OpenCv

[2]:

```
import cv2 as cv
import numpy as np
```

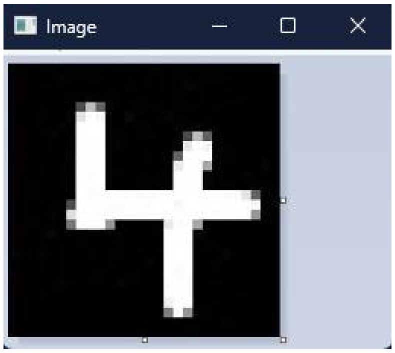
[3]:

```
image=cv.imread("D:\MIT ADT\Third Year - Sem 2\ML LAB\Assign 11 - OpenCV\\4.
 ↪jpeg")
# gray = cv.cvtColor(image, cv.COLOR_BGR2GRAY)
```

[3]:

```
print(image)
```

```
[[[237 251 245]
  [237 251 245]
  [239 251 245]
  …
  [250 246 245]
  [250 246 245]
  [250 246 245]]

 [[238 248 242]
  [238 248 242]
  [240 247 242]
  …
  [252 247 246]
  [251 248 244]
  [252 247 246]]

 [[247 249 243]
  [247 249 243]
  [247 249 243]
  …
  [255 249 246]
  [254 250 245]
  [255 249 246]]

 …

 [[227 212 203]
  [227 212 203]
  [227 212 203]
  …
  [227 212 203]
  [227 212 203]
  [227 212 203]]

 [[227 212 203]
  [227 212 203]
  [227 212 203]
  …
  [227 212 203]
  [227 212 203]
  [227 212 203]]

 [[227 212 203]
  [227 212 203]
  [227 212 203]
  …
  [227 212 203]
  [227 212 203]
  [227 212 203]]]
```



[4]:
```
image.shape
```

[4]:
```
(196, 257, 3)
```

[5]:
```
cv.imshow("Image",image)
cv.waitKey(0)
cv.destroyAllWindows()
cv.waitKey(1)
```

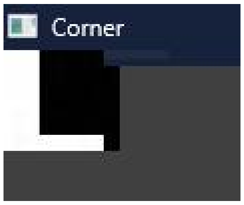
[5]:
```
-1
```

[6]:
```
image.shape
```

[6]:
```
(196, 257, 3)
```

[7]:
```
corner=image[50:100,50:100]
```

[9]:
```
cv.imshow("Corner", corner)
cv.waitKey(0)
cv.destroyAllWindows()
cv.waitKey(1)
```



[9]:
```
-1
```

[10]:
```
canvas = np.zeros((300, 300, 3), dtype = "uint8")
#RGB(255,255,255)
```

```
green = (0, 255, 0)
cv.line(canvas, (0, 0), (300, 300), green,4)
cv.imshow("Canvas", canvas)
cv.waitKey(0)
```

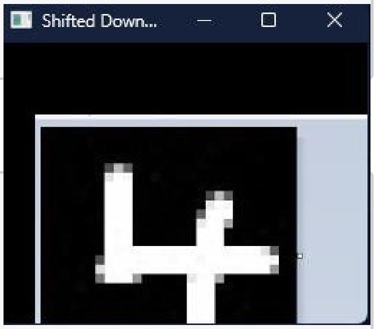
[10]: -1

[11]:
```
red = (0, 0, 255)
cv.line(canvas, (300, 0), (0, 300), red, 3)
cv.imshow("Canvas", canvas)
cv.waitKey(0)
```

[11]: -1

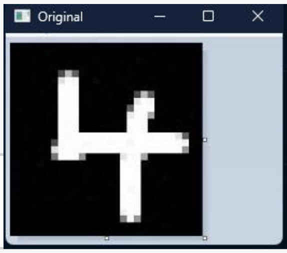
1 Image Processing

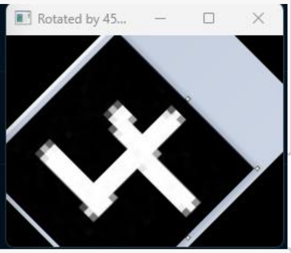
[12]:
```
# Image Translation
#The first row of the matrix is [1, 0, tx], where tx is the number of pixels we will shift the image left or right.
#Negative values of tx will shift the image to the left and positive values will shift the image to the right
M = np.float32([[1, 0, 25], [0, 1, 50]])
shifted = cv.warpAffine(image, M, (image.shape[1], image.shape[0]))
cv.imshow("Shifted Down and Right", shifted)
cv.waitKey(0)
```



[12]: -1

[14]:
```
# Rotation
cv.imshow("Original", image)
(h, w) = image.shape[:2]

center = (w // 2, h // 2)
M = cv.getRotationMatrix2D(center, 45, 1.0)
rotated = cv.warpAffine(image, M, (w, h))
cv.imshow("Rotated by 45 Degrees", rotated)
cv.waitKey(0)
```

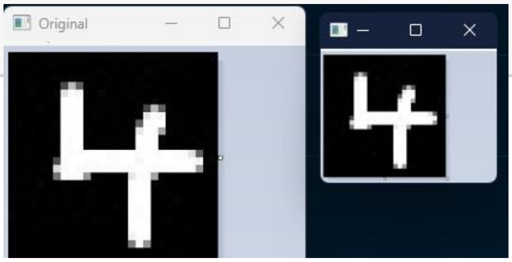




[14]: -1

[4]:
```
cv.imshow("Original", image)
print(image.shape)
r = 150.0 / image.shape[1]
dim = (150, int(image.shape[0] * r))
resized = cv.resize(image, dim, interpolation = cv.INTER_AREA)
print(resized.shape)
```

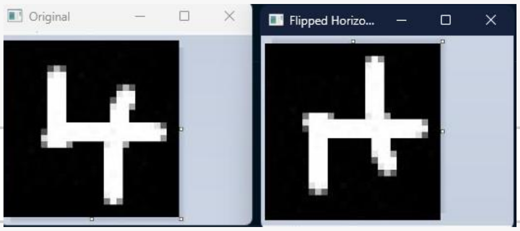
```
cv.imshow("Resized (Width)", resized)
cv.waitKey(0)
```



```
(196, 257, 3)
(114, 150, 3)
```

[4]:
```
-1
```

[5]:
```
cv.imshow("Original", image)
flipped = cv.flip(image, 0)
cv.imshow("Flipped Horizontally", flipped)
cv.waitKey(0)
```



[5]:
```
-1
```

[ ]:
```
python opencv cheat sheet pdf
```