Visão Computacional: reconhecimento de texto com OCR e OpenCV

cursos.alura.com.br/course/visao-computacional-reconhecimento-texto-ocr-opencv/task/112863

Além de definirmos as linguagens do projeto, os idiomas que vamos utilizar, outra parte importante, é de **PSM** ou **Page Segmentation Mode** que, em tradução livre, significa Modo de Segmentação da Página.

O OCR, em geral, espera receber uma imagem de formato específico, por exemplo, de um PDF, quer dizer, uma imagem "normal". Porém, nem sempre trabalharemos com imagens padrão, às vezes os textos são de formatos diferentes, a criatividade extrapola o que imaginamos e o OCR precisa se adaptar.

Para isso, utilizamos o PSM. Ele contém alguns modos que podem ser usados para a segmentação. Para visualizarmos esses modos, utilizaremos o `tesseract` e o comando `help-psm`.

```
!tesseract --help-psm
```

> !tesseract --help-psm

Page segmentation modes:

0	Orientation and script detection (OSD) only.
1	Automatic page segmentation with OSD.
2	Automatic page segmentation, but no OSD, or OCR.
3	Fully automatic page segmentation, but no OSD. (Default)
4	Assume a single column of text of variable sizes.
5	Assume a single uniform block of vertically aligned text.
6	Assume a single uniform block of text.
7	Treat the image as a single text line.
8	Treat the image as a single word.
9	Treat the image as a single word in a circle.
10	Treat the image as a single character.
11	Sparse text. Find as much text as possible in no particular order.
12	Sparse text with OSD.

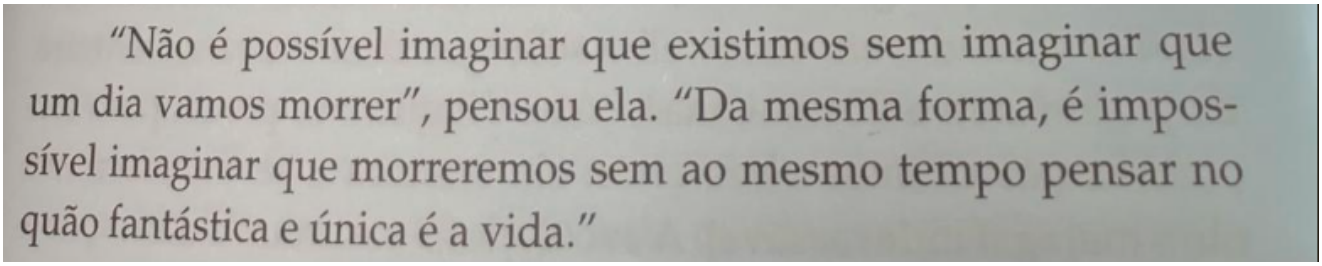
	Page segmentation modes:
13	Raw line. Treat the image as a single text line,
	bypassing hacks that are Tesseract-specific.

Ele vai gerar uma lista de "Page segmentation modes", que são os 14 modos com os quais podemos fazer segmentação de página pelo PSM. Nós temos, "Orientation and script detection (OSD)", uma linguagem que já estudamos anteriormente e fornece informações sobre a imagem.

Também temos "Assume a single uniform block of vertically aligned text", isto é, assume que ele tem uma forma uniforme de bloco com texto verticalmente alinhado. Além disso, "Treat the image as a single text line" e "Treat the image as a single word", ou seja, tratamos a imagem como uma linha de texto única e com uma única palavra dentro de um círculo.

Enfim, são vários tipos e modos disponíveis e nós precisamos encontrar qual funciona melhor. Em geral, o PSM se ajusta bem e nós entenderemos como isso acontece. Desta vez, chamaremos uma imagem diferente, usando o mesmo código de chamada de imagens, `img = cv2.imread` . Ainda neste código, ao invés do `undersampling` , vamos chamar "Aula2-trehco-livro.png". Para isso, basta copiar o caminho e substituir no código.

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



Existe um hífen na palavra "impossível" que a divide. Podemos conferir como esse bloco de texto está reagindo sem o PSM.

```
config_tesseract = '--tessdata-dir tessdata'
texto = pytesseract.image_to_string(rgb, lang='por', config=config_tesseract)
print(texto)
```

> "Não é possível imaginar que existimos sem imaginar que um dia vamos morrer", pensou ela. "Da mesma forma, é impos- sível imaginar que morreremos sem ao mesmo tempo pensar no quão fantástica e única é a vida."

Aparentemente, ele já encarou a imagem como um bloco de texto e fez a tradução corretamente. Mas, se tentarmos encaixar esse texto dentro de um PSM, acredito que seria o "6 Assume a single uniform block of text" ou seja, o tópico 6 assume que a imagem é um bloco de texto uniforme.

Vamos utilizar o trecho de código anterior e, no `config_tesseract`, depois do `tessdata`, daremos espaço e escreveremos `--psm 6` para verificarmos o que ele vai retornar.

```
config_tesseract = '--tessdata-dir tessdata --psm 6'
texto = pytesseract.image_to_string(rgb, lang='por', config=config_tesseract)
print(texto)
```

> “Não é possível imaginar que existimos sem imaginar que um dia vamos morrer”, pensou ela. “Da mesma forma, é impos- sível imaginar que morreremos sem ao mesmo tempo pensar no quão fantástica e única é a vida.”

Deu certo também! Ele está reconhecendo corretamente! Mas, o que aconteceria se ele não reconhecesse? Isto é, se o Tesseract não conseguisse identificar que esse era um PSM? Podemos fazer um teste exatamente com a mesma imagem. Para isso, copiaremos a configuração anterior, mas com `--psm 7`. O PSM 7 trata a imagem como se ela fosse uma única linha de texto.

```
config_tesseract = '--tessdata-dir tessdata --psm 7'
texto = pytesseract.image_to_string(rgb, lang='por', config=config_tesseract)
print(texto)
```

Ele não retornou nada. Quando isso acontece, as pessoas costumam ficar preocupadas, pensando que o erro é muito mais complexo, mas, ele apenas não conseguiu entender o que estava na imagem. Outro erro que podemos testar é com o PSM 8, que trata a imagem como se fosse uma única palavra, considerando que a nossa imagem é de um trecho de livro, portanto, composta por várias palavras.

```
config_tesseract = '--tessdata-dir tessdata --psm 8'
texto = pytesseract.image_to_string(rgb, lang='por', config=config_tesseract)
print(texto)
```

> ªfâ%êããããââª%%?ªííÉ:D;Z:Én:gíríãrgâgíie::ugfâ|

O resultado são várias letras sem sentido. É um erro que assusta bastante até descobrirmos o que aconteceu. Esses dois resultados errados demonstram o que acontece quando o Tesseract não consegue reconhecer qual é o PSM. Neste caso, nós indicamos o PSM errado de propósito, mas esse erro pode ocorrer de forma natural.

Vamos adicionar uma imagem diferente com o PSM certo para conferirmos o que acontece. A imagem será a "Aula2-Saida.png".

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

Vamos rodá-la. Em seguida, copiaremos o `tessdata` do PSM 7, tratando-a como uma única linha.

```
config_tesseract = '--tessdata-dir tessdata --psm
7'
texto = pytesseract.image_to_string(rgb,
lang='por', config=config_tesseract)
print(texto)
```

> SAÍDAS

Ele leu a seta como "S", mas nós não realizamos todo o tratamento necessário na imagem. Inclusive, ela está com cores complicadas de se entender. Mesmo assim, a leitura foi boa, a acurácia está bem melhor que não entregar nada. O PSM é uma ferramenta que merece atenção. Caso ele não esteja retornando o esperado, precisamos entender se estamos lidando com um bloco de texto, uma linha ou uma palavra na vertical.

Na próxima aula, estudaremos novos tratamentos e a como melhorar nossas saídas dentro do Tesseract. Até já!!