inverting image in Python with OpenCV

I want to load a color image, convert it to grayscale, and then invert the data in the file.

What I need: to iterate over the array in OpenCV and change every single value with this formula (it might be wrong but it seems reasonable for me):

```
img[x,y] = abs(img[x,y] - 255)
```

but I don't understand why doesn't it works:

```
def inverte(imagem, name):
    imagem = abs(imagem - 255)
    cv2.imwrite(name, imagem)


def inverte2(imagem, name):
    for x in np.nditer(imagem, op_flags=['readwrite']):
        x = abs(x - 255)
    cv2.imwrite(name, imagem)


if __name__ == '__main__':
    nome = str(sys.argv[1])
    image = cv2.imread(nome)
```

I don't want to do an explicit loop (I am trying to be more pythonic). I can see that in one image that got a white background it turned black, but only this it doesn't looks like the other colors are having much (if any) change.

python arrays opencv numpy image-processing

edited May 11 '16 at 20:02

Saullo G. P. Castro
34.9k 12 115 173

asked Oct 25 '13 at 2:25

Mansueli
2,848 4 20 46

2 Answers

You almost did it. You were tricked by the fact that `abs(imagem-255)` will give a wrong result since your `dtype` is an unsigned integer. You have to do `(255-imagem)` in order to keep the integers unsigned:

```
def inverte(imagem, name):
    imagem = (255-imagem)
    cv2.imwrite(name, imagem)
```

As recommended by some comments here, you could invert the image using the bitwise_not function of OpenCV:

```
imagem = cv2.bitwise_not(imagem)
```

edited May 22 '17 at 12:58

34.9k 12 115 173

2 Thanks you! I didn't know that in python the standard would be unsigned int ... – Mansueli Oct 31 '13 at 20:36

3 Neat, didn't think it would be this simple, thanks. – Nit Mar 22 '15 at 12:09

Alternatively, you could invert the image using the `bitwise_not` function of OpenCV:

```
imagem = cv2.bitwise_not(imagem)
```

I liked the example at: https://www.learnopencv.com/filling-holes-in-an-image-using-opencv-python-c/

answered Dec 4 '16 at 0:36

Eric Olmon
291 3 5