

# Visão Computacional: reconhecimento de texto com OCR e OpenCV

 [cursos.alura.com.br/course/visao-computacional-reconhecimento-texto-ocr-opencv/task/112859](https://cursos.alura.com.br/course/visao-computacional-reconhecimento-texto-ocr-opencv/task/112859)

Nosso mundo, atualmente, é todo digitalizado. Normalmente, as nossas contas chegam por e-mail ou aplicativos de celular, mas, algumas empresas ainda enviam faturas de papel via Correios e outras exigem que as pessoas levem documentos físicos com cópias para comprovarem que realmente são elas. Neste caso, surge uma questão: onde armazenar esses papéis?

Por exemplo, as empresas de advocacia e afins recebem contratos, faturamentos e outros documentos físicos e precisam armazená-los. O problema principal, portanto, é o **armazenamento**.

Além disso, em algum momento, as empresas desejarão consultar esse material, seja porque tiveram dúvida em algum ponto ou porque o cliente entrou em contato e pediu alguma informação sobre o contrato, ou por qualquer outro motivo. Então, um segundo problema seria a **procura pelos dados armazenados**.

Se o documento tem cinco folhas, fica fácil encontrar, mas, imagine o caso de vários documentos, cada um com cinco folhas. Seria bem mais complicado. Para esses casos, podemos aplicar o OCR, que é o **Optical Character Recognition** ou, em português, o Reconhecimento Ótico de Caracteres. Mas, como aplicaremos o OCR se tudo está no papel?

Primeiro, vamos entender o que é o OCR. Como já dissemos, ele é o Reconhecimento de Caracteres Óticos e converte uma imagem de texto em um formato de texto legível para a máquina. Então, para que ele funcione, teremos que transformar todos os papéis em um formato de entrada para a máquina.

A primeira parte é a **aquisição da imagem**. Podemos escanear todas as imagens, tirar fotografias com o celular ou qualquer outra câmera e essa será a etapa de entrada. Depois disso, faremos o **pré-processamento**. Nesta etapa, trataremos a imagem, nos atentando a alguns detalhes, por exemplo, se a imagem está um pouco rotacionada ou se é necessário ajustar a iluminação, dentre outros.

Em seguida, passamos para o reconhecimento do texto em si. Este reconhecimento é composto por duas etapas distintas, isto é, o reconhecimento pode ser **por padrões**, quando a ferramenta já viu tantas letras e sabe, por exemplo, o que é uma letra "a" minúscula ou uma letra "A" maiúscula.

O reconhecimento também pode ser **detecção de recursos**. Neste caso, ele sabe, por exemplo, que quando escrevemos a letra "A" maiúscula, criamos uma intersecção de linhas. Assim, é possível prever que a letra provavelmente será a "A". Desta forma, vai encontrando pontos correlacionados nas letras para reconhecê-las.

O próximo passo é de **pós-processamento**, que é como entregará a informação. Existem diversas ferramentas que usam o método OCR, mas, desta vez, escolheremos a **Pytesseract**, uma biblioteca do Python que utiliza o **Tesseract OCR**, do Google.

O Tesseract OCR é um recurso *open source* da Google e funciona com a OCR engine. O OCR é um método que utilizamos e podemos treinar. Nós optamos pela Tesseract por ser *open source*. Em seu repositório do GitHub, conseguimos visualizar todas as línguas que foram treinadas e também podemos treiná-la com outro idioma.

É interessante entendermos como funciona esse processo de aprendizagem e também como funciona o OCR, mas, agora estudaremos especificamente o Tesseract OCR. Vamos lá?