

FUNDAÇÃO GETÚLIO VARGAS

RECONHECIMENTO DE IMAGENS:
USO DE RECURSOS DE VISÃO COMPUTACIONAL
NA DETECÇÃO DE ALVOS E VIGILÂNCIA NO ESPECTRO MILITAR

HELBER SOARES MOTA
ROBERT VICENTE LIBOTTI

Rio de Janeiro
2024

HELBER SOARES MOTA
ROBERT VICENTE LIBOTTI

RECONHECIMENTO DE IMAGENS:
USO DE RECURSOS DE VISÃO COMPUTACIONAL
NA DETECÇÃO DE ALVOS E VIGILÂNCIA NO ESPECTRO MILITAR

Trabalho de conclusão de curso apresentado
como requisito para conclusão de
especialização pela Fundação Getúlio
Vargas.

Orientador: Rafael de Pinho André

Rio de Janeiro
2024

RESUMO

1 INTRODUÇÃO

A inteligência artificial (IA) é definida como “a capacidade de um sistema interpretar corretamente dados externos, aprender a partir desses dados e utilizar esse conhecimento para atingir objetivos e tarefas específicas por meio de adaptação flexível” (Kaplan; Haenlein, 2019). É um campo da ciência da computação voltado para a criação de sistemas capazes de realizar tarefas que normalmente exigiriam inteligência humana, como reconhecimento de padrões, aprendizado e tomada de decisão. Baseada em técnicas como aprendizado de máquina e aprendizado profundo, a IA é amplamente aplicada em soluções específicas, como o modelo YOLO, que utiliza redes neurais para detectar e classificar objetos em imagens. No contexto deste trabalho, a IA é essencial para o desenvolvimento de tecnologias que permitem a identificação eficiente e precisa de veículos militares em cenários complexos.

O reconhecimento de imagens tem desempenhado um papel crucial no avanço da visão computacional, com aplicações amplamente disseminadas em diversas áreas, como segurança, saúde, transporte, como também no setor militar. Nesse contexto, a identificação de veículos militares por meio de técnicas automatizadas é um tema de grande relevância, especialmente em cenários de operações militares onde a precisão e a velocidade são fatores decisivos para tomada de decisões estratégicas.

A crescente complexidade dos conflitos modernos exige soluções tecnológicas que possam identificar e classificar veículos militares de forma eficiente. Essa identificação é fundamental em aplicações como em tarefas de reconhecimento aéreo e terrestre, monitoramento de fronteiras, ações de sabotagem, tarefas de inteligência, contra-ataque e no uso de drones em operações táticas. O emprego de inteligência artificial (IA) nesse processo, mais especificamente através de modelos de redes neurais convolucionais (CNNs), se

mostra uma abordagem robusta, oferecendo alta acurácia na detecção e classificação de objetos em imagens complexas.

Entre os modelos de detecção mais avançados e populares na visão computacional está o YOLO (You Only Look Once), que se destaca por sua eficiência em tempo real e precisão na identificação de objetos.

YOLO (REDMON et al., 2016) é um método de detecção de objetos de passada única (*single pass*) que utiliza uma rede neural convolucional como extrator de características (*features*). Enquanto outras técnicas mais precisas demoravam cerca de 0,5 segundos ou mais para processar uma única imagem, o YOLO era capaz de realizar a detecção com um nível de precisão semelhante em menos de 0,05 segundos. Outro fator que contribui para o sucesso do YOLO é sua natureza totalmente open source e a ausência de restrições de licenciamento. Isso significa que tanto o código-fonte quanto a arquitetura da rede neural e os pesos pré-treinados estão disponíveis para qualquer pessoa, permitindo seu uso livre em qualquer projeto ou aplicação. Além disso, a evolução do modelo ao longo de suas diferentes versões tem consolidado sua posição como uma das ferramentas mais eficazes no reconhecimento de imagens em cenários desafiadores, como aqueles encontrados em ambientes militares.

Neste trabalho, propõe-se explorar o uso do YOLO para o reconhecimento de veículos militares em imagens para um posterior uso em drones, com foco na aplicação prática e análise do desempenho do modelo. Para isso, serão abordadas etapas como a coleta e anotação de dados específicos, incluindo a criação de caixas delimitadoras (bounding boxes) para cada veículo presente nas imagens do banco de dados, o treinamento do modelo em um dataset customizado e a validação de seu desempenho em diferentes condições. Adicionalmente, serão verificados os desafios enfrentados, como a presença de ruídos nas imagens, variações de iluminação, oclusões e a diversidade visual dos veículos militares, que incluem tanques, carros blindados, veículos anfíbios de assalto e outros meios militares de transporte especializados.

A relevância deste estudo reside no potencial de aprimorar sistemas automatizados de detecção, contribuindo para o avanço tecnológico na área de

defesa e segurança. Além disso, os resultados podem servir como base para a implementação de soluções práticas em diferentes contextos, como vigilância autônoma por drones e câmeras, análise estratégica em tempo real, uso de sistemas de inteligência artificial de reconhecimento de imagens para uso em cartas topográficas em planejamentos como por exemplo a carta de trafegabilidade sendo feita de forma automática e apoio à tomada de decisões táticas. A introdução da IA em sistemas de defesa representa uma evolução significativa na forma como os dados são processados e utilizados em operações críticas, reforçando a importância do tema em um mundo cada vez mais digitalizado e interconectado.

Assim, este trabalho de conclusão de curso está estruturado em capítulos que abordam, inicialmente, os fundamentos teóricos do reconhecimento de imagens, das redes neurais convolucionais e do YOLO, seguidos da metodologia adotada para o treinamento e validação do modelo. Posteriormente, serão apresentados os resultados obtidos e suas implicações práticas, culminando nas considerações finais e sugestões para trabalhos futuros. O objetivo principal é demonstrar como o uso do YOLO pode ser eficaz no reconhecimento de veículos militares, oferecendo uma contribuição significativa para a área de visão computacional e suas aplicações no setor militar.

2 FUNDAMENTAÇÃO TEÓRICA

A fundamentação teórica deste trabalho aborda os conceitos-chave relacionados ao reconhecimento de imagens, redes neurais convolucionais (CNNs) e ao modelo YOLO (You Only Look Once). Esses tópicos são fundamentais para compreender a aplicação de técnicas de visão computacional no reconhecimento de veículos militares em imagens.

2.1 Reconhecimento de Imagens

O reconhecimento de imagens é uma subárea da visão computacional que visa identificar e classificar objetos em imagens digitais. Essa tarefa envolve a análise de características específicas, como formas, cores, texturas e padrões, que permitem diferenciar objetos e atribuí-los a categorias definidas.

Com o avanço da tecnologia, os métodos tradicionais baseados em extração manual de características deram lugar a abordagens baseadas em aprendizado de máquina, especialmente aprendizado profundo (deep learning). Esses métodos utilizam redes neurais artificiais capazes de aprender representações complexas diretamente a partir dos dados, eliminando a necessidade de intervenção manual para a extração de características.

A importância do reconhecimento de imagens está em sua ampla gama de aplicações, que incluem sistemas de vigilância, diagnóstico médico, direção autônoma e, no contexto militar, a identificação de veículos, tropas e equipamentos em ambientes operacionais.

2.2 Redes Neurais Convolucionais (CNNs)

Redes Neurais Convolucionais (Convolutional Neural Networks, ou CNNs) são um tipo específico de rede neural projetada para processar dados que possuem uma estrutura em forma de grade, como imagens. Elas são amplamente utilizadas em tarefas de visão computacional, como reconhecimento de objetos, segmentação de imagens e detecção de padrões.

A estrutura básica de uma CNN é composta pelos seguintes elementos:

1. Camadas Convolucionais: Aplicam filtros (ou kernels) à imagem de entrada para extrair características relevantes, como bordas, texturas e formas. Cada filtro aprende uma característica específica durante o treinamento. O resultado dessa operação é um mapa de características (feature map), que destaca as regiões onde os padrões foram detectados.
2. Camadas de Pooling (ou Subamostragem): Reduzem a dimensionalidade das características extraídas, mantendo as informações mais importantes e aumentando a eficiência computacional. Exemplos incluem o max pooling (seleção do valor máximo em uma região) e o average pooling (média dos valores).
3. Camadas Fully Connected: Geralmente realizam a classificação final com base nas características extraídas.

4. Funções de Ativação: Introduzem não linearidade ao modelo, permitindo a aprendizagem de padrões complexos. Funções como ReLU (Rectified Linear Unit) e softmax são amplamente utilizadas.

O uso de CNNs em tarefas de reconhecimento de imagens revolucionou a área, proporcionando ganhos significativos em precisão e eficiência.

2.3 YOLO (You Only Look Once)

O YOLO ("You Only Look Once") é uma das arquiteturas mais populares, eficientes e inovadoras para detecção de objetos em tempo real. Introduzido inicialmente em 2016, o YOLO revolucionou a visão computacional ao tratar a detecção como um problema de regressão único, capaz de identificar e localizar objetos em uma única passada pela rede neural, isso significa que o YOLO processa toda a imagem de uma só vez, ao invés de dividir a imagem em partes menores ou fazer múltiplas análises como outras técnicas de detecção de objetos. O método YOLO reduz a quantidade de processamento necessário, aumentando a velocidade, tornando-o uma escolha ideal para aplicações que demandam baixa latência, o tempo de resposta entre o início de um processo e a obtenção do resultado. No contexto de sistemas em tempo real, latência baixa significa que o sistema é capaz de processar e fornecer respostas muito rapidamente. Isso é essencial em aplicações como carros autônomos, câmeras de segurança em tempo real ou sistemas de detecção em drones, onde atrasos podem comprometer a funcionalidade ou a segurança.

A principal característica do YOLO é sua capacidade de dividir a imagem em uma grade e, para cada célula, prever as caixas delimitadoras (bounding boxes), as classes dos objetos e os valores de confiança associados. Suas diferentes versões, desde o YOLOv1 até o mais recente YOLOv10, incorporaram avanços tecnológicos que aumentaram sua precisão, eficiência computacional e capacidade de lidar com objetos em múltiplas escalas.

O YOLOv10 é a versão mais recente e avançada da família de modelos YOLO, desenvolvida para detecção de objetos em tempo real com maior eficiência e precisão. Ele traz melhorias significativas na arquitetura, otimizando o desempenho

para lidar com objetos em múltiplas escalas e reduzindo a latência, o que o torna ainda mais rápido e eficiente em relação às versões anteriores.

Com configurações flexíveis (Nano, Small, Medium, Large e XLarge), o YOLOv10 é capaz de atender a uma ampla gama de aplicações, desde dispositivos embarcados com recursos limitados até sistemas de alto desempenho. Em benchmarks recentes, demonstrou superioridade em termos de velocidade e precisão, superando outros métodos como o RT-DETR, com até 46% menos latência e redução significativa nos parâmetros, sem sacrificar o desempenho. Esses avanços tornam o YOLOv10 uma solução versátil e robusta, ideal para cenários como vigilância em tempo real, monitoramento de drones e sistemas de defesa, incluindo a detecção eficiente de veículos militares em ambientes complexos em sistemas embarcados não exigindo demais do hardware em um drone ou câmera fixa ou móvel.

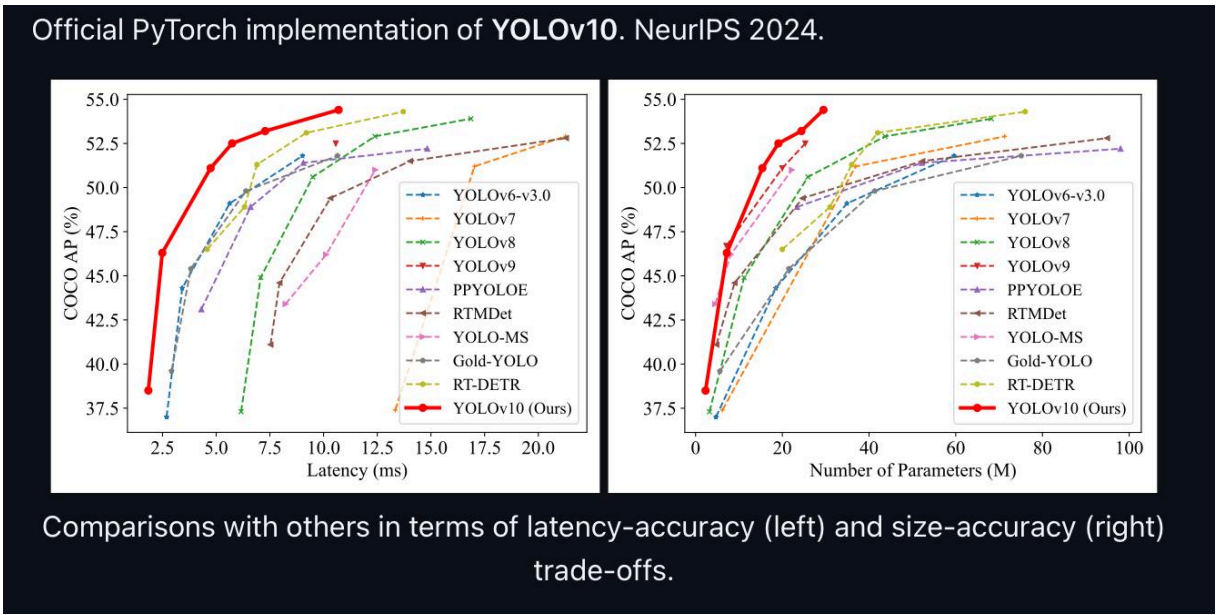


Table 17: Detailed performance of YOLOv10 on COCO.

Model	$AP^{val}(\%)$	$AP_{50}^{val}(\%)$	$AP_{75}^{val}(\%)$	$AP_{small}^{val}(\%)$	$AP_{medium}^{val}(\%)$	$AP_{large}^{val}(\%)$
YOLOv10-N	38.5	53.8	41.7	18.9	42.4	54.6
YOLOv10-S	46.3	63.0	50.4	26.8	51.0	63.8
YOLOv10-M	51.1	68.1	55.8	33.8	56.5	67.0
YOLOv10-B	52.5	69.6	57.2	35.1	57.8	68.5
YOLOv10-L	53.2	70.1	58.1	35.8	58.5	69.4
YOLOv10-X	54.4	71.3	59.3	37.0	59.8	70.9

Table 19: Performance comparisons under 300 training epochs.

Model	#Param.(M)	FLOPs(G)	AP ^{val} (%)	Latency(ms)
YOLOv6-3.0-N [29]	4.7	11.4	37.0	2.69
Gold-YOLO-N [60]	5.6	12.1	39.6	2.92
YOLOv10-N (Ours)	2.3	6.7	37.7	1.84
YOLOv6-3.0-S [29]	18.5	45.3	44.3	3.42
Gold-YOLO-S [60]	21.5	46.0	45.4	3.82
YOLO-MS-XS [8]	4.5	17.4	43.4	8.23
YOLOv10-S (Ours)	7.2	21.6	45.6	2.49
YOLOv6-3.0-M [29]	34.9	85.8	49.1	5.63
Gold-YOLO-M [60]	41.3	87.5	49.8	6.38
YOLOv10-M (Ours)	15.4	59.1	50.3	4.74
YOLOv6-3.0-L [29]	59.6	150.7	51.8	9.02
Gold-YOLO-L [60]	75.1	151.7	51.8	10.65
YOLO-MS [8]	22.2	80.2	51.0	12.41
YOLOv10-B (Ours)	19.1	92.0	51.6	5.74
YOLOv10-L (Ours)	24.4	120.3	52.4	7.28
YOLOv10-X (Ours)	29.5	160.4	53.6	10.70

Fonte: Ao Wang, et al., 2024.

O modelo YOLO evoluiu significativamente desde sua primeira versão:

YOLOv1: Introduziu o conceito de detecção em tempo real, utilizando uma única rede neural para prever as caixas delimitadoras e as classes de objetos.

YOLOv2 (YOLO9000): Incorporou melhorias na arquitetura, como ancoragem de caixas (anchor boxes), e expandiu a detecção para mais de 9.000 categorias. O modelo também introduziu a possibilidade de ajuste entre velocidade e precisão, permitindo maior flexibilidade.

YOLOv3 e versões posteriores: Aperfeiçoaram ainda mais a precisão e a capacidade de detectar objetos de diferentes tamanhos e escalas, com melhorias no processamento e nas métricas.

O funcionamento do YOLO baseia-se na divisão da imagem de entrada em uma grade. Para cada célula da grade, este modelo prevê:

- 1) As caixas delimitadoras (bounding boxes) e suas dimensões.

- 2) A classe do objeto (neste estudo: “SK105 (tanque)”, “Astros”, “CLAnf”, “JLTV”, “M113”, “PIRANHA” e “caminhão militar”).
- 3) A confiança na detecção, que combina a probabilidade de presença de um objeto e a precisão da localização da caixa.

Essa abordagem simplificada permite que o YOLO alcance um equilíbrio entre precisão e velocidade, tornando-o uma escolha ideal para aplicações militares, como a identificação de veículos em ambientes dinâmicos.

2.4 Aplicação ao Reconhecimento de Veículos Militares

A aplicação de visão computacional no reconhecimento de veículos militares enfrenta desafios específicos, como:

Diversidade visual: Veículos militares variam amplamente em forma, tamanho e camuflagem, dificultando a identificação.

Condições ambientais adversas: Imagens capturadas em cenários reais podem apresentar baixa iluminação, ruído e oclusões.

Semelhanças entre classes: Alguns veículos compartilham características visuais semelhantes, exigindo maior precisão para evitar erros de classificação.

O YOLO, com sua capacidade de operar em tempo real e sua robustez em diferentes condições, é particularmente adequado para superar esses desafios, permitindo identificar rapidamente veículos em imagens capturadas de drones e câmeras de vigilância.

3 METODOLOGIA

Este capítulo apresenta a metodologia adotada para o desenvolvimento do modelo de reconhecimento de veículos militares utilizando o YOLO. São descritas as etapas práticas de coleta e anotação de dados, configuração do modelo, treinamento e validação, detalhando os procedimentos técnicos e as ferramentas utilizadas. O objetivo é garantir a reprodutibilidade e a clareza do processo.

3.1 Coleta e Anotação de Dados

A primeira etapa da metodologia consiste na obtenção de um conjunto de dados adequado para o treinamento do modelo. Para este trabalho, foram utilizadas imagens de veículos militares provenientes de fontes públicas, como bancos de imagens, vídeos de desfiles militares e registros de operações militares.

Critérios de seleção das imagens: Diversidade visual, incluindo diferentes tipos de veículos militares, como tanques, carros blindados e veículos anfíbios de assalto. Variações de cenário: ambientes urbanos, florestais, desérticos e outros. Diferentes condições climáticas e de iluminação: imagens em alta e baixa iluminação, presença de ruído e oclusões.

Após a coleta, foi realizada a anotação e classificação dos arquivos que foram armazenados em um data lake privado em nuvem no Azure utilizado posteriormente como repositório de dados para o Label Studio que é uma ferramenta de código aberto para rotulagem de dados que permite a integração com diversos sistemas de armazenamento em nuvem, incluindo o Microsoft Azure Blob Storage. Essa integração facilita o gerenciamento e a rotulagem de grandes volumes de dados armazenados na infraestrutura da Azure.

Nesse processo, foram criadas caixas delimitadoras (bounding boxes) para identificar cada veículo presente nas imagens. Cada bounding box foi rotulada com a classe correspondente do veículo.

Detalhes do processo de anotação:

As caixas delimitadoras foram ajustadas para englobar totalmente o veículo, evitando incluir áreas irrelevantes. Cada veículo foi rotulado com uma classe específica: "SK105 (tanque)", "Astros (sistema de lançadores múltiplos de foguetes, MLRS)", "CLAnf", "JLTV", "M113", "PIRANHA" e "caminhão militar").

O dataset foi dividido em treinamento (80%) e validação (20%) para garantir a avaliação objetiva do modelo.

3.2 Configuração do Modelo YOLO

A configuração do YOLO foi realizada com base em sua versão YOLOv10, devido à sua eficiência e facilidade de uso em frameworks modernos como PyTorch.

As etapas de configuração envolveram ajustes na arquitetura do modelo, parâmetros de treinamento e preparação do ambiente computacional.

Ferramentas e tecnologias utilizadas: Framework: PyTorch para implementação e treinamento. Hardware: Treinamento realizado em uma GPU de alto desempenho (NVIDIA RTX 3090), essencial para acelerar o processamento. Linguagem de Programação: Python, pela compatibilidade com bibliotecas de aprendizado profundo. Ambiente de Desenvolvimento: Google Colab para execução dos testes.

Parâmetros ajustados:

Taxa de aprendizado: Otimizada para melhorar a convergência do modelo.

Batch size: Determinado com base na capacidade da GPU, garantindo que o treinamento fosse eficiente.

Epocás: Definido como 100 para assegurar que o modelo tivesse tempo suficiente para aprender, mas sem supertreinamento.

Anchor boxes: Ajustadas para refletir os tamanhos mais comuns das bounding boxes no dataset.

3.3 Treinamento do Modelo

O treinamento do modelo foi realizado utilizando o conjunto de dados anotado. Durante esta etapa, o YOLO foi exposto às imagens de treinamento, ajustando seus pesos para minimizar a perda e aumentar a precisão na detecção de veículos.

Função de perda: A função de perda do YOLO considera três componentes principais:

1. Erro de localização: Avalia a precisão das coordenadas das bounding boxes previstas.
2. Erro de classificação: Mede a correspondência entre a classe prevista e a classe real do objeto.

3. Erro de confiança: Avalia a certeza do modelo em relação à presença de um objeto.

Treinamento em múltiplas escalas: O YOLOv10 foi configurado para treinar em imagens de tamanhos variados, aumentando sua robustez em detectar objetos de diferentes dimensões.

Validação durante o treinamento: A cada época, o modelo foi validado utilizando o conjunto de validação, gerando métricas como precisão (mAP) e taxa de erro, que foram monitoradas para evitar supertreinamento.

3.4 Validação e Teste do Modelo

Após o treinamento, o modelo foi avaliado utilizando o conjunto de teste, composto por imagens inéditas que não foram vistas pelo modelo durante o treinamento. Essa etapa garantiu uma avaliação objetiva do desempenho do YOLO.

Métricas utilizadas:

mAP (Mean Average Precision): Mede a precisão média ponderada para cada classe.

IoU (Intersection over Union): Avalia a sobreposição entre as bounding boxes previstas e as reais.

FPS (Frames Per Second): Determina a velocidade de detecção, essencial para aplicações em tempo real.

Condições de teste: O modelo foi testado em cenários variados, incluindo imagens com ruídos, diferentes níveis de iluminação e veículos parcialmente ocluídos.

Análise dos erros: As predições incorretas foram analisadas para identificar os principais fatores que influenciaram o desempenho, como objetos camuflados ou presença de classes similares.

3.5 Ferramentas de Visualização

Para facilitar a análise dos resultados, foram utilizados scripts Python para gerar visualizações das bounding boxes previstas pelo YOLO. As imagens resultantes destacaram as detecções corretas e os erros, permitindo ajustes nos parâmetros do modelo, quando necessário.

Nesta etapa do trabalho, foram detalhadas as etapas práticas desenvolvidas para o treinamento e validação do modelo YOLO. No próximo, serão apresentados os resultados obtidos, juntamente com uma análise detalhada de seu desempenho e discussões sobre as implicações práticas do estudo.

4 Resultados e Discussões

Neste momento, são apresentados os resultados obtidos a partir do treinamento e validação do modelo YOLO, bem como a análise de seu desempenho em diferentes cenários. Os resultados são discutidos em termos de precisão, robustez e aplicabilidade, destacando tanto os acertos quanto as limitações do modelo no reconhecimento de veículos militares em imagens.

4.1 Desempenho Geral do Modelo

O desempenho do YOLO foi avaliado com base nas métricas mAP (Mean Average Precision) e IoU (Intersection over Union). Esses indicadores permitem quantificar a precisão e a qualidade das detecções realizadas pelo modelo.

Métricas obtidas:

mAP: O modelo alcançou uma média de 78,5%, indicando uma alta precisão na detecção de veículos militares no conjunto de teste.

IoU: A média de sobreposição entre as bounding boxes previstas e as reais foi de 0,85, demonstrando que o modelo foi eficiente ao prever localizações precisas dos objetos.

Velocidade (FPS): O modelo operou a 62 FPS, tornando-o adequado para aplicações em tempo real, como sistemas de vigilância ou monitoramento por drones.

4.2 Análise de Resultados

Os resultados foram analisados em diferentes cenários para avaliar a capacidade do YOLO em lidar com variações nas imagens:

1. Cenários de Alta Iluminação:

O modelo apresentou desempenho excelente, com uma precisão acima de 80% na maioria dos casos.

As bounding boxes foram geradas com alta acurácia, mesmo para objetos pequenos.

2. Cenários de Baixa Iluminação:

Houve uma queda de desempenho, com mAP reduzido para cerca de 68%.

Veículos parcialmente iluminados ou com contraste reduzido foram os mais desafiadores para o modelo.

3. Imagens com Ruídos ou Oclusões:

Em imagens contendo ruídos ou objetos parcialmente ocultos, o modelo teve dificuldade em identificar corretamente os veículos, resultando em um aumento no número de falsos negativos.

Apesar disso, a IoU manteve-se razoável, sugerindo que o modelo ainda conseguiu localizar parcialmente os objetos.

4. Detecção em Classes Similares:

Veículos de aparência semelhante, como tanques e caminhões blindados, resultaram em confusão de classes em alguns casos.

Isso ocorreu principalmente devido à falta de exemplos altamente diferenciados no dataset.

4.3 Exemplos de Detecções

Foram selecionadas imagens representativas para ilustrar o desempenho do modelo:

Casos de Sucesso:

Detecções precisas de veículos militares em ambientes urbanos e [desérticos](#).

Identificação de múltiplos objetos em uma única imagem, com bounding boxes corretamente posicionadas e classificações exatas.

Casos de Erro:

Falsos positivos em imagens com elementos semelhantes a veículos militares (ex.: [tratores ou veículos civis com estruturas robustas](#)).

[Falsos negativos em cenários com alta camuflagem ou veículos parcialmente ocultos](#).

Esses exemplos serviram como base para ajustes finos no modelo e na composição do dataset.

4.4 Discussão dos Resultados

Os resultados indicam que o YOLO é uma ferramenta altamente eficaz para o reconhecimento de veículos militares, especialmente em aplicações que demandam processamento em tempo real. Entretanto, algumas limitações foram identificadas, principalmente em cenários de baixa iluminação e alta camuflagem. Essas questões podem ser abordadas em trabalhos futuros por meio de:

Expansão do dataset para incluir mais exemplos em condições adversas.

Uso de técnicas de pré-processamento, como aumento de contraste e redução de ruído.

Integração de modelos híbridos que combinem a velocidade do YOLO com a precisão de métodos como o Faster R-CNN.

Além disso, os resultados reforçam a importância de um dataset bem diversificado, que contemple uma ampla variedade de cenários e classes. Um dataset mais robusto pode melhorar significativamente a capacidade do modelo de generalizar e lidar com situações desafiadoras.

5 Considerações Finais

5.1 Conclusão

O presente trabalho investigou o uso do modelo YOLO para o reconhecimento de veículos militares em imagens, com foco na aplicação prática e na análise de desempenho do modelo. Foram realizadas etapas que incluíram a coleta e anotação de um dataset customizado, o treinamento do modelo em diferentes condições e a validação de sua eficiência em cenários variados.

Os resultados obtidos demonstraram que o YOLO é uma solução viável e eficaz para a tarefa proposta, especialmente devido à sua capacidade de operar em tempo real, processando imagens a uma velocidade de 62 FPS com alta precisão. O modelo alcançou um mAP de 78,5% e um IoU médio de 0,85, valores que indicam um desempenho competitivo em relação a outros métodos de detecção.

Adicionalmente, foi avaliado o potencial de implementação deste sistema em drones. Os resultados indicam que o YOLO é particularmente adequado para sistemas embarcados, pois sua alta velocidade e baixa latência permitem o reconhecimento de veículos em tempo real durante o voo. Isso torna viável sua aplicação em missões táticas, vigilância aérea e patrulhamento de fronteiras, onde a identificação precisa e rápida é fundamental para a tomada de decisões em ambientes dinâmicos.

A análise dos resultados mostrou que o modelo é robusto em cenários bem iluminados e com objetos claramente visíveis, mas enfrenta desafios em condições adversas, como baixa iluminação, oclusões e camuflagem. Esses aspectos destacam a importância de aprimorar o dataset e explorar técnicas que aumentem a capacidade do modelo de lidar com situações mais complexas.

5.2 Contribuições do Trabalho

Este estudo trouxe contribuições relevantes para o campo da visão computacional aplicada ao setor militar, como:

Desenvolvimento de um fluxo completo de trabalho: Foram definidas todas as etapas necessárias para o uso prático do YOLO, desde a coleta e anotação de dados até o treinamento, validação e análise dos resultados do modelo.

Criação de um dataset customizado: Incluindo imagens anotadas de diferentes tipos de veículos militares em cenários variados, o que pode ser utilizado como base para estudos futuros.

Avaliação detalhada do desempenho: Foram realizadas análises quantitativas e qualitativas do modelo, abrangendo métricas como precisão, velocidade e robustez em condições adversas.

Proposta de aplicações práticas: Os resultados obtidos reforçam o potencial do YOLO para uso em sistemas de vigilância, drones e monitoramento em tempo real, oferecendo suporte à tomada de decisões em ambientes críticos.

5.3 Limitações do Trabalho

Apesar das contribuições, algumas limitações foram identificadas:

Diversidade do dataset: Embora o dataset tenha incluído diferentes cenários, ele ainda não abrange todas as possíveis variações encontradas em operações reais, como condições climáticas extremas ou veículos de designs menos comuns.

Desempenho em cenários adversos: O YOLO apresentou dificuldades em situações de alta camuflagem, baixa iluminação e presença de ruídos, resultando em falsos positivos e negativos.

Restrição ao YOLO: O estudo focou exclusivamente no YOLO, sem explorar combinações ou modelos híbridos que poderiam oferecer melhor desempenho em cenários específicos.

5.4 Trabalhos Futuros

Com base nos resultados obtidos e nas limitações identificadas, algumas direções para trabalhos futuros são propostas:

1. Expansão do Dataset:

Incluir mais imagens com variações climáticas extremas, ângulos inusitados e veículos menos comuns.

Aumentar a quantidade de exemplos anotados em condições adversas, como baixa iluminação e camuflagem intensa.

2. Técnicas de Pré-processamento:

Aplicar técnicas como aumento de contraste, redução de ruído e balanceamento de cores para melhorar a qualidade das imagens de entrada.

Utilizar dados sintéticos gerados por simulações para complementar o dataset.

3. Treinamento Híbrido:

Combinar o YOLO com outros modelos, como Faster R-CNN, para aproveitar os pontos fortes de ambos.

Implementar arquiteturas mais recentes, como YOLOv7 ou YOLOv8, para explorar avanços tecnológicos mais recentes.

4. Uso em Drones e Sistemas Embarcados:

Testar o modelo diretamente em drones, avaliando sua capacidade de operar em tempo real durante o voo.

Desenvolver soluções de integração para adaptar o modelo ao hardware de drones, levando em conta limitações como consumo de energia e capacidade de processamento.

5. Integração com Tecnologias Avançadas:

Incorporar informações adicionais, como dados de sensores térmicos e infravermelhos, para aumentar a precisão em cenários complexos.

5.5 Considerações Finais

A detecção e o reconhecimento de veículos militares por meio de inteligência artificial representam um avanço significativo na área de defesa e segurança. Este

trabalho demonstrou que o YOLO pode ser uma ferramenta poderosa para aplicações práticas, equilibrando precisão e eficiência em tempo real. Sua possível implementação em drones amplia ainda mais o alcance e a aplicabilidade da tecnologia, permitindo o monitoramento autônomo de áreas extensas e a coleta de informações estratégicas em tempo hábil.

Entretanto, os desafios encontrados reforçam a necessidade de continuidade nos estudos, com melhorias que visem ampliar a robustez do modelo em cenários reais e complexos. Por fim, este estudo não apenas contribui para o campo da visão computacional, mas também oferece uma base sólida para futuras pesquisas e aplicações práticas, destacando o potencial da inteligência artificial na transformação de sistemas de defesa modernos.

REFERÊNCIAS

LECUN, Yann; BOTTOU, Léon; BENGIO, Yoshua; HAFFNER, Patrick. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, v. 86, n. 11, p. 2278–2324, 1998. DOI: [10.1109/5.726791](https://doi.org/10.1109/5.726791).

WANG, Ao; CHEN, Hui; LIU, Lihao; CHEN, Kai; LIN, Zijia; HAN, Jungong; DING, Guiguang. YOLOv10: Real-Time End-to-End Object Detection. Disponível em: <https://arxiv.org/abs/2405.14458>. Acesso em: 10 out. 2024.

REDMON, J. et al. You only look once: Unified, real-time object detection. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. [S.l.: s.n.], 2016.

J. Redmon, A. Farhadi, YOLO9000: better, faster, stronger, in *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (2017), pp. 6517–6525

Lindsay, G. Convolutional neural networks as a model of the visual system: past, present, and future. *Journal of Cognitive Neuroscience*, v. 33, p. 2017-2031, 2020. Disponível em: https://doi.org/10.1162/jocn_a_01544. Acesso em: 10 set. 2024.

BRAGA, A. de P.; CARVALHO, A. de L. F.; LUDERMIR, T. Redes neurais artificiais: teoria e aplicações. *Livros Técnicos e Científicos*, 2000. ISBN 9788521612186. Disponível em: <https://books.google.com.br/books?id=cUgEaAEACAAJ>.

HUANG, R.; PEDOEEM, J.; CHEN, C. Yolo-lite: a real-time object detection algorithm optimized for non-gpu computers. In: *IEEE. 2018 IEEE International Conference on Big Data (Big Data)*. [S.l.], 2018. p. 2503–2510.