

## Desafio Técnico PDI

### 1. Descrição

Para este projeto serão utilizados algoritmos de processamento de imagens com objetivo de destacar a região de interesse de um encarte de supermercado (nome e preço do produto), que serão posteriormente extraídos pelo time de NLP.

### 2. Arquivos

Os arquivos desenvolvidos durante a resolução do projeto são listados a seguir:

- **pre\_processamento\_imagens.py**: Arquivo principal, conteúdo dos pré-processamento utilizando técnicas clássicas de processamento digital de imagens para extrair as regiões de interesse das imagens.
- **./notebooks/pre\_processamento\_imagens.ipynb**: O arquivo em questão é uma cópia do arquivo anterior no formato de notebook para facilitar a visualização dos resultados.
- **./notebooks/treinamento\_CNN\_OCR.ipynb**: Criação do modelo de rede neural convolucional para reconhecimento de caracteres.
- **./notebooks/reconhecimento\_numeros\_cnn.ipynb**: Teste para reconhecimento dos preços dos produtos por meio de uma rede neural convolucional.
- **./notebooks/reconhecimento\_caracteres\_pytesseract.ipynb**: Teste para reconhecimento dos nomes dos produtos por meio do pytesseract.
- **./notebooks/imagens**: Imagens utilizadas e resultantes.
- **./notebooks/modelo**: modelo treinado

### 3. Técnicas utilizadas

Visando encontrar o melhor resultado, realizou-se diversos experimentos e aplicações de diversas técnicas. Por fim, manteve-se as técnicas que apresentaram os melhores resultados na etapa de extração de caracteres.

Ademais, as imagens utilizadas para extração de caracteres com Pytesseract e CNN foram submetidas a técnicas diferentes, uma vez que, cada técnica se adequou melhor a um tipo de imagem.

As técnicas utilizadas são descritas a seguir:

- **Destaque da região de interesse**: Gerou-se as regiões de interesse (ROIs) da imagem (como nome e preço do produto) para realizar o pré-processamento e eliminar as informações não importantes.
- **Tons de cinza**: Conversão da imagem RGB para tons de cinza.
- **Filtro Bilateral**: Utilizado para realizar a suavização da imagem e remover possíveis ruídos, preservando os detalhes de bordas e contornos.
- **Binarização de Nobuyuki Otsu**: Realizada com objetivo de separar o objeto de interesse do fundo.
- **Deteção de Bordas de Canny**: Algoritmo utilizado para detectar as bordas presentes nas imagens.

- **Operação Morfológica Dilatação:** Utilizada para dilatar a área do objeto de interesse, ou seja, o objeto do primeiro plano ficará maior do que era inicialmente.
- **Operação Morfológica Erosão:** Utilizada para realizar a corrosão das arestas do objeto de interesse, resultando em uma imagem "encolhida" do objeto.
- **Rede Neural Convolucional:** Rede Neural treinada para reconhecer os dígitos da imagem.
- **Pytesseract:** Ferramenta de reconhecimento óptico de caracteres (OCR) para Python, que reconhece e retorna o texto embutido nas imagens.

#### 4. Resultados

Para a extração de caracteres utilizando Pytesseract destacou-se as regiões de interesse da imagem, converteu a imagem para tons de cinza, suavizou a imagem e por fim realizou-se a binarização de Otsu.



O Pytesseract conseguiu reconhecer bem os nomes dos produtos, entretanto, para os numerais, a técnica não apresentou resultados interessantes. Ademais, para melhorar o resultado, destacou-se apenas o texto desejado. O resultado do Pytesseract para o texto em destaque:

Texto Extraído:

Cerveja Puro  
Malte Petra 350ml

**Cerveja Puro  
Malte Petra 350ml**

Em seguida, para realizar a extração dos dígitos dos preços dos produtos texto realizou-se o treinamento de um modelo CNN. Foi necessário adaptar as imagens enviadas para rede neural, pois, a rede será treinada com a base de dados MNIST. Logo, é interessante que a imagem de teste possua um formato semelhante, ou seja, a área de interesse (numeral) branca e o fundo preto. Assim, a imagem foi transformada para escala de cinza, suavizada, detectou-se as bordas com o filtro de Canny e por fim, aplicou-se as operações morfológicas de dilatação e erosão, resultando na seguinte imagem:



Por fim, a rede CNN exibiu os seguintes resultados para reconhecimento dos dígitos:

Caractere Extraído: 2  
Probabilidade = 99.96829032897949



Caractere Extraído: 7  
Probabilidade = 92.56790280342102



Caractere Extraído: 9  
Probabilidade = 94.8762059211731

