

**Universidade de Brasília**  
**Faculdade de Tecnologia**  
**Departamento de Engenharia Mecânica**

**Detecção e contagem automáticas de  
Euschistus heros (Percevejo marrom)  
em lavouras de soja usando  
imagens e aprendizagem profunda**

Bruno Pinheiro de Melo Lima

DISSERTAÇÃO DE MESTRADO  
PROGRAMA DE PÓS-GRADUAÇÃO EM SISTEMAS MECATRÔNICOS

Brasília  
2024

**Universidade de Brasília**  
**Faculdade de Tecnologia**  
**Departamento de Engenharia Mecânica**

**Detecção e contagem automáticas de  
Euschistus heros (Percevejo marrom)  
em lavouras de soja usando  
imagens e aprendizagem profunda**

Bruno Pinheiro de Melo Lima

Dissertação de Mestrado submetida ao Departamento de Engenharia Mecânica da Universidade de Brasília como parte dos requisitos necessários para a obtenção do grau de Mestre

Orientador  
Prof. Dr. Díbio Leandro Borges

Brasília  
2024

## FICHA CATALOGRÁFICA

L732d Lima, Bruno Pinheiro de Melo.  
Detecção e contagem automáticas de *Euschistus heros* (Percevejo marrom) em lavouras de soja usando imagens e aprendizagem profunda / Bruno Pinheiro de Melo Lima; prof. dr. Díbio Leandro Borges. -- Brasília, 2024.  
78 p.

Dissertação de Mestrado (Programa de Pós-Graduação em Sistemas Mecatrônicos) -- Universidade de Brasília, 2024.

1. Aprendizagem profunda. 2. YOLO. 3. Rastreamento e contagem de insetos. 4. Soja. I. Díbio Leandro, Borges, orient. II. Título.

**Universidade de Brasília  
Faculdade de Tecnologia  
Departamento de Engenharia Mecânica**

**Detecção e contagem automáticas de *Euschistus heros*  
(Percevejo marrom) em lavouras de soja usando imagens e  
aprendizagem profunda**

Bruno Pinheiro de Melo Lima

Dissertação de Mestrado submetida ao Departamento de Engenharia Mecânica da Universidade de Brasília como parte dos requisitos necessários para a obtenção do grau de Mestre

Trabalho aprovado. Brasília, 15 de março de 2024:

---

**Prof. Dr. Díbio Leandro Borges, UnB/CIC**  
Presidente

---

**Prof. Dr. Hélio Pedrini, Unicamp/IC**  
Examinador externo

---

**Prof. Dr. Daniel Mauricio Muñoz Arboleda,  
UnB/FGA**  
Examinador interno

*À minha amada filha, Cecília,  
Este trabalho é dedicado a você, que ilumina meus dias com sua  
alegria e pureza. Que você possa crescer em graça, rodeada por amor,  
aprendizado e inspiração. Com todo meu amor.*

# Agradecimentos

Agradeço, em primeiro lugar, a Deus, cujo amparo foi fundamental para superar todas as adversidades e alcançar esta etapa da jornada sem esmorecer.

Expresso minha profunda gratidão à minha família pelo incansável apoio, dedicação à minha educação e pelo amor que sempre me impulsionou. Faço uma menção especial à minha saudosa avó (*in memoriam*). Reconheço, também, o carinho e incentivo de minha esposa, Lara, cujo apoio familiar e emocional foram fundamentais para me fortalecer ao longo desta caminhada.

Agradeço ao meu orientador, Professor Díbio Leandro Borges, pela dedicação exemplar, orientação precisa e atenção dedicada ao meu trabalho. Sem seus ensinamentos, este trabalho seria consideravelmente mais desafiador. Estendo meus agradecimentos aos pesquisadores doutores Edson Hirose, da EMBRAPA Soja, e Lurdineide de Araújo Barbosa Borges, da EMBRAPA Cerrados, pelos préstimos de suas contribuições como meus coautores em publicação de artigos e fornecimento de conhecimentos técnicos fundamentais no que tange à problemática do Percevejo-marrom-da-soja.

# Resumo

No Brasil, a produção de soja aumentou consideravelmente nas últimas décadas, impulsionada por tecnologias agrícolas avançadas. No entanto, o uso excessivo de pesticidas, que representa uma parte significativa dos custos, apresenta desafios econômicos e ambientais. Métodos de detecção e contagem de pragas usando Visão Computacional mostram-se promissores, uma vez que as abordagens tradicionais são trabalhosas e demoradas. Este estudo apresenta um método projetado para identificar e contar insetos em campos de soja, construído sobre um modelo YOLOv8 aprimorado, especificamente desenvolvido para essa missão. Para aprimorar a precisão na detecção de pequenos insetos e reduzir a complexidade do modelo YOLOv8, conduzimos experimentos de ablação para avaliar o impacto da integração de um nível de recursos mais profundo e um bloco com menor número de filtros na espinha dorsal e no pescoço da estrutura original. Por meio desses experimentos, o novo algoritmo foi treinado e testado usando um conjunto de dados de pesquisa público, composto por amostras de *Euschistus Heros* (PMN), um percevejo de interesse em lavouras de soja no Brasil, e seu desempenho foi comparado com três configurações em condições idênticas: A) YOLOv8n, B) YOLOv8n com apenas C2f2 e C) YOLOv8n com apenas P2. Nossa avaliação empregou diversas métricas, incluindo precisão, revocação, mAP0.5 e mAP0.5-0.95. Também consideramos a complexidade do modelo como fatores essenciais na avaliação da eficiência dos modelos YOLO para aplicações específicas, comparando parâmetros, FLOPs, Inferência e Tempo. O modelo proposto foi então integrado a um *framework* capaz de rastrear e contar os PMNs em questão em um *stream* de vídeo. Este *stream* foi criado, animando 42 novas imagens, capturadas sob diversas condições de iluminação e fundo para abordar potenciais desafios em aplicações práticas, testando a capacidade de generalização do modelo, bem como o desempenho do modelo em aplicações de vídeo. O modelo demonstra uma medida de acurácia, representada pela métrica de MOTA, satisfatória, com cerca de 62%. Isso indica que a técnica proposta é eficaz em rastrear bem os percevejos ao longo das cenas apresentadas, o que é corroborado pela alta taxa de acerto na contagem final dos PMNs (desvio para cima de 5,3%). Esses resultados apontam para um potencial desse tipo de *framework* e sinaliza potencial em aplicações futuras de mapeamento de pragas a partir de rastreamentos em tempo real.

**Palavras-chave:** Aprendizagem profunda. YOLO. Rastreamento e contagem de insetos. Soja.

# Abstract

In Brazil, soybean production has increased considerably in recent decades, driven by advanced agricultural technologies. However, the excessive use of pesticides, which represents an important portion of the costs, presents economic and environmental challenges. Pest detection and counting methods using Computer Vision appear promising, since traditional approaches are laborious and time-consuming. This study introduces a method tailored for real-time detecting and counting insects in soybean fields, built upon an improved YOLOv8 model, designed for this mission. To enhance the accuracy in detecting small insects and reduce the complexity of YOLOv8, we conducted ablation experiments to assess the impact of integrating a deeper feature level and a proposed C2f2 layer into the insect detection model. Through these ablation experiments, new algorithm underwent training and testing using a public research dataset, composed of samples of *Euschistus Heros* (NBSB), a bug of interest in soybean crops in Brazil, and its performance was compared against three configurations under identical conditions: A) YOLOv8n, B) YOLOv8n with C2f2 only and C) YOLOv8n with P2 only. Our evaluation employed various metrics, including Precision, Recall, mAP0.5 and mAP0.5-0.95. We also considered model complexity as essential factors in assessing the efficiency of YOLO models for specific applications, by comparing Params, FLOPs, Inference, and Time. The proposed model was then integrated into a framework capable of track and count NBSBs from a video stream. This video stream was created, animating 42 new images, captured under diverse lighting and background conditions to address potential challenges in practical applications, testing the generalization capacity of the model as well as the model's performance in video applications. The model demonstrates a satisfactory measure of accuracy, represented by the MOTA metric, with around 62%. This indicates that the proposed technique is effective in tracking bedbugs well throughout the scenes presented, which is corroborated by the high hit rate in the final insect count (upward deviation of 5.3%). These results point to the potential of this type of *framework* and signal potential in future pest mapping applications based on real-time tracking.

**Keywords:** Deep Learning. YOLO. Insect track and count. Soybean.



# Lista de figuras

Figura 1.1	Presença das principais espécies de percevejos na cultura da soja em diversas regiões produtoras do Brasil, safra 2006/07. Fonte: <a href="#">Corrêa-Ferreira, Krzyzanowski e Minami (2009)</a> . . . . .	17
Figura 1.2	Open REALM, software de mapeamento aéreo em tempo real para imagens de drones. Fonte: <a href="#">Kern et al. (2020)</a> . . . . .	18
Figura 1.3	Esquema geral do método proposto, com o emprego do Modelo obtido na detecção de insetos registrados em vídeos. Adaptado de: <a href="#">Lima et al. (2024)</a>	19
Figura 2.1	<i>Euschistus heros</i> : (a) adulto, (b) ovos, (c) ninfas recém-eclodidas, (d) ninfas de 5º instar. Fonte: <a href="#">Hoffmann-Campo et al. (2000)</a> . . . . .	22
Figura 2.2	Pano-de-batida, técnica empregada para monitorar a presença de percevejos em plantações de soja. Fonte: <a href="#">Prando (2019)</a> . . . . .	23
Figura 2.3	Detecção simultânea de objetos baseada em grade. Adaptado de: <a href="#">Redmon et al. (2016)</a> . . . . .	26
Figura 2.4	Arquitetura simplificada de detectores YOLO, consistindo em <i>Backbone</i> (extração de características), <i>Neck</i> (refino de informações espaciais e semânticas) e <i>Head</i> (ponta final da arquitetura, predição). Adaptado de: <a href="#">Terven, Córdova-Esparza e González (2023)</a> . . . . .	27
Figura 2.5	Backbone da YOLOv8. Adaptado de: <a href="#">King (2023)</a> . . . . .	28
Figura 2.6	Módulo CBS da YOLOv8. Adaptado de: <a href="#">King (2023)</a> . . . . .	29
Figura 2.7	Módulo de agrupamento piramidal espacial rápido (SPPF) da YOLOv8. Adaptado de: <a href="#">King (2023)</a> . . . . .	29
Figura 2.8	Rede Piramidal de atributos (FPN). Fonte: <a href="#">Lin et al. (2017)</a> . . . . .	30
Figura 2.9	<i>Path Aggregation Network for Instance Segmentation</i> (PANet). (a) Estrutura principal da FPN. (b) Caminho de <i>augmentation</i> de baixo para cima. (c) Agrupamento das características adaptativas. (d) Ramo das caixas. (e) Fusão ( <i>fully connected</i> ). Fonte: <a href="#">Liu et al. (2018)</a> . . . . .	30
Figura 2.10	Pescoço ( <i>Neck</i> ) da YOLOv8. Adaptado de: <a href="#">King (2023)</a> . . . . .	31
Figura 2.11	Cabeça ( <i>Head</i> ) da YOLOv8. Adaptado de: <a href="#">King (2023)</a> . . . . .	31
Figura 3.1	[a] Dados principais do conjunto de dados INSECT10K7C640_SAT; [b] Distribuição espacial de PMS nas imagens do conjunto de dados; [c] Contagem de insetos nas imagens e [d] Imagem de amostra no conjunto de dados, compreendendo a cena com PMSs anotados após o pré-processamento.	41
Figura 3.2	Representação da estrutura do modelo YOLO aprimorado. Adaptado de: <a href="#">Lima et al. (2024)</a> . . . . .	43
Figura 3.3	Estrutura da camada C2f2. . . . .	45

Figura 3.4	[a] Dados principais do conjunto de dados EMBRAPA_EUSCHISTUS; [b] Distribuição espacial de PMS nas imagens do conjunto de dados; [c] Contagem de insetos nas imagens e [d] Imagem de amostra no conjunto de dados, compreendendo a cena com PMSs anotados após o pré-processamento.	46
Figura 3.5	Curva de Precisão-Revocação . . . . .	49
Figura 3.6	Resumo das métricas e suas possibilidades de registro no problema de rastreamento para posterior aplicação na contagem de insetos. Adaptado de: <a href="#">Lima et al. (2024)</a> . . . . .	50
Figura 3.7	Esquema geral do processo de transformar imagens em um vídeo dinâmico simulando vídeos obtidos por drone. . . . .	52
Figura 4.1	Relatório de perdas ( <i>losses</i> ) de treinamento para cada modelo ablacionado.	54
Figura 4.2	Relatório de perdas ( <i>losses</i> ) de validação para cada modelo ablacionado.	54
Figura 4.3	Relatório de mAP0.5 para o experimento de ablação sobre YOLOv8. . .	56
Figura 4.4	Relatório de mAP0.5:0.95 para o experimento de ablação sobre YOLOv8.	57
Figura 4.5	Precisão (Prec.), para o experimento de ablação sobre YOLOv8. . . . .	58
Figura 4.6	Revocação (Revoc.), para o experimento de ablação sobre YOLOv8. . . .	58
Figura 4.7	Resultados obtidos para número de parâmetros, em milhões, e para número de operações de ponto flutuante por segundo, em FLOPS(G) para o experimento de ablação sobre YOLOv8. A cor predominante representa o modelo ablacionado vencedor em cada medida. . . . .	60
Figura 4.8	Amostras de inferência (detecção de PMS com o modelo proposto) sobre o conjunto de dados INSECT10K7C640_SAT (conjunto de teste). Os casos verdadeiros positivos são indicados quando o limiar de confiança é maior ou igual a 0.3. No canto inferior direito, em azul, apresentamos insetos detectados ou presentes na imagem em questão. . . . .	60
Figura 5.1	Matriz de confusão expondo os resultados obtidos (VP, VN, FN, FP e IDS), valores absolutos, rastreando PMSs ao longo dos <i>frames</i> do <i>stream</i> de vídeo fornecido. . . . .	63
Figura 5.2	PMS com espinhos no pronoto e mancha branca no dorso bem visível. .	66
Figura 5.3	PMS em vista frontal clara. . . . .	66
Figura 5.4	Dois PMSs visíveis na mesma cena. . . . .	67
Figura 5.5	Ninfa de PMS, ignorada pelo <i>framework</i> . . . . .	67
Figura 5.6	PMS em posição frontal, dificultando sua identificação. . . . .	68
Figura 5.7	PMSs em acasalamento. . . . .	68

## Lista de tabelas

Tabela 1.1	Pragas da soja, parte da planta atacada e sua importância relativa. Fonte: Hoffmann-Campo <i>et al.</i> (2000) . . . . .	16
Tabela 2.1	Evolução dos modelos YOLO e suas contribuições propostas. Fontes: (Ter-ven; Córdova-Esparza; González, 2023; Wang; Yeh; Liao, 2024) . . . . .	25
Tabela 2.2	Trabalhos envolvendo a detecção e classificação de insetos em contexto agrícola. . . . .	33
Tabela 2.3	Trabalhos abordando modificações em arquiteturas YOLO com vistas à detecção de pequenos objetos. . . . .	35
Tabela 2.4	Comparativo entre os algoritmos de rastreamento de objetos DeepSORT e ByteTrack. Inspirado em: Aadi, Sadiq e Labd (2023) . . . . .	39
Tabela 3.1	Estrutura detalhada da YOLOv8n com nível P2. . . . .	44
Tabela 4.1	Comparação do desempenho dos algoritmos em termos de precisão, re-vocação, mAP@0.5, mAP@0.5:0.95, número de parâmetros do modelo, operações de ponto flutuante, tempo de inferência e tempo total de proces-samento. . . . .	55
Tabela 5.1	Resultados quantitativos para cada lote do extrato de <i>frames</i> obtido. . . .	64
Tabela 5.2	Resultados qualitativos para cada lote do extrato de <i>frames</i> obtido. . . .	65

# Lista de abreviaturas e siglas

AP	Average Precision
BCE	Binary Cross Entropy
CIoU	Complete Intersection over Union
CNN	Convolutional Neural Network
CSP	Cross Stage Partial
DFL	Distribution Focal Loss
ELAN	Efficient Layer Aggregation Network
ESPP	Efficient Spatial Pyramid Pooling
FEBlock	Feature Enhancement Block
FPN	Feature Pyramid Network
FPS	Frames por Segundo
GELAN	Generalized Efficient Layer Aggregation Network
IoU	Intersection over Union
mAP	Mean Average Precision
MOTA	Multiple Object Tracking Accuracy
PANet	Path Aggregation Network
PMS	Percevejo-marrom-da-soja
R-CNN	Region based Convolutional Neural Network
SA	Simulated Annealing
SCEP	Self-Characteristic Expansion Plate
SiLU	Sigmoid Linear Unit
SPP	Spatial Pyramid Pooling layer
SPPF	Spatial Pyramid Pooling Fast layer
SSD	Single-shot detectors
TAA	Task-Aligned Assigner
UAV	Unmanned aerial vehicle
YOLO	You Only Look Once

# Sumário

<b>1</b>	<b>INTRODUÇÃO</b>	<b>14</b>
1.1	Caracterização do Problema	14
1.2	Motivação e Justificativa	16
1.3	Objetivos	18
1.3.1	Objetivo Geral	18
1.3.2	Objetivos Específicos	18
1.3.3	Escopo e Contribuições do Trabalho	19
1.4	Organização do Trabalho	20
<b>2</b>	<b>FUNDAMENTAÇÃO TEÓRICA</b>	<b>21</b>
2.1	Percevejo-marrom-da-soja ( <i>Euschistus heros</i> )	21
2.2	Método do Pano-de-batida	22
2.3	Redes Neurais Convolucionais (CNNs)	23
2.4	YOLO	24
2.4.1	YOLOv8	27
2.5	Detecção e Classificação de Insetos em Contexto Agrícola Empregando Modelos YOLO	32
2.6	Modificações de Modelos YOLO para Detecção de Pequenos Objetos	34
2.6.1	Influência do Nível de Característica	37
2.6.2	Influência do Número de Filtros	37
2.7	Métodos de Rastreamentos de Objetos	38
<b>3</b>	<b>MATERIAL E MÉTODOS</b>	<b>40</b>
3.1	Base de Dados	40
3.2	Proposição do Modelo de Detecção	42
3.3	Validação Cruzada do Modelo	45
3.3.1	Aplicação do Modelo de Detecção: Rastreamento e Contagem de PMSs em <i>Stream</i> de Vídeo	47
3.4	Métricas de Desempenho	47
3.4.1	Métricas de Eficácia do Modelo de Detecção	47
3.4.2	Métricas de Avaliação do <i>Framework</i> de Rastreamento e Contagem	50
3.5	Recursos Computacionais Empregados	51
<b>4</b>	<b>ANÁLISE DO MODELO YOLOV8 APRIMORADO</b>	<b>53</b>
4.1	Dados de Treinamento do Modelo	53
4.2	Desempenho do Modelo	55
4.3	Conclusões do Capítulo	61

5	<b>ANÁLISE DO <i>FRAMEWORK</i> DE RASTREAMENTO E CONTAGEM DE INSETOS</b> . . . . .	<b>63</b>
5.1	<b>Conclusões do Capítulo</b> . . . . .	<b>69</b>
6	<b>CONCLUSÃO</b> . . . . .	<b>70</b>
6.1	<b>Conjunto de Dados Proposto</b> . . . . .	<b>70</b>
6.2	<b>Modelo de Detecção Proposto</b> . . . . .	<b>70</b>
6.3	<b>Processo de Rastreamento e Contagem Proposto</b> . . . . .	<b>71</b>
6.4	<b>Sugestão de Trabalhos Futuros</b> . . . . .	<b>71</b>
6.5	<b>Artigo Publicado</b> . . . . .	<b>72</b>
6.6	<b>Repositório</b> . . . . .	<b>72</b>
	<b>REFERÊNCIAS</b> . . . . .	<b>73</b>

# 1 Introdução

O capítulo em questão aborda a caracterização do problema relacionado à cultura da soja e aos desafios associados ao uso indiscriminado de pesticidas, especialmente no controle de pragas de insetos. Explora-se a necessidade de adotar abordagens mais criteriosas e sustentáveis para o manejo dessas pragas, considerando os impactos econômicos e ambientais. O capítulo também destaca a importância do monitoramento inteligente de insetos em sistemas de cultivo e apresenta perguntas de pesquisa pertinentes. Em seguida, são discutidas a motivação e justificativa para a pesquisa, com foco na detecção e contagem do percevejo-marrom-da-soja (*Euschistus heros*) em plantações de soja, destacando a urgência de soluções eficientes e em tempo real. Por fim, são apresentados os objetivos geral e específicos do estudo, bem como o escopo e as contribuições esperadas, além da organização do trabalho em diferentes seções.

## 1.1 Caracterização do Problema

A cultura da soja destaca-se como uma das mais relevantes mundialmente, valorizada por seu elevado teor de proteínas e óleo, por ser uma fonte crucial de óleo vegetal e ração para proteína animal (Masuda; Goldsmith, 2009). Devido a esses atrativos e a valorização nos mercados internacionais de *commodities*, seu cultivo vem se expandindo consistentemente no Brasil desde os anos 1970, atraindo o interesse de agricultores, governos e entidades de pesquisa.

Apesar de sua relevância econômica, o uso indiscriminado e excessivo de pesticidas apresenta desafios econômicos e ambientais substanciais, com os inseticidas contribuindo com mais de 20% dos custos de produção (Bueno *et al.*, 2011). O uso inadequado de defensivos agrícolas representa uma séria ameaça ambiental, contaminando lençóis freáticos, rios e colocando em risco a saúde humana. Para evitar tais consequências, é fundamental adotar uma abordagem criteriosa na escolha e aplicação de inseticidas, integrando métodos de controle químico e biológico. Estratégias de manejo mais precisas, que evitem aplicações uniformes em toda a plantação, não só reduzem custos como também promovem um equilíbrio ecológico mais sustentável.

Os impactos ambientais derivados do uso excessivo de defensivos agrícolas são diversos e abrangentes, incluindo o surgimento de pragas resistentes, a diminuição de organismos benéficos e a contaminação dos recursos hídricos. Diante disso, existe uma pressão socioeconômica crescente para a adoção de práticas agrícolas mais sustentáveis, que incorporem o controle químico de maneira equilibrada e consciente. A busca por soluções sustentáveis é crucial para mitigar os efeitos prejudiciais, fomentando uma agricultura responsável que

preserve tanto o meio ambiente quanto a viabilidade econômica e social.

Em resposta a essa problemática, técnicas de agricultura de precisão surgiram, visando localizar com precisão pragas, doenças e deficiências, permitindo intervenções direcionadas e redução de desperdícios.

Como parte desse avanço, o monitoramento inteligente de insetos em sistemas de cultivo representa uma área de estudo de suma importância, dadas as implicações econômicas e de sustentabilidade associadas a essa possível ameaça. Em ciclos de cultivo específicos, mais de uma centena de espécies de insetos podem coexistir, como pode ser visto na tabela 1.1, especialmente em regiões tropicais, sendo que apenas algumas são classificadas como pragas específicas para uma determinada cultura. Algumas dessas espécies são consideradas neutras, enquanto outras podem ser até mesmo benéficas para o ecossistema agrícola. A popularização do acesso e o barateamento de câmeras com alta qualidade de resolução e computadores potentes, aliado às técnicas de inteligência artificial, notadamente o aprendizado profundo, proporciona uma variedade de possibilidades para abordar a detecção de pragas de insetos em tempo real (Butera *et al.*, 2021).

Revisões abrangentes exploraram técnicas de aprendizado de máquina e aprendizado profundo para o gerenciamento eficiente de insetos, destacando abordagens bem-sucedidas e os desafios a serem superados, especialmente ao lidar com imagens obtidas diretamente em campo (Li *et al.*, 2021; Kasinathan; Singaraju; Uyyala, 2021). A maioria das estratégias recentes é adaptada para uma cultura ou mesmo para uma praga específica, considerando que a importância de controlar pragas de insetos de maneira sustentável apresenta desafios econômicos e práticos (Høye *et al.*, 2021). Essa evolução tecnológica está transformando a identificação de insetos na natureza, e novos avanços são imperativos.

Nesse contexto, modelos de visão computacional têm ganhado destaque, sendo empregados em diversas plataformas para a identificação de insetos e ervas daninhas. Essa abordagem busca, em linhas gerais, identificar a praga o mais rapidamente e precisamente possível em seus estágios iniciais na cultura, representando uma promissora vertente de pesquisa no âmbito do monitoramento inteligente em sistemas de cultivo.

Com isso, concluímos essa caracterização do problema com as seguintes perguntas de pesquisa:

- É viável empregar modelos baseados em YOLO (You Only Look Once) para detectar objetos muito pequenos, como *Euchistus Heros*?
- Uma vez detectados, é possível rastreá-los e contá-los de modo único, garantindo que futuramente possam ser mapeados de modo preciso?



Tabela 1.1 – Pragas da soja, parte da planta atacada e sua importância relativa. Fonte: Hoffmann-Campo *et al.* (2000)

Inseto	Parte da planta atacada	Importância
Anticarsia gemmatalis	Fo	Praga principal
Epinotia aporema	Fo, Br, Va	Secundária, com alguma importância em áreas restritas
Omiodes indicatus	Fo	Secundária, geralmente ocorrendo no final do ciclo da cultura, quando a desfolha não é importante
Pseudoplusia includens	Fo	Secundária
Rachiplusia nu	Fo	Secundária
Cerotoma sp.	Fo(A), No(L)	Secundária, em áreas de soja precedida por feijão
Diabrotica speciosa	Fo(A), Ra(L)	Secundária, em áreas de soja precedida por milho "safriinha"
Aracanthus mourei	Fo, Pe	Secundária, ocorrência no início do crescimento da soja
Maecolaspis calcarifera	Fo	Secundária
Megascelis sp.	Fo	Secundária
Chalcoedermus sp.	Fo	Secundária, praga regionalmente importante
Bemisia argentifolii	Fo	Secundária, com potencial alto de dano
Gafanhotos	Fo	Esporádica
Ácaros	Fo	Esporádica
Trips	Fj	Secundária, importante em áreas muito restritas, vetores de vírus da "queima do broto"
Nezara viridula	Va, Se	Praga principal
Piezodorus guildinii	Va, Se	Praga principal
<b>Euschistus heros</b>	<b>Va, Se</b>	<b>Praga principal</b>
Dichelops furcatus	Va	Secundária
Edessa mediatubunda	Va	Secundária
Thyanta perditor	Va	Secundária
Acrosternum sp.	Va	Secundária
Ethiella zinckenella	Va	Secundária, com alguma importância em áreas restritas
Spodoptera latifascia	Va	Esporádica
Spodoptera eridania	Va	Esporádica
Maruca testulalis	Va	Esporádica
Sternechus subsignatus	Ha	Praga regionalmente importante
Elasmopalpus lignosellus	Ha	Esporádica, usualmente importantes em anos com prolongado período seco, na fase inicial da cultura
Myochrous armatus	Ha	Esporádica
Blapstinus sp.	Pl, Ha	Esporádica
Piolhos-de-cobra	Pl, Pp	Secundária, importante em áreas de semeadura direta
Caracóis e lesmas	Pl, Co, Fj	Secundária, importantes em áreas de semeadura direta
Phyllophaga spp. (Corós)	Ra	Praga regionalmente importante
Scaptocoris castanea	Ra	Praga regionalmente importante
Cochonilhas-da-raiz	Ra	Secundária, importantes em áreas de semeadura direta

Br = brotos; Co = cotilédones; Fj = folhas jovens; Fo = folhas; Ha = hastes; No = nódulos; Pe = pecíolos; Pl = plântulas; Pp = plantas pequenas; Ra = raízes; Se = sementes; Va = vagens. (A) = adulto, (L) = larva.

## 1.2 Motivação e Justificativa

O percevejo-marrom-da-soja (PMS) *Euschistus heros* (Hemiptera: Pentatomidae) destaca-se como uma das pragas mais prejudiciais em plantações de soja no Brasil (Bortolotto *et al.*, 2015), conforme descrito na Figura 1.1. A detecção eficaz do PMS em campos de soja é crucial devido ao seu potencial para causar danos significativos às colheitas. Métodos

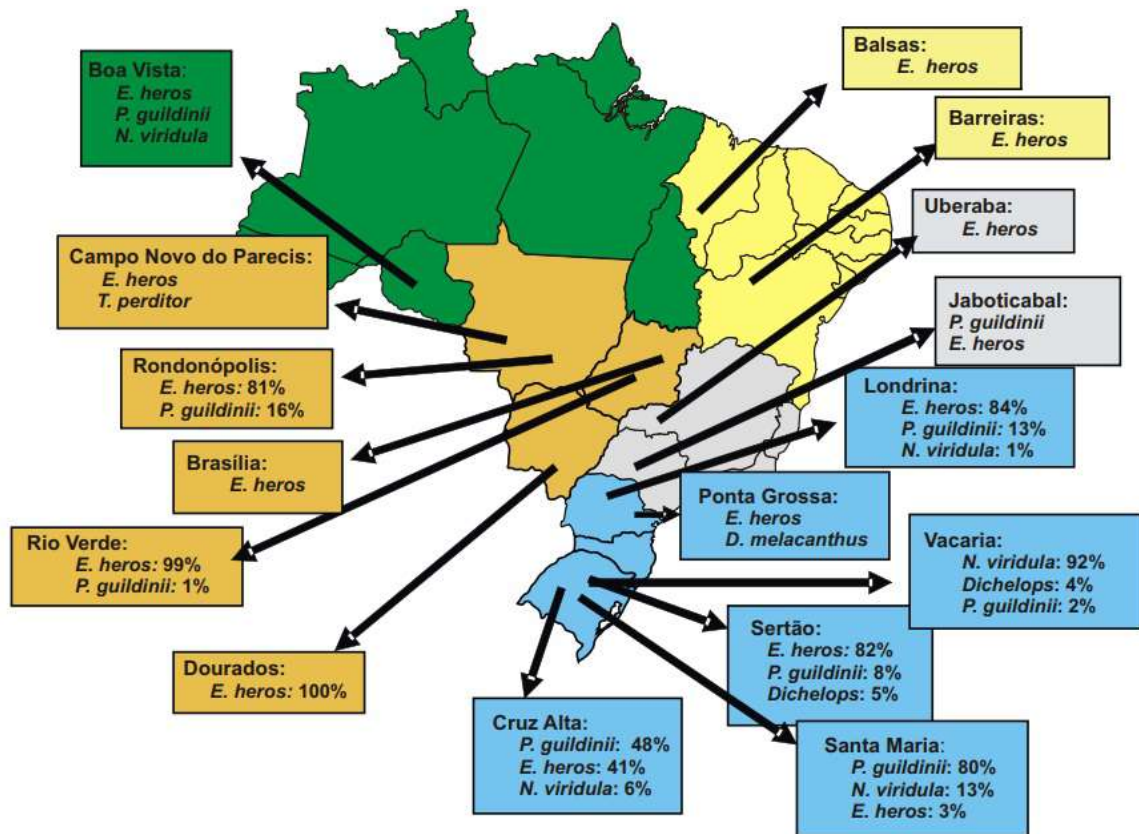


Figura 1.1 – Presença das principais espécies de percevejos na cultura da soja em diversas regiões produtoras do Brasil, safra 2006/07. Fonte: Corrêa-Ferreira, Krzyzanowski e Minami (2009).

tradicionais, como o uso de pano de batida, apresentam desafios, sendo demorados, intensivos em mão de obra e requerendo a contratação de profissionais qualificados, acarretando custos substanciais. Diante disso, a necessidade urgente de explorar soluções eficientes e em tempo real para a detecção e contagem precisa dessas pragas nos estágios iniciais da infestação da cultura se torna evidente.

Adicionalmente, esforços recentes, têm se concentrado no desenvolvimento de métodos viáveis de mapeamento em tempo real utilizando drones (Kern *et al.*, 2020). Essas abordagens visam gerar mapas a partir de transmissões de vídeo, como na Figura 1.2, indicando uma tendência promissora na obtenção, a partir de vídeo, de informações georreferenciadas que podem ser posteriormente mapeadas. Este avanço tecnológico sugere uma direção de pesquisa que, não apenas se alinha às demandas atuais de monitoramento em tempo real, mas também abre possibilidades de aplicação prática na detecção eficiente e posterior mapeamento do PMS e outras pragas em ambientes agrícolas.

Nesse contexto, a presente pesquisa propõe explorar e aprimorar ainda mais essas metodologias, explorando modelos de aprendizado profundo e visão computacional no estado da arte, adaptando-as ao contexto específico da detecção do *Euschistus heros* em plantações de soja, contribuindo assim para a evolução contínua no monitoramento e manejo integrado

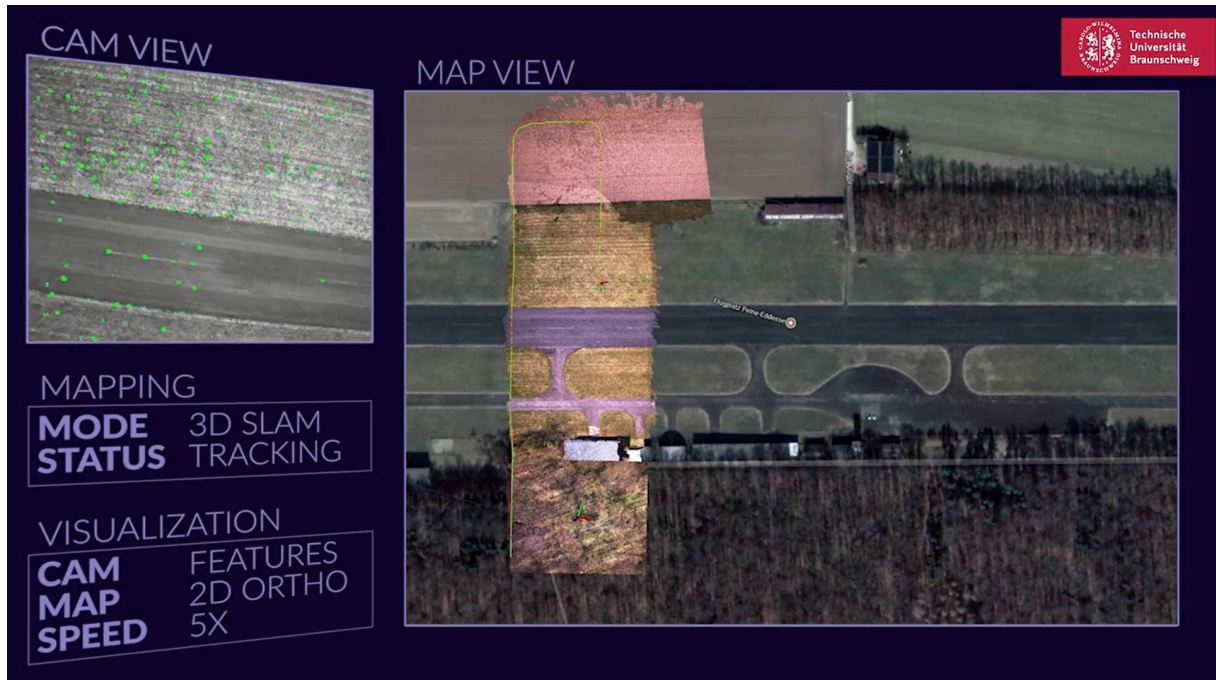


Figura 1.2 – Open REALM, software de mapeamento aéreo em tempo real para imagens de drones.  
Fonte: Kern *et al.* (2020)

de pragas agrícolas ao estudar também sua aplicação na contagem desses insetos em uma cena.

## 1.3 Objetivos

### 1.3.1 Objetivo Geral

O objetivo geral do presente estudo é propor um método viável de detecção e contagem populacional de PMSs em lavouras empregando modelos baseados na arquitetura YOLO e imagens reais de campo.

### 1.3.2 Objetivos Específicos

Esta pesquisa aborda esse objetivo geral, através dos seguintes objetivos específicos:

- Propor um modelo YOLOv8 aprimorado, com nível de característica P2 mais profundo e bloco C2f2 na espinha dorsal e no pescoço da estrutura, computacionalmente mais leve, adaptado para detecção de pequenos objetos, como insetos em imagens de campo;
- Realizar experimentos de ablação com as modificações propostas (nível P2 e bloco C2f2) para avaliar a eficácia de sua inclusão;
- Testar com um conjunto de dados público e com novas imagens coletadas em campos de cultivo de soja, transpostos em forma de vídeo, conforme exposto na Figura 1.3;

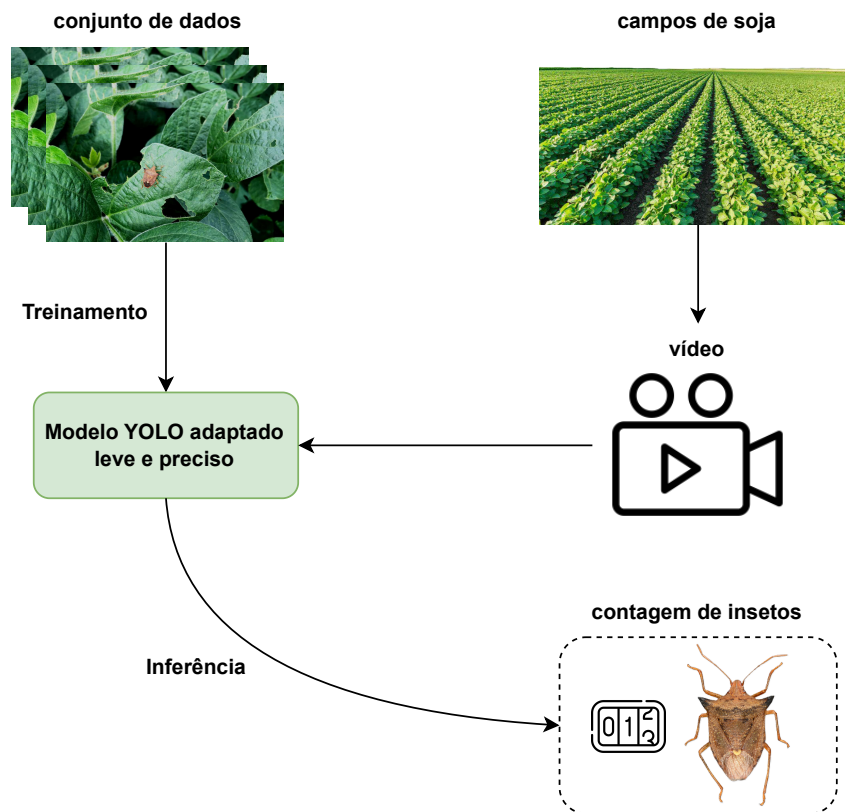


Figura 1.3 – Esquema geral do método proposto, com o emprego do Modelo obtido na detecção de insetos registrados em vídeos. Adaptado de: [Lima et al. \(2024\)](#)

### 1.3.3 Escopo e Contribuições do Trabalho

Portanto, este trabalho contribui com o estado atual de pesquisa acerca do emprego de Visão Computacional e Inteligência Artificial em problemas de detecção de pragas de interesse agrícola, delimitando-se ao seguinte escopo:

Propõe-se um modelo YOLOv8 aprimorado, explorando os aspectos de profundidade da rede e inserção de elementos visando leveza computacional na espinha dorsal e no pescoço da estrutura, adaptado para detecção, rastreamento e contagem de percevejo-marrom-da-soja (PMS) *Euschistus heros* (Hemiptera: Pentatomidae) em sua fase adulta na cultura agrícola de soja (*Glycine max*). Empregou-se conjunto de dados disponível publicamente em pesquisas anteriores e imagens inéditas obtidas em campo para testes de avaliação cruzada. As imagens em ambos os conjuntos foram produzidas em ambientes agrícolas reais a até 1,5 metro do solo, em dias distintos e condições climáticas diversas, capturando os insetos em diferentes ângulos e em situações desafiadoras como acasalamento e oclusão parcial.

## 1.4 Organização do Trabalho

Este trabalho está organizado de modo que a fundamentação teórica, está exposta no Capítulo 2. Os materiais e métodos da pesquisa, expondo os conjuntos de dados utilizados, as métricas pertinentes e o método para avaliação do Modelo e do *framework* de aplicação estão cobertos no Capítulo 3. Os resultados obtidos na construção do Modelo e na sua aplicação no *framework* de rastreamento e contagem foram analisados e discutidos nos capítulos 4 e 5, respectivamente. Por fim, as conclusões são abordadas no Capítulo 6.

## 2 Fundamentação Teórica

Este capítulo aborda a problemática da detecção e mapeamento dos percevejos-marrom-da-soja (*Euschistus heros*) em áreas de cultivo de soja. Inicialmente, são discutidos os impactos negativos dos PMSs na produção de soja e a predominância deles como inseto de interesse, por seu impacto negativo, em diversas regiões produtoras do Brasil. A aplicação excessiva de inseticidas e suas consequências também são abordadas. Em seguida, o capítulo explora o método do pano-de-batida como uma técnica tradicional de monitoramento de populações de insetos na cultura da soja, e são destacadas suas fraquezas para o controle do PMS. Posteriormente, são introduzidas as Redes Neurais Convolucionais (CNNs) como uma ferramenta promissora na detecção de pragas agrícolas, seguidas pela apresentação do modelo YOLO (*You Only Look Once*) e sua evolução até o YOLOv8, com ênfase em sua aplicação na detecção de insetos. Além disso, são discutidas modificações específicas em modelos YOLO para aprimorar a detecção de objetos pequenos, como os PMS, e métodos de rastreamento de objetos, com destaque para o ByteTrack, considerando sua relevância para operações em tempo real, como o monitoramento de insetos em lavouras.

### 2.1 Percevejo-marrom-da-soja (*Euschistus heros*)

Os percevejos representam uma ameaça significativa para a cultura de soja, afetando negativamente o rendimento e a qualidade das sementes. Dentre as espécies mais relevantes, se destacam *Nezara viridula*, *Piezodorus guildinii* e *Euschistus heros* se destacam, sendo esta última adaptada a climas mais quentes (Hoffmann-Campo *et al.*, 2000). Outro dado importante é que o *Euschistus heros* é responsável por mais de 90% das infestações de percevejos sugadores em diversas regiões produtoras de grãos do Brasil (Corrêa-Ferreira; Sosa-Gómez, 2017). Sua atividade como praga se intensifica durante o enchimento de grãos, atingindo densidades máximas no período de maturação.

O ciclo de vida do percevejo-marrom-da-soja (*E. heros*) abrange três gerações durante a safra de soja, com a capacidade de se alimentar de amendoim-bravo após a colheita e após a diapausa na palhada da cultura anterior, *E. heros* sobrevive utilizando as reservas de lipídios armazenadas (Hoffmann-Campo *et al.*, 2000). Além disso, em que pese a espécie ser menos polífaga em comparação com outros percevejos, sua presença exige monitoramento devido aos danos irreversíveis que pode causar.

A pressão por aumentar a produtividade, associada a preços favoráveis da soja e custos relativamente baixos de inseticidas no passado, levou a aplicações excessivas e indiscriminadas. Porém, o manejo eficaz dos percevejos na cultura de soja é crucial para evitar tornar os cultivos de soja mais eficientes economicamente e reduzir seus impactos ambientais em

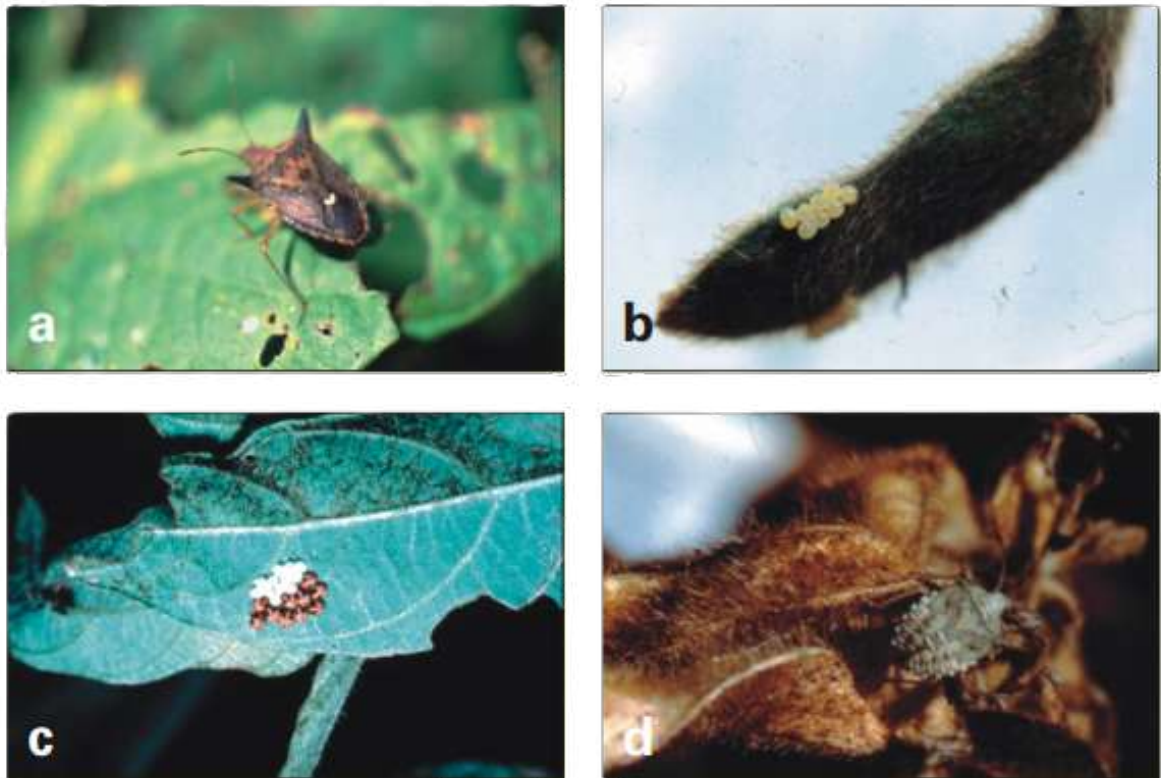


Figura 2.1 – *Euschistus heros*: (a) adulto, (b) ovos, (c) ninfas recém-eclodidas, (d) ninfas de 5º ínstar.  
Fonte: Hoffmann-Campo *et al.* (2000)

momento de pressões crescentes nesse sentido. Ademais, além de o uso indiscriminado de inseticidas resultar em aumento dos custos de produção, pode gerar seleção de populações resistentes e surtos futuros de pragas mais severos (Bueno; Sosa-Gómez; Silva, 2020).

Portanto, a abordagem racional do uso de inseticidas é essencial tanto para equilibrar a eficácia no controle de *Euschistus heros* quanto para promover maior sustentabilidade agrícola.

## 2.2 Método do Pano-de-batida

O monitoramento eficaz das populações de insetos na cultura de soja é fundamental para um manejo integrado bem-sucedido, evitando prejuízos à colheita e reduzindo o uso desnecessário de inseticidas. O método usual para esse tipo de controle é o do pano-de-batida, exemplificado na Figura 2.2, que envolve amostragens periódicas em diferentes pontos da lavoura, registrando ninfas e adultos. Durante a colonização da soja, a atenção deve ser redobrada nas bordaduras, onde os percevejos costumam iniciar a infestação. Recomenda-se que as amostragens ocorram nos períodos mais frescos do dia para maior eficácia. Além disso, a observação pontual não é suficiente para determinar a real população de percevejos, destacando a importância do monitoramento sistemático desde a floração até a maturação das plantas (Corrêa-Ferreira; Krzyzanowski; Minami, 2009).



Figura 2.2 – Pano-de-batida, técnica empregada para monitorar a presença de percevejos em plantações de soja. Fonte: Prando (2019).

Apesar do pano-de-batida ser eficiente para lagartas, é menos adotado para percevejos, devido à mobilidade das ninfas e adultos. Uma alternativa viável é o uso de armadilhas com iscas feromonais para capturar adultos (Pires *et al.*, 2006). É importante também destacar que o emprego do pano-de-batida deve considerar diversos fatores, como o espaçamento entre linhas e a densidade populacional alvo. O pano-de-batida, pano-de-batida largo e pano-de-batida vertical são métodos utilizados para amostrar lagartas e percevejos em soja. Com a diminuição do espaçamento entre linhas, o pano-de-batida foi comprometido, surgindo alternativas como o pano-de-batida largo e o pano-de-batida vertical, este último recomendado na Argentina para espaçamentos reduzidos (Stürmer *et al.*, 2014).

Trabalhos recentes apontaram para possíveis alternativas ao método do pano-de-batida empregando técnicas de visão computacional, como a Regressão Logística e a Análise Discriminante Linear, aplicadas a imagens de drones na identificação do *Euschistus heros* em plantação de soja (Sabará, 2018). O presente trabalho reconhece esse esforço como motivação ao emprego de outras técnicas no estado da arte para a solução desse problema.

## 2.3 Redes Neurais Convolucionais (CNNs)

As Redes Neurais Convolucionais (CNNs) representam um avanço crucial no âmbito do aprendizado profundo, especialmente na área de visão computacional. Essa classe de redes neurais foram projetadas para processar e analisar dados visuais, inspiradas no sistema visual humano. A arquitetura fundamental das CNNs incorpora camadas convolucionais,



camadas de *pooling* e camadas *fully connected*. As camadas convolucionais utilizam filtros ou *kernels* para realizar convolução sobre os dados de entrada, capturando hierarquias espaciais e permitindo a extração de características. As camadas de *pooling* reduzem as dimensões espaciais, facilitando uma representação robusta de características. As camadas *fully connected* conectam as características extraídas para a classificação final.

A evolução das CNNs remonta ao trabalho pioneiro de [LeCun, Bengio e Hinton \(2015\)](#), que introduziram a arquitetura LeNet para o reconhecimento de dígitos manuscritos. A AlexNet, desenvolvida por [Krizhevsky, Sutskever e Hinton \(2012\)](#), marcou um momento crucial na história das CNNs, exibindo desempenho sem precedentes no Desafio de Reconhecimento Visual em Grande Escala do ImageNet. Desde então, arquiteturas subsequentes como VGG, GoogLeNet, ResNet e outras refinaram ainda mais as CNNs, introduzindo modelos mais profundos e eficientes.

A detecção de objetos, uma tarefa fundamental em visão computacional, se beneficia de arquiteturas como *region-based CNNs* (R-CNN), *single-shot detectors* (SSD) e *You Only Look Once* (YOLO). Esses modelos possibilitam a localização precisa e a classificação de objetos em imagens ou vídeos. Tarefas de classificação fazem uso do poder discriminativo das CNNs, permitindo distinguir entre diversas categorias dentro de um conjunto de dados. Isso é exemplificado pela habilidade das CNNs em alcançar precisão de ponta em conjuntos de dados de referência, consolidando sua posição como ferramentas indispensáveis na classificação de imagens.

Alguns trabalhos recentes abordaram de maneira diligente o problema da detecção e da classificação de imagens de pragas de insetos em plantações de soja. [Tetila et al. \(2019\)](#) avaliaram três modelos de CNN, DenseNet-201, Inception-Resnet-v2 e Resnet-50, em imagens de campo de soja após segmentação de superpixels SLIC. Originalmente, eles tinham 1.000 imagens com 7 classes (6 pragas de insetos e uma não praga). Com os pesos da ImageNet, eles relataram uma precisão de classificação de imagens dos insetos de 94,89%. Uma continuação desse trabalho foi publicada com 5.000 imagens para avaliar estratégias de ajuste fino de Inception-v3, Resnet-50, VGG-16, VGG-19 e Xception, onde foram relatadas precisões de classificação acima de 90% ([Tetila et al., 2020](#)).

## 2.4 YOLO

Os modelos YOLO, introduzidos em 2015 ([Redmon et al., 2016](#)), representam um avanço inovador no campo da detecção de objetos em tempo real utilizando redes neurais profundas. Em essência, as redes YOLO são projetadas para identificar e localizar múltiplos objetos em imagens e vídeos de maneira rápida e eficiente. A arquitetura original passou por várias fases evolutivas, conforme verificamos na Tabela 2.1, cada uma contribuindo para a eficácia da série em aplicações de visão computacional.

Tabela 2.1 – Evolução dos modelos YOLO e suas contribuições propostas. Fontes: (Terven; Córdova-Esparza; González, 2023; Wang; Yeh; Liao, 2024)

Modelo	Ano	Contribuição e Avanços
YOLO	2016	Introdução da detecção rápida de objetos em tempo real, embora com limitações na precisão da localização e na detecção de múltiplos objetos da mesma classe.
YOLOv2	2017	Aprimoramentos significativos com a inclusão de normalização em lote, arquitetura <i>fully convolutional</i> , previsão direta de caixas delimitadoras e melhorias na detecção multi-escala.
YOLOv3	2018	Avanços em previsão de caixas delimitadoras e classes, introdução de um novo <i>backbone</i> , aprimoramentos em previsões multi-escala e determinação de caixas âncora com k-means.
YOLOv4	2020	Incorporação de estratégias avançadas de treinamento, como mosaico de imagens e <i>DropBlock</i> , treinamento auto-adversarial e otimização de hiperparâmetros, resultando em melhorias significativas na detecção de objetos.
YOLOv5	2020	Utilização do Pytorch e do algoritmo AutoAnchor, melhorias na previsão de caixas delimitadoras e classes, contribuindo para uma detecção mais precisa e eficiente.
YOLOX	2021	Retorno à arquitetura sem âncora, simplificação do treinamento e decodificação, introdução de amostragem centrada e separação de confiança e precisão, resultando em melhorias significativas na detecção de objetos.
YOLOv6	2022	Introdução de uma nova espinha dorsal baseado em RepVGG, estratégia de auto-dissipação, melhorias nas perdas de classificação e regressão, contribuindo para um aumento na precisão e eficácia da detecção.
YOLOv7	2022	Desenvolvimento da extensão E-ELAN para aprendizagem eficiente, estratégia de escalonamento para modelos de concatenação, proporcionando melhorias no aprendizado e na eficiência dos modelos.
YOLOv8	2023	Implementação de um modelo sem âncora com cabeça dupla para processamento independente de tarefas, introdução de um modelo de segmentação semântica YOLOv8-Seg, proporcionando melhorias significativas em <i>benchmarks</i> de detecção de objetos e segmentação semântica.
YOLOv9 <sup>1</sup>	2024	Introduz o Gradiente de Informações Programável (PGI) para resolver o gargalo de informações no processo de <i>feed-forward</i> , gerando gradientes confiáveis, pensando principalmente em modelos menores. Arquitetura ELAN generalizada (GELAN), projetada para considerar parâmetros, complexidade computacional, precisão e velocidade de inferência simultaneamente.

As redes YOLO se utilizam de uma grade fixa sobreposta à imagem de entrada,

<sup>1</sup> A arquitetura YOLOv9 foi desenvolvido após a conclusão desta pesquisa. Porém achamos importante citá-la brevemente, bem como suas contribuições, durante a escrita final da Dissertação.

dividindo-a em células, conforme a Figura 2.3. Ao contrário de abordagens tradicionais, a CNN da YOLO processa todas as células simultaneamente, fornecendo uma única saída para a imagem inteira. Essa metodologia baseada em grade estabelece os fundamentos para a detecção de objetos em tempo real, conforme demonstrado por sua capacidade de identificar objetos de forma rápida e precisa em vídeos. A saída da CNN da YOLO é caracterizada por informações de classificação de objetos e informações de localização, representadas por caixas delimitadoras etiquetadas. Essa extração simultânea de detalhes de classificação e localização é alcançada pela incorporação de caixas ancoradas dentro das células da grade. Essas caixas possibilitam à YOLO detectar múltiplos objetos dentro de uma única célula de grade, aprimorando sua versatilidade no tratamento de cenários complexos (Mehta, 2023).

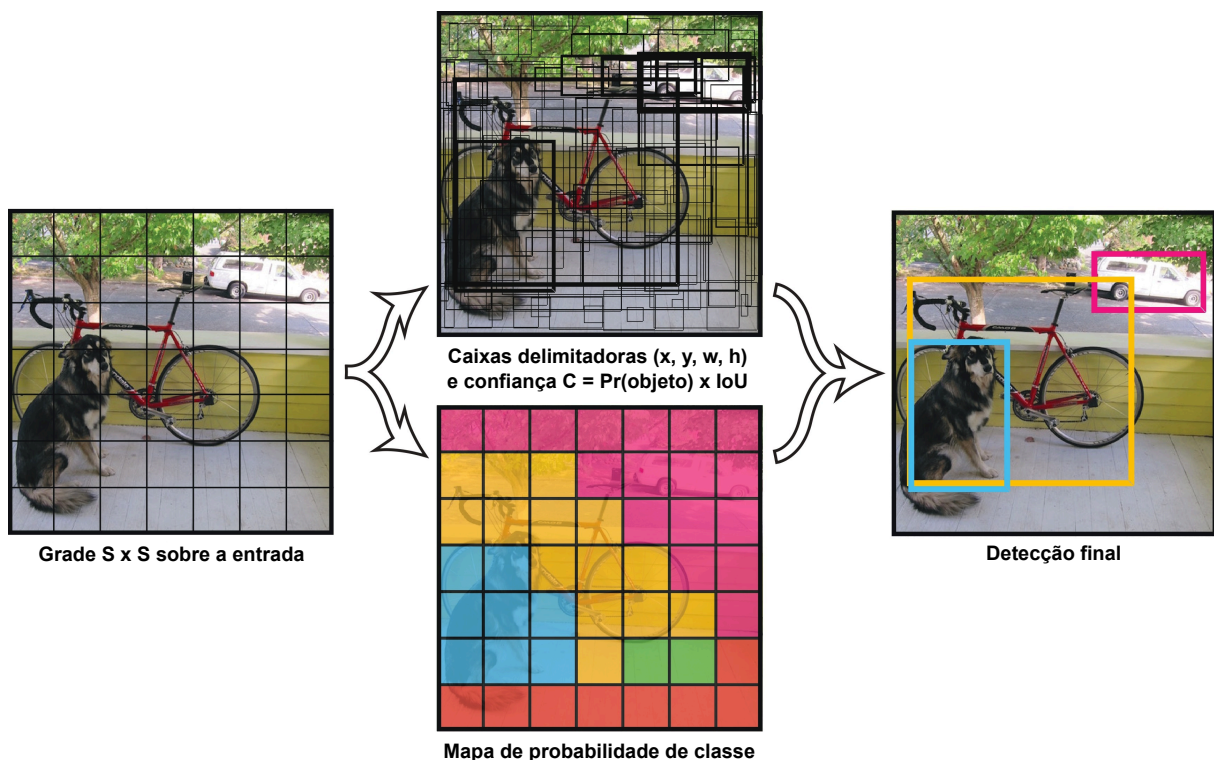


Figura 2.3 – Deteccção simultânea de objetos baseada em grade. Adaptado de: Redmon *et al.* (2016)

A arquitetura básica de um modelo YOLO é dividida em três partes: a espinha dorsal (*backbone*), o pescoço (*neck*) e a cabeça (*head*), conforme Figura 2.4. A espinha dorsal desempenha um papel crucial na extração de características valiosas de imagens de entrada, normalmente utilizando uma CNN treinada em conjuntos de dados extensos de classificação de imagens, como o ImageNet. Esse componente captura características hierárquicas em diferentes escalas, extraindo características de baixo nível (como bordas e texturas) nas camadas iniciais, e características de alto nível (incluindo partes de objetos e informações semânticas) nas camadas mais profundas. O pescoço atua como um componente intermediário conectando a espinha dorsal à cabeça (Terven; Córdova-Esparza; González, 2023).

Além das inovações arquiteturais, as distinções na série YOLO abrangem vários aspectos

tos no processo de treinamento, nas funções de perda e nas técnicas de *data augmentation*. Além disso, diversas configurações de hiperparâmetros contribuem para a complexidade da configuração para cada caso específico que se quer implementar. Por exemplo, variações como detecções baseados em âncoras e *anchor free* (sem âncoras) em diferentes modelos YOLO introduzem desafios multidimensionais, exigindo uma compreensão sutil dos avanços desenvolvidos nas arquiteturas em relação a outros fatores que têm também influência no desempenho do modelo (Lazarevich, 2023).

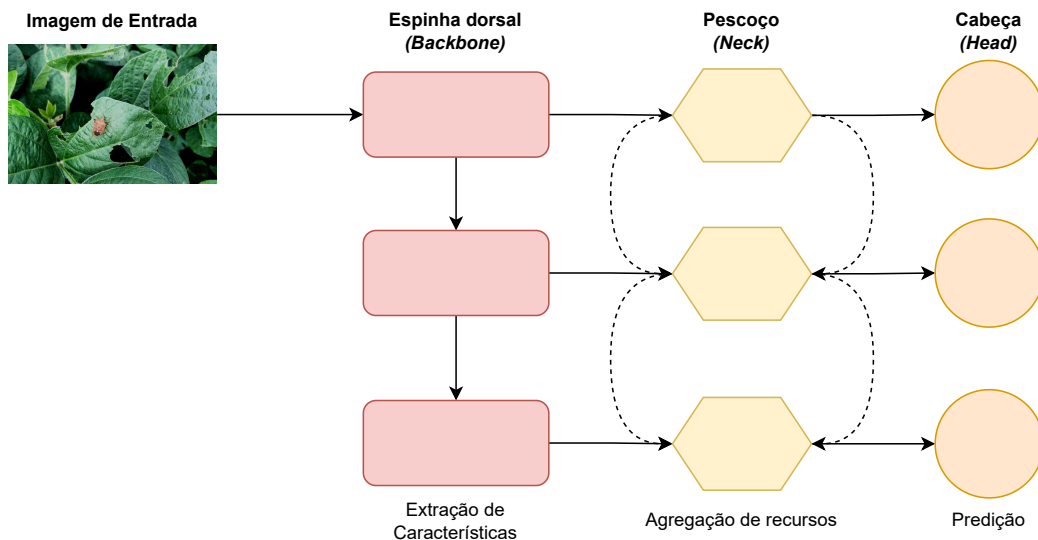


Figura 2.4 – Arquitetura simplificada de detectores YOLO, consistindo em *Backbone* (extração de características), *Neck* (refino de informações espaciais e semânticas) e *Head* (ponta final da arquitetura, predição). Adaptado de: Terven, Córdova-Esparza e González (2023)

### 2.4.1 YOLOv8

A arquitetura mais recente desenvolvida pela Ultralytics é a YOLOv8, a qual representa um avanço notável na detecção de objetos em tempo real, construída sobre os fundamentos da bem-sucedida YOLOv5, amplamente explorada em trabalhos acadêmicos. Esse novo modelo se destaca por sua arquitetura de rede leve e eficazes técnicas de fusão de características. Além disso, adota uma abordagem *Anchor-Free*, inovando ao eliminar a necessidade de caixas âncoras predefinidas, proporcionando maior flexibilidade e eficácia durante a detecção de objetos (Jocher, 2023).

A estrutura da rede YOLOv8, conforme visto anteriormente para outros modelos YOLO, pode ser resumida por uma espinha dorsal (*Backbone*), um pescoço (*Neck*) e uma cabeça (*Head*), detalhados a seguir:

### 2.4.1.1 Espinha Dorsal

Na espinha dorsal da YOLOv8, Figura 2.5, uma rede neural CSPDarknet53 reformulada é empregada como a rede principal (Redmon; Farhadi, 2018). Essa rede consiste em cinco estágios de amostragem, cada um gerando características em diferentes escalas (B1 a B5). Em comparação com a versão anterior, YOLOv5, a YOLOv8 adota o módulo C2f, substituindo o módulo *Cross Stage Partial* (CSP) anterior. O módulo C2f é projetado para aprimorar o fluxo de informações dentro da rede de extração de características, mantendo uma arquitetura compacta e introduz uma conexão de desvio de gradiente, mecanismo de conexão dentro de rede neural que facilita o fluxo de gradientes durante o processo de treinamento, contornando certas camadas ou módulos e permitindo que os gradientes alcancem camadas mais profundas de forma mais eficiente, sem comprometer o desempenho (Wang et al., 2023).

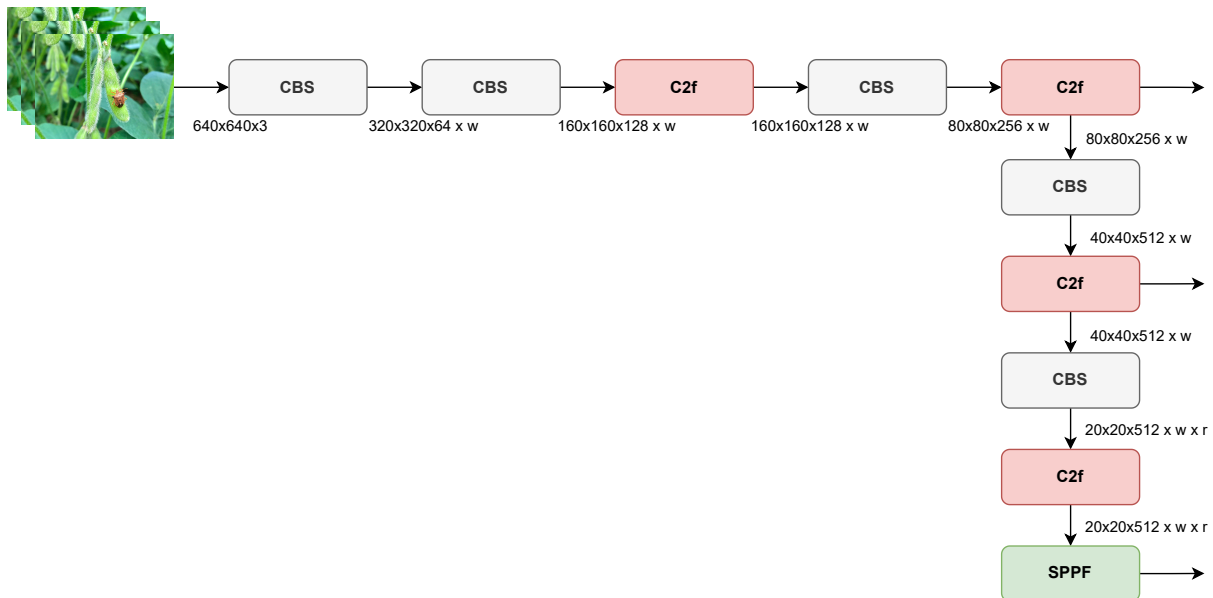


Figura 2.5 – Backbone da YOLOv8. Adaptado de: King (2023)

Adicionalmente, o módulo CBS desempenha um papel crucial no processamento dos dados de entrada. Ele executa uma operação de convolução seguida por normalização em lote e ativação por meio da função *Sigmoid Linear Unit* (SiLU), conforme Figura 2.6, fornecendo o resultado desejado com eficácia.

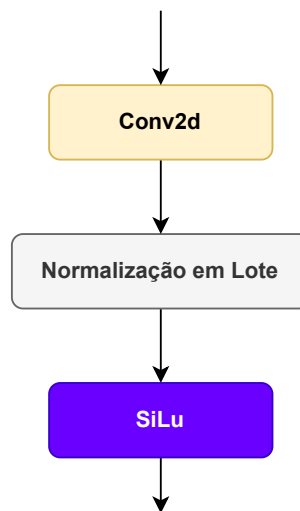


Figura 2.6 – Módulo CBS da YOLOv8. Adaptado de: [King \(2023\)](#)

Para melhorar a eficiência computacional e minimizar a latência, a rede principal da YOLOv8 incorpora o módulo de agrupamento piramidal espacial rápido (*Spatial Pyramid Pooling Fast layer, SPPF*), conforme a Figura 2.7. Este módulo é essencial para produzir uma saída de tamanho fixo, superando a tradicional estrutura de agrupamento piramidal espacial (*Spatial Pyramid Pooling layer, SPP*) em termos de complexidade computacional e reduzindo o tempo de execução, garantindo uma detecção de objetos eficiente e ágil ([He et al., 2015](#)).

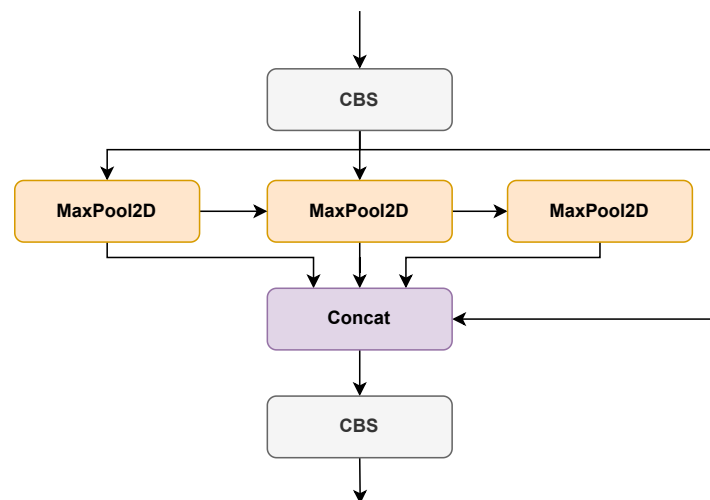


Figura 2.7 – Módulo de agrupamento piramidal espacial rápido (SPPF) da YOLOv8. Adaptado de: [King \(2023\)](#)

#### 2.4.1.2 Pescoço

Uma Rede Piramidal de atributos, ou (*Feature Pyramid Network, FPN*), é um extrator de características projetado para lidar com o desafio de detectar objetos em várias escalas dentro de uma imagem, produzindo múltiplas camadas de mapas de características ([Lin et](#)

*al.*, 2017). Essas camadas oferecem informações detalhadas em diferentes níveis de resolução espacial, tornando-se especialmente úteis para tarefas de detecção de objetos com variação significativa de tamanho. Com um caminho ascendente para a extração de características e um caminho descendente para reconstruir camadas de resolução mais alta, a FPN facilita a detecção precisa e robusta.

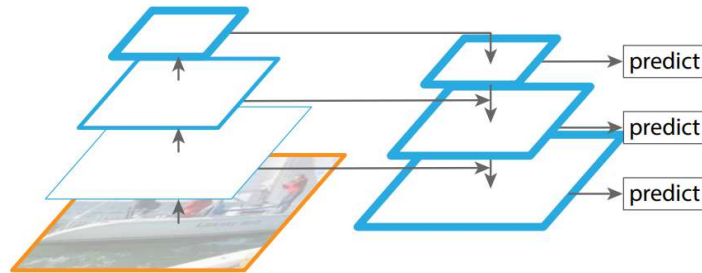


Figura 2.8 – Rede Piramidal de atributos (FPN). Fonte: Lin *et al.* (2017)

Na YOLOv8, uma estrutura inspirada na PANet, Figura 2.9, complementa a FPN tradicional aprimorando-a semanticamente, embora tendo como efeito indesejado uma perda de informações de posicionamento, o que é compensado fortalecendo o aprendizado de características de localização ao fundir pares específicos de mapas de características Liu *et al.* (2018).

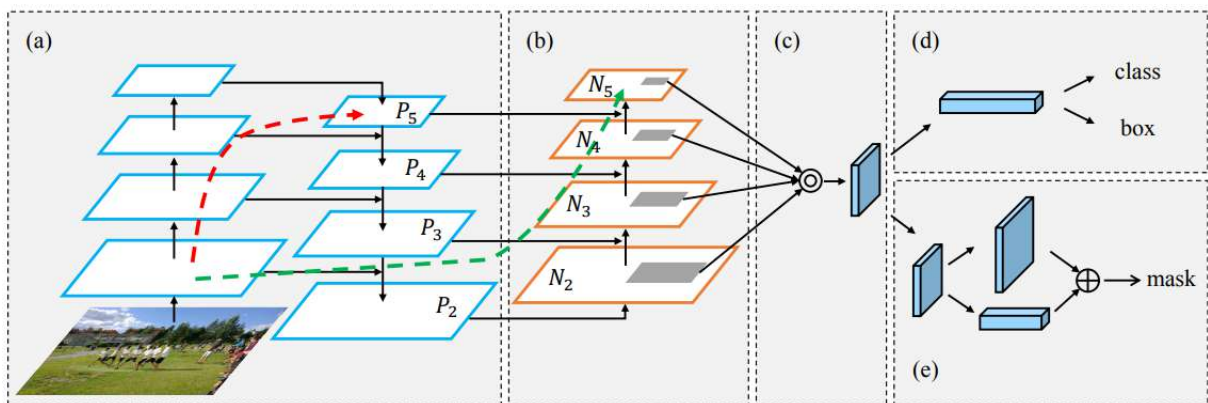


Figura 2.9 – Path Aggregation Network for Instance Segmentation (PANet). (a) Estrutura principal da FPN. (b) Caminho de *augmentation* de baixo para cima. (c) Agrupamento das características adaptativas. (d) Ramo das caixas. (e) Fusão (*fully connected*). Fonte: Liu *et al.* (2018)

Ao final do processo de fusão, os seguintes pares foram unidos: B3–P3, B4–P4, P4–N4, P5–N5, obtendo-se o pescoço conforme a Figura 2.10. Essa união alcança então um equilíbrio entre aprimoramento semântico e precisão posicional.

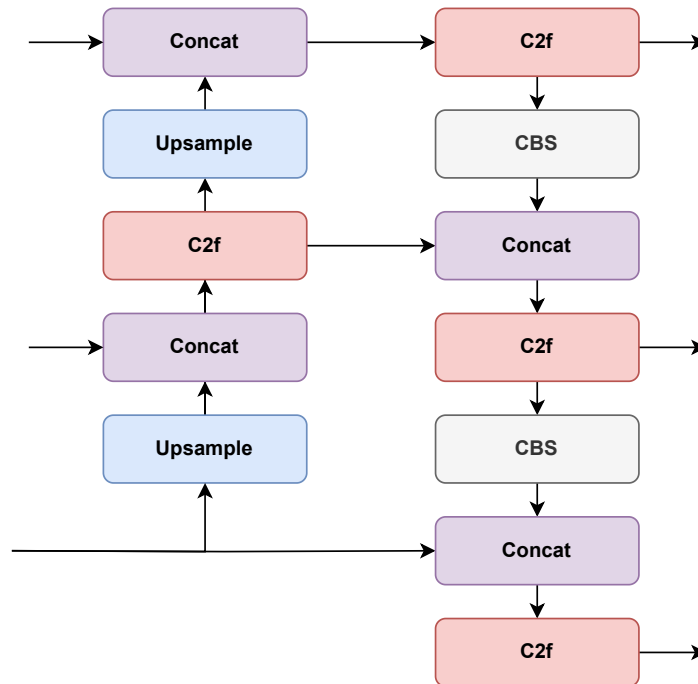


Figura 2.10 – Pescoço (*Neck*) da YOLOv8. Adaptado de: King (2023)

#### 2.4.1.3 Cabeça

A cabeça da YOLOv8, exposta na Figura 2.11, emprega uma estrutura de cabeça desacoplada, que separa os processos de classificação de objetos e regressão das caixas delimitadoras previstas em dois ramos distintos da rede.

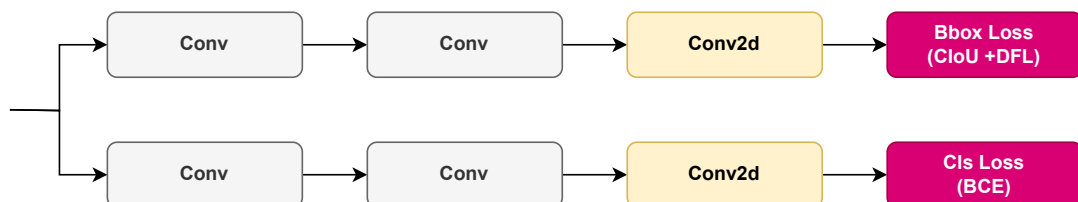


Figura 2.11 – Cabeça (*Head*) da YOLOv8. Adaptado de: King (2023)

- **Classificação de Objetos:** é realizada em um ramo específico usando a perda de entropia cruzada binária (BCE Loss). Essa função de perda é comumente usada para problemas de classificação binária, onde cada objeto pode ser classificado como presente ou ausente.
- **Regressão de Caixa Delimitadora:** é tratada em outro ramo usando duas funções de perda diferentes: a perda focal de distribuição (DFL) (Li *et al.*, 2020) e a Interseção Completa sobre a União (CIoU) (Zheng *et al.*, 2020). Essas funções de perda são utilizadas para ajustar a posição e tamanho das caixas delimitadoras previstas em relação às caixas delimitadoras reais dos objetos na imagem.



A definição explícita de amostras positivas e negativas ocorre através da identificação clara e precisa de exemplos de treinamento que contenham objetos de interesse (amostras positivas) e aqueles que não contenham esses objetos (amostras negativas). Por exemplo, em um problema de detecção de objetos em imagens, as amostras positivas seriam imagens que contêm os objetos que desejamos detectar, enquanto as amostras negativas seriam imagens que não contêm esses objetos.

O uso dinâmico do atribuidor alinhado à tarefa (*Task-Aligned Assigner*, TAA) é uma técnica que se refere à maneira como essas amostras de treinamento são distribuídas ou atribuídas aos diferentes ramos da rede neural durante o treinamento. Dessa forma, o TAA é um componente que gerencia essa atribuição dinâmica, garantindo que cada ramo receba uma quantidade adequada de amostras relevantes para aprender a distinguir corretamente os objetos de interesse, enquanto o ramo de regressão recebe amostras que ajudam a ajustar precisamente as caixas delimitadoras em torno desses objetos (Feng *et al.*, 2021).

Essa abordagem dinâmica de atribuição de amostras de treinamento ajuda a otimizar o desempenho da rede neural, garantindo que cada parte dela receba informações úteis e relevantes para sua tarefa específica, o que pode melhorar a precisão e robustez da detecção de objetos durante o treinamento.

## 2.5 Detecção e Classificação de Insetos em Contexto Agrícola Empregando Modelos YOLO

Os modelos YOLO são amplamente utilizados para detecção de objetos, podendo, porém, enfrentar dificuldades em detectar precisamente objetos pequenos e apresentar alto custo computacional. Apesar dessas limitações, o processamento em tempo real e o desempenho geral dos modelos YOLO tornaram-nos os preferido em aplicações de visão computacional. O trabalho de Silveira *et al.* (2021) avaliou a YOLOv3 para detecção de pragas de insetos em tempo real em campos de soja e falhou em alguns casos, como regiões caracterizadas por variações intrincadas de iluminação, instâncias com presença de atividade herbívora, outros danos às folhas e visibilidade limitada de insetos. Verma *et al.* (2021) compararam YOLOv4 e YOLOv5, com essa última alcançando a maior precisão na detecção de insetos. Além disso, Önlér (2021) utilizou o YOLOv5 para identificar lagartas de cardo em cultivo de girassol com 59% de mAP (Média das Precisões Médias) e conseguiu detectar diferentes espécies de lagartas. Por sua vez, Ahmad *et al.* (2022) desenvolveram um sistema de reconhecimento de objetos para vários insetos com base em várias arquiteturas YOLO, sendo que o YOLOv5, particularmente YOLOv5x, superou os outros. E, mais recentemente, em Khalid *et al.* (2023), a YOLOv8 alcançou a maior mAP para detecção precoce de pragas em comparação com outras arquiteturas YOLO.

A Tabela 2.2 apresenta uma comparação resumida desses estudos conduzidos com

YOLO e outras arquiteturas predecessoras, a metodologia empregada no estudo, os procedimentos de aquisição de imagens e, finalmente, a aplicação final desejada.

Tabela 2.2 – Trabalhos envolvendo a detecção e classificação de insetos em contexto agrícola.

Autor	Arquitetura (baseada em)	Método	Conjunto de Dados (Fonte)	Aplicação
Tenório <i>et al.</i> (2019)	Rede Neural Convolutio-nal Profunda Hierárquica (HD-CNN) e SegNet	Comparação entre as duas técnicas e Rede Neural Convolutio-nal Profunda (DCNN) como referência	Drone	Identificação de lagartas em ambientes complexos
Tetila <i>et al.</i> (2019)	Várias arquiteturas de aprendizado profundo (DenseNet, Resnet e outras)	Comparação entre arquiteturas, com/sem ajuste fino e transfer learning	Câmera de Smartphone	Contagem de insetos em planta-ção de soja
Silveira <i>et al.</i> (2021)	YOLOv3	Avaliação do desem-penho da arquitetura para detecção de obje-tos em tempo real	Câmera de Smartphone	Detecção de vá-rios insetos em plantação de soja
Verma <i>et al.</i> (2021)	YOLO v3, v4 e v5	Comparação entre ar-quiteturas	Conjunto de dados de pesquisa agrícola	Detecção e classi-ficação de vários insetos em plan-tação de soja
Önler (2021)	YOLOv5	Comparação entre ar-quiteturas com quatro escalas diferentes (s, m, l e x)	Imagens públicas e imagens coletadas no campo (generaliza-ção)	Detecção em tempo real de lagartas e outros insetos
Ahmad <i>et al.</i> (2022)	YOLOv5	Comparação entre ar-quiteturas com cinco escalas diferentes (n, s, m, l e x)	Imagens pú-blicas (Goo-gle, Kaggle etc)	Detecção e classi-ficação de vários insetos
Khalid <i>et al.</i> (2023)	YOLO v3, YOLOv3-Tiny, YOLOv4, YOLOv4-Tiny, YOLOv6 e YOLOv8	Comparação entre ar-quiteturas	Conjunto de dados de pesquisa anterior e Câmera de Smartphone	Detecção e classi-ficação de três espécies de insetos

## 2.6 Modificações de Modelos YOLO para Detecção de Pequenos Objetos

Visando lidar com os desafios enfrentados pelos modelos YOLO na detecção precisa de objetos pequenos e suas demandas computacionais, pesquisadores têm dedicado esforços significativos envolvendo modificações e adaptações de suas arquiteturas, dando origem então a novos modelos derivados dos originais e específicos para cada missão. Essas modificações arquiteturais foram propostas para aprimorar a eficácia e a eficiência dos sistemas baseados em modelos YOLO, de modo a aprimorar a detecção de objetos pequenos e otimizar os recursos computacionais para aplicações do mundo real.

No contexto de construção de uma variante de modelo YOLO, os experimentos de ablação referem-se a testes realizados para entender o impacto de remover ou modificar componentes específicos do modelo original, como camadas, blocos de convolução, módulos de atenção, ou outras características com o objetivo de iterar sobre a arquitetura do modelo, refinando-o para atender melhor às necessidades específicas da aplicação ou para alcançar um desempenho superior.

A Tabela 2.3 resume os estudos realizados em modelos YOLO e suas modificações arquiteturais voltadas para enfrentar os desafios da detecção precisa de objetos pequenos e redução das demandas computacionais. A tabela oferece uma visão geral de diferentes trabalhos de pesquisa, incluindo as arquiteturas nas quais se baseiam, as modificações propostas, os métodos experimentais e as aplicações alvo.

Uma abordagem para aprimorar a detecção de objetos pequenos da YOLOv5 é o modelo YOLO-Z proposto por [Benjumea et al. \(2021\)](#). Esse modelo substitui elementos estruturais específicos da YOLOv5, resultando em uma melhoria de até 6,9% no mAP ao detectar objetos menores com 50% de interseção sobre união (IOU), com apenas um leve aumento no tempo de inferência. O aprimoramento da YOLOv5 com a incorporação de mecanismos de atenção, especificamente módulos de atenção de canal e espacial, mostrou um refinamento significativo no foco dos mapas de características em [Yuan et al. \(2021\)](#). Essa modificação produz resultados notáveis em conjuntos de dados de insetos, atingindo um notável mAP@0.5 de 92,5% e um F1 score de 0,91, superando o modelo YOLOv5 base com uma melhoria de 1,7% e 0,02, respectivamente. [Yuan et al. \(2022\)](#) introduziram o YOLOv5-ytiny, um modelo agregado de detecção e classificação em miniatura, obtido substituindo a camada C3 da YOLOv5 pela proposta estrutura CI, superando outros algoritmos de detecção de objetos em termos de Média de Precisões Médias (mAP). Para aprimorar a detecção de objetos pequenos em cenas capturadas por drones, [Zhan et al. \(2022\)](#) propuseram quatro métodos, incluindo redesenhar o tamanho da âncora, adicionar um módulo SE, adotar a perda CIOU e adicionar um nível de característica mais profundo. Esses métodos aumentaram significativamente o mAP no conjunto de dados VisDrone-2020, mantendo uma alta velocidade de detecção

Tabela 2.3 – Trabalhos abordando modificações em arquiteturas YOLO com vistas à detecção de pequenos objetos.

Autor	Arquitetura (baseada em)	Modificação Proposta	Método	Aplicação
Benjumea <i>et al.</i> (2021)	YOLOv5	<i>Backbone</i> : ResNet ou DenseNet. <i>Neck</i> : Pan-Net simplificado para FPN e substituído por um biFPN. Outro: mapa de características de resolução mais alta.	Experimento de ablação	Detecção e identificação de pequenos objetos em sistemas de direção autônoma
Yuan <i>et al.</i> (2021)	YOLOv5	Módulo de atenção de canal + módulo de atenção espacial (CBAM) para o <i>Neck</i>	Comparação com YOLOv5l	Detecção de vários insetos
Yuan <i>et al.</i> (2022)	YOLOv5	<i>Backbone</i> e <i>Neck</i> : Estrutura C3 substituída por CI proposto. Perda de GIoU substituída por CIoU	Comparação com modelos YOLOv4 e YOLO v5 e SSD	Identificação e classificação de agregados (concreto)
Zhan <i>et al.</i> (2022)	YOLOv5	Redesenho da âncora; módulo de <i>squeeze-and-excitation</i> (SE) incorporado na espinha dorsal ResNet50; função de perda CIoU; Expansão de nível de característica.	Experimento de ablação	Detecção de pequenos objetos capturados por imagens de UAV
Cheng <i>et al.</i> (2022)	YOLOv3	Redução de parâmetros YOLOv3; correspondência de caixa anterior via k-means++; blocos de ampulheta leves e atenção de coordenadas para extração de características aprimoradas	Experimento de ablação	Detecção de pragas de insetos
Liu <i>et al.</i> (2023)	YOLOv4	Módulo de atenção tripla para CSPDarknet53; função de perda focal; algoritmo de <i>clustering</i> K-means++ para caixas âncora.	Comparação com DPM, R-CNN, Fast R-CNN, Faster R-CNN e SSD	Detecção de insetos em tomate
Li <i>et al.</i> (2023)	YOLOv5	Função de perda PLDIoU; CBAM para o <i>Neck</i> ; algoritmo de aumento de dados mixup (entrada).	Experimento de ablação	Detecção de insetos em maracujá
Mahaur e Mishra (2023)	YOLOv5	SPP aprimorado: substituição da função de <i>pooling</i> por convolução dilatada; estrutura PANet aprimorada; novo bloco funcional (NCSP)	Experimento de ablação	Detecção e identificação de pequenos objetos em sistemas de direção autônoma
Liu <i>et al.</i> (2023)	YOLOv5	Inserção de nível de característica mais profundo; <i>pooling</i> piramidal espacial aprimorado (ESPP) do SPP; Placa de Expansão de Características Próprias (SCEP).	Experimento de ablação	Detecção de pequenos objetos capturados por imagens de UAV
Kang <i>et al.</i> (2023)	YOLOv8	Algoritmo de Reconhecimento Simulado (SA) na última camada da CNN.	Comparação com YOLOv7	Segmentação de imagem de doenças foliares
Lou <i>et al.</i> (2023)	YOLOv8	Módulo MDC para realizar operação de <i>downsampling</i> ; módulo C2f substituído pelo módulo DC; Melhorias na Fusão de Características.	Experimento de ablação	Detecção de objetos pequenos (Visdrone, VOS e Tinyperson)

de 55FPS. Outro método *lightweight* de detecção de objetos, YOLOLite-CSG, foi proposto por Cheng *et al.* (2022), projetado para dispositivos de baixo desempenho em ambientes agrícolas. Com base no YOLOLite com precisão otimizada, parâmetros reduzidos e informações espaciais aprimoradas por meio de k-means++, blocos de ampulheta e atenção de coordenadas, alcançou maior precisão e número reduzido de parâmetros. Liu *et al.* (2023) desenvolveram um algoritmo YOLOv5 aprimorado para cenas de captura por UAV, incorporando o Bloco de Aprimoramento de Recursos (FEBlock), Espaço Piramidal Espacial Aprimorado (ESPP) e Placa de Expansão Auto-Characterística (SCEP) para aprimorar a capacidade de extração de características e desempenho de detecção para objetos de médio e longo alcance. Li *et al.* (2023) propôs um algoritmo com uma nova função de perda de distância ponto-linha, módulo de atenção e aumento de dados online mixup para alcançar alto mAP, mantendo as características leves da YOLOv5 tradicional. Mahaur e Mishra (2023) introduziram a iS-YOLOv5, uma versão modificada da YOLOv5, com alterações arquiteturais como módulo SPP, estrutura PANet e caminhos de informação, resultando em uma precisão e velocidade de detecção aprimoradas. Liu *et al.* (2023) propuseram um algoritmo aprimorado de detecção de objetos para cenas de captura por YOLOv5 UAV, incorporando Bloco de Aprimoramento de Recursos (FEBlock), Pirâmide Espacial Espacial (SPP) e Placa de Expansão Auto-Characterística (SCEP) para aprimorar a capacidade de extração de características e desempenho de detecção para objetos de médio e longo alcance.

Portanto, a literatura destaca esforços contínuos para enfrentar o desafio de detectar pequenos insetos, como o PMS, indicando que o problema continua relevante e não resolvido. Trabalhos recentes sobre a YOLOv8 incluem otimização usando o algoritmo de Recozimento Simulado (SA) para detecção de pragas em culturas (Kang *et al.*, 2023), bem como uma rede de fusão de características aprimorada e nova estrutura de rede para maior precisão de detecção (Lou *et al.*, 2023).

Dadas as afinidades arquiteturais entre YOLOv5 e YOLOv8, é concebível que certas adaptações propostas para o YOLOv5 tenham potencial para modelar um YOLOv8 aprimorado, adaptado para o discernimento do PMS em campos de soja. Nesse sentido, a introdução do nível de característica mais profundo, até a camada P2, e uma modificação no módulo C2f dentro da espinha dorsal da YOLOv8 representa um passo estratégico à frente. A adição do nível de característica P2 alinha-se com estratégias bem-sucedidas vistas em modelos aprimorados da YOLOv5, como o YOLO-Z, que demonstraram melhorias na detecção de objetos pequenos. Além disso, a modificação no módulo C2f aborda a necessidade de otimizar a complexidade computacional do modelo, mantendo sua eficácia. Isso é especialmente crucial no contexto da detecção de insetos, onde precisão e eficiência são fundamentais.

### 2.6.1 Influência do Nível de Característica

Semelhante à rede YOLOv5 predecessora, a YOLOv8 emprega uma arquitetura de pirâmide de características, tratada anteriormente, na qual extrai características de diferentes escalas da imagem de entrada e as combina para gerar um conjunto de previsões de detecção. Na arquitetura YOLOv5, a camada P2 constitui uma camada de CNN que desempenha um papel fundamental no processo de detecção de objetos, responsável por receber as saídas das camadas anteriores e gerar as saídas necessárias para a camada subsequente. Mais especificamente, a camada P2 é parte de uma seção da rede denominada "*neck*" (pescoço), que integra características de baixo e alto nível extraídas de camadas anteriores da rede para obter informações mais ricas sobre a imagem.

A camada P2 recebe como entrada o tensor de saída da camada anterior. Aplica uma série de operações de convolução, normalização e ativação para gerar um novo tensor que representa características da imagem em diferentes escalas. A camada subsequente utiliza então essa saída para gerar as saídas finais que representam detecções de objetos na imagem. Podemos afirmar que as camadas P2 são especificamente projetadas para extrair características em uma escala mais minuciosa, o que pode auxiliar na identificação de objetos pequenos, que outras camadas da rede podem negligenciar. Por outro lado, ao adicionar a camada P2, a rede torna-se mais profunda, com mais camadas de computação. Essa profundidade adicional pode ajudar a rede a aprender representações complexas da imagem de entrada e extrair características mais informativas, resultando em um desempenho aprimorado em tarefas de detecção de objetos pequenos, como o PMS. Como parte deste trabalho, uma camada P2, até então não implementada, foi proposta para ser acoplada à YOLOv8.

### 2.6.2 Influência do Número de Filtros

A redução do número de filtros no modelo de detecção de objetos YOLOv8 pode acarretar diversos efeitos esperados em seu desempenho e características. O número de filtros nas camadas convolucionais do modelo impacta diretamente sua capacidade de aprender e representar características a partir dos dados de entrada.

Um dos efeitos mais imediatos da redução do número de filtros é a diminuição da complexidade do modelo. O número de parâmetros e gradientes será reduzido, resultando em um modelo mais leve. Isso pode ser benéfico em cenários onde os recursos computacionais são limitados, tornando o modelo mais eficiente para implementação em dispositivos com restrição de recursos computacionais (Zhang; Zhong; Li, 2019). Embora um modelo menor possa demandar menos potência computacional, também é provável que resulte em uma diminuição na precisão de detecção, limitando a capacidade do modelo de representar efetivamente uma ampla gama de características de objetos. Isso poderia ocasionar dificuldades na detecção de objetos com escalas, formas ou orientações variadas, embora isso não

pareça ser um problema imediato, considerando a baixa variabilidade do PMS, quando nos limitamos à detecção apenas das fases mais avançadas do inseto, como é o caso.

Por outro lado, um modelo menor pode generalizar melhor em dados de domínios ou categorias previamente não observados. Com uma capacidade reduzida de memorização dos dados de treinamento, o modelo pode concentrar-se na aprendizagem de características mais genéricas que são úteis em diferentes conjuntos de dados (Seemakurthy *et al.*, 2022). Como veremos mais adiante, este é um objetivo que buscamos em nosso modelo.

A redução no número de filtros pode resultar em tempos de inferência mais rápidos. O modelo pode processar imagens de forma mais ágil, sendo adequado para aplicações em tempo real ou cenários que exigem detecção rápida, como a detecção e contagem de PMSs em tempo real. No entanto, é necessário ter cuidado, pois se o número de filtros for significativamente reduzido, o modelo pode tornar-se mais suscetível ao *overfitting* nos dados de treinamento e pode enfrentar dificuldades em capturar padrões complexos, resultando em perda de desempenho ao lidar com novos exemplos não vistos (Diwan; Anirudh; Tembhurne, 2023).

## 2.7 Métodos de Rastreamentos de Objetos

A detecção e rastreamento de objetos em tempo real são tarefas fundamentais em diversas aplicações, incluindo a vigilância aérea por meio de drones e câmeras de trânsito. Essas tarefas podem também ser viáveis para emprego como parte de sistemas de mapeamento em tempo real, como aquele exposto na Figura 1.2.

Trabalhos recentes envolvendo a detecção e o rastreamento de animais pequenos vêm sendo desenvolvidos, conforme atesta a literatura. Leocádio, Segundo e Pessin (2023) compararam a YOLOv7 e YOLOv8 como modelos para classificação e detecção de abelhas jataí, comparando o desempenho em rastreamento através da aplicação subsequente de distância Euclidiana e ByteTrack. Já Qian, Chen e Jiang (2023), propuseram seu detector baseado em YOLOv5 e o acoplaram com um rastreador, também baseado em ByteTrack, para o rastreamento de trajetórias de peixes. Também Cao *et al.* (2020) desenvolveram um *framework* de rastreamento on-line de vários objetos combinando as informações de movimento e aparência das formigas usando o modelo ResNet para rastrear o movimento de formigas. Por fim, Pani *et al.* (2021) treinou um método iterativo fracamente supervisionado, devido ao número reduzido de imagens no conjunto de dados, e o acoplou com um módulo de rastreamento de dois estágios para rastrear múltiplas vespas *Trichogramma*.

Dentre os principais algoritmos de rastreamento disponíveis, apresentam-se o DeepSORT e o ByteTrack, cujas características estão sintetizadas na Tabela 2.4.

Tabela 2.4 – Comparativo entre os algoritmos de rastreamento de objetos DeepSORT e ByteTrack. Inspirado em: [Aadi, Sadiq e Labd \(2023\)](#)

Algoritmo	Acurácia	Velocidade	Flexibilidade	Fraquezas
DeepSORT	✓			occlusão
ByteTrack		✓	✓	occlusão, associação de falha

Em cenários que demandam detecção e rastreamento de objetos em tempo real, como operações de mapeamento com drones, a velocidade de processamento é essencial. O ByteTrack demonstra na literatura tempos de inferência mais rápidos do que o DeepSORT ([Aadi; Sadiq; Labd, 2023](#)), garantindo uma resposta ágil em situações dinâmicas. Drones e sistemas de mapeamento em tempo real frequentemente operam em ambientes com recursos computacionais e energéticos limitados. O ByteTrack se destaca por sua arquitetura leve e eficiente, requerendo menos recursos em comparação com o DeepSORT ([Munawar, 2023](#)), tornando-se uma escolha atraente para implementações embarcadas, onde a eficiência energética é crucial, em que pese a maior acurácia obtida alcançada pelo DeepSORT.

O ByteTrack estabelece conexões entre trilhas e caixas de alta pontuação, utilizando o filtro de Kalman para lidar com desafios como desfoque de movimento ou oclusão. Esse filtro extrapola o estado atual dos quadros com base em estimativas anteriores, garantindo rastreamento preciso de objetos ao longo do tempo. Com sua abordagem baseada em redes neurais siamesas e técnicas de *data augmentation*, o ByteTrack tem se mostrado robusto na detecção e rastreamento de objetos em condições desafiadoras, como ocultações e variações de iluminação. Sua eficácia o torna adequado para identificar e monitorar insetos, como proposto para avanços futuros. Sua arquitetura simplificada facilita sua implementação e manutenção, inclusive para usuários com menos experiência em visão computacional, o que é relevante em contextos que demandam implantação rápida de sistemas de detecção em tempo real.

As características do ByteTrack o tornam vantajoso, distribuindo eficientemente recursos computacionais e alcançando altas velocidades de rastreamento ([Zhang et al., 2022](#)). Isso possibilita rastreamento e contagem rápidos e precisos de insetos em ambientes agrícolas a partir de vídeos de câmeras em movimento.



## 3 Material e Métodos

O capítulo aborda diversos aspectos relacionados à detecção e contagem de percevejos-marrom-da-soja (PMS) em plantações de soja, utilizando um conjunto de dados denominado INSECT10K7C. Primeiramente, é detalhada a composição e o processo de preparação desse conjunto de dados, incluindo a coleta das imagens, as etapas de anotação manual, pré-processamento e aumento de dados. Em seguida, é apresentada uma proposição para um modelo aprimorado de detecção de objetos, baseado na arquitetura YOLOv8, com modificações específicas para melhorar a detecção de alvos pequenos, como os PMSs. A estrutura proposta inclui a adição de um novo nível de característica à arquitetura YOLOv8 e uma modificação na camada C2f da arquitetura original, denominada C2f2, visando equilibrar a complexidade do modelo. Além disso, são discutidas as métricas de avaliação do modelo de detecção, incluindo Precisão, Revocação, mAP0.5 e mAP0.5:0.95, bem como métricas de complexidade do modelo, como número de parâmetros e FLOPs. O capítulo também descreve a validação cruzada do modelo proposto, aplicando o modelo de detecção em um *framework* de rastreamento e contagem de PMS em *stream* de vídeo, construído sobre um novo conjunto de dados de imagens de PMSs, EMBRAPA\_EUSCHISTUS, destacando o uso do algoritmo ByteTrack e as métricas de avaliação associadas, como MOTA. Por fim, são detalhados os recursos computacionais empregados na pesquisa, incluindo ferramentas de anotação, treinamento e inferência e processamento do vídeo.

### 3.1 Base de Dados

O presente estudo empregou uma base de dados composta por imagens de plantações coletadas em uma área agrícola de soja em Dourados-MS, Brasil, denominada INSECT10K7C e disponível em (Tetila, 2019). O conjunto de dados é composto por 1.000 imagens digitais capturadas localmente pelos pesquisadores, utilizando uma câmera digital equipada com um sensor Samsung SM-G930F de resolução de 12,2 megapixels e tamanho de 1/2,5". As imagens foram registradas a uma altitude de 1 metro acima da plantação, com um ângulo aproximado de 45° da câmera para o solo. A coleta das imagens ocorreu durante as fases fenológicas R1–R6 do ciclo reprodutivo da soja na safra 2018/19, em dias distintos e em condições climáticas diversas, entre 18h e 19h30.

O conjunto de dados original utilizado neste estudo consistiu de 903 imagens com pelo menos uma anotação em cada. Essas imagens iniciais foram primeiramente anotadas manualmente, utilizando uma única classe "inseto", que englobava completamente o contorno visível dos PMSs. Posteriormente, aplicamos diversos passos de pré-processamento e *data augmentation* às imagens anotadas, com o intuito de aprimorar a diversidade do

conjunto de dados e aumentar seu tamanho para um treinamento mais eficaz do modelo. Essas etapas resultaram na geração de pontos de dados adicionais, expandindo o conjunto de dados para um total de 4.000 imagens. Ao final dessas operações, o conjunto de dados foi obtido conforme ilustrado na Figura 3.1.

Durante a fase de pré-processamento, técnicas como auto-orientação, redimensionamento, divisão em telhas e filtragem foram empregadas para padronizar e enriquecer o conjunto de dados. Essas etapas garantiram que as imagens estivessem consistentemente orientadas<sup>2</sup>, possuísem um tamanho uniforme de 640x640 pixels e contivessem uma quantidade substancial de dados anotados, atendendo ao limiar de 90% de anotação.

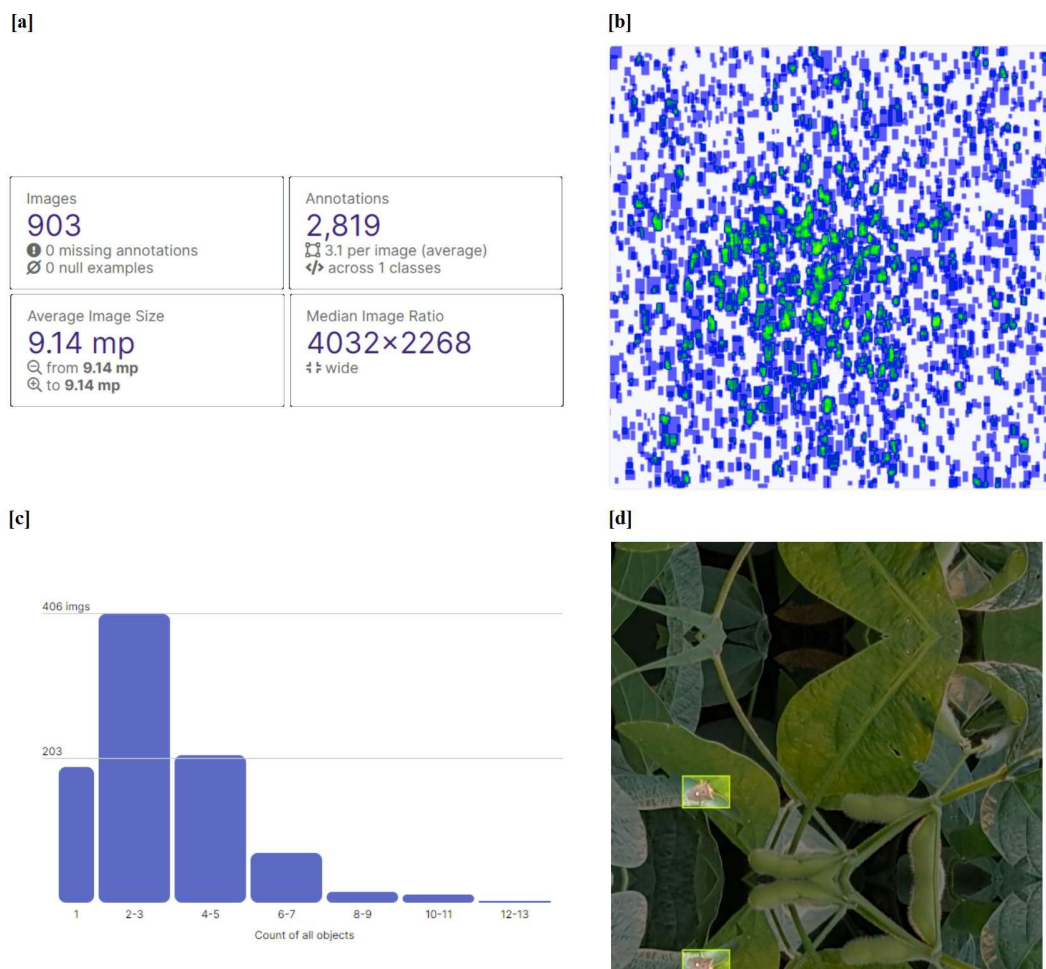


Figura 3.1 – [a] Dados principais do conjunto de dados INSECT10K7C640\_SAT; [b] Distribuição espacial de PMS nas imagens do conjunto de dados; [c] Contagem de insetos nas imagens e [d] Imagem de amostra no conjunto de dados, compreendendo a cena com PMSs anotados após o pré-processamento.

Ao introduzir ajustes aleatórios de saturação para cada exemplo de treinamento,

<sup>2</sup> Trata-se de uma questão de pós-processamento. Em uma aplicação real bastaria que a orientação nadiral fosse garantida com o uso de um gimbal.

o número de saídas por exemplo foi aumentado para dois. Essa estratégia de aumento introduziu variações na intensidade de cor, contribuindo para um conjunto de dados de treinamento mais abrangente e robusto.

O conjunto de treinamento, que compreende a maioria dos dados (82% das 4.000 imagens), é utilizado para treinar o modelo e ajustar seus parâmetros, permitindo que ele aprenda a partir de uma variedade diversificada de exemplos e padrões nos dados. O conjunto de validação (13%), composto por 607 imagens, é utilizado durante o treinamento para ajustar hiperparâmetros e avaliar o desempenho do modelo em dados não vistos, auxiliando na prevenção de *overfitting* e validação do modelo treinado. Por fim, o conjunto de teste (5%), totalizando 260 imagens, serve como uma avaliação independente do desempenho do modelo em dados completamente não vistos, fornecendo uma medida confiável de sua eficácia no mundo real e indicando capacidade de generalização.

## 3.2 Proposição do Modelo de Detecção

O modelo original YOLOv8 é altamente eficaz; no entanto, ainda enfrenta desafios na detecção precisa de alvos pequenos, como o PMS, sendo esse um desafio de pesquisa relevante, conforme a literatura exposta anteriormente. A questão reside principalmente no processo de extração de características, onde características maiores muitas vezes ofuscam alvos pequenos. As características extraídas carecem de informações sobre alvos pequenos, resultando em resultados de detecção inadequados. Além disso, alvos pequenos têm maior probabilidade de sobrepor-se a outros objetos, tornando-os mais difíceis de distinguir e localizar na imagem (Lou *et al.*, 2023). Para superar esses problemas, propomos um algoritmo de detecção aprimorado, cuja estrutura é apresentada na Figura 3.2, capaz de detectar alvos de pequeno porte, como o *Euschistus Heros*, sendo também um modelo mais leve e rápido que o YOLOv8 original.

A arquitetura YOLOv8 oferece uma variedade de estruturas de rede, incluindo YOLOv8n, YOLOv8s, YOLOv8m, YOLOv8l e YOLOv8x. Embora difiram em largura e profundidade, seguem os mesmos princípios e podem ser escolhidos de acordo com necessidades específicas. Quanto mais profunda a estrutura, maior a precisão, mas menor a velocidade de treinamento e inferência. Para priorizar a velocidade sem comprometer a precisão, escolhemos YOLOv8n como estrutura base, com aprimoramentos adicionais para melhorar o desempenho.

A adição do nível P2 à arquitetura YOLOv8 torna a rede mais profunda, pois uma camada adicional é incorporada à estrutura global da rede. A arquitetura YOLOv8 já inclui uma série de camadas convolucionais e uma seção de *neck* que combina características de diferentes escalas. Ao adicionar o nível P2, conforme detalhado na Tabela 3.1, de maneira semelhante ao YOLOv5 (Benjumea *et al.*, 2021; Zhan *et al.*, 2022), espera-se que a rede se torne

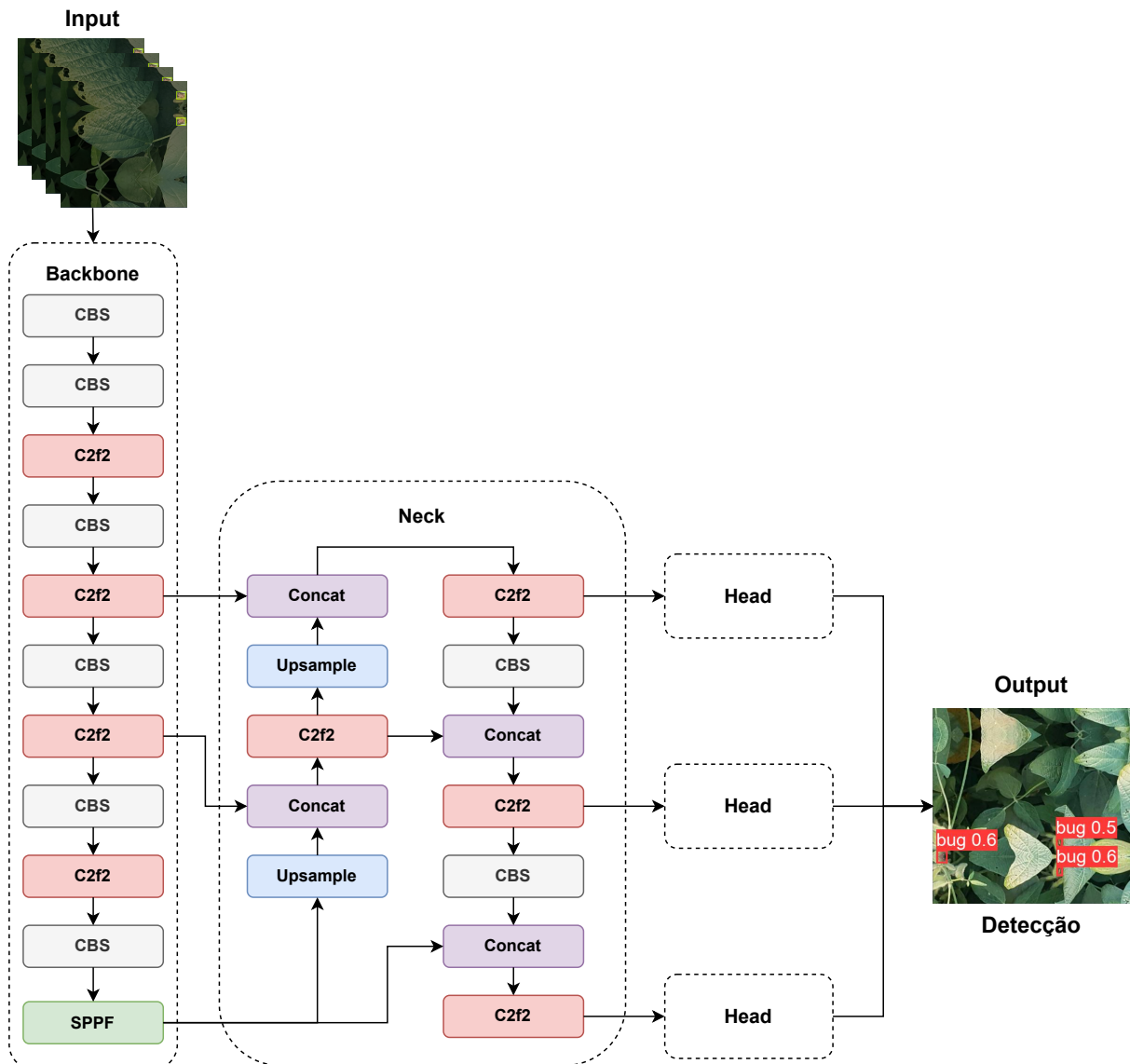


Figura 3.2 – Representação da estrutura do modelo YOLO aprimorado. Adaptado de: Lima *et al.* (2024)

mais profunda, com mais camadas de computação. Essa maior profundidade pode auxiliar a rede a aprender representações complexas da imagem de entrada e extrair características mais informativas, levando a um desempenho aprimorado em tarefas de detecção de objetos, especialmente para objetos pequenos.

Vale ressaltar, no entanto, que aumentar a profundidade da rede também apresenta potenciais desvantagens, como aumento da complexidade computacional e um maior risco de overfitting do modelo aos dados de treinamento. Portanto, adicionar a camada P2 à arquitetura YOLOv8 é um compromisso que precisa ser cuidadosamente equilibrado com outras considerações, como tamanho do modelo e requisitos de desempenho. Nesse sentido, propomos neste trabalho uma modificação na camada C2f, devido à sua importância na arquitetura geral do YOLOv8, no sentido de torná-la mais leve e contrabalancear o aumento

Tabela 3.1 – Estrutura detalhada da YOLOv8n com nível P2.

de	repetições	módulo	argumentos
-1	1	Conv	[64, 3, 2]
-1	1	Conv	[128, 3, 2]
-1	3	C2f2	[128, True]
-1	1	Conv	[256, 3, 2]
-1	6	C2f2	[256, True]
-1	1	Conv	[512, 3, 2]
-1	6	C2f2	[512, True]
-1	1	Conv	[1024, 3, 2]
-1	3	C2f2	[1024, True]
-1	1	SPPF	[1024, 5]
-1	1	Upsample	[None, 2, "nearest"]
(-1, 6)	1	Concat	[1]
-1	3	C2f2	[512]
-1	1	Upsample	[None, 2, "nearest"]
(-1, 4)	1	Concat	[1]
-1	3	C2f2	[256]
-1	1	Upsample	[None, 2, "nearest"]
(-1, 2)	1	Concat	[1]
-1	3	C2f2	[128]
-1	1	Conv	[128, 3, 2]
(-1, 15)	1	Concat	[1]
-1	3	C2f2	[256]
-1	1	Conv	[256, 3, 2]
(-1, 12)	1	Concat	[1]
-1	3	C2f2	[512]
-1	1	Conv	[512, 3, 2]
(-1, 9)	1	Concat	[1]
-1	3	C2f2	[1024]
(18, 21, 24, 27)	1	Detect	[nc]

obtido com a adição da camada de nível de recurso P2. Em nosso trabalho, batizamos essa nova estrutura como C2f2.

A camada C2f2 difere da C2f na medida em que possui um número diferente de filtros nos blocos *bottleneck*, mas possui uma estrutura de rede semelhante, como mostrado na Figura 3.3. Ambas as implementações utilizam o bloco *Bottleneck* CSP, que incorpora comumente duas convoluções encontradas em arquiteturas de redes neurais convolucionais. O bloco consiste em uma camada de convolução 1x1, uma camada de convolução *depthwise* 3x3 e outra camada de convolução 1x1. O tensor de entrada é dividido em duas partes iguais, com a primeira parte passando pela primeira convolução 1x1 e sendo dividida novamente. A segunda parte passa por módulos *Bottleneck*, que consistem em duas camadas de convolução separáveis e uma conexão opcional de atalho. As duas partes divididas e as saídas dos

módulos *Bottleneck* são concatenadas e passam pela segunda camada de convolução 1x1 para produzir a saída do bloco. Um parâmetro 'n' determina o número de módulos *Bottleneck*, neste caso 1.

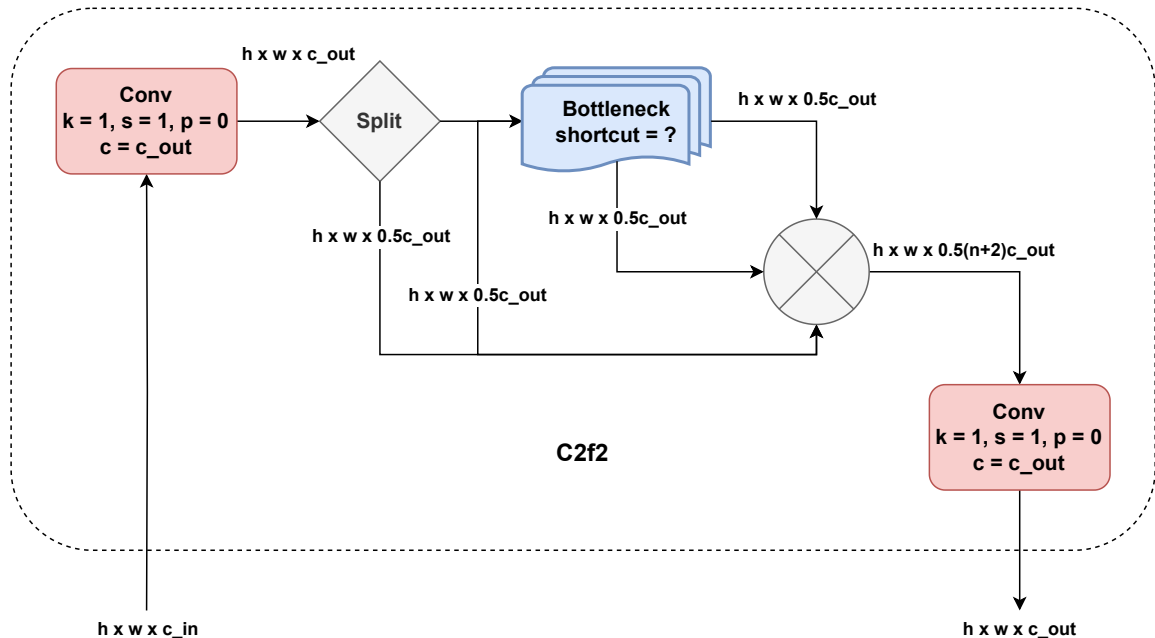


Figura 3.3 – Estrutura da camada C2f2.

Neste estudo, foram realizados experimentos de ablação para avaliar o impacto de diferentes módulos (nível P2 e camada C2f2) no desempenho do algoritmo de detecção de PMSs nas mesmas condições experimentais. Para esse fim, o novo algoritmo foi treinado e testado no conjunto de dados INSECT10K7C640\_SAT e comparado com: A) YOLOv8n, B) YOLOv8n apenas com C2f2 e C) YOLOv8n apenas com P2. Escolhemos a versão 8.0.99 do YOLOv8n como modelo de referência para os experimentos de ablação. A resolução da imagem de entrada foi definida como  $640 \times 640$ , e foram treinadas 100 épocas com um tamanho de lote de 150.

### 3.3 Validação Cruzada do Modelo

Utilizamos, também, um conjunto inédito de imagens de PMS para avaliar se o modelo proposto de detecção de objetos, treinado com imagens do conjunto de dados INSECT10K7C640\_SAT, funcionaria sob condições diferentes daquelas nas quais foi treinado e também para testar a capacidade de contagem do sistema proposto em um conjunto inédito. Esse conjunto é composto por um total de 42 imagens de *Euschistus Heros* em plantações de soja, obtidas entre fevereiro e março de 2021, na área experimental da Embrapa em Santo Antônio de Goiás, Goiás, Brasil. As imagens, que denominamos neste estudo como EMBRAPA\_EUSCHISTUS, Figura 3.4, foram capturadas usando a câmera traseira de um

smarVPhone modelo G9 - LG a uma altura de 1,2 a 1,5 m do solo, em uma orientação de aproximadamente 45°.

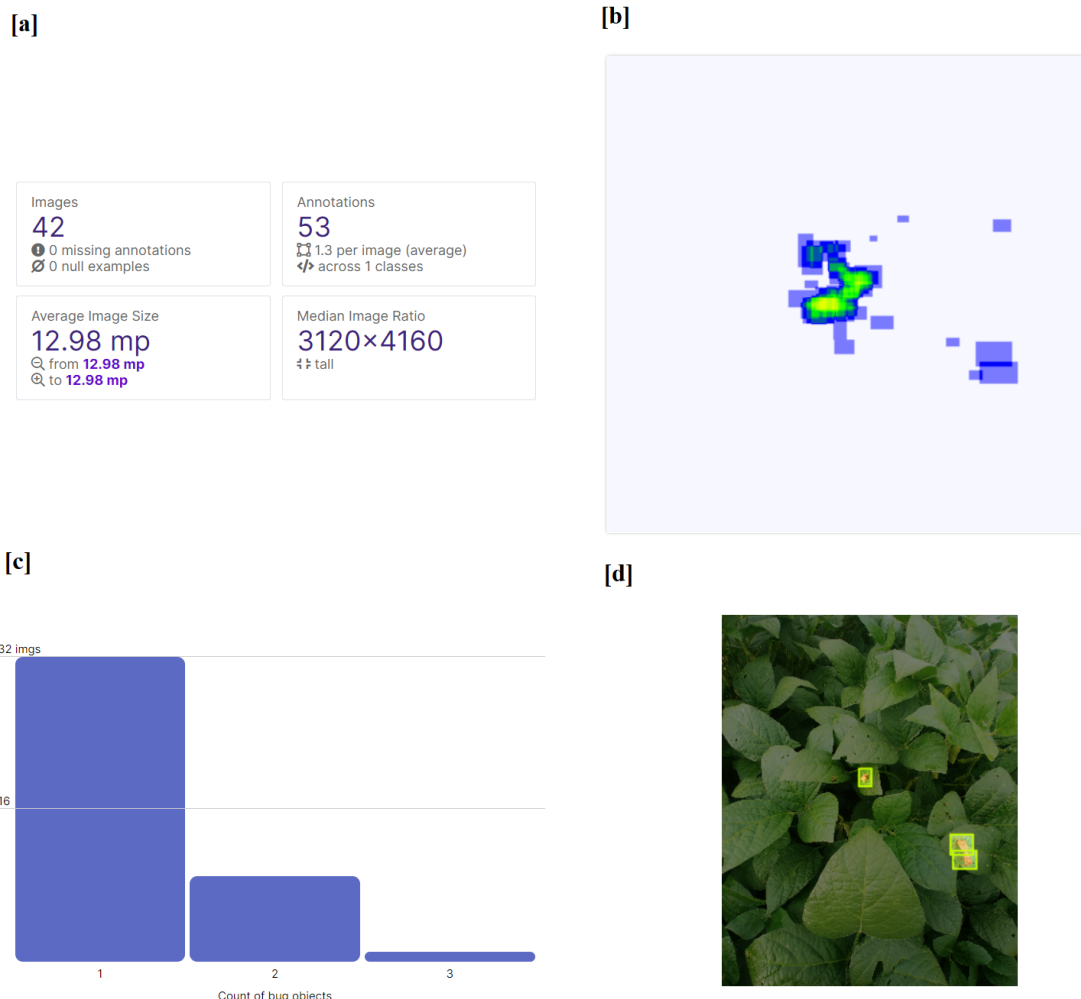


Figura 3.4 – [a] Dados principais do conjunto de dados EMBRAPA\_EUSCHISTUS; [b] Distribuição espacial de PMS nas imagens do conjunto de dados; [c] Contagem de insetos nas imagens e [d] Imagem de amostra no conjunto de dados, compreendendo a cena com PMS anotados após o pré-processamento.

Esse novo conjunto de dados foi convertido em um vídeo, permitindo também testar a capacidade de rastreamento e contagem do método proposto. Ajustamos o modo de transição dos slides usando o efeito "push", de modo que cada transição durasse 3 segundos e, uma vez concluída, avançaria automaticamente para um novo slide. As imagens ocupavam toda a tela de vídeo 1:1, resultando em um vídeo com duração de 2 minutos e 7 segundos. O controle preciso sobre efeitos de transição e cronometragem permitiu transições contínuas e fluidas entre imagens, imitando a fluidez de um vídeo obtido a partir de um drone. Embora essa abordagem possa não abranger todas as capacidades de drones, ela representa uma alternativa criativa e econômica. Por fim, o vídeo foi dividido em frames a uma taxa de 5 FPS, compatível com a análise do rastreamento e contagem que queremos realizar, nos

aprofundando no nível do frame.

### 3.3.1 Aplicação do Modelo de Detecção: Rastreamento e Contagem de PMSs em *Stream* de Vídeo

No contexto do rastreamento e da contagem de insetos em plantios de soja utilizando *streams* de vídeo, o algoritmo ByteTrack, introduzido por (Zhang *et al.*, 2022), surge como uma ferramenta valiosa, conforme tratado previamente. Especificamente desenvolvido para sequências de vídeo, o ByteTrack categoriza eficientemente caixas de detecção em classificações de alta e baixa pontuação, retendo informações abrangentes.

O processo de aplicação do ByteTrack, cujo código se encontra no [Repositório](#) do trabalho, para a contagem dos percevejos via stream de vídeo pode ser descrito em várias etapas:

No início do processo, são realizados ajustes iniciais para preparar o ambiente de trabalho. Isso inclui definir o diretório de trabalho e o caminho do vídeo de origem. Além disso, é instalada a biblioteca Roboflow Supervision para auxiliar na manipulação e processamento de dados. Também é carregado o modelo pré-treinado YOLOv8, que será utilizado para realizar as previsões nos quadros de vídeo.

Uma vez realizados os ajustes iniciais, o processo prossegue com a predição e anotação de um único quadro de vídeo. Primeiramente, são definidas as classes de interesse e criado um gerador de quadros de vídeo. Em seguida, o modelo YOLOv8 realiza a predição em um único quadro de vídeo, e as detecções resultantes são convertidas para o formato esperado pela biblioteca Supervision. Por fim, os quadros são anotados com as detecções correspondentes e o resultado é exibido.

A etapa seguinte envolve a predição e anotação de todo o vídeo, onde o modelo é aplicado em todos os quadros do vídeo de origem. Para isso, são definidas as zonas de interesse e instanciados os objetos necessários para a anotação, como a criação de uma função de *callback* para processar cada quadro do vídeo. Durante o processamento, o modelo realiza a predição em cada quadro e as detecções são anotadas conforme necessário. Ao final deste processo, obtém-se o vídeo anotado e processado, pronto para ser utilizado na contagem de PMSs.

## 3.4 Métricas de Desempenho

### 3.4.1 Métricas de Eficácia do Modelo de Detecção

Para avaliar o desempenho dos modelos abatidos, utilizamos seis métricas, a saber, Precisão (Eq. 3.1), Revocação (Eq. 3.2), mAP0.5 e mAP0.5:0.95, relacionadas à Eq. 3.3, Params(M), Flops(G). Observamos também as métricas de Inferência(ms) e Tempo(h).



### 3.4.1.1 Precisão

A medida de precisão para um modelo de visão computacional é uma métrica que avalia a proporção de predições corretas feitas pelo modelo em relação ao total de predições feitas. Essa medida é particularmente relevante quando estamos lidando com problemas de classificação, onde o modelo tenta atribuir rótulos ou categorias a determinadas imagens.

Matematicamente, é expressa pela Eq. 3.1.

$$\text{Precisão} = \frac{VP}{VP + FP} \quad (3.1)$$

### 3.4.1.2 Revocação

A revocação mede a proporção de exemplos positivos que foram corretamente identificados pelo modelo em relação ao total de exemplos positivos presentes nos dados de teste. Em outras palavras, a revocação é a capacidade do modelo de encontrar todos os exemplos relevantes e é calculado de acordo com a Eq. 3.2.

$$\text{Revocação} = \frac{VP}{VP + FN} \quad (3.2)$$

### 3.4.1.3 mAP

Para o caso específico de detecção de insetos em lavouras de soja, a medida de mAP é uma métrica fundamental para avaliar o desempenho do modelo de detecção. A curva de precisão-revocação mostra como a precisão varia conforme a revocação é ajustada. À medida que o modelo detecta mais insetos (aumento da revocação), a precisão pode diminuir, pois o modelo pode começar a obter mais falsos positivos. O mAP é então calculado pela área sob a curva de precisão-revocação, conforme a Figura 3.5, representando a média das precisões médias (AP) para diferentes níveis de revocação, tal como matematicamente descrito pela Eq. 3.3. Isso nos fornece uma medida abrangente do desempenho do modelo na detecção de insetos na lavoura, levando em conta essa compensação entre precisão e revocação.

A diferença entre mAP0.5 e mAP0.5:0.95 refere-se à variação do mAP dependendo do limiar de confiança usado para considerar uma detecção como correta, sendo que, para mAP0.5 o valor do *threshold* de confiança é fixado em 0.5. Ou seja, as detecções feitas pelo algoritmo são consideradas corretas apenas se estiverem associadas a uma probabilidade de confiança igual ou superior a 0.5 (50%), sendo mais utilizada em casos onde é necessário um equilíbrio entre a precisão e a taxa de detecção. Já o mAP0.5:0.95 é o mAP é calculado para uma variedade de *thresholds* de confiança, variando de 0.5 a 0.95, com intervalos regulares. Isso permite uma análise detalhada do desempenho do algoritmo em uma ampla gama de níveis de confiança. O mAP0.5:0.95 fornece uma visão completa do desempenho

do algoritmo, desde detecções mais confiáveis (0.95) até detecções menos confiáveis (0.5), oferecendo insights sobre como o algoritmo se comporta em diferentes cenários.

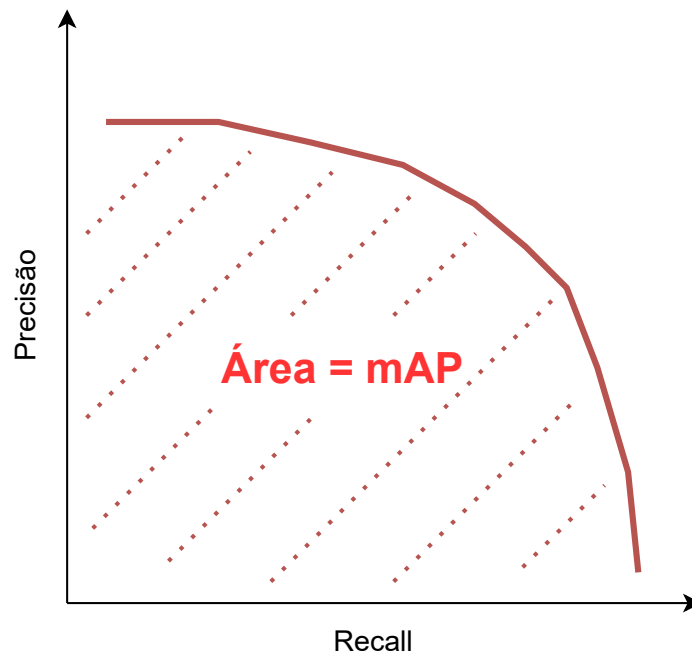


Figura 3.5 – Curva de Precisão-Revocação

$$\text{mAP} = \frac{1}{k} \sum_{i=1}^k \text{AP}_i \quad (3.3)$$

#### 3.4.1.4 Métricas de Eficiência Computacional do Modelo

Consideramos também o número de parâmetros (Params.) e FLOPs para avaliar o modelo. O número de parâmetros em um modelo de rede neural refere-se ao número de variáveis aprendidas utilizadas para fazer previsões. É uma métrica importante para avaliar a complexidade do modelo e sua eficiência computacional. Modelos com mais parâmetros geralmente exigem mais recursos para treinamento e inferência, mas podem apresentar maior precisão. FLOPS é o acrônimo que significa "Operações de Ponto Flutuante por Segundo". Mede quantas operações aritméticas de ponto flutuante um modelo pode realizar por segundo. É uma indicação de quão eficiente computacionalmente o modelo é.

Parâmetros de desempenho, como tempo de inferência (em ms) e tempo de treinamento (em horas), são métricas úteis para avaliar a eficiência computacional dos modelos YOLO. O tempo de inferência mede a velocidade com que o modelo processa dados de entrada e gera previsões de saída. É importante minimizar o tempo de inferência para aplicações em tempo real, onde a velocidade é crucial, como detecção de objetos em tempo real em vídeos. Por outro lado, o tempo de treinamento mede quanto tempo o modelo leva para

aprender com os dados de treinamento e melhorar sua precisão. Essa métrica é essencial para medir a eficiência geral de treinamento de modelos YOLO. Considerar esses dois parâmetros ao selecionar um modelo YOLO para um caso de uso específico é fundamental, pois podem impactar o desempenho e o custo computacional do modelo.

### 3.4.2 Métricas de Avaliação do *Framework* de Rastreamento e Contagem

Para contagem de insetos em stream de vídeo, as métricas são baseadas nas diferenças registradas entre cada quadro do vídeo. As diversas situações possíveis para registro dessas métricas estão resumidas na Figura 3.6. Um verdadeiro positivo (VP) é considerado quando um novo inseto (quando consideramos a passagem do quadro  $k$  ao  $k + 1$ ) recebe um novo ID. Um falso negativo (FN) ocorre quando um novo inseto no vídeo não recebe um novo ID, ou sua trajetória é interrompida entre os quadros  $k$  e  $k + 1$ . Um falso positivo (FP) é computado quando um inseto previamente identificado recebe outro ID simultaneamente com o ID atribuído anteriormente. Uma troca de ID (IDS) ocorre quando um inseto altera seu ID entre o quadro  $k$  e o quadro subsequente  $k + 1$ . Finalmente, um verdadeiro negativo (VN) ocorre quando corretamente o modelo não realiza uma nova contagem de insetos. Vale a pena também ressaltar que cada um desses eventos em cena é registrado como *Ground Truth* (GT), de forma que quando há 2 insetos no local temos um GT igual a 2. O mesmo ocorre quando há um ID duplicado, ou seja, um FP e um VP simultaneamente.

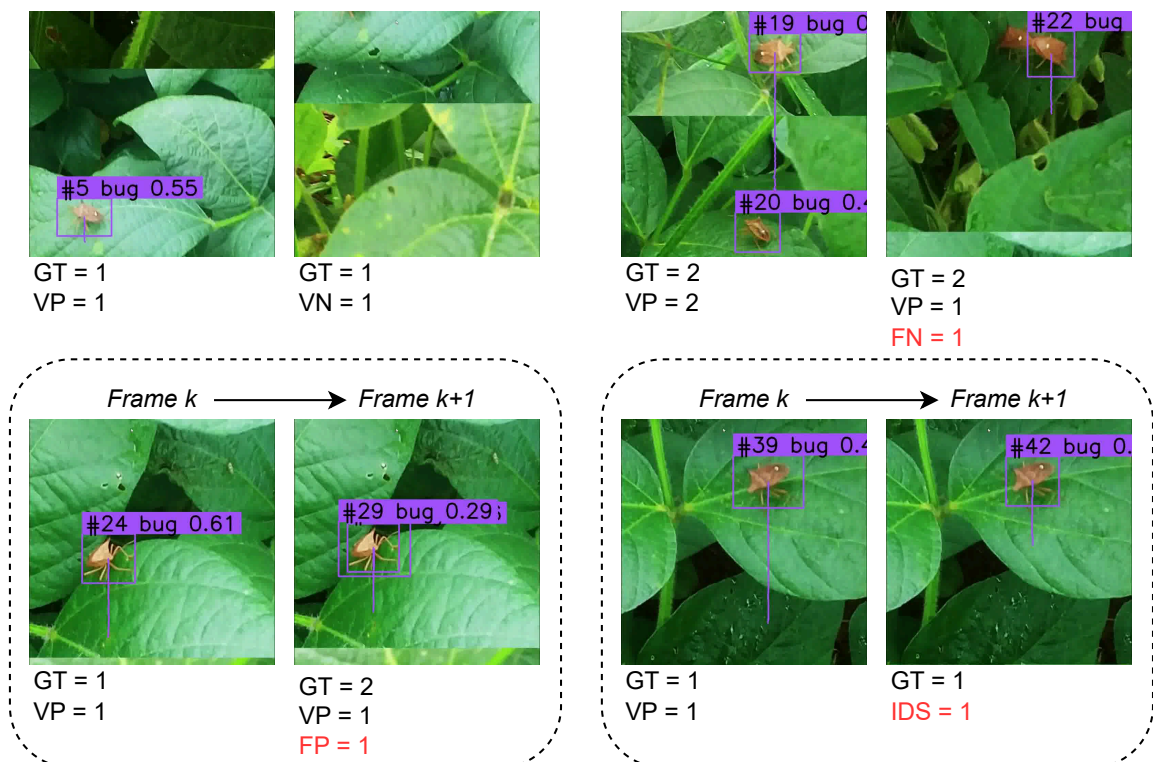


Figura 3.6 – Resumo das métricas e suas possibilidades de registro no problema de rastreamento para posterior aplicação na contagem de insetos. Adaptado de: Lima *et al.* (2024)

Com essas métricas primárias, calculamos a precisão do rastreamento multiobjeto (MOTA) (Bernardin; Elbs; Stiefelhagen, 2006) do experimento, uma das principais métricas de avaliação para o desempenho de rastreamento de objetos, calculada conforme a Eq. 3.4. MOTA varia de  $-\infty$  a 1 e pode ser multiplicado por 100 para obter MOTA em termos de porcentagem. A qualidade do rastreamento é considerada mais alta à medida que o valor de MOTA se aproxima de 1 e é considerada deficiente quando esse valor é menor ou igual a 0.

$$\text{MOTA} = 1 - \frac{\text{FN} + \text{FP} + \text{IDS}}{\text{GT}} \quad (3.4)$$

### 3.5 Recursos Computacionais Empregados

As imagens da base de dados INSECT10K7C640 foram anotadas manualmente utilizando a ferramenta *web Make Sense* (MAKE SENSE, 2019). Pré-processamento e *data augmentation* foram aplicados às imagens anotadas, utilizando ferramentas disponíveis na plataforma *Roboflow* (ROBOFLOW, 2020).

Para os treinamentos e inferências do modelo empregou-se o *Google Colaboratory* (Colab), uma plataforma de codificação baseada em navegador que oferece recursos gratuitos de GPU (GOOGLE, 2017). Especificamente, utilizamos a versão *Google Colab Pro+*, a qual oferece acesso prioritário a GPUs mais poderosas e máquinas virtuais com maior memória em comparação com a versão gratuita. Utilizamos a GPU NVIDIA A100-SXM4-40GB, uma GPU de alto desempenho com 40GB de memória. Para a avaliação cruzada sobre o novo conjunto de dados, o tamanho das imagens foi fixado em 640 px, o mesmo no qual os modelos foram treinados. Para a geração dos gráficos a serem analisados no próximo capítulo, transferiu-se os dados dos extratos produzidos pelo *Google Colaboratory* para o *Microsoft Excel*.

O conjunto de dados EMBRAPA\_EUSCHISTUS foi convertido em vídeo através do recurso de apresentação de slides do programa *Microsoft PowerPoint*, conforme ilustrado na Figura 3.7.

O vídeo passou também pela ferramenta de recorte do *Windows 11* e foi salvo no formato *mp4*. Por fim, o vídeo foi dividido em frames a uma taxa de 5 FPS, utilizando a aplicação web ASPOSE (ASPOSE, 2024).

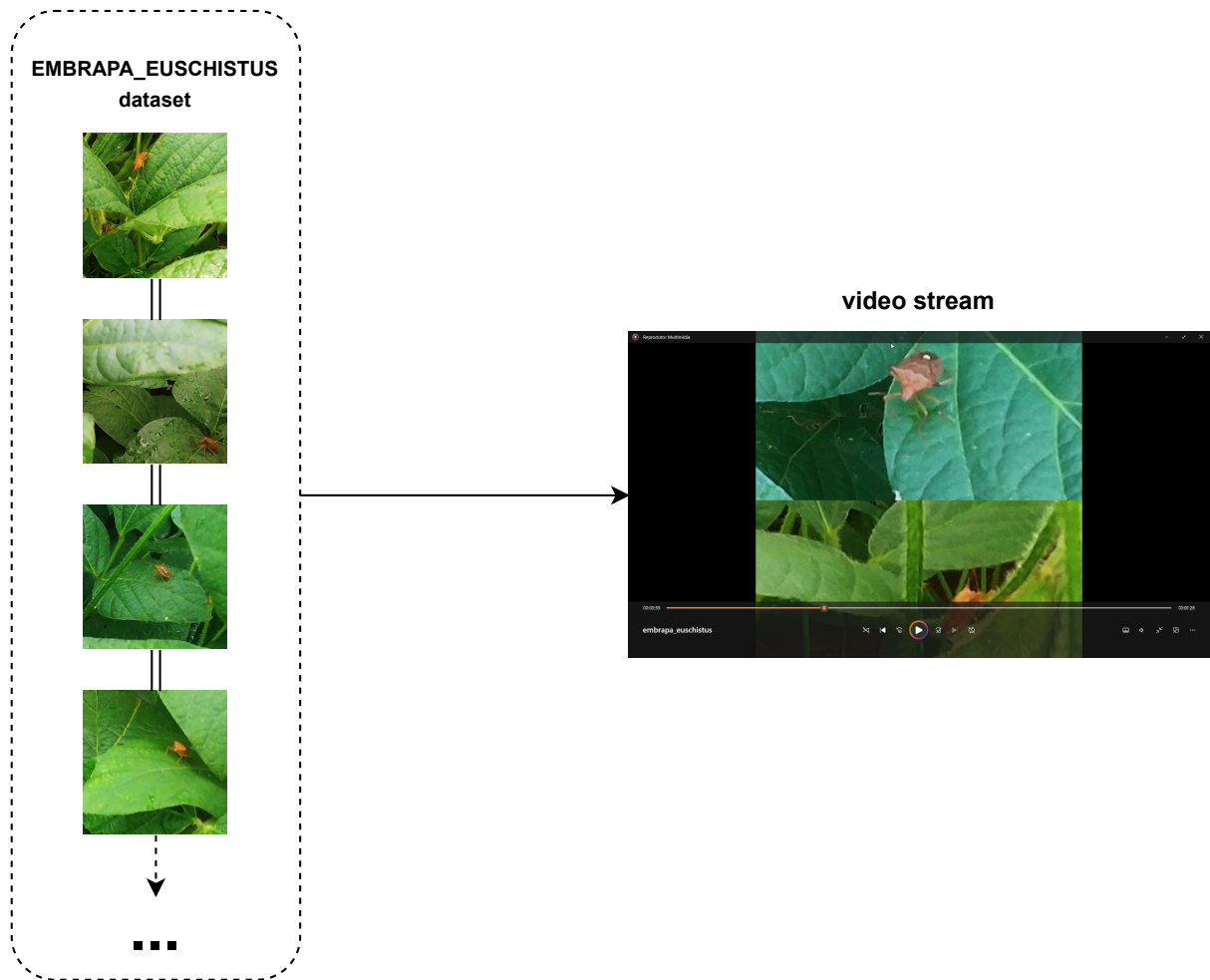


Figura 3.7 – Esquema geral do processo de transformar imagens em um vídeo dinâmico simulando vídeos obtidos por drone.

## 4 Análise do Modelo YOLOv8 Aprimorado

Este capítulo aborda a análise dos desempenhos de treinamento de quatro modelos ablacionados: YOLOv8n, YOLOv8n + P2, YOLOv8n + C2f2, e o Modelo Proposto (YOLOv8n + P2 + C2f2). A análise é iniciada métricas de perda de treinamento e validação, revelando que o Modelo Proposto supera os demais ao demonstrar diminuições mais significativas nas perdas de treinamento e validação. Além disso, o modelo proposto exibe melhor estabilidade em relação às magnitudes de perda. A avaliação também inclui a comparação dos modelos em termos de precisão, revocação, mAP@0.5, mAP@0.5:0.95, número de parâmetros, operações de ponto flutuante por segundo (FLOPS), tempo de inferência e tempo total de processamento. Os resultados destacam que o Modelo Proposto apresenta o melhor equilíbrio entre os resultados de detecção e eficiência computacional, com menor número de parâmetros e FLOPS em comparação com os outros modelos avaliados. A análise detalhada dos resultados destaca tanto os pontos fortes quanto as áreas para melhoria do modelo proposto na detecção de pragas em campos de soja, ressaltando a necessidade de maiores investigações e otimização.

### 4.1 Dados de Treinamento do Modelo

Previamente à análise do Modelo propriamente dita, cabe examinar as perdas de treinamento e validação, na medida em que esses dados fornecem indícios sobre a estabilidade geral do processo de treinamento. Perdas mais altas podem indicar desafios na aprendizagem a partir dos dados ou instabilidade do modelo, enquanto perdas mais baixas sugerem uma aprendizagem eficaz e convergência do modelo.

Os dados de treinamento incluem valores para "train/box\_loss", "train/cls\_loss" e "train/df\_l\_loss", representando a evolução da perda na aprendizagem da caixa, da classe e do aprendizado profundo de recursos. Analisando esses valores, podemos entender como cada componente contribui para o processo geral de treinamento. Uma tendência decrescente nessas perdas indica que o modelo está efetivamente aprendendo e otimizando seus parâmetros ao longo do tempo.

Da mesma forma, os dados de validação apresentam valores para "val/box\_loss", "val/cls\_loss" e "val/df\_l\_loss", representando a evolução da perda durante a validação. Comparar as tendências de perda de treinamento e validação ajuda a avaliar o desempenho de generalização do modelo. Se a perda de validação permanecer próxima da perda de treinamento, isso sugere que o modelo não está sobreajustando aos dados de treinamento.

Os desempenhos do treinamento dos 4 modelos ablacionados (YOLOv8n; YOLOv8n + P2; YOLOv8n + C2f2 e o Modelo Proposto) se encontram nas Figuras 4.1 e 4.2 a seguir.

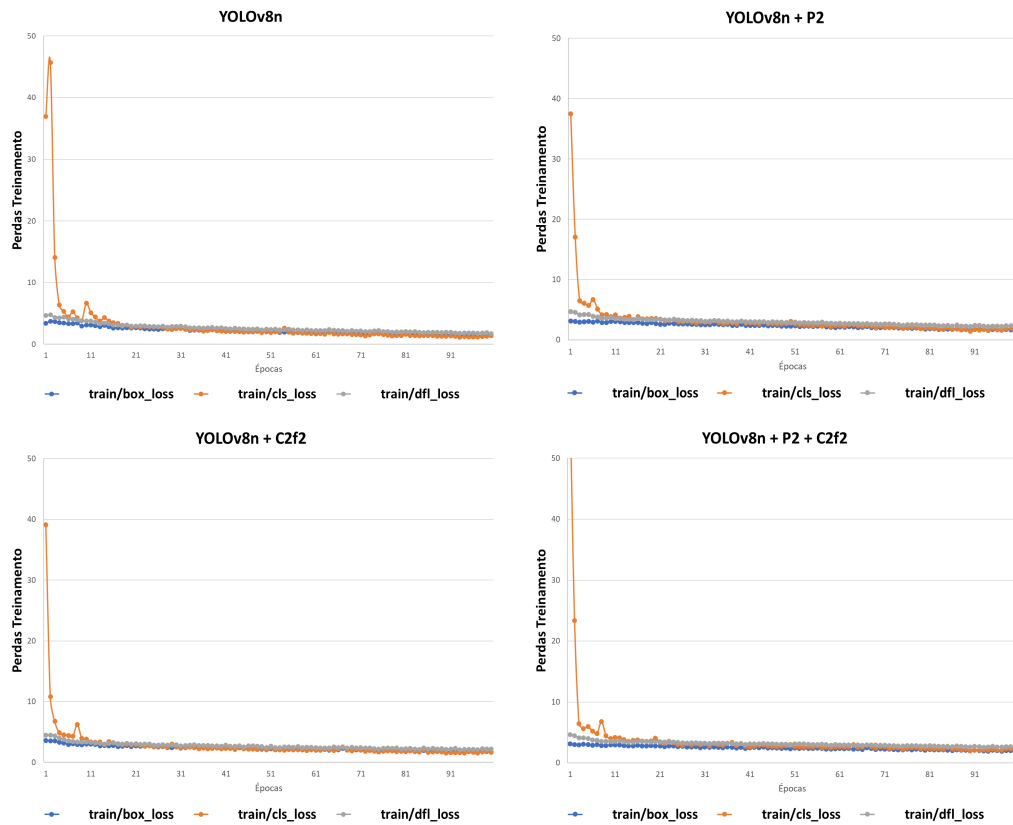


Figura 4.1 – Relatório de perdas (*losses*) de treinamento para cada modelo ablacionado.

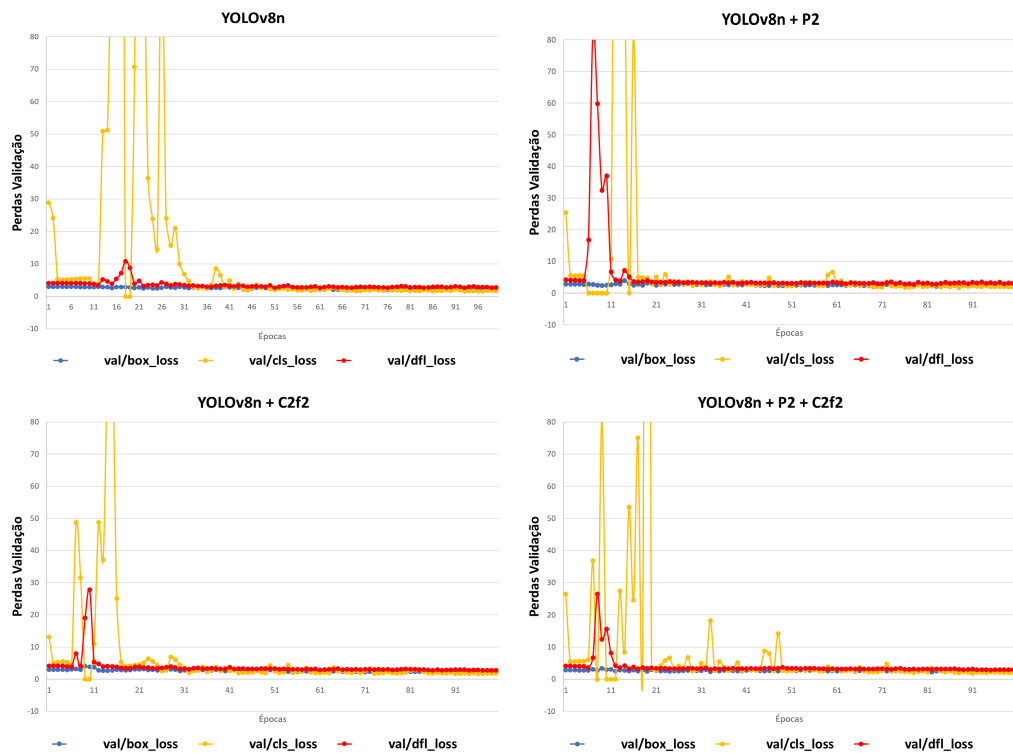


Figura 4.2 – Relatório de perdas (*losses*) de validação para cada modelo ablacionado.

O modelo proposto (YOLOv8n + C2f2 + P2) demonstra desempenho superior em comparação com YOLOv8n, YOLOv8n + C2f2 e YOLOv8n + P2, conforme indicado pela diminuição mais significativa nas perdas de treinamento e validação. Além disso, o modelo proposto apresenta melhor estabilidade em termos de magnitudes de perdas em comparação com os outros modelos.

## 4.2 Desempenho do Modelo

Nosso estudo comparou, conforme apresentado na Tabela 4.1, o desempenho de diferentes algoritmos de detecção de objetos para o conjunto de dados fornecido, conforme mostrado na Figura 3.1. Entre os algoritmos avaliados<sup>3</sup>, possuíam um tamanho uniforme de 640x640 pixels e contivessem uma quantidade substancial de dados anotados, atendendo ao limiar de 90% de anotação., estão YOLOv8n, YOLOv8n + C2f2, YOLOv8n + P2 e o modelo proposto (YOLOv8n + P2 + C2f2). Observamos, a seguir, que a adição tanto de P2 quanto de C2f2 ao YOLOv8n resultou no melhor desempenho geral entre os modelos YOLOv8.

Tabela 4.1 – Comparação do desempenho dos algoritmos em termos de precisão, revocação, mAP@0.5, mAP@0.5:0.95, número de parâmetros do modelo, operações de ponto flutuante, tempo de inferência e tempo total de processamento.

Modelo	Prec.	Revoc.	mAP0.5	mAP0.5:0.95	Par.(M)	FLOPS(G)	Inf.(ms)	Tempo Trein. (h)
YOLOv8n	84,4	55,5	61,5	34,4	3,01	8,1	0,4	0,33
YOLOv8n+C2f2	62,5	50,7	56,5	35	1,81	<b>5,1</b>	0,3	0,32
YOLOv8n+P2	77,1	59,2	65	<b>39,2</b>	2,92	12,2	0,7	0,33
<b>(Modelo Proposto)</b>	<b>78,3</b>	<b>68,4</b>	<b>71,1</b>	38,8	<b>1,69</b>	8,6	0,6	0,33

Os resultados do mAP0.5, mostrados Figura 4.3, apresentam uma visão abrangente do desempenho do modelo proposto em comparação com variantes alternativas da YOLOv8 na detecção do PMS em campos de soja. Ao longo das 100 épocas de treinamento, o modelo proposto demonstra consistentemente valores mais elevados de mAP0.5, indicando sua capacidade superior na detecção de insetos com um limiar de confiança de pelo menos 0,5. Este desempenho elevado sublinha a eficácia do modelo na captura de um amplo espectro de instâncias de PMSs, essencial para estratégias abrangentes de manejo de pragas na agricultura. No entanto, um exame matizado revela flutuações esporádicas e declínios intermitentes nos valores de mAP0.5 em épocas específicas, sugerindo desafios potenciais na manutenção de uma precisão de detecção consistente.

<sup>3</sup> Os dados em questão foram retirados dos extratos obtidos após os treinamentos no *Google Colaboratory* (GOOGLE, 2017) e, em seguida, transpostos para gráficos através do programa Microsoft Excel.



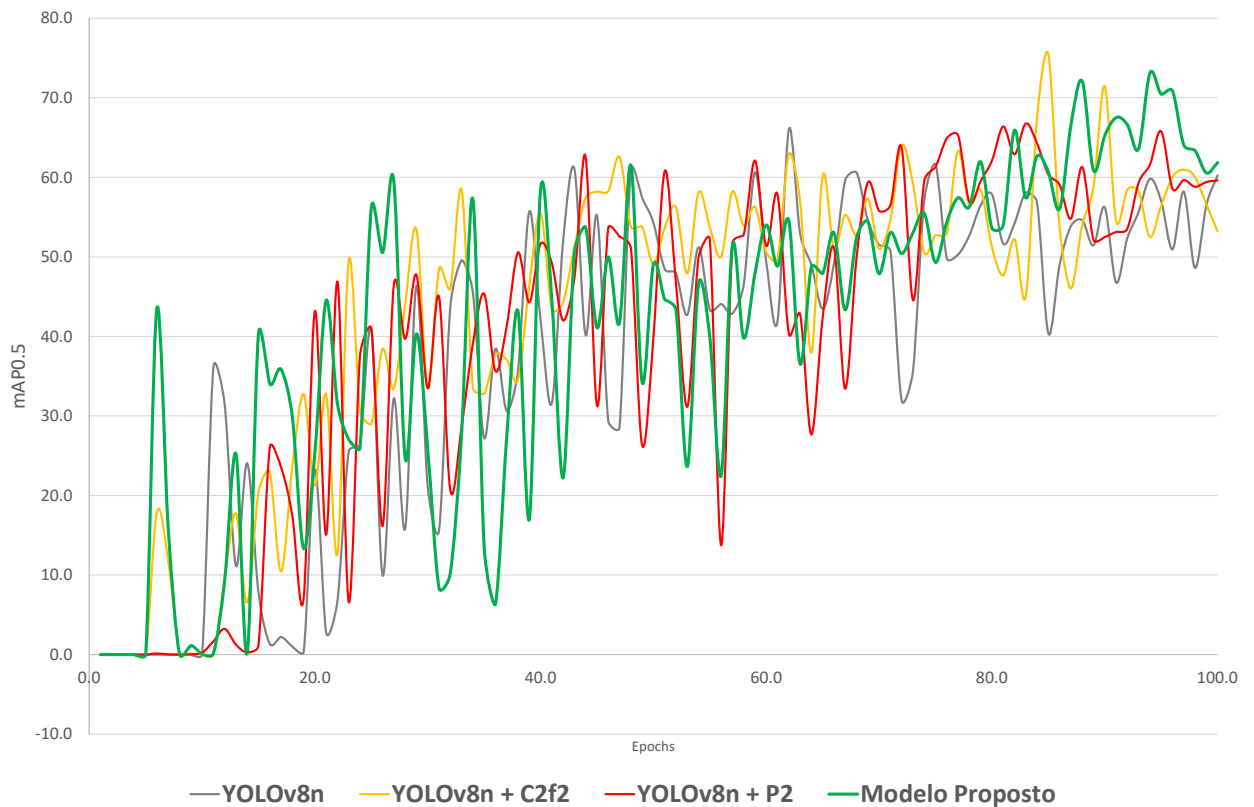


Figura 4.3 – Relatório de mAP0.5 para o experimento de ablação sobre YOLOv8.

Em contraste, os valores  $mAP_{0.5:0.95}$ , Figura 4.4, que avalia a precisão da detecção de PMSs em uma faixa de *thresholds* de confiança, de 0.5 a 0.95, indicam que o modelo proposto supera consistentemente as demais variantes ablacionadas da YOLOv8 (exceto a YOLOv8n + P2) em termos de valores de  $mAP_{0.5:0.95}$ , sugerindo que o algoritmo é capaz de detectar PMS com alta precisão em uma ampla gama de confianças. O resultado obtido aponta ainda para a robustez do algoritmo, suficiente para detectar PMS nas diferentes condições em que se obteve as imagens e o algoritmo é capaz de manter uma alta taxa de detecção mesmo em condições desafiadoras, como baixa luminosidade ou alta umidade. O resultado sugere também que o algoritmo minimiza a ocorrência de falsos positivos (identificação incorreta de objetos como PMS) e falsos negativos (falha na detecção de PMS presentes na imagem). No entanto, os resultados revelam novamente flutuações e declínios ocasionais nos valores de  $mAP_{0.5:0.95}$  durante épocas específicas, sugerindo desafios potenciais na manutenção de detecções consistentemente de alta confiança.

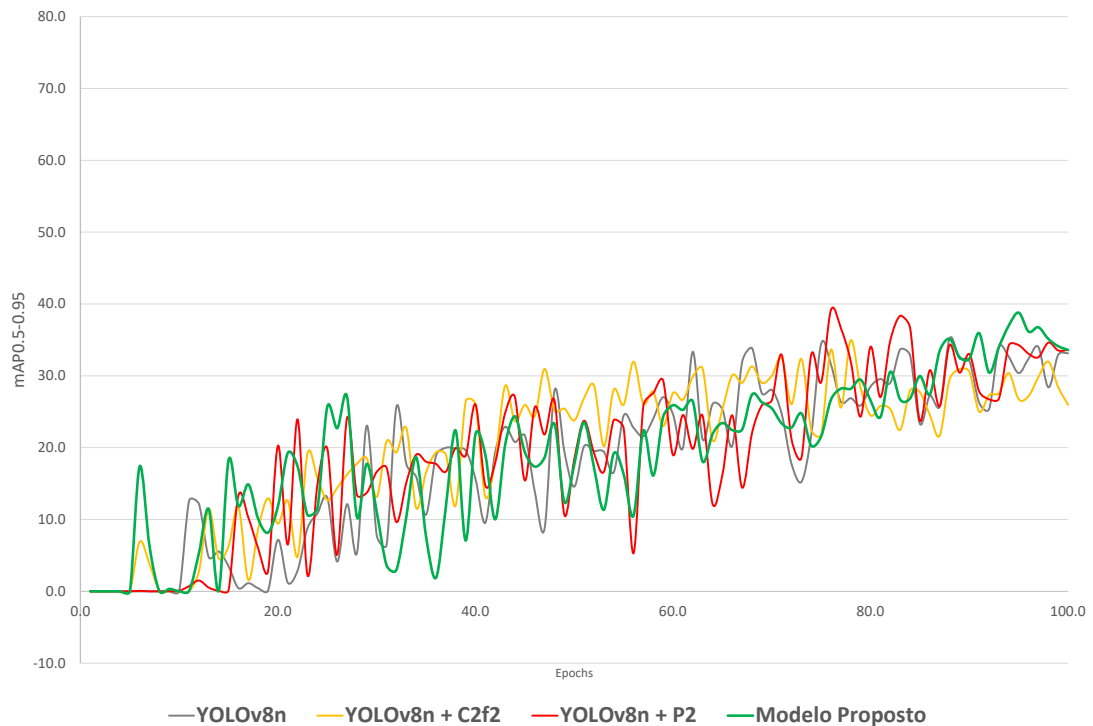


Figura 4.4 – Relatório de mAP0.5:0.95 para o experimento de ablação sobre YOLOv8.

Focando agora nos números obtidos para a precisão dos modelos ablacionados, Figura 4.5, os valores obtidos para cada modelo corroboram as tendências observadas nas análises de mAP0.5 e mAP0.5:0.95. YOLOv8n atinge a maior precisão de 84,4%, superando o modelo proposto (YOLOv8n + C2f2 + P2) e outras variantes. Porém, o modelo proposto demonstra valores de precisão competitivos, superando YOLOv8n + C2f2 e YOLOv8n + P2, destacando sua eficácia e mantendo-se competitivo, desde que comprovados os demais benefícios, como se verá a seguir.

Quanto aos resultados obtidos em termos de revocação, Figura 4.6, observa-se que o modelo proposto supera significativamente os outros modelos em termos de revocação. Isso significa que o modelo proposto tem uma capacidade muito maior de detectar corretamente os PMS na lavoura de soja, com uma taxa de sucesso de 68,4%. A revocação é especialmente importante em cenários agrícolas, onde a detecção precisa de pragas é crucial para o manejo eficaz de culturas. Um modelo com alta revocação como o modelo proposto significa que ele está identificando a maioria dos insetos PMS presentes nas imagens.

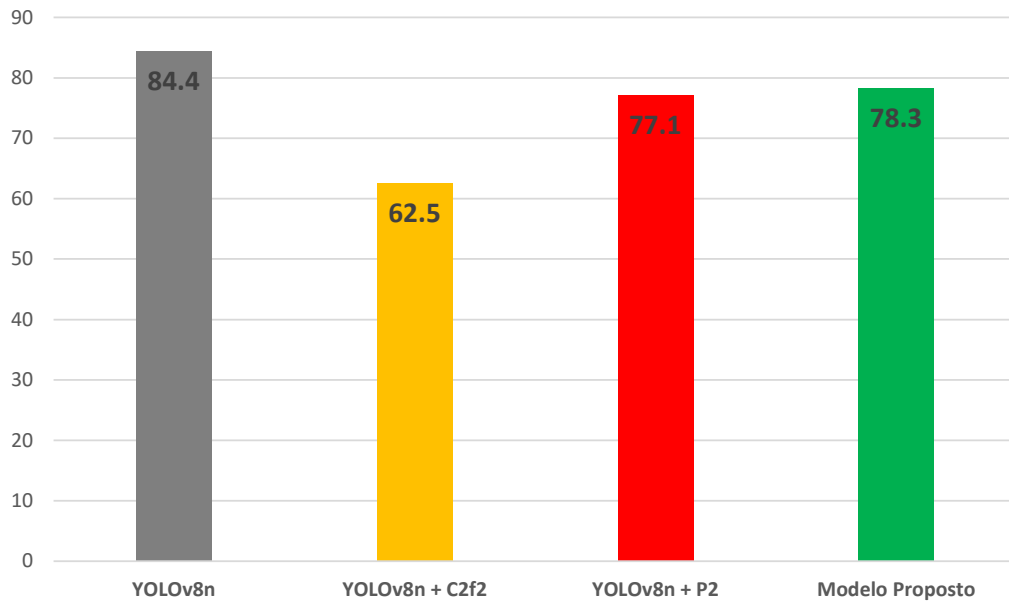


Figura 4.5 – Precisão (Prec.), para o experimento de ablação sobre YOLOv8.

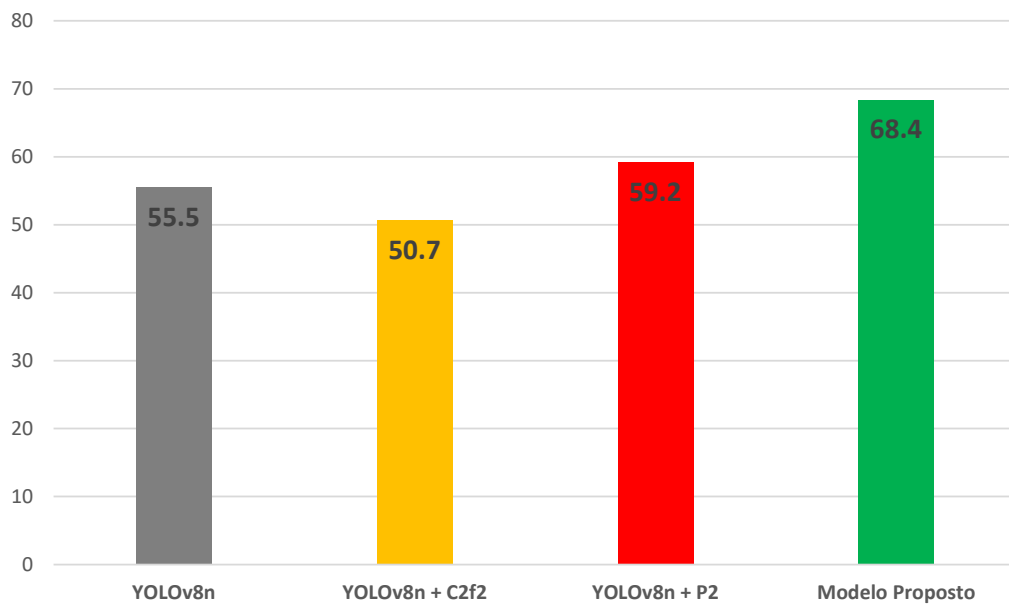


Figura 4.6 – Revocação (Revoc.), para o experimento de ablação sobre YOLOv8.

O número de parâmetros em um modelo, como vimos, é uma métrica crucial que reflete sua complexidade e capacidade de aprendizado a partir de dados. Modelos com contagens de parâmetros mais altas tendem a ser mais complexos e podem potencialmente capturar padrões intrincados nos dados. No entanto, também podem ser mais propensos ao *overfitting*, onde o modelo aprende ruído nos dados em vez de padrões verdadeiros. Por outro lado, modelos com contagens de parâmetros mais baixas são frequentemente mais simples e podem generalizar melhor para dados não vistos. Da mesma forma, a métrica FLOPS, que mede a complexidade computacional de um modelo durante a inferência, é crucial para avaliar sua eficiência. Valores de FLOPS mais baixos indicam tempos de inferência mais rápidos e custos computacionais reduzidos, tornando-os desejáveis para implantação em aplicações do mundo real onde os recursos computacionais são limitados. Em nosso estudo, avaliamos os quatro modelos ablacionados - YOLOv8n, YOLOv8n + C2f2, YOLOv8n + P2 e o modelo proposto (YOLOv8n + P2 + C2f2) - e os resultados foram plotados na Figura 4.7. Entre esses modelos, o modelo proposto se destaca por seu desempenho superior em termos tanto do número de parâmetros quanto de FLOPS.

O modelo proposto (YOLOv8n + P2 + C2f2) exibe o menor número de parâmetros, totalizando 1,69 milhão. Essa redução nos parâmetros sugere uma arquitetura mais simples em comparação com as outras configurações, indicando benefícios potenciais em termos de mitigação do *overfitting* e melhoria no quesito de generalização. Ao manter uma contagem de parâmetros enxuta, o modelo proposto demonstra uma utilização eficiente de recursos enquanto alcança desempenho competitivo em tarefas de detecção de PMSs. Além disso, o modelo proposto apresenta um valor moderado de FLOPS de 8,6, encontrando um equilíbrio entre complexidade computacional e eficácia na detecção de PMSs. Embora esse valor de FLOPS seja menor do que o de YOLOv8n + P2, é maior do que o de YOLOv8n + C2f2. Isso indica que o modelo proposto utiliza otimamente recursos computacionais, oferecendo um compromisso razoável entre desempenho do modelo e custo computacional.

O bloco C2f2, apesar de possuir menos filtros convolucionais, pode estar capturando características mais relevantes e discriminativas para a detecção de insetos PMS em comparação com o bloco proposto. Isso pode ser resultado de uma melhor adaptação do bloco C2f2 às características específicas do problema em questão, permitindo uma extração mais eficaz de informações relevantes das imagens das lavouras de soja. Além disso, é importante considerar que a redução no número de filtros convolucionais no bloco C2f2 pode levar a uma redução na dimensionalidade das representações intermediárias da rede neural, o que pode facilitar o treinamento e melhorar a generalização do modelo.

Parâmetros

FLOPS



Figura 4.7 – Resultados obtidos para número de parâmetros, em milhões, e para número de operações de ponto flutuante por segundo, em FLOPS(G) para o experimento de ablação sobre YOLOv8. A cor predominante representa o modelo ablacionado vencedor em cada medida.

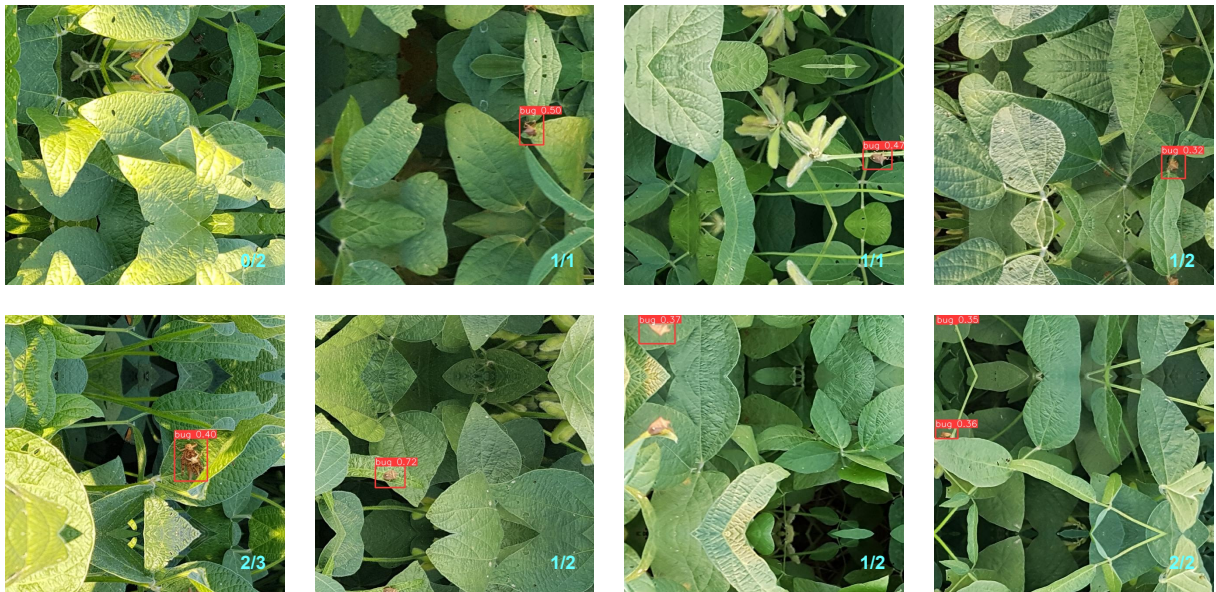


Figura 4.8 – Amostras de inferência (detecção de PMS com o modelo proposto) sobre o conjunto de dados INSECT10K7C640\_SAT (conjunto de teste). Os casos verdadeiros positivos são indicados quando o limiar de confiança é maior ou igual a 0.3. No canto inferior direito, em azul, apresentamos insetos detectados ou presentes na imagem em questão.

Por fim, parte dos resultados de inferência sobre o conjunto de testes do conjunto de dados INSECT10K7C640\_SAT é resumida na Figura 4.8. Analisando as amostras de inferência da esquerda para a direita e de cima para baixo, nós temos que primeiramente o modelo não conseguiu detectar os dois insetos PMS em posição lateral no canto da imagem.

Isso indica uma limitação na capacidade do modelo de detectar insetos quando estão em uma posição específica e em uma região específica da imagem. Uma possível razão para isso pode ser a falta de variação nos dados de treinamento para capturar esses casos específicos. Passando à próxima imagem, o modelo conseguiu detectar corretamente o inseto PMS, que estava exibindo bem o dorso e partes de sua coloração característica. Isso demonstra que o modelo é capaz de identificar corretamente os insetos quando apresentam características distintivas, o que é uma indicação positiva da eficácia do modelo. Quando analisamos a terceira imagem, novamente o modelo conseguiu detectar corretamente o inseto PMS em uma posição dorsal perfeita, exibindo todas as características distintivas. Isso sugere uma capacidade robusta do modelo em identificar insetos mesmo quando estão em posições específicas e exibindo características particulares. Quanto à quarta imagem, o modelo detectou corretamente o inseto original, mas falhou em detectar o inseto duplicado, que ficou parcialmente ocluído. Isso destaca uma limitação do modelo em lidar com casos de oclusão parcial, o que pode afetar sua capacidade de detectar todos os insetos presentes na imagem. Passando à próxima linha, temos que o modelo identificou incorretamente dois insetos com coloração parecida como PMS, sendo que apenas um deles era realmente um PMS. Além disso, um PMS em posição completamente lateral foi desprezado. Isso sugere que o modelo pode estar tendo dificuldades em discernir entre insetos PMS e outros insetos com características semelhantes, além de ter limitações na detecção de insetos em certas posições. Seguindo à próxima imagem, o modelo novamente falhou em detectar o inseto duplicado, que ficou parcialmente ocluído, enquanto detectava corretamente o inseto original. Isso reforça a observação anterior sobre a limitação do modelo em lidar com oclusões parciais. Por fim, na última imagem, o modelo conseguiu detectar corretamente os dois insetos PMS em posição lateral, incluindo o inseto duplicado. Isso sugere uma capacidade satisfatória do modelo em lidar com esses casos específicos de duplicação e oclusão parcial.

### 4.3 Conclusões do Capítulo

A análise abrangente do desempenho do modelo proposto na detecção de PMS em plantios de soja revela pontos fortes e áreas para melhoria. O modelo proposto apresenta o desempenho mais promissor em termos de  $mAP@0.5$ , indicando uma alta precisão na detecção de PMS em níveis moderados de confiança. Com uma pontuação elevada de  $mAP@0.5$ , este modelo pode ser confiável para identificar corretamente a presença de PMS na lavoura de soja em uma variedade de condições. Além disso, uma revocação significativa sugere que o modelo é capaz de recuperar a maioria dos insetos PMS presentes nas imagens. Embora o  $mAP@0.5-0.95$  seja ligeiramente menor do que o do modelo YOLOv8n+P2, a diferença não é significativa. Isso sugere que o modelo proposto mantém um desempenho competitivo mesmo em níveis mais altos de confiança (0.5-0.95), proporcionando uma detecção confiável de PMS em diferentes cenários. No entanto, um exame matizado dos valores de  $mAP@0.5$

---

indica flutuações esporádicas e declínios intermitentes em épocas específicas, tanto para mAP0.5 quanto para mAP0.5:0.95, sugerindo desafios potenciais na manutenção de uma precisão de detecção consistente. Estas flutuações podem surgir de vários fatores, incluindo problemas de convergência de modelos, disparidades na distribuição de dados ou complexidades de otimização. Enfrentar estes desafios é crucial para garantir a integridade e robustez do modelo proposto em cenários práticos de implantação.

Os resultados do número de parâmetros e FLOPS sugerem que adicionar P2 e C2f2 simultaneamente ao YOLOv8n pode efetivamente aprimorar o desempenho do modelo em tarefas de detecção e contagem de insetos, mantendo um equilíbrio entre complexidade computacional e eficácia na detecção. Ao incorporar ambos os componentes P2 e C2f2, o modelo proposto alcança uma redução nos parâmetros sem sacrificar a eficiência computacional, o que é crucial para muitas aplicações, como detecção de objetos em tempo real em vídeos, como se deseja a fim de obter um sistema de mapeamento a partir de vídeo. Importante também sublinhar que as modificações no modelo proposto resultaram em um impacto mínimo no tempo de inferência, corroborando a análise quanto à complexidade do modelo.

No geral, a discussão ressalta o potencial promissor do modelo proposto na detecção de PMSs em lavouras de soja. São necessários, porém, mais esforços de investigação e otimização para enfrentar os desafios identificados e maximizar a utilidade prática e o impacto do modelo em ambientes agrícolas.

## 5 Análise do *Framework* de Rastreamento e Contagem de Insetos

Os resultados do *framework* de rastreamento e contagem de PMSs em vídeos são resumidos na Figura 5.1 e apresentam diversos aspectos positivos que merecem menção.

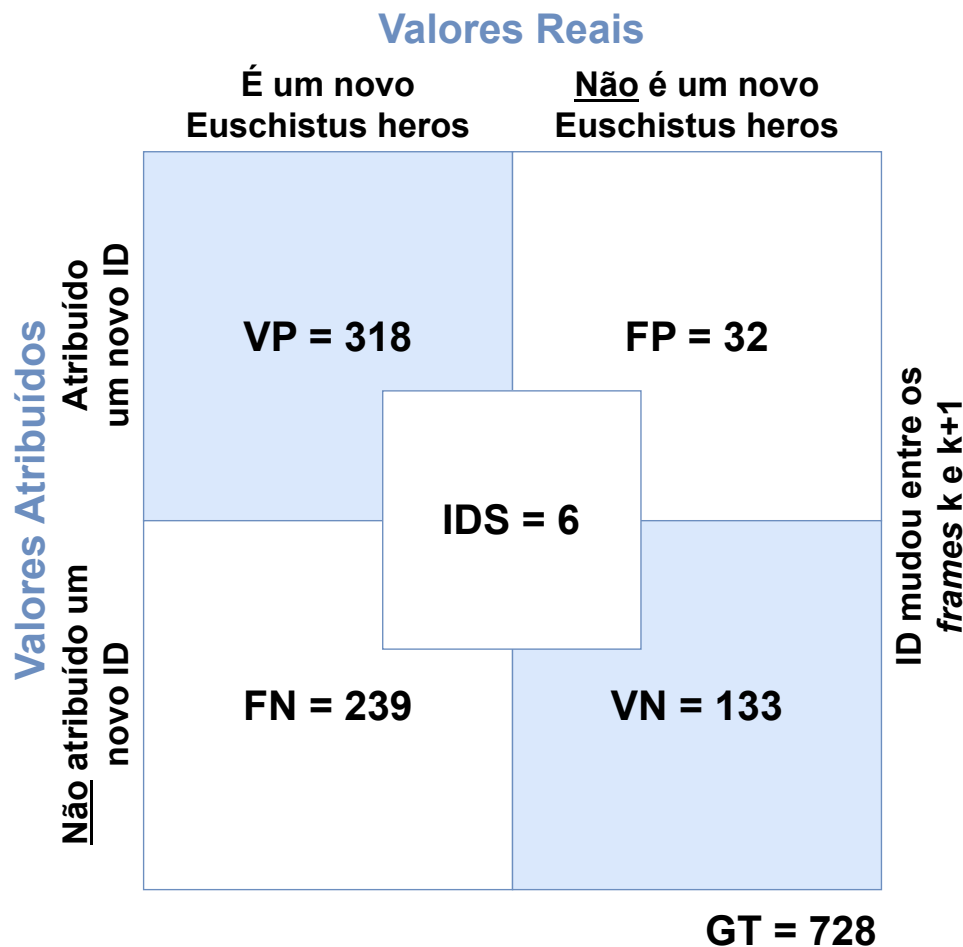


Figura 5.1 – Matriz de confusão expondo os resultados obtidos (VP, VN, FN, FP e IDS), valores absolutos, rastreando PMSs ao longo dos *frames* do *stream* de vídeo fornecido.

Em termos de falsos negativos (FN), o *framework* proposto exibiu um total combinado de FN de 239, indicando casos em que os modelos falharam em detectar PMSs que estavam presentes no *ground truth*, o que pode levar à subestimação das populações de PMSs. Embora essa métrica destaque áreas para melhoria potencial em sensibilidade, é importante observar que os modelos detectaram uma parte significativa dos indivíduos em *ground truth*, como evidenciado pelo total de verdadeiros positivos (VP) de 318.

Os números de falsos positivos (FP) foram mínimos, com uma contagem de 32, indicando casos em que o modelo identificou incorretamente objetos que não eram PMSs



como tais, ou duplicaram falsamente a contagem em um PMS. As trocas de identidade (IDS), representando casos em que os modelos trocaram incorretamente a identidade de um objeto entre quadros, foram limitadas a 6 instâncias.

O escore de precisão de rastreamento de múltiplos objetos (MOTA), métrica que considera FN, FP e IDS, foi calculado em cerca de 62%. Esse valor razoável de MOTA reflete a robustez do modelo em manter trajetórias de objetos consistentes, apesar de desafios como oclusões e mudanças nas condições de iluminação. Um valor de MOTA alto destaca a confiabilidade e eficácia do modelo em captar e rastrear a população de insetos em *stream* de vídeo, posicionando o modelo como uma ferramenta promissora para aplicações de monitoramento de insetos em tempo real.

Também consideramos uma divisão dos *frames* em lotes, conforme os dados quantitativos da Tabela 5.1 e qualitativos da Tabela 5.2. Os *frames* de cada lote encontram-se no [Repositório](#) do trabalho. Com esses dados conseguimos analisar mais profundamente os resultados obtidos e cada caso de falha para melhorias posteriores.

Tabela 5.1 – Resultados quantitativos para cada lote do extrato de *frames* obtido.

Lote	Nº Frames	VP	VN	FP	FN	IDS	GT	MOTA(%)	PMSs presentes	PMSs detectados
Lote 1	150	113	21	6	28	1	169	79,29	10	13
Lote 2	150	40	49	8	64	3	164	54,27	7	6
Lote 3	150	102	21	18	43	1	185	66,49	10	12
Lote 4	150	63	35	0	91	1	190	51,58	10	9
Lote 5	20	0	7	0	13	0	20	35,00	1	0

Nos lotes onde as características dos PMSs são bem visíveis e consistentes ao longo do vídeo (como espinhos no pronoto e mancha branca no dorso), Figura 5.2, o *framework* tende a ter um bom desempenho, identificando-os com precisão na maior parte dos frames. Isso é evidenciado em casos como o Lote 1 e parte do Lote 3, onde o modelo obteve expressivamente mais VPs que FNs.

Tabela 5.2 – Resultados qualitativos para cada lote do extrato de *frames* obtido.

Lote	Observações relevantes
Lote 1	<p>1 PMS em vista semi dorsal (ligeiramente inclinada lateralmente), características bem visíveis (espinhos no pronoto e mancha branca no dorso), identificado com VP em todos os frames (15); 1 PMS em vista traseira (descendo a folha), identificado com VP em 13 frames, 4 FP (duplicatas); 1 PMS em vista completamente dorsal, identificado com VP em 14 de 15 dos frames, com 1 IDS; 1 PMS em vista semi dorsal (ligeiramente inclinada lateralmente), características bem visíveis (espinhos no pronoto e mancha branca no dorso), mais próximo à câmera (zoom), identificado com VP em 11 frames e não identificado (FN) em 3 frames no início da aparição. 1 PMS em vista dorsal, levemente frontal (subindo a folha), características bem visíveis (espinhos no pronoto e mancha branca no dorso), identificado com VP em todos os frames (15); 1 PMS com vista dorsal, levemente lateral e traseiro, características bem visíveis (espinhos no pronoto e mancha branca no dorso), identificado com VP em todos os frames, mesmo na presença de outros PMSs iguais em alguns frames. 1 dupla de PMSs em acasalamento, vista dorsal de ambos, características bem visíveis (espinhos no pronoto e mancha branca no dorso), somente um dos indivíduos identificado com VP em todos os frames, o outro indivíduo não identificado em todos os frames (FN); 1 PMS em vista frontal, mancha branca parcialmente visível, identificado com VP em todos os frames.</p>
Lote 2	<p>1 PMS em vista frontal (subindo a folha), mancha branca parcialmente visível, não identificado (FN) em todos os frames (14); 1 PMS em vista dorsal, levemente frontal (subindo a folha), características bem visíveis (espinhos no pronoto e mancha branca no dorso) identificado com VP em todos os frames (15); 1 PMS em vista predominantemente lateral, visível parte do dorso, espinhos no pronoto e mancha branca levemente visíveis, identificado com VP em todas os frames; 1 PMS em instar de ninfa, coloração ofuscada pelo sol, corretamente não identificado como VP em nenhum frame; 1 PMS em vista completamente frontal, espinhos no pronoto levemente visíveis, não identificado (FN) em todos os frames (14); 1 PMS em vista dorsal, características bem visíveis (espinhos no pronoto e mancha branca no dorso), identificado como VP em 14 frames, porém com a presença de duplicatas (FP) em 8 frames e 3 IDS; 1 PMS em vista frontal, parcialmente dorsal, características bem visíveis (espinhos no pronoto e mancha branca no dorso), não identificado (FN) em todos os frames (14); 1 PMS em vista dorsal, levemente lateral, características bem visíveis (espinhos no pronoto e mancha branca no dorso), identificado como VP em apenas 1 de 11 dos frames, no restante FN.</p>
Lote 3	<p>1 PMS em vista lateral, espinhos no pronoto e mancha branca levemente visíveis, identificado como VP em todos os frames (15); 1 PMS em vista dorsal, características bem visíveis (espinhos no pronoto e mancha branca no dorso), não identificado (FN) em todos os frames (13); 1 PMS em vista dorsal, parcialmente frontal, identificado (VP) em todos os frames (15), mesmo após o surgimento de um segundo PMS em cena; 1 PMS em vista completamente dorsal, características bem visíveis (espinhos no pronoto e mancha branca no dorso), identificado (VP) em 13 de 14 frames, 1 IDS; 1 PMS em vista dorsal, levemente lateral, características bem visíveis (espinhos no pronoto e mancha branca no dorso), identificado (VP) em 11/15 dos frames, duplicata (FP) em 1 frame, FN em 4 de 15 frames; 1 PMS em vista frontal, parcialmente dorsal (subindo a folha), mancha branca visível, identificado em todos os frames, porém com duplicatas (FP) em 3 de 16 deles; 1 PMS em vista dorsal, características bem visíveis (espinhos no pronoto e mancha branca no dorso), não identificado (FN) em todos os frames (13); 1 PMS em vista totalmente dorsal, mancha branca visível, identificado (VP) em todos os frames, porém com a presença de duplicatas em 14/16 deles; 1 PMS em vista completamente lateral, com mancha branca parcialmente visível, identificado (VP) em todos os frames (15)</p>
Lote 4	<p>1 PMS em vista frontal, parcialmente dorsal, com mancha branca visível, identificado (VP) em 14 de 15 frames, com 1 IDS; 2 PMSs em acasalamento em vista completamente dorsal, características bem visíveis (espinhos no pronoto e mancha branca no dorso), apenas um indivíduo identificado em 1/12 dos frames; 1 PMS em vista dorsal, parcialmente lateralizado, no canto da imagem, mancha branca visível, não identificado (FN) em todos os frames (14); 1 PMS em vista totalmente lateral, mancha branca levemente visível, identificado (VP) em todos os frames (15); 1 PMS em vista dorsal, levemente lateral traseira, características bem visíveis (espinhos no pronoto e mancha branca no dorso), identificado (VP) em todos os frames (15); 1 PMS em vista lateral, parcialmente dorsal e frontal (subindo a folha), no canto da imagem, mancha branca e espinhos levemente visíveis, não identificado (FN) em 13 de 14 frames, VP em 1 frame; 2 PMSs em acasalamento, vista completamente lateral de ambos, sem características distintivas visíveis (mancha e espinhos), não identificados (FN) em todos os frames (15); 1 PMS em vista frontal, com dorso e mancha branca visíveis, em zoom out, identificado (VP) em 13 de 14 frames.</p>
Lote 5	<p>1 PMS em vista frontal, parte do dorso e mancha branca visíveis, subindo a folha, não identificado (FN) em 13 frames.</p>

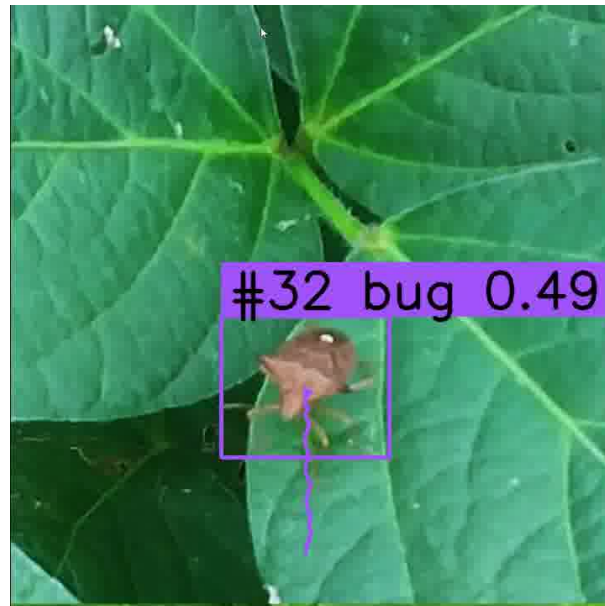


Figura 5.2 – PMS com espinhos no pronoto e mancha branca no dorso bem visível.

O *framework* também se sai bem na identificação de insetos em vistas frontais e dorsais claras, como na Figura 5.3, especialmente quando as características dos insetos são bem visíveis. Isso é observado em alguns casos, como a maioria dos insetos identificados no Lote 4.

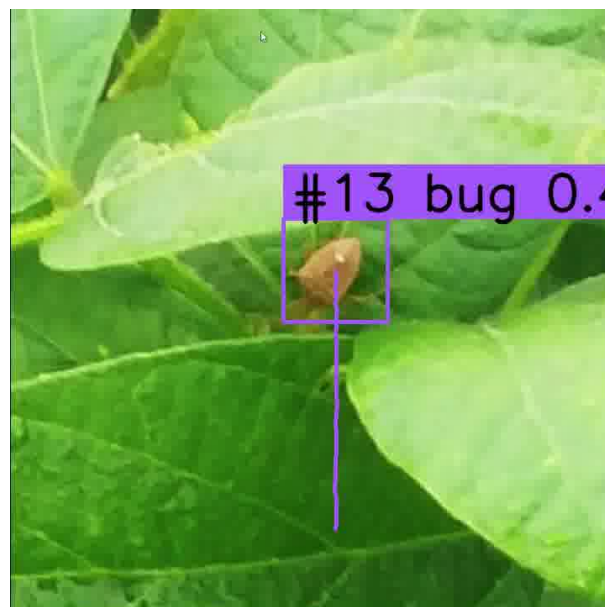


Figura 5.3 – PMS em vista frontal clara.

O *framework* se mostrou eficiente também em identificar (marcar VPs) quando há insetos distintos no mesmo *frame*, como na Figura 5.4. Isso ficou evidente especialmente nos lotes 3 e 4.

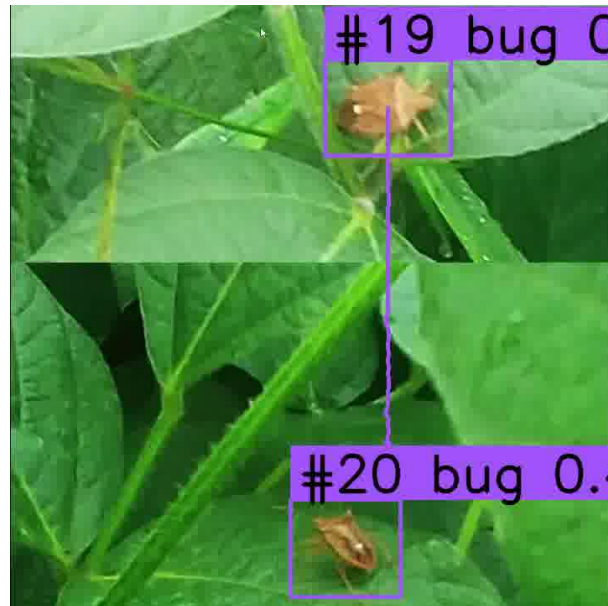


Figura 5.4 – Dois PMSs visíveis na mesma cena.

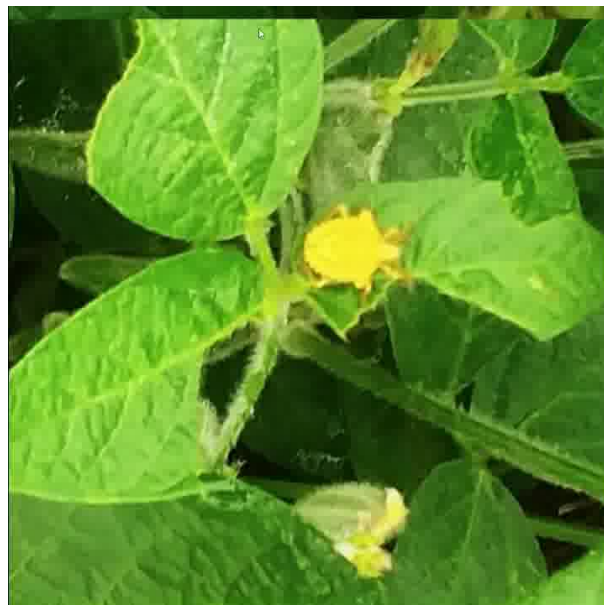


Figura 5.5 – Ninf de PMS, ignorada pelo *framework*.

Ainda, com relação ao lote 2, verificamos que o modelo, que foi projetado para identificar e contar indivíduos adultos, foi capaz de ignorar em sua contagem um PMS no instar de ninf, Figura 5.5, ressaltando a sua capacidade de ser específico.

Por outro lado, o *framework* mostra inconsistência na identificação dos insetos em condições variadas, como diferentes ângulos de visão, como na Figura 5.6, iluminação e distância da câmera. Isso é observado em vários lotes, onde alguns insetos não são identificados (falsos negativos). O modelo enfrenta dificuldades na identificação de insetos menos visíveis, como aqueles com características menos distintas ou que estão parcialmente obscurecidos.

Isso é evidente nos lotes 2, 4 e 5, onde os insetos não foram identificados corretamente em muitos *frames*.

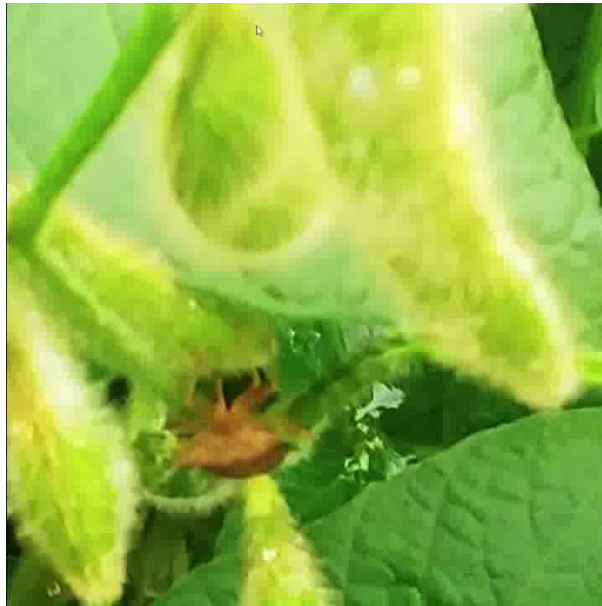


Figura 5.6 – PMS em posição frontal, dificultando sua identificação.

Vale mencionar ainda que o modelo enfrenta dificuldades notáveis na identificação correta de insetos durante o acasalamento, Figura 5.7, visto nos lotes 1 e 4. Nesse tipo de evento o modelo considerou apenas um dos indivíduos e deixou de marcar o outro, resultando em sucessivos falsos negativos.

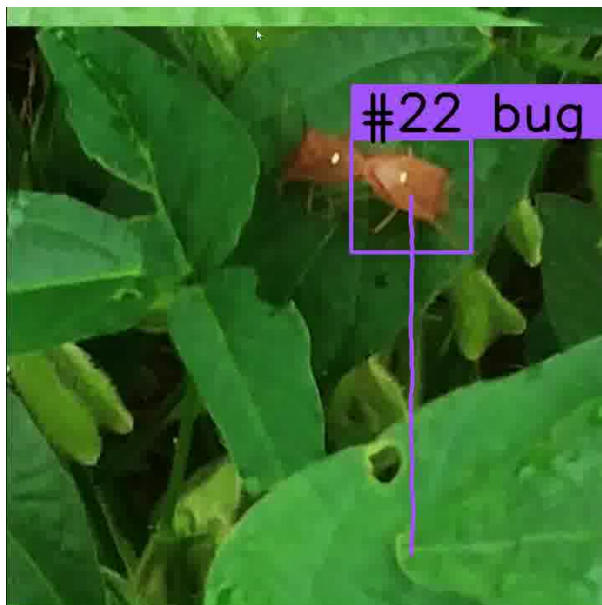


Figura 5.7 – PMSs em acasalamento.

Por fim, destacamos que no *stream* de vídeo havia 38 PMSs únicos presentes e que

o modelo proposto (YOLOv8 aprimorado + ByteTrack) foi capaz de rastrear (atribuir IDs únicos) em 40 ocasiões, demonstrando no caso em questão um certo equilíbrio entre efeitos somativos e subtrativos, alcançando um valor ligeiramente próximo ao real (aumentado em 5,3% em relação ao valor real).

## 5.1 Conclusões do Capítulo

No contexto da detecção e contagem de insetos, ter um *framework* baseado em alto número de Verdadeiros Positivos (VPs) é vantajoso por várias razões, apesar da presença de alto número de Falsos Negativos (FNs). VPs elevados indicam que o modelo está capturando efetivamente uma parte significativa da população de insetos, fornecendo dados valiosos para o monitoramento agrícola.

Em primeiro lugar, um alto número de VPs garante que o modelo represente com precisão a verdadeira distribuição e densidade de insetos no ambiente. Essas informações são cruciais para avaliar a dinâmica populacional, identificar áreas de alta atividade de insetos e implementar intervenções direcionadas na agricultura para mitigar danos causados por pragas. Em segundo lugar, um sistema com alto número de VPs inspira confiança na confiabilidade da saída do modelo, aprimorando sua utilidade nos processos de tomada de decisão, na medida em que altos VPs contribuem para a confiabilidade geral e credibilidade dos dados coletados.

A baixa taxa de FPs obtida ressalta a especificidade dos modelos em distinguir insetos de outros elementos do ambiente e em compreender um indivíduo em uma trajetória, sem apontar que se trata de um novo, contribuindo para a precisão geral da contagem de insetos. Também, o reduzido valor de IDs demonstra a capacidade dos modelos de manter a consistência no rastreamento de insetos individuais ao longo do tempo, crucial para avaliar com precisão a contagem populacional.

Em resumo, os resultados da detecção de insetos em vídeo com o modelo YOLOv8 aprimorado e contagem com ByteTrack demonstram a eficácia do *framework* em identificar e contar insetos com precisão em *stream* de vídeo. Embora haja áreas para melhoria, como a redução de falsos negativos e o aumento da sensibilidade, podemos verificar que o resultado de falsos negativos ao longo dos *frames* não é tão grave para missões de contagem, tendo em vista que aqui o principal objetivo é contar os PMSs e não necessariamente rastreá-los *frame a frame*, de modo que um inseto perdido em um frame pode ser "recuperado" nos frames seguintes e devidamente contado.

Por fim, o *framework* apresenta ainda um desempenho promissor quanto ao resultado final de MOTA obtido e o cômputo de insetos (objetivo principal do *framework*) com erro reduzido. Esses resultados demonstram seu potencial para aplicações na agricultura de precisão, especificamente no manejo ecológico de insetos.

## 6 Conclusão

Esta dissertação forneceu contribuições para obter um modelo YOLOv8 aprimorado, adaptado para detecção de insetos em imagens de campo, particularmente o Percevejo-marrom-da-soja (PMS) *Euschistus heros* (Hemiptera: Pentatomidae) através de experimentos de ablação com novos módulos mostrando a eficácia de sua inclusão. Testes com um conjunto de dados público e com novas imagens coletadas em campo de campos de soja, transpostos em forma de vídeo, foram realizados de modo a testar a capacidade de generalização do modelo, empregando-o na contagem de PMSs nesse novo conjunto de dados.

### 6.1 Conjunto de Dados Proposto

Este trabalho empregou para o treinamento de seu modelo de detecção uma base de dados publicada anteriormente, o INSECT10K7C (Tetila, 2019), composta por 1.000 imagens de plantações de soja em Dourados-MS, Brasil, capturadas durante diferentes fases fenológicas. Após um processo de pré-processamento e aumento de dados, o conjunto foi expandido para 4.000 imagens, com técnicas como auto-orientação, redimensionamento e aumento de saturação, visando diversificar e enriquecer o conjunto para um treinamento mais eficaz do modelo.

Além disso, um conjunto de imagens inéditas foi proposto para testar a capacidade de generalização do modelo em diferentes condições, a fim de demonstrar sua eficácia na detecção de pragas em plantações de soja. A abordagem do estudo incluiu a compilação das imagens em um vídeo utilizando o PowerPoint para testar a capacidade de contagem do método proposto. Este vídeo, com duração de 2 minutos e 7 segundos, foi gravado e dividido em frames a uma taxa de 5 FPS para análise. Embora essa abordagem não substitua completamente o uso de drones, ela ofereceu uma alternativa criativa e econômica para a análise em questão.

### 6.2 Modelo de Detecção Proposto

O modelo proposto para detecção de PMS em plantios de soja, que combina YOLOv8n com uma nova camada C2f2 e um nível de profundidade P2, demonstrou competitividade em relação às outras variantes ablaçadas sobre a YOLOv8, especialmente em termos de mAP0.5 e mAP0.5-0.95. No entanto, uma análise mais detalhada revelou flutuações ocasionais e declínios intermitentes nessas métricas, sugerindo desafios na manutenção de uma precisão consistente de detecção, possivelmente devido a problemas de convergência de modelos ou disparidades na distribuição de dados.

Adicionalmente, a inclusão simultânea de P2 e C2f2 à YOLOv8n mostrou-se eficaz para aprimorar o desempenho do modelo, obtendo, dentre os modelos ablacionados, o maior mAP0.5, o segundo maior mAP0.5-0.95 e um grande salto em revocação, na missão de detecção de PMSs, mantendo um equilíbrio entre complexidade computacional e eficácia na detecção dos PMSs. Isso resultou em uma redução nos parâmetros do modelo da ordem de 56%, quando comparado como modelo original YOLOv8n, sem sacrificar a eficiência computacional, essencial para aplicações que demandam detecção de objetos em tempo real, como sistemas de mapeamento a partir de vídeo. Embora sejam necessários mais esforços de pesquisa e otimização para enfrentar os desafios identificados e maximizar a utilidade prática do modelo em ambientes agrícolas, os resultados globais destacam seu potencial promissor.

### 6.3 Processo de Rastreamento e Contagem Proposto

O uso de um *framework*, como o proposto, capaz de obter alto número de Verdadeiros Positivos (VPs) no rastreamento e contagem de insetos mostrou-se vantajoso, apesar da presença de Falsos Negativos (FNs) significativos. Esse cenário garantiu uma representação precisa da distribuição e densidade de PMSs em cena, essencial para avaliar a dinâmica populacional e implementar medidas de controle em áreas de alta infestação. Além disso, a confiabilidade dos dados coletados é reforçada, pois altos VPs inspiram confiança na saída do modelo, aumentando sua utilidade nos processos de tomada de decisão.

A especificidade do modelo em distinguir PMSs de outros elementos do ambiente (ou mesmo de PMSs em outras fases) e sua consistência no rastreamento ao longo do tempo contribuíram para a precisão geral da contagem de insetos. Embora existam áreas para melhorias, como a redução de Falsos Negativos e o aumento da sensibilidade, a análise revela que, para missões de contagem, os FNs não comprometeram gravemente o resultado, uma vez que o foco principal é contar os insetos, não necessariamente rastreá-los em todos os frames. Além disso, o *framework* apresenta um desempenho promissor no resultado final de MOTA e no cômputo de insetos, demonstrando seu potencial para aplicações na agricultura e no manejo ecológico de pragas.

### 6.4 Sugestão de Trabalhos Futuros

A solução proposta demonstrou resultados promissores sob as condições testadas, mas necessita de testes adicionais em condições climáticas, luminosidade e tipos de cultivo mais variados e com a presença de outras variedades de insetos para ser plenamente ajustada antes de ser integrada em uma plataforma operacional. Como trabalhos futuros, é interessante também considerar a adoção de modelos treinados em conjuntos de imagens maiores e também obtidos a partir de drones de modo a testar as limitações e dificuldades do emprego



dessa plataforma no registro dos PMS, principalmente quanto à resolução obtida com as câmeras disponíveis no mercado. Realizar inferência com câmeras de vídeo embarcadas em drones permitiria obter, além da detecção e rastreamento, os metadados de geolocalização de cada pixel obtido (GeoTIFF), de modo que o framework de detecção e contagem dos insetos possa ser ampliado para também mapeá-los a partir de sua posição. É importante também responder às seguintes dúvidas quanto ao emprego de drones para mapeamento de insetos: qual a distância necessária, bem como a resolução do equipamento empregado de modo a obter qualidade de imagem e evitar interferência na cena registrada? Qual o consumo energético com um sistema embarcado com o modelo em questão durante uma operação de inferência? E finalmente, é importante passar o método final pelo crivo de especialistas na área a fim de validar se o mesmo é aceitável (e em que medida) como substituto ao método do pano-de-batida.

## 6.5 Artigo Publicado

Como parte das contribuições resultantes desta pesquisa, um trabalho foi publicado na revista *Ecological Informatics*, Fator de Impacto de 5,1 (Clarivate Analytics, 2023) e avaliação Qualis A1, Fator de Impacto 5.1, conforme citação abaixo:

LIMA, Bruno P.M.; BORGES, Lurdineide A.B.; HIROSE, Edson; BORGES, Díbio L. A lightweight and enhanced model for detecting the Neotropical brown stink bug, *Euschistus heros* (Hemiptera: Pentatomidae) based on YOLOv8 for soybean fields. **Ecological Informatics**, p. 102543, 2024. <https://doi.org/10.1016/j.ecoinf.2024.102543>

## 6.6 Repositório

Encorajando pesquisas futuras, compartilhamos prontamente nossos conjuntos de dados e códigos no [Repositório](https://github.com/pinheimelobruno/Euschistus) do trabalho <https://github.com/pinheimelobruno/Euschistus>

## Referências

- AADI, F. Z. A. H.; SADIQ, A.; LABD, Z. Comparing object tracking algorithms for real-time applications: Performance analysis and implementation study. *In: IEEE. 2023 10th International Conference on Wireless Networks and Mobile Communications (WINCOM)*. [S.l.], 2023. p. 1–5. Citado nas pp. 10 e 39.
- AHMAD, I. *et al.* Deep learning based detector yolov5 for identifying insect pests. **Applied Sciences**, MDPI, v. 12, n. 19, p. 10167, 2022. Citado nas pp. 32 e 33.
- ASPOSE. **Convert video file to image online app - Free Online Video file to Image Converter**. 2024. Disponível em: <https://products.aspose.app/video/video-to-image>. Citado na p. 51.
- BENJUMEA, A. *et al.* Yolo-z: Improving small object detection in yolov5 for autonomous vehicles. **arXiv preprint arXiv:2112.11798**, 2021. Citado nas pp. 34, 35 e 42.
- BERNARDIN, K.; ELBS, A.; STIEFELHAGEN, R. Multiple object tracking performance metrics and evaluation in a smart room environment. *In: CITESEER. Sixth IEEE International Workshop on Visual Surveillance, in conjunction with ECCV*. [S.l.], 2006. v. 90, n. 91. Citado na p. 51.
- BORTOLOTTO, O. C. *et al.* The use of soybean integrated pest management in brazil: a review. **Agronomy Science and Biotechnology**, v. 1, n. 1, p. 25–25, 2015. Citado na p. 16.
- BUENO, A.; BATISTELA, M. J.; BUENO, R. C. O.; FRANÇA-NETO, J. de B.; NISHIKAWA, M. A. N.; FILHO, A. L. Effects of integrated pest management, biological control and prophylactic use of insecticides on the management and sustainability of soybean. **Crop Protection**, Elsevier, v. 30, n. 7, p. 937–945, 2011. Citado na p. 14.
- BUENO, A. d. F.; SOSA-GÓMEZ, D. R.; SILVA, D. M. da. Medidas de controle eficazes contra percevejos em soja. **Medidas de controle eficazes contra percevejos em soja - Revista Cultivar**, Grupo Cultivar, Jun 2020. Disponível em: <https://revistacultivar.com.br/artigos/medidas-de-controle-eficazes-contrapercevejos-em-soja>. Citado na p. 22.
- BUTERA, L. *et al.* Precise agriculture: Effective deep learning strategies to detect pest insects. **IEEE/CAA Journal of Automatica Sinica**, IEEE, v. 9, n. 2, p. 246–258, 2021. Citado na p. 15.
- CAO, X. *et al.* Online tracking of ants based on deep association metrics: method, dataset and evaluation. **Pattern Recognition**, Elsevier, v. 103, p. 107233, 2020. Citado na p. 38.

- CHENG, Z. *et al.* A lightweight crop pest detection method based on convolutional neural networks. **Applied Sciences**, MDPI, v. 12, n. 15, p. 7378, 2022. Citado nas pp. 35 e 36.
- CORRÊA-FERREIRA, B. S.; KRZYZANOWSKI, F. C.; MINAMI, C. A. Percevejos e a qualidade da semente de soja-série sementes. Londrina: Embrapa Soja, 2009., 2009. Citado nas pp. 8, 17 e 22.
- CORRÊA-FERREIRA, B. S.; SOSA-GÓMEZ, D. R. Percevejos e o sistema de produção soja-milho. Londrina: Embrapa Soja, 2017., 2017. Citado na p. 21.
- DIWAN, T.; ANIRUDH, G.; TEMBHURNE, J. V. Object detection using yolo: Challenges, architectural successors, datasets and applications. **multimedia Tools and Applications**, Springer, v. 82, n. 6, p. 9243–9275, 2023. Citado na p. 38.
- FENG, C. *et al.* Tood: Task-aligned one-stage object detection. *In: IEEE COMPUTER SOCIETY. 2021 IEEE/CVF International Conference on Computer Vision (ICCV)*. [S.l.], 2021. p. 3490–3499. Citado na p. 32.
- GOOGLE. **Google Colaboratory**. 2017. Disponível em: <https://colab.google/>. Citado nas pp. 51 e 55.
- HE, K. *et al.* Spatial pyramid pooling in deep convolutional networks for visual recognition. **IEEE transactions on pattern analysis and machine intelligence**, IEEE, v. 37, n. 9, p. 1904–1916, 2015. Citado na p. 29.
- HOFFMANN-CAMPO, C. B. *et al.* **Pragas da soja no Brasil e seu manejo integrado**. [S.l.]: Embrapa soja Londrina, 2000. v. 30. Citado nas pp. 8, 10, 16, 21 e 22.
- HØYE, T. T. *et al.* Deep learning and computer vision will transform entomology. **Proceedings of the National Academy of Sciences**, National Acad Sciences, v. 118, n. 2, p. e2002545117, 2021. Citado na p. 15.
- JOCHER, G. **New - yolov8 multi-object tracking · issue #1429 · ultralytics/ultralytics**. 2023. Disponível em: <https://github.com/ultralytics/ultralytics/issues/1429>. Citado na p. 27.
- KANG, J. *et al.* Research on an improved yolov8 image segmentation model for crop pests. **Advances in Computer, Signals and Systems**, Clausius Scientific Press, v. 7, n. 3, p. 1–8, 2023. Citado nas pp. 35 e 36.
- KASINATHAN, T.; SINGARAJU, D.; UYYALA, S. R. Insect classification and detection in field crops using modern machine learning techniques. **Information Processing in Agriculture**, Elsevier, v. 8, n. 3, p. 446–457, 2021. Citado na p. 15.
- KERN, A. *et al.* Openrealm: Real-time mapping for unmanned aerial vehicles. *In: IEEE. 2020 International Conference on Unmanned Aircraft Systems (ICUAS)*. [S.l.], 2020. p. 902–911. Citado nas pp. 8, 17 e 18.

- KHALID, S. *et al.* Small pests detection in field crops using deep learning object detection. **Sustainability**, MDPI, v. 15, n. 8, p. 6815, 2023. Citado nas pp. 32 e 33.
- KING, R. **Brief summary of Yolov8 Model Structure issue #189**. 2023. Disponível em: <https://github.com/ultralytics/ultralytics/issues/189>. Citado nas pp. 8, 28, 29 e 31.
- KRIZHEVSKY, A.; SUTSKEVER, I.; HINTON, G. E. Imagenet classification with deep convolutional neural networks. **Advances in neural information processing systems**, v. 25, 2012. Citado na p. 24.
- LAZAREVICH, I. **Introducing Yolobench: Benchmarking efficient object detectors on embedded systems**. Medium, 2023. Disponível em: <https://medium.com/p/ceff516ccbca>. Citado na p. 27.
- LECUN, Y.; BENGIO, Y.; HINTON, G. Deep learning. **nature**, Nature Publishing Group UK London, v. 521, n. 7553, p. 436–444, 2015. Citado na p. 24.
- LEOCÁDIO, R. R.; SEGUNDO, A. K. R.; PESSIN, G. Multiple object tracking in native bee hives: A case study with jataí in the field. *In*: SPRINGER. **Brazilian Conference on Intelligent Systems**. [S.l.], 2023. p. 176–191. Citado na p. 38.
- LI, K. *et al.* A fast and lightweight detection algorithm for passion fruit pests based on improved yolov5. **Computers and Electronics in Agriculture**, Elsevier, v. 204, p. 107534, 2023. Citado nas pp. 35 e 36.
- LI, W. *et al.* Classification and detection of insects from field images using deep learning for smart pest management: A systematic review. **Ecological Informatics**, Elsevier, v. 66, p. 101460, 2021. Citado na p. 15.
- LI, X. *et al.* Generalized focal loss: Learning qualified and distributed bounding boxes for dense object detection. **Advances in Neural Information Processing Systems**, v. 33, p. 21002–21012, 2020. Citado na p. 31.
- LIMA, B. P. d. M. *et al.* A lightweight and enhanced model for detecting the neotropical brown stink bug, *euschistus heros* (hemiptera: Pentatomidae) based on yolov8 for soybean fields. **Ecological Informatics**, Elsevier, p. 102543, 2024. Citado nas pp. 8, 9, 19, 43 e 50.
- LIN, T.-Y. *et al.* Feature pyramid networks for object detection. *In*: **Proceedings of the IEEE conference on computer vision and pattern recognition**. [S.l.: s.n.], 2017. p. 2117–2125. Citado nas pp. 8 e 30.
- LIU, S. *et al.* Path aggregation network for instance segmentation. *In*: **Proceedings of the IEEE conference on computer vision and pattern recognition**. [S.l.: s.n.], 2018. p. 8759–8768. Citado nas pp. 8 e 30.
- LIU, Z. *et al.* An improved yolov5 method for small object detection in uav capture scenes. **IEEE Access**, v. 11, p. 14365–14374, 2023. Citado nas pp. 35 e 36.

- LOU, H. *et al.* Dc-yolov8: Small size object detection algorithm based on camera sensor. Preprints, 2023. Citado nas pp. 35, 36 e 42.
- MAHAUR, B.; MISHRA, K. Small-object detection based on yolov5 in autonomous driving systems. **Pattern Recognition Letters**, Elsevier, v. 168, p. 115–122, 2023. Citado nas pp. 35 e 36.
- MAKE SENSE. **Make sense**. 2019. Disponível em: <https://www.makesense.ai/>. Citado na p. 51.
- MASUDA, T.; GOLDSMITH, P. D. World soybean production: area harvested, yield, and long-term projections. **International food and agribusiness management review**, v. 12, n. 1030-2016-82753, p. 1–20, 2009. Citado na p. 14.
- MEHTA, R. **Real-time object detection using Yolo**. 2023. Disponível em: <https://imerit.net/blog/real-time-object-detection-using-yolo/>. Citado na p. 26.
- MUNAWAR, M. R. **ByteTrack vs. Deepsort: A comparative analysis**. 2023. Disponível em: <https://www.visobyte.com/2023/07/ByteTrack-vs-Deepsort.html>. Citado na p. 39.
- ÖNLER, E. Real time pest detection using yolov5. **International Journal of Agricultural and Natural Sciences**, v. 14, n. 3, p. 232–246, 2021. Citado nas pp. 32 e 33.
- PANI, V. *et al.* Trichtrack: Multi-object tracking of small-scale trichogramma wasps. *In: IEEE. 2021 17th IEEE International Conference on Advanced Video and Signal Based Surveillance (AVSS)*. [S.l.], 2021. p. 1–8. Citado na p. 38.
- PIRES, C. S. S. *et al.* Potencial de uso de armadilhas iscadas com o feromônio sexual do percevejo marrom, *euschistus heros* (heteroptera: Pentatomidae), para o monitoramento populacional de percevejos praga da soja. **Manejo Integrado de Plagas y Agroecología Número 77 (Abril 2006)**, CATIE, Turrialba (Costa Rica), 2006. Citado na p. 23.
- PRANDO, A. M. **Pano de Batida**. Embrapa Soja, 2019. Disponível em: <https://www.embrapa.br/busca-de-imagens/-/midia/4829001/pano-de-batida>. Citado nas pp. 8 e 23.
- QIAN, Z.-M.; CHEN, X.; JIANG, H. Fish tracking based on yolo and bytetrack. *In: IEEE. 2023 16th International Congress on Image and Signal Processing, BioMedical Engineering and Informatics (CISP-BMEI)*. [S.l.], 2023. p. 1–5. Citado na p. 38.
- REDMON, J.; FARHADI, A. Yolov3: An incremental improvement. **arXiv preprint arXiv:1804.02767**, 2018. Citado na p. 28.
- 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*. [S.l.: s.n.], 2016. p. 779–788. Citado nas pp. 8, 24 e 26.
- ROBOFLOW. **Everything you need to build and deploy computer vision models**. 2020. Disponível em: <https://roboflow.com/>. Citado na p. 51.

- SABARÁ, H. H. R. **O uso de veículos aéreos não tripulados (VANT) na identificação do percevejo marrom em lavouras de soja usando técnicas de reconhecimento de padrões e aprendizado de máquinas.** Dissertação (Mestrado) — Universidade Tecnológica Federal do Paraná, 2018. Citado na p. 23.
- SEEMAKURTHY, K. *et al.* Domain generalisation for object detection. **arXiv preprint arXiv:2203.05294**, 2022. Citado na p. 38.
- SILVEIRA, F. A. G. d. *et al.* Performance analysis of yolov3 for real-time detection of pests in soybeans. *In: SPRINGER. Intelligent Systems: 10th Brazilian Conference, BRACIS 2021, Virtual Event, November 29–December 3, 2021, Proceedings, Part II.* [S.l.], 2021. p. 265–279. Citado nas pp. 32 e 33.
- STÜRMER, G. R. *et al.* Eficiência do pano-de-batida na amostragem de insetos-praga de soja em diferentes espaçamentos entre linhas e cultivares. **Semina: Ciências Agrárias**, Universidade Estadual de Londrina, v. 35, n. 3, p. 1177–1186, 2014. Citado na p. 23.
- TENÓRIO, G. L. *et al.* Comparative study of computer vision models for insect pest identification in complex backgrounds. *In: IEEE. 2019 12th International Conference on Developments in eSystems Engineering (DeSE).* [S.l.], 2019. p. 551–556. Citado na p. 33.
- TERVEN, J.; CÓRDOVA-ESPARZA, D.-M.; GONZÁLEZ, J. A. R. A comprehensive review of yolo architectures in computer vision: From yolov1 to yolov8 and yolo-nas. **Machine Learning and Knowledge Extraction**, MDPI, v. 5, n. 4, p. 1680–1716, 2023. Citado nas pp. 8, 10, 25, 26 e 27.
- TETILA, E. C. 2019. Disponível em: <https://bit.ly/2XK1XXv>. Citado nas pp. 40 e 70.
- TETILA, E. C. *et al.* A deep-learning approach for automatic counting of soybean insect pests. **IEEE Geoscience and Remote Sensing Letters**, IEEE, v. 17, n. 10, p. 1837–1841, 2019. Citado nas pp. 24 e 33.
- TETILA, E. C. *et al.* Detection and classification of soybean pests using deep learning with uav images. **Computers and Electronics in Agriculture**, Elsevier, v. 179, p. 105836, 2020. Citado na p. 24.
- VERMA, S. *et al.* Insect detection and identification using yolo algorithms on soybean crop. *In: IEEE. TENCON 2021-2021 IEEE Region 10 Conference (TENCON).* [S.l.], 2021. p. 272–277. Citado nas pp. 32 e 33.
- WANG, C.-Y.; YEH, I.-H.; LIAO, H.-Y. M. Yolov9: Learning what you want to learn using programmable gradient information. **arXiv preprint arXiv:2402.13616**, 2024. Citado nas pp. 10 e 25.
- WANG, G. *et al.* Uav-yolov8: a small-object-detection model based on improved yolov8 for uav aerial photography scenarios. **Sensors**, MDPI, v. 23, n. 16, p. 7190, 2023. Citado na p. 28.

- 
- YUAN, S. *et al.* Yolov5-ytiny: A miniature aggregate detection and classification model. **Electronics**, MDPI, v. 11, n. 11, p. 1743, 2022. Citado nas pp. 34 e 35.
- YUAN, Z.; FANG, W.; ZHAO, Y.; SHENG, V. S. Research of insect recognition based on improved yolov5. **Journal of Artificial Intelligence**, Tech Science Press, v. 3, n. 4, p. 145, 2021. Citado nas pp. 34 e 35.
- ZHAN, W. *et al.* An improved yolov5 real-time detection method for small objects captured by uav. **Soft Computing**, Springer, v. 26, p. 361–373, 2022. Citado nas pp. 34, 35 e 42.
- ZHANG, P.; ZHONG, Y.; LI, X. Slimyolov3: Narrower, faster and better for real-time uav applications. *In: Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV) Workshops*. [S.l.: s.n.], 2019. Citado na p. 37.
- ZHANG, Y. *et al.* Bytetrack: Multi-object tracking by associating every detection box. *In: SPRINGER. European Conference on Computer Vision*. [S.l.], 2022. p. 1–21. Citado nas pp. 39 e 47.
- ZHENG, Z. *et al.* Distance-iou loss: Faster and better learning for bounding box regression. *In: Proceedings of the AAAI conference on artificial intelligence*. [S.l.: s.n.], 2020. v. 34, n. 07, p. 12993–13000. Citado na p. 31.