Heisthen Mazzei Scarparo

Estudo comparativo de detecção e rastreamento de elementos no trânsito utilizando imagens ominidirecionais

Brasil

2024

Heisthen Mazzei Scarparo

Estudo comparativo de detecção e rastreamento de elementos no trânsito utilizando imagens ominidirecionais

Dissertação apresentada ao Programa de Pós-Graduação em Engenharia Elétrica do Centro Tecnológico da Universidade Federal do Espírito Santo, como requisito parcial para obtenção do Grau de Mestre em Engenharia Elétrica.

Universidade Federal do Espírito Santo

Programa de Pós-Graduação em Engenharia Elétrica

Orientadora Raquel Frizera Vassallo

Brasil

2024

Ficha catalográfica disponibilizada pelo Sistema Integrado de Bibliotecas - SIBI/UFES e elaborada pelo autor

---

Scarparo, Heisthen Mazzei, 1991-
S285e Estudo comparativo de detecção e rastreamento de elementos no trânsito utilizando imagens ominidirecionais / Heisthen Mazzei Scarparo. - 2024.
69 p. : il.

Orientador: Raquel Frizera Vassallo.
Dissertação (Mestrado em Engenharia Elétrica) - Universidade Federal do Espírito Santo, Centro Tecnológico.

1. Detecção de objetos de trânsito. 2. Rastreamento de objetos de trânsito. 3. Dataset de objetos de trânsito com imagens Ominidirecionais. 4. Cidades Inteligentes. I. Vassallo, Raquel Frizera. II. Universidade Federal do Espírito Santo. Centro Tecnológico. III. Título.

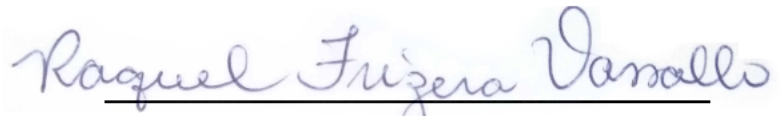
CDU: 621.3

Heisthen Mazzei Scarparo

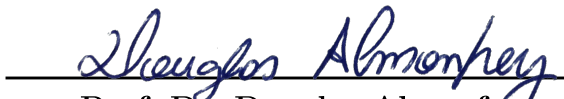
Estudo comparativo de detecção e rastreamento de elementos no trânsito utilizando imagens ominidirecionais

Dissertação apresentada ao Programa de Pós-Graduação em Engenharia Elétrica do Centro Tecnológico da Universidade Federal do Espírito Santo, como requisito parcial para obtenção do Grau de Mestre em Engenharia Elétrica.

Trabalho aprovado. Brasil, 22 de Novembro de 2024:



Profa. Dra. Raquel Frizera Vassallo
Universidade Federal do Espírito Santo
Orientadora



Prof. Dr. Douglas Almonfrey
Instituto Federal de Educação, Ciência e Tecnologia do Espírito Santo - Vitória
Examinador Externo

Prof. Dr. Daniel Cruz Cavalieri
Instituto Federal de Educação, Ciência e Tecnologia do Espírito Santo - Serra
Examinador Externo

Brasil
2024

Agradecimentos

Agradeço primeiramente a Deus e Nossa Senhora da Penha, que me dão forças para prosseguir na caminhada. Aos meus pais, Débora e Elias, que não mediram esforços, amor e dedicação para minha evolução como estudante e pessoa. A minha esposa Jeniffer, pelo amor, companherismo e por não me deixar vacilar nos momentos mais difíceis. Aos meus familiares, amigos e colegas de trabalho pelo companheirismo e incentivo. Por fim e não menos importante, aos meus professores, em especial à Raquel, por pavimentar a estrada do saber que culmina nesse trabalho.

Meu muito obrigado, de coração. Amo vocês!

Heisthen Mazzei Scarparo

Resumo

A detecção e rastreamento de elementos no trânsito desempenham um papel fundamental no avanço das cidades inteligentes. Essas tecnologias têm o potencial de aliviar o congestionamento, otimizar o uso de recursos e melhorar a qualidade de vida da população. No entanto, um aspecto ainda pouco explorado nesse campo é a utilização de vídeos omnidirecionais, que proporcionam um campo visual de 360°.

As imagens omnidirecionais oferecem uma perspectiva abrangente do ambiente viário, permitindo uma análise mais completa do tráfego e dos objetos em movimento. Essa visão panorâmica possibilita a detecção de veículos, pedestres, ciclistas e outros elementos em todas as direções, incluindo ângulos difíceis de capturar com câmeras convencionais. O uso desse tipo de imagens para o controle semafórico facilita a obtenção de informações da trajetória dos veículos em tempo real e, portanto, a configuração dos semáforos de forma mais inteligente e eficiente.

Além disso, as imagens omnidirecionais podem ser usados para monitorar áreas de alta densidade de tráfego, identificar pontos de congestionamento e analisar padrões de comportamento dos usuários da via. Essas informações são valiosas para o planejamento urbano, o desenvolvimento de políticas de mobilidade e a implementação de estratégias que visem melhorar o fluxo de tráfego e a segurança nas ruas. Embora o uso de imagens panorâmicas em 360° no contexto da detecção e rastreamento no trânsito ainda seja um campo pouco explorado, ele representa uma boa ferramenta para o avanço das cidades inteligentes através da sua integração com os sistemas de controle semafórico e de gestão de tráfego nas cidades.

Nesse sentido, esse trabalho apresenta uma base de dados contendo 25 vídeos em 360°, com suas respectivas anotações. Tal base de dados está disponível para utlização pela comunidade acadêmica. Também apresenta um estudo comparativo entre aplicação das redes YOLOv5, YOLOv7 e YOLO-NAS, juntamente com o uso do algoritmo DEEPSORT, para detecção e rastreamento dos objetos de trânsito, presentes na base de dados. Para realizar a comparação entre as redes, foram utilizadas as métricas de Precisão, Revocação, F1-Score, mAP@.5 e mAP@.5:.95. No estudo aqui realizado, o melhor resultado foi obtido utilizando a rede YOLOv7 com treinamento. Tal estudo mostra a viabilidade de se considerar o uso de imagens omnidirecionais como uma ferramenta na tarefa de monitoramento de tráfego e auxiliar na mobilidade urbana.

Palavra-chave: Estudo comparativo; Imagens Ominidirecionais; Rastreamento; Detecção; YOLO; Deepsort; Dataset.

Abstract

Traffic detection and tracking play an important role in the context of smart cities. These technologies have the potential to alleviate congestion, optimize the use of resources, and improve the quality of life of the population. However, one aspect of this field that has not yet been explored is the use of omnidirectional videos, which provide a 360° field of view.

Omnidirectional images offer a large field of view of the road environment, allowing for a more complete analysis of traffic and moving objects. This panoramic view makes it possible to detect vehicles, pedestrians, cyclists, and other elements in all directions, including angles that are difficult to capture with conventional cameras. Using this type of imagery for traffic light control makes it easier to obtain information on the trajectory of vehicles in real time and, therefore, configure traffic lights in a more intelligent and efficient way.

In addition, omnidirectional images can be used to monitor areas of high traffic density, identify congestion points, and analyze road user behavior patterns. This information is valuable for urban planning, the development of mobility policies, and the implementation of strategies aimed at improving traffic flow and street safety. Although the use of 360° panoramic images in the context of traffic detection and tracking is still an underexplored field, it represents a good tool for the implementation of smart cities through its integration with traffic light control and traffic management systems in cities.

In this context, this work presents a database containing 25 panoramic videos, with their respective annotations. This database is available for use by the academic community. It also presents a comparative study between the application of the YOLOv5, YOLOv7, and YOLO-NAS networks, together with the use of the DEEPSORT algorithm, for detection and tracking of traffic objects present in the database. To compare the networks, the metrics of Precision, Recall, F1-Score, mAP@.5, and mAP@.5:.95 were used. In this study, the best result was obtained using the YOLOv7 network with training. Such result shows the feasibility of considering the use of omnidirectional images as a tool in the task of traffic monitoring and helping provide urban mobility.

Keywords: Comparative study; Omnidirectional images; Tracking; Detection; YOLO; Deepsort; Dataset.

Lista de ilustrações

Lista de tabelas

Lista de quadros

Sumário

1 Introdução

De acordo com o Relatório Mundial das Cidades 2022, publicado pelo ONU-Habitat, mais da metade da população do planeta (55%) vive em cidades, com uma previsão de que este valor chegue a 68% até 2050. Os centros urbanos seguem crescendo, mesmo com a desaceleração da urbanização observada em razão da pandemia do COVID-19. Neste período, razões como a busca por lugares mais afastados, trabalho na modalidade de *home office* e, em muitos casos, a necessidade de se manter as crianças em casa, motivaram muitas migrações das grandes cidades para regiões do interior ou para cidades menores, em busca de mais segurança sanitária e qualidade de vida devido ao isolamento (AL, 2022).

Mesmo assim, a população urbana segue em crescimento e o término da pandemia tende a trazer de volta o fluxo para os grandes centros, onde bilhões de pessoas, cada dia mais, são afetadas de diversas formas por problemas de infraestrutura, segurança, mobilidade, poluição e acesso limitado a recursos.

A partir do desenvolvimento tecnológico atual, acredita-se que, tornando as cidades mais inteligentes, será possível melhorar os serviços urbanos e aumentar a qualidade de vida da população (KON FáBIO; SANTANA, 2017).

Cidades inteligentes podem ser definidas como cidades que utilizam a tecnologia para promover o desenvolvimento de forma sustentável e eficiente (LAZZARETTI K., 2019). Nesse contexto, em uma considerável parte dos projetos de cidades inteligentes é comum o uso de visão computacional como forma de sensoriamento, tanto em ambientes públicos como privados, principalmente para monitoramento, controle de tráfego e segurança pública.

De acordo com PINHEIRO Alex; DA SILVA (2018), o uso de visão computacional em mobilidade urbana envolve tecnologias e conhecimentos em áreas multidisciplinares, objetivando o desenvolvimento de ferramentas que reconheçam certos cenários, objetos ou mesmo características de seus contextos. Isso contribui para o desenvolvimento de sistemas que buscam tomar decisões e ações que melhorem a mobilidade, a segurança e a relação entre os diferentes agentes de uma cidade. Dentro do contexto de mobilidade, destaca-se a temática do controle de tráfego, o qual será o objeto de estudo neste trabalho. Neste sentido, uma das utilizações da visão computacional inclui o reconhecimento de veículos (carros, motocicletas, ônibus e caminhões), ciclistas e pedestres, e a utilização dessa informação em sistemas que possam gerar maior fluidez no trânsito e, consequentemente, maior segurança e eficiência.

No Brasil, as dificuldades de mobilidade urbana geram prejuízos econômicos e grande impacto socioambiental. Estudos mostram que cerca de 9 milhões de brasileiros

demoram mais de uma hora para chegar ao trabalho e a situação se agrava a cada dia, o que gera uma perda de R$ 267 bilhões por ano com congestionamentos, representando, por exemplo, quase 4% de todo o Produto Interno Bruto (PIB) do país em 2017. Além disso, os acidentes de trânsito no Brasil custam mais de R$ 19 bilhões por ano, valor superior ao PIB de 11 capitais (GORGULHO CRISTIANE FERNANDES; TREDINNICK, 2020). Considerando exclusivamente esse dado econômico, tais valores poderiam ser muito melhor investidos em conforto para a sociedade.

Atualmente, muitos sistemas de monitoramento de tráfego estão em implantação, porém normalmente são utilizadas câmeras convencionais com campo de visão limitado. Uma alternativa muito mais interessante, seria a utilização de imagens panorâmicas, obtidas por meio de câmeras ou lentes especiais, que representam uma opção mais ampla e com menor uso de banda de transmissão, devido ao seu campo visual ampliado e utilização de uma única imagem para retratar todo o ambiente.

Essas imagens são construídas principalmente para retratar um ambiente com maior ângulo de visão e foram muito utlizadas em trabalhos científicos, entre os anos de 1990 e 2010, para navegação em robótica, quando processar imagens panorâmicas de uma única câmera representava uma alternativa muito interessante, devido ao menor custo computacional para processamento, considerando a capacidade dos computadores da época. Além disso, essa alternativa também representava menor custo financeiro no momento de se equipar as plataformas robóticas com sensores (TOšIć; FROSSARD, 2009), (SCHROEDER, 2000), (ISHIGURO; UEDA; TSUJI, 1993), (VASSALLO; SCHNEEBELI; SANTOS-VICTOR, 2000) (VLASSIS et al., 2001), (CHEN et al., 2007), (GAVA et al., 2007), (ANDRUSSOW et al., 2023). Mais recentemente, imagens panorâmicas vêm sendo utilizadas, por exemplo, na construção de cenários de jogos ou em simuladores, principalmente em jogos de aviação ou de corrida (HIRAGA ALAN KAZUO; DA SILVA, 2013).

As primeiras imagens panorâmicas, foram obtidas em 1839 (CURTIN, 2004), quando elas eram formadas por meio de várias fotos. Tais fotos eram reveladas em papel e depois recortadas e coladas na sequência que foram capturadas, surgindo o termo *Stitching* de imagens ou costura de imagens. Atualmente a construção de imagens panorâmicas é muito mais simples, feita principalmente por meio de lentes especiais e software para a sua planificação (HIRAGA ALAN KAZUO; DA SILVA, 2013), ampliando sua utilização para muito além de jogos ou simulações.

Desta forma, a utilização de imagens panorâmicas com 360° de campo visual podem amplificar também a região vista por um sistema de mobilidade urbana e facilitar a realização da tarefa de monitoramento e controle de tráfego. Essas imagens, afinal, permitem a identificação de mais objetos e pessoas ao mesmo tempo, e possuem um custo de instalação, tráfego de banda e custo de processamento menor do que a utilização de

várias câmeras para cobrir o mesmo espaço (AZEVEDO et al., 2020). Logo, é fácil concluir que o uso de imagens panorâmicas pode contribuir no processo de se automatizar as cidades e torná-las cada vez mais inteligentes, apresentando melhor mobilidade urbana, segurança e eficiência. Com esta motivação, este trabalho focará na utilização de imagens panorâmicas com 360° em tarefas de mobilidade urbana e monitoramento de tráfego.

As vias monitoradas correspondem a uma área que engloba duas avenidas e um calçadão à beira mar. A razão para escolha desta região se deve a intenção de aumentar o número de pedestres e ciclistas presentes nas cenas, diversificando os elementos de trânsito a serem detectados e rastreados.

Assim, considerando-se esse ambiente de litoral e calçadão, um estudo comparativo para a detecção, identificação e rastreamento de elementos no trânsito, utilizando imagens 360°, será apresentado como uma abordagem para monitoramento do fluxo de veículos, a qual poderá ser usada para controle de tráfego, contribuindo para tornar cidades mais inteligentes e sustentáveis.

1.1 Objetivos

O principal objetivo deste trabalho de dissertação é realizar um estudo comparativo entre detectores de objetos no trânsito (pedestres, ciclistas e veículos), a partir de um conjunto de vídeos panorâmicos, utilizando para isso uma câmera 360°. Tais vídeos foram capturados em vias públicas próximas ao litoral, onde há a presença de um calçadão, o que favorece o aparecimento de mais ciclistas e pedestres, com o intuito de aumentar o número de diferentes elementos do trânsito presentes nas imagens. Além da detecção dos objetos, também foi realizado o seu seguimento aplicando-se um único modelo de rastreador. A sua escolha foi baseada no bom desempenho do mesmo em várias situações diferentes.

A câmera utilizada foi do tipo Rico Theta (https://theta360.com) e a metodologia aplicada neste trabalho está descrita no Capítulo 3.

Para alcançar o objetivo geral, foram elencados os seguintes objetivos específicos:

1. Realizar uma revisão bibliográfica em revistas e periódicos para avaliar o estado da arte nas técnicas de detecção e rastreamento de objetos em imagens 360°.
2. Criar um banco de dados com vídeos de trânsito com a câmera 360° no local escolhido para captura.
3. Desenvolver *pipelines* que realizem a detecção e rastreamento automaticamente.
4. Realizar testes de validação dos experimentos e análise dos resultados.
5. Comparar os resultados obtidos com os diferentes modelos de detecção de objetos.

6. Identificar qual modelo de adequa melhor ao problema juntamente com o rastreador.

1.2 Estrutura da dissertação

Essa dissertação está dividida em cinco capítulos: Introdução; Referencial Teórico e Trabalhos Relacionados; Base de Dados; Estudo Comparativo - Detecção de Objetos e Rastreamento e, por último, Conclusão e Trabalhos Futuros. O Capítulo 1 trata da introdução e motivação desse trabalho, as premissas iniciais e o porquê desse trabalho se mostrar relevante. O Capítulo 2 - Referencial Teórico e Trabalhos Relacionados traz o embasamento teórico necessário para o desenvolvimento desse trabalho e também discute sobre outros trabalhos correlacionados.

O Capítulo 3 - Base de Dados descreve a metodologia utilizada para a elaboração da base de dados. O Estudo Comparativo é apresentado no Capítulo 4, que mostra o objetivo geral desse trabalho, utilizando as redes YOLOv5, YOLOv7 e YOLO-NAS, para a detecção, e a rede DEEPSORT, para o rastreamento dos objetos no trânsito, além da apresentação da comparação de desempenho entre as redes testadas. Por último, no Capítulo 5 - Conclusão e Trabalhos Futuros é realizada uma discussão sobre os resultados alcançados, as contribuições e os próximos passos que ainda devem ser desenvolvidos.

2 Referencial Teórico e Trabalhos Relacionados

Neste capítulo, serão apresentados aspectos teóricos dos temas abordados neste trabalho, bem como trabalhos relacionados. Por opção, esse capítulo foi dividido em três partes. A primeira se refere a obtenção e planificação das imagens em 360°. A segunda se refere ao aprendizado profundo e técnicas de detecção, classificação e rastreamento de objetos e, por último, os trabalhos relacionados que abordam o tema de mobilidade urbana e detecção e rastreamento de objetos em imagens panorâmicas.

2.1 Obtenção de imagens em 360°

A obtenção de imagens em 360°, também conhecida como imagens ominidirecionais ou panorâmicas, é uma técnica fotográfica que tem aplicabilidade em diferentes áreas, como robótica, videomonitoramento, jogos virtuais, identificação de objetos, marketing imobiliário e turismo. Esta técnica permite capturar uma imagem panorâmica completa do ambiente, diminuindo o tamanho dos arquivos para retratar um mesmo ambiente, além de proporcionar uma experiência imersiva para os espectadores, nos casos de jogos e simuladores virtuais.

Existem diversas técnicas para obtenção de imagens em 360°, desde a utilização de câmeras acopladas com lentes olho de peixe, câmeras combinadas com espelhos hiberbólicos, parabólicos ou esféricos (os chamados sistemas catadióptricos), e sistemas com múltiplas câmeras apontadas para diferentes direções, cujas imagens são combinadas por software para criar uma imagem panorâmica final (GäCHTER, 2001), (VIRTUAIS, 2020).

Atualmente, as câmeras com lentes olho de peixe são as mais comuns na obtenção de imagens com 360°. Essas lentes possuem um ângulo de visão de aproximadamente 180°, o que permite capturar toda a área circundante da câmera. Esse tipo de sistema utiliza uma ou duas câmeras perfeitamente alinhadas, cada uma com uma lente olho de peixe. No caso de uma imagem, esta é processada e aberta como uma panorâmica. No caso de duas imagens, as imagens circulares capturadas são tratadas por software, obtendo uma imagem esférica com 360° de campo visual que pode também ser aberta como panorâmica. Alguns exemplos de câmeras com lentes olho de peixe são a Ricoh Theta (©RICOHCOMPANY, 2023) e a GoPro Max (GOPROINC, 2023).

Os chamados sistemas de visão omnidirecional catadióptricos são formados pela combinação de uma câmera alinhada e apontada para um espelho do tipo hiperbólico, parabólico ou esférico. A imagem capturada pela câmera mostra o reflexo do ambiente, na

superfície do espelho, e representa uma imagem em 360° do espaço em torno do sistema. Se o sistema câmera e espelho se encontrar perfeitamente alinhado, com a câmera e espelhos posicionados corretamente, a planificação da imagem ocorre de acordo com a equação matemática da superfície do espelho e a informação perpectiva da cena é recuperada com boa qualidade (GäCHTER, 2001),(MAUGEY, 2023).

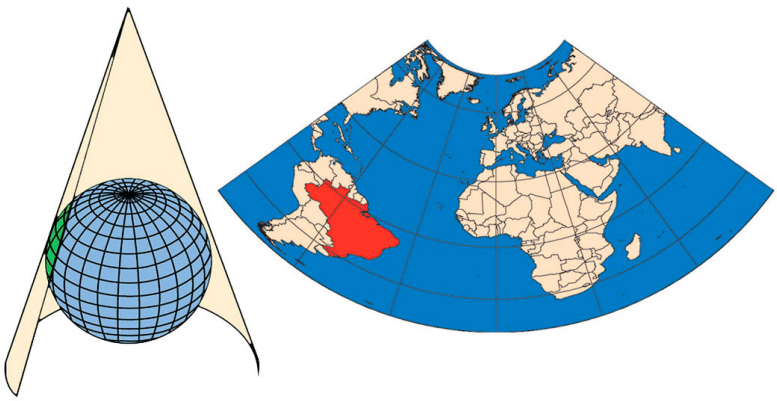
Já a técnica de múltiplas imagens consiste em capturar várias imagens de um determinado local a partir de diferentes ângulos e, em seguida, uni-las para criar uma imagem panorâmica em 360°. O processo de unir as diferentes imagens para compor uma única imagem panorâmica com 360° é comumente conhecido como *stitching* (FURUKAWA; HERNáNDEZ, 2015), (NIR; KARPEL, 2008).

2.1.1 Planificação de imagens

Para a exibição das imagens com 360° em telas bidimensionais como celulares, televisores e computadores, é necessário realizar a planificação, que transforma a imagem esférica em um formato plano. Existem diversas técnicas de planificação como as projeções cônica, cilíndrica, cúbica e equirectangular.

A projeção cônica é usada para criar imagens com 360° de uma área específica, como uma sala ou um jardim. Segundo Silva (1998), essa técnica envolve projetar um cone sobre o ambiente e, em seguida, projetar a imagem resultante no cone em um plano. A projeção cônica é mais adequada para projetos em que o objetivo é criar uma imagem detalhada e precisa de uma área específica. A projeção cônica pode ser dividida em duas categorias: a projeção cônica equidistante e a projeção cônica conforme. Na projeção cônica equidistante, a distância entre os pontos da imagem é proporcional à distância entre os pontos no ambiente, conforme pode ser visto na Figura 1. Já na projeção cônica conforme, há distorção dos pontos, mantendo a forma e a proporção dos objetos (SILVA, 1998).

Figura 1 – Projeção cônica de Albers

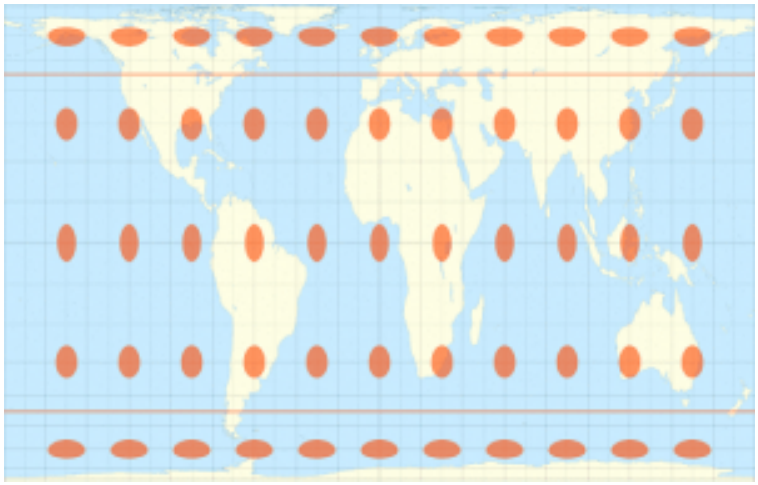


Fonte: IBGE (2024)

A projeção cilíndrica é usada para criar imagens em 360° de áreas mais amplas,

como paisagens ou cidades. De acordo com Silva (1998), nessa técnica, um cilindro é projetado sobre o ambiente e, em seguida, a imagem resultante no cilindro é projetada em um plano. Ainda segundo Silva (1998), assim como a projeção cônica, a projeção cilíndrica também pode ser dividida em duas categorias: a projeção cilíndrica equidistante e a projeção cilíndrica conforme. Na projeção cilíndrica equidistante, a distância entre os pontos da imagem é proporcional à distância entre os pontos no ambiente, como mostrado na Figura 2. Já na projeção cilíndrica conforme, há distorção dos pontos, mantendo a forma e a proporção dos objetos, como pode ser visto na Figura 3.

Figura 2 – Projeção equidistante de Peters



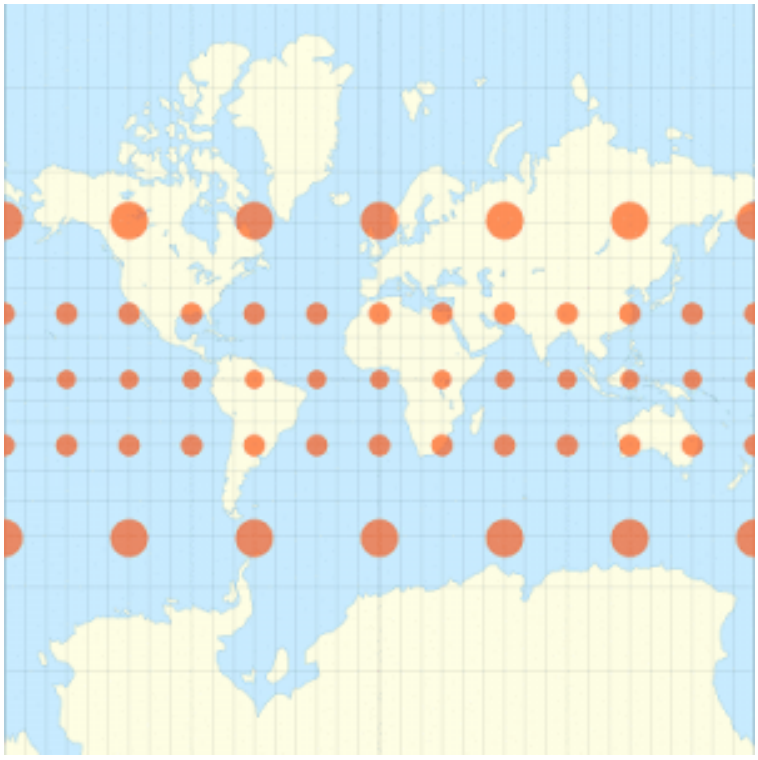
Fonte: Jesus (2018)

Segundo Azevedo et al. (2020), as projeções cúbicas são realizadas projetando seis imagens retangulares em um cubo, cada uma representando uma das seis faces do cubo. Essas imagens são projetadas com distorções mínimas para manter a perspectiva correta. Esta técnica é amplamente utilizada para imagens panorâmicas de alta resolução e para aplicações de realidade virtual. Para criar uma projeção cúbica, as imagens devem ser projetadas usando uma projeção de Snyder (AZEVEDO et al., 2020), que mantém a forma dos continentes e dos oceanos, mas distorce as áreas e as distâncias conforme pode ser visto na Figura 4. É importante garantir que as imagens sejam projetadas com a mesma distorção para que possam ser unidas sem problemas.

Ainda segundo Azevedo et al. (2020) as projeções equirectangulares são realizadas projetando uma imagem plana em uma superfície esférica como pode ser visto na Figura 5, formando a imagem panorâmica. Esta técnica é amplamente utilizada em aplicações de realidade virtual e jogos. É uma das projeções mais utilizadas para imagens panorâmicas, pois é fácil de implementar e é suportada por muitos softwares de edição de imagens. No entanto, essa técnica pode levar a distorções de perspectiva como pode ser visto na Figura 6.

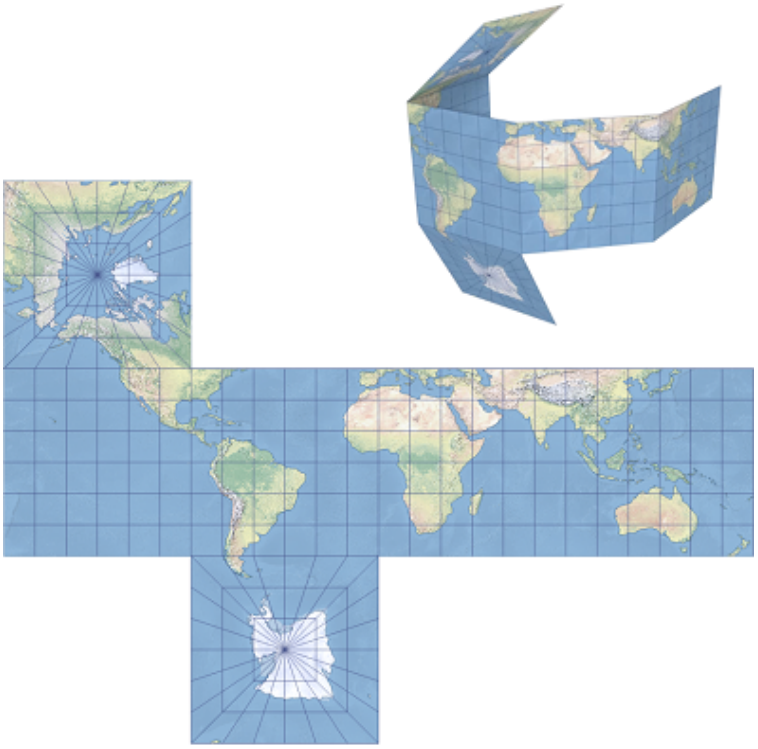
Por fim, todas as técnicas de planificação levam à descontinuidade horizontal onde

Figura 3 – Projeção conforme de Mercator



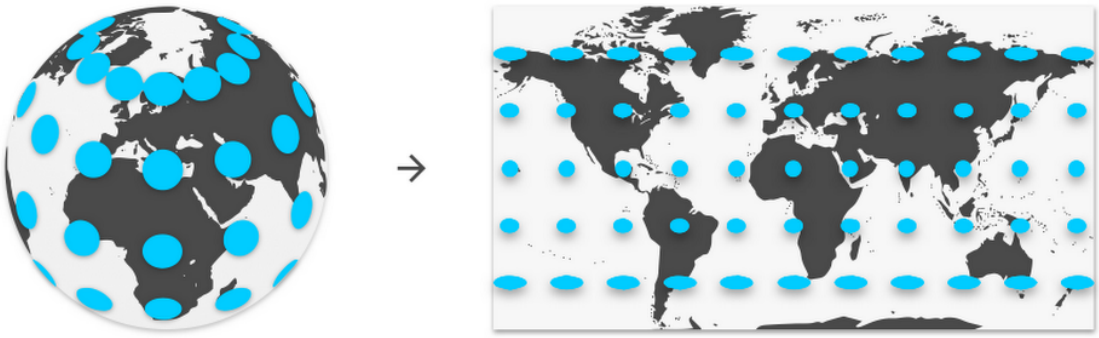
Fonte: Jesus (2018)

Figura 4 – Projeção cúbica



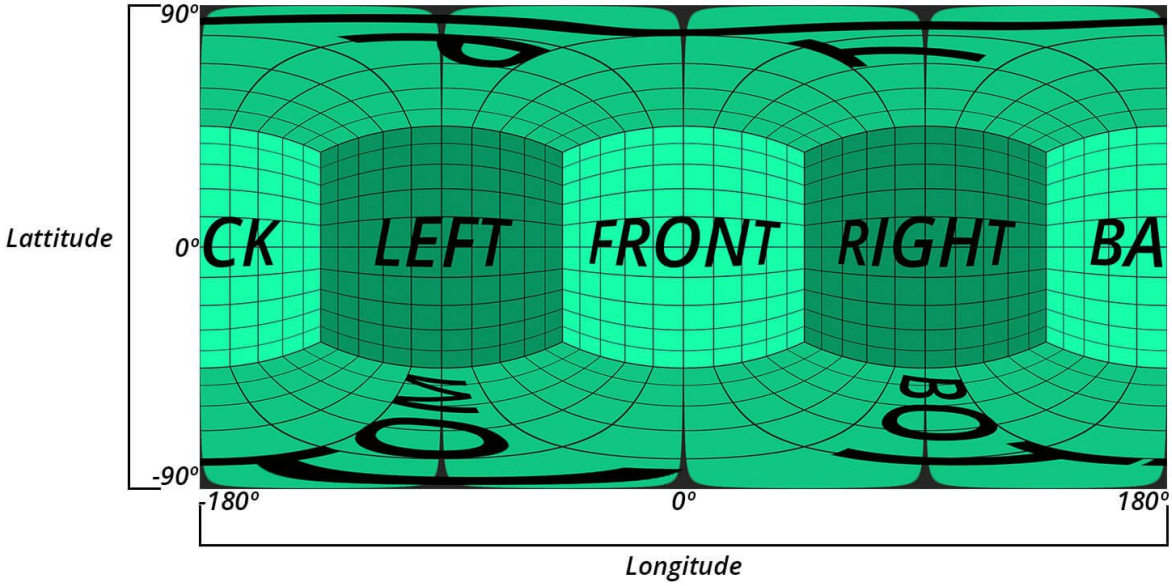
Fonte: ESRI (2024)

Figura 5 – Planificação equirectangular do Globo



Fonte: G. (2021)

Figura 6 – Linhas de distorção da imagem equirectangular



Fonte: G. (2021)

há um "corte" na imagem esférica ou com 360° para a realização da projeção na imagem paronâmica. Nesse caso, objetos próximos podem ficar em lados opostos da imagem como o extremo leste da Rússia e o Alaska no mapa mundi.

2.2 Aprendizado profundo, detecção e rastreamento de objetos

O aprendizado profundo é uma área de estudo da inteligência artificial que tem ganhado grande destaque nos últimos anos (KASHANI; IVRY, 2022). Segundo Goodfellow Yoshua Bengio (2016), o aprendizado profundo é um conjunto de técnicas de *machine*

learning que permitem a criação de modelos que podem aprender a representar e manipular dados complexos, tais como imagens, áudio e texto.

Um dos principais aspectos do aprendizado profundo é a utilização de redes neurais artificiais profundas. Essas redes são compostas por várias camadas interconectadas, o que permite que elas sejam capazes de aprender a representar padrões de alta complexidade (LECUN; BENGIO; HINTON, 2015).

Para treinar uma rede neural profunda, é necessário utilizar um algoritmo de otimização capaz de ajustar os pesos das conexões entre as camadas. Uma das técnicas mais populares é o algoritmo de retropropagação de erro (*backpropagation*), que permite calcular o gradiente da função de custo em relação aos pesos da rede (RUMELHART; HINTON; WILLIAMS, 1986). Atualmente, o aprendizado profundo tem sido aplicado com sucesso em várias áreas, tais como reconhecimento de imagem, processamento de linguagem e reconhecimento de voz (PEREIRA, 2018), (LORENTE; RIERA; RANA, 2021), (MOHAMED; HEMEIDA; HASSAN, 2022).

A detecção e rastreamento de objetos é uma área de grande importância para diversas aplicações, como vigilância e segurança, navegação autônoma de veículos, monitoramento de tráfego, entre outras. Esta técnica consiste em identificar e acompanhar objetos em um ambiente por meio de algoritmos de visão computacional e aprendizado de máquina (PEREIRA, 2018), (LORENTE; RIERA; RANA, 2021), (MOHAMED; HEMEIDA; HASSAN, 2022).

Um dos métodos mais utilizados para a detecção de objetos é a técnica de classificação de imagens por meio de redes neurais convolucionais (CNNs) (GWAK; SAVARESE; BOHG, 2022),(PANDA, 2019) (MANE; MANGALE, 2018). Segundo Krizhevsky, Sutskever e Hinton (2012), as CNNs são capazes de detectar objetos com alta precisão, mesmo em imagens com grande variação de iluminação e ângulos de visão. Para o treinamento das redes, é necessário um grande conjunto de imagens anotadas com as informações dos objetos presentes na cena.

Já para o rastreamento de objetos, uma das técnicas clássicas é a utilização de filtros de Kalman e suas variações. De acordo com Julier, Uhlmann e Durrant-Whyte (1995) e Pei et al. (2019), o filtro de Kalman é um método recursivo para estimar o estado de um sistema dinâmico a partir de medições ruidosas. Com o uso deste filtro, é possível prever a posição futura do objeto a partir da sua posição atual e da velocidade estimada, assim como corrigir a sua posição estimada com base nas novas medições.

Outra abordagem mais antiga e tradicional para o rastreamento de objetos é a técnica de detecção de características (*feature detection*), como descrito por Shi e Tomasi (1994). Esta técnica consiste em encontrar pontos de interesse nas imagens, como cantos e bordas, e associá-los ao objeto a ser rastreado. Com o uso de algoritmos de correspondência

de características, é possível identificar os pontos correspondentes nas imagens subsequentes, permitindo o rastreamento do objeto.

Além das técnicas mencionadas, existem outras abordagens mais recentes que têm sido exploradas na detecção e rastreamento de objetos com bons resultados. Uma delas é o uso de métodos baseados em redes neurais recorrentes (RNNs), que são capazes de lidar com sequências de dados e informações temporais (BEWLEY et al., 2016), (WOJKE; BEWLEY; PAULUS, 2017).

Outra técnica é a detecção e rastreamento de objetos baseada em técnicas de segmentação de imagens. De acordo com Ling e Fidler (2017), a segmentação de imagens é capaz de fornecer informações mais precisas sobre a posição e forma dos objetos, o que pode ser utilizado para melhorar a detecção e rastreamento. Esta abordagem utiliza técnicas como o algoritmo de Felzenszwalb e Huttenlocher (2004), que segmenta a imagem em regiões com similaridades de cor e textura.

É importante ressaltar que a detecção e rastreamento de objetos ainda apresentam desafios, como a detecção de objetos em imagens de baixa qualidade e a identificação de objetos em situações de oclusão ou mudanças bruscas de iluminação. Por isso, é importante continuar explorando novas abordagens e algoritmos que possam melhorar a precisão e eficiência dessas técnicas.

2.2.1 YOLO

A rede YOLO (*You Only Look Once*), criada pela equipe liderada por Joseph Redmon em 2016, é uma arquitetura baseada em uma única etapa que usa uma única rede neural convolucional para prever caixas delimitadoras e classes de objetos em uma imagem. A rede YOLO divide a imagem em uma grade de células e, em cada célula, prevê várias caixas delimitadoras, cada caixa tem uma confiança associada, que indica a probabilidade de haver um objeto dentro da caixa delimitadora. Essa rede também prevê uma classe para cada caixa delimitadora, indicando que tipo de objeto está contido nela (REDMON et al., 2015).

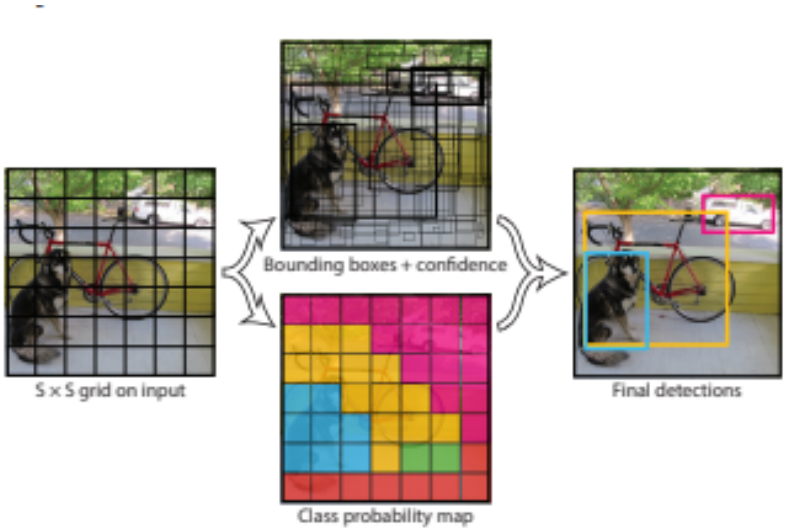
A YOLO é composta por três componentes principais: *backbone*, *neck* e *head*. O *backbone* é a primeira parte da arquitetura da rede YOLO e é responsável por extrair características da imagem de entrada. Geralmente, o *backbone* é uma rede neural convolucional pré-treinada em grandes conjuntos de dados de imagens, como a Darknet (TERVEN; CóRDOVA-ESPARZA; ROMERO-GONZáLEZ, 2023). O *backbone* consiste em várias camadas convolucionais seguidas por camadas de *pooling* para reduzir a resolução espacial da imagem de entrada. A saída do *backbone* é uma representação de características da imagem de entrada que é então passada para o próximo componente da rede, o *neck* (REDMON et al., 2015), (TERVEN; CóRDOVA-ESPARZA; ROMERO-GONZáLEZ, 2023).

O *neck* é responsável por integrar as características extraídas pelo *backbone* para detectar objetos na imagem. Ele é composto por camadas convolucionais adicionais, que permitem que a rede combine as características de baixo nível e de alto nível extraídas pelo *backbone*. Uma das técnicas comuns usadas no *neck* é a FPN (*Feature Pyramid Network*), que é usada para construir uma pirâmide de características que combina informações de diferentes escalas espaciais (REDMON et al., 2015). Isso permite que a rede detecte objetos em diferentes tamanhos e resoluções na imagem de entrada (REDMON et al., 2015).

O *head* é a terceira e última parte da arquitetura da rede YOLO e é responsável por detectar objetos na imagem de entrada, é composto por várias camadas convolucionais que são responsáveis por gerar as saídas da rede, que incluem as coordenadas dos objetos detectados e suas respectivas classes. Também inclui camadas de detecção baseadas em caixas delimitadoras, que são responsáveis por detectar os objetos e atribuí-los a diferentes classes. Uma técnica comum usada no *head* é a NMS (*Non-Maximum Suppression*), que é usada para remover detecções redundantes e selecionar apenas as detecções mais confiáveis (REDMON et al., 2015), (HOSANG; BENENSON; SCHIELE, 2017).

Há várias vantagens da YOLO em relação a outras técnicas de detecção de objetos como a velocidade, pois usa uma única rede neural convolucional para prever todas as caixas delimitadoras e classes de objetos; a detectção de objetos pequenos, porque usa uma grade de células para prever as caixas delimitadoras; e também a consistência de desempenho em várias tarefas de detecção de objetos (SHAFIEE M. J.; CHYWL, 2017), conforme pode ser visto na Figura 7.

Figura 7 – Construção das caixas delimitadoras



Fonte: Redmon et al. (2015)

No entanto, a rede YOLO também tem algumas limitações como a dificuldade

em detectar objetos muito próximos uns dos outros. Também apresenta dificuldades em detectar objetos com tamanhos muito diferentes e pode ter dificuldades em detectar objetos com formas muito irregulares (SHAFIEE M. J.; CHYWL, 2017).

Desde o lançamento da rede YOLO original, várias variantes foram propostas para melhorar a precisão e a velocidade da detecção de objetos, sendo possível descrever a seguinte linha cronológica:

- Lançado no final de 2016, pelo criador da YOLO, a YOLOv2 ou YOLO9000 é capaz de detectar mais de 9000 categorias de objetos. Apresenta melhorias significativas frente à YOLO (REDMON; FARHADI, 2016);
- Lançado em 2018, ainda por Redmon, a YOLOv3 possui um dos artigos mais legíveis no campo de visão computacional pela sua linguagem coloquial. Apresenta uma nova abordagem na detecção de objetos menores, realizando-a em três níveis separados de granularidade, melhorando o desempenho (REDMON; FARHADI, 2018);
- Lançado em abril de 2020, por Alexey Bochkovskiy, a YOLOv4 apresenta velocidade e acurácia de detecção superior a modelos anteriores. Foram implementados novos recursos como WRC (*Weighted-Residual-Connections*), CSP (*Cross-Stage-Partial-connections*), CmBN (*Cross mini-Batch Normalization*), SAT (*Self-adversarial-training*) e ativação Mish (BOCHKOVSKIY; WANG; LIAO, 2020);
- Em trabalhos mais recentes foram desenvolvidas as versões YOLOv5 (JOCHER et al., 2022), YOLOv6 (LI et al., 2022) e YOLOv7 (WANG; BOCHKOVSKIY; LIAO, 2022b), que buscam uma maior precisão e menor custo computacional.
- Em 2024 foram desenvolvidas as versões YOLOv8 (SOHAN; RAM; CH, 2024), YOLOv9 (WANG; YEH; LIAO, 2024) e YOLOv10 (WANG et al., 2024). Essas versões não foram consideradas para testes neste trabalho, pois surgiram depois da realização dessa pesquisa.

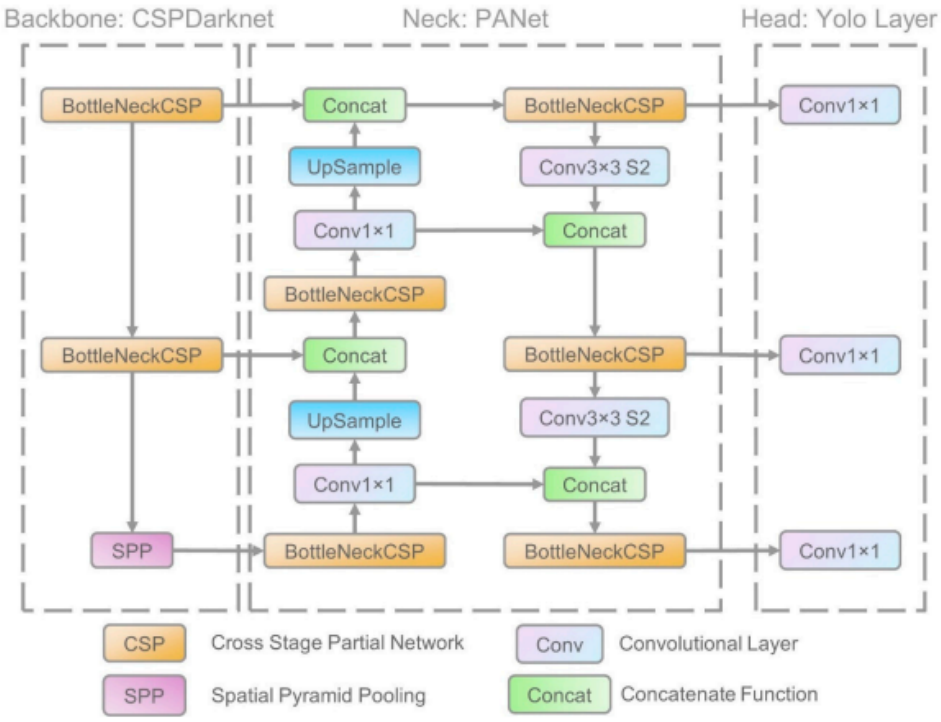
Tudo isso faz da família YOLO uma abordagem interessante para ser avaliada e possivelmente aplicada nesse projeto, pois possui melhor desempenho frente a outros métodos como *EfficientDet*, *CenterMask* ou *ASFF* (HUANG et al., 2021),(BOCHKOVSKIY; WANG; LIAO, 2020). Além do melhor desempenho, a família YOLO apresenta uma vasta documentação, além de uma comunidade atuante.

2.2.2 YOLOv5

A rede YOLOv5 foi desenvolvida pela equipe da Ultralytics, em 2020, sendo sucessora da YOLOv4 (JOCHER et al., 2022). A YOLOv5 introduziu várias melhorias em relação às versões anteriores, como o uso do método de ativação *swish* em vez do ReLU, o que proporciona resultados mais precisos. Além disso, a YOLOv5 emprega a técnica de *focal loss*, que atribui maior importância a objetos difíceis de detectar, melhorando a

detecção em geral. A arquitetura da YOLOv5 pode ser observada na Figura 8.

Figura 8 – Arquitetura da YOLOv5



Fonte: Xu et al. (2021)

A YOLOv5 utiliza o CSPDarknet como *backbone*, uma estrutura leve e eficiente que reduz a complexidade computacional da rede. O CSPDarknet é composto por várias camadas convolucionais que capturam informações contextuais e formam a base para a detecção de objetos. O *neck* utiliza uma arquitetura conhecida como PAN (*Path Aggregation Network*) que integra informações de várias resoluções espaciais para melhorar a precisão da detecção de objetos. E o *head* utiliza uma estrutura de cabeça complexa, composta por várias camadas convolucionais, onde são aplicadas técnicas como detecção de múltiplas escalas e regressão das coordenadas dos objetos (XU et al., 2021).

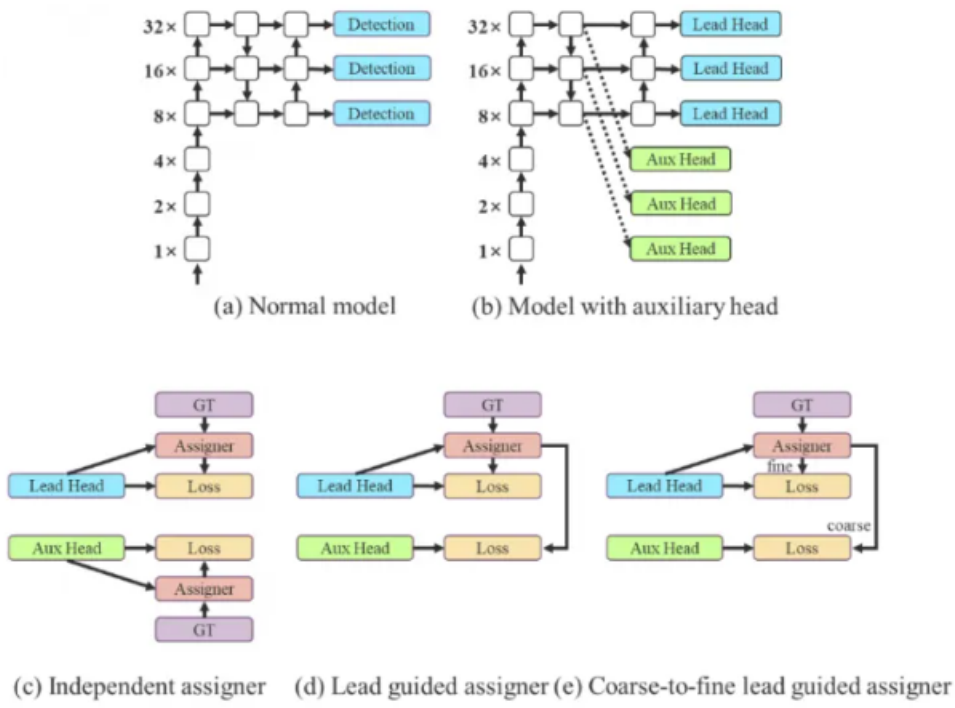
2.2.3 YOLOv7

No estado da arte das redes YOLO foi lançada, no final de 2022, a YOLOv7, tendo uma evolução frente a YOLOv5 e a YOLOv6. A principal diferença entre a YOLOv7 e as versões anteriores é a adição de novos módulos de detecção de objetos, como o módulo SAM (*Spatial Attention Module*), permitindo que o modelo concentre sua atenção em áreas específicas da imagem (WANG; BOCHKOVSKIY; LIAO, 2022b).

A YOLOv7 utiliza como o *backbone* a E-LAN (*Extended Efficient Layer Aggregation Network*), um método de escalonamento de pilha no *neck* e uma abordagem de múltiplas

heads, uma *head* na saída da rede chamada de *lead head* e várias *auxiliary heads* nas camadas intermediárias (XU et al., 2021), conforme pode ser visto na Figura 9.

Figura 9 – Estrutura da *Head* da YOLOv7



Fonte: Boesch (2023)

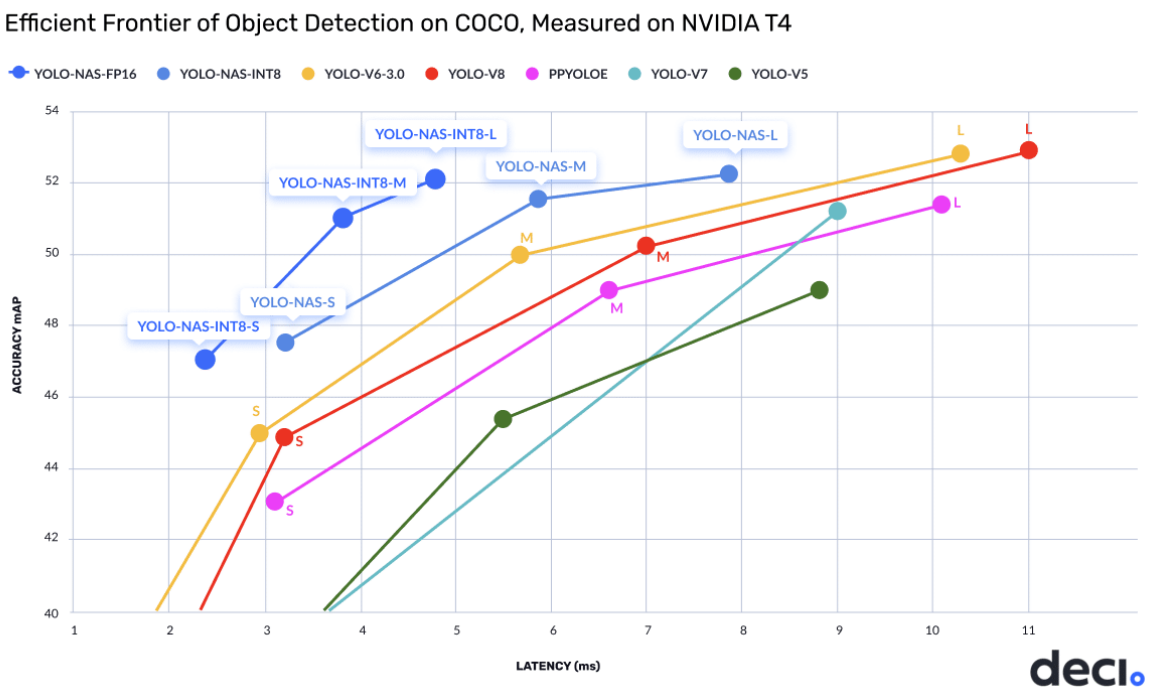
Uma das principais características da YOLOv7 é sua velocidade. A arquitetura do modelo é projetada para executar a detecção de objetos em tempo real, com taxas de quadros de até 90 FPS (*frames per second* ou quadros por segundo) em GPUs de última geração. Além disso, a YOLOv7 tem um baixo tempo de treinamento, o que a torna uma boa opção para treinamento em grandes conjuntos de dados (WANG; BOCHKOVSKIY; LIAO, 2022b).

Outra característica importante da YOLOv7 é sua precisão. O modelo foi treinado em grandes conjuntos de dados de detecção de objetos, como o MS COCO (*Common Objects in Context*), e alcançou uma precisão média de 45,6 mAP (*Mean Average Precision*), que é uma métrica padrão para avaliar o desempenho de modelos de detecção de objetos (WANG; BOCHKOVSKIY; LIAO, 2022b).

2.2.4 YOLO-NAS

Disponibilizada em maio de 2023 pela equipe de pesquisa da DECI, a YOLO-NAS surge na vanguarda dos detectores YOLO focando em melhor desempenho, velocidade e detecção de objetos pequenos (KHVEDCHENYA EUGENE E SAHOTA, 2023). Conforme mostrado na Figura 10, a YOLO-NAS apresenta um desempenho melhor e um custo computacional menor de detecção na base de dados COCO.

Figura 10 – Comparação entre a YOLO-NAS e outras redes detectando objetos na base COCO



Fonte: Khvedchenya Eugene e Sahota (2023)

A principal característica da YOLO-NAS é utilização de algoritmo de procura de arquitetura neural (NAS - *Neural Architecture Search*), proprietário da DECI chamado AutoNAC. Esse algoritmo realiza uma varredura automática buscando a melhor arquitetura entre 10^{14} arquiteturas possíveis.

2.2.5 DEEPSORT

Na literatura, o método clássico para a realização do rastreamento em visão computacional é o filtro de Kalman. Apesar de antigo, esse método vem sendo utilizado também em trabalhos recentes como Taylor, Mirdanies e Saputra (2016) e Gunjal et al. (2018). O filtro de Kalman utiliza informações sobre as posições anteriores do objeto e suas velocidades para prever a posição futura do objeto.

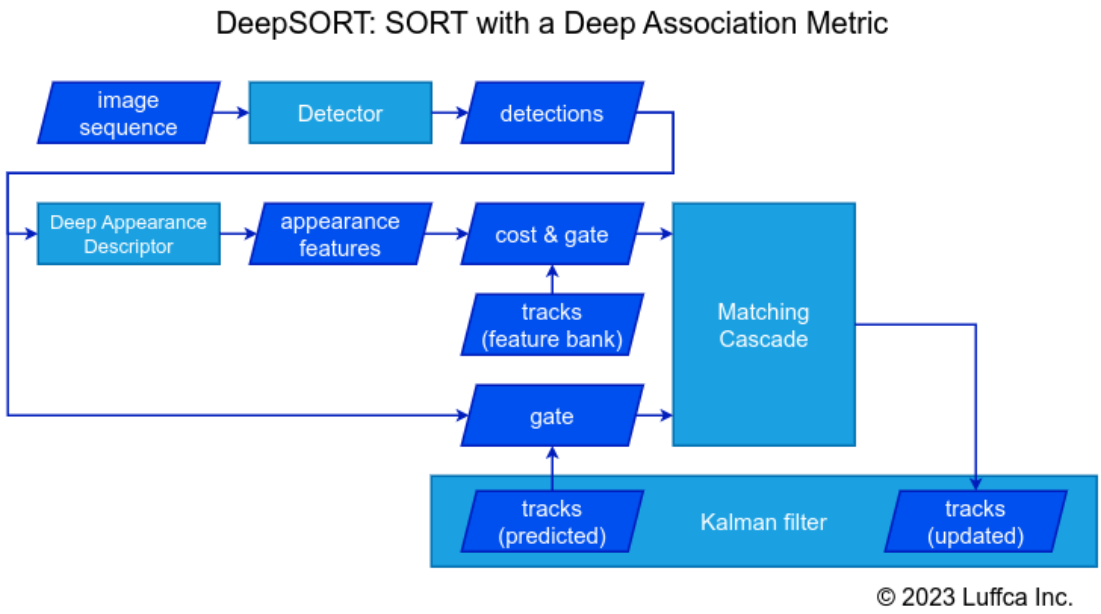
Entretanto, atualmente, tem-se utilizado cada vez mais os modelos SORT (BEWLEY et al., 2016) e DEEP-SORT (WOJKE; BEWLEY; PAULUS, 2017) para rastreamento de objetos. Esses modelos são baseados no filtro de Kalman em conjunto com o algoritmo Húngaro, apresentando melhoras significativas em comparação à utilização dos modelos clássicos e outros modelos para rastreamento de objetos.

Segundo a proposta de Wojke, Bewley e Paulus (2017), o algoritmo DEEPSORT realiza a detecção de objetos em cada quadro do vídeo utilizando uma rede neural

convolucional, que produz uma lista de detecções candidatas. Essas detecções candidatas são então associadas a objetos existentes. Além disso, o DEEPSORT utiliza uma técnica de re-identificação para lidar com o problema de oclusão parcial do objeto, que ocorre quando um objeto é temporariamente coberto por outro objeto na imagem. Essa técnica utiliza um modelo de rede neural para extrair características discriminativas do objeto que podem ser usadas para identificar o objeto, mesmo quando ele está parcialmente oculto.

De acordo com os resultados experimentais apresentados por Wojke, Bewley e Paulus (2017), o DEEPSORT (*Simple Online and Realtime Tracking with a Deep Association Metric*) é capaz de rastrear objetos em vídeos com maior precisão do que o SORT pois utiliza uma métrica de associação profunda, permitindo que ele lide com diferentes perspectivas de objetos, o que o torna ideal para aplicações em cenários dinâmicos. Um fluxograma do DEEP-SORT pode ser visto na Figura 11. Além disso, é capaz de lidar com objetos em movimento rápido e oclusão parcial, que são desafios comuns em sistemas de rastreamento de objetos em vídeo conforme pode ser visto na Figura 12.

Figura 11 – Fluxograma do DEEP-SORT

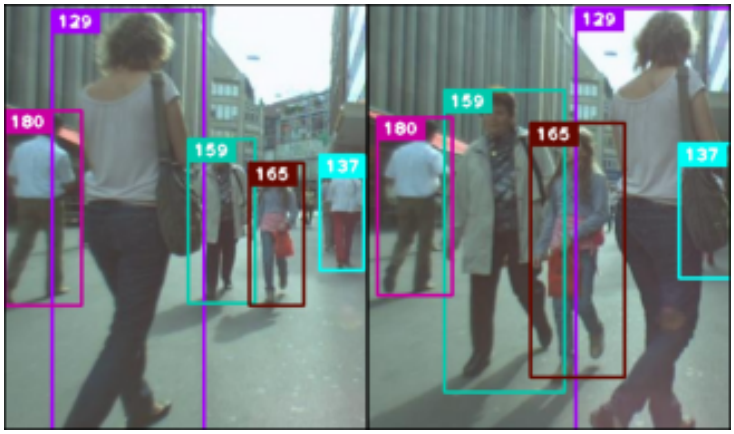


Fonte: LuffcaInc (2024)

2.3 Métricas

Segundo Mariano (2021), um modelo de classificação de dados tem como objetivo fazer previsões com base em ocorrências passadas. Para isso, o modelo utiliza um conjunto de dados contendo indivíduos e suas propriedades. Além disso, é necessário conhecer

Figura 12 – Utilização do DEEPSORT com oclusão



Fonte: (WOJKE; BEWLEY; PAULUS, 2017)

os resultados esperados para esse conjunto de dados, que são os rótulos. Todas essas informações são usadas para treinar um modelo que será capaz de prever os resultados esperados para novos dados que surgirem no futuro.

Durante o treinamento, é importante utilizar um conjunto de dados separado, que não tenha sido usado anteriormente, para avaliar o desempenho do modelo. No entanto, simplesmente contar a quantidade de acertos do modelo não é suficiente para determinar se ele é bom ou não. Dependendo do problema em estudo, diferentes métricas devem ser utilizadas para essa avaliação. Antes de apresentar essas métricas, é necessário compreender alguns conceitos relacionados à classificação binária: as classes que os dados preditos podem receber.

Uma forma simples de apresentar os dados do modelo de classificação é através de uma matriz de confusão, a qual indica os acertos e erros do sistema de detecção, conforme o Quadro 1.

Quadro 1 – Modelo da Matriz de Confusão

		REAL	
		Sim	Não
DETECTADA	Sim	VP	FP
	Não	FN	VN

onde:

- VP (Verdadeiro Positivo) é a classificação correta do positivo para a classe;
- VN (Verdadeiro Negativo) é a classificação correta do negativo para a classe;
- FP (Falso Positivo) é a classificação positiva (errônea) prevista pelo detector quando deveria ser negativa;

- FN (Falso Negativo) é a classificação negativa (errônea), ou seja, o detector não realizou a detecção da classe, quando deveria ser positiva.

A partir da matriz de confusão são calculadas as métricas de *Precision* (Eq. 2.1) ou Precisão; *Recall* (Eq. 2.2) ou Revocação; e *F1-Score* (Eq. 2.3).

A Precisão mede a quantidade de vezes que o modelo acertou a classificação em relação ao total de vezes que ele fez alguma classificação. Já o *Recall* mede a quantidade de vezes que o modelo acertou a classificação em relação ao total de vezes que ele deveria ter acertado. O *F1-Score* é a média harmônica entre Precisão e *Recall*, (MARIANO, 2021).

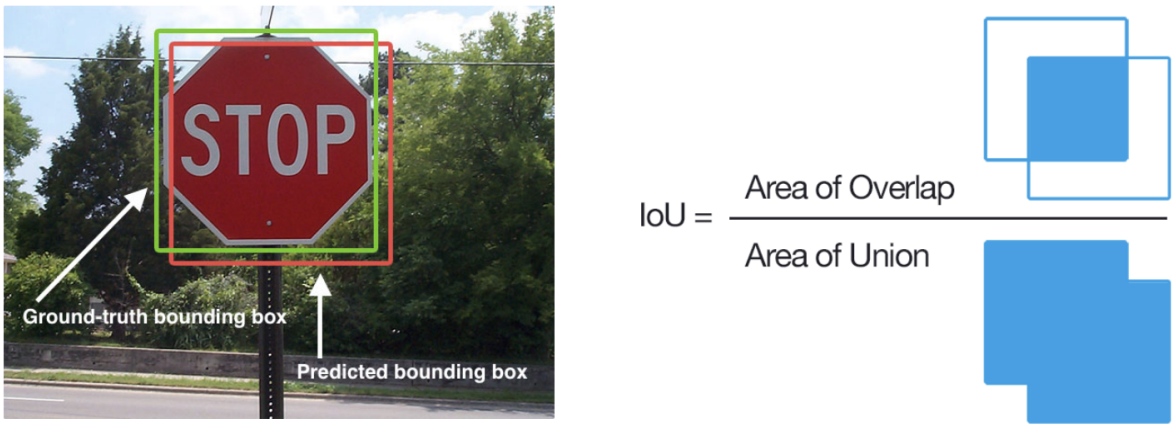
$$Precision = \frac{VP}{(VP + FP)} \tag{2.1}$$

$$Recall = \frac{VP}{(VP + FN)} \tag{2.2}$$

$$F1\text{-}Score = \frac{2 * (Precision * Recall)}{(Precision + Recall)} \tag{2.3}$$

A métrica mAP (*Mean Average Precision*), em especial na forma de mAP@.5 e mAP@.5:.95 são duas das métricas muito utilizadas nos sistemas de detecção em imagens. Essas métricas possuem como base de cálculo a área sob a curva de Precisão-Recall (AUC). Para definir se um objeto foi corretamente detectado, utiliza-se a *Intersection over Union* (IoU) que é a razão entre a interseção da caixa delimitadora (*bounding box*) anotada e a predita, pela união das mesmas conforme exemplificada pela Figura 13.

Figura 13 – *Intersection over Union (IoU)*



Fonte: Shah (2022)

A partir do valor de IoU, é definida o tipo da classificação conforme a matriz de confusão do Quadro 1, realizando-se uma comparação com o *threshold IoU*. Para valores de IoU maiores ou iguais ao *threshold IoU* a detecção é considerada como verdadeiro

positivo, o contrário é considerado como falso positivo (SHAH, 2022). A partir dessa definição é calculada a precisão. A mAP (*Mean Average Precision*) representa a média da AP (*Average Precision*) para cada classe e é calculada através da Eq. 2.4:

$$mAP = \frac{1}{N} \sum_{i=1}^{N} Precision_i \tag{2.4}$$

O sufixo @.5 ou @.5:95 se refere ao valor do *threshold IoU*. A mAP@.5 possuí um *threshold IoU* de 0,5, enquanto a mAP@.5:95 ou também conhecida como mAP[0.5:0.05:0.95] é a média das mAP com o *threshold IoU* partindo de 0,5 até 0,95 com aumento de 0,05 a cada acréscimo (REN et al., 2016).

Por fim, as métricas MOTA (*Multiple Object Tracking Accuracy*) e MOTP (*Multiple Object Tracking Precision*) proposta por (BERNARDIN; STIEFELHAGEN, 2008), são duas das principais métricas utilizadas em sistemas de rastreamento.

A MOTA contabiliza a acurácia do objeto rastreado, *frame* a *frame*, contabilizando falsos positivos, erros na posição da caixa de detecção e a troca de categoria do objeto através da Eq. 2.5:

$$MOTA = 1 - \frac{\sum_t (m_t + fp_t + mme_t)}{\sum_t g_t} \tag{2.5}$$

onde:

- m_t são os erros na posição da caixa de detecção no *frame*;
- fp_t são os falsos positivos no *frame*;
- mme_t são as trocas de categoria do objeto;
- g_t é a quantidade de objetos no *frame*;
- t é o *frame*.

A MOTP calcula o erro total da posição estimada da caixa delimitadora, pelo total de correspondências realizadas, *frame* à *frame*, através da Eq. 2.6:

$$MOTP = \frac{\sum_{i,t} d_t^i}{\sum_t c_t} \tag{2.6}$$

onde:

- d_t^i diferença entre a posição anotada de cada caixa delimitadora e a posição rastreada da caixa delimitadora no *frame*;
- c_t total de correspondências no *frame*;
- t é o *frame*.

2.4 Trabalhos relacionados

Foi realizada uma pesquisa bibliográfica sobre bases de dados semelhantes a proposta por esse trabalho e também trabalhos de detecção e rastreamento de objetos em imagens. As bases disponíveis serão divididas usando-se duas características presentes nesse trabalho: formato da imagem (perspectiva normal ou em 360°) e o tipo de local onde as imagens foram capturadas.

Das bases de dados contendo imagens em 360°, pode-se citar Su, Jayaraman e Grauman (2016) e Hu et al. (2017) que possuem como foco os esportes, Morgado Nuno Vasconcelos e Wang (2018) e Morgado, Li e Vasconcelos (2020) que são baseadas em vídeos do *Youtube*, principalmente apresentções musicais.

Entre as bases de dados que possuem como cenário o trânsito, baseadas em imagens perspectivas, pode-se citar Xu, Huang e Liu (2021), Zhang et al. (2021), Kataoka et al. (2018).

Em relação aos trabalhos de detecção e rastreamento de elementos do trânsito, pode-se citar o trabalho de Brasil Rafael H. e Machado (2014) que realizou a detecção de semáforos em imagens convencionais utilizando OpenCV para detecção e Camshift para rastreamento. Por sua vez, Júnior (2018) utilizou imagens planas obtidas a partir de um drone para detectar veículos (aplicando a YOLOv2) em um semáforo, afim de criar um sistema de controle semafórico em tempo real. Já Cortez (2022) traz uma excelente revisão bibliográfica acerca de trabalhos relacionados, também mostra um estudo comparativo entre as redes YOLOv3 e YOLOv4 no desenvolvimento de controle semafórico. Olivatto (2021) trabalha com imagens 360° para identificação em tempo real de rampas de acessibilidade. Sugimoto, Ikehata e Aizawa (2022) utilizam uma única câmera 360° para detectar a direção que carros trafegam. No trabalho de Liao, Xie e Geiger (2021) utiliza-se imagens panorâmicas para a detecção de elementos de trânsito, afim de se navegar veículos de forma autônoma. Lavrenko et al. (2024) utiliza imagens 360° e a rede YOLOv3 para detectar pessoas, bicicletas e animais com o objetivo de evitar colisões com veículos.

Entre os trabalhos encontrados, nenhum dos trabalhos possui como proposta trabalhar com objetos de trânsito em imagens 360° graus, em um cenário próximo a sinais de trânsito e ciclovias, com a presença de muitos pedestres e ciclistas, além dos veículos. Assim, espera-se que o trabalho desenvolvido nesta dissertação possa trazer uma nova contribuição para a comunidade científica no tema de mobilidade urbana e cidades inteligentes.

3 Base de Dados

Este capítulo apresenta a metodologia para a elaboração da base de dados denominada VV360. O desenvolvimento de uma base de dados sólida é essencial para o estudo comparativo proposto nesse trabalho, pois garante a isonomia dentre as detecções realizadas com as variadas redes YOLO descritas no Capítulo 2.

Nessa seção são descritos inicialmente os materiais utilizados para a elaboração da base de dados e também como os vídeos foram gravados. Também é descrita a metodoligia de anotação das imagens e a montagem da base de dados.

3.1 Materiais e método de gravação de vídeo

Nessa seção são apresentados os materiais e o método utilizados para a gravação dos vídeos a fim de compor a base de dados VV360.

3.1.1 Materiais

Os materiais utilizados nesse trabalho podem ser divididos em dois grupos. O primeiro para a gravação e planificação da base de dados e o segundo para a anotação, detecção e rasteamento dos vídeos através do *benchmark* proposto.

Para a criação da base de dados VV360, foram gravados 25 vídeos, uma câmera que gera imagens com 360º graus, do tipo e modelo Ricoh Theta V (RICOH...,), com as seguintes configurações:

1. Resolução 4K, H264: 3840×1920;
2. Frame Rate de 29,97fps;
3. Taxa de transmissão de 56Mbps;
4. Altura de 1,2m do tripé, nivelado por nível de bolha.

Os vídeos foram gravados através do modo de *streaming* disponibilizado pelo aplicativo do fabricante da câmera para Android. O *streaming* foi realizado para o celular Poco M3 Pro, sendo as imagens planificadas pelo próprio aplicativo da câmera.

Para a anotação foram utilizados vários computadores, pois foi empregado o serviço *web* CVAT (*Computer Vision Annotation Tool*) (CVAT...,) para tal tarefa. As imagens com rótulos foram salvas em nuvem e o processo será descrito em detalhes na Seção 3.2.

Já para a detecção e rastreamento foi empregada uma máquina específica com as seguintes configurações:

1. Processador Intel Core i7 11700k;
2. Memória RAM DDR4 64GB 3200MHz;
3. GPU RTX 3080;
4. Armazenamento 8TB-HD/1TB SSD.

3.1.2 Método de gravação de vídeo

Para a elaboração da base de dados VV360 foram gravados 25 vídeos com média de um minuto cada, na cidade de Vila Velha, Estado do Espírito Santo, Brasil. A gravação ocorreu na esquina da Avenida Champagnat com a Avenida Antônio Gil Veloso, no dia 07 de julho de 2022, no período entre as 15:53 e 16:20h. Um exemplo do tipo de imagem panorâmica contida na base de dados está mostrado na Figura 14.

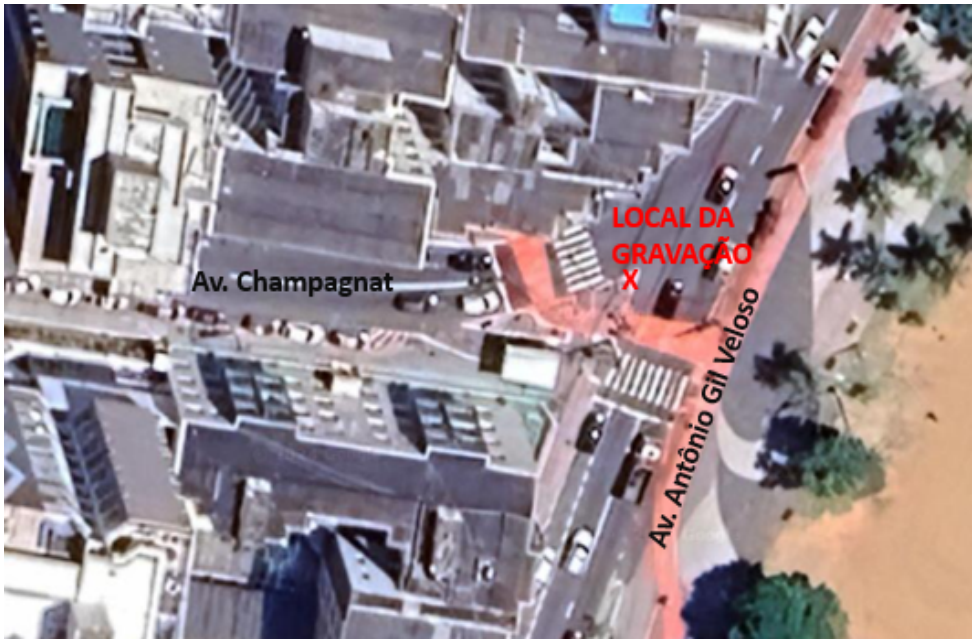
Figura 14 – Imagem panorâmica do local da gravação

Fonte: do Autor

Essa esquina está localizada entre dois semáforos para veículos, os quais também possuem os respectivos semáforos para pedestres. Há ainda duas faixas para pedestres, uma em cada semáforo, como pode ser visto na Figura 15. Outro ponto importante, é que esse local fica em frente a um calçadão à beira da praia, que possui uma ciclovia, havendo, portanto, uma grande circulação de pedestres e bicicletas.

Foi utilizado um tripé com 1,2 metro de altura para fixação da câmera e captura das imagens. Para garantir que a câmera não sofresse nenhuma inclinação vertical ou horizontal, o tripé escolhido possuia um indicador de nivelamento por bolha. Também

Figura 15 – Mapa do local da gravação



Fonte: do Autor

tomou-se um cuidado especial com a descontinuidade horizontal, que foi discutida na Subseção 2.1.1, evitando "cortar" a rua e alinhando a descontinuidade junto ao meio fio. Isso permitiu manter a continuidade no movimento dos veículos, porém sacrificando a continuidade de pedestres e ciclistas ao atravessarem a faixa à esquerda da imagem.

O alinhamento foi realizado através do *streaming* da câmera no celular pelo aplicativo THETHA disponível na *Play Store* nos celulares Android. Primeiramente, é necessário realizar a conexão via *Bluethoot* entre o celular e a câmera e acessar o modo *streaming*. Por fim, depois de gravados os vídeos, os arquivos foram salvos em computador.

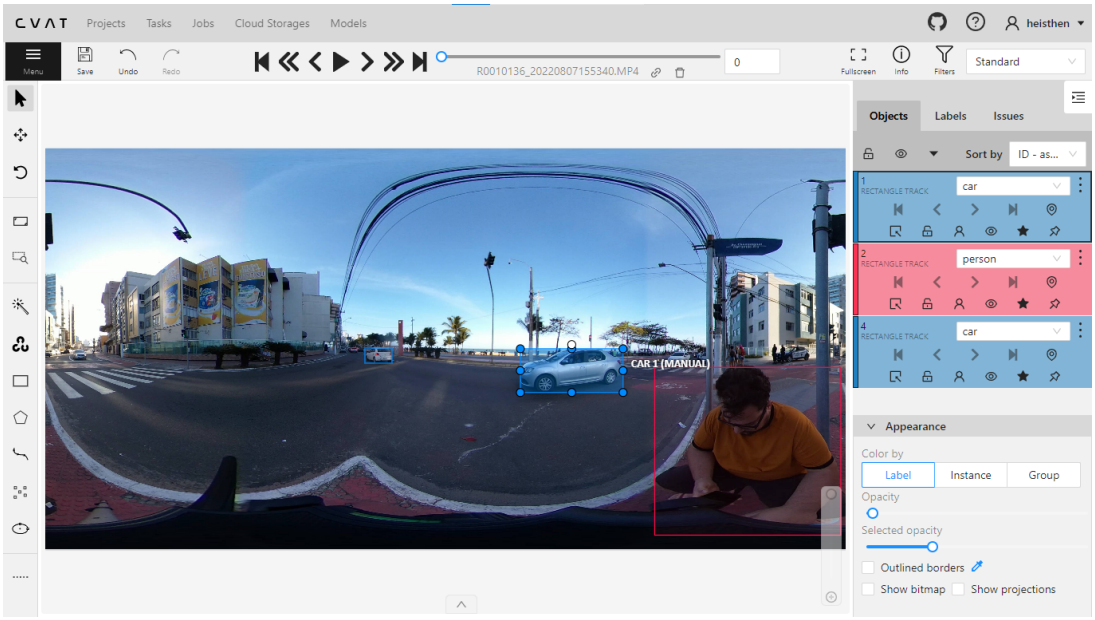
3.2 Anotação

Segundo Rebinth e S (2019), a anotação manual é uma etapa fundamental para a detecção automática de objetos em imagens, pois é necessário treinar os algoritmos com rótulos bem definidos e que se aproximem à percepção humana. Além disso, as imagens rotuladas são utilizadas para comparar a detecção realizada com a anotação, extraindo-se as métricas de detecção para a avaliação de desempenho do sistema. A anotação manual é um método lento e trabalhoso, porém garante os melhores resultados, devido à grande capacidade humana de detecção de padrões, frente aos algoritmos existentes (PAGARE; SHINDE, 2012).

Para realizar o processo de anotação manual existe uma série de *softwares* e aplicações *web* como (SUPERANNOTATE,), (V7LABS,), (LABELBOX,), entre outros.

A plataforma utilizada neste trabalho foi o CVAT (*Computer Vision Annotation Tool*) (CVAT...,), o qual era gratuito na época da aquisição das imagens para a base de dados, além de ter uma interface agradável e intuitiva como pode ser visto na Figura 16. Isso facilitou e acelerou bastante o trabalho de anotação e rotulação dos veículos, pedestres e ciclistas nas imagens.

Figura 16 – Interface de anotação do CVAT



Fonte: do Autor

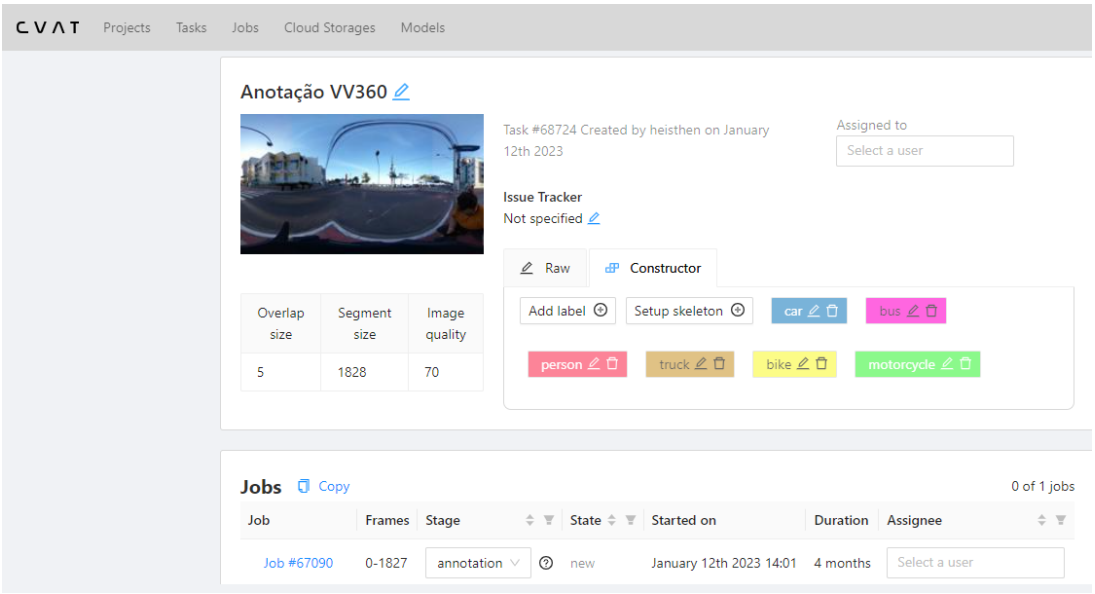
Para a anotação da base de dados, primeiramente foi realizado o *upload* dos vídeos na plataforma CVAT e definidos os *labels* dos objetos das classes: carro (*car*); bicicleta (*bike*); motocicleta (*motorcycle*); pedestre (*person*); ônibus (*bus*); caminhão (*truck*), como mostrado na Figura 17.

A anotação foi realizada em modo *tracking*, o que facilita o rastreamento dos objetos. A base de dados possui um total de 12.813 *frames* com a observação de todas as classes, mas com a predominância de pedestres, bicicletas e carros.

3.2.1 Metodologia de Anotação

Um sistema inteligente, utilizando reconhecimento de objetos através de visão computacional, tem que ter a capacidade de discernir os objetos que realmente são importantes para o sistema, daqueles que não o são. No contexto de calçadões, muito comuns em áreas litorâneas, essa premissa é fundamental, pois muitos dos pedestres e ciclistas, que estão presentes nas imagens, não constituem elementos de interesse para o sistema porque não necessariamente possuem a intenção de atravessar nas faixas de pedestre. Se a detecção destes elementos fosse usada para controlar um semáforo, por

Figura 17 – Labels de anotação



Fonte: do Autor

exemplo, tais detecções modificariam a frequência de abertura do semáforo para pedestres, sem a necessidade efetiva de que isso fosse feito.

Devido a essas considerações, nessa base de dados, foi utilizada uma abordagem para a detecção, tentando identificar a real importância dos objetos para o sistema. Assim, os objetos que transitavam pelo calçadão, sem parada nos semáforos, foram considerados como *outliers*, sendo priorizada a detecção dos elementos que se aproximavam do semáforo. Ao entrar nas àreas de detecção, que serão explicadas mais a frente neste texto, esses objetos que estavam no calçadão deixam de ser *outliers* e passam a ser considerados pelo sistema.

As regras de detecção foram separadas em dois grupos diferentes. O primeiro grupo são dos pedestres e ciclistas, enquanto o segundo grupo inclui carros, motociclistas, caminhões e ônibus. Essa separação foi adotada devido aos caminhos pelos quais os objetos se movem. Enquanto o primeiro grupo transita pelas calçadas, ciclovias e faixas de pedestres, o segundo grupo transita pelas ruas. As áreas de detecção dos pedestres e ciclistas estão exemplificadas na Figura 18.

A área 1 é a transição entre o calçadão e a calçada que dá acesso à faixa de pedestres. Essa área é formada pelos pontos A, B, C e a faixa de corte da imagem à direita da imagem. Nessa área, os objetos são detectados ao encostarem na ciclovia, quando transitam do sentido calçadão para a faixa de pedestres; e deixam de ser detectados quando encostam no calçadão, no sentido da faixa de pedestres para o calçadão.

A área 2 corresponde somente à ciclovia, onde os ciclistas passam a ser dectados ao passarem pelas plantas (ponto A) no sentido da esquerda para a direita, e deixam de

Figura 18 – Limites de detecção

Fonte: do Autor

ser detectados ao se aproximarem do recuo para ônibus (ponto B). Quando transitam da direita para a esquerda, a detecção passa a ser no ponto B e termina no ponto A. Essa detecção é necessária, pois geralmente os ciclistas param na faixa e atravessam para a outra calçada, não sendo possível prever antecipadamente quais ciclistas farão o trajeto (pontos A-B), e quais atravessarão na faixa.

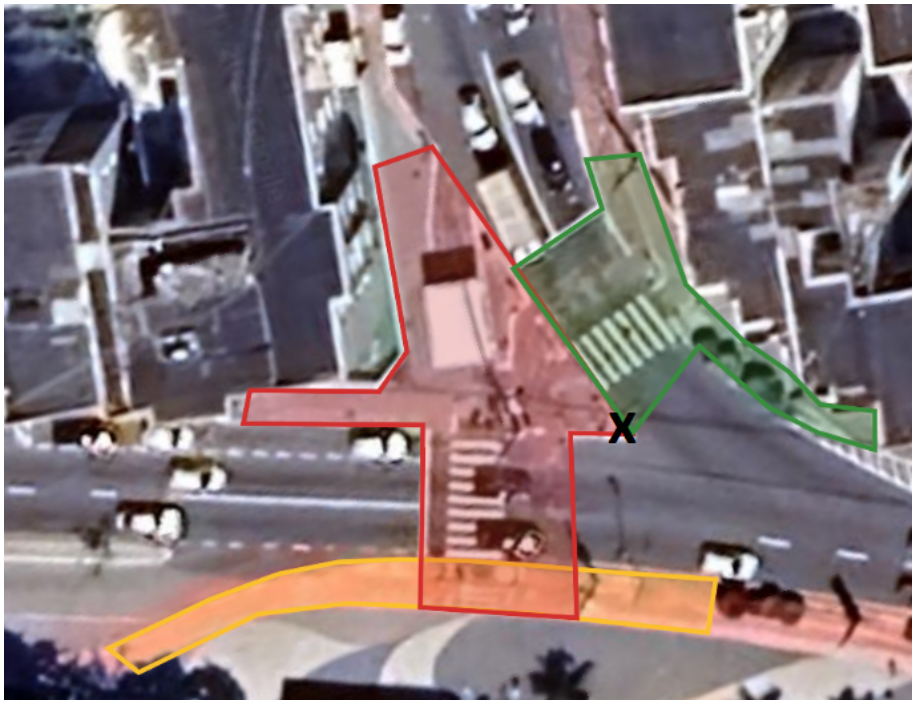
A área 3 consiste na calçada onde foram gravados os vídeos. Nesse caso, a detecção ocorre a partir do ponto C. As áreas podem ser melhor identificadas através da Figura 19, onde a área 1 está em vermelho, a área 2 em amarelo, a área 3 em verde e o **X** marca o local de gravação.

Na área 3 ocorre a descontinuidade da imagem característica da planificação equirectangular da imagem omnidirecional, conforme é descrito por G. (2021) e discutido anteriormente. Como pode ser observado nas Figuras 20 e 21 e indicado pelas setas, ao se atravessar a borda da imagem planificada, a detecção atual é terminada sendo, a seguir, gerada um nova detecção do outro lado da imagem. Essa transição ocorre quando a maior parte do objeto atravessa de uma borda para outra.

A área 4 consiste na outra calçada, onde a detecção ocorre entre os pontos D e E. As últimas áreas são as faixas de pedestres, onde a detecção ocorre sem problemas.

Os limites de detecção do segundo grupo (carros, motocicletas, ônibus e caminhões) são definidos a partir da sua observação inicial na imagem e da redução de sua área para um valor menor que 900 pixels quadrados, valor arbitrado de forma empírica, ao passarem pelas vias dos semáforos.

Figura 19 – Vista aérea das áreas de detecção



Fonte: do Autor

Figura 20 – Detecção anterior à descontinuidade

Fonte: do Autor

3.3 Disponibilização da base de dados

A base da dados será disponibilizada para a comunidade científica no link [1], sendo necessária a leitura e aceite do termo de responsabilidade em cumprimento da Lei Geral de Proteção de Dados Pessoais. Através desse link é possível baixar um arquivo compactado que contém os vídeos originais e suas respectivas anotações no formato YOLO 1.1.

[1] https://forms.gle/CUwddRdMP4oGwX32A

Figura 21 – Detecção posterior à descontinuidade

Fonte: do Autor

Os arquivos são nomeados usando-se uma taxonomia exemplificada através do Quadro 2, através da qual é possível rastrear a data e horário exato do início da gravação do vídeo conforme o Quadro 2.

Quadro 2 – Taxonomia da VV360° database

TAXONOMIA VV360° DATABASE				
MODELO	R + ID DO VÍDEO	_	ANO/MÊS/DIA	HORA/MINUTO/SEGUNDO
EXEMPLO	R0010136	_	20220807	155340

Fonte: Produção do próprio autor.

A Tabela 1 traz o total de objetos anotados por classe.

Tabela 1 – Objetos anotados na VV360º

CLASSE					
car	bus	person	truck	bike	moto
136282	1291	69880	551	66186	6956

4 Estudo comparativo da detecção e rastreamento de objetos do trânsito em imagens panorâmicas

Este capítulo apresenta, em um primeiro momento, um estudo comparativo entre algumas das redes YOLO, enumeradas no Capítulo 2, na tarefa de detecção de objetos de trânsito, utilizando a base de dados VV360, apresentada no Capítulo 3. Depois disso, também foi realizado o rastreamento utilizando o algoritmo DEEP-SORT juntamente com as redes de detecção. A partir dos resultados, mais uma vez foi feita a comparação do desempenho das redes na tarefa conjunta de detecção e rastreamento.

De acordo com Fachin (2001), o método comparativo consiste em examinar objetos ou eventos e explicá-los com base em suas semelhanças e diferenças. Esse método possibilita a análise de dados concretos e a dedução de similaridades e discrepâncias entre elementos específicos, abstratos e gerais, possibilitando investigações indiretas.

Dito isso, foram escolhidas as seguintes redes para o estudo comparativo:

1. Rede YOLOv7 com Treinamento;
2. Rede YOLOv7 sem Treinamento na base de VV360°;
3. Rede YOLOv5 com Treinamento;
4. Rede YOLO-NAS com Treinamento.

Vale ressaltar que esse presente trabalho é uma continuação do trabalho Scarparo, Santos e Vassallo (2023), onde foram obtidos bons resultados utilizando a rede YOLOv7 com treinamento. Nesse sentido, foi utilizada a rede YOLOv7 sem treinamento na base VV360° (utilizando o treinamento padrão na base de dados COCO) para demonstrar a importância de se treinar a rede previamente. A rede YOLOv5 foi escolhida por ser uma rede consolidada na comunidade científica e a YOLO-NAS por ser a rede YOLO mais recente, na época do desenvolvimento desse trabalho. Para o trabalho dessa disseetação, não foi possível realizar o estudo com mais redes, devido a limitações de tempo e equipamento. Foi considerada a utilização de outras redes além das baseadas na YOLO, mas devido às características suprecitadas e a familiaridade de utilização, foram selecionadas somente as redes baseadas em YOLO.

A metodologia empregada no estudo comparativo é descrita a seguir. Primeiramente, foram sorteados 18 vídeos de forma aleatória, do total de 25 vídeos da base de dados.

Esses vídeos foram utilizados no treinamento das redes, correspondendo a 72% da base de dados. Os outros 7 vídeos, que constituem 28% da base de dados, foram apresentados de forma inédita aos algoritmos para a detecção e classificação dos objetos.

Neste estudo serão apresentados os resultados de detecção e rastreamento utilizando as métricas descritas na Seção 2.3. A partir da utilização de cada rede selecionada, será realizada uma comparação e discussão dos resultados obtidos.

4.1 Detecção utilizando YOLOv7

A YOLOv7 foi a primeira rede a ser utilizada neste trabalho. Nesta seção serão apresentados os resultados utilizando a YOLOv7 com e sem treinamento. Essa abordagem visa demonstrar a importância do treinamento para a obtenção de melhores resultados.

Para o treinamento foi utilizada a rede YOLOv7, proposta em Wang, Bochkovskiy e Liao (2022b). Foram sorteados 18 vídeos com suas anotações (tendo como classes de interesse as classes citadas na Subseção 3.2.1), os quais foram utilizados para a construção da base de dados de treinamento utilizando a metodologia proposta em Wang, Bochkovskiy e Liao (2022a). Para o modelo de treinamento foram utilizados os hiperparâmetros da Tabela 2, sendo o modelo treinado até sua estabilização como pode ser observado nas Figuras 22 e 23.

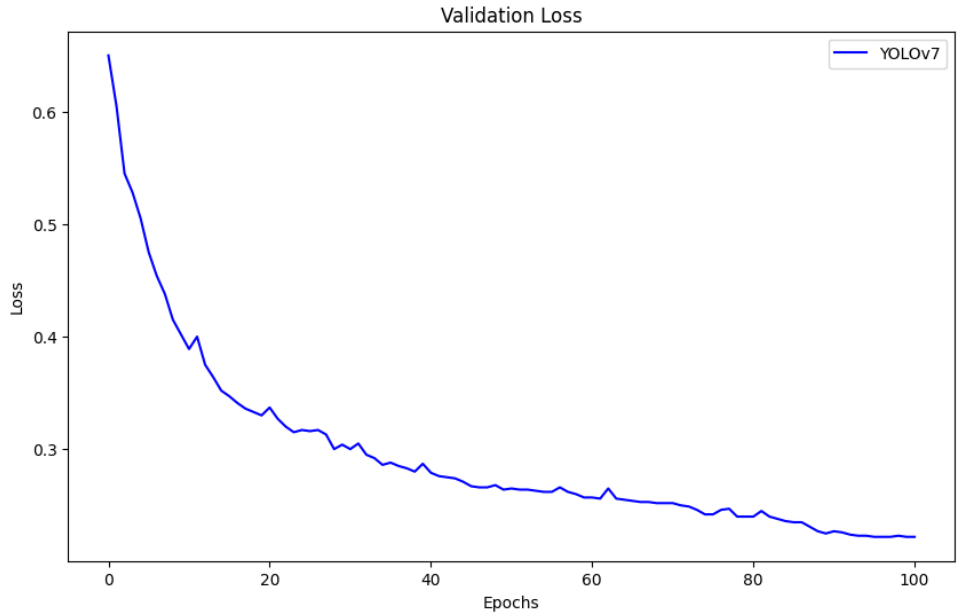
Tabela 2 – Hiperparâmetros para o treinamento da YOLOv7

HIPERPARÂMETRO	VALOR UTILIZADO
Image Size	3840x1920
Batch Size	8
Epochs	100
Loss	0,02
lr	0,01
Solver	SGD (0.937 momentum)
hsv	h: 0,015 s: 0,7 v: 0,4
Translate	0,2
Scale	0,5
Flip left-right	0,5
Mosaic	1,0

Feito o treinamento, foram realizados os testes obtendo a detecção e classificação dos objetos como exemplificado na Figura 24.

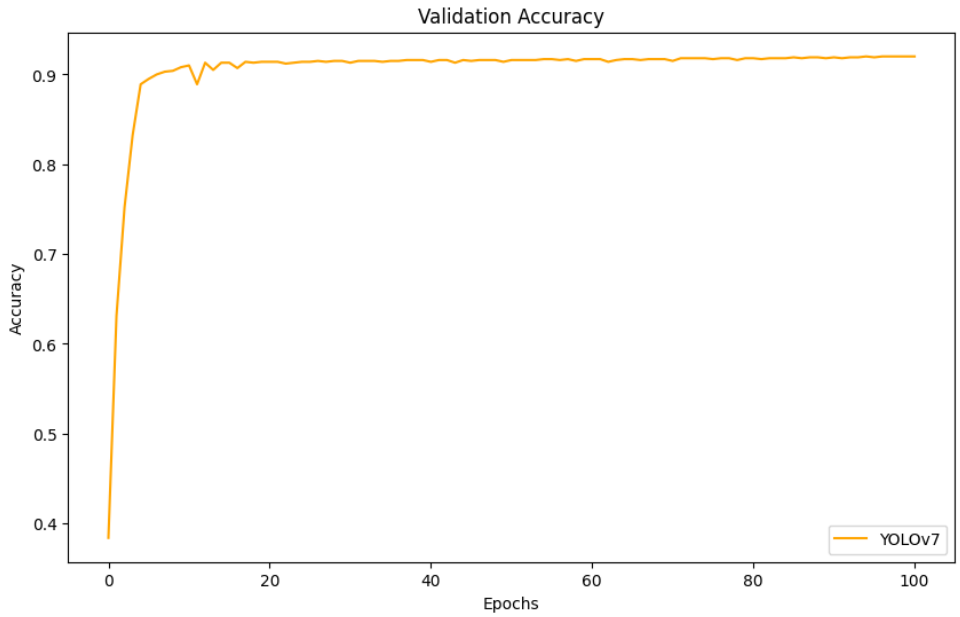
A base de teste possui sete vídeos escolhidos de forma aleatória. Essa base possui 9.225 imagens, com 184.673 objetos anotados no total. Destes, 89.203 são carros, 978

Figura 22 – Gráfico de Perda de Validação (Validation Loss)



Fonte: do Autor

Figura 23 – Gráfico de Ganho de Acurácia (Validation Accuracy)



Fonte: do Autor

são ônibus, 43.770 são pessoas, 548 são caminhões, 45.395 são bicicletas e 4.779 são motocicletas. A partir dessa base de teste, foram realizadas as detecções e classificações utilizando a rede YOLOv7 com treinamento, gerando a matriz de confusão conforme pode ser visto em Tabela 3.

Como observado na matriz de confusão, os *labels* de caminhões foram detectados em outras classes, principalmente como ônibus. Por essa questão não foram calculadas

Figura 24 – Detecção e classificação com a YOLOv7 treinada

Fonte: do Autor.

Tabela 3 – Tabela de Confusão - YOLOv7 com treinamento

DETECTADO		**ROTULADO**						
		car	bus	person	truck	bike	moto	BG FP
	car	82171	1	0	3	10	8	8472
	bus	8	814	0	548	0	0	561
	person	1	0	35125	0	254	0	11115
	truck	0	0	0	0	0	0	258
	bike	15	0	291	0	31863	114	5218
	moto	25	0	19	0	136	4274	432
	BG FN	6983	163	8335	0	13132	373	0

métricas dos caminhões. Em relação as demais classes, a maioria dos *labels* foram detectados corretamente, sendo a segunda classe mais recorrente a não detecção como fundo (BG FN), em consequência das áreas de detecção descritas na Subseção 3.2.1.

A partir da matriz de confusão, foram calculados a Precisão, o *Recall* e o *F1-Score*, para todas as detecções e também individualizados por classe conforme a Tabela 4.

Na sequência, foram obtidas as métricas mAP@.5, mAP@.5:.95 (Tabela 5).

Com base nas Tabelas 4 e 5, as classes de carros e motos apresentaram as melhores precisões, sendo que a classe carro também apresentou a melhor revocação, F1-Score, mAP@.5 e mAP@.5:.95. A classe de pessoas e bicicletas apresentaram um resultado inferior, devido à metodologia de áreas de detecção utilizadas nesse trabalho e também pelo maior número de oclusões que essas classes sofrem.

Após ser realizada a detecção e classificação na base de dados utilizando a rede

Tabela 4 – Precisão, *Recall* e *F1-Score* - YOLOv7 com treinamento

	Precisão	**Recall**	**F1-Score**
Total	0,889	0,808	0,847
car	0,919	0,903	0,911
bus	0,951	0,787	0,861
person	0,778	0,777	0,777
bike	0,879	0,688	0,772
motorcycle	0,919	0,886	0,902

Tabela 5 – mAP@.5 e mAP@.5:.95 - YOLOv7 com treinamento

	mAP@.5	**mAP@.5:.95**
Total	0,863	0,436
car	0,925	0,610
bus	0,907	0,244
person	0,789	0,344
bike	0,769	0,466
motorcycle	0,926	0,515

YOLOv7 com treinamento, foi realizado o mesmo experimento com a rede YOLOv7 sem treinamento na base de dados proposta.

Ao contrário das outras análises feitas nesse trabalho, foram extraídas somente as métricas totais para a detecção e identificação utilizando a rede YOLOv7 sem treinamento, conforme a Tabela 6.

Tabela 6 – Métricas Totais - YOLOv7 sem treinamento

	Total
Precisão	0,436
Recall	0,232
F1-Score	0,303
mAP@.5	0,247
mAP@.5:.95	0,1

Essa abordagem foi utilizada para a comparação direta com a YOLOv7 com treinamento, estudando a viabilidade da utilização da YOLOv7 sem a necessidade de treinamento. Na Tabela 7 é mostrada a comparação entre a rede YOLOv7 com e sem treinamento e o percentual de queda de desempenho.

Após a análise e comparação entre as redes YOLOv7, fica clara a razão para descartar a utilização da rede YOLOv7 sem treinamento como uma opção viável, devido ao seu desempenho muito inferior se comparada à rede YOLOv7 treinada.

Tabela 7 – Queda de desempenho entre a YOLOv7 treinada e a YOLOv7 sem treinamento

	YOLOv7		Queda de
Métrica	Com treinamento	Sem treinamento	desempenho (%)
Precisão	0,889	0,436	50,9%
Recall	0,808	0,232	71,3%
F1-Score	0,847	0,303	64,2%
mAP@.5	0,863	0,247	71,3%
mAP@.5:.95	0,436	0,1	77,1%

4.2 Detecção utilizando YOLOv5

Após o estudo realizado com a YOLOv7, passou-se a trabalhar com a YOLOv5, sendo realizados o treinamento, detecção e classificação de forma análoga ao que foi descrito na seção anterior. A YOLOv5 foi selecionada pois é uma rede consolidada na comunidade científica, sendo amplamente utilizada.

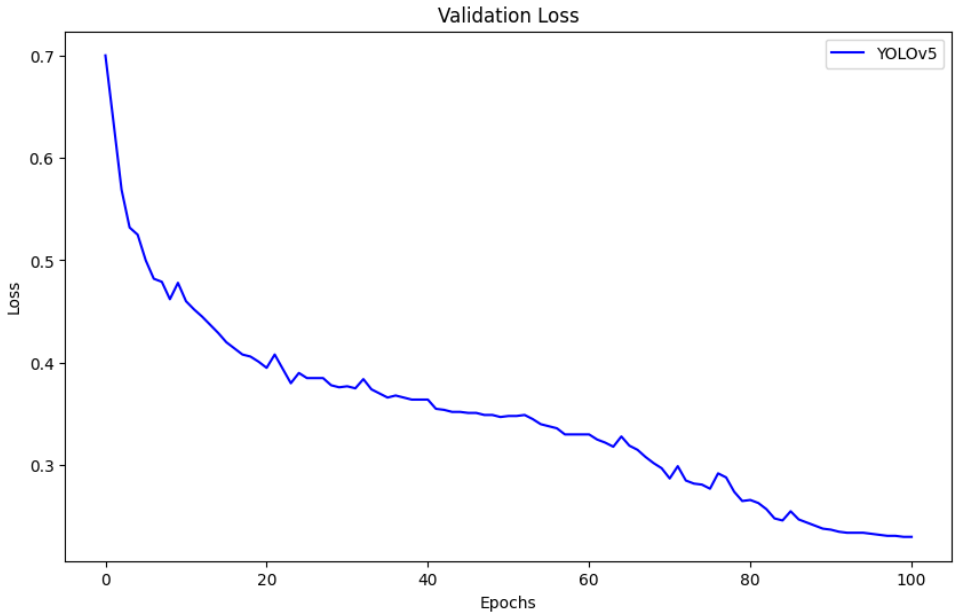
Nessa seção serão apresentados os resultados utilizando a YOLOv5 original proposta em Jocher et al. (2022). A partir das bases de dados de treinamento e teste já definidas anteriormente, os vídeos foram utilizados junto com suas anotações para os dois procedimentos. Com os hiperparâmetros mostrados na Tabela 8, o modelo foi treinado até sua estabilização, como pode ser observado nas Figuras 25 e 26.

Tabela 8 – Hiperparâmetros para o treinamento da YOLOv5

HIPERPARÂMETRO	VALOR UTILIZADO
Image Size	3840x1920
Batch Size	8
Epochs	100
Loss	0,02
lr	0,01
Solver	SGD (0.937 momentum)
hsv	h: 0,015 s: 0,7 v: 0,4
Translate	0,2
Scale	0,5
Flip left-right	0,5
Mosaic	1,0

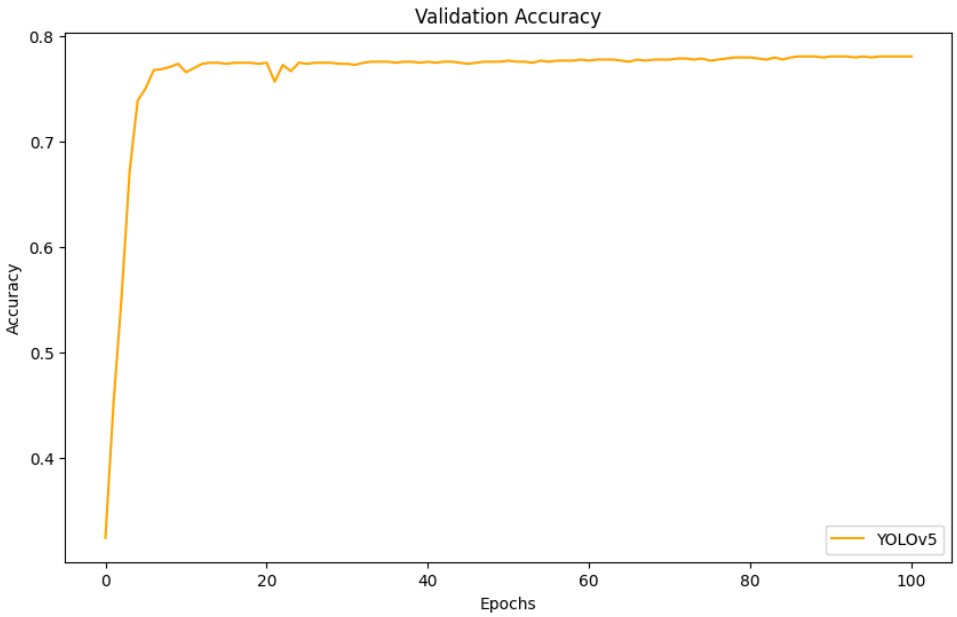
Após o treinamento da YOLOv5 foram realizados os testes, obtendo a detecção e classificação dos objetos. Na sequência, foram extraídos os resultados dos sete vídeos já sorteados anteriormente e mencionados na Seção 4.1. Nesse processo foram utilizadas as

Figura 25 – Gráfico de Perda de Validação (Validation Loss)



Fonte: do Autor

Figura 26 – Gráfico de Ganho de Acurácia (Validation Accuracy)



Fonte: do Autor

mesmas 9.225 imagens, com 184.673 objetos anotados no total. Destes, 89.203 são carros, 978 são ônibus, 43.770 são pessoas, 548 são caminhões, 45.395 são bicicletas e 4.779 são motocicletas. Foram realizadas as detecções e classificações utilizando a rede YOLOv5 com treinamento, gerando a matriz de confusão que pode ser vista na Tabela 9.

Como observado na matriz de confusão, os *labels* de caminhões não foram detectados devido aos problemas já listados na Seção 4.1. Por essa questão, aqui também não foram

Tabela 9 – Matriz de Confusão - YOLOv5 com treinamento

DETECTADO		**ROTULADO**						
		car	bus	person	truck	bike	moto	BG FP
	car	82067	0	0	5	0	0	11204
	bus	0	812	0	543	0	0	0
	person	0	0	35016	0	454	0	9901
	truck	0	0	0	0	0	0	0
	bike	0	0	438	0	31776	96	4690
	moto	0	0	0	0	0	4253	782
	BG FN	7136	166	8316	0	13165	430	0

calculadas as métricas dos caminhões. Em relação as demais classes, semelhante com o que ocorreu com a detecção utilizando a rede YOLOv7 com treinamento, a maioria dos *labels* foram detectados corretamente, sendo a segunda classe mais recorrente a não detecção como fundo (BG FN), em consequência das áreas de detecção descritas na Subseção 3.2.1.

A partir da matriz de confusão, foram calculados a Precisão, o *Recall* e o *F1-Score*, para todas as detecções e também individualizados por classe conforme a Tabela 10.

Tabela 10 – Precisão, *Recall* e *F1-Score* - YOLOv5 com treinamento

	Precisão	**Recall**	**F1-Score**
Total	0,698	0,604	0,648
car	0,755	0,847	0,798
bus	0,638	0,171	0,270
person	0,556	0,643	0,596
bike	0,729	0,591	0,653
motorcycle	0,812	0,768	0,789

Por fim, foram obtidas as métricas mAP@.5, mAP@.5:.95 (Tabela 11).

Com base nas Tabelas 10 e 11, assim como ocorreu na detecção utilizando a rede YOLOv7, as classes de carros e motos apresentaram as melhores precisões, sendo que a classe carro também apresentou a melhor revocação, F1-Score, mAP@.5 e mAP@.5:.95. A classe de pessoas e bicicletas apresentaram um resultado inferior, devido à metodologia de áreas de detecção utilizadas nesse trabalho e também pelo maior número de oclusões que essas classes sofrem.

Após a análise entre a rede YOLOv7 com treinamento e a rede YOLOv5 com treinamento, a rede YOLOv7 apresenta um desempenho superior em Precisão geral 88,9% frente a 69,8% da YOLOv5, essa superioriade prossegue em todas as classes. Em relação à Revocação a YOLOv7 apresenta 80,8% contra 60,4% da sua versão mais antiga, com esse resultado superior também se confirmando nas classes individuais. A YOLOv7 também apresenta melhores resultados de mAP@.5 e mAP@.5:.95 com 86,3% e 43,6%

Tabela 11 – mAP@.5 e mAP@.5:.95 - YOLOv5 com treinamento

	mAP@.5	**mAP@.5:.95**
Total	0,638	0,315
car	0,859	0,544
bus	0,275	0,057
person	0,621	0,254
bike	0,591	0,661
motorcycle	0,768	0,361

respectivamente, já a YOLOv5 apresentou 63,8% para mAP@.5 e 31,5% para mAP@.5:.95. Além desses fatores a YOLOv7 apresentou também um desempenho computacional melhor se comparado à YOLOv5.

4.3 Detecção utilizando YOLO-NAS

Após o estudo realizado com a YOLOv5, passou-se a trabalhar com a YOLO-NAS, sendo realizados o treinamento, detecção e classificação de forma análoga às seções anteriores. A YOLO-NAS foi selecionada pois era a rede YOLO mais recente quando esse trabalho foi elaborado, apresentando como principal característica um melhor custo computacional se comparado à YOLOv7 e YOLOv5 por exemplo.

Nessa seção serão apresentados os resultados utilizando-se a YOLO-NAS original proposta em Khvedchenya Eugene e Sahota (2023). Mais uma vez, as bases de dados de treinamento e teste são as mesmas definidas anteriormente, com os vídeos e suas anotações. Para o treinamento, foram utilizados os hiperparâmetros da Tabela 12, sendo o modelo treinado até sua estabilização como pode ser observado nas Figuras 27 e 28.

Após o treinamento da YOLO-NAS, foram realizados os testes, obtendo a detecção e classificação dos objetos.

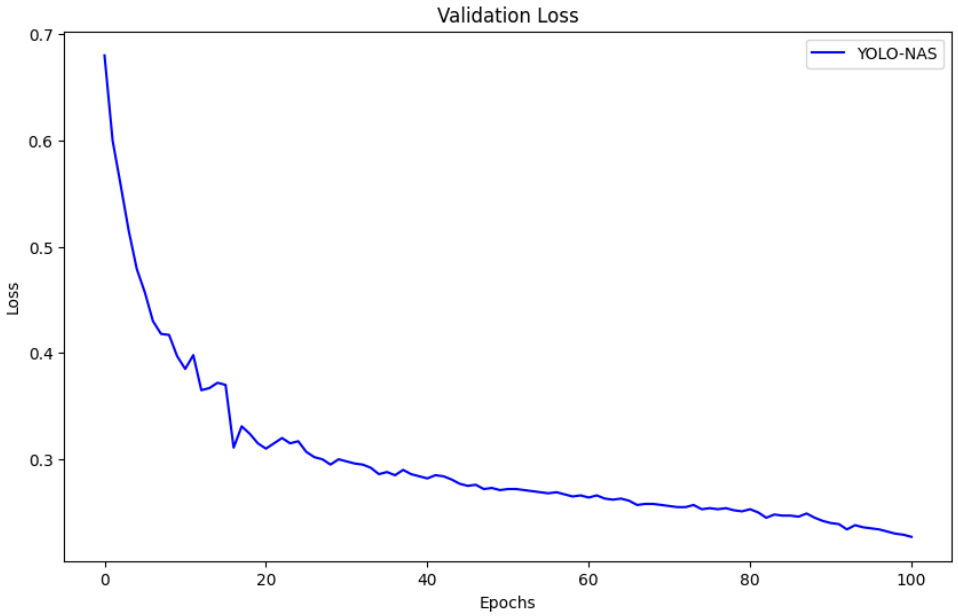
A seguir, foram extraídos os resultados dos sete vídeos selecionados anteriormente para teste e já mencionados na Seção 4.1. Com isso, foram realizadas as detecções e classificações utilizando a rede YOLO-NAS com treinamento, gerando a matriz de confusão mostrada na Tabela 13.

Também nesse caso os *labels* de caminhões não foram detectados devido aos problemas já listados na Seção 4.1. Por essa questão não foram calculadas métricas dos caminhões. Além disso, todos os ônibus foram detectados como carros. Em relação as demais classes, semelhante com o que ocorreu com a detecção utilizando a rede YOLOv7 com treinamento e a rede YOLOv5 com treinamento, a maioria dos *labels* foram detectados corretamente, sendo a segunda classe mais recorrente a não detecção como fundo (BG FN), em consequência das áreas de detecção descritas na Subseção 3.2.1.

Tabela 12 – Hiperparâmetros para o treinamento da YOLO-NAS

HIPERPARÂMETRO	VALOR UTILIZADO
Image Size	3840x1920
Batch Size	8
Epochs	100
Loss	0,02
lr	0,01
Solver	SGD (0.937 momentum)
hsv	h: 0,015 s: 0,7 v: 0,4
Translate	0,2
Scale	0,5
Flip left-right	0,5
Mosaic	1,0

Figura 27 – Gráfico de Perda de Validação (Validation Loss)



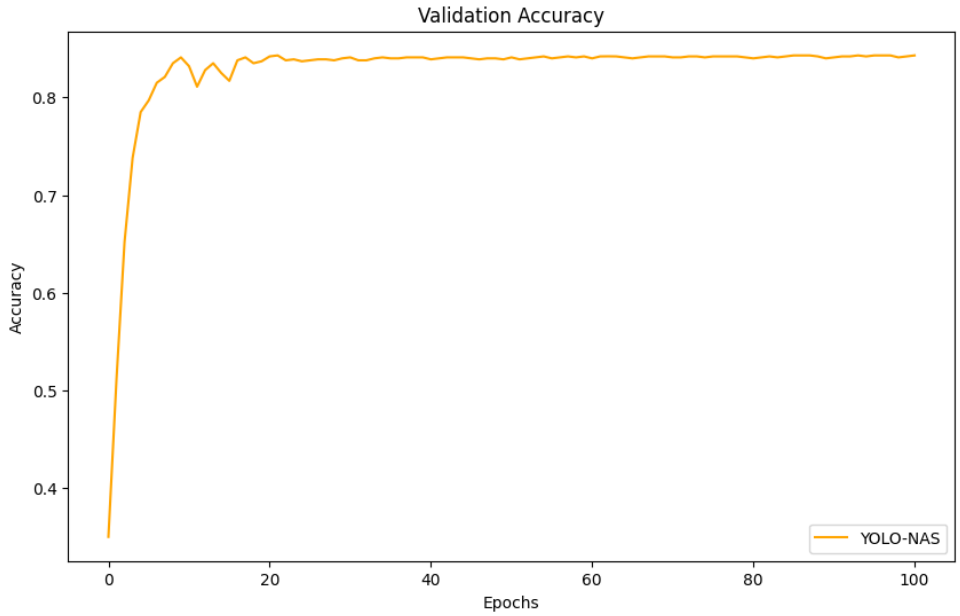
Fonte: do Autor

A partir da matriz de confusão, foram calculados a Precisão, o *Recall* e o *F1-Score*, para todas as detecções e também individualizados por classe conforme a Tabela 14.

Por fim, foram obtidas as métricas mAP@.5, mAP@.5:.95 (Tabela 15).

Após a análise entre a rede YOLOv7 com treinamento, a rede YOLOv5 com treinamento e a rede YOLO-NAS com treinamento, a rede YOLOv7 apresenta um desempenho superior em Precisão geral 88,9% frente a 69,8% da YOLOv5 e 71,2% da YOLO-NAS, a YOLO-NAS apresenta melhor precisão nas classes de carros 93,3% e motocicletas 99% se

Figura 28 – Gráfico de Perda de Validação (Validation Loss)



Fonte: do Autor

Tabela 13 – Matriz de Confusão - YOLO-NAS com treinamento

DETECTADO		**ROTULADO**						
		car	bus	person	truck	bike	moto	BG FP
	car	84743	978	0	0	0	239	3721
	bus	0	0	0	0	0	0	0
	person	0	0	35016	0	454	48	7007
	truck	0	0	0	0	0	0	0
	bike	0	0	0	0	40402	239	3250
	moto	0	0	0	0	0	4205	1352
	BG FN	4460	0	8754	548	4539	48	0

comparada as outras redes. Em relação à Revocação a YOLOv7 apresenta 80,8% contra 60,4% da YOLOv5 e 34,5% da YOLO-NAS, com esse resultado superior também se confirmando nas classes individuais. A YOLOv7 também apresenta melhores resultados de mAP@.5 e mAP@.5:.95 com 86,3% e 43,6% respectivamente, já a YOLOv5 apresentou 63,8% para mAP@.5 e 31,5% para mAP@.5:.95 e a YOLO-NAS apresentou um desempenho de 34% para mAP@.5 e 17,9% para mAP@.5:.95. Importante ainda considerar a detecção errônea da classe ônibus por parte da YOLO-NAS. Além desses fatores a YOLO-NAS apresentou um desempenho computacional melhor se comparado as outras redes.

Tabela 14 – Precisão, *Recall* e *F1-Score* - YOLO-NAS com treinamento

	Precisão	**Recall**	**F1-Score**
Total	0,712	0,345	0,465
car	0,933	0,704	0,802
bus	0	0	NA
person	0,760	0,328	0,458
bike	0,875	0,424	0,571
motorcycle	0,990	0,267	0,421

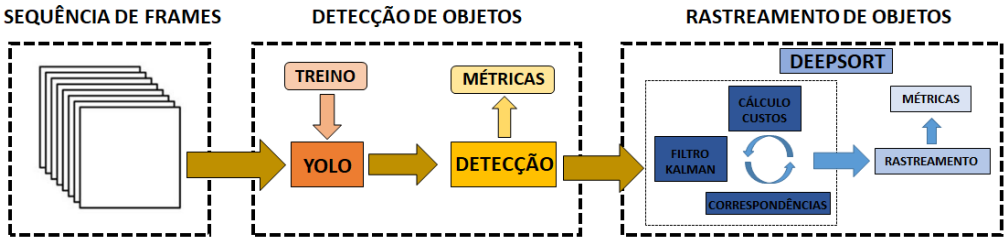
Tabela 15 – mAP@.5 e mAP@.5:.95 - YOLO-NAS com treinamento

	mAP@.5	**mAP@.5:.95**
Total	0,340	0,179
car	0,705	0,414
bus	0	0
person	0,315	0,145
bike	0,411	0,217
motorcycle	0,270	0,119

4.4 Rastreamento utilizando DEEPSORT

Após a detecção e identificação dos objetos, foi aplicado o rastreamento através do algoritmo DEEP-SORT Wojke, Bewley e Paulus (2017) com o *pipeline* de detecção e rastreamento conforme o fluxograma da Figura 29. Também foram calculadas as métricas MOTA e MOTP para a detecção, identificação e rastreamento dos objetos para as redes YOLOv7, YOLOv5 e YOLO-NAS, todas com treinamento, segundo as Tabelas 16, 17, 18.

Figura 29 – *Pipeline* de detecção e rastreamento de objetos



Fonte: do Autor

4.5 Estudo comparativo

Obtidas todas as métricas propostas nesse trabalho é possível realizar um estudo comparativo entre as três redes YOLO trabalhadas nas seções anteriores: YOLOv7, YOLOv5 e YOLO-NAS, todas treinadas. Nesta seção será apresentada uma comparação direta

Tabela 16 – MOTA e MOTP - YOLOv7 com treinamento

	MOTA	MOTP
Total	0,871	0,795
car	0,902	0,889
bus	0,946	0,762
person	0,761	0,752
bike	0,863	0,659
motorcycle	0,907	0,868

Tabela 17 – MOTA e MOTP - YOLOv5 com treinamento

	MOTA	MOTP
Total	0,783	0,654
car	0,852	0,801
bus	0,602	0,204
person	0,701	0,605
bike	0,752	0,703
motorcycle	0,804	0,752

Tabela 18 – MOTA e MOTP - YOLO-NAS com treinamento

	MOTA	MOTP
Total	0,700	0,333
car	0,920	0,692
bus	0	0
person	0,747	0,314
bike	0,862	0,414
motorcycle	0,981	0,261

entre as métricas já supracitadas. Primeiramente, a Tabela 19 apresenta a comparação entre as métricas totais.

Tabela 19 – Comparação entre as métricas totais

	YOLOv7	YOLOv5	YOLO-NAS
Precisão	0,889	0,698	0,712
Recall	0,808	0,604	0,345
F1-Score	0,847	0,648	0,465
mAP@.5	0,863	0,638	0,340
mAP@.5:.95	0,436	0,315	0,179
MOTA	0,871	0,783	0,700
MOTP	0,795	0,654	0,333

A Tabela 20 apresenta a comparação entre as métricas dos carros.

A Tabela 21 apresenta a comparação entre as métricas dos ônibus.

Tabela 20 – Comparação entre as métricas dos carros

	YOLOv7	YOLOv5	YOLO-NAS
Precisão	0,919	0,755	0,933
Recall	0,903	0,847	0,704
F1-Score	0,911	0,798	0,802
mAP@.5	0,925	0,859	0,705
mAP@.5:.95	0,610	0,544	0,414
MOTA	0,902	0,852	0,920
MOTP	0,889	0,801	0,692

Tabela 21 – Comparação entre as métricas dos ônibus

	YOLOv7	YOLOv5	YOLO-NAS
Precisão	0,951	0,638	0
Recall	0,787	0,171	0
F1-Score	0,861	0,270	NA
mAP@.5	0,907	0,275	0
mAP@.5:.95	0,244	0,057	0
MOTA	0,946	0,602	0
MOTP	0,762	0,204	0

A Tabela 22 apresenta a comparação entre as métricas das pessoas.

Tabela 22 – Comparação entre as métricas das pessoas

	YOLOv7	YOLOv5	YOLO-NAS
Precisão	0,778	0,556	0,760
Recall	0,777	0,643	0,328
F1-Score	0,777	0,596	0,458
mAP@.5	0,789	0,621	0,315
mAP@.5:.95	0,344	0,254	0,145
MOTA	0,761	0,701	0,747
MOTP	0,752	0,605	0,314

A Tabela 23 apresenta a comparação entre as métricas das bicicletas.

A Tabela 24 apresenta a comparação entre as métricas das motocicletas.

A partir das métricas geradas é possível realizar uma discussão sobre os resultados obtidos nesse trabalho. Primeiramente, foi demonstrada a necessidade do treinamento para a obtenção de melhores métricas através do estudo realizado na rede YOLOv7. A rede treinada obteve um melhor desempenho em todas as métricas propostas na Seção 2.3.

Também é possível afirmar que a YOLOv7 obteve um melhor desempenho geral do que as outras redes, porém para as classes de carros e motocicletas a YOLO-NAS apresentou precisão maior que a YOLOv7.

Tabela 23 – Comparação entre as métricas das bicicletas

	YOLOv7	YOLOv5	YOLO-NAS
Precisão	0,879	0,729	0,875
Recall	0,688	0,591	0,424
F1-Score	0,772	0,653	0,571
mAP@.5	0,769	0,591	0,411
mAP@.5:.95	0,466	0,661	0,217
MOTA	0,863	0,752	0,862
MOTP	0,659	0,703	0,414

Tabela 24 – Comparação entre as métricas das motocicletas

	YOLOv7	YOLOv5	YOLO-NAS
Precisão	0,919	0,812	0,990
Recall	0,886	0,768	0,267
F1-Score	0,902	0,789	0,421
mAP@.5	0,926	0,768	0,270
mAP@.5:.95	0,515	0,361	0,119
MOTA	0,907	0,804	0,981
MOTP	0,868	0,752	0,261

É evidente que esse estudo não visa definir a rede YOLOv7 como uma rede melhor que YOLOv5 ou a YOLO-NAS. Não foram discutidos outros fatores como o custo computacional ou a complexidade para a aplicação. Esse estudo demonstra que para a situação específica proposta nesse trabalho, imagens ominidirecionais no contexto de trânsito e com as restrições propostas no Capítulo 3, a rede YOLOv7 apresenta-se como a melhor solução.

Também cria-se através desse estudo um *benchmark* a ser utilizado por outros trabalhos, apresentando uma base de dados com suas anotações, as redes utilizadas, seus hiperparâmetros e os resultados obtidos.

4.6 Comparação com outros trabalhos

Apesar de não ser possível uma comparação direta deste estudo com outros, pelo fato de não haver trabalhos que empregam imagens omnidirecionais com o mesmo objetivo, uma comparação qualitativa será realizada para efeito de avaliação deste trabalho em relação a outros existentes na literatura.

Desta forma, trabalhos que utilizam outras bases de dados e *benchmarks*, e que também empregam YOLOv7 na detecção de objetos do trânsito, serão aqui mencionados e brevemente discutidos.

Em Zhang et al. (2023) são usadas imagens perspectivas para a detecção de objetos do trânsito utilizando uma versão da YOLOv7. Os objetos detectados são carros, ônibus e caminhões e o intuito do trabalho era o desenvolvimento e validação uma versão da YOLOv7, chamada YOLOv7. Os valores obtidos para as métricas de precisão, revocação e F1-Score são, respectivamente 92,5%, 80% e 85,8%.

Já em Yu et al. (2024), o foco está na detecção de carros, ônibus, motocicletas e caminhões com o objetivo de segurança no transporte de produtos químicos. Nesse caso, os valores para precisão, revocação e F1-Score alcançam 81,5%, 74,4% e 77,6%. As imagens utilizadas no trabalho também são do tipo plana.

O trabalho de Tang, Yang e Tian (2023) emprega a YOLOv7 com imagens em perspectiva a fim de realizar a detecção de pedestres a longas distâncias e com oclusão. A métrica precisão fica em torno de 50,1%, enquanto a revocação é de 32,2% e F1-Score de 39,2%.

Considerando-se que o estudo aqui desenvolvido, usando imagens omnidirecionais e a rede YOLOv7 com treinamento, atingiu uma precisão de 88,9%, uma revocação de 80,8% e um F1-Score de 84,7%, além das métricas de MOTA com 87,1% e MOTP com 79,5%, quando a rede de detecção foi combinada com o rastredor DEEPSORT, pode-se dizer que este trabalho apresenta desempenho semelhante, utilizando imagens ominidirecionais que possuem as distorções discutidas na Subseção 2.1.1, pode-se citar também a ocorrência de oclusões e também da detecção em áreas de interesse, dicutidas em 3.2.1.

Portanto, tal resultado mostra que a abordagem proposta representa um opção para sua utilização como ferramenta de monitoramento de tráfego no contexto de cidades inteligentes, utilizando uma infraestrutura mais enxuta, com apenas uma câmera e consequêntemente com menor tráfego de dados. A principal diferença, com relação a outros trabalhos, é que, nesse caso, a detecção e rastreamento de elementos nas vias são feitas utilizando-se imagens ominidirecionais, uma abordagem até então pouco explorada.

5 Conclusões e Trabalhos Futuros

No presente trabalho, foi realizada uma discussão sobre a importância do controle do tráfego no contexto das cidades inteligentes e a utilização das imagens ominidirecionais como uma opção viável, devido aos menores custos de instalação e transmissão de dados.

Também foi realizada uma revisão bibliográfica sobre as imagens ominidirecionais, seu processo de planificação e as distorções geradas a partir desse processo. Em sequência, realizou-se uma revisão bibliográfica sobre as redes YOLOv5, YOLOv7 e YOLO-NAS além do rastreador DEEPSORT.

Como um dos objetivos e contribuições desse trabalho, foi gerada uma base de dados contendo imagens omnidirecionais de vias públicas, onde são observados veículos, pedestres e ciclistas.Para isso, foram capturadas e rotuladas imagens omnidirecionais (360° de campo visual) para a detecção e rastreamento de elementos de trânsito no contexto de mobilidade urbana, buscando oferecer gratuitamente uma base de dados com vídeos e imagens para a comunidade científica. Tal base foi nomeada VV360 e é apresentada como uma opção para trabalhos em detecção e rastreamentos de objetos no trânsito em imagens omnidirecionais.

A seguir, como segunda contribuição, um estudo comparativo entre as três redes YOLOv5, YOLOv7 e YOLO-NAS, associadas ao rastreador DEEPSORT, foi conduzido a fim de avaliar a possibilidade da utilização de imagens omnidirecionais para monitoramento de tráfego e fornecer um novo *benchmark* para trabalhos futuros que decidam utilizar a base de dados gerada. Assim, foi realizada uma apresentação das métricas utilizadas (precisão, revocação, F1-Score, mAP@0.5, mAP@0.5:95, MOTA e MOTP) e apresentado o *pipeline* de detecção e rastreamento utilizando as redes YOLOv7 (com e sem treinamento), YOLOv5 e YOLO-NAS.

No estudo comparativo supracitado, o melhor resultado foi obtido utilizando-se a rede YOLOv7 com treinamento, alcançando-se uma precisão de 88,9%, uma revocação de 80,8% e um F1-Score de 84,7%.

Comparando qualitativamente esses resultados com outros trabalhos que empregam imagens perspectivas e a rede YOLOv7 para detecção de elementos associados ao trânsito, observa-se que este trabalho apresenta desempenho similar, porém utilizando imagens ominidirecionais.

Isso indica que a abordagem proposta obteve resultados condizentes com outros trabalhos que possuem objetivo semelhante. Conforme já comentado anteriormente, vale ressaltar que a abordagem apresentada neste trabalho usa imagens omnidirecionais, en-

quanto os demais trabalhos citados utilizam imagens perspectivas, as quais não apresentam as distorções que normalmente dificultam o processo de detecção.

Como trabalhos futuros, pretende-se melhorar o desempenho do *pipeline* usando-se mais vídeos para treinamento, de forma a aumentar as amostras de classes menos frequentes, assim como o estudo e a realização de balanceamento das classes no processo de treinamento, utilizando-se a técnica SMOTE gerando amostras sintéticas das classes minoritárias para equilibrar a distribuição das classes.

Espera-se também testar outras combinações de detecção e rastreamento de veículos e pedestres para aplicação em sistemas de mobilidade urbana, avaliando o seu desempenho em relação ao tempo de processamento, precisão e acurácia. Outro objetivo será a captura e preparação de novas imagens e vídeos omnidirecionais, disponibilizando novas bases para a comunidade, com mais amostras de veículos que não obtiveram muitos exemplares nesta primeira base de dados, como o ocorrido com os caminhões.

Também deseja-se utilizar métodos de anotação supervisionada, o que resultará em um trabalho mais rápido e menos custoso, utilizando para isso a ferramenta existente no CVAT, assim como estudar a utilização da validação cruzada no treinamento das redes. Além disso, pode ser interessante a realização de um estudo mais aprofundado de como a distorção das imagens ominidirecionais afeta o desempenho da detecção e rastreamento, prevendo assim, suas limitações para o uso.

Por fim, deseja-se também utilizar os resultados do rastreamento para novas aplicações como cálculo de velocidade de veículos, cálculo de métricas de caracterização do fluxo do trânsito, detecção de conversões em cruzamentos e controle automático semafórico.

Referências

AL, K. et. *Envisaging the Future of Cities - World Cities Report 2022*. [S.l.]: ONU, 2022. Citado na página 15.

ANDRUSSOW, I. et al. *Minsight: A Fingertip-Sized Vision-Based Tactile Sensor for Robotic Manipulation*. 2023. Disponível em: <https://arxiv.org/abs/2304.10990>. Citado na página 16.

AZEVEDO, R. G. de A. et al. Visual distortions in 360° videos. *IEEE Transactions on Circuits and Systems for Video Technology*, Institute of Electrical and Electronics Engineers (IEEE), v. 30, n. 8, p. 2524–2537, aug 2020. Disponível em: <https://doi.org/10.1109%2Ftcsvt.2019.2927344>. Citado 2 vezes nas páginas 17 e 21.

BERNARDIN, K.; STIEFELHAGEN, R. Evaluating multiple object tracking performance: The clear mot metrics. *EURASIP Journal on Image and Video Processing*, v. 2008, 01 2008. Citado na página 34.

BEWLEY, A. et al. Simple online and realtime tracking. In: *2016 IEEE International Conference on Image Processing (ICIP)*. IEEE, 2016. Disponível em: <https://doi.org/10.1109%2Ficip.2016.7533003>. Citado 2 vezes nas páginas 25 e 30.

BOCHKOVSKIY, A.; WANG, C.-Y.; LIAO, H.-Y. M. *YOLOv4: Optimal Speed and Accuracy of Object Detection*. arXiv, 2020. Disponível em: <https://arxiv.org/abs/2004.10934>. Citado na página 27.

BOESCH, G. *Yolov7: The Most Powerful Object Detection Algorithm (2023 guide)*. 2023. Disponível em: <https://viso.ai/deep-learning/yolov7-guide/>. Citado na página 29.

BRASIL RAFAEL H. E MACHADO, A. M. C. Detecção de avanço de semáforo vermelho utilizando câmera embarcada em veículos automotores. In: *Anais - 7º Simpósio de Instrumentação e Imagens Médicas (SIIM) / 6º Simpósio de Processamento de Sinais da UNICAMP (SPS-UNICAMP'2015)*. [s.n.], 2014. Disponível em: <https://www.sps.fee.unicamp.br/anais/vol01/VSPS_a36_RBrasil.pdf>. Citado na página 35.

CHEN, D. et al. Multi-View Human Pose Estimation using Modified Five-point Skeleton Model. p. 17–19, 2007. Citado na página 16.

CORTEZ, D. E. d. S. Desenvolvimento de um sistema de controle de tráfego inteligente baseado em visão computacional. In: *Dissertação (mestrado) – UFRN/ Programa de Pós-graduação em Tecnologia da Informação*. [s.n.], 2022. Disponível em: <https://repositorio.ufrn.br/bitstream/123456789/47158/1/Desenvolvimentosistemacontrole_Cortez_2022.pdf>. Citado na página 35.

CURTIN, D. P. The textbook of digital photography. 2004. Citado na página 16.

CVAT | Computer Vision Annotation Tool. <https://https://www.cvat.ai/>. Accessed: 2023-01-15. Citado 2 vezes nas páginas 37 e 40.

ESRI. *CUBE - ArcMap*. 2024. Urlhttps://desktop.arcgis.com/en/arcmap/latest/map/projections/cube.htm. Citado na página 22.

FACHIN, O. In: *Fundamentos de Metodologia*. [S.l.: s.n.], 2001. Citado na página 45.

FELZENSZWALB, P.; HUTTENLOCHER, D. International journal of computer vision 59. In: *Efficient Graph-Based Image Segmentation*. [S.l.: s.n.], 2004. p. 167–181. Citado na página 25.

FURUKAWA, Y.; HERNáNDEZ, C. [S.l.: s.n.], 2015. Citado na página 20.

G., D. *A deeper look at 360 video projections*. 2021. Urlhttps://www.trekview.org/blog/2021/projection-type-360-photography/. Citado 2 vezes nas páginas 23 e 42.

GAVA, C. C. et al. Nonlinear control techniques and omnidirectional vision for team formation on cooperative robotics. In: *Proceedings 2007 IEEE International Conference on Robotics and Automation*. [S.l.: s.n.], 2007. p. 2409–2414. Citado na página 16.

GOODFELLOW YOSHUA BENGIO, A. C. I. Deep learning. MIT Press, 2016. Citado na página 23.

GOPROINC. *GoPro Max*. 2023. Urlhttps://www.goprobr.com/chdhz-202-rx-max/p. Citado na página 19.

GORGULHO CRISTIANE FERNANDES; TREDINNICK, M. R. A. d. C. *O Controle de Tráfego em Cidades Inteligentes: um panorama dos depósitos de patente no Brasil e no mundo.* Rio de Janeiro, RJ: INPI, 2020. Citado na página 16.

GUNJAL, P. R. et al. Moving object tracking using kalman filter. In: *2018 International Conference On Advances in Communication and Computing Technology (ICACCT)*. [S.l.: s.n.], 2018. p. 544–547. Citado na página 30.

GWAK, J.; SAVARESE, S.; BOHG, J. *Minkowski Tracker: A Sparse Spatio-Temporal R-CNN for Joint Object Detection and Tracking*. 2022. Disponível em: <https://arxiv.org/abs/2208.10056>. Citado na página 24.

GäCHTER, S. Mirror design for an omnidirectional camera with a uniform cylindrical projection when using the svavisca sensor. 01 2001. Citado 2 vezes nas páginas 19 e 20.

HIRAGA ALAN KAZUO; DA SILVA, F. A. A. A. O. Algoritmos para construÇÃo de panorama de imagens 360 e visualizaÇÃo. *Jornal Unoeste*, 2013. Citado na página 16.

HOSANG, J. H.; BENENSON, R.; SCHIELE, B. Learning non-maximum suppression. *CoRR*, abs/1705.02950, 2017. Disponível em: <http://arxiv.org/abs/1705.02950>. Citado na página 26.

HU, H. et al. Deep 360 pilot: Learning a deep agent for piloting through 360° sports video. *CoRR*, abs/1705.01759, 2017. Disponível em: <http://arxiv.org/abs/1705.01759>. Citado na página 35.

HUANG, X. et al. *PP-YOLOv2: A Practical Object Detector*. arXiv, 2021. Disponível em: <https://arxiv.org/abs/2104.10419>. Citado na página 27.

IBGE, I. B. de Geografia e E. *Atlas Geográfico Escolar, As projeções cartográficas*. 2024. Urlhttps://atlasescolar.ibge.gov.br/cartografia/21733-as-projecoes-cartograficas.html. Citado na página 20.

ISHIGURO, H.; UEDA, K.; TSUJI, S. Omnidirectional visual information for navigating a mobile robot. In: *[1993] Proceedings IEEE International Conference on Robotics and Automation*. [S.l.: s.n.], 1993. p. 799–804 vol.1. Citado na página 16.

JESUS, F. S. de. *Tipos de projeções cartográficas: equivalentes, conformes, equidistantes e afiláticas*. 2018. Disponível em: <https://www.geografiaopinativa.com.br/2018/03/tipos-de-projecoes-cartograficas-equivalentes-conformes-equidistantes-e-afilaticas.html>. Citado 2 vezes nas páginas 21 e 22.

JOCHER, G. et al. *ultralytics/yolov5: v7.0 - YOLOv5 SOTA Realtime Instance Segmentation*. Zenodo, 2022. Disponível em: <https://doi.org/10.5281/zenodo.7347926>. Citado 2 vezes nas páginas 27 e 50.

JULIER, S.; UHLMANN, J.; DURRANT-WHYTE, H. A new approach for filtering nonlinear systems. In: *Proceedings of 1995 American Control Conference - ACC'95*. [S.l.: s.n.], 1995. v. 3, p. 1628–1632 vol.3. Citado na página 24.

JúNIOR, L. d. S. B. Análise de veículos em cruzamentos com semáforos utilizando deep learning. In: *Monografia (graduação) – UFPB*. [S.l.: s.n.], 2018. Citado na página 35.

KASHANI, S.; IVRY, A. *Deep Learning Interviews: Hundreds of fully solved job interview questions from a wide range of key topics in AI*. 2022. Citado na página 23.

KATAOKA, H. et al. Drive video analysis for the detection of traffic near-miss incidents. In: *2018 IEEE International Conference on Robotics and Automation (ICRA)*. [S.l.: s.n.], 2018. p. 3421–3428. Citado na página 35.

KHVEDCHENYA EUGENE E SAHOTA, H. Yolo-nas by deci achieves state-of-the-art performance on object detection using neural architecture search. In: . [s.n.], 2023. Disponível em: <https://deci.ai/blog/yolo-nas-object-detection-foundation-model/>. Citado 3 vezes nas páginas 29, 30 e 53.

KON FáBIO; SANTANA, E. F. Z. Computação aplicada a cidades inteligentes: como dados, serviços e aplicações podem melhorar a qualidade de vida nas cidades. *Sociedade Brasileira de Computação*, 2017. Citado na página 15.

KRIZHEVSKY, A.; SUTSKEVER, I.; HINTON, G. Imagenet classification with deep convolutional neural networks. In: *Proceedings of the 25th International Conference on Neural Information Processing Systems - Volume 1*. [S.l.]: Curran Associates Inc., 2012. Citado na página 24.

LABELBOX. <https://labelbox.com/>. Accessed: 2023-06-17. Citado na página 39.

LAVRENKO, T. et al. *Real-Time Detection and Classification for a 360°-Camera Using a YOLO Algorithm*. 2024. Disponível em: <https://www.cea-wismar.de/asim2021/tagungsband/data/ASIM_WS_2021_paper_33.pdf>. Citado na página 35.

LAZZARETTI K., S. S. . B. F. F. M. H. P. V. *Cidades inteligentes: insights e contribuições das pesquisas brasileiras.* Revista Brasileira de Gestão Urbana, v. 11, e20190118, 2019. Disponível em: <https://www.scielo.br/j/urbe/a/3LscvBK8vN86Q3fyFvzx7Fw/?lang=pt&format=pdf>. Citado na página 15.

LECUN, Y.; BENGIO, Y.; HINTON, G. Learning representations by back-propagating errors. Nature, 2015. Citado na página 24.

LI, C. et al. *YOLOv6: A Single-Stage Object Detection Framework for Industrial Applications*. 2022. Citado na página 27.

LIAO, Y.; XIE, J.; GEIGER, A. *KITTI-360: A Novel Dataset and Benchmarks for Urban Scene Understanding in 2D and 3D.* arXiv, 2021. Disponível em: <https://arxiv.org/abs/2109.13410>. Citado na página 35.

LING, H.; FIDLER, S. *Teaching Machines to Describe Images via Natural Language Feedback*. 2017. Citado na página 25.

LORENTE Òscar; RIERA, I.; RANA, A. *Image Classification with Classic and Deep Learning Techniques*. 2021. Disponível em: <https://arxiv.org/abs/2105.04895>. Citado na página 24.

LUFFCAINC. *DeepSORT: SORT with a Deep Association Metric*. 2024. Urlhttps://www.luffca.com/2023/05/multiple-object-tracking-deepsort/. Citado na página 31.

MANE, S.; MANGALE, S. Moving object detection and tracking using convolutional neural networks. In: *2018 Second International Conference on Intelligent Computing and Control Systems (ICICCS)*. [S.l.: s.n.], 2018. p. 1809–1813. Citado na página 24.

MARIANO, D. Métricas de avaliação em machine learning: acurácia, sensibilidade, precisão, especificidade e f-score. In: _____. [S.l.: s.n.], 2021. ISBN 9786599275326. Citado 2 vezes nas páginas 31 e 33.

MAUGEY, T. Chapter 2 - acquisition, representation, and rendering of omnidirectional videos. In: VALENZISE, G. et al. (Ed.). *Immersive Video Technologies*. Academic Press, 2023. p. 27–48. ISBN 978-0-323-91755-1. Disponível em: <https://www.sciencedirect.com/science/article/pii/B9780323917551000080>. Citado na página 20.

MOHAMED, A.; HEMEIDA, A.; HASSAN, M. Image classification based deep learning: A review. *Aswan University Journal of Sciences and Technology*, v. 2, 06 2022. Citado na página 24.

MORGADO NUNO VASCONCELOS, T. L. P.; WANG, O. Self-supervised generation of spatial audio for 360deg *video.In : Neural Information Processing Systems (NIPS)*.[*S.l.* : *s.n.*], 2018. *Citadonapágina*35.

MORGADO, P.; LI, Y.; VASCONCELOS, N. *Learning Representations from Audio-Visual Spatial Alignment*. arXiv, 2020. Disponível em: <https://arxiv.org/abs/2011.01819>. Citado na página 35.

NIR, T.; KARPEL, N. Example based learning of image stitching for an omni-directional camera using a variational optical flow methodology. 05 2008. Citado na página 20.

OLIVATTO, T. F. Identificação automática de rampas de acessibilidade apoiada por visão computacional a partir de imagens panorâmicas street-level. In: *Dissertação (pós-graduação) – UFSCar/ Programa de Pós-Graduação em Engenharia Urbana*. [S.l.: s.n.], 2021. Citado na página 35.

PAGARE, R.; SHINDE, A. A study on image annotation techniques. *International Journal of Computer Applications*, v. 37, p. 42–45, 01 2012. Citado na página 39.

PANDA, C. Object detection and tracking using faster r-cnn. *International Journal of Recent Technology and Engineering (IJRTE)*, v. 8, p. 4894–4900, 09 2019. Citado na página 24.

PEI, Y. et al. *An Elementary Introduction to Kalman Filtering*. 2019. Citado na página 24.

PEREIRA, C. A. R. P. N. S. Deep learning conceitos e utilizaÇÃo nas diversas Áreas do conhecimento. *Universidade Evangélica de Goiás*, 2018. Disponível em: <http://repositorio.aee.edu.br/jspui/handle/aee/1104>. Citado na página 24.

PINHEIRO ALEX; DA SILVA, T. L. A importÂncia da visÃo computacional e suas aplicaÇÕes na mobilidade urbana. *IMED*, 2018. Citado na página 15.

REBINTH, A.; S, M. K. *Importance of Manual Image Annotation Tools and Free Datasets for Medical Research*. 2019. 1880-1885 p. Citado na página 39.

REDMON, J. et al. *You Only Look Once: Unified, Real-Time Object Detection*. arXiv, 2015. Disponível em: <https://arxiv.org/abs/1506.02640>. Citado 2 vezes nas páginas 25 e 26.

REDMON, J.; FARHADI, A. *YOLO9000: Better, Faster, Stronger*. arXiv, 2016. Disponível em: <https://arxiv.org/abs/1612.08242>. Citado na página 27.

REDMON, J.; FARHADI, A. *YOLOv3: An Incremental Improvement*. arXiv, 2018. Disponível em: <https://arxiv.org/abs/1804.02767>. Citado na página 27.

REN, S. et al. *Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks*. 2016. Citado na página 34.

RICOH Theta V, Product Description. <https://theta360.com/en/about/theta/v.html>. Accessed: 2023-01-08. Citado na página 37.

RUMELHART, D.; HINTON, G.; WILLIAMS, R. Deep learning. Nature, 1986. Citado na página 24.

SCARPARO, H. M.; SANTOS, C. C. dos; VASSALLO, R. F. Vv360 database: Vídeos omnidirecionais para detecção e rastreamento de elementos no trânsito. In: *2023 15th IEEE International Conference on Industry Applications (INDUSCON)*. [S.l.: s.n.], 2023. p. 1005–1012. Citado na página 45.

SCHROEDER, D. J. Chapter 9 - auxiliary optics for telescopes. In: SCHROEDER, D. J. (Ed.). *Astronomical Optics (Second Edition)*. Second edition. San Diego: Academic Press, 2000. p. 206–239. ISBN 978-0-12-629810-9. Disponível em: <https://www.sciencedirect.com/science/article/pii/B9780126298109500104>. Citado na página 16.

SHAFIEE M. J.; CHYWL, B. L. F. W. A. Fast yolo: A fast you only look once system for realtime embedded object detection in video. 2017. Disponível em: <https://arxiv.org/abs/1709.05943v1>. Citado 2 vezes nas páginas 26 e 27.

SHAH, D. *Mean Average Precision (mAP) Explained: Everything You Need to Know*. 2022. Urlhttps://www.v7labs.com/blog/mean-average-precision. Citado 2 vezes nas páginas 33 e 34.

SHI, J.; TOMASI. Good features to track. In: *1994 Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*. [S.l.: s.n.], 1994. p. 593–600. Citado na página 24.

SILVA, I. d. F. T. e. a. Noções básicas de cartografia. In: *Ministério do Planejamento e Orçamento / Instituto Brasileiro de Geografia e Estatística - IBGE / Diretoria de Geociências - DGC*. [S.l.: s.n.], 1998. Citado 2 vezes nas páginas 20 e 21.

SOHAN, M.; RAM, T.; CH, V. A review on yolov8 and its advancements. In: _____. [S.l.: s.n.], 2024. p. 529–545. ISBN 978-981-99-7999-8. Citado na página 27.

SU, Y.; JAYARAMAN, D.; GRAUMAN, K. Pano2vid: Automatic cinematography for watching 360° videos. *CoRR*, abs/1612.02335, 2016. Disponível em: <http://arxiv.org/abs/1612.02335>. Citado na página 35.

SUGIMOTO, N.; IKEHATA, S.; AIZAWA, K. *Intersection Prediction from Single 360° Image via Deep Detection of Possible Direction of Travel*. arXiv, 2022. Disponível em: <https://arxiv.org/abs/2204.04634>. Citado na página 35.

SUPERANNOTATE. <https://www.superannotate.com/>. Accessed: 2023-06-17. Citado na página 39.

TANG, F.; YANG, F.; TIAN, X. Long-distance person detection based on yolov7. 2023. ISSN 2079-9292. Citado na página 60.

TAYLOR, L. E.; MIRDANIES, M.; SAPUTRA, R. P. Optimized object tracking technique using kalman filter. *Journal of Mechatronics, Electrical Power, and Vehicular Technology*, National Research and Innovation Agency, v. 7, n. 1, p. 57–66, jul. 2016. ISSN 2087-3379. Disponível em: <http://dx.doi.org/10.14203/j.mev.2016.v7.57-66>. Citado na página 30.

TERVEN, J.; CóRDOVA-ESPARZA, D.-M.; ROMERO-GONZáLEZ, J.-A. A comprehensive review of yolo architectures in computer vision: From yolov1 to yolov8 and yolo-nas. *Machine Learning and Knowledge Extraction*, MDPI AG, v. 5, n. 4, p. 1680–1716, nov. 2023. ISSN 2504-4990. Disponível em: <http://dx.doi.org/10.3390/make5040083>. Citado na página 25.

TOšIć, I.; FROSSARD, P. Chapter 10 - spherical imaging in omnidirectional camera networks. In: AGHAJAN, H.; CAVALLARO, A. (Ed.). *Multi-Camera Networks*. Oxford: Academic Press, 2009. p. 239–264. ISBN 978-0-12-374633-7. Disponível em: <https://www.sciencedirect.com/science/article/pii/B9780123746337000124>. Citado na página 16.

V7LABS. <https://www.v7labs.com/>. Accessed: 2023-06-17. Citado na página 39.

VASSALLO, R. F.; SCHNEEBELI, H. J.; SANTOS-VICTOR, J. Visual servoing and appearance for navigation. *Robotics and Autonomous Systems*, v. 31, n. 1, p. 87–97, 2000. ISSN 0921-8890. Disponível em: <https://www.sciencedirect.com/science/article/pii/S0921889099000834>. Citado na página 16.

VIRTUAIS, B. R. e. *Saiba como é possível captar Fotografias 360*. 2020. Disponível em: <https://reaisevirtuais.com/2020/10/09/saiba-como-e-possivel-captar-fotografias-360/>. Citado na página 19.

VLASSIS, N. et al. Edge-based features from omnidirectional images for robot localization. In: *Proceedings 2001 ICRA. IEEE International Conference on Robotics and Automation (Cat. No.01CH37164)*. [S.l.: s.n.], 2001. v. 2, p. 1579–1584 vol.2. Citado na página 16.

WANG, A. et al. *YOLOv10: Real-Time End-to-End Object Detection*. 2024. Disponível em: <https://arxiv.org/abs/2405.14458>. Citado na página 27.

WANG, C.-Y.; BOCHKOVSKIY, A.; LIAO, H.-Y. M. *Repositório YOLOv7*. arXiv, 2022. Disponível em: <https://github.com/WongKinYiu/yolov7>. Citado na página 46.

WANG, C.-Y.; BOCHKOVSKIY, A.; LIAO, H.-Y. M. *YOLOv7: Trainable bag-of-freebies sets new state-of-the-art for real-time object detectors*. arXiv, 2022. Disponível em: <https://arxiv.org/abs/2207.02696>. Citado 4 vezes nas páginas 27, 28, 29 e 46.

WANG, C.-Y.; YEH, I.-H.; LIAO, H.-Y. M. *YOLOv9: Learning What You Want to Learn Using Programmable Gradient Information*. 2024. Disponível em: <https://arxiv.org/abs/2402.13616>. Citado na página 27.

WOJKE, N.; BEWLEY, A.; PAULUS, D. *Simple Online and Realtime Tracking with a Deep Association Metric*. arXiv, 2017. Disponível em: <https://arxiv.org/abs/1703.07402>. Citado 5 vezes nas páginas 25, 30, 31, 32 e 56.

XU, L.; HUANG, H.; LIU, J. Trafficqa: A question answering benchmark and an efficient network for video reasoning over traffic events. *CoRR*, abs/2103.15538, 2021. Disponível em: <https://arxiv.org/abs/2103.15538>. Citado na página 35.

XU, R. et al. A forest fire detection system based on ensemble learning. *Forests*, v. 12, p. 217, 02 2021. Citado 2 vezes nas páginas 28 e 29.

YU, C. et al. G-YOLO: A YOLOv7-based target detection algorithm for lightweight hazardous chemical vehicles. *PLoS One*, v. 19, n. 4, p. e0299959, abr. 2024. Citado na página 60.

ZHANG, Y. et al. Yolov7-rar for urban vehicle detection. 2023. Citado na página 60.

ZHANG, Y. et al. VIL-100: A new dataset and A baseline model for video instance lane detection. *CoRR*, abs/2108.08482, 2021. Disponível em: <https://arxiv.org/abs/2108.08482>. Citado na página 35.

©RICOHCOMPANY. *Theta homepage*. 2023. Urlhttps://theta360.com/en/. Citado na página 19.