

Image Prepossessing

October 10, 2022

1 Image Prepossessing

Table of Contents

1. [Install Necessary Libraries Python](#)
2. [Remove Background from Image](#)
3. [Remove Noise from Image](#)
4. [Improving Image Quality](#)

1.1 Install Necessary Libraries Python

```
[1]: #!pip install rembg
```

```
[ ]: #!pip install Pillow
```

```
[2]: #!pip install opencv-python
```

```
[3]: #!pip install numpy
```

```
[4]: #!pip install matplotlib
```

1.2 Remove Background from Image

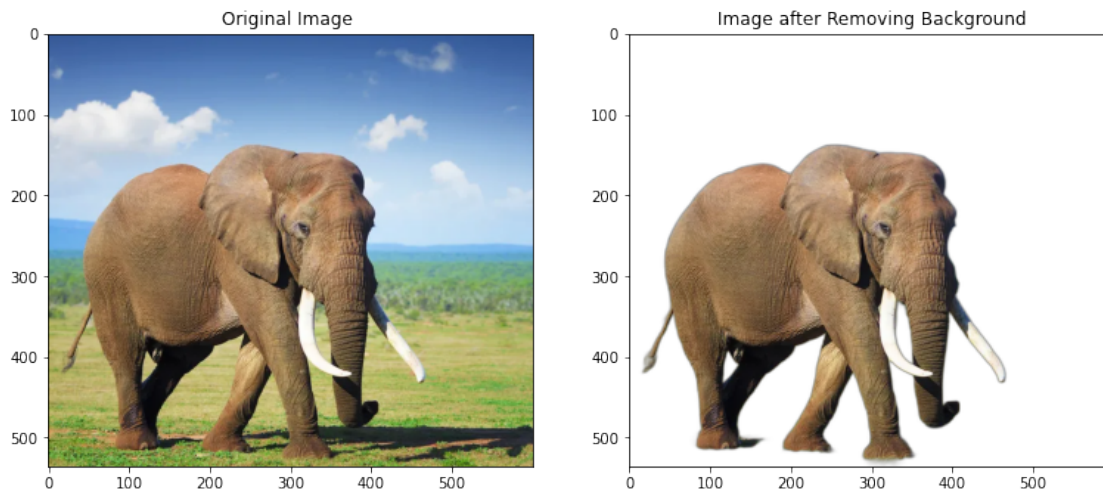
```
[5]: from rembg import remove
from PIL import Image
input_path = 'Input_Images/elephant.jpg'
output_path = 'output_elephant.png'
input = Image.open(input_path)
output = remove(input)
output.save(output_path)
```

```
[6]: import matplotlib.pyplot as plt
import matplotlib.image as mpimg
fig = plt.figure(figsize=(20,12))
plt.subplot(2,3,1)
```

```
plt.imshow(input)
plt.title('Original Image ')

plt.subplot(2,3,2)
plt.imshow(output)
plt.title('Image after Removing Background')
```

```
[6]: Text(0.5, 1.0, 'Image after Removing Background')
```



1.3 Remove Noise from Image

```
[7]: import numpy as np

import cv2

from matplotlib import pyplot as plt

image = cv2.imread('Input_Images/Discovery_Museum.jpg',1)

image_bw = cv2.imread('Input_Images/Discovery_Museum.jpg',0)
```

```
[8]: noiseless_image_bw = cv2.fastNlMeansDenoising(image_bw, None, 20, 7, 21)

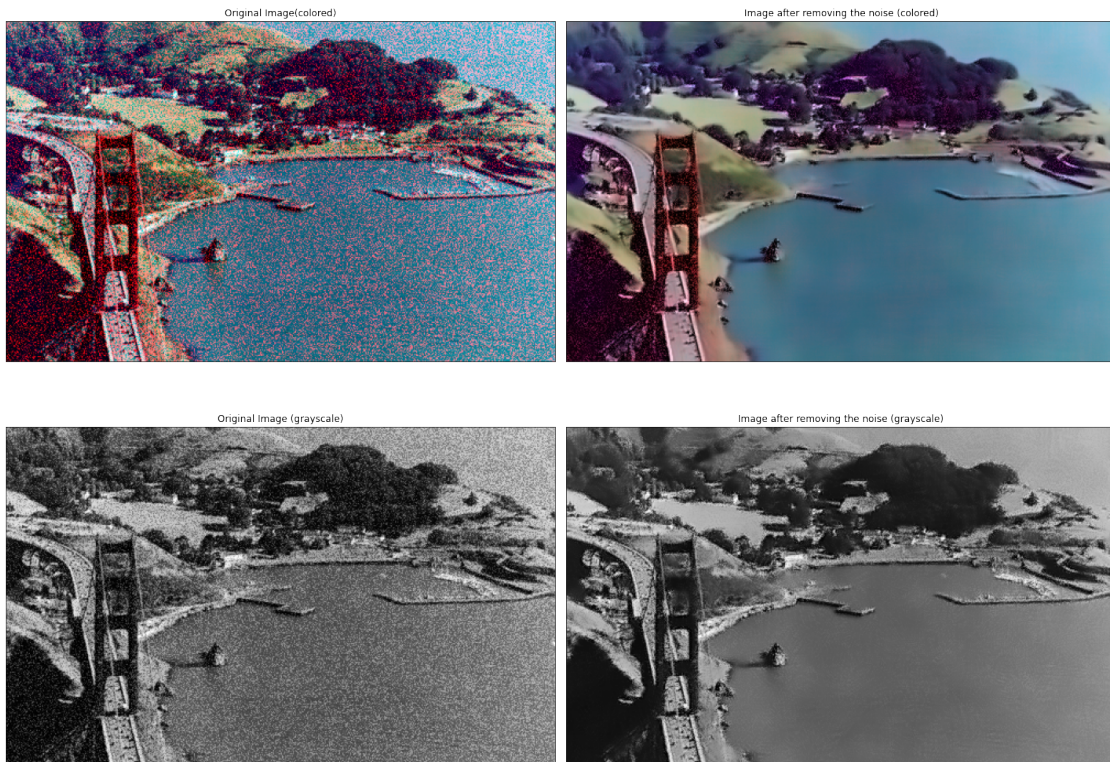
noiseless_image_colored = cv2.fastNlMeansDenoisingColored(image, None, 20, 20, 7, 21)
```

```
[9]: titles = ['Original Image(colored)', 'Image after removing the noise (colored)',
             ↪ 'Original Image (grayscale)', 'Image after removing the noise (grayscale)']
images = [image, noiseless_image_colored, image_bw, noiseless_image_bw]
plt.figure(figsize=(20,15))
```

```

for i in range(4):
    plt.subplot(2,2,i+1)
    plt.imshow(cv2.cvtColor(images[i],cv2.COLOR_BGR2RGB))
    plt.title(titles[i])
    plt.xticks([])
    plt.yticks([])
plt.tight_layout()
plt.show()

```



1.4 Improving Image Quality

```

[10]: import cv2
import numpy as np
import matplotlib.pyplot as plt

```

```

[11]: img = cv2.imread('Input_Images/xray.jpeg')
hsv_img = cv2.cvtColor(img, cv2.COLOR_BGR2HSV)
h, s, v = hsv_img[:, :, 0], hsv_img[:, :, 1], hsv_img[:, :, 2]

```

```

[12]: clahe = cv2.createCLAHE(clipLimit = 15.0, tileGridSize = (20,20))
v = clahe.apply(v)

```

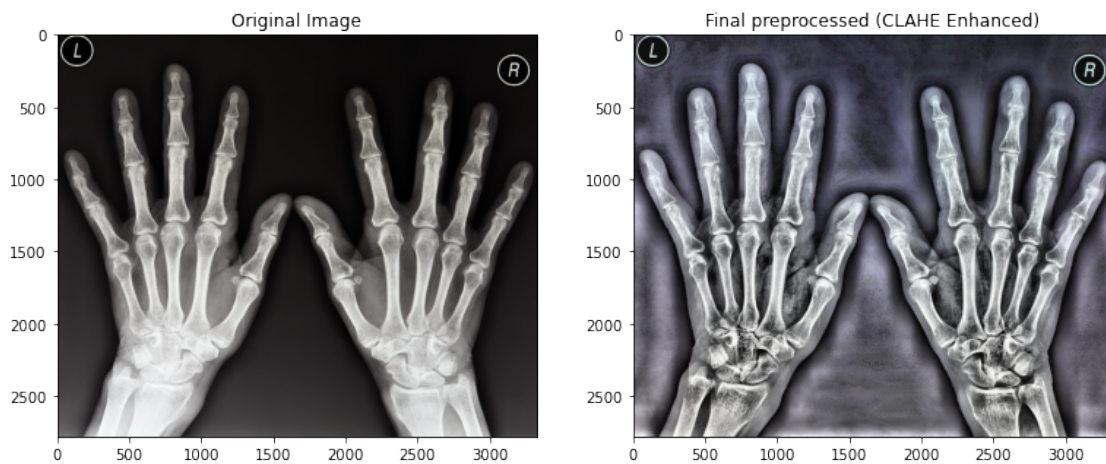
```
hsv_img = np.dstack((h,s,v))

rgb = cv2.cvtColor(hsv_img, cv2.COLOR_HSV2RGB)
#plt.imshow(rgb);
```

```
[13]: from matplotlib import pyplot as plt
fig = plt.figure(figsize=(20,15))
plt.subplot(2,3,1)
plt.imshow(img)
plt.title('Original Image')

plt.subplot(2,3,2)
plt.imshow(rgb)
plt.title('Final preprocessed (CLAHE Enhanced)')
```

```
[13]: Text(0.5, 1.0, 'Final preprocessed (CLAHE Enhanced)')
```



```
[ ]:
```