Visão Computacional: reconhecimento de texto com OCR e OpenCV

cursos.alura.com.br/course/visao-computacional-reconhecimento-texto-ocr-opencv/task/112862

Até aqui, lidamos com imagens coloridas e em preto e branco, mas apenas com frases curtas. O primeiro texto era "TESTE INICIAL OCR" e o segundo, "Tesseract OCR" com a logo do Google. O que acontece no caso de textos mais complexos?

Nós vamos reutilizar os códigos das aulas passadas, começando pelo da imagem, `.imread`.

```
img = cv2.imread('/content/text-recognize/Imagens/Aula1-teste.png')
cv2_imshow(img)
```

O `content` será diferente. Na pasta "text-recognize > Imagens", nós localizaremos a "Aula2-undersampling.png", copiaremos esse caminho e substituiremos pelo caminho antigo, dentro das aspas.

```
img = cv2.imread('/content/text-recognize/Imagens/Aula2-undersampling.png')
cv2_imshow(img)
```

Ao fazermos isso, ele mostra uma imagem abaixo, onde está escrito, em azul e negrito, "Undersampling". Na linha abaixo, em letras menores, "É uma técnica que consiste em manter todos os dados da classe com menor frequência e diminuir a quantidade dos que estão na classe de maior frequência, fazendo com que as observações no conjunto possuam dados com a variável alvo equilibrada."

Não queremos que a imagem apareça como está, mas de forma direta, com `rgb`, por isso, substituiremos a segunda linha de código pelo código do `rgb`, realizando, desta maneira, a conversão para BGR.

```
img = cv2.imread('/content/text-recognize/Imagens/Aula2-undersampling.png')
rgb = cv2.cvtColor(img, cv2.COLOR_BGR2RGB)
cv2_imshow(rgb)
```

Antes, as letras da imagem eram azuis e agora, com a conversão de cores, o texto continua o mesmo, mas as letras estão amareladas. Nosso próximo passo é retirar/capturar o texto.

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

> Undersampling
>
> E uma técnica que consiste em manter todos os dados da classe com menor frequéncia e diminuir a quantidade dos que estdo na classe de maior frequéncia, fazendo com que as observagées no conjunto possuam dados com a variavel alvo equilibrada.

Sobre a tradução, o título está correto. No parágrafo, ele não traduziu o acento agudo em "É", mas traduziu em "técnica". Já a palavra "frequência" deveria conter acento circunflexo, mas foi traduzida com acento agudo, "frequéncia". Depois, o trecho "diminuir a quantidade dos que estão na classe", ele traduziu por "diminuir a quantidade dos que estdo na classe", portanto, faltou o "a" em "estão".

Continuando, em "que estão na classe de maior frequência", outra vez "frequência" foi traduzida com acento agudo, "frequéncia". O parágrafo segue com "fazendo com que as observações" que foi traduzido como "fazendo com que as observagées". O final do parágrafo está traduzido corretamente, para além da palavra "variável", que está sem o acento na letra "a".

As palavras **frequência**, **estão** e **observações** não conseguiram ser traduzidas de uma forma boa o suficiente. Por que isso aconteceu, considerando que as demais palavras estão corretas?

Na primeira aula, estudamos que o Tesseract OCR faz reconhecimentos por padrão, por exemplo, reconhece que determinada letra se parece com outra que ele já conhece e vai fazendo substituições. Se algumas palavras foram traduzidas de maneira incorreta, significa que está ocorrendo algum problema de interpretação das letras.

Vamos conferir com quais idiomas ele está trabalhando, porque, se não está reconhecendo acentos da língua portuguesa, talvez esteja utilizando outra língua. Então, com o `!tesseract --list-langs` visualizaremos as línguas que estão no Tesseract.

```
!tesseract --list-langs
```

> List of available languages (3): osd eng por

Ele retornou uma lista de linguagens que estão disponíveis: osd; e eng, que é o inglês. O português não está na lista, portanto, precisamos instalá-lo, assim, o Tesseract entenderá o texto e fará uma tradução melhor. O comando para essa instalação, será `!apt-get`, espaço, `install tesseract-ocr-por`, sendo que `por` se refere ao português.

```
!apt-get install tesseract-ocr-por
```

> Reading package lists... Done Building dependency treeReading state information... Done tesseract-ocr-por is already the newest version (4.00~git24-0e00fe6-1.2). The following package was automatically installed and is no longer >required: libnvidia-common-460 Use 'apt autoremove' to remove it. 0 upgraded, 0 newly installed, 0 to remove and 19 not upgraded.

Agora, podemos rodar a lista de linguagens outra vez e verificar se ele de fato instalou o português.

```
!tesseract --list-langs
```

> List of available languages (3): osd eng por

Temos 3 linguagens e uma delas é o português. Como o português instalado, podemos fazer a leitura da frase que está na imagem outra vez. Para isso, basta copiar o mesmo código de `texto`, passando também o parâmetro `lang` definido como `por`, de "português". Sem isso, o Tesseract definirá o inglês como língua principal e não entenderá o que estamos pedindo que ele leia.

```
texto = pytesseract.image_to_string(rgb, lang='por')
print(texto)
```

> Undersampling

> É uma técnica que consiste em manter todos os dados da classe com menor frequência e diminuir a quantidade dos que estão na classe de maior frequência, fazendo com que as observações no conjunto possuam dados com a variável alvo equilibrada.

Desta vez, todas as palavras foram acentuadas de forma certa, inclusive "frequência", "estão" e "observações". Com essa pequena mudança, dentro de um parâmetro, já conseguimos um bom resultado com o OCR. A próxima etapa é trabalhar com elementos diferentes dentro do OCR para alcançarmos resultados ainda melhores. Vamos lá?!