ESCOLA NAVAL

talante bien faire

João Alexandre Basílio Martins

Seguimento de alvos utilizando veículos aéreos não tripulados para o apoio a operações militares

Dissertação para obtenção do Grau de Mestre em Ciências Militares Navais, na especialidade de Fuzileiros

Alfeite

2024

João Alexandre Basílio Martins

Seguimento de alvos utilizando veículos aéreos não tripulados para o apoio a operações militares

Dissertação para obtenção do Grau de Mestre em Ciências Militares Navais, na especialidade de Fuzileiros

Orientação de: Victor Lobo
Co-orientation of Vitor Rodrigues

O Aluno Mestrando,

Assinado por: **João Alexandre Basílio Martins**
Num. de Identificação: 14522089
Data: 2024.06.22 16:44:16+01'00'

João Martins

O Orientador,

Assinado por: **Victor José de Almeida e Sousa Lobo**
Num. de Identificação: 09302569
Data: 2024.07.03 16:24:40+01'00'

Victor Lobo

Alfeite
2024

Dedico este trabalho á minha família.

Agradecimentos

Durante a realização deste projeto de pesquisa, recebi apoio, orientação e estímulo de várias pessoas e instituições, desde o início até a conclusão. Expresso minha sincera gratidão a todos os envolvidos. Sou grato ao GMAR FZ Batista Pinto por permitir usufruir das gravações adicionais retiradas pelo próprio, o que foi essencial para o rumo do meu estudo.

Quero expressar minha profunda apreciação ao meu orientador, PROF Sousa Lobo e co-orientador 1TEN FZ Borges Rodrigues. Sua acessibilidade, preocupação, compartilhamento de experiências e conhecimentos, além das sugestões valiosas, foram de grande ajuda em momentos de obstáculos e incertezas.

Meus camaradas do curso CALM Manuel Armando Ferraz também merecem meu agradecimento, pois compartilhamos experiências e nos apoiamos durante esses cinco anos. Um reconhecimento especial aos camaradas da minha especialidade, ASPOF FZ Fé Santos e ASPOF FZ Fatela Figueiredo, que sempre estiveram ao meu lado.

À Escola Naval, Escola de Fuzileiros e Corpo de Fuzileiros, expresso meu agradecimento por todas as oportunidades de aprendizado e acesso. Por último, mas não menos importante, minha família e amigos merecem uma profunda gratidão por todo o apoio e encorajamento que me deram.

Resumo

Neste tralho é feito o seguimento de multiplos objectos (*Multiple Object Tracking* - MOT), de forma automática, a partir de videos gravados por uma câmera integrada num veículo aéreo não tripulado (*Unmanned Aerial Vehicle* - UAV). Para tal, fez-se uma pesquisa bibliográfica sobre os métodos mais usados para esta tarefa, e optou-se por usar o sistema ByteTrack para fazer o processamento. O sistema ByteTrack é de utilização muito simples, mas internamente usa o detector YOLOX para fazer o reconhecimento de alvos em cada imagem do video, seguido de um filtro de Kalman e várias métricas de distância para fazer a associação dos alvos nas diferentes imagens. Um dos aspectos críticos do sistema ByteTrack é o detector (YOLOX) que é uma rede neuronal cujos pesos são ajustados treinando o sistema com diversos conjuntos de dados. Numa primeira instância o sistema ByteTrack foi inicializado com parâmetros obtidos no treino com o conjunto de dados MOTChallange e MS COCO. Por forma a ajustar os pesos à realidade a ser testada (imagens de pessoas tiradas a partir de UAV), foi realizado treino adicional (trasnsfer learning e fine-tuning) com mais 3 conjuntos de dados. Esses dados incluiram imagens genéricas de pessoas (conjunto VISDRONE), imagens gravadas durante exercícios militares com cadetes na península de Troia e com fuzileiros da Força Nacional Destacada na Lituânia, e um conjunto de imagens "open source"com militares retirada de um site da internet. Finalmente, o desempenho do sistema de seguimento foi testado em imagens obtidas em ambiente militar na Ucrânia e na República Centro-Africana (RCA). O treino com estes conjuntos de dados foi feito usando o CooLab (da Google). Com as melhorias introduzidas neste trabalho, o desempenho nas imagens de teste foi bastante melhor que o do sistema genérico ByteTrack (inicializado com apenas MS COCO), abrindo perspectivas de poder ser utilizado operacionalmente no futuro.

Palavras-chave: Seguimento (Tracking), Seguimento de Múltiplos Objectos (MOT) Reconhecimento de Imagens, UAVs, VisDrone

Abstract

This work presents the automatic tracking of multiple objects (Multiple Object Tracking - MOT) from videos recorded by a camera integrated into an Unmanned Aerial Vehicle (UAV). For this purpose, a bibliographic research on the most used methods for this task was conducted, and the ByteTrack system was chosen for processing. The ByteTrack system is straightforward to use, but internally it employs the YOLOX detector for target recognition in each video frame, followed by a Kalman filter and various distance metrics for target association across frames. One of the critical aspects of the ByteTrack system is the detector (YOLOX), which is a neural network whose weights are adjusted by training the system with various datasets. Initially, the ByteTrack system was initialized with parameters obtained from training with the MOTChallenge and MS COCO datasets. To adapt the weights to the reality being tested (images of people taken from UAVs), additional training (transfer learning and fine-tuning) was performed with three more datasets. These datasets included generic images of people (VISDRONE dataset), images recorded during military exercises with cadets in the Troia Peninsula and with marines from the Detached National Force in Lithuania, and a set of "open source"images with military personnel taken from an internet site. Finally, the performance of the tracking system was tested on images obtained in military environments in Ukraine and the Central African Republic (CAR). Training with these datasets was performed using Google's Colab. With the improvements introduced in this work, the performance on test images was significantly better than that of the generic ByteTrack system (initialized with only MS COCO), opening perspectives for potential operational use in the future.

Keywords: Tracking, MOT, Image Recognition, UAVs, Visdrone

Índice

Lista de Figuras

Lista de Tabelas

Lista de Abreviaturas

MOT	**M**ulti **O**bject **T**racking
SOT	**S**ingle **O**bject **T**racking
FPS	**F**rame **P**er **S**econd
UAV	**U**nmaned **A**erial **V**ehicle
CNN	**C**onvulutinioal **N**eural **N**etworks
ViT	**Vi**sion **T**ransformer
DBT	**D**etection **B**ased **T**racking
DFT	**D**etection **F**ree **T**racking
IoU	**I**ntersection **O**ver **U**nion
DETR	**De**tection **Tr**ansformer
Re-ID	**R**e **ID**enditificação
RMN	**R**ede **M**ovimento **R**elativo
TAO	**T**racking **A**ny **O**bject
MS COCO	**M**icro**s**oft **C**ommon **O**bjects in **Co**ntext
KITTI	**K**arlsruhe **I**nstitute of **T**echnology and **T**oyota Technological **I**nstitute
MOTP	**M**ulti **O**bject **T**racking **P**recision
GT	**G**round **T**ruth
IDSW	**Id**entity **Sw**itches
CLEAR	**Cl**assification of **E**vents **A**ctivities and **R**elationships
IDTP	**Id**entity **T**rue **P**ositives
IDFP	**Id**entity **F**alse **P**ositives
IDFN	**Id**entity **F**alse **N**egatives
FN	**F**alsos **N**egativos
FP	**F**alsos **P**ositivos
MOTA	**M**ulti **O**bject **T**racking **A**ccuracy
RGB	**R**ed **G**reen **B**lue
YOLO	**Y**ou **O**nly **L**ook **O**nce
RCA	**R**épublica **C**entro **A**fricana
MT	**M**ostly **T**racked
PT	**P**artially **T**racked
ML	**M**ostly **L**ost

Capítulo 1

Introdução

Atualmente, os UAVs são utilizados para fins militares, comerciais e industriais. Um veiculo aéreo não tripulado pode se definir como o nome indica (UAV - *Unmanned Aerial Vehicle*) por uma aeronave sem um piloto humano a bordo (Ma'sum, Arrofi, Jati, Arifin, Kurniawan, Mursanto e Jatmiko 2013). No contexto militar, a obtenção de informações precisas e em tempo real é crucial para o sucesso das operações. Os UAVs desempenham um papel vital nesse aspeto, fornecendo informações, vigilância e reconhecimento (ISR - *Intelligence, Surveillance, and Reconnaissance*), o que permite aos comandantes tomar decisões informadas e estratégicas no campo de batalha.

Considerando a complexidade e a periculosidade do campo de batalha moderno, segundo Udeanu, Dobrescu e Oltean 2016 o uso de UAVs em operações militares é obrigatório. Uma aplicação importante da vigilância aérea para fins de segurança é detetar, classificar e efetuar seguimento de objetos em movimento na área observada. Para o desenvolvimento deste trabalho o alvo de estudo será métodos de seguimento de múltiplos objetos aplicados a cenários militares realistas.

Existem inúmeros modelos de algoritmos de seguimento disponíveis, e que serão abordados no estado de arte desta dissertação, assim como base de dados voltadas quer para fins militares ou imagens adquiridas através de veículos aéreos.

Esta dissertação mostrará como um modelo de seguimento de múltiplos alvos de arquitetura leve e atual (ByteTrack e YOLOX Nano) se comporta em situações de operações militares reais, a partir de imagens captadas por um veiculo aéreo não tripulado (UAV). Foi feita uma revisão do estado de arte atual acerca de datasets existentes para avaliar desempenho de modelos de seguimento, e foi adquirido por nós uma base de dados também que se aproxima da realidade de uma operação militar. Foram realizados vários testes com vários parâmetros, incluindo testes do

modelo com e sem treino, que nos levou á conclusão que apesar dos resultados medianos mas em conformidade com os valores do estado de arte para seguimento, que a tecnologia assim que for refinada terá grande capacidade de empregabilidade.

1.1 Motivação

A motivação para escolher o tema "Seguimento de alvos utilizando veículos aéreos não tripulados para o apoio a operações militares"para uma dissertação é impulsionada pela necessidade de aprimorar as capacidades de vigilância e segurança nas operações militares. Este tema é crucial devido à sua relevância em várias áreas:

A segurança nacional é a principal prioridade de qualquer nação. A capacidade de monitorizar e controlar fronteiras, identificar atividades suspeitas e garantir a segurança de instalações militares é vital para a proteção do território nacional. Os veículos aéreos não tripulados (UAVs) desempenham um papel crucial nesse contexto, permitindo a vigilância constante de grandes áreas com eficiência e precisão (Paucar, Morales, K. Pinto, Sánchez, Rodríguez, Gutierrez e Palacios 2018). A utilização de UAVs em operações de patrulha aérea e marítima (Jingbo Wang, K. Zhou, W. Xing, H. Li e Z. Yang 2023) facilita a deteção precoce de ameaças, aumentando a capacidade de resposta das forças de segurança.

"Os UAVs fazem contribuições significativas para a capacidade de combate das forças operacionais"(Udeanu, Dobrescu e Oltean 2016). Em operações de combate, o seguimento de objetos é essencial para missões de reconhecimento, operações especiais e minimizando danos colaterais

A utilização de UAVs em missões de seguimento de alvos e reconhecimento permite que as forças militares realizem tarefas perigosas sem colocar vidas humanas em risco. Isso é particularmente importante em cenários de combate onde a exposição direta ao inimigo pode resultar em baixas significativas.

Por fim a escolha deste tema para uma dissertação de mestrado é justificada pela importância estratégica e pela oportunidade de contribuir para o avanço da tecnologia militar.

1.2 Objetivos

Relativamente aos objetivos da dissertação e começando pelo principal, será propor um sistema de seguimento de alvos utilizando veículos aéreos não tripulados

para apoio às operações terrestres. E, portanto, a questão central que deriva deste objetivo será – qual o sistema de seguimento de alvos utilizando veículos aéreos não tripulados que deve ser utilizado para o apoio às operações terrestres?

Relativamente a objetivos específicos vão ser tidos em conta 4:

- Avaliar os sistemas de seguimento de alvos utilizando veículos aéreos não tripulados atualmente existentes;
- Identificar os algoritmos de deteção e seguimento de alvos passíveis de serem utilizados em veículos aéreos não tripulados para apoio às operações terrestres;
- Adaptar os algoritmos de deteção e seguimento de alvos usando veículos aéreos não tripulados para o problema em estudo (apoio às operações terrestres);
- Adquirir uma base de dados representativa do problema em estudo para aferir o desempenho dos algoritmos.

Relativamente às questões derivadas destes objetivos:

- Quais os sistemas de seguimento de alvos utilizando veículos aéreos não tripulados que teriam aplicabilidade em operações terrestres?
- Que algoritmos de deteção e seguimento de alvos podem ser adaptados e aplicados para apoio às operações terrestres?
- De que forma tenho de adaptar/alterar os algoritmos existentes por forma a conseguir detetar e seguir alvos em operações terrestres?
- Qual o desempenho (erro) obtido pelos algoritmos adaptados?

1.3 Estrutura da dissertação

A presente dissertação tem como objetivo principal investigar e propor soluções para o seguimento de múltiplos objetos no contexto militar. A estrutura da dissertação é organizada da seguinte forma:

- **Revisão de Literatura** - O segundo capítulo da dissertação consiste em uma revisão da literatura sobre o seguimento de múltiplos objetos. São discutidos vários aspetos, incluindo técnicas de associação de objetos e os desafios enfrentados no processo de seguimento;
- **Metodologia** - Este capítulo descreve a metodologia adotada no desenvolvimento da pesquisa. Inclui informações sobre o modelo escolhido e a explicação

detalhada de sua arquitetura. Além disso, são apresentados os conjuntos de dados utilizados para testar o modelo, juntamente com as configurações de teste e treino;

- **Discussão dos Resultados** - Neste capítulo serão apresentados os resultados de desempenho obtidos pelo algoritmo, será discutido a robustez que o algoritmo apresenta para diversos cenários, análise dos erros obtidos e comparação com trabalhos anteriores,
- **Conclusões** - No último capítulo, são apresentadas as conclusões da dissertação, juntamente com as contribuições da pesquisa para o campo do seguimento de múltiplos objetos no contexto militar. São discutidas as limitações do estudo e propostas para trabalhos futuros.

Capítulo 2

Revisão de Literatura

Neste capítulo será feita uma revisão da literatura referente ao processamento de imagens de vídeo para seguimento (*tracking*) de objetos (W. Luo, J. Xing, Milan, Xiaoqin Zhang, Wei Liu e T.-K. Kim 2021), em particular quando essas imagens são obtidas por UAV. Pela importância que têm nestas tarefas, é feita uma revisão sobre redes neuronais convolucionais (CNN) (Zewen Li, W. Yang, S. Peng e F. Liu 2020) e "Transformers"para visão (*Vision Transformers*) para reconhecimento de imagens, bem como dos diversos passos, datasets de referência, e métricas de desempenho que são usados ao fazer o *tracking* de múltiplos objetos (MOT),

2.1 Imagens e Vídeos obtidos por UAV

Um vídeo é essencialmente uma sequência de imagens em rápida sucessão. Cada *frame* de um vídeo é uma imagem estática capturada em um instante específico no tempo. As imagens usadas são *bitmaps* (conjuntos de pixels), e podem ser caracterizadas pela resolução [1], profundidade de côr (ou número de bits por pixel), comprimento e altura ou "aspect ratio". No processamento das imagens, é vulgar usar um formato igual para todas as imagens, pelo que por vezes é preciso redimensioná-las, de modo a terem todas o mesmo tamanho (nº total de pixels), *"aspect ratio"* (largura e altura), e profundidade de cor. O sistema ByteTrack (Y. Zhang, P. Sun et al. 2021) usado nesta dissertação faz este redimensionamento automático, pelo que podemos usar imagens com qualquer resolução.

Os vídeos gravados a partir de UAVs têm algumas características específicas que são amplamente discutidos em diversos artigos (Du et al. 2018b; Tang, Ni, Y.

[1]Em rigôr, aquilo que popularmente, mas impropriamente, se designa "resolução"é o número total de pixels. A resolução, propriamente dita, é o número de pixels por unidade de comprimento, normalmente "Dots Per Inch - DPI", ou "Pixels Per Inch - PPI", que em literatura popular é designada por "densidade"da imagem. Nesta dissertação, embora reconhecendo o erro, optou-se por usar o termo popular por ser mais comum.

Zhao, Y. Gu e W. Cao 2024; H. Yao, Qin e X. Chen 2019). Assim, ao processar as imagens de vídeo é preciso ter em atenção os seguintes aspetos:

- **Existência de Objetos Pequenos** - Devido á altitude em que os drones operam por vezes a definição (qualidade) não é das melhores tornando os alvos pouco perceptíveis;
- **Movimento da Câmara** - O movimento da câmara e do próprio drone pode prejudicar as imagens se estes não forem realizados de forma suave. Para o algoritmo será sempre mais difícil identificar um alvo que se move mais rápido do que o suposto;
- **Estabilidade da Imagem** - As imagens obtidas pelos UAV têm frequentemente um problema de estabilidade, devido à vibrações e movimentos rápidos do veículo. Por isso, é vulgar que as câmaras estejam montadas sobre estabilizadores que amortecem essas vibrações. No entanto, a qualidade desses estabilizadores pode variar consideravelmente, e muitas vezes, em vez de estabilizadores mecânicos, faz-se um pós processamento para "estabilização digital"(REF). A qualidade dos estabilizadores da câmara influencia também a qualidade de imagem, podendo por vezes resultar *blurs* que é habitual em imagens UAV;
- **Ângulo de Visão** - O ângulo de visão da câmara do UAV pode afetar a aparência dos objetos na imagem, especialmente em relação à sua forma e tamanho. Ângulos de visão amplos podem capturar uma área maior, mas podem distorcer a perspetiva, enquanto ângulos estreitos podem proporcionar uma visão mais detalhada, mas com menos contexto;
- **Iluminação** - Condições de iluminação variáveis, como sombras, reflexos e mudanças na luminosidade, podem afetar a qualidade das imagens e a deteção dos objetos;
- **Aplicabilidade prática** - Deve ser considerado nos algoritmos os problemas que acarreta operar com estes dispositivos em tempo-real assim como considerar integrar todo o processamento na plataforma.

Essas características das imagens capturadas por UAVs precisam ser consideradas e tratadas pelos algoritmos de seguimento para garantir um desempenho eficaz e preciso na deteção e no acompanhamento de objetos em ambientes aéreos.

2.2 Processamento de Imagens

Os algoritmos de processamento de imagem, como as redes neuronais convolucionais (CNNs)(Zewen Li, W. Yang, S. Peng e F. Liu 2020) e as *Transformers* (ViTs)(Vaswani et al. 2017), desempenham um papel fundamental nas tarefas de visão computacional. Esses algoritmos têm revolucionado a forma como as máquinas interpretam e compreendem imagens, permitindo uma ampla gama de aplicações em diversas áreas, como reconhecimento de objetos, deteção de padrões, segmentação de imagem e muito mais.

A importância desses algoritmos reside no fato de que eles permitem que as máquinas compreendam e interpretem o mundo visual de forma semelhante aos seres humanos, abrindo caminho para uma ampla gama de aplicações práticas em áreas como medicina, segurança, automação industrial, reconhecimento facial, veículos autónomos, entre outras(Zewen Li, W. Yang, S. Peng e F. Liu 2020).

2.2.1 Algoritmos CNN

A força das Redes Neuronais Profundas (*Deep Neural Networks* - DNN) reside em sua capacidade de aprender representações ricas e extrair características complexas e abstratas de seus *insputs*"(Ciaparrone, Luque Sánchez, Tabik, Troiano, Tagliaferri e Herrera 2020). As redes neuronais convolucionais (CNN) atualmente constituem o estado-da-arte na extração de características e são empregues em tarefas como classificação de imagens(Rawat e Zenghui Wang 2017), deteção de objetos(Dhillon e Verma 2020) e segmentação (Minaee, Boykov, Porikli, Plaza, Kehtarnavaz e Terzopoulos 2021). Como os métodos de aprendizagem profunda têm sido capazes de alcançar alto desempenho em muitas dessas tarefas, estamos progressivamente vendo-os a ser utilizados na maioria dos algoritmos MOT de melhor desempenho (**tr**; Aharon, Orfaig e Bobrovsky 2022; W. Luo, J. Xing, Milan, Xiaoqin Zhang, Wei Liu e T.-K. Kim 2021; Y. Zhang, P. Sun et al. 2021).

Em visão computacional, várias arquiteturas de redes neurais convolucionais (CNN) têm sido amplamente utilizadas. Entre elas, a LeNet(Prashanth, Mehta, Ramana e Bhaskar 2022), desenvolvida por Yann LeCun, é notável por sua aplicação bem-sucedida no reconhecimento de dígitos manuscritos. A AlexNet(Alom et al. 2018), concebida por Alex Krizhevsky, Ilya Sutskever e Geoffrey Hinton, foi um marco ao vencer o Desafio ImageNet em 2012, impulsionando o interesse em deep learning. A VGGNet(L. Wang, Guo, W. Huang e Qiao 2015), criada pelo

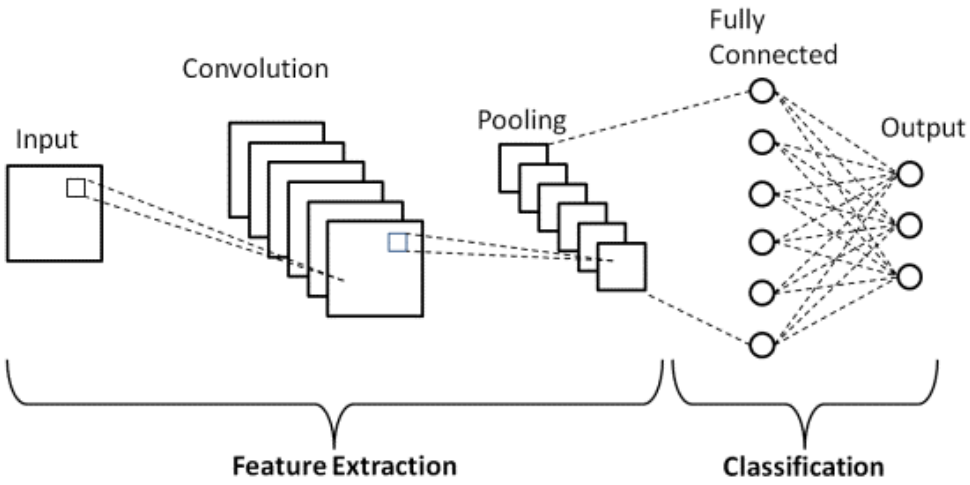

FIGURA 2.1: Arquitetura simples de uma CNN.

Visual Geometry Group (VGG) da Universidade de Oxford, destaca-se pela simplicidade e profundidade de suas camadas convolucionais. A GoogleNet(Al-Qizwini, Barjasteh, Al-Qassab e Radha 2017), introduzida pelo Google, tornou-se conhecida pela arquitetura Inception, que utiliza módulos de convolução paralela para melhorar a eficiência computacional. A ResNet(Szegedy, Ioffe, Vanhoucke e Alemi 2017), desenvolvida pela Microsoft Research, inovou ao empregar conexões residuais, possibilitando a construção de redes mais profundas e facilitando o treino. A MobileNet(Sinha e El-Sharkawy 2019), projetada para dispositivos móveis e sistemas com recursos limitados, utiliza operações de convolução separáveis em profundidade para eficiência computacional. Por fim, a EfficientNet(Tan e Le 2019), também do Google, utiliza técnicas de escalonação composto para optimizar o desempenho e a eficiência de parâmetros.

"As CNNs não precisam de extração manual de características, imitam neurónios biológicos, usam *kernels* para detetar diferentes características e as funções de ativação simulam a transmissão de sinais neuronais(Dhillon e Verma 2020). Na construção de uma CNN, são importantes quatro componentes: convolução para extração de características, preenchimento para compensar perda de informações nas bordas, passo (stride) para controlar a densidade da convolução e *pooling* para reduzir redundâncias. Após a extração de características, as *fully connected layers* são frequentemente utilizadas para realizar a classificação final ou outras tarefas específicas (Dhillon e Verma 2020).

2.2.2 ViT (Vision Transformer)

Nesta secção abordaremos as ViT (*Vision Transformer*) que têm vindo a demonstrar resultados promissores no campo da visão computacional. Ao contrário das CNNs tradicionais, que operam com convoluções espaciais, as ViT tratam imagens como sequências de *patches* e as processam usando blocos de atenção (Dosovitskiy et al. 2020).

É proposto por Dosovitskiy et al. 2020 no seu trabalho em "*An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale*"(Dosovitskiy et al. 2020) uma arquitetura que não necessita de operações de convolução e recorrência anteriormente propostas como o caso das CNNs. Baseado apenas em mecanismos de atenção com a proposta de trazer mais qualidade com menos tempo de treino, no caso, testado em tarefas de tradução. Com base nesta proposta outras aplicações foram testadas como tarefas de reconhecimento(Hatamizadeh et al. 2022; Hong, Han, J. Yao, L. Gao, B. Zhang, Plaza e Chanussot 2021; Xizhou Zhu, Su, L. Lu, B. Li, Xiaogang Wang e Dai 2020), tarefas multimodais(Xu, Xiatian Zhu e Clifton 2023) e tarefas de processamento de vídeo(S. Shi, J. Gu, Xie, Xintao Wang, Yujiu Yang e C. Dong 2022).

Em particular, na área da deteção e seguimento de objetos, é habitualmente utilizado o DETR (*Detection Transformer*) (Carion, Massa, Synnaeve, Usunier, Kirillov e Zagoruyko 2020) e vários sistemas de algoritmo de seguimento foram também desenvolvidos (Chu, Jiang Wang, You, Ling e Zicheng Liu 2021; Meinhardt, Kirillov, Leal-Taixe e Feichtenhofer 2022; P. Sun, J. Cao, Jiang, R. Zhang et al. 2021; F. Zeng, B. Dong, T. Wang, Xiangyu Zhang e Wei 2021). .

2.3 Seguimento

Nesta secção abordaremos vários aspetos acerca do seguimento de objetos. O seguimento de objetos em vídeos envolve acompanhar a posição, o movimento e outras características dos objetos ao longo do tempo. Isso é feito analisando uma sequência de *frames* consecutivos e determinando como os objetos se movem de um *frame* para o próximo (W. Luo, J. Xing, Milan, Xiaoqin Zhang, Wei Liu e T.-K. Kim 2021). As técnicas de seguimento de objetos muitas vezes se baseiam em informações extraídas de cada *frame* individual para inferir a trajetória dos objetos ao longo do tempo.

2.3.1 Deteção em Seguimento de Objetos

Existem duas abordagens principais na maioria dos trabalhos de seguimento de múltiplos objetos, dependendo de como os objetos são inicializados: Seguimento baseado em Deteção (*Detection Based Tracking* - DBT) e Seguimento livre de Deteção (*Detection Free Tracking* - DFT)(W. Luo, J. Xing, Milan, Xiaoqin Zhang, Wei Liu e T.-K. Kim 2021). A principal diferença entre os dois reside na inicialização, ao passo que um é automático (DBT) o outro é manual. DBT é a abordagem mais utilizada porque os objetos podem ser descobertos e objetos desaparecidos são terminados automaticamente, o que no caso do DFT(Fragkiadaki e J. Shi 2011; M. Yang, T. Yu e Y. Wu 2007; L. Zhang e Maaten 2014), este não consegue lidar com novas aparições.

Falando da abordagem mais usada, a deteção de objetos desempenha um papel fundamental no seguimento de objetos em cenários complexos e dinâmicos. Ao identificar e localizar objetos em imagens ou vídeos, a deteção de objetos fornece a base necessária para acompanhar esses objetos ao longo do tempo e do espaço.

Sem uma deteção precisa de objetos, o seguimento de múltiplos objetos seria ineficaz, pois não haveria informações sobre quais objetos estão presentes em cada *frame* da sequência de vídeo. Além disso, a deteção de objetos influencia diretamente a qualidade e a confiabilidade das trajetórias de seguimento resultantes. Deteções imprecisas ou ausentes podem levar a erros de associação, o que resulta em trajetórias erráticas ou incompletas. Portanto, a precisão e a robustez da deteção de objetos são cruciais para garantir um seguimento preciso e consistente de objetos em diferentes aplicações.

Como demonstra a figura 2.2 primeiramente, um detetor independente é aplicado a todos os *frames* de imagem para obter deteções prováveis de pedestres. Em seguida, um *tracker* é executado no conjunto de deteções para realizar a associação de dados, ou seja, vincular as deteções para obter trajetórias completas.

A deteção de objetos para o seguimento de múltiplos objetos é frequentemente realizada por modelos como DPM(Felzenszwalb, Mcallester e Ramanan 2008), Faster R-CNN(S. Ren, K. He, R. B. Girshick e J. Sun 2015), SDP(F. Yang, Choi e Y. Lin 2016), RetinaNet(T.-Y. Lin, Goyal, R. Girshick, K. He e Dollár 2017), CenterNet(X. Zhou, Dequan Wang e Krähenbühl 2019) e YOLO(Redmon e Farhadi 2018). Métodos de MOT utilizam esses resultados de deteção para melhorar o desempenho do seguimento.

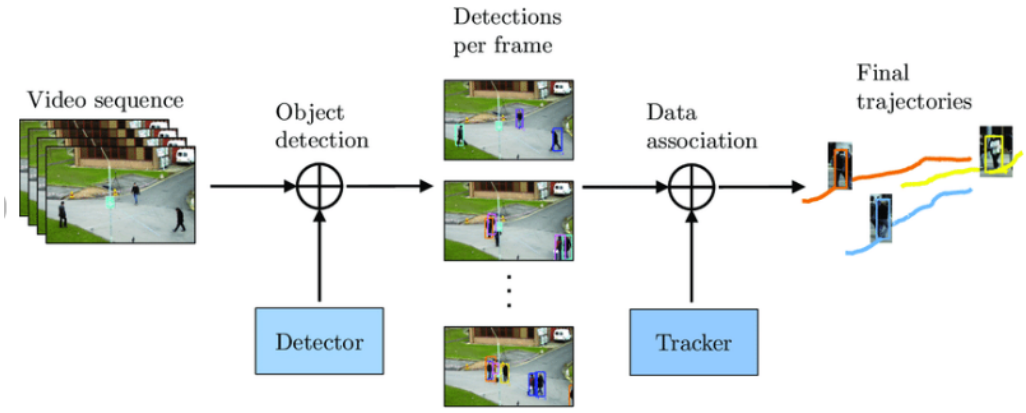

FIGURA 2.2: Abordagem Seguimento por Deteção. Retirado de Leal-Taixé 2014.

Abordagens baseadas em *Transformer* no contexto do seguimento de múltiplos objetos, têm tido avanços recentes para a previsão de trajetórias. Especificamente, estudos como C. Yu, X. Ma, J. Ren, Haiyu Zhao e Yi 2020 utilizaram modelos *Transformer* para prever trajetórias de objetos com resultados impressionantes. Além disso, investigações pioneiras conduzidas em Meinhardt, Kirillov, Leal-Taixe e Feichtenhofer 2022; P. Sun, J. Cao, Jiang, R. Zhang et al. 2021 exploraram a integração de arquiteturas *Transformer* em estruturas de MOT.

2.3.2 *Single-Tracking vs Multi-Tracking*

Nesta secção serão abordadas as principais diferenças entre *Single-Tracking* e *Multi-Tracking* e os principais desafios que representa o seguimento de objetos. E portanto, como os próprios nomes indicam, a principal diferença entre ambos é que em SOT o objetivo é especificar um determinado alvo de interesse e portanto único a seguir, enquanto que em MOT é efetuado o seguimento de múltiplos alvos de interesse. A escolha de cada abordagem vai depender das intenções do utilizador, se quer só um alvo ou vários.

Como é , diferentes abordagens para o mesmo problema implicam diferenças quer de implementação quer de optimização:

- **Objetivo:** No caso de SOT o objetivo é efetuar seguimento de um alvo e em MOT de vários;
- **Complexidade:** MOT será sempre mais complexo, porque terá de lidar com sobreposições e contagem de objetos enquanto que SOT se preocupa exclusivamente com um objeto;

- **Inicialização:** Normalmente SOT é iniciado manualmente (Bolme, Beveridge, Draper e Lui 2010; Kalal, Mikolajczyk e Matas 2012) e MOT de forma automática através de deteção;
- **Associação:** No caso de MOT a associação de objetos é um dos passos importantes e várias técnicas são utilizadas como é o exemplo dos filtros de Kalman (Welch, Bishop et al. 1995a), enquanto que em SOT a *bounding box* é associada a um único objeto;
- **Robustez:** Pelos fatores descritos acima, SOT em princípio terá sempre algoritmos mais robustos por não ter que lidar com problemas como sobreposição, erros de associação e novas aparições;
- **Aplicações:** SOT é comummente usado em aplicações como seguimento de veículos, monitorização de tráfego, reconhecimento facial e seguimento de objetos em vídeos de vigilância, enquanto que em MOT se utiliza em análise de multidões(Loy, K. Chen, Gong e Xiang 2013), contagem de pessoas(Ye, J. Shen, G. Lin, Xiang, Shao e Hoi 2021), seguimento de múltiplos veículos em estradas(Betke, Haritaoglu e Davis 2000) e monitorização de comportamento de grupo(Rezaei e Yazdi 2021).

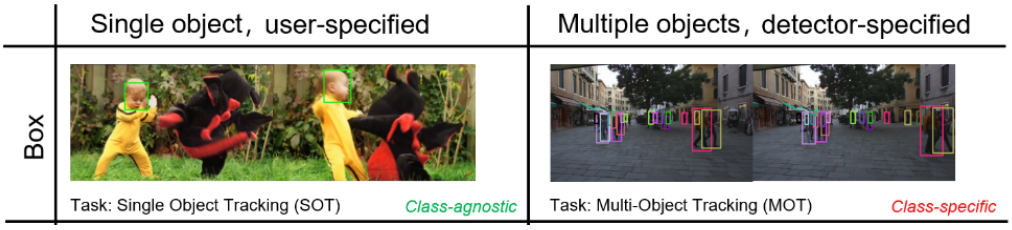

FIGURA 2.3: Diferença entre SOT e MOT. Retirado de Zhongdao Wang, Hengshuang Zhao, Y.-L. Li, S. Wang, Torr e Bertinetto 2021b.

2.3.3 Inicialização do alvo a seguir

Uma das formas de categorizar um algoritmo de seguimento é em relação ao método de inicialização. Existem duas opções possíveis, seguimento baseado em deteção e seguimento sem deteção(W. Luo, J. Xing, Milan, Xiaoqin Zhang, Wei Liu e T.-K. Kim 2021).

Falando primeiro do mais conhecido e usado, seguimento baseado em deteção, como o próprio nome indica o desempenho do algoritmo neste caso dependerá diretamente da deteção de objetos. Isto porque, após ser inserido uma sequência de *frames* (vídeo) a primeira ação do algoritmo é detetar e só depois conectar as

deteções a trajetórias. Daí retira-se logo algumas especificidades, o algoritmo para funcionar terá de ser previamente treinado para um dado número de classes de objetos específicos para que possa efetuar a deteção de forma automática. Ou seja apesar de conseguir á partida lidar com vários objetos aparecendo e saindo dos *frames* as classes sujeitas a análise são limitadas (W. Luo, J. Xing, Milan, Xiaoqin Zhang, Wei Liu e T.-K. Kim 2021).

Por sua vez o seguimento sem detetor já não necessita de treino prévio. No entanto é obrigatório para o seu funcionamento que os objetos que o utilizador deseja manter no algoritmo sejam introduzidos manualmente no primeiro *frame* (W. Luo, J. Xing, Milan, Xiaoqin Zhang, Wei Liu e T.-K. Kim 2021). Isto implica que os objetos apesar de não limitados ao número de classes, são limitados ao números de objetos a ser seguidos pelo algoritmo, pelo motivo que não existe um detetor para lidar com aparições de novos objetos em *frame*s seguintes.

2.3.4 Desafios do Seguimento

Comparado com o seguimento de um único objeto (SOT), que se concentra principalmente em criar modelos sofisticados para lidar com desafios como mudanças de escala e iluminação, o seguimento de múltiplos objetos apresenta desafios adicionais. Além de determinar o número de objetos em movimento e manter suas identidades ao longo do tempo, há problemas específicos, como oclusões frequentes, iniciar e encerrar seguimentos, objetos com aparência semelhante e interações entre eles.

Para enfrentar esses desafios, várias soluções foram propostas ao longo dos anos, abordando diferentes aspetos do problema de seguimento de múltiplos objetos. Essa diversidade de soluções pode se tornar difícil para os pesquisadores, especialmente os novatos, entenderem completamente o problema e escolherem a abordagem mais adequada segundo W. Luo, J. Xing, Milan, Xiaoqin Zhang, Wei Liu e T.-K. Kim 2021:

- **Iluminação:** A disposição das fontes de luz pode afetar o tipo de objeto apresentado, e múltiplas fontes de luz podem criar efeitos de iluminação distintos.O problema da iluminação é definido como o grau de visibilidade ou mudança de aparência de um objeto sob diferentes condições de iluminação (Ranipa e Bhatt 2014).;
- **Oclusão:** As estratégias para combater a oclusão são predominantemente baseadas em métodos de região e são difíceis de serem aplicadas aos modelos

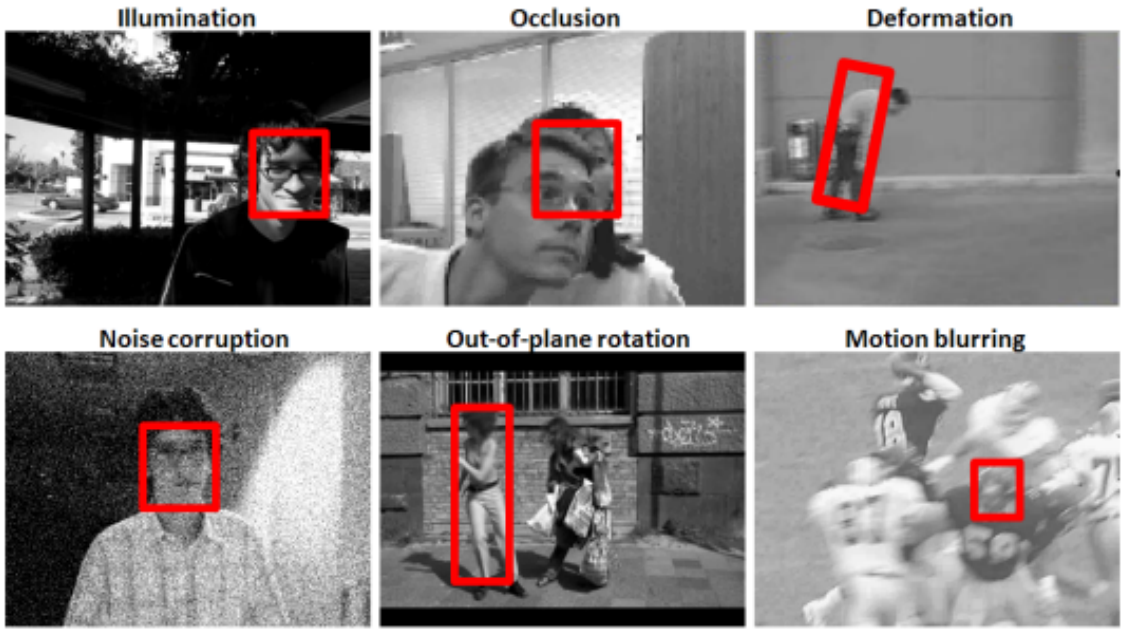

FIGURA 2.4: Ilustração de mudanças de aparência difíceis de lidar em MOT. Retirado de Xi Li, W. Hu, C. Shen, Z. Zhang, Dick e Hengel 2013.

baseados em pontos. Em contraste, os modelos de duas etapas são todos baseados em regiões, oferecendo condições adequadas para várias contramedidas eficazes contra a oclusão(Y.-H. Wang, Hsieh, P.-Y. Chen, Chang, So e Xin Li 2024; Ranyang Zhao, Xinyan Zhang e J. Zhang 2024).

- **Deformação/Alterações abruptas de aparência:** Representa ainda um grande desafio para seguimento de objetos especialmente por normalmente o objeto de estudo tratar-se de classes "deformáveis"como pedestres. Apesar de algumas tentativas em desenvolver modelos de aparência mais robustos o treino em condições que cubram todas as possibilidades existentes parece ser o caminho que se tem seguido por enquanto como *datasets* de larga escala.
- ***Motion Blur*:** Na maior parte dos algoritmos não está previsto a presença de *motion blur* devido ás sequências de vídeo serem retiradas a partir de câmaras estáticas. No entanto existem alguns trabalhos que prevêm essa dificuldade(B. Ma, L. Huang, J. Shen, Shao, M.-H. Yang e Porikli 2016).
- **Ruído:** Fenómenos físicos, movimentos da câmara e condições ambientais imprevisíveis, como a presença de poeira, podem causar ruído nas imagens de vídeo (C.-H. H. Yang, Chhabra, Y.-C. Liu, Kong, Yoshinaga e Murakami 2021).

- **Alterações de ID:** Erros na aquisição de informação, no processamento e na previsão podem levar a dados perdidos ou incorretos durante a associação de objetos resultando em alterações de identidade em MOT(J. Huang, X. He e S. Zhao 2023). Modelos de Re-ID(Re-identificação) são implementados e desenvolvidos em quase todos os trabalhos ultimamente para tentar recuperar objetos perdidos por vários motivos como oclusões(Aharon, Orfaig e Bobrovsky 2022; Zelin Liu, Xinggang Wang, Cheng Wang, Wenyu Liu e X. Bai 2023; Meinhardt, Kirillov, Leal-Taixe e Feichtenhofer 2022; Y. Zhang, P. Sun et al. 2021; Y. Zhang, Chunyu Wang, Xinggang Wang, W. Zeng e Wenyu Liu 2020).
- **Deteções Imprecisas:** Os detetores fornecem aos algoritmos de seguimento baseados em deteção observações de potenciais localizações de objetos em cada *frame*, enquanto que os *trackers* associam estas deteções ao longo dos *frames* para gerar trajetórias.(W. Luo, J. Xing, Milan, Xiaoqin Zhang, Wei Liu e T.-K. Kim 2021). Daqui se retira, que o desempenho do algoritmo de MOT, depende diretamente da qualidade e rapidez do detetor. Os erros que ocorrem numa primeira fase acumulam para a fase de associação.
- **Arquiteturas Leves:** Apesar da maior parte das soluções depender de arquiteturas pesadas, isso se torna contraintuitivo quase se fala de seguimento em tempo real. Por isso, a relação rapidez/eficiência se torna um desafio quando implementamos algoritmos mais "leves". Requerem um bom *fine tune* e os melhores pesos possíveis para um boa inicialização (Yan, H. Peng, K. Wu, Dong Wang, Fu e H. Lu 2021).

2.3.5 Seguimento *Offline* e *online*

O seguimento de objetos pode ser categorizado em seguimento *online* e seguimento *offline*, dependendo se as observações dos *frames* futuros são utilizadas ao lidar com o *frame* atual. No seguimento *online*, os métodos dependem apenas das informações passadas disponíveis até o *frame* atual. Por outro lado, abordagens de seguimento *offline* empregam observações tanto do passado quanto do futuro(W. Luo, J. Xing, Milan, Xiaoqin Zhang, Wei Liu e T.-K. Kim 2021).

O caminho que a comunidade tem seguido é de cada vez mais desenvolver algoritmos de um estágio para aplicações em tempo real e são exemplo de seguimento online por exemplo os seguintes algoritmos: ByteTrack(Y. Zhang, P. Sun et al. 2021), SmileTrack(Y.-H. Wang, Hsieh, P.-Y. Chen, Chang, So e Xin Li 2024), SparseTrack(Zelin Liu, Xinggang Wang, Cheng Wang, Wenyu Liu e X. Bai 2023), SORT(Bewley, ZongYuan Ge, Ott, Ramos e Upcroft 2016), TransTrack(P. Sun, J.

Cao, Jiang, R. Zhang et al. 2021), BoT-SORT(Aharon, Orfaig e Bobrovsky 2022), FairMOT(Y. Zhang, Chunyu Wang, Xinggang Wang, W. Zeng e Wenyu Liu 2020)e entre muitos outros.

No entanto também existe investigação na área do seguimento *offline* como o Trackformer(Meinhardt, Kirillov, Leal-Taixe e Feichtenhofer 2022). Devido a limitações computacionais e de memória, nem sempre é possível lidar com todos os *frames* de uma vez. Uma solução alternativa é dividir os dados em clipes de vídeo mais curtos e inferir os resultados de forma hierárquica ou sequencial para cada lote.

2.3.6 Associação de Objetos

No seguimento de múltiplos objetos, a abordagem de seguimento por deteção consiste em duas etapas distintas. Primeiramente, cada *frame* de um vídeo é analisado por um detetor de objetos para identificar as possíveis localizações de todos os objetos presentes. Em seguida, na fase de associação de objetos, as deteções falsas positivas são eliminadas e as deteções corretas são vinculadas às identidades correspondentes, gerando assim trajetórias individuais para cada objeto(Ganian, Hamm e Ordyniak 2021). Este capitulo concentra-se na segunda fase deste processo.

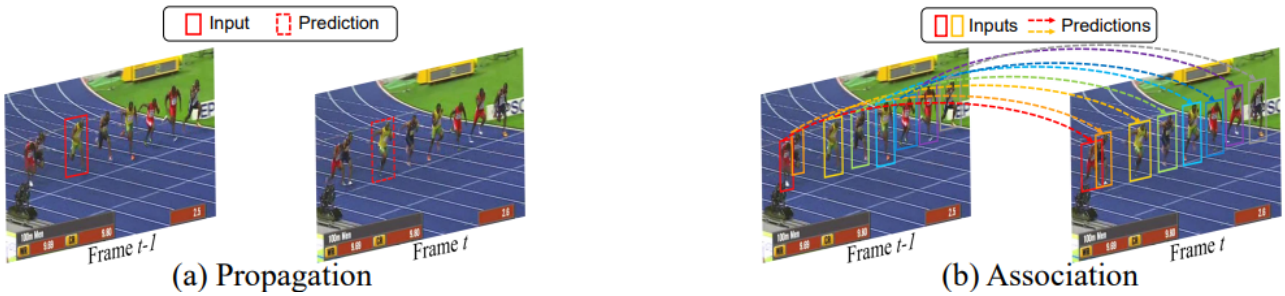

FIGURA 2.5: Propagação vs Associação. Retirado de Zhongdao Wang, Hengshuang Zhao, Y.-L. Li, S. Wang, Torr e Bertinetto 2021a.

Como demonstra a figura 2.5, no problema de associação, são fornecidas observações nos *frames* anterior e atual, e o objetivo é determinar correspondências entre os dois conjuntos A forma mais recorrente de colmatar este problema é utilizando filtros de kalman(Aharon, Orfaig e Bobrovsky 2022; Bewley, ZongYuan Ge, Ott, Ramos e Upcroft 2016; Welch, Bishop et al. 1995b; Wojke, Bewley e Paulus 2017; Y. Zhang, P. Sun et al. 2021; Y. Zhang, Chunyu Wang, Xinggang Wang, W. Zeng e Wenyu Liu 2020).

Os filtros de Kalman(Welch, Bishop et al. 1995b) são utilizados em MOT para prever a posição dos objetos no próximo *frame* com base nas regiões encontradas

no *frame* atual. Primeiramente, o centro do objeto é identificado e, em seguida, o filtro de Kalman é empregue para fazer a previsão da posição do objeto no próximo *frame*. O filtro de Kalman é um método baseado em regiões que estima o estado do sistema linear, assumindo uma distribuição gaussiana. Ele consiste em duas etapas principais: previsão e correção. Durante o processo de previsão, o filtro de Kalman fornece uma estimativa ótima da posição do objeto em movimento para cada instante de tempo.

Já em algoritmos mais recentes parece não só haver uma escolha predominante em utilizar filtros de kalman, mas também o uso de algoritmo húngaro(Y. Zhang, Chunyu Wang, Xinggang Wang, W. Zeng e Wenyu Liu 2020), IoU (*Intersection Over Union*)(Bewley, ZongYuan Ge, Ott, Ramos e Upcroft 2016) e ferramentas de Re-ID(Re-Identificação)(Y. Zhang, P. Sun et al. 2021).

Trackformer (Meinhardt, Kirillov, Leal-Taixe e Feichtenhofer 2022)) estende sua eficácia na deteção de objetos para o seguimento de múltiplos objetos ao abordar o problema como uma previsão sequencial entre *frames*. Em vez de tratar cada *frame* individualmente, ele usa um mecanismo de atenção para associar dados entre *frames*, permitindo prever conjuntos de trajetórias ao longo do tempo com base nas deteções de objetos. Similarmente, o TransTrack (P. Sun, J. Cao, Jiang, R. Zhang et al. 2021)) utiliza um mecanismo com base no *Deformable DETR*(Xizhou Zhu, Su, L. Lu, B. Li, Xiaogang Wang e Dai 2020). No entanto, é importante notar que todas essas abordagens baseadas em *Transformers* são computacionalmente intensas, o que as torna inadequadas para aplicações *on-board* que exigem tempo real.

2.3.7 Modelação

Para que os algoritmos interpretem e entendam melhor as informações visuais fornecidas é essencial desenvolver modelos que retiram características dos objetos, com a finalidade de melhorar a precisão e robustez.

Entre estes modelos os mais conhecidos são os de aparência e movimento. No entanto os algoritmos podem adotar outros ou mesmo desenvolver para determinados *benchmarks* como é o caso de modelos padrão de movimento de multidões (Gaidon, Q. Wang, Cabon e Vig 2016), modelos para lidar com oclusão (Possegger, Mauthner, P. M. Roth e Bischof 2014) e modelos de interação entre objetos (Bolshakov 2024). Isto deve-se pelo estado de arte na área de seguimento de múltiplos objetos seguir *datasets* de teste que se focam em multidões de pedestres como é o caso do MOTChallenge (Dendorfer, Osep et al. 2021).

Similaridade de aparência pode ser benéfico para a re-identificação de objetos a longo prazo(Y. Zhang, P. Sun et al. 2021).O DeepSORT (Wojke, Bewley e Paulus 2017) utiliza um modelo de Re-ID próprio para extrair características de aparência das *bounding boxes* de deteção.

Dentro dos modelos de aparência existe também alguns avanços que levam em conta oclusão de objetos. Nomeadamente em Y. Wu, T. Yu e Hua 2003 é proposto a integração do modelo de oclusão e do modelo de múltiplas ângulos de visão que resulta em uma rede bayesiana complexa. Também em Specker, Stadler, Florin e Beyerer 2021 é proposto um sistema de seguimento de múltiplos alvos e múltiplas câmaras (MTMCT - *Multi Target Multi Camera Tracking*) que lida com o desafio da oclusão em vídeos de vigilância. Onde quando um objeto seguido fica oculto, em vez de encerrar imediatamente sua trilha, o sistema verifica se há outra trilha ativa com alta sobreposição. Se isso ocorrer, as duas trilhas são consideradas como um par de oclusão.

No trabalho de Pellegrini, Ess, Schindler e Van Gool 2009 os autores explicam que os modelos dinâmicos convencionais preveem onde cada objeto estará no futuro com base apenas em sua trajetória passada, sem considerar outros objetos na cena. Eles só lidam com colisões quando elas realmente acontecem. Essa abordagem não leva em conta aspetos importantes do comportamento humano: as pessoas planeiam sua rota com antecedência, levando em conta para onde estão indo, o ambiente ao seu redor e antecipando colisões para ajustar sua trajetória e evitá-las. E neste âmbito os pesquisadores então introduziram um modelo de comportamento social dinâmico, inspirado em modelos desenvolvidos para simulação de multidões.

No mesmo contexto de ter em conta todas as trajetórias para prever futuras posições de objetos, no artigo "*Discriminative Appearance Modeling With Multi-Track Pooling for Real-Time Multi-Object Tracking*"(C. Kim, Fuxin, Alotaibi e Rehg 2021) os autores afirmam que muitas abordagens modelam cada alvo de forma isolada e não conseguem usar todos os alvos na cena para atualizar conjuntamente.

No artigo "*Bayesian Multi-object Tracking Using Motion Context from Multiple Objects*"(J. H. Yoon, M.-H. Yang, Lim e K.-J. Yoon 2015) é descrito um problema desafiador de seguimento de múltiplos objetos em tempo real com uma única câmara em movimento, onde os modelos de movimento convencionais em 2D não são aplicáveis devido ao movimento global da câmara. O artigo propõe o uso de contexto de movimento de vários objetos para descrever o movimento relativo entre eles e construir uma Rede de Movimento Relativo (RMN) para eliminar os efeitos do movimento inesperado da câmara.

2.3.8 Re-ID

Re-ID (Re-Identificação), é definido como um problema de recuperação de um objeto detetado anteriormente. Seu objetivo é determinar se o objeto consultado apareceu em outro lugar em um momento distinto capturado por uma câmara diferente, ou mesmo na mesma câmara em um instante de tempo diferente. No entanto, vários desafios complicam essa tarefa, incluindo diferentes pontos de vista, baixa resolução de imagens, mudanças de iluminação, oclusões, entre outros(Ye, J. Shen, G. Lin, Xiang, Shao e Hoi 2021).

Além disso, há vários conjuntos de dados públicos disponíveis para Re-ID, como Market-1501(L. Zheng, L. Shen, L. Tian, S. Wang, Jingdong Wang e Q. Tian 2015), DukeMTMC-reID(Gou, Karanam, Wenqian Liu, Camps e Radke 2017), CUHK03(W. Li, Rui Zhao, Xiao e Xiaogang Wang 2014), entre outros, que podem ser usados para experimentar algoritmos de Re-ID e comparar seu desempenho com outros métodos existentes.

Figura 2.6: Funcionamento básico de um método Re-ID. Retirado de W.-S. Zheng, Gong e Xiang 2012.

Como demonstra a figura 2.6, se a imagem de consulta for correspondida a uma pessoa errada (como mostrado pela linha azul tracejada), então a saída está incorreta. Em uma verificação generalizada de pessoas baseada em conjuntos por meio de *transfer learning*, se a imagem de consulta for correspondida a uma das pessoas-alvo (como mostrado pela linha vermelha), então a saída está correta.

Em suma, embora a pesquisa em Re-ID tenha feito progressos significativos, a aplicação prática dessas técnicas enfrenta desafios únicos que precisam ser superados para alcançar implantações bem-sucedidas em ambientes do mundo real(Leng, Ye e Q. Tian 2019).

2.3.9 Datasets MOT de Teste

O desenvolvimento de *benchmarks* em larga escala é fundamental para a maioria das aplicações relacionadas à inteligência artificial (L. Liu et al. 2023). Esses *benchmarks* fornecem conjuntos de dados compartilhados para o treino de modelos e competições justas na avaliação, promovendo assim o desenvolvimento de algoritmos eficazes de seguimento de múltiplos objetos. É importante ressaltar que o desenvolvimento desses algoritmos também depende fortemente da disponibilidade de conjuntos de dados de alta qualidade(Dave, Khurana, Tokmakov, Schmid e Ramanan 2020; Dendorfer, Rezatofighi et al. 2020; Milan, Leal-Taixe, Reid, S. Roth e Schindler 2016; Wen et al. 2019).

A série de MOTChallenge (Dendorfer, Osep et al. 2021; Dendorfer, Rezatofighi et al. 2020; Milan, Leal-Taixe, Reid, S. Roth e Schindler 2016) é talvez o conjunto de *benchmarks* mais conhecido e utilizado para avaliar desempenho de algoritmos de seguimento de múltiplos objetos. É um conjunto de *datasets* com foco em imagens estáticas retiradas do ar livre em que existe multidões de pessoas a caminhar, onde o objetivo é stressar o algoritmo com oclusão e novas aparições.

FIGURA 2.7: Imagens referentes ao MOT16. Sequência de treino (superior) e teste (inferior). Retirado de Milan, Leal-Taixe, Reid, S. Roth e Schindler 2016.

A série de desafios do VisDrone (P. Zhu, Wen, Du, Bian, Fan, Q. Hu e Ling 2021) também contribui muito para o desenvolvimento e aperfeiçoamento de algoritmos de MOT nomeadamente aqueles que se focam em imagens captadas por aparelhos UAV. Apesar das imagens retiradas por um veículo aéreo impactarem no

desempenho de um algoritmo devido as características inerentes de uma imagem captada por estas fontes, como explicado na secção 2.1, o emprego de UAVs no âmbito da vigilância e controlo por exemplo têm sido cada vez mais frequentes e correspondem ao caminho que a comunidade tende a seguir ao desenvolver cada vez melhores algoritmos para atuarem em tempo real.

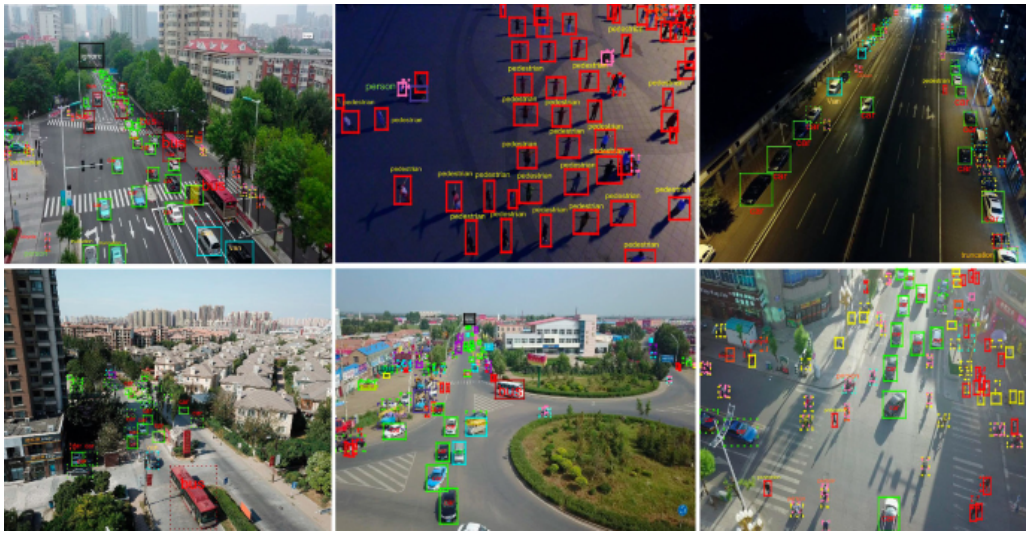

FIGURA 2.8: Exemplos de imagens referentes ao VisDrone. Retirado de P. Zhu, Wen, Du, Bian, Fan, Q. Hu e Ling 2021.

O SportsMOT(Cui, C. Zeng, X. Zhao, Yichun Yang, G. Wu e L. Wang 2023) foca-se em vídeos retirados de uma perspetiva televisiva de 3 desportos: voleibol, futebol e basquetebol. A proposta é através de 240 vídeos com mais de 15 mil *frames* anotados testar algoritmos para que sejam capazes de desenvolver ferramentas de análise desportiva de forma automática.

O DanceTrack(P. Sun, J. Cao, Jiang, Yuan, S. Bai, Kitani e P. Luo 2022) é um *dataset* com imagem estática de concursos de dança desportiva que visa testar dois aspetos de MOT: aparências semelhantes e padrões de movimento pouco habituais. No fundo põe á prova as técnicas empregues de modelos de aparência e movimento existentes.

O WildTrack(Chavdarova et al. 2017) é semelhante ao MOTChallenge porém com menos fontes e com a diferença de este ser multi câmara. O que segundo os pesquisadores que desenvolveram este projeto, visam testar novos algoritmos para imagens de diferentes ângulos que se sobrepõem.

O KITTI(Geiger, Lenz e Urtasun 2012) é um dataset que visa a implementação de sistemas de reconhecimento empregues em aparelhos robóticos como

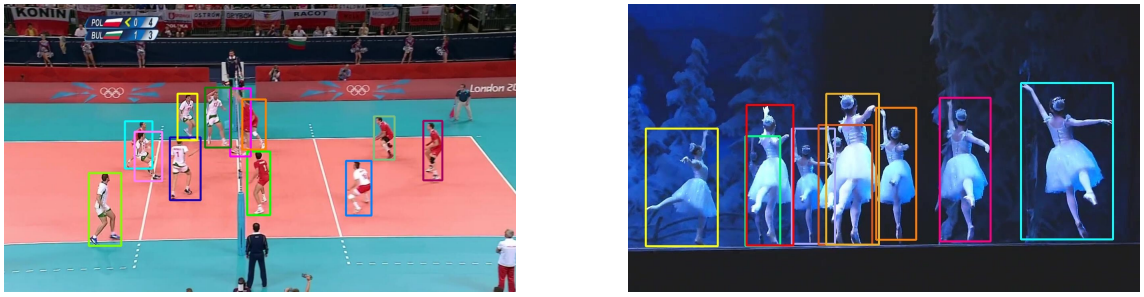

FIGURA 2.9: Á esquerda um exemplo de uma imagem de SportsMOT num jogo de voleibol e á direita uma imagem do DanceTrack.Imagens retiradas de Cui, C. Zeng, X. Zhao, Yichun Yang, G. Wu e L. Wang 2023 e de P. Sun, J. Cao, Jiang, Yuan, S. Bai, Kitani e P. Luo 2022

condução autónoma. Contém mais de 200 mil anotações com até 15 carros e 30 pedestres em cada imagem, de inúmeros ângulos de visão.

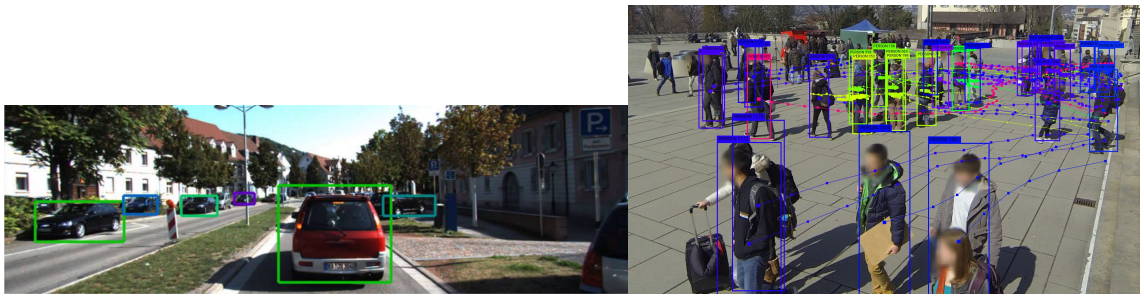

FIGURA 2.10: Á esquerda um exemplo de uma imagem de KITTI e á direita uma imagem do WildTrack.Imagens retiradas de Geiger, Lenz e Urtasun 2012 e de Chavdarova et al. 2017

Para além disso o TAO (Tracking Any Object) é motivado pela existência de um outro *dataset* de deteção muito conhecido, MS COCO (T.-Y. Lin, Maire et al. 2015). Com a proposta de colmatar a pouca diversidade de classes avaliadas em outros *benchmarks* os autores em Dave, Khurana, Tokmakov, Schmid e Ramanan 2020 propõem um *dataset* com mais de 833 classes em 2907 vídeos.

2.3.10 Métricas de Avaliação

As métricas de avaliação são utilizadas para retirar *feedbacks* do desempenho dos algoritmos e assim conseguirmos aperfeiçoar ou corrigir os mesmos. As métricas de avaliação de seguimento de múltiplos alvos que serão estudadas nesta secção fazem parte de um conjunto de métricas utilizadas pela maior plataforma de teste de algoritmos de seguimento MOTChallenge (Dendorfer, Osep et al. 2021). Esta plataforma utiliza as métricas CLEAR-MOT propostas em 2006 (Stiefelhagen, Bernardin, Bowers, Garofolo, Mostefa e Soundararajan 2006) , adiciona métricas

Dataset	Cenário	Nºimagens	Classes	Anotações	Resolução	Ano
MOT17	Ar livre	11k	1	900k	1920x1080	2017
MOT20	Ar livre	13k	1	2100k	1920x1080	2020
VisDrone	UAV	33k	11	X	3840X2160	2019
TAO	Ar livre	88k	833	X	640x480	2020
DanceTrack	Desporto	105k	1	X	X	2021
SportsMOT	Desporto	150k	1	1600k	1280x720	2023
WildTrack	Ar livre	2,8k	1	40k	1920x1080	2018
KITTI	Ar livre	41k	2	200k	1241x376	2012

Tabela 2.1: Datasets Existentes.

relacionadas á qualidade do seguimento (B. Wu e Nevatia 2006) e acrescenta IDF1 que avalia a trajetória(Ristani, Solera, Zou, Cucchiara e Tomasi 2016). Para além de ser um conjunto de métricas completas estas são utilizadas pelo grupo de algoritmos escolhido para o desenvolvimento desta dissertação, por isso também mais um motivo para serem abordadas.

Segundo a plataforma MOTChallenge existem 4 tipos de métricas consoante as características que são avaliadas:

- MOTA - *Multiple Object Tracking Accuracy*
- MOTP - *Multiple Object Tracking Precision*
- *Precision, Recall* e F1
- Métricas de Qualidade

Começando então por MOTA descrito na formula 2.1 abaixo onde t é o número de *frames*, GT o número de objetos *ground-truth*, FN os falsos negativos (ou seja ou número de *ground-truth* não detetados pelo método), FP os falsos positivos (ou seja o número de objetos detetados que não existem nos *ground-truth*) e $IDSW$ o número de vezes que a identidade (ID) se altera como por exemplo as vezes que uma trajetória é associada a diferentes GT. Os valores de MOTA variam entre (-infinito, 100) podendo assim ter resultados negativos se o numero de erros for superior ao numero de todos os objetos.

$$MOTA = 1 - \frac{\sum_t (FN_t + FP_t + IDSW_t)}{\sum_t GT_t} \tag{2.1}$$

Como se pode absorver desta expressão, esta medida tem em conta três tipos de erros: falsos negativos, falsos positivos e mudanças de ID.

Apesar Patrick Dendofer em "*MOTChallenge: A Benchmark for Single-Camera Multiple Target Tracking*"(Dendorfer, Osep et al. 2021) reconhecer que esta medida é por vezes criticada por várias fontes da literatura por não conter diferentes tipos de erros devidamente balanceados, ainda assim afirma e com fundamento que é a métrica com mais expressividade no campo de avaliação de desempenho de um algoritmo de seguimento múltiplo.

Abordando agora MOTP descrito na equação 2.2 logo abaixo

$$MOTP = \frac{\sum_t d_{t,i}}{\sum_t C_t} \tag{2.2}$$

Esta métrica mede a proximidade (sobreposição) entre as previsões de localização dos objetos seguidos e suas posições reais ao longo do tempo. A MOTP é calculada pela média das distâncias entre os objetos seguidos e os alvos reais em todos os *frames* de uma sequência. Quanto menor a MOTP, maior a precisão do seguimento.

Abordando agora as métricas de identificação elas estão divididas em dois grupos, *precision* e *recall* (IDP e IDR respetivamente). Onde o prefixo *ID* relaciona-se com a identificação do objeto (ou seja classe e número) e *IDTP, IDFP* e *IDFN*, verdadeiros positivos, falsos positivos e falsos negativos respetivamente.

E por ultimo *IDF1* basicamente junta as duas equações anteriores numa só como demonstra a equação 2.5.

$$IDP = \frac{IDTP}{IDTP + IDFP} \tag{2.3}$$

$$IDR = \frac{IDTP}{IDTP + IDFN} \tag{2.4}$$

$$IDF1 = \frac{2 * IDTP}{2 * IDTP + IDFP + IDFN} \tag{2.5}$$

Ristani, em *"Performance Measures and a Data Set for Multi-target, Multi-camera Tracking"* (Ristani, Solera, Zou, Cucchiara e Tomasi 2016) propõe uma nova medida de avaliação de desempenho onde afirma que o desenvolvimento desta medida teve por motivação complementar as métricas já existentes CLEAR. Argumentando que seria mais vantajoso para alguns desenvolvedores saber o quão bem o algoritmo

determina quem e onde os alvos são detetados durante todo o tempo ao invés de olhar somente aos erros cometidos.

Este conjunto de medidas têm por objetivo avaliar a capacidade de um algoritmo em preservar IDs ao longo do tempo (Ristani, Solera, Zou, Cucchiara e Tomasi 2016).

No contexto do seguimento de múltiplos objetos (MOT), as métricas de precisão (*precision*), *recall* e F1 são amplamente utilizadas para avaliar o desempenho dos algoritmos de seguimento.

Precisão (*Precision*): A precisão mede a proporção de deteções corretas em relação ao total de deteções feitas pelo algoritmo. Em outras palavras, indica a precisão com que o algoritmo identifica corretamente os objetos de interesse, sem fazer muitos erros de falsos positivos.

Revocação (*Recall*): A revocação mede a proporção de deteções corretas em relação ao número total de objetos reais presentes na cena. Ela indica a capacidade do algoritmo de identificar todos os objetos relevantes, minimizando os falsos negativos.

F1 Score: O F1 Score é uma métrica que combina precisão e *recall* em uma única medida, fornecendo uma avaliação geral do desempenho do algoritmo de seguimento. É calculado como a média harmónica entre precisão e *recall* e é especialmente útil quando se deseja equilibrar a importância de ambas as métricas.

2.4 Trabalhos Relacionados

No artigo "*Military object detection in defense using multi-level capsule networks*"(Janakiramaiah, Kalyani, Karuna, Prasad e Krishna 2023) é proposta uma arquitetura de rede neuronal baseada em cápsulas de múltiplos níveis para deteção automática de objetos militares em imagens. A metodologia proposta envolve várias etapas, começando com a extração de características usando camadas convolucionais, seguida por camadas de cápsulas primárias e cápsulas de classe. O estudo é validado usando um conjunto de dados contendo imagens de cinco diferentes objetos militares e objetos civis semelhantes. Experimentos comparativos são realizados com métodos tradicionais de classificação, como máquinas de vetores de suporte (SVM) e redes neuronais convolucionais com transferência de aprendizado (CNN-TL). Os resultados mostram que a arquitetura proposta alcança uma precisão média de 95por

cento para a deteção de objetos militares, superando os métodos tradicionais em termos de precisão e eficácia na classificação de objetos.

Em "*A Military Object Detection Model of UAV Reconnaissance Image and Feature Visualization*"(H. Liu, Y. Yu, Shengzong Liu e W. Wang 2022) é discutido métodos para deteção de objetos militares em imagens de reconhecimento de veículos aéreos não tripulados (UAVs). O artigo destaca os desafios enfrentados nessa tarefa, como a falta de dados de imagem, imagens de baixa qualidade e objetos pequenos. Para lidar com esses desafios, os autores propõem uma melhoria no modelo YOLOv5, criando o UAVT-YOLOv5, e apresentam uma base de dados chamada UAVT-3 para treino e avaliação do modelo.

No trabalho "*A Benchmark and Simulator for UAV Tracking*"(Mueller, Smith e Ghanem 2016b) é proposto um novo conjunto de dados e uma avaliação de referência para o seguimento de alvos por UAVs em baixas altitudes, juntamente com um simulador de UAV foto-realista. Ele destaca a importância crescente do seguimento aéreo em diversas aplicações e a falta de conjuntos de dados abrangentes e *benchmarks* para avaliar algoritmos de seguimento nessas condições. O conjunto de dados, chamado UAV123, contém 123 sequências de vídeo anotadas e é projetado para representar uma variedade de desafios de seguimento aéreo. O simulador permite testar algoritmos de seguimento em cenários sintéticos antes da implantação em UAVs reais. O artigo contribui para a pesquisa nessa área compilando o UAV123, realizando uma extensa avaliação de *trackers* de última geração e desenvolvendo o simulador de seguimento de UAV.

FIGURA 2.11: Exemplos de imagens referentes ao simulador utilizado para testar o uav123. Retirado de Mueller, Smith e Ghanem 2016b.

Para além do UAV123 também foi proposto em "*The Unmanned Aerial Vehicle Benchmark: Object Detection and Tracking*"(Du et al. 2018a) um dataset de teste chamado UAVDT. Este conjunto de dados trata da criação de um *benchmark* para UAVs (Veículos Aéreos Não Tripulados) que visa representar cenários complexos e desafiadores. O objetivo é impulsionar pesquisas relacionadas ao uso de UAVs em visão computacional, fornecendo um conjunto de dados que aborda uma variedade de condições e desafios encontrados em ambientes reais. O conjunto de dados é composto por aproximadamente 80.000 *frames* totalmente anotados com *bounding boxes*, além de até 14 tipos diferentes de atributos, como condição meteorológica, altitude de voo, vista da câmara, categoria de veículo e oclusão. Ele é projetado para apoiar duas tarefas fundamentais de visão computacional: deteção de objetos, e seguimento de objetos. Os resultados experimentais mostram que os métodos mais recentes têm desempenho relativamente pior neste conjunto de dados devido aos novos desafios apresentados em cenas reais de UAV, como alta densidade, objetos pequenos e movimento da câmara.

Por ultimo no trabalho "*Automatic Detection of Civilian and Military Personnel in Reconnaissance Missions using a UAV*"(Santos, Rodrigues, A. B. Pinto e Damas 2023) desenvolvido por Batista Pinto treinou um modelo (YOLOv3) para detetar duas classes, militares e civis com um dataset anotado e adquirido pelo próprio em exercícios de treino das Forças de Fuzileiros portugueses empenhados na Lituânia assim como de exercícios da Escola Naval.

Figura 2.12: Exemplos de imagens referentes ao dataset utilizado por Pinto. Retirado de Santos, Rodrigues, A. B. Pinto e Damas 2023.

Capítulo 3

Metodologia

3.1 Arquitetura do Modelo: ByteTrack

A arquitetura do modelo de seguimento múltiplo de objetos é crucial para garantir um desempenho preciso e eficiente. Nesta secção, descrevemos a estrutura da rede neuronal utilizada, incluindo camadas convolucionais, camadas de agrupamento (*pooling*), camadas de seguimento e qualquer outra arquitetura específica empregue no modelo.

Esta metodologia propõe uma estratégia simples, porém eficaz, que primeiro realiza a deteção de objetos em um vídeo utilizando um detetor previamente treinado, como o YOLOX (Zheng Ge, Songtao Liu, F. Wang, Zeming Li e J. Sun 2021), e em seguida, executa as associações entre as deteções para formar seguimentos contínuos (trajetórias) ao longo do tempo.

Inicialmente, o ByteTrack(Y. Zhang, P. Sun et al. 2021) processa cada *frame* do vídeo de entrada utilizando o detetor YOLOX, gera assim as deteções de objetos juntamente com suas respetivas pontuações de confiança. Em seguida, as deteções são separadas em duas categorias: caixas de deteção (*bounding boxes*) de alta pontuação (Dhigh) e caixas de deteção de baixa pontuação (Dlow), com base em um limiar de pontuação (*threshold*) de deteção definido.

Após essa etapa inicial de deteção, o ByteTrack adota um filtro de Kalman(Welch, Bishop et al. 1995b) para prever as novas localizações dos objetos em cada *frame*, com base nos seguimentos existentes. A associação dos objetos ocorre em dois estágios distintos. No primeiro estágio, as caixas de deteção de alta pontuação são associadas aos seguimentos existentes, utilizando métricas de similaridade como a sobreposição de caixas delimitadoras (IoU) ou distâncias de características de Re-ID. O algoritmo de associação húngara é então aplicado para realizar a correspondência entre as deteções e os seguimentos.

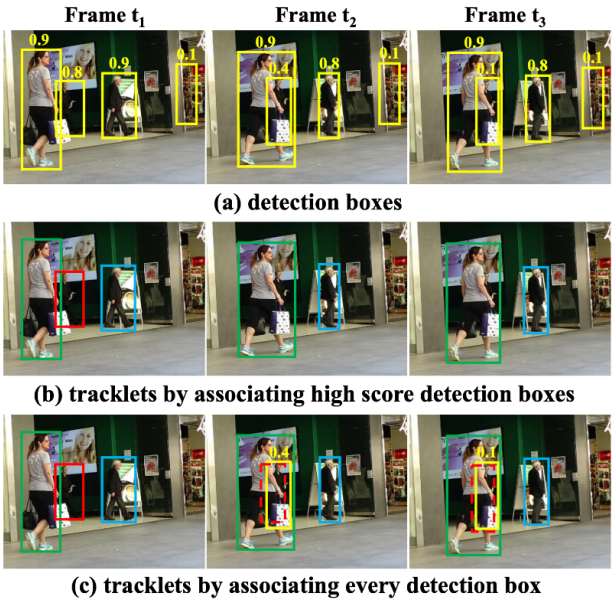

FIGURA 3.1: ByteTrack também aproveita as caixas de deteção de baixa pontuação como mostra a última linha de *frames*. Retirado de LE 2024.

Após a primeira associação, as deteções não associadas e as caixas de deteção de baixa pontuação são analisadas em um segundo estágio de associação. Nesse estágio, a similaridade entre as deteções e os seguimentos remanescentes é calculada com base apenas na sobreposição de caixas delimitadoras (IoU), devido à confiabilidade reduzida das características de aparência nas deteções de baixa pontuação.

$$\text{IOU} = \frac{\text{area of overlap}}{\text{area of union}} =$$

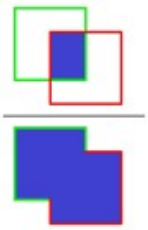

FIGURA 3.2: Imagem referente ao IoU assim como a forma que é calculado. Retirado de LE 2024.

O ByteTrack demonstrou ser altamente flexível e compatível com diferentes métodos de associação. Quando combinado com outros *trackers*, como o FairMOT(Y. Zhang, Chunyu Wang, Xinggang Wang, W. Zeng e Wenyu Liu 2020), pode ser facilmente adaptado para incorporar características específicas de cada método. Essa abordagem robusta e eficaz, aliada ao poderoso detetor YOLOX, estabelece o ByteTrack como uma referência no seguimento de objetos em tempo real.

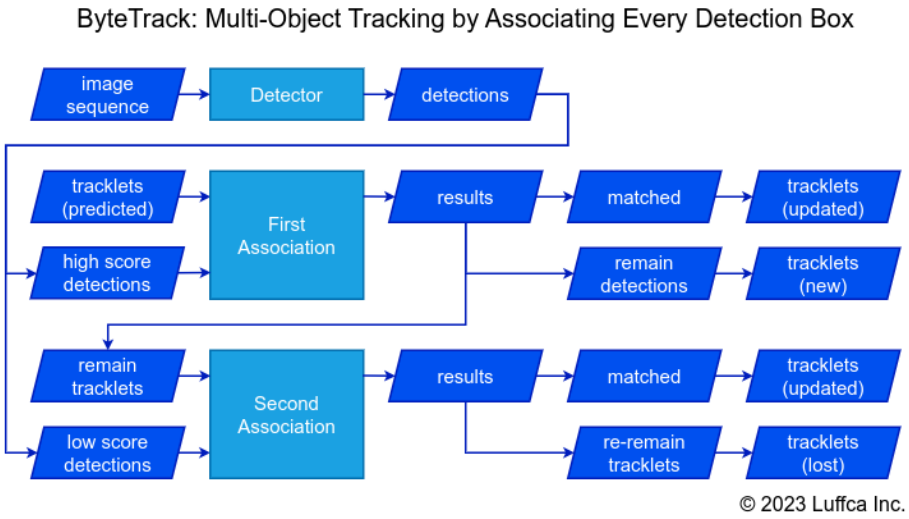

FIGURA 3.3: Arquitetura completa de ByteTrack. Retirado de LE 2024.

3.2 Teste sem treino

Testar o modelo em um conjunto de dados completamente diferente daquele usado no treino permite avaliar sua capacidade de generalização. Se o modelo tiver um desempenho decente no novo conjunto de dados, isso sugere que ele aprendeu padrões úteis que podem ser aplicados em diferentes contextos. E portanto, um refinamento numa base de dados dedicada pode trazer um aumento significativo de desempenho.

3.2.1 Conjunto de dados de teste

O dataset de teste utilizado neste estudo é composto por sete sequências, conforme apresentado na Tabela 3.1. Cada sequência representa uma série de imagens capturadas em diferentes contextos e condições, abrangendo cenários tanto da RCA (República Centro-Africana) quanto da Ucrânia.

Conforme ilustrado na figura 3.4, pode-se observar que as sequências provenientes da RCA consistem em dois tipos de classes, pessoas e veículos, e são compostas por um número variado de anotações e imagens. Essas sequências foram gentilmente cedidas pela Força Aérea Portuguesa, contribuindo para a diversidade da base de dados utilizada neste estudo.

Por outro lado, as sequências da Ucrânia apresentam apenas uma classe e foram capturadas por drones ucranianos em situações reais, refletindo cenários mais

FIGURA 3.4: Duas imagens retiradas do dataset referentes ás sequências da RCA.

próximos de uma implementação prática do estudo em questão. Esses vídeos são caracterizados por uma maior movimentação de câmara e, devido à transmissão em tempo real para um posto de comando, a qualidade das imagens é significativamente inferior em comparação com as sequências da RCA. Essa diferença de qualidade pode prejudicar os resultados, especialmente devido à presença de borrões (*blur*) nas imagens, uma característica mais proeminente nos vídeos da Ucrânia. Exemplos dessas sequências podem ser observados na figura 3.5 logo abaixo.

FIGURA 3.5: Duas imagens retirados do dataset referentes ás sequências da Ucrânia.

Em resumo, o dataset de teste abrange uma variedade de condições e contextos, oferecendo uma amostra representativa para avaliar a eficácia do estudo proposto. A diversidade presente nas sequências contribui para uma análise abrangente e robusta dos resultados obtidos. É importante destacar que o dataset difere significativamente dos *benchmarks* convencionais, o que implica em resultados distintos daqueles obtidos pelos modelos estado de arte. O propósito é testar esses modelos em situações militares realistas, proporcionando *insights* valiosos sobre sua eficácia e adequação para cenários específicos.

Tabela 3.1: Dataset de Teste

Sequências	Nº classes	Nº Anotações	Nº Imagens	Resolução	Oclusão
RCA 1	2	2177	364	1920x1080	Sim
RCA 2	2	1445	498	1920x1080	Sim
RCA 3	2	1509	139	1920x1080	Sim
Ucrânia 1	1	500	123	840x560	Não
Ucrânia 2	1	511	511	840x560	Não
Ucrânia 3	1	499	306	840x560	Não
Ucrânia 4	1	785	282	840x560	Não

3.2.2 Modelo pré-treinado em MOTChallenge

Ao comparar os resultados dos testes com os modelos X e NANO em MOT (*Multiple Object Tracking*), observamos diferenças significativas nas métricas de desempenho. É importante ressaltar que os modelos X e NANO representam diferentes tamanhos e complexidades de arquitetura, onde o modelo X é maior e mais complexo, enquanto o modelo NANO é menor e mais simples.

Em relação à métrica MOTA , que avalia a precisão global do seguimento de objetos, notamos que o modelo X apresenta um desempenho superior em comparação com o modelo NANO. Isso sugere que o modelo X é mais eficaz em lidar com uma variedade maior de cenários e condições, devido à sua capacidade de aprendizagem e representação de características complexas dos objetos.

Tabela 3.2: Teste com modelo X em MOT

SEQ.	IDF1	IDP	IDR	MT	PT	ML	FP	FN	IDsw	FM	MOTA	MOTP
RCA 1	0.75	0.91	0.63	5	5	1	2	656	11	22	0.69	0.77
RCA 2	0.45	0.84	0.31	1	2	9	0	912	10	20	0.36	0.50
RCA 3	0.53	0.56	0.51	10	2	2	75	220	27	27	0.79	0.56
Ucrânia 1	0.70	0.94	0.56	0	4	1	0	203	3	9	0.59	0.38
Ucrânia 2	0.57	0.85	0.43	0	1	0	0	253	1	25	0.50	0.25
Ucrânia 3	0.19	0.78	0.11	0	2	3	1	431	2	6	0.13	0.30
Ucrânia 4	0.14	0.81	0.07	0	1	3	0	713	4	7	0.09	0.76

Tabela 3.3: Teste com modelo NANO em MOT

SEQ.	IDF1	IDP	IDR	MT	PT	ML	FP	FN	IDsw	FM	MOTA	MOTP
RCA 1	0.44	1.00	0.28	0	4	7	0	1569	2	5	0.28	0.34
RCA 2	0.22	0.99	0.12	0	3	9	0	1264	2	7	0.12	0.49
RCA 3	0.42	0.79	0.29	0	10	4	0	962	17	28	0.35	0.40
Ucrânia 1	0.29	1.00	0.17	0	1	4	0	415	1	5	0.17	0.31
Ucrânia 2	0.73	0.89	0.62	0	1	0	0	154	1	31	0.70	0.24
Ucrânia 3	0.14	0.68	0.08	0	2	3	0	440	3	6	0.11	0.37
Ucrânia 4	0.02	1.00	0.01	0	0	4	0	775	0	0	0.01	0.25

Por outro lado, ao analisar a métrica MOTP , que mede a precisão espacial do seguimento de objetos, observamos que o modelo NANO tende a ter valores mais altos em algumas sequências, enquanto o modelo X se destaca em outras. Isso pode indicar que a precisão não depende muito do tamanho do modelo.

Quanto às métricas IDF1 , IDP e IDR , que avaliam o desempenho do seguimento de identidades de objetos, o modelo X também tende a superar o modelo NANO. Isso sugere que o modelo X é melhor em manter e reconhecer a identidade dos objetos ao longo do tempo.

Em resumo, analisando as tabelas , podemos observar que o modelo X se destaca principalmente nas sequências "RCA 1", "RCA 2"e "RCA 3". Nessas sequências, o modelo X apresenta valores mais altos em métricas como IDF1 e MOTA em comparação com o modelo NANO. Isso sugere que o modelo X tem um desempenho superior na deteção e seguimento de objetos nestas sequências específicas.

3.2.3 Modelo pré-treinado em MS COCO

Começando com o modelo pré-treinado em MS COCO, podemos ver que ele alcançou resultados relativamente altos em termos de precisão (IDP), *recall* (IDR) e pontuação F1 (IDF1) para a maioria das sequências. Por exemplo, na sequência "RCA 1", o modelo obteve uma pontuação IDF1 de 0.88, indicando uma alta precisão e *recall* na deteção de objetos. No entanto, ao observar a sequência "Ucrânia 1", vemos uma queda significativa no desempenho, com um IDF1 de apenas 0.55. Isso sugere que o modelo pode ter dificuldade em lidar com certos tipos de cenários, objetos específicos ou baixas resoluções de imagem.

Em termos de seguimento de múltiplos objetos, o modelo pré-treinado em MS COCO obteve resultados variados. Por exemplo, na sequência "RCA 1", ele conseguiu efetuar seguimento na maioria dos objetos corretamente (MT = 9), enquanto na sequência "Ucrânia 1", não conseguiu efetuar seguimento em nenhum objeto consistentemente (MT = 0). Isso pode indicar que o modelo pode não ser tão eficaz em lidar com movimentos rápidos ou baixas resoluções.

Por outro lado, ao analisar os resultados do modelo pré-treinado em MOT, vemos uma tendência diferente. Este modelo parece ter um desempenho inferior em comparação com o modelo MS COCO em termos de precisão e *recall*. Por exemplo, na sequência "RCA 1", o modelo alcançou um IDF1 de apenas 0.50, em comparação com 0.88 do modelo MS COCO. Isso sugere que o modelo MOT pode ter dificuldade em detetar objetos fora do padrão para o qual foi treinado, no entanto

se a deteção é efetuada ele consegue manter as suas identidades por mais tempo de forma consistente em comparação com os modelos pré-treinados em MS COCO.

Tabela 3.4: Teste com modelo X em MS COCO

SEQ.	IDF1	IDP	IDR	MT	PT	ML	FP	FN	IDsw	FM	MOTA	MOTP
RCA 1	0.88	0.94	0.82	9	2	0	7	285	9	19	0.86	0.91
RCA 2	0.56	0.78	0.43	1	6	5	8	650	10	17	0.54	0.88
RCA 3	0.63	0.77	0.54	4	8	2	1	451	18	27	0.69	0.89
Ucrânia 1	0.55	0.94	0.39	0	4	1	0	292	4	9	0.41	0.83
Ucrânia 2	0.36	0.51	0.28	0	1	0	0	229	3	57	0.55	1.00
Ucrânia 3	0.18	0.93	0.10	0	2	3	0	444	1	11	0.11	1.00
Ucrânia 4	0.19	0.69	0.11	0	1	3	19	677	5	11	0.11	1.00

Além disso, o modelo MOT também parece ter dificuldade em efetuar seguimento de múltiplos objetos, como evidenciado pela baixa contagem de objetos seguidos corretamente (MT) em várias sequências. Por exemplo, na sequência "Ucrânia 2", o modelo só conseguiu seguir corretamente um único objeto (MT = 1), enquanto na sequência "Ucrânia 4", não conseguiu seguir nenhum objeto corretamente (MT = 0).

Tabela 3.5: Teste com modelo NANO em MS COCO

SEQ.	IDF1	IDP	IDR	MT	PT	ML	FP	FN	IDsw	FM	MOTA	MOTP
RCA 1	0.50	0.95	0.34	1	3	7	0	1409	6	10	0.35	0.92
RCA 2	0.37	0.89	0.23	1	4	7	0	1068	7	21	0.26	0.89
RCA 3	0.25	1.00	0.14	1	3	10	0	1297	6	10	0.14	0.83
Ucrânia 1	0.07	1.00	0.03	0	0	5	0	483	0	2	0.03	0.71
Ucrânia 2	0.22	0.50	0.14	0	1	0	0	364	4	11	0.28	1.00
Ucrânia 3	0.03	1.00	0.01	0	0	5	0	492	0	0	0.01	1.00
Ucrânia 4	0.01	1.00	0.00	0	0	4	0	783	0	0	0.00	1.00

3.2.4 Discussão e escolha do modelo para treino

Com base na proposta de realizar seguimento em tempo real e considerando os recursos disponíveis para o treino do modelo, a decisão foi escolher o YOLO Nano. Embora a tabela tenha sido útil para destacar as diferenças de desempenho entre os diferentes tamanhos de modelos, a escolha final foi influenciada principalmente pela capacidade do YOLO Nano de realizar o seguimento em tempo real de forma eficiente. Com apenas 3 milhões de parâmetros, o YOLO Nano é mais leve e mais rápido do que os modelos maiores, como o YOLO X e o Byte X. Isso é particularmente relevante considerando que o treino será conduzido na plataforma Coolab da Google, que oferece recursos computacionais limitados em comparação com infraestruturas dedicadas. Para contextualizar, enquanto o YOLO Nano pode executar o seguimento em 7ms (142 FPS), o YOLO X, com 68 milhões de parâmetros, leva cerca

de 200ms (5 FPS) para a mesma tarefa. Além disso, é importante mencionar que a escolha do YOLO Nano é compatível com a disponibilidade da GPU A100 (com 40 GB de VRAM) na plataforma Coolab da Google. Isso contrasta com os requisitos de hardware do modelo Byte X, que foram treinados usando 8 unidades NVIDIA V100. Portanto, a escolha do YOLO Nano alinha-se com a necessidade de realizar experimentos práticos em cenários militares realistas, garantindo um desempenho satisfatório dentro das restrições de hardware e de tempo real.

Tabela 3.6: Média das métricas para todos os modelos

Modelo	**IDF1**	**MOTA**	**MOTP**
BYTE X	0.43	0.38	0.55
BYTE NANO	0.26	0.20	0.41
YOLO X	0.50	0.43	0.89
YOLO NANO	0.21	0.18	0.90

3.3 Conjunto de Dados de Treino

O conjunto de dados de treino desempenha um papel fundamental no desenvolvimento de um modelo de seguimento. Nesta secção, detalhamos o conjunto de dados utilizados, incluindo seu tamanho, a variedade de objetos seguidos e a qualidade das anotações.

3.3.1 Visdrone

O VisDrone-DET2019 Dataset(P. Zhu, Wen, Du, Bian, Fan, Q. Hu e Ling 2021), é focado na deteção de múltiplas classes. O conjunto de dados consiste em 8.599 imagens capturadas por plataformas de drones em diferentes locais e alturas, totalizando mais de 540 mil caixas delimitadoras anotadas para dez categorias pré-definidas. Estas categorias incluem pedestres, pessoas, carros, vans, autocarros, caminhões, motas, bicicletas e triciclos.

O conjunto de dados é dividido em subconjuntos de treino, validação e teste, consistindo em 6.471, 548 e 1.580 imagens, respetivamente. Embora as imagens sejam coletadas de diferentes locais, elas compartilham ambientes semelhantes, garantindo uma representação abrangente dos cenários de deteção como mostra a figura 3.6.

O conjunto de dados VisDrone foi escolhido como o principal conjunto de dados para treinar o detetor de objetos, devido à sua relevância e adequação ao

FIGURA 3.6: Exemplo de imagens do VisDrone. Retirado de P. Zhu, Wen, Du, Bian, Fan, Q. Hu e Ling 2021

.

escopo do projeto. O VisDrone se destaca como o único *benchmark* disponível para o seguimento de múltiplos objetos (MOT) que utiliza imagens de veículos aéreos não tripulados (UAVs). Embora existam outros conjuntos de dados, como o UAVDT(*ResearchGate* s.d.) e o UAV123(Mueller, Smith e Ghanem 2016a), esses são destinados principalmente ao seguimento de objetos únicos (SOT - *Single Object Tracking*), o que limita sua aplicabilidade para o treino de deteção de objetos em cenários de múltiplos alvos.

3.3.2 Base de dados Escola Naval/ Fuzileiros

Utilizei um dataset de Santos, Rodrigues, A. B. Pinto e Damas 2023 retirado de uma dissertação desenvolvida pelo GMAR Batista Pinto, "Deteção de alvos terrestres a partir de um UAV, em apoio a Operações Anfíbias". Esse conjunto de dados contém imagens dos Fuzileiros Portugueses na Lituânia e exercícios da Escola Naval em Troia, todos capturados com um drone. Exemplos desses dois cenários podem ser vistos na figura 3.7.

A escolha dessa base de dados se deve à sua especificidade militar, que o VisDrone não possui por ser voltado para fins civis. Além disso, enquanto o VisDrone inclui uma variedade de classes além de pedestres, nosso interesse está focado especificamente em pedestres.

3.3.3 Base de dados militar

Como será evidenciado pelos resultados obtidos nos conjuntos de dados VisDrone(Wen et al. 2019) e militar da Escola Naval/Fuzileiros(Santos, Rodrigues,

FIGURA 3.7: Duas imagens retiradas do dataset militar de Pinto Santos, Rodrigues, A. B. Pinto e Damas 2023.

A. B. Pinto e Damas 2023), os resultados obtidos até então não foram satisfatórios. Diante dessa constatação, foi decidido treinar o modelo com mais um conjunto de dados. Este conjunto de dados militar, obtido de uma plataforma *open-source*(2 2023), consiste em uma base de dados com 9263 imagens extraídas da plataforma Roboflow. As imagens abrangem uma ampla variedade de resoluções, ângulos de visão e ambientes diversos.

FIGURA 3.8: Duas imagens retiradas do dataset militar do Roboflow 2 2023.

Acredito que essa diversidade, como será demonstrado no próximo capítulo, pode ter um impacto significativo nos resultados do *tracker*. Considerando que o principal problema para o desempenho insatisfatório do *tracker* parece ser o detetor, a inclusão deste novo conjunto de dados pode contribuir para melhorar a precisão e a robustez do seguimento de objetos.

Exemplos de imagens retiradas desse dataset podem ser observados na figura 3.8.

O dataset tem apenas uma classe anotada cujo autor deu o nome de "*soldier-person*", ou seja tanto militares e pessoas são anotadas numa única classe só, o que vem de em conta com o objetivo do trabalho de detetar qualquer pedestre nas sequências de vídeo.

3.4 Configurações de Treino do detetor

As configurações de treino, como taxa de aprendizagem, número de épocas de treino, tamanho do lote (*batch size*), algoritmo de optimização e outros hiperparâmetros, são cruciais para o sucesso do modelo de seguimento. Nesta secção, fornecemos uma visão geral das configurações de treino utilizadas.

Tabela 3.7: Parâmetros de Treino

Parâmetro	**Valor**
Epochs	300
Batch	16
Imgsz	640
Pretrained	true
Optimizer	auto
Learning Rate	0.01
Augment	false
Plots	true
Val	true

Apartir da tabela 3.1 vamos explicar a escolha de cada argumento para o treino:

- **Épocas:** O número total de épocas de treino é 300. Uma época é uma passagem completa pelos dados de treino.
- **Batch size:** O tamanho do lote de treino é 16. Isso significa que 16 amostras são processadas antes de atualizar os pesos do modelo.
- **Tamanho da Imagem:** O tamanho da imagem de entrada é 640x640 pixels.
- **Pré-Treino:** O modelo está pré-treinado, o que significa que os pesos do modelo foram inicializados com pesos de outro modelo treinado em um conjunto de dados semelhante.
- **Optimizador:** O optimizador é definido como "auto", o que pode significar que o optimizador será escolhido automaticamente com base no modelo e na tarefa.

- **Taxa de aprendizagem:** A taxa de aprendizagem inicial é 0.01 e é ajustada automaticamente durante o treino.
- **Aumento:** O aumento de dados durante o treino está desativado (*false*). O aumento de dados é uma técnica comum para aumentar a diversidade do conjunto de dados de treino, o que pode ajudar a melhorar a generalização do modelo.
- **Plots:** Os gráficos de treino serão gerados durante o treino (*true*), o que pode ser útil para monitorizar o progresso do treino.
- **Validação:** O conjunto de validação será usado durante o treino (*true*), o que permite avaliar o desempenho do modelo em um conjunto de dados separado.

3.5 Avaliação do Modelo durante o Treino

Durante o treino do modelo de seguimento, é importante avaliar seu desempenho para garantir que esteja progredindo na direção desejada. Esta secção discute as métricas de desempenho utilizadas para avaliar o modelo durante o treino, como perda (*loss*), precisão (*accuracy*), *recall* e outras métricas relevantes.

A curva de precisão e *recall* é gerada variando o *threshold* de decisão do modelo e calculando a precisão e a *recall* correspondentes para cada valor de *threshold*. Geralmente, o *threshold* é ajustado para controlar o equilíbrio entre precisão e *recall*, já que aumentar um geralmente leva à diminuição do outro. A área sob a curva de precisão e *recall* (*Area Under the Precision-Recall Curve - AUC-PR*) é frequentemente usada como uma medida do desempenho do modelo, onde valores mais altos indicam um desempenho melhor.

Abordando agora as métricas de perda avaliadas no conjunto de validação, elas são essenciais para monitorizar o desempenho do modelo em dados nunca vistos durante o treino e para evitar o sobre-ajuste (*overfitting*), garantindo que o modelo generaliza bem para novos dados.

- **Loss de caixas (Validação):** Esta curva representa a perda associada à predição das coordenadas das caixas delimitadoras (*bounding boxes*) no conjunto de validação. Assim como na métrica de treino correspondente, a perda de caixas no conjunto de validação é calculada com base na diferença entre as coordenadas das caixas previstas pelo modelo e as coordenadas verdadeiras das caixas no conjunto de validação;

- **Loss de Classificação (Validação):** Esta curva representa a perda associada à predição das classes dos objetos contidos nas caixas delimitadoras no conjunto de validação. Similarmente à métrica de treino correspondente, a perda de classificação no conjunto de validação é calculada com base na diferença entre as probabilidades previstas para cada classe e as classes verdadeiras dos objetos no conjunto de validação;
- **Loss de Filtragem dinâmica (Validação):** Esta curva representa a perda associada à aplicação de filtros dinâmicos no conjunto de validação durante o treino do modelo. A perda de filtragem dinâmica no conjunto de validação é calculada com base na diferença entre as predições filtradas e as caixas verdadeiras no conjunto de validação, da mesma forma que na métrica de treino correspondente.

As métricas mAP50 e mAP50-95 são variações da métrica de *Average Precision (mAP)* usadas para avaliar o desempenho de modelos de deteção de objetos em tarefas de deteção de múltiplas classes. No caso da mAP50, esta métrica calcula a média da precisão para cada classe de objeto em um determinado intervalo de confiança, geralmente definido em 50 por cento, considera apenas as deteções cuja confiança está acima de um limiar específico (50 por cento deste caso). No caso da mAP50-95, esta métrica calcula a média da precisão para cada classe de objeto em um intervalo de confiança variável, geralmente de 50 a 95 por cento.

3.5.1 Visdrone

Para o dataset Visdrone foram realizadas todas as 300 épocas de treino. Como se pode observar pela figura 3.10 os valores de mAP obtidos para cada classe são distintos (entre 0.095 e 0.768). Isso se deve em princípio ao número de instâncias em que as classes são treinadas, ou seja existe muitos mais dados na classe carro (140 mil) do que na classe bicicleta (10 mil). No entanto a classe a ter em conta será apenas a de pedestres que obteve 0.366 mAP.

Nas próximas figuras logo abaixo, nomeadamente as figuras 3.11 e 3.12 o objetivo, da primeira é verificar se o comportamento do modelo durante o treino é adequado. Ou seja se os valores de perda(loss) vão diminuindo ao longo das épocas de treino. Pela análise da figura 3.11, ao chegar perto das 300 épocas se verifica que houve sim um aumento o que indica um ligeiro *overfitting*. No entanto, durante o treino são sempre guardados dois pesos, o último e o com melhor avaliação (mAP). Resumindo o gráfico indica que treino adicional não melhoraria os resultados adquiridos.

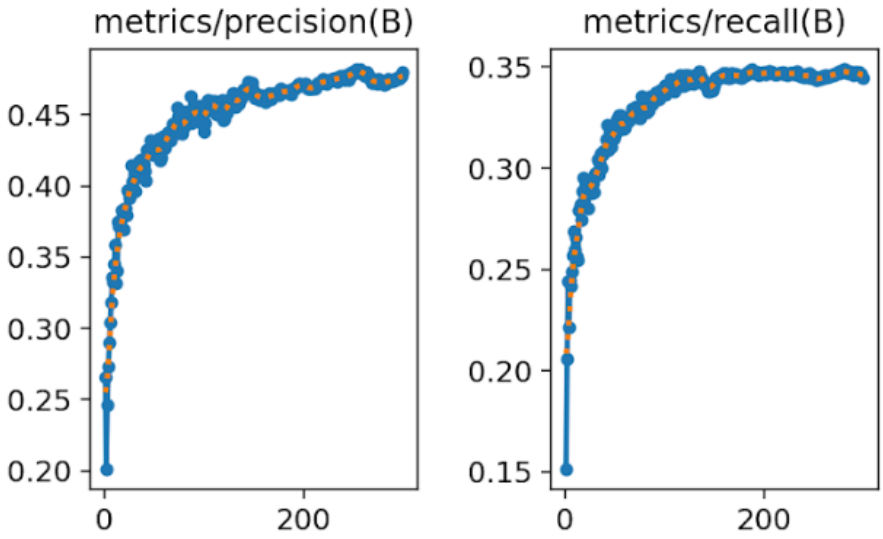

FIGURA 3.9: Imagem referente aos gráficos de *precision* e *recall* ao longo das épocas no dataset Visdrone.

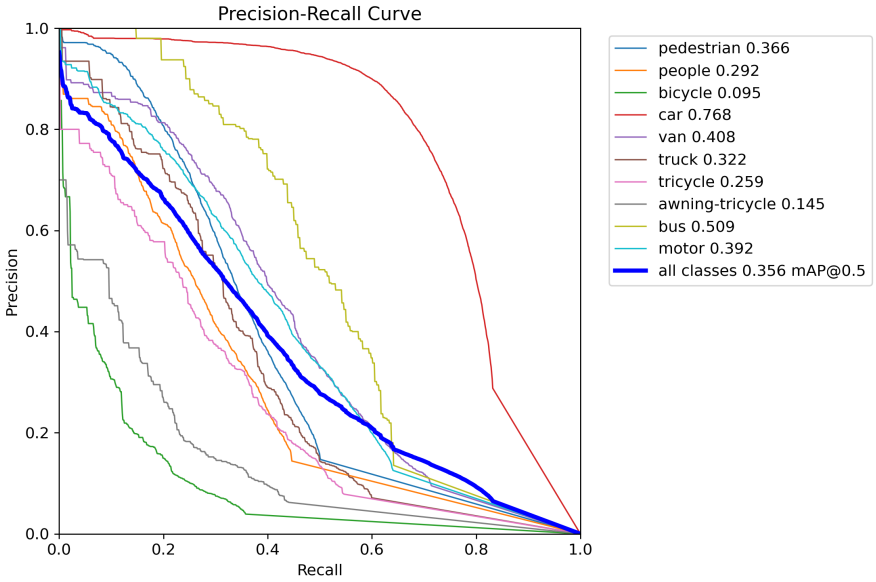

FIGURA 3.10: Imagem referente ao gráfico de de curva *Precision-Recall* no dataset VisDrone.

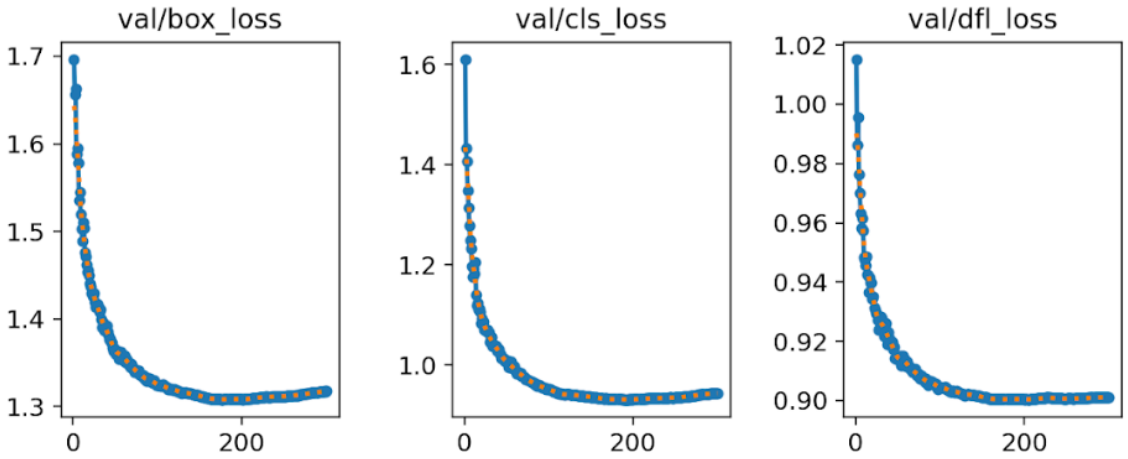

FIGURA 3.11: Imagem referente aos gráficos de validação ao longo das épocas no dataset VisDrone.

A mesma lógica se verifica na figura 3.12, próximo da época 200 o modelo atinge um *plateou* e desce os valores de mAP ligeiramente no final das 300 épocas, o que indica mais uma vez que não é necessário mais treino para melhores resultados.

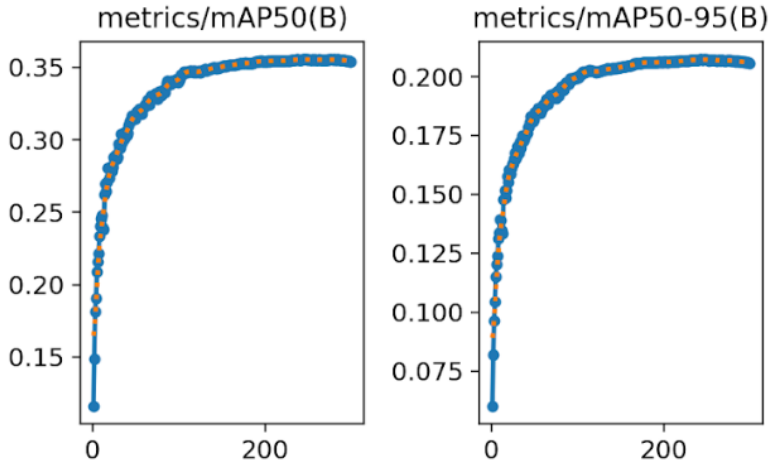

FIGURA 3.12: Imagem referente aos gráficos de mAP do dataset VisDrone.

3.5.2 Base de Dados Escola Naval/Fuzileiros

Passando agora á base de dados de GMAR Batista Pinto e adaptada para esta dissertação foram realizados ao todo 300 épocas de treino. Pela leituras das figuras 3.13 e 3.14 verificamos que o treino no dataset militar de Pinto, obteve melhores resultados. Apesar do treino ter sido efetuado apenas para uma classe e que isso possa ser uma das justificativas para melhores resultados, o valor de mAP obtido, 0.931, é consideravelmente superior. O que não indica que necessariamente

representará um bom desempenho nos testes de seguimento abordados no próximo capítulo.

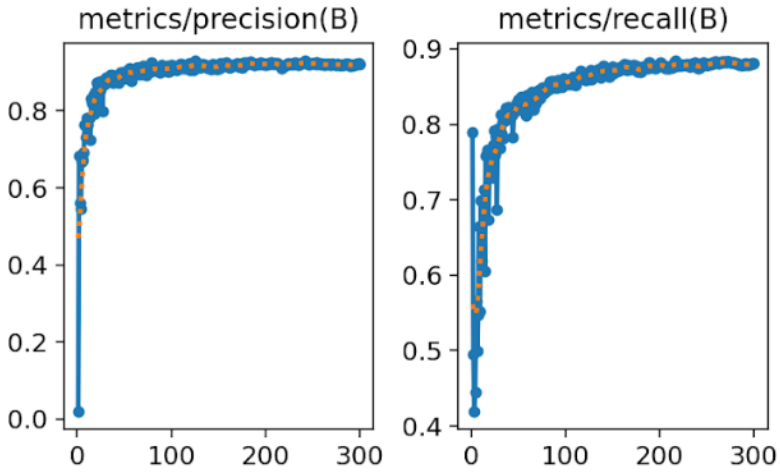

FIGURA 3.13: Imagem referente aos gráficos de precision e recall ao longo das épocas no dataset escola naval/fuzileiros.

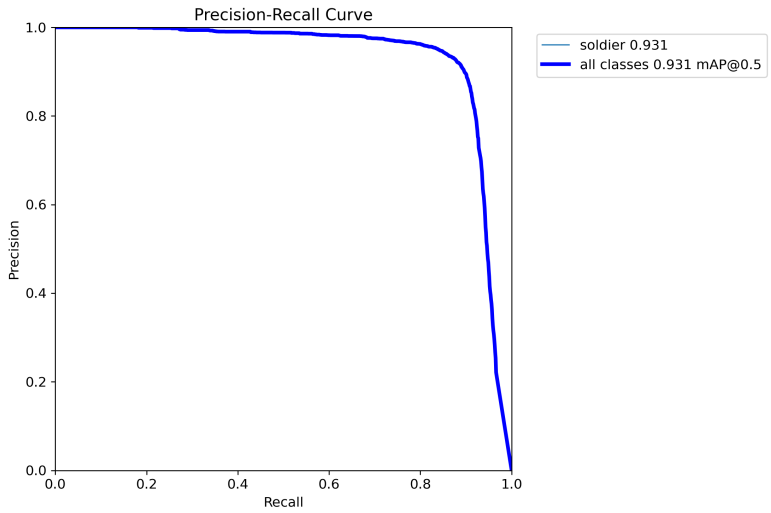

FIGURA 3.14: Imagem referente aos gráficos de precision e recall ao longo das épocas no dataset escola naval/fuzileiros.

Abordando agora os gráficos da figura 3.15 referentes ás curvas de *loss* ou validação, o objetivo é verificar se existe um *overfitting* ou sobre ajuste. Como os valores de loss, vão diminuindo ao longo das 300 épocas isso indica que não houve treino desnecessário. Mesmo os melhores pesos não tendo sido obtidos após a última época de treino, mais treino poderia por ventura trazer um melhor resultado, mas nada de muito substancial, analisando o comportamento dos gráficos estes aparentam ter atingido um *plateou*.

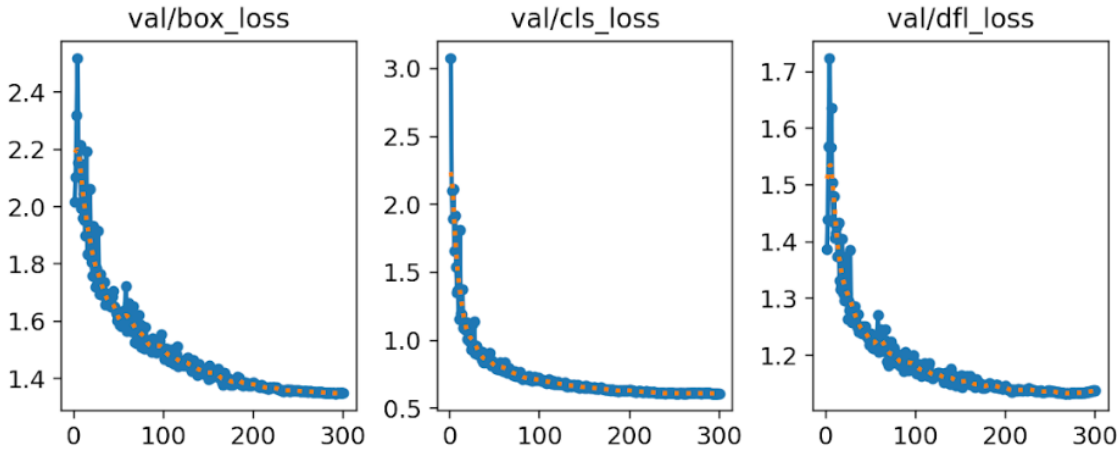

FIGURA 3.15: Imagem referente aos gráficos de validação ao longo das épocas no dataset escola naval/fuzileiros.

Por fim, como demonstra a figura 3.16, os valores obtidos para mAP50 e mAP50-95 foram respetivamente, 0.931 e 0.590. Como já referido, foram resultados mais satisfatórios que os anteriores obtidos no VisDrone.

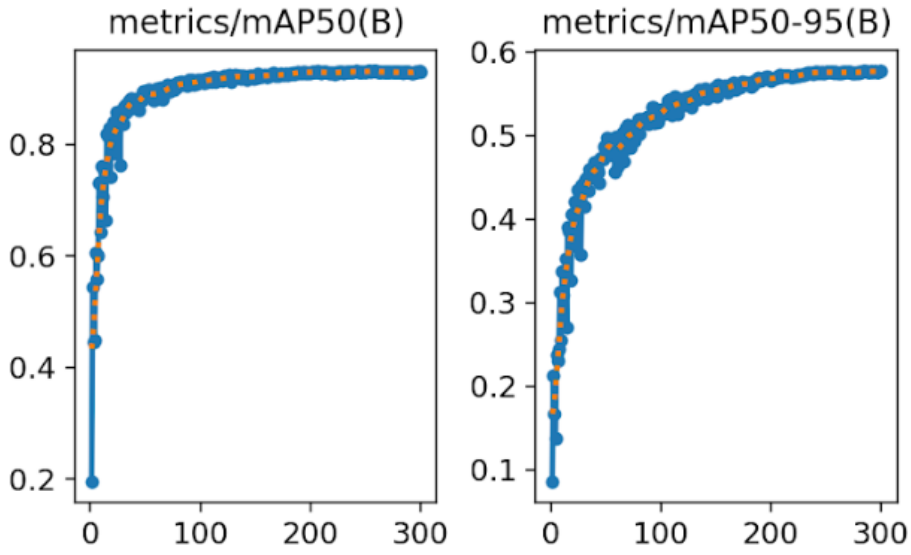

FIGURA 3.16: Imagem referente aos gráficos de mAP do dataset escola naval/fuzileiros.

3.5.3 Base de Dados Militar

Ao todo foram realizadas 270 épocas de treino, tendo em conta que se não houver nas últimas 50 épocas um valor superior de mAP o treino é interrompido automaticamente.

Começando pelos gráficos de precisão e *recall* representados pela figura 3.17, o valor máximo de precisão atingido foi de 0.86 na época 59, enquanto que o do *recall* o seu pico deu-se apenas por volta da época 200 com 0.77.

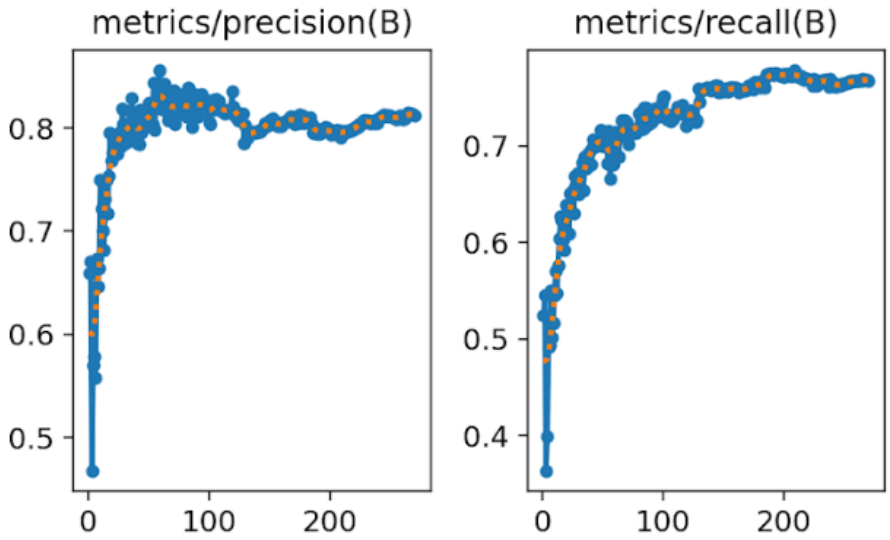

FIGURA 3.17: Imagem referente aos gráficos de precision e recall ao longo das épocas no dataset militar.

Passando agora á curva de Precisão-*Recall*, como seria de esperar, o aumento de uma representa a diminuição da outra, e portanto o gráfico representa o comportamento esperado para o treino. Sendo assim o resultado do mAP onde as duas métricas se relacionam melhor de 0.811 considerando uma confiança a partir de 0.5.

Passando agora aos gráficos de validação representados pela figura 3.19, o importante é verificar se existe o *overfitting* ao fim de algumas épocas de treino, isso pode ser observado pela subida do gráfico ao longo das épocas. Perto das 300 épocas já se pode observar uma leve subida mas como o algoritmo por si só interrompe o treino ao fim de 50 épocas sem melhorias, não existe esse problema.

Por fim na figura 3.20 é possível visualizar os resultados para mAP50 e mAP50-95 ao longo das épocas. Os valores obtidos foram de 0.811 e 0.545 respetivamente.

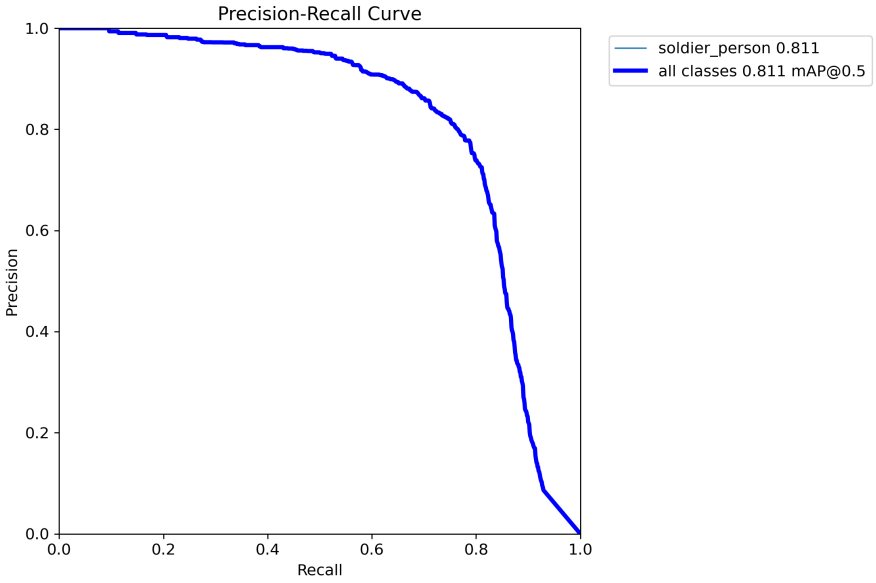

FIGURA 3.18: Imagem referente ao gráfico da curva *Precision-Recall* ao longo das épocas no dataset militar.

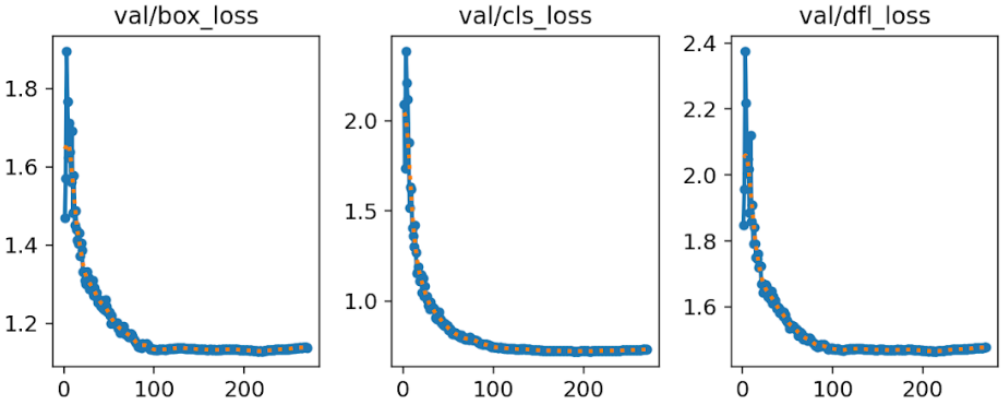

FIGURA 3.19: Imagem referente aos gráficos de validação ao longo das épocas no dataset militar.

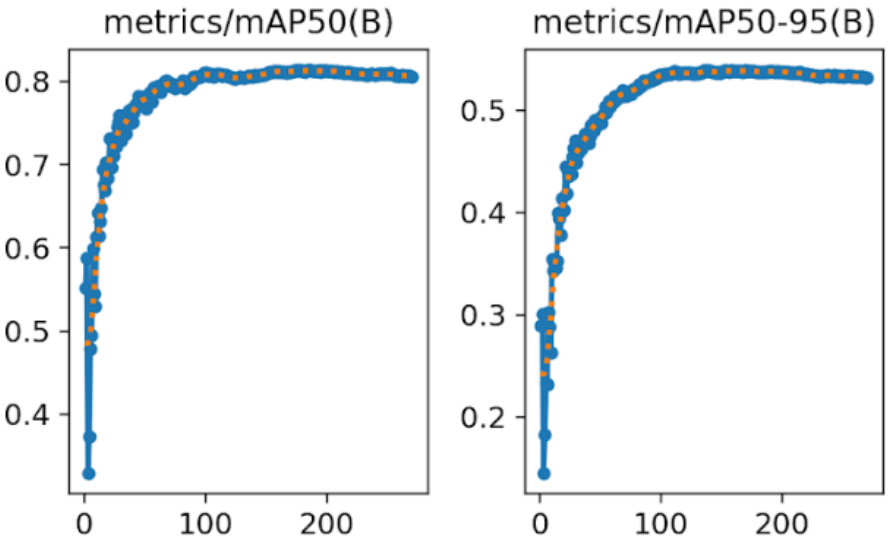

FIGURA 3.20: Imagem referente aos gráficos de mAP do dataset militar.

Capítulo 4

Apresentação e Discussão dos Resultados

Neste capítulo, faremos uma análise em detalhe dos resultados obtidos após treino adicional, comparando-os com os resultados obtidos com o ByteTrack inicializado com o MS-COCO, e descritos na secção 2.4.10. Além disso, compararemos o desempenho do nosso modelo com os resultados de outros trabalhos relacionados, para contextualizar sua eficácia em relação ao estado da arte no seguimento de múltiplos objetos.

4.1 Desempenho do Modelo após treino adicional

Como descrito no capítulo anterior, o detetor inicial foi tentativamente melhorado através do treino com três conjuntos de dados representativos de imagens de UAV, incluindo dois com imagens de militares. Analisamos de seguida os resultados obtidos após treino com cada um desses conjuntos.

4.1.1 Visdrone

Ao analisar os resultados obtidos no modelo nano (pré-treinado em MS COCO) treinado no Visdrone, podemos identificar uma variedade de desempenhos nas diferentes sequências avaliadas como se pode ver na tabela 4.1.

Começando pela sequência RCA 1, observamos um IDF1 relativamente baixo, indicando uma precisão e *recall* deficientes. Isso pode ser atribuído ao alto número de falsos negativos (FN), sugerindo que o modelo teve dificuldade em detetar corretamente objetos e em classificá-los adequadamente.

Na sequência RCA 2, embora o IDF1 seja um pouco mais alto do que na sequência anterior, ainda é consideravelmente baixo. Especificamente, a taxa

de deteção (IDR) é notavelmente baixa, indicando que muitos objetos não foram detetados pelo modelo.

Por outro lado, na sequência RCA 3, observamos uma melhoria significativa no IDF1 em comparação com as sequências anteriores. Embora ainda existam áreas onde o modelo pode ser aprimorado, especialmente em termos de atribuição de identidades aos objetos seguidos, como indicado pelo alto número de mudança de ID (IDsw).

Infelizmente, não temos dados disponíveis para as sequências da Ucrânia (Ucrânia 1, 2 e 3), dificultando qualquer análise específica dessas sequências, pelo que o modelo não foi capaz de fazer seguimento com as poucas deteções que obteve.

Tabela 4.1: Resultados obtidos com treino no Visdrone

SEQ.	**IDF1**	**IDP**	**IDR**	**MT**	**PT**	**ML**	**FP**	**FN**	**IDsw**	**FM**	**MOTA**	**MOTP**
RCA 1	0.08	1.0	0.04	0	1	10	0	2085	1	3	0.04	1
RCA 2	0.11	1.0	0.6	0	1	11	0	1364	0	1	0.06	0.89
RCA 3	0.39	0.93	0.25	1	5	8	0	1109	12	26	0.26	0.87
Ucrânia 1	X	X	X	X	X	X	X	X	X	X	X	X
Ucrânia 2	X	X	X	X	X	X	X	X	X	X	X	X
Ucrânia 3	X	X	X	X	X	X	X	X	X	X	X	X
Ucrânia 4	0.51	0.55	0.47	1	3	0	132	254	21	16	0.48	0.82

Por fim, na sequência Ucrânia 4, vemos um IDF1 relativamente melhor em comparação com as sequências RCA. No entanto, ainda há uma quantidade considerável de falsos positivos e falsos negativos, destacando áreas onde o modelo ainda precisa ser refinado.

Em resumo, as pontuações de MOTA variaram entre as sequências avaliadas, refletindo os desafios enfrentados pelo modelo. Nas sequências RCA 1, 2 e 3, observou-se baixas pontuações de MOTA, indicando dificuldades em manter o seguimento preciso dos objetos devido a falsos positivos, falsos negativos e erros na atribuição de identidades. Por outro lado, na sequência Ucrânia 4, onde foram obtidos dados de deteção, houve uma pontuação relativamente melhor de MOTA, sugerindo um desempenho mais robusto do modelo nessa configuração específica. No entanto, ainda há espaço para melhorias, especialmente na redução de falsos positivos e falsos negativos.

4.1.2 Base de dados Escola Naval/Fuzileiros

Ao analisar os resultados obtidos no modelo nano treinado no dataset da escola naval/fuzileiros, observamos uma variedade de desempenhos nas diferentes sequências avaliadas, conforme apresentado na Tabela 4.2.

Tabela 4.2: Resultados obtidos com treino no dataset Escola Naval/Fuzileiros

SEQ.	IDF1	IDP	IDR	MT	PT	ML	FP	FN	IDsw	FM	MOTA	MOTP
RCA 1	0.57	0.99	0.40	1	4	6	0	1301	4	11	0.40	0.97
RCA 2	0.14	0.97	0.07	0	1	11	3	1340	2	7	0.07	0.96
RCA 3	X	X	X	X	X	X	X	X	X	X	X	X
Ucrânia 1	0.24	1.0	0.13	0	2	3	0	433	1	2	0.13	0.99
Ucrânia 2	0.01	0.4	0.00	0	1	0	0	506	2	2	0.01	0.91
Ucrânia 3	0.49	0.35	0.86	1	4	0	732	1	4	1	-0.48	0.99
Ucrânia 4	0.57	0.58	0.56	1	3	0	153	183	24	16	0.54	1.00

Ao analisar os resultados apresentados na tabela, é evidente que enfrentamos desafios significativos em algumas das sequências avaliadas. Na sequência RCA 3, por exemplo, não foi possível realizar o seguimento. Isso sugere que pode haver problemas ou limitações específicas relacionadas a essa sequência que impossibilitaram a avaliação do desempenho do modelo.

Além disso, observamos um número relativamente elevado de falsos negativos em todas as sequências, o que implica que muitos objetos não foram detetados pelo modelo. Isso é preocupante, pois os falsos negativos afetam diretamente a capacidade do modelo de realizar o seguimento correto dos objetos ao longo do tempo.

Por outro lado, nas sequências Ucrânia 3 e 4, observamos um número considerável de falsos positivos, representados pelos valores nas colunas "FP". Isso indica que o modelo detetou erroneamente objetos que na verdade não estavam presentes na cena.

Além disso, a maioria dos objetos em todas as sequências foram classificados como "PT"(parcialmente seguidos) ou "ML"(principalmente perdidos), indicando que o modelo teve dificuldade em manter o seguimento consistente de muitos objetos ao longo do tempo. Isso sugere a necessidade de melhorias no algoritmo de seguimento para lidar com essas situações.

Por fim, ao observarmos as pontuações de MOTA, notamos que a sequência Ucrânia 3 registou uma pontuação negativa (-0.48) nesta métrica. Isso é incomum e sugere um desempenho muito pobre do modelo nesta sequência específica, o que requer uma investigação mais aprofundada para entender as causas subjacentes desse resultado negativo.

É possível obter valores negativos na métrica MOTA quando a soma dos falsos negativos (FN), falsos positivos (FP) e mudanças de identidade incorretos (IDsw) é maior do que o número total de objetos presentes nas imagens de referência (GT). Isso pode ocorrer quando o sistema de seguimento comete muitos erros, como

falhas na deteção de objetos, deteções incorretas e erros na atribuição de identidades aos objetos seguidos.

4.1.3 Base de Dados Militar

Ao comparar os resultados obtidos nos três conjuntos de dados (Visdrone, Escola Naval/Fuzileiros e Militar), podemos observar que os resultados do conjunto de dados Militar são consistentemente superiores em várias métricas.

Primeiramente, notamos que no conjunto de dados Militar, o número de objetos maioritariamente seguidos (MT) e parcialmente seguidos (PT) é significativamente maior do que nos outros conjuntos de dados, indicando um melhor desempenho na tarefa de seguimento de objetos. Isso sugere que o modelo foi capaz de acompanhar mais objetos ao longo do tempo, o que é uma métrica importante para aplicações práticas como pode ser observado na tabela 4.3.

Tabela 4.3: Resultados obtidos com treino no dataset Militar

SEQ.	**IDF1**	**IDP**	**IDR**	**MT**	**PT**	**ML**	**FP**	**FN**	**IDsw**	**FM**	**MOTA**	**MOTP**
RCA 1	0.81	0.80	0.83	8	2	1	280	211	19	15	0.77	0.95
RCA 2	0.49	0.63	0.41	2	8	2	78	587	33	36	0.52	0.96
RCA 3	0.52	0.81	0.39	1	8	5	1	785	19	40	0.47	0.89
Ucrânia 1	0.31	0.89	0.19	0	3	2	0	394	4	10	0.20	0.91
Ucrânia 2	0.89	0.84	0.94	1	0	0	90	30	0	12	0.77	0.87
Ucrânia 3	0.54	0.96	0.37	0	4	1	0	307	1	17	0.38	1.00
Ucrânia 4	0.49	0.56	0.43	1	3	0	106	295	17	24	0.47	0.96

Além disso, o número de falsos negativos (FN) no conjunto de dados Militar é menor em comparação com os outros conjuntos de dados, o que significa que houve menos objetos não detetados. Isso é crucial para o sucesso do seguimento de objetos, pois falsos negativos podem resultar na perda de objetos de interesse.

As métricas de precisão (IDP) e *recall* (IDR) também são notavelmente melhores no conjunto de dados Militar, indicando uma maior precisão nas deteções e um maior sucesso em recuperar objetos verdadeiros. Isso significa que o modelo no conjunto de dados Militar teve um desempenho mais equilibrado entre a precisão das deteções e a capacidade de encontrar todos os objetos verdadeiros.

Além disso, é importante destacar que todas as sequências no conjunto de dados Militar obtiveram resultados, enquanto nos outros não. Isso demonstra uma consistência maior no desempenho do modelo no conjunto de dados Militar.

Globalmente, os resultados do conjunto de dados Militar parecem ser mais consistentes e robustos, o que se reflete na métrica MOTA, indicando um desempenho geral melhor na tarefa de seguimento de objetos em comparação com os outros conjuntos de dados.

4.2 Discussão

Como visto na secção anterior, os resultados obtidos após treino adicional com os dados do Visdrone e os da Escola Naval e Fuzileiros foram piores do que os resultados iniciais. Assim sendo, considera-se que não vale a pena usar estes dados para treino. No entanto, os resultados obtidos após treino com o "Dataset Militar"foram melhores e por isso usaremos estes para comparar com resultados obtidos com os modelos padrão (de origem) sem treino.

4.2.1 Comparação com os modelos Pré-treinados

Com base nos resultados obtidos com o modelo treinado com o dataset militar, podemos observar uma melhoria significativa em relação aos valores anteriores. Utilizando o modelo YOLO NANO pré-treinado no conjunto de dados COCO como ponto de partida, o desempenho melhorou em todas as métricas avaliadas como demonstra a tabela 4.4.

Tabela 4.4: Média das métricas para todos os modelos no dataset de teste

Modelo	**Dataset treino**	***Fine-Tuning***	**IDF1**	**MOTA**	**MOTP**
YOLO X	MS COCO	X	0.50	0.43	0.89
YOLO NANO	MS COCO	X	0.21	0.18	0.90
YOLO X	MS COCO	MOT17 e CH	0.43	0.38	0.55
YOLO NANO	MS COCO	MOT17 e CH	0.26	0.20	0.41
YOLO NANO	MS COCO	Dataset Militar	0.58	0.53	0.93

Começando pela métrica IDF1, que representa a média harmónica de precisão (IDP) e *recall* (IDR), observamos um aumento para 0.58 em comparação com os 0.21 obtidos com o modelo YOLO NANO pré-treinado no COCO. Isso indica uma melhoria na capacidade do modelo em encontrar e identificar corretamente os objetos, o que é essencial para tarefas de deteção de objetos.

A métrica MOTA, também apresentou uma melhoria substancial, subindo para 0.53 em comparação com os 0.18 do modelo pré-treinado no COCO. Isso sugere

que o modelo treinado com o dataset militar teve menos falsos positivos e erros de correspondência, o que é crucial para um sistema de seguimento confiável.

Por fim, a métrica MOTP, como se pode ler pelos gráficos tem valores muito elevados o que é mau para a tarefa de deteção. No entanto é explicado na subsecção 4.2.2 o motivo destes valores obtidos.

Considerando que o dataset MS COCO possui 80 classes diferentes e o dataset militar avaliado aqui se concentra exclusivamente na classe de pedestres, os resultados indicam que o treino foi eficaz, mesmo com uma tarefa mais específica. Além disso, é importante notar que o modelo treinado com o dataset militar superou os resultados do modelo mais robusto, YOLO X, o que mostra que treino específico pode ser mais eficaz para tarefas específicas.

4.2.2 Análise de Erros

Através das tabelas 4.2 e 4.1 é possível observar que algumas sequências não obtiveram resultados de seguimento nomeadamente para os modelos treinados nos conjuntos de dados Visdrone e Escola Naval/Fuzileiros. O primeiro passo será observar se está a ser efetuada deteção nestas sequências em específico antes de procurar resolver qualquer problema no *tracker*. Após averiguação prática nas sequencias de vídeo é possível verificar que os modelos treinados em praticamente todos os *frames* executam a deteção mas por algum motivo não conseguem manter as IDs ao longo do tempo.

Daqui só pode ser uma de duas opções: o detetor está a efetuar a deteção mas a sobreposição está abaixo do limiar definido por defeito 0.5 e portanto não é considerado deteção, ou as deteções são consideradas mas a posição prevista para o *frame* seguinte (através do filtro de kalman) não atinge o mínimo de sobreposição. Ou seja, ou temos que ajustar o limiar de deteção ou o limiar de correspondência (seguimento). Por norma primeiro se avalia o desempenho do detetor antes de passar ao *tracker*, e esse foi o rumo tomado. Mesmo com o limiar de deteção elevado para 0.9 ou seja só 10 por cento dos objetos precisariam se sobrepor aos seus *ground thruths*, verificou-se que muitas sequencias sem resultados passaram logo a obter resultados mais satisfatórios nas métricas de MOTA e IDF1.

Ou seja sabemos então que se forem dadas deteções ao algoritmo de seguimento ele é capaz de atribuir IDs e mantê-los ao longo do tempo por mais que as

deteções sejam imperfeitas. E portanto saí os valores de MOTP serem extremamente elevados porque praticamente foram consideradas todas as deteções, com o propósito de avaliar o *tracker*.

Se voltarmos á tabela 4.2, especificamente na sequencia "Ucrânia 3"foi obtido um valor negativo para MOTA (-0.48). Ora se analisarmos a fórmula de MOTA presente na subsecção 2.4.10 do capitulo 2, resultados negativos para MOTA só são possíveis se o somatório de falsos positivos com falsos negativos e mudanças de ID for superior ao número total de *Ground Truths*. Ou seja existem mais erros do que deteções e associações corretas. No entanto apenas os número de falsos positivos está fora do normal 732, comparativamente com os falsos negativos (1) e mudanças de ID (4). Ao analisar a tabela 3.1 vemos realmente que a sequencia em causa apenas tem 499 anotações (*ground truths*) o que vai de em conta com o valor negativo obtido em MOTA.

4.2.3 Robustez

Para avaliar a robustez do modelo de seguimento, será realizado um comparativo entre os resultados obtidos em dois conjuntos de sequências: o conjunto de sequências da Ucrânia e o conjunto de sequências da RCA.

No conjunto de sequências da RCA, os vídeos apresentam uma qualidade superior e estão mais próximos dos alvos. Essa proximidade proporciona uma visualização mais clara e detalhada dos objetos em movimento. No entanto, é importante ressaltar que essa condição é aplicável apenas a operações que não dependem de fatores surpresa ou que não sejam afetadas pela aproximação do drone ao alvo. Em cenários onde a aproximação do drone pode influenciar o desfecho da operação, a qualidade e proximidade dos vídeos podem não ser tão relevantes.

Por outro lado, os vídeos do conjunto de sequências da Ucrânia apresentam uma qualidade inferior devido à maior altura de captura. Essa característica pode resultar em uma menor capacidade de visualização dos objetos em movimento. No entanto, essa altura também pode ser vantajosa, pois pode reduzir a probabilidade de deteção pelo alvo e permitir a transmissão de informações a longas distâncias sem a necessidade de uma conexão de alta velocidade, como o 5G, para transmitir em tempo real.

Considerando essas diferenças nos conjuntos de sequências, é esperado que o modelo enfrente mais desafios na deteção e seguimento de objetos nos vídeos com características semelhantes aos da Ucrânia.

Tabela 4.5: Valores médios das métricas para os conjuntos de sequências da RCA e da Ucrânia

Conjunto	**IDF1**	**MOTA**	**MOTP**
RCA	0.607	0.586	0.933
Ucrânia	0.557	0.455	0.935

Os resultados apresentados na Tabela 4.5 indicam que o desempenho do modelo de seguimento permanece praticamente constante ao mudar de cenários, representados pelos conjuntos de sequências da RCA e da Ucrânia.

Além disso, os valores médios de MOTA *(Multi-Object Tracking Accuracy)* e MOTP *(Multi-Object Tracking Precision)* também mostraram estabilidade. Embora haja uma pequena diferença entre os conjuntos da RCA e da Ucrânia, com MOTA de 0.586 e 0.455, respetivamente, e MOTP de 0.933 e 0.935, respetivamente, essas diferenças são relativamente pequenas e podem ser atribuídas a variações naturais nos dados ou características específicas dos cenários.

4.2.4 Aplicação Prática

Um ponto destacável é o tempo de resposta do modelo, que foi de 7 milissegundos, correspondendo a uma taxa de 142 quadros por segundo (FPS). Esse tempo de processamento rápido é fundamental para aplicações em tempo real, como o seguimento de alvos em UAVs militares, garantindo uma resposta rápida e eficiente do sistema.

No entanto, é importante destacar que a viabilidade prática do modelo também depende da infraestrutura de comunicação disponível para transmitir os dados do drone para o posto de comando. Além disso, a capacidade de processamento disponível no posto de comando também é um fator determinante.

Com base nos testes realizados no artigo "*Intelligent UAV Based Flexible 5G EmergencyNetworks: Field Trial and System Level Results*"(Y. Gao et al. 2020), considerando a velocidade de movimento do drone (180km/h), a distância da base (4km) e a taxa de transmissão(390Mbps), é viável transmitir um vídeo de até 4K para o posto de comando através de um drone. Os resultados da simulação mostraram que o sistema UAV 5G foi capaz de satisfazer a demanda de transmissão de vídeo em diferentes resoluções, incluindo 4K, mesmo em altitudes elevadas e com velocidades consideráveis.

Mas não é necessário qualidade 4K para fins militares e operacionais, ao invés disso, se fizermos uma simples estimativa com base na resolução de entrada, a resolução aplicada nestes testes foi de 640x640, uma imagem Full HD (1920x1080) representa 5 vezes mais pixels para serem processados. O que poderia significar a princípio uma velocidade de processamento de 28fps ao invés de 142fps o que já é bom, muito perto dos 30fps ideais para exibição de um vídeo.

Além dos desafios enfrentados durante os testes, é importante reconhecer que algumas condições ideais para operações militares não foram abordadas nos cenários de teste. Por exemplo, operações noturnas com imagens infravermelhas e condições climáticas adversas, como chuva, não foram incluídas nos testes realizados.

No entanto, isso não parece ser um problema intrínseco à tecnologia em si, mas sim uma questão de disponibilidade de dados de treino adequados. Com um dataset abrangente e diversificado, é possível treinar modelos capazes de lidar com uma ampla gama de condições, incluindo aquelas encontradas em operações militares durante a noite e em condições climáticas desfavoráveis.

A tecnologia presente já demonstrou sua prontidão para ser implementada em aplicações militares como apresentado nas sequências da Ucrânia, oferecendo soluções eficazes de seguimento de alvos em tempo real. O principal desafio reside, portanto, na aquisição de dados de treino o mais vastos e diversificados possíveis. A disponibilidade de um conjunto de dados representativo e abrangente é essencial para permitir que o modelo aprenda com uma variedade de situações e condições.

Capítulo 5

Conclusão

Neste capítulo falamos sobre algumas considerações finais do trabalho, as limitações que foram surgindo ao longo do desenvolvimento, e com base nisso apresentamos propostas para trabalhos futuros na área de seguimento de objetos militares.

5.1 Limitações

Primeiramente, para a avaliação do algoritmo, foi utilizada uma GPU Nvidia L4 com 22,5 GB de VRAM, o que pode limitar os resultados obtidos neste trabalho. Apesar de ter atingido a velocidade mínima para processamento de imagem em tempo real, outras configurações com modelos mais robustos poderiam ter trazido resultados mais próximos daqueles que se observam no estado de arte.

Outra limitação a ser considerada é a escolha dos conjuntos de dados. Embora o modelo tenha sido treinado e testado em um dataset específico pertinente ao contexto militar, a representatividade desses dados em relação a uma variedade maior de cenários pode ser limitada, o que pode afetar a generalização do modelo para situações diversas. Por exemplo transpondo este trabalho para uma operação militar seria ideal ter sequencias de vídeo retirados com câmaras infravermelhas para operações noturnas, que são ideias para estas situações.

5.2 Propostas para trabalhos futuros

Considerando as limitações e os resultados deste estudo, há várias direções para trabalhos futuros que podem aprimorar e expandir o modelo proposto. Algumas propostas incluem:

- **Aquisição de Novos Dados para Treino:** A expansão do conjunto de dados de treino pode aumentar a capacidade do modelo de generalizar para uma variedade maior de cenários. A inclusão de mais dados relevantes para o contexto militar permitirá que o modelo seja mais eficaz em situações do mundo real;
- **Implementação em Sistemas UAVs:** Uma proposta interessante é a implementação do algoritmo em sistemas de Veículos Aéreos Não Tripulados (UAVs). Isso permitirá a utilização do modelo em operações militares, proporcionando vigilância aérea e seguimento de objetos em tempo real;
- **Teste em Tempo Real com Conexão 5G:** A possibilidade de testar o modelo em tempo real, por exemplo, por meio de uma conexão 5G entre o dispositivo aéreo e um servidor, para realizar o processamento, é uma proposta que visa avaliar o desempenho do modelo em condições reais. Isso permitirá uma validação mais precisa e confiável do algoritmo em ambientes operacionais.

A realização dessas propostas de trabalhos futuros proporcionará avanços significativos no campo do seguimento de objetos no contexto militar, permitindo o desenvolvimento de soluções mais eficazes e adaptáveis às necessidades específicas das operações militares.

5.3 Considerações finais

Esta dissertação visou avaliar métodos de seguimento de múltiplos alvos utilizando veículos aéreos não tripulados (UAVs) para apoio a operações militares. A principal contribuição deste estudo foi a implementação de algoritmos avançados (ByteTrack) de seguimento que demonstraram eficácia em cenários complexos e dinâmicos, mesmo que os resultados não sejam promissores, com as justificações apresentadas anteriormente concluímos sim que é possível fazer seguimento com os meios atuais tendo ao nosso dispor capacidade computacional e bases de dados extensas e diversificadas.

Os objetivos propostos no inicio deste trabalho, na nossa perspetiva foram atingidos, porque foi feita uma investigação do estado de arte acerca algoritmos de seguimento , houve uma escolha ponderada de qual se adequaria melhor ás capacidades computacionais assim como possível empregabilidade numa situação militar, otimização do algoritmo e aquisição de base de dados e utilização de já existentes.

Bibliografia

2, CIVIL MIL PERSON (jan. de 2023). – *Dataset*. https://universe.roboflow.com/civil-mil-person-2/-ibrjb. Open Source Dataset. visited on 2024-05-02. URL: https://universe.roboflow.com/civil-mil-person-2/-ibrjb.

Aharon, Nir, Roy Orfaig e Ben-Zion Bobrovsky (2022). *BoT-SORT: Robust Associations Multi-Pedestrian Tracking*.

Alom, Md Zahangir, Tarek M Taha, Christopher Yakopcic, Stefan Westberg, Paheding Sidike, Mst Shamima Nasrin, Brian C Van Esesn, Abdul A S Awwal e Vijayan K Asari (2018). «The history began from alexnet: A comprehensive survey on deep learning approaches». Em: *arXiv preprint arXiv:1803.01164*.

Betke, Margrit, Esin Haritaoglu e Larry S Davis (2000). «Real-time multiple vehicle detection and tracking from a moving vehicle». Em: *Machine vision and applications* 12, pp. 69–83.

Bewley, Alex, ZongYuan Ge, Lionel Ott, Fabio Ramos e Ben Upcroft (2016). «Simple Online and Realtime Tracking». Em: *CoRR* abs/1602.00763. URL: http://arxiv.org/abs/1602.00763.

Bolme, David S., J. Ross Beveridge, Bruce A. Draper e Yui Man Lui (2010). «Visual object tracking using adaptive correlation filters». Em: *2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pp. 2544–2550. URL: https://api.semanticscholar.org/CorpusID:2451356.

Bolshakov, Vladislav E (2024). «Multi-Agent Reinforcement Learning as Interaction Model for Online Multi-Object Tracking». Em: *2024 6th International Youth Conference on Radio Electronics, Electrical and Power Engineering (REEPE)*. IEEE, pp. 1–6.

Carion, Nicolas, Francisco Massa, Gabriel Synnaeve, Nicolas Usunier, Alexander Kirillov e Sergey Zagoruyko (2020). «End-to-End Object Detection with Transformers». Em: *CoRR* abs/2005.12872. URL: https://arxiv.org/abs/2005.12872.

Chavdarova, Tatjana, Pierre Baqué, Stéphane Bouquet, Andrii Maksai, Cijo Jose, Louis Lettry, Pascal Fua, Luc Van Gool e François Fleuret (2017). *The WILDTRACK Multi-Camera Person Dataset*.

Chu, Peng, Jiang Wang, Quanzeng You, Haibin Ling e Zicheng Liu (2021). *TransMOT: Spatial-Temporal Graph Transformer for Multiple Object Tracking*.

Ciaparrone, Gioele, Francisco Luque Sánchez, Siham Tabik, Luigi Troiano, Roberto Tagliaferri e Francisco Herrera (mar. de 2020). «Deep learning in video multi-object tracking: A survey». Em: *Neurocomputing* 381, pp. 61–88. ISSN: 0925-2312. DOI: 10.1016/j.neucom.2019.11.023. URL: http://dx.doi.org/10.1016/j.neucom.2019.11.023.

Cui, Yutao, Chenkai Zeng, Xiaoyu Zhao, Yichun Yang, Gangshan Wu e Limin Wang (out. de 2023). «SportsMOT: A Large Multi-Object Tracking Dataset in Multiple Sports Scenes». Em: *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 9921–9931.

Dave, Achal, Tarasha Khurana, Pavel Tokmakov, Cordelia Schmid e Deva Ramanan (2020). *TAO: A Large-Scale Benchmark for Tracking Any Object*.

Dendorfer, Patrick, Aljosa Osep, Anton Milan, Konrad Schindler, Daniel Cremers, Ian Reid, Stefan Roth e Laura Leal-Taixé (2021). «Motchallenge: A benchmark for single-camera multiple target tracking». Em: *International Journal of Computer Vision* 129, pp. 845–881.

Dendorfer, Patrick, Hamid Rezatofighi, Anton Milan, Javen Shi, Daniel Cremers, Ian Reid, Stefan Roth, Konrad Schindler e Laura Leal-Taixé (2020). *MOT20: A benchmark for multi object tracking in crowded scenes*.

Dhillon, Anamika e Gyanendra K Verma (2020). «Convolutional neural network: a review of models, methodologies and applications to object detection». Em: *Progress in Artificial Intelligence* 9.2, pp. 85–112.

Dosovitskiy, Alexey, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit e Neil Houlsby (2020). «An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale». Em: *CoRR* abs/2010.11929. URL: https://arxiv.org/abs/2010.11929.

Du, Dawei, Yuankai Qi, Hongyang Yu, Yifan Yang, Kaiwen Duan, Guorong Li, Weigang Zhang, Qingming Huang e Qi Tian (set. de 2018a). «The Unmanned Aerial Vehicle Benchmark: Object Detection and Tracking». Em: *Proceedings of the European Conference on Computer Vision (ECCV)*.

— (2018b). «The Unmanned Aerial Vehicle Benchmark: Object Detection and Tracking». Em: *CoRR* abs/1804.00518. URL: http://arxiv.org/abs/1804.00518.

Felzenszwalb, Pedro, David Mcallester e Deva Ramanan (jun. de 2008). «A Discriminatively Trained, Multiscale, Deformable Part Model». Em: vol. 8: DOI: 10.1109/CVPR.2008.4587597.

Fragkiadaki, Katerina e Jianbo Shi (2011). «Detection free tracking: Exploiting motion and topology for segmenting and tracking under entanglement». Em: *CVPR 2011*, pp. 2073–2080. DOI: 10.1109/CVPR.2011.5995366.

Gaidon, Adrien, Qiao Wang, Yohann Cabon e Eleonora Vig (2016). «Virtual worlds as proxy for multi-object tracking analysis». Em: *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 4340–4349.

Ganian, Robert, Thekla Hamm e Sebastian Ordyniak (2021). «The complexity of object association in multiple object tracking». Em: *Proceedings of the AAAI Conference on Artificial Intelligence*. Vol. 35. 2, pp. 1388–1396.

Gao, Yuan, Jiang Cao, Ping Wang, Junsong Yin, Ming He, Ming Zhao, Mugen Peng, Su Hu, Yunchuan Sun, Jing Wang, Shaochi Cheng, Yang Guo, Yanchang Du, Yanxi Cai, Jinhui Huang e Kai Qiu (2020). «Intelligent UAV Based Flexible 5G Emergency Networks: Field Trial and System Level Results». Em: *IEEE INFOCOM 2020 - IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS)*, pp. 138–143. DOI: 10.1109/INFOCOMWKSHPS50562.2020.9162724.

Ge, Zheng, Songtao Liu, Feng Wang, Zeming Li e Jian Sun (2021). «Yolox: Exceeding yolo series in 2021». Em: *arXiv preprint arXiv:2107.08430*.

Geiger, Andreas, Philip Lenz e Raquel Urtasun (2012). «Are we ready for autonomous driving? The KITTI vision benchmark suite». Em: *2012 IEEE Conference on Computer Vision and Pattern Recognition*, pp. 3354–3361. DOI: 10.1109/CVPR.2012.6248074.

Gou, Mengran, Srikrishna Karanam, Wenqian Liu, Octavia Camps e Richard J Radke (2017). «Dukemtmc4reid: A large-scale multi-camera person re-identification dataset». Em: *Proceedings of the IEEE conference on computer vision and pattern recognition workshops*, pp. 10–19.

Hatamizadeh, Ali, Yucheng Tang, Vishwesh Nath, Dong Yang, Andriy Myronenko, Bennett Landman, Holger R Roth e Daguang Xu (2022). «Unetr: Transformers for 3d medical image segmentation». Em: *Proceedings of the IEEE/CVF winter conference on applications of computer vision*, pp. 574–584.

Hong, Danfeng, Zhu Han, Jing Yao, Lianru Gao, Bing Zhang, Antonio Plaza e Jocelyn Chanussot (2021). «SpectralFormer: Rethinking hyperspectral image classification with transformers». Em: *IEEE Transactions on Geoscience and Remote Sensing* 60, pp. 1–15.

Huang, Junchao, Xiaoqi He e Sheng Zhao (2023). «The detection and rectification for identity-switch based on unfalsified control». Em: *arXiv preprint arXiv:2307.14591*.

Janakiramaiah, B, Gadupudi Kalyani, Arava Karuna, LV Narasimha Prasad e M Krishna (2023). «Military object detection in defense using multi-level capsule networks». Em: *Soft Computing* 27.2, pp. 1045–1059.

Kalal, Zdenek, Krystian Mikolajczyk e Jiri Matas (2012). «Tracking-Learning-Detection». Em: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 34.7, pp. 1409–1422. DOI: 10.1109/TPAMI.2011.239.

Kim, Chanho, Li Fuxin, Mazen Alotaibi e James M. Rehg (jun. de 2021). «Discriminative Appearance Modeling With Multi-Track Pooling for Real-Time Multi-Object Tracking». Em: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 9553–9562.

LE, BEN (2024). *An Introduction to BYTETrack: Multi-Object Tracking by Associating Every Detection Box*. https://www.datature.io/blog/introduction-to-bytetrack-multi-object-tracking-by-associating-every-detection-box. Accessed: March 30, 2024.

Leal-Taixé, Laura (nov. de 2014). «Multiple object tracking with context awareness». Em.

Leng, Qingming, Mang Ye e Qi Tian (2019). «A survey of open-world person re-identification». Em: *IEEE Transactions on Circuits and Systems for Video Technology* 30.4, pp. 1092–1108.

Li, Wei, Rui Zhao, Tong Xiao e Xiaogang Wang (2014). «Deepreid: Deep filter pairing neural network for person re-identification». Em: *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 152–159.

Li, Xi, Weiming Hu, Chunhua Shen, Zhongfei Zhang, Anthony R. Dick e Anton van den Hengel (2013). «A Survey of Appearance Models in Visual Object Tracking». Em: *CoRR* abs/1303.4803. URL: http://arxiv.org/abs/1303.4803.

Li, Zewen, Wenjie Yang, Shouheng Peng e Fan Liu (2020). *A Survey of Convolutional Neural Networks: Analysis, Applications, and Prospects*.

Lin, Tsung-Yi, Priya Goyal, Ross Girshick, Kaiming He e Piotr Dollár (2017). «Focal Loss for Dense Object Detection». Em: *2017 IEEE International Conference*

on Computer Vision (ICCV), pp. 2999–3007. DOI: 10.1109/ICCV.2017.324.

Lin, Tsung-Yi, Michael Maire, Serge Belongie, Lubomir Bourdev, Ross Girshick, James Hays, Pietro Perona, Deva Ramanan, C. Lawrence Zitnick e Piotr Dollár (2015). *Microsoft COCO: Common Objects in Context*.

Liu, Huanhua, Yonghao Yu, Shengzong Liu e Wei Wang (2022). «A military object detection model of UAV reconnaissance image and feature visualization». Em: *Applied Sciences* 12.23, p. 12236.

Liu, Lihao, Yanqi Cheng, Zhongying Deng, Shujun Wang, Dongdong Chen, Xiaowei Hu, Pietro Liò, Carola-Bibiane Schönlieb e Angelica Aviles-Rivero (2023). *TrafficMOT: A Challenging Dataset for Multi-Object Tracking in Complex Traffic Scenarios*.

Liu, Zelin, Xinggang Wang, Cheng Wang, Wenyu Liu e Xiang Bai (2023). *SparseTrack: Multi-Object Tracking by Performing Scene Decomposition based on Pseudo-Depth*.

Loy, Chen Change, Ke Chen, Shaogang Gong e Tao Xiang (out. de 2013). «Crowd Counting and Profiling: Methodology and Evaluation». Em: vol. 11. ISBN: 978-1-4614-8482-0. DOI: 10.1007/978-1-4614-8483-7_14.

Luo, Wenhan, Junliang Xing, Anton Milan, Xiaoqin Zhang, Wei Liu e Tae-Kyun Kim (2021). «Multiple object tracking: A literature review». Em: *Artificial Intelligence* 293, p. 103448. ISSN: 0004-3702. DOI: https://doi.org/10.1016/j.artint.2020.103448. URL: https://www.sciencedirect.com/science/article/pii/S0004370220301958.

Ma, Bo, Lianghua Huang, Jianbing Shen, Ling Shao, Ming-Hsuan Yang e Fatih Porikli (2016). «Visual tracking under motion blur». Em: *IEEE Transactions on Image Processing* 25.12, pp. 5867–5876.

Ma'sum, M. Anwar, M. Kholid Arrofi, Grafika Jati, Futuhal Arifin, M. Nanda Kurniawan, Petrus Mursanto e Wisnu Jatmiko (2013). «Simulation of intelligent Unmanned Aerial Vehicle (UAV) For military surveillance». Em: *2013 International Conference on Advanced Computer Science and Information Systems (ICACSIS)*, pp. 161–166. DOI: 10.1109/ICACSIS.2013.6761569.

Meinhardt, Tim, Alexander Kirillov, Laura Leal-Taixe e Christoph Feichtenhofer (2022). *TrackFormer: Multi-Object Tracking with Transformers*.

Milan, Anton, Laura Leal-Taixe, Ian Reid, Stefan Roth e Konrad Schindler (2016). *MOT16: A Benchmark for Multi-Object Tracking*.

Minaee, Shervin, Yuri Boykov, Fatih Porikli, Antonio Plaza, Nasser Kehtarnavaz e Demetri Terzopoulos (2021). «Image segmentation using deep learning: A

survey». Em: *IEEE transactions on pattern analysis and machine intelligence* 44.7, pp. 3523–3542.

Mueller, Matthias, Neil Smith e Bernard Ghanem (2016a). «A Benchmark and Simulator for UAV Tracking». Em: *Computer Vision – ECCV 2016*. Ed. por Bastian Leibe, Jiri Matas, Nicu Sebe e Max Welling. Cham: Springer International Publishing, pp. 445–461. ISBN: 978-3-319-46448-0.

— (2016b). «A benchmark and simulator for uav tracking». Em: *Computer Vision–ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11–14, 2016, Proceedings, Part I 14*. Springer, pp. 445–461.

Paucar, Carlos, Lilia Morales, Katherine Pinto, Marcos Sánchez, Rosalba Rodríguez, Marisol Gutierrez e Luis Palacios (2018). «Use of drones for surveillance and reconnaissance of military areas». Em: *Developments and Advances in Defense and Security: Proceedings of the Multidisciplinary International Conference of Research Applied to Defense and Security (MICRADS 2018)*. Springer, pp. 119–132.

Pellegrini, Stefano, Andreas Ess, Konrad Schindler e Luc Van Gool (2009). «You'll never walk alone: Modeling social behavior for multi-target tracking». Em: *2009 IEEE 12th international conference on computer vision*. IEEE, pp. 261–268.

Possegger, Horst, Thomas Mauthner, Peter M Roth e Horst Bischof (2014). «Occlusion geodesics for online multi-object tracking». Em: *proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1306–1313.

Prashanth, Duddela Sai, R Vasanth Kumar Mehta, Kadiyala Ramana e Vidhyacharan Bhaskar (2022). «Handwritten devanagari character recognition using modified lenet and alexnet convolution neural networks». Em: *Wireless Personal Communications* 122.1, pp. 349–378.

Al-Qizwini, Mohammed, Iman Barjasteh, Hothaifa Al-Qassab e Hayder Radha (2017). «Deep learning algorithm for autonomous driving using googlenet». Em: *2017 IEEE intelligent vehicles symposium (IV)*. IEEE, pp. 89–96.

Ranipa, Kalpesh R e Kiritkumar Bhatt (2014). «Illumination condition effect on object tracking: a review». Em: *Global Journal of Computer Science and Technology* 14.5-F, p. 9.

Rawat, Waseem e Zenghui Wang (2017). «Deep convolutional neural networks for image classification: A comprehensive review». Em: *Neural computation* 29.9, pp. 2352–2449.

Redmon, Joseph e Ali Farhadi (abr. de 2018). «YOLOv3: An Incremental Improvement». Em.

Ren, Shaoqing, Kaiming He, Ross B. Girshick e Jian Sun (2015). «Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks». Em: *CoRR* abs/1506.01497. URL: http://arxiv.org/abs/1506.01497.

ResearchGate (s.d.). http://http://https://www.researchgate.net/figure/Several-scenario-examples-of-our-UAVDT-dataset_fig1_337724906. Accessed: 2024-01-08.

Rezaei, Fariba e Mehran Yazdi (2021). «Real-time crowd behavior recognition in surveillance videos based on deep learning methods». Em: *Journal of Real-Time Image Processing* 18, pp. 1669–1679.

Ristani, Ergys, Francesco Solera, Roger Zou, Rita Cucchiara e Carlo Tomasi (2016). «Performance measures and a data set for multi-target, multi-camera tracking». Em: *European conference on computer vision*. Springer, pp. 17–35.

Santos, Nuno Pessanha, Vitor Borges Rodrigues, André Batista Pinto e Bruno Damas (mai. de 2023). «Automatic Detection of Civilian and Military Personnel in Reconnaissance Missions using a UAV». eng. Em: *2023 IEEE International Conference on Autonomous Robot Systems and Competitions (ICARSC)*. IEEE. DOI: 10.1109/ICARSC58346.2023.10129575.

Shi, Shuwei, Jinjin Gu, Liangbin Xie, Xintao Wang, Yujiu Yang e Chao Dong (2022). «Rethinking alignment in video super-resolution transformers». Em: *Advances in Neural Information Processing Systems* 35, pp. 36081–36093.

Sinha, Debjyoti e Mohamed El-Sharkawy (2019). «Thin mobilenet: An enhanced mobilenet architecture». Em: *2019 IEEE 10th annual ubiquitous computing, electronics & mobile communication conference (UEMCON)*. IEEE, pp. 0280–0285.

Specker, Andreas, Daniel Stadler, Lucas Florin e Jurgen Beyerer (jun. de 2021). «An Occlusion-Aware Multi-Target Multi-Camera Tracking System». Em: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, pp. 4173–4182.

Stiefelhagen, Rainer, Keni Bernardin, Rachel Bowers, John Garofolo, Djamel Mostefa e Padmanabhan Soundararajan (abr. de 2006). «The CLEAR 2006 evaluation». Em: vol. 4122, pp. 1–44. ISBN: 978-3-540-69567-7. DOI: 10.1007/978-3-540-69568-4_1.

Sun, Peize, Jinkun Cao, Yi Jiang, Zehuan Yuan, Song Bai, Kris Kitani e Ping Luo (2022). *DanceTrack: Multi-Object Tracking in Uniform Appearance and Diverse Motion*.

Sun, Peize, Jinkun Cao, Yi Jiang, Rufeng Zhang, Enze Xie, Zehuan Yuan, Changhu Wang e Ping Luo (2021). *TransTrack: Multiple Object Tracking with Transformer*.

Szegedy, Christian, Sergey Ioffe, Vincent Vanhoucke e Alexander Alemi (2017). «Inception-v4, inception-resnet and the impact of residual connections on learning». Em: *Proceedings of the AAAI conference on artificial intelligence*. Vol. 31. 1.

Tan, Mingxing e Quoc Le (2019). «Efficientnet: Rethinking model scaling for convolutional neural networks». Em: *International conference on machine learning*. PMLR, pp. 6105–6114.

Tang, Guangyi, Jianjun Ni, Yonghao Zhao, Yang Gu e Weidong Cao (2024). «A Survey of Object Detection for UAVs Based on Deep Learning». Em: *Remote Sensing* 16.1. ISSN: 2072-4292. DOI: 10.3390/rs16010149. URL: https://www.mdpi.com/2072-4292/16/1/149.

Udeanu, Gheorghe, Alexandra Dobrescu e Mihaela Oltean (2016). «Unmanned aerial vehicle in military operations». Em: *Sci. Res. Educ. Air Force* 18.1, pp. 199–206.

Vaswani, Ashish, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser e Illia Polosukhin (2017). «Attention Is All You Need». Em: *CoRR* abs/1706.03762. URL: http://arxiv.org/abs/1706.03762.

Wang, Yu-Hsiang, Jun-Wei Hsieh, Ping-Yang Chen, Ming-Ching Chang, Hung Hin So e Xin Li (2024). *SMILEtrack: SiMIlarity LEarning for Occlusion-Aware Multiple Object Tracking*.

Wang, Jingbo, Kaiwen Zhou, Wenbin Xing, Huanhuan Li e Zaili Yang (2023). «Applications, Evolutions, and Challenges of Drones in Maritime Transport». Em: *Journal of Marine Science and Engineering* 11.11. ISSN: 2077-1312. DOI: 10.3390/jmse11112056. URL: https://www.mdpi.com/2077-1312/11/11/2056.

Wang, Limin, Sheng Guo, Weilin Huang e Yu Qiao (2015). «Places205-vggnet models for scene recognition». Em: *arXiv preprint arXiv:1508.01667*.

Wang, Zhongdao, Hengshuang Zhao, Ya-Li Li, Shengjin Wang, Philip Torr e Luca Bertinetto (jul. de 2021a). *Do Different Tracking Tasks Require Different Appearance Models?*

— (2021b). «Do different tracking tasks require different appearance models?» Em: *Thirty-Fifth Conference on Neural Infromation Processing Systems*.

Welch, Greg, Gary Bishop et al. (1995a). «An introduction to the Kalman filter». Em.

— (1995b). «An introduction to the Kalman filter». Em.

Wen, Longyin et al. (2019). «VisDrone-MOT2019: The Vision Meets Drone Multiple Object Tracking Challenge Results». Em: *2019 IEEE/CVF International Conference on Computer Vision Workshop (ICCVW)*, pp. 189–198. DOI: 10.1109/ICCVW.2019.00028.

Wojke, Nicolai, Alex Bewley e Dietrich Paulus (2017). «Simple Online and Realtime Tracking with a Deep Association Metric». Em: *CoRR* abs/1703.07402. URL: http://arxiv.org/abs/1703.07402.

Wu, Bo e Ram Nevatia (2006). «Tracking of multiple, partially occluded humans based on static body part detection». Em: *2006 IEEE computer society conference on computer vision and pattern recognition (CVPR'06)*. Vol. 1. IEEE, pp. 951–958.

Wu, Ying, Ting Yu e Gang Hua (2003). «Tracking appearances with occlusions». Em: *2003 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2003. Proceedings.* Vol. 1, pp. I–I. DOI: 10.1109/CVPR.2003.1211433.

Xu, Peng, Xiatian Zhu e David A Clifton (2023). «Multimodal learning with transformers: A survey». Em: *IEEE Transactions on Pattern Analysis and Machine Intelligence*.

Yan, Bin, Houwen Peng, Kan Wu, Dong Wang, Jianlong Fu e Huchuan Lu (2021). *LightTrack: Finding Lightweight Neural Networks for Object Tracking via One-Shot Architecture Search*.

Yang, C-H Huck, Mohit Chhabra, Y-C Liu, Quan Kong, Tomoaki Yoshinaga e Tomokazu Murakami (2021). «Robust Unsupervised Multi-Object Tracking In Noisy Environments». Em: *2021 IEEE International Conference on Image Processing (ICIP)*. IEEE, pp. 2239–2243.

Yang, Fan, Wongun Choi e Yuanqing Lin (jun. de 2016). «Exploit All the Layers: Fast and Accurate CNN Object Detector with Scale Dependent Pooling and Cascaded Rejection Classifiers». Em: pp. 2129–2137. DOI: 10.1109/CVPR.2016.234.

Yang, Ming, Ting Yu e Ying Wu (nov. de 2007). «Game-Theoretic Multiple Target Tracking». Em: pp. 1–8. ISBN: 978-1-4244-1631-8. DOI: 10.1109/ICCV.2007.4408942.

Yao, Huang, Rongjun Qin e Xiaoyu Chen (2019). «Unmanned Aerial Vehicle for Remote Sensing Applications—A Review». Em: *Remote Sensing* 11.12. ISSN:

2072-4292. DOI: 10.3390/rs11121443. URL: https://www.mdpi.com/2072-4292/11/12/1443.

Ye, Mang, Jianbing Shen, Gaojie Lin, Tao Xiang, Ling Shao e Steven CH Hoi (2021). «Deep learning for person re-identification: A survey and outlook». Em: *IEEE transactions on pattern analysis and machine intelligence* 44.6, pp. 2872–2893.

Yoon, Ju Hong, Ming-Hsuan Yang, Jongwoo Lim e Kuk-Jin Yoon (2015). «Bayesian Multi-object Tracking Using Motion Context from Multiple Objects». Em: *2015 IEEE Winter Conference on Applications of Computer Vision*, pp. 33–40. DOI: 10.1109/WACV.2015.12.

Yu, Cunjun, Xiao Ma, Jiawei Ren, Haiyu Zhao e Shuai Yi (2020). «Spatio-Temporal Graph Transformer Networks for Pedestrian Trajectory Prediction». Em: *CoRR* abs/2005.08514. URL: https://arxiv.org/abs/2005.08514.

Zeng, Fangao, Bin Dong, Tiancai Wang, Xiangyu Zhang e Yichen Wei (2021). «End-to-end multiple-object tracking with transformer». Em: *arXiv preprint arXiv:2105.03247* 2.3, p. 7.

Zhang, Lu e Laurens van der Maaten (abr. de 2014). «Preserving Structure in Model-Free Tracking». Em: *Pattern Analysis and Machine Intelligence, IEEE Transactions on* 36, pp. 756–769. DOI: 10.1109/TPAMI.2013.221.

Zhang, Yifu, Peize Sun, Yi Jiang, Dongdong Yu, Zehuan Yuan, Ping Luo, Wenyu Liu e Xinggang Wang (2021). «ByteTrack: Multi-Object Tracking by Associating Every Detection Box». Em: *CoRR* abs/2110.06864. URL: https://arxiv.org/abs/2110.06864.

Zhang, Yifu, Chunyu Wang, Xinggang Wang, Wenjun Zeng e Wenyu Liu (2020). «A Simple Baseline for Multi-Object Tracking». Em: *CoRR* abs/2004.01888. URL: https://arxiv.org/abs/2004.01888.

Zhao, Ranyang, Xinyan Zhang e Jianwei Zhang (2024). «PSMOT: Online Occlusion-Aware Multi-Object Tracking Exploiting Position Sensitivity». Em: *Sensors* 24.4. ISSN: 1424-8220. DOI: 10.3390/s24041199. URL: https://www.mdpi.com/1424-8220/24/4/1199.

Zheng, Liang, Liyue Shen, Lu Tian, Shengjin Wang, Jingdong Wang e Qi Tian (2015). «Scalable person re-identification: A benchmark». Em: *Proceedings of the IEEE international conference on computer vision*, pp. 1116–1124.

Zheng, Wei-Shi, Shaogang Gong e Tao Xiang (jun. de 2012). «Transfer re-identification: From person to set-based verification». Em: pp. 2650–2657. ISBN: 978-1-4673-1226-4. DOI: 10.1109/CVPR.2012.6247985.

Zhou, Xingyi, Dequan Wang e Philipp Krähenbühl (2019). «Objects as Points». Em: *CoRR* abs/1904.07850. URL: http://arxiv.org/abs/1904.07850.

Zhu, Pengfei, Longyin Wen, Dawei Du, Xiao Bian, Heng Fan, Qinghua Hu e Haibin Ling (2021). *Detection and Tracking Meet Drones Challenge*.

Zhu, Xizhou, Weijie Su, Lewei Lu, Bin Li, Xiaogang Wang e Jifeng Dai (2020). «Deformable DETR: Deformable Transformers for End-to-End Object Detection». Em: *CoRR* abs/2010.04159. URL: https://arxiv.org/abs/2010.04159.