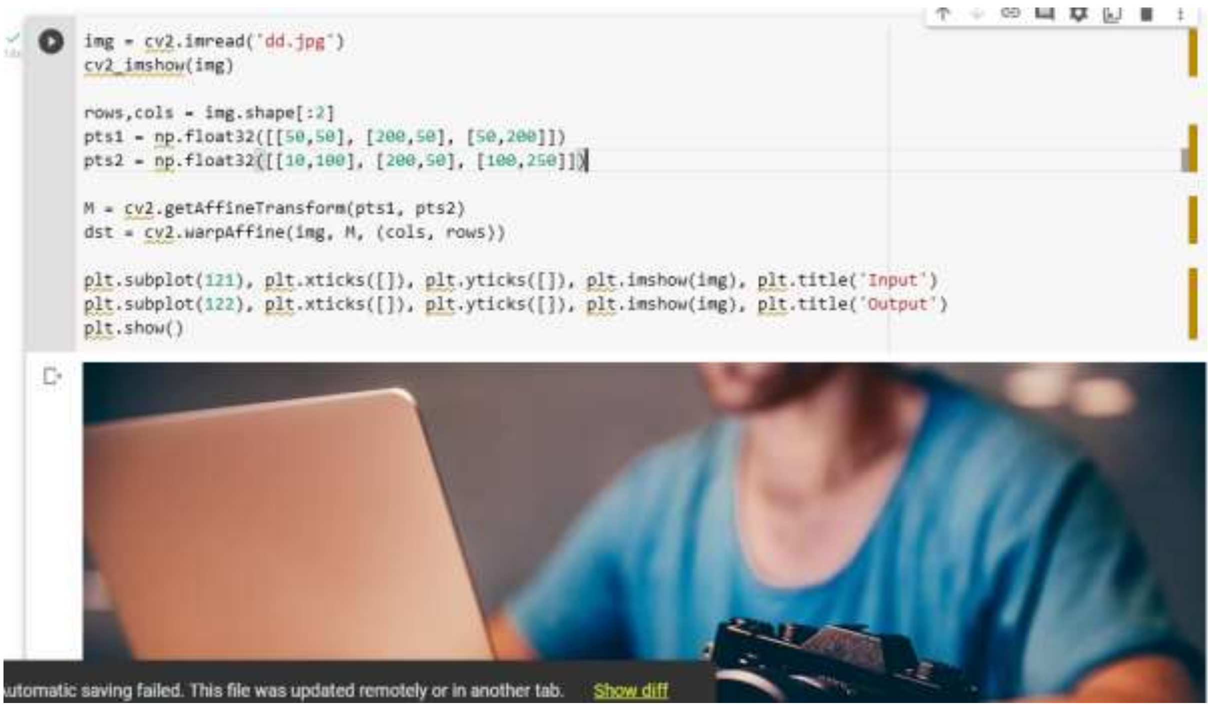


Automatic saving failed. This file was updated remotely or in another tab. Show diff

```
img = cv2.imread('dd.jpg')
```

02_GAURAV BANE

```
cv2_imshow(img)

rows,cols = img.shape[:2]
pts1 = np.float32([[50,50], [200,50], [50,200]])
pts2 = np.float32([[10,100], [200,50], [100,250]])

M = cv2.getAffineTransform(pts1, pts2)
dst = cv2.warpAffine(img, M, (cols, rows))

plt.subplot(121), plt.xticks([]), plt.yticks([]), plt.imshow(img), plt.
title('Input')
plt.subplot(122), plt.xticks([]), plt.yticks([]), plt.imshow(img), plt.
title('Output')
plt.show()
```



```
img = cv2.imread('dd.jpg')
cv2_imshow(img)

rows,cols = img.shape[:2]
pts1 = np.float32([[50,50], [200,50], [50,200]])
pts2 = np.float32([[10,100], [200,50], [100,250]])

M = cv2.getAffineTransform(pts1, pts2)
dst = cv2.warpAffine(img, M, (cols, rows))

plt.subplot(121), plt.xticks([]), plt.yticks([]), plt.imshow(img), plt.title('Input')
plt.subplot(122), plt.xticks([]), plt.yticks([]), plt.imshow(img), plt.title('Output')
plt.show()
```

Automatic saving failed. This file was updated remotely or in another tab. Show diff