

Visão Computacional: reconhecimento de texto com OCR e OpenCV

 cursos.alura.com.br/course/visao-computacional-reconhecimento-texto-ocr-opencv/task/112860

Agora que já sabemos o que é o OCR, podemos começar a trabalhar com ele dentro do Colab. Nosso primeiro passo será a instalação das bibliotecas. Nós instalaremos o **pytesseract** e também utilizaremos o **opencv** para visualizarmos algumas imagens e realizarmos alguns tratamentos nelas. O **Tesseract OCR** também será instalado.

```
!pip install opencv-python==4.6.0
!sudo apt install tesseract-ocr
!pip install pytesseract==0.3.9
```

Dois dos comandos dessas três bibliotecas que estamos instalando são `pip install`. O segundo, é um comando que vem do Linux, `sudo apt`, que usamos para a instalação do `tesseract`. Depois de rodar, ele enviará uma mensagem bastante grande. Ao final, um aviso, "you must restart the runtime in order to use newly installed versions", isto é, "você precisa restartar o seu runtime", assim, ele conseguirá entender o que está acontecendo.

Então, vamos apertar o botão "Restart Runtime". Fazendo isso, receberemos um *warning* com a mensagem "Você quer mesmo reiniciar o ambiente de execução? O estado dele e de todas as variáveis locais será perdido". Nós não temos nada rodado ainda, então, podemos apertar "sim". Após o reiniciou, verificaremos que o valor de "RAM" e "Disco rígido" voltaram ao normal.

Todos os dados que usaremos no curso serão importados via **git clone**, do GitHub. Dentro da pasta "text-recognize", disponível neste link, encontraremos outra pasta, de "imagens". Nela, encontraremos o Projeto e todas as imagens por aula. Na próxima célula, chamaremos `! git clone` seguido do link.

```
! git clone https://github.com/sthemonica/text-recognize
```

Na lateral superior do Colab, vamos selecionar os arquivos e a pasta "text_recognize" já estará aqui, com todas as nossas imagens. É muito mais simples do que importarmos todas as imagens de uma vez só ou adicionarmos uma a uma.

Agora, importaremos todas as bibliotecas que usaremos: `numpy as np`; `cv2`, que é a biblioteca `openCV`; e `cv2_imshow`, que mostrará as imagens no Colab.

```
import pytesseract
import numpy as np
import cv2
from google.colab.patches import cv2_imshow
```

Dentro do `openCV` , temos o `cv2.imshow` . Esse comando não funciona no Colab para mostrarmos as imagens. Quando estamos usando o Anaconda ou o Jupyter, ele funcionará corretamente, mas dentro do Google Colab, não, então precisaremos dos `patches` para evitarmos um erro.

Rodamos a célula e deu tudo certo. Agora podemos conferir se ele instalou corretamente a versão do `pytesseract` .

```
pytesseract.__version__
```

```
| 0.3.9
```

Também vamos conferir a versão do `OpenCV` .

```
cv2.__version__
```

```
| 4.6.0
```

A versão do `OpenCV` é a 4.6.0 e a do `pytesseract` é a 0.3.9, exatamente as versões que pedimos anteriormente no código. Nosso ambiente está configurado, então, podemos importar uma imagem. Para isso, utilizaremos o `OpenCV` , que fará a leitura. Faremos `img` igual a `cv2.imread()` .

Para importarmos a primeira imagem, na lateral esquerda da tela, acessaremos a pasta "text_recognize > Imagens", que importamos do GitHub, e localizaremos o arquivo "Aula1-teste.png". Em seguida, apertaremos o botão de "três pontinhos (...)" que está à frente do arquivo e selecionaremos a opção "Copiar caminho" e colaremos na célula, entre aspas simples ou duplas.

Na linha abaixo, faremos `cv2.imshow()` , que é o comando que importamos com o *google patches* e passaremos `img` .

```
img = cv2.imread('/content/text-recognize/Imagens/Aula1-teste.png')  
cv2.imshow(img)
```

Com isso, aparecerá a nossa primeira imagem dentro do Google Colab, portanto, o `OpenCV` está funcionando.

TESTE INICIAL OCR

A primeira parte está feita. Agora, precisamos conferir se o `tesseract` está funcionando. Vamos tentar retirar o texto a partir da imagem. Vamos chamar uma variável nova, `texto` e ela será igual a `pytesseract.image_to_string(img)`. Abaixo, faremos `print(texto)`.

```
texto = pytesseract.image_to_string(img)
print(texto)
```

| TESTE INICIAL OCR

A resposta foi "TESTE INICAL OCR", portanto, ele já começou a ler a imagem, sem que fosse necessário fazer qualquer tratamento. O próximo passo é testar a reação do `tesseract` com outras imagens.