

LetsGrowMore

Priyanshi Badaya

Data Science Intern

Image to Pencil Sketch with Python

Description- We need to read the image in RBG format and then convert it to a grayscale image. This will turn an image into a classic black and white photo. Then the next thing to do is invert the grayscale image also called negative image, this will be our inverted grayscale image. Inversion can be used to enhance details. Then we can finally create the pencil sketch by mixing the grayscale image with the inverted blurry image. This can be done by dividing the grayscale image by the inverted blurry image. Since images are just arrays, we can easily do this programmatically using the divide function from the cv2 library in Python.

IMPORTING LIBRARIES

```
In [12]: pip install --user opencv-python

Requirement already satisfied: opencv-python in c:\users\A\appdata\roaming\python\python39\site-packages (4.5.5.62)
Requirement already satisfied: numpy>=1.17.3 in c:\users\A\appdata\roaming\python\python39\site-packages (from opencv-python) (1.22.2)
Note: you may need to restart the kernel to use updated packages.

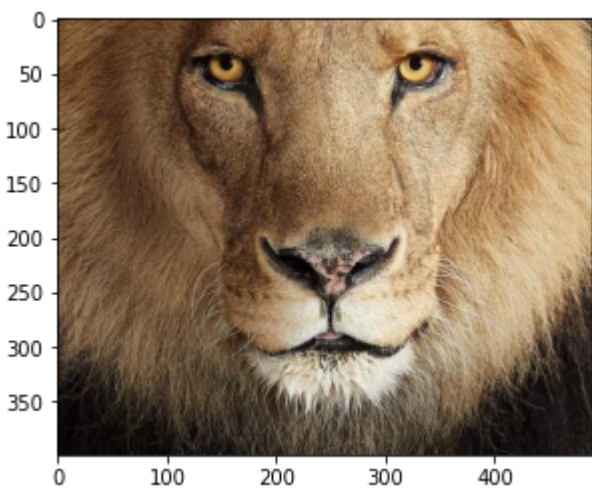
In [2]: pip install cv2-plt-imshow

Defaulting to user installation because normal site-packages is not writeable
Requirement already satisfied: cv2-plt-imshow in c:\users\A\appdata\roaming\python\python39\site-packages (0.0.1)
Requirement already satisfied: opencv-python=4.2.0.34 in c:\users\A\appdata\roaming\python\python39\site-packages (from cv2-plt-imshow) (4.5.5.62)
Requirement already satisfied: matplotlib>=3.2.2 in c:\users\A\appdata\roaming\python\python39\site-packages (from cv2-plt-imshow) (3.5.1)
Requirement already satisfied: fonttools>=4.22.0 in c:\users\A\appdata\roaming\python\python39\site-packages (from matplotlib>=3.2.2->cv2-plt-imshow) (4.29.1)
Requirement already satisfied: python-dateutil>=2.7 in c:\users\A\appdata\roaming\python\python39\site-packages (from matplotlib>=3.2.2->cv2-plt-imshow) (2.8.2)
Requirement already satisfied: cycler>=0.10 in c:\users\A\appdata\roaming\python\python39\site-packages (from matplotlib>=3.2.2->cv2-plt-imshow) (0.11.0)
Requirement already satisfied: numpy>=1.17 in c:\users\A\appdata\roaming\python\python39\site-packages (from matplotlib>=3.2.2->cv2-plt-imshow) (1.22.2)
Requirement already satisfied: packaging>=20.0 in c:\users\A\appdata\roaming\python\python39\site-packages (from matplotlib>=3.2.2->cv2-plt-imshow) (21.3)
Requirement already satisfied: kiwisolver>=1.0.1 in c:\users\A\appdata\roaming\python\python39\site-packages (from matplotlib>=3.2.2->cv2-plt-imshow) (1.3.2)
Requirement already satisfied: pillow>=6.2.0 in c:\users\A\appdata\roaming\python\python39\site-packages (from matplotlib>=3.2.2->cv2-plt-imshow) (9.0.1)
Requirement already satisfied: pyparsing>=2.2.1 in c:\users\A\appdata\roaming\python\python39\site-packages (from matplotlib>=3.2.2->cv2-plt-imshow) (3.0.7)
Requirement already satisfied: six>=1.5 in c:\users\A\appdata\roaming\python\python39\site-packages (from python-dateutil>=2.7->matplotlib>=3.2.2->cv2-plt-imshow) (1.16.0)
Note: you may need to restart the kernel to use updated packages.
```

```
In [3]: import cv2
from cv2_plt_imshow import cv2_plt_imshow
import matplotlib.pyplot as plt
```

```
In [4]: original_image=cv2.imread("D:\\Lion.jpg")
cv2_plt_imshow(original_image)
```

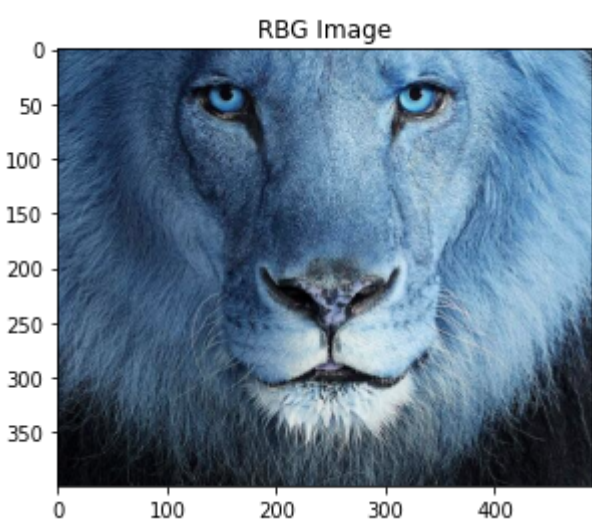
Out[4]: <matplotlib.image.AxesImage at 0x2c1ce9d9ac0>



CONVERTING IMAGE INTO RBG FORMAT

```
In [5]: rgb_image = cv2.cvtColor(original_image, cv2.COLOR_BGR2RGB)
plt.title('RBG Image')
cv2_plt_imshow(rgb_image)
```

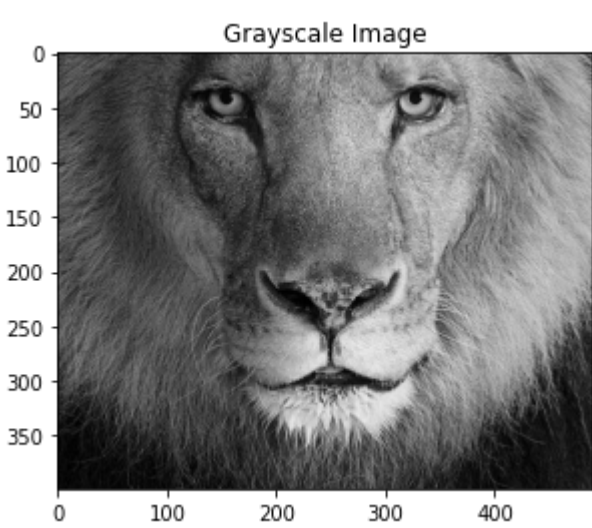
Out[5]: <matplotlib.image.AxesImage at 0x2c1cea454f0>



CONVERTING IMAGE INTO GRAYSCALE FORMAT

```
In [6]: grayscale_image = cv2.cvtColor(rgb_image, cv2.COLOR_BGR2GRAY)
plt.title('Grayscale Image')
cv2_plt_imshow(grayscale_image)
```

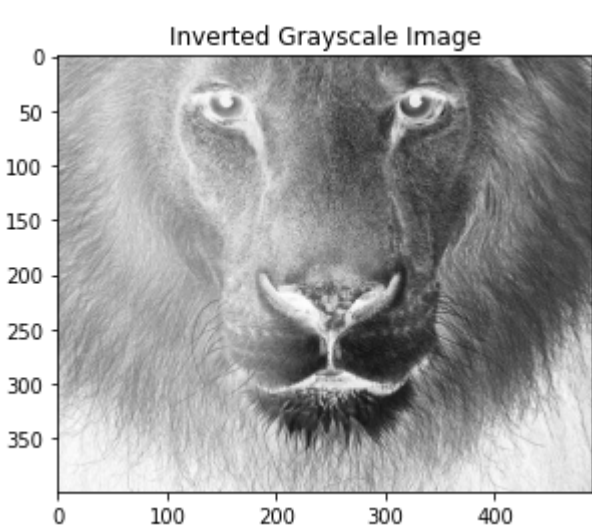
Out[6]: <matplotlib.image.AxesImage at 0x2c1ceaf9d30>



INVERTING THE GRAYSCALE IMAGE

```
In [7]: inverted_image = 255 - grayscale_image
plt.title('Inverted Grayscale Image')
cv2_plt_imshow(inverted_image)
```

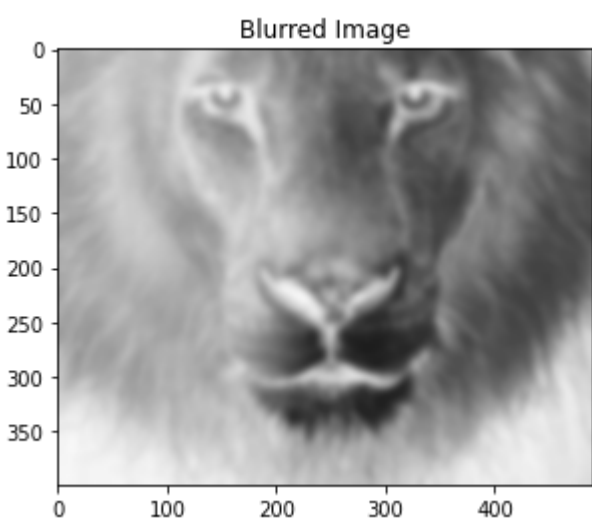
Out[7]: <matplotlib.image.AxesImage at 0x2c1ceb7b430>



CONVERTING TO BLURRED IMAGE

```
In [8]: blurred_image = cv2.GaussianBlur(inverted_image,(21, 21), 0)
plt.title('Blurred Image')
cv2_plt_imshow(blurred_image)
```

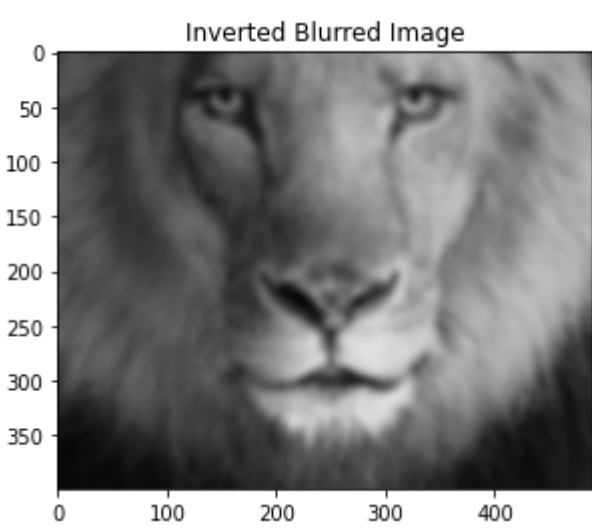
Out[8]: <matplotlib.image.AxesImage at 0x2c1cebd0670>



INVERTING THE BLURRED IMAGE

```
In [9]: inverted_blurred = 255 - blurred_image
plt.title('Inverted Blurred Image')
cv2_plt_imshow(inverted_blurred)
```

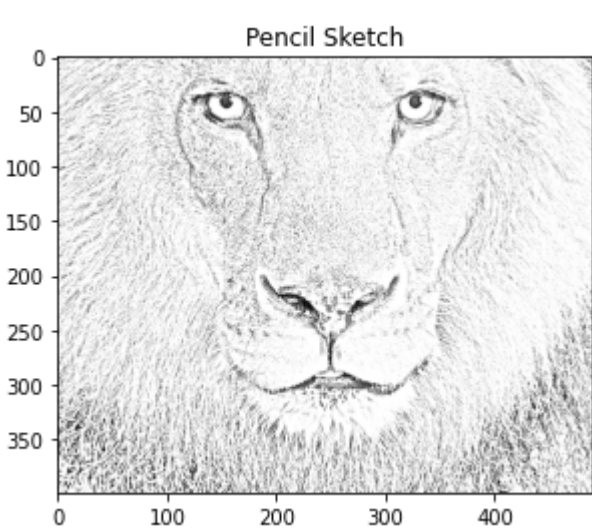
Out[9]: <matplotlib.image.AxesImage at 0x2c1cf11c10>



PENCIL SKETCH OF THE IMAGE

```
In [10]: pencil_sketch = cv2.divide(grayscale_image, inverted_blurred, scale=256.0)
plt.title('Pencil Sketch')
cv2_plt_imshow(pencil_sketch)
```

Out[10]: <matplotlib.image.AxesImage at 0x2c1ceb0d6a0>



SHOWING ORIGINAL IMAGE AND THE PENCIL SKETCH

```
In [11]: img1 = cv2.cvtColor(original_image, cv2.COLOR_BGR2RGB)
img2 =cv2.cvtColor(pencil_sketch, cv2.COLOR_BGR2RGB)
NUM_ROWS = 1
IMGs_IN_ROW = 2
f, ax = plt.subplots(NUM_ROWS, IMGs_IN_ROW, figsize=(16,6))
ax[0].imshow(img1)
ax[1].imshow(img2)
ax[0].set_title('Original Image')
ax[1].set_title('Pencil Sketch')
plt.tight_layout()
plt.show()
```

