1] image acquisition and representation

Code :

import cv2

path = 'E:/pranav sirsufale/spidy.jpg'

img = cv2.imread('spidy.jpg')

cv2.imshow('image',img)

cv2.waitKey(0)

cv2.destroyAllWindows()

**OUTPUT:**



2] image compression and storage.

Code :

### Image compresson and storage

import cv2

img = cv2.imread('spidy.jpg',1)

compressed\_img\_param = [cv2.IMWRITE\_JPEG\_QUALITY,50]

cv2.imwrite('compressed\_sharp\_img2.jpg',img,compressed\_img\_param)

OUTPUT:

|  |  |
| --- | --- |
| ORIGINAL IMAGE | COMPRESSED IMAGE |

3] image filtering and Restoration – Spatial domain filtering, Frequency domain filtering Image denoising and restoration.

Code :

import cv2

import numpy as np

img = cv2.imread('noicy.jpg',0)

denoised = cv2.GaussianBlur(img,(5,5),0)

cv2.imshow('original Image',img)

cv2.imshow('denoised iamge',denoised)

if( cv2.waitKey() == ord('q') ):

    cv2.destroyAllWindows()

OUTPUT:

|  |  |
| --- | --- |
| ORIGINAL IMAGE | DENOISED IMAGE |

4] Convert bright image into dark image.

Code :

import cv2

brigh\_image = cv2.imread('fish.jpg')

darkening\_factor = 50

dark\_image = cv2.subtract(brigh\_image, darkening\_factor)

dark\_image = cv2.max(dark\_image,0)

cv2.imshow('Original Image',brigh\_image)

cv2.imshow('Dark Image', dark\_image)

cv2.waitKey(0)

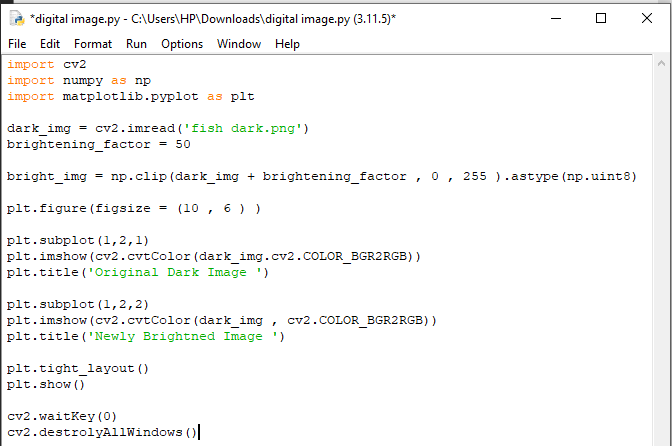
cv2.destroyAllwindows()

OUTPUT :

|  |  |
| --- | --- |
| ORIGINAL IMAGE | DARK IMAGE |

5] Convert dark image into bright image.

Code :

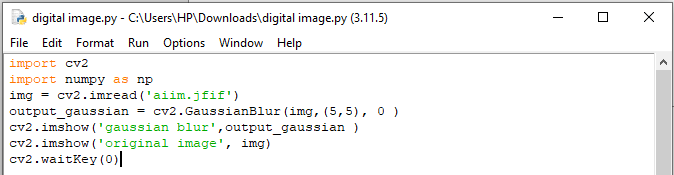


OUTPUT :

|  |  |
| --- | --- |
| ORIGINAL DARK IMAGE | NEWLY BRIGHTED IMAGE |

6] Take an image and apply Gaussian blur filter on it to sharp the image.

Code :

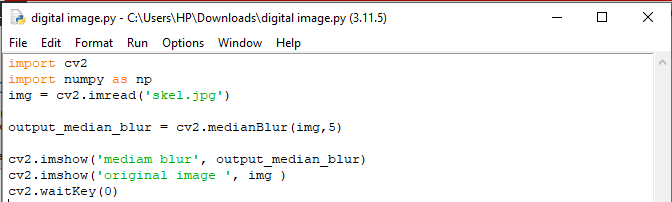


OUTPUT :

|  |  |
| --- | --- |
| ORIGINAL IMAGE | GAUSSIAN BLURED IMAGE |

7] Take an image and apply medium blur filter on it.

Code :

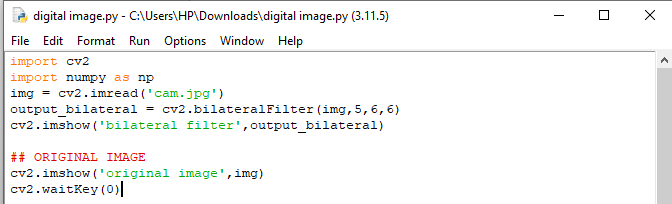


OUTPUT :

|  |  |
| --- | --- |
| ORIGINAL IMAGE | MEDIUM BLUR |

8] Take an image and apply bilateral filter on it.

Code :

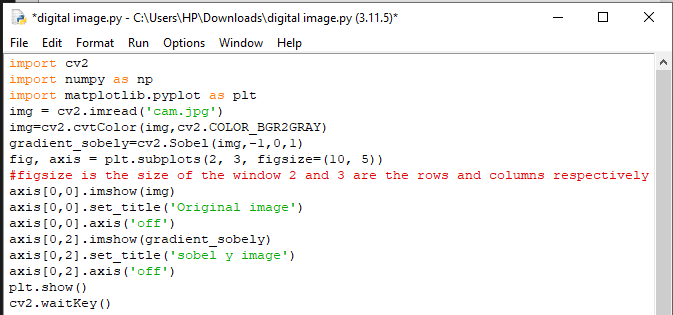


OUTPUT :

|  |  |
| --- | --- |
| ORIGINAL IMAGE | BILATERAL FILTERED IMAGE |

9] Write a program to show only horizontal feature using sobel operator.

Code :

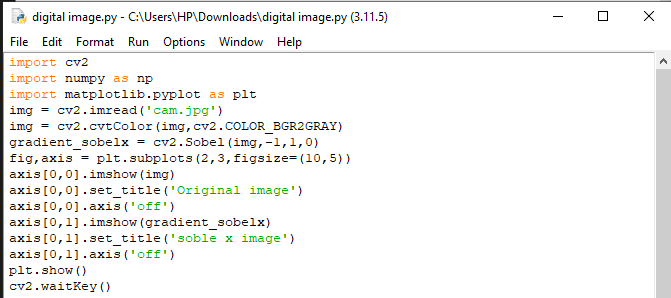


OUTPUT :

|  |  |
| --- | --- |
| ORIGINAL IMAGE | SOBEL Y IMAGE |

10] Write a program to show only vertical feature using sobel operator.

Code :



OUTPUT :

|  |  |
| --- | --- |
|  |  |

11] Write a program to apply laplacian filter on an image.

Code :

OUTPUT :

|  |  |
| --- | --- |
|  |  |

4] image compression and storage.

Code :

OUTPUT :

|  |  |
| --- | --- |
|  |  |

4] image compression and storage.

Code :

OUTPUT :

|  |  |
| --- | --- |
|  |  |