Introduction to python

Contents

- PIL – python Imaging Library
- Matplotlib
- OpenCV

PIL – Python Imaging library

- Image module
  ```
  from PIL import Image
  ```

- Open the Image
  ```
  pil_im = Image.open('Image_1.jpg')
  ```

- Convert color image to greyscale
  ```
  pil_im = Image.open('Image_1.jpg').convert('L')
  ```

- Save the greyscale image
  ```
  pil_im.save('Image_2.jpg')
  ```

Copy and paste regions

- Crop a part of the Image

```
box = (100,100,400,400) #Where coordinates are (left, upper, right, lower)
region = pil_im.crop(box)
```

- Rotate the region by 180 degrees

```
region = region.transpose(Image.ROTATE_180)
```

- Paste the region on the Image

```
pil_im.paste(region,box)
```

Resize and Rotate

- Resize the Image

  Out = pil_im.resize((128,128))

- Rotate the Image

  Out = pil_im.rotate(45)

Matplotlib

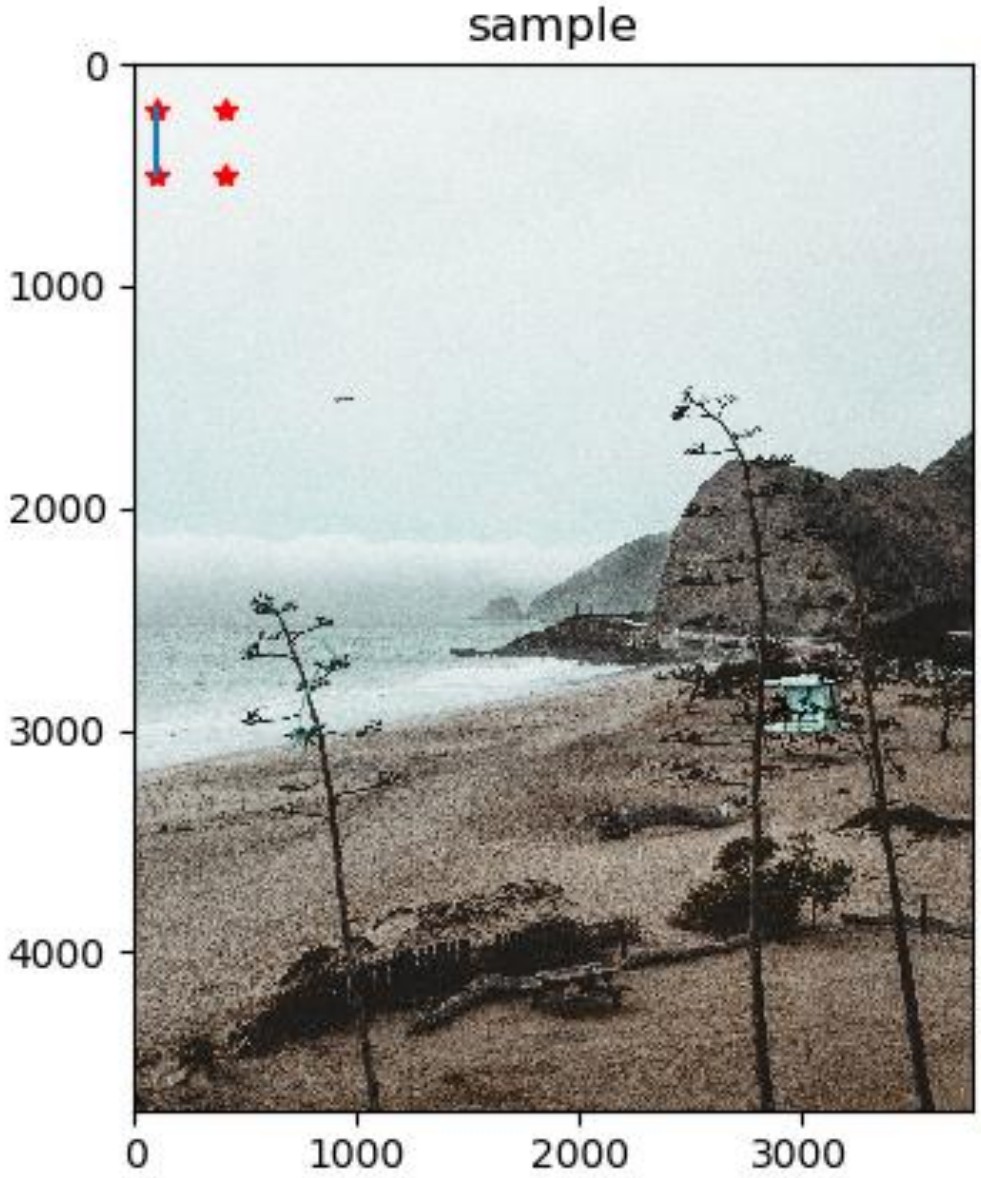
- Matplotlib is a good graphics library with much more powerful features than the plotting available in PIL.

- Matplotlib's PyLab is the set of functions that allows the user to create plots.

Plotting Images, points and Lines

```
from PIL import Image
from pylab import *

# read image to array
im = array(Image.open('Image_1.jpg'))

# Plot the image
imshow(im)

# Some points
x = [100,100,400,400]
y = [200,500,200,500]

# Plot the points with red star-markers
plot(x, y, 'r*')

# line plot connecting the first 2 points
plot(x[:2], y[:2])

# add title and show the plot
title('Sample plot')
show()
```

output



- The show() command starts the figure GUI and raises the figure windows.
- The GUI loop blocks the script until the last figure window is closed.
- You should call show() only once per script, usually at the end.

OpenCV

- OpenCV is a C++ library for computer vision. It has a python interface as well.

- Following document might be useful for installing OpenCV, you may follow any other latest reference as well

https://www.learnopencv.com/install-opencv-3-and-dlib-on-windows-python-only/

Reading and writing images

```
import cv2

# read image
im = cv2.imread('Image_1.jpg')
h,w = im.shape[:2]
print h,w

# save image
cv2.imwrite('out.jpg', im)
```

Color spaces

- In OpenCV, Images are not stored in conventional RGB format; they are stored in BGR format (reverse order)

- Colorspace conversions are done using the function cvtcolor()

- Convert to grayscale

```
gray = cv2.cvtcolor(im, cv2.COLOR_BGR2GRAY)

# show the result in a OpenCV window
cv2.imshow('temp image', im)
waitKey(0)
```

- waitkey(0) will display the image until its closed
- waitkey(t) will display the image for t milliseconds.

References

› Programming computer vision with python by Jan Erik Solem (Chap 1.1-1.2 and Chap 10.1-10.2)

π

Thanks