

# Sistema de Reconhecimento de Placas de Automóveis Usando Algoritmo de Detecção YOLO

Lucas da Silva Pereira  
180125699

Departamento de Ciência da Computação  
Universidade de Brasília  
Brasília, Brasil  
pereira-silva.sp@aluno.unb.br

Felipe Costa de Sousa  
211055236

Departamento de Ciência da Computação  
Universidade de Brasília  
Brasília, Brasil  
211055236@aluno.unb.br

**Abstract**—This report presents our experience and results of the implementation of the article "An Efficient and Layout-Independent Automatic License Plate Recognition System Based on the YOLO Detector", for the Introduction to Artificial Intelligence course.

**Index Terms**—Reconhecimento de Placas de Automóveis, Redes Neurais, Inteligência Artificial, Detecção de Objetos

## I. INTRODUÇÃO

Este trabalho consiste em um estudo, aplicação, implementação e testes de um sistema de reconhecimento, classificação e identificação de veículos e placas construído com base no artigo "An Efficient and Layout-Independent Automatic License Plate Recognition System Based on the YOLO Detector". Esse artigo aborda a detecção de placas de carros baseado em Inteligência Artificial, Deep Learning, Regras de Pós Processamento e Redes Neurais. Essa pesquisa foi resultado de um trabalho de aprimoramento de outros trabalhos semelhantes a esse e que tem como diferencial a velocidade e a eficiência de seu modelo.

## II. SISTEMA DE RECONHECIMENTO DE PLACAS DE AUTOMÓVEIS

O Sistema de Reconhecimento de Placas de Automóveis foi basicamente dividido nas seguintes etapas:

- Detecção de Veículos
- Segmentação de Veículos
- Classificação de Veículos (Carro/Moto)
- Detecção de Placas de Automóveis
- Segmentação de Placas de Automóveis
- Classificação de Placas de Automóveis (País/Região)
- Reconhecimento dos Caracteres das Placas dos Automóveis

## III. ALGORITMO YOLO

YOLO - You Only Look Once - é um algoritmo de detecção de objetos em imagens e vídeos. O algoritmo aborda o problema de detecção de objetos como um problema de regressão para localizar objetos diretamente na imagem, em vez de abordar o problema por meio de propostas de regiões, como muitos métodos antigos faziam.

A principal ideia do YOLO é dividir a imagem em uma grade e prever caixas delimitadoras e probabilidades de classe para cada célula da grade. O processo de detecção é realizado em uma única passagem pela rede neural, o que o torna mais rápido em comparação com métodos baseados em regiões, como o R-CNN.

## IV. CONSTRUÇÃO

A construção desse relatório foi totalmente baseada na aplicação dos passos disponibilizados pelos autores do artigo em seu site.

No início do trabalho, o nosso grupo tentou realizar os testes localmente na nossa máquina pessoal, o que foi muito difícil pela complexidade de se baixar e configurar instrumentos como CMake, Darknet e CUDA.

Logo em seguida, passamos para a implementação no Google Colab, o que tornou a realização dos testes muito mais fácil. Nesse ambiente de simulação, a darknet pode ser instalada usando 5 linhas de código e de forma muito rápida devido à utilização remota de GPUs da Google. Isso facilitou a nossa experiência de forma que nada mais precisaria estar rodando localmente.

O que pode ser considerado um ponto negativo do Google Colab seria a limitação em relação ao armazenamento de dados. Quando a sessão de utilização do Colab é finalizada, todas aquelas informações armazenadas temporariamente são perdidas e é necessário realizar a instalação e configuração toda novamente.

## V. ANÁLISE DE RESULTADOS

A análise dos resultados obtidos nesse experimento é bem diversa e com fatos curiosos. Em geral, o algoritmo YOLOv2 utilizado nesse artigo já está bastante antigo, considerando que a versão 8 já existe. Logo, alguns erros são esperados e serão abordados nessa seção.

No que se refere a detecção dos carros na imagem bruta, foi possível concluir que as imagens tiradas com menos zoom não conseguiram reconhecer os veículos na pista. Isso foi bastante evidente ao testar a seguinte imagem:



Fig. 1. Figura com a presença de 1 carro não detectado

Portanto, testamos imagens com um zoom maior ou tiradas de uma região mais próxima e conseguimos obter um resultado mais satisfatório (precisão média de 0.90, considerada uma eficiência excelente).

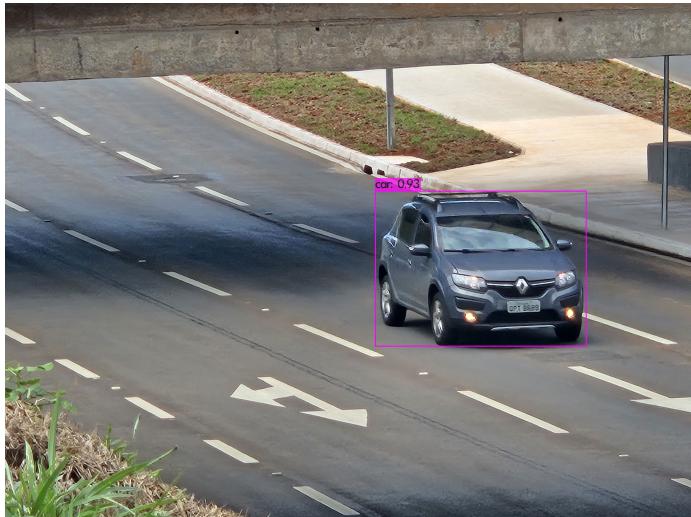


Fig. 2. PRECISÃO MEDIDA 0.93



Fig. 3. PRECISAO MEDIDA 0.89

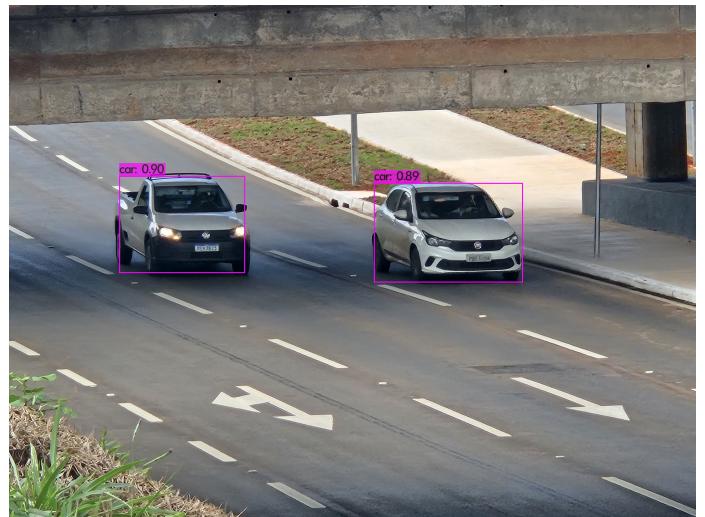


Fig. 4. PRECISÃO MEDIDA 0.89 e 0.90

Seguindo, na implementação dos passos do artigo, passamos para a parte de reconhecer a placa na imagem. Esse passo teve uma eficiência boa (média de 0.80), quase sempre reconhecendo a placa dos carros como sendo apenas brasileiras.



Fig. 5. PRECISÃO MEDIDA 0.84



Fig. 6. PRECISÃO MEDIDA 0.75

Por fim, a parte mais importante, a identificação dos caracteres da placa dos carros e em seguida a classificação desses caracteres. Essa parte depende muito da qualidade da imagem, inclusive depois de vários zoom e cortes que precisam ser aplicados na figura. Em imagens de boa qualidade, a eficiência ficou em média de 0.90 e na de qualidade ruim na casa dos 0.65

Além disso, é possível que erros como a não identificação de uma letra ou número de uma placa aconteça com frequência.

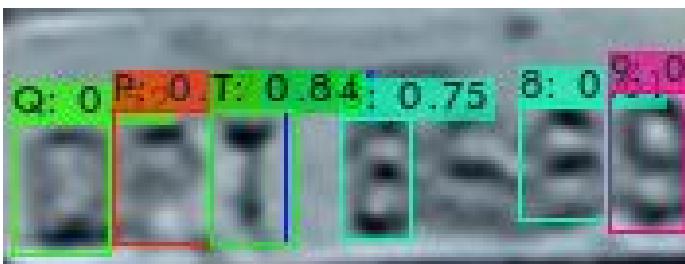


Fig. 7. Divisão de cada letra e número na placa de baixa qualidade

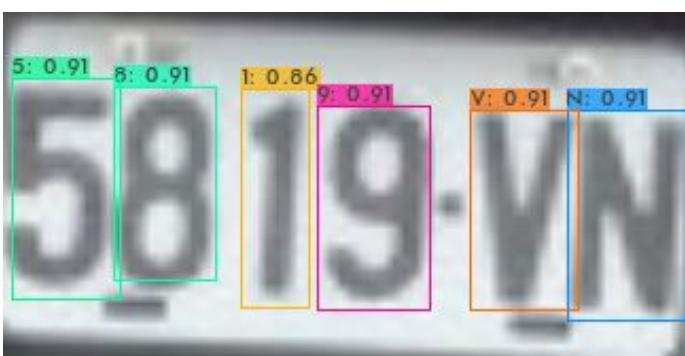


Fig. 8. Divisão de cada letra e número na placa de alta qualidade

## VI. ERROS

Durante a execução desse trabalho obtivemos diversos erros interessantes de serem relatados.

Primeiramente, quando tratamos de fotos em engarrafamentos, o algoritmo tem uma dificuldade muito grande de identificar os veículos, ou quando identifica, é com uma eficiência muita baixa (precisão média de 0.50).



Fig. 9. Nenhuma Moto Foi Identificada



Fig. 10. Borda da moldura "englobando" 2 motos



Fig. 11. Dois veículos identificados com baixa precisão

Outro erro interessante foi a identificação de uma placa de uma foto tirada pelo nosso grupo em que o sistema colocou a placa como sendo AMERICANA e BRASILEIRA. Porém, a probabilidade maior era da placa ser americana por isso a borda aparecendo é dos Estados Unidos e não do Brasil.



Fig. 12. Placa identificada como americana

## VII. CONCLUSÃO

Portanto, concluimos que o artigo analisado nesse trabalho possui excelentes resultados para fotos tirados em posições mais próximas, com poucos veículos na pista e a com boa qualidade de imagem.

Dessa forma, conseguimos adquirir um conhecimento muito válido e enriquecedor testando um projeto que se aproxima da vida real e como a Inteligência Artificial está sendo utilizada no mercado e no desenvolvimento de novas tecnologias.

## VIII. MATERIAIS

Aqui estão os materiais utilizados para a elaboração deste trabalho, dos testes, da apresentação e da confecção deste relatório.

- GitHub do projeto com os Notebooks Google Colab:  
<https://github.com/lucsap/ia-alpr-project>
- GitHub das imagens do projeto:  
[https://github.com/felipecostadesousa/imagens\\_trabalho\\_IA](https://github.com/felipecostadesousa/imagens_trabalho_IA)
- Imagens brutas:  
<https://drive.google.com/file/d/1AzLzxccZ1k62eJfNCEZimrXjXcmjdwtS/view?usp=sharing>

- Imagens após os testes do trabalho:  
<https://drive.google.com/file/d/1blcMXNdbuTYrY8Ij5tgNR7KMqZVZrhz0/view?usp=sharing>

## REFERENCES

- [1] Laroca R, Zanlorensi LA, Gonçalves GR, Todt E, Schwartz WR, Menotti D. *An efficient and layout-independent automatic license plate recognition system based on the YOLO detector* IETIntell Transp Syst. 2021;15:483–503, 2023, <https://doi.org/10.1049/itr2.12030>
- [2] Alexey Bochkovskiy and Chien-Yao Wang and Hong-Yuan Mark Liao. *YOLOv4: Optimal Speed and Accuracy of Object Detection* 2020, eprint=2004.10934,
- [3] Wang, Chien-Yao and Bochkovskiy, Alexey and Liao, Hong-Yuan Mark. *Scaled-YOLOv4: Scaling Cross Stage Partial Network* booktitle = Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), June, 2021, pages = 13029-13038
- [4] Redmon, J., Farhadi, A *YOLOv3: An incremental improvement* arXivpreprint 2018 Available from:<http://arxiv.org/abs/1804.02767>