

DESAFIO FINAL

Atlântico Academy AsCan

Computação Cognitiva

Caio Martim Barros

Atlântico



Objetivo *do Desafio*

Reconhecer os caracteres das placas de carros do dataset contido no Kaggle

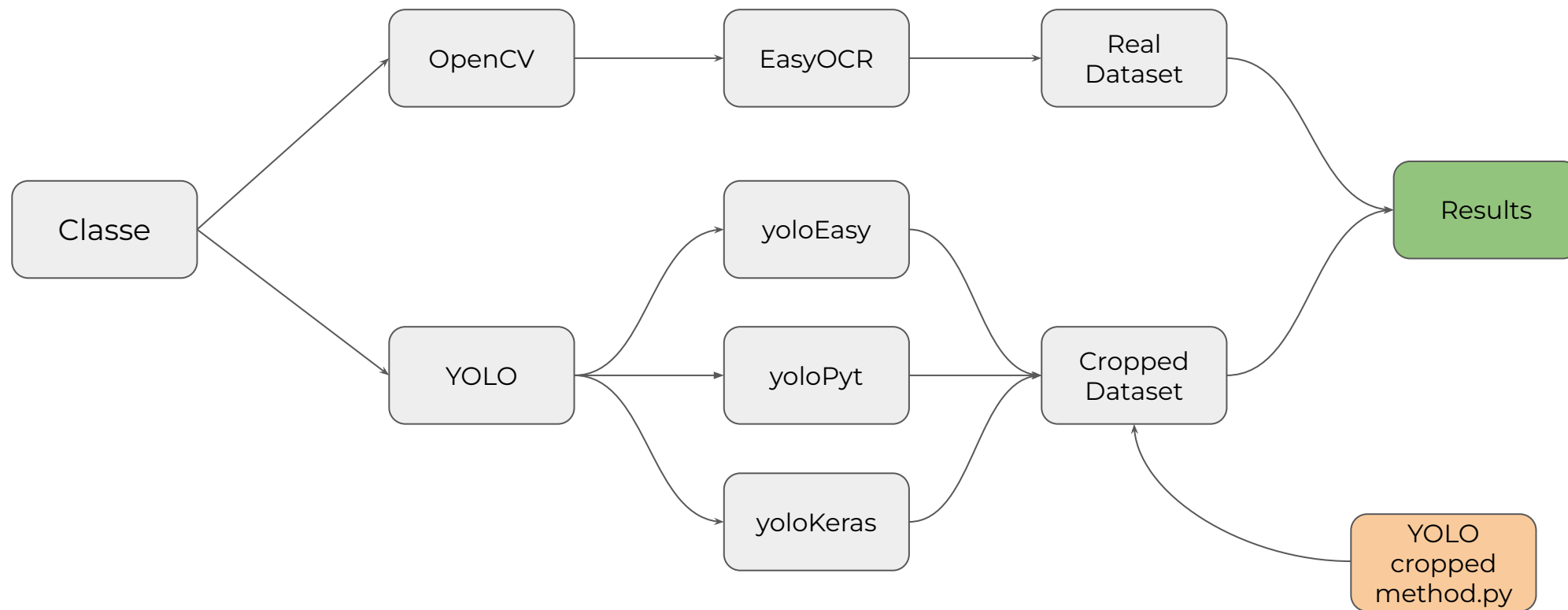




Tecnologia *utilizada*

- Google Colab - Python
 - OpenCV
 - YOLO
 - Algoritmos de OCR
 - EasyOCR
 - Pytesseract
 - KerasOCR

Descrição do Projeto



Simple Detection

(OpenCV)

gray



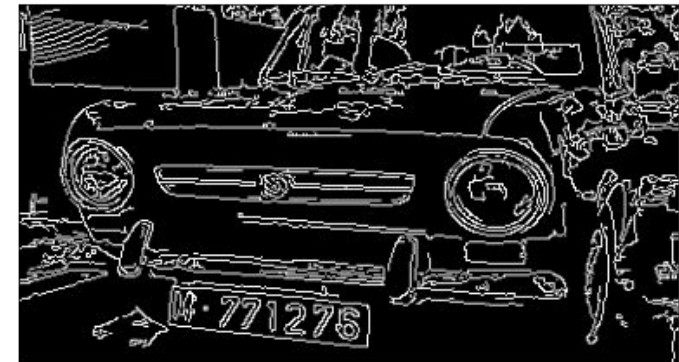
bfilter



bfilter



edged



original



cropped_image



YOLOv5

(You only look once)





Resultados | *obtidos*

Nesse espaço, descreva o resultado
obtido ao finalizar o desafio.

Processo



yolov5



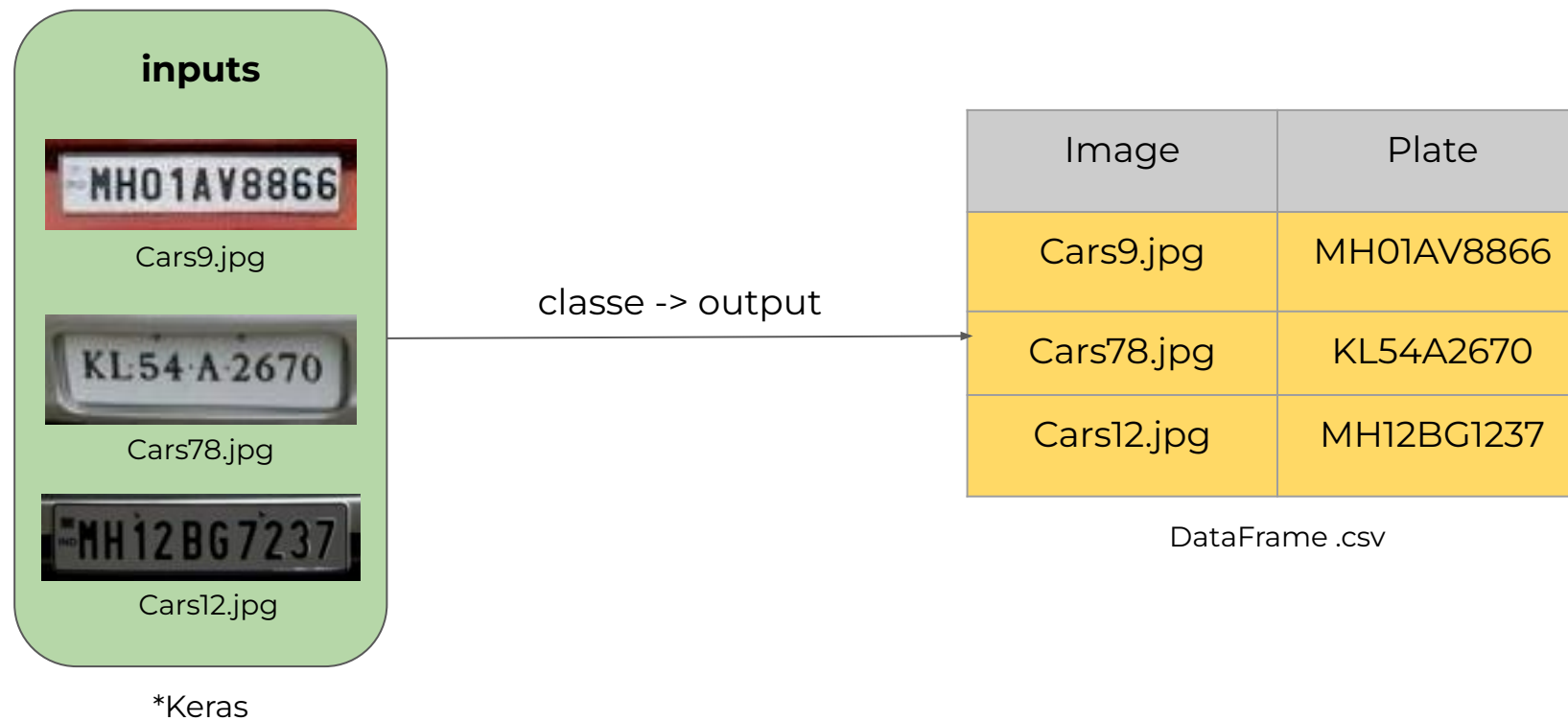
classe

mho1av8866



output 1

Processo



KerasOCR - 'Problema'



Image	Plate
Cars34.jpg	DLZC
Cars34.jpg	N
Cars34.jpg	5617

Cars34.jpg

DLZCN5617



Image	Plate
Cars299.jpg	784U
Cars299.jpg	SDN

784USDN

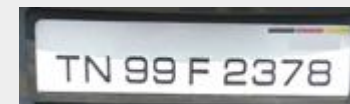
Comparação



Cars299.jpg



Cars418.jpg



Cars73.jpg



Cars393.jpg

EasyOCR

Image	Plate
Cars299.jpg	SDN7484U
Cars393.jpg	M:
Cars418.jpg	AND
Cars73.jpg	TN99F2378

Pytesseract

Image	Plate
Cars299.jpg	SDN7484U!
Cars393.jpg	M 771276)
Cars418.jpg	[OX65 AND
Cars73.jpg	TN SSF 2378

KerasOCR

Image	Plate
Cars299.jpg	7484USDN
Cars393.jpg	MA771275
Cars418.jpg	AWDOX855
Cars73.jpg	TN59F2878



Considerações *finais*

KerasOCR parece ser o mais promissor por conta da efetividade em reconhecer os caracteres corretamente.

Porém, Pytesseract pode ser mais efetivo com as configurações e preprocessamentos corretos.



Considerações *finais*

Pontos de Melhorias

- Configurar a parte dos resultados do dataset
- Realizar pre-processamento para melhorias nas imagens
- Verificar a existência de uma configuração melhor para o pytesseract (parâmetro)
- Adicionar métricas para avaliar o desempenho das OCR

Demonstração

Atlântico



Obrigado!

Atlântico

