



UNIVERSIDADE FEDERAL DE PERNAMBUCO
CENTRO DE INFORMÁTICA
PROGRAMA DE PÓS-GRADUAÇÃO EM CIÊNCIAS DA COMPUTAÇÃO

ALLAN SANTOS DE FREITAS

**Uma metodologia de contagem automática de ovos de *Aedes aegypti* a partir de
imagens obtidas por smartphone**

Recife

2025

ALLAN SANTOS DE FREITAS

Uma metodologia de contagem automática de ovos de *Aedes aegypti* a partir de imagens obtidas por smartphone

Dissertação de Mestrado apresentada ao Programa de Pós-Graduação em Ciências da Computação da Universidade Federal de Pernambuco como requisito parcial para a obtenção do título de Mestre em Ciência da Computação.

Área de Concentração: Inteligência Computacional

Orientador: Leandro Maciel Almeida

Recife

2025

.Catalogação de Publicação na Fonte. UFPE - Biblioteca Central

Freitas, Allan Santos de.

Uma metodologia de contagem automática de ovos de Aedes aegypti a partir de imagens obtidas por smartphone / Allan Santos de Freitas. - Recife, 2025.

145f.: il.

Dissertação (Mestrado)- Universidade Federal de Pernambuco, Centro de Informática, Programa de Pós-Graduação em Ciências da Computação, 2025.

Orientação: Leandro Maciel Almeida.

1. Contagem automática; 2. Aedes aegypti; 3. Aprendizagem profunda; 4. Dispositivo de captura de baixo custo; 5. Base de dados de ovos; 6. Palheta de ovitrampa. I. Almeida, Leandro Maciel. II. Título.

UFPE-Biblioteca Central

Allan Santos de Freitas

**“Uma metodologia de contagem automática de ovos de Aedes aegypti
a partir de imagens obtidas por smartphone”**

Dissertação de mestrado apresentada ao Programa de Pós-Graduação em Ciência da Computação da Universidade Federal de Pernambuco, como requisito parcial para a obtenção do título de Mestre em Ciência da Computação. Área de Concentração: Inteligência Computacional.

Aprovado em: 30/07/2025.

BANCA EXAMINADORA

Prof. Dr. Fernando Maciano de Paula Neto
Centro de Informática / UFPE

Prof. Dr. João Fausto Lorenzato de Oliveira
Escola Politécnica de Pernambuco / UPE

Prof. Dr. Leandro Maciel Almeida
Centro de Informática / UFPE
(orientador)

Dedico esse trabalho a todos os pesquisadores, profissionais de saúde e entidades, empenhados no combate ao mosquito *Aedes aegypti*, no desenvolvimento de tecnologias promissoras de combate ao mosquito e aos profissionais de saúde envolvidos no tratamento de doenças relacionadas.

AGRADECIMENTOS

Antes de tudo, agradeço ao meu Deus, que ao longo dessa trajetória me proveu de sua graça e misericórida, para que eu pudesse concluir essa jornada incrível e desafiadora. Sem ele, não chegaria até aqui.

Aos meus pais, que me aconselharam e me motivaram a suplantar os obstáculos e dificuldades que enfrentei. Estendo, também, os meus agradecimento a toda a minha família.

Ainda, sou imensamente grato à minha amada noiva e futura esposa, Ester, cujo apoio, presença, amizade, motivação, alegria, otimismo e bons conselhos nos momentos mais importantes me possibilitaram alcançar a conclusão deste trabalho.

Também expresso aos meus amigos e irmãos em Cristo que frequentemente me fortaleciam a esperança, a despeito de todas os desafios a serem enfrentados.

Sou muito agradecido aos meus colegas do Centro de Informática com os quais pude aprender e cooperar ao longo da jornada como discente.

Aos professores que, cada um em sua área do conhecimento, me abriram horizontes na Inteligência Computacional, instigando-me a explorar ainda mais essa fascinante área, bem como apreciá-la.

Agradeço a professora Aida Ferreira pela oportunidade de participar do programa de Iniciação Científica durante à graduação, bem como aos meus colegas do LabGeo no IFPE com quem pude aprender bastante.

Minha gratidão aos meus colegas da Emprel que me incentivavam, especialmente nos momentos finais, como Dani, Cinthya, Diego, dentre tantos outros.

Também registro meus agradecimentos à FIOCRUZ-PE e ao Instituto Aggeu Magalhães pela concessão de numerosas palhetas de ovitrampas utilizadas neste trabalho; sem essa contribuição e colaboração, este trabalho não seria possível.

Por fim, não poderia deixar de agradecer ao meu professor e orientador Leandro Almeida, que, ao longo desse tempo, bem me encaminhou e me instruiu no decorrer desse curso.

RESUMO

Insetos transmissores de doenças, como o *Aedes aegypti*, causam danos significativos em várias áreas da sociedade. Doenças relacionadas, como Dengue, Zika e Chikungunya têm causado prejuízos sanitários, sociais e econômicos ao redor do globo, principalmente em países tropicais. Dentre os métodos de combate, a vigilância entomológica provê mecanismos preditivos baseados em indicadores-chave, envolve a metodologia de ovitrampas e consequente contagem manual de ovos do mosquito em palhetas de ovitrampa para obtenção de informações para posterior tomada de decisões no combate ao mosquito. Problemas inerentes à laboriosa contagem manual dos ovos demandam métodos acurados de contagem automática. Este trabalho propõe o desenvolvimento de uma metodologia de contagem automática de ovos de *Aedes aegypti* em palhetas de ovitrampa de viável aplicação operacional, de baixo custo e que, além de resolver problemas fundamentais da captura de imagens, como dificuldades de iluminação, resolve também problemas frequentes na literatura, como a captura redundante de regiões da palheta. Abordagens anteriores promoveram considerável avanço no desenvolvimento de métodos de contagem automática, restando, entretanto, numerosas dificuldades e pontos de melhoria. Contribuições importantes desta pesquisa incluem a definição de uma metodologia de contagem de ovos baseada em redes neurais de detecção de objetos, além da criação de um dispositivo de baixo custo para captura adequada de imagens em palhetas de ovitrampa, a criação de base de dados numericamente significativa dessas imagens, avaliação experimental do impacto de técnicas de pré-processamento baseadas em processamento de imagens, bem como de rede neural de restauração de imagens degradadas. Também, treinamento e avaliação de modelos, como D-FINE e RF-DETR, obtendo erro médio de 91,26 ovos e -2,77% de erro global percentual no reconhecimento de ovos em imagens com ruídos expressivos, bem como, por fim, o uso de quantidade significativa de 28 palhetas como conjunto independente de validação da aplicabilidade desta metodologia em cenários de uso real, número bastante expressivo considerando-se trabalhos encontrados na literatura.

Palavras-chave: Contagem automática. *Aedes aegypti*. Aprendizagem Profunda. Dispositivo de captura de baixo custo. Base de dados de ovos. Palheta de ovitrampa.

ABSTRACT

ABSTRACT

Disease-transmitting insects, such as *Aedes aegypti*, cause significant damage in various areas of society. Related diseases such as Dengue, Zika, and Chikungunya have led to health, social, and economic losses worldwide, especially in tropical countries. Among the control methods, entomological surveillance provides predictive mechanisms based on key indicators, involving the use of ovitraps and subsequent manual counting of mosquito eggs on oviposition paddles to support decision-making in mosquito control. Problems inherent in the labor-intensive manual counting process demand accurate automatic counting methods. This work proposes the development of an automatic *Aedes aegypti* egg-counting methodology for oviposition paddles, with low cost and operational feasibility, which, in addition to solving fundamental issues in image acquisition such as lighting difficulties, also addresses frequent problems reported in the literature, including redundant capture of paddle regions. Previous approaches have promoted considerable progress in developing automatic counting methods; however, numerous challenges and areas for improvement remain. Important contributions of this research include the definition of an egg-counting methodology based on object detection neural networks, the development of a low-cost device for proper image acquisition in oviposition paddles, the creation of a numerically significant image dataset, and an experimental evaluation of the impact of preprocessing techniques based on image processing, as well as a neural network for restoring degraded images. Additionally, training and evaluation of models such as D-FINE and RF-DETR achieved a mean error of 91.26 eggs and a mean percentage error of -2.77% in recognizing eggs in images with expressive noise. Finally, a substantial number of 28 paddles were used as an independent set to validate the applicability of this methodology in real-world scenarios, a figure that is considerably significant considering works reported in the literature.

Keywords: Automatic counting. *Aedes aegypti*. Deep learning. Low-cost capture device. Egg database. Oviposition paddle.

LISTA DE FIGURAS

| | |
|--|----|
| Figura 1 – Ovos de <i>Aedes aegypti</i> | 26 |
| Figura 2 – Ovitrampa | 31 |
| Figura 3 – Sistema Visual Humano | 37 |
| Figura 4 – Semelhanças entre neurônios biológicos e sistemas artificiais | 38 |
| Figura 5 – Rede Neural Artificial | 39 |
| Figura 6 – Diagrama de Perceptron | 40 |
| Figura 7 – Multi-Layer Perceptron | 40 |
| Figura 8 – Arquitetura em alto nível de uma CNN | 42 |
| Figura 9 – Exemplo de Classificação de Imagens | 43 |
| Figura 10 – Exemplo de Detecção de Objetos | 44 |
| Figura 11 – Arquitetura do YOLOv10 | 46 |
| Figura 12 – Exemplo de aplicação de <i>Area Attention</i> do YOLOv12 | 47 |
| Figura 13 – Classificação das metodologias de modelos de detecção de objetos. A figura exibe de baixo para cima (em ordem cronológica), as metodologias utilizadas. Destaca-se as abordagens do estado da arte, como <i>CNN-based</i> e <i>Transformer-based methods</i> , os quais são amplamente utilizados atualmente, além dos VLMs, que estão em uso crescente. | 49 |
| Figura 14 – Arquitetura do RF-DETR | 50 |
| Figura 15 – Visão geral do componente FDR do modelo D-FINE: Observa-se que as distribuições de probabilidade operam como uma representação intermediária de granulação mais fina e que posteriormente são refinadas pelas camadas de decodificação considerando as conexões residuais na arquitetura da rede. | 51 |
| Figura 16 – Arquitetura multiestágio do MPRNet: Observa-se que os estágios primeiros extram características multiescala contextualizadas e os últimos estágios atuam na resolução da imagem original para se obter saídas precisas do ponto de vista espacial. | 54 |
| Figura 17 – Fluxo Completo de Contagem | 70 |
| Figura 18 – Palheta de ovitrampa | 71 |
| Figura 19 – Câmera traseira - Samsung A03S | 72 |
| Figura 20 – Segmento de palheta com iluminação não uniforme | 73 |

| | |
|--|-----|
| Figura 21 – Divisão de segmentos com áreas comuns | 74 |
| Figura 22 – Suporte de captura de imagens de palheta de ovitrampa | 75 |
| Figura 23 – Suporte de captura de imagens de palheta de ovitrampa com <i>smartphone</i> posicionado | 75 |
| Figura 24 – Características do Suporte de Captura | 76 |
| Figura 25 – Plataforma inferior do Suporte de Captura de imagens de palheta de ovitrampa | 76 |
| Figura 26 – Imagens resultantes após uso do Suporte de Captura | 78 |
| Figura 27 – Exemplo de detecção de ovo após aplicação de <i>zoom</i> | 78 |
| Figura 28 – Processo de geração de <i>crops</i> a partir de um segmento de palheta | 79 |
| Figura 29 – Exemplo de <i>crop</i> obtido | 80 |
| Figura 30 – Exemplo de imagens da base criada por (SILVA, 2021) | 82 |
| Figura 31 – Exemplo de imagem ruidosa de palheta, especialmente com iluminação irregular | 84 |
| Figura 32 – Sub-rede <i>encoder-decoder</i> | 86 |
| Figura 33 – Bloco Original Resolution Block (ORB) da sub-rede Original Resolution Subnetwork (ORSNet) | 86 |
| Figura 34 – Cross-stage Feature Fusion (CSFF) entre os estágios 1 e 2 | 87 |
| Figura 35 – CSFF entre os estágios 2 e 2 | 87 |
| Figura 36 – Módulo Supervised Attention Module (SAM) | 88 |
| Figura 37 – Imagens com a presença de ruído de borramento | 89 |
| Figura 38 – Imagens de palheta de ovitrampa após aplicação de variações de contraste, brilho e saturação | 90 |
| Figura 39 – Definição de <i>Intersection over Union</i> (IoU) | 101 |
| Figura 40 – Exemplo de valores de <i>Intersection over Union</i> (IoU) | 102 |
| Figura 41 – Impacato da restauração do ruído de borramento com MPRNet no reconhecimento de ovos | 110 |

LISTA DE TABELAS

| | |
|---|-----|
| Tabela 1 – Comparativo de trabalhos na literatura que utilizam Processamento de Imagens para contagem automática de ovos de <i>A. aegypti</i> | 60 |
| Tabela 2 – Comparativo de trabalhos na literatura que utilizam Aprendizagem Profunda para contagem automática de ovos de <i>A. aegypti</i> | 67 |
| Tabela 3 – Divisão do dataset de ovos de <i>Aedes aegypti</i> em palhetas de ovitrampa . . | 81 |
| Tabela 4 – Divisão das imagens do dataset derivado | 83 |
| Tabela 5 – Principais hiperparâmetros utilizados no treinamento do YOLOv10-M . . . | 93 |
| Tabela 6 – Técnicas de <i>Data Augmentation</i> aplicadas no treinamento do YOLOv10-M | 93 |
| Tabela 7 – Hiperparâmetros utilizados no treinamento do modelo YOLOv12-M | 95 |
| Tabela 8 – Técnicas de <i>Data Augmentation</i> aplicadas no treinamento do YOLOv12-M | 96 |
| Tabela 9 – Principais hiperparâmetros do treinamento do modelo RF-DETR-B | 97 |
| Tabela 10 – Hiperparâmetros utilizados no treinamento do modelo D-FINE-M | 99 |
| Tabela 11 – Técnicas de <i>Data Augmentation</i> aplicadas no treinamento do D-FINE-M . | 99 |
| Tabela 12 – Palhetas contadas manualmente por técnicos da FIOCRUZ e quantidade de ovos correspondente. | 106 |
| Tabela 13 – Protocolo de experimentos do modelo base YOLOv10-M em relação às etapas de pré-processamento | 107 |
| Tabela 14 – Desempenho do base YOLOv10-M nos conjuntos de treino, validação e teste | 109 |
| Tabela 15 – Desempenho do modelo base na contagem de ovos por palheta com MPR-Net como pré-processamento | 112 |
| Tabela 16 – Métricas globais de desempenho do modelo base com MPRNet | 113 |
| Tabela 17 – Desempenho do modelo base na contagem de ovos por palheta com <i>Color Jitter</i> como pré-processamento | 114 |
| Tabela 18 – Métricas globais de desempenho do modelo YOLOv10-M com <i>Color Jitter</i> | 115 |
| Tabela 19 – Desempenho do modelo YOLOv10-M na contagem de ovos por palheta com Equalização de histograma como pré-processamento | 116 |
| Tabela 20 – Métricas globais de desempenho do modelo YOLOv10-M com Equalização de histograma | 117 |
| Tabela 21 – Desempenho do modelo YOLOv10-M na contagem de ovos por palheta com CLAHE como pré-processamento | 118 |

| | |
|--|-----|
| Tabela 22 – Diferença entre contagem do modelo base e contagem de referência entre os grupos de palhetas | 119 |
| Tabela 23 – Métricas globais de desempenho do modelo YOLOv10-M com <i>CLAHE</i> . . | 119 |
| Tabela 24 – Desempenho do modelo YOLOv10-M na contagem direta de ovos em palheta de ovitrampa | 120 |
| Tabela 25 – Métricas globais de desempenho do modelo YOLOv10-M na contagem direta | 121 |
| Tabela 26 – Desempenho do modelo YOLOv12-M na contagem direta de ovos em palheta de ovitrampa | 122 |
| Tabela 27 – Métricas globais de desempenho do modelo YOLOv12-M na contagem direta | 123 |
| Tabela 28 – Desempenho do modelo RF-DETR-B na contagem direta de ovos em palhetas de ovitrampa | 124 |
| Tabela 29 – Faixas de diferença de ovos e quantidade de palhetas na contagem com RF-DETR-B | 125 |
| Tabela 30 – Diferença entre contagem do RF-DETR-B e contagem de referência entre os grupos de palhetas | 125 |
| Tabela 31 – Métricas globais de desempenho do modelo RF-DETR-B na contagem direta de ovos em palheta de ovitrampa | 126 |
| Tabela 32 – Desempenho do modelo D-FINE-M na contagem direta de ovos em palhetas de ovitrampa | 127 |
| Tabela 33 – Métricas globais de desempenho do modelo D-FINE-M na contagem direta de ovos em palhetas de ovitrampas | 128 |
| Tabela 34 – Comparativo de trabalhos na literatura para contagem de ovos de <i>A. aegypti</i> | 130 |

LISTA DE ABREVIATURAS E SIGLAS

| | |
|----------------|--|
| AM | Aprendizagem de Máquina |
| AP | Average Precision |
| CLAHE | Contrast Limited Adaptive Histogram Equalization |
| CNN | Convolutional Neural Network |
| COCO | Common Objects in Context |
| CSFF | Cross-stage Feature Fusion |
| CSPNet | Cross Stage Partial Network |
| DETR | Deformable Transformers |
| DL | Deep Learning |
| DNN | Deep Neural Network |
| EDL | Estação Disseminadora de Larvicida |
| ELAN | Efficient Layer Aggregation Networks |
| FDR | Fine-Grained Distribution Refinement |
| Fiocruz | Fundação Oswaldo Cruz |
| FPS | Frames Per Second |
| GO-LSD | Global Optimal Localization Self-Distillation |
| HDMI | High-Definition Multimedia Interface |
| HSV | Hue, Saturation, Value |
| IA | Inteligência Artificial |
| IDO | Índice de Densidade de Ovo |
| IDV | Índice de Densidade Vetorial |
| IoU | Intersection over Union |
| IPO | Índice de Positividade de Ovo |
| JPG | Joint Photographic Experts Group |
| LED | Light-Emitting Diode |

| | |
|-------------------|--|
| LIA | Levantamento de Índice Amostral |
| LIRa | Levantamento Rápido de Índices para <i>Aedes aegypti</i> |
| MAE | Mean Absolute Error |
| mAP | mean Average Precision |
| mAR | Mean Average Recall |
| Mask R-CNN | Mask Region-based Convolutional Neural Network |
| MLP | Multi-Layer Perceptron |
| MPRNet | Multi-Stage Progressive Image Restoration |
| MSE | Mean Squared Error |
| NMS | Non-Maximum Suppression |
| ORB | Original Resolution Block |
| ORSNet | Original Resolution Subnetwork |
| PAN | Path Aggregation Network |
| PLN | Processamento de Linguagem Natural |
| PR Curve | Precision-Recall Curve |
| PSA | Partial Self-Attention |
| R-ELAN | Residual Efficient Layer Aggregation Networks |
| R-FCN | Region-based Fully Convolutional Networks |
| RGB | Red, Green, Blue |
| RNA | Rede Neural Artificial |
| RT-DETR | Real-Time DEtection TRansformer |
| SABL | Side-Aware Boundary Localization |
| SAM | Supervised Attention Module |
| SIF | Sistema de Inferência Fuzzy |
| TIE | Técnica do Inseto Estéril por Irradiação |
| TN | True Negative |
| USB | Universal Serial Bus |

VLMs Vision Language Models

YOLO You Only Look Once

LISTA DE SÍMBOLOS

\in Pertence

SUMÁRIO

| | | |
|----------|---|-----------|
| 1 | INTRODUÇÃO | 20 |
| 1.1 | ORGANIZAÇÃO DO DOCUMENTO | 23 |
| 1.2 | OBJETIVOS | 23 |
| 1.3 | PRINCIPAIS CONTRIBUIÇÕES | 24 |
| 2 | FUNDAMENTAÇÃO TEÓRICA | 25 |
| 2.1 | O MOSQUITO AEDES AEGYPTI | 25 |
| 2.1.1 | Doenças relacionadas | 27 |
| 2.1.2 | Dengue | 27 |
| 2.1.3 | Chikungunya | 27 |
| 2.1.4 | Zika | 28 |
| 2.2 | SISTEMAS DE VIGILÂNCIA | 29 |
| 2.2.0.1 | <i>Vigilância Entomológica</i> | 29 |
| 2.2.0.2 | <i>Ovitrampa</i> | 30 |
| 2.2.0.3 | <i>Monitoramento</i> | 31 |
| 2.2.0.4 | <i>Contagem manual de ovos de A. aegypti</i> | 32 |
| 2.3 | PROCESSAMENTO DE IMAGENS | 33 |
| 2.3.1 | Realce de Imagens | 34 |
| 2.3.2 | Color Jitter | 34 |
| 2.3.3 | Processamento de histograma | 35 |
| 2.3.4 | Equalização de Histograma | 35 |
| 2.3.5 | Contrast Limited Adaptative Histogram Equalization (CLAHE) | 35 |
| 2.4 | VISÃO COMPUTACIONAL | 36 |
| 2.4.1 | Aprendizagem Profunda | 38 |
| 2.4.2 | Redes Neurais Convolucionais | 41 |
| 2.4.3 | Modelos de Detecção de Objetos | 42 |
| 2.4.4 | YOLOv10 | 44 |
| 2.4.5 | YOLOv12 | 46 |
| 2.4.6 | RF-DETR | 48 |
| 2.4.7 | D-FINE | 50 |
| 2.4.8 | MPRNet | 52 |

| | | |
|--------------|---|-----------|
| 2.5 | CONCLUSÃO DO CAPÍTULO | 54 |
| 3 | TRABALHOS RELACIONADOS | 55 |
| 3.1 | CONTAGEM AUTOMÁTICA COM PROCESSAMENTO DE IMAGENS | 55 |
| 3.2 | CONTAGEM AUTOMÁTICA COM APRENDIZAGEM PROFUNDA | 60 |
| 3.3 | CONCLUSÃO DO CAPÍTULO | 68 |
| 4 | MATERIAIS E MÉTODOS | 69 |
| 4.1 | FLUXO DE DESENVOLVIMENTO DO TRABALHO | 69 |
| 4.2 | AQUISIÇÃO DE PALHETAS | 70 |
| 4.3 | CAPTURA DE IMAGENS | 71 |
| 4.3.1 | Dispositivo móvel de captura | 71 |
| 4.3.1.1 | <i>Limitações da captura direta por smartphone</i> | 72 |
| 4.3.2 | Suporte de captura desenvolvido | 74 |
| 4.3.3 | Geração de grids (crops) | 77 |
| 4.3.4 | Base de imagens | 81 |
| 4.3.5 | Base de dados criada | 81 |
| 4.3.6 | Base de dados derivada | 82 |
| 4.3.7 | Conjunto de palhetas contadas | 83 |
| 4.3.7.1 | <i>Desafios à contagem automática</i> | 83 |
| 4.4 | MÉTODOS DE CONTAGEM AUTOMÁTICA COM APRENDIZAGEM PROFUNDA | 84 |
| 4.4.1 | Ambiente de treinamento | 84 |
| 4.5 | FLUXO I: CONTAGEM ALIADA À APLICAÇÃO DE MODELOS DE RESTAURAÇÃO DE IMAGENS BASEADOS EM APRENDIZAGEM PROFUNDA | 85 |
| 4.5.1 | MPRNet | 85 |
| 4.6 | FLUXO II: CONTAGEM ALIDADA A TÉCNICAS DE PROCESSAMENTO DE IMAGENS | 89 |
| 4.6.1 | Aplicação inicial de técnicas de realce de imagens | 89 |
| 4.6.1.1 | <i>Color Jitter</i> | 90 |
| 4.6.1.2 | <i>Equalização de Histograma</i> | 91 |
| 4.6.1.3 | <i>Contrast Limited Adaptive Histogram Equalization (CLAHE)</i> | 91 |
| 4.7 | FLUXO III: CONTAGEM DIRETA COM REDES NEURAIS | 92 |
| 4.7.1 | YOLOv10 | 92 |
| 4.7.2 | YOLOv12 | 94 |

| | | |
|---------|--|-----|
| 4.7.3 | RF-DETR | 96 |
| 4.7.4 | D-FINE | 97 |
| 4.8 | MÉTRICAS DE AVALIAÇÃO DE DESEMPENHO | 100 |
| 4.8.1 | <i>Intersection Over Union (IoU)</i> | 101 |
| 4.8.2 | <i>mean Average Precision (mAP)</i> | 102 |
| 4.8.3 | <i>Recall</i> | 102 |
| 4.8.4 | <i>Precision</i> | 103 |
| 4.8.5 | <i>F1-Score</i> | 103 |
| 4.8.6 | Contagem absoluta | 103 |
| 4.8.6.1 | <i>Mean Absolute Error (MAE)</i> | 104 |
| 4.8.6.2 | <i>Mean Squared Error (MSE)</i> | 104 |
| 4.8.6.3 | <i>Erro Relativo Percentual</i> | 104 |
| 4.9 | PROTOCOLO EXPERIMENTAL | 105 |
| 4.9.1 | Base para treinamento de modelos | 105 |
| 4.9.2 | Palhetas contadas | 105 |
| 4.9.3 | Comparação dos fluxos | 105 |
| 4.9.4 | Avaliação final | 107 |
| 4.9.5 | Métricas de comparação | 107 |
| 4.10 | CONCLUSÃO DO CAPÍTULO | 108 |
| 5 | RESULTADOS E DISCUSSÃO | 109 |
| 5.1 | FLUXOS I E II: AVALIAÇÃO DAS ETAPAS DE PRÉ-PROCESSAMENTO | 109 |
| 5.1.1 | Desempenho do modelo base YOLOv10-M no treinamento | 109 |
| 5.1.2 | Pré-processamento com MPRNet | 110 |
| 5.1.2.1 | <i>Desempenho do modelo base com MPRNet</i> | 111 |
| 5.1.3 | Pré-processamento com Color Jitter | 113 |
| 5.1.3.1 | <i>Desempenho do modelo base com Color Jitter</i> | 114 |
| 5.1.4 | Pré-processamento com Equalização de histograma | 115 |
| 5.1.4.1 | <i>Desempenho do modelo base com Equalização de histograma</i> | 115 |
| 5.1.5 | Pré-processamento com CLAHE | 117 |
| 5.1.5.1 | <i>Desempenho do modelo base com CLAHE</i> | 117 |
| 5.2 | FLUXO III: CONTAGEM DIRETA SEM PRÉ-PROCESSAMENTO | 119 |
| 5.2.1 | YOLOv10-M | 119 |
| 5.2.2 | YOLOv12-M | 121 |

| | | |
|-------|---|-----|
| 5.2.3 | RF-DETR-B | 123 |
| 5.2.4 | D-FINE-M | 126 |
| 5.3 | AVALIAÇÃO FINAL | 128 |
| 5.3.1 | Comparativo com trabalhos da literatura na contagem de ovos de Ae. aegypti | 129 |
| 6 | CONCLUSÃO E TRABALHOS FUTUROS | 132 |
| | REFERÊNCIAS | 135 |

1 INTRODUÇÃO

O *Aedes aegypti* é um mosquito transmissor de doenças como dengue, febre amarela, chikungunya e zika, além de, associado a esta última, poder causar microcefalia por infecção congênita (ARAÚJO et al., 2018). Sua provável origem é o Nordeste da África, especificamente o Egito (PESSÔA; MARTINS, 1982), de modo que foi introduzido no Brasil durante o período colonial, provavelmente por meio do transporte marítimo, em especial, de navios negreiros (PAIXAO, 2007) (CONSOLI; OLIVEIRA, 1994). O mosquito incide sobre toda faixa tropical e sub-tropical do planeta e, por ser considerado cosmopolita, acompanha o ser humano e, desse modo, manifesta-se principalmente em regiões urbanas, tanto no domicílio como também no peridomicílio humanos.

Historicamente, o Brasil tem sofrido surtos e epidemias de arboviroses transmitidas pelo *Ae. aegypti*, a exemplo da tríplice epidemia dos vírus zika, dengue e chikungunya, ocorrida em 2016 e da epidemia de dengue em 2019. No País, no período compreendido entre 2008 e 2019, foram notificados cerca de 11,6 milhões de casos suspeitos de dengue, zika e chikungunya, sendo, somente a dengue, responsável por cerca de 10,6 milhões desse total (SAÚDE, 2020). Os dois anos seguintes foram marcados pela queda acentuada no número de casos dessas doenças. Porém, em 2022, houve um aumento abrupto, de modo que somente o número de casos notificados de dengue atingiu cerca de 1,4 milhões até a semana epidemiológica 51, com um total de 1.053 mortes no ano. Isto põe em clareza o constante risco de ressurgimento de surtos e epidemias de arboviroses no País (Ministério da Saúde – Secretaria de Vigilância em Saúde, 2022; SAÚDE, 2024).

Ainda mais, em 2023, o número de casos de dengue teve ainda aumento, de maneira a atingir cerca de 1,6 milhões de casos prováveis e 1094 mortes confirmadas (Ministério da Saúde – Brasil, 2024b). No ano seguinte, o número de casos teve expressivo aumento e alcançou cerca de 6,5 milhões de casos prováveis de dengue e 6.297 óbitos (o maior da série histórica), o que representa um aumento de 400% em relação ao ano de 2023 (Ministério da Saúde – Brasil, 2024a). Isso fez com que o número de óbitos por dengue em 2024 superasse a quantidade de óbitos por COVID-19 no mesmo ano (Ministério da Saúde – Brasil, 2025a). Portanto, dessa forma, como constatado pelas autoridades sanitárias, o Brasil tem enfrentado consecutivas epidemias de dengue ao longo dos últimos três anos (2022, 2023 e 2024) (Ministério da Saúde – Secretaria de Vigilância em Saúde e Ambiente, 2024). Por fim, somente até a 27^a semana epidemiológica de

2025, já foram notificados mais de 1,6 milhões de casos prováveis de dengue com 1.420 óbitos ao total, além de mais de 3.900 casos de Zika e cerca de 109 mil de Chikungunya (Ministério da Saúde – Brasil, 2024a).

Ante esse cenário, os sistemas de vigilância provêm meios capazes de se detectar, com antecedência, o surgimento de novos surtos de doenças transmitidas pelo vetor. Por um lado, a vigilância epidemiológica estuda, entre outras coisas, o processo da saúde-doença na sociedade (FILHO; ROUQUAYROL, 1992), de modo a tratar-se da observação sistemática e ativa de casos suspeitos ou confirmados de doenças transmissíveis e suas causas. Tal abordagem, entretanto, configura-se como um método tardio, pois suas ações visam primeiro conhecer fatores determinantes na saúde coletiva ou individual, para só então adotar ações de controle dessas doenças (ACOSTA, 2016). Por outro lado, a vigilância entomológica abrange escolha, coleta e acompanhamento de dados de indicadores eleitos associados ao vetor de doenças com o objetivo de determinar mudanças na distribuição geográfica de um espaço, no sentido de se obter dados da população do mosquito ao longo do tempo numa dada região, além de fornecer informações que deêm suporte a oportunas tomadas de decisão no âmbito de programas de combate ao vetor. (ORGANIZATION et al., 2016).

A vigilância entomológica faz uso de duas técnicas, a saber: Pesquisa larvária e vigilância pela presença de ovos em ovitrampas. A primeira consiste na busca de larvas de mosquitos em recipientes com água, e em levantamentos quantitativos a partir desses dados. Por outro lado, a segunda, trata-se de contagem de ovos do mosquito depositados em armadilhas especiais chamadas ovitrampas.

Nesse contexto, ovitrampa consiste numa armadilha de deposição de ovos do mosquito (ou ovoposição) e é formada por um recipiente cilíndrico preto abastecido de água de torneira no seu interior, tendo nele fixado um substrato de ovoposição, na forma de uma palheta de superfície áspera que serve para deposição de ovos pela fêmea adulta do mosquito. Essa técnica possibilita, através da contagem de ovos de *Aedes aegypti* nas palhetas de ovitrampa, a geração de inferências acerca da densidade do vetor em regiões específicas antes que seu adensamento populacional resulte no aparecimento de doenças relacionadas. Desenvolvida nos Estados Unidos, em estudos partir de 1965 (FAY; PERRY et al., 1965; FAY; ELIASON et al., 1966), tal ferramenta vem, desde então, sendo amplamente utilizada em várias partes do mundo, (PERICH et al., 2003), também sendo mais encontrada na literatura, além de aplicada por municípios no Brasil, por recomendação do Ministério da Saúde.

Desta forma, a metodologia de implementação de ovitrampas envolve como etapa indis-

pensável a contagem manual de ovos de *Aedes aegypti* para a obtenção de índices de ovos. Estes indicadores (e a precisão deles) são cruciais para, no âmbito da vigilância entomológica, realizar o monitoramento periódico, a obtenção de informações sobre a infestação dos mosquitos de forma oportuna para direcionar ações de controle vetorial a partir do bloqueio de focos. Também, a implementação e avaliação de tecnologias de controle, como o uso de Estação Disseminadora de Larvicida (EDL), do método Wolbachia e da Técnica do Inseto Estéril por Irradiação (TIE) e visita de casa em casa em áreas prioritárias dependem obrigatoriamente da prévia aplicação de ovitrampas (Ministério da Saúde – Brasil, 2025d).

Entretanto, métodos tradicionais de contagem manual, visual e não-automática são frequentemente sujeitos a erros e vieses, além de serem laboriosos, tediosos e demorados e poderem levar ao acúmulo de palhetas (MAINS; MERCER; DOBSON, 2008), o que, em última instância, pode atrasar o monitoramento e ações de controle vetorial. Desta forma, diversos métodos de contagem automática têm sido estudados na literatura com vistas a acelerar o tempo de contagem e obter performance satisfatória e indicadores fidedignos de forma tempestiva e menos laboriosa. Então, métodos de contagem automática de ovos do *Aedes aegypti*, além de aumentar a confiabilidade da análise, podem favorecer a tomada de ações de controle em tempo hábil para evitar surtos da doença.

Outrossim, esta contagem automática apresenta desafios peculiares que envolvem o uso de dispositivos de captura, geralmente custosos, baseados em componentes eletrônicos para auxiliar na captura como, uso de sistemas de iluminação, para assim, obterem-se imagens digitais para contagem automática. Também a contagem automática a partir de imagens é, em si, desafiadora e alvo de pesquisas ao longo dos anos, a exemplo de (SANTOS et al., 2008; FEITOSA et al., 2015; SILVA, 2021; VICENTE et al., 2024), com limitações e pontos de melhoria ainda observados, como, a captura redundante de regiões da palheta pelo dispositivo de captura, o aumento expressivo de erro de contagem em imagens ruidosas, bases de validação final da solução com poucas imagens, dentre outros. Nesse sentido, atualmente, há a necessidade do desenvolvimento de uma metodologia de contagem que envolva, tanto o uso de dispositivos eficientes, mas também de baixo custo, fácil transporte e manuseio, como também o uso de métodos computacionais mais eficientes, do estado da arte, a exemplo de modelos baseados em *Transformers*, além da aplicação de mecanismos de *attention* para contagem de automática de ovos de *A. aegypti* em palhetas de ovitrampas, para tornar, por fim, mais célere o monitoramento e aplicação de ações de controle no âmbito da implementação da vigilância entomológica.

À luz disso, este trabalho ocupa-se de desenvolver dispositivo eficiente de baixo custo e de fácil uso operacional para captura de imagens em palhetas de ovitrampa, aliado à criação de uma base de tamanho expressivo, além do uso de métodos de inteligência computacional para o reconhecimento de ovos de *A. aegypti* em palhetas de ovitrampa com performance significativa, em especial de redes neurais de restauração de imagens degradadas, como o MPRNet, modelos de detecção de objetos baseados em *Transformers*, como RF-DETR. Estas ferramentas propostas visam auxiliar na aplicação da metodologia do uso de ovitrampas e contribuir para a celeridade na obtenção de indicadores precisos, que por sua vez, no âmbito da vigilância entomológica, servem de base para o direcionamento de ações efetivas no monitoramento e controle do mosquito vetor de doenças como Dengue, Zika e Chikungunya e febre amarela.

1.1 ORGANIZAÇÃO DO DOCUMENTO

Este documento está posto estruturado da seguinte forma: O capítulo 1 contém a introdução e relata os objetivos deste trabalho. No capítulo 2, está contida a fundamentação teórica, onde são expostos os conceitos a serem tratados noutros capítulos. O capítulo 3 apresenta os trabalhos relacionados à contagem automática de ovos de *Aedes aegypti*. Por sua vez, o capítulo 4, apresenta a metodologia empregada neste trabalho e no capítulo 5 são expostos os resultados. No capítulo 6, é feita a conclusão e relato de trabalhos futuros.

1.2 OBJETIVOS

O objetivo geral desta pesquisa é propor uma metodologia de contagem automática de ovos de *Aedes aegypti* em ovitrampas a partir de imagens obtidas por *smartphone*, aliada ao uso de dispositivo de captura de baixo custo.

Para alcançar este objetivo, seguem-se os objetivos específicos:

- a) Definir um protocolo de contagem automática que envolva captura de imagens de palhetas com *smartphone* e ferramentas de baixo custo e de fácil manuseio operacional;
- b) Desenvolver um dispositivo de baixo custo para auxiliar a captura adequada de imagens em palhetas de ovitrampa que resolva problemas comuns apontados na literatura;
- c) Criar uma base de dados a partir de palhetas de ovitrampa capturadas usando a metodologia definida e dispositivo de captura de baixo custo;

-
- d) Treinar modelos de detecção de objetos do estado da arte para a tarefa de reconhecimento de ovos de *Aedes aegypti* usando a base de dados criada;
 - e) Tratar imagens ruidosas com técnicas de processamento de imagens;
 - f) Lidar com imagens degradadas e de baixa resolução usando rede neural de restauração de imagens;
 - g) Testar o desempenho dos modelos de contagem de ovos de *A. aegypti* em palhetas contadas por técnicos da Fundação Oswaldo Cruz (Fiocruz) em Pernambuco.

1.3 PRINCIPAIS CONTRIBUIÇÕES

- a) Definição de um protocolo de contagem de ovos de *A. aegypti* de melhor uso operacional e de baixo custo;
- b) Criação de um dispositivo de captura de baixo custo que possibilita a captura de imagens da superfície de uma palheta de ovitrampa com iluminação adequada e de forma a evitar sobreposição de regiões da palheta (redundância de captura);
- c) Criação de uma base de dados de imagens de palhetas de ovitrampa para treinamento de modelos de aprendizagem profunda;
- d) Testar técnicas de processamento de imagens e rede neural de restauração de imagens para lidar com imagens de palhetas com ruídos;
- e) Aplicar redes neurais do estado da arte voltadas à detecção de ovos de *A. aegypti*, como D-FINE, RF-DETR e da série de modelos YOLO num protocolo de contagem de baixo custo e relacionado a imagens de baixa resolução capturadas por *smartphone*.

2 FUNDAMENTAÇÃO TEÓRICA

2.1 O MOSQUITO AEDES AEGYPTI

Ao longo da história, a espécie humana tem ocupado espaços que já eram *habitat* de várias outras espécies, o que pode causar, tanto a extinção das espécies que tiveram seus *habitat* invadido, como também causar o comensalismo. Nesse último caso, as espécies não humanas permanecem se reproduzindo em territórios ocupados pelo homem e, tratando-se de espécies de insetos que se alimentam de sangue de animais vertebrados (preferencialmente humano), essa combinação pode gerar consequências graves (POWELL; TABACHNICK, 2013).

Uma vez que insetos transmissores de doenças humanas são partícipes dessa domesticação, de modo a viverem em proximidade com humanos (antropofilia), ocorre uma mudança na preferência da fonte de sangue, deixando de ser de fonte animal (zoofagia) para ser de fonte humana (antropofagia). Tal permuta, faz com que humanos sofram com doenças infecciosas antes restritas aos animais. Isso acarreta também que, os insetos vetores de doenças passar a ser espécies invasoras, de modo que, não só passam acompanhar o homem em seu domicílio e peridomicílio, como também ser espalhados por ele à medida em que o homem ocupa outros espaços geográficos (LOUNIBOS, 2002).

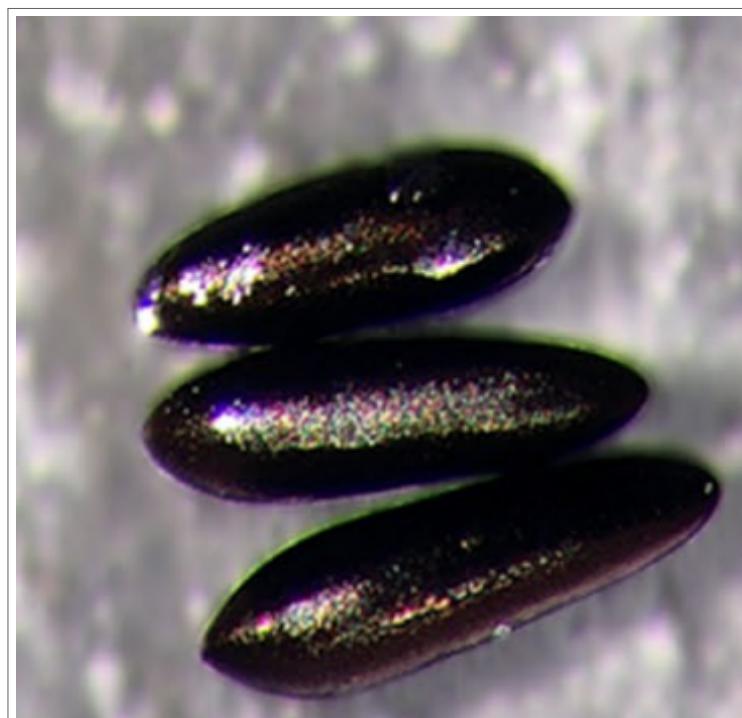
Nessa circunstância, insere-se o mosquito *Aedes aegypti* ("odioso do Egito"), vetor de doenças como febre amarela, dengue, chikungunya e zika (Ministério da Saúde – Brasil, 2023). Descrito cientificamente pela primeira vez em 1757 por Fredrik Hasselqvist (HASSELQUIST, 1757) como *Cullex aegypti* (GARCES-AYALA et al., 2014), é de origem africana e, com menos de um centímetro de comprimento, é menor do que mosquitos comuns, sendo caracterizado pela sua cor preta com listras brancas no tronco e nas pernas e asas translúcidas, além de ruído praticamente inaudível aos humanos ao voar.

Enquanto que o mosquito macho alimenta-se apenas de frutas, a fêmea, porém é hematófaga, de maneira que, após a cópula, precisa alimentar-se de sangue como fonte de proteína para realizar a maturação completa de seus ovos. Durante seu tempo aproximado de vida de trinta dias, pode dar origem a cerca 1.500 mosquitos. Após a fecundação e postura dos ovos, inicia-se o ciclo de vida do *Ae. Aegypti*, o qual, após a eclosão do ovo, atinge a fase adulta em cerca de dez dias (CRUZ, 2022b).

Cada mosquito fêmea põe entre 150 a 200 ovos, os quais inicialmente são de cor branca, tornando-se pretos e brilhantes ao passar do tempo, como mostra a Figura 1 (MUNDIM-POMBO

et al., 2021). Possuem menos de 1 mm de comprimento e cerca de 0,2 mm de largura, e são depositados pelo mosquito em criadouros naturais ou artificiais, de modo a ficarem a milímetros de distância de água parada e limpa. Após cerca de 15h, adquirem resistência à secura, de modo que podem suportar até cerca de 450 dias sem contato com água (CRUZ, 2022a). Após ocorrer contato com água, os ovos ecodem em pouco menos de 30 min, e entre sete e nove dias tornam-se mosquitos adultos, tendo passado pelas suas quatro fases: Ovo, larva, pupa e adulto (SANTO, 2025).

Figura 1 – Ovos de *Aedes aegypti*



Fonte: (MUNDIM-POMBO et al., 2021)

Uma vez infectado com os vírus, o mosquito adulto fêmea transmite o vírus por toda vida e, é possível que pelo menos uma parte de seus descendentes já nasçam portadores do vírus (SANTO, 2025). Em adição a isso, em cada ciclo de reprodutivo, a fêmea pode depositar cerca de 100 ovos; isso pode ocorrer a cada quatro dias, o que favorece a proliferação do vírus (Ministério da Saúde – Brasil, 2025b).

2.1.1 Doenças relacionadas

2.1.2 Dengue

É uma doença causada pelo vírus da dengue (DENV), que possui quatro sorotipos: DENV-1, DENV-2, DENV-3 e DENV-4. Os sorotipos são variações do mesmo vírus e, mesmo pertencendo à mesma espécie, distinguem-se entre si quanto à sua composição antigênica (Ministério da Saúde – Brasil, 2025c).

Estima-se que há cerca de 390 milhões de infecções anualmente ao redor do mundo. A doença ocorre em lugares como, na América do Sul, Central, nordeste da Ásia e Pacífico Sul e é transmitida pela picada do *Ae. aegypti* e tem histórico considerável de epidemias no Brasil (KAUFMAN, 2019).

É importante salientar que o mosquito é infectado pelo vírus após sugar o sangue de um humano infectado com a dengue (no curto período em que o humano tem o vírus circulando em sua corrente sanguínea) e, só então, depois de cerca de 10 a 12 dias, o mosquito se torna infectivo, portanto transmissor da doença (Instituto Oswaldo Cruz – Fiocruz, 2025).

Após cerca de 5 a 8 dias, os sintomas da doença manifestam-se, quais sejam febre, erupção cutânea, dor de cabeça severa, dor nas articulações e nos músculos, dentre outros sintomas (KAUFMAN, 2019). Inclusive, a palavra “dengue” é de origem espanhola e significa “manha”, “melindre” e faz referência ao estado em que o infectado se encontra, devido aos sintomas (Secretaria de Estado da Saúde do Espírito Santo, 2015).

Ainda, o infectado pode se recuperar da doença em cerca de uma semana, porém, a doença pode ser severa nos casos de Dengue Hemorrágica e causar até óbito (KAUFMAN, 2019).

2.1.3 Chikungunya

Causada pelo vírus chikungunya (CHIKV), é uma doença febril aguda, transmitida pelo *Ae. aegypti* e *Aedes albopictus*, conhecido como muriçoca. Foi primeiramente descoberta em 1952 no Planalto de Makonde (hoje, Tanzânia), no leste da África. A palavra “chikungunya” significa “aqueles que se dobram”, de modo à fazer referência às dores articulares causadas pela doença (Agência Fiocruz de Notícias, 2023; BARTHOLEMEEUSEN et al., 2023).

Quanto aos sintomas, a doença acomete as articulações, de modo que é potencialmente debilitante, dada a intensidade e cronicidade do quadro de dor (ESPORCATTE; PORTES, 2019).

Após o período de incubação do vírus (3 a 7 dias), a maioria dos pacientes sofre com dor em múltiplas articulações (poliartralgia), além de dor muscular (mialgia), que podem ser crônicos (THIBERVILLE et al., 2013; BARTHOLOMEEUSEN et al., 2023).

Ainda, a doença pode progredir em três fases: aguda, com duração de sete até 14 dias; subaguda, que permanece até três meses e, por fim, crônica, em que os sintomas duram mais de três meses (MARQUES et al., 2017).

No caso da chikungunya, a incidência de pacientes que precisam de atenção médica é maior, em relação a outras arboviroses comuns. Ainda após a fase aguda, alguns pacientes pode ocorrer reincidência da doença (recidiva), persistente dor em uma ou mais articulações (artralgia) ou dores musculoesqueléticas (THIBERVILLE et al., 2013). Inclusive, a artralgia pode perdurar por até três anos (SCHILTE et al., 2013).

Nos casos graves, a doença pode causar encefalite, miocardite, hepatite e falência múltipla de órgãos. Afora, o acometimento neurológico pode causar convulsões, alteração mental, paralisia fáctica, de maneira que pode culminar na morte do paciente (SINGH et al., 2008; RAJAPAKSE; RODRIGO; RAJAPAKSE, 2010). Ainda, também a doença tem um efeito expressivo na saúde e qualidade de vida de indivíduos que já possuem doenças crônicas, de maneira a resultar também, em prejuízos econômicos, em especial em países emergentes (THIBERVILLE et al., 2013).

2.1.4 Zika

O Zika vírus (ZIKV) é um flavivírus, que, a propósito pertence à família dos vírus da Febre Amarela, Dengue e Chikungunya. É transmitido vetorialmente através da picada do *Ae. aegypti* ou do *Aedes albopictus*, também por meio de relações sexuais sem proteção com pessoas portadoras do vírus, transfusão de sangue e transplante de órgãos, também de pessoas portadoras da doença. Ainda, a transmissão se dá vetorialmente, de mãe para filho, durante a gestação ou no momento do parto (ANTONIOU et al., 2021; VIVEIROS et al., 2025).

Assim, infecções pelo ZIKV têm sido reportados desde os anos 1950 em vários países do continente africano e desde 1966 na Ásia. Contudo, a partir de 2015, a doença se espalhou pelas Américas, com o pico de 500 mil infectados. Alcançou a Europa em 2019 e Índia em 2021 (World Health Organization, 2022; JONG; GROBUSCH, 2025). Em dezembro de 2023, já haviam 91 países ou territórios com casos já registrados de infecção por vírus Zika (RABE et al., 2025).

Os casos de infecção pelo ZIKV são, em sua maioria, assintomáticos ou com sintomas leves.

Entretanto, quando há, entre os sintomas estão: Erupções cutâneas, dor de cabeça, febre, dor nas articulações, conjuntivite. Podem ser observados ainda, vômitos, diarreia, edema, dor abdominal, dentre outros sintomas, a exemplo da pressão intraocular.

Para além disso, destaca-se que, apesar de atingir uma parte menor no conjunto de infectados, as queixas clínicas neurais ou graves podem também ocorrer, de maneira a causar danos ao desenvolvimento neurológico (FACCINI et al., 2022; VIVEIROS et al., 2025).

Entretanto, apesar do risco de doença vírus Zika grave em adultos ser baixo, de modo a causar poucas mortes, a transmissão vertical, de mãe para filho, tem um enorme impacto em questões de saúde, devido ao fato do nascimento de bebês com microcefalia. Nessa perspectiva, o termo Síndrome Congênita do Zika descreve os casos de recém-nascidos afetados pela combinação de microcefalia com outras anomalias de desenvolvimento (PEREIRA et al., 2020; GIRALDO; GONZALEZ-OROZCO; RAJSBAUM, 2023).

Inclusive, em 2015 e 2016, houve o maior surto de vírus Zika já observado, além de estar associado a anomalias graves no desenvolvimento do sistema nervoso central em fetos. Nesse período, houve um aumento do número de bebês com microcefalia potencialmente associado ao surto de ZIKV (LEBOV et al., 2023).

2.2 SISTEMAS DE VIGILÂNCIA

No contexto de Sistemas de Vigilância, a vigilância epidemiológica proporciona a observação sistemática e ativa de ocorrência de doenças e de infectados, o que é uma vigilância de pessoas. Assim, é um método tardio para detectar epidemias. Por outro lado, a Vigilância Entomológica envolve atividades referentes a fatores de riscos biológicos, baseados na coleta e tratamento de dados referentes ao vetor de doenças. É observada a periodicidade na coleta desses dados, de maneira a se ater a intervalos pré-definidos para se obter indicadores escolhidos (MARTINEZ; LEGALL, 1998).

2.2.0.1 *Vigilância Entomológica*

Haja vista a rápida ascenção de doenças transmitidas por mosquitos, como é o caso da dengue, Chikungunya e Zika, além do alto porcentual de infecções, a Vigilância Entomológica é aplicada por governos na forma de planos de ação para combater essas doenças, na detecção, controle e, quando possível, erradicação dos vetores (VILLA, 2020). Assim, a Vigilância Ento-

mológica do *Ae. aegypti* é uma ferramenta empregada para monitorar, de maneira tempestiva e permanente, a presença dos vetores de doenças, seus principais criadouros, bem como a distribuição geográfica deles no território. A propósito, tem sido aplicados no Brasil, sistemas de Vigilância Entomológica, complementados com uso de Sistemas de Informação Geográfica para auxílio na tomada de decisões e controle do vetor (EIRAS; RESENDE, 2009; PEPIN et al., 2013).

Adicionalmente, recomenda-se realizar pelo menos um levantamento de índice larvário – Levantamento de Índice Amostral (LIA) ou Levantamento Rápido de Índices para *Aedes aegypti* (LIRAA). Entretanto, é importante realizar a análise dos dados considerando-se também distintas metodologias de levantamento entomológico, haja vista os índices estimados pelo LIRAA/LIA representarem um retrato momentâneo, de maneira a apresentarem variações nos níveis de infestação e dos tipos de recipientes predominantes, devido a diferentes períodos do ano em que são realizados, principalmente por conta das variações climáticas. Por certo, ainda, podem ser empregados métodos complementares, tais como contagem de pupas e coleta de mosquito adultos.

Nesse ínterim, para aplicação das demais técnicas de controle do vetor, o emprego de ovitrampa constitui-se num critério obrigatório para a caracterização do território, com vistas à implementação e avaliação das outras tecnologias de controle, como o uso do método Wolbachia, uso de EDL e inseto estéril por irradiação. Assim, o Ministério da Saúde preconiza que a vigilância entomológica seja realizada por meio de ovitrampas, para todos os municípios brasileiros, estando infestados ou não, com monitoramento periódico, com uma distribuição ampla por todo o território urbano ou em locais estratégicos (Ministério da Saúde – Brasil, 2025d).

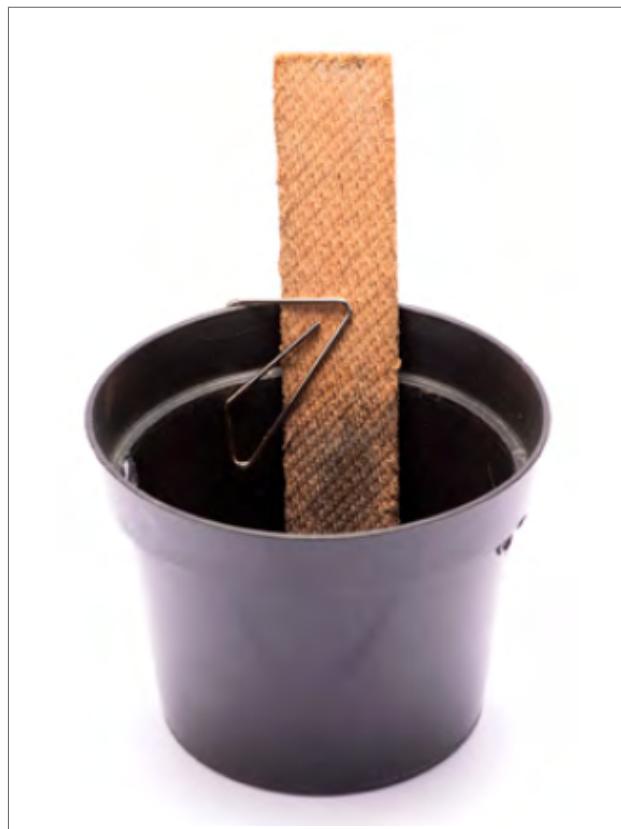
2.2.0.2 *Ovitrampa*

É uma armadilha utilizada como substrato de ovoposição, isto é, deposição (coleta) de ovos de *Ae. aegypti* ou até mesmo do *Ae. albopictus*. É constituída por um vaso de plástico preenchido com água, uma palheta de madeira, um clipe de metal, o qual prende a palheta ao vaso. A superfície áspera da ovitrampa é voltada para o interior do vaso.

O vaso de plástico na cor preta, funciona como um depósito de água e possui capacidade para 1L; apesar disso, a quantidade de água é limitada a 500 ml por um orifício realizado na lateral do vaso, o qual limita a quantidade de água. A palheta com dimensões próximas de 15 por 1,5 cm colocada e presa ao vaso pelo clipe, é feita de madeira aglomerada, do tipo

Eucatex® (Eucatex S.A. Industria e Comercio) e sobre sua superfície áspera, o mosquito fêmea deposita seus ovos, como se vê na Figura 2. Podem ser acrescentados, na água, atrativos ao mosquito fêmea, como levedo de cerveja (Ministério da Saúde – Brasil, 2025d).

Figura 2 – Ovitrampa



Fonte: (Ministério da Saúde – Brasil, 2025d)

Assim, o uso dessas armadilhas constitui-se num método sensível, de baixo custo e fácil manuseamento em campo para detectar a presença do vetor e guiar as ações subsequentes, como o levantamento de índices de ovos, seguido de ações posteriores (BRAGA et al., 2000).

Ademais, ao contrário dos índices estimados pelo LIRAA/LIA, que oferecem um retrato momentâneo, o monitoramento por ovitrampa fornece índices de ovos de maneira periódica. Estes índices podem indicar um nível maior de infestação por conta da alta sensibilidade que a ovitrampa apresenta.

2.2.0.3 Monitoramento

Desta feita, as ovitrampas são distribuídas obedecendo a um raio de distância de 300 ou 400m uma da outra, considerada a capacidade operacional do local. Isto feito, o monitoramento

em áreas prioritárias é feito semanal ou quinzenalmente, enquanto que, nas demais áreas monitoradas, é mensal, a depender também da capacidade operacional.

A partir da aplicação, coleta e transporte e contagem de ovos de ovitrampa, são levantados indicadores entomológicos, quais sejam: Índice de Densidade de Ovo (IDO), Índice de Positividade de Ovo (IPO) e Índice de Densidade Vetorial (IDV). O IDO, indica a quantidade média de ovos por ovitrampa positiva (que contém ovos). É obtido a partir da divisão do número geral de ovos pela quantidade de armadilhas positivas. Por outro prisma, o IPO obtém o porcentual de armadilhas positivas em relação ao total de armadilhas. Por fim, o IDV, indica a média de ovos por armadilhas em geral (positivas ou não). Obtém-se através da divisão do número geral de ovos pela quantidade total de armadilhas examinadas, contendo elas ovos ou não (Ministério da Saúde – Brasil, 2025d).

Este monitoramento, feito em continuidade por no mínimo três meses, é requisito para a implementação de outras tecnologias no combate ao vetor, a exemplo de uso de estações disseminadoras de larvicidas, a liberação de mosquitos com a bactéria Wolbachia, bem como também da liberação de mosquito estéril por irradiação.

*2.2.0.4 Contagem manual de ovos de *A. aegypti**

A fidelidade desses índices de ovos supracitados é determinante para a correta aplicação das posteriores ações de controle vetorial e mobilização da sociedade, bem como da tomada de decisões de controle e prevenção e definição de estratégias de combate por parte dos gestores. Uma vez que as ovitrampas são instaladas, distribuídas, recolhidas e, por fim, transportadas para o laboratório, é feita a contagem dos ovos (Ministério da Saúde – Brasil, 2025d).

A contagem é realizada manualmente por um técnico treinado que utiliza um microscópio estereoscópico (lupa). Desta forma, a quantidade de ovos é contada e as palhetas contendo ovos são consideradas positivas e as que não têm, negativas (Instituto Oswaldo Cruz – Fiocruz, 2024). Ovos inviáveis, isto é, secos ou já eclodidos são desconsiderados da contagem.

É recomendado que contagem de ovos seja feita sequencialmente por campos da palheta, fazendo-se uso de um lápis ou de uma lâmina de microscopia para delimitação das regiões, visando evitar a leitura de regiões já consideradas na contagem e, assim, obter uma leitura equivocada. Para tanto, pode-se adotar a marcação com lápis dos campos já lidos para evitar a contagem de campos repetidos.

Com isso, a implementação da Vigilância Entomológica na avaliação do controle do Ae.

aegypti pode ser limitada, devido à falta de indicadores confiáveis e práticos. Nesse sentido, a contagem de ovos do vetor é determinante para o emprego mais efetivo de ações subsequentes de controle do mosquito (SCOTT; MORRISON, 2009; SCOTT; MORRISON, 2010; BARRERA, 2016).

2.3 PROCESSAMENTO DE IMAGENS

Uma imagem é definida como uma função bidimensional, $f(x, y)$, em que x e y são coordenadas espaciais, e a intensidade (ou nível de cinza) é a amplitude de f em qualquer ponto $f(x, y)$ no plano (GONZALEZ; WOODS, 2015). Quando tanto x , y quanto os valores de intensidade são quantidades finitas e discretas, tem-se uma imagem digital. Ou seja, uma imagem digital é uma representação concreta de objetos do campo visual com informação espacial (*layout*) e intensidade (cores ou tons de cinza). Nesse cenário, cada ponto da imagem é chamado de *pixel* (PETERS, 2017).

Outrossim, uma imagem binária é formada somente por *pixels* de intensidade 0 (preto) ou 1 (branco). Já uma imagem em escala de cinza é representada por uma função de intensidade de luz, com coordenadas x , y e contém *pixels* visíveis como preto, branco ou cinza (cores intermediárias entre preto e branco). Já a imagem colorida (como com o sistema Red, Green, Blue (RGB)) possui, para cada ponto (x, y) , um array 1×3 , em que cada elemento desse array refere-se a um canal de cor (vermelho, verde e azul) e indica o nível de brilho de cada cor. Portanto, trata-se de uma imagem multidimensional com três canais (PETERS, 2017).

A área de Processamento Digital de Imagens refere-se ao estudo de informação de imagens digitais, manipulação, extração de características, descrição e visualização, ou seja, abrange (não só) processos cujas entradas e saídas são imagens, mas também extração de atributos de imagens até o reconhecimento de objetos individuais (GONZALEZ; WOODS, 2015).

Uma vez que este campo envolve processamento de imagens digitais por um computador digital, logo, esta área está associada diretamente ao computador, uma vez que imagens digitais necessitam, tanto de capacidade de armazenamento e desempenho computacional (PETERS, 2017). Quanto aos processamentos aplicados, pode resumir-se em três níveis: Baixo (como o pré-processamento para reduzir o ruído), médio (a exemplo da segmentação e descrição) e alto, que envolve “dar sentido” a objetos reconhecidos através da análise de imagens (PETERS, 2017).

Processamento de Imagens envolve um conjunto de metodologias, como: Aquisição de imagens, realce de imagens, restauração de imagens, processamento de imagens coloridas,

processamento morfológico, segmentação, reconhecimento, entre outras. No entanto, a definição e emprego de quais metodologias é orientada ao problema, de modo que, para uma dada aplicação ou sistema de processamento de imagens é possível que nem todos os passos são requeridos (GONZALEZ; WOODS, 2015).

2.3.1 Realce de Imagens

A partir do plano ou domínio da imagem, pode ser aplicado processamento espacial, cujas categorias principais são: Transformação de intensidade e filtragem espacial. Aquela lida de forma individual nos *pixels* de uma imagem para manipulação de contraste e limiarização, por exemplo. Esta, porém, trata de operações como o realce de imagens e distingue-se por envolver em seu processamento os *pixels* vizinhos (SOLOMON; BRECKON, 2011).

Neste sentido, Realce de Imagens é a aplicação de técnicas de manipulação de imagem para torná-la mais adequada ao domínio específico de um problema; assim, é, por natureza, orientada ao problema (PRATT, 2007). Em outras palavras, significa tornar a informação contida na imagem mais visível (NDE-Ed, 2000). Sendo desta forma, a priori, o observador é quem avalia o êxito da aplicação do processamento. Em outros casos, porém, no contexto de uma aplicação de computacional de reconhecimento de objetos, por exemplo, a definição do melhor realce está associada ao sucesso na taxa de detecção do sistema, por exemplo. Ou seja, o a melhor técnica é a que resulta em melhor reconhecimento de objetos por parte da aplicação (GONZALEZ; WOODS, 2015).

2.3.2 *Color Jitter*

É uma técnica aplicada para simular diferentes variações de ambiente e de iluminação numa imagem, através da alteração de propriedades de cor de uma imagem. A exemplo de alterações aleatórias no brilho, contraste, saturação ou matiz de uma imagem (PYTORCH, 2025a). No contexto de *Data Augmentation*, quando se pretende aumentar um conjunto de dados, pode ser aplicada (em conjunto com outras técnicas existentes) para aumentar a diversidade dos dados de um modo mais natural (ZENG et al., 2023). As transformações de intensidade propostas nesta técnica podem realçar os ovos de *A. aegypti* na superfície das palhetas de ovitrampa, promovendo uma melhor separação visual entre objeto de interesse e *background*.

2.3.3 Processamento de histograma

O histograma de uma imagem exibe a frequência relativa de ocorrência dos valores de intensidade dos *pixels* em relação aos valores em si. São definidos intervalos de intensidade, chamados *bins* e assim, as intensidades são agrupadas em cada *bin* (PETERS, 2017).

Então, a partir do histograma de uma imagem, podem ser aplicados diversos processamentos espaciais no âmbito da manipulação de histograma com intuito de realçar imagens; entre eles estão o alargamento de contraste, equalização de histograma, especificação de histograma e equalização adaptativa (OPENCV, 2025a).

2.3.4 Equalização de Histograma

No contexto das técnicas de processamento de histograma, a equalização de histograma é uma técnica que uniformiza a distribuição dos níveis de intensidades dos pixels presentes na imagem, de modo a obter uma imagem de saída cujos valores do histograma sejam mais bem distribuídos ao longo das intensidades possíveis da imagem.

Para tanto, baseia-se no histograma da imagem de entrada e lida com diversos cenários. Entre os casos, estão: Imagens escuras, quando os componentes do histograma estão concentrados no nível inferior; imagens muito claras, quando os componentes estão abundantes do lado direito ou, por fim, quando as intensidades estão estreitas num intervalo curto, o que produz imagens de baixo contraste (histograma estreito). Estes casos, são alguns dos exemplos de problemas que podem ser tratados com equalização de histograma. Objetiva-se atingir uma imagem de alto contraste em que seja exibida uma variedade uniforme de tons (GARG; JAIN, 2017). Em imagens de palhetas de ovitrampa, onde é comum naturalmente haver baixo contraste (especialmente regiões muito escuras), esta transformação torna-se promissora para distribuir de forma mais equânime os níveis de intensidade na imagem e realçar sua superfície e ovos contidos nela (MUNGRA et al., 2020; CHOWDHURY; LIU; RAMANNA, 2024; SAIFULLAH; DREŽEWSKI, 2024).

2.3.5 Contrast Limited Adaptive Histogram Equalization (CLAHE)

Uma das limitações da equalização de histograma é que, por aumentar o contraste da imagem, o contraste do ruído também é aumentado. Ante isso, O Contrast Limited Adaptive

Histogram Equalization (CLAHE) surge como uma extensão adaptativa da Equalização de Histograma, acrescida da aplicação de um limiar de contraste (MUSTAFA; KADER, 2018). Esta transformação divide a imagem em regiões chamadas *tiles* e aplica a Equalização de Histograma em cada uma delas. Em seguida, um *clip limit* (ou fator de contraste) pré-definido é aplicado a cada histograma obtido e o excesso é redistribuído igualmente abaixo desse limiar. Devido a isso, o contraste é limitado, o que impede a amplificação do ruído. À propósito, esta característica é promissora para o tratamento das imagens de palhetas, que naturalmente possuem ruído visual devido à textura do material e à presença de detritos, além dos próprios ovos. Por fim, para reunir as regiões outrora separadas, é aplicada Interpolação Bilinear (PIZER et al., 1990) (MATHWORKS, 2024). Possui requisitos computacionais modestos e requer apenas o fator de contraste como parâmetro de entrada (ZUIDERVELD, 1994).

2.4 VISÃO COMPUTACIONAL

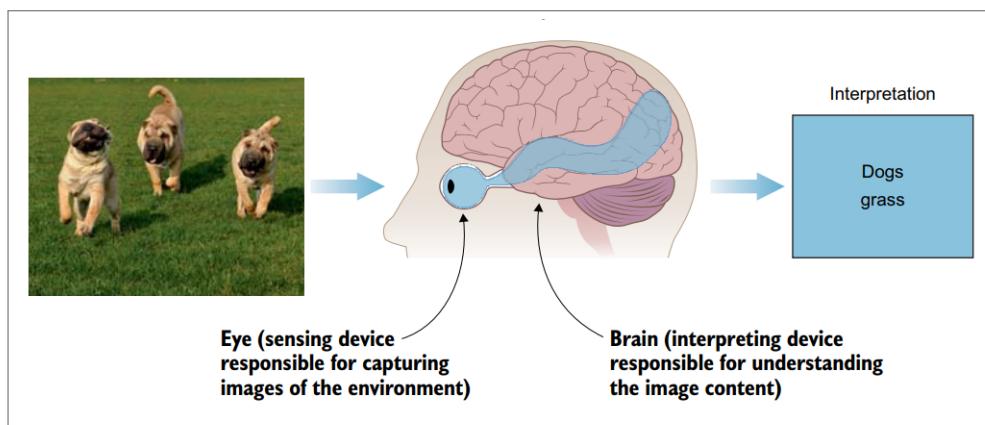
No âmbito da Inteligência Artificial (IA), espera-se que um sistema seja capaz de perceber o ambiente a sua volta e tomar decisões baseado em suas percepções. Dentro desse conjunto de percepções possíveis, está a percepção visual e é nela que a visão computacional está concentrada. Em outras palavras, é a ciência de perceber e compreender o ambiente através de imagens e vídeos, de maneira a construir um modelo físico desse ambiente e tomar ações apropriadas (ELGENDY, 2020). Assim, é um ramo da Inteligência Artificial e tem como objetivo utilizar computadores para emular o sistema visual humano, de forma a aprender, ter a capacidade de realizar inferências e agir com base nas informações visuais percebidas (GONZALEZ; WOODS, 2015). Distingue-se do processamento de imagens, por envolver o entendimento do que acontece numa imagem, que é uma tarefa muito além do processamento (ELGENDY, 2020). Em outras palavras, é a ciência de entender ou manipular imagens e vídeos (SHANMUGAMANI, 2018). Possui diversas aplicações, como realidade aumentada, direção autônoma, inspeção industrial, dentre outras (SHANMUGAMANI, 2018).

Baseada no sistema visual humano, objetiva reconstruir e interpretar cenas naturais a partir de imagens (Szeliski, 2022). Em alto nível, o sistema visual humano assemelha-se ao dos animais e consiste de um sensor (nesse caso, o olho) que captura a imagem; o cérebro processa essa informação e a interpreta e, por fim, o sistema retorna uma predição dos elementos da imagem, baseado nos dados extraídos dela (ELGENDY, 2020). A Figura 3 sintetiza o sistema visual humano aos agentes sensor e interpretador.

Para observadores humanos, ao visualizar uma imagem ou cena contendo cães, a interpretação e entendimento do cenário são instantâneos; os componentes da imagem são reconhecidos e podem ser contados de maneira bastante espontânea. O reconhecimento de que na cena exibida na Figura 3 há cachorros só é possível porque o observador foi treinado ao longo de anos a identificar cães. No entanto, se esta fosse mostrada pela primeira vez a um humano, seria necessário fornecer também a informação de que se tratam de cachorros. Este é o processo de treinamento. Ao ver porém um outro animal, como um cavalo, o humano poderia julgar, pela semelhança, tratar-se de um cão também. Ao receber a informação de que a identificação está errada e que trata-se de um cavalo, o cérebro ajusta seu aprendizado. Assim, de modo geral, um cérebro é treinado a identificar classes de objetos em geral, bastaria adicionar mais classes (ELGENDY, 2020).

O mesmo, em alto nível, acontece no treinamento de sistemas de visão computacional para aprenderem a identificar objetos. Enquanto para humanos, basta poucos exemplos para aprender o padrão, para computadores, são necessárias milhares e até milhões (a depender da complexidade) de imagens para aprender-se a identificar os objetos vistos no treinamento (ELGENDY, 2020).

Figura 3 – Sistema Visual Humano

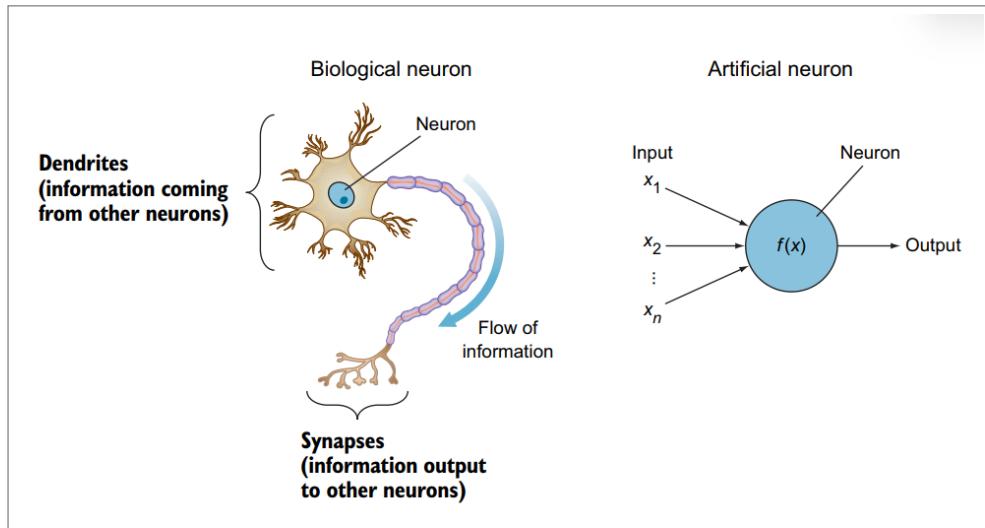


Fonte: (ELGENDY, 2020)

Um aspecto importante em sistemas de visão computacional consiste na escolha do melhor sensor adequado ao problema para captura de dados do ambiente. Câmera digital, radar, raios-x, entre outros tipos, combinados ou não, simulam o que é o olho humano e dos animais. Para atuar como dispositivos de interpretação e fazer o que o cérebro faz, são usados algoritmos de visão computacional, os quais, cumpririam o papel de cérebros artificiais e serem capazes de, receber os sinais obtidos pelo sensor, extrair características, detectar padrões e identificar

objetos (ELGENDY, 2020). No caso do cérebro biológico, a informação vinda de outro neurônio é recebida por um neurônio através dos dendritos, passa por um fluxo interno ao neurônio e, por fim, é externada para outro neurônio, por meio das sinapses, como mostra a Figura 4. De forma análoga, os neurônios artificiais possuem uma entrada, um processamento interno e uma saída de sinais (ELGENDY, 2020).

Figura 4 – Semelhanças entre neurônios biológicos e sistemas artificiais



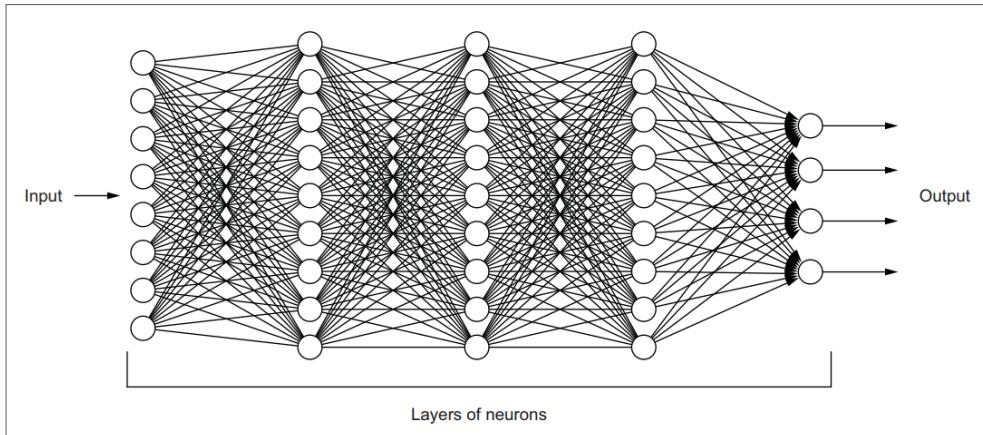
Fonte: (ELGENDY, 2020)

Semelhante ao processamento pelo qual a informação passa no cérebro biológico, cada neurônio artificial, conectado a outros, ao receber uma quantidade suficiente de sinais de entrada, dispara um sinal para cada neurônio com qual está conectado. Este fluxo individual é tornado mais complexo, ao envolver, por exemplo, milhões de neurônios conectados entre si. Ao ter diversas camadas de neurônios, produz-se um comportamento de aprendizagem. Ao criar-se uma rede de neurônios artificiais conectados e com várias camadas, como exemplifica a Figura 5, obtém-se uma Rede Neural Artificial (RNA); desta forma, está a se usar Aprendizagem Profunda ou Deep Learning (DL). Na figura, observa-se os neurônios da camada de entrada (*input*), os neurônios das camadas intermediárias ocultas (*hidden layers*) e os neurônios da camada de saída (*output*).

2.4.1 Aprendizagem Profunda

Aprendizagem profunda é um tipo específico da Aprendizagem de Máquina (AM), que, subsequentemente, é um campo da IA, de modo que técnicas de AM influenciaram no de-

Figura 5 – Rede Neural Artificial



Fonte: (ELGENDY, 2020)

senvolvimento de algoritmos de aprendizagem profunda (GOODFELLOW; BENGIO; COURVILLE, 2016).

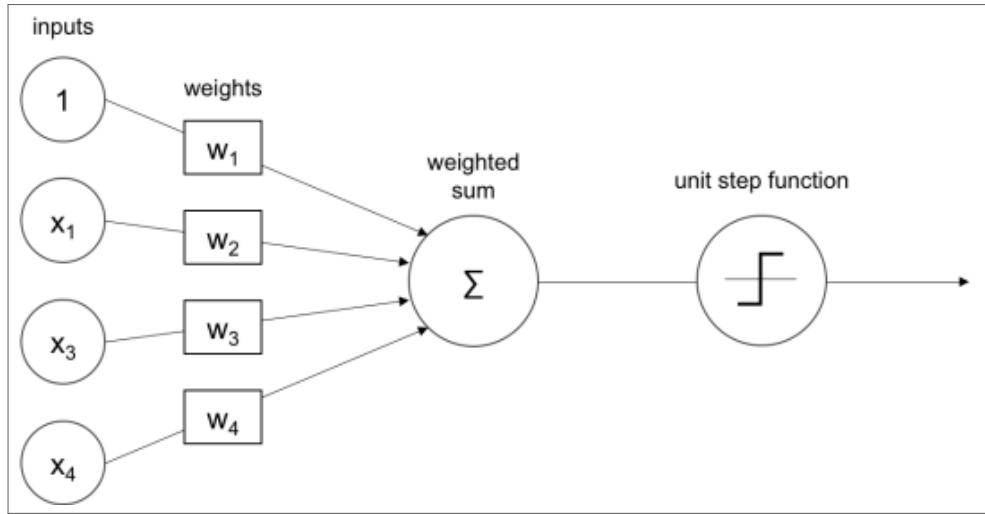
Neste campo, há o conceito elementar de que um neurônio artificial (ou *perceptron*) possui entradas, realiza uma soma ponderada e retorna um resultado. Este modelo de arquitetura que foi proposto inicialmente por (ROSENBLATT, 1958) é modelo mais simples de uma RNA (GÉRON, 2019). No contexto da aprendizagem, a atribuição dos pesos do *perceptron*, que vão influenciar na saída, é determinada durante o treinamento, tendo como base os dados de treinamento utilizados (SHANMUGAMANI, 2018).

Por princípio, como mostra a Figura 6, as entradas passam por soma ponderada pelos pesos *weights*. O *perceptron* aprende funções simples ao atualizar seus pesos de acordo com os dados e este processo de aprendizagem é chamado de treinamento. Por fim, a saída, passa para uma função de ativação (ou *step function*), que serve para introduzir não linearidade às redes neurais e possibilita o aprendizado em cenários complexos. Ainda, a função de ativação decide se o sinal do *perceptron* deve ser emitido ou passar para as seguintes camadas (*layers*) (SHANMUGAMANI, 2018).

Adicionalmente, o uso de *perceptrons* possui capacidade limitada, como apontado por (MARVIN; SEYMOUR, 1969), a exemplo da resolução trivial de problemas de “Ou Exclusivo” (XOR). Entretanto, ao adotar o uso de múltiplos *perceptrons* associados, estas limitações são resolvidas. Assim, Multi-Layer Perceptron (MLP) consiste numa camada de entrada, camadas ocultas e uma saída. Cada camada é completamente conectada com a próxima e, a excessão da camada final, possui um viés (*bias*), como exibe a Figura 7.

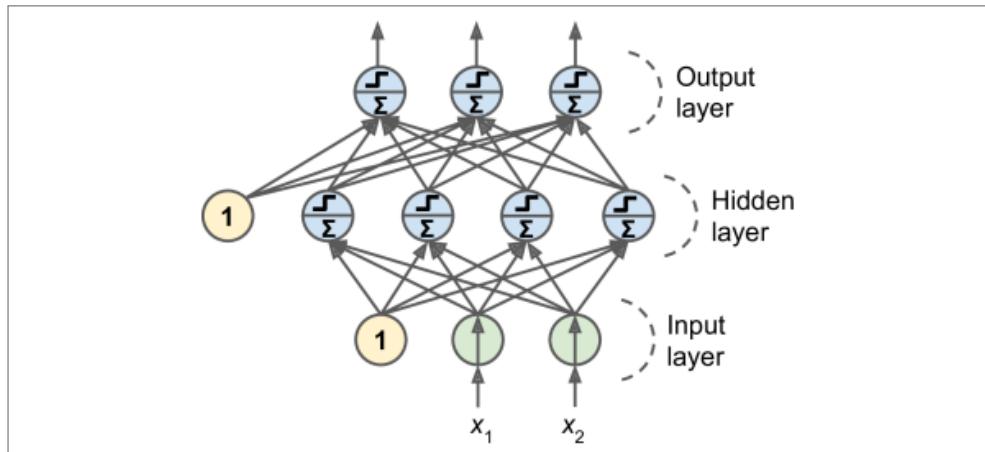
À luz do exposto, dadas estas bases, uma RNA é uma coleção de *perceptrons*, conectados

Figura 6 – Diagrama de Perceptron



Fonte: (SHANMUGAMANI, 2018)

Figura 7 – Multi-Layer Perceptron



Fonte: (GÉRON, 2019)

entre si de maneira a formar camadas ocultas que, em decorrência disso, formam a base não linear que mapeiam as camadas de entrada para as camadas de saída. A RNA faz, então, um mapeamento entrada-saída, de forma que considera a soma ponderada das entradas e as entradas dos vieses (*bias*). Por fim, chama-se arquitetura a estrutura geral da RNA e modelo o conjunto de valores dos pesos e *bias* (ELGENDY, 2020).

Isto dito, quando uma RNA é formada por uma pilha de camadas ocultas, passa a ser chamada de Rede Neural Profunda ou Deep Neural Network (DNN), nome que faz referência à profundidade da arquitetura do modelo, dado número de camadas ocultas (GÉRON, 2019).

O processo de treinamento do modelo é o que determina os valores dos pesos e dos vieses. No início do treinamento, esses valores são inicializados de forma aleatória e ajustados ao longo do treinamento com base numa função de erro. Já o erro é calculado ao comparar-se a

saída do modelo com a saída esperada, chamada de verdade fundamental (*ground-truth*), os quais, na aprendizagem supervisionada, são fornecidos pelo supervisor que determina a saída correta esperada.

Desta forma, o erro serve de base para ajuste dos valores do peso, de modo a refinar o aprendizado do modelo. O treinamento é interrompido quando o erro não mais é reduzido ou por vontade do humano que opera o treinamento do modelo, por meio de gatilhos no algoritmo de treinamento. Desta forma, o modelo aprende as características dos dados e aprende a identificar padrões.

No caso de imagens, digitais, o conteúdo básico é formado por *pixels* com *features* elementares, como, bordas, ângulos, cor, forma e textura ou mais complexas como olhos, boca, faces etc. (Szeliski, 2022). Assim, as *features* representam as características das imagens e são utilizadas para aprendizado de características durante o treinamento (Shanmugamani, 2018).

O emprego de DL para visão computacional pode ser dividido em várias categorias de tarefa: Classificação, detecção, segmentação e geração, tanto de imagens, como de vídeos (Szeliski, 2022).

2.4.2 Redes Neurais Convolucionais

Baseadas no córtex visual humano, as Redes Neurais Convolucionais ou Convolutional Neural Network (CNN) assemelham-se à RNA, no sentido de possuírem pesos, viés, função de ativação e saída. Devido ao fato de imagens serem grandes, usar RNA em imagens produziria uma arquitetura com um número muito expressivo de neurônios, além de não produzir o aprendizado eficiente das *features* presentes na imagem (Shanmugamani, 2018).

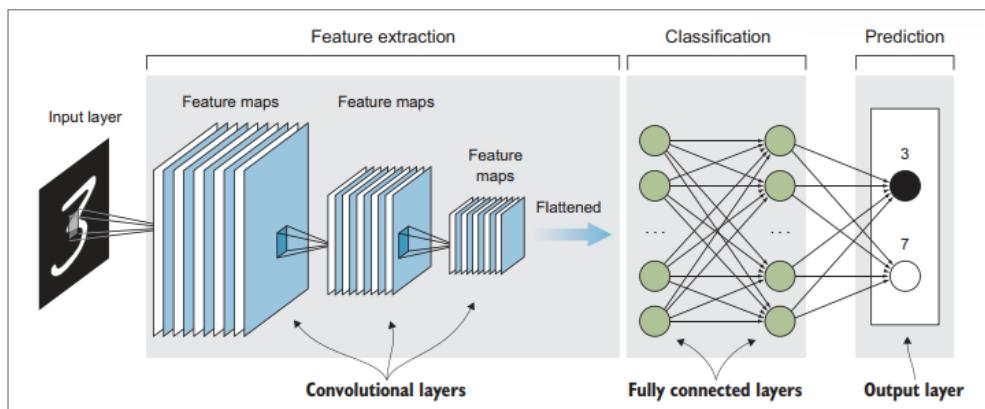
À vista disso, a estrutura de uma imagem digital, como descrito na seção 2.3, uma imagem é considerada um volume e possui dimensões de altura, largura e profundidade; esta, inclusive, associada à quantidade de canais da imagem, por exemplo: No sistema de cores RGB, a profundidade da imagem seria 3 (Shanmugamani, 2018). Nesse sentido, os neurônios da CNN são organizados também de forma volumétrica para tomar vantagens a partir da imagem. Cada camada transforma o volume de entrada num volume de saída.

A priori, o bloco básico de uma CNN é uma camada convolucional; cada neurônio da primeira camada é conectado a apenas aos *pixels* em seu campo receptivo, ou seja, uma região específica da imagem. Assim, na segunda camada convolucional, cada neurônio conecta-se da

mesma forma a apenas uma região. Essa arquitetura faz com que, nas camadas iniciais, a rede concentre-se nas *features* de baixo nível (genéricas, como bordas) e nas camadas finais, nas de alto nível (como faces). Essa estrutura é comum em imagens do mundo real e torna apropriado o uso de CNN (GéRON, 2019).

Ainda mais, as *features* são extraídas através dos filtros (ou *kernels*). Os valores desses *kernels* são aprendidos no treinamento. Os filtros deslizam sobre a imagem e realizam uma operação de convolução, o que resulta num conjunto de mapas de ativação (*feature maps*); o objetivo é detectar padrões e extrair *features*. Após as camadas de convolução, as *features* passadas para uma camada de neurônios totalmente conectados e, por fim, para a camada de saída, onde ocorre a predição. A Figura 8 exibe, em alto nível a síntese da arquitetura de uma CNN, com amada de entrada, camadas convolucionais, uma camada totalmente conectada (*Fully connected layer*) e a camada de saída.

Figura 8 – Arquitetura em alto nível de uma CNN



Fonte: (ELGENDY, 2020)

Assim, Redes Neurais Convolucionais têm obtido desempenhos consideráveis em cenários visuais complexos. Tarefas como condução autônoma, condução autônoma de veículos, além de tarefas como reconhecimento de voz ou Processamento de Linguagem Natural (PLN), são algumas das tarefas possíveis por conta do uso dessas redes (GéRON, 2019).

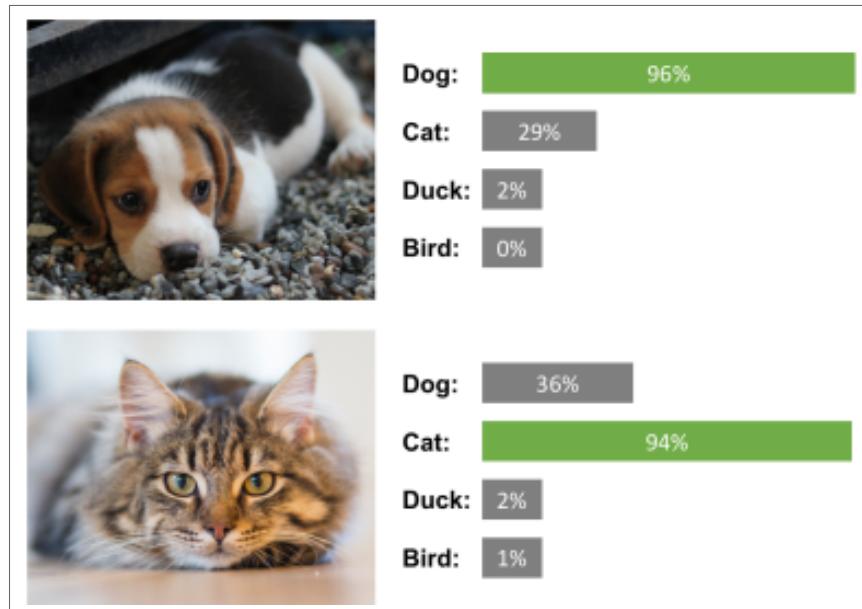
2.4.3 Modelos de Detecção de Objetos

Classificação de imagens, detecção de objetos e segmentação semântica são alguns exemplos de tarefas que envolvem o uso de DL para visão computacional.

Classificação de imagens significa rotular ou classificar uma imagem com base num objeto ou conceito presente nela (SHANMUGAMANI, 2018). A classificação atribui um rótulo a uma

imagem com base num conjunto pré-definido de categorias, por exemplo, rotular uma imagem em classes, como gato ou cachorro, a exemplo da Figura 9 (ELGENDY, 2020).

Figura 9 – Exemplo de Classificação de Imagens



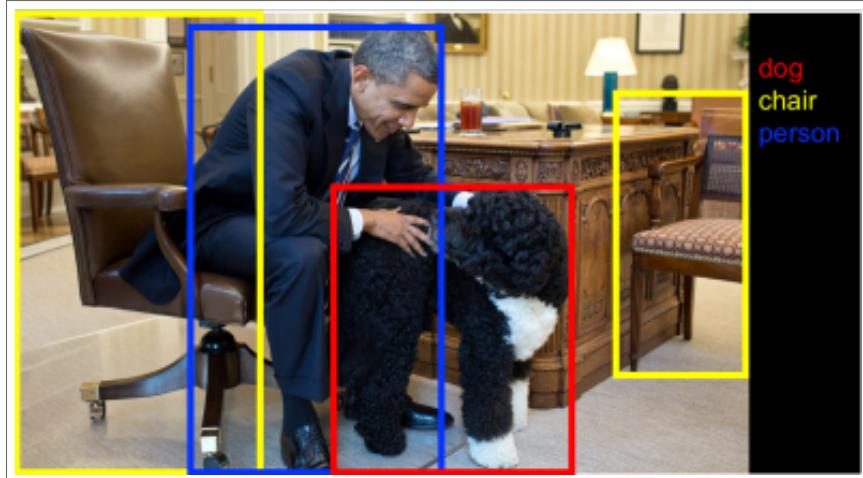
Fonte: (SHANMUGAMANI, 2018)

Por outro lado, detecção de objetos envolve não somente classificar a imagem, mas também localizar e detectar cada tipo de objeto nela presente, além de apresentar pontuações de confiança (*confidence scores* para cada objeto. Ou seja, enquanto a classificação rotula a imagem como um todo, a detecção de objetos localiza a posição de cada objeto e rotula cada um (SHANMUGAMANI, 2018). Assim, uma imagem pode contar múltiplos objetos de diferentes classes (ELGENDY, 2020). Outras tarefas, como segmentação semântica, que classifica à nível de *pixels* e segmentação por instância, que, além disso, individualiza objetos distintos (GÉRON, 2019).

Portanto, detecção de objetos objetiva classificar múltiplos objetos numa imagem e atribuir caixas delimitadoras (*bounding boxes*) para cada objeto (GÉRON, 2019), como exibe a a Figura 10, em que os objetos das classes “*dog*”, “*chair*” e “*person*” são localizados e rotulados.

Nesse tipo de tarefa, a arquitetura dos modelos de detecção de objetos baseados em CNN, de modo geral, é composta por: Backbone, Neck e Head. O *backbone*, usado para extração de características da imagem, de modo a obter a imagem de entrada e obtém mapas de ativação da imagem. Geralmente, usam-se modelos de classificação com a última camada *fully connected* removida. (JIAO et al., 2019). Já o *head*, propõe e refina as classificações e *scores* dos objetos, bem como, numa tarefa de regressão, as caixas delimitadoras (*bounding*

Figura 10 – Exemplo de Detecção de Objetos



Fonte: (SHANMUGAMANI, 2018)

box).

Em suma, o *backbone* recebe uma imagem como entrada e extrai as *features*, em seguida o *neck*, agrupa e combina as *features* de múltiplas escalas e, ao fim, o *head*, a partir delas, retorna a detecção (caixas delimitadoras, rótulos e pontuação de confiança). Como usam-se modelos de classificação como *backbone*, o *neck* e *head* são determinantes definir um *framework* de detecção (MATHWORKS, 2025a).

2.4.4 YOLOv10

O YOLOv10, pertencente à família de modelos You Only Look Once (YOLO), é uma arquitetura para detecção de objetos e segmentação por instância. Representa uma melhoria em várias abordagens em relação aos modelos de detecção de objetos em tempo real de ponta a ponta (*end-to-end*), tanto no desempenho verificado em *dataset* de *benchmark* quanto no tempo de inferência, dentre outras métricas. Os experimentos de então demonstraram que o YOLOv10 significativamente superou modelos do estado da arte ao considerar-se o *tradeoff* entre acurácia e custo computacional nas várias escalas de arquitetura disponibilizadas (WANG et al., 2024).

O YOLOv10 introduz melhorias em vários pontos-chave de melhoria nos componentes de sua arquitetura, de modo a aproveitar pontos fortes dos predecessores, mas também introduzir mudanças. No *backbone*, que opera a extração de *features*, usa-se uma versão aprimorada do Cross Stage Partial Network (CSPNet) para melhorar o fluxo do gradiente e reduzir a

redundância computacional. Também, no *neck*, que agrupa características extraídas e as passa para o *head*, são incluídas camadas Path Aggregation Network (PAN) para obter fusão eficaz de *features* em várias escalas. Ainda, o *One-toMany Head*, gera diversas predições por objeto durante o treinamento para fornecer ricos sinais de supervisão e melhorar a precisão. Por fim, o *One-to-One Head* gera uma única e melhor predição por objeto durante a inferência para remover a necessidade de uso do NMS, assim então, reduz a latência e melhora a eficiência (WANG et al., 2024; ULTRALYTICS, 2024).

Quanto aos avanços propostos, está a estratégia de *NMS-free training*, ou treinamento sem Non-Maximum Suppression (NMS), que utiliza atribuições duplas consistentes (*consistent dual assignments*) para poder tornar o NMS dispensável durante o treinamento e reduzir a latência de inferência (WANG et al., 2024).

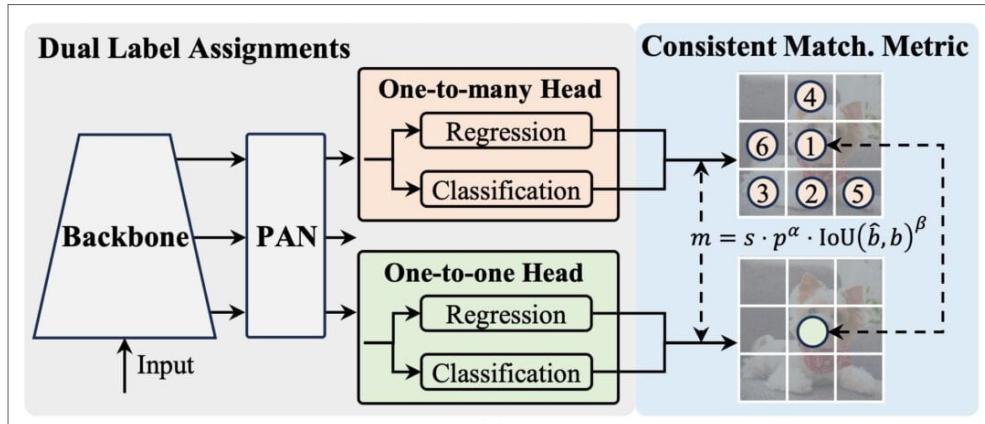
Além disso, destaca-se a introdução de uma arquitetura holística voltada à eficiência e precisão, que abrange a inclusão de um *head* de classificação mais leve, nomeado *Lightweight classification head*. Assim como a *head* de classificação assume maior importância em relação à *head* de regressão, reduziu-se a sobrecarga da *head* de classificação. Dessa forma, adotou-se uma arquitetura mais leve para este componente, composto por duas convoluções à nível de dimensão (*depthwise*) e com tamanho de *filtro* 3×3 , seguidas por uma convolução 1×1 (WANG et al., 2024).

Ainda, foram feitos ajustes diversos no modelo, como o uso de maiores *filtros* convolucionais para ter um *receptive field* maior e aumentar a capacidade de extração de *features*. Ainda, foram incluídos módulos parciais de *self-attention* (Partial Self-Attention (PSA)) para melhorar a performance sem introduzir custo computacional colateral (WANG et al., 2024).

Além disso, outro refinamento proposto foi a *Spatial-channel decoupled downsampling*, que desacopla a redução espacial da modulação de canal para minimizar a perda de informação e custo computacional. Além disso, *Rank-guided block design*, adapta o *design* dos blocos com base na redundância instrínseca dos estágios, de modo a garantir um uso ideal dos parâmetros. Por fim, a *consistent matching metric* alinha a supervisão entre as duas estratégias (*One-to-One Head* e *One-to-Many Head*) para melhorar a qualidade das predições. A arquitetura geral do YOLOv10 é exibida na Figura 11.

Quanto a desempenho superior a modelos de então, YOLOv10-S/X são, respectivamente, 1.8 e 1.3 vezes mais rápidos do o RT-DETR-R18/R101, mantendo desempenho similar nas demais métricas. Ainda, com desempenho semelhante, alcança uma redução de cerca de 46% na latência em relação ao YOLOv9-C e YOLOv10-B. Acrescenta-se que, utiliza os parâmetros

Figura 11 – Arquitetura do YOLOv10



Fonte: (WANG et al., 2024)

de forma mais eficiente, de maneira que, no YOLOv10 na escala L e X, excede o desempenho do YOLOv8 nas escalas L e X, com 0.3 e 0.5 AP e é 1.8 e 2.3 vezes menor em quantidade de parâmetros, respectivamente. Inclusive, YOLOv10 na escala M alcança AP similar ao YOLOv9-M e YOLO-MS, mesmo possuindo 23% e 31% menos parâmetros de treinamento. Desta forma, YOLOv10 supera o desempenho dos modelos de detecção de objetos predecessores com uma cobertura ampla de experimentos (WANG et al., 2024).

Assi, seu melhor equilíbrio entre desempenho e precisão tornam este modelo aplicável ao cenário de contagem de ovos, tanto para se obter uma metodologia de contagem de custo computacional viável, quanto atingir níveis de precisão satisfatórios ao cenário de contagem em imagens de baixa resolução.

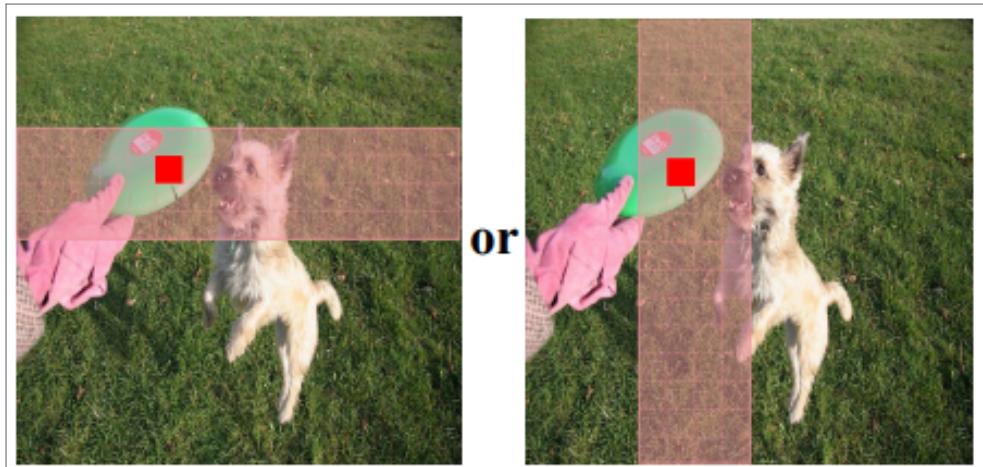
2.4.5 YOLOv12

Apesar de o uso de mecanismos de atenção (*attention*) afetarem de forma negativa a velocidade dos modelos, de modo a serem inferiores nesse ponto a modelos baseados em CNN, YOLOv12, propõe mudanças na arquitetura atual desta família de modelos, de maneira a utilizar uma estrutura centrada em *attention* para usar suas vantagens e, simultaneamente, igualar-se à velocidade dos modelos anteriores somente baseados em CNN (TIAN; YE; DOERMANN, 2025).

Caracteriza-se pela inserção do mecanismo de atenção, como *Area Attention Mechanism*, que envolve autoatenção para processamento de grandes *receptive fields* de maneira eficiente. Divide os mapas de ativação em l áreas de tamanhos iguais (por padrão, 4), tanto horizontal

como verticalmente, como exibe a Figura 12. Esse mecanismo evita operações complexas enquanto mantém um *receptive field* grande, o que resulta numa alta eficiência, pois reduz o custo computacional comparado ao uso de *self-attention* padrão (TIAN; YE; DOERMANN, 2025).

Figura 12 – Exemplo de aplicação de *Area Attention* do YOLOv12



Fonte: (TIAN; YE; DOERMANN, 2025)

Outro acréscimo importante, é Residual Efficient Layer Aggregation Networks (R-ELAN). As Efficient Layer Aggregation Networks (ELAN) são projetadas para melhorar a agregação de *features*, porém esta arquitetura pode causar instabilidade. Nesse sentido, no YOLOv12, foi proposta a R-ELAN para introduzir conexões residuais da entrada para a saída através do bloco, com um fator de escala (por padrão 0.01). Outro ponto é que a arquitetura aplica uma camada de transição para ajustar as dimensões do canal e produzir somente um *feature map*. Este, é processado por meio de blocos em sequência, sucedidos pela concatenação, de modo que forma uma estrutura de gargalo (TIAN; YE; DOERMANN, 2025).

Ainda, utiliza-se uma estrutura otimizada de atenção adaptada à estrutura do YOLO, de modo a manter a eficiência. Inclui *Flash attention*, para minizar a sobrecarga no acesso à memória. Ademais, dentro de um módulo de atenção, foi feito um ajuste na proporção entre as camadas de entrada e a camada de saída (*MLP ratio*); por padrão essa razão é 4, porém o YOLOv12 obteve melhor performance com um *MLP ratio* de 1.2. Isso equilibra o custo computacional entre a rede e o mecanismo de atenção e reduz a profundidade das camadas ocultas e facilita a otimização, além de aproveitar ao máximo as operações de convolução com eficiência no uso dos recursos computacionais (TIAN; YE; DOERMANN, 2025). Na detecção de ovos de *A. aegypti* em imagens palhetas de ovitrampa caracterizadas pela presença de ruídos

visuais, aplicação de mecanismos de *attention* nesta arquitetura YOLO de modo computacionalmente eficiente pode prover um modelo capaz de focar em *features* importantes nas imagens em detrimento de ruídos ou características secundárias.

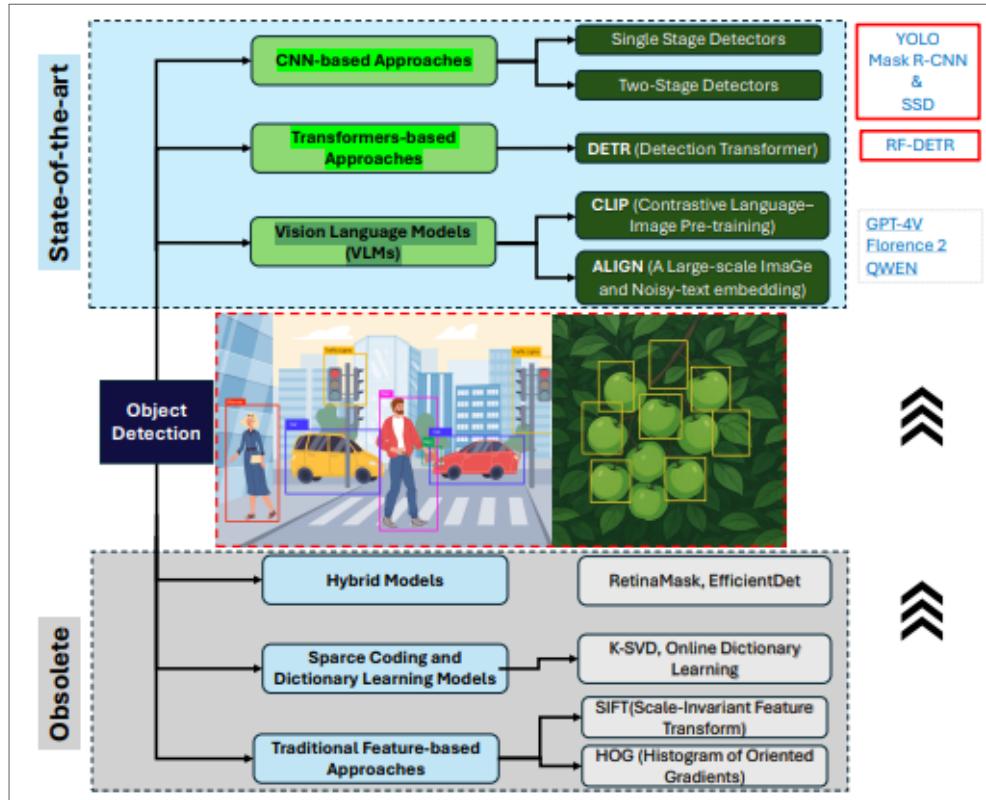
Este modelo supera todos os modelos de detecção de objetos em tempo real em desempenho de precisão, enquanto mantém velocidade similar. A exemplo, numa GPU NVIDIA T4, o YOLOv12-N alcança 40.6% de mean Average Precision (mAP) (Ou precisão média) com uma latência de 1.64 ms e supera YOLOv12-N e YOLOV11-N por 2.1% mAP com velocidade próxima. Não só supera modelos atuais de ponta a ponta, como Deformable Transformers (DETR), mas também suas melhorias, quais sejam: RT-DETR, RTDETRv2; YOLOv12-S mostra-se superior ao RF-DETR-R18 e RT-DETRv2-R18 e mostra-se 42% mais rápido, mesmo com o uso de 36% do custo computacional e 45% dos parâmetros (TIAN; YE; DOERMANN, 2025).

2.4.6 RF-DETR

Similarmente ao YOLO, DETR é uma série de modelos e o RF-DETR, parte desse conjunto, apresenta desempenho semelhante ao YOLOv12. Os detectores baseados em transformers, como o DETR, tiveram início com o Real-Time DEtection TRansformer (RT-DETR) por (ZHAO et al., 2024). Dentre as categorias mais atuais de modelos de detecção de objetos, como modelos de visão de linguagem ou Vision Language Models (VLMs), modelos baseados em CNN (como o YOLO), o grupo de modelos DETR enquadram-se numa abordagem baseada em transformers, como exibe a Figura 13. Abordagens obsoletas, que incluem modelos como EfficientDet (TAN; PANG; LE, 2020), K-SVD (AHARON; ELAD; BRUCKSTEIN, 2006) apresentam desempenho inferior em comparação com as metodologias atuais (SAPKOTA et al., 2025). Dentre outras melhorias, o conjunto de modelos DETR propôs uma alternativa ao NMS, já que este afeta o desempenho (ZHAO et al., 2024).

O RF-DETR, segundo o estudo de (SAPKOTA et al., 2025), utiliza um *backbone* DINOV2, proposto por (OQUAB et al., 2023). Nesse estudo feito também num conjunto de dados de frutas verdes, a arquitetura do RF-DETR utiliza mecanismos de atenção e destacou-se na modelagem de contexto global, de maneira a obterem-se, em casos de objetos ambíguos ou com oclusão, detecções particularmente eficientes. Como exibide a Figura 14, RF-DETR é constituído sobre uma arquitetura codificador-decodificador (chamada *encoder-decoder*) baseada em *transformers*. São utilizados também módulos *multi-scale* de atenção para substituir os mapas

Figura 13 – Classificação das metodologias de modelos de detecção de objetos. A figura exibe de baixo para cima (em ordem cronológica), as metodologias utilizadas. Destaca-se as abordagens do estado da arte, como *CNN-based* e *Transformer-based methods*, os quais são amplamente utilizados atualmente, além dos VLMs, que estão em uso crescente.

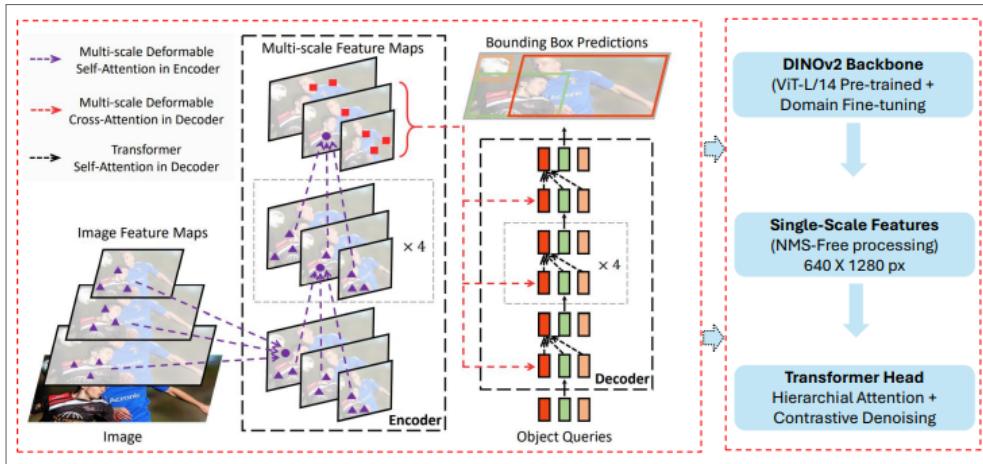


Fonte: (SAPKOTA et al., 2025)

de ativação dos módulos de atenção (ZHU et al., 2020). Devido à rápida convergência do modelo e eficiência computacional, inclusive dos recursos de memória, foi adotado um mecanismo para refinamento das *bounding boxes*, além de a proposição de regiões ser também feita a partir de uma variante do *Deformable DETR*, para servir como entrada para o decodificador para, por fim, ocorrer o refinamento da *bounding box* (ZHU et al., 2020).

No caso de problemas de detecção de classe única, RF-DETR obteve uma melhor mAP@50 de 0.9464, o que mostra a capacidade de detectar frutas verdes mesmo em cenários desordenados. A maior mAP@50:95 obtida foi pelo YOLOv12-N foi 0.7620, entretanto, RF-DETR obtém melhor performance em cenários espaciais mais complexos. Na detecção multi-classe, obteve um desempenho superior com mAP@50 de 0.8298, o que superou o modelo predecessor, de forma a evidenciar a capacidade do modelo de bem diferenciar objetos com ou sem oclusão. Outro ponto importante do RF-DETR é sua rápida convergência durante o treinamento, de maneira que o treinamento atingiu um platô e alcançou desempenhos importantes após ser treinado por apenas 10 épocas, o que é um evidente caso de destaque

Figura 14 – Arquitectura do RF-DETR



Fonte: (Sapkota et al., 2025)

(Sapkota et al., 2025).

Por fim, em aplicações do mundo real, há a indefinição de (entre o RF-DETR ou YOLOv12) qual modelo obtém desempenho superior. É evidenciado, que apesar de a arquitetura do RF-DETR permitir alta velocidade, este é mais recomendado para tarefas onde a precisão é mais crítica, já que obteve superior desempenho neste ponto em geral, enquanto que o YOLOv12 permanece mais adequado para detecção em aplicações de tempo real. Destarte, a capacidade do RF-DETR de modelar o contexto global e seu bom desempenho em cenários com oclusão são particularmente relevantes para este trabalho, uma vez que os ovos nas palhetas frequentemente se encontram aglomerados e sobrepostos

2.4.7 D-FINE

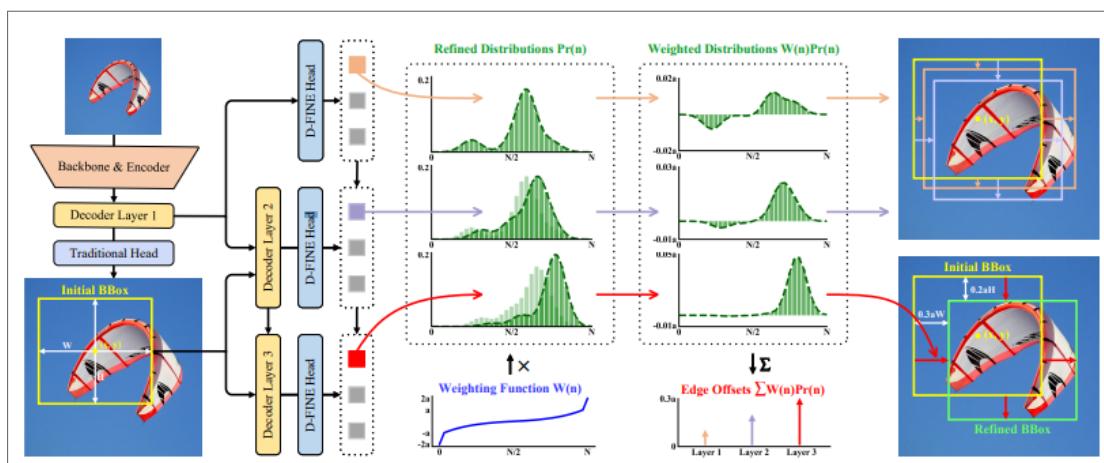
É um modelo de detecção de objetos em tempo real, que representa um aprimoramento a partir da série de modelos DETR, principalmente quanto à precisão de localização, justo por conta de redefinir a tarefa de regressão referente às caixas delimitadoras. Seu ponto chave busca resolver um problema na delimitação das *bounding boxes*, qual seja: Modelos tradicionais de detecção de objetos tratam a localização das caixas delimitadoras como pontos fixos e as tratam apenas com funções de perda (*loss functions*, a exemplo de L1 e Intersection over Union (IoU)). Esta forma, porém, falha em modelar a incerteza da localização, o que limita a precisão e atrasa a convergência. Portanto, em aplicações *real-time*, com as quais o D-FINE visa lidar, a alta precisão e a baixa latência são requisitos essenciais. Assim, por esta razão, D-FINE lida com a tarefa de regressão referente às caixas delimitadoras de modo a utilizar

uma arbordagem probabilística e autodestilação (PENG et al., 2024).

Quanto às suas melhorias, o D-FINE, propõe importantes abordagens arquiteturais relacionadas aos componentes: O Fine-Grained Distribution Refinement (FDR) e o Global Optimal Localization Self-Distillation (GO-LSD). O FDR, transforma o processo de regressão para não mais ser de coordenadas fixas e sim para envolver modelagens de distribuição de probabilidade, o que proporciona um maior ajuste fino nas representações intermediárias. Este componente, portanto, refina essas distribuições de forma iterativa e residual, de maneira a possibilitar de modo gradativo um ajuste na precisão de localização, tarefa importante em modelos de detecção de objetos. (PENG et al., 2024).

Ainda como mostra a Figura 15, o FDR, em repetição, otimiza a distribuição de granulação fina gerada pelas camadas. Por início, a primera camada decodificadora prevê caixas delimitadoras preliminares e distribuições de probabilidade também preliminares. Por sequência, cada uma das quatro bordas de uma caixa é associada a uma distribuição, para ser como uma caixa delimitadora de referência, para então, as camadas posteriores refiná-las por meio do ajuste das distribuições de forma residual. Por fim, as distribuições refinadas são aplicadas às quatro bordas da caixa delimitadora preliminar; esse processo ocorre de forma iterativa e por fim, espera-se obter melhor precisão nas caixas delimitadoras (PENG et al., 2024).

Figura 15 – Visão geral do componente FDR do modelo D-FINE: Observa-se que as distribuições de probabilidade operam como uma representação intermediária de granulação mais fina e que posteriormente são refinadas pelas camadas de decodificação considerando as conexões residuais na arquitetura da rede.



Fonte: (PENG et al., 2024)

Como é sabido que as camadas mais profundas produzem previsões mais corretas em termos de localização, outro ponto de ajuste envolve o componente GO-LSD. Este, então, utiliza uma estratégia de otimização bidirecional que transfere a tarefa de regressão de localização

(referente às *bounding boxes*) das camadas mais profundas para as camadas mais superficiais através da autodestilação, ao mesmo tempo que simplifica as tarefas de predição residual para camadas mais profundas. As camadas superficiais são as camadas iniciais do modelo, enquanto que as camadas mais profundas são as camadas finais; salienta-se que este ajuste introduz baixos custos no treinamento (PENG et al., 2024).

Assim, o GO-LSD destila o conhecimento de localização das distribuições refinadas para as camadas superficiais. Ao longo do treinamento, as camadas finais produzem cada vez mais rótulos (*labels*) mais precisos. As camadas superficiais alinham suas predições com as *labels* através do GO-LSD, de modo a obter mais precisão nas predições. À medida que as predições iniciais melhoram, as camadas posteriores focam em refinar resíduos menores. Em suma, são utilizadas as distribuições refinadas das camadas finais para destilar conhecimento para as camadas superficiais. Assim, este mecanismo de reforço leva a, ao longo do processo, obter localizações mais precisas (PENG et al., 2024). A precisão na delimitação das bounding boxes é crucial no presente estudo, pois um erro de localização pode levar à contagem dupla ou à omissão de ovos, impactando diretamente a acurácia do índice entomológico final.

Com isso, outros ajustes menores incorporam otimizações leves no custo computacional em módulos intensivos e em operações, o que por fim produz um melhor equilíbrio entre latência e precisão. Portanto, em experimentos, o D-FINE alcança 54% Average Precision (AP) no dataset Common Objects in Context (COCO) a 124 Frames Per Second (FPS), bem como 55.8% a 78 FPS. Pré-treinado no dataset Objects365, D-FINE na escala L e X, obtém 57.1% AP e 59.3% AP, respectivamente, de modo a superar os modelos de detecção de objetos de então. Ao considerarem-se os modelos DETR, há um aumento de desempenho de 5.3% AP em favor do D-FINE, com a introdução de parâmetros e custos de treinamento também mínimos. Portanto, constitui-se num modelo competitivo tanto para uso em aplicações com foco em precisão quanto ainda para uso em cenários de tempo real (PENG et al., 2024).

2.4.8 MPRNet

No contexto de restauração de imagens com redes neurais, a tarefa visa, a partir de uma imagem degradada e tendo conhecimento do processo que causou a degradação, obter-se sua versão original (GONZALEZ; WOODS, 2015; ZAMIR et al., 2021). Nesse ínterim, há uma tarefa desafiadora, dado *trade-off* entre valorizar detalhes espaciais ou detalhes contextualizados de alto nível no processo de recuperação de imagens. Nesse quesito, o Multi-Stage Progressive

Image Restoration (MPRNet) insere nesse ambiente uma arquitetura para otimizar esse equilíbrio (ZAMIR et al., 2021).

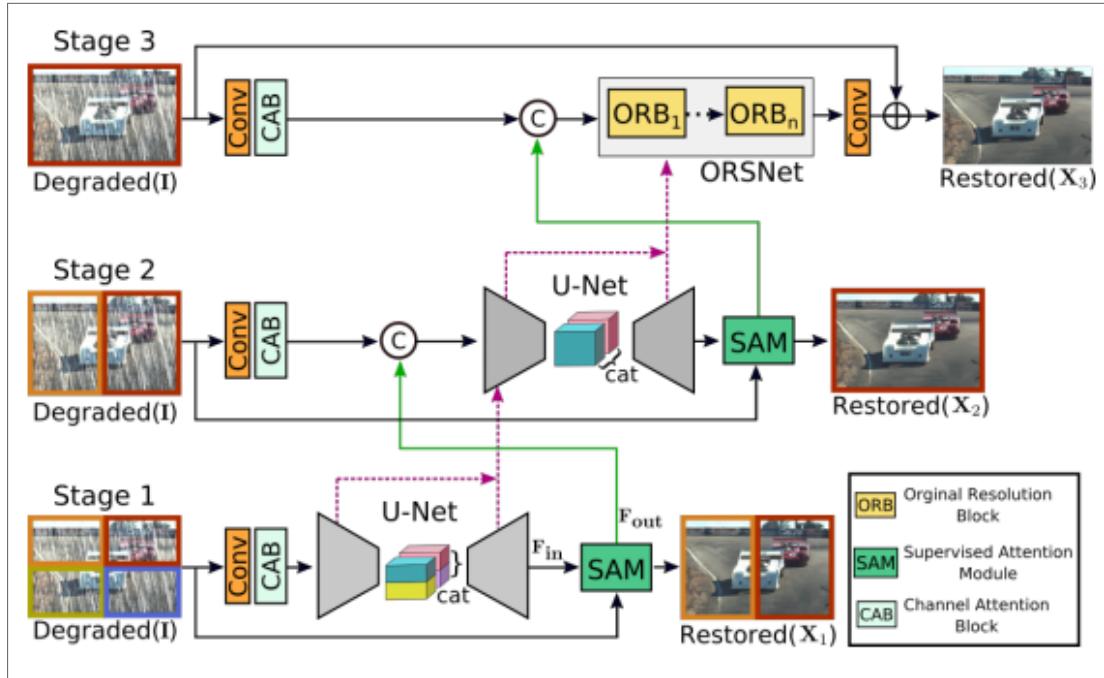
Para este tipo de tarefa, modelos baseados em CNN, dividem-se entre três abordagens principais: Abordagens *single-stages* (de estágio único), *multi-stage* (multiestágio) e de *attention*. O MPRNet, enquadra-se na abordagem multi-estágio, que caracteriza-se por realizar a restauração de forma progressiva, tendo em cada estágio uma sub-rede leve. Assim, a arquitetura do MPRNet aprende de maneira progressiva as funções de restauração, de modo a dividir este processo em etapas ou estágios gerenciáveis. *A priori*, o MPRNet aprende as *features* contextualizadas com o uso de arquiteturas de codificador e decodificador e, *a posteriori*, combina essas *features* com um ramo de alta resolução que retém informação local. Em cada etapa ou estágio, é introduzido um *design* adaptativo à nível de *pixel* que utiliza o mecanismo de atenção supervisionada no próprio local para novamente ponderar a *features* locais. A troca de informações entre os estágios é um ponto chave dessa arquitetura (ZAMIR et al., 2021).

Assim, do ponto de vista arquitetural, o MPRNet divide-se em três estágios para restaurar a imagem de forma progressiva. Então, os primeiros dois estágios, como mostra a Figura 16, baseiam-se em sub-redes de codificação e decodificação (*encoder-decoder*) que aprendem as informações mais amplas da imagem, devido ao uso de *recpetive fields* grandes. Dado que esta tarefa é senível e requer que os *pixels* da imagem entrada correspondam aos da saída, o terceiro estágio da arquitetura emprega uma sub-rede que atua na resolução da imagem de entrada sem reduzir suas dimensões, de modo a preservar as texturas finas na imagem final restaurada (ZAMIR et al., 2021).

Deste modo, ainda, os estágios incorporam um módulo de atenção supervisionada entre cada dois estágios. Com a supervisão das imagens de referência (*ground-truth*), o módulo reescalas os mapas de ativação dos estágios anteriores antes de passá-los para o próximo estágio. Outrossim, há um mecanismo de fusão de *features* entre os estágios de maneira que as *features* contextualizadas multiescala intermediárias da sub-rede anterior auxiliam a consolidar as *features* intermediárias da próxima sub-rede. Em que pese o MPRNet agregar vários estágios, cada um tem acesso à imagem de entrada. É também adaptada uma hierarquia *multi-patch* na imagem de entrada, de forma a dividí-la em *patches* não sobrepostos (sem intersecção) da seguinte forma: 4 para o estágio 1, 2 para o estágio 2 e a imagem original para o terceiro estágio.

Quanto às suas tarefas divide-se em: *Image deblurring* (para remover borramento), *image denoising* (para remover ruído em geral) e *image derraining* (para remover traços de chuva).

Figura 16 – Arquitetura multiestágio do MPRNet: Observa-se que os estágios primeiros extraem características multiescala contextualizadas e os últimos estágios atuam na resolução da imagem original para se obter saídas precisas do ponto de vista espacial.



Fonte: (ZAMIR et al., 2021)

Quanto à *image deblurring*, por exemplo, o modelo foi avaliado no *dataset* GoPro e HIDE. De um modo geral, o modelo supera modelos de então, com os quais foi comparado. De modo geral, o MPRNet obtém desempenho significativo em *datasets* de *benchmark*. Quanto à tamanho, o modelo é considerado leve e eficiente quanto à tempo de execução, o que torna seu uso favorável em ambientes de baixos recursos computacionais.

2.5 CONCLUSÃO DO CAPÍTULO

O capítulo 2 apresentou uma revisão teórica dos conceitos básicos de que este trabalho trata. Por início, foram abordados conceitos relacionados ao mosquito *Aedes aegypti* e doenças relacionadas a este vetor. Por sequência, foi discorrida uma elucidação acerca dos sistemas de vigilância do vetor, em especial da vigilância entomológica, destacando-se o uso de ovitrampas. Ainda, foi introduzido a área de processamento de imagens, incluídas algumas técnicas de realce de imagens. Ademais, deteve-se sobre a visão computacional e suas aplicações. Além disso, expôs-se de forma sucinta a aprendizagem profunda, bem como sua aplicação na visão computacional. Por fim, tratou-se dos modelos envolvidos no contexto deste trabalho: YOLOv10, YOLOv12, RF-DETR, D-FINE e MPRNet.

3 TRABALHOS RELACIONADOS

Neste capítulo, serão apresentados trabalhos voltados à contagem automática de ovos de *Aedes aegypti* em ovitrampas, abrangendo abordagens encontradas na literatura tanto voltadas à aplicação de Processamento de Imagens, por início, e Aprendizagem Profunda, por fim. Esta divisão baseia-se nas categorias de principais abordagens usadas na literatura para esse caso ao longo do tempo.

3.1 CONTAGEM AUTOMÁTICA COM PROCESSAMENTO DE IMAGENS

Abordagens primárias de contagem automática de ovos de *A. aegypti* em ovitrampas envolvem Processamento de Imagens. Assim, (SANTOS et al., 2008) propôs uma abordagem nesse sentido, que consiste em aplicar segmentação de cor e filtros não lineares baseados em morfologia matemática.

As imagens foram obtidas por câmera digital com 7.2 MP (*megapixels*), LCD 2.5, com aplicação de *zoom* óptico de 4.5 vezes e com lente Leica DC Vario Elmarit. As ovitrampas foram digitalizadas com resolução de 700 dpi e *zoom* óptico de 4 vezes. Ao fim, obtém-se uma imagem digital no sistema RGB com 3000 x 2300 *px* de dimensão. Para fins de experimentos, as imagens foram divididas em seções (sub-imagens) e a contagem do número correto de ovos (contagem de referência) para fins de comparação com os métodos desenvolvidos se deu manualmente de forma visual através das imagens. Os autores propuseram três métodos de contagem. No “Método 1”, as imagens são convertidas para o sistema YIQ, o canal I foi segmentado e é aplicado uma binarização com limiar fixo de 130. Após, é aplicado um algoritmo de componentes conectados (rotulação de componentes conexos), de modo a aplicar um rótulo em cada área branca. Em seguida, as áreas menores do que 100 *px* (tamanho médio de um ovo) são desconsideradas. Após isso, a imagem é filtrada com a operação morfológica de fechamento, tendo uma imagem de um ovo levemente inclinado como elemento estruturante da operação. As dimensões deste elemento são de 8 x 13 *px*. Por fim, foi calculada a média de *pixels* brancos na imagem, considerando que um ovo ocupa 170 *px*. Assim, portanto, a quantidade de ovos é obtida.

Por sua vez, no “Método 2 - Limiar Fixo”, a imagem também é convertida para o sistema YIQ e o canal I é segmentado. A partir daí, se introduzem mudanças pontuais no fluxo, com

o restante do fluxo permanecendo inalterado. Nesse caso, a limiarização é feita com o limiar de 200 e esse valor obtido após análise do histograma da imagem. Outro, o “Método 2 - K-Médias”, faz a binarização automática com base no método k-médias [7, 8, 9, 10] com 3 entradas, 4 classes, taxa de aprendizagem de 0.1 e máximo de 200 iterações. Para a contagem, é considerado o tamanho médio de um ovo como 220 *px*.

Os experimentos foram feitos considerando 6 imagens. Por fim, os resultados evidenciaram que o “Método 1” atingiu erro máximo de 25% na segunda imagem, o que equivale a uma diferença de dois ovos. Em geral, o erro foi de 9%. Por sua vez, o “Método 2 - Limiar Fixo” apresentou resultados superiores aos demais. Por fim, o “Método 2 - K-Médias” obteve taxa de erro média de 7.33%, de maneira que as duas versões do Método 2 obtiveram erro menor do que 10%.

O método apresentado neste trabalho possui aplicação limitada, dada à forte dependência da intensidade dos *pixels* da imagem, considerando que uma palheta pode ter várias condições de iluminação. Ademais, a base de experimentos é pequena para evidenciar o êxito da solução desenvolvida.

Semelhantemente, (GUSMÃO; MACHADO; RODRIGUES, 2009) propõe contagem de ovos do vetor através de Processamento de Imagens voltadas a exploração dos sistemas de cor e aplicação do algoritmo de agrupamento K-Means. Em princípio, as imagens RGB são convertidas para o espaço L*a*b*. Os componentes a* e b* são usados para agrupar a imagem de entrada usando o mapa K-Means. O número de *clusters* (agrupamentos) é definido como 3, referentes a: Ovo, ovitrampa e regiões intermediárias. O algoritmo aleatoriamente seleciona os vetores iniciais de média e como critério de parada foram usados dois argumentos: O valor máximo de 100 iterações e a estimativa de coesão. O algoritmo, então para quando a soma das distâncias de cada objeto não pode mais diminuir. O algoritmo repete o agrupamento 3 vezes, cada qual com um novo conjunto de posições iniciais do vetor de média e retorna a solução com menor valor de soma das distâncias dos vetores ponto-média. Após isso obtém imagens de agrupamento; a imagem é convertida para o sistema Hue, Saturation, Value (HSV) e o canal H (matiz) é segmentado. Em seguida, para cada *cluster*, é obtido o valor médio de matiz e o valor de cada grupo caso, é atribuído para 1 (caso seja superior a 0.5) e para 0 (caso contrário). Com isso, obtém-se uma imagem binária e é então aplicado um algoritmo de rotulação de componentes conexos. Após isso, são desconsideradas as regiões com menos de 140 *pixels*. Por fim, considerou-se que um ovo ocupa uma área de 357 *px* e então, calculou-se a quantidade de ovos com base numa média aritmética considerando esse valor.

Os resultados mostraram um erro individual máximo de 45.45% na primeira imagem, o que correspondeu a uma diferença de dez ovos. No geral, o erro médio foi de 1.16% com um desvio padrão de 4.15. O método apresenta uma melhoria de erro médio de 6.66% para 1.16% em relação ao proposto por (SANTOS et al., 2008), em que pese o erro individual ter aumentado de 25% para 45.45% e o desvio padrão, de 1.6 para 4.15.

Esta abordagem, novamente, apresenta uma dependência importante em relação à iluminação e cor da imagem, desconsiderando totalmente outros traços e características do objeto de interesse, além de ser experimentada com poucas imagens.

(SILVA; RODRIGUES; ARAUJO, 2012a) desenvolveu um sistema de aquisição de imagens de palhetas de ovitrampa associado a um método de contagem, tanto automática como também semi-automática (assistida). Observa-se nesse trabalho, ainda a aplicação de técnicas de processamento de imagens. As informações obtidas pelo sistema são enviadas para um servidor *web*, onde são analisadas as palhetas por técnicas de processamento de imagens, em especial baseada nos processos de segmentação, filtragem e quantificação. No geral, a contagem é feita em três etapas, em que a primeira é segmentação, que é composta por uma operação AND entre duas máscaras geradas por duas formas de segmentação distintas (por cor e por limiarização). A segunda, é uma filtragem morfológica, em que utilizou-se um filtro de moda com dimensões de 3 x 3 *pixels*. Em seguida, foi feita uma filtragem morfológica de abertura com um elemento estruturante no formato de um disco com raio de 2 *px*. A terceira etapa, por fim, consiste na contagem dos ovos por meio do cálculo de média tendo considerando a área média de um ovo como 160 *px* (valor determinado após experimentos em 100 imagens de ovos). Na análise de 100 imagens, o método automático obteve um erro médio de 16.26% e o erro global, considerando a totalização das contagens manual e automática, foi de 2.67%.

Observa-se que este método é estreitamente vinculado à características das imagens nessas condições, haja vista a própria menção de que o valor médio do tamanho de um ovo pode variar ao tratar-se de imagens de tamanho diferente. Aqui, apesar de o conjunto de imagens ser maior, ainda é baixo dado o cenário e o método de contagem é estreitamente associado às características específicas dessas imagens.

Ademais, em (FEITOSA et al., 2015), a partir de imagens capturadas com microscópio Universal Serial Bus (USB), com 96 dpi de resolução em 800 x 600 *pixels* de dimensão (além das imagens disponibilizadas por (PORTELLA, 2009), foram tratados métodos de processamento de imagens. A partir de uma imagem RGB, o canal R foi segmentado e o histograma da imagem foi normalizado. Após, utilizou-se a técnica de limiarização *Otsu*, seguida por uma operação

morfológica em que o elemento estruturante tem o formato de um disco com raio de 1 *pixel*. Em seguida, foi aplicado um algoritmo de rotulação de componentes conexos. Após, as áreas com menos de 140 *pixels* foram desconsideradas e, a partir de um valor determinado de área de ovo (362 *pixels*), foi calculada a média de *pixels* brancos e, assim, obteve-se a quantidade de ovos. Outro método de contagem proposto, distingue-se em pontos-chave, como: Utiliza as imagens no espaço de cores L*a*b o algoritmo K-Means e pelo valor definido como área de um ovo para fins de cálculo (357 *pixels*). Por fim, a contagem é feita por cálculo de média, como no primeiro método. Em geral, o método 1 obteve o melhor erro global (erro 0) e o método 2 obteve o melhor resultado em média de erro: 18.12%.

Entre as limitações do trabalho, destacam-se: A necessidade de controle de iluminação por parte do dispositivo de captura, além da necessidade de obter uma massa maior de imagens. Qual os anteriores, este apresenta forte associação às características específicas das imagens, como alta dependência em relação à cor dos ovos, além de ser especializado apenas nessas condições de captura, já que é estreitamente ligado a esse tamanho de imagem.

Por outro lado, (COSTA, 2017), utiliza um Sistema de Inferência Fuzzy (SIF) e verifica os resultados com base em 206 imagens. Como resultados, obteve acurácia de 98.94% e coeficiente de correlação de Matthews de 0.9 para classificação e contagem dos ovos, comparado com a contagem manual. Contudo, há a presença de um valor alto de falsos positivos. Ainda, observa-se, ainda ligação dessa abordagem proposta com as características específicas das imagens (como de cor, luminosidade etc) justo por utilizar técnicas de processamento de imagens como ponto chave, a exemplo de técnicas de binarização.

Observa-se o uso de Processamento de imagens, agora, aliado à Aprendizagem de Máquina, no trabalho proposto por (ROCHA; BIZERRA; MAGALHÃES, 2019), onde utiliza-se principalmente o algoritmo *Haar Cascade* para otimizar o reconhecimento dos objetos e buscar evitar falsos positivos. Menciona-se que imagem passa antes por um filtro de realce (*high-boost*). Usa-se um equipamento eletrônico para captura (que, inclusive, usa Light-Emitting Diode (LED) de branca) e processamento de imagens para realização da contagem. Apresentou resultados como: De 26 ovos contidos numa imagem, reconheceu 14, no geral teve alguns falsos positivos.

Como limitações desse trabalho, tem-se que: Há a necessidade de uma base maior de imagens e de um dispositivo de captura, que além de iluminar bem as imagens (este apresentou alguns pontos de melhoria), seja de baixo custo e de fácil manuseio. Além disso, a contagem necessitaria ser mais capaz de lidar com problemas comuns em palhetas: Ovos adjuntos, sobrepostos, segmentados, dentre outras disposições.

Como observa-se, a somente aplicação de métodos de processamento de imagens é altamente vinculada às condições das imagens para as quais são implementadas, como intensidade de cor e dimensões da imagem. Ademais, é imperioso a adoção de métodos computacionais capazes de, em fato, aprender a reconhecer ovos de *Aedes aegypti*, não somente aplicar métodos em condições controladas e que sejam estritamente localizados. Imagens com sobreposição de ovos, oclusão, além de *background* escuro ou ruidoso ou mesmo imagens ruidosas no geral são encontradas em cenários reais.

Portanto, para cenários e imagens de mundo real, são necessários métodos capazes de modelar suficientemente a complexidade das imagens e dos objetos de interesse, de modo atender os pontos supracitados. Ante isso, dados os avanços da Aprendizagem Profunda e sua aplicação para Visão Computacional, a literatura passou a envolver estas áreas na tarefa de contagem de ovos de *A. aegypti*, ou seja, a aplicação de técnicas de Processamento de Imagens (ainda que associadas a AM) para o problema de contagem automática de ovos de *Ae. aegypti*, tem perdido, progressivamente, importante espaço para dar lugar a métodos de DL.

A respeito da contagem automática com processamento de imagens, a Tabela 1 faz um comparativo dos trabalhos acima mencionados.

Tabela 1 – Comparativo de trabalhos na literatura que utilizam Processamento de Imagens para contagem automática de ovos de *A. aegypti*

| Trabalho | Técnica | Dispositivo de captura de imagens | Melhores resultados | Limitações |
|------------------------------------|---|---|--|---|
| (SANTOS et al., 2008) | Segmentação de cor e filtros não lineares baseados em morfologia matemática | Câmera digital com 7.2 MP, LCD 2.5", com aplicação de zoom óptico de 4.5 vezes e com lente Leica DC Vario Elmarit. | Erro médio de 7.33% | Dependência estrita da intensidade dos pixels da imagem. Modelo não aprende a identificar ovos propriamente dito, mas se baseia em técnicas de filtragem e cálculos de área média de ovo. |
| (GUSMÃO; MACHADO; RODRIGUES, 2009) | Segmentação de cor e filtros não lineares baseados em morfologia matemática | Câmera digital 7.2 MP, LCD 2.5", zoom óptico 4.5x e lente LEICA DC Vario Elmarit | Erro médio de 1.16% e desvio padrão de 4.15. | Abordagem testada em poucas imagens e altamente vinculada às características particulares das imagens e não ao reconhecimento de padrões que caracterizam um ovo. |
| (SILVA; RODRIGUES; ARAUJO, 2012a) | Segmentação, filtragem e quantificação. | Dispositivo com placa de controle, câmera digital com 7,1 megapixels e modo "super-macro", sistema de iluminação e suporte para translação de palhetas. | Erro médio de 16.26% e erro global de 2.67%. | Método específico às características das imagens utilizadas, como iluminação, cor e tamanho médio de ovos. |
| (FEITOSA et al., 2015) | Processamento de histograma, segmentação, filtragem, limiarização e agrupamento | Microscópio USB com 96 dpi de resolução. | Erro global 0 erro médio de 18,12% | Método fortemente associado às características específicas das imagens e alta dependência em relação à cor dos ovos, além de ser específico para essas condições de captura. |
| (COSTA, 2017) | Segmentação, extração de características e lógica fuzzy. | Câmera BIOFOCUS acoplada a um microscópio Stemi 305 body (Carl Zeiss Microscopy GmbH). | Acurácia de 98.94% para classificação de ovos e coeficiente de correlação de Matthews de 0.9 em relação à contagem manual. | Ligação da abordagem com as características específicas das imagens, como cor, luminosidade etc. |
| (ROCHA; BIZERRA; MAGALHÃES, 2019) | Filtragem e algoritmo Haar Cascade para reconhecimento de objetos. | Equipamento com dispositivos eletrônicos, como câmera de 5MP, luzes de LED branco, dentre outros. | Taxa de recall de cerca de 60% | Base experimentos pequena, dificuldade ainda com iluminação e com ovos adjuntos ou sobrepostos, além de dispositivo de captura com custo considerável. |

Fonte: Elaborada pelo autor (2025).

3.2 CONTAGEM AUTOMÁTICA COM APRENDIZAGEM PROFUNDA

A priori, (SANTANA et al., 2019) propõe uma abordagem baseada em redes neurais para contagem dos ovos de *A. aegypti* (e também de *A. albopictus*) em palhetas de ovitrampa. Propôs um dispositivo de *hardware* que possui integrado nele uma câmera (microscópio digital) para captura de imagens, um módulo de contagem dos ovos e uma palicação *web* para exibir os resultados das contagens efetuadas.

Então, foi usado um modelo de detecção de objetos, o Region-based Fully Convolutional

Networks (R-FCN), proposto por (DAI et al., 2016), pré-treinado no *dataset* COCO e com um conjunto de 984 imagens. O R-FCN alcançou precisão de 91% na classificação dos ovos e para validar a utilização do dispositivo de captura, foram utilizadas 10 palhetas reais obtidas.

Entretanto, pontua-se que a solução proposta tem redução do desempenho em relação à disposição de ovos (occlusão de ovos, ovos adjuntos *etc*). Destaca-se, ainda, o custo associado ao dispositivo de captura, dado que possui diversas interfaces de conexão, como USB, MicroUSB, High-Definition Multimedia Interface (HDMI), interface de áudio, porta *Ethernet* *etc*. Um outro ponto importante é a sobreposição na captura das imagens pelo dispositivo de captura, dado que o trabalho menciona que há uma sobreposição de 10%. Esta sobreposição causa recontagem de ovos (ovos são contados mais de uma vez) e leva a erro a contagem de uma palheta completa, que é requisito essencial para obtenção dos índices de ovos, como o IDO e o IDV. Estes índices são cruciais e parte do fluxo de implementação da Vigilância Entomológica, como recomendada o Ministério da Saúde do Brasil (Ministério da Saúde – Brasil, 2025d).

Ainda com uso de *Deep Learning*, (SILVA et al., 2021) propõe uma metodologia para segmentação de ovos para auxiliar a contagem. Com imagens obtidas por câmera digital de 7.2 MP LCD 2,5 polegadas, zoom óptico de 4.5 vezes e lente Leica DC Vario Elmari Foram digitalizadas 10 palhetas e obteve-se um total de 30 imagens de 1024 x 768 *pixels* de dimensão. A partir de uma imagem, gera 12 sub-imagens de dimensão 256 x 256 *pixels* e então, obteve-se 228 imagens após esse pré-processamento. Após isso, ocorre segmentação, etapa na qual são experimentados o uso de uma rede U-Net e de uma SegNet, além de uma Resnet50 pré-treinada. Em seguida, o pós-processamento é feito com uma operação morfológica de fechamento em que o elemento estruturante é um disco com raio de 7 *pixels*. Logo depois, utilizou-se uma técnica para fechar os buracos segmentados em regiões de ovos e foi aplicado o algoritmo de rotulação de componentes conectados. Para fins de contagem de ovos, se uma região contiver a área menor do 1.5 vezes o tamanho médio de um ovo (1321 *pixels*), é considerado como apenas um ovo. Caso contrário, é realizada a divisão do tamanho da região pelo tamanho médio de um ovo e o resultado inteiro obtido corresponde à quantidade de ovos. Quanto a resultados, os melhores foram com U-Net, com acurácia de segmentação de 98.65% e erro médio quadrático de 4.25% na contagem dos ovos.

Em que pese a abordagem atual ter substituído a limiarização de imagem por segmentação usando redes neurais, ainda apresenta limitações relacionadas a: Tamanho (área) dos ovos não é um valor seguro para estimar a quantidade de ovos numa imagem, já que diferentes

tamanhos de ovos ou mesmo ovos segmentados (parcialmente na imagem, parcialmente não) deturpariam a estimativa feita de contagem. Ademais, a quantidade de imagens testadas poderia ser maior, de modo a envolver um número expressivo de palhetas. Por fim, o método não aprende a reconhecer os ovos de fato, mas se baseia, em última instância, em estimativas e aplicação de redes neurais apenas atua na tarefa de segmentação, não na detecção em si.

Também, (SILVA, 2021) criou um dispositivo de *hardware* que possui uma câmera digital acoplada a ele para captura de imagens. A partir de 100 palhetas, aplicou técnicas de *data augmentation* e, assim, criou uma base de dados com 3703 imagens. Analisou a discrepância entre contagem automática e manual, além de criar um *software* para auxiliar técnicos na contagem dos ovos. Com a base de dados de imagens criada, treinou modelos de aprendizagem profunda voltados a detecção de objetos, como EfficientDet, YOLOv4 e RFCN e foi verificado nos experimentos que os modelos EfficientDet e YOLOv4 foram superiores.

O dispositivo de captura poderia envolver componentes que o tornasse de mais baixo custo e principalmente de fácil manuseio, operação e transporte, dado que isto tornaria mais viável seu uso por parte dos profissionais que atuam em campo na implementação da vigilância e controle do vetor, como instalação, contagem de ovitrampas e monitoramento em geral. Inclusive, o fato de o *hardware* de captura envolver uma câmera digital e de principalmente ser razoavelmente grande relativo a uma palheta, é um fator de peso em desfavor da praticidade de uso, ainda mais em campo. Ainda, o dispositivo causa sobreposição na captura de imagens, o que leva a estatísticas equivocadas sobre a contagem em palheta completa, pois regiões da palheta são consideradas mais de uma vez. Ainda sobre o dispositivo de captura, o autor refere-se à limitações por conta do foco da câmera digital, uma vez que esta se move sobre a palheta e, por vezes, captura imagens com ruído relacionado ao foco, o *defocus blur*.

Outro ponto é que não foi considerada nem medida a contagem em palheta completa, dado fundamental para obter os indicadores de ovos, como o IDO e outros para fins de vigilância entomológica. Ainda, o autor menciona a questão de imagens ruidosas ou de baixa qualidade como fatores de comprometem a capacidade dos modelos de reconhecer os objetos, o que impacata diretamente no resultado da contagem dos ovos de mosquito. Como palhetas de ovitrampas são instaladas em diversos locais ao ar livre, estão, assim, suscetíveis à diversos fatores que contribuem para que a palheta, ao ser considerada para contagem, tenha ruídos. Assim, as metodologias de contagem de ovos, devem lidar com este fator, de maneira que lide também com imagens ruidosas de diversos ruídos existentes. Acerca disso, a literatura atual menciona abordagens possíveis, como a restauração de imagens degradadas por redes neurais.

Ainda, a validação da metodologia de contagem quanto à comparação da contagem manual com a contagem automática, considerou 13 palhetas, número que poderia ser mais expressivo para se avaliar uma metodologia de contagem automática proposta. Por fim, dados os recentes avanços em termos de modelos de detecção de objetos com desempenhos superiores aos de então, com alta precisão e baixa latência amparados em amplos experimentos, é importante treinar esses modelos do estado da arte, refiná-los com o fim de se obter um desempenho melhor. Em geral, estas melhorias, seria ideal uma metodologia de contagem que considerasse estes pontos-chave com vistas a contribuir com mais avanços nesta área do conhecimento e relacionados a contagem automática de ovos de *A. aegypti*.

Por sua vez, (JAVED et al., 2023) apresenta uma solução que consiste num *software* gratuito de contagem (EggCountAI), cuja contagem baseia-se no modelo Mask Region-based Convolutional Neural Network (Mask R-CNN). As palhetas, nesse caso, distinguem-se por serem de tiras de lixa (*sandpaper strip*). As imagens foram capturadas com uma câmera Olympus Tough TG-6 para obterem-se as imagens macroscópicas, ao passo que, as imagens microscópicas foram coletadas com um microscópio Nikon SMZ18 ajustado para uma ampliação de 8x com uma lente objetiva de 1x, utilizando o *software* *NISElements* da Nikon.

Para cada tipo de imagem (microscópica e macroscópica), um modelo é treinado para detectar ovos e obtém-se, para imagens microscópicas, mAP de 0.92, Mean Average Recall (mAR) de 0.90 e F1-Score de 0.91. No modelo treinado para imagens macroscópicas, obtém-se 0.91 mAP, 0.90 mAR e 0.90 de F1-Score. O *software* EggCountAI conta automaticamente ovos em um diretório contendo imagens de tiras de lixa.

Para testar a solução, comparou-se com as ferramentas de contagem *ICount* e *MECVision* usando 10 imagens microscópicas e 10 macroscópicas. Os resultados evidenciaram superior desempenho em relação às outras ferramentas, de modo que se obteve acurácia geral de 98.88% para as imagens microscópicas e de 96.06% para imagens macroscópicas. Por sua vez, *ICount* obteve 81.71% e 82.22%, respectivamente para imagens micro e macro, enquanto que o *MECVision* obteve 68.01% e 51.71% para imagens micro e macro, nessa mesma ordem, o que ressalta a superioridade da abordagem proposta. Quanto a erro médio, a solução, o EggCountAI obteve 1.90 para imagens micro e 74.30 para imagens macro, além de obter desempenho melhor ao lidar com ovos sobrepostos ou agrupados.

Em relação a essa metodologia, observa-se o conjunto bastante reduzido de imagens para validação do desempenho da solução, o que limita a quantidade de instâncias de ovos. Observa-se que as imagens não apresentam ruídos significativos (ou mesmo forte presença deles), pelo

fato de o substrato de ovoposição ser de tiras de lixa com fundo amarelado. A propósito, isso limita a aplicação desta metodologia apenas a esses tipos de imagens, já que não representa todos os cenários de utilização quanto ao substrato para deposição de ovos utilizado. Inclusive, não é feito um tratamento mais específico para lidar com imagens ruidosas, como muito escuras ou com ruído de borramento. Ademais, a forma de captura das imagens envolve dispositivos que aumentam o custo do processo como um todo. O uso câmera e do microscópio supracitados encarecem financeiramente o processo de aquisição das imagens e suscita a necessidade de utilizarem-se dispositivos de baixo custo para alcançar resultados similares. Com dispositivo de mais baixo custo para captura e de melhor manuseio, transporte e utilização e campo (principalmente), uma metodologia, como de contagem, proporcionaria maior reprodutibilidade para cenários de aplicação real.

Ademais, o trabalho (NARANJO-ALCAZAR et al., 2024) também aplicou aprendizagem profunda para classificar ovos em duas classes: Eclodidos e não eclodidos. Propôs também reconstruir a imagem de uma palheta a partir das imagens parciais diminuindo a duplicidade e corte de ovos entre imagens (ovos segmentados), mas ainda mantendo certo grau de sobreposição. O dispositivo de captura desenvolvido possui acoplado a si um microscópio DinoLite AM4013MZT e é capaz de capturar imagens de várias palhetas sem ser necessário remover e colocar as palhetas ao longo do processo e também proveu a contagem em palheta completa. A partir de uma palheta, obtém-se 165 imagens parciais e a movimentação do microscópio em relação à palheta gera sobreposição, como mencionado pelos autores e *dataset* obtido consistiu de 96 imagens em que 24 foram separadas para o conjunto de teste, as quais continham 215 instâncias de ovos eclodidos e 1160 de ovos não eclodidos. Em relação à contagem, utilizaram-se modelos de segmentação semântica, quais sejam: Mask R-CNN e Cascade Mask-RCNN. Os resultados mostraram que o primeiro obteve melhor desempenho com 0.91 mAP@.5, contra 0.89 mAP@.5 do outro modelo.

Em relação a limitações, destaca-se que o dispositivo de captura, em que pese seja prático, poderia ser de melhor transporte e manuseio para atividades de vigilância entomológica em campo. Ademais, há uma sobreposição importante na captura das imagens, fato que inclusive, fez com que imagens consecutivas de uma palheta não fossem rotuladas para evitar considerar o mesmo ovo duas vezes. Portanto, há a necessidade de se tratar o problema da sobreposição de regiões da superfície da palheta na captura. Ademais, o conjunto de imagens treinamento é ainda muito pequeno, bem como a quantidade de instâncias de ovos também o é, nem foi mencionada a aplicação de técnicas de aumento de dados. Ademais, modelos mais recentes

de segmentação estão disponíveis na literatura, os quais possuem desempenho importante evidenciado com base em amplas experimentações. Por fim, não foi apresentada a contagem geral de uma palheta completa para assim, compararar-se com a contagem manual e calcular o erro em métricas como erro médio absoluto, por exemplo.

Por fim, (VICENTE et al., 2024) com o uso de um estereomicroscópio Leica MC170 HD, capturou imagens de palhetas de ovitrampas, criou um conjunto de treinamento e propôs três modelos de detecção de objetos, como Faster R-CNN, Side-Aware Boundary Localization (SABL) e *FoveaBox*. A base de imagens é composta de imagens obtidas em campo e em laboratório. No geral, há 247 imagens, sendo que 123 foram coletadas em campo e 124, em laboratório, o que totaliza 12.513 instâncias de ovos anotados. Pontua-se que os ovos obtidos em laboratório são provenientes de ovitrampas e são maturados, mas distinguem-se pelo fato de que um filtro de papel foi colocado nas ovitrampa para manter os níveis adequados de umidade para facilitar a eclosão e, por fim, na fase adulta, os mosquitos fêmeas foram alimentados com uma solução de capim e repasto sanguíneo para aumentar a postura de ovos com maior viabilidade.

Os resultados mostraram que o *FoveaBox* obteve melhor desempenho que os demais modelos, especialmente na contagem de ovos muito próximos uns dos outros (disposição que é comum nessa base de dados) ou agrupados (sobrepostos). Foi verificado que o erro médio aumenta de forma expressiva à medida que mais imagens são consideradas, sendo de 2.68 com até 100 imagens, 26.21 entre 101 e 300 imagens e, por fim, 123.25 para mais de 300 imagens.

Isto colocado, com relação a pontos-chave de melhoria, alguns substratos de opositoção utilizados não são de superfície planas, o que levou a dificuldades no reconhecimento de objetos posicionados lateralmente à estrutura, o que levou a inviabilidade de anotação e de reconhecimento pelo modelo; isto pode levar, num cenário de inferência em aplicações do mundo real, a expressiva subcontagem dos ovos nas palhetas, fato este que inviabilizaria a solução para fins de obtenção de indicadores relacionados à vigilância entomológica. Ademais, houve dificuldades também na contagem dos ovos devido à presença numerosa de ovos nas imagens, dispostos muito próximos uns aos outros e em grupos; ainda, a presença de ruídos foi apontada como um fator que traz dificuldades à contagem. Também, modelos do estado da arte podem ser aplicados com o fim de se obter desempenho melhor na contagem, além da aplicação de modelos de restauração de imagens para tratar cenários desafiadores como imagens ruidosas. Por fim, há a necessidade de coleta de mais imagens para treinamento e

obtenção de modelos com melhor capacidade de generalização, considerando o cenário de mundo real em que devem ser aplicados.

Para fins de comparação, a Tabela 2 expõe de forma resumida as abordagens utilizadas nos trabalhos desta seção.

Tabela 2 – Comparativo de trabalhos na literatura que utilizam Aprendizagem Profunda para contagem automática de ovos de *A. aegypti*

| Trabalho | Técnica | Dispositivo de captura de imagens | Melhores resultados | Limitações |
|--------------------------------|---|--|--|---|
| (SANTANA et al., 2019) | Algoritmo R-FCN para detecção de ovos | Dispositivo de hardware como um scanner com microscópio digital USB com zoom óptico de 500x | Precisão de 91% | 10% sobreposição em áreas da palheta, dispositivo de captura com custo significativo, dificuldade com sobreposição e ovos adjuntos |
| (SILVA et al., 2021) | U-Net, SegNet e Resnet50, morfologia matemática e segmentação | Câmera digital de 7.2 MP LCD 2,5 polegadas, zoom óptico de 4.5x e lente Leica DC Vario Elmari | Acurácia de segmentação de 98.65% e erro médio quadrático de 4.25% | Método não aprende a contar ovos, mas segmenta imagens com base em suas características específica. Base de experimentos ainda pouco expressiva |
| (SILVA, 2021) | EfficientDet, YOLOv4 e RFCN para detecção de ovos | Dispositivo de hardware feito com impressora 3D e com câmera de microscopia de 5MP | YOLOv4 com 92%, 94% e 93% nos conjuntos de treino, validação e teste e leva 14 segundos para inferência em 204 imagens | Dispositivo de hardware de captura ainda com custo considerável, acentuadamente grande e de difícil manuseio em campo no âmbito da vigilância entomológica. Não houve um tratamento de imagens ruidosas |
| (JAVED et al., 2023) | Mask-RCNN para detecção de ovos | Câmera Olympus Tough TG-6 e microscópio Nikon SMZ18 com zoom de 8x e lente objetiva de 1x, respectivamente macroscópica e microscópica | Acurácia geral de 98.88% para as imagens microscópicas e de 96.06% para imagens macroscópicas erro médio 74.30 (macro) e de 1.90 (micro) | Conjunto reduzido de imagens de validação. Não houve tratamento para imagens ruidosas já que as imagens não têm ruído significativo. Uso das referidas câmeras macroscópicas e microscópicas encarecem a metodologia de contagem de ovos |
| (NARANJO-ALCAZAR et al., 2024) | Modelos Mask-RCNN e Cascade Mask-RCNN para segmentação de ovos | Dispositivo de hardware com microscópio DinoLite AM4013MZT | 0.91 mAP@.5 | Dispositivo de hardware de captura significativamente grande e por isso, de difícil transporte e manuseio para atuação em campo no âmbito da vigilância entomológica. Há sobreposição considerável na captura dos ovos, além de o conjunto de treinamento ser pequeno. Não foi apresentada contagem total de uma palheta. Modelos do estado da arte podem ser aplicados |
| (VICENTE et al., 2024) | Faster R-CNN, Side-Aware Boundary Localization (SABL) e FoveaBox para detecção de ovos. | Estereomicroscópio Leica MC170 HD | FoveaBox obteve melhor correlação | Os ovos coletados em laboratório não estão em superfície plana, o que desfavorece a contagem, pois esconde ovos. Podem ser experimentados modelos mais recentes e com melhor desempenho |

Fonte: Elaborada pelo autor (2025).

3.3 CONCLUSÃO DO CAPÍTULO

Este capítulo apresentou trabalhos relacionados à tarefa de contagem automática de ovos de *A. aegypti* em palhetas de ovitrampa, divididos em duas abordagens principais: Processamento de Imagens, inicialmente, e Aprendizagem Profunda, por fim.

Evidencia-se o esforço contínuo dedicado à esta tarefa, bem como os desafios a ela relacionados. Em que pesem avanços importantes terem sido realizados, ainda permanecem áreas importantes de melhoria, como: Desenvolver dispositivo de captura de mais baixo custo e de fácil utilização, manuseio e transporte; utilizar maiores bases de dados, tanto para treinamento, quanto para fins de comparação entre contagem automática e contagem manual; treinar modelos do estado da arte na área de aprendizagem profunda, como, modelos que adotam mecanismos mais robustos, como *attention*, como YOLOv12 e D-FINE, baseados em *transformers*, como RF-DETR, dentre outros modelos e tecnologias com melhorias verificadas em reconhecidos *datasets* de *benchmark*, como COCO; buscar resolver problemas referentes à captura de imagens, como sobreposição de regiões; lidar com imagens ruidosas, como ruído de borramento por meio de modelos de restauração de imagens ou outra técnica com o mesmo fim. Portanto, esses pontos principais enfatizam a atual necessidade de avanços nessa área do conhecimento.

Assim, nesse contexto, o presente trabalho distingue-se positivamente pela criação de uma base de dados maior. Ainda, pontua-se a criação de um dispositivo para captura de imagens feito com impressora 3D. Além disso, caracteriza-se pela criação de um protocolo autoral para os modelos de contagem, além de serem experimentados vários modelos de aprendizagem profunda do estado da arte com treinamento para contagem a partir de imagens de baixa resolução capturadas por câmeras de *smartphones*. Também ressalta-se a aplicação de modelo de restauração de imagens baseado em redes neurais e sua inclusão no protocolo autoral de contagem e são incluídas e avaliadas também o impacto do uso técnicas de realce com processamento de imagens na contagem dos ovos, na estrutura do protocolo de contagem proposto. Desta forma, o capítulo seguinte detalha os procedimentos metodológicos que foram utilizados com vistas a atingir esses objetivos acima pontuados.

4 MATERIAIS E MÉTODOS

No âmbito da Metodologia Científica, este capítulo tem o intuito de esclarecer os materiais e métodos aplicados no desenvolvimento deste trabalho.

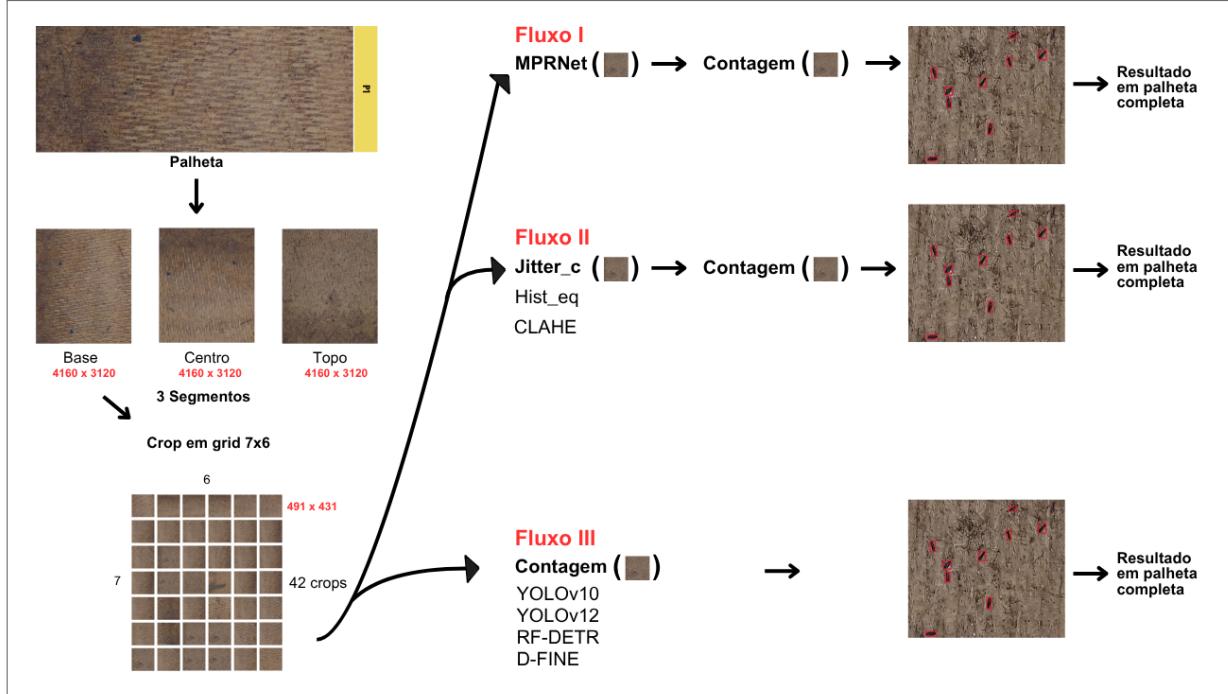
4.1 FLUXO DE DESENVOLVIMENTO DO TRABALHO

O fluxo proposto de contagem de ovos de *Aedes aegypti* é apresentado Figura 17. Dado como entrada uma palheta de ovitrampa (cerca de 15 x 5 cm), são capturadas 3 imagens digitais com apoio do dispositivo de captura; essas imagens são chamadas de “segmentos” e representam três divisões da palheta, assim nomeadas: Base, centro e topo, cada qual com dimensão de 4160 x 3120 pixels. Para cada cada segmento, é feita uma operação de corte na imagem em forma de grade, num corte de 7 linhas e 6 colunas. Esse processamento, resulta em 42 imagens chamadas “crops”, em que cada uma tem cerca de 491 x 431 px. Ao considerar os três segmentos da palheta, após essa operação obtém-se 126 “crops”.

Em seguida, o fluxo de contagem subdivide-se em três: Fluxo I, voltado à aplicação modelos de restauração de imagens baseados em Redes Neurais (tal como o MPRNet) como operação intermediária para após isso ser realizada a contagem por modelos de Aprendizagem Profunda; Fluxo II, cujo foco são técnicas de Processamento de Imagens, como Equalização de Histograma, para depois serem aplicados os modelos de contagem; e, por fim, Fluxo III, em que a contagem é feita diretamente por modelos sem a aplicação de operações intermediárias.

Deste modo, dado que uma palheta foi a entrada deste fluxo, obtém-se ao final o resultado da contagem de ovos de *Aedes aegypti* nesta palheta: As imagens com as detecções e o total de ovos na palheta.

Figura 17 – Fluxo Completo de Contagem



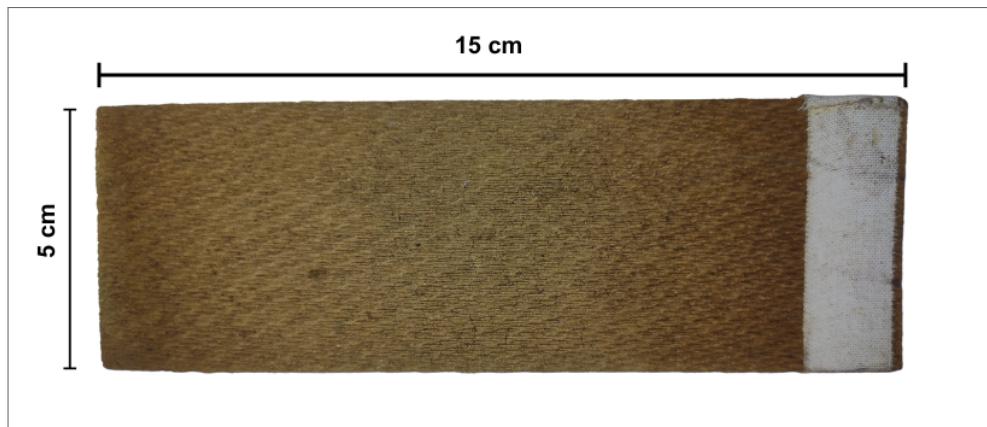
Fonte: Figura do autor (2025)

4.2 AQUISIÇÃO DE PALHETAS

Dada a colaboração existente entre a FIOCRUZ-PE e o Centro de Informática, foram fornecidas palhetas de ovitrampa utilizadas na vigilância entomológica, já abastecidas de ovos de *Ae. aegypti*. Ao total, cerca de 152 palhetas foram disponibilizadas e consideradas para este trabalho, cada qual com as características e dimensões apresentas na Figura 18.

Dentre essas palhetas, 28, foram fornecidas juntamente com a contagem estimada de ovos, conforme técnicos da FIOCRUZ-PE, e foram, portanto, consideradas na verificação dos resultados dos experimentos descritos nas seções posteriores deste trabalho.

Figura 18 – Palheta de ovitrampa



Fonte: SILVA (2021)

4.3 CAPTURA DE IMAGENS

Dentre várias formas possíveis de captura de imagens a partir das palhetas de ovitrampa, foram utilizados um *smartphone* e um suporte de captura desenvolvido no âmbito deste trabalho, os quais são utilizados em conjunto e serão descritos logo abaixo.

4.3.1 Dispositivo móvel de captura

Para captura das imagens digitais foi utilizado um *smartphone* Samsung Galaxy A03s, lançado em 2021 e considerado um modelo de entrada. Possui um processador Octa-Core (4 x 2.3Ghz + 4 x 1.8Ghz), 3GB de memória RAM e 32GB de armazenamento interno (SAMSUNG, 2021b).

O dispositivo possui também uma câmera frontal de 5MP (não utilizada neste trabalho) e uma câmera traseira composta por três sensores: Um principal de 13MP, um de 2MP para captura macro e um sensor de profundidade de 2MP. Enquanto a câmera principal de 13MP é indicada para obter imagens grandes e com alta resolução, a câmera macro, por outro lado é de baixa resolução e é utilizada para objetos muito próximos da lente (cerca de 4 cm, segundo o fabricante) (BRASIL, 2021). Entretanto, foi utilizada a câmera principal de 13MP por fornecer melhor resolução. Além disso, a câmera traseira possui um Flash com luz de LED branca, como mostra a Figura 19.

4.3.1.1 Limitações da captura direta por smartphone

Em que pese a qualidade considerável na captura de imagens digitais com o dispositivo acima mencionado, foram observadas limitações que inviabilizaram o uso definitivo do *smartphone* por si só, isto é, sem instrumento auxiliar. Iluminação não uniforme, intersecção entre os segmentos da palheta na captura são os principais pontos limitadores desta abordagem sendo ela adotada sem acessórios ou suporte.

Figura 19 – Câmera traseira - Samsung A03S



Fonte: SAMSUNG (2021a)

Dado que o Flash de LED branco está posto lateralmente ao sensor principal de 13MP, conforme Figura 19, a imagem digital capturada da palheta, nesse caso, apresenta iluminação não uniforme, isto é, um lado da imagem recebia mais reflectância e isto produzia uma imagem com o lado esquerdo mais iluminado e o lado direito (especialmente a extremidade inferior) mais escuro. Como apontado por (GONZALEZ; WOODS, 2015), um ponto de uma imagem

digital é formado por luminância e reflectância, e neste caso, pelo fato de a luminância ser maior no lado esquerdo, a luz não era uniformemente distribuída nas regiões do segmento de uma palheta, como observado na Figura 20, o que, por conseguinte compromete o resultado dos processamentos posteriores.

Figura 20 – Segmento de palheta com iluminação não uniforme

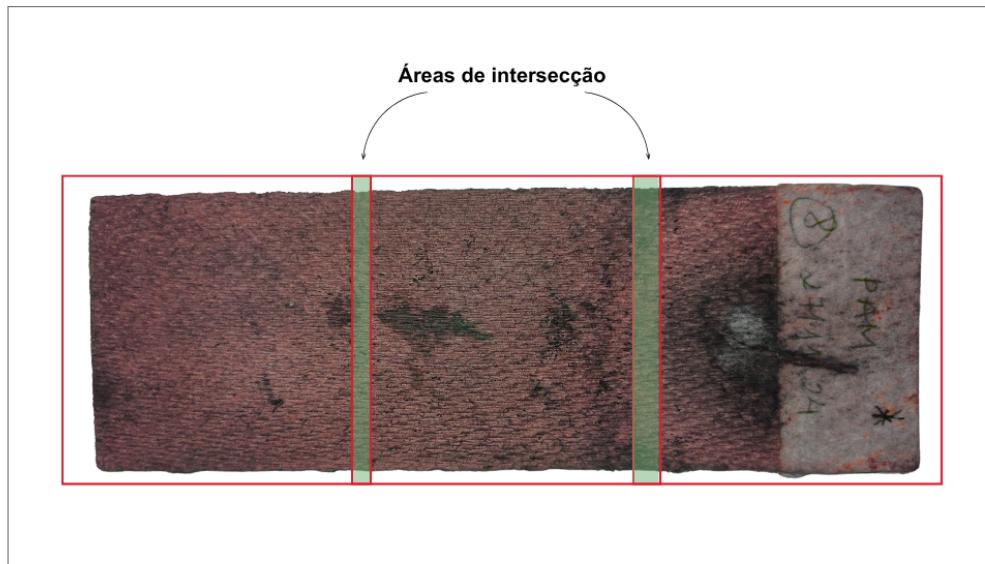


Fonte: Figura do autor (2025)

Ademais, outro ponto limitador da abordagem de captura manual dos segmentos de uma palheta, consiste na intersecção de segmentos da imagem, ou seja, ao capturar imagens dos três segmentos de uma palheta (base, centro e topo), sem o auxílio de um suporte, por consequência haverão regiões comuns ou de intersecção por não haver um delimitador físico para cada segmento, como mostra a Figura 21. Tal inconsistência levaria a uma contagem futura dúbia do número de ovos numa palheta em que os objetos de interesse seriam considerados duas

vezes.

Figura 21 – Divisão de segmentos com áreas comuns



Fonte: Figura do autor (2025)

Por fim, outros problemas tais como a falta de uma distância fixa pré-estabelecida entre a câmera do *smartphone* para a superfície da palheta também produziriam imagens instáveis. Afora outros fatores, tais como inclinação (ainda que leve) da câmera em relação à superfície também resultariam em imagens com condições diversas entre si, logo, despadronizadas. Tais razões suscitaram o desenvolvimento de um Suporte de Captura, com o intuito de sanar os principais problemas da captura puramente manual e desestruturada.

4.3.2 Suporte de captura desenvolvido

Dada a necessidade de sanar os problemas de despadronização na captura das fotos de uma palheta, foi desenvolvido no âmbito deste projeto, um Suporte de Captura (ou Dispositivo de Captura), criado com impressora 3D, com o objetivo de ser uma estrutura fixa de sustentação para a câmera do *smartphone*, bem como um provedor de iluminação padronizada, sanando assim, os problemas supracitados.

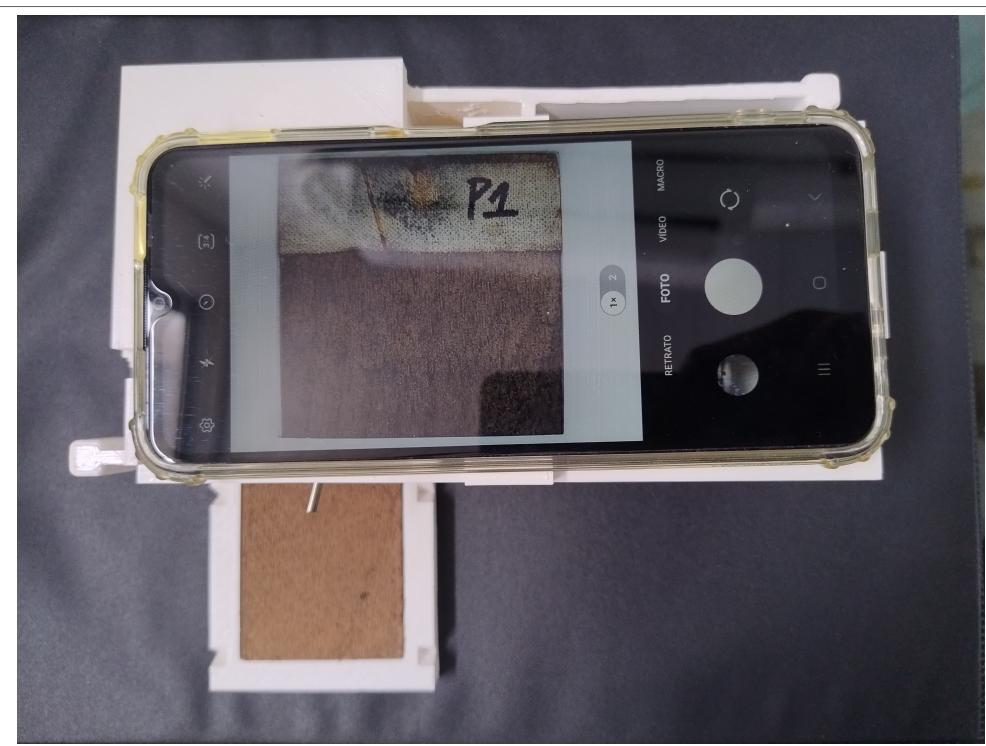
Este Dispositivo de Captura mede 18 cm de comprimento, 10 cm de largura e 7 cm de altura. É composto por duas partes: A plataforma inferior de captura, sobre a qual fica a palheta e o suporte superior de captura, onde apoia-se o celular. O suporte de captura, visto de cima, é exibido na Figura 22 e Figura 23.

Figura 22 – Suporte de captura de imagens de palheta de ovitrampa



Fonte: Figura do autor (2025)

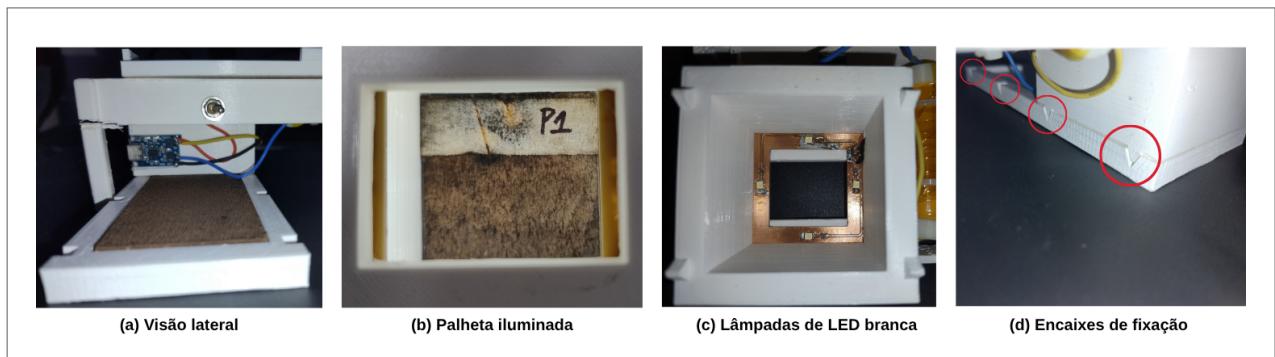
Figura 23 – Suporte de captura de imagens de palheta de ovitrampa com smartphone posicionado



Fonte: Figura do autor (2025)

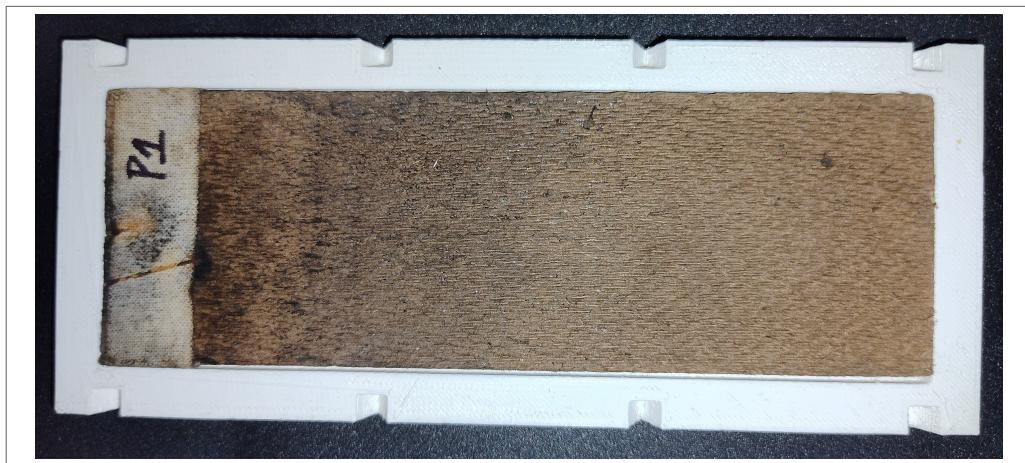
Assim sendo, a plataforma inferior de captura mede 17 cm x 5 cm, e sobre ela é colocada a palheta. Possui 8 ranhuras triangulares ao longo de sua superfície, as quais servem para encaixe do ressalto (saliência) da parte principal superior, conforme Figura 25. De modo semelhante, o suporte superior de captura possui as saliências para fixação com a plataforma inferior de captura, conforme Figura 24. Desta forma, esses encaixes de fixação mantém bem delimitados e sem intersecção os segmentos da palheta durante a captura das fotos.

Figura 24 – Características do Suporte de Captura



Fonte: Figura do autor (2025)

Figura 25 – Plataforma inferior do Suporte de Captura de imagens de palheta de ovitrampa



Fonte: Figura do autor (2025)

Ademais, o suporte superior possui acoplada a ele 4 pequenas lâmpadas de LED branca, alimentadas por uma bateria. Possui ainda, uma entrada USB-C para carregamento da bateria, além de uma pequena alavanca mecânica, para o operador humano controlar o acionamento e desligamento das lâmpadas.

Desta forma, ao utilizar o Suporte, o operador humano captura três fotos, uma para cada segmento da imagem: Base, centro e topo. Cada segmento é bem delimitado devido às ranhuras

e encaixes, o que evita a captura de áreas comuns. As lâmpadas de LED branca, direcionam-se da parte superior para a plataforma inferior, deste modo iluminando os quatro cantos de cada segmento da palheta. Pelo fato de a câmera estar fixa e imóvel numa distância de cerca de 6 cm da palheta, de modo a evitar o problema de variação de inclinação na captura.

Então, o Suporte de Captura, é usado para obter imagens digitais com excelente qualidade de iluminação, de maneira a sanar os fatores limitadores acima expostos, como iluminação irregular, intersecção de regiões e variações de inclinação. Além do mais, é um instrumento pequeno, prático e de fácil utilização por operadores humanos, o que torna mais segura e estabelecida esta etapa do processo como um todo de contagem de ovos de *Ae. aegypti* em palhetas de ovitrampa.

Nessa situação, o uso de sistemas de iluminação no contexto de captura de imagens de palhetas de ovitrampa com ovos de *Aedes aegypti* são também adotados em sistemas de capturas de ovos do mosquito, como em (SILVA; RODRIGUES; ARAUJO, 2012b), que menciona LED branca e (HAMESSE et al., 2023).

Ademais, no contexto de a metodologia de contagem de ovos aqui proposta, em especial, o dispositivo de captura ser de baixo custo, assevera-se que no contexto desta solução a diferença de qualidade e quantidade de componentes eletrônicos, como câmeras digitais e sistemas de iluminação, serve como premissa forte para concluir que o custo final do dispositivo de captura proposto neste trabalho é menor do que o custo associado ao empregado em outros trabalhos da literatura.

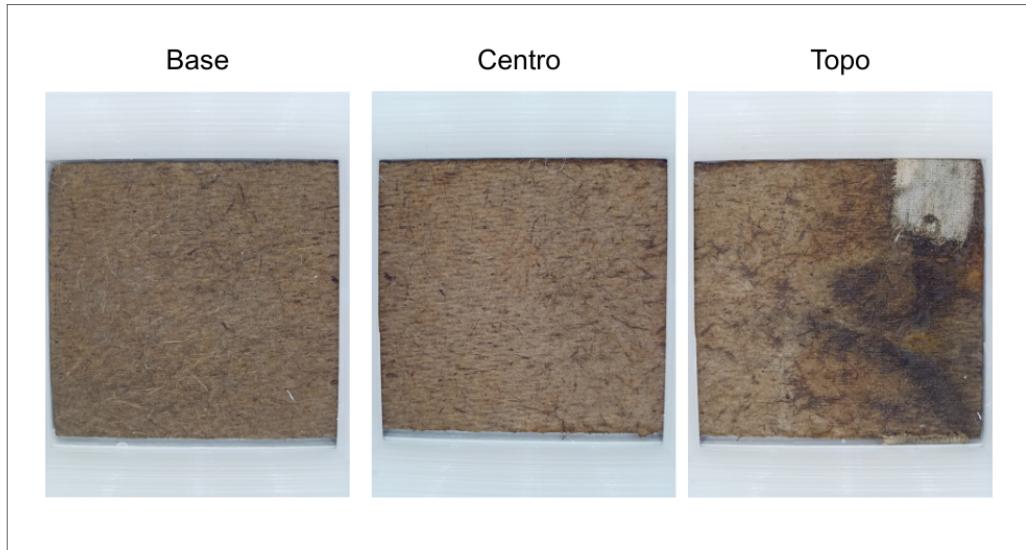
Nesse sentido, tem-se que, a câmera do *smartphone* Samsung Galaxy A03 adotada neste trabalho, em comparação com a câmera “Olympus Tough TG-6 e microscópio Nikon”, utilizado como dispositivo de captura em (JAVED et al., 2023), bem como o molde feito com impressora 3D (deste trabalho), comparado com sistemas mais complexos de captura, como o “Dispositivo de hardware com microscópio DinoLite AM4013MZT”, empregado por (NARANJO-ALCAZAR et al., 2024).

Por fim, para cada palheta, as três imagens resultantes podem ser exemplificadas na Figura 26.

4.3.3 Geração de *grids* (*crops*)

Durante testes iniciais, dado um segmento de uma palheta (por exemplo, como visto na Figura 20), verificou-se que ao aplicar *zoom* nestas imagens, os ovos do mosquito começam

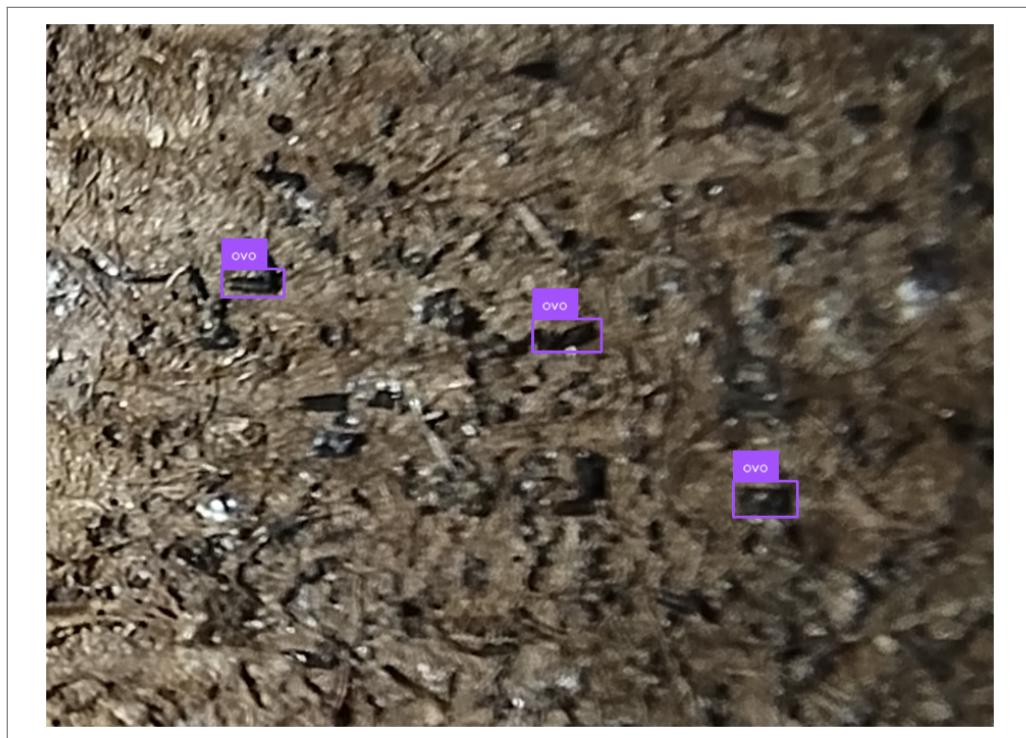
Figura 26 – Imagens resultantes após uso do Suporte de Captura



Fonte: Figura do autor (2025)

a tornar-se mais claramente visíveis na superfície da palheta (como mostra a Figura 27), bem como também é possível a um modelo treinado reconhecer os ovos na imagem.

Figura 27 – Exemplo de detecção de ovo após aplicação de zoom



Fonte: Figura do autor (2025)

Com isso, visando a cobertura de toda a superfície da palheta, realizou-se o seguinte processamento descrito no Algoritmo 1 para geração de *crops* para cada segmento da palheta.

Algoritmo 1: Geração de crops a partir de uma palheta

Input: Conjunto de segmentos S de tamanho 3 da palheta Px .

Output: Um conjunto de recortes de imagem (*crops*), com C *crops* por segmento s de entrada, totalizando $C \times 3$ *crops*.

foreach $segmento s \in S$ **do**

Dividir o segmento s em um grid de tamanho 7×6 , resultando em $C = 42$ *crops*;

for $i \leftarrow 0$ **to** $C - 1$ **do**

Salvar o crop i como arquivo: $<Px>_<s>_grid_<i>.$;

Exemplo: $P1_base_grid_0, P1_base_grid_1, \dots, P1_base_grid_41.$;

end

end

Dada uma palheta como entrada, ao utilizar-se o Dispositivo de Captura desenvolvido, obtém-se a partir dela os segmentos base, centro e topo. Cada segmento possui dimensão de 4160×3120 *pixels*. A partir destes segmentos passa-se a geração dos *crops*.

Este próximo processamento é uma operação em que a imagem é dividida em pequenas porções de imagem, chamadas de *crop*. Adotou-se o valor de 7 linhas e 6 colunas e dividiu-se a imagem como uma matriz; isto resulta em 42 imagens, cada qual com de 491×431 px. Esta etapa do processamento é expressa na Figura 28.

Figura 28 – Processo de geração de *crops* a partir de um segmento de palheta



Fonte: Figura do autor (2025)

Nesse contexto, é sabido que há um *thresholding* quanto à quantidade de linhas e colunas escolhida. Inicialmente, por natureza, sabe-se que um segmento de imagem possui ovos

excessivamente pequenos e até dificilmente vistos por um humano ou reconhecidos por um modelo. Assim, ao cortar essa imagem em outras micro imagens, não se deve aumentar em excesso a quantidade de cortes, sob pena de cortar os objetos de interesse em partes ou de se obter imagens excessivamente pequenas.

Deste modo, testou-se vários valores de linha e coluna, tais como 4×4 , 5×5 e 7×6 . Nos testes iniciais, cada um deles deu origem a imagens resultantes com resoluções de, respectivamente, 611×815 , 489×652 e 407×466 pixels.

Adicionalmente, constatou-se que a qualidade das detecções aumenta assim que a quantidade de *crops* aumenta. Ou seja, quanto mais *crops* de imagem, obtém-se mais detecções de qualidade, como mostra a Figura 27. A ressalva consiste em não aumentar muito o número de *crops*, haja vista que, ao se aumentar o número de *crops*, a dimensão das imagens finais diminui.

Figura 29 – Exemplo de *crop* obtido



Fonte: Figura do autor (2025)

Portanto, o dispositivo de captura obtém, ao fim do processo 126 imagens, que representam em conjunto, a totalidade da superfície de uma palheta completa. Cada imagem, a exemplo da Figura 29, possui 491×431 de dimensão, com cores especificadas no sistema de cores RGB e

comprimidas sob o formato de arquivo Joint Photographic Experts Group (JPG) (GONZALEZ; WOODS, 2015).

4.3.4 Base de imagens

A base de dados utilizada, disponibilizada por (SILVA, 2021) foi utilizada neste trabalho. Esta base foi feita a partir de 100 palhetas, capturada utilizando câmera digital e cada imagem possui $680 \times 480 \text{ px}$ de dimensão. É formada por 1.162 imagens originais e, consideradas as obtidas, por *data augmentation*, possui 3.703 imagens.

4.3.5 Base de dados criada

No âmbito deste trabalho, com Dispositivo de Captura desenvolvido com impressora 3D, foi criada uma base de imagens de ovos de *Ae. aegypti* a partir de 124 palhetas de ovitrampa disponibilizadas pela FIOCRUZ-PE, capturadas com o *smartphone* acima mencionado. Ao seguir-se os passos descritos nas seções anteriores, foi criada uma base própria de imagens que formada por 2.340 imagens e com o total de 8.745 ovos do mosquito. Cada imagem possui $491 \times 430 \text{ px}$, aproximadamente.

Por consequência, foi feita a divisão deste conjunto de dados com vistas ao treinamento de modelos de detecção de objetos voltados ao reconhecimento de ovos de *Ae. aegypti*. A divisão das imagens obedece ao exibido na Tabela 3.

Tabela 3 – Divisão do dataset de ovos de *Aedes aegypti* em palhetas de ovitrampa

| Partição | Quantidade de Imagens | Quantidade de Ovos |
|--------------------|-----------------------|--------------------|
| Treinamento | 1.872 | 6.970 |
| Validação | 234 | 922 |
| Teste | 234 | 853 |
| Total | 2.340 | 8.745 |

Fonte: Elaborada pelo autor (2025)

No âmbito de modelos de Aprendizagem Profunda, a obtenção, rotulação e demais tratamentos dos dados constituem-se em parte fundamental do fluxo de criação de um modelo, além de exigir tempo e esforço humanos consideráveis (GOODFELLOW; BENGIO; COURVILLE, 2016).

Portanto, este conjunto de dados criado é uma contribuição importante para o refinamento posterior de modelos de detecção de objetos voltados ao domínio deste trabalho.

4.3.6 Base de dados derivada

Com vista a aumentar o conjunto de dados, aproveitou-se a base de dados proposta por (SILVA, 2021). Esta base é formada por 1.162 imagens de ovos de *Ae. aegypti* em palhetas de ovitrampa; cada imagem possui 640 x 480 px. As condições de captura, porém, distinguem-se, pois o dispositivo de captura é uma câmera digital (em vez de um *smartphone*) e o suporte de captura é bem maior e de funcionamento distinto.

Estas imagens apresentam uma qualidade visual bem melhor, em que pese haver imagens ruidosas, dada a natureza do problema. A Figura 30 mostra exemplos de imagens desta base em que fica à mostra a diferença na qualidade das imagens.

Figura 30 – Exemplo de imagens da base criada por (SILVA, 2021)



Fonte: Elaborada pelo autor (2025)

Visando aumentar o conjunto de treinamento, bem como a variedade de condições de iluminação, cor, brilho em que os ovos estão presentes na imagem e até, fez-se o redimensionamento das imagens desta referida base e as imagens foram agregadas ao dataset de imagens capturadas por celular criado neste trabalho. Assim, as imagens da base obtidas por câmera digital, foram redimensionadas para um tamanho próximo o suficiente, de modo que o tamanho dos objetos de interesse ficasse próximo do que são na base captura por *smartphone*. Assim, as imagens foram redimensionadas de 640 x 480 para aproximadamente 491 x 430 px.

Por consequência, a partir disto, obteve-se um conjunto de imagens ainda maior, com mais exemplos de treinamento e objetos com mais variações de condições. Este *dataset* está assim dividido, como mostra a Tabela 4.

Tabela 4 – Divisão das imagens do dataset derivado

| Partição | Quantidade de Imagens | Quantidade de Ovos |
|--------------------|-----------------------|--------------------|
| Treinamento | 2.799 | 11.113 |
| Validação | 349 | 1.397 |
| Teste | 351 | 1.326 |
| Total | 3.499 | 13.836 |

Fonte: Elaborada pelo autor (2025)

4.3.7 Conjunto de palhetas contadas

Ainda, foi obtido pelo projeto um conjunto de 28 palhetas de ovitrampa contadas pelos técnicos da FIOCRUZ-PE. Esse conjunto de imagens totaliza 3.528 imagens e totaliza 8.035 ovos. Esse conjunto de palhetas foi capturado com o Dispositivo de Captura baseado em impressora 3D desenvolvido neste trabalho. Tal grupo de palhetas foi rotulado e também contado pelo autor utilizando-se da ferramenta de rotulação *LabelImg* (TZUTALIN, 2015). Esta rotulação é necessária para obter-se a mAP.

Essas palhetas constituem-se numa coleção importante (e mais atual) para servir como um conjunto de dados *Real World* (do mundo real) e assim serem usadas para, em última análise, verificar o desempenho dos modelos de contagem automática desenvolvidos e comparar com a contagem humana manual.

4.3.7.1 Desafios à contagem automática

A contagem automática de ovos de *Ae. aegypti* abrange alguns desafios: Quanto à qualidade das imagens capturadas e à disposição dos ovos na palheta. Nesse sentido, ruídos são bastante comuns nessas imagens, o que requer robustez por parte dos métodos de contagem, a fim de lidar com esses problemas. Observa-se na Figura 31, uma imagem com iluminação irregular, que é um dos ruídos comuns nas imagens. Somado a isso, problemas na disposição dos ovos, como, quando os ovos estão sobrepostos, adjuntos, além de estarem segmentados;

ademas, ruídos gerais existentes nas imagens, que são característicos do uso de ovitrampas, como objetos estranhos, também são obstáculos para a contagem automática.

Figura 31 – Exemplo de imagem ruidosa de palheta, especialmente com iluminação irregular



Fonte: Elaborada pelo autor (2025)

4.4 MÉTODOS DE CONTAGEM AUTOMÁTICA COM APRENDIZAGEM PROFUNDA

Conforme o fluxo descrito na Figura 17, adotou-se o uso de diversas abordagens para a contagem automática de ovos de *Ae. aegypti* em palhetas de ovitrampa, cada qual com suas operações intermediárias. As seções seguintes detalham cada fluxo alternativo para contagem automática.

4.4.1 Ambiente de treinamento

O processo de treinamento dos modelos foi realizado no *Google Colab Pro+*, o qual fornece um ambiente virtual dotado de 83.5 GB de memória RAM, além de uma GPU de NVIDIA A100-SXM4-40GB, com memória RAM de 40 GB.

4.5 FLUXO I: CONTAGEM ALIADA À APLICAÇÃO DE MODELOS DE RESTAURAÇÃO DE IMAGENS BASEADOS EM APRENDIZAGEM PROFUNDA

Neste fluxo, utilizou-se Aprendizagem Profunda voltada a restauração de imagens, haja vista os desafios à contagem automática, em especial a presença de tipos de ruídos na base de dados de ovos de *Ae. aegypti*, com os quais um modelo de restauração poderia lidar.

Assim, neste fluxo, um modelo de melhoria foi aplicado como etapa intermediária, antes da realização da inferência pelo modelo de contagem automática.

4.5.1 MPRNet

Como define (GONZALEZ; WOODS, 2015), a Restauração de Imagens, de modo geral, distingue-se do Ralce de Imagens, por ser este um processo objetivo, ao contrário daquele (realce), que é subjetivo e baseado na percepção do observador humano. Ademais, a restauração de imagens visa recuperar uma imagem corrompida em cenários onde se tem o conhecimento de qual foi o fenômeno de degradação que causou o ruído e, a partir daí prover a aplicação do processo contrário, para, assim, a partir de uma imagem degradada, obter sua versão dita limpa. Assim, ruídos como borramento, ruídos de chuva e ruídos em geral podem ser tratados por esta técnica.

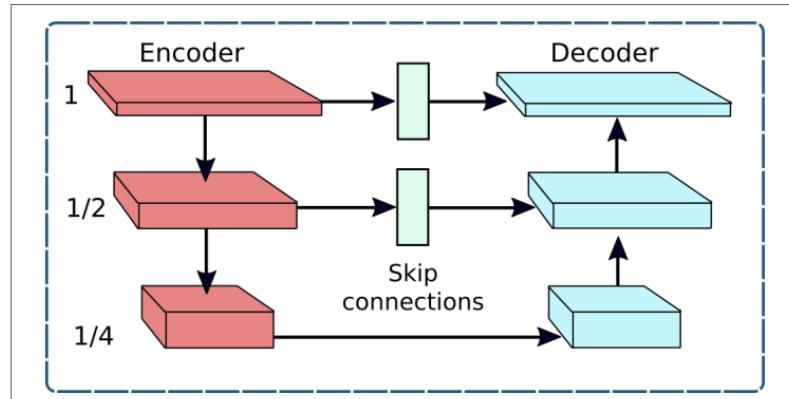
Então, dada a tarefa de lidar com imagens de baixa qualidade e com presença de ruídos específicos, optou-se por um tratamento com Redes Neurais de restauração de imagens, uma vez que são amplamente aplicados atualmente, como, tal qual exposto por (ARCHANA; JEEVARAJ, 2024).

Assim, utilizou-se o modelo de restauração MPRNet, proposto por (ZAMIR et al., 2021), que, no contexto de restauração de imagens, busca em sua transformação, equilibrar tanto detalhes espaciais, como também informações de alto nível nas imagens, de modo a buscar manter uma transformação final que apresente nitidez nos detalhes, como texturas e bordas e ainda nas informações de alto nível.

O MPRNet provê de arquitetura que lida com a Restauração de Imagens de três tipos de ruídos: *Image Deblurring*, *Image Derraining* e *Image Denoising* (RAJAEI; RAJAEI; DAMAVANDI, 2023). Sua arquitetura de três estágios, como mostra a Figura 16, restaura gradativamente a imagem ao longo desses estágios e faz, nesse processo, uso de *skip connections* entre eles, tanto internamente, quanto entre os estágios. Quanto aos estágios, assim divide-se: Nos dois

primeiros, uma estrutura codificador-decodificador (ou *encoder-decoder*) é implementada com base no padrão de uma rede U-Net (RONNEBERGER; FISCHER; BROX, 2015), tal qual exibe a Figura 32 com o intuito de aprender informações contextuais completas da imagem degradada.

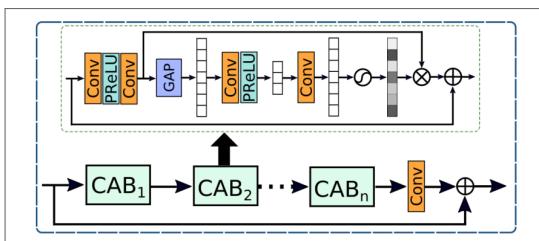
Figura 32 – Sub-rede *encoder-decoder*



Fonte: (ZAMIR et al., 2021)

Por conseguinte, no terceiro estágio, uma ORSNet opera também na imagem original, mais precisamente em sua resolução, para gerar saídas espacialmente precisas e obter-se uma imagem de saída com correspondência espacial *pixel* a *pixel* em relação à imagem original, como exibe a Figura 33.

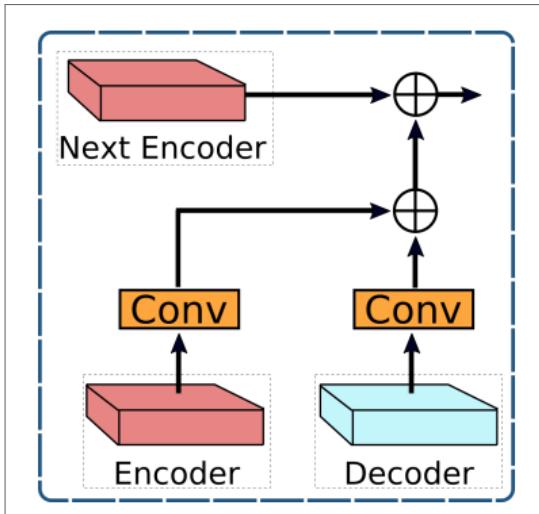
Figura 33 – Bloco ORB da sub-rede ORSNet



Fonte: (ZAMIR et al., 2021)

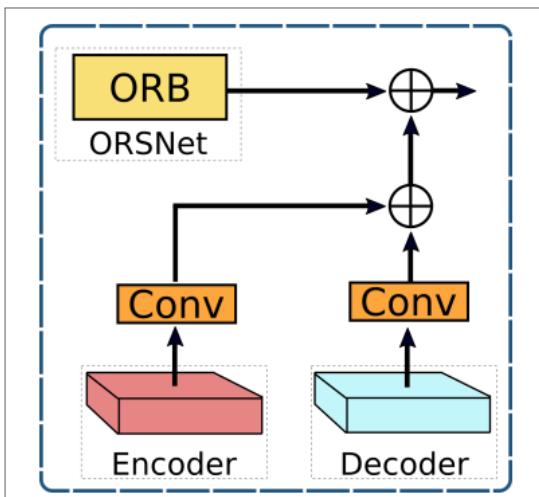
Entre os estágios 1 e 2, em especial entre os dois *encoder-decoder*, há um módulo denominado CSFF, como aponta a Figura 34. Já entre os estágios 2 e 3, entre os *encoder-decoder* e ORSNet, de igual modo está o módulo CSFF, conforme Figura 35. Devido ao uso repetido de operações de *up-sampling* e *down-sampling*, pode ocorrer perda de informação. Nesse ponto, o CSFF torna a rede menos vulnerável à perda de informação. Acrescenta-se que, o uso do CSFF, permite com que *features* multi-escala oriundas de um estágio auxiliem a enriquecer as *features* do estágio subsequente. Por fim, à medida que o CSFF facilita o fluxo de informação, o processo de otimização da rede se torna mais estável, o que permite com que a arquitetura da rede seja aumentada, de modo a adicionarem-se mais estágios.

Figura 34 – CSFF entre os estágios 1 e 2



Fonte: (ZAMIR et al., 2021)

Figura 35 – CSFF entre os estágios 2 e 2

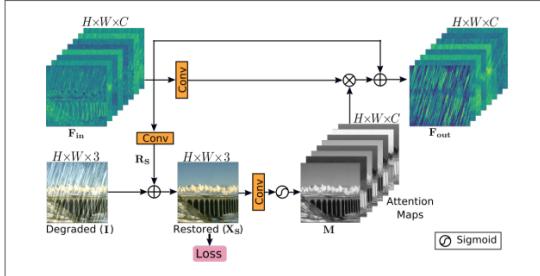


Fonte: (ZAMIR et al., 2021)

Ao contrário de outras redes multi-estágio de restauração de imagens, tais como, (SUIN; PUROHIT; RAJAGOPALAN, 2020; ZHANG et al., 2019), que fazem a predição numa imagem, para então passá-la para o próximo estágio, a MPRNet, usa SAM como módulo de *attention* entre os dois estágios, o que facilita o ganho de performance significativa (ZAMIR et al., 2021). Por a restauração ser progressiva entre os estágios, o SAM, exibido na Figura 36, é útil por fornecer sinais de supervisão de *ground-truth* ao longo deste processo. Ainda, previsões supervisionadas localmente são usadas também para gerar mapas de *attention* e, assim, suprimir *features* menos significativas, de maneira a passar para o próximo estágio somente informações úteis.

No contexto de imagens de ovos de *Ae. aegypti* em palhetas de ovitrampa, em que há, ruídos diversos abundantes, considerou-se adotar o MPRNet dada sua arquitetura de restauração

Figura 36 – Módulo SAM



Fonte: (ZAMIR et al., 2021)

progressiva de imagem adotar mecanismos e estratégias para tentar evitar a propagação de ruídos para a imagem de saída. Portanto, o uso de diversos módulos com estratégias pertinentes, aliadas a mecanismos de atenção, como acima citado (entre outros), tornam a arquitetura do MPRNet propícia para restauração de imagens no âmbito deste trabalho.

Ainda, o MPRNet, em 10 *datasets* de *benchmark*, nas suas três tarefas (*Image Deblurring*, *Image Deraining* e *Image Denoising*) obtém desempenho superior em relação a outros modelos com os quais é comparado. Além disso, caracteriza-se por ser eficiente em questão de tamanho e em tempo de execução, o que favorece até sua aplicação em dispositivos com recursos limitados (ZAMIR et al., 2021).

Destarte, este modelo é pré-treinado no conjunto de dados GoPro (NAH; KIM; LEE, 2017), que possui 2.103 pares de imagens na partição de treinamento e 1.111, na de validação. Ainda, o modelo teve o desempenho verificado na partição de teste do *dataset* HIDE (SHEN et al., 2019) e RealBlur (RIM et al., 2020; ZAMIR et al., 2021).

Então, como acima mencionado, o MPRNet é disponibilizado para três distintas tarefas de restauração de imagens: *Image Deblurring*, *Image Deraining* e *Image Denoising*. Aquela (*Deblurring*) é voltada a restauração de ruído causado por falta de foco (*Defocus blur*), movimento do sensor ou do objeto (*Motion blur*) ou por limitações óticas. Essa, a tarefa de *Image Derraining*, por sua vez visa remover traços de chuva presentes na imagem, mantendo sua nitidez. Por fim, o *Image Denoising*, consiste em remover ruídos de modo geral da imagem, como o ruído gaussiano e ainda assim manter detalhes de alta qualidade na imagem (ELAD; KAWAR; VAKSMAN, 2023).

Nesse contexto, observa-se na base de dados de imagens de palhetas de ovitrampa capturadas por *smartphone*, a presença de numerosa de ruído de borramento (ou *Blur*). A propósito, a Figura 37, exibe a presença de ruídos do tipo *blur*. Logo, ao utilizar o modelo optou-se por um tratamento específico com a tarefa de remoção de ruído de borramento: *Image Deblurring*.

Figura 37 – Imagens com a presença de ruído de borramento



Fonte: Figura do autor (2025)

4.6 FLUXO II: CONTAGEM ALIDADA A TÉCNICAS DE PROCESSAMENTO DE IMAGENS

4.6.1 Aplicação inicial de técnicas de realce de imagens

Como define (GONZALEZ; WOODS, 2015), Realce de Imagens é o processo de manipulação de imagens para torná-las mais adequadas ao fim específico de uma aplicação em relação ao que elas inicialmente eram. Por natureza, estas técnicas são orientadas ao problema, isto é, o conjunto de técnicas aplicadas obedece ao domínio específico da aplicação ou sistema de Visão Computacional.

Assim, a partir de uma imagem original da superfície de uma palheta proveniente da base capturada por *smartphone*, foi aplicado *zoom*, a ponto de se visualizar bem os ovos e, em sequência, foram aplicadas diversas técnicas de Realce de Imagens, como *Color Jitter*,

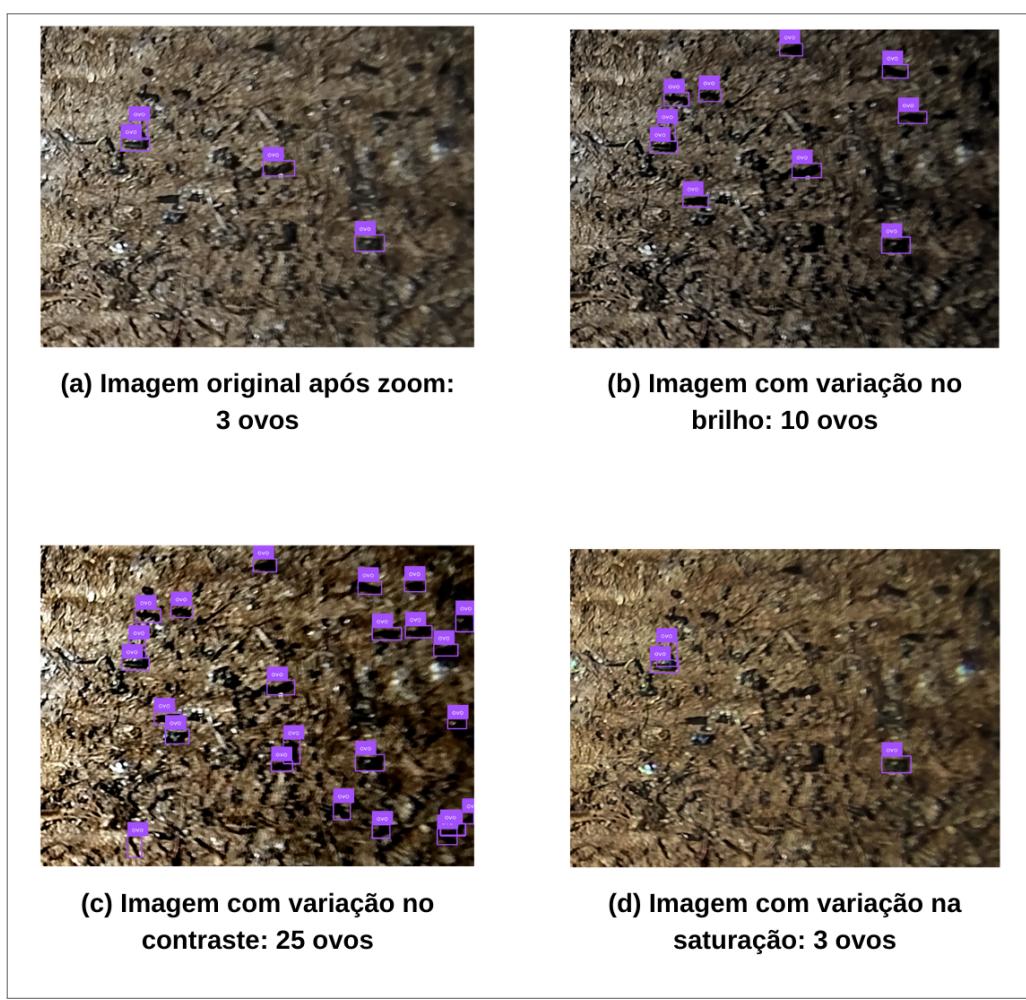
Equalização de Histograma e CLAHE.

Então, a partir daí, utilizou-se um modelo YOLOv10, treinado no âmbito deste trabalho na base proposta por (SILVA, 2021) e realizou-se a inferência em um grupo de imagens capturadas por *smartphone* para verificar o efeito da aplicação de técnicas de realce na contagem de ovos. Portanto, foram aplicadas as transformações conforme o que as subseções seguintes discriminam.

4.6.1.1 Color Jitter

É uma técnica comumente aplicada para *Data Augmentation* e consiste em gerar variações de cor em imagens, quanto à brilho, saturação, contraste e matiz (PYTORCH, 2025a). Dessa maneira, ao aplicar estas transformações, verificou-se que mais objetos de interesse eram reconhecidos pelo YOLOv10. Isto é expresso, a exemplo, na Figura 38.

Figura 38 – Imagens de palheta de ovitrampa após aplicação de variações de contraste, brilho e saturação



Fonte: Figura do autor (2025)

4.6.1.2 Equalização de Histograma

Dentre as várias transformações de Transformação de Histograma, adotou-se, a Equalização de Histograma. Esta transformação é útil para realçar imagens de baixo contraste e produz uma distribuição mais uniforme das intensidades dos *pixels* presentes na imagem (OPENCV, 2025b; MATHWORKS, 2025b).

Desta forma, como menciona a Figura 17, antes da inferência por um modelo de *Deep Learning*, foram aplicadas individualmente as Técnicas de Realce intermediárias com o fim de submeter ao modelo uma imagem já realçada e assim, verificar o desempenho da contagem automática. Esta operação é padrão e não recebe quaisquer parâmetros para ajuste, ao contrário da transformação seguinte.

4.6.1.3 Contrast Limited Adaptive Histogram Equalization (CLAHE)

Justo por que a Equalização de Histograma, que para fins de realce de contraste, considera o contraste global da imagem, o que nem sempre produz resultados satisfatórios, como no caso em que há tanto *pixels* muito escuros e muito claros.

Assim, CLAHE (Ou Equalização Adaptativa), em vez de considerar o contraste global, divide a imagem em regiões, chamadas tiles e considera o contraste de cada região e então, cada *tile* é equalizado normalmente. Ainda, para evitar que o ruído possa ser aumentado nas regiões, é aplicado um limitador de contraste, de modo que, se um *bin* da equalização de histograma ultrapassar este limite (*clipLimit*), os *pixels* excedentes serão distribuídos uniformemente ao longo do histograma. Por fim, uma Interpolação Bilinear é aplicada.

Portanto, foram utilizados como parâmetros, os seguintes valores: O tamanho do *tile* (*tileGridSize*) foi definido como 8x8 px e o limiar de corte, o *clipLimit* foi definido como 2.0 (OPENCV, 2025c). Estes valores de parâmetros produzem um efeito equilibrado na transformação, com um *tileGridSize* adequado ao tamanho das imagens de dimensão 491 x 431 px e com um *clipLimit* conservador com intuito de obter um bom equilíbrio de contraste, evitando transformações extremas.

4.7 FLUXO III: CONTAGEM DIRETA COM REDES NEURAIS

Entende-se como a utilização de um modelo de Aprendizagem Profunda para realizar inferência diretamente nas imagens sem que haja etapas intermediárias, como pré-processamento ou realce de imagens.

Então, já de posse da base de dados, procedeu-se ao treinamento dos modelos de Aprendizagem Profunda, como mencionado em seguida.

4.7.1 YOLOv10

Foi utilizado um modelo YOLOv10, proposto por (WANG et al., 2024). Este, conforme documentação oficial, possui 6 tamanhos disponibilizados, divididos por quantidade de parâmetros e complexidade. Dado que a base de dados utilizada neste trabalho não é suficientemente grande, como se exige para modelos de Aprendizagem Aprendizagem como este, optou-se por utilizar o YOLOv10-M.

Este modelo é disponibilizado já pré-treinado no *dataset* COCO 2017, que é composto por 80 classes genéricas, como pessoa, carro, maçã, laranja etc. O conjunto possui cerca de 118 mil imagens para treino e 5 mil para validação, além de cerca de 40 mil imagens de teste.

Considerando a literatura, optou-se por utilizar este modelo pré-treinado em COCO 2017 e aplicar *Transfer Learning* para obter vantagens desta técnica e obter um desempenho melhor, considerando o amplo conjunto de características gerais extraídas pelo modelo dado seu extenso número de 80 classes.

Quanto aos hiper-parâmetros, o modelo foi treinado como descreve a Tabela 5. O modelo foi treinado por 100 épocas, com *batch-size* de 32 e o otimizador escolhido foi o AdamW, além de ter sido aplicada uma regularização L2 (conhecida como *Ridge Regression* com valor de 0.0005 para ajudar a controlar o *overfitting*.

Assim, a partir desse YOLOv10m pré-treinado em COCO 2017, o treinamento foi feito na base de dados de treinamento proposta por (SILVA, 2021) a uma taxa de aprendizagem de 0.003, o que é uma taxa baixa e apropriada para aplicar *Fine-tuning*, sem sobreescrivar excessivamente os valores dos pesos e desperdiçar o aprendizado anterior e mais geral do modelo e que também é recomendado na documentação oficial para o ajuste fino com *datasets* personalizados. Além disso, foi utilizado um *Batch size* de 32 para dar maior estabilidade ao treinamento. O parâmetro *imgsz* com valor 640 refere-se ao tamanho para o qual as imagens

de entrada são redimensionadas antes de serem submetidas ao modelo durante o treino e geralmente, quanto maior for, pode produzir resultados melhores.

Tabela 5 – Principais hiperparâmetros utilizados no treinamento do YOLOv10-M

| Hiperparâmetro | Valor |
|----------------------|--------|
| Épocas | 100 |
| Batch size | 32 |
| Learning rate | 0,003 |
| Otimizador | AdamW |
| Weight decay | 0,0005 |
| imgsz | 640 |

Fonte: Elaborada pelo autor (2025)

Além disso, como recomendado pela literatura para evitar *overfitting*, foram aplicadas algumas técnicas de *Data Augmentation*, conforme exibido na Tabela 6. Estas técnicas aplicadas visando fornecer ao modelos dados em mais variadas condições de posicionamento (como objetivam as transformações como *Flip Horizontal*) ou de condições de iluminação (a exemplo da transformação de Saturação), de tamanho (como usado com a transformação de *Scale*). Isto, por fim, ajuda o modelo a ser mais tolerante a essas variações e generalizar melhor (GÉRON, 2019).

Tabela 6 – Técnicas de *Data Augmentation* aplicadas no treinamento do YOLOv10-M

| Técnica | Valor |
|------------------------|-------------|
| Flip horizontal | 0,5 |
| Mosaic | 0,5 |
| HSV Saturation | 0,7 |
| AutoAugment | RandAugment |
| Erasing | 0,3 |
| Translate | 0,1 |
| Scale | 0,3 |

Fonte: Elaborada pelo autor (2025)

Como apontado, foi utilizado *horizontal flip* com valor de 0.5, o qual inverte horizontalmente 50% das imagens, além de *mosaic* com valor 0.5, que monta 4 imagens numa só para auxiliar na generalização e é aplicado, nesse caso, sobre 70% das imagens (ULTRALYTICS, 2025e).

Ainda, foi aplicada um ajuste de saturação (*HSV Saturation*) que modifica a intensidade da imagem aleatoriamente numa faixa de -0.7 a 0.7. Inclusive, *AutoAugment* com valor *RandAug-*

ment, o qual aplica uma seleção aleatória de transformações. Também foi utilizado *Erasing*, que, por sua vez apaga partes da imagem para simular oclusão de objetos, que é um problema importante de ser tratado em virtude dos desafios inerentes à contagem automática (nesse caso, objetos sobrepostos).

Também, foi aplicada um *Translate* leve de 0.1, o qual desloca as imagens horizontal e verticalmente de acordo com esta magnitude definida. Por fim, foi empregada uma transformação *scale* de 0.3, que redimensiona as imagens dentro desse valor definido como margem (para mais ou para menos). Ela é útil para simular objetos a diferentes distâncias: mais perto ou mais longe.

Observa-se, portanto, que a definição dos parâmetros supracitados, obedeceu às seguintes condições: Proporcionar aumento de dados, produzir dados sintéticos em situações mais diversas, bem como, não causar ruído expressivo nos dados, ou mesmo amplificar ruídos já existentes numa imagem.

4.7.2 YOLOv12

Considerando o modelo proposto em (TIAN; YE; DOERMANN, 2025) apresentar bons desempenhos superiores ao YOLOv10, adotou-se o YOLOv12 em sua versão YOLOv12-M, que conta com 20.2 milhões de parâmetros e AP50:95 de 52.5, que é levemente superior ao anterior (51.1); isso medido no conjunto de validação do *dataset* COCO 2017.

Dado este modelo base YOLOv12, pré-treinado na base de treinamento COCO 2017 e visto que ele possui 5 tamanhos de escala, utilizou-se o modelo intermediário, o YOLOv12-M, dado sua referida quantidade de parâmetros e haja vista que os de maior escala, como YOLOv12-L ou YOLOv12-X são mais complexos e possuem uma quantidade consideravelmente maior de parâmetros, respectivamente 26.5 e 59.3 milhões de parâmetros. Por conseguinte, dado que o tamanho do dataset utilizado neste trabalho ser considerado pequeno, isto poderia levar a *overfitting*, já que estes modelos maiores escala são mais recomendados quando se tem uma quantidade bem maior e mais variada de dados de treinamento.

Nesse sentido, foi feito *fine-tuning* a partir do modelo base pré-treinado, com os hiperparâmetros definidos conforme a Tabela 7.

A quantidade de épocas foi mantida em 100, mesmo o *dataset* sendo pequeno, por conta do *patience* 15, que interromperia o treinamento. Além disso, ao fim do treinamento, são obtidos os pesos do modelo com o melhor desempenho e os últimos pesos; ainda, o trei-

Tabela 7 – Hiperparâmetros utilizados no treinamento do modelo YOLOv12-M

| Hiperparâmetro | Valor |
|----------------------|-------|
| Épocas | 100 |
| Batch size | 32 |
| Learning rate | 0,002 |
| Otimizador | AdamW |
| Dropout | 0.35 |
| Weight decay | 0,005 |
| Patience | 15 |
| imgsz | 1024 |

Fonte: Elaborada pelo autor (2025)

namento foi monitorado através das métricas como mAP@0.50, por exemplo, para obter os últimos pesos associados a uma boa performance, principalmente nas primeiras dezenas de épocas para evitar *overfitting*. O parâmetro *imgsz* foi mantido com valor de 640, pois, mesmo que aumentando-o, o desempenho do modelo poderia melhorar, entretanto, este parâmetro influencia significativamente no custo computacional envolvido no treinamento, em especial a memória RAM da GPU do *Google Colab Pro+*. Deste modo, verificou-se em experimentos que, ao aumentar *imgsz* para 1024, por exemplo, o tamanho do *batch* precisaria ser diminuído significativamente. Então, decidiu-se por manter o *batch size* 32 para dar mais estabilidade ao treinamento.

Ademais, o hiperparâmetro *patience*, referido acima, está associado à técnica de *Early Stopping* e produz o seguinte efeito: O treinamento é interrompido quando, mesmo após 15 épocas consecutivas, o modelo não apresenta melhoria nas métricas de validação; isto é útil para evitar *overfitting* (ULTRALYTICS, 2025d).

Também, dada a ocorrência de *overfitting* em experimentos intermediários, optou-se também por utilizar uma regularização mais forte com *dropout* de 0.35. O *Weight decay* com valor 0,0005 é um valor padrão e apropriado para uma regularização leve. Dado o valor mais forte do *dropout*, optou-se por não penalizar muito o modelo para não incorrer em *underfitting*, como ocorreu em experimentações anteriores deste modelo. Junto às outras técnicas de regularização como L2 e *Weight decay* utilizadas, servem como mecanismo para controlar o *overfitting*.

Ainda, visando tornar o modelo mais resiliente a variações nas características dos dados de palhetas de ovitrampa presentes no mundo real, como variações de iluminação de posiciona-

mento, rotação e outras variações, foi aplicado um conjunto de técnicas de *Data Augmentation*, como mostra a Tabela 8.

Tabela 8 – Técnicas de *Data Augmentation* aplicadas no treinamento do YOLOv12-M

| Técnica | Valor |
|------------------------|-------------|
| HSV Hue | 0,015 |
| HSV Saturation | 0,7 |
| HSV Value | 0,4 |
| Translate | 0,1 |
| Scale | 0,5 |
| Flip horizontal | 0,5 |
| Mosaic | 0,5 |
| AutoAugment | RandAugment |
| Erasing | 0,4 |

Fonte: Elaborada pelo autor (2025)

O *HSV Hue* altera as cores da imagem na faixa de intervalo especificada (nesse caso -0.15 a 0.15) e é útil para simular mais condições de cor de um objeto de interesse. Doutra forma, o *HSV Value* opera ajustes no brilho da imagem, de acordo com o valor fornecido. É importante para cenários em que o objeto de interesse aparece mais claro ou mais escuro a depender da luz do ambiente.

4.7.3 RF-DETR

Considerou-se o modelo RF-DETR, mencionado em (SAPKOTA et al., 2025), o qual apresenta desempenho concorrente ao do YOLOv12, de modo a superá-lo com mAP@50 de 0.9464 num cenário de classe única, bem como também, em vários cenários multiclasse. Dado principalmente seu bom desempenho em tarefas complexas, com oclusão de objetos (que é caso comum em ovos de *Ae. aegypti* em palhetas de ovitrampa, bem como sua rápida convergência, especialmente em problemas de classe única, de modo o treinamento do modelo alcançou platô em 10 épocas. Por estas razões, adotou-se este modelo para o problema atual. Ainda, o RF-DETR é mais voltado a cenários onde é mais importante a acurácia em cenários críticos (como *backgrounds* complexos) do que a velocidade de inferência, por exemplo, em cenários de detecção em tempo real, para o qual o YOLOv12 é mais recomendado.

Dado que este modelo possui duas versões de tamanhos disponíveis, quais sejam: RF-DETR-base (ou RF-DETR-B), com 29 milhões de parâmetros e RF-DETR-large, com 128

milhões de parâmetros, foi adotado o modelo com menos parâmetros, considerando o cenário atual deste trabalho.

O modelo é pré-treinado no *dataset* ImageNet-1K, que é composto por 1000 classes de objetos variados, como cachorro, gato, avisão, morango, dentre outros. É composto por cerca de 1.2 milhão de imagens de treinamento, 50 mil imagens de validação e 100 mil de teste. Fornece, portanto, um conjunto variado e objetos e principalmente, de características genéricas que apoiam o uso da técnica de *Fine-tuning*.

Assim sendo, o modelo foi ajustado, conforme os hiperparâmetros de treinamento apontados na Tabela 9.

Tabela 9 – Principais hiperparâmetros do treinamento do modelo RF-DETR-B

| Hiperparâmetro | Valor |
|-----------------------------|--------|
| Épocas | 15 |
| Batch size | 16 |
| Learning rate | 0,0001 |
| Otimizador | AdamW |
| Weight decay | 0,0001 |
| Resolução de entrada | 560 |

Fonte: Elaborada pelo autor (2025)

A quantidade baixa de épocas em relação ao YOLOv12 sustenta-se pelo fato de o RF-DETR alcançar um desempenho estável de maneira rápida em poucas épocas, como por exemplo, 10 épocas em cenários de classe única num problema agrícola, apontado por (SAPKOTA et al., 2025). O tamanho do *batch* foi reduzido dadas as restrições de memória do ambiente virtual do *Google Colab*. A taxa de aprendizagem foi assim definida para usufruir mais das vantagens do aprendizado anterior do modelo, de modo a utilizar os benefícios do *Fine-tuning*. A regularização L2 (*Weight decay*) foi mantida em 0.0001 por ser leve e ainda assim ser útil para controlar o sobreajuste. A resolução de entrada não foi aumentada, dadas as restrições do ambiente, de modo que ficou com um valor de 560 x 560 px.

4.7.4 D-FINE

Foi utilizada uma recente contribuição de (PENG et al., 2024), o D-FINE, um modelo de detecção de objetos voltado a obter um equilíbrio entre velocidade e acurácia, além de obter melhor acurácia na regressão referente à tarefa de localização de objetos, ponto importante

em modelos de detecção de objetos. Além do mais, no COCO *Dataset*, obteve desempenho superior aos modelos existentes, quando se considerou acurácia e eficiência em geral (como custo computacional, por exemplo).

O D-FINE é disponibilizado em duas versões: pré-treinado somente na base COCO 2017 e na base Objects365 + COCO 2017. Utilizou-se a segunda versão pré-treinada, por ser melhor para tarefas de *Fine-tuning*, além de ser mais ideal para generalização. Quanto a escala, o D-FINE pré-treinado em Objects365+COCO possui 4 tamanhos: D-FINE-S, D-FINE-M, D-FINE-L e D-FINE-X, respectivamente com 10, 19, 31 62 milhões de parâmetros de treinamento. Por considerar mais apropriado ao *dataset* utilizado neste trabalho, adotou-se o D-FINE-M, já que, por ser um equilíbrio importante em capacidade de aprendizado, dados os desafios envolvendo ruídos na base de dados obtida por smartphone e o controle de *overfitting*, que poderia ocorrer com mais facilidade num modelo com muitos parâmetros, dado o tamanho da base.

Ainda, destaca-se o amplo conjunto de classes contidas no *dataset* Objects365, no qual D-FINE é pré-treinado. Seu número extenso de 365 classes, cerca de 2 milhões de imagens e a presença de 30 milhões de objetos nelas, fornece um conjunto amplo de características e diferentes cenários para aprendizado do modelo. Portanto, apropriar-se dessa ampla capacidade de extração de características fornece um modelo com um bom ponto de partida para aprendizado específico de objetos, como é o caso de ovos de *Ae. aegypti*, através da aplicação de ajuste fino neste modelo.

Assim, a partir da implementação proposta por (PENG et al., 2024), implementou-se ajustes na implementação padrão dos autores com vista a tornar melhor a reproduzibilidade dos experimentos. Ademais, o modelo requer um conjunto de dados anotado no formato COCO, o que exigiu uma conversão no formato de anotação do dataset utilizado neste trabalho, de PASCAL VOC para COCO.

Então, acerca dos hiperparâmetros, o modelo foi configurado como mostra a Tabela 10. Observa-se a distinção entre taxas de aprendizagem, com um valor mais baixo (0.00002) para o *backbone* (extrator de características) do modelo e uma maior taxa geral de 0.0002. Tal diferenciação visa aproveitar melhor o aprendizado geral do modelo e sua capacidade de extração de características, dado seu referido amplo número de classes genéricas, no qual é pré-treinado e, em conjunto, ainda assim manter uma taxa de aprendizagem geral ainda baixa para ajuste fino.

Utilizou-se o *batch* de tamanho 32 para proporcionar maior estabilidade ao modelo, dentro

dos limites do ambiente de execução. O tamanho das imagens manteve-se em $640 \times 640 \text{ px}$ e o treinamento durou 132 épocas. Por fim, após a conclusão do treinamento, obtém-se os pesos que contém o melhor resultado do modelo no conjunto de validação.

Tabela 10 – Hiperparâmetros utilizados no treinamento do modelo D-FINE-M

| Hiperparâmetro | Valor |
|---------------------------------|---------|
| Épocas | 132 |
| Batch size | 32 |
| Learning rate (backbone) | 0,00002 |
| Learning rate geral | 0,0002 |
| Otimizador | AdamW |
| base size | 640 |

Fonte: Elaborada pelo autor (2025)

Ainda, a transformação *RandomPhotometricDistort*, também usada em (LIU et al., 2016), aleatoriamente distorce a imagem, baseado em *Color Jitter*, de modo a manipular brilho, contraste, saturação e matiz (PYTORCH, 2025d). O intento é tornar o modelo mais resiliente a variações de cor e luz. Por sua vez, *RandomZoomOut* aplica produz o efeito de um *Zoom out*, o que faz com que a imagem pareça mais distante; os pixels em volta são preenchidos com o valor definido de 0 (preto) (PYTORCH, 2025e).

Também, *RandomIoUCrop* faz alterações na imagem, de modo a deslocar o objeto de interesse para fora da imagem, mas mantendo um IoU mínimo, nesse caso 0.8; Na prática, grande parte do objeto ainda continuará visível na imagem e o restante, não (PYTORCH, 2025c). Outrossim, a transformação *RandomHorizontalFlip*, de maneira aleatória, faz um *flip* (ou giro) na imagem dada uma probabilidade, definida como 0.4, nesse caso (PYTORCH, 2025b).

Tabela 11 – Técnicas de *Data Augmentation* aplicadas no treinamento do D-FINE-M

| Técnica | Valor |
|---------------------------------|-------|
| RandomPhotometricDistort | 0,05 |
| RandomZoomOut | 0 |
| RandomIoUCrop | 0,8 |
| RandomHorizontalFlip | 0,4 |

Fonte: Elaborada pelo autor (2025)

Por fim, com vistas a obter um desempenho melhor quanto a generalização, controlar o ajuste fino e ampliar o conjunto de imagens de treinamento, utilizou-se técnicas de aumento de dados, as quais estão descritas na Tabela 11. Há uma particularidade nesse caso, no que tange

à aplicação de transformações nas imagens: São usadas técnicas mais agressivas de aumento de dados sobre as imagens até a época 120 e, seguidas por 4 épocas com transformações mais leves e, por fim, não são feitas mais transformações. Isto visa propor mais imagens ao modelo no início e assim tentar controlar o ajuste fino e, após isso, propor ao modelo imagens com transformações leves e, por fim, sem transformações, as quais são mais parecidas com as imagens do mundo real, com as quais o modelo terá que lidar na inferência. Esta estratégia de aplicação de transformações de *data augmentation* em fases é baseada no treinamento do próprio D-FINE, aplicado pelos autores em (PENG et al., 2024) e proposta no trabalho original de (LV et al., 2024).

4.8 MÉTRICAS DE AVALIAÇÃO DE DESEMPENHO

As métricas de avaliação permitem avaliar o sistema que está sendo desenvolvido. No contexto de modelos de Aprendizagem de Máquina, intenta-se saber quão bom é o funcionamento do modelo que se está desenvolvendo (ELGENDY, 2020). Isto posto, já obtidas as as métricas de desempenho, têm-se um guia para as ações futuras no treinamento e ajuste de modelos (GOODFELLOW; BENGIO; COURVILLE, 2016).

Também, é importante salientar que a definição de métricas é orientada ao problema que se tenta resolver e acrescenta-se, que, problemas específicos podem requerer o uso de métricas mais avançadas (GOODFELLOW; BENGIO; COURVILLE, 2016).

Logo, no âmbito deste trabalho, foram utilizadas as métricas descritas nesta seção, tanto as gerais, como, em especial, as voltadas especificamente para a tarefa de Detecção de Objetos.

Ainda, pontua-se que, métricas que dependam de True Negative (TN) não são aplicadas em contextos de modelos de detecção de objetos pelas seguintes razões: O *background* de uma imagem pode possuir um número muito expressivo de combinações de *pixels* e quaisquer dessas combinações seriam caracterizadas como TN. Assim, se fora adotada, haveria um conjunto vasto e, principalmente, indefinido de regiões TN, o que inflacionaria enganosamente a acurácia do modelo, de modo a levar a medições errôneas (ApX Machine Learning, 2025); O fim da tarefa de detecção de objetos é reconhecer e localizar os objetos de interesse na imagem e não em considerar o *background* para fins de medição de acerto. Por fim, trabalhos na literatura, como os que empregam *datasets* de *benchmark*, quais (LIN et al., 2014; EVERINGHAM et al., 2010) assim o fazem. Portanto, o uso de TN em detecção de objetos não é significativo e, logo, são comumente ignorados (SuperAnnotate, 2023; Neuralception, 2024).

Isto posto, no contexto de Detecção de Objetos, além das métricas usuais, usam-se métricas específicas como, por exemplo, mAP, que avalia a tarefa de detecção em si e é muito utilizada também em *benchmarking*.

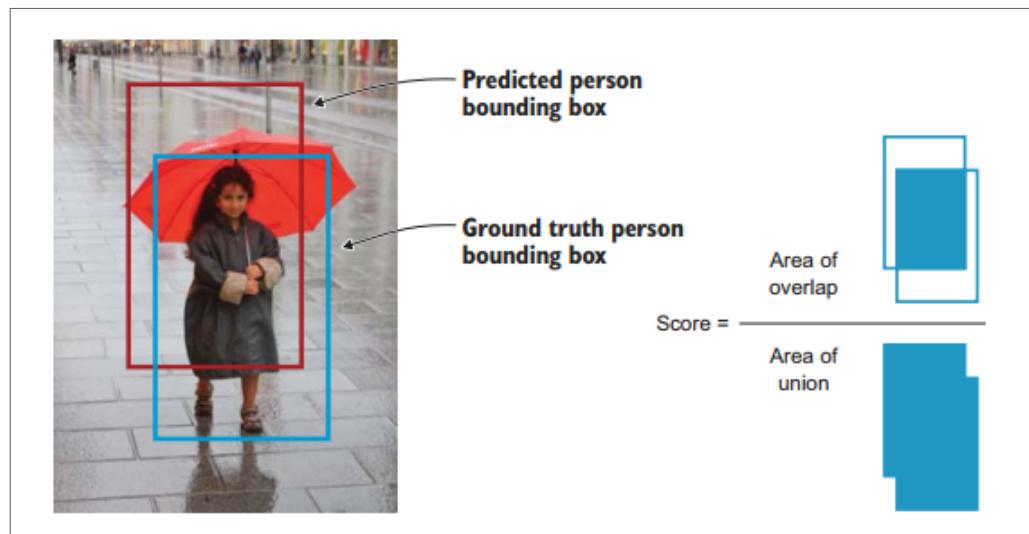
Além do mais, no contexto de Detecção de Objetos, as métricas são peculiares também devido ao fato de ser feita por humanos. A anotação das *bounding boxes* feita pelo rotulador é chamada de *ground-truth* e define onde o objeto de interesse está localizado na imagem. Como essa demarcação do *ground-truth* poder variar, em alguns *pixels*, de humano para humano, é difícil ao modelo detectar exatamente a caixa delimitadora rotulada pelo humano (SHANMUGAMANI, 2018).

Portanto, para determinar se uma previsão é correta ou errada, usa-se a métrica auxiliar IoU, que avalia a tarefa de localização. Esta métrica é descrita na subseção abaixo.

4.8.1 *Intersection Over Union (IoU)*

Dada uma rotulação verdadeira (*ground-truth*) e uma *bounding box* predita pelo modelo, esta métrica auxiliar mensura quanto boa é a detecção, baseado na interseção entre o posicionamento de ambas as caixas. Portanto, calcula-se a razão entre a área de interseção (*Area of Overlap*) entre elas e a área da união (*Area of Union*), como exibe a Figura 39.

Figura 39 – Definição de *Intersection over Union (IoU)*

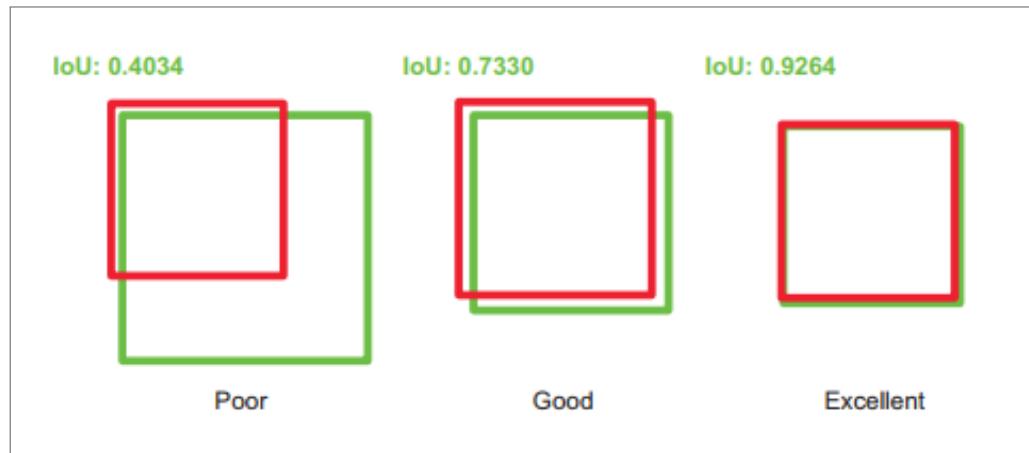


Fonte: (ELGENDY, 2020)

O valor dessa métrica auxiliar é definido de 0 a 100, de maneira que, quanto maior for a interseção entre as *bounding boxes* verdadeiras e as preditas pelo modelo, maior seu valor,

como ilustra a Figura 40 (ELGENDY, 2020).

Figura 40 – Exemplo de valores de *Intersection over Union* (IoU)



Fonte: (ELGENDY, 2020)

4.8.2 *mean Average Precision* (mAP)

mAP é obtida a partir da precisão (*Precision*) e revocação (*Recall*) e seus valores são porcentuais, portanto, variam de 0 a 100, de maneira que quanto maior o valor, melhor é o resultado. É diferente da acurácia medida em problema de classificação (ELGENDY, 2020).

Para tanto, mAP é obtida a partir da AP de cada classe, também chamada de precisão média, que, por sua vez é obtida após cálculo da área sob a Curva *precision-recall* (Ou Precision-Recall Curve (PR Curve)) e fornece um único valor que engloba o desempenho de *precision* e *recall* do modelo, de modo que provê uma avaliação geral do desempenho (ULTRALYTICS, 2025c). Em suma, AP agrupa PR Curve em um único valor, de modo a representar a média de todas as precisões.

Por fim, mAP é obtida após cálculo da média dos AP para cada classe. A predição é considerada correta se o mAP é superior a 0.5 (SHANMUGAMANI, 2018).

4.8.3 *Recall*

Também nominada de sensibilidade, mede a capacidade de o modelo identificar os casos positivos. Indica a proporção de instâncias (nesse caso, objetos) que o modelo identificou corretamente. Um *recall* alto indica que o modelo reconheceu a maioria dos objetos, com alguns falsos negativos (ULTRALYTICS, 2025b). É definida pela Fórmula 4.1.

$$\text{Recall} = \frac{VP}{VP + FN} \quad (4.1)$$

4.8.4 Precision

Também chadamada de especificidade, é o oposto do *Recall* e mede quantos dos objetos identificados estão realmente corretos. Alta precisão significa que o modelo tem poucos casos de falsos positivos (ULTRALYTICS, 2025b). Obedece à Fórmula 4.2.

$$\text{Precision} = \frac{VP}{VP + FP} \quad (4.2)$$

4.8.5 F1-Score

F1-Score sintetiza as *precision* e *recall* numa métrica só e representa ambas. É obtida calculando-se a média harmônica dessas métricas, como exposto na Fórmula 4.3 (ELGENDY, 2020). O valor *p* refere-se à precisão e *r*, ao *recall*. Isto posto, para obter um alto F1-Score, o modelo precisa também ter boa precisão e revocação, cumulativamente. Esta métrica fornece uma medição equilibrada do desempenho do modelo e é especialmente importante nos casos de *datasets* desbalanceados ou no caso em que, para o modelo, é importante tanto evitar falsos positivos como falsos negativos (ULTRALYTICS, 2025a).

$$F_1 = \frac{2pr}{p + r} \quad (4.3)$$

4.8.6 Contagem absoluta

Como a natureza do problema necessita obter a quantidade de ovos de *Aedes aegypti* presentes em palhetas de ovitrampa, foi utilizada também a contagem absoluta. Assim, a partir da contagem absoluta de ovos do mosquito e da contagem feita por técnicos da FIOCRUZ-PE, foram também obtidas as seguintes métricas:

4.8.6.1 *Mean Absolute Error (MAE)*

Mean Absolute Error (MAE), também chamada de Erro Médio Absoluto, é usada para medir erro em sistemas de previsão. Essa métrica, como aponta a Fórmula 4.4, expressa a média aritmética dos erros absolutos, i.e. a diferença absoluta entre o valor real e o valor previsto pelo modelo. Y_i refere-se ao valor observado, enquanto que X_i é o valor previsto.

$$MAE = \frac{1}{n} \sum_{i=1}^n |Y_i - X_i| \quad (4.4)$$

4.8.6.2 *Mean Squared Error (MSE)*

Mean Squared Error (MSE), também chamada de Erro Médio Quadrático, é semelhante ao MAE, no sentido de medir a média dos erros absolutos das previsões. Todavia, distingue-se desta por dar um peso maior às diferenças verificadas entre os valores real e previsto, em virtude da potenciação aplicada. Na prática, penaliza mais erros significativos. Assim, na Fórmula 4.5 e, tal como na Fórmula 4.4, Y_i e X_i também referem-se, respectivamente, ao valor real e ao valor estipulado pelo modelo.

$$MSE = \frac{1}{n} \sum_{i=1}^n (Y_i - X_i)^2 \quad (4.5)$$

4.8.6.3 *Erro Relativo Percentual*

É a diferença porcentual entre a contagem automática e a contagem de referência após inferência num conjunto de imagens. Nesse caso, é considerada as contagens automáticas e manuais totais num conjunto de palhetas para expressar uma diferença porcentual, seja positiva (o modelo contou a mais) ou negativa (o modelo contou a menos). Essa métrica é adotada também por (SANTOS et al., 2008; GUSMÃO; MACHADO; RODRIGUES, 2009; SILVA; RODRIGUES; ARAUJO, 2012b; FEITOSA et al., 2015) e é chamada igualmente de Erro Porcentual Global e neste trabalho, é também considerada na contagem geral de ovos por palheta.

$$\text{Erro Relativo Percentual} = \left(\frac{\text{Total Automático} - \text{Total Manual}}{\text{Total Manual}} \right) \times 100 \quad (4.6)$$

4.9 PROTOCOLO EXPERIMENTAL

4.9.1 Base para treinamento de modelos

O *dataset* derivado, de que trata 4.3.6, foi utilizado. O motivo é por fornecer um número maior de imagens, o que, por conseguinte, provê mais instâncias de objetos, além também de aumentar a variação dos dados em termos de formas em que o objeto está posto, como condições distintas de iluminação, de cor, bem como *backgrounds* diferentes *etc*. Portanto, a generalização por parte do modelo pode ser favorecida quando de sua aplicação para inferência em situações de uso real.

Assim sendo, o dataset derivado foi utilizado nos experimentos para treinamento dos modelos de detecção de objetos.

4.9.2 Palhetas contadas

Para fins de avaliação e comparação do desempenho dos modelos e abordagens propostas nos três fluxos apresentados, foi realizada a inferência nas 28 palhetas fornecidas pela FIOCRUZ-PE; estas possuem uma contagem de ovos realizada por técnicos treinados desta fundação.

As palhetas contadas são identificadas e possuem as quantidades de ovos, como detalha a Tabela 12.

4.9.3 Comparação dos fluxos

Para avaliar o impacto das etapas de pré-processamento, um mesmo modelo base de contagem (YOLOv10-M) será utilizado como base nos três fluxos. No Fluxo I, a inferência será realizada sobre as imagens resultantes da aplicação do modelo de restauração MPRNet. No Fluxo II, cada técnica de realce será avaliada de forma independente. O YOLOv10-M será aplicado separadamente às imagens tratadas com *Color Jitter*, com Equalização de Histograma e

Tabela 12 – Palhetas contadas manualmente por técnicos da FIOCRUZ e quantidade de ovos correspondente.

| # | Palheta | Quantidade de ovos |
|----|---------|--------------------|
| 1 | ASF_CP4 | 213 |
| 2 | ASF_CP5 | 361 |
| 3 | ASF_CP6 | 113 |
| 4 | ASF_CP7 | 137 |
| 5 | ASF_CP8 | 133 |
| 6 | ASF_CP9 | 172 |
| 7 | PAM_1 | 238 |
| 8 | PAM_2 | 95 |
| 9 | PAM_3 | 168 |
| 10 | PAM_4 | 234 |
| 11 | PAM_5 | 412 |
| 12 | PAM_6 | 210 |
| 13 | PAM_7 | 196 |
| 14 | PAM_8 | 408 |
| 15 | PAM_9 | 604 |
| 16 | PAM_10 | 402 |
| 17 | WAN_1 | 1212 |
| 18 | WAN_2 | 755 |
| 19 | WAN_3 | 48 |
| 20 | LESSA_1 | 293 |
| 21 | LESSA_2 | 348 |
| 22 | LESSA_3 | 323 |
| 23 | LESSA_4 | 211 |
| 24 | LESSA_5 | 130 |
| 25 | LESSA_6 | 133 |
| 26 | LESSA_7 | 258 |
| 27 | LESSA_8 | 153 |
| 28 | LESSA_9 | 75 |

Fonte: Elaborada pelo autor (2025)

com CLAHE. O objetivo é identificar qual dessas três técnicas de pré-processamento resulta no melhor desempenho de contagem. No Fluxo III, o desempenho do YOLOv10-M em contagem direta (sem pré-processamento) será avaliado e comparado com os outros modelos de detecção propostos (YOLOv12-M, RF-DETR-B e D-FINE-M) para determinar o modelo mais eficaz nesta abordagem.. Este protocolo de experimentos é sintetizado na Tabela 13.

A comparação do desempenho do modelo base (YOLOv10-M) nos três cenários permitirá

avaliar a contribuição relativa de cada fluxo (restauração, realce ou contagem direta).

Tabela 13 – Protocolo de experimentos do modelo base YOLOv10-M em relação às etapas de pré-processamento

| Modelo | Pré-processamento aplicado |
|-----------|----------------------------|
| YOLOv10-M | MPRNet |
| YOLOv10-M | Color Jitter |
| YOLOv10-M | Equalização de histograma |
| YOLOv10-M | CLAHE |
| YOLOv10-M | Nenhum (contagem direta) |

Fonte: Elaborada pelo autor (2025)

4.9.4 Avaliação final

Obtida a melhor abordagem de pré-processamento com o modelo base, na avaliação final, os modelos do Fluxo III (contagem direta sem pré-processamento) serão também avaliados. Esta avaliação utilizará o conjunto de 28 palhetas contadas por especialistas da FIOCRUZ-PE, que funciona como um conjunto de teste independente e representa um cenário de aplicação real. A contagem automática gerada pelos modelos será diretamente comparada com a contagem manual dos especialistas.

4.9.5 Métricas de comparação

A eficácia dos modelos será avaliada por duas perspectivas. A performance na tarefa de detecção será medida pela métrica mAP@.50. A acurácia na tarefa de contagem, que é o objetivo final deste trabalho, será medida primariamente pelo MAE e secundariamente pelo MSE e comparando os totais de ovos por palheta com os valores de referência (*ground-truth*).

Este processo culminará na identificação não apenas do melhor modelo dentro de cada fluxo, mas também na definição da abordagem geral (inter-fluxo) mais acurada para a contagem automática dos ovos.

4.10 CONCLUSÃO DO CAPÍTULO

No presente capítulo, foram apresentados o fluxo geral de desenvolvimento do trabalho, a forma de captura de imagens e tratamento posterior, o dispositivo auxiliar de captura de fotos, a criação da base de dados, os modelos treinados, o protocolo experimental e as métricas de avaliação utilizadas.

5 RESULTADOS E DISCUSSÃO

5.1 FLUXOS I E II: AVALIAÇÃO DAS ETAPAS DE PRÉ-PROCESSAMENTO

Seguindo o disposto na seção 4.9, o modelo base YOLOv10-M obteve performance conforme as subseções abaixo.

5.1.1 Desempenho do modelo base YOLOv10-M no treinamento

Conforme o protocolo experimental definido na seção 4.9, o modelo YOLOv10-M foi treinado na base derivada com os hiperparâmetros e técnicas de *data augmentation* expostos na subseção 4.7.1. Com o *score* de confiança de 0.25, o modelo obteve o desempenho conforme a Tabela 14.

Tabela 14 – Desempenho do base YOLOv10-M nos conjuntos de treino, validação e teste

| Conjunto | Precisão | Revocação | F1-Score | mAP@0,50 |
|--------------------|----------|-----------|----------|----------|
| Treinamento | 0,6824 | 0,9120 | 0,7807 | 0,862 |
| Validação | 0,6884 | 0,8933 | 0,7776 | 0,845 |
| Teste | 0,6466 | 0,8982 | 0,7519 | 0,852 |

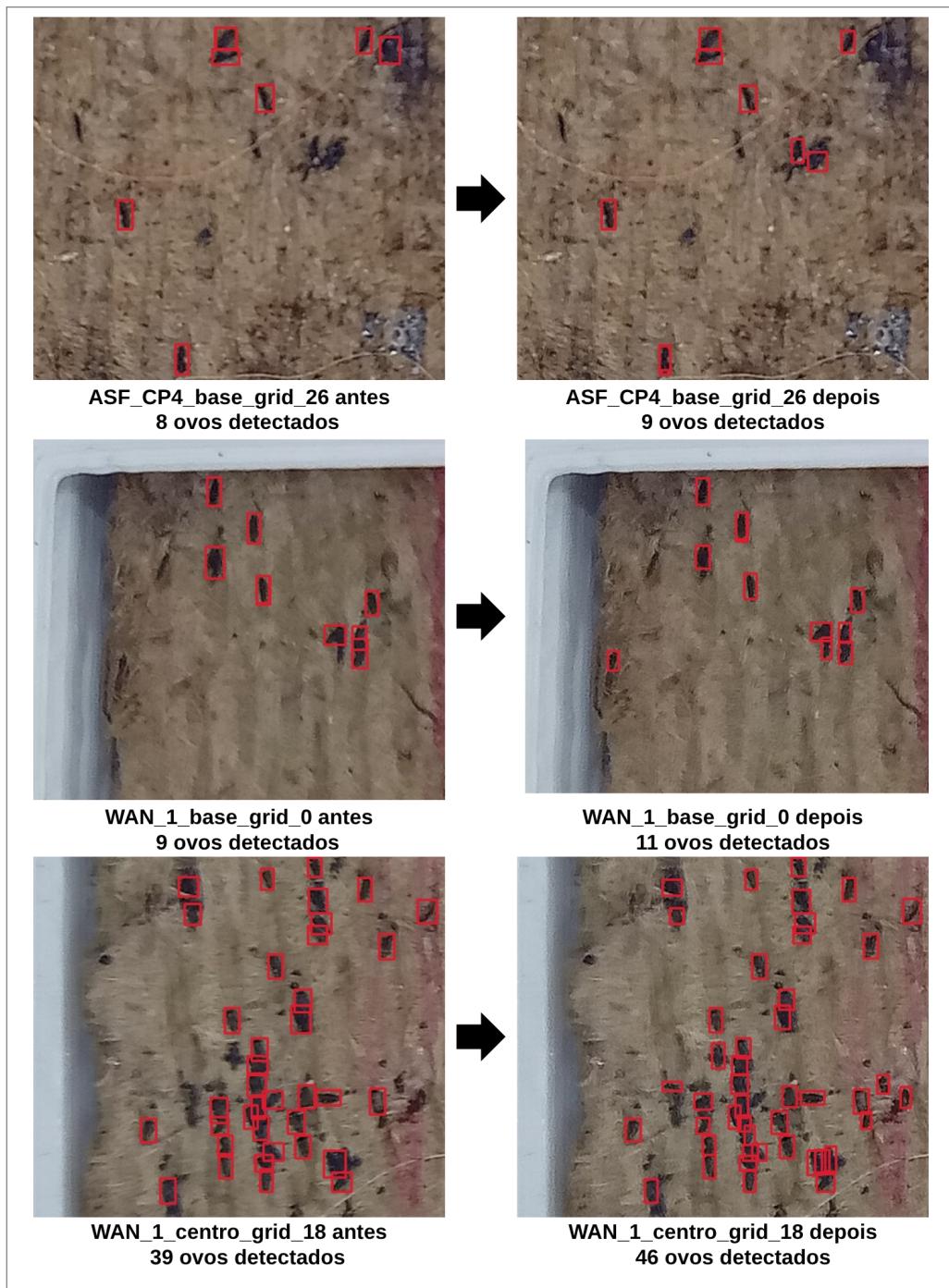
Fonte: Elaborada pelo autor (2025)

A priori, observa-se a estabilidade das métricas entre os conjuntos, o que minimiza a presença de sobreajustamento considerável. Os valores de precisão apontam um desempenho bom quanto ao acerto do modelo ao apontar ovos nas imagens. Com base nesse valor, há, porém, uma apresenta presença significante de falsos positivos, que é também explicada pelo cenário desafiador de contagem em imagens ruidosas. A revocação alta e estável mostra que o modelo consegue perceber boa parte dos ovos presentes nas imagens e desconsidera uma margem pequena na contagem, o que é uma boa performance. O valor de *F1-Score* confirma o bom desempenho geral do modelo e expressa o bom equilíbrio entre precisão e *recall*. Por fim, mAP@0,50, indica que, com IoU com limiar de 0.5 na predição das *boudning boxes* de detecção de ovos, o modelo obtém bom equilíbrio entre precisão e revocação no geral. Assim, o modelo mantém um bom desempenho entre as tarefas de classificação e de localização, mesmo em imagens de baixa resolução e com ruídos, como é o caso.

5.1.2 Pré-processamento com MPRNet

A partir do modelo base obtido, a etapa de pré-processamento com MPRNet com a tarefa de *Image deblurring* feita nas imagens das 28 palhetas contadas, resultou num conjunto de imagens que foram submetidas ao modelo base (YOLOv10-M) para avaliação. Esta abordagem teve impacto nas detecções de ovos, a exemplo do que mostra a Figura 41.

Figura 41 – Impacato da restauração do ruído de borramento com MPRNet no reconhecimento de ovos



Fonte: Figura do autor (2025)

A restauração aplicada pelo modelo produziu melhorias muito leves, o que pode ser por conta da complexidade visual do cenário. Ainda assim, conforme experimento, levou a aumento na detecção de ovos, o que favoreceu principalmente quando se tem uma quantidade numerosa de ovos na imagem e que não eram reconhecidos por conta do ruído. Nesses casos, ovos eram desconsiderados, principalmente quando estavam oclusos ou adjuntos.

Em contrapartida, a abordagem aumentou o número de detecções de falsos positivos em alguns casos, de maneira que o modelo base reconheceu ruído como sendo ovo.

5.1.2.1 Desempenho do modelo base com MPRNet

A avaliação do modelo base foi feita nas 28 palhetas contadas por técnicos da FIOCRUZ-PE. O modelo obteve o desempenho conforme a Tabela 15.

Tabela 15 – Desempenho do modelo base na contagem de ovos por palheta com MPRNet como pré-processamento

| Palheta | Contagem Manual | Contagem YOLOv10-M | mAP@0.50 |
|----------------------|-----------------|--------------------|----------|
| ASF_CP4 | 213 | 213 | 0,817 |
| ASF_CP5 | 361 | 419 | 0,753 |
| ASF_CP6 | 113 | 87 | 0,717 |
| ASF_CP7 | 137 | 145 | 0,762 |
| ASF_CP8 | 133 | 78 | 0,671 |
| ASF_CP9 | 172 | 150 | 0,764 |
| PAM_1 | 238 | 211 | 0,705 |
| PAM_2 | 95 | 537 | 0,739 |
| PAM_3 | 168 | 416 | 0,799 |
| PAM_4 | 234 | 519 | 0,897 |
| PAM_5 | 412 | 917 | 0,767 |
| PAM_6 | 210 | 222 | 0,876 |
| PAM_7 | 196 | 514 | 0,681 |
| PAM_8 | 408 | 449 | 0,890 |
| PAM_9 | 604 | 433 | 0,740 |
| PAM_10 | 402 | 320 | 0,573 |
| WAN_1 | 1212 | 1657 | 0,738 |
| WAN_2 | 755 | 1037 | 0,782 |
| WAN_3 | 48 | 49 | 0,938 |
| LESSA_1 | 293 | 166 | 0,761 |
| LESSA_2 | 348 | 775 | 0,864 |
| LESSA_3 | 323 | 379 | 0,787 |
| LESSA_4 | 211 | 219 | 0,854 |
| LESSA_5 | 130 | 119 | 0,897 |
| LESSA_6 | 133 | 158 | 0,845 |
| LESSA_7 | 258 | 182 | 0,875 |
| LESSA_8 | 153 | 217 | 0,910 |
| LESSA_9 | 75 | 72 | 0,734 |
| Total de ovos | 8.035 | 10.660 | – |

Nota: O símbolo “–” indica valor não aplicável.

Fonte: Elaborada pelo autor (2025).

A contagem absoluta indica que o modelo com esta abordagem superou a contagem de referência de forma significativa no somatório total de ovos. Em algumas palhetas, por exemplo, a contagem foi inferior e noutras, foi muito superior. No geral, em 17 palhetas, a contagem do modelo foi superior à contagem manual, em 10 foi inferior e por uma vez o resultado

correspondeu (palheta ASF_CP4).

As demais métricas, como mostra a Tabela 16, apontam um erro médio absoluto de cerca de 136 unidades, MSE muito expressivo e erro relativo percentual indica que no geral, o modelo superou em 32% a contagem de referência. A média aritmética dos mAP@0.50 das palhetas aponta um valor razoável.

Tabela 16 – Métricas globais de desempenho do modelo base com MPRNet

| Métrica | Valor |
|--------------------------|-----------|
| MAE | 136,61 |
| MSE | 43.936,04 |
| Erro Relativo Percentual | 32,66% |
| Média geral mAP@0,50 | 0,7906 |

Fonte: Elaborada pelo autor (2025).

Nesse caso, essa abordagem de pré-processamento produziu resultados ainda insuficientes, mesmo no cenário de imagens ruidosas.

5.1.3 Pré-processamento com Color Jitter

Esta abordagem como etapa intermediária à contagem proporcionou os resultados, conforme a Tabela 17.

5.1.3.1 Desempenho do modelo base com *Color Jitter*

Tabela 17 – Desempenho do modelo base na contagem de ovos por palheta com *Color Jitter* como pré-processamento

| Palheta | Contagem Manual | Contagem YOLOv10-M | mAP@0.50 |
|----------------------|-----------------|--------------------|----------|
| ASF_CP4 | 213 | 253 | 0,766 |
| ASF_CP5 | 361 | 499 | 0,733 |
| ASF_CP6 | 113 | 132 | 0,691 |
| ASF_CP7 | 137 | 202 | 0,745 |
| ASF_CP8 | 133 | 165 | 0,572 |
| ASF_CP9 | 172 | 281 | 0,634 |
| PAM_1 | 238 | 321 | 0,738 |
| PAM_2 | 95 | 681 | 0,591 |
| PAM_3 | 168 | 274 | 0,756 |
| PAM_4 | 234 | 571 | 0,830 |
| PAM_5 | 412 | 1007 | 0,782 |
| PAM_6 | 210 | 248 | 0,857 |
| PAM_7 | 196 | 672 | 0,660 |
| PAM_8 | 408 | 624 | 0,850 |
| PAM_9 | 604 | 626 | 0,672 |
| PAM_10 | 402 | 426 | 0,482 |
| WAN_1 | 1212 | 1812 | 0,687 |
| WAN_2 | 755 | 1067 | 0,759 |
| WAN_3 | 48 | 51 | 0,893 |
| LESSA_1 | 293 | 181 | 0,652 |
| LESSA_2 | 348 | 847 | 0,831 |
| LESSA_3 | 323 | 528 | 0,676 |
| LESSA_4 | 211 | 279 | 0,786 |
| LESSA_5 | 130 | 160 | 0,830 |
| LESSA_6 | 133 | 187 | 0,741 |
| LESSA_7 | 258 | 216 | 0,815 |
| LESSA_8 | 153 | 242 | 0,871 |
| LESSA_9 | 75 | 81 | 0,711 |
| Total de ovos | 8.035 | 12.633 | – |

Nota: O símbolo “–” indica valor não aplicável.

Fonte: Elaborada pelo autor (2025).

Em alguns casos, esta abordagem resultou num realce que possibilitou o reconhecimento de ovos antes não detectados. Entretanto, produziu um número muito expressivo na contagem de ovos. Em algumas palhetas, a diferença entre as contagens foi de 600 (palheta WAN_1) ovos. Em 25 de 28 palhetas o modelo superou a contagem de referência. Em 15 palhetas, o grau de erro ficou abaixo de 100 unidades e para as demais treze, superou esse valor, o que evidencia o grau de erro geral dessa abordagem.

Com 4598 ovos contados a mais no total, o que equivale a 57% de erro relativo percentual, conforme a Tabela 18, o modelo apresentou erro médio considerável (175,21), o que representa um erro de 175 ovos por palheta, que é um valor expressivo. O MSE também foi acentuado nessa abordagem e mAP@0.50 teve leve redução.

17.

Tabela 18 – Métricas globais de desempenho do modelo YOLOv10-M com *Color Jitter*

| Métrica | Valor |
|--------------------------|-----------|
| MAE | 175,21 |
| MSE | 68.645,36 |
| Erro relativo percentual | 57,22% |
| Média geral mAP@0.50 | 0,73 |

Fonte: Elaborada pelo autor (2025).

Dessarte, esta também abordagem resultou em erros elevados, especialmente em relação à contagem de ovos.

5.1.4 Pré-processamento com Equalização de histograma

5.1.4.1 Desempenho do modelo base com Equalização de histograma

O uso desta abordagem ainda fez a contagem total de ovos como um todo superar significativamente a contagem de referência, de maneira que em 23 palhetas foram contados ovos a mais e a diferença geral atingiu 2697 ovos. Em relação a graus de erro, em 16 palhetas, a diferença entre as contagens não ultrapassou 100 ovos. Todavia, em 12 palhetas, a contagem superou este valor e contribui para um erro considerável. Os resultados são mostrados na Tabela 19.

Tabela 19 – Desempenho do modelo YOLOv10-M na contagem de ovos por palheta com Equalização de histograma como pré-processamento

| Palheta | Contagem Manual | Contagem YOLOv10-M | mAP@0.50 |
|----------------------|-----------------|--------------------|----------|
| ASF_CP4 | 213 | 245 | 0,433 |
| ASF_CP5 | 361 | 391 | 0,419 |
| ASF_CP6 | 113 | 115 | 0,142 |
| ASF_CP7 | 137 | 149 | 0,190 |
| ASF_CP8 | 133 | 157 | 0,323 |
| ASF_CP9 | 172 | 302 | 0,260 |
| PAM_1 | 238 | 307 | 0,550 |
| PAM_2 | 95 | 275 | 0,489 |
| PAM_3 | 168 | 424 | 0,498 |
| PAM_4 | 234 | 480 | 0,534 |
| PAM_5 | 412 | 897 | 0,682 |
| PAM_6 | 210 | 255 | 0,468 |
| PAM_7 | 196 | 601 | 0,464 |
| PAM_8 | 408 | 528 | 0,610 |
| PAM_9 | 604 | 561 | 0,513 |
| PAM_10 | 402 | 351 | 0,352 |
| WAN_1 | 1212 | 1404 | 0,627 |
| WAN_2 | 755 | 883 | 0,662 |
| WAN_3 | 48 | 56 | 0,283 |
| LESSA_1 | 293 | 104 | 0,194 |
| LESSA_2 | 348 | 680 | 0,545 |
| LESSA_3 | 323 | 370 | 0,409 |
| LESSA_4 | 211 | 253 | 0,493 |
| LESSA_5 | 130 | 181 | 0,388 |
| LESSA_6 | 133 | 229 | 0,318 |
| LESSA_7 | 258 | 209 | 0,397 |
| LESSA_8 | 153 | 248 | 0,492 |
| LESSA_9 | 75 | 77 | 0,276 |
| Total de ovos | 8.035 | 10.732 | – |

Nota: O símbolo “–” indica valor não aplicável.

Fonte: Elaborada pelo autor (2025).

A partir da Tabela 20, identifica-se valor razoável de erro médio, ao considerar-se o cenário de contagem, porém um valor ainda expressivo para o MSE é observado, o que reflete as diferenças entre as contagens. A média dos valores de mAP@0.50 está diminuta e reflete a piora na tarefa de detecção de objetos.

Tabela 20 – Métricas globais de desempenho do modelo YOLOv10-M com Equalização de histograma

| Métrica | Valor |
|--------------------------|-----------|
| MAE | 120,04 |
| MSE | 29.620,11 |
| Erro relativo percentual | 33,56% |
| Média geral mAP@0.50 | 0,43 |

Fonte: Elaborada pelo autor (2025).

5.1.5 Pré-processamento com CLAHE

5.1.5.1 Desempenho do modelo base com CLAHE

Como mostra a Tabela 21, a equalização adaptativa, por sua vez, obteve desempenho conservador, muito próximo à abordagem anterior e com um leve aumento na quantidade total de ovos contados.

Tabela 21 – Desempenho do modelo YOLOv10-M na contagem de ovos por palheta com CLAHE como pré-processamento

| Palheta | Contagem Manual | Contagem YOLOv10-M | mAP@0.50 |
|----------------------|-----------------|--------------------|----------|
| ASF_CP4 | 213 | 214 | 0,769 |
| ASF_CP5 | 361 | 376 | 0,704 |
| ASF_CP6 | 113 | 79 | 0,571 |
| ASF_CP7 | 137 | 141 | 0,630 |
| ASF_CP8 | 133 | 117 | 0,583 |
| ASF_CP9 | 172 | 221 | 0,659 |
| PAM_1 | 238 | 262 | 0,707 |
| PAM_2 | 95 | 554 | 0,659 |
| PAM_3 | 168 | 407 | 0,782 |
| PAM_4 | 234 | 510 | 0,857 |
| PAM_5 | 412 | 877 | 0,815 |
| PAM_6 | 210 | 229 | 0,878 |
| PAM_7 | 196 | 547 | 0,704 |
| PAM_8 | 408 | 515 | 0,859 |
| PAM_9 | 604 | 525 | 0,703 |
| PAM_10 | 402 | 356 | 0,525 |
| WAN_1 | 1212 | 1609 | 0,746 |
| WAN_2 | 755 | 967 | 0,803 |
| WAN_3 | 48 | 51 | 0,924 |
| LESSA_1 | 293 | 154 | 0,579 |
| LESSA_2 | 348 | 733 | 0,854 |
| LESSA_3 | 323 | 391 | 0,714 |
| LESSA_4 | 211 | 228 | 0,806 |
| LESSA_5 | 130 | 127 | 0,860 |
| LESSA_6 | 133 | 167 | 0,798 |
| LESSA_7 | 258 | 192 | 0,760 |
| LESSA_8 | 153 | 214 | 0,856 |
| LESSA_9 | 75 | 75 | 0,672 |
| Total de ovos | 8.035 | 10.838 | – |

Nota: O símbolo “–” indica valor não aplicável.

Fonte: Elaborada pelo autor (2025).

A diferença entre as contagens do modelo em relação às contagens de referência separadas por grupos de palhetas, exibida na Tabela 22 evidencia a tendência do modelo superar a contagem com valores de diferença consideravelmente altos, em que pese estar associado à detecção em imagens ruidosas.

Tabela 22 – Diferença entre contagem do modelo base e contagem de referência entre os grupos de palhetas

| Grupo de Palhetas | Contagem do modelo |
|-------------------|--------------------|
| ASF_CP | 19 ovos a mais |
| PAM | 1560 ovos a mais |
| WAN | 612 ovos mais |
| LESSA | 357 ovos a mais |

Fonte: Elaborada pelo autor (2025).

Conforme Tabela 23, o erro médio é ainda conservador, com cerca de 127 ovos de diferença por palheta e MSE também alto refletem a contagem com considerada discrepância. Erro relativo percentual é ainda significativo e a média dos valores de mAP@0.50 apresenta desempenho similar à abordagem anterior.

Tabela 23 – Métricas globais de desempenho do modelo YOLOv10-M com CLAHE

| Métrica | Valor |
|--------------------------|-----------|
| MAE | 127,46 |
| MSE | 39.016,61 |
| Erro relativo percentual | 34,88% |
| Média geral mAP@0.50 | 0,74 |

Fonte: Elaborada pelo autor (2025).

5.2 FLUXO III: CONTAGEM DIRETA SEM PRÉ-PROCESSAMENTO

Neste fluxo, os modelos treinados na base derivada foram usados para inferência nas palhetas contadas sem etapa intermediária de pré-processamento.

5.2.1 YOLOv10-M

Conforme a Tabela 24, o modelo YOLOV10-M apresentou um desempenho considerável em relação à contagem total de ovos, com um total de 1867 ovos a mais que a contagem manual, o que representa um valor mais próximo da quantidade real de ovos. Em 12 palhetas, a diferença entre as contagens foi no máximo de 30 ovos; ainda, em 6 palhetas, a diferença esteve entre 31 e 100 e nas demais 11, a diferença ultrapassou 100 ovos.

Tabela 24 – Desempenho do modelo YOLOv10-M na contagem direta de ovos em palheta de ovitrampa

| Palheta | Contagem Manual | Contagem YOLOv10-M | mAP@0.50 |
|----------------------|--------------------|-----------------------|----------|
| ASF_CP4 | 213 | 192 | 0,855 |
| ASF_CP5 | 361 | 370 | 0,770 |
| ASF_CP6 | 113 | 80 | 0,734 |
| ASF_CP7 | 137 | 120 | 0,782 |
| ASF_CP8 | 133 | 65 | 0,709 |
| ASF_CP9 | 172 | 132 | 0,795 |
| PAM_1 | 238 | 203 | 0,791 |
| PAM_2 | 95 | 488 | 0,752 |
| PAM_3 | 168 | 379 | 0,833 |
| PAM_4 | 234 | 491 | 0,898 |
| PAM_5 | 412 | 875 | 0,813 |
| PAM_6 | 210 | 213 | 0,903 |
| PAM_7 | 196 | 476 | 0,736 |
| PAM_8 | 408 | 434 | 0,908 |
| PAM_9 | 604 | 404 | 0,749 |
| PAM_10 | 402 | 289 | 0,579 |
| WAN_1 | 1212 | 1551 | 0,765 |
| WAN_2 | 755 | 987 | 0,799 |
| WAN_3 | 48 | 46 | 0,955 |
| LESSA_1 | 293 | 151 | 0,762 |
| LESSA_2 | 348 | 718 | 0,891 |
| LESSA_3 | 323 | 344 | 0,798 |
| LESSA_4 | 211 | 196 | 0,873 |
| LESSA_5 | 130 | 107 | 0,924 |
| LESSA_6 | 133 | 146 | 0,854 |
| LESSA_7 | 258 | 175 | 0,862 |
| LESSA_8 | 153 | 202 | 0,936 |
| LESSA_9 | 75 | 68 | 0,787 |
| Total de ovos | 8.035 | 9.902 | – |

Nota: O símbolo “–” indica valor não aplicável.

Fonte: Elaborada pelo autor (2025).

Como mostra a Tabela 25, tanto MAE quanto MSE apresentam valores razoáveis. O erro relativo porcentual mostra uma diferença de 23% entre as contagens automática e manual. A média dos mAP@0.50 apresenta um considerável equilíbrio entre precisão e sensibilidade, além de uma melhora significativa na tarefa de detecção de objetos.

Tabela 25 – Métricas globais de desempenho do modelo YOLOv10-M na contagem direta

| Métrica | Valor |
|--------------------------|-----------|
| MAE | 123,75 |
| MSE | 34.182,39 |
| Erro relativo percentual | 23,23% |
| Média geral mAP@0.50 | 0,81 |

Fonte: Elaborada pelo autor (2025).

Este método, portanto, apresenta resultados promissores na contagem automática de ovos de *A. aegypti*.

5.2.2 YOLOv12-M

Com este modelo, conforme a Tabela 26, a contagem automática de ovos apresenta também valores mais próximos à contagem manual, o que é um resultado importante nesta tarefa, principalmente no caso de contagem a partir de imagens capturadas por *smartphone*. A menor diferença entre as contagens foi na palheta PAM_6, em que a diferença foi de 2 ovos a mais. No geral, a contagem do modelo foi maior que a manual em 14 palhetas e menor noutras 14; ainda, quanto a grau de erro, em 11 palhetas, a diferença foi de até 30 ovos.

Tabela 26 – Desempenho do modelo YOLOv12-M na contagem direta de ovos em palheta de ovitrampa

| Palheta | Contagem Manual | Contagem YOLOv12-M | mAP@0.50 |
|----------------------|--------------------|-----------------------|----------|
| ASF_CP4 | 213 | 197 | 0,839 |
| ASF_CP5 | 361 | 353 | 0,723 |
| ASF_CP6 | 113 | 93 | 0,683 |
| ASF_CP7 | 137 | 146 | 0,750 |
| ASF_CP8 | 133 | 80 | 0,646 |
| ASF_CP9 | 172 | 135 | 0,727 |
| PAM_1 | 238 | 182 | 0,743 |
| PAM_2 | 95 | 354 | 0,699 |
| PAM_3 | 168 | 309 | 0,816 |
| PAM_4 | 234 | 472 | 0,900 |
| PAM_5 | 412 | 780 | 0,835 |
| PAM_6 | 210 | 212 | 0,917 |
| PAM_7 | 196 | 426 | 0,736 |
| PAM_8 | 408 | 364 | 0,880 |
| PAM_9 | 604 | 372 | 0,728 |
| PAM_10 | 402 | 385 | 0,581 |
| WAN_1 | 1212 | 1421 | 0,774 |
| WAN_2 | 755 | 872 | 0,803 |
| WAN_3 | 48 | 77 | 0,970 |
| LESSA_1 | 293 | 159 | 0,683 |
| LESSA_2 | 348 | 659 | 0,882 |
| LESSA_3 | 323 | 124 | 0,742 |
| LESSA_4 | 211 | 191 | 0,838 |
| LESSA_5 | 130 | 57 | 0,887 |
| LESSA_6 | 133 | 159 | 0,798 |
| LESSA_7 | 258 | 164 | 0,867 |
| LESSA_8 | 153 | 206 | 0,922 |
| LESSA_9 | 75 | 77 | 0,742 |
| Total de ovos | 8.035 | 9.026 | – |

Nota: O símbolo “–” indica valor não aplicável.

Fonte: Elaborada pelo autor (2025).

Este modelo ainda obteve um erro médio, bem como MSE bastante reduzidos, o que minimiza a diferença entre as contagens automática e manual, como mostra a Tabela 27. O erro relativo percentual batante diminuto aponta para uma inferência que, no geral, aproxima-se mais da contagem de referência. Ainda, a média dos mAP@0.50 é um valor consideravelmente

bom e demonstra boa performance do modelo na detecção com IoU de 0.5 e boa relação entre precisão e *recall*.

Tabela 27 – Métricas globais de desempenho do modelo YOLOv12-M na contagem direta

| Métrica | Valor |
|--------------------------|-----------|
| MAE | 107,04 |
| MSE | 22.380,04 |
| Erro relativo percentual | 12,33% |
| Média geral mAP@0.50 | 0,79 |

Fonte: Elaborada pelo autor (2025).

5.2.3 RF-DETR-B

Como exibe a Tabela 28, o RF-DETR-B apresenta desempenho bastante elevado na contagem total de ovos, com uma diferença de 223 ovos a menos comparado à contagem manual. Ainda, com imagens com a presença de diversos tipos de degradação, o modelo consegue aproximar-se da contagem real de ovos com diferença acentuadamente baixa, como nas lhetas WAN_1 (diferença de 56 ovos), LESSA_5 (8 ovos), LESSA_6 (7 ovos) e LESSA_9 (5 ovos).

Tabela 28 – Desempenho do modelo RF-DETR-B na contagem direta de ovos em palhetas de ovitrampa

| Palheta | Contagem Manual | Contagem RF-DETR-B |
|----------------------|--------------------|-----------------------|
| ASF_CP4 | 213 | 170 |
| ASF_CP5 | 361 | 310 |
| ASF_CP6 | 113 | 96 |
| ASF_CP7 | 137 | 125 |
| ASF_CP8 | 133 | 69 |
| ASF_CP9 | 172 | 100 |
| PAM_1 | 238 | 149 |
| PAM_2 | 95 | 306 |
| PAM_3 | 168 | 261 |
| PAM_4 | 234 | 384 |
| PAM_5 | 412 | 654 |
| PAM_6 | 210 | 183 |
| PAM_7 | 196 | 376 |
| PAM_8 | 408 | 315 |
| PAM_9 | 604 | 290 |
| PAM_10 | 402 | 239 |
| WAN_1 | 1212 | 1268 |
| WAN_2 | 755 | 668 |
| WAN_3 | 48 | 67 |
| LESSA_1 | 293 | 168 |
| LESSA_2 | 348 | 530 |
| LESSA_3 | 323 | 237 |
| LESSA_4 | 211 | 167 |
| LESSA_5 | 130 | 122 |
| LESSA_6 | 133 | 130 |
| LESSA_7 | 258 | 157 |
| LESSA_8 | 153 | 191 |
| LESSA_9 | 75 | 80 |
| Total de ovos | 8.035 | 7.812 |

Fonte: Elaborada pelo autor (2025).

Outrossim, a quantidade de ovos detectada por palheta pelo modelo superou a quantidade real em 10 ocasiões e foi inferior em 18. Ainda, como mostra a Tabela 29, quanto às faixas de diferença entre as contagens automática e manual, a diferença não ultrapassou 100 ovos em 20 das 28 palhetas. Estas medições representam uma contagem geral com performance bastante acentuada.

Tabela 29 – Faixas de diferença de ovos e quantidade de palhetas na contagem com RF-DETR-B

| Faixa de diferença de ovos | Número de palhetas |
|-----------------------------------|---------------------------|
| 0 – 10 | 6 |
| 11 – 30 | 4 |
| 31 – 100 | 12 |
| ≥ 101 | 8 |

Fonte: Elaborada pelo autor (2025).

Doutra forma, à nível de grupo de palhetas, o modelo contou mais ovos do que a contagem manual em alguma e menos, noutras, como exibe a Tabela 30.

Tabela 30 – Diferença entre contagem do RF-DETR-B e contagem de referência entre os grupos de palhetas

| Grupo de Palhetas | Contagem RF-DETR-B |
|--------------------------|-------------------------------|
| ASF_CP (6 palhetas) | 259 ovos a menos |
| PAM (10 palhetas) | 164 ovos a mais |
| WAN (3 palhetas) | 12 ovos a menos |
| LESSA (9 palhetas) | 142 ovos a menos |

Fonte: Elaborada pelo autor (2025).

Entretanto, ao considerar a quantidade de palhetas em cada grupo, observa-se que o modelo teve um desempenho substancial dada a quantidade de palhetas em cada grupo. Por exemplo, mesmo na contagem das 10 palhetas PAM, o modelo contou apenas 164 ovos a mais. Este desempenho é ainda mais elevado ao se considerar que nesse grupo há 2.967 ovos. Destaca-se também a performance nas palhetas WAN, que mesmo com 2.015 ovos, a contagem automática reconheceu somente 12 ovos a menos. Também, nas 9 palhetas LESSA, que possui 1.924 ovos, a contagem automática apresentou uma diferença bastante diminuta de 142 ovos a menos. Por fim, nas palhetas ASF_CP, a contagem apresentou a maior diferença, entretanto, ainda razoável, considerando que possui 1.129 ovos.

Quanto aos demais resultados, como exibe a Tabela 31, o erro médio foi de somente cerca de 91 ovos, o que corrobora o alto desempenho da contagem automática. Semelhantemente, o MSE apresenta valor bastante reduzido, o que demonstra o baixo grau de diferença entre a contagem do modelo e a contagem real de ovos. Ainda, o erro relativo percentual é acentuadamente baixo e destaca o potencial deste modelo para contagem de ovos em aplicações de mundo real no contexto da vigilância entomológica.

Tabela 31 – Métricas globais de desempenho do modelo RF-DETR-B na contagem direta de ovos em palheta de ovitrampa

| Métrica | Valor |
|--------------------------|-----------|
| MAE | 91,96 |
| MSE | 14.433,96 |
| Erro relativo percentual | -2,77% |

Fonte: Elaborada pelo autor (2025).

5.2.4 D-FINE-M

O D-FINE-M, por sua vez, distancia-se de forma considerável em termos de contagem absoluta de ovos, como exibe a Tabela 32,. Nesse caso, a diferença entre as contagens é de 3.116 ovos a mais em relação à contagem manual. O melhor desempenho foi nas palhetas PAM_10, LESSA_3, LESSA_1, ASF_CP9, PAM_1 em que o modelo apresentou diferença porcentual de, respectivamente, 2,29%, 4,44%, 4,64%, -7,03% e -9,16%.

Em relação aos graus de erro na contagem absoluta, em 17 palhetas, a diferença não ultrapassou 100 ovos, entretanto ultrapassou este limiar noutras 10 palhetas e apenas uma vez, o erro esteve abaixo de 10.

Tabela 32 – Desempenho do modelo D-FINE-M na contagem direta de ovos em palhetas de ovitrampa

| Palheta | Contagem Manual | Contagem D-FINE-M |
|----------------------|--------------------|----------------------|
| ASF_CP4 | 213 | 254 |
| ASF_CP5 | 361 | 445 |
| ASF_CP6 | 113 | 180 |
| ASF_CP7 | 137 | 224 |
| ASF_CP8 | 133 | 151 |
| ASF_CP9 | 172 | 185 |
| PAM_1 | 238 | 262 |
| PAM_2 | 95 | 481 |
| PAM_3 | 168 | 378 |
| PAM_4 | 234 | 503 |
| PAM_5 | 412 | 855 |
| PAM_6 | 210 | 276 |
| PAM_7 | 196 | 485 |
| PAM_8 | 408 | 438 |
| PAM_9 | 604 | 433 |
| PAM_10 | 402 | 393 |
| WAN_1 | 1212 | 1430 |
| WAN_2 | 755 | 893 |
| WAN_3 | 48 | 143 |
| LESSA_1 | 293 | 280 |
| LESSA_2 | 348 | 706 |
| LESSA_3 | 323 | 338 |
| LESSA_4 | 211 | 267 |
| LESSA_5 | 130 | 201 |
| LESSA_6 | 133 | 236 |
| LESSA_7 | 258 | 269 |
| LESSA_8 | 153 | 283 |
| LESSA_9 | 75 | 162 |
| Total de ovos | 8.035 | 11.151 |

Fonte: Elaborada pelo autor (2025).

Ainda, o modelo contou ovos a mais em 25 das 28 palhetas, o que equivale a cerca de 38% ovos a mais, como exibe a Tabela 33. O erro médio foi razoável, bem como o MSE também; isso ressalta que, mesmo o modelo contando objetos a mais na maioria dos casos, a diferença permanece diminuta em outros casos, o que pode explicar o valor não tão elevado para MAE e MSE.

Tabela 33 – Métricas globais de desempenho do modelo D-FINE-M na contagem direta de ovos em palhetas de ovitrampas

| Métrica | Valor |
|--------------------------|-----------|
| MAE | 125,07 |
| MSE | 30.328,43 |
| Erro relativo percentual | 38,78% |

Fonte: Elaborada pelo autor (2025).

Desta forma, mesmo em imagens de baixa resolução, o D-FINE aproxima-se da contagem real em algumas palhetas, porém distancia-se acentuadamente noutras.

5.3 AVALIAÇÃO FINAL

No âmbito das atividades de vigilância entomológica é fundamental obter indicadores importantes sobre o vetor de doenças numa região. No uso de ovitrampas para deposição de ovos, é passo importante a obtenção de indicadores de ovos; ainda, é importante que sejam fidedignos pois sobre eles baseiam-se passos posteriores importantes para uma implementação exitosa da vigilância entomológica no controle do *A. aegypti*.

Destarte, dado que a contagem de ovos influencia de forma direta no planejamento de ações para combate ao mosquito, tal contagem automática de ovos deve aproximar-se o máximo possível da quantidade real de ovos nas palhetas de ovitrampa examinadas. Ademais, na contagem de palhetas, o erro médio é o mais importante do que os erros individuais, pois o método será aplicado sobre várias palhetas em uso prático; mesmo assim, por óbvio, a abordagem deve obter bom desempenho em palhetas individuais, mantendo equilíbrio entre entre falsos positivos e falsos negativos e não cometer erros consideravelmente grandes.

Em relação ao impacto de técnicas de pré-processamento na contagem automática, a aplicação do modelo de restauração de imagens (MPRNet) especializado na remoção de ruído de borramento obteve menor diferença entre as contagens automática e manual, bem como maior mAP@0.50 no geral (0,79) e menor erro relativo percentual (32,66%). Todavia, a aplicação de equalização de histograma apresentou menor MAE (120,04) dentre as técnicas de pré-processamento, enquanto que CLAHE obteve performance inferior a esta, porém com métricas próximas, como MAE de 127,46. Dentre todas as técnicas, *color jitter* apresentou menor performance com erros significativos, como diferença de 4598 ovos em relação à contagem manual nas 28 palhetas utilizadas. Desta forma, as abordagens de MPRNet e equalização de

histograma apresentam resultados mais promissores quanto à contagem associada técnicas de pré-processamento.

Por outro lado, a respeito da abordagem de contagem direta (sem pré-processamento), ao considerar principalmente os resultados da contagem absoluta, erro médio absoluto, erro médio quadrático e erro relativo percentual, o modelo RF-DETR-B apresentou performance mais próxima da adequada para aplicação em cenários reais de contagem, de modo a obter a contagem absoluta mais próxima da manual, com MAE de 91,96, MSE de 14.433,96 e erro relativo percentual de -2,77% em relação à contagem manual nas 28 palhetas consideradas. De igual modo, em comparação com os demais modelos associados ou não a técnicas de pré-processamento, obteve o melhor desempenho, o que leva à avaliação de que, nesse cenário experimental, a contagem direta levou a melhor desempenho nas principais métricas avaliadas.

5.3.1 Comparativo com trabalhos da literatura na contagem de ovos de *Ae. aegypti*

No conjunto de abordagens de Processamento de Imagens, o algoritmo proposto em (ROCHA; BIZERRA; MAGALHÃES, 2019), foi validado numa única imagem contendo 26 ovos, dos quais o algoritmo detectou 14, o que resulta numa acurácia de cerca de 53,85%, abaixo do atingido pelo RF-DETR-B neste trabalho (97,23%), ainda que este tenha sido submetido a um conjunto expressivo de imagens.

De modo similar, o uso da técnica de Processamento de Imagens, agora aliada à lógica fuzzy, o sistema de inferência referido em (COSTA, 2017), obteve acurácia de 98,94% num conjunto de 206 imagens que continham 386 ovos do mosquito *Ae. aegypti*. No presente trabalho, mesmo com um número mais expressivo de objetos e imagens, o valor observado de acurácia foi bastante próximo, de modo a ser, assim, levemente inferior.

Por outro lado, ao considerarem-se abordagens de Aprendizagem Profunda, o modelo de contagem em imagens macroscópicas, proposto por (JAVED et al., 2023), obteve acurácia de 96,06% no conjunto de 10 imagens a que foi submetido. No âmbito deste trabalho o RF-DETR-B, alcançou 97,23% na mesma métrica tendo por base o total de ovos contados, conforme Tabela 28. Isto mostra que, mesmo num conjunto bastante aumentado de imagens, distribuídas em 28 palhetas distintas, conjunto esse com presença importante de imagens ruidosas, o modelo de detecção de objetos melhor avaliado neste trabalho obteve desempenho superior ao observado no referido estudo. Pontua-se, porém, que para imagens microscópicas, (JAVED et al., 2023) obteve acurácia levemente superior à alcançada neste trabalho: 98,88%.

Ainda, no (VICENTE et al., 2024) trabalho, observou-se que o modelo *FoveaBox* atingiu maior erro proporcionalmente ao número de ovos nas imagens a que foi submetido, de maneira que, quanto mais imagens o MAE era progressivamente maior. Para imagens que continha ao menos 100 ovos, o *FoveaBox* obteve MAE de 123,25, enquanto que o RF-DETR-B, ao considerar o quantitativo total de ovos de 8.035, obteve um MAE de 91,96, conforme Tabela 31. Isto mostra que o RF-DETR-B, apesar de erros individuais em imagens ou em palhetas, obteve, nos experimentos deste trabalho desempenho superior, ainda que com um número muito mais expressivo de imagens e capturadas por *smartphone*.

Acrescenta-se que, quanto ao observado em (SILVA et al., 2021), o erro médio quadrado obtido foi de 4,25, enquanto que o melhor modelo deste trabalho, obteve o valor de 14.433,96, conforme Tabela 31. Ressalta-se a limitada amostra do referido trabalho (4 imagens e 71 ovos), enquanto que neste trabalho, 28 palhetas com 126 imagens cada, o que totaliza 3.528 imagens parciais que abrigam 8.035 ovos.

Tabela 34 – Comparativo de trabalhos na literatura para contagem de ovos de *A. aegypti*

| Abordagem | Técnica | Acurácia (%) | MAE | MSE | Número de imagens | Número de ovos |
|-----------------------------------|---------|--------------|--------------|-------------|-------------------|----------------|
| (ROCHA; BIZERRA; MAGALHÃES, 2019) | PI | 53,85 | - | - | 1 | 26 |
| (COSTA, 2017) | PI | 98,94 | - | - | 206 | 386 |
| (JAVED et al., 2023) | DL | 96,06 | - | - | 10 | 18773 |
| (VICENTE et al., 2024) | DL | - | 123,25 | - | - | - |
| (SILVA et al., 2021) | DL | - | - | 4,25 | 4 | 71 |
| RF-DETR-B | DL | 97,23 | 91,96 | 14433,96 | 3528 | 8035 |

Fonte: Elaborada pelo autor (2025).

Portanto, como sintetiza a Tabela 34, o modelo RF-DETR-B destaca-se dentre outras abordagens por obter desempenho superior às demais abordagens na métrica de MAE e segunda melhor acurácia com 97,23%, ainda que o tamanho do conjunto de imagens utilizadas nos experimentos foi bem maior (3.528). Por outro lado, em termos de quantidade de ovos, houve menos objetos (8.035) em comparação com os 18.773 ovos em (JAVED et al., 2023).

Portanto, mesmo sem utilizar dispositivos de captura melhores, tais como Estereomicroscópio Leica MC170 HD utilizado por (VICENTE et al., 2024) ou Câmera Olympus Tough TG-6 e microscópio Nikon SMZ18, como em (JAVED et al., 2023), o presente trabalho obteve desempenho próximo ou superior nas métricas observadas, mesmo utilizando um *smartphone* com sensor de 13MP para captura das fotos. Tais resultados comparativos evidenciam o desempe-

nho observado na abordagem de contagem direta com RF-DETR-B numa base de imagens capturadas por *smartphone* no contexto deste trabalho.

6 CONCLUSÃO E TRABALHOS FUTUROS

No âmbito de combate ao mosquito *Aedes aegypti*, vetor de doenças como Dengue, Zika e Chikungunya, têm sido aplicadas metodologias, dentre as quais, a de implementação de ovitrampas para obtenção de indicadores fundamentais para a tomada eficiente de ações de controle. Nesse ínterim, o processo de contagem de ovos em palhetas de ovitrampa é uma atividade trabalhosa, que pode ser simplificada com a aplicação de técnicas de visão computacional.

Este trabalho tratou de propor uma metodologia de contagem automática de ovos de *Aedes aegypti* em palhetas de ovitrampa associada à utilização da câmera de *smartphone* para captura de imagens. Também, foi criado um dispositivo de baixo custo para captura de imagens de palhetas de ovitrampa para ser utilizado em conjunto com um *smartphone* a fim de prover a obtenção de imagens, considerando requisitos importantes como captura sem regiões repetidas, distância padrão entre palheta e câmera e iluminação regular da superfície da palheta.

Ainda, foi criada uma base de dados com significativa quantidade de imagens obtidas a partir do dispositivo de captura. Por fim, foi avaliado o impacto da aplicação de pré-processamento nas imagens antes da contagem automática, como o modelo MPRNet para restauração de imagens degradadas com ruído de borramento, *Color Jitter*, equalização de histograma e CLAHE. Além disso, foram também avaliados modelos para a contagem sem a utilização de pré-processamento antes da inferência, como YOLOv10, YOLOv12, RF-DETR e D-FINE.

As contribuições desta pesquisa podem ser resumidas a:

- Definição de um protocolo de contagem de ovos de *Aedes aegypti* em palhetas de ovitrampas associado ao uso de *smartphone* para captura das imagens.
- Desenvolvimento de um dispositivo de captura de baixo custo feito com impressora 3D para ser utilizado em conjunto com um *smartphone* na obtenção de imagens de palhetas de ovitrampas.
- Criação de base de dados de imagens de ovos de *A. aegypti* em palhetas de ovitrampas para o treinamento de redes neurais de detecção de objetos.

- Avaliação do impacto da contagem automática em conjunto com a aplicação de técnicas de processamento de imagens e de rede neural de restauração de imagens.
- Utilização de redes neurais artificiais do estado da arte voltadas à detecção de objetos para o reconhecimento e contagem de ovos do mosquito em palhetas de ovitrampa no contexto de imagens ruidosas e de baixa resolução, além de comparação com resultados de trabalhos semelhantes na literatura.

Esta pesquisa resultou na obtenção de um modelo de detecção de objetos RF-DETR-B aplicado à contagem direta em imagens (sem pré-processamento) com erro global percentual de -2,77% comparado à contagem manual num conjunto de 28 palhetas e erro médio de 91,26. Ademais, a avaliação do impacto de técnicas de pré-processamento na contagem com redes neurais demonstrou desempenhos similares entre equalização de histograma e equalização adaptativa, respectivamente com erro médio de 120,04 e 127,46 também na contagem de 28 palhetas de ovitrampa. A aplicação do modelo de restauração MPRNet apresentou, no entanto, maior mAP@0.50 no geral, além de menor erro relativo percentual e menor diferença entre contagens no âmbito das técnicas de pré-processamento.

Os trabalhos futuros podem ser sintetizados em:

- Obtenção de mais imagens como base de dados para o treinamento de modelos de aprendizagem profunda.
- Validação da abordagem num conjunto ainda mais expressivo de palhetas contadas.
- Realização de experimentos em campo utilizando a metodologia proposta num fluxo completo de implementação de ovitrampas visando testá-la em situações de uso real, obter pontos de melhoria e ajustar a metodologia e seus componentes, como o dispositivo de captura.
- Desenvolver aplicativo móvel para sistematizar a captura de imagens de palhetas de ovitrampa.
- Projetar e implementar sistema de informações geográficas voltado ao monitoramento da aplicação de ovitrampas, obtenção de dados relacionados à contagem de ovos, dentre outras funcionalidades.

- Elicitar requisitos, junto à FIOCRUZ-PE e Instituto Aggeu Magalhães acerca do desenvolvimento de eventuais aplicações *web* e móveis, bem como de outras tecnologias e ferramentas de suporte no âmbito da implementação da metodologia de implantação de ovitrampas.

REFERÊNCIAS

- ACOSTA, L. M. W. Conceitos e definições de vigilância epidemiológica. *Aula Vigilância em Saúde do Curso de Especialização em Saúde da Família UFCSPA/UNASUS*, 2016.
- Agência Fiocruz de Notícias. *Fiocruz: Chikungunya*. 2023. Disponível em: <<https://agencia.fiocruz.br/chikungunya>>.
- AHARON, M.; ELAD, M.; BRUCKSTEIN, A. K-svd: An algorithm for designing overcomplete dictionaries for sparse representation. *IEEE Transactions on signal processing*, IEEE, v. 54, n. 11, p. 4311–4322, 2006.
- ANTONIOU, E.; OROVOU, E.; ANDRONIKIDI, P. E.; OROVAS, C.; RIGAS, N.; PALASKA, E.; SARELLA, A.; IATRAKIS, G.; VOYIATZAKI, C. Congenital zika infection and the risk of neurodevelopmental, neurological, and urinary track disorders in early childhood. a systematic review. *Viruses*, MDPI, v. 13, n. 8, p. 1671, 2021.
- ApX Machine Learning. *Média de Precisão (mAP) em Detecção de Objetos*. 2025. Acesso em: 30 set. 2025. Disponível em: <https://apxml.com/courses/cnns-for-computer-vision/chapter-3-object-detection-algorithms/object-detection-evaluation-metrics?utm_source=chatgpt.com>.
- ARAÚJO, T. V. B. de; XIMENES, R. A. de A.; MIRANDA-FILHO, D. de B.; SOUZA, W. V.; MONTARROYOS, U. R.; MELO, A. P. L. de; VALONGUEIRO, S.; BRAGA, C.; FILHO, S. P. B.; CORDEIRO, M. T. et al. Association between microcephaly, zika virus infection, and other risk factors in brazil: final report of a case-control study. *The Lancet infectious diseases*, Elsevier, v. 18, n. 3, p. 328–336, 2018.
- ARCHANA, R.; JEEVARAJ, P. E. Deep learning models for digital image processing: a review. *Artificial Intelligence Review*, Springer, v. 57, n. 1, p. 11, 2024.
- BARRERA, R. Recomendaciones para la vigilancia de aedes aegypti. *Biomédica*, Instituto Nacional de Salud, v. 36, n. 3, p. 454–462, 2016.
- BARTHOLOMEEUSEN, K.; DANIEL, M.; LABEAUD, D. A.; GASQUE, P.; PEELING, R. W.; STEPHENSON, K. E.; NG, L. F.; ARIËN, K. K. Chikungunya fever. *Nature Reviews Disease Primers*, Nature Publishing Group UK London, v. 9, n. 1, p. 17, 2023.
- BRAGA, I. A.; GOMES, A. D. C.; NELSON, M.; MELLO, R. D. C. G.; BERGAMASCHI, D. P.; SOUZA, J. M. P. D. Comparação entre pesquisa larvária e armadilha de oviposição, para detecção de *aedes aegypti*. *Revista da Sociedade Brasileira de Medicina Tropical*, v. 33, p. 347–353, 2000.
- BRASIL, S. N. *Samsung apresenta o novo Galaxy A03s*. 2021. Disponível em: <<https://news.samsung.com/br/samsung-apresenta-o-novo-galaxy-a03s>>.
- CHOWDHURY, J. H.; LIU, Q.; RAMANNA, S. Simple histogram equalization technique improves performance of vgg models on facial emotion recognition datasets. *Algorithms*, MDPI, v. 17, n. 6, p. 238, 2024.
- CONSOLI, R. A.; OLIVEIRA, R. L. d. *Principais mosquitos de importância sanitária no Brasil*. [S.I.]: Editora Fiocruz, 1994.

COSTA, E. D. *Desenvolvimento de aplicação computacional para detecção e contagem de ovos do Aedes aegypti utilizando processamento digital de imagens e lógica fuzzy*. Monografia (Trabalho de Conclusão de Curso) — Universidade Federal do Rio Grande do Norte, 2017.

CRUZ, I. O. *Conheça o comportamento do mosquito Aedes aegypti e entenda a razão que leva este pequeno inseto a ser taxado desta forma*. 2022. Disponível em: <<https://www.ioc.fiocruz.br/dengue/textos/oportunista.html>>.

CRUZ, I. O. *Dengue: prevenção deve priorizar a eliminação de focos do Aedes aegypti*. 2022. Disponível em: <<http://www.fiocruz.br/ioc/cgi/cgilua.exe/sys/start.htm?infoid=336&query=simple&search%5Fby%5Fauthorname=all&search%5Fby%5Ffield=tax&search%5Fby%5Fkeywords=any&search%5Fby%5Fpriority=all&search%5Fby%5Fsection=all&search%5Fby%5Fstate=all&search%5Ftext%5Foptions=all&sid=32&site=fio&text=Dengue%3A+preven%E7%E3o+deve+priorizar+a+elimina%E7%E3o+de+focos+do+Aedes+aegypti>>.

DAI, J.; LI, Y.; HE, K.; SUN, J. R-fcn: Object detection via region-based fully convolutional networks. *Advances in neural information processing systems*, v. 29, 2016.

EIRAS, Á. E.; RESENDE, M. C. Avaliação preliminar da tecnologia "mi-dengue" para o monitoramento e controle do aedes aegypti. *Cadernos de Saúde Pública*, SciELO Brasil, v. 25, p. S45–S58, 2009.

ELAD, M.; KAWAR, B.; VAKSMAN, G. Image denoising: The deep learning revolution and beyond—a survey paper. *SIAM Journal on Imaging Sciences*, SIAM, v. 16, n. 3, p. 1594–1654, 2023.

ELGENDY, M. *Deep learning for vision systems*. [S.l.]: Manning, 2020.

ESPORCATTE, L. P. G.; PORTES, A. J. F. Manifestações oculares na febre chikungunya. *Revista Brasileira de Oftalmologia*, SciELO Brasil, v. 78, p. 338–341, 2019.

EVERINGHAM, M.; GOOL, L. V.; WILLIAMS, C. K. I.; WINN, J.; ZISSERMAN, A. *The PASCAL Visual Object Classes (VOC) Challenge*. 2010.

FACCINI, L. S.; FRIEDRICH, L.; MOURA, S. K. de; MARIA, F. D. S.; BONE, S. d. S. I. de. Neurological evaluation of microcephalic children with zika syndrome and congenital cytomegalovirus infection. *ENeurologicalSci*, Elsevier, v. 29, p. 100417, 2022.

FAY, R.; ELIASON, D. A. et al. A preferred oviposition site as a surveillance method for aedes aegypti. *Mosquito News*, v. 26, n. 4, p. 531–5, 1966.

FAY, R.; PERRY, A. et al. Laboratory studies of ovipositional preferences of aedes aegypti. *Mosquito News*, v. 25, n. 3, p. 276–81, 1965.

FEITOSA, L. et al.

Sistema de contagem automática de ovos do Aedes Aegypti a partir de processamento de imagens das palhetas de Ovitrampas, 2015.

FILHO, N. d. A.; ROUQUAYROL, M. Z. Introdução à epidemiologia moderna. In: *Introdução à epidemiologia moderna*. [S.l.: s.n.], 1992. p. 184–184.

GARCES-AYALA, F.; ESCOBAR-ESCAMILLA, N.; VAZQUEZ-PICHARDO, M. et al. Aedes aegypti [a-edēz a-jiptē]. *Salud Publica Mex*, v. 56, p. 402–4, 2014.

- GARG, P.; JAIN, T. A comparative study on histogram equalization and cumulative histogram equalization. *International Journal of New Technology and Research*, Nextgen Research Publication, v. 3, n. 9, p. 263242, 2017.
- GIRALDO, M. I.; GONZALEZ-OROZCO, M.; RAJSBAUM, R. Pathogenesis of zika virus infection. *Annual Review of Pathology: Mechanisms of Disease*, Annual Reviews, v. 18, n. 1, p. 181–203, 2023.
- GONZALEZ, R. C.; WOODS, R. E. *Processamento Digital de Imagens - 3a Edição*. [S.I.]: Pearson, 2015.
- GOODFELLOW, I.; BENGIO, Y.; COURVILLE, A. *Deep Learning*. [S.I.]: MIT Press, 2016. <<http://www.deeplearningbook.org>>.
- GUSMÃO, G.; MACHADO, S. C.; RODRIGUES, M. A. A new algorithm for segmenting and counting aedes aegypti eggs in ovitraps. In: IEEE. *2009 Annual International Conference of the IEEE Engineering in Medicine and Biology Society*. [S.I.], 2009. p. 6714–6717.
- GÉRON, A. *Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow: Concepts, Tools, and Techniques to Build Intelligent Systems - 2nd Edition*. [S.I.]: O'Reilly Media, 2019.
- HAMESSE, C.; ANDREO, V.; GONZALEZ, C. R.; BEUMIER, C.; RUBIO, J.; PORCASI, X.; LOPEZ, L.; GUZMAN, C.; HELTERMAN, R.; SHIMONI, M. et al. Ovitrap monitor-online application for counting mosquito eggs and visualisation toolbox in support of health services. *Ecological Informatics*, Elsevier, v. 75, p. 102105, 2023.
- HASSELQUIST, F. *Iter Palaestinum eller Resa til Heliga Landet forrattad ifran ar 1749 til 1752*. [S.I.]: Salvii, 1757.
- Instituto Oswaldo Cruz – Fiocruz. *Manual de Ovitrampas – Fiocruz*. 2024. Disponível em: <<https://saude.rn.gov.br/upload/arquivos/202402/01155325-06-manual-ovitrampas-fiocruz.pdf>>.
- Instituto Oswaldo Cruz – Fiocruz. *Fiocruz: Aedes aegypti – vetor da dengue*. 2025. Disponível em: <<https://www.ioc.fiocruz.br/dengue/textos/aedesvetoredoenca.html>>.
- JAVED, N.; LÓPEZ-DENMAN, A. J.; PARADKAR, P. N.; BHATTI, A. Eggcountai: a convolutional neural network-based software for counting of aedes aegypti mosquito eggs. *Parasites & Vectors*, Springer, v. 16, n. 1, p. 341, 2023.
- JIAO, L.; ZHANG, F.; LIU, F.; YANG, S.; LI, L.; FENG, Z.; QU, R. A survey of deep learning-based object detection. *IEEE access*, IEEE, v. 7, p. 128837–128868, 2019.
- JONG, H. K. de; GROBUSCH, M. P. Zika virus: an overview update. *Current Opinion in HIV and AIDS*, LWW, v. 20, n. 3, p. 294–302, 2025.
- KAUFMAN, P. E. Medical and veterinary entomology. *Florida Entomologist*, BioOne, v. 102, n. 3, p. 666–666, 2019.
- LEBOV, J. F.; HOOPER, S. R.; PUGH, N.; BECKER-DREPS, S.; BOWMAN, N. M.; BROWN, L. M.; MACDONALD, P. D.; LAKSHMANANE, P.; JADI, R.; BUCARDO, F. et al. Neurological and neuropsychological sequelae of zika virus infection in children in león, nicaragua. *Revista Panamericana de Salud Pública*, SciELO Public Health, v. 46, p. e90, 2023.

- LIN, T.-Y.; MAIRE, M.; BELONGIE, S.; BOURDEV, L.; URTASUN, R.; PONCE, J. M.; ZISSERMAN, A. *Microsoft COCO: Common Objects in Context*. 2014.
- LIU, W.; ANGUELOV, D.; ERHAN, D.; SZEGEDY, C.; REED, S.; FU, C.-Y.; BERG, A. C. Ssd: Single shot multibox detector. In: SPRINGER. *Computer Vision–ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11–14, 2016, Proceedings, Part I* 14. [S.I.], 2016. p. 21–37.
- LOUNIBOS, L. P. Invasions by insect vectors of human disease. *Annual review of entomology*, Annual Reviews, Inc., v. 47, p. 233, 2002.
- LV, W.; ZHAO, Y.; CHANG, Q.; HUANG, K.; WANG, G.; LIU, Y. Rt-detrv2: Improved baseline with bag-of-freebies for real-time detection transformer. *arXiv preprint arXiv:2407.17140*, 2024.
- MAINS, J. W.; MERCER, D. R.; DOBSON, S. L. Digital image analysis to estimate numbers of aedes eggs oviposited in containers. *Journal of the American Mosquito Control Association*, v. 24, n. 4, p. 496, 2008.
- MARQUES, C. D. L.; DUARTE, A. L. B. P.; RANZOLIN, A.; DANTAS, A. T.; CAVALCANTI, N. G.; GONÇALVES, R. S. G.; JUNIOR, L. F. da R.; VALADARES, L. D. de A.; MELO, A. K. G. de; FREIRE, E. A. M. et al. Recomendações da sociedade brasileira de reumatologia para diagnóstico e tratamento da febre chikungunya. parte 1–diagnóstico e situações especiais. *Revista Brasileira de reumatologia*, Elsevier, v. 57, p. 421–437, 2017.
- MARTINEZ, S. R.; LEGALL, S. W. E. G. A comparison of surveillance systems for the dengue vector aedes aegypti in port of spain, trinidad. *Journal of the American Mosquito Control Association*, v. 14, n. 2, p. 131–136, 1998.
- MARVIN, M.; SEYMOUR, A. P. Perceptrons. Cambridge, MA: MIT Press, v. 6, n. 318-362, p. 7, 1969.
- MATHWORKS. *Contrast Limited Adaptive Histogram Equalization*. 2024. Disponível em: <<https://www.mathworks.com/help/visionhdl/ug/contrast-adaptive-histogram-equalization.html>>.
- MATHWORKS. *MathWorks: Getting Started with YOLO v4*. 2025. Disponível em: <<https://www.mathworks.com/help/vision/ug/getting-started-with-yolo-v4.html>>.
- MATHWORKS. *MATLAB: Histogram Equalization*. 2025. Disponível em: <<https://www.mathworks.com/help/images/histogram-equalization.html>>.
- Ministério da Saúde – Brasil. *Gov.br: Dengue*. 2023. Disponível em: <<https://www.gov.br/saude/pt-br/assuntos/saude-de-a-a-z/d/dengue>>.
- Ministério da Saúde – Brasil. *Gov.br: Painel de Monitoramento das Arboviroses (Aedes aegypti)*. 2024. Atualizado continuamente. Último acesso: consulte a seção “Atualização de Casos de Arboviroses”. Disponível em: <<https://www.gov.br/saude/pt-br/assuntos/saude-de-a-a-z/a/aedes-aegypti/monitoramento-das-arboviroses>>.
- Ministério da Saúde – Brasil. *Gov.br: Série histórica – Casos prováveis de dengue (2000–2023)*. 2024. Atualizado em 12 jan. 2024 às 20h35. Dados até Semana Epidemiológica 52/2023. Disponível em: <<https://www.gov.br/saude/pt-br/assuntos/saude-de-a-a-z/d/dengue/situacao-epidemiologica/serie-historica-casos-provaveis-de-dengue-2000-2023>>.

Ministério da Saúde – Brasil. *Gov.br: Brasil atinge menor número de casos e mortes por COVID-19 desde 2020*. 2025. Publicado em 7 mar. 2025 às 18h25. Dados comparativos de 2024, 2023 e preliminares até 25 fev. 2025. Disponível em: <<https://www.gov.br/saude/pt-br/assuntos/noticias/2025/marco/brasil-atinge-menor-numero-de-casos-e-mortes-por-covid-19-desde-2020>>.

Ministério da Saúde – Brasil. *Gov.br: Ciclo de vida do Aedes aegypti – do ovo ao adulto: quanto tempo temos para agir*. 2025. Disponível em: <<https://www.gov.br/saude/pt-br/assuntos/noticias/2025/janeiro/ciclo-de-vida-de-aedes-aegypti-do-ovo-ao-adulto-2013-quanto-tempo-temos-para-agir>>.

Ministério da Saúde – Brasil. *Gov.br: Dengue – entenda os sorotipos da doença e por que o tipo 3 preocupa no Brasil*. 2025. Disponível em: <<https://www.gov.br/saude/pt-br/assuntos/noticias/2025/janeiro/dengue-entenda-o-que-sao-os-sorotipos-da-doenca-e-porque-o-tipo-3-e-o-que-mais-preocupa-actualmente>>.

Ministério da Saúde – Brasil. *Gov.br: Diretrizes Nacionais para Prevenção e Controle das Arboviroses Urbanas – Vigilância Entomológica e Controle Vetorial*. 2025. Disponível em: <<https://www.gov.br/saude/pt-br/centrais-de-conteudo/publicacoes/svsa/dengue/diretrizes-nacionais-para-prevencao-e-controle-das-arboviroses-urbanas-vigilancia-entomologica-e-controle.pdf/view>>.

Ministério da Saúde – Secretaria de Vigilância em Saúde. *Boletim Epidemiológico — Volume 53, Nº 48*. 2022. <<https://www.gov.br/saude/pt-br/media/pdf/2022/dezembro/22/boletim-epidemiologico-48.pdf>>. Dezembro de 2022.

Ministério da Saúde – Secretaria de Vigilância em Saúde e Ambiente. *Boletim Epidemiológico — Volume 55, N.º 11*. 2024. <<https://www.gov.br/saude/pt-br/centrais-de-conteudo/publicacoes/boletins/epidemiologicos/edicoes/2024/boletim-epidemiologico-volume-55-no-11.pdf/view>>. Publicado em 4 jul. 2024; dados monitoram arboviroses e encerramento do COE Dengue e outras Arboviroses.

MUNDIM-POMBO, A. P. M.; CARVALHO, H. J. C. d.; RIBEIRO, R. R.; LEÓN, M.; MARIA, D. A.; MIGLINO, M. A. *Aedes aegypti: egg morphology and embryonic development*. *Parasites & Vectors*, BioMed Central, v. 14, n. 1, p. 1–12, 2021.

MUNGRA, D.; AGRAWAL, A.; SHARMA, P.; TANWAR, S.; OBAIDAT, M. S. *Pratit: a cnn-based emotion recognition system using histogram equalization and data augmentation*. *Multimedia Tools and Applications*, Springer, v. 79, n. 3, p. 2285–2307, 2020.

MUSTAFA, W. A.; KADER, M. M. M. A. *A review of histogram equalization techniques in image enhancement application*. In: IOP PUBLISHING. *Journal of Physics: Conference Series*. [S.I.], 2018. v. 1019, p. 012026.

NAH, S.; KIM, T. H.; LEE, K. M. *Deep multi-scale convolutional neural network for dynamic scene deblurring*. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. [S.I.: s.n.], 2017. p. 3883–3891.

NARANJO-ALCAZAR, J.; GRAU-HARO, J.; ZUCCARELLO, P.; ALMENAR, D.; LOPEZ-BALLESTER, J. *Automatic counting and classification of mosquito eggs in field traps*. *arXiv preprint arXiv:2405.20656*, 2024.

NDE-Ed. *NDE-Ed: Real-Time Radiography — Image Processing Techniques*. 2000. Disponível em: <https://www.nde-ed.org/NDETechniques/Radiography/AdvancedTechniques/Real-Time_Radiography/ImageProcessingTechniques.xhtml>.

Neuralception. *Média de Precisão (mAP) em Detecção de Objetos*. 2024. Acesso em: 30 set. 2025. Disponível em: <https://www.neuralception.com/objectdetection-evaluationmetrics/?utm_source=chatgpt.com>.

OPENCV. *OpenCV: Histogram Equalization*. 2025. Disponível em: <https://docs.opencv.org/4.x/d5/daf/tutorial_py_histogram_equalization.html>.

OPENCV. *OpenCV: Histogram Equalization (Version 3.4)*. 2025. Disponível em: <https://docs.opencv.org/3.4/d4/d1b/tutorial_histogram_equalization.html>.

OPENCV. *OpenCV: Histogram Equalization (Version 4.x)*. 2025. Disponível em: <https://docs.opencv.org/4.x/d5/daf/tutorial_py_histogram_equalization.html>.

OQUAB, M.; DAR CET, T.; MOUTAKANNI, T.; VO, H.; SZA FRANIEC, M.; KHALIDOV, V.; FERNANDEZ, P.; HAZIZA, D.; MASSA, F.; EL-NOUBY, A. et al. Dinov2: Learning robust visual features without supervision. *arXiv preprint arXiv:2304.07193*, 2023.

ORGANIZATION, W. H. et al. *Vector control operations framework for Zika virus*. [S.I.], 2016.

PAIXAO, K. da S. Avaliação do controle químico de adultos de aedes (stegomyia) aegypti (linnaeus, 1762)(diptera: Culicidae) de fortaleza por meio de métodos convencionais e das armadilhas bg-sentinel® e mosquitrap®. Universidade Federal de Minas Gerais, 2007.

PENG, Y.; LI, H.; WU, P.; ZHANG, Y.; SUN, X.; WU, F. D-fine: redefine regression task in detrs as fine-grained distribution refinement. *arXiv preprint arXiv:2410.13842*, 2024.

PEPIN, K. M.; MARQUES-TOLEDO, C.; SCHERER, L.; MORAIS, M. M.; ELLIS, B.; EIRAS, A. E. Cost-effectiveness of novel system of mosquito surveillance and control, brazil. *Emerging infectious diseases*, v. 19, n. 4, p. 542, 2013.

PEREIRA, H. V. F. S.; SANTOS, S. P. D.; AMÂNCIO, A. P. R. L.; OLIVEIRA-SZEJNFELD, P. S. de; FLOR, E. O.; TAVARES, J. de S.; FERREIRA, R. V. B.; TOVAR-MOLL, F.; AMORIM, M. M. R. de; MELO, A. Neurological outcomes of congenital zika syndrome in toddlers and preschoolers: a case series. *The Lancet Child & Adolescent Health*, Elsevier, v. 4, n. 5, p. 378–387, 2020.

PERICH, M.; KARDEC, A.; BRAGA, I.; PORTAL, I.; BURGE, R.; ZEICHNER, B.; BROGDON, W.; WIRTZ, R. Field evaluation of a lethal ovitrap against dengue vectors in brazil. *Medical and veterinary entomology*, Wiley Online Library, v. 17, n. 2, p. 205–210, 2003.

PESSÔA, S. B.; MARTINS, A. Parasitologia médica. 11* ed. Rio de Janeiro: Guanabara Koo-gan, 1982.

PETERS, J. F. *Foundations of computer vision: computational geometry, visual image structures and object shape detection*. [S.I.]: Springer, 2017. v. 124.

- PIZER, S.; JOHNSTON, R.; ERICKSEN, J.; YANKASKAS, B.; MULLER, K. Contrast-limited adaptive histogram equalization: speed and effectiveness. In: *[1990] Proceedings of the First Conference on Visualization in Biomedical Computing*. [S.l.: s.n.], 1990. p. 337–345.
- PORTELLA, N. M. *CONTAGEM AUTOMÁTICA DE OVOS DE AEDES AEGYPTI EM IMAGENS DE OVITRAMPAS*. Monografia (Trabalho de Conclusão de Curso) — Departamento de Sistemas e Computação, Universidade de Pernambuco, Recife, 2009.
- POWELL, J. R.; TABACHNICK, W. J. History of domestication and spread of aedes aegypti-a review. *Memórias do Instituto Oswaldo Cruz*, SciELO Brasil, v. 108, p. 11–17, 2013.
- PRATT, W. K. *Digital image processing*. [S.I.]: Wiley-interscience, 2007.
- PYTORCH. *torchvision.transforms.ColorJitter*. 2025. Disponível em: <<https://docs.pytorch.org/vision/main/generated/torchvision.transforms.ColorJitter.html>>.
- PYTORCH. *torchvision.transforms.RandomHorizontalFlip*. 2025. Disponível em: <<https://docs.pytorch.org/vision/main/generated/torchvision.transforms.RandomHorizontalFlip.html>>.
- PYTORCH. *torchvision.transforms.v2.RandomIoUCrop*. 2025. Disponível em: <<https://docs.pytorch.org/vision/0.19/generated/torchvision.transforms.v2.RandomIoUCrop.html>>.
- PYTORCH. *torchvision.transforms.v2.RandomPhotometricDistort*. 2025. Disponível em: <<https://docs.pytorch.org/vision/main/generated/torchvision.transforms.v2.RandomPhotometricDistort.html>>.
- PYTORCH. *torchvision.transforms.v2.RandomZoomOut*. 2025. Disponível em: <<https://docs.pytorch.org/vision/0.16/generated/torchvision.transforms.v2.RandomZoomOut.html>>.
- RABE, I. B.; HILLS, S. L.; HAUSSIG, J. M.; WALKER, A. T.; SANTOS, T. D.; MARTIN, J. L. S.; GUTIERREZ, G.; MENDEZ-RICO, J.; RODRIGUEZ, J. C.; ELIZONDO-LOPEZ, D. et al. A review of the recent epidemiology of zika virus infection. *The American journal of tropical medicine and hygiene*, v. 112, n. 5, p. 1026, 2025.
- RAJAEI, B.; RAJAEI, S.; DAMAVANDI, H. An Analysis of Multi-stage Progressive Image Restoration Network (MPRNet). *Image Processing On Line*, v. 13, p. 140–152, 2023. <<https://doi.org/10.5201/itol.2023.446>>.
- RAJAPAKSE, S.; RODRIGO, C.; RAJAPAKSE, A. Atypical manifestations of chikungunya infection. *Transactions of the Royal Society of Tropical Medicine and Hygiene*, Royal Society of Tropical Medicine and Hygiene, v. 104, n. 2, p. 89–96, 2010.
- RIM, J.; LEE, H.; WON, J.; CHO, S. Real-world blur dataset for learning and benchmarking deblurring algorithms. In: SPRINGER. *Computer vision–ECCV 2020: 16th European conference, glasgow, UK, August 23–28, 2020, proceedings, part XXV 16*. [S.I.], 2020. p. 184–201.
- ROCHA, C. D. F. da; BIZERRA, A. M. C.; MAGALHÃES, D. A. Contagem automática de ovos do aedes aegypti em palhetas de ovitrampas: um sistema para aquisição e processamento de imagens. *Encontro Internacional de Jovens Investigadores (JOIN)*, p. 67–74, 2019.

- RONNEBERGER, O.; FISCHER, P.; BROX, T. U-net: Convolutional networks for biomedical image segmentation. In: SPRINGER. *International Conference on Medical image computing and computer-assisted intervention*. [S.I.], 2015. p. 234–241.
- ROSENBLATT, F. The perceptron: a probabilistic model for information storage and organization in the brain. *Psychological review*, American Psychological Association, v. 65, n. 6, p. 386, 1958.
- SAIFULLAH, S.; DREŽEWSKI, R. Advanced medical image segmentation enhancement: a particle-swarm-optimization-based histogram equalization approach. *Applied sciences*, MDPI, v. 14, n. 2, p. 923, 2024.
- SAMSUNG. *Galaxy A03s - Overview and Specifications*. 2021. Disponível em: <<https://www.samsung.com/us/mobile/phones/galaxy-a/galaxy-a03s-unlocked-sm-a037uzkdxaa/>>.
- SAMSUNG. *Galaxy A03s - Spec Sheet*. 2021. Disponível em: <https://image-us.samsung.com/SamsungUS/samsungbusiness/pdf/spec-sheets/Galaxy_A03s_Spec_Sheet_VZN.pdf>.
- SANTANA, C. J. de; FIRMO, A. C. A.; OLIVEIRA, R. F. A. P. de; LINS, P. J. B.; LIMA, G. A. de; LIMA, R. A. de. A solution for counting aedes aegypti and aedes albopictus eggs in paddles from ovitraps using deep learning. *IEEE Latin America Transactions*, IEEE, v. 17, n. 12, p. 1987–1994, 2019.
- SANTO, S. de Estado da Saúde do E. *SESA-ES: Mosquito Aedes aegypti*. 2025. Disponível em: <<https://mosquito.saude.es.gov.br/aedes-aegypti>>.
- SANTOS, W.; MELLO, C.; MESQUITA, R.; RODRIGUES, M.; CANDEIAS, A.; GUSMÃO, C. Um algoritmo para contagem automática de ovos do mosquito aedes aegypti em ovitrampas para controle da dengue. In: *CBEB2008: Anais do 21º Congresso Brasileiro de Engenharia Biomédica*. [S.I.: s.n.], 2008.
- SAPKOTA, R.; CHEPPALLY, R. H.; SHARDA, A.; KARKEE, M. Rf-detr object detection vs yolov12: A study of transformer-based and cnn-based architectures for single-class and multi-class greenfruit detection in complex orchard environments under label ambiguity. *arXiv preprint arXiv:2504.13099*, 2025.
- SAÚDE, S. de Vigilância em Saúde Ministério da. *Boletim Epidemiológico 33 - Óbito por arboviroses no Brasil, 2008 a 2019*. 2020. Disponível em: <<http://plataforma.saude.gov.br/anomalias-congenitas/boletim-epidemiologico-SVS-33-2020.pdf>>.
- SAÚDE, S. de Vigilância em Saúde Ministério da. *Gov.br: Série histórica de casos e óbitos de dengue (2000–2023)*. 2024. Atualizado em 12 jan. 2024. Dados até Semana Epidemiológica 52/2023. Disponível em: <<https://www.gov.br/saude/pt-br/assuntos/saude-de-a-a-z/d/dengue/situacao-epidemiologica/serie-historica-casos-de-obitos-dengue-2000-2023/view>>.
- SCHILTE, C.; STAIKOVSKY, F.; COUDERC, T.; MADEC, Y.; CARPENTIER, F.; KASSAB, S.; ALBERT, M. L.; LECUIT, M.; MICHAULT, A. Chikungunya virus-associated long-term arthralgia: a 36-month prospective longitudinal study. *PLoS neglected tropical diseases*, Public Library of Science San Francisco, USA, v. 7, n. 3, p. e2137, 2013.
- SCOTT, T. W.; MORRISON, A. C. Vector dynamics and transmission of dengue virus: implications for dengue surveillance and prevention strategies: vector dynamics and dengue prevention. *Dengue virus*, Springer, p. 115–128, 2009.

- SCOTT, T. W.; MORRISON, A. C. Longitudinal field studies will guide a paradigm shift in dengue prevention. *Vector biology, ecology and control*, Springer, p. 139–161, 2010.
- Secretaria de Estado da Saúde do Espírito Santo. *SESA-ES: A doença (Arboviroses)*. 2015. Disponível em: <<https://mosquito.saude.es.gov.br/a-doenca>>.
- SHANMUGAMANI, R. *Deep Learning for Computer Vision: Expert techniques to train advanced neural networks using TensorFlow and Keras*. [S.I.]: Packt Publishing Ltd, 2018.
- SHEN, Z.; WANG, W.; LU, X.; SHEN, J.; LING, H.; XU, T.; SHAO, L. Human-aware motion deblurring. In: *Proceedings of the IEEE/CVF international conference on computer vision*. [S.I.: s.n.], 2019. p. 5572–5581.
- SILVA, L. M. de S.; SILVA, L. L.; BARROS, V.; ARAUJO, F. H. Contagem de ovos do aedes aegypti em palhetas de ovitrampas baseada em deep learning. In: SBC. *Encontro Unificado de Computação do Piauí (ENUCOMPI)*. [S.I.], 2021. p. 128–135.
- SILVA, M. G. N. M. d.; RODRIGUES, M. A. B.; ARAUJO, R. E. d. Sistema de aquisição e processamento de imagens de ovitrampas para o combate a dengue. *Revista Brasileira de Engenharia Biomédica*, SciELO Brasil, v. 28, p. 364–374, 2012.
- SILVA, M. G. N. M. d.; RODRIGUES, M. A. B.; ARAUJO, R. E. d. System for acquisition and processing of ovitraps images to fight dengue. *Revista Brasileira de Engenharia Biomédica*, SciELO Brasil, v. 28, p. 364–374, 2012.
- SILVA, R. E. V. da. *Contagem automática de ovos do mosquito Aedes aegypti utilizando métodos de aprendizagem profunda e dispositivo de baixo custo*. Dissertação (Mestrado) — Universidade Federal de Pernambuco, 2021.
- SINGH, S.; MANIMUNDA, S.; SUGUNAN, A.; VIJAYACHARI, P. et al. Four cases of acute flaccid paralysis associated with chikungunya virus infection. *Epidemiology & Infection*, Cambridge University Press, v. 136, n. 9, p. 1277–1280, 2008.
- SOLOMON, C.; BRECKON, T. *Fundamentals of Digital Image Processing: A practical approach with examples in Matlab*. [S.I.]: John Wiley & Sons, 2011.
- SUIN, M.; PUROHIT, K.; RAJAGOPALAN, A. Spatially-attentive patch-hierarchical network for adaptive motion deblurring. In: *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. [S.I.: s.n.], 2020. p. 3606–3615.
- SuperAnnotate. *Mean Average Precision (mAP) em Detecção de Objetos*. 2023. Acesso em: 30 set. 2025. Disponível em: <https://www.superannotate.com/blog/mean-average-precision-and-its-uses-in-object-detection?utm_source=chatgpt.com>.
- SZELISKI, R. *Computer vision: algorithms and applications*. [S.I.]: Springer Nature, 2022.
- TAN, M.; PANG, R.; LE, Q. V. Efficientdet: Scalable and efficient object detection. In: *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. [S.I.: s.n.], 2020. p. 10781–10790.
- THIBERVILLE, S.-D.; MOYEN, N.; DUPUIS-MAGUIRAGA, L.; NOUGAIREDE, A.; GOULD, E. A.; ROQUES, P.; LAMBALLERIE, X. de. Chikungunya fever: epidemiology, clinical syndrome, pathogenesis and therapy. *Antiviral research*, Elsevier, v. 99, n. 3, p. 345–370, 2013.

- TIAN, Y.; YE, Q.; DOERMANN, D. Yolov12: Attention-centric real-time object detectors. *arXiv preprint arXiv:2502.12524*, 2025.
- TZUTALIN. *LabelImg: Graphical image annotation tool*. 2015. <<https://github.com/HumanSignal/labelImg>>. Accessed: 2025-09-30.
- ULTRALYTICS. *Ultralytics: YOLOv10 — Real-Time End-to-End Object Detection*. 2024. Disponível em: <<https://docs.ultralytics.com/models/yolov10/>>.
- ULTRALYTICS. *Ultralytics Glossary: F1 Score*. 2025. Disponível em: <<https://www.ultralytics.com/glossary/f1-score>>.
- ULTRALYTICS. *Ultralytics Glossary: Mean Average Precision (mAP)*. 2025. Disponível em: <<https://www.ultralytics.com/glossary/mean-average-precision-map>>.
- ULTRALYTICS. *Ultralytics: YOLO Performance Metrics*. 2025. Disponível em: <<https://docs.ultralytics.com/guides/yolo-performance-metrics/#introduction>>.
- ULTRALYTICS. *YOLO Configuration: Train Settings*. 2025. Disponível em: <<https://docs.ultralytics.com/pt/usage/cfg/#train-settings>>.
- ULTRALYTICS. *YOLO Data Augmentation: AutoAugment*. 2025. Disponível em: <https://docs.ultralytics.com/pt/guides/yolo-data-augmentation/#auto-augment-auto_augment>.
- VICENTE, M. N. d. O.; HIGA, G. T. H.; PORTO, J. V. d. A.; HENRIQUE, H.; NUCCI, P.; SANTANA, A. B.; PORTO, K. R. d. A.; ROEL, A. R.; PISTORI, H. Aedes aegypti egg counting with neural networks for object detection. *arXiv preprint arXiv:2403.08016*, 2024.
- VILLA, K. R. Trabajo Fin de Grado, *Vigilancia entomológica*. 2020.
- VIVEIROS, J. W. S. de; SOUZA, J. C. F. de; OLIVEIRA, M. C. V.; CRISTINO, S. N.; ALMEIDA, G. K. S.; SANTOS, Y. M.; SAMPAIO, A. F.; COTA, M. E. D. V. Desfechos neurológicos em crianças expostas ao zika vírus: uma revisão da literatura. *Caderno Pedagógico*, v. 22, n. 1, p. e13530–e13530, 2025.
- WANG, A.; CHEN, H.; LIU, L.; CHEN, K.; LIN, Z.; HAN, J. et al. Yolov10: Real-time end-to-end object detection. *Advances in Neural Information Processing Systems*, v. 37, p. 107984–108011, 2024.
- World Health Organization. *WHO: Zika virus*. 2022. Disponível em: <<https://www.who.int/news-room/fact-sheets/detail/zika-virus>>.
- ZAMIR, S. W.; ARORA, A.; KHAN, S.; HAYAT, M.; KHAN, F. S.; YANG, M.-H.; SHAO, L. Multi-stage progressive image restoration. In: *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. [S.l.: s.n.], 2021. p. 14821–14831.
- ZENG, L.; LI, Z.; LIU, H.; LI, Y. CJT: Boosting the Transferability of Adversarial Attacks with Color Jitter Transformation. In: *2023 International Conference on High Performance Big Data and Intelligent Systems (HDIS)*. [S.l.]: IEEE, 2023. p. 218–225.
- ZHANG, H.; DAI, Y.; LI, H.; KONIUSZ, P. Deep stacked hierarchical multi-patch network for image deblurring. In: *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. [S.l.: s.n.], 2019. p. 5978–5986.

- ZHAO, Y.; LV, W.; XU, S.; WEI, J.; WANG, G.; DANG, Q.; LIU, Y.; CHEN, J. Detrs beat yolos on real-time object detection. In: *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. [S.l.: s.n.], 2024. p. 16965–16974.
- ZHU, X.; SU, W.; LU, L.; LI, B.; WANG, X.; DAI, J. Deformable detr: Deformable transformers for end-to-end object detection. *arXiv preprint arXiv:2010.04159*, 2020.
- ZUIDERVELD, K. Contrast limited adaptive histogram equalization. In: *Graphics gems IV*. [S.l.: s.n.], 1994. p. 474–485.