

INSTITUTO FEDERAL DE SANTA CATARINA

LUCAS RANIERE GONCALVES

O PAPEL DOS ALGORITMOS DE VISÃO COMPUTACIONAL DENTRO DO  
SISTEMA DE SENSORIAMENTO DE VEÍCULOS AUTÔNOMOS

JOINVILLE  
DEZEMBRO/2022

LUCAS RANIERE GONÇALVES

O PAPEL DOS ALGORITMOS DE VISÃO COMPUTACIONAL DENTRO DO  
SISTEMA DE SENSORIAMENTO DE VEÍCULOS AUTÔNOMOS

Monografia apresentada ao  
curso de Engenharia  
Elétrica Campus Joinville do  
Instituto Federal de Santa  
Catarina para a obtenção do  
diploma de Engenheiro  
Eletricista

Orientador: Msc. Stefano  
Romeu Zeplin

Joinville  
Dezembro 2022

Gonçalves, Lucas Raniere.

O papel dos algoritmos de visão computacional dentro do sistema de sensoriamento de veículos autônomos / Lucas Raniere Gonçalves – Joinville, SC, 2022.

66 p.

Trabalho de Conclusão de Curso (Graduação) - Instituto Federal de Educação Ciência e Tecnologia de Santa Catarina, Curso de Bacharelado em Engenharia Elétrica, Joinville, 2022.

Orientador: Stefano Romeu Zeplin.

1. Detecção de objetos. 2. Redes neurais convolucionais. 3. YOLO. I. Instituto Federal de Educação Ciência e Tecnologia de Santa Catarina. II. Título.

LUCAS RANIERE GONCALVES

O PAPEL DOS ALGORITMOS DE VISÃO COMPUTACIONAL DENTRO DO  
SISTEMA DE SENSORIAMENTO DE VEÍCULOS AUTÔNOMOS

Este trabalho foi julgado adequado para obtenção do título em Engenheiro  
Eletricista, pelo Instituto Federal de Educação, Ciência e Tecnologia de Santa  
Catarina Campus Joinville, e aprovado na sua forma final pela comissão avaliadora  
abaixo indicada.

Joinville, 14 de dezembro de 2022.




Prof. Stefano Romeu Zeplin, Me. Eng.  
Orientador

Instituto Federal de Santa Catarina Campus Joinville



Prof. Neury Boaretto, Dr. Eng.  
Instituto Federal de Santa Catarina Campus Joinville



Prof. Rodrigo Coral, Dr. Eng.  
Instituto Federal de Santa Catarina Campus Joinville

## **AGRADECIMENTOS**

Agradecer primeiramente ao senhor Deus pelo dom da vida e capacidade intelectual.

Aos meus professores que ao longo do curso compartilharam seus conhecimentos nas mais variadas áreas e permitiram que chegasse este momento, em especial a meu orientador Stefano que ministrou as disciplinas de meu maior interesse pessoal.

A minha esposa Tayana que sempre esteve ao meu lado, responsável por auxiliar a manter meu equilíbrio emocional em um momento desafiador.

Aos meus pais Rene e Mariza que desde a infância tem incentivado na busca pelo conhecimento.

Aos meus irmão Gabriel e Rafael que compartilharam momentos de aprendizado e descontração.

*O conhecimento é em si mesmo um poder.*

*(Francis Bacon)*

## RESUMO

O trânsito tem feito muitas vítimas ao longo da história. Além das mortes a incapacidade gerada por estes acidentes também é um grande problema, provocando diversos impactos diretos e indiretos na sociedade. Grande parte destes acidentes tem origem no comportamento humano, ou seja, são evitáveis. Diante deste cenário a indústria automotiva tem empenhado esforços em desenvolver sistemas auxiliares que promovem conforto e segurança ao condutor e passageiros. Estes sistemas têm se tornado cada vez mais sofisticados, com o objetivo de alcançar a plena autonomia veicular. A proposta do trabalho é realizar um estudo acerca do sensoriamento em veículos que os levem à autonomia, em particular será abordado o papel dos algoritmos de visão computacional. Para tal objetivo serão apresentados a classificação dos diferentes níveis de automação em veículos, um panorama dos sensores que fazem parte da estrutura tecnológica hoje empregada, enfatizando a visão computacional, mostrando onde e de que forma são utilizados algoritmos de visão. No desenvolvimento foi feito o estudo de um dos algoritmos mais modernos na área de visão computacional, sendo este para identificação de objetos em tempo real, conhecido como *YOLO*, acrônimo para *You Only Look Once*, o qual teve sua primeira versão apresentada em 2015 por Joseph Redmon e desde então tem recebido otimizações. Através da implementação do algoritmo pode-se demonstrar seu funcionamento bem como as particularidades de detecção e os impactos do ambiente, sendo que foi demonstrado uma efetividade de detecção dos objetos mesmo sob condições adversas do ambiente, como a baixa iluminação, baixo contraste entre o ambiente e o objeto a ser detectado, oclusão das detecções, bem como altas quantidades objetos alvo.

Palavras-Chave: Detecção de objetos. Redes neurais convolucionais. YOLO. Visão computacional. Veículos autônomos

## **ABSTRACT**

Traffic has claimed many victims throughout history. In addition to deaths, the incapacity generated by these accidents is also a major problem, causing several direct and indirect impacts on society. Most of these accidents have their origin in human behavior, that is, they are preventable. Given this scenario, the automotive industry has made efforts to develop auxiliary systems that promote comfort and safety to the driver and passengers. These systems have become increasingly sophisticated, with the objective of achieving full vehicular autonomy. The purpose of the work is to carry out a study about the sensing in vehicles that take them to autonomy, in particular the role of computer vision algorithms will be addressed. For this purpose, the classification of the different levels of automation in vehicles will be presented, an overview of the sensors that are part of the technological structure used today, emphasizing computer vision, showing where and in what way vision algorithms are used. In the development, a study was made of one of the most modern algorithms in the area of computer vision, which is for identifying objects in real time, known as YOLO, acronym for You Only Look Once, which had its first version presented in 2015 by Joseph Redmon and has since received optimizations. Through the implementation of the algorithm, its operation can be demonstrated, as well as the detection particularities and the impacts of the environment, and an effectiveness of object detection was demonstrated even under adverse environmental conditions, such as low lighting, low contrast between the environment and the object to be detected, occlusion of detections, as well as high amounts of target objects.

Keywords: Autonomous vehicles. Computer vision. Convolutional neural networks. Machine learning.. Object detection. YOLO.





## LISTA DE ILUSTRAÇÕES

|   |    |
|---|----|
| Figura 1: Sensoriamento em veículos .....   | 20 |
| Figura 2: Níveis de automação em veículos .....   | 21 |
| Figura 3: Sistema de controle de velocidade de cruzeiro .....                             | 23 |
| Figura 4: Princípio de medição do sistema TOF (time-of-flight) .....                      | 24 |
| Figura 5: Sensor LiDAR HDL-64E (esq.) Montagem em um Toyota Priuse (dir.) .....           | 25 |
| Figura 6: Mapeamento gerado pelo LiDAR.....   | 25 |
| Figura 7: Sistema de câmeras carro Tesla.....   | 26 |
| Figura 8: Conjunto de câmeras Tesla Vision.....   | 27 |
| Figura 9: Exemplo processo de aquisição imagem.....                                       | 28 |
| Figura 10: Quantização de imagem digitalizada em escala de cinza .....                    | 29 |
| Figura 11: Aplicação do Filtro Sobel Y .....  | 30 |
| Figura 12: Comparativo entre machine learning convencional e deep learning .....          | 32 |
| Figura 13: Sistema de detecção YOLO .....   | 33 |
| Figura 14: Etapas camada convolucional .....  | 34 |
| Figura 15: Aplicação de um kernel de bordas .....   | 34 |
| Figura 16: Aplicação função Relu .....  | 35 |
| Figura 17: Aplicação Maxpooling 2x2 .....   | 36 |
| Figura 18: Flattering da matriz de características .....                                  | 36 |
| Figura 19: Arquitetura de rede YOLO .....   | 37 |
| Figura 20: Resultado detecção YOLO .....  | 37 |
| Figura 21: Aplicação da supressão não máxima .....  | 38 |
| Figura 22: Comparação do desempenho detectores de objetos.....                            | 39 |
| Figura 23: Intersecção através da união.....  | 41 |
| Figura 24: Ambiente Google Colab.....   | 42 |
| Figura 25: Exemplo de imagem anotada .....  | 43 |
| Figura 26: Anotações de classe.....   | 44 |
| Figura 27: Padrão de anotação YOLO.....   | 45 |
| Figura 28: Labellmg software de anotação.....   | 46 |
| Figura 29: Anotação feita (esq) e caixa delimitadora prevista pelo algoritmo(dir.) .....  | 47 |
| Figura 30: Original (esq.) e caixa delimitadora prevista pelo algoritmo (dir.).....       | 48 |
| Figura 31: Original (esq.) e caixa delimitadora prevista pelo algoritmo (dir.).....       | 48 |
| Figura 32: Original (acima.) e caixa delimitadora prevista pelo algoritmo (abaixo.) ..... | 49 |
| Figura 33: Imagem original para testes de oclusão .....                                   | 50 |
| Figura 34: Primeira e segunda etapa de redução oclusão.....                               | 51 |
| Figura 35: Terceira e quarta etapa de oclusão.....  | 51 |
| Figura 36: Imagem com detecção fantasma de classe inexistente .....                       | 52 |
| Figura 37: Detecção original sem distorção.....   | 53 |
| Figura 38: Nível de desfoque em 5%.....   | 53 |
| Figura 39: Nível de desfoque em 10%.....  | 54 |
| Figura 40: Nível de desfoque em 15%.....  | 54 |
| Figura 41: Nível de desfoque em 20%.....  | 55 |
| Figura 42: Detecção lateral caminhão .....  | 56 |
| Figura 43: Detecção traseira caminhão .....   | 56 |
| Figura 44: Detecção noturna .....   | 57 |
| Figura 45: Ambiente noturno em movimento .....  | 58 |
| Figura 46: Detecção noturna com luminosidade.....   | 58 |
| Figura 47: Detecção noturna múltiplas classes .....                                       | 59 |
| Figura 48: Visão da via de outra perspectiva .....  | 60 |
| Figura 49: Vista panorâmica frontal.....  | 61 |
| Figura 50: Parte do conjunto de imagens anotadas.....                                     | 61 |

## **LISTA DE ABREVIATURAS E SIGLAS**

ABS – Antilock Braking System

ADAS – Advanced Driver Assistance Systems

CCD – Charge Coupled Device

GPS - Global Positioning System

GPU - Graphics Processing Unit

IOU - Intersection over Union

OMS – Organização Mundial da Saúde

TOF – Time Of Flight

YOLO – You Only Look Once

## SUMÁRIO

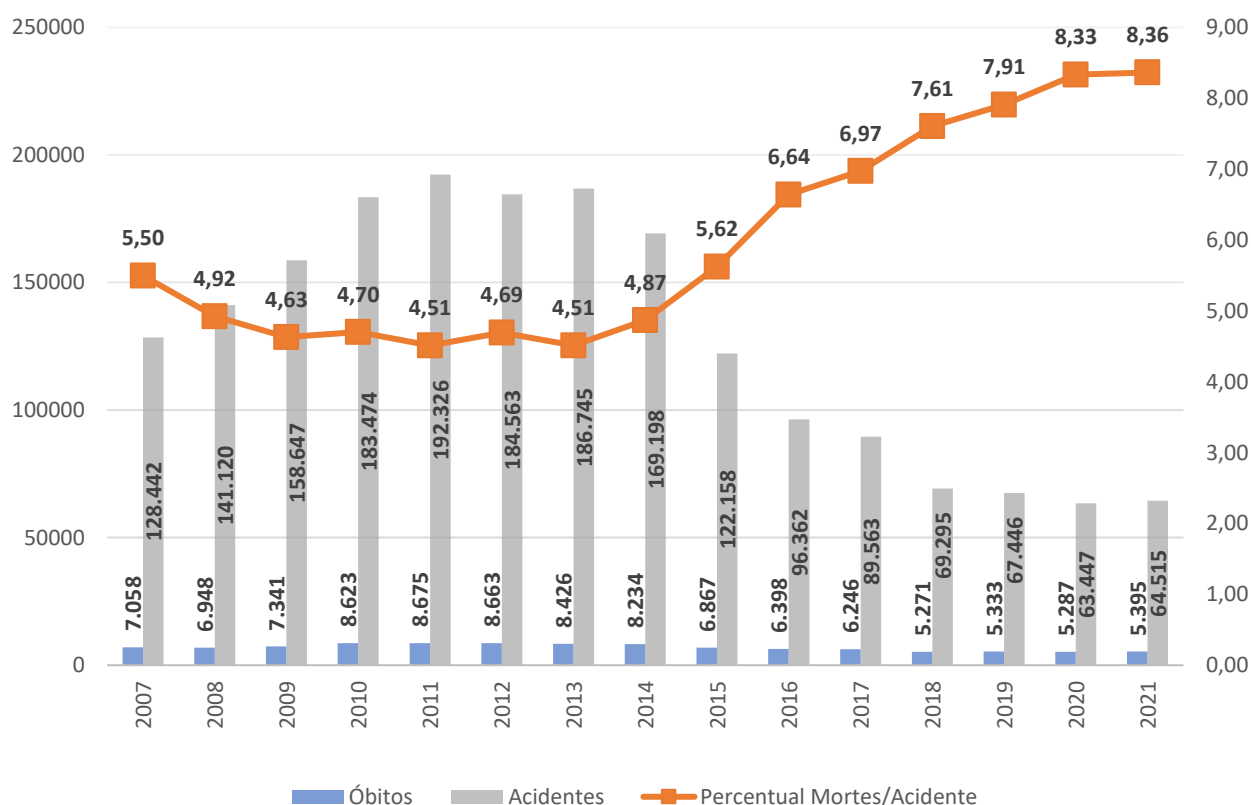
|   |    |
|---|----|
| 1 INTRODUÇÃO .....  | 15 |
| 1.1 Objetivos.....  | 18 |
| 1.1.1 Objetivo geral .....  | 18 |
| 1.1.2 Objetivo específico.....  | 18 |
| 2 DESENVOLVIMENTO.....  | 19 |
| 2.1 Revisão de literatura .....                                       | 19 |
| 2.1.1 Veículos Autônomos.....   | 19 |
| 2.1.1.1 Sensores .....  | 22 |
| 2.1.1.1.1 Radar .....   | 23 |
| 2.1.1.1.2 LiDAR.....  | 24 |
| 2.1.1.1.3 Câmera .....  | 26 |
| 2.1.2 Visão Computacional.....  | 28 |
| 2.1.3 Algoritmos de aprendizagem profunda em visão computacional..... | 31 |
| 2.1.4 YOLO – You Only Look Once .....                                 | 33 |
| 2.1.5 Métricas de avaliação .....                                     | 40 |
| 3 METODOLOGIA.....  | 42 |
| 4 ANÁLISE E DISCUSSÃO DOS RESULTADOS.....                             | 47 |
| 5 CONCLUSÃO.....  | 63 |
| REFERÊNCIAS.....  | 65 |

# 1 INTRODUÇÃO

Um relatório global de segurança, publicado pela (OMS, 2018) revela que mais de 1,35 milhão de pessoas perderam a vida em acidentes de trânsito em 2016, isso representa aproximadamente 3700 mortes por dia, além de que em torno de 50 milhões de pessoas sofrem com as sequelas dos acidentes.

A alta representatividade do trânsito no número de óbitos globais, segue a mesma tendência no cenário brasileiro. Dados da (PRF, 2021) demonstram uma diminuição no número de acidentes entre 2007 e 2021, contudo o número de mortes, percentualmente, aumentou, conforme visto no gráfico abaixo.

Gráfico 1 – Acidentes e mortes em Rodovias Federais



Fonte: Elaborado pelo Autor a partir dos dados da Polícia Rodoviária Federal (2021).

O aumento percentual do número de mortes em relação ao número de acidentes indica que existem situações que não foram amenizadas, mesmo com todas as campanhas de prevenção e enrijecimento das leis de trânsito.

As principais causas de acidentes no ano de 2021 em rodovias federais, foram a falta de atenção (34,9%), desobediência a sinalização (12,9%), velocidade incompatível (10,7%), não guardar a distância de segurança (7,9%), ingestão de álcool (7,9%), dormir ao volante (3,6%) e ultrapassagem indevida (2,5%). Sendo assim o condutor têm sido o principal causador de acidentes no trânsito, seja por imprudência ou falta de atenção.

O desenvolvimento de dispositivos que tornassem os veículos autônomos teve seu marco após a segunda guerra mundial, período em que as tecnologias desenvolvidas durante a guerra, como o *GPS*, passaram a ser aplicadas nos veículos. Em 1958 ocorreu a primeira demonstração de condução autônoma, o carro da General Motors denominado Impala, possuía um sistema de condução, que consistia em um conjunto de bobinas fixas ao veículo e um cabo enterrado ao solo o qual circulava corrente alternada que era percebida pelas bobinas do veículo (WETMORE, 2003), permitindo a condução sem motorista.

A partir dos anos 60 com as pesquisas desenvolvidas no campo da robótica móvel, promoveu-se uma série de iniciativas envolvendo dispositivos capazes de fornecer dados ao veículo para que pudesse tomar suas próprias ações, sem a necessidade de intervenção humana, atividades que contribuíram para avanços no campo dos carros autônomos (WEI, 2013).

Partindo de uma iniciativa das montadoras o desenvolvimento e implementação gradual de dispositivos capazes de auxiliar na condução dos veículos, dão suporte para a implementação de sistemas cada vez mais integrados, os quais conduzem aos sistemas autônomos. Para que isso seja possível há a necessidade de sensores capazes de captar as variáveis de ambiente de maneira assertiva e um trabalho conjunto entre hardware e software para que sejam capazes de interpretar corretamente as variáveis e tomar as melhores decisões.

Os sistemas de visão incorporados nos carros autônomos desempenham um papel importante no quesito segurança, para isso é necessária uma capacidade de processamento em tempo real, exigindo que o processamento de um quadro se conclua dentro de 30 a 40 milissegundos, requisito desafiador para sistemas embarcados (PULLI et al., 2012). Algoritmos que não onerem de maneira abusiva os sistemas de processamento viabilizam a aplicação dos sistemas de visão dentro dos veículos, uma vez que determinados algoritmos exigem um hardware muito mais

robusto e por consequência mais caro.

O presente trabalho tem o intuito de desenvolver um estudo acerca do sensoriamento em veículos autônomos, dando ênfase na aplicação da visão computacional, apresentando um dos algoritmos mais modernos na área de visão computacional, classificado como uma técnica *single pass*, havendo a possibilidade de uso para a identificação de objetos em tempo real, conhecido como *YOLO (You Only Look Once)*.

## 1.1 Objetivos

### 1.1.1 Objetivo geral

Este trabalho demonstra um estudo acerca do sensoriamento em veículos autônomos voltado a aplicação do algoritmo de visão computacional, no campo da classificação e identificação.

### 1.1.2 Objetivo específico

Para desenvolvimento do trabalho, uma série de conceitos serão vistos sendo necessário contemplar alguns pontos como:

- Estudo sobre veículos autônomos;
- Estudo dos conceitos de visão computacional;
- Estudo dos conceitos de redes neurais convolucionais;
- Estudo do algoritmo *YOLO*;
- Aplicação do algoritmo *YOLO*;
- Captação das imagens e testes;
- Análise dos resultados qualitativos e quantitativos baseados na experimentação do algoritmo.



## 2 DESENVOLVIMENTO

### 2.1 Revisão de literatura

#### 2.1.1 Veículos Autônomos

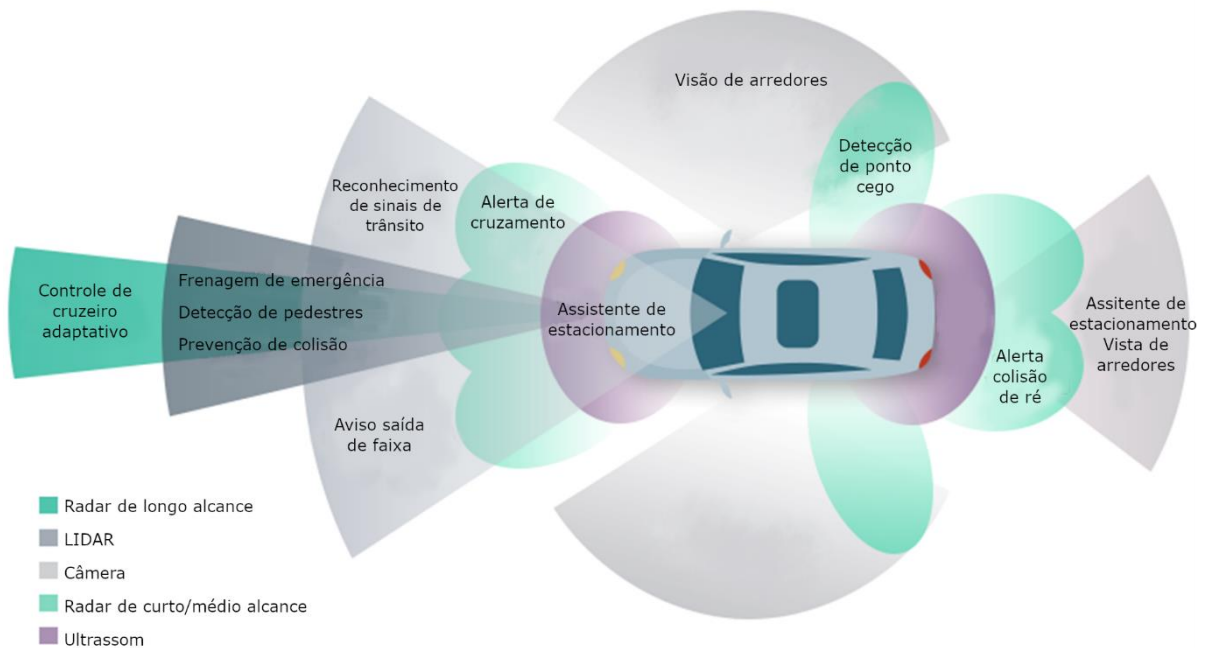
Ao longo da história os veículos passaram por uma série de mudanças, tanto nos aspectos físicos quanto conceituais. Nos últimos anos os veículos autônomos têm ganhado espaço nas rodovias, o objetivo de transportar pessoas e bens partindo de um ponto inicial e indo até determinado local, definido pelo usuário, exige uma integração total dos sistemas do veículo, sistemas estes, que gradualmente já vem sendo implementados em veículos convencionais (NAKATA, 2018).

Para que o veículo autônomo fosse possível, foram antes desenvolvidas e implementadas de maneira gradual pelas montadoras, uma série de tecnologias que contribuem de maneira assistiva na condução de veículos, sendo elas:

- *Cruise Control* (Controle de velocidade de cruzeiro);
- *Parking Assistance* (Assistente de estacionamento);
- *Lane Sterring* (Assistente de guiamento);
- *ABS* (Sistemas de freio antibloqueio);
- Sistemas de detecção de obstáculos;
- Sistemas alerta de proximidade;
- Monitoramento das condições de funcionamento dos veículos;

Estes dispositivos são conhecidos como sistemas avançados de assistência ao condutor (*Advanced driver assistance systems - ADAS*). Estes sistemas atuam de forma conjunta com o intuito de contemplar as mais diferentes direções e distâncias no veículo conforme indicado na figura 1.

Figura 1: Sensoriamento em veículos



Fonte: Interlias (2018) Adaptado pelo autor.

Através de uma central, que nada mais é que um computador, todos estes dispositivos são monitorados e controlados pelo veículo, fazendo com que em certos níveis de autonomia o condutor seja dispensável.

Tendo isto em vista, pode-se dizer que o carro autônomo é aquele que faz a condução e a tomada de decisões sem a intervenção de um ser humano (OZGUNER, 2007).

Dentro dos sistemas autônomos há normativas que os classificam de acordo com seu nível de autonomia, segundo a SAE International (2021), órgão mundial que fomenta o conhecimento acerca dos veículos e sistemas automotores, estes sistemas podem ser classificados em seis níveis.

Figura 2: Níveis de automação em veículos



Fonte: SAE International (2014). Adaptado pelo autor

A figura 2 apresenta os níveis propostos pela SAE Internacional, partindo de veículos sem autonomia alguma (nível 0), onde podem até existir dispositivos como o ABS, avisos e alertas, porém as decisões competem totalmente ao condutor.

Já no primeiro nível existe uma autonomia limitada, segurança ativa e algumas funções autônomas como a centralização na faixa ou controle de velocidade adaptativa (mantendo distância do veículo a frente), porém o condutor ainda possui um nível altíssimo de condução.

No nível dois a autonomia ainda é limitada, possuindo segurança ativa, porém existem mais conveniências em relação ao nível anterior, englobando as mesmas funções, além de tarefas como estacionamento autônomo e a utilização de funções simultâneas, como por exemplo, controle de faixa e aceleração. Neste nível ainda na maior parte do tempo o condutor permanece no controle, o sistema assume o controle apenas em certas condições

O nível três apresenta a autonomia condicional, ou seja, ela é aciona sob algumas condições, como por exemplo em rodovias que são bem demarcadas e com boa infraestrutura. O motorista pode habilitar e desabilitar a automação, dependendo do cenário ainda requer a ação do motorista em determinadas situações.

No quarto nível é apresentada um alto nível de autonomia, sendo condicionada a locais autorizados e em regiões definidas, nestes casos não há necessidade de condutor. O último nível diz respeito a autonomia total, onde não é mais necessário a atuação ou até mesmo a presença do motorista em nenhum momento.

#### 2.1.1.1 Sensores

Um sensor tem como objetivo informar a respeito da ocorrência de algum evento externo (WENDLING, 2010). De modo semelhante ao ser humano, os sensores são capazes de reproduzir os sentidos e assim, digitalizar os acontecimentos do mundo real, transformando grandezas físicas em grandezas elétricas. Com esta conversão é possível manipular estes sinais para o controle dos mais variados processos.

Dentro dos veículos autônomos não é diferente, os sensores têm papel fundamental na correta operação do sistema. Os sensores dentro do campo da robótica móvel podem ser divididos em três segmentos (MURPHY, 2000).

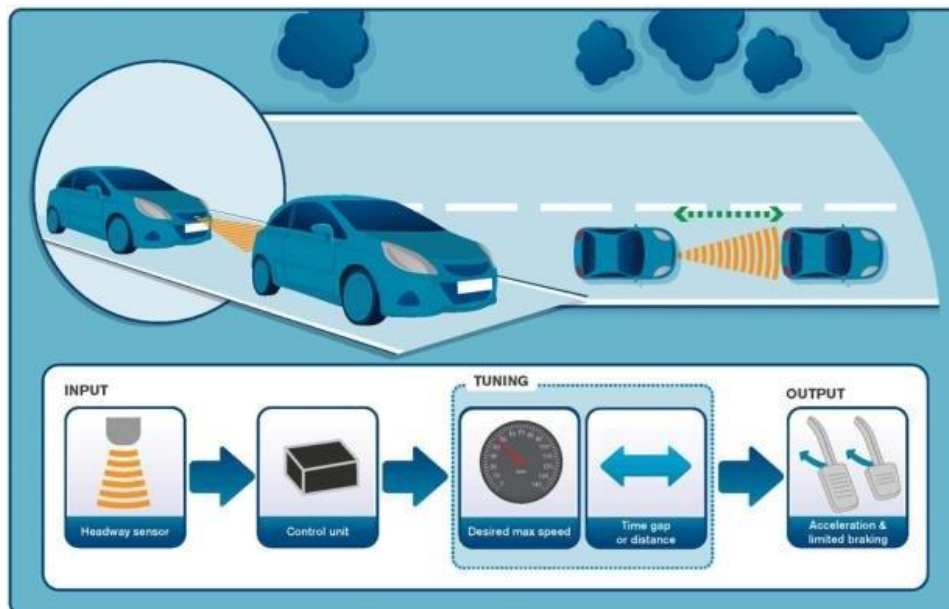
- Proprioceptivos: os quais fornecem as informações internas, relacionadas com o sistema robótico, como a velocidade das rodas, nível de bateria;
- Exteroceptivo: coletam informações acerca da área de atuação do robô, ou seja, medidas de distância, luminosidade, temperatura, estes desempenhados por câmeras, LIDAR, RADARs, sensores de ultrassom entre outros;
- Exproprioceptivo: fornecem a informação a respeito da posição das partes do robô em relação ao ambiente.

Diretamente relacionados com a percepção de ambiente os sensores que serão apresentados a seguir estão localizados dentro do campo exteroceptivo, coletando informações do ambiente e possibilitando uma reconstrução.

### 2.1.1.1.1 Radar

Segundo Brain (2003, p. 6) o radar, diferente do sonar, utiliza onda de rádio para detecção de objetos, elas viajam a longas distâncias e permitem detecção, mesmo com sinais fracos. Já implementadas em veículos os sistemas *ADAS* (*Advanced Drive Assistance Systems*), possuem radares que contribuem para a geração de alarmes na detecção de colisões com outros veículos e oferecem controle do *adaptive cruise control* que nada mais é do que o controle da velocidade de cruzeiro, conforme visto na figura 3.

Figura 3: Sistema de controle de velocidade de cruzeiro



Fonte: Çupi (2019).

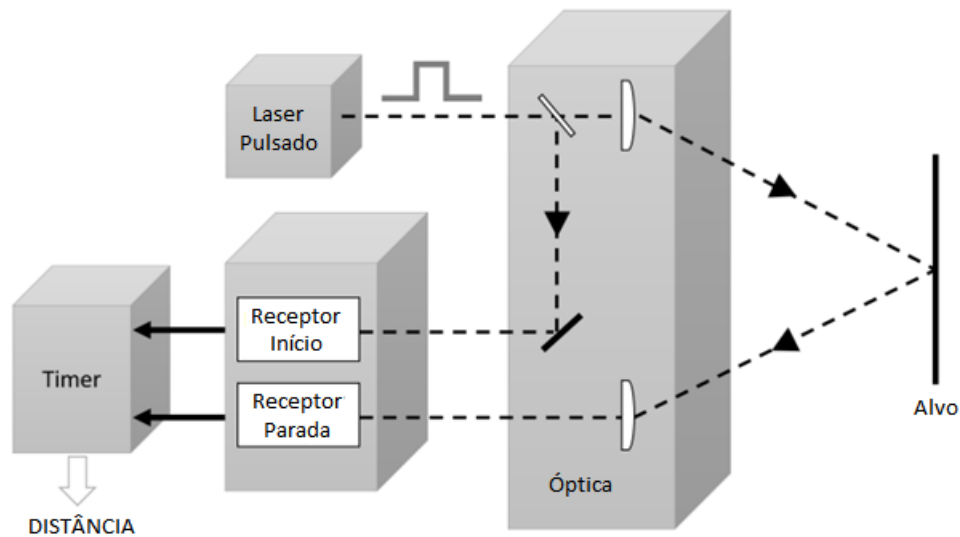
Nos sistemas que controlam velocidade o radar é responsável por monitorar a distância do veículo a frente, enviar o sinal para a central de controle que faz o processamento dos dados, levando em consideração a velocidade atual e a distância do veículo a frente, podendo assim realizar o controle de velocidade do veículo.

Hoje existem radares que operam na faixa dos 77Ghz, oferecendo assim distancias de alcance em torno de 200m em um arco de 10°, já para um arco de 30° a distância fica reduzida a 30m. Quanto mais alta a frequência melhor a resolução obtida, isto permite que o sensor possa distinguir diferentes objetos em tempo, promovendo um melhor controle (DIGI-KEY'S EUROPEAN EDITORS, 2016).

### 2.1.1.1.2 LiDAR

Nos últimos anos o LiDAR (*light detection and ranging*), evoluiu, sendo uma das principais técnicas para estudos de mapeamento. Seu funcionamento é baseado no princípio *TOF (time-of-flight)*, como visto na figura 4, que opera através da contagem do tempo entre emissões de feixe pulsado de luz e sua captação, com estas medições de tempo a velocidade da luz, que é conhecida, é utilizada para computar a distância e realizar o mapeamento (ROYO e BALLESTA-GARCIA, 2019).

Figura 4: Princípio de medição do sistema TOF (*time-of-flight*)



Fonte: Royo e Ballesta-Garcia (2019). Adaptado pelo autor.

Com o uso dos dados gerados pelo sistema LiDAR o veículo sabe a posição exata dos objetos em volta, permitindo que o veículo seja guiado, evitando colisões, desviando de objetos. Usualmente estes sensores têm sido instalados na parte superior do veículo, conforme apresentado na figura 5, permitindo uma captação ampla do ambiente em volta.

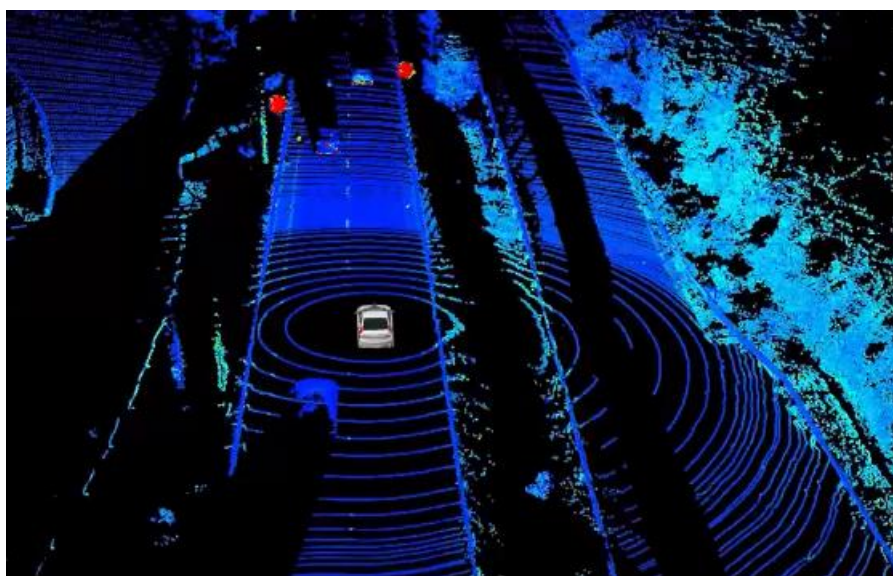
Figura 5: Sensor LiDAR HDL-64E (esq.) Montagem em um Toyota Priuse (dir.)



Fonte: Mukhtar et al. (2015).

Os produtos disponíveis hoje no mercado demonstram uma alta eficácia em seu funcionamento, porém ainda ocorrem problemas relacionados à qualidade dos sensores, os quais necessitam atender uma série de requisitos para que possam ser aplicados, uma vez que é necessário uma varredura precisa de dados em tempo real e robustez sobre diversas condições climáticas (MUKHTAR *et al.*, 2015). Dentro dos produtos comerciais o custo é um ponto crítico, há casos em que o preço do equipamento é superior ao do veículo. A figura 6 apresenta o mapa formado pela nuvem de pontos captados pelo LiDAR.

Figura 6: Mapeamento gerado pelo LiDAR



Fonte: RoboSense (2022).



### 2.1.1.1.3 Câmera

Nos veículos, inicialmente, as câmeras foram utilizadas em aplicações menos complexas como dispositivos de auxílio ao condutor, sendo implementadas como por exemplo em sistemas de assistência de estacionamento, permitindo a visão mais clara do local a ser estacionado, segundo Tsuchiya et al. (2012) este tipo de aplicação se popularizou ao ponto de que agências nacionais de trânsito de alguns países como Estados Unidos propusessem novas regras exigindo este tipo de dispositivo nos veículos.

No campo dos veículos autônomos as câmeras têm auxiliado na detecção dos mais variados objetos, o desenvolvimento de algoritmos de visão computacional, permitiu que elas desempenhassem um papel importante dentro do segmento, uma vez que as câmeras possuem um custo muito inferior a sistemas como o LiDAR.

Figura 7: Sistema de câmeras carro Tesla

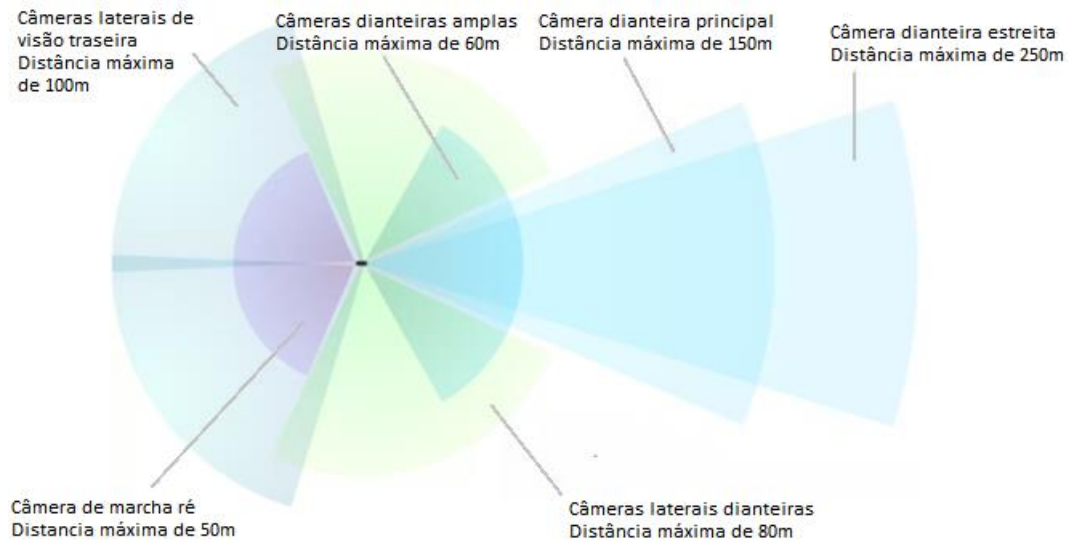


Fonte: Tesla (2022). Adaptado pelo autor.

O exemplo pode ser visto na figura 7, os veículos da fabricante de carros Tesla, não possuem sistema LiDAR, para realizar a movimentação autônoma através do sistema *Autopilot* o carro é orientado por um conjunto de câmeras com capacidades de distâncias que variam de 50 a 250 metros (TESLA, 2022), conforme pode ser visto na figura 8.



Figura 8: Conjunto de câmeras Tesla Vision



Fonte: Tesla (2022). Adaptado pelo autor

Nos últimos anos ocorreu a remoção de sensores ultrassônicos de alguns modelos de veículos (Model 3, Model Y, Model S e Model X), fazendo com que o sistema *Tesla Vision* passasse a ser totalmente baseado nas câmeras instaladas no veículo (TESLA, 2022).

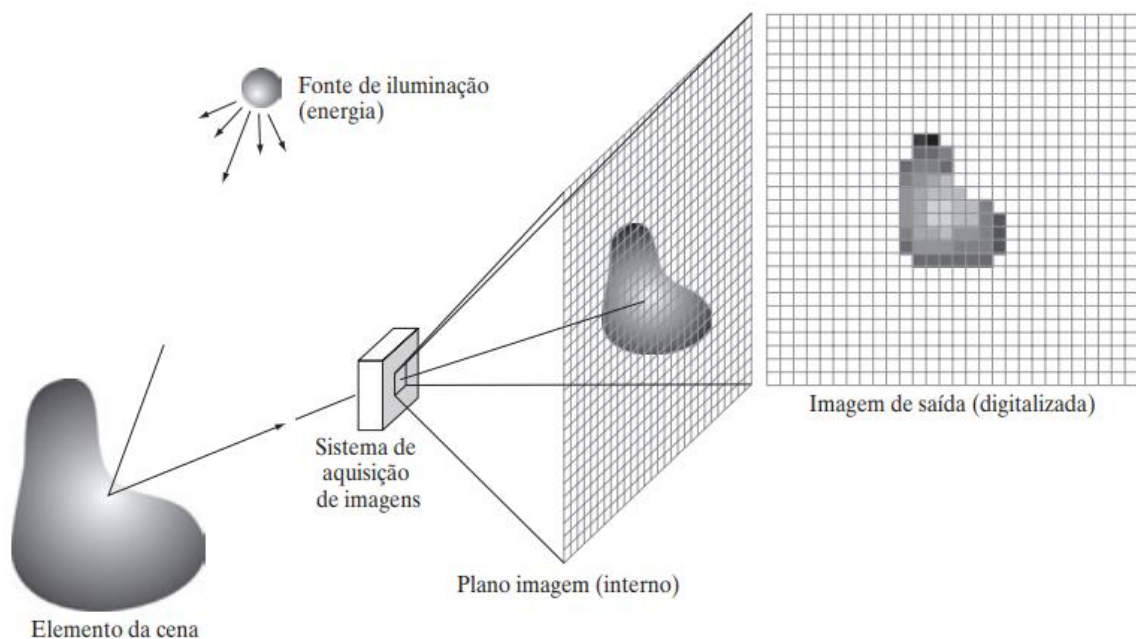
As câmeras são responsáveis por captar o ambiente e fornecer as informações para que os algoritmos possam reproduzir o mundo a volta do veículo, identificando cores e formatos, permitindo a análise e interpretação de sinalizações de trânsito, detecção das pistas, pessoas, animais entre outros. Estas tarefas são diretamente ligadas ao software ao qual realiza estas detecções e ao hardware que processa estas informações, exigindo altos níveis de desenvolvimento tanto dos algoritmos quanto do hardware.

## 2.1.2 Visão Computacional

Segundo Fu et al. (1987, p.296) a capacidade de visão dá ao sistema um mecanismo sofisticado de sensoriamento, fazendo com que a máquina possa responder ao ambiente de maneira inteligente. O intuito de utilizar sistemas de visão, está ligado diretamente com o aumento da flexibilidade e gama de aplicação de sistemas robóticos. Pode-se dividir o segmento de visão computacional em seis principais etapas, aquisição, pré-processamento, segmentação, descrição, reconhecimento e interpretação. Cada etapa dentro do processo tem um objetivo específico.

Conforme elucidado por Gonzalez e Woods (2010 p.32) o processo de aquisição de imagens em câmeras digitais que possuem sensores CCD, ocorre de maneira que a energia de uma fonte luminosa reflete os elementos da cena e é captada por um sistema de aquisição que é basicamente uma matriz de elementos fotossensíveis, quanto maior o sensor CCD, maior será a resolução da imagem obtida, a seguir é apresentado um esquemático, figura 9, do sistema de aquisição de imagens.

Figura 9: Exemplo processo de aquisição imagem

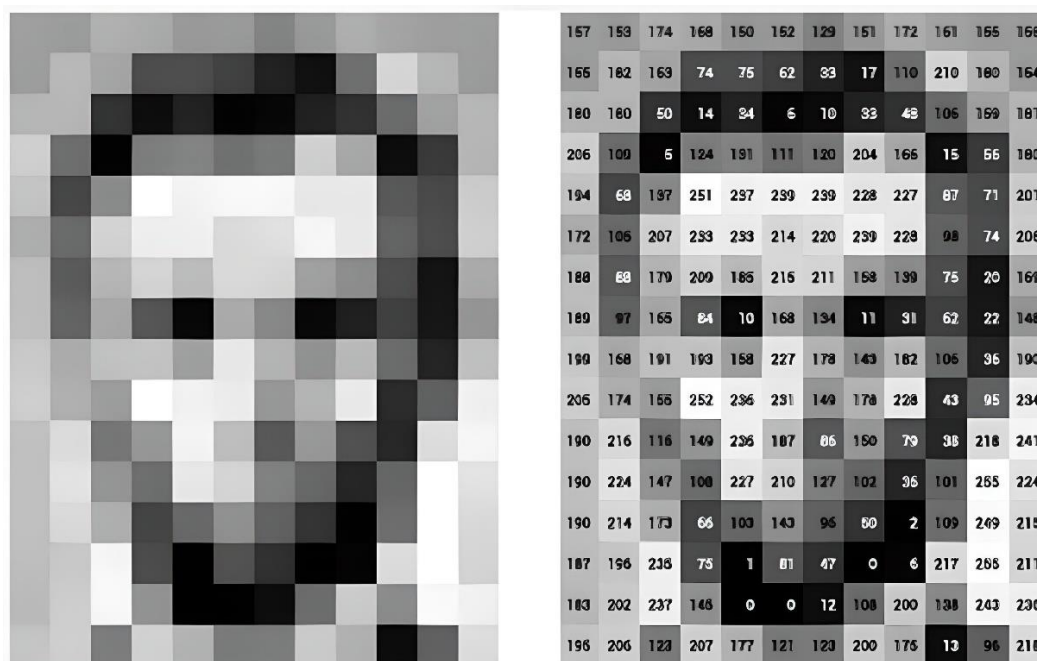


Fonte: Gonzalez e Woods (2010). Adaptado pelo autor.

Dentro do sistema de aquisição, existe um arranjo de sensores que produzem saídas proporcionais a integral da luz que incide sobre eles, através de circuitos digitais e analógicos acontece uma varredura destas saídas as quais são convertidas em sinais analógicos para posteriormente formar a imagem digitalizada.

A imagem digitalizada consiste em matrizes compostas com diferentes níveis de intensidade, as imagens coloridas no sistema RGB possuem 3 camadas (*red*, *green* e *blue*), já imagens na escala de cinza possuem apenas uma camada, com valores que variam de 0 a 255, conforme visto na figura 10.

Figura 10: Quantização de imagem digitalizada em escala de cinza



Fonte: Prathmesh Muchhala (2020). Adaptado pelo autor.

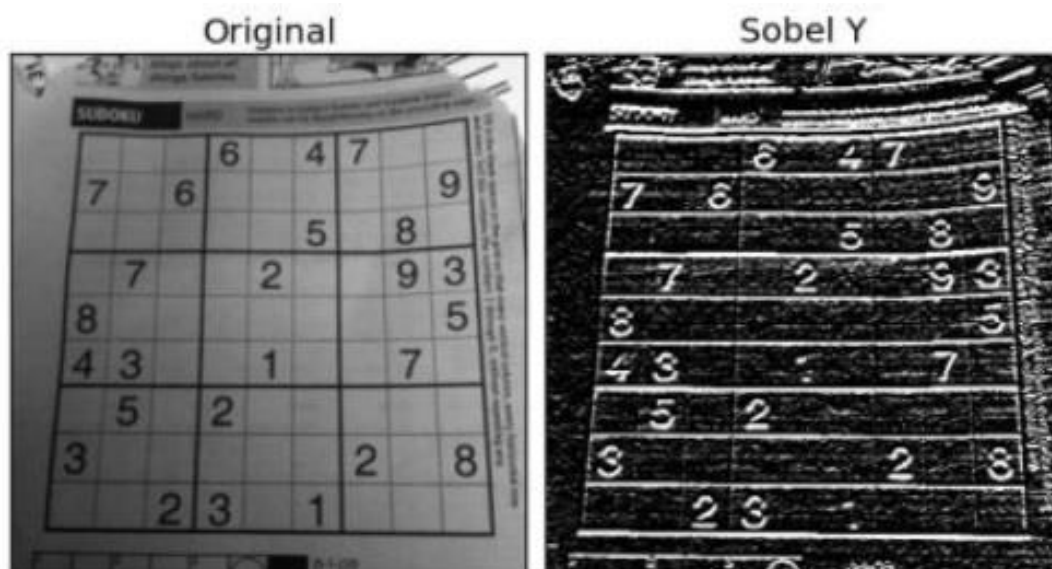
O pré-processamento de imagens tem como objetivo a aplicação de técnicas para realce da imagem, removendo ruídos e promovendo destaque de uma determinada região de interesse (SOUZA e CORREIA, 2007). Dentre as técnicas estão os filtros passa-alta, que fazem a ampliação de bordas, filtros passa-baixa, que fazem a suavização das bordas, normalização de histograma e transformada discreta de Fourier.

A segmentação da imagem tem como objetivo selecionar áreas de interesse ou objetos que a compõem. Os algoritmos de segmentação baseiam-se em propriedades básicas de valores de intensidade de descontinuidade e similaridade. Algumas das

abordagens consistem em dividir uma imagem baseado em mudanças bruscas de intensidade, como as bordas, já outras abordagens dividem a imagem em regiões que são semelhantes em determinado critério, são exemplos destes métodos a limiarização, crescimento de região, divisão e fusão (GONZALEZ e WOODS, 2010).

Na figura 11 é demonstrada a aplicação de um filtro para intensificar as linhas horizontais, denominado *Sobel*, a aplicação do filtro consiste no produto da matriz do filtro pela matriz da imagem. Dependendo da composição do filtro, diferentes características são intensificadas ou atenuadas.

Figura 11: Aplicação do Filtro Sobel Y



Fonte: OpenCV (2022). Adaptado pelo autor.

Na aplicação de visão de máquina, a extração de características é necessária para a identificação e distinção de objetos uns dos outros, isso é possível através da correlação com características que identificam os objetos de maneira única (VIANNA, 2009). Área, momentos da imagem e centro de área, são características que são utilizadas para tal. É importante que estas características não tenham relação direta com posição, orientação do objeto na cena, nem tão pouco a fatores de escala.

O reconhecimento de objetos, baseia-se na comparação entre as características extraídas da imagem com as características do objeto alvo. Pode-se dividir as metodologias de reconhecimento em duas áreas, as que utilizam de padrões descritores quantitativos como comprimento, área e textura. A outra área trata de comparação com descritores qualitativos (GONZALEZ e WOODS, 2010).

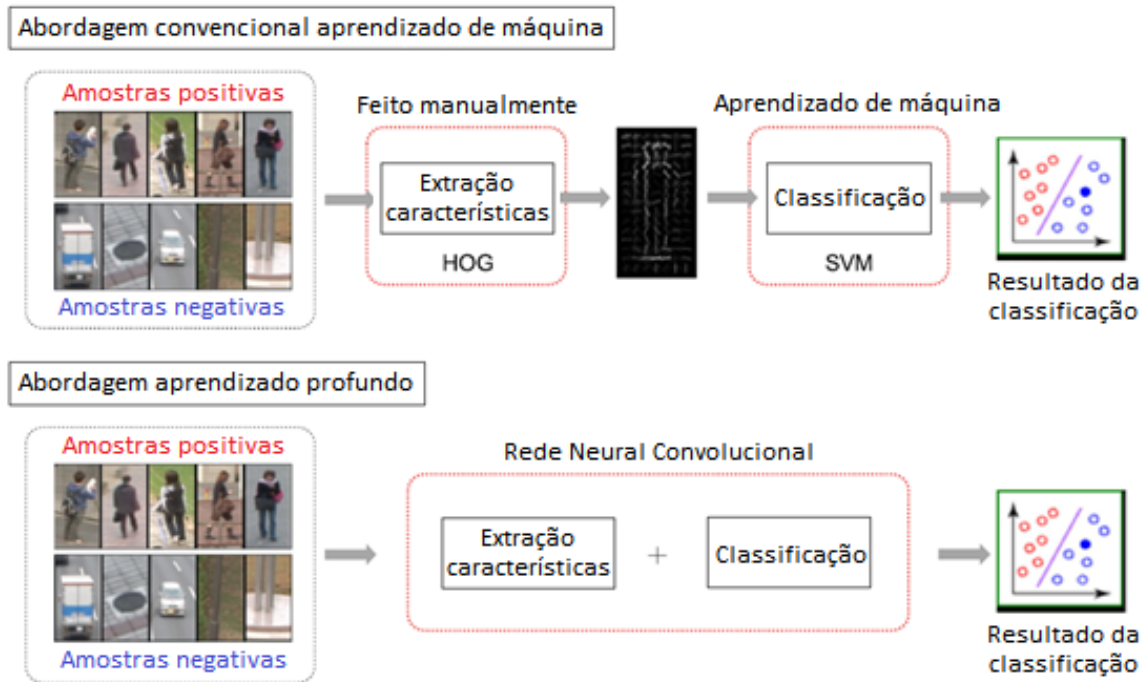
### 2.1.3 Algoritmos de aprendizagem profunda em visão computacional

Conforme apresentado por Fujiyoshi et al (2019), com a evolução dos computadores na década de 1990, foi possível processar uma quantidade maior de dados com certa velocidade. Utilizando-se disto nos sistemas de visão computacional a principal técnica utilizada era a extração do vetor de características da imagem, ou seja, recursos locais da imagem, e utilização do aprendizado de máquina para executar as detecções.

Nos anos 2000 as pesquisas na área de detecção de objetos envolvendo visão computacional avançaram, e apresentaram alguns algoritmos, que ainda não utilizavam de redes neurais para detecção, e dependiam ainda do conhecimento humano para definição dos detectores, dentre eles se destacam o *Viola Jones Algorithm* (VIOLA; JONES, 2001), que utiliza de *Haar cascade classifiers*, *Scale Invariant Feature Transform (SIFT)* (LOWE, 1999), *Histogram Of Oriented gradients (HOG)* (DALAL; TRIGGS, 2005) entre outras.

Em 2010 o campo do aprendizado profundo entrou em destaque, em particular no campo dos algoritmos de visão foram desenvolvidas as redes neurais convolucionais (CNN's) esta nova abordagem, permitiu que as redes fossem utilizadas para realizar a extração das características da imagem por meio do aprendizado e posterior detecção, o que tornava o processo mais robusto, uma vez que as características importantes na detecção são determinadas pela rede (FUJIYOSHI et al, 2019). São alguns exemplos de algoritmos que utilizam de aprendizado profundo: *Region Proposal CNN (R-CNN)*, *Fast R-CNN*, *Faster R-CNN* e *You Only Look Once (YOLO)*. A figura 12 demonstra um comparativo entre as abordagens do aprendizado de máquina e o aprendizado profundo.

Figura 12: Comparativo entre machine learning convencional e deep learning



Fonte: Fujiyoshi et al. (2019). Adaptado pelo autor

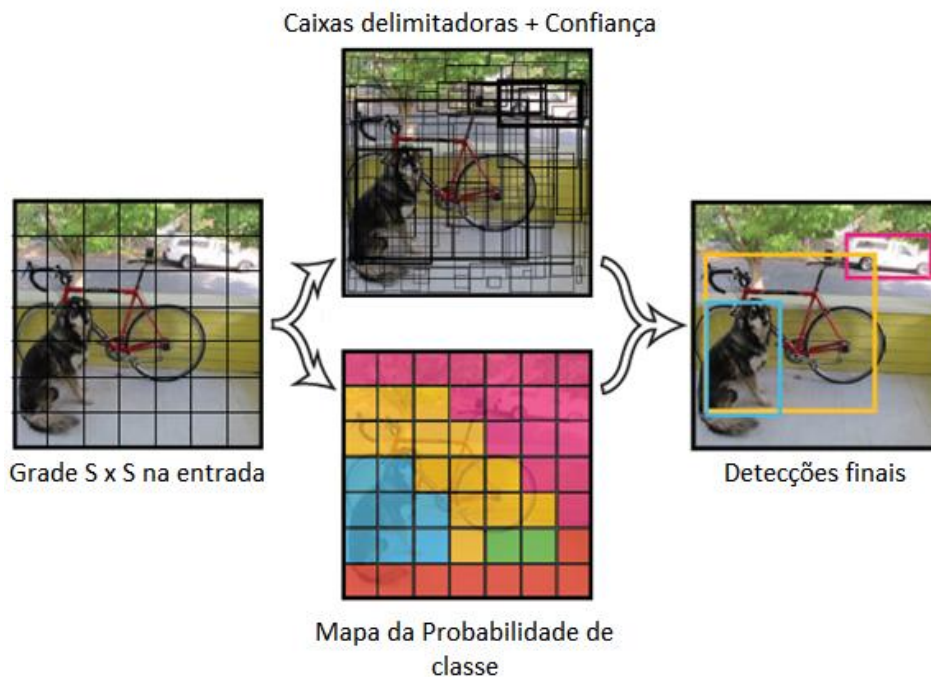
O surgimento de aplicações dinâmicas com a necessidade de detecção em tempo real, como por exemplo nos veículos autônomos, demandou o desenvolvimento de algoritmos cada vez mais ágeis e assertivos, dentre eles um modelo se diferenciou dos métodos de detecção convencionais, o YOLO. O ponto no qual este algoritmo se destaca é devido a não utilização o método de janela deslizante para detecção (SHARIFI *et al.*, 2021), ou seja, é necessário que a imagem seja processada apenas uma vez para que ocorram as detecções, promovendo rapidez na execução.

## 2.1.4 YOLO – You Only Look Once

O YOLO (REDMOND et al., 2016) é uma arquitetura para detecção de objetos que utiliza de redes neurais convolucionais e é extremamente rápida e precisa, sendo capaz de processar imagens em tempo real, em uma taxa de aproximadamente, 45 quadros por segundo utilizando uma GPU Titan X da Nvidia. Sua primeira versão foi desenvolvida em 2015 e passou por aprimoramentos em suas versões subsequentes.

O YOLO realiza a detecção de caixas delimitadoras (*bounding boxes*) e da probabilidade de classes para posterior combinação dos resultados, conforme visto na figura 13.

Figura 13: Sistema de detecção YOLO



Fonte: Redmon et al.(2016). Adaptado pelo autor

Nas camadas convolucionais é feita a extração das características da imagem enquanto na camada totalmente conectada é feita a predição das saídas, com os dados de probabilidade e coordenadas dos objetos (REDMOND et al., 2016).

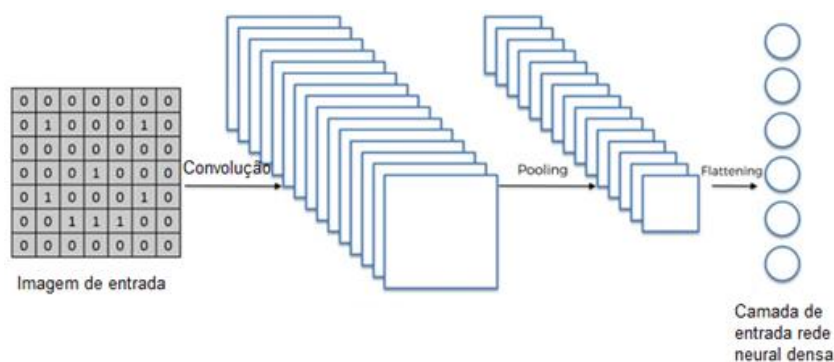
Os métodos convencionais de detecção que utilizam apenas redes neurais densas normalmente empregam classificadores pré-definidos, exigindo que o usuário determine quais as características importantes para a detecção dos objetos em questão. Já as redes neurais convolucionais no quesito extração de características,



são capazes de determinar quais os pontos mais importantes da imagem para a detecção dos objetos, sem a necessidade de escolha e influência do usuário.

A operação das camadas de convolução pode ser definido em quatro etapas, operador de convolução, *pooling*, flattening e pôr fim a aplicação dos dados na rede neural densa, conforme topologia da figura 14.

Figura 14: Etapas camada convolucional



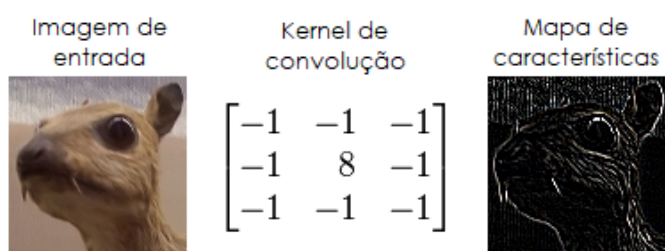
Fonte: Amir Ali (2019). Adaptado pelo autor

A etapa de convolução funciona com a aplicação de um *kernel* (filtro) a imagem, este denominado *feature detector* (detector de características), através do filtro obtém-se o mapa de características da imagem de acordo com a composição do detector.

Neste processo o detector percorre cada região da imagem e é realizado o produto de cada elemento da matriz da imagem com cada elemento da matriz do detector, somam-se estes resultados gerando um novo valor para o *pixel* da matriz de características. Este procedimento resulta em uma matriz menor do que a imagem original, facilitando o processamento.

A figura 15 demonstra a convolução de um kernel que destaca as bordas.

Figura 15: Aplicação de um kernel de bordas



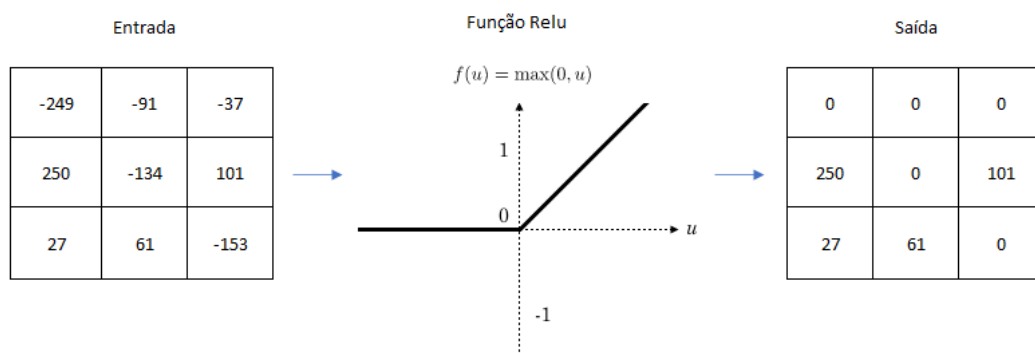
Fonte: Nvidia (2022). Editado pelo autor



No YOLO o processo é para extração das características é semelhante, porém, a matriz do detector não é predefinida e não é apenas uma, o próprio algoritmo é responsável por otimizá-las, através das técnicas de retro propagação durante o treinamento da rede, fazendo com que as melhores características sejam utilizadas para detecção dos objetos. É formado um conjunto denominado *feature maps* (mapas de características). Assim o algoritmo estabelece os detectores que possuem melhor desempenho para o grupo de imagens a serem detectadas.

Ainda no processo de convolução ocorre o produto da matriz de características pela função Relu, que trata de uma função identidade para valores maior que zero e zero para valores negativos, ela tem o intuito de eliminar valores negativos da matriz e intensificar características, conforme apresentado na figura 16.

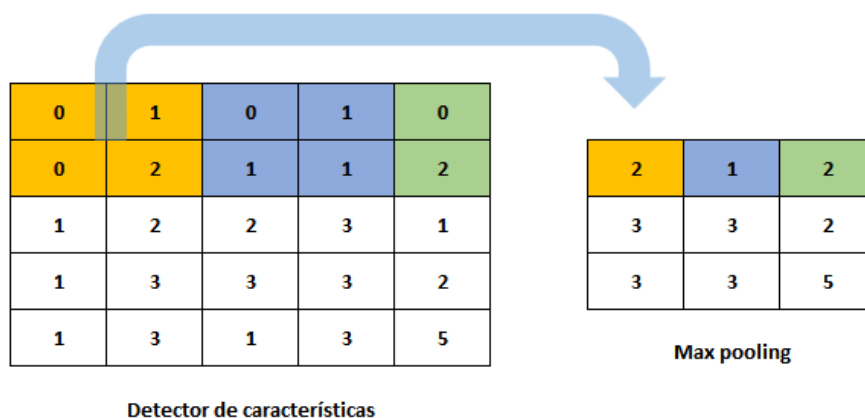
Figura 16: Aplicação função Relu



Fonte: Autor (2022)

A etapa de *pooling* tem o objetivo de otimizar o detector, intensificando as principais características da matriz e reduzindo o *overfitting*. Um exemplo de *pooling* aplicado ao YOLO é o *max pooling*, que seleciona o maior valor da matriz dentro de uma dimensão determinada, na figura 17 é aplicado um *max pooling* de 2x2.

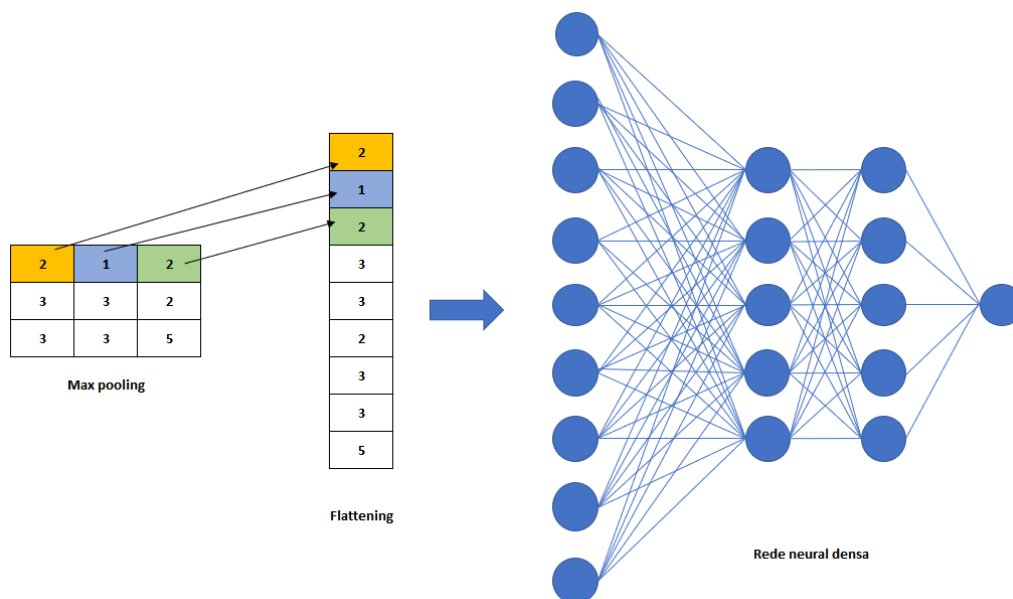
Figura 17: Aplicação Maxpooling 2x2



Fonte: Autor (2022)

Já a etapa de *flattening*, figura 18, diz respeito a transformação da matriz de características em forma de vetor para que possa ser submetida a rede neural densa.

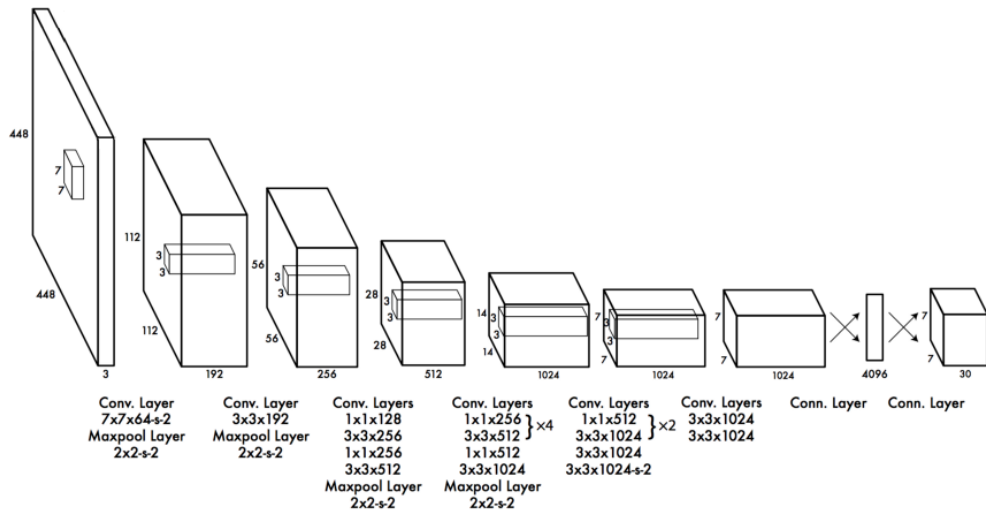
Figura 18: Flattering da matriz de características



Fonte: Autor (2022)

O algoritmo *YOLO* integra os componentes da detecção de objetos em apenas uma rede neural, a qual utiliza as características da imagem, extraídas pelas camadas convolucionais e armazenadas nos mapas de características para prever cada caixa delimitadora, sendo que todas as caixas, de todas as classes são analisadas simultaneamente. A arquitetura da rede pode ser vista na figura 19.

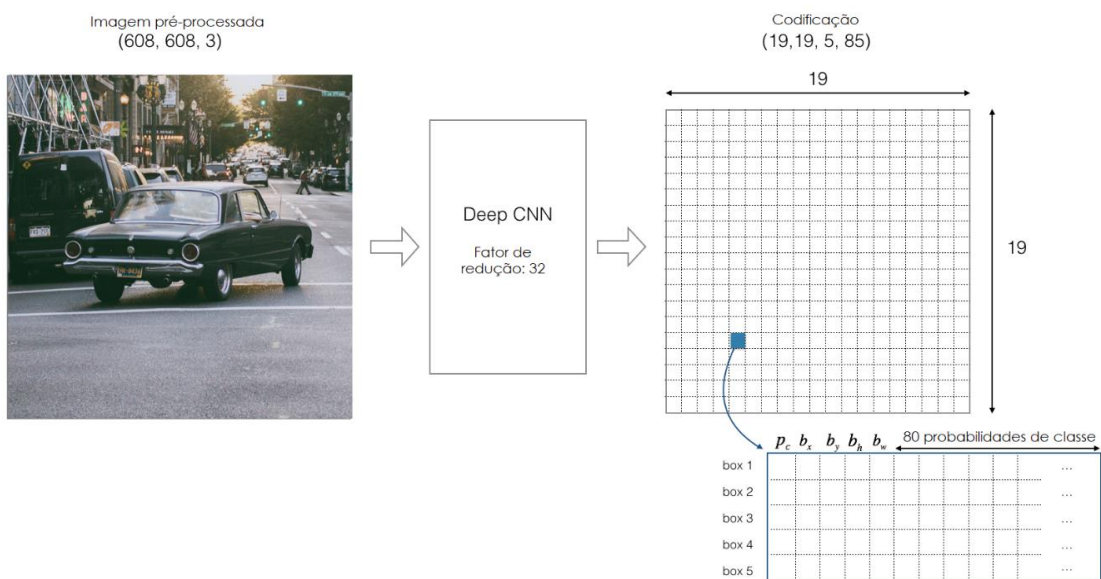
Figura 19: Arquitetura de rede YOLO



Fonte: Redmon et al.(2016).

A arquitetura original continha 24 camadas convolucionais e duas camadas totalmente conectadas. Em relação a detecção, quando uma imagem é submetida ao YOLO ela é dividida em uma grade SxS cada célula da grade é responsável por prever 5 caixas delimitadoras bem como a confiança de que há um objeto em seu interior e as probabilidades das classes (REDMOND et al., 2016), a imagem 20 apresenta uma imagem submetida ao YOLO a qual foi dividida em uma grade 19x19.

Figura 20: Resultado detecção YOLO



Fonte: Jędrzej Świeżewski (2020). Adaptado pelo autor

Para cada célula da grade são retornadas 5 caixas delimitadoras contendo as seguintes variáveis: confiança,  $x$ ,  $y$ ,  $w$ ,  $h$  e as probabilidades de classe. O valor de confiança apresentado expressa quão o modelo está confiante de que a caixa identificada contém um objeto, os termos  $x$  e  $y$ , indicam as coordenadas do centro da caixa detectada em relação aos limites da célula da grade em que a previsão está inserida,  $w$  e  $h$ , são altura e largura em relação a imagem total. (REDMOND et al., 2016).

Nem todas as caixas conterão um objeto em seu interior, desse modo é aplicada uma técnica denominada *Non Max-Supression*, ou seja, supressão não máxima, que tem o intuito de remover as caixas delimitadoras com as menores probabilidades de conter um objeto e unir as caixas que possuem a mesma região compartilhada, conforme visto na figura 21.

Figura 21: Aplicação da supressão não máxima



Fonte: Jędrzej Świeżewski (2020). Adaptado pelo autor

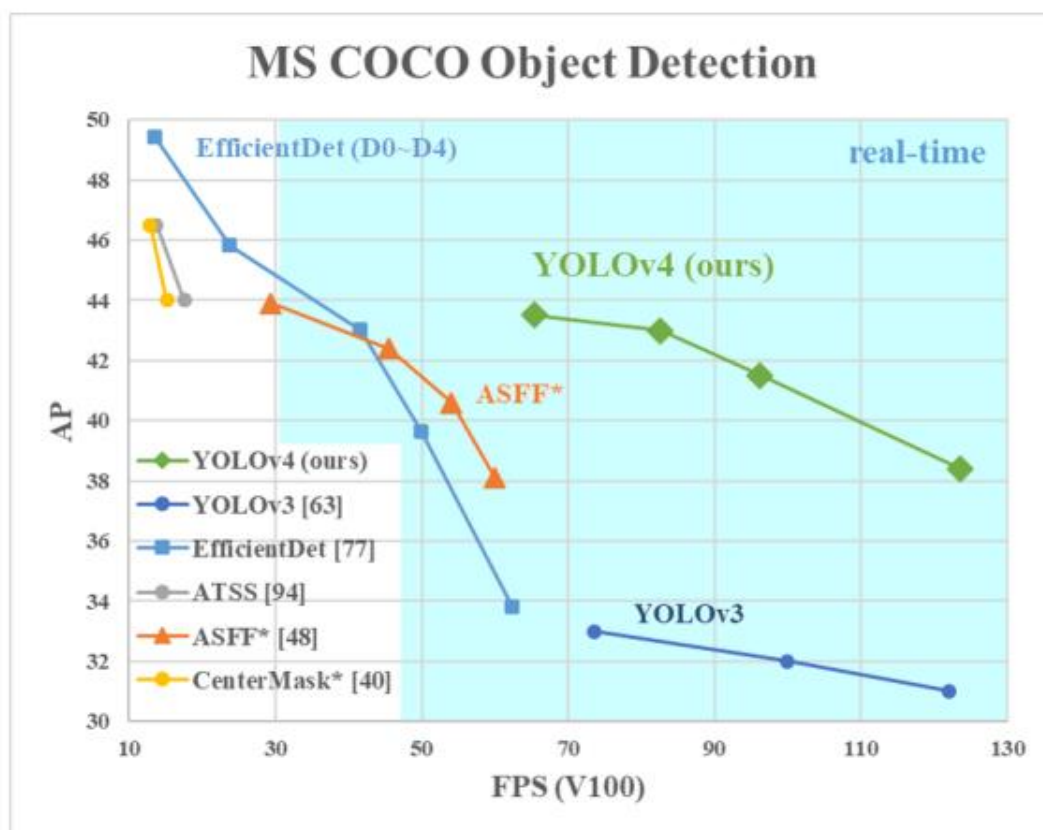
Posteriormente o valor de confiança estabelecido para a caixa delimitadora e o valor da probabilidade de classe são combinados para indicar a pontuação final da detecção, para que uma detecção seja mostrada a pontuação levantada deve ser superior ao parâmetro denominado *threshold*, em tradução direta “gatilho”, que se trata do limiar para detecções.

Nas versões subsequentes o YOLO teve melhoras significativas no que diz respeito a detecção, o conceito de âncoras foi implementado, o que otimizou a

detecção de caixas delimitadoras (REDMON; FARHADI, 2017) e a classificação das imagens foi aprimorada, sendo composta por 19 camadas convolucionais e 5 camadas de maxpooling. Na versão posterior foi implementado um novo extrator de características, o qual continha 53 camadas convolucionais, tal mudança melhorou o desempenho da detecção (REDMON; FARHADI, 2018). Após a terceira versão, o algoritmo passou a ser aprimorado por (BOCHKOVSKIY et al. ,2020) que apresentaram a versão YOLOv4, nesta versão foram feitas algumas alterações e implementadas uma série de melhorias as quais inferiam diretamente na otimização do sistema para processamentos paralelos, melhorando o desempenho.

Na figura 22 pode-se ver um comparativo do desempenho do YOLO em relação a outros detectores.

Figura 22: Comparação do desempenho detectores de objetos



Fonte: Bochkovskiy et al. ,2020

Como observado na figura 21 o YOLOv4 roda duas vezes mais rápido que o EfficientDet com um desempenho comparável, e teve melhora nas métricas AP e FPS em relação ao YOLOv3, 10% e 12% respectivamente. Desse modo o YOLO se

apresenta como um detector de última geração, demonstrando ótimos resultados em taxas de quadros razoáveis, podendo ser aplicado em situações de processamento em tempo real

### 2.1.5 Métricas de avaliação

Dentro dos sistemas de detecção existem algumas métricas de avaliação do detector, através de desafios no campo da detecção de objetos cada uma das instituições que os promovem competições online utiliza uma determinada métrica para avaliar os resultados, essas métricas foram trazidas para o campo da pesquisa e foram adotadas para avaliar o detector (PADILLA et al. 2021). A detecção é representada por um conjunto de três atributos, a classe do objeto, a caixa delimitadora e a pontuação de confiança.

A precisão é a habilidade do detector em identificar apenas objetos relevantes, sendo o percentual de corretos positivos.

- $TP$  indica o número de verdadeiros positivos;
- $FP$  é o número de falsos positivos;
- $FN$  é o número de falso negativos.

$$Precisão = \frac{TP}{TP+FP} \quad (1)$$

O *recall* é a capacidade em encontrar todas as caixas delimitadoras, sendo assim, representa o percentual de verdadeiros positivos detectados entre todas as caixas.

$$Recall = \frac{TP}{TP+FN} \quad (2)$$

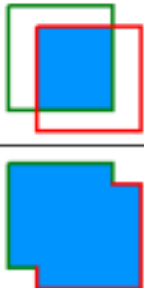
Para que se tenha esta noção do que se trata ou não de uma detecção correta estabeleceu-se uma maneira de avaliar os resultados, a intersecção sobre a união, ou simplesmente IOU (*intersection over union*), sendo uma medida baseada na no índice

Jaccard, um coeficiente de similaridade para dois conjuntos (PADILLA et al. 2020). É feita então a medida de sobreposição da caixa delimitadora prevista com a caixa delimitadora real, conforme visto abaixo

$$IOU = \frac{area(B_p \cap B_{gt})}{area(B_p \cup B_{gt})} \quad (3)$$

Na expressão acima  $B_p$  representa a caixa delimitadora estimada e  $B_{gt}$  é a caixa delimitadora que foi rotulada, a expressão acima pode ser melhor entendida através da imagem 23.

Figura 23: Intersecção através da união

$$IOU = \frac{\text{área de sobreposição}}{\text{área de união}} = \frac{\text{Imagem 23}}{\text{Imagem 23}}$$


Fonte: Padilla et al. (2020)

Desse modo estabelecido um *threshold* “ $t$ ” ou valor de referência, pode-se utilizar como parâmetro para classificar se a detecção está correta ou incorreta. Se o índice IOU for maior ou igual ao parâmetro de referência “ $t$ ” a detecção então é considerada correta, caso seja inferior, é considerada incorreta.

Dentro das métricas de algoritmos de visão é comum ver a seguinte variável para indicar o desempenho do detector: mAP ou AP, seguidos de um valor, como por exemplo  $AP_{50}$  ou  $AP_{75}$ . Esta nomenclatura se refere a precisão média de detecção em todas as classes, seguida do parâmetro de IOU (*Intersection Over Union*).

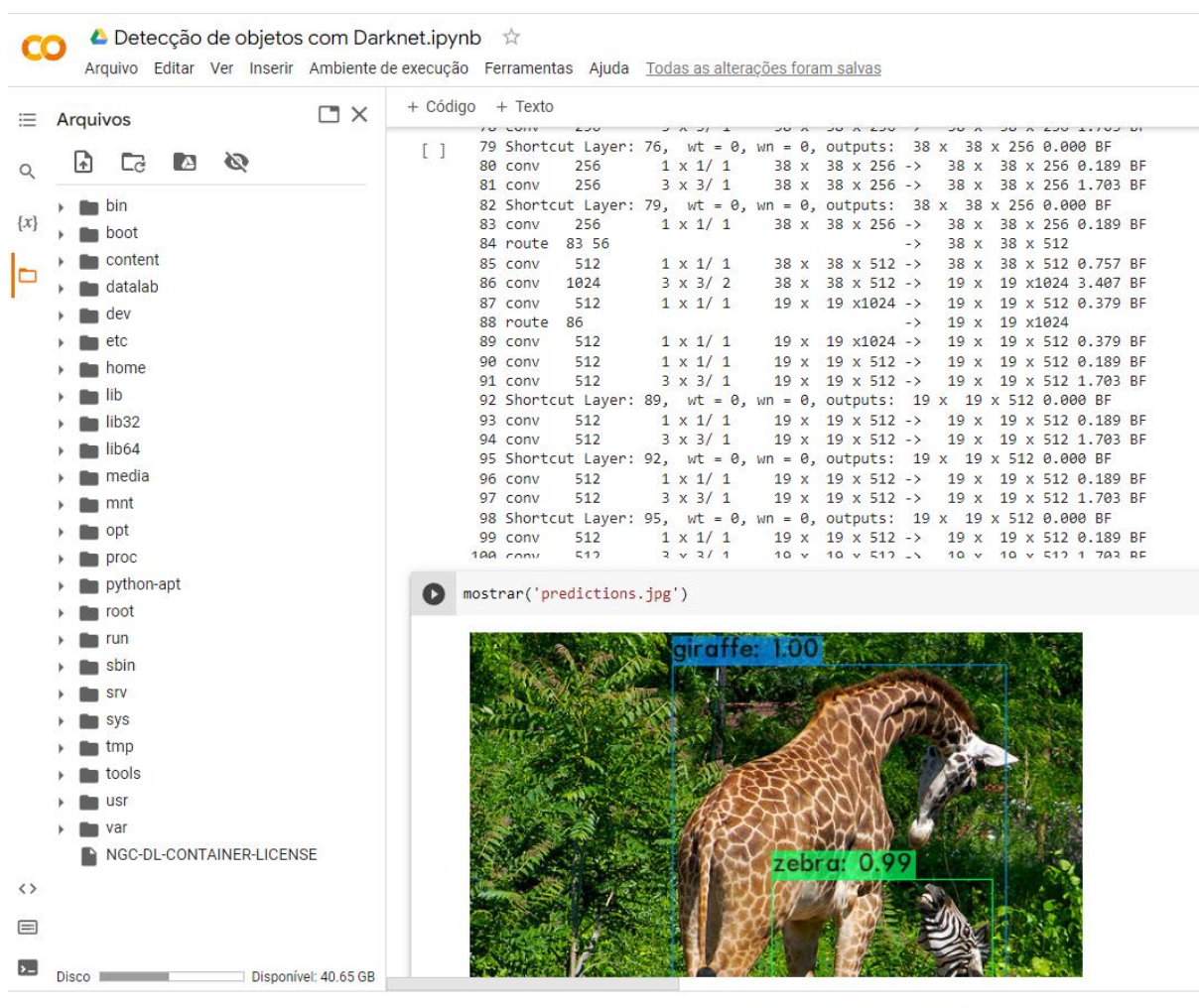


### 3 METODOLOGIA

Com o intuito de demonstrar o funcionamento de um dos modelos dos algoritmos de detecção, foi utilizada a plataforma do Google Colaboratory (Google Colab), figura 24, que nada mais é que um serviço em nuvem, que permite a execução de códigos sem a necessidade da instalação de aplicativos ou bibliotecas no computador.

A plataforma conta com os mais variados recursos para o desenvolvimento de pesquisa na área do aprendizado de máquina como o uso de GPU's remotas para processamento, recurso que fornece agilidade durante o treinamento das redes e no processo de detecção, no trabalho a GPU fornecida para os testes foi uma Tesla T4 da Nvidia.

Figura 24: Ambiente Google Colab



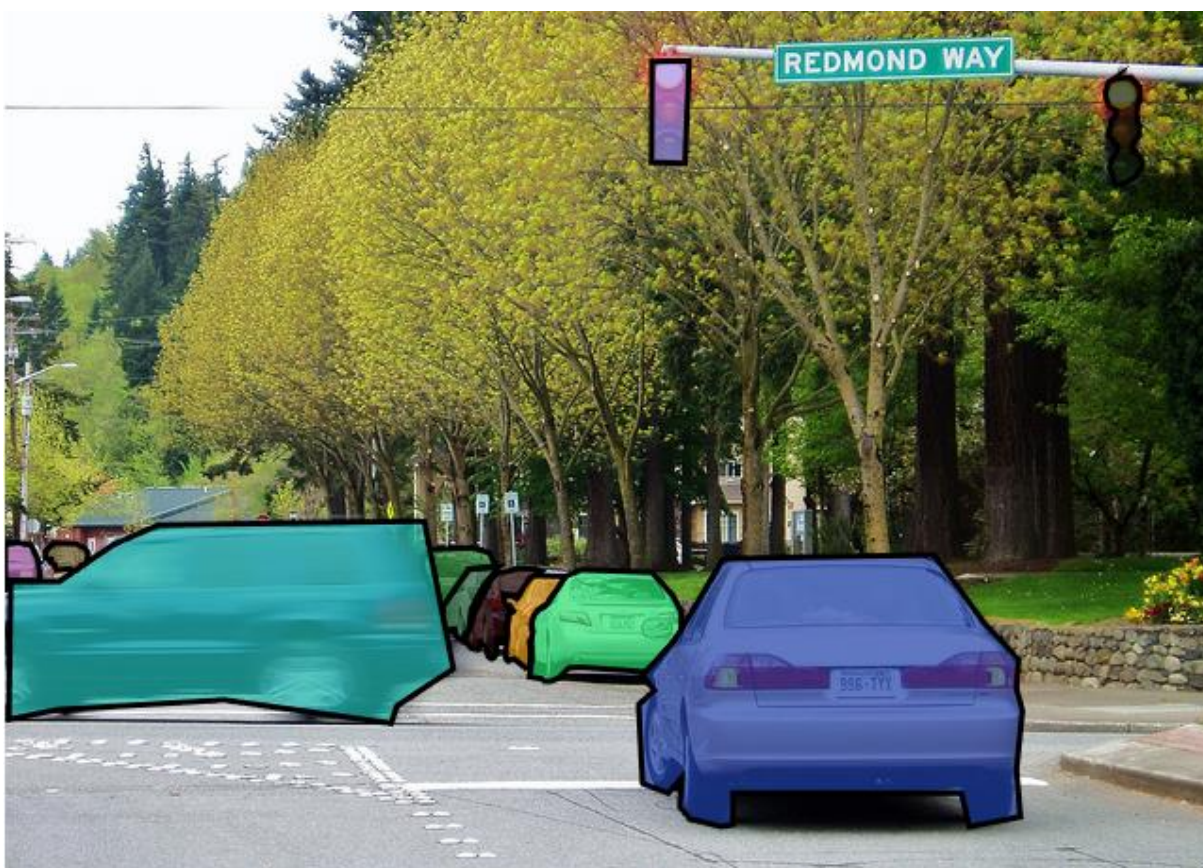
Fonte: Autor (2022)



O modelo de detecção apresentado será o *YOLO v4* que é implementado através de um framework para *deep learning* denominado *Darknet*.

O *framework* nada mais é que a estrutura da rede neural utilizada para execução do modelo *YOLO* tanto na etapa de treinamento quanto na de detecção. Através da importação deste *framework*, foram carregados os arquivos da estrutura do *YOLO* e para realização das detecções foi utilizado um modelo de pesos já pré-treinados com a base de dados *MS COCO*, que conta com mais de oitenta classes de objetos e um número de imagens superior a 330 mil arquivos, sendo que mais de 200 mil estão anotadas, conforme exemplo da imagem 25.

Figura 25: Exemplo de imagem anotada



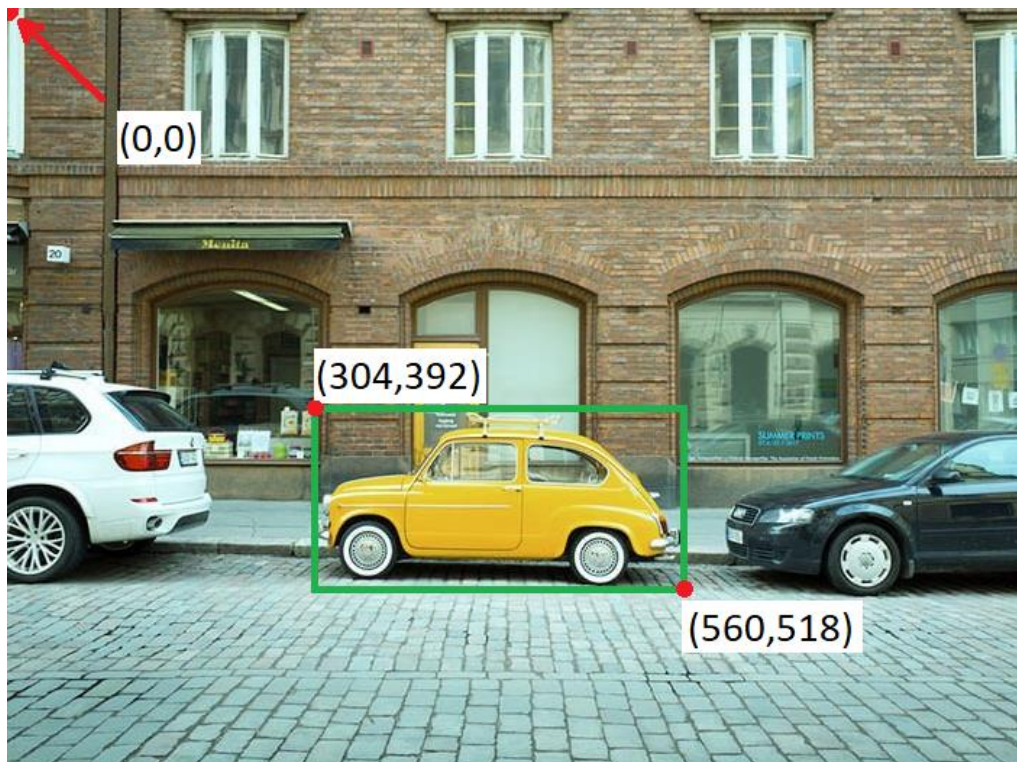
Fonte: COCO Dataset (2022).

Desse modo foi possível aplicar a detecção de objetos, sem a necessidade de executar as épocas de treinamento. Caso fosse necessário realizar o treinamento da rede neural seriam necessárias algumas etapas para que fossem feitas as especificações do treinamento, detalhes acerca das classes objetos que seriam

detectadas, bem como um número razoável de imagens anotadas, além da execução das épocas de treinamento.

A anotação da imagem, conforme visto na figura 26, nada mais é que a indicação das coordenadas onde o objeto se localiza, trazendo a posição em relação às dimensões bem como altura e largura.

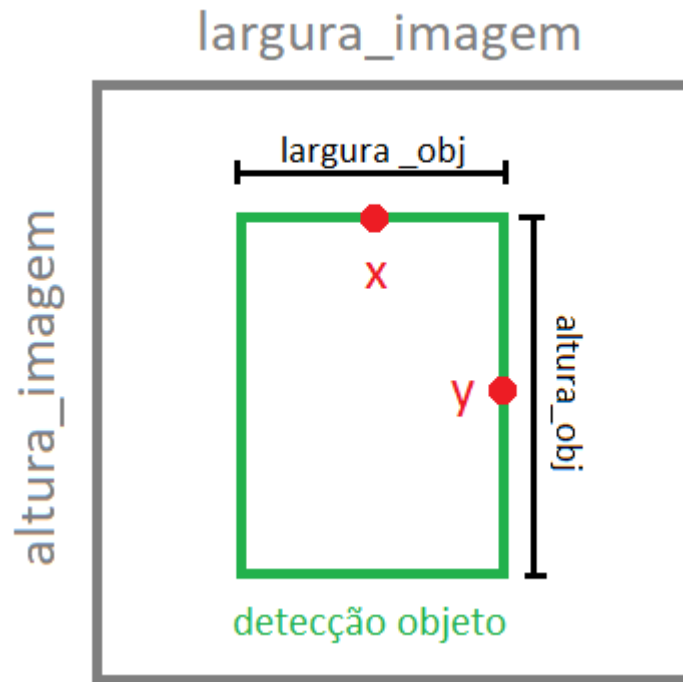
Figura 26: Anotações de classe



Fonte: Autor (2022).

A anotação para o treinamento do YOLO deve fornecer os pontos centrais relacionadas à altura e largura, bem como as os coeficientes entre a largura e altura da caixa delimitadora em relação a largura e altura da imagem, conforme visto na figura 27.

Figura 27: Padrão de anotação YOLO



Fonte: Autor (2022).

Tendo as coordenadas iniciais e finais de x e y, conforme visto na figura 25, pode-se converter a anotação para o padrão seguido pelo YOLO através das expressões abaixo.

$$x = \frac{\left(\frac{x_{final} - x_{inicio}}{2}\right) + x_{inicio}}{largura_{imagem}}$$

$$y = \frac{\left(\frac{y_{final} - y_{inicio}}{2}\right) + y_{inicio}}{altura_{imagem}}$$

$$largura = \frac{x_{final} - x_{inicio}}{largura_{imagem}}$$

$$altura = \frac{y_{final} - y_{inicio}}{altura_{imagem}}$$



Como o *dataset* utilizado já foi previamente treinado é possível verificar seu desempenho através de um conjunto de imagens anotadas as quais servirão de parâmetro para verificar se a detecção está de acordo com a anotação.

As imagens utilizadas para teste do algoritmo não fizeram parte do conjunto de dados de treinamento, ou seja, as imagens submetidas a rede nunca foram processadas por ela, uma vez que as captações foram todas feitas pelo autor.

Para realizar anotação das imagens foi utilizado um *software open-source*, denominado Labellmg, Figura 28, a caixa delimitadora é desenhada pelo usuário e então é criado o arquivo com a extensão .txt contendo a classe e coordenadas dos objetos presentes na imagem.

Figura 28: Labellmg software de anotação



Fonte: Autor (2022).

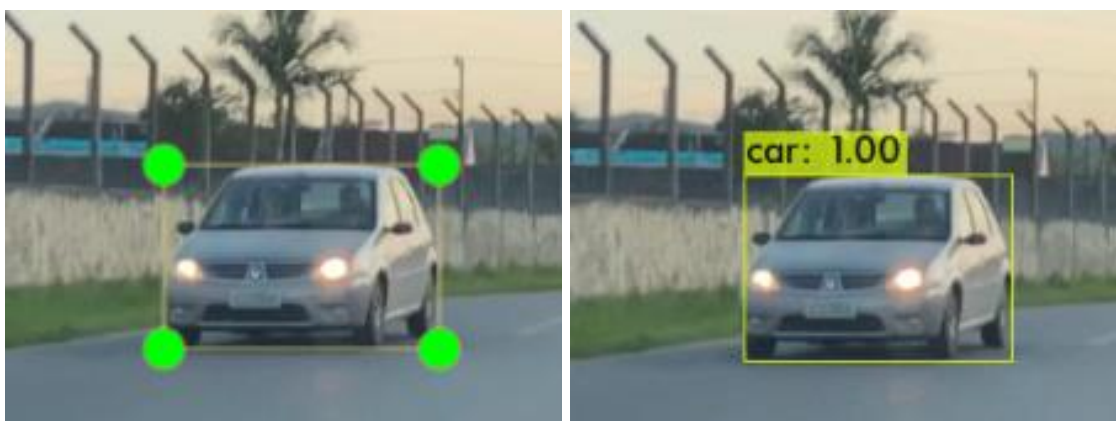
As imagens anotadas são fornecidas como padrões ouro para a rede na etapa de treinamento, mas podem ser utilizadas como ferramenta de comparação sendo possível avaliar de que modo ocorreram as detecções e se estão coerentes com as anotações.

Para captação das imagens foi utilizada a câmera de um dispositivo celular que possui 12 *megapixels*, resolução de 4032x3024, uma lente 28mm com a abertura *f* de 1.8.

## 4 ANÁLISE E DISCUSSÃO DOS RESULTADOS

Com o intuito de submeter o algoritmo às condições diferentes, foi elaborado um conjunto de imagens, variando as situações captadas, possibilitando de maneira qualitativa verificar o desempenho do algoritmo. Abaixo é demonstrada uma imagem anotada (esquerda) bem como o resultado do seu processamento na rede neural (direita).

Figura 29: Anotação feita (esq) e caixa delimitadora prevista pelo algoritmo(dir.)



Fonte: Autor (2022).

A imagem da figura 29 demonstra um exemplo simples onde o algoritmo identificou um objeto da classe *car* em uma caixa delimitadora muito semelhante a anotação feita e com um grau de certeza de 100%, a diferença entre caixa prevista e a anotada foi mínima, conforme demonstrado na tabela 1.

Tabela 1: Coordenadas de anotação vs. caixa delimitadora prevista.

| Coordenada | Anotada | Previsão Algoritmo |
|------------|---------|--------------------|
| X inicial  | 508     | 511                |
| Y inicial  | 272     | 271                |
| Largura    | 157     | 144                |
| Altura     | 98      | 101                |

Fonte: Autor (2022).

Para o mesmo caso foi proposta a imagem com outra orientação (invertida), figura 30, caso que raramente acontecerá na prática, mas para efeito de comparação desejou-se avaliar se haveria alterações na detecção, conforme visto a seguir.

Figura 30: Original (esq.) e caixa delimitadora prevista pelo algoritmo (dir.)

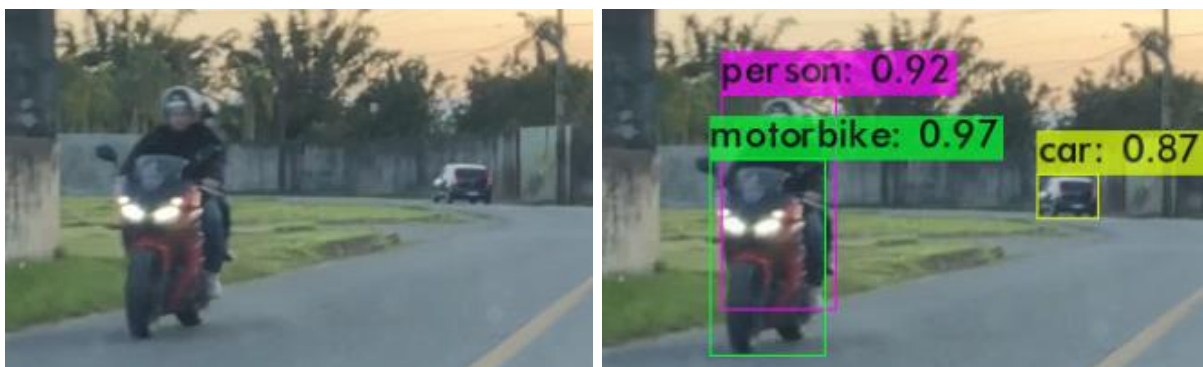


Fonte: Autor 2022.

Com alteração da orientação houve uma mudança na detecção, a confiança foi reduzida pela metade, porém a caixa delimitadora foi detectada corretamente. Do mesmo modo foi verificada mais uma vez o impacto da orientação na detecção, quando a imagem original foi rotacionada em 90°, o algoritmo não foi capaz de realizar a detecção.

Com o intuito de verificar quais seriam as detecções optou-se por utilizar uma imagem um pouco mais complexa contendo mais objetos a serem detectados e com certa proximidade entre eles.

Figura 31: Original (esq.) e caixa delimitadora prevista pelo algoritmo (dir.)



Fonte: Autor (2022).

Na figura 31 foi possível verificar que neste ambiente ocorreu a detecção das classes com altos valores de certeza, mesmo os objetos estando próximos (*motorbike / person*), além do veículo ao fundo.

Para outro caso foram avaliadas também condições em que a detecção é parcial, onde o objeto em questão não consta totalmente no *frame*, como visto a seguir.

Figura 32: Original (acima.) e caixa delimitadora prevista pelo algoritmo (abaixo.)



Fonte: Autor (2022).

Mesmo para uma detecção parcial do carro, como visto na Figura 32, obteve-se altos índices de confiança (99%).

Para outro caso foi avaliado até que ponto ocorrem as detecções mesmo com a oclusão do objeto a ser detectado, desse modo, foram feitos recortes na figura 33 com o intuito de avaliar a capacidade de detecção de objeto quando a presença é parcial.

*Figura 33: Imagem original para testes de oclusão*

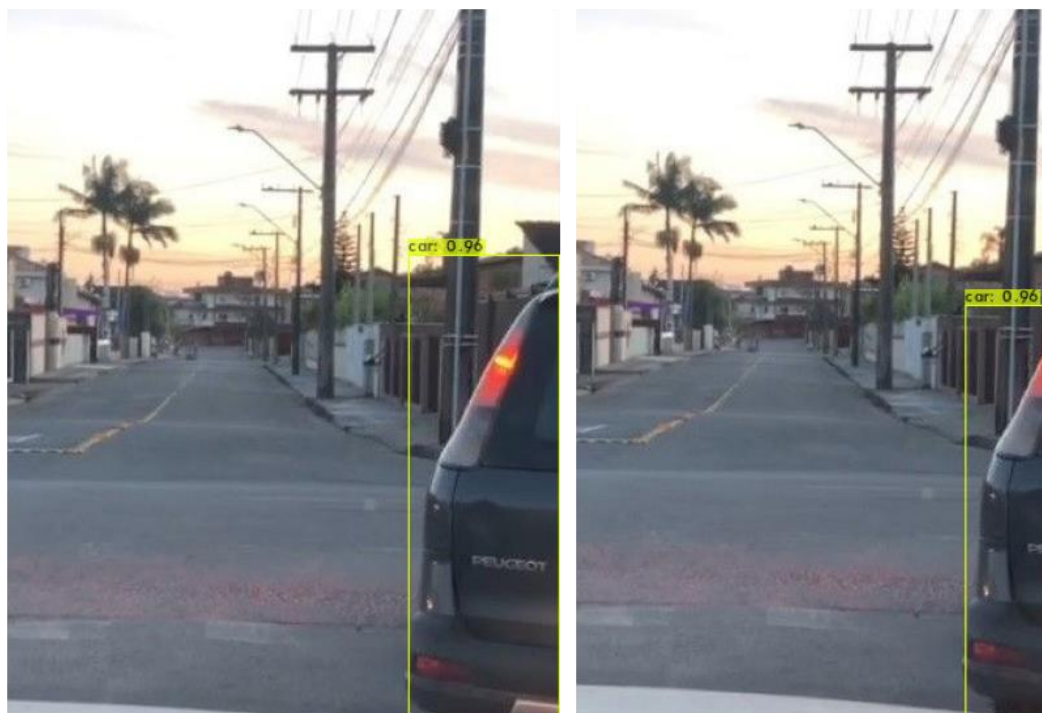


Fonte: Autor (2022).

Foram feitas 4 novas imagens a partir da original, cada uma delas com uma região cada vez menor contendo o veículo, o resultado pode ser observado nas figuras 34 e 35.

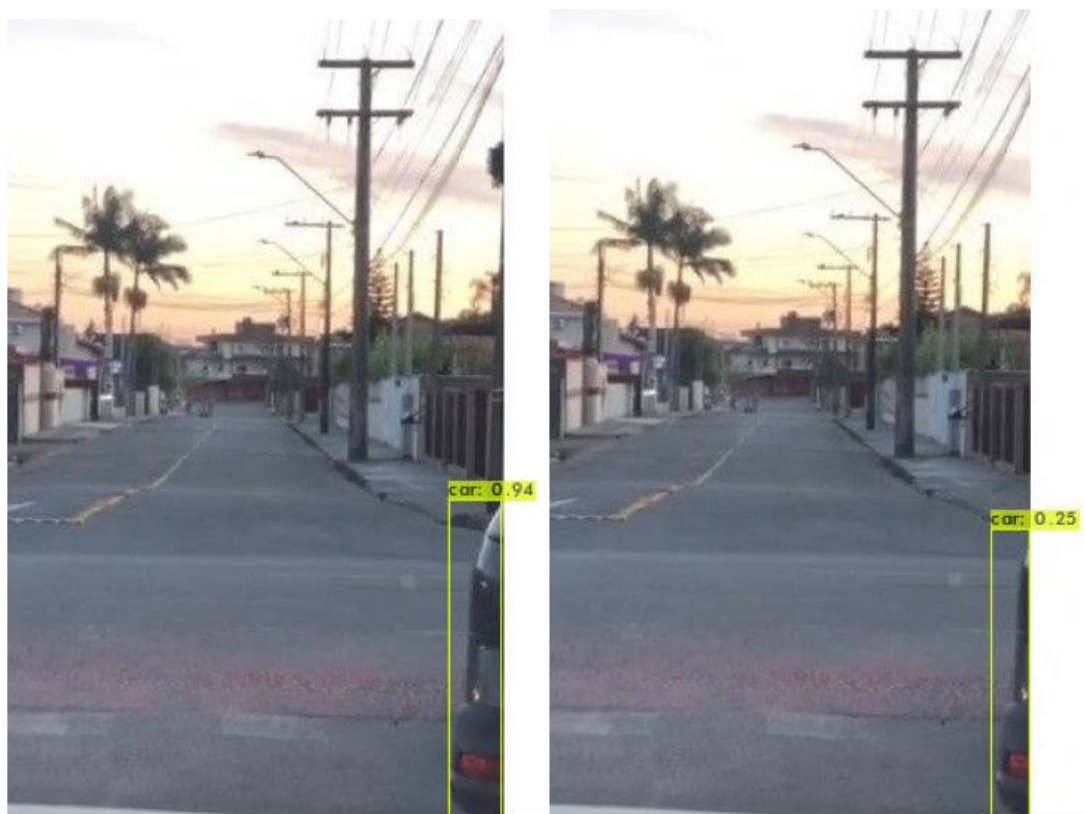


Figura 34: Primeira e segunda etapa de redução oclusão



Fonte: Autor (2022).

Figura 35: Terceira e quarta etapa de oclusão

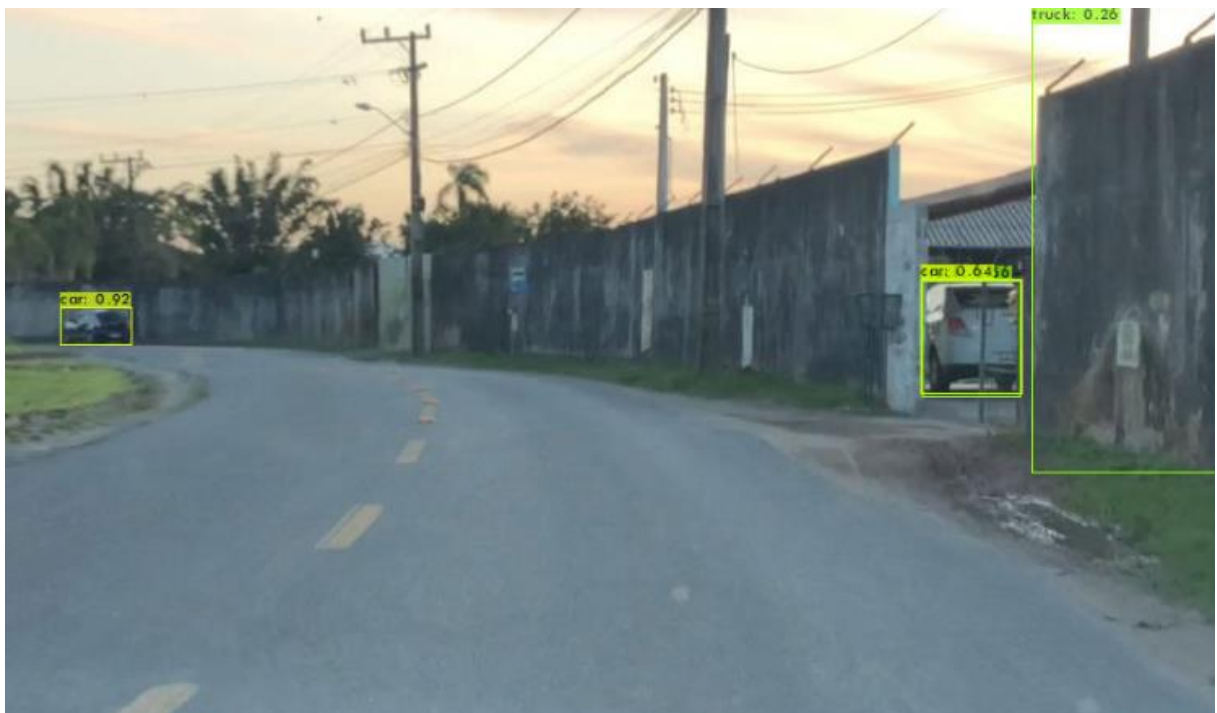


Fonte: Autor (2022).

Apesar da redução da confiabilidade no último caso (25%), observou-se uma capacidade de detectar o objeto mesmo com uma pequena parcela presente na imagem.

Em outros casos o desempenho foi demonstrou alguns desvios uma vez que a detecção fugiu totalmente no quesito área de detecção quanto de classe presente, conforme visto na figura 36.

Figura 36: Imagem com detecção fantasma de classe inexistente

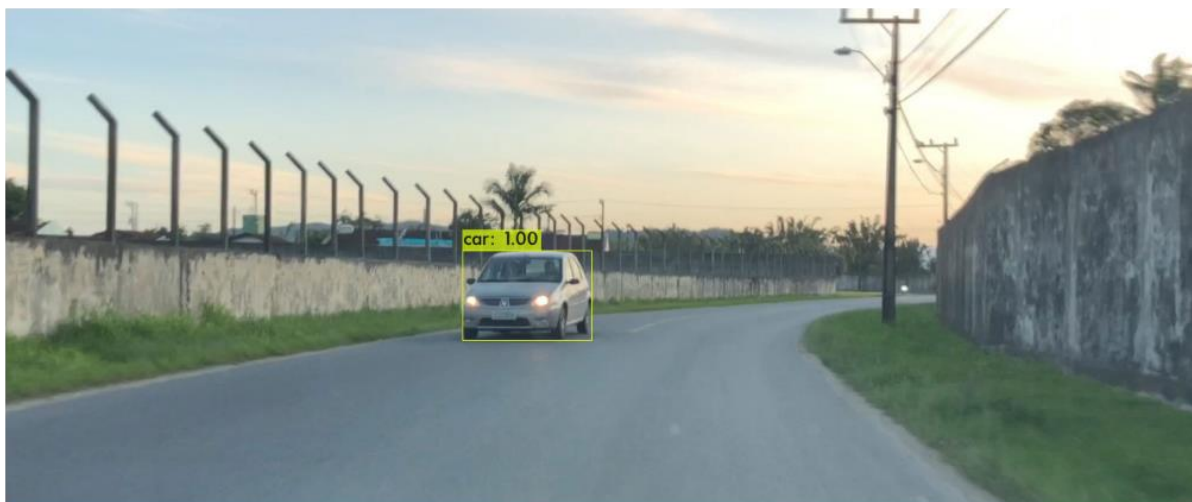


Fonte: Autor (2022).

A figura acima é um exemplo onde ocorreu a detecção dupla no veículo da direita em níveis de confiança moderados além de uma detecção “fantasma” da classe *truck*.

Além das condições apresentadas anteriormente foi considerado que em alguns casos podem ocorrer distorções nas imagens capturadas, como o desfoque do *frame*, com o intuito de avaliar esta condição a figura 37 foi alterada inserindo diferentes níveis de desfoque.

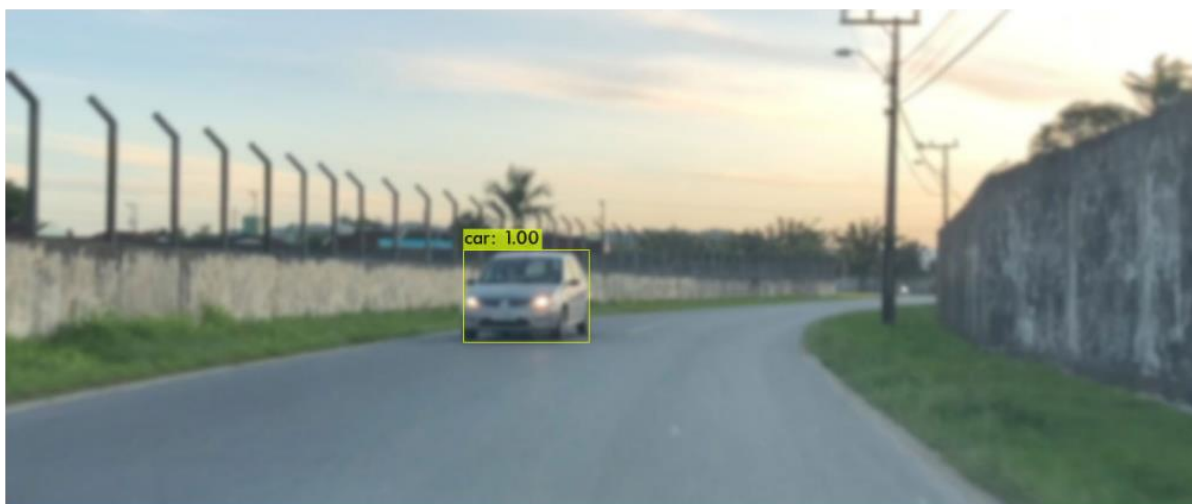
*Figura 37: Detecção original sem distorção*



Fonte: Autor (2022).

Inicialmente foi implementado um nível de desfoque com intensidade de 5%, na figura 38 pode-se observar a detecção sob estas condições.

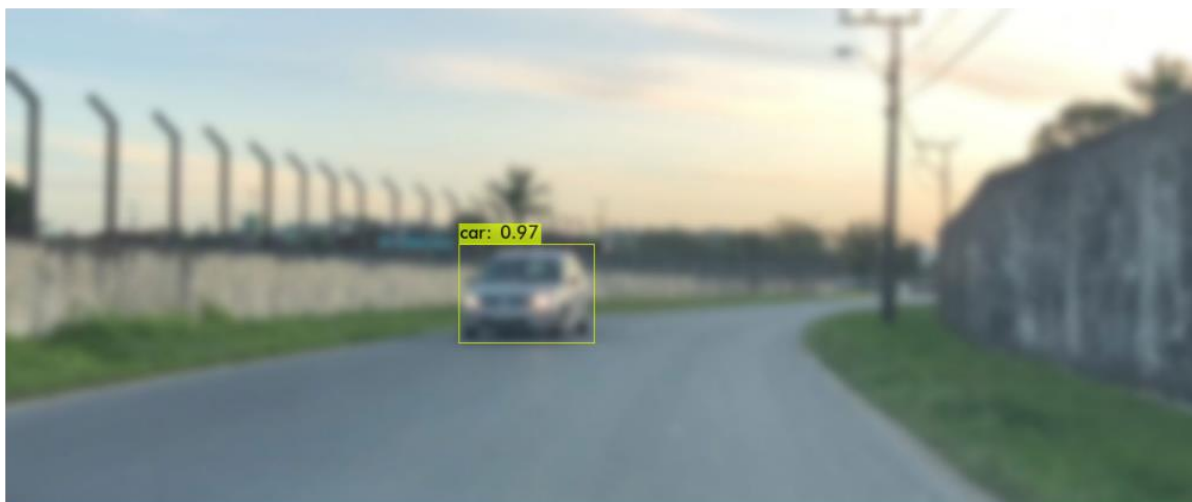
*Figura 38: Nível de desfoque em 5%*



Fonte: Autor (2022).

Não foram identificados impactos na detecção com este nível de desfoque, a intensidade foi incrementada passando a ser 10%, conforme visto a seguir na figura 39.

Figura 39: Nível de desfoque em 10%



Fonte: Autor (2022).

Nesta intensidade de desfoque a imagem apresenta uma pequena redução no nível de confiança, porém a caixa delimitadora foi detectada corretamente. Para mais uma etapa a intensidade de desfoque foi aumentada, passando a ser 15% conforme visto na figura 40.

Figura 40: Nível de desfoque em 15%

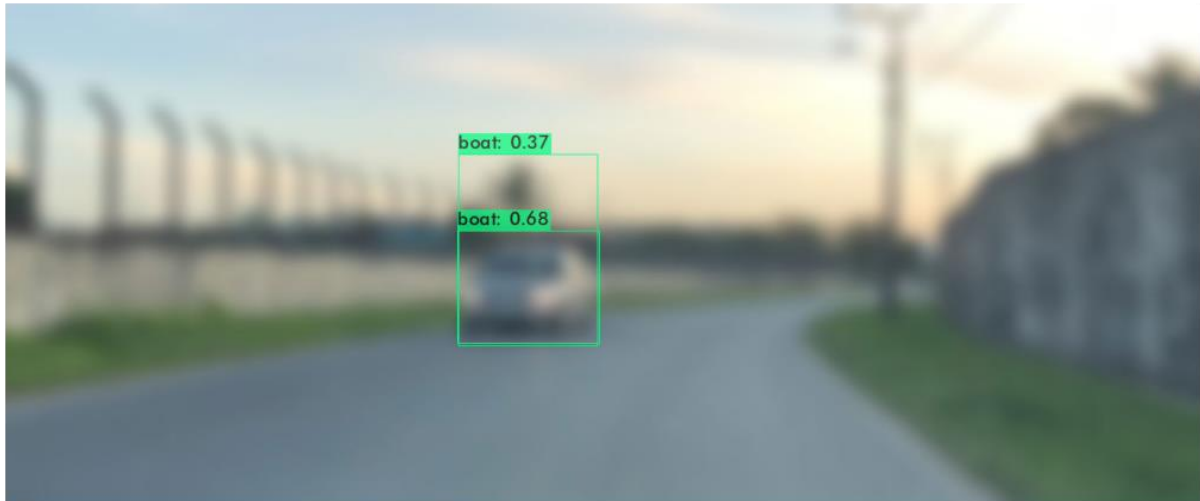


Fonte: Autor (2022).

Com a intensidade de desfoque de 15% foram feitas diferentes detecções de maneira duplicada, ou seja, no mesmo objeto foram identificadas classes diferentes, 71% de certeza para a classe *car* e 59% para a classe *truck*.

Para última etapa deste quesito foi verificado a resposta do algoritmo quando a intensidade do desfoque é aumentada para 20%, conforme figura 41.

Figura 41: Nível de desfoque em 20%



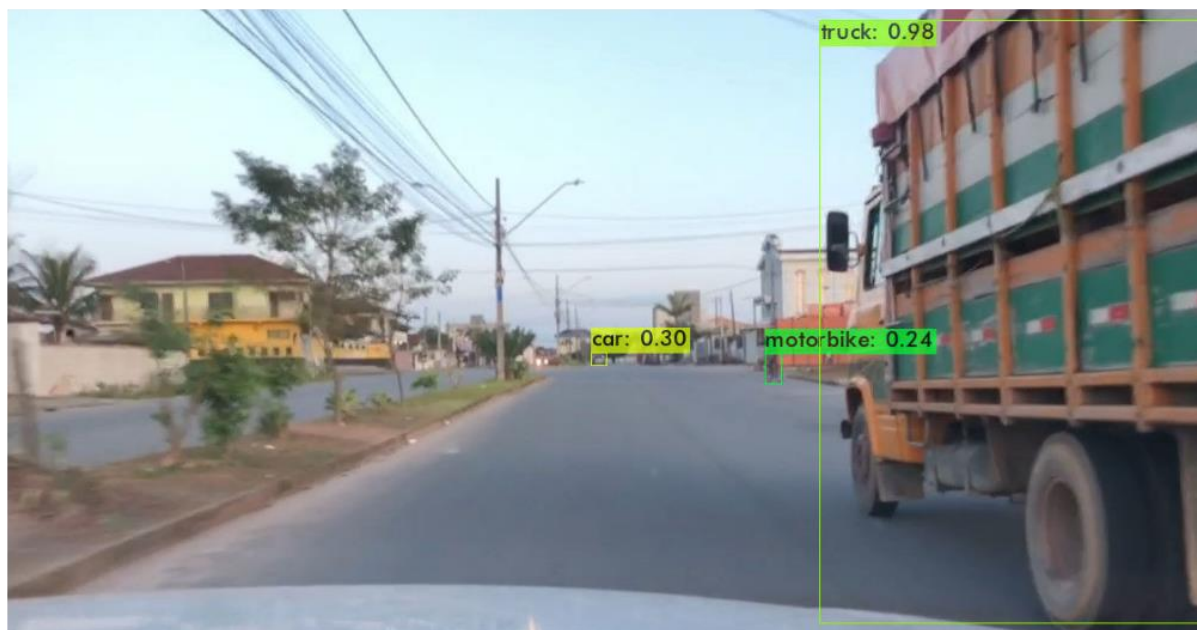
Fonte: Autor (2022).

Observaram-se detecções incorretas tanto da demarcação da caixa delimitadora quando na classe do objeto, foram identificadas duas caixas delimitadoras com as seguintes classes e confianças *boat* com 68% e *boat* com 37%. Desfoques acima de 20% implicaram em imagens em que não ocorreram nenhuma detecção. Desse modo identificou-se que certos níveis de desfoque são aceitáveis e promovem um baixo impacto da detecção.

Para outra caso desejou-se analisar se uma detecção em outra orientação seria validada e detectada corretamente, foi utilizado o exemplo de um caminhão, onde em uma das imagens a vista é lateral, mostrando a cabine e em outra a vista é traseira, estando a vista apenas a parte do baú, conforme figura.



Figura 42: Detecção lateral caminhão



Fonte: Autor (2022).

Na figura 42 foi observado que a detecção ocorreu com uma confiança de 98% da classe *truck* que era a de interesse. Para o mesmo veículo foi submetida uma imagem apenas da parte traseira, com o intuito de avaliar qual seria a detecção, sem que a cabine estivesse presente.

Figura 43: Detecção traseira caminhão

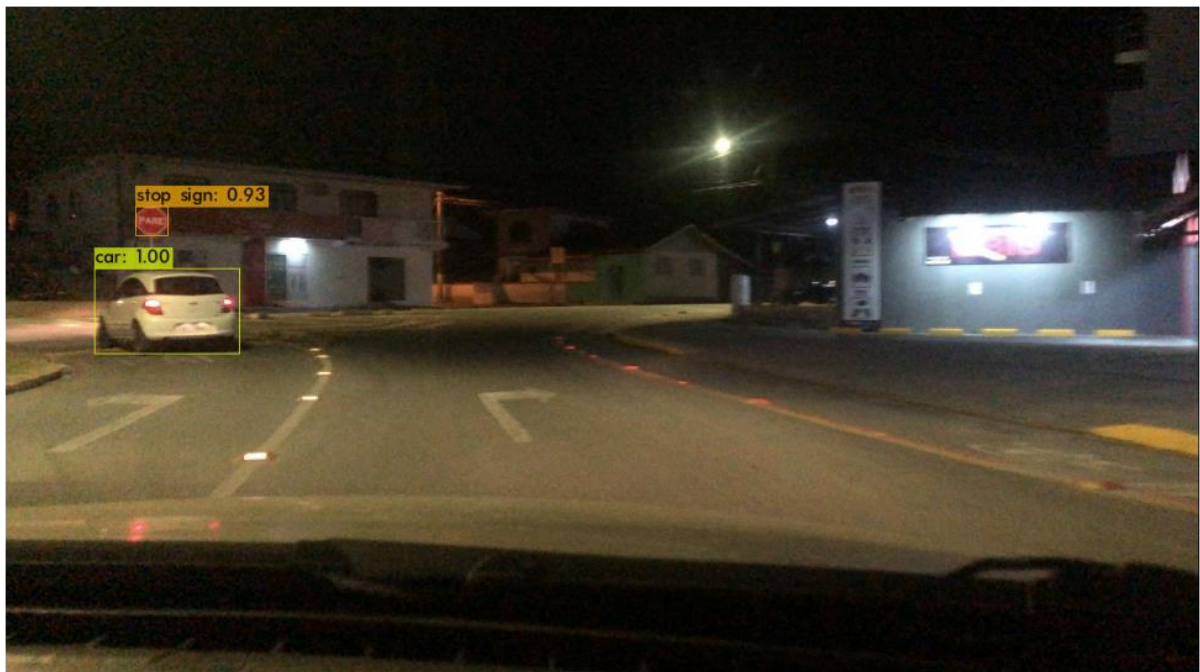


Fonte: Autor (2022).

Observou-se que mesmo com a omissão da cabine ocorreu a detecção classe de interesse de maneira correta na figura 43, tanto na classificação quanto na demarcação do objeto.

Outro ponto que desejou ser avaliado são as detecções noturnas onde o nível de luminosidade é reduzido e a iluminação dos outros veículos causa ofuscamento na imagem. A seguir será apresentado um exemplo de detecção feita em ambiente noturno, inicialmente optou-se por um cenário mais simples com poucos objetos conforme visto abaixo.

*Figura 44: Detecção noturna*



Fonte: Autor (2022).

Mesmo com a baixa luminosidade da captura vista na imagem 44 foi possível identificar as classes presentes com altos níveis de certeza a uma distância razoável.

Para um caso um pouco mais complexo foi fornecido ao algoritmo uma imagem onde o objeto a ser detectado estava em movimento, o que gerou uma certa distorção na imagem e o ambiente mais escuro conforme visto a seguir.

Figura 45: Ambiente noturno em movimento



Fonte: Autor (2022).

Mesmo com a distorção da imagem 45 a detecção foi feita corretamente com altos níveis de confiança.

Ainda com relação ao cenário noturno foi submetida ao algoritmo uma imagem com mais objetos e luminosidades variadas advindas dos veículos próximos.

Figura 46: Detecção noturna com luminosidade



Fonte: Autor (2022).



A figura 46 apresenta um cenário um pouco mais caótico onde há diferentes fontes de luz, fazendo com que algumas detecções sejam comprometidas, a exemplo do veículo a esquerda onde a detecção teve uma redução do nível de certeza (71%) e o veículo que está imediatamente atrás deste não foi identificado

Em sequência foi submetida mais uma imagem com múltiplas classes presentes em um ambiente noturno.

*Figura 47: Detecção noturna múltiplas classes*

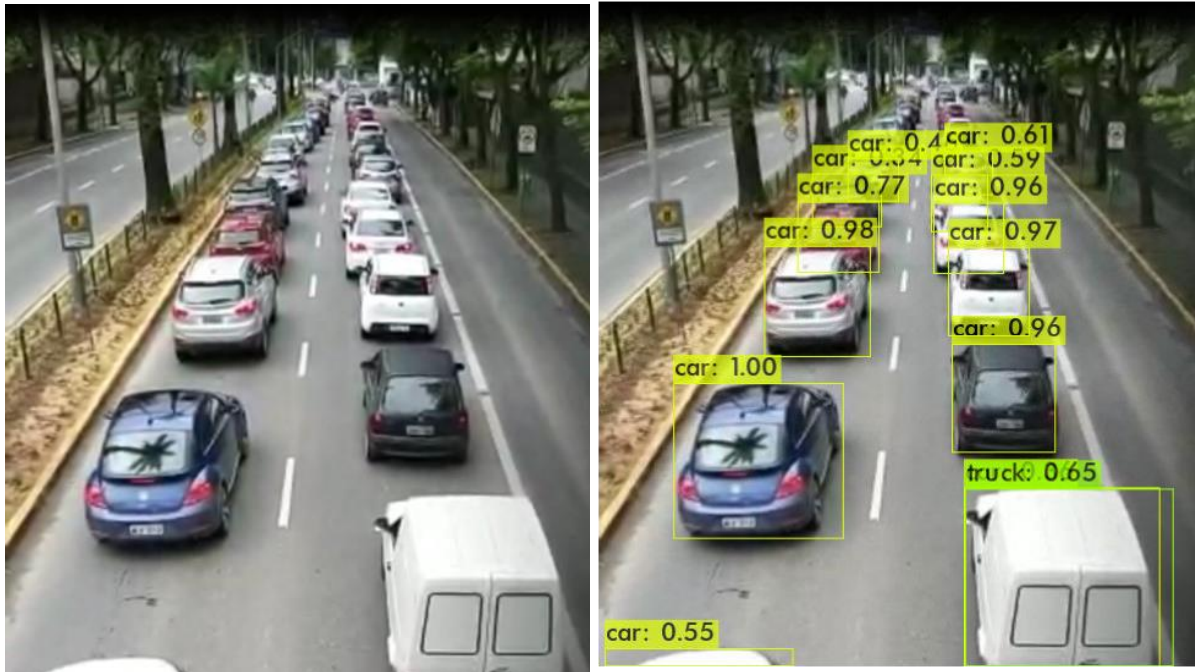


Fonte: Autor (2022).

Na figura 47 o cenário é semelhante ao da figura anterior, porém há a presença de mais classes na imagem. Mesmo com baixo destaque os pedestres eles foram detectados de maneira correta apesar dos baixos níveis de certeza.

Foi proposto também um ambiente com outra perspectiva de observação e com um volume maior de veículos, com o interesse na observação se todos os objetos que estão claramente nítidos seriam detectados.

Figura 48: Visão da via de outra perspectiva

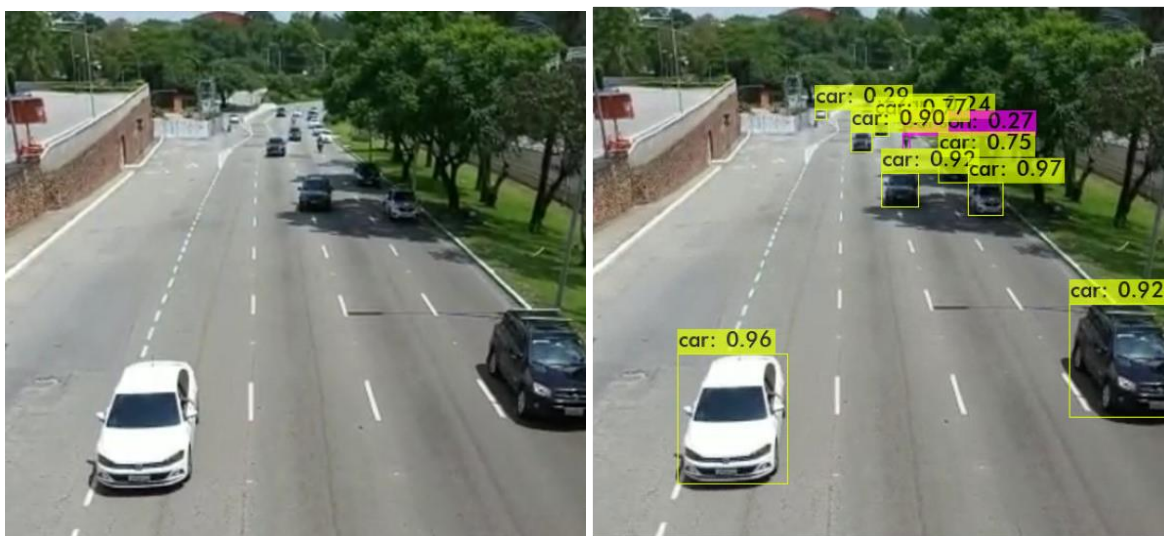


Fonte: Autor (2022).

Na figura 48 pode-se notar que os veículos foram detectados até um pouco mais da metade da via e com níveis de confiança decrescentes a medida que se distanciavam da captura, um destaque maior para as detecções que foram feitas mais próximas da borda inferior, uma parcela muito pequena do veículo a esquerda foi o suficiente para a classificação como a classe *car*, ponto que é positivo, porém, a direita um veículo utilitário obteve duas classificações (*car* e *truck*), sendo que uma delas foi incorreta.

Para outro caso pegou-se uma imagem mais distante com certos complicadores, como a presença de uma sombra a direita da imagem como pode ser observado a seguir

Figura 49: Vista panorâmica frontal



Fonte: Autor (2022).

O destaque para as detecções da figura 49 foram para os veículos a direita que se encontravam, a sombra e mesmo assim obtiveram altos níveis de confiança na detecção (92%, 97% e 75%), para o veículo a esquerda que está mais distante que mesmo com uma pequena representação na imagem foi identificado com 29% de certeza, entretanto a motocicleta presente na parte mais distante a direita foi classificada como uma pessoa.

Por fim, para avaliar o desempenho do modelo pré-treinado utilizado, foi montado um conjunto com 100 imagens anotadas através do *software Labellmg*, identificando os objetos e as respectivas classes presentes na imagem, forme visto na figura 50.

Figura 50: Parte do conjunto de imagens anotadas



Fonte: Autor (2022).

As imagens e os arquivos de anotação foram fornecidos ao algoritmo e realizada a avaliação do detector através do comando *detector map*, foram retornadas as métricas de desempenho do modelo conforme apresentado na tabela 2.

*Tabela 2: Avaliação do modelo de pesos pré treinado*

| Métrica      | Resultado |
|--------------|-----------|
| Precision    | 0,66      |
| Recall       | 0,93      |
| TP           | 278       |
| FP           | 144       |
| FN           | 22        |
| Média de IoU | 55,59%    |

Fonte: Autor (2022).

Observou-se que a precisão ficou em torno de 66%, isso significa que este modelo pré-treinado com este conjunto de imagens anotadas foi eficaz em fazer 66% de detecções corretas, já o recall ficou próximo a 93%, o que indica que foi capaz de detectar a classe correta quando feita a detecção. Já a média da intersecção sobre a união (IoU) indica a relação entre as caixas anotadas e as previstas esta métrica ficou em torno de 55,6%.

## 5 CONCLUSÃO

Considerando o elevado número de óbitos provenientes das rodovias, destaca-se que os veículos autônomos promoverão um grande avanço no quesito segurança, uma vez que o comportamento humano é o principal causador de acidentes, inibindo esta variável, haverá avanços e melhorias nestes indicadores.

Dentre as tecnologias hoje existentes para a implementação em veículos autônomos os algoritmos de visão possuem um alto potencial. Considerando que a área de desenvolvimento de programação tem tido grandes avanços, é aberto um horizonte para desenvolvimento de sistemas integrados. Em particular nos veículos autônomos estes sistemas servirão como balizadores de guiamento do veículo e como dispositivos de sistemas de segurança, uma vez que são capazes de identificar os mais variados objetos em um intervalo cada vez menor de tempo.

Há uma série de desafios a serem vencidos, pois o desenvolvimento de sistemas autônomos possui inúmeras variáveis, e neste quesito o aprendizado de máquina terá papel crucial fornecendo robustez ao sistema, tornando-o capaz de fazer previsões e identificar as mais variadas situações mesmo sem antes tê-las “vivenciado”.

Diferentes tecnologias têm sido empregadas para elaborar o veículo autônomo, sejam elas radares, LiDAR ou sistemas com câmeras, todas possuem o mesmo objetivo, dar ao sistema a capacidade de enxergar, replicando o mundo real de maneira virtual, para que seja possível as tomadas de ação.

A conectividade em alta velocidade advinda das telecomunicações como por exemplo da tecnologia 5G abrirá um leque antes inimaginável, ou seja, o processo de aprendizado dos veículos poderá ser aprimorado em tempo real, uma vez que esta coleta de dados do ambiente fornecerá estes dados como conteúdo para o treinamento do sistema, fazendo com que as decisões dos veículos sejam cada vez mais assertivas

A topologia do algoritmo YOLO demonstrou ser o caminho para estes sistemas de visão, onde o sistema de detecção é integrado totalmente em uma rede neural e o problema de classificação é reduzido a um problema de regressão, tornando o sistema cada vez mais veloz.

Como sugestão para trabalhos futuros seguem algumas linhas de pesquisa:

- Implementação sistemas embarcados com o YOLO integrado com o intuito de avaliar o desempenho deste processamento e se levantar as métricas obtidas durante a detecção;
- Realizar o treinamento de detectores personalizados de determinado objeto e avaliar o desempenho das épocas de treinamento nas detecções finais;
- Desenvolver um protótipo com o YOLO em um sistema integrado com atuação de saídas como por exemplo para a seleção de peças;

O trabalho proposto atingiu seus objetivos de maneira sucinta, uma vez que esclareceu diferentes conceitos básicos dentro da área de pesquisa e possibilitou verificar qual seria o comportamento do algoritmo estudado diante das diferentes condições do ambiente.

## REFERÊNCIAS

BRAIN, Marshall. **How Radar Works**. 2003. Disponível em: <https://science.howstuffworks.com/radar.htm>. Acesso em: 09 jul. 2022.

BRASIL. POLÍCIA RODOVIÁRIA FEDERAL. . **Dados acidentes**. 2022. Disponível em: <https://www.gov.br/prf/pt-br/acesso-a-informacao/dados-abertos/dados-abertos-acidentes>. Acesso em: 15 jun. 2022.

DALAL, N.; TRIGGS, B. **Histograms of oriented gradients for human detection**. In: 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05). [S.l.: s.n.], 2005. v. 1, p. 886–893 vol. 1.

DIGI-KEY'S EUROPEAN EDITORS. **Radar Sensing for Driverless Vehicles**. 2016. Disponível em: <https://www.digikey.com.br/en/articles/radar-sensing-for-driverless-vehicles>. Acesso em: 09 jul. 2022.

FU, K. S. *et al.* **Robotics: control, sensing, vision and intelligence**. Estados Unidos: McGraw-Hill, 1987.

FUJIYOSHI, Hironobu *et al.* Deep learning-based image recognition for autonomous driving. **Iatss Research**, [S.L.], v. 43, n. 4, p. 244-252, dez. 2019. Elsevier BV. <http://dx.doi.org/10.1016/j.iatssr.2019.11.008>.

GONZALEZ, Rafael C.; WOODS, Richard E.. **Processamento Digital de Imagens**. 3. ed. São Paulo: Pearson Prentice Hall, 2010.

LOWE, D. **Object recognition from local scale-invariant features**. In: Proceedings of the Seventh IEEE International Conference on Computer Vision. [S.l.: s.n.], 1999. v.2, p.1150–1157 vol.2.

MURPHY, Robin R.. **Introduction to AI robotics**. 2. ed. Cambridge: The Mit Press, 2019.

MUKHTAR, Amir *et al.* Vehicle Detection Techniques for Collision Avoidance Systems: a review. **Ieee Transactions On Intelligent Transportation Systems**, [S.L.], v. 16, n. 5, p. 2318-2338, out. 2015. Institute of Electrical and Electronics Engineers (IEEE). <http://dx.doi.org/10.1109/tits.2015.2409109>.

OZGUNER, U.; STILLER, C.; REDMILL, K.. Systems for Safety and Autonomous Behavior in Cars: the darpa grand challenge experience. **Proceedings Of The Ieee**, [S.L.], v. 95, n. 2, p. 397-412, fev. 2007. Institute of Electrical and Electronics Engineers (IEEE). <http://dx.doi.org/10.1109/jproc.2006.888394>.

ROYO, Santiago; BALLESTA-GARCIA, Maria. An Overview of Lidar Imaging Systems for Autonomous Vehicles. **Applied Sciences**, [S.L.], v. 9, n. 19, p. 4093, 30 set. 2019. MDPI AG. <http://dx.doi.org/10.3390/app9194>

PADILLA, R.; NETTO, S. L.; DA SILVA, E. A. B. **A survey on performance metrics for object-detection algorithms**. In: 2020 International Conference on Systems, Signals and Image Processing (IWSSIP). [S.l.: s.n.], 2020. p. 237–242.

PADILLA, Rafael; PASSOS, Wesley L.; DIAS, Thadeu L. B.; NETTO, Sergio L.; SILVA, Eduardo A. B. da. A Comparative Analysis of Object Detection Metrics with a Companion Open-Source Toolkit. **Electronics**, [S.L.], v. 10, n. 3, p. 279, 25 jan. 2021. MDPI AG. <http://dx.doi.org/10.3390/electronics10030279>.

PULLI, Kari *et al.* Real-time computer vision with OpenCV. **Communications Of The Acm**, [S.L.], v. 55, n. 6, p. 61-69, jun. 2012. Association for Computing Machinery (ACM). <http://dx.doi.org/10.1145/2184319.2184337>.

REDMON, Joseph; DIVVALA, Santosh; GIRSHICK, Ross; FARHADI, Ali. **You Only Look Once: unified, real-time object detection**, [S.L.], jun. 2015. ArXiv. <http://dx.doi.org/10.48550/ARXIV.1506.02640>.



REDMON, J.; FARHADI, A. **Yolo9000: Better, faster, stronger**. 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Jul 2017. Disponível em: <<https://ieeexplore.ieee.org/document/8100173>>.

REDMON, J.; FARHADI, A. Yolov3: An incremental improvement. CoRR, abs/1804.02767, 2018. Disponível em: <<http://arxiv.org/abs/1804.02767>>.

SHARIFI, Amir *et al.* A deep learning based hazardous materials (HAZMAT) sign detection robot with restricted computational resources. **Machine Learning With Applications**, [S.L.], v. 6, p. 100104, dez. 2021. Elsevier BV.

SOCIETY OF AUTOMOTIVE ENGINEERS INTERNATIONAL. **SAE J3016: SURFACE VEHICLE RECOMMENDED PRACTICE**. Canada, 2021.

SOUZA, Taciana; CORREIA, Suzete. **ESTUDO DE TÉCNICAS DE REALCE DE IMAGENS DIGITAIS E SUAS APLICAÇÕES**. In: CONGRESSO DE PESQUISA E INOVAÇÃO DA REDE NORTE NORDESTE DE EDUCAÇÃO TECNOLÓGICA, 2., 2007, João Pessoa. 2007. p. 1-11.

TESLA. **Autopilot and Full Self-Driving Capability**. Disponível em: <https://www.tesla.com/support/autopilot>. Acesso em: 02 nov. 2022.

TSUCHIYA, Chikao *et al.* Real-Time Vehicle Detection using a Single Rear Camera for a Blind Spot Warning System. **Sae International Journal Of Passenger Cars - Electronic And Electrical Systems**, [S.L.], v. 5, n. 1, p. 146-153, 16 abr. 2012. SAE International. <http://dx.doi.org/10.4271/2012-01-0293>.

VIANNA, Marcela Esteves. **Calibração de Sistemas de Visão Computacional para Aplicação em Automação e Robótica**. 2009. 98 f. Dissertação (Mestrado) - Curso de Engenharia Mecânica, Pontifícia Universidade Católica de Minas Gerais, Belo Horizonte, 2009.

VIOLA, P.; JONES, M.. Rapid object detection using a boosted cascade of simple

features. **Proceedings Of The 2001 Ieee Computer Society Conference On Computer Vision And Pattern Recognition. Cvpr 2001**, [S.L.], p. 1-9, dez. 2001. IEEE Comput. Soc. <http://dx.doi.org/10.1109/cvpr.2001.990517>.

WEI, Daniel Chin Min *et al.* VEÍCULOS AUTÔNOMOS: CONCEITOS, HISTÓRICO E ESTADO-DA-ARTE. 2013. **XXVII Congresso de Pesquisa e Ensino em Transportes**, Belém, nov. 2013.

WENDLING, Marcelo. **Sensores**. 2010. 20 f., Universidade Estadual Paulista, Guaratinguetá, 2010.

WETMORE, Jameson M.. Systems for Safety and Autonomous Behavior in Cars: The DARPA Grand Challenge Experience. **Automotive History Review**, Whashington, v. 7, n. 1, p. 4-19, jan. 2003.

WHO, World Health Organization (org.). **Global status report on road safety 2018**. France: Paprika, 2018.