

UNIVERSIDADE FEDERAL DE VIÇOSA

**Quantificação do volume sólido de madeira de eucalipto utilizando visão
computacional**

Gianmarco Goycochea Casas
Doctor Scientiae

**VIÇOSA - MINAS GERAIS
2026**

GIANMARCO GOYCOCHEA CASAS

Quantificação do volume sólido de madeira de eucalipto utilizando visão computacional

Tese apresentada à Universidade Federal de Viçosa, como parte das exigências do Programa de Pós-Graduação em Ciência Florestal, para obtenção do título de *Doctor Scientiae*.

Orientador: Helio Garcia Leite

Coorientadores: Carlos P. B. Soares
Carlos A. Araújo Júnior
Zool Hilmi Bin Ismail

**Ficha catalográfica elaborada pela Biblioteca Central da Universidade
Federal de Viçosa - Campus Viçosa**

T

C335q
2026

Casas, Gianmarco Goycochea, 1991-
Quantificação do volume sólido de madeira de eucalipto
utilizando visão computacional / Gianmarco Goycochea Casas. –
Viçosa, MG, 2026.
1 tese eletrônica (234 f.): il. (algumas color.).

Orientador: Hélio Garcia Leite.
Tese (doutorado) - Universidade Federal de Viçosa,
Departamento de Engenharia Florestal, 2026.
Inclui bibliografia.
DOI: <https://doi.org/10.47328/ufvbbt.2026.541>
Modo de acesso: World Wide Web.

1. Madeira - Medição. 2. Madeira serrada -
Armazenamento. 3. Visão por computador. I. Leite, Hélio
Garcia, 1960-. II. Universidade Federal de Viçosa. Departamento
de Engenharia Florestal. Programa de Pós-Graduação em
Ciência Florestal. III. Título.

GFDC adapt. CDD 22. ed. 634.9365

GIANMARCO GOYCOCHEA CASAS

Quantificação do volume sólido de madeira de eucalipto utilizando visão computacional

Tese apresentada à Universidade Federal de Viçosa, como parte das exigências do Programa de Pós-Graduação em Ciência Florestal, para obtenção do título de *Doctor Scientiae*.

APROVADA: 3 de setembro de 2026.

Assentimento:

Gianmarco Goycochea Casas
Autor

Helio Garcia Leite
Orientador

Essa tese foi assinada digitalmente pelo autor em 16/09/2026 às 11:12:17 e pelo orientador em 24/09/2026 às 22:29:50. As assinaturas têm validade legal, conforme o disposto na Medida Provisória 2.200-2/2001 e na Resolução nº 37/2012 do CONARQ. Para conferir a autenticidade, acesse <https://siadoc.ufv.br/validar-documento>. No campo 'Código de registro', informe o código **WMWI.E9BW.FDKX** e clique no botão 'Validar documento'.

DEDICO

A Deus, Todo-Poderoso, pela vida.

À minha mãe, Mirtha Casas, e ao meu pai, Victor Goycochea (in memoriam).

Ao meu irmão, Paul Goycochea.

Aos meus avós, Justo Casas (in memoriam) e Irma Perez.

À minha família mais próxima, Syafiqah, Beba e Baby.

AGRADECIMENTOS

À Universidade Federal de Viçosa (UFV) e à Coordenação de Aperfeiçoamento de Pessoal de Nível Superior (CAPES), pela oportunidade e pelo apoio financeiro para a realização deste trabalho.

Ao professor Hélio Garcia Leite, pela orientação, paciência, dedicação e elevado profissionalismo demonstrados durante o desenvolvimento desta pesquisa e de outros trabalhos ao longo da minha formação.

Ao professor Zool Hilmi Ismail e ao professor Mohd Ibrahim Shapiai, do Center for Artificial Intelligence and Robotics (CAIRO), da Universiti Teknologi Malaysia (UTM), pela orientação, confiança, incentivo e elevado profissionalismo no desenvolvimento de pesquisas e projetos científicos. Agradeço pelas valiosas contribuições, pelo compartilhamento de conhecimentos e pela colaboração que fortaleceram significativamente esta pesquisa.

Aos funcionários do Departamento de Engenharia Florestal, em especial aos secretários da pós-graduação, Dilson e Alexandre, ao porteiro Chiquinho e à equipe de limpeza do departamento, pelo apoio, dedicação e cordialidade.

Aos funcionários da Diretoria de Relações Internacionais (DRI) da UFV, pela atenção e pelo suporte prestado durante os processos relacionados à documentação internacional.

Aos colegas do Programa de Pós-Graduação em Ciência Florestal, especialmente Manu, Caro, Kemely, Mathaus, Silas e Pedro, pela amizade, convivência e apoio durante esta trajetória.

Ao professor Marinaldo e aos integrantes do GT Manejo, pelas discussões, pela troca de experiências e pelo aprendizado compartilhado ao longo deste período.

Aos funcionários responsáveis pela manutenção das áreas da UFV, especialmente das áreas verdes e da infraestrutura do campus. À natureza que envolve a universidade (o céu, a lua, o sol, as aves e as nuvens), por proporcionar um ambiente inspirador para o desenvolvimento deste trabalho.

À minha família, pelo incentivo e apoio constantes, especialmente aos meus pais, Mirtha Casas e Victor Goycochea (in memoriam); ao meu irmão, Paul Goycochea; aos meus tios Silva Casas, Ramon Burga, Edgar Casas (in memoriam), Maritza, Walter Casas e Luz (in memoriam); e aos meus primos Luis e Ramon Burga, Eric e Evelyn Casas, Walter e Toño Casas.

Este trabalho foi realizado com o apoio das seguintes agências de pesquisa brasileiras: Coordenação de Aperfeiçoamento de Pessoal de Nível Superior – Brasil (CAPES) – Código de Financiamento 001, Fundação de Amparo à Pesquisa do Estado de Minas Gerais (FAPEMIG) e Conselho Nacional de Desenvolvimento Científico e Tecnológico (CNPq).

Às demais pessoas que, de forma direta ou indireta, contribuíram para a realização deste trabalho, os meus sinceros agradecimentos.

O importante é não parar de questionar. A curiosidade tem a sua própria razão para existir

Albert Einstein

Wa kalimatu Allahi hiya al-ulya

RESUMO

CASAS, Gianmarco Goycochea, D.Sc., Universidade Federal de Viçosa, setembro de 2026. **Quantificação do volume sólido de madeira de eucalipto utilizando visão computacional.** Orientador: Helio Garcia Leite. Coorientadores: Carlos Pedro Boechat Soares, Carlos Alberto Araújo Júnior e Zool Hilmi Bin Ismail.

O Capítulo 1 fundamenta a evolução e o funcionamento dos detectores de uma passada, descrevendo como as versões iniciais do YOLO unificaram extração de atributos e predição de caixas e classes em um único fluxo, e como convoluções, pooling e blocos totalmente conectados permitem ao backbone aprender mapas de características que alimentam a cabeça de detecção. O Capítulo 2 apresenta um mapeamento sistemático de 1.019 artigos indexados na base Scopus entre 1983 e 2026. O controle de qualidade de superfície concentra 48,3% da produção, as abordagens clássicas respondem por 73,6% do corpus frente a 26,4% do aprendizado profundo e 63,7% dos artigos são posteriores a 2016. As aplicações de mensuração, entre as quais se insere a quantificação de madeira empilhada, somam menos de 9% da literatura, lacuna que delimita o problema tratado adiante. O Capítulo 3 descreve a montagem do conjunto de dados, o treinamento com YOLOv8 e a validação em campo com toras isoladas e madeira empilhada. Em pilhas reais, as imagens estáticas apresentam subestimativa consistente, com erro relativo entre -32,252% e -47,465%, enquanto o vídeo alcança +1,561% no total da pilha, sem viés estatisticamente detectável. O vídeo é, portanto, preferível para a quantificação do total da pilha, ao passo que fotografias com configuração de inferência ajustada podem ser utilizadas na mensuração de trechos delimitados. O Capítulo 4 introduz a ponte estatística que converte contagem global em volume sólido: estabelece 35 classes diamétricas, calcula área basal e volume por fórmulas consolidadas no contexto florestal e seleciona o modelo probabilístico entre distribuições ajustadas por máxima verossimilhança e ranqueadas por AIC e BIC. A Generalized Extreme Value mostrou-se adequada às assimetrias observadas, com parâmetro de forma correspondente ao tipo III (Weibull), de cauda superior limitada; a combinação da probabilidade por classe com a contagem total fornece o número esperado de toras e o volume por classe. O Capítulo 5 operacionaliza o fluxo completo em um sistema reprodutível que integra detecção, rastreamento, modelagem diamétrica e cálculo de volume, adotando YOLOv9 em vídeo acoplado ao ByteTrack, que associa detecções quadro a quadro e busca manter identificadores persistentes por tora, evitando duplicidades. O Capítulo 6 apresenta o NeuroTimber Mobile, versão concebida para operar dentro de um aplicativo de mensageria, que

combina a segmentação de contexto pelo Segment Anything Model 3 (SAM 3), a segmentação de instâncias das faces das toras pelo YOLO26 e uma calibração métrica fornecida pelo operador para converter, pela fórmula de Smalian, a área observada em volume sólido; em 30 pilhas de *Eucalyptus grandis*, obteve erro médio percentual absoluto de 4,66% e correlação de 0,795 com os volumes de referência. Em conjunto, os seis capítulos compõem um fluxo de trabalho auditável e ampliável, que parte da arquitetura dos detectores, delimita a lacuna da literatura, diferencia fotografia e vídeo na contagem em pilhas, estabelece metodologia probabilística transparente para transportar contagem a volume e culmina em implementações integradas para uso em campo.

Palavras-chave: detecção de objetos; contagem de toras; distribuição de diâmetros; pilhas de madeira; rastreamento em vídeo; YOLO

ABSTRACT

CASAS, Gianmarco Goycochea, D.Sc., Universidade Federal de Viçosa, September, 2026. **Quantification of solid eucalypt wood volume using computer vision.** Adviser: Helio Garcia Leite. Co-advisers: Carlos Pedro Boechat Soares, Carlos Alberto Araújo Júnior and Zool Hilmi Bin Ismail.

Chapter 1 lays out the evolution and mechanics of single-shot detectors, describing how the early YOLO versions unified feature extraction and box and class prediction in a single pass, and how convolutions, pooling, and fully connected blocks let the backbone learn the feature maps that feed the detection head. Chapter 2 presents a systematic mapping of 1,019 Scopus-indexed papers published between 1983 and 2026. Surface quality control accounts for 48.3% of the output, classical approaches for 73.6% of the corpus against 26.4% for deep learning, and 63.7% of the papers postdate 2016. Measurement applications, which include the quantification of stacked wood, account for less than 9% of the literature, a gap that delimits the problem addressed in the following chapters. Chapter 3 describes dataset assembly, YOLOv8 training, and field validation with isolated logs and stacked wood. On real stacks, static images show consistent underestimation, with relative error between -32.252% and -47.465%, whereas video reaches +1.561% for the stack total, with no statistically detectable bias. Video is therefore preferable for quantifying the stack as a whole, whereas photographs with an adjusted inference configuration can be used to measure delimited sections. Chapter 4 introduces the statistical bridge that converts a global count into solid volume: it establishes 35 diameter classes, computes basal area and volume with formulas consolidated in the forestry context, and selects the probabilistic model among distributions fitted by maximum likelihood and ranked by AIC and BIC. The Generalized Extreme Value distribution proved suitable to the observed skewness, with a shape parameter matching the type III (Weibull) member of the family, with a bounded upper tail; combining the per-class probability with the total count yields the expected number of logs and the volume per class. Chapter 5 operationalizes the whole workflow in a reproducible system integrating detection, tracking, diameter modeling, and volume calculation, adopting YOLOv9 on video coupled with ByteTrack, which associates detections frame to frame and seeks to maintain persistent identifiers per log, avoiding duplicates. Chapter 6 presents NeuroTimber Mobile, a version designed to run within a messaging application, combining context segmentation by the Segment Anything Model 3 (SAM 3), instance segmentation of the log faces by YOLO26, and a metric calibration

supplied by the operator to convert, through Smalian's formula, the observed wood area into solid volume; on 30 stacks of *Eucalyptus grandis* it achieved a mean absolute percentage error of 4.66% and a correlation of 0.795 with the reference volumes. Taken together, the six chapters compose an auditable and extensible workflow that starts from detector architecture, delimits the gap in the literature, distinguishes photographs from video for counting in stacks, establishes a transparent probabilistic method to translate counts into volume, and culminates in integrated implementations for field use.

Keywords: object detection; log counting; diameter distribution; wood piles; video tracking; YOLO

LISTA DE FIGURAS

1.1	Inspiração biológica do funcionamento do cérebro humano aplicada na visão computacional	37
1.2	Exemplo de processo de convolução em processamento de imagem	39
1.3	Exemplo de aplicação de pooling máximo e médio com a função de reduzir a resolução dos mapas de recursos	40
1.4	Exemplo de camadas totalmente conectadas em redes neurais convolucionais .	41
1.5	Processo de classificação de imagens com rede neural convolucional	42
1.6	Gráfico da função Rectified Linear Unit (ReLU)	43
1.7	Descrição do processo de detecção de objetos do modelo YOLO	48
1.8	Exemplo de formato de anotação e divisão de conjunto de dados em uma imagem de madeira de eucalipto empilhada	49
1.9	Exemplo dos diferentes desempenhos existentes na avaliação da detecção de objetos usando a Interseção sobre União (IoU)	51
1.10	Exemplo de detecção como verdadeiro positivo, falso positivo e falso negativo em diferentes limiares de IoU	52
1.11	Curva de relação entre precisão e recall em diferentes limites de IoU	53
1.12	Métricas de precisão, recuperação, comparação mAP@50 e mAP@50:95 avaliados no conjunto de dados de teste no limite de IoU variando de 20% a 95% . .	54
1.13	Exemplo da Curva de Confiança F1, representando o desempenho de um modelo de classificação	55
1.14	Evolução histórica e progressiva do modelo YOLO na visão computacional . .	57
1.15	A imagem à esquerda mostra a saída típica de um modelo de detecção de objetos contendo várias caixas sobrepostas e a imagem à direita mostra a saída após o algoritmo NMS	59
1.16	Desenvolvimento inicial da arquitetura para o YOLO V1	60
1.17	Descrição das características da primeira versão do modelo YOLO	62
1.18	Descrição das características da segunda versão do modelo YOLO	63
1.19	Arquitetura de detecção multi-escala do YOLO V3	66
1.20	Descrição das características da terceira versão do modelo YOLO	66
1.21	Descrição das características da quarta versão do modelo YOLO	71
1.22	Diagrama do processo Backbone-Neck-Head de detecção em uma e duas fases do modelo YOLO V4	72

1.23	Desenvolvimento da arquitetura do modelo YOLO V5	74
1.24	Descrição das características da quinta versão do modelo YOLO	74
1.25	Descrição das características da sexta versão do modelo YOLO	75
1.26	Descrição das características da sétima versão do modelo YOLO	77
1.27	Descrição das características da oitava versão do modelo YOLO	78
1.28	Arquitetura detalhada do modelo YOLOV8	81
1.29	Descrição das características da nona versão do modelo YOLO	82
1.30	Síntese da arquitetura e das inovações do YOLO V10	83
1.31	Síntese da arquitetura e dos blocos principais do YOLO V11	84
1.32	Síntese da arquitetura centrada em atenção do YOLO V12	85
1.33	Síntese da arquitetura e do pipeline de treinamento do YOLO26	87
1.34	Esquema unificado da estrutura comum e da genealogia das versões YOLO, destacando o esqueleto Backbone-Neck-Head compartilhado	94
2.1	Fluxo PRISMA de seleção dos estudos	108
2.2	Produção anual, crescimento acumulado e distribuição por período	113
2.3	Tipos documentais e principais veículos de publicação	114
2.4	Distribuição geográfica dos dez países com maior número de artigos	115
2.5	Análise de citações do corpus	116
2.6	Rede de coocorrência de palavras-chave	118
2.7	Distribuição dos artigos entre as nove categorias de aplicação	119
2.8	Distribuição dos artigos por classe metodológica	123
2.9	Distribuição das classes metodológicas entre as categorias de aplicação	124
2.10	Participação da visão computacional clássica em cada categoria de aplicação	125
2.11	Adoção do aprendizado profundo	126
2.12	Composição temática do corpus em cada período	128
3.1	O banco de dados é composto por um conjunto de dois tipos de imagens uti- lizadas para treinar o modelo YOLOv8	146
3.2	Formato de anotação e divisão do conjunto de dados em uma imagem de madeira de eucalipto empilhada	149
3.3	Avaliação das diferentes curvas de desempenho do modelo treinado, incluindo a curva F1-confiança, a curva...	153
3.4	Imagens utilizadas na generalização do modelo, classificadas como toras es- paçadas com diâmetros grandes (A) e diâmetros...	154
3.5	Imagens (A-J) utilizadas na generalização do modelo para a contagem e de- tecção de madeira empilhada na indústria de...	155
3.6	Detecções na fotografia D nos limiares de 0,25 e 0,10	160
3.7	Caixas sobrepostas sobre a mesma face no limiar de 0,10	161

4.1	A) Distribuição das classes diamétricas (DAP) do inventário pré-colheita e B) exemplo da imagem de frame utilizada para...	177
4.2	A) Modelagem de afilamento de eucalipto aos 66 e 86 meses, com alturas do fuste a cada 6 m, e B) ajuste da distribuição...	180
4.3	Comparação das técnicas de segmentação e detecção no processamento de imagens	192
4.4	Guia passo a passo para estimar o volume total de madeira de eucalipto empilhada	193
5.1	Aba Detecção em Vídeo do NeuroTimber (versão web)	208
5.2	Parte superior da aba Diâmetros & Volume	210
6.1	Fluxograma de funcionamento do NeuroTimber Mobile	218
6.2	Etapas do processamento de uma pilha (P24, face A)	219
6.3	Segundo exemplo de processamento (P16, face A)	219
6.4	Avaliação do NeuroTimber Mobile nas 30 pilhas	221
6.5	NeuroTimber Mobile (@NeuroTimber_bot) - etapa 1	223
6.6	NeuroTimber Mobile - etapa 2	224
6.7	NeuroTimber Mobile - etapa 3	225
6.8	Proposta de fluxo operacional do NeuroTimber Mobile	226

LISTA DE TABELAS

1.1	Comparação das funções de ativação mais usadas em redes neurais convolucionais na etapa da rede neural totalmente conectada	45
1.2	Descrição das camadas convolutivas e totalmente conectadas, quantidade de filtros, dimensões e passos dos filtros, e dimensões das saídas na arquitetura de YOLO V1	60
1.3	Descrição das camadas convolutivas, quantidade de filtros, dimensões e passos dos filtros, e dimensões das saídas na arquitetura DarkNet-19 de YOLO V2 . .	62
1.4	Descrição das camadas convolutivas, quantidade de filtros, dimensões e passos dos filtros, e dimensões das saídas na arquitetura DarkNet-53 de YOLO V3 . .	65
1.5	Parâmetros que definem as diferentes variantes do modelo YOLOV8	80
2.1	Distribuição dos artigos entre as categorias de aplicação e participação das abordagens clássicas	119
3.1	Métricas de desempenho avaliadas na divisão dos dados de madeira de eucalipto empilhada	152
3.2	Dados dos arquivos e análises de erro para imagens e vídeo testados	156
3.3	Análise de sensibilidade da contagem automática aos parâmetros de inferência .	157
3.4	Contagem manual, por fotografia e por vídeo nas dez seções da pilha	163
4.1	Parâmetros de diferentes modelos de distribuições de diâmetro ajustadas	181
4.2	Comparação dos resultados dos pressupostos estatísticos das distribuições diamétricas ajustadas	183
4.3	Estimativa do volume sólido por classe diamétrica utilizando densidade de probabilidade	185
4.4	Simulação para a estimativa do volume sólido de 30 pilhas de 5000 toras	187
4.5	Relatório dos valores médios, máximos e mínimos das variáveis medidas . . .	189
4.6	Comparação do volume estimado pelo processo proposto com o volume observado obtido pelo método de Smalian	189
6.1	Volume de referência (V_{cc}), volume estimado pelo NeuroTimber Mobile, erro relativo e número de toras segmentadas...	220

LISTA DE TERMOS TÉCNICOS

A relação a seguir reúne os termos técnicos empregados ao longo do texto, organizados em quatro grupos: visão computacional e aprendizado de máquina, métricas de desempenho, estatística e distribuições diamétricas. As definições são voltadas ao leitor de formação florestal, sem pretensão de formalismo matemático.

Aprendizado profundo. Ramo do aprendizado de máquina baseado em redes neurais com muitas camadas, capazes de aprender representações cada vez mais abstratas dos dados a partir de exemplos anotados.

Backbone. Parte inicial de um detector, responsável por extrair as características visuais da imagem. É a rede convolucional que converte os pixels em mapas de características.

Neck. Parte intermediária do detector, que combina os mapas de características extraídos em diferentes escalas, de modo a permitir a detecção de objetos de tamanhos variados.

Head. Parte final do detector, que converte as características em previsões: coordenadas das caixas delimitadoras, escores de confiança e probabilidades de classe.

Caixa delimitadora (*bounding box*). Retângulo que circunscreve um objeto na imagem, definido pelas coordenadas do seu centro e por sua largura e altura.

Detector *anchor-free*. Detector que prediz diretamente a posição e as dimensões das caixas, sem partir de um conjunto fixo de caixas de referência (âncoras) previamente definidas.

Inferência. Etapa de uso do modelo já treinado, em que ele é aplicado a novas imagens para produzir previsões. Distingue-se do treinamento, em que os pesos são ajustados.

Latência de inferência. Tempo necessário para o modelo processar uma imagem e produzir suas previsões. Determina quantas imagens por segundo podem ser processadas.

Inferência *end-to-end*. Inferência em que a rede produz diretamente o conjunto final de detecções, sem etapas de pós-processamento como a supressão de não máximos.

Supressão de não máximos (NMS). Procedimento de pós-processamento que elimina caixas redundantes: quando várias caixas se sobrepõem fortemente sobre o mesmo objeto, retém-se apenas a de maior confiança.

Interseção sobre União (IoU). Medida de sobreposição entre duas caixas, calculada como a razão entre a área comum e a área total ocupada por ambas. Varia de 0, sem sobreposição, a 1, coincidência perfeita.

Limiar de confiança. Valor mínimo de confiança exigido para que uma detecção seja aceita. Limiares altos reduzem falsos positivos e aumentam detecções omitidas; limiares baixos produzem o efeito inverso.

Mecanismos de atenção. Componentes que permitem à rede ponderar de forma não uniforme diferentes regiões da imagem, concentrando o processamento nas mais informativas para a tarefa.

Data augmentation. Ampliação artificial do conjunto de treinamento por meio de transformações das imagens disponíveis, como rotações, recortes e alterações de brilho, com o objetivo de aumentar a capacidade de generalização.

Dados anotados. Imagens acompanhadas da marcação manual dos objetos de interesse, que constituem a referência a partir da qual o modelo aprende e contra a qual seu desempenho é avaliado.

Conjuntos de treinamento, validação e teste. Partições do conjunto de dados: o primeiro é usado para ajustar os pesos, o segundo para acompanhar o desempenho durante o treinamento e o terceiro, mantido à parte, para a avaliação final.

Época (epoch). Uma passagem completa do algoritmo de treinamento por todas as imagens do conjunto de treinamento.

Early stopping. Interrupção antecipada do treinamento quando o desempenho no conjunto de validação deixa de melhorar por um número determinado de épocas, evitando o sobreajuste.

Batch. Subconjunto de imagens processado de uma só vez antes de cada atualização dos pesos do modelo.

Rastreamento (tracking). Associação das detecções obtidas em quadros consecutivos de um vídeo, de modo que cada objeto receba um identificador único mantido enquanto permanecer visível.

Segmentação de instâncias. Delimitação do contorno exato de cada objeto, e não apenas de um retângulo que o circunscreve, permitindo medir a área que ele ocupa na imagem.

Detecção de vocabulário aberto. Detecção que não se restringe a uma lista fixa de classes definida no treinamento, sendo capaz de localizar objetos descritos por texto no momento da inferência.

Mapeamento sistemático. Revisão de literatura conduzida por protocolo explícito e reproduzível, cujo objetivo é caracterizar a distribuição da produção de um campo entre temas, métodos e períodos, e não sintetizar os resultados de estudos individuais, como faz a revisão sistemática clássica.

PRISMA. Protocolo internacional de relato de revisões sistemáticas (*Preferred Reporting Items for Systematic Reviews and Meta-Analyses*), que padroniza o registro das etapas de identificação, triagem, elegibilidade e inclusão dos estudos, tornando o processo auditável.

Corpus. Conjunto final de documentos que resta após a aplicação de todos os critérios de elegibilidade e sobre o qual incidem as análises do mapeamento.

Bibliometria. Análise quantitativa da produção científica a partir de seus metadados: volume de publicações, distribuição temporal e geográfica, veículos, autoria e citações.

Coocorrência de palavras-chave. Técnica que mede quantas vezes dois termos aparecem juntos nos mesmos documentos, revelando os agrupamentos temáticos de uma literatura e as pontes entre eles.

F1 macro e F1 ponderada. Formas de resumir o escore *F1* de um classificador de várias categorias: a macro é a média simples entre as categorias, tratando todas com o mesmo peso, e a ponderada pesa cada uma pelo número de itens que contém.

Métricas de desempenho

Verdadeiro positivo, falso positivo e falso negativo. Numa tarefa de detecção, o verdadeiro positivo é a detecção que corresponde a um objeto realmente presente; o falso positivo, a detecção que não corresponde a objeto algum; e o falso negativo, o objeto presente que o modelo não detectou.

Precisão. Proporção das detecções produzidas pelo modelo que são corretas. Responde à pergunta: do que o modelo apontou, quanto estava certo?

Revocação (*recall*). Proporção dos objetos presentes que o modelo conseguiu detectar. Responde à pergunta: do que existia, quanto o modelo encontrou?

Escore F1. Média harmônica entre precisão e revocação, que resume as duas em um único valor e penaliza o desequilíbrio entre elas.

mAP50 e mAP50-95. Precisão média das detecções, calculada com limiar de sobreposição de 0,50 no primeiro caso e como média entre os limiares de 0,50 a 0,95 no segundo, sendo esta última a medida mais exigente quanto à precisão da caixa.

Erro relativo (RE). Diferença entre o valor estimado e o observado, expressa como proporção do observado. É negativo quando há subestimação e positivo quando há superestimação.

Erro médio percentual absoluto (EMPA). Média dos erros relativos tomados em valor absoluto, que mede a magnitude típica do erro sem que desvios de sinais opostos se cancelem.

Raiz do erro quadrático médio (RMSE). Medida de dispersão do erro na mesma unidade da variável estimada, que atribui peso maior aos desvios grandes.

Estatística

Viés. Tendência sistemática de um estimador a subestimar ou a superestimar o valor real. Distingue-se da imprecisão: um método pode ser preciso e enviesado, ou não enviesado e disperso.

Desvio-padrão. Medida de dispersão dos valores em torno da média, na mesma unidade da

variável.

Intervalo de confiança. Faixa de valores que, sob o procedimento adotado, contém o parâmetro de interesse com a probabilidade declarada, usualmente 95%. Quando o intervalo contém o zero, não se rejeita a ausência do efeito.

Valor-p. Probabilidade de se observar um resultado ao menos tão extremo quanto o obtido, caso a hipótese de ausência de efeito seja verdadeira. Valores baixos indicam que o resultado é pouco compatível com essa hipótese.

Teste t pareado. Teste que compara dois métodos aplicados às mesmas unidades amostrais, avaliando se a média das diferenças difere de zero.

Teste de Wilcoxon. Alternativa ao teste t pareado que se apoia nos postos das diferenças, e não em seus valores, sendo por isso menos sensível à suposição de normalidade e a valores extremos.

Coefficiente de determinação (R^2). Proporção da variação da variável resposta explicada pelo modelo ajustado.

Coefficiente de correlação de Pearson. Medida da associação linear entre duas variáveis, que varia de -1 a $+1$.

Máxima verossimilhança. Método de estimação que escolhe os parâmetros que tornam os dados observados os mais prováveis sob o modelo assumido.

AIC e BIC. Critérios de informação de Akaike e bayesiano, que ordenam modelos concorrentes conjugando qualidade de ajuste e número de parâmetros. Valores menores indicam modelos preferíveis.

Amostragem estratificada. Sorteio conduzido dentro de subgrupos definidos previamente, de modo a assegurar representação de todos eles na amostra.

Distribuições diamétricas e mensuração

Distribuição diamétrica. Descrição de como os diâmetros se distribuem em um povoamento ou em um lote de toras, usualmente representada por uma função de probabilidade ajustada aos dados medidos.

Classe diamétrica. Intervalo de diâmetros tratado como uma unidade. Neste trabalho adotaram-se 35 classes de 1 cm de amplitude, identificadas pelo seu centro de classe.

Função densidade de probabilidade (FDP). Função que descreve a concentração relativa de probabilidade ao longo dos valores possíveis da variável. A probabilidade de uma classe corresponde à área sob a curva no intervalo da classe.

Função de distribuição acumulada (FDA). Função que fornece a probabilidade de a variável assumir valor igual ou inferior a um dado ponto. A probabilidade de uma classe é obtida pela diferença da FDA entre os limites da classe.

Distribuição de Weibull. Distribuição de dois ou três parâmetros amplamente empregada na descrição de distribuições diamétricas, por acomodar assimetria à direita e à esquerda conforme o parâmetro de forma.

Distribuição Generalizada de Valores Extremos (GEV). Família que reúne, conforme o parâmetro de forma, as distribuições de Gumbel, Fréchet e Weibull. O tipo III, de cauda superior limitada, é o adotado neste trabalho, coerente com a existência de um diâmetro máximo no material.

Parâmetros de locação, escala e forma. Respectivamente, o deslocamento da distribuição ao longo do eixo, a sua dispersão e o formato da curva, incluindo a assimetria e o comportamento das caudas.

Área basal. Área da seção transversal do fuste ou da face da tora, obtida a partir do diâmetro.

Fórmula de Smalian. Procedimento de cubagem que estima o volume de uma tora pela média das áreas seccionais de suas duas extremidades multiplicada pelo comprimento.

Volume sólido e volume estéreo. O volume sólido corresponde à madeira efetivamente presente; o volume estéreo, ao espaço ocupado pela pilha, incluindo os vazios entre as toras.

Fator de empilhamento. Razão entre o volume estéreo e o volume sólido de uma pilha, empregada na conversão entre as duas medidas.

Diâmetro à altura do peito (DAP). Diâmetro do fuste medido a 1,30 m do solo na árvore em pé. Distingue-se do diâmetro da face da tora, empregado na mensuração de madeira empilhada.

Afilamento (*taper*). Redução progressiva do diâmetro do fuste com a altura, descrita por funções que permitem estimar o diâmetro em qualquer posição.

SUMÁRIO

LISTA DE FIGURAS	11
LISTA DE TABELAS	14
LISTA DE TERMOS TÉCNICOS	15
SUMÁRIO	20
INTRODUÇÃO GERAL	26
REFERÊNCIAS	32
1 VISÃO COMPUTACIONAL E OPERAÇÃO DO MODELO YOLO	34
RESUMO	34
1.1 INTRODUÇÃO	34
1.2 REVISÃO BIBLIOGRÁFICA	36
1.2.1 Visão computacional na era da inteligência artificial	36
1.2.2 Componentes em redes neurais convolucionais	38
1.2.2.1 Camada convolucional	38
1.2.2.2 Camada de pooling	39
1.2.2.3 Camadas totalmente conectadas	40
1.2.3 Funções de ativação em redes neurais convolucionais	42
1.2.4 Operação de detecção de objetos do algoritmo YOLO	46
1.2.5 Evolução progressiva do modelo YOLO na visão computacional	56
1.2.5.1 YOLO V1: Você só olha uma vez	58
1.2.5.2 YOLO V2: YOLO9000, melhor, mais rápido, mais forte	62
1.2.5.3 YOLO V3: Uma melhoria incremental	63
1.2.5.4 YOLO V4: Velocidade e precisão ideais da detecção de objetos	67
1.2.5.5 YOLO V5	72
1.2.5.6 YOLO V6: Marco de detecção de objetos em uma única fase para aplicações industriais	75
1.2.5.7 YOLO V7: Conjunto treinável de recursos define um novo padrão para detectores de objetos em tempo real	76

1.2.5.8	YOLO V8	77
1.2.5.9	YOLO V9: Aprendendo o que você quer aprender usando informações de gradiente programável	81
1.2.5.10	YOLO V10: detecção end-to-end sem NMS	82
1.2.5.11	YOLO V11: blocos C3k2, SPPF e C2PSA	83
1.2.5.12	YOLO V12: detectores em tempo real centrados em atenção	84
1.2.5.13	YOLO26: modelos unificados end-to-end para visão em tempo real	85
1.2.6	Versões complementares do modelo YOLO	87
1.2.6.1	Scaled-YOLOV4	87
1.2.6.2	YOLOR	88
1.2.6.3	YOLOX	89
1.2.6.4	DAMO-YOLO	89
1.2.6.5	PP-YOLO	90
1.2.6.6	PP-YOLOV2	90
1.2.6.7	PP-YOLOE	91
1.2.6.8	YOLO-NAS	92
1.2.7	Síntese estrutural: elementos comuns às versões YOLO	92
1.3	CONCLUSÕES	94
1.4	REFERÊNCIAS	95

2	VISÃO COMPUTACIONAL APLICADA À MADEIRA: UM MAPEAMENTO SISTEMÁTICO DA LITERATURA	104
	RESUMO (Capítulo II)	104
2.1	INTRODUÇÃO	105
2.2	MATERIAIS E MÉTODOS	106
2.2.1	Fonte de dados e estratégia de busca	106
2.2.2	Triagem inicial e critérios de elegibilidade	107
2.2.3	Pré-processamento textual e filtragem por regras	109
2.2.4	Taxonomia de aplicações e procedimento de classificação	109
2.2.5	Taxonomia metodológica	110
2.2.6	Aferição da confiabilidade da classificação	111
2.2.7	Limitações da classificação	112
2.3	PANORAMA BIBLIOMÉTRICO	112
2.3.1	Tendência de publicação	112
2.3.2	Tipos documentais e veículos de publicação	113
2.3.3	Distribuição geográfica	114
2.3.4	Análise de citações	115
2.3.5	Rede de coocorrência de palavras-chave	117

2.4	CATEGORIAS DE APLICAÇÃO	118
2.4.1	Controle de qualidade de superfície	119
2.4.2	Imageamento tridimensional e interno	120
2.4.3	Identificação de espécie e origem	120
2.4.4	Microestrutura e caracterização	121
2.4.5	Produção de painéis e chapas	121
2.4.6	Processamento de toras e madeira roliça	121
2.4.7	Medição dimensional e geometria	122
2.4.8	Automação e robótica em serrarias	122
2.4.9	Predição de propriedades da madeira	122
2.5	EVOLUÇÃO METODOLÓGICA	123
2.5.1	Distribuição das classes metodológicas	123
2.5.2	Perfil metodológico por categoria de aplicação	123
2.5.3	Tendências de adoção do aprendizado profundo	125
2.6	ANÁLISE TEMPORAL	126
2.6.1	Período de fundação (1983–1999)	126
2.6.2	Período de crescimento (2000–2015)	126
2.6.3	Era do aprendizado profundo (2016–presente)	127
2.6.4	Evolução das categorias ao longo do tempo	127
2.7	DISCUSSÃO	128
2.8	DIREÇÕES FUTURAS DE PESQUISA	129
2.9	CONCLUSÕES	130
2.10	REFERÊNCIAS	131
3	DETECÇÃO E CONTAGEM AUTOMÁTICA DE MADEIRA DE EU-CALIPTO EMPILHADA USANDO O MODELO YOLOv8	142
	RESUMO (Capítulo III)	142
3.1	INTRODUÇÃO	143
3.2	MATERIAIS E MÉTODOS	146
3.2.1	Conjunto de dados	146
3.2.2	Aquisição e contagem em vídeo	147
3.2.3	Formato de Anotação e Divisão do Conjunto de Dados	148
3.2.4	Arquitetura e Configuração do Modelo	149
3.2.5	Procedimento de generalização	150
3.2.6	Medidas de desempenho	151
3.3	RESULTADOS	152
3.3.1	Desempenho do Modelo	152
3.3.2	Generalização do Modelo	153
3.3.3	Análise de sensibilidade dos parâmetros de inferência	157

3.3.4	Comparação estatística entre fotografia e vídeo	162
3.4	DISCUSSÃO	164
3.5	CONCLUSÕES	167
3.6	REFERÊNCIAS	168
4	QUANTIFICAÇÃO DO VOLUME SÓLIDO DE MADEIRA DE EUCALIPTO EMPILHADA UTILIZANDO MODELOS DE DETECÇÃO-SEGMENTAÇÃO E DE DISTRIBUIÇÃO DE DIÂMETROS	173
	RESUMO (Capítulo IV)	173
4.1	INTRODUÇÃO	174
4.2	MATERIAIS E MÉTODOS	176
4.2.1	Conjunto de dados	176
4.2.2	Volume sólido por meio de distribuição de probabilidade	177
4.2.3	Detecção e segmentação automática de toras	178
4.3	RESULTADOS	180
4.3.1	Volume sólido por meio de distribuição de probabilidade	180
4.3.2	Simulação da estimativa de volume sólido por densidade de probabilidade	184
4.3.3	Verificação por meio da distribuição acumulada	188
4.3.4	Validação de um método de estimativa de volume de toras por meio de comparação com o método de Smalian	189
4.3.5	Análise abrangente do desempenho do modelo e técnica de processa- mento de imagens	190
4.3.5.1	Segmentação	190
4.3.5.2	Detecção	191
4.3.6	Aplicação da estimativa do volume de madeira empilhada por meio de contagem automática	193
4.4	DISCUSSÃO	193
4.5	CONCLUSÕES	196
4.6	REFERÊNCIAS	196
5	NEUROTIMBER: ESTIMATIVA DE VOLUME DE MADEIRA EMPIL- HADA USANDO DETECÇÃO POR IA E MODELOS DE DISTRIBUIÇÃO DIAMÉTRICA	200
	RESUMO (Capítulo V)	200
5.1	INTRODUÇÃO	201
5.2	VISÃO GERAL DO PIPELINE DE IMPLEMENTAÇÃO	202
5.2.1	Preparação do ambiente	202
5.2.1.1	Instalação e versões	202
5.2.1.2	Repositório YOLOv9	202
5.2.1.3	Imports e supressão de avisos	202

5.2.2	Configuração de caminhos e artefatos	203
5.2.3	Detecção com YOLOv9 e integração com Supervision	203
5.2.3.1	Classe Extended Detections	203
5.2.3.2	Carregamento e post-processing	203
5.2.3.3	Parâmetros de inferência/NMS	204
5.2.3.4	Anotação simples de vídeo	204
5.2.4	Rastreamento (ByteTrack) e contagem global de toras	204
5.2.4.1	Contador global	204
5.2.4.2	ByteTrack e anotadores	205
5.2.4.3	Loop de anotação com contagem	205
5.2.5	Modelagem de distribuição diamétrica (SciPy)	205
5.2.5.1	Leitura dos dados e seleção da coluna	205
5.2.5.2	Lista de distribuições candidatas	206
5.2.5.3	Ajuste e seleção via AIC	206
5.2.5.4	Probabilidades por classe de diâmetro	206
5.2.6	Estimativa do volume empilhado	207
5.2.6.1	Fórmula geométrica e unidades	207
5.2.6.2	Combinação com probabilidades e contagem	207
5.3	TUTORIAL DE USO DO NEUROTIMBER (VERSÃO WEB - HUGGING FACE SPACES)	208
5.3.1	Visão geral da interface	208
5.3.2	Detecção de toras em vídeo (aba “Detecção em Vídeo”)	208
5.3.2.1	Carregar o vídeo	208
5.3.2.2	Processar o vídeo	209
5.3.2.3	Ler o resultado	209
5.3.3	Estimativa de volume (aba “Diâmetros & Volume”)	209
5.3.3.1	Carregar os dados de diâmetro	209
5.3.3.2	Definir intervalos e comprimento	209
5.3.3.3	Ajustar distribuições	209
5.3.3.4	Escolher a distribuição e calcular volume	209
5.4	CONCLUSÕES	210

6	NEUROTIMBER MOBILE: ESTIMATIVA DE VOLUME SÓLIDO DE MADEIRA EMPILHADA POR SEGMENTAÇÃO DE IMAGENS E CALI- BRAÇÃO PELA LARGURA DA PILHA EM APLICATIVO DE MENSAGERIA	212
	RESUMO (Capítulo VI)	212
6.1	INTRODUÇÃO	213
6.2	MATERIAIS E MÉTODOS	214
6.2.1	Conjunto de dados e volumes de referência	214

6.2.2	Segmentação e recorte da pilha com SAM 3	215
6.2.3	Segmentação individual das toras com YOLO26	215
6.2.4	Calibração métrica pela largura da pilha	215
6.2.5	Cálculo do volume pela fórmula de Smalian	216
6.2.6	Implementação móvel: NeuroTimber Mobile	216
6.2.7	Avaliação estatística	218
6.3	RESULTADOS	219
6.3.1	Segmentação da pilha e das toras	219
6.3.2	Estimativas de volume	220
6.3.3	Consistência física da calibração	221
6.3.4	O aplicativo NeuroTimber Mobile em operação	222
6.4	APLICAÇÃO EM MODO OPERACIONAL: PROPOSTA	225
6.4.1	Recomendações para a aquisição das fotografias	227
6.5	DISCUSSÃO	229
6.6	CONCLUSÕES	231
6.7	REFERÊNCIAS	231

CONCLUSÕES GERAIS

234

INTRODUÇÃO GERAL

A visão computacional desponta como uma área chave na era da inteligência artificial, permitindo que máquinas extraíam informações significativas de imagens e vídeos. Em especial, técnicas de detecção de objetos evoluíram rapidamente na última década. Entre elas, destaca-se a família de algoritmos YOLO (You Only Look Once), conhecida por seu equilíbrio notável entre velocidade e acurácia na detecção em tempo real (TERVEN; CÓRDOVA-ESPARZA; ROMERO-GONZÁLEZ, 2023). O YOLO foi introduzido por Redmon et al. (2016) como um modelo unificado de detecção que processa a imagem completa em uma única passada de rede neural convolucional, combinando as etapas de proposição de regiões e classificação em um só estágio (SAPKOTA et al., 2026). Essa abordagem revolucionária reduziu drasticamente o tempo de inferência sem sacrificar a precisão, transformando YOLO em uma referência para aplicações que demandam resposta rápida (por exemplo, veículos autônomos, vigilância e robótica).

Desde a publicação original, o algoritmo passou por várias iterações (YOLOv2, v3, v4, v5, v6 e v7, até as versões mais recentes, como YOLOv8, YOLOv9, YOLOv10, YOLOv11, YOLOv12 e o YOLO26, esta última a mais atual), cada uma incorporando melhorias arquiteturais e de treinamento para aprimorar desempenho. Revisões abrangentes recentes detalham essa evolução, evidenciando as inovações de cada versão, como backbones mais robustos, mecanismos de atenção e estratégias de data augmentation, e confirmando os ganhos contínuos em velocidade e acurácia ao longo da série YOLO. Estudos comparativos recentes reforçam esse avanço ao contrastar versões como YOLOv9 e YOLOv10 com arquiteturas baseadas em transformadores em aplicações em tempo real (SALTÍK; ALLMENDINGER; STEIN, 2024), enquanto revisões sistemáticas discutem os desafios da detecção de objetos em ambientes abertos e não controlados (LIANG et al., 2024). Uma tendência emergente é a detecção e a segmentação de vocabulário aberto, que dispensam listas fixas de classes e ampliam a flexibilidade dos detectores em tempo real (SAPKOTA; KARKEE, 2026).

Vale salientar que o sucesso desses modelos está alicerçado no poder das redes neurais profundas, em particular das CNNs, capazes de aprender hierarquias de características visuais (bordas, formas, texturas) e identificar objetos independentemente de sua posição ou escala na imagem. Ademais, a literatura enfatiza a importância de treinar os modelos de visão com dados específicos da tarefa alvo; ou seja, ao se preparar um detector para uma aplicação particular, o uso de conjuntos de treinamento anotados de forma pertinente tende a tornar a detecção mais eficiente e precisa (ZHU; WU, 2023). Esses fundamentos teóricos, envolvendo arquiteturas de CNN, estratégias de treinamento orientadas à tarefa e aprimoramentos iterativos nos algoritmos, constituem a base conceitual abordada no Capítulo 1, proporcionando uma visão ampla dos conceitos de visão computacional e do funcionamento interno do modelo YOLO, pilar central desta pesquisa.

O Capítulo 2 situa esse arcabouço conceitual no panorama da literatura. Por meio de um mapeamento sistemático de 1.019 artigos indexados na base Scopus e publicados entre 1983 e 2026, quantifica-se como a pesquisa em visão computacional aplicada à madeira se distribui entre temas de aplicação, entre abordagens metodológicas e ao longo do tempo. O levantamento cumpre duas funções nesta tese. A primeira é oferecer um panorama reproduzível de um campo cuja produção se encontra dispersa entre comunidades científicas que raramente dialogam, o que dificulta discernir quais problemas já dispõem de solução consolidada. A segunda, e mais relevante para o encadeamento do trabalho, é demonstrar de forma quantitativa que as aplicações voltadas à mensuração, nas quais se inclui a quantificação volumétrica de madeira empilhada, permanecem sub-representadas: somam menos de 9% da literatura, contra quase metade dedicada ao controle de qualidade de superfície. A lacuna assim delimitada não decorre de impressão, mas de contagem, e constitui a justificativa para os capítulos subsequentes.

No Capítulo 3, o foco recai sobre a aplicação prática do YOLO na detecção e contagem automática de toras de madeira empilhadas. A motivação para tal aplicação reside nas limitações dos métodos tradicionais de quantificação florestal. Nas operações de colheita e comercialização de madeira, é comum a medição de toras empilhadas em metros cúbicos estéreos (m^3 estéreo) por meio de estimativas manuais de dimensões da pilha. Tais procedimentos manuais são sujeitos a erros humanos e pouco viáveis em larga escala (PURFÜRST et al., 2023). Contagens visuais de toras a partir de fotografias, por exemplo, tornam-se impraticáveis em ambientes industriais com grandes volumes, exceto em pilhas muito pequenas (CASAS et al., 2023). Consequentemente, há uma forte demanda por soluções automatizadas que aumentem a eficiência e confiabilidade dessa tarefa.

A visão computacional oferece essa possibilidade: ao treinar um modelo de detecção de objetos para reconhecer as extremidades das toras nas imagens, podemos automatizar a contagem de maneira consistente. O YOLO, por sua arquitetura de detector de estágio único extremamente rápido, mostrou-se uma escolha apropriada para esse fim. Sua versão, YOLOv8, lançada em 2023, incorpora avanços de rede neural (como backbone CSPDarknet aprimorado e cabeçotes de detecção atualizados) e suporta diretamente tarefas de detecção e segmentação de objetos, tudo com implementações acessíveis via biblioteca Python e interface de linha de comando (JOCHER; CHAURASIA; QIU, 2023). Graças a esses aprimoramentos, o YOLOv8 alcança desempenho de ponta em detecção de objetos. No contexto florestal, a adoção de YOLOv8 permite detectar automaticamente dezenas ou centenas de toras em uma imagem em segundos, algo inviável manualmente.

Diversos estudos já demonstram a eficácia de modelos YOLO em cenários análogos: por exemplo, Fang et al. (2021) empregaram o YOLOv5 para detecção precisa de nós em tábuas de madeira serrada, obtendo alta acurácia na identificação de imperfeições como nós e ressaltos (FANG et al., 2021). De modo similar, Ma et al. (2022) aplicaram algoritmos YOLO para extrair contornos de trincas em estruturas de madeira de construções históricas, mesmo sob texturas complexas e irregulares, comprovando a robustez do método em diferentes padrões visuais (MA

et al., 2022). Também no domínio de detecção de defeitos, abordagens baseadas em YOLO têm alcançado sucesso notável na identificação de falhas em produtos de madeira, como ilustrado no trabalho de Kurdthongmee (2023) com melhorias no YOLOv3 para localizar regiões defeituosas com alta exatidão (KURDTHONGMEE, 2023). Ademais, esforços recentes mostram a versatilidade dessas técnicas em ambientes industriais: Hasanzadeh Fard et al. (2025) propuseram uma solução de baixo custo utilizando YOLOv5 para detectar toras de madeira em pátios e estimar automaticamente o diâmetro de cada peça a partir da caixa delimitadora, alcançando desempenho confiável (mAP superior a 0,6) mesmo com hardware modesto. Avanços recentes na inspeção de madeira reforçam essa tendência, com redes YOLO leves e mecanismos de atenção voltados à detecção de defeitos em painéis e à implantação em dispositivos de borda (CAO et al., 2023; KANG et al., 2025); no inventário em pé, o aprendizado profundo aliado à varredura laser terrestre já permite identificar e segmentar verticilos de galhos e toras comerciais em árvores individuais (PEHKONEN et al., 2025). Esses exemplos da literatura corroboram a viabilidade e vantagem da contagem automatizada de toras via visão computacional, apoiando teoricamente a abordagem desenvolvida neste trabalho.

No capítulo, são discutidos os detalhes do modelo YOLOv8 treinado para toras de eucalipto, incluindo preparação do conjunto de dados anotado, métricas de avaliação (precisão, recall, mAP) e técnicas de pós-processamento como Non-Maximum Suppression, estabelecendo uma base metodológica sólida para a detecção e contagem automáticas.

No Capítulo 4, a investigação aprofunda-se na estimativa do volume sólido das toras empilhadas, integrando a etapa de detecção previamente mencionada com modelos estatísticos de distribuição diamétrica. A base teórica para esse capítulo reside na combinação de técnicas de visão computacional com princípios clássicos de inventário florestal. Uma vez que um modelo como o YOLO pode fornecer a contagem total de toras numa pilha (e potencialmente identificar as dimensões de suas secções transversais nas imagens), é possível estimar o volume total combinando essa informação com a distribuição dos diâmetros das toras. Em cenários onde as toras da pilha possuem comprimentos similares (como é comum em plantações manejadas onde as toras são cortadas com comprimentos padronizados), a distribuição de frequências dos diâmetros serve como base para o cálculo do volume. Essencialmente, conhecendo-se quantas toras existem em cada classe de diâmetro, pode-se multiplicar esse número pelo volume unitário médio de uma tora daquela classe (considerando o diâmetro médio da classe e um comprimento fixo) e então somar sobre todas as classes para obter o volume total estimado.

A utilização de modelos de distribuição de probabilidade para descrever os diâmetros de árvores ou toras é uma técnica bem estabelecida em mensuração florestal. Diversas funções estatísticas, como as distribuições Normal, Lognormal, Gama, Weibull, Johnson SB e outras, já foram aplicadas com sucesso para modelar a distribuição diamétrica de povoamentos florestais (EGONMWAN; OGANA, 2020). Em particular, a distribuição Weibull destaca-se historicamente pela flexibilidade em ajustar dados de diâmetro e é amplamente empregada para esse fim, embora outras distribuições possam ser mais adequadas dependendo do caso. Estudos re-

centes seguem aprimorando esse ajuste, seja avaliando métodos de estimação dos parâmetros da função Weibull (BONČINA et al., 2022), seja comparando múltiplas funções densidade de probabilidade em florestas mistas (GUZMÁN-SANTIAGO et al., 2024). Ao caracterizar matematicamente a estrutura diamétrica de um lote de toras, ganhamos a capacidade de prever volumes por classe de tamanho e fazer inferências sobre o estoque total. Por exemplo, Egonmwan e Ogana (2020) demonstraram em um estudo de povoamentos de *Tectona grandis* que o ajuste da distribuição de diâmetros via função Weibull, combinado com modelos alométricos altura-diâmetro e volume, permitiu estimar de forma acurada o volume por hectare do povoamento analisado. No geral, a hipótese subjacente é que a distribuição diamétrica de um conjunto de toras pode ser representada por uma função contínua, e os parâmetros dessa função (estimados a partir de amostras de campo ou dados prévios do talhão) possibilitam extrapolar o volume total quando integrados com a informação de contagem fornecida pela visão computacional.

O capítulo teórico aborda os detalhes desses modelos: discute-se a seleção de distribuições candidatas (incluindo distribuições de extremos generalizadas, dado que em alguns casos há interesse nas caudas da distribuição de diâmetros), os critérios de ajuste e seleção de modelos (e.g., uso do critério de informação Akaike, AIC, para comparar o quão bem cada distribuição se ajusta aos dados observados) e a forma de calcular o volume a partir da distribuição escolhida. Fundamentado na literatura estatística florestal, o texto ressalta que a predição de volume com base em distribuições de probabilidade é uma extensão natural das práticas de inventário, apresentando potencial de alta acurácia e eficiência quando aliada a dados coletados automaticamente. Assim, o Capítulo 4 fornece o embasamento teórico e prático para integrar a detecção automática (contagem de toras) com a modelagem diamétrica, resultando em um procedimento inovador de estimativa volumétrica automatizada.

Por fim, o Capítulo 5 sintetiza a materialização prática de todos esses conceitos na forma do NeuroTimber, um sistema integrado que une detecção por IA e modelos estatísticos em uma aplicação coesa. Do ponto de vista conceitual, este capítulo está fundamentado em princípios de engenharia de sistemas e na tendência de transformação digital na área florestal, muitas vezes referida como Floresta 4.0 ou silvicultura de precisão. A ideia central é que, ao combinar diferentes tecnologias (visão computacional, aprendizado de máquina, interfaces web interativas e computação em nuvem), podemos criar soluções completas que automatizam e aprimoram cadeias de trabalho tradicionais no setor florestal, trazendo ganhos de produtividade e confiabilidade.

Estudos recentes em logística florestal ressaltam a falta de abordagens holísticas que integrem todas as etapas do processo, do monitoramento à quantificação e tomada de decisão, indicando que a ausência de integração entre soluções pontuais limita o impacto pleno das tecnologias de IA no setor (NUNES, 2025). O NeuroTimber almeja justamente preencher essa lacuna integrando, num único fluxo, a contagem automatizada de toras (via modelo YOLO) com a estimativa estatística do volume (via ajuste de distribuições diamétricas), entregando essa funcionalidade por meio de uma interface de fácil uso. Na prática, isso significa que um

usuário pode carregar um vídeo ou imagem de uma pilha de madeira e obter instantaneamente não apenas a contagem de toras identificadas, mas também uma estimativa confiável do volume total daquela pilha, tudo de forma automatizada. A disponibilização dessa solução tanto em uma plataforma web (web app) quanto como repositório de código aberto reforça os conceitos de reprodutibilidade e transparência científica.

Ao abrir o código e os pesos do modelo treinado, garante-se que outros pesquisadores possam auditar, testar e aprimorar o método, atendendo às boas práticas de pesquisa reproduzível. Em paralelo, a interface gráfica simplifica a adoção pela indústria, permitindo que profissionais sem treinamento em programação utilizem a ferramenta no dia a dia, o que atende às demandas de usabilidade e transferência de tecnologia. Essa dualidade, código aberto e aplicação amigável, alinha-se com as recomendações atuais de promover soluções de IA acessíveis e confiáveis no setor florestal, maximizando seu impacto prático. Adicionalmente, a digitalização do processo de medição de madeira traz benefícios de rastreamento e padronização: um sistema automatizado reduz a variabilidade entre medições de diferentes operadores e armazena os resultados de forma sistemática, facilitando auditorias e análises históricas.

O Capítulo 6 estende a proposta do NeuroTimber para o contexto móvel, apresentando o NeuroTimber Mobile, uma solução que opera integralmente em um aplicativo de mensageria (Telegram) e substitui o catálogo de diâmetros por uma calibração métrica direta informada pelo operador. Nessa abordagem, a segmentação de contexto pelo Segment Anything Model 3 (SAM 3) isola a pilha do fundo, a segmentação de instâncias pelo YOLO26 delimita a face de cada tora e a largura física da face, fornecida pelo usuário, converte a área de madeira observada em metros quadrados, à qual se aplica a fórmula de Smalian para obter o volume sólido. Modelos de segmentação de fundação, capazes de operar em modo zero-shot, já vêm sendo aplicados à detecção e à segmentação de árvores em imagens aéreas (CHEN et al., 2025), e abordagens baseadas em nuvens de pontos têm sido propostas para estimar o volume de pilhas de materiais diversos (LING et al., 2024), o que reforça a viabilidade de estimativas volumétricas automatizadas a partir de imagens. Fundamentado em princípios de mensuração florestal e na tendência de aproximar a inteligência artificial do usuário de campo por meio de dispositivos móveis, o capítulo avalia o método em 30 pilhas de *Eucalyptus grandis* e discute sua acurácia, sua consistência física e sua operacionalização em arquitetura cliente-servidor.

Em conclusão, esta introdução geral apresentou os alicerces teóricos correspondentes a cada um dos seis capítulos da pesquisa. Abordamos (i) os fundamentos da visão computacional e do modelo YOLO, que provê a capacidade de detecção automática de objetos com elevada eficiência; (ii) o mapeamento sistemático da literatura de visão computacional aplicada à madeira, que delimita quantitativamente a lacuna das aplicações de mensuração; (iii) a aplicação dessa capacidade na contagem de toras de madeira; (iv) a extensão da metodologia para a estimativa de volume, combinando a detecção com modelos de distribuição diamétrica da mensuração florestal, em prol de estimativas robustas do volume sólido; (v) a integração dessas componentes em um sistema completo, refletindo tendências modernas de digitalização e propondo uma fer-

ramenta de alto potencial para silvicultura de precisão (NeuroTimber); e (vi) a extensão dessa solução para um aplicativo de mensageria (NeuroTimber Mobile), que estima o volume sólido por segmentação de imagens e calibração métrica direta. Nos próximos capítulos, esses conceitos são explorados em profundidade e aplicados experimentalmente, buscando demonstrar empiricamente os benefícios e contribuições do trabalho para a área de inventário florestal automatizado.

REFERÊNCIAS

- BONČINA, Z.; TRIFKOVIĆ, V.; ROSSET, C.; KLOPČIČ, M. Evaluation of estimation methods for fitting the three-parameter Weibull distribution to European beech forests. **iForest - Biogeosciences and Forestry**, v. 15, p. 484-490, 2022. <https://doi.org/10.3832/ifor4145-015>.
- CAO, Y.; LIU, F.; JIANG, L.; BAO, C.; MIAO, Y.; CHEN, Y. **Lightweight wood panel defect detection method incorporating attention mechanism and feature fusion network**. arXiv:2306.12113, 2023. <https://doi.org/10.48550/arXiv.2306.12113>.
- CASAS, G. G.; ISMAIL, Z. H.; LIMEIRA, M. M. C.; DA SILVA, A. A. L.; LEITE, H. G. Automatic Detection and Counting of Stacked Eucalypt Timber Using the YOLOv8 Model. **Forests**, v. 14, n. 12, p. 2369, 2023. <https://doi.org/10.3390/f14122369>.
- CHEN, M.; RUSSELL, D.; PALLAVOOR, A.; YOUNG, D.; WU, J. **Zero-Shot Tree Detection and Segmentation from Aerial Forest Imagery**. arXiv:2506.03114, 2025. <https://doi.org/10.48550/arXiv.2506.03114>.
- EGONMWAN, I. Y.; OGANA, F. N. Application of diameter distribution model for volume estimation in *Tectona grandis* L.f. stands in the Oluwa forest reserve, Nigeria. **Tropical Plant Research**, v. 7, n. 3, p. 573-580, 2020. <https://doi.org/10.22271/tpr.2020.v7.i3.070>.
- FANG, Yiming; GUO, Xianxin; CHEN, Kun; ZHOU, Zhu; YE, Qing. Accurate and automated detection of surface knots on sawn timbers using YOLO-V5 model. **BioResources**, v. 16, n. 3, p. 5390-5406, 2021. <https://doi.org/10.15376/biores.16.3.5390-5406>.
- GUZMÁN-SANTIAGO, J. C.; DE LOS SANTOS-POSADAS, H. M.; VARGAS-LARRETA, B.; GÓMEZ-CÁRDENAS, M.; SANTIAGO-GARCÍA, W.; NAVA-NAVA, A. Predicting Diameter Distributions in Mixed Forests in Southern Mexico. **South-east European Forestry**, v. 15, n. 2, p. 161-173, 2024. <https://doi.org/10.15177/see-for.24-16>.
- HASANZADEH FARD, F.; HASANZADEH FARD, S.; JONOBI, M. **A Low-Cost Machine Learning Approach for Timber Diameter Estimation**. arXiv preprint arXiv:2507.17219, 2025. <https://doi.org/10.48550/arXiv.2507.17219>.
- JOCHER, G.; CHAURASIA, A.; QIU, J. **YOLO by Ultralytics (Version 8.0.0)**. 2023. Disponível em: <https://github.com/ultralytics/ultralytics>. Acessado em: 22 jul. 2023.
- KANG, J.; CEN, Y.; CEN, Y.; WANG, K.; LIU, Y. **CFIS-YOLO: A Lightweight Multi-Scale Fusion Network for Edge-Deployable Wood Defect Detection**. arXiv:2504.11305, 2025. <https://doi.org/10.48550/arXiv.2504.11305>.
- KURDTHONGMEE, Wattanapong. **Improving Wood Defect Detection Accuracy with Yolo V3 by Incorporating Out-of-Defect Area Annotations**. 2023. <https://doi.org/10.2139/ssrn.4395580>.
- LIANG, S.; WANG, W.; CHEN, R.; LIU, A.; WU, B.; CHANG, E.-C.; CAO, X.; TAO, D. **Object Detectors in the Open Environment: Challenges, Solutions, and Outlook**. arXiv:2403.16271, 2024. <https://doi.org/10.48550/arXiv.2403.16271>.
- LING, Y.; ZHAO, R.; SHEN, Y.; LI, D.; JIN, J.; LIU, J. **DIVESPOT: Depth Integrated Volume Estimation of Pile of Things Based on Point Cloud**. arXiv:2407.05415, 2024. <https://doi.org/10.48550/arXiv.2407.05415>.
- MA, Jian; YAN, Weidong; LIU, Guoqi; XING, Shiyu; NIU, Siqi; WEI, Tong. Complex

Texture Contour Feature Extraction of Cracks in Timber Structures of Ancient Architecture Based on YOLO Algorithm. **Advances in Civil Engineering**, v. 2022, p. 1-13, 2022. <https://doi.org/10.1155/2022/7879302>.

NUNES, L. J. R. The Role of Artificial Intelligence (AI) in the Future of Forestry Sector Logistics. **Future Transportation**, v. 5, n. 2, p. 63, 2025. <https://doi.org/10.3390/futuretransp5020063>.

PEHKONEN, M.; VASTARANTA, M.; HOLOPAINEN, M.; HYYPPÄ, J.; KUKKO, A.; PYÖRÄLÄ, J. Identification and segmentation of branch whorls and sawlogs in standing timber using terrestrial laser scanning and deep learning. **Forestry: An International Journal of Forest Research**, v. 98, n. 5, p. 712-725, 2025. <https://doi.org/10.1093/forestry/cpaf006>.

PURFÜRST, T; DE MIGUEL-DÍEZ, F; BERENDT, F; ENGLER, B; CREMER, T. Comparison of wood stack volume determination between manual, photo-optical, iPad-LiDAR and handheld-LiDAR based measurement methods. **iForest - Biogeosciences and Forestry**, v. 16, n. 4, p. 243-252, 2023. <https://doi.org/10.3832/ifor4153-016>.

REDMON, J.; DIVVALA, S.; GIRSHICK, R.; FARHADI, A. You Only Look Once: Unified, Real-Time Object Detection. 2016. **Proceedings of the IEEE conference on computer vision and pattern recognition** [...]. [S. l.: s. n.], 2016. p. 779-788.

SALTÍK, A. O.; ALLMENDINGER, A.; STEIN, A. **Comparative Analysis of YOLOv9, YOLOv10 and RT-DETR for Real-Time Weed Detection**. arXiv:2412.13490, 2024. <https://doi.org/10.48550/arXiv.2412.13490>.

SAPKOTA, R.; CHEPPALLY, R. H.; SHARDA, A.; KARKEE, M. **YOLO26: Key Architectural Enhancements and Performance Benchmarking for Real-Time Object Detection**. arXiv:2509.25164v5, 16 mar. 2026. <https://doi.org/10.48550/arXiv.2509.25164>.

SAPKOTA, R.; KARKEE, M. **YOLOE-26: Integrating YOLO26 with YOLOE for Real-Time Open-Vocabulary Instance Segmentation**. arXiv:2602.00168, 2026. <https://doi.org/10.48550/arXiv.2602.00168>.

TERVEN, J.; CÓRDOVA-ESPARZA, D.-M.; ROMERO-GONZÁLEZ, J.-A. A Comprehensive Review of YOLO Architectures in Computer Vision: From YOLOv1 to YOLOv8 and YOLO-NAS. **Machine Learning and Knowledge Extraction**, v. 5, n. 4, p. 1680-1716, 2023. <https://doi.org/10.3390/make5040083>.

ZHU, S.-C.; WU, Y. Statistics of Natural Images. **Computer Vision**. Cham: Springer International Publishing, 2023. p. 19-35. https://doi.org/10.1007/978-3-030-96530-3_2.

CAPÍTULO I

VISÃO COMPUTACIONAL E OPERAÇÃO DO MODELO YOLO

RESUMO

A detecção automática de objetos em imagens tem se consolidado como alternativa às medições manuais na mensuração florestal, mas sua aplicação criteriosa exige a compreensão dos fundamentos que governam o desempenho dos detectores, tema pouco sistematizado na literatura do setor. Este capítulo tem por objetivo reunir os fundamentos de visão computacional e o funcionamento do modelo YOLO (You Only Look Once), situando-os no contexto da detecção de toras em imagens de madeira de eucalipto empilhada. A exposição parte das redes neurais convolucionais e de seus componentes clássicos, a saber, convolução, pooling, camadas totalmente conectadas e funções de ativação, e avança até a operação do detector, do pré-processamento ao pós-processamento, descrevendo o pipeline do YOLO como processo unificado no qual a imagem é dividida em uma grade e cada célula prediz, em uma única passada pela rede, caixas delimitadoras, escores de confiança e probabilidades de classe. Na avaliação, formalizam-se a Interseção sobre União (IoU), a relação precisão-revocação, as métricas $mAP@50$ e $mAP@50:95$ e a curva de confiança F1, e detalha-se, por meio de pseudocódigo, a Supressão de Não Máximos. A revisão sistematiza ainda a evolução do modelo, do YOLOv1 às versões mais recentes, incluindo variantes complementares, e evidencia que todas compartilham o esqueleto Backbone-Neck-Head e o paradigma de detecção em uma passada, diferenciando-se pela implementação interna de cada componente e por uma trajetória que caminha das cabeças baseadas em âncoras rumo a detectores *anchor-free* e de inferência *end-to-end*. Conclui-se que o encadeamento entre conceitos, métricas e arquiteturas, associado às práticas de anotação e de particionamento do conjunto de dados aqui descritas, fornece o alicerce metodológico para implementar, avaliar e aprimorar detectores em cenários florestais, subsidiando as etapas de detecção, contagem e estimativa de volume desenvolvidas nos capítulos seguintes.

Palavras-chave: visão computacional; redes neurais convolucionais; detecção de objetos; modelo YOLO; madeira empilhada.

1.1. INTRODUÇÃO

A visão computacional, conforme descrita em diversos livros sobre o assunto, representa um campo essencial dentro da inteligência artificial, concentrando-se na criação de algoritmos

capazes de extrair informações significativas a partir de dados visuais, tais como imagens e vídeos. Este campo multidisciplinar combina conceitos e técnicas da computação, matemática, estatística e até mesmo neurociência, buscando desenvolver sistemas capazes de compreender e interpretar o mundo visual de forma análoga aos seres humanos (BHUYAN, 2019; PRINCE, 2012; SZELISKI, 2022).

A abordagem contemporânea da visão computacional se concentra na aprendizagem e na inferência em modelos matemáticos como um tema central. Demonstra-se como utilizar os dados de treinamento para compreender as relações entre os dados de imagem observados e os elementos do mundo que desejamos estimar, como a detecção de múltiplos objetos em 2D e 3D, segmentação e estimativa de pose humana, aprendendo com famílias como R-CNN, SSD, arquiteturas YOLO, U-Net, e plataformas como Detectron2. Também envolve atividades como trocas de expressões faciais, geração de novos rostos, manipulação de expressões faciais e o uso de técnicas de aprendizado por reforço, como Deep Q-learning, para implementar reconhecimento óptico de caracteres (OCR), legendagem de imagens e detecção de objetos com implantação do modelo na produção em nuvem como Amazon Web Services (AWS), Microsoft Azure ou Google Cloud.

As Redes Convolucionais (CNNs) dominam a visão computacional por aprenderem diretamente de imagens, extraindo características e padrões ideais para classificação, detecção e reconhecimento de objetos. No entanto, outras técnicas como redes recorrentes (RNNs) auxiliam na análise de dados sequenciais de imagens para estabelecer relações entre elas. Pesquisadores buscam novas técnicas e combinações, como unir características definidas por especialistas com modelos de deep learning, para aprimorar ainda mais o desempenho da visão computacional (WANG, FENG.; TAX, 2016).

A visão humana é frequentemente influenciada pela tarefa específica que estamos tentando realizar. Portanto, é importante que os modelos visuais sejam capazes de interpretar estímulos visuais de maneira relevante para a tarefa específica em execução. Uma maneira de alcançar isso é treinar os modelos de visão com dados anotados especificamente para a tarefa que se espera que o modelo execute. Ao considerar a tarefa específica que o modelo está tentando realizar, podemos treinar os modelos de visão para serem mais eficientes e precisos (ZHU; WU, 2023).

Uma alternativa consolidada para tarefas específicas de classificação e detecção de objetos é o YOLO. Introduzido por Redmon et al. (2016), o YOLO (You Only Look Once) é um algoritmo de detecção reconhecido por conciliar rapidez e precisão: ele analisa a imagem inteira em uma única passada, dividindo-a em uma grade e predizendo, para cada célula, caixas delimitadoras, escores de confiança e probabilidades de classe. Ao reformular a detecção como um problema de regressão e prescindir de uma etapa separada de proposição de regiões, o YOLO reduz substancialmente a latência de inferência e viabiliza aplicações em tempo real em domínios industriais e científicos.

Desde a versão original, a família YOLO passou por sucessivas iterações que aprimoraram *backbone*, *neck* e *head*, incorporando mecanismos de atenção, novas funções de perda e

estratégias de treinamento (BOCHKOVSKIY; WANG; LIAO, 2020; TERVEN; CÓRDOVA-ESPARZA; ROMERO-GONZÁLEZ, 2023). As versões mais recentes acentuam a tendência de detectores *anchor-free* e de inferência *end-to-end* sem supressão de não máximos, como no YOLOv10 e no YOLO26 (WANG et al., 2024; JOCHER et al., 2026). Apesar dessa diversidade, todas as versões preservam o mesmo princípio de detecção em uma passada e o esqueleto Backbone-Neck-Head, o que motiva, ao final do capítulo, uma síntese estrutural que interliga as diferentes versões.

Neste capítulo, aborda-se o cerne da visão computacional, elucidando seus conceitos fundamentais e oferecendo uma análise detalhada do modelo YOLO, que representa uma das principais forças motrizes da percepção artificial moderna. Investiga-se sua arquitetura, seu funcionamento e suas métricas de avaliação, com o objetivo de fornecer uma visão abrangente que fundamente as escolhas metodológicas dos capítulos seguintes. O entendimento consistente da visão computacional e do modelo YOLO é essencial para o avanço da pesquisa e para o desenvolvimento de tecnologias aplicáveis ao setor florestal, em particular à detecção, contagem e quantificação automatizada de madeira empilhada.

1.2. REVISÃO BIBLIOGRÁFICA

1.2.1. Visão computacional na era da inteligência artificial

A visão computacional, um ramo empolgante da ciência da computação, permite que os computadores "enxerguem" e interpretem o mundo visual. Essa tecnologia inovadora, inspirada na visão humana, capacita máquinas com a capacidade de analisar imagens e vídeos, extraindo informações valiosas e realizando tarefas complexas com precisão e velocidade impressionantes (DAVIES, 2022).

As aplicações da visão computacional são vastas e impactam diversos setores. No reconhecimento de objetos, por exemplo, permite que carros autônomos identifiquem e categorizem elementos no trânsito, como pedestres e placas, e que plataformas de mídia social ofereçam recursos de marcação automática de fotos. Na área de segurança e vigilância, a visão computacional é crucial para a detecção de atividades suspeitas, reconhecimento facial e rastreamento de objetos, aprimorando a eficácia dos sistemas de segurança. Já no controle robótico, a tecnologia permite que robôs interajam com o ambiente de forma segura e eficiente, auxiliando na navegação, manipulação de objetos e execução de tarefas complexas (SZELISKI, 2022).

Com o progresso constante neste campo, antecipa-se a emergência de uma ampla gama de inovações e aplicações da visão computacional, que prometem revolucionar diversos setores industriais, incluindo a indústria florestal e agrícola. Esta tecnologia tem o potencial de desempenhar um papel significativo na moldagem do nosso futuro, não apenas otimizando processos existentes, mas também abrindo novas possibilidades e soluções para desafios contemporâneos.

A inteligência artificial (IA) em visão computacional é derivada do funcionamento do cére-

bro humano. Uma das principais descobertas da neurociência é que o cérebro humano possui duas vias de processamento visual distintas: a via ventral e a via dorsal. A via ventral é responsável pela identificação e reconhecimento de objetos. Ela se inicia no lobo occipital e projeta-se para o lobo temporal, onde está envolvida no processamento das características dos objetos, como sua forma, cor e textura. A via dorsal é responsável pelo processamento da localização espacial dos objetos. Ela se inicia no lobo occipital e projeta-se para o lobo parietal, onde está envolvida no rastreamento dos movimentos dos objetos e sua relação com o corpo (Figura 1.1) (GOODALE; MILNER, 1992).

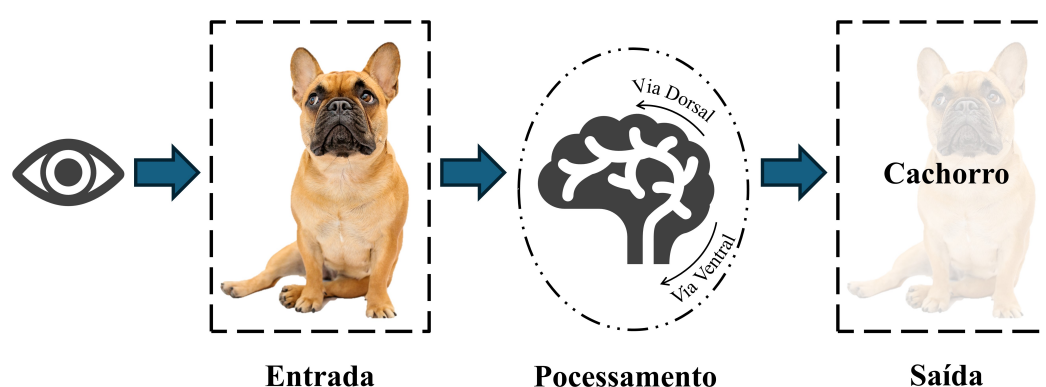


Figura 1.1. Inspiração biológica do funcionamento do cérebro humano aplicada na visão computacional. Fonte: Elaboração própria.

A visão computacional recebeu um grande impulso graças às redes neurais convolucionais (CNNs). As bases para o desenvolvimento de CNNs foram lançadas por Kunihiro Fukushima em 1980 e Yann LeCun em 1989. O trabalho de Fukushima explorou as propriedades matemáticas das CNNs, enquanto LeCun aplicou com sucesso a técnica de retropropagação para treinar uma CNN para reconhecer dígitos em códigos postais manuscritos.

Essa pesquisa pioneira foi expandida por LeCun e sua equipe na década de 1990, culminando com o desenvolvimento da arquitetura LeNet-5, que obteve resultados impressionantes no reconhecimento de documentos.

A partir daí, o campo das CNNs explodiu, com a introdução de novos conjuntos de dados como MNIST e CIFAR-10 e competições como o ImageNet Large Scale Visual Recognition Challenge (ILSVRC). Em torno de 2010, diversas arquiteturas variantes de CNNs foram surgindo ao longo do tempo, como AlexNet, VGGNet, GoogLeNet e ResNet, cada uma com suas vantagens e desvantagens específicas.

O desenvolvimento contínuo de CNNs contribuiu significativamente para o avanço do campo da visão computacional, permitindo a realização de tarefas complexas como reconhecimento de objetos, segmentação de imagens e detecção de faces.

1.2.2. Componentes em redes neurais convolucionais

1.2.2.1. Camada convolucional

Uma rede neural convolucional (*Convolutional Neural Network*, CNN) é um tipo de rede neural concebido especificamente para dados dispostos em grade, como imagens. Sua característica distintiva é que, em vez de conectar cada neurônio a todos os pixels da imagem, cada neurônio processa apenas uma pequena vizinhança, e o mesmo conjunto de parâmetros é reaplicado em todas as posições. Esses parâmetros aprendidos são denominados *pesos*: são os números que a rede ajusta durante o treinamento e que determinam quais padrões visuais ela reconhece. Essa organização reduz drasticamente o número de parâmetros e permite que um padrão aprendido em uma região da imagem seja reconhecido em qualquer outra.

A operação central dessas redes é a *convolução*. Ela consiste em deslizar uma pequena matriz de pesos sobre a imagem e, em cada posição, multiplicar os valores dessa matriz pelos valores dos pixels correspondentes e somar os produtos, gerando um único valor de saída. Repetindo essa operação em todas as posições, obtém-se uma nova imagem, denominada *mapa de características* (*feature map*), na qual valores elevados indicam as regiões em que o padrão procurado está presente. A pequena matriz de pesos utilizada nessa varredura é o *filtro* ou *kernel*.

As camadas convolucionais são responsáveis por aprender a representação de características das entradas, como diferenciar imagens de gatos e cachorros ou dígitos numéricos. Em imagens em escala de cinza codificadas em 8 bits, formato usual das fotografias digitais, cada pixel assume um valor inteiro entre 0 e 255. Tons escuros, cinzas intermediários e claros são representados por valores próximos a 0, 128 e 255, respectivamente. Imagens coloridas seguem o mesmo princípio, com três desses valores por pixel, correspondentes aos canais vermelho, verde e azul.

Para detectar linhas verticais automaticamente, aplica-se à imagem um filtro (ou *kernel*) com orientação vertical, isto é, uma matriz de pesos, tipicamente de 3×3 ou 5×5 elementos, cujos valores são elevados na direção vertical e baixos nas demais. Ao ser deslizada sobre a imagem, essa matriz produz respostas altas onde há transições verticais de intensidade e respostas próximas de zero nas demais regiões. Em uma CNN, os valores desses filtros não são definidos manualmente: são aprendidos durante o treinamento, de modo que a rede descobre por si mesma quais padrões são úteis para a tarefa.

O tamanho dos filtros, geralmente representado como $(n \times n)$, é uma variável crucial na resolução de problemas usando redes neurais convolucionais. Esses filtros são aplicados diretamente à entrada, que é frequentemente uma imagem, para extrair características convolucionais.

O processo de aplicação do filtro envolve a multiplicação de elementos correspondentes do filtro e da imagem, seguida pela soma desses produtos (Figura 1.2).

Este processo é repetido em toda a imagem, resultando em um mapa de características. O resultado da convolução é então passado para a próxima camada após a adição de um viés e a aplicação de uma função de ativação adequada.

Após cada operação de convolução, temos a aplicação de uma função de Unidade Linear Retificada (ReLU), que transforma o mapa de recursos e introduz a não linearidade.

$$\begin{pmatrix} 6 & 3 & 3 & 7 & 8 \\ 4 & 8 & 6 & 2 & 5 \\ 3 & 6 & 2 & 9 & 4 \\ 5 & 4 & 7 & 7 & 7 \\ 3 & 3 & 6 & 5 & 3 \end{pmatrix} \times \begin{pmatrix} 1 & 0 & -1 \\ 1 & 0 & -1 \\ 1 & 0 & -1 \end{pmatrix} = \begin{pmatrix} 2 & -1 & -6 \\ -3 & 0 & -1 \\ -4 & -8 & 1 \end{pmatrix}$$

Imagem
Filtro
Kernel

Representação de
Características

Figura 1.2. Exemplo de processo de convolução em processamento de imagem. Fonte: Elaboração própria.

1.2.2.2. Camada de pooling

A camada de *pooling* é inserida entre as camadas convolucionais e tem por função resumir a informação contida no mapa de características, reduzindo sua resolução. A operação divide o mapa em pequenas janelas, em geral de 2×2 elementos, e substitui cada janela por um único valor: o maior deles, no *max pooling*, ou a média, no *average pooling* (Figura 1.3). O efeito prático é duplo. Por um lado, diminui o custo computacional das camadas seguintes, já que o mapa passa a ter cerca de um quarto dos elementos. Por outro, confere *invariância a translações*: como se retém apenas o valor representativo de cada janela, pequenos deslocamentos do objeto na imagem deixam de alterar a saída, e a rede passa a reconhecer o padrão independentemente de sua posição exata.

Ao reduzir a dimensão dos mapas de características, o pooling diminui o número de conexões entre as camadas convolucionais, o que alivia a carga computacional do processamento.

Tipos de pooling:

- Pooling Máximo (Max Pooling): Seleciona o valor máximo dentro de uma região específica do mapa de características.
- Pooling Médio (Average Pooling): Calcula a média dos valores dentro de uma região específica do mapa de características.

- Pooling L_p (onde p pode ser qualquer valor): É uma generalização do pooling médio e máximo, permitindo o uso de diferentes potências para o cálculo.
- Pooling Misto (Mixed Pooling): Combina diferentes tipos de pooling em uma única camada.

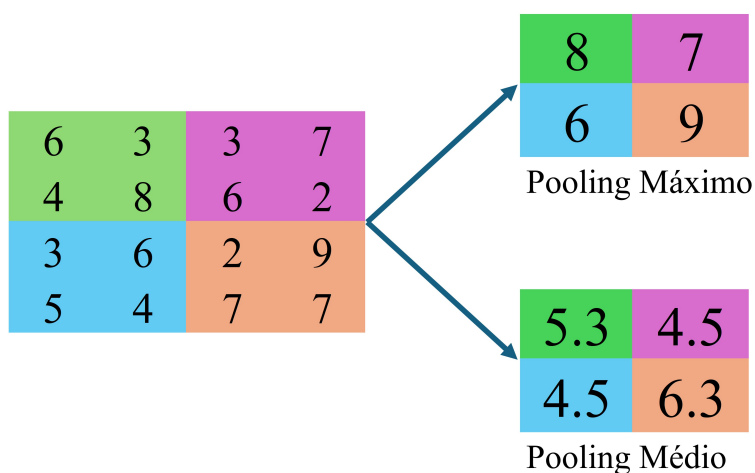


Figura 1.3. Exemplo de aplicação de pooling máximo e médio com a função de reduzir a resolução dos mapas de recursos. Fonte: Elaboração própria.

1.2.2.3. Camadas totalmente conectadas

Após sequências de camadas convolucionais e de pooling, podem-se ter diversas camadas totalmente conectadas. Ao contrário das camadas convolucionais, nas quais cada neurônio se conecta apenas a uma pequena região da camada anterior e compartilha os mesmos pesos com os demais, aqui cada neurônio se conecta a todos os neurônios da camada anterior, com pesos próprios (Figura 1.4). Enquanto as camadas convolucionais extraem características visuais locais, as camadas totalmente conectadas combinam essas características para produzir a decisão final da rede, como a classe do objeto ou as coordenadas de uma caixa delimitadora.

A camada final em uma CNN é a camada de saída, responsável pela predição final. Para tarefas de classificação com múltiplas classes (por exemplo, o dataset MNIST de dígitos escritos à mão), a função Softmax é comumente usada para prever a probabilidade de cada classe. Já para classificações binárias (por exemplo, diferenciar imagens de gatos e cachorros), a função Sigmoid é mais indicada.

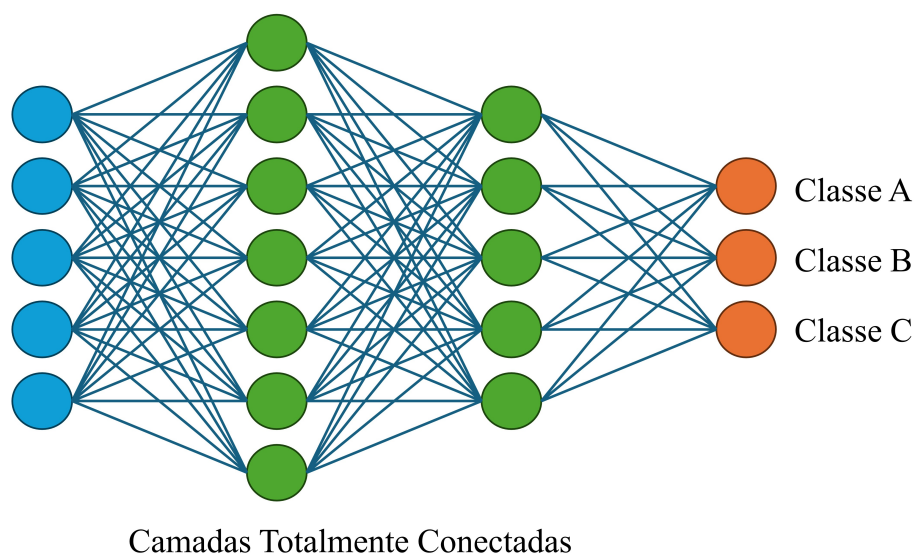


Figura 1.4. Exemplo de camadas totalmente conectadas em redes neurais convolucionais. Fonte: Elaboração própria.

Na Figura 1.5, é apresentado o conjunto dos componentes da rede neural convolucional. As camadas são organizadas em um fluxo de dados que se estende da entrada (a imagem do animal) até a saída (a classificação do animal). O fluxo de dados inicia na Entrada com a imagem de um animal, como um cachorro. O Mapa de Características processa a imagem através de convolução (retângulos verdes empilhados) e pooling, extraindo características essenciais. A aplicação de filtros (Kernels) e da função de ativação Unidade Linear Retificada (ReLU) transformam os dados para a próxima etapa. As Camadas Totalmente Conectadas, representadas por círculos coloridos interconectados, funcionam como neurônios que processam as informações e geram previsões. Na Saída, duas imagens, uma de Cachorro e outra de Gato, demonstram a capacidade da CNN de classificar imagens nestas categorias.

No entanto, embora seja uma abordagem simplificada para entender e começar a estudar as redes neurais convolucionais, é importante ressaltar que agora existem processos e arquiteturas muito mais complexos.

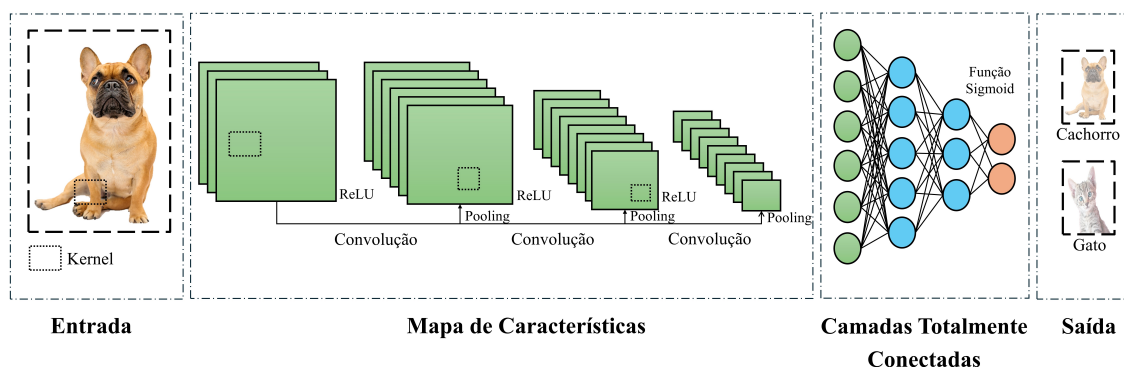


Figura 1.5. Processo de classificação de imagens com rede neural convolucional. Fonte: Elaboração própria.

1.2.3. Funções de ativação em redes neurais convolucionais

As redes neurais se consolidaram como ferramentas excepcionais para solucionar problemas complexos em diversos campos. A ascensão da aprendizagem profunda elevou ainda mais sua importância e aplicabilidade prática. Um fator crucial para o sucesso das redes neurais reside na escolha de uma função de ativação adequada, responsável por introduzir a não linearidade no modelo, permitindo que a rede aprenda relações complexas e modele com maior precisão os dados. Atualmente, o universo das redes neurais artificiais conta com mais de 400 funções de ativação, cada uma com suas características e vantagens, o que exige uma seleção criteriosa para otimizar o desempenho da rede em cada caso (KUNC; KLÉMA, 2024).

Para camadas ocultas de CNNs, a função Rectified Linear Unit (ReLU) é a preferida devido à sua diferenciação simples e rapidez de processamento, sendo normalmente aplicada após a operação de convolução. Essa preferência é corroborada por avaliações sistemáticas, nas quais a ReLU e suas variantes Leaky ReLU e PReLU figuram como as escolhas mais frequentes em redes convolucionais, ao passo que as funções Sigmoid e Tanh devem ser evitadas nas camadas ocultas desse tipo de rede, por conduzirem a uma convergência deficiente (DUBEY; SINGH; CHAUDHURI, 2022).

A ReLU acelera o aprendizado da rede e exige menos recursos computacionais do que outras funções como sigmoid ou tanh. Porém, a ReLU tem um ponto fraco: alguns neurônios podem "morrer" durante o treinamento, tornando sua saída permanentemente zero (GLOROT; BORDES; BENGIO, 2011). Isso acontece quando uma atualização de peso gera um sinal muito forte na unidade. Apesar disso, a ReLU costuma levar a um aprendizado mais rápido que a função sigmoide e pode ser aproximada por outras funções matemáticas eficientes. Ademais, ela serviu de base para a verificação de redes neurais usando técnicas de lógica formal (ALEKSANDROV; VÖLLINGER, 2023). Uma ReLU é definida matematicamente como o máximo de zero e o

valor de entrada:

Onde: $f(x)$ é a saída da função ReLU, e x é a entrada da função ReLU.

A ReLU se define da seguinte forma:

- Se o valor de entrada (x) for positivo, a função ReLU retorna o próprio valor (x).
- Se o valor de entrada (x) for negativo, a função ReLU retorna zero.

Em outras palavras, a ReLU "corta" a parte negativa da entrada e deixa passar a parte positiva. Na Figura 1.6 pode-se observar o gráfico da função ReLU. O gráfico mostra uma linha reta com inclinação 1 para valores de x maiores ou iguais a 0, e uma linha horizontal em $y = 0$ para valores de x menores que 0. Essa característica torna a ReLU mais eficiente do que outras funções de ativação, como a sigmoide, que são mais complexas e computacionalmente custosas.

Para resolver o problema de "neurônios mortos" no ReLU original, diversas modificações e derivações foram propostas, como Leaky ReLU, Very Leaky ReLU e ReLU Paramétrico, que permitem fluxo de gradiente em qualquer entrada. Outras abordagens, como a Unidade Linear Exponencial e Softplus, utilizam suavização para o mesmo fim.

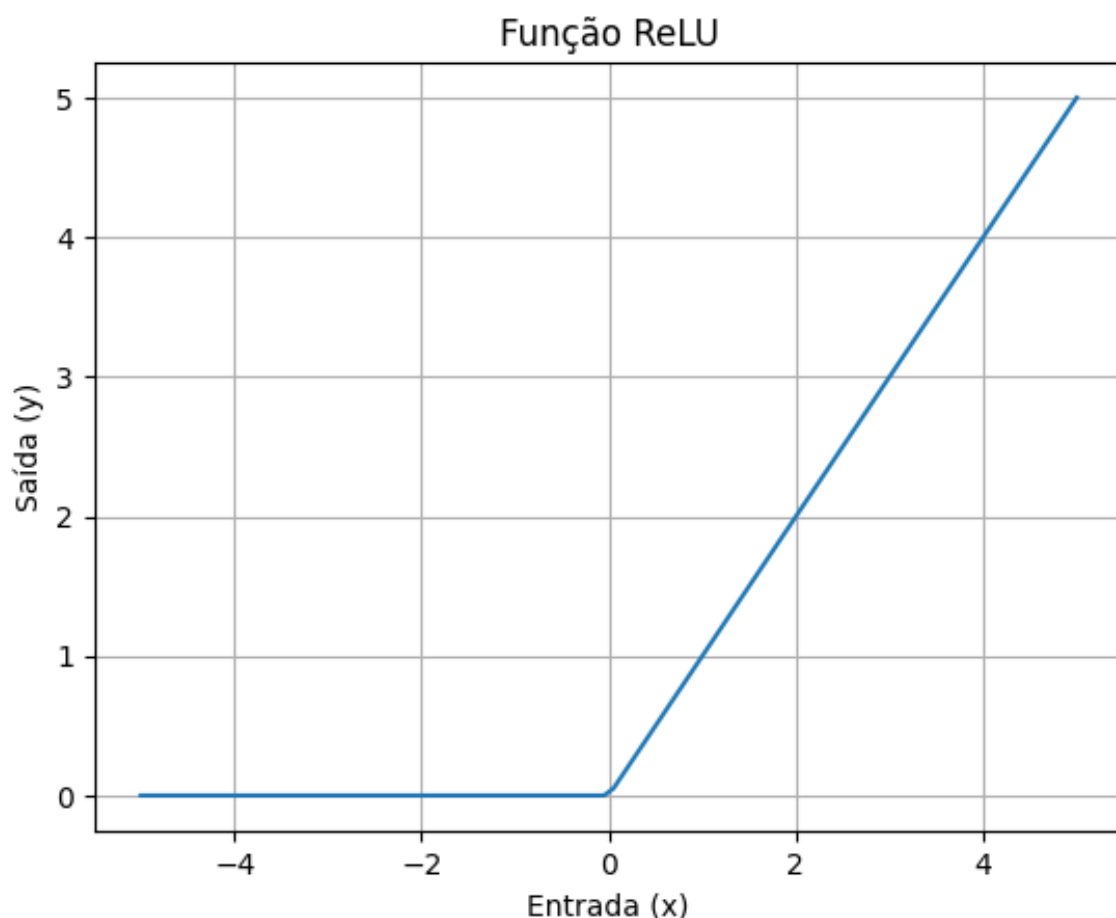


Figura 1.6. Gráfico da função Rectified Linear Unit (ReLU). Fonte: Elaboração própria.

Na etapa de convolução de imagens em CNNs, as funções Sigmoid e tanh não são tão populares quanto a ReLU. A razão para isso reside no comportamento dos gradientes dessas funções.

Para entradas muito grandes, Sigmoid e tanh geram gradientes próximos a zero, o que dificulta o processo de aprendizado da rede (HAO et al., 2020).

A etapa da rede neural totalmente conectada tipicamente utiliza diversas funções de ativação ao longo de suas camadas. A escolha da função varia entre camadas ocultas e a camada de saída. Qualquer função contínua pode ser utilizada como ativação, incluindo a função linear ($g(z) = z$), comum na camada de saída para tarefas de regressão. Já nas camadas ocultas, onde a rede aprende características complexas dos dados, funções não-lineares são predominantes. Nesta etapa as funções de ativação mais usadas são: ReLU, Sigmoid, Tanh, e Softmax (Tabela 1.1).

A escolha da função de ativação ideal depende muito do problema que sua rede neural precisa resolver e do intervalo esperado para os valores de saída. Na camada de saída, por exemplo, se a rede precisa prever valores maiores que 1, funções como tanh e sigmoid não são adequadas, sendo a ReLU a opção mais viável. Já para saídas entre 0 e 1 ou -1 e 1, a ReLU perde espaço para sigmoid e tanh. Para classificação multiclasse, onde o objetivo é gerar uma distribuição de probabilidade entre classes mutuamente exclusivas, a função softmax é imprescindível na última camada. Nas camadas ocultas, no entanto, a ReLU geralmente reina, introduzindo a não-linearidade necessária com relativa simplicidade (SEYREK; UYSAL, 2023).

Tabela 1.1. Comparação das funções de ativação mais usadas em redes neurais convolucionais na etapa da rede neural totalmente conectada. Fonte: Seyrek e Uysal (2023).

Função	Descrição	Vantagens	Desvantagens	Tarefas	Camadas
ReLU	Unidade Retificadora Linear	Simples, eficiente, não linear	Gradiente zero para entradas negativas	Regressão, classificação	Ocultas
Sigmoid	Função Sigmoid	Curva em S, valores entre 0 e 1	Gradiente decrescente para entradas grandes	Regressão, classificação binária	Saída
Tanh	Tangente Hiperbólica	Curva em S, valores entre -1 e 1	Gradiente decrescente para entradas grandes	Regressão, classificação binária	Saída
Softmax	Normalização de Probabilidades	Converte em probabilidades	-	Classificação multiclasse	Saída

A escolha da função de ativação pode ser complexa, no entanto, existem algumas considerações a serem levadas em conta que podem ajudar diferentes desenvolvedores. As recomendações a seguir sintetizam as diretrizes propostas por Dubey, Singh e Chaudhuri (2022), que compararam 18 funções de ativação em diferentes arquiteturas e conjuntos de dados, por Apicella et al. (2021) e por Kunc e Kléma (2024):

- Evitar as funções Sigmoid e Tanh nas camadas ocultas de redes neurais convolucionais, uma vez que a saturação dos gradientes compromete a convergência do treinamento (HAO et al., 2020; DUBEY; SINGH; CHAUDHURI, 2022).
- Adotar a ReLU como escolha inicial para as camadas ocultas, pela simplicidade, pelo baixo custo computacional e pelo desempenho consistente entre diferentes arquiteturas (DUBEY; SINGH; CHAUDHURI, 2022).
- Recorrer às variantes Leaky ReLU, PReLU ou ELU quando houver evidência de neurônios mortos, pois essas funções preservam o fluxo de gradiente para entradas negativas (GLOROT; BORDES; BENGIO, 2011; DUBEY; SINGH; CHAUDHURI, 2022).
- Considerar funções suaves e auto-reguladas, como Swish (SiLU) e Mish, sobretudo em redes profundas com conexões residuais, nas quais elas têm apresentado ganhos de acurácia em relação à ReLU (MISRA, 2019; DUBEY; SINGH; CHAUDHURI, 2022).
- Utilizar funções não lineares nas camadas ocultas para extrair características complexas e, na camada de saída, adotar a função compatível com a natureza da tarefa e com o intervalo de valores desejado: linear para regressão, sigmoide para classificação binária e softmax para classificação multiclasse (SEYREK; UYSAL, 2023).
- Testar empiricamente mais de uma função para a mesma tarefa, mantendo fixos os demais elementos do protocolo de treinamento, uma vez que o desempenho relativo das funções de ativação depende da arquitetura e do conjunto de dados (APICELLA et al., 2021; DUBEY; SINGH; CHAUDHURI, 2022).

Essas recomendações são coerentes com as configurações adotadas nas arquiteturas YOLO utilizadas nos capítulos seguintes desta tese, que combinam ReLU e Leaky ReLU nos blocos convolucionais do *backbone* com funções suaves do tipo SiLU e Mish nos blocos de detecção (MISRA, 2019; KUNC; KLÉMA, 2024).

1.2.4. Operação de detecção de objetos do algoritmo YOLO

O YOLO (You Only Look Once) é um método de detecção de objetos rápido e preciso que pode ser utilizado em tempo real, sendo aplicável em diversas áreas, tais como veículos autônomos, segurança e controle de qualidade. Em veículos autônomos, por exemplo, o YOLO

é empregado para detectar outros veículos, pedestres e ciclistas, enquanto na segurança é utilizado para identificar pessoas e objetos em locais públicos. Além disso, o YOLO oferece vantagens significativas, como velocidade, sendo mais rápido que outros métodos de detecção de objetos, precisão na detecção e eficiência no uso de memória e poder de processamento. Essas características tornam o YOLO uma opção altamente eficaz e versátil para uma variedade de aplicações (TERVEN; CORDOVA-ESPARZA, 2023; TERVEN; CORDOVA-ESPARZA; ROMERO-GONZÁLEZ, 2023).

O YOLO processa a imagem inteira de uma só vez, em vez de analisar pedaços menores, o que permite uma detecção de objetos muito mais rápida. A Figura 1.7 ilustra o processo de detecção de objetos do modelo YOLO, conforme descrito no artigo original do YOLO (You Only Look Once: Unified, Real-Time Object Detection), que essencialmente aborda a detecção como um problema de regressão. A imagem é dividida em uma grade $S \times S$, onde cada célula é responsável por detectar objetos dentro de sua área. Para cada classe de objeto (C classes), o YOLO prevê B caixas delimitadoras em cada célula da grade. Cada caixa delimitadora é definida por cinco valores: a pontuação de confiança (P_c), que indica a probabilidade de a caixa conter um objeto da classe especificada, as coordenadas do centro da caixa (b_x, b_y), que indicam sua posição na célula da grade, e a altura (b_h) e largura (b_w) da caixa. O YOLO gera um tensor de tamanho $S \times S \times (B \times 5 + C)$, contendo todas as informações sobre as caixas delimitadoras previstas. O fator 5 corresponde justamente aos cinco valores descritos acima, necessários para caracterizar completamente cada caixa, e o termo C , ao número de classes consideradas (REDMON et al., 2016).

A descrição do processo de detecção de objetos do modelo YOLO na Figura 1.7 revela uma abordagem detalhada e eficiente. À esquerda, a imagem original é subdividida em uma grade 5×5 , destacando áreas que contêm um cachorro, um carro e uma bicicleta. No painel central, as caixas delimitadoras são aplicadas aos objetos identificados (cachorro, carro e bicicleta), cada uma acompanhada de um nível de confiança associado. Abaixo deste painel, um mapa de probabilidade de classe é apresentado, onde cores distintas representam diferentes classes ou tipos de objetos detectados. Por fim, no painel à direita, as detecções finais são exibidas, com o cachorro, o carro e a bicicleta envolvidos em caixas delimitadoras, destacando o êxito na identificação dos três objetos.

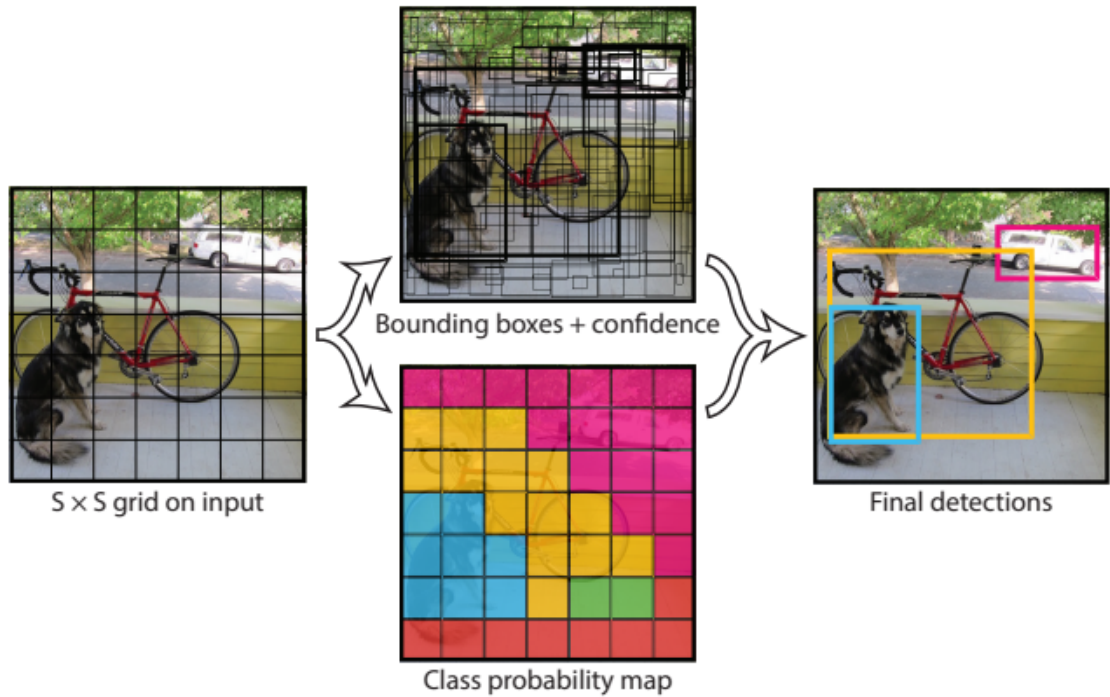


Figura 1.7. Descrição do processo de detecção de objetos do modelo YOLO. Fonte: Redmon et al. (2016).

Para entender os valores da caixa delimitadora, vamos nos referir diretamente à imagem mostrada na Figura 1.8. O primeiro valor (Object-Class) corresponde à identificação da classe do objeto, enquanto o segundo (x) (Equação (1.1)) e o terceiro (y) (Equação (1.2)) valores representam o ponto médio da caixa delimitadora. Esse ponto médio é relativo à largura e à altura da imagem. Os valores seguintes, o quarto (largura ou width [w]) (Equação (1.3)) e o quinto (altura ou height [h]) (Equação (1.4)), denotam a largura e a altura da caixa delimitadora, respectivamente. Normalmente, os conjuntos de dados são divididos em conjuntos de treinamento, teste e validação, com uma distribuição de tamanho de 80%, 10% e 10%, respectivamente (CASAS et al., 2023).

$$x_n = \frac{x_c}{W_{\text{imagem}}} \quad (1.1)$$

$$y_n = \frac{y_c}{H_{\text{imagem}}} \quad (1.2)$$

$$w_n = \frac{W_{\text{caixa}}}{W_{\text{imagem}}} \quad (1.3)$$

$$h_n = \frac{H_{\text{caixa}}}{H_{\text{imagem}}} \quad (1.4)$$

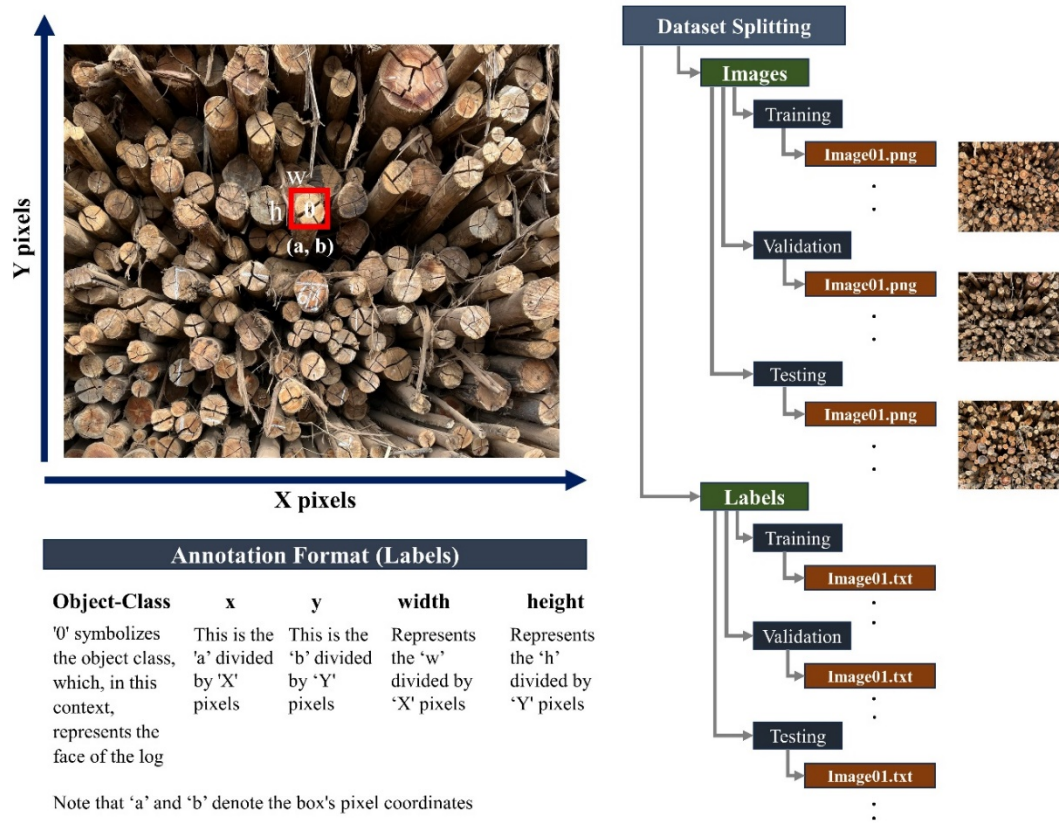


Figura 1.8. Exemplo de formato de anotação e divisão de conjunto de dados em uma imagem de madeira de eucalipto empilhada. Fonte: Casas et al. (2023).

Essas células de grade simplificam operações relacionadas à estimativa de caixas delimitadoras e probabilidades de classe. Cada célula é responsável por detectar os objetos cujo centro nela se situa e atribui a cada caixa predita um *escore de confiança*, que não se confunde com a probabilidade de existência do objeto: o escore resulta do produto entre essa probabilidade e a Interseção sobre União entre a caixa predita e a caixa de referência (Equação 1.5). Dessa forma, uma caixa só recebe confiança elevada quando o modelo julga provável a presença do objeto e a caixa predita se sobrepõe adequadamente à caixa verdadeira.

$$C = p \times \text{IoU}(B, B_{gt}) \quad (1.5)$$

Em que C é o escore de confiança da caixa predita, p é a probabilidade de o objeto estar contido na caixa e IoU é a Interseção sobre União entre a caixa predita (B) e a caixa de referência (B_{gt}).

A interseção sobre a união é calculada usando a seguinte equação (Equação 1.6):

$$\text{IoU}(B, B_{gt}) = \frac{\text{área}(B \cap B_{gt})}{\text{área}(B \cup B_{gt})} \quad (1.6)$$

Em que a Interseção sobre União (IoU) é definida como a área da interseção entre as caixas delimitadoras verdadeira e prevista ($B \cap B_{gt}$) dividida pela área da união das duas caixas

($B \cup Bgt$).

No contexto do algoritmo de detecção de objetos, a caixa alvo ou meta é indicada como Bgt , enquanto a caixa prevista é representada como B . A probabilidade (p) indica se o objeto existe dentro da caixa delimitadora detectada. A métrica de Interseção sobre União (IoU), definida pela Equação 1.6, calcula a área de interseção entre as caixas delimitadoras verdadeiras e previstas. Isso determina uma área aceitável para cada objeto detectado na imagem de entrada e toma decisões com base nisso. Para obter a caixa delimitadora mais adequada, o valor de confiança é aplicado após a estimativa.

O cálculo do IoU envolve três etapas: a determinação da interseção, que é a área de sobreposição entre as caixas predita e real; a união, que é a área total coberta pelas duas caixas, calculada como a soma das áreas individuais das caixas menos a área de interseção; e, finalmente, a fórmula do IoU, que é a razão entre a área de interseção e a área de união. O IoU varia de 0 a 1, onde um valor próximo de 1 indica uma sobreposição significativa entre as caixas, o que é desejável, enquanto um valor próximo de 0 indica que as caixas não se sobrepõem ou têm uma sobreposição mínima. Essa métrica é crucial na avaliação da precisão das previsões de caixas delimitadoras em tarefas como detecção de objetos, onde a precisão da localização é fundamental, ajudando a penalizar previsões imprecisas (BOESCH, 2024).

O IoU varia com diferentes níveis de desempenho de detecção:

- Desempenho de detecção fraco: Neste cenário, a caixa delimitadora prevista mal se sobrepõe à caixa delimitadora verdadeira, resultando em uma pontuação de IoU baixa, próxima a 0. Isso indica uma grande disparidade entre as localizações do objeto previsto e real.
- Bom desempenho de detecção: Quando a caixa delimitadora prevista se alinha de perto com a caixa delimitadora verdadeira, capturando a maior parte do objeto, a pontuação de IoU é alta, se aproximando de 1. Os valores de IoU aceitáveis geralmente estão acima de 0,5, enquanto os bons valores de IoU estão acima de 0,7. Isso demonstra uma localização precisa e uma detecção eficaz.
- Excelente desempenho de detecção: No caso de alinhamento perfeito entre as caixas delimitadoras previstas e verdadeiras, a pontuação de IoU atinge quase a seu valor máximo de 1. Isso significa uma detecção impecável, onde a caixa delimitadora prevista corresponde precisamente aos limites reais do objeto.

A Figura 1.9 exemplifica a variação de desempenho na avaliação da detecção de objetos através do cálculo da Interseção sobre União (IoU). Em diferentes casos, podemos observar como a IoU varia, refletindo a qualidade das detecções realizadas pelo modelo em relação aos objetos reais presentes na cena. Essa análise detalhada da IoU proporciona insights importantes sobre a capacidade do algoritmo em identificar com precisão os objetos de interesse, contribuindo para o aprimoramento contínuo dos sistemas de detecção de objetos.

Outras métricas como a precisão e o recall são duas métricas fundamentais utilizadas para avaliar o desempenho de um modelo de visão computacional em uma tarefa de detecção. A precisão mede a acurácia das caixas delimitadoras previstas, enquanto o recall mensura a capacidade do modelo de detectar todas as instâncias do objeto. A precisão é a proporção de detecções corretas em relação ao total de detecções realizadas pelo modelo, ou seja, a razão entre os verdadeiros positivos e a soma dos verdadeiros positivos com os falsos positivos (Equação 1.7). Ela enfoca a qualidade das previsões, indicando o quanto se pode confiar em uma detecção anunciada pelo modelo. A revocação, por sua vez, é a proporção de objetos existentes que o modelo conseguiu detectar, isto é, a razão entre os verdadeiros positivos e a soma dos verdadeiros positivos com os falsos negativos (Equação 1.8). Ele enfatiza a capacidade do modelo de identificar todas as instâncias relevantes, independentemente de algumas detecções irrelevantes também estarem incluídas. A Figura 1.10 mostra um exemplo de detecção como verdadeiro positivo, falso positivo e falso negativo em diferentes limiares de IoU.

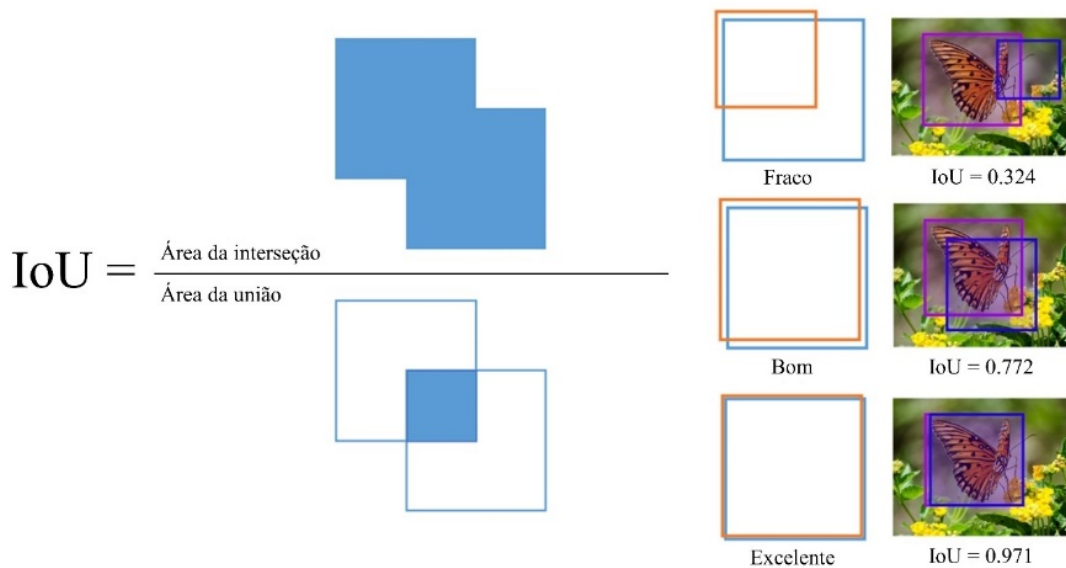


Figura 1.9. Exemplo dos diferentes desempenhos existentes na avaliação da detecção de objetos usando a Interseção sobre União (IoU). Nota: Imagem modificada baseada em Boesch (2024).

Essas duas métricas são frequentemente utilizadas em conjunto para avaliar o equilíbrio entre precisão e recall. Um modelo de alta precisão pode perder algumas instâncias relevantes, enquanto um modelo de alto recall pode retornar alguns resultados irrelevantes. Equilibrar essas duas métricas é crucial para alcançar um desempenho ótimo em tarefas de detecção de objetos (TING, K., 2011).

$$\text{Precisão} = \frac{TP}{TP + FP} \quad (1.7)$$

$$\text{Recall} = \frac{TP}{TP + FN} \quad (1.8)$$

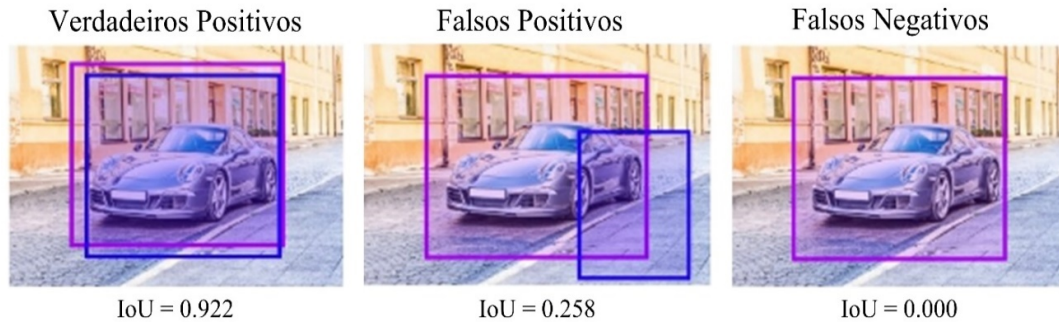


Figura 1.10. Exemplo de detecção como verdadeiro positivo, falso positivo e falso negativo em diferentes limiares de IoU. Nota: Imagem modificada baseada em Boesch (2024).

A relação entre precisão e recall pode ser visualizada graficamente através de uma curva chamada Curva Precision-Recall. A Figura 1.11 ilustra essa relação, em que o eixo x, temos o Recall, variando de 0.0 a 0.5, enquanto no eixo y, temos a Precisão, variando de 0.6 a 1.0. Existem três curvas no gráfico, cada uma representando diferentes limiares de IoU (Intersecção sobre União): 0.4, 0.55 e 0.7. Todas as curvas começam com alta precisão em valores baixos de recall, mas diminuem em precisão à medida que o recall aumenta. A curva azul para o limiar de IoU = 0.4 mantém uma precisão mais alta em comparação com as outras duas curvas em todos os níveis de recall. A curva laranja para o limiar de IoU = 0.55 mostra valores moderados de precisão em toda a gama de recalls. A curva verde para o limiar de IoU = 0.7 tem o declínio mais acentuado, indicando uma precisão mais baixa em níveis aumentados de recall (JAIN et al., 2019). Essas curvas mostram a troca entre precisão e recall para diferentes limiares de IoU. À medida que ao tentar aumentar o recall (reduzindo os falsos negativos), podemos acabar aumentando os falsos positivos (diminuindo a precisão). Um limiar mais alto aumenta a precisão ao reduzir os falsos positivos, mas diminui o recall ao perder mais positivos. Por outro lado, um limiar mais baixo aumenta o recall ao capturar mais positivos, mas diminui a precisão ao permitir mais falsos positivos.

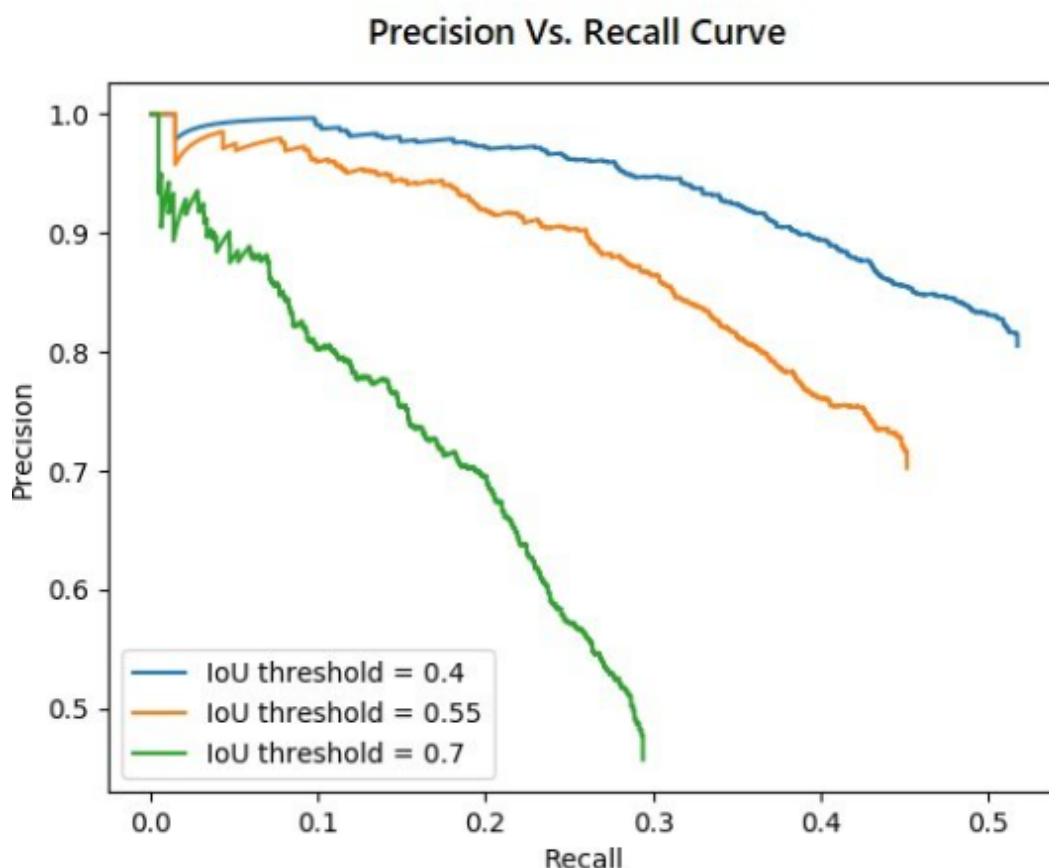


Figura 1.11. Curva de relação entre precisão e recall em diferentes limites de IoU. Fonte: Jain et al. (2019).

A Precisão Média (mAP) calculada a um limiar de *intersection over union* (IoU) de 50% (mAP50) é outra métrica comum utilizada para avaliar o desempenho dos modelos de detecção de objetos com YOLO. A mAP representa a média das precisões médias (AP) calculadas para cada classe de objeto. O limiar de 50% refere-se à sobreposição mínima exigida entre a caixa predita e a caixa de referência para que a detecção seja considerada correta, e não ao grau de confiança atribuído pelo modelo à predição.

Um valor alto de mAP50 (próximo a 1,0) indica que o modelo possui uma alta precisão, demonstrando sua habilidade em fazer previsões corretas e minimizar previsões falsas. Além disso, um alto valor de mAP50 sugere consistência na precisão do modelo em todas as classes de objetos. Por outro lado, um valor baixo de mAP50 (próximo a 0,0) indica uma baixa precisão do modelo, o que implica em um grande número de previsões falsas. Além disso, um valor baixo de mAP50 pode sugerir uma falta de consistência na precisão do modelo entre diferentes classes de objetos.

Assim como o mAP50, também é possível avaliar outros limiares de IoU, como o mAP50-95, que indica a média da precisão média calculada com diferentes limiares de IoU, variando de 0,50 a 0,95. Isso oferece uma visão abrangente do desempenho do modelo em diferentes níveis de dificuldade de detecção.

A Figura 1.12, exibe a precisão, recall e mAP (Mean Average Precision) em diferentes limiares de IoU, treinados em conjuntos de dados específicos. No eixo X, temos os limiares de IoU em porcentagem, variando de 20% a 95%, enquanto o eixo Y representa a porcentagem das métricas mencionadas, variando de 0,6 a 1. Os valores máximos alcançados por cada curva foram de 91,1% para a precisão, 94,8% para o recall, 93,4% para o mAP@50 e 84,0% para o mAP@50:95. Cabe observar que esses máximos não ocorrem no mesmo limiar de IoU, razão pela qual não devem ser interpretados como um conjunto de resultados obtidos em uma única condição de avaliação: a precisão atinge seu valor máximo em limiares baixos, ao passo que o recall o atinge em limiares elevados. Observa-se, ainda, que quanto maior o limiar de IoU, mais desafiadora se torna a obtenção de detecções eficientes (MATHEW et al., 2023).

A pontuação F1 é uma métrica de avaliação crítica que representa a média harmônica de precisão e recall. Ela fornece uma avaliação equilibrada do desempenho do modelo, levando em conta tanto os falsos positivos quanto os falsos negativos. Além disso, o escore F1 é crucial em situações em que a precisão e o recall são igualmente importantes, havendo a necessidade de equilibrar essas duas métricas. Torna-se especialmente relevante em tarefas de classificação binária e multi-classe, onde a precisão sozinha pode não refletir com precisão a capacidade do modelo de distinguir entre diferentes classes (KUNDU, 2022).

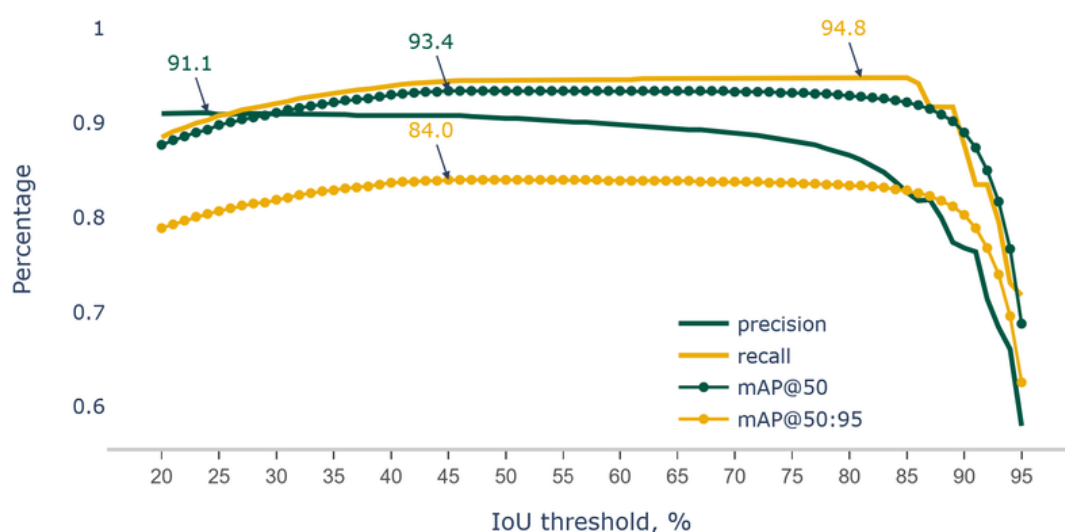


Figura 1.12. Métricas de precisão, recuperação, comparação mAP@50 e mAP@50:95 avaliados no conjunto de dados de teste no limite de IoU variando de 20% a 95%. Fonte: Mathew et al. (2023).

O escore F1 varia de 0 a 1, onde 0 representa o resultado mais baixo possível e 1 denota um resultado perfeito, indicando que o modelo previu com precisão cada rótulo. Um escore F1 alto geralmente indica um desempenho equilibrado do modelo, demonstrando que ele pode alcançar alta precisão e alto recall simultaneamente. Um escore F1 baixo frequentemente significa um compromisso entre recall e precisão, indicando que o modelo enfrenta dificuldades para equilibrar essas duas métricas.

Para calcular a pontuação F1 pode ser usada a seguinte fórmula (Equação 1.9):

$$F_1 = 2 \times \frac{\text{Precisão} \times \text{Recall}}{\text{Precisão} + \text{Recall}} \quad (1.9)$$

Graficamente, também é possível avaliar o desempenho dos modelos de classificação por meio da Curva de Confiança F1. Na Figura 1.13, apresenta-se um exemplo desse gráfico. O modelo demonstra uma pontuação F1 alta (aproximadamente 0,79) em um nível específico de confiança (aproximadamente 0,268), indicando um bom equilíbrio entre precisão e revocação nesse ponto. Contudo, à medida que o limiar de confiança aumenta, a pontuação F1 tende a diminuir. Cabe observar que a curva varia o limiar de aceitação das detecções, e não o modelo: em limiares elevados um número crescente de detecções corretas é descartado, o que reduz a revocação e aumenta os falsos negativos, deprimindo o F1 mesmo com precisão alta. Assim, o gráfico evidencia que o modelo mantém um bom desempenho até certo ponto de confiança, mas além desse limiar, seu desempenho tende a declinar. A análise do gráfico pode ser útil para ajustar o limiar de confiança do modelo e maximizar sua pontuação F1.

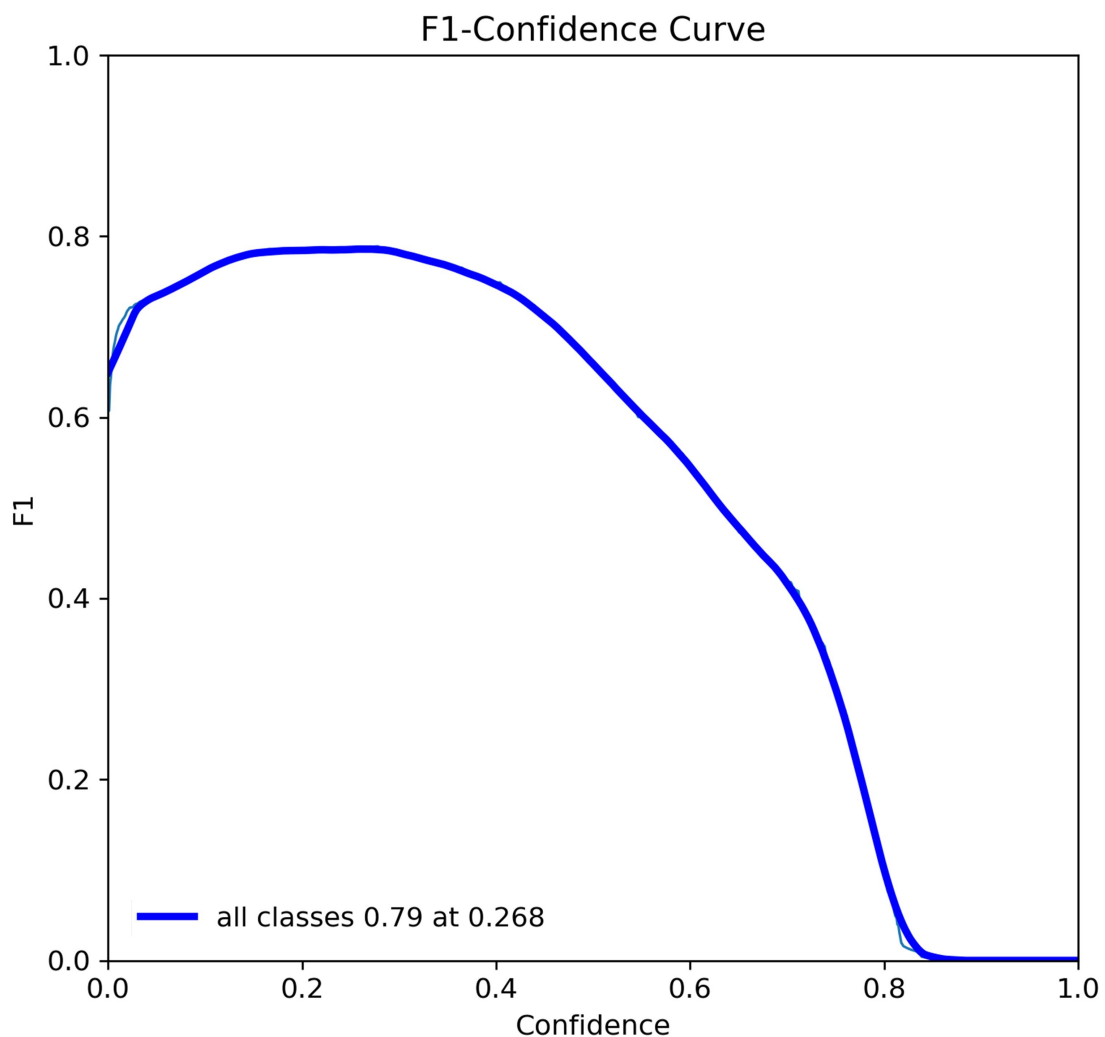


Figura 1.13. Exemplo da Curva de Confiança F1, representando o desempenho de um modelo de classificação. Fonte: Elaboração própria.

Ao interpretar os resultados, é essencial considerar o contexto e os objetivos da tarefa de detecção. Nem sempre um alto valor nas métricas de avaliação indica o sucesso absoluto do modelo. Por exemplo, se o objetivo é contar objetos em uma imagem, pode ser que uma detecção com valores elevados nessas métricas não seja tão relevante quanto a precisão na contagem exata dos objetos (CASAS et al., 2023).

Portanto, ao avaliar o desempenho de um modelo de detecção, é fundamental levar em conta não apenas as métricas padrão, mas também as necessidades específicas e as nuances do contexto em que o modelo será aplicado. Isso garantirá que o modelo seja avaliado de forma completa e adequada às exigências da tarefa em questão.

1.2.5. Evolução progressiva do modelo YOLO na visão computacional

O modelo YOLO (You Only Look Once) é um avanço relativamente recente no campo da visão computacional, com sua primeira iteração, YOLOv1, sendo introduzida em 2016. O YOLOv1 se destacou por sua abordagem unificada e de detecção em uma única etapa. Posteriormente, em 2017, o YOLOv2 foi introduzido, incorporando recursos inovadores como caixas âncora e treinamento em múltiplas escalas. Aproveitando o progresso feito, a versão YOLOv3 emergiu em 2018, incorporando a rede de pirâmide de características para um desempenho aprimorado. Ela envolve a construção de uma pirâmide de escalas (ou níveis de resolução) de uma imagem original. Cada nível da pirâmide representa uma versão da imagem em uma escala diferente. Por exemplo, em tarefas como detecção de objetos ou rastreamento, a rede de pirâmide de características permite que o algoritmo analise a imagem em várias escalas, tornando-o mais robusto a variações de tamanho e aparência (LIN; DOLLÁR; et al., 2017). O ano de 2020 viu a introdução do YOLOv4, que introduziu o "bag of freebies" e o "bag of specials", refinando ainda mais as capacidades do modelo. O termo "Bag of Freebies" refere-se a uma coleção de várias técnicas ou aprimoramentos que são incorporados a um modelo ou algoritmo de aprendizado de máquina sem aumentar significativamente os custos ou a complexidade computacional. Esses acréscimos têm o objetivo de melhorar o desempenho, a robustez ou os recursos de generalização do modelo sem exigir recursos adicionais (ZHANG, ZHI. et al., 2019). No mesmo ano, o YOLOv5 foi estabelecido, ostentando um recurso de âncoras de caixa delimitadora de aprendizado automático, representando mais um avanço significativo na evolução da série YOLO. Em 2022, a versão YOLOv6 foi introduzida, apresentando características de cabeça desacopladas amigáveis ao hardware para otimizar o desempenho em dispositivos específicos. Continuando a série, o YOLOv7 foi estabelecido no mesmo ano, oferecendo recursos treináveis de "bag-of-freebies" e aprimorando ainda mais a adaptabilidade e o desempenho. Finalmente, em 2023, a última iteração YOLOv8 foi lançada, introduzindo recursos de cabeçalho de detecção sem âncora, prometendo melhor precisão e flexibilidade. O YOLOv8 utiliza funções de Complete Intersection over Union (CIoU) e Distribution Focal Loss (DFL) para a perda de caixa delimitadora e entropia cruzada binária para a perda de classificação. Essas funções de perda aprimoraram o desempenho da detecção de objetos, especialmente no contexto de objetos

menores. Em comparação com outros modelos no campo da visão computacional, o modelo YOLO se destaca como uma técnica eficiente de detecção, reformulando a detecção como um problema de regressão, possibilitando previsões rápidas de imagem e superando métodos como o Modelo de Partes Deformáveis (DPM) e a Rede Convolutiva Baseada em Região (R-CNN) em termos de aprendizado de representações de objetos altamente generalizáveis (LIU, WEI et al., 2016; REDMON et al., 2016). Essa evolução progressiva do modelo YOLO demonstra os esforços constantes e inovações na visão computacional para avançar na detecção e reconhecimento de objetos, tornando-o uma ferramenta versátil e poderosa para diversas aplicações do mundo real. Com base nesses critérios, podemos identificar várias vantagens do modelo YOLO, incluindo sua implementação eficiente de detecção de objetos em uma única etapa; seu status de acesso aberto; e o fato de receber atualizações regulares para acompanhar os últimos avanços no campo. YOLOv9 é a mais recente versão do YOLO, lançada em fevereiro de 2024, por Chien-Yao Wang, I-Hau Yeh e Hong-Yuan Mark Liao. É um modelo de detecção de objetos em tempo real aprimorado que visa superar todos os métodos baseados em convolução e baseados em transformadores. O YOLOv9 é lançado em quatro modelos, ordenados pelo número de parâmetros: v9-S, v9-M, v9-C e v9-E. Para melhorar a precisão, ele introduz informações de gradiente programáveis (PGI) e a Rede de Agregação de Camadas Eficientes Generalizadas (GELAN). O PGI evita a perda de dados e garante atualizações de gradiente precisas e o GELAN otimiza modelos leves com planejamento de caminho de gradiente (WANG, CHIEN-YAO; YEH; LIAO, 2024). A evolução histórica e progressiva do modelo YOLO na visão computacional é mostrada na Figura 1.14.

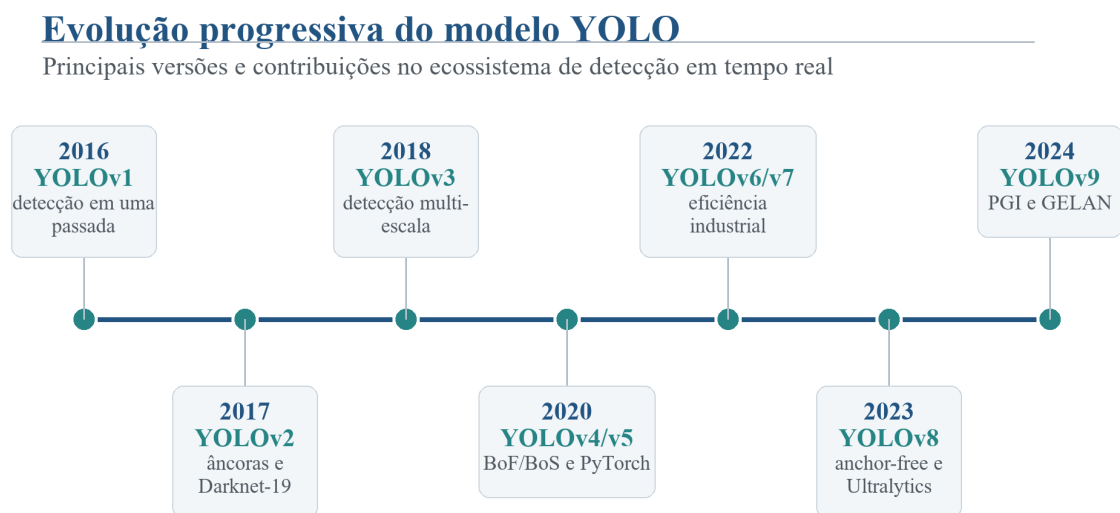


Figura 1.14. Evolução histórica e progressiva do modelo YOLO na visão computacional. Fonte: Elaboração própria.

1.2.5.1. YOLO V1: Você só olha uma vez

O YOLO, de Joseph Redmon e colaboradores, foi publicado no ano 2016 (REDMON et al., 2016). Apresentou, pela primeira vez, uma abordagem de ponta a ponta em tempo real para detecção de objetos. O nome YOLO significa "You Only Look Once" (Você Só Olha Uma Vez), referindo-se ao fato de que era capaz de realizar a tarefa de detecção com uma única passagem pela rede. A abordagem proposta com base no conceito "unificado" possibilitou a previsão simultânea de múltiplas caixas delimitadoras e probabilidades de classe, melhorando tanto a velocidade quanto a precisão (TERVEN; CORDOVA-ESPARZA, 2023).

O objetivo central do YOLO e da detecção de objetos em geral é localizar e detectar objetos por meio do uso de caixas delimitadoras. São necessários dois conjuntos de vetores de caixas delimitadoras: um representando a verdade fundamental e outro sendo a predição do modelo. Para lidar com a presença de múltiplas caixas delimitadoras que não contêm nenhum objeto ou contêm o mesmo objeto, o YOLO utiliza a técnica de Non-Maximum Suppression (NMS) ou "supressão de não máximos" (Figura 1.15). O NMS é uma técnica de pós-processamento comumente usada em tarefas de detecção de objetos. Sua finalidade é eliminar as detecções duplicadas e selecionar as caixas delimitadoras mais relevantes correspondentes aos objetos detectados (RUI; TANG; HAO, 2022). Ao definir um valor de limite para o NMS, seleciona-se primeiro a caixa de maior confiança e eliminam-se todas as demais cuja Intersection Over Union (IoU) com ela seja maior ou igual ao limiar definido, repetindo-se o procedimento com as caixas remanescentes. A sequência do pseudocódigo NMS é mostrada abaixo:

Algoritmo de Supressão Não Máxima (NMS)

Requer: Conjunto de caixas delimitadoras previstas B , pontuações de confiança S , limiar IoU τ , limiar de confiança T

Garante: Conjunto de caixas delimitadoras filtradas F

- 1: $F \leftarrow \emptyset$ (Inicialize F como um conjunto vazio)
- 2: Filtrar as caixas: $B \leftarrow \{b \in B \mid S(b) \geq T\}$ (Mantenha apenas as caixas em B que têm uma pontuação de confiança maior ou igual a T)
- 3: Ordenar as caixas B por suas pontuações de confiança em ordem decrescente
- 4: enquanto $B \neq \emptyset$ faça (Enquanto B não estiver vazio)
- 5: Selecione a caixa b com a pontuação de confiança mais alta
- 6: Adicione b ao conjunto de caixas finais F : $F \leftarrow F \cup \{b\}$ (Adicione b ao conjunto F)
- 7: Remova b do conjunto de caixas B : $B \leftarrow B - \{b\}$ (Remova b de B)
- 8: para todas as caixas restantes r em B faça (Para cada caixa restante r em B)
- 9: Calcule o IoU entre b e r : $iou \leftarrow IoU(b, r)$ (Calcule a intersecção sobre a união (IoU) entre b e r)
- 10: se $iou \geq \tau$ então (Se a IoU for maior ou igual ao limiar τ)
- 11: Remova r do conjunto de caixas B : $B \leftarrow B - \{r\}$ (Remova r de B)
- 12: fim se (end if)

13: fim para (end for)

14: fim enquanto (end while)



Figura 1.15. A imagem à esquerda mostra a saída típica de um modelo de detecção de objetos contendo várias caixas sobrepostas e a imagem à direita mostra a saída após o algoritmo NMS. Fonte: Terven e Cordova-Esparza (2023).

A versão original do YOLO, baseada no framework Darknet, consistia em duas subvariantes. A primeira arquitetura era composta por 24 camadas convolucionais, com a camada final conectada à primeira de duas camadas totalmente conectadas. Enquanto isso, a variante "Fast YOLO" consistia apenas em nove camadas convolucionais com menos filtros cada uma. Inspirado no módulo de início do GoogleNet, foi implementada uma sequência de camadas convolucionais 1x1 para reduzir o espaço de características resultante das camadas anteriores (HUSSAIN, 2023). A arquitetura preliminar para o YOLO-v1 é mostrada na Figura 1.16 e representa o processo de uma imagem sendo processada através do Framework DarkNet. Segue uma análise passo a passo:

- **Entrada (Input):** A entrada é uma imagem de tamanho 448x448x3, que é alimentada no Framework DarkNet.
- **Framework DarkNet:** Este é um modelo de rede neural que processa a imagem de entrada. A estrutura do framework é representada pelas dimensões 7x7x1024.
- **Vetor de Características:** Após o processamento, a imagem é reduzida a um vetor de características de tamanho 4096. Este vetor contém características importantes extraídas da imagem.
- **Saída (Output):** O vetor de características é então transformado em uma saída de dimensão 7x7x30. Esta saída inclui informações sobre a caixa delimitadora (xB), coordenadas (x, y), largura e altura (w, h), pontuação do objeto e probabilidade de classe (xC).

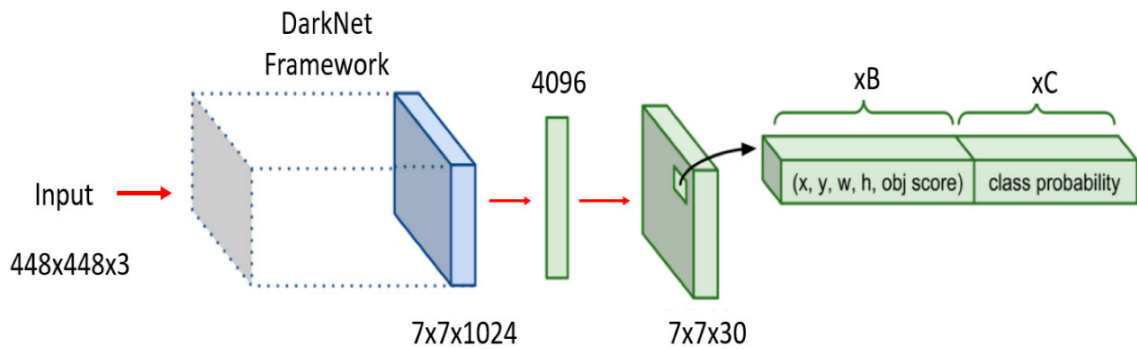


Figura 1.16. Desenvolvimento inicial da arquitetura para o YOLO V1. Fonte: Hussain (2023).

A arquitetura do YOLO versão compreende várias camadas convolucionais, camadas de pooling máximo e camadas totalmente conectadas. Na Tabela 1.2 está um exemplo detalhado da arquitetura:

Tabela 1.2. Descrição das camadas convolutivas e totalmente conectadas, quantidade de filtros, dimensões e passos dos filtros, e dimensões das saídas na arquitetura de YOLO V1. Fonte: Redmon et al. (2016).

Tipo	Filtros	Tamanho/passo	Saída	Repetições
Convolutional	64	$7 \times 7 / 2$	224×224	1
Max pooling	—	$2 \times 2 / 2$	112×112	1
Convolutional	192	$3 \times 3 / 1$	112×112	1
Max pooling	—	$2 \times 2 / 2$	56×56	1
Convolutional	128	$1 \times 1 / 1$	56×56	1
Convolutional	256	$3 \times 3 / 1$	56×56	1
Convolutional	256	$1 \times 1 / 1$	56×56	1
Convolutional	512	$3 \times 3 / 1$	56×56	1
Max pooling	—	$2 \times 2 / 2$	28×28	1
Convolutional	256	$1 \times 1 / 1$	28×28	4
Convolutional	512	$3 \times 3 / 1$	28×28	4
Convolutional	512	$1 \times 1 / 1$	28×28	1
Convolutional	1024	$3 \times 3 / 1$	28×28	1
Max pooling	—	$2 \times 2 / 2$	14×14	1
Convolutional	512	$1 \times 1 / 1$	14×14	2
Convolutional	1024	$3 \times 3 / 1$	14×14	2
Convolutional	1024	$3 \times 3 / 1$	14×14	1
Convolutional	1024	$3 \times 3 / 2$	7×7	1
Convolutional	1024	$3 \times 3 / 1$	7×7	1
Convolutional	1024	$3 \times 3 / 1$	7×7	1
Totalmente conectada	—	—	4096	1

Dropout (0,5)	—	—	4096	1
Totalmente conectada	—	—	$7 \times 7 \times 30$	1

A arquitetura (Tabela 1.2) foi projetada para extrair recursos da imagem de entrada e prever caixas delimitadoras e probabilidades de classe. As camadas convolucionais (convolutional layers) são responsáveis pela extração de recursos. Elas aplicam uma série de filtros à imagem de entrada, criando um conjunto de mapas de recursos. As camadas de pooling máximo (max pooling layers) são usadas para reduzir as dimensões espaciais do volume de entrada, diminuindo a complexidade computacional e consolidando os recursos. As camadas totalmente conectadas (FC) no final da rede pegam os recursos de alto nível aprendidos pelas camadas convolucionais e os utilizam para classificar a imagem de entrada em várias classes (REDMON et al., 2016).

A arquitetura do YOLO se destaca por ser simplificada, ou seja, em vez de realizar a detecção em várias etapas, o YOLO utiliza uma única etapa para processar toda a imagem. Essa técnica inovadora, conhecida como regressão em uma única etapa, impulsiona a velocidade do YOLO, permitindo a operação em tempo real. Embora o YOLO apresente um desempenho veloz, sua precisão na localização de objetos deixa a desejar quando comparada a métodos mais avançados naquela época, como o Fast R-CNN. Essa diferença de performance pode ser atribuída a três fatores principais:

- Limitação na detecção de no máximo dois objetos da mesma classe em cada célula da grade, o que restringia sua capacidade de prever objetos próximos.
- Dificuldade em prever objetos com proporções de aspecto não observadas nos dados de treinamento.
- Aprendizado baseado em características tendenciosas dos objetos devido às camadas de amostragem reduzida.

A Figura 1.17 mostra as principais características da primeira versão do modelo YOLO. YOLO V1

YOLOv1: primeira versão

Modelo unificado para detecção em tempo real

Entrada e grade • Imagem 448 x 448. • Grade S x S.	Predição única • Caixas, confiança e classes. • Um tensor de saída.	Limitações • Objetos próximos. • Objetos pequenos.
Arquitetura • Backbone Darknet. • Camadas totalmente conectadas.		

Figura 1.17. Descrição das características da primeira versão do modelo YOLO.

1.2.5.2. YOLO V2: YOLO9000, melhor, mais rápido, mais forte

O YOLO-v2/9000 foi introduzido para melhorar as ineficiências observadas no YOLO original, mantendo a mesma velocidade. Se destaca por ser destinada à detecção em tempo real de mais de 9000 objetos diferentes. A normalização em lote foi introduzida para melhorar a convergência do modelo e acelerar o treinamento. A arquitetura foi treinada com imagens de 448 × 448 pixels para ajustar os filtros internos ao lidar com imagens de maior resolução. As caixas de ancoragem foram adicionadas para as previsões das caixas delimitadoras, substituindo a camada totalmente conectada responsável por prever as caixas delimitadoras. As caixas delimitadoras são usadas para descrever a localização exata de um objeto, enquanto caixas de âncora são uma técnica para gerar várias caixas delimitadoras com diferentes características para melhorar a detecção de objetos em imagens. O YOLO V2 foi treinado em diferentes arquiteturas, como VGG-16 e GoogleNet, e finalmente foi desenvolvida a arquitetura Darknet-19, contendo 19 camadas convolucionais e cinco camadas de maxpooling (Tabela 1.3); desta forma conseguiu a superioridade sobre as arquiteturas da época, como SSD e Faster R-CNN. A Figura 1.18 mostra as principais características da segunda versão do modelo YOLO.

Tabela 1.3. Descrição das camadas convolutivas, quantidade de filtros, dimensões e passos dos filtros, e dimensões das saídas na arquitetura DarkNet-19 de YOLO V2. Fonte: Redmon e Farhadi (2017).

Tipo Filtros	Tamanho/Stride	Saida
Conv	32 3 × 3	224 × 224
Max Pool	2 × 2 / 2	112 × 112

Conv 64 3 × 3 112 × 112
Max Pool 2 × 2 / 2 56 × 56 Conv 128 3 × 3 56 × 56
Conv 64 1 × 1 56 × 56
Conv 128 3 × 3 56 × 56
Max Pool 2 × 2 / 2 28 × 28 Conv 256 3 × 3 28 × 28
Conv 128 1 × 1 28 × 28
Conv 256 3 × 3 28 × 28 Max Pool 2 × 2 / 2 14 × 14 Conv 512 3 × 3 14 × 14
Conv 256 1 × 1 14 × 14
Conv 512 3 × 3 14 × 14
Conv 256 1 × 1 14 × 14
Conv 512 3 × 3 14 × 14
Max Pool 2 × 2 / 2 7 × 7 Conv 1024 3 × 3 7 × 7
Conv 512 1 × 1 7 × 7 Conv 1024 3 × 3 7 × 7
Conv 512 1 × 1 7 × 7
Conv 1024 3 × 3 7 × 7
Conv 1000 1 × 1 7 × 7
Avgpool Global 1000

YOLO V2

YOLOv2 / YOLO9000

Melhorias de precisão, velocidade e capacidade de generalização

Darknet-19 • 19 camadas convolucionais. • Backbone mais eficiente.	Anchor boxes • Melhor localização. • Mais objetos por célula.	Multi-escala • Entrada variável. • Mais robustez.
---	--	--

YOLO9000

- Detecção + classificação.
- Vocabulário ampliado.

Figura 1.18. Descrição das características da segunda versão do modelo YOLO.

1.2.5.3. YOLO V3: Uma melhoria incremental

O YOLO V3 é a terceira versão do algoritmo de detecção de objetos YOLO, que busca superar as limitações das versões anteriores. Enquanto a versão base do YOLO não tinha solução

para erros de localização e a segunda versão falhava em detectar objetos menores, o YOLO V3 oferece uma maneira eficiente de detectar objetos de diversos tamanhos com desempenho aprimorado.

O YOLO V3 utiliza uma poderosa rede backbone chamada Darknet-53 (Figura 1.19). Esta rede, composta por 53 camadas convolucionais (Tabela 1.4), supera sua predecessora, YOLO V2, ao substituir camadas de pool máximo por convoluções com passada e incorporando conexões residuais. Essas modificações aprimoram a eficiência de extração de recursos enquanto mantêm a precisão, tornando o Darknet-53 significativamente mais rápido, utilizando a normalização em lote e ativação do Leaky ReLU (REDMON; FARHADI, 2018).

As conexões residuais, também conhecidas como "skip connections", são um componente fundamental da arquitetura YOLO V3, contribuindo significativamente para seu desempenho aprimorado. Elas funcionam como atalhos que conectam camadas não adjacentes da rede neural, permitindo que o fluxo de informações percorra um caminho mais direto e eficiente (ZHANG, XINLIANG et al., 2021).

Uma inovação chave no YOLO V3 é sua capacidade de prever em múltiplas escalas, resolvendo uma limitação importante em versões anteriores do YOLO que enfrentavam dificuldades na detecção de objetos pequenos. Ao gerar previsões em várias grades de diferentes tamanhos, o YOLO V3 melhora significativamente a precisão da detecção de objetos pequenos, capturando tanto objetos grandes quanto pequenos com caixas delimitadoras mais precisas. Esta abordagem aborda duas melhorias essenciais: detalhamento mais preciso das caixas, graças ao uso de diferentes tamanhos de grade, e aprimoramento na detecção de objetos pequenos, concentrando-se nos recursos extraídos de grades menores. Em vez de utilizar softmax para classificação, adota a entropia cruzada binária (Binary Cross-Entropy, BCE) para treinar classificadores logísticos independentes, viabilizando a atribuição de múltiplos rótulos a uma mesma caixa, ou seja, para cada objeto detectado, se calcula a probabilidade de pertencer a cada classe e compara com os rótulos reais (0 ou 1) para calcular a perda de classificação, dessa forma, a BCE é usada para calcular a perda de classificação para cada classe individualmente (CRESWELL; ARULKUMARAN; BHARATH, 2017; DIWAN; ANIRUDH; TEMBHURNE, 2023).

O YOLO V3 adota uma estratégia de previsão multi-escala, resultando em três saídas (y_1 , y_2 e y_3) distintas com escalas diferentes. A detecção multi-escala é essencial para identificar objetos de diferentes tamanhos de forma eficaz. Para objetos grandes (y_1), a detecção ocorre na camada 82, onde um mapa de recursos de 13×13 é gerado após convoluções com um stride de 32. Em seguida, para objetos médios (y_2), a detecção é realizada na camada 94 com um stride de 16. Nesse processo, convoluções adicionais são aplicadas nas camadas 79 e 61, resultando em um mapa de recursos de 26×26 . Já para objetos pequenos (y_3), o mapa de recursos de 52×52 é utilizado na camada 106, após aplicar um stride de 8. Aqui, convoluções adicionais são realizadas na camada 91 e o resultado é mesclado com a camada 36 usando um kernel de 1×1 . Essa combinação de mapas de recursos sub-amostrados e aumentados em diferentes estágios permite a extração de características detalhadas para detectar objetos menores de várias

dimensões (TERVEN; CORDOVA-ESPARZA, 2023).

A integração do Spatial Pyramid Pooling (SPP) no YOLO V3 foi fundamental para aprimorar a extração de recursos em múltiplas escalas, permitindo uma identificação mais precisa desses objetos de menor tamanho. A adição da camada SPP ao YOLO V3 representa uma solução eficaz para aprimorar o desempenho da detecção de objetos em dados provenientes de Veículos Aéreos Não Tripulados (UAVs) (PEBRIANTO et al., 2023).

Tabela 1.4. Descrição das camadas convolutivas, quantidade de filtros, dimensões e passos dos filtros, e dimensões das saídas na arquitetura DarkNet-53 de YOLO V3. Fonte: Redmon e Farhadi (2018).

Tipo	Filtros	Tamanho/ passo	Saída	Repetições
Convolutacional	32	3×3	256×256	—
Convolutacional	64	$3 \times 3 / 2$	128×128	—
Convolutacional	32	1×1	128×128	1
Convolutacional	64	3×3	128×128	1
Residual	—	—	128×128	1
Convolutacional	128	$3 \times 3 / 2$	64×64	—
Convolutacional	64	1×1	64×64	2
Convolutacional	128	3×3	64×64	2
Residual	—	—	64×64	2
Convolutacional	256	$3 \times 3 / 2$	32×32	—
Convolutacional	128	1×1	32×32	8
Convolutacional	256	3×3	32×32	8
Residual	—	—	32×32	8
Convolutacional	512	$3 \times 3 / 2$	16×16	—
Convolutacional	256	1×1	16×16	8
Convolutacional	512	3×3	16×16	8
Residual	—	—	16×16	8
Convolutacional	1024	$3 \times 3 / 2$	8×8	—
Convolutacional	512	1×1	8×8	4
Convolutacional	1024	3×3	8×8	4
Residual	—	—	8×8	4

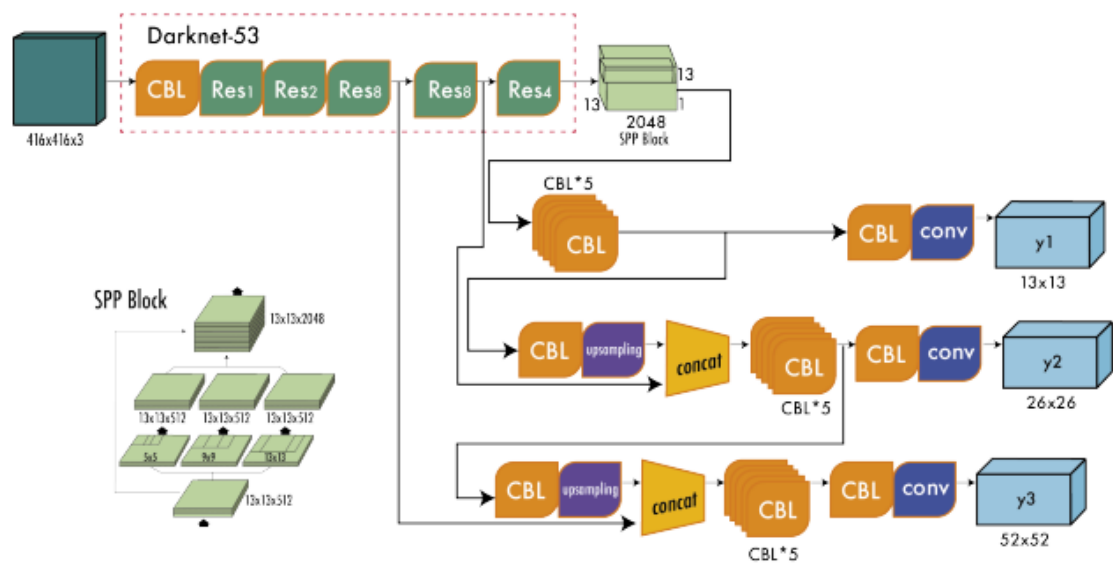


Figura 1.19. Arquitetura de detecção multi-escala do YOLO V3. A saída do backbone Darknet-53 é ramificada em três saídas diferentes marcadas como y1, y2 e y3, cada uma com resolução aumentada. Fonte: Terven e Cordova-Esparza (2023).

A Figura 1.20 mostra as principais características da terceira versão do modelo YOLO. YOLO V3

YOLOv3: melhoria incremental

Detecção multi-escala e backbone mais profundo

Darknet-53	Três escalas	Classificação
• Conexões residuais. • Maior profundidade.	• Pequenos, médios e grandes. • Mapas multi-resolução.	• Logística independente. • Classes sobrepostas.
Pipeline • Estágio único. • Mais precisão.		

Figura 1.20. Descrição das características da terceira versão do modelo YOLO.

1.2.5.4. YOLO V4: Velocidade e precisão ideais da detecção de objetos

A quarta versão da série YOLO foi criada por Alexey Bochkovskiy, Chien-Yao Wang e Hong-Yuan Mark Liao. A versão 4 é mais rápida e precisa do que todas as versões anteriores, tendo como objetivo produzir um detector de objetos para sistemas de produção. Os autores compararam CSPResNeXt50, CSPDarknet53 e EfficientNetB3 para a construção da arquitetura do YOLO V4. O CSPDarknet53, uma rede com 29 camadas convolucionais com filtros 3×3 e aproximadamente 27,6 milhões de parâmetros, foi escolhido como backbone por ter superado as demais arquiteturas. O "CSP" significa "Cross-stage partial connections" (conexões parciais entre estágios). As CSPNets ajudam na combinação rica de gradientes com custo computacional mínimo (BOCHKOVSKIY; WANG; LIAO, 2020).

Para aprimorar ainda mais o desempenho do sistema, podem ser implementadas algumas abordagens adicionais. O uso do Pooling Piramidal Espacial (SPP), também empregado pelo RCNN, permite ao modelo lidar eficientemente com imagens de tamanhos variados, sem a necessidade de redimensionamento ou alteração de formato, mapeando qualquer tamanho de entrada para uma saída de tamanho fixo. Além disso, a adoção da Rede de Agregação de Caminhos (PAN) em vez do Feature Pyramid Network (FPN) pode oferecer vantagens, já que a PAN utiliza o pooling adaptativo de recursos, funcionando de maneira semelhante ao FPN, porém com melhorias na propagação de recursos. Por fim, a aplicação de algoritmos genéticos para o ajuste de hiperparâmetros e o refinamento do modelo pode ser benéfica, pois oferecem uma abordagem eficaz para otimizar parâmetros complexos e explorar o espaço de busca de forma abrangente, potencialmente resultando em melhorias significativas no desempenho geral do sistema (DIWAN; ANIRUDH; TEMBHURNE, 2023).

O YOLO V4 buscou encontrar o equilíbrio ideal experimentando diversas mudanças categorizadas como "bag-of-freebies" e "bag-of-specials". Bag-of-freebies são métodos que alteram apenas a estratégia de treinamento e aumentam o custo computacional, mas não o tempo de inferência. O exemplo mais comum é a ampliação de dados (data augmentation). Por outro lado, bag-of-specials são métodos que aumentam ligeiramente o tempo de inferência, mas melhoram significativamente a precisão (TERVEN; CORDOVA-ESPARZA, 2023). Segue o detalhamento das técnicas implementadas:

Bag-of-Freebies:

Aumento de dados

- CutMix: Combina aleatoriamente patches de diferentes imagens para aumentar a diversidade do conjunto de dados (YUN et al., 2019).
- MixUp: Combina linearmente features de diferentes imagens para gerar exemplos interpolados (ZHANG, HONGYI et al., 2017).
- CutOut: Remove aleatoriamente patches de imagens para estimular a robustez do modelo a oclusões (DEVRIES; TAYLOR, 2017).

Funções de ativação

- Mish: Função de ativação suave e não monótona, definida por $f(x) = x \tanh(\text{softplus}(x))$, que preserva pequenos valores negativos e tem apresentado melhor desempenho que a ReLU em diversos cenários (MISRA, 2019).
- ReLU, LeakyReLU, SELU, ReLU-6: Funções de ativação com diferentes propriedades, exploradas para encontrar a mais adequada para cada camada da rede (KUNC; KLÉMA, 2024; LEDERER, 2021).

Regularização

- DropPath: Desativa aleatoriamente sub-redes durante o treinamento, evitando o sobreajuste (LARSSON; MAIRE; SHAKHNAROVICH, 2016).
- DropOut: Desativa aleatoriamente neurônios durante o treinamento, promovendo a generalização do modelo (CAI, SHAOFENG et al., 2019).
- DropBlock: Desativa aleatoriamente blocos de neurônios, similar ao DropOut, mas com maior granularidade (GHIASI; LIN; LE, 2018).
- Spatial DropOut: Desativa aleatoriamente patches de features, incentivando a robustez a oclusões (CAI, SHAOFENG et al., 2019).

Normalização

- Batchnorm: Padroniza as ativações das camadas, facilitando o treinamento e aumentando a estabilidade (IOFFE; SZEGEDY, 2015).
- Cross GPU Batchnorm: Sincroniza as estatísticas de Batchnorm entre diferentes GPUs, permitindo um treinamento mais eficiente em ambientes multi-GPU (YAO et al., 2020).
- BN: A BN é uma técnica usada para normalizar as ativações intermediárias em uma rede neural durante o treinamento. Ela normaliza as ativações de cada mini-batch, tornando o treinamento mais estável e acelerando a convergência (GAO; SUN, 2023).
- CBN: A CBN é uma variante da BN que leva em consideração informações contextuais adicionais. Ela ajusta os parâmetros de normalização com base em condições específicas, como classes de saída ou outras características. A CBN é útil em tarefas como transferência de estilo, onde diferentes estilos podem ser aplicados a uma imagem de entrada (SHETH et al., 2022).

Funções de perda

- MSE: Mede o erro quadrático médio entre os valores previstos e reais, sendo uma função de perda clássica (BISHOP, 2006).
- IoU: Mede a interseção sobre união entre as caixas delimitadoras preditas e reais, penalizando previsões imprecisas (REZATOFIGHI et al., 2019).
- GIoU, CIoU: Evoluções da IoU que incorporam informações geométricas adicionais para uma avaliação mais precisa da qualidade das previsões. A GIoU acrescenta a menor caixa envolvente comum (REZATOFIGHI et al., 2019) e a CIoU incorpora também a distância entre centros e a razão de aspecto (ZHENG et al., 2020).

Bag-of-Specials:

- CSPDarknet53: Estrutura "backbone". Arquitetura de rede neural convolucional robusta e eficiente, com 29 camadas convolucionais e aproximadamente 27,6 milhões de parâmetros (BOCHKOVSKIY; WANG; LIAO, 2020; MAHASIN; DEWI, 2022).
- Spatial Pyramid Pooling (SPP): Permite que o modelo processe imagens de tamanhos variados sem necessidade de redimensionamento, aumentando a flexibilidade. O SPP mapeia qualquer tamanho de entrada para uma saída de tamanho fixo, facilitando a detecção de objetos de diferentes tamanhos (HE et al., 2015b).
- Path Aggregation Network (PAN): Estrutura de "neck" que combina features de diferentes escalas de forma mais eficiente que o Feature Pyramid Network (FPN). O PAN utiliza pooling adaptativo de features para uma melhor propagação de informações e detecção de objetos em diferentes escalas (LIU, SHU et al., 2018).
- Treinamento Autoadversarial (SAT): O Self-Adversarial Training (SAT) empregado no YOLOv4 opera em duas etapas: na primeira, a rede altera de forma adversarial a própria imagem de entrada, sem modificar os pesos; na segunda, o detector é treinado sobre a imagem assim modificada (BOCHKOVSKIY; WANG; LIAO, 2020).
- Otimização de Hiperparâmetros com Algoritmos Genéticos: Uma técnica que utiliza algoritmos genéticos para encontrar os melhores valores para os hiperparâmetros da rede (por exemplo: Taxa de Aprendizagem, Tamanho do Lote (batch), Número de Épocas (epochs) e Especificidades da Arquitetura), dessa forma, ajuda a encontrar a melhor configuração para o modelo, otimizando seu desempenho (BOCHKOVSKIY; WANG; LIAO, 2020).

A partir de YOLO V4, começa-se a estabelecer firmemente a estrutura Backbone-Neck-Head. A evolução do YOLO ao longo de suas versões, é marcada pela introdução e refinamento dos componentes Backbone, Neck e Head. Inicialmente, o YOLOv1 operava com uma única rede convolucional para extrair features e gerar caixas delimitadoras, sem uma diferenciação explícita desses elementos. No entanto, o YOLOv2 introduziu a distinção entre Backbone e

Head, com a rede Darknet-19 atuando como Backbone para a extração de features e uma Head convolucional de detecção, que substituiu as camadas totalmente conectadas do YOLOv1. O YOLO V3 refinou ainda mais a arquitetura ao incluir o Neck, implementando o FPN (Feature Pyramid Network) para combinar features de diferentes escalas, resultando em uma detecção mais precisa de objetos de tamanhos variados.

Já o YOLO V4 manteve a estrutura Backbone-Neck-Head, porém com otimizações significativas, como a introdução do Backbone CSPDarknet53 e do PAN (Path Aggregation Network) no Neck, além de melhorias na Head com a utilização de funções de ativação otimizadas e métodos de regularização.

A partir de YOLOv5 para a frente, segue a estrutura Backbone-Neck-Head, que é uma fundamental na detecção moderna de objetos e em cada versão são implementados com módulos independentes, permitindo uma construção do modelo mais flexível, por exemplo, a versão base do YOLO V5 incorpora o Backbone Focus, o Neck PAN e a Head YOLOX, enfatizando modularidade e escalabilidade.

Seguem os conceitos básicos sobre a estrutura Backbone-Neck-Head:

Backbone

O Backbone é uma rede neural pré-treinada usada para extrair representações ricas de características das imagens. Ele reduz a resolução espacial da imagem e aumenta a resolução das características (canais). O Backbone é responsável por capturar informações de baixo nível, como bordas, texturas e padrões. Este componente, geralmente uma rede neural convolucional (CNN) profunda pré-treinada em um enorme conjunto de dados de classificação de imagem como o ImageNet, extrai recursos da imagem de entrada. Esses recursos variam de detalhes de baixo nível, como bordas e texturas (extraídos em camadas anteriores), a informações de alto nível, como partes e semântica do objeto (extraídas em camadas mais profundas). A escolha do Backbone ideal depende de fatores como o tipo de aplicação, os requisitos de desempenho e a disponibilidade de recursos computacionais (REDMON et al., 2016).

Neck

Sua função é extrair pirâmides de características. As pirâmides de características ajudam o modelo a generalizar bem para objetos de diferentes tamanhos e escalas. Agindo como uma camada intermediária, o neck refina e agrega os recursos extraídos pelo backbone. Ele frequentemente utiliza camadas convolucionais adicionais ou arquiteturas especializadas como Redes Piramidais de Recursos (FPNs) para melhorar a informação espacial e semântica em diferentes escalas. A escolha do Neck ideal depende das características do Backbone e das necessidades específicas da aplicação.

Head

O componente final, o head, aproveita os recursos fornecidos pelo neck para fazer previsões. Ele normalmente consiste em sub-redes especializadas responsáveis por tarefas como classificação, localização e até mesmo segmentação de instância e estimativa de pose em modelos mais recentes. Este estágio final gera previsões de candidatos a objetos, que são então filtradas por

meio de técnicas de pós-processamento como supressão não máxima para remover detecções redundantes e reter apenas as mais confiáveis. A escolha da Head ideal depende do tipo de detector de objetos utilizado e dos requisitos de precisão e velocidade.

A Figura 1.22 ilustra a estrutura completa do Backbone-Neck-Head para detecção de objetos em uma ou duas fases. As entradas incluem imagens, patches e uma pirâmide de imagens. Diversos backbones podem ser utilizados, como VGG16 , ResNet-50 , SpineNet , EfficientNet-B0/B7 , CSPResNeXt50 e CSPDarknet53 (WANG, CHIEN-YAO et al., 2019). O Neck, por sua vez, é composto por blocos adicionais como SPP , ASPP , RFB e SAM , além de blocos de agregação de caminho como FPN , PAN , NAS-FPN , Fully-connected FPN, BiFPN , ASFF e SFAM (ZHAO et al., 2018). Já as Heads podem ser modelos de predição densa de uma fase como RPN , SSD , YOLO , RetinaNet , CornerNet , CenterNet , MatrixNet e FCOS , ou modelos de predição esparsa de duas fases como Faster R-CNN , R-FCN , Mask R-CNN e RepPoints (YANG et al., 2019). É importante destacar que o YOLO V4 utiliza um detector de uma fase com CSPDarknet53 como backbone, SPP e PAN como Neck, e YOLOv3 como head (Figura 1.22) (BOCHKOVSKIY; WANG; LIAO, 2020).

A Figura 1.21 mostra as principais características da quarta versão do modelo YOLO.
YOLO V4

YOLOv4: velocidade e precisão

Combinação de backbone robusto, neck multi-escala e técnicas de treinamento

Backbone	Neck	BoF / BoS
- CSPDarknet53. - Mish e normalização.	- SPP + PAN. - Fusão multi-escala.	- Mosaic, CutMix, DropBlock. - Mais acurácia.
Produção - Detector rápido. - Boa relação custo-acurácia.		

Figura 1.21. Descrição das características da quarta versão do modelo YOLO.

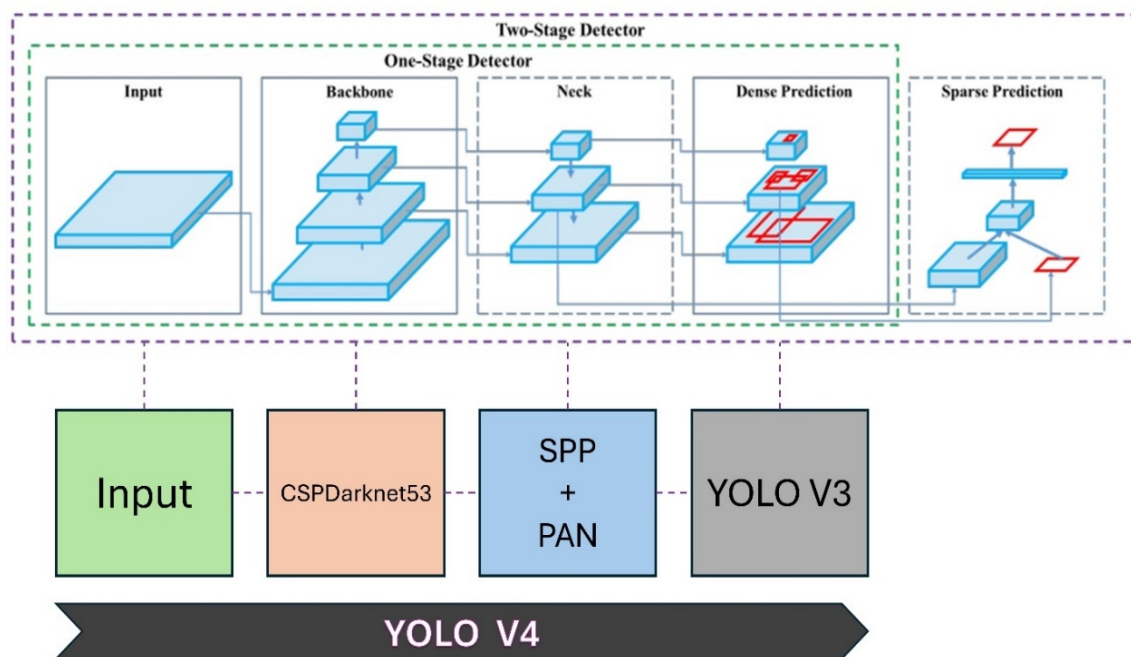


Figura 1.22. Diagrama do processo Backbone-Neck-Head de detecção em uma e duas fases do modelo YOLO V4. Nota: Imagem modificada baseada em Bochkovskiy et al. (2020).

1.2.5.5. YOLO V5

O YOLO V5 representa a quinta iteração do YOLO, um método amplamente utilizado para detecção de objetos em tempo real. Duas características distintas se destacam nesta versão: Autoaprendizado, em que o YOLO V5 emprega um algoritmo de aprendizado automático para aprimorar as caixas de ancoragem, permitindo que o modelo determine as melhores configurações para o conjunto de dados fornecido através de um algoritmo de agrupamento k-means. Além disso, as Caixas de Ancoragem desempenham um papel crucial no YOLO V5, sendo caixas delimitadoras pré-definidas de alturas e larguras específicas que servem como referências para prever as caixas delimitadoras reais dos objetos. O modelo prevê a largura e altura das caixas delimitadoras como deslocamentos das caixas de ancoragem, permitindo que a rede produza uma transformação adequada para ajustar as caixas de ancoragem aos objetos reais (JANI et al., 2023; JOCHER, 2020).

Embora mantenha a arquitetura básica das versões anteriores, como YOLO V3 e YOLO V4, cada nova versão traz consigo melhorias e inovações significativas. O modelo YOLO V5 resume-se da seguinte forma: sua espinha dorsal (backbone) utiliza a estrutura Focus e a rede CSP para extração de características, o pescoço (neck) combina blocos SPP e PANet para processamento multi-escala, e a cabeça (head) baseia-se na cabeça do YOLO V3 com perda GIoU. A principal melhoria do YOLO V5 sobre o YOLO V4 é a utilização da rede CSPDarknet53, que resolve o problema de informações repetitivas do gradiente encontradas no YOLO V4 e YOLO V3, reduzindo o número de parâmetros da rede e consequentemente a velocidade de inferência, ao mesmo tempo que aumenta a precisão (NEPAL; ESLAMIAT, 2022; OLORUNSHOLA;

IRHEBHUDE; EVWIEKPAEFE, 2023). A arquitetura do YOLO V5 é ilustrada na Figura 1.23.

Além disso, foi desenvolvido em PyTorch em vez de Darknet. A Darknet e o PyTorch são dois frameworks amplamente empregados para aprendizado profundo e visão computacional. Darknet, programado em C, é altamente otimizado para velocidade e frequentemente empregado em detecção de objetos. Por outro lado, o PyTorch, escrito em Python, é um framework versátil para aprendizado profundo, oferecendo suporte a diversas tarefas além da detecção de objetos. Beneficiando-se de uma comunidade extensa, integra-se de forma harmoniosa com outras bibliotecas e ferramentas Python.

YOLO V5 é de código aberto e é ativamente mantido pela Ultralytics. A Ultralytics fornece uma versão móvel para iOS e Android, além de numerosas integrações para rotulação, treinamento e implantação. YOLOv5 oferece cinco versões em escala: YOLO V5n (nano), YOLO V5s (pequena), YOLO V5m (média), YOLO V5l (grande) e YOLO V5x (extragrande).

As diferenças entre os modelos YOLO V5 oferecem opções variadas para atender às necessidades específicas de detecção de objetos, cada uma com características distintas. O YOLO V5n (Nano) é o menor modelo, adequado para dispositivos com recursos limitados, priorizando velocidade sobre precisão. Já o YOLO V5s (Pequena) equilibra velocidade e precisão, sendo uma boa escolha para aplicações em tempo real. O YOLO V5m (Média) oferece maior precisão que o YOLO V5s, mas com uma velocidade um pouco menor, sendo útil para aplicações que demandam mais precisão sem sacrificar a capacidade em tempo real. O YOLO V5l (Grande) destaca-se pela alta precisão, embora seja mais lento que o YOLO V5m, sendo indicado para aplicações que exigem a máxima precisão possível. Por fim, o YOLO V5x (Extra-grande) é o maior modelo, com a mais alta precisão, mas também a menor velocidade, sendo recomendado apenas para tarefas que demandam a máxima precisão e alto desempenho computacional. Além disso, cada modelo possui um número diferente de camadas, variando de 3 camadas para o YOLO V5n a 27 camadas para o YOLO V5x, refletindo em diferentes níveis de complexidade, sendo os modelos menores mais simples e rápidos, enquanto os maiores são mais complexos e mais lentos (HORVAT; JELEČEVIĆ; GLEDEC, 2022; JOCHER, 2020).

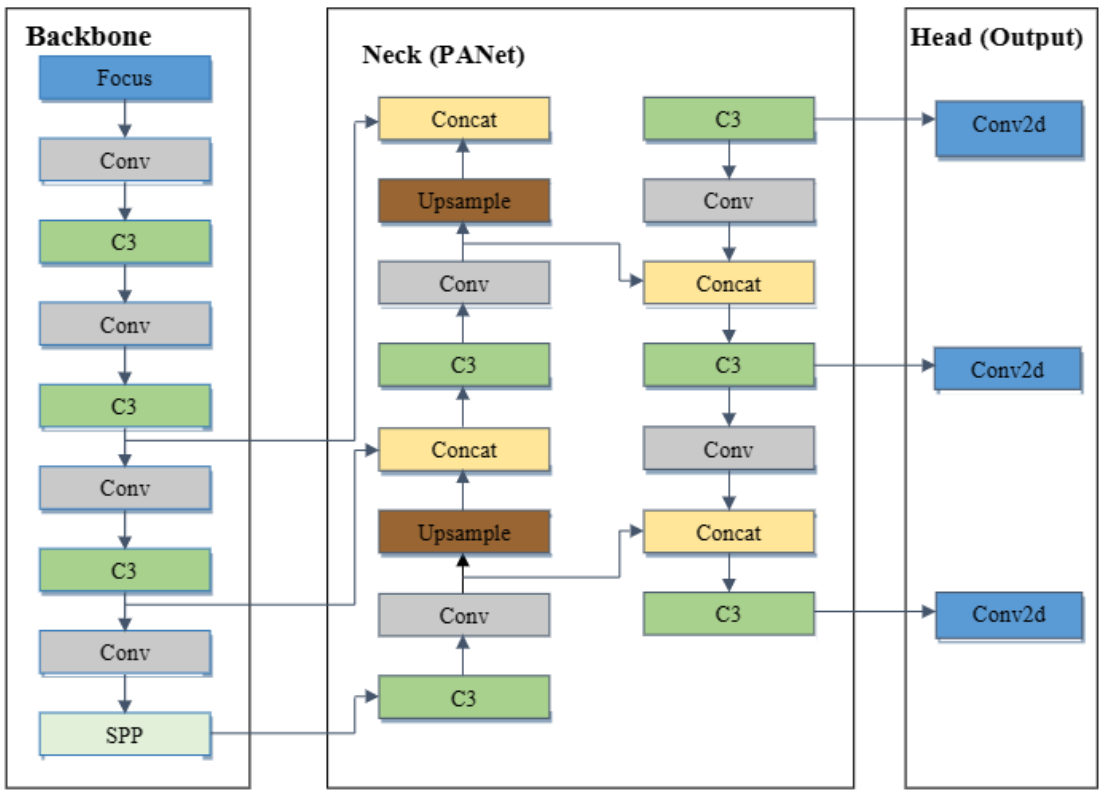


Figura 1.23. Desenvolvimento da arquitetura do modelo YOLO V5. Fonte: Nepal e Eslamiat (2022).

A Figura 1.24 mostra as principais características da quinta versão do modelo YOLO. YOLO V5

YOLOv5

Implementação prática em PyTorch orientada à experimentação e implantação

Ultralytics • Treino, validação e exportação. • Uso industrial.	Arquitetura • Backbone CSP. • Neck PAN/FPN.	Escalas • n, s, m, l, x. • Latência vs. precisão.
--	--	--

Aplicação

- Protótipos rápidos.
- Grande comunidade.

Figura 1.24. Descrição das características da quinta versão do modelo YOLO.

1.2.5.6. YOLO V6: Marco de detecção de objetos em uma única fase para aplicações industriais

Seguindo a tendência dos métodos baseados em pontos de ancoragem, o YOLOv6 adotou um detector livre de ancoragem. As principais novidades deste modelo foram (LI, CHUYI. et al., 2022; TERVEN; CÓRDOVA-ESPARZA; ROMERO-GONZÁLEZ, 2023):

- Uma nova espinha dorsal baseada no RepVGG chamada EfficientRep que utiliza maior paralelismo do que as espinhas dorsais YOLO anteriores.
- Para o pescoço, eles utilizaram PAN aprimorado com RepBlocks ou Blocos CSPStackRep para os modelos maiores.
- Inspirados pelo YOLOX, desenvolveram uma cabeça desacoplada eficiente. Atribuição de rótulo usando a abordagem de aprendizado de alinhamento de tarefas introduzida no TOOD.
- Novas perdas de classificação e regressão. Eles utilizaram uma perda de classificação VariFocal e uma perda de regressão SIOU/GIOU.
- Uma estratégia de auto-destilação para as tarefas de regressão e classificação.
- Um esquema de quantização para detecção usando RepOptimizer e destilação por canal que ajudou a alcançar um detector mais rápido.

A Figura 1.25 mostra as principais características da sexta versão do modelo YOLO. YOLO V6

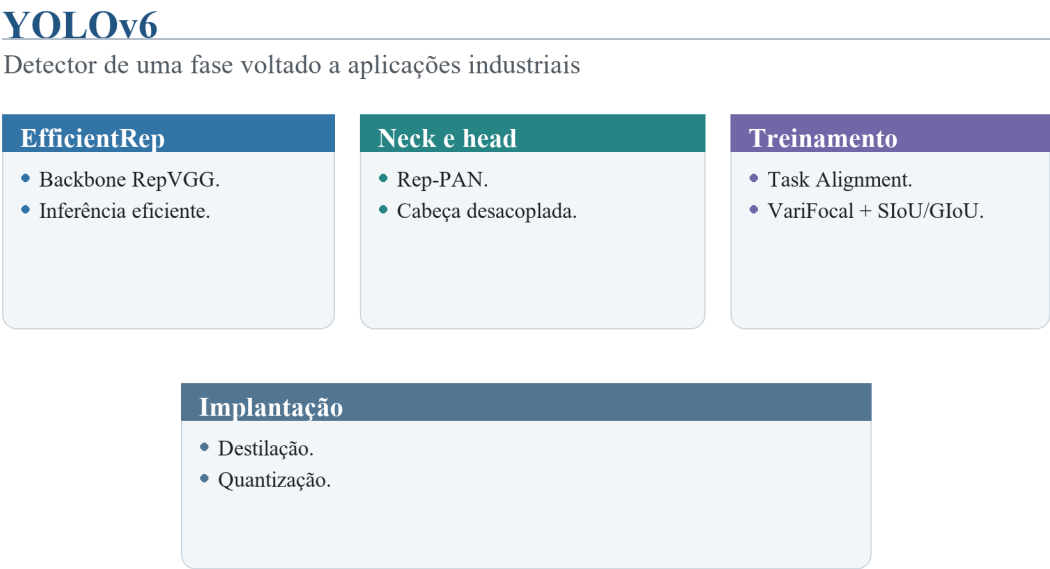


Figura 1.25. Descrição das características da sexta versão do modelo YOLO.

1.2.5.7. YOLO V7: Conjunto treinável de recursos define um novo padrão para detectores de objetos em tempo real

O YOLO V7 emerge como um farol revolucionário na detecção de objetos em tempo real, especialmente para aplicações industriais. Este modelo inovador se destaca por sua excepcional velocidade e precisão, superando as capacidades de seus predecessores. Seus avanços em arquitetura e treinamento, como as Redes de Agregação Eficiente de Camadas Estendidas (E-ELAN), a escalabilidade de modelos baseados em concatenação e "bag-of-freebies", o tornam uma ferramenta inestimável para uma ampla gama de aplicações (TERVEN; CÓRDOVA-ESPARZA; ROMERO-GONZÁLEZ, 2023).

O E-ELAN permite aprendizado e convergência mais eficientes, baseados na análise de trajetórias de gradientes, potencializando a capacidade expressiva do modelo treinado ao melhorar a capacidade de aprendizado da rede e combinando características de diferentes grupos (WANG, CHIEN-YAO; LIAO; YEH, 2022). Além disso, o escalonamento de modelos para arquiteturas baseadas em concatenação gera modelos de diferentes tamanhos, preservando a estrutura ideal (SHARMA; INDU; JAYANTHI, 2023).

A escalabilidade de modelos permite gerar modelos de diferentes tamanhos sem afetar a estrutura ideal. Isso significa que podem ser criados modelos menores e mais rápidos para dispositivos com recursos limitados, ou modelos maiores e mais precisos para tarefas mais exigentes. Os "Bag-of-freebies" é uma técnica que aumenta a precisão do modelo sem afetar a velocidade de inferência. Isso é logrado através da incorporação de diversas técnicas de treinamento, como a regularização e a otimização do hiperparâmetro (WANG, C. Y.; BOCHKOVSKIY; LIAO, 2023).

No aspecto do treinamento, o YOLO V7 incorpora técnicas "bag-of-freebies" como a Convolução Reparametrizada Planejada (RepConvN), que elimina a conexão de identidade na arquitetura original, resultando em melhor desempenho. Atribuição de Rótulo Grossa e Fina ou "Coarse and Fine Label Assignment" é implementada em duas etapas para aumentar a precisão e eficiência do treinamento. O rótulo grossa é gerado relaxando as restrições do processo de atribuição de amostras positivas, tratando mais grades como alvos positivos. A Normalização em Lote (NL) na Convolução-bn-Ativação otimiza o desempenho do modelo na fase de inferência. A NL é uma técnica amplamente utilizada no campo do Aprendizado Profundo. Ela melhora a velocidade de aprendizado das Redes Neurais e fornece regularização, evitando o sobreajuste. A NL é uma técnica de normalização realizada entre as camadas de uma Rede Neural em vez de nos dados brutos. É feita ao longo de mini-lotes em vez do conjunto de dados completo. No processo de treinamento de redes neurais convolucionais (CNN), uma camada de NL é frequentemente inserida após uma camada de convolução para acelerar a convergência do treinamento da CNN (OGUNDOKUN et al., 2022; PENG; YU; YU, 2024; TING, YU-SHENG; TENG; CHIUEH, 2021).

Além disso, o modelo incorpora conhecimento implícito inspirado no YOLOR, aumentando

a robustez e generalização. Por fim, a Média Móvel Exponencial or "Exponential Moving Average" (EMA) é utilizada como modelo de inferência final para fornecer resultados mais estáveis e precisos. EMA é um tipo de média móvel que atribui mais peso aos pontos de dados mais recentes do que aos que ocorreram no passado. Isso significa que ela dá mais importância às últimas experiências ou memórias, assumindo que estas são representadas pelos pontos de dados. No contexto de redes neurais e aprendizado profundo, EMA ajuda a estabilizar o processo de treinamento ao minimizar o ruído nas atualizações de parâmetros, o que leva a uma convergência mais estável e melhor generalização do modelo. EMA suaviza as flutuações e o ruído no processo de treinamento ao calcular a média dos pesos ao longo do tempo. Isso pode resultar em um modelo mais estável e robusto (CAI, ZHAOWEI et al., 2021).

A Figura 1.26 mostra as principais características da sétima versão do modelo YOLO. YOLO V7

YOLOv7

Recursos treináveis para detectores em tempo real

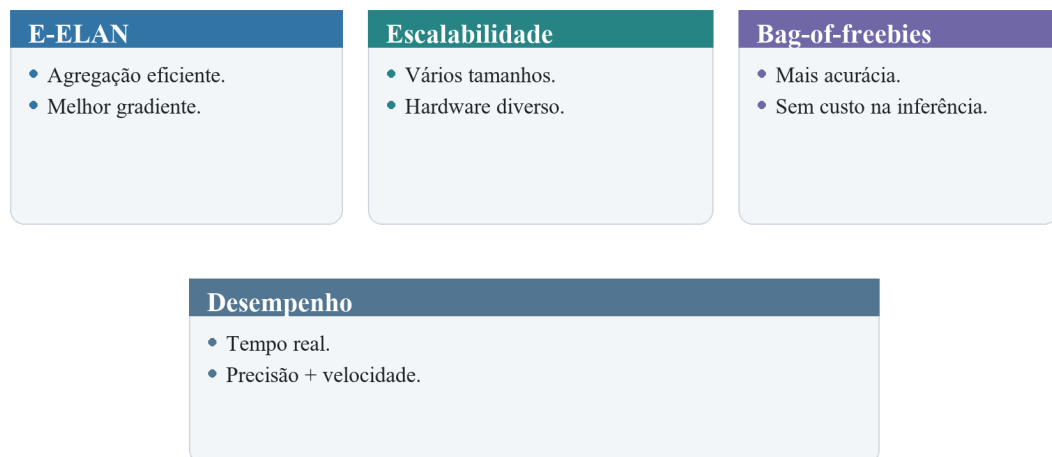


Figura 1.26. Descrição das características da sétima versão do modelo YOLO.

1.2.5.8. YOLO V8

O YOLO V8, lançado em janeiro de 2023 pela Ultralytics, a empresa criadora do YOLOv5, representa mais um avanço significativo na tecnologia de detecção de objetos em tempo real. Ainda não há um artigo científico oficial sobre o YOLO V8, mas vamos explorar suas inovações arquitetônicas em relação às versões anteriores.

O YOLO V8 apresenta diversas características distintas:

- Elimina a necessidade de âncoras pré-definidas, reduzindo o número de previsões de caixas delimitadoras e acelerando o processo de supressão de não-máximos (NMS).

- Durante o treinamento, utiliza a técnica de aumento por mosaico, combinando quatro imagens diferentes em uma única entrada, embora desabilite esse aumento nas últimas dez épocas de treinamento devido a possíveis efeitos prejudiciais.
- É versátil em sua utilização, podendo ser executado a partir da linha de comando (CLI) ou instalado como um pacote PIP, oferecendo integrações para rotulagem, treinamento e implementação de modelos.
- Disponibiliza cinco versões com diferentes tamanhos e pesos (YOLO V8n, YOLO V8s, YOLO V8m, YOLO V8l e YOLO V8x), permitindo seleções ideais para diversas necessidades de precisão e velocidade.

A Figura 1.27 mostra as principais características da oitava versão do modelo YOLO. YOLO V8

YOLOv8

Arquitetura moderna sem âncoras e ecossistema Ultralytics

Anchor-free	C2f e SPPF	Tarefas
• Sem âncoras fixas. • Predição simplificada.	• Blocos eficientes. • Custo menor.	• Detecção e segmentação. • CLI ou Python.

Modelos
• n, s, m, l, x. • Precisão vs. memória.

Figura 1.27. Descrição das características da oitava versão do modelo YOLO.

1.2.5.8.1. Blocos individuais na arquitetura YOLOV8 Para explicar a arquitetura YOLOV8 detalhadamente, veremos os blocos individuais que a compõem para facilitar o entendimento:

Bloco Convolutacional

É o bloco fundamental, composto por uma camada convolutacional 2D, normalização Batch 2D e ativação SiLU. Esses elementos estão unidos em um único bloco.

Bloco C2F

Possui um bloco convolucional inicial. O mapa de características resultante é dividido, com uma parte indo para o bloco Bottleneck e a outra diretamente para o bloco Concat. O bloco C2F pode ter vários blocos Bottleneck. Finaliza com outro bloco convolucional.

Bloco Bottleneck

É uma sequência de blocos convolucionais com um atalho.

Bloco SPPF (Spatial Pyramid Pooling Fast)

É uma modificação do SPP (Spatial Pyramid Pooling) com maior velocidade. Possui um bloco convolucional inicial seguido de três camadas de Max Pooling 2D. O interessante é que cada mapa de características resultante é concatenado antes do final do SPPF.

Bloco de Detecção

Diferente das versões anteriores do YOLO, o YOLOv8 é um modelo sem âncoras. As previsões acontecem na célula da grade. O bloco de detecção possui duas ramificações: uma para previsão da caixa delimitadora e outra para previsão da classe. Ambas as ramificações possuem a mesma sequência de blocos: 2 blocos convolucionais e uma única camada convolucional 2D.

1.2.5.8.2. Componentes fundamentais de uma rede neural convolucional Kernel

É uma matriz bidimensional. O kernel também é chamado de detector de características. Os valores no kernel são pesos que podem ser atualizados durante o treinamento. O kernel se move pela imagem e realiza uma operação de ponto entre a entrada e o valor do kernel para produzir uma saída, também conhecida como mapa de características.

Stride

Define a distância de deslocamento durante o processo de convolução. Quanto menor a saída resultante, maior o stride.

Padding

É a adição de valor aos elementos da borda da imagem. No Python, existem vários tipos de padding. O padding "same" preenche os pixels com valor 0, enquanto o padding "replication" replica o valor do pixel real mais próximo. Os cantos preenchidos terão o mesmo valor dos cantos reais.

1.2.5.8.3. Arquitetura unificada Em geral, a arquitetura YOLO é dividida em 3 partes: backbone, neck e head.

Backbone

É a arquitetura de aprendizado profundo que funciona como um extrator de características.

Neck

Combina as características adquiridas das várias camadas do modelo backbone.

Head

Prediz as classes e regiões da caixa delimitadora, que é a saída final produzida pelo modelo de detecção de objetos.

1.2.5.8.4. Explicação detalhada da arquitetura YOLOV8 Esta explicação baseia-se no arquivo de arquitetura YOLOV8 pela Ultralytics. Também se inspira fortemente nas análises do Dr. Priyanto Hidayatullah, especialista em visão computacional e aprendizado profundo (<https://www.linkedin.com/in/dr-priyanto-hidayatullah-38632715/>).

A explicação começa com três parâmetros que definem a variante YOLOv8: depth multiple (d), width multiple (w) e max channels (mc) (Tabela 1.5). O depth multiple determina quantos blocos Bottleneck estão no bloco C2F. Os parâmetros width multiple e max channels determinam o canal de saída. A entrada do YOLOv8 é uma imagem com três canais.

Tabela 1.5. Parâmetros que definem as diferentes variantes do modelo YOLOV8.

Variante do modelo de YOLO	Depth multiple (d)	width multiple (w)	Max channels (mc)
n	0,33	0,25	1024
s	0,33	0,50	1024
m	0,67	0,75	768
l	1	1	512
x	1	1,25	512

Cada bloco da arquitetura YOLOV8 é explicado em detalhes abaixo e pode ser visto na Figura 1.28.

- Backbone

O nome do backbone não é declarado diretamente. Ele é composto por várias camadas convolucionais que extraem diferentes características em vários níveis de resolução.

A numeração na arquitetura YOLOV8 baseia-se no arquivo YOLOV8.yml. A numeração começa na seção backbone a partir de zero. Por exemplo, o primeiro bloco convolucional é #0.

O backbone começa com dois blocos convolucionais com tamanho de kernel 3, stride 2 e padding 1. O stride 2 reduz a resolução espacial da saída.

- Bloco C2F

Este bloco possui três parâmetros: shortcut, n e d. O shortcut indica se o atalho será usado no bloco Bottleneck. O n determina quantos blocos Bottleneck serão usados, e o d é o valor depth multiple (d).

- Neck

O neck combina as características de diferentes resoluções para fornecer uma entrada ideal para o bloco de detecção. Ele usa uma camada upsample para aumentar a resolução do mapa de características do SPPF para corresponder à resolução do mapa de características do bloco C2F anterior. Em seguida, usa concat para combinar os mapas de características.

Dois blocos C2F adicionais são usados no neck, com diferentes parâmetros para reduzir o tamanho do canal do mapa de características.

- Bloco de Detecção

Existem três blocos de detecção especializados em detectar objetos de diferentes tamanhos (pequeno, médio e grande). Cada um recebe como entrada o mapa de características de um bloco C2F específico e usa blocos convolucionais para reduzir a resolução do mapa de características. Por fim, concat é usado para combinar o mapa de características resultante com o mapa de características de outro bloco C2F ou SPPF.

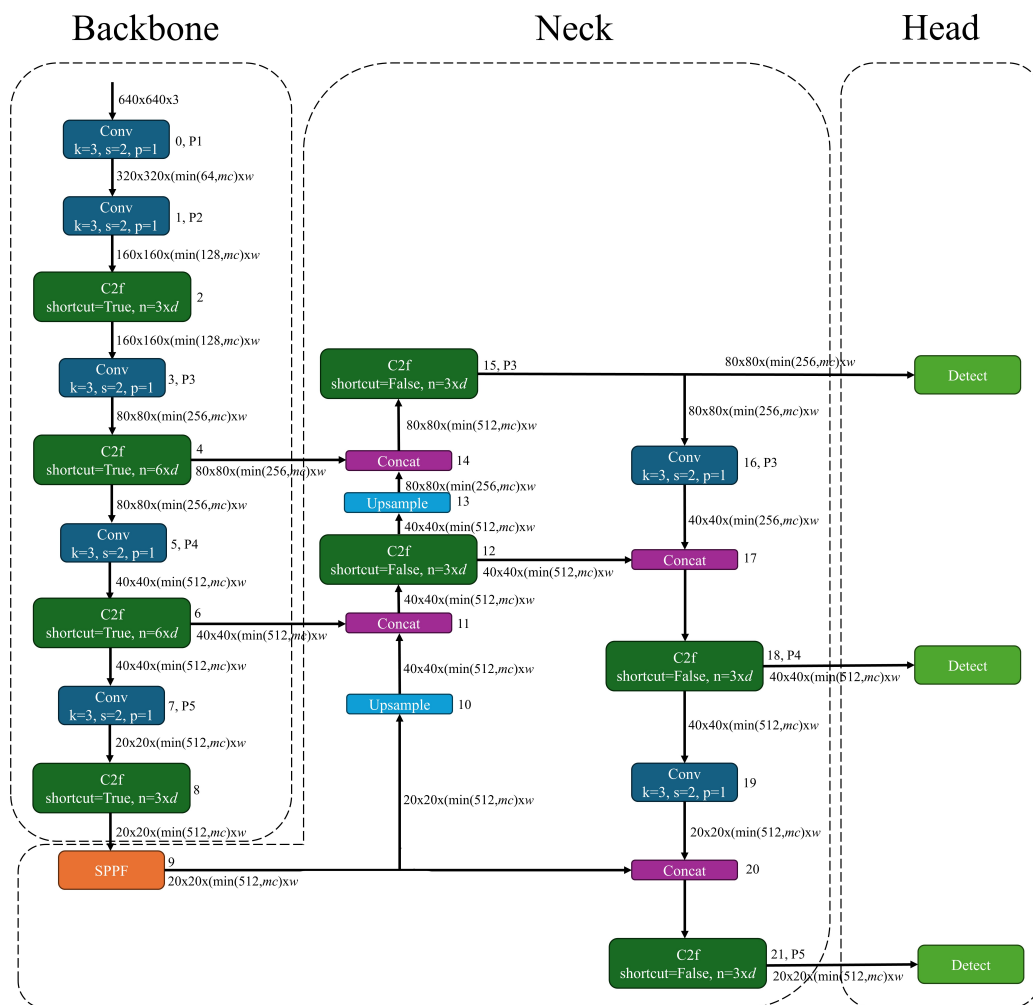


Figura 1.28. Arquitetura detalhada do modelo YOLOV8. Nota: Imagem modificada com base na análise da Ultralytics e do Dr. Priyanto Hidayatullah.

1.2.5.9. YOLO V9: Aprendendo o que você quer aprender usando informações de gradiente programável

YOLO V9 é a mais recente versão do YOLO, lançada em fevereiro de 2024, por Chien-Yao Wang, I-Hau Yeh e Hong-Yuan Mark Liao. É um modelo de detecção de objetos em tempo real

aprimorado que visa superar todos os métodos baseados em convolução e baseados em transformadores. O YOLO V9 é lançado em quatro modelos, ordenados pelo número de parâmetros: v9-S, v9-M, v9-C e v9-E. Para melhorar a precisão, ele introduz informações de gradiente programáveis (PGI) e a Rede de Agregação de Camadas Eficientes Generalizadas (GELAN). O PGI evita a perda de dados e garante atualizações de gradiente precisas e o GELAN otimiza modelos leves com planejamento de caminho de gradiente. Além disso, YOLO V9 oferece melhorias significativas em termos de precisão de detecção de objetos, utilizando PGI para evitar a perda de dados durante o processamento e introduzindo a GELAN, que otimiza modelos leves. Essas contribuições de pesquisa promovem avanços na arquitetura de redes neurais profundas para detecção de objetos, destacando-se análises teóricas inovadoras e o desenvolvimento de mecanismos como o PGI e a GELAN. A combinação dessas técnicas resulta em um modelo altamente competitivo, superando os detectores existentes em precisão e eficiência (WANG, CHIEN-YAO; YEH; LIAO, 2024). A Figura 1.29 mostra as principais características da nona versão do modelo YOLO.

YOLO V9

YOLOv9

Aprendizado com informação de gradiente programável

PGI	GELAN	Modelos
• Gradiente programável. • Menos perda de informação.	• Agregação eficiente. • Modelos leves.	• v9-S, M, C, E. • Escala por parâmetros.
Objetivo • Mais acurácia. • Tempo real.		

Figura 1.29. Descrição das características da nona versão do modelo YOLO.

1.2.5.10. YOLO V10: detecção end-to-end sem NMS

O YOLOv10 foi proposto por Wang et al. (2024) como uma resposta direta a uma limitação recorrente da família YOLO: a dependência da Supressão de Não Máximos (NMS) como etapa externa de pós-processamento. Embora o NMS tenha sido historicamente útil para remover caixas redundantes, ele adiciona latência, introduz hiperparâmetros sensíveis e dificulta a implantação completamente end-to-end. A contribuição central do YOLOv10 consiste em

combinar treinamento NMS-free com uma revisão arquitetural orientada ao compromisso entre eficiência e acurácia.

Na arquitetura, o YOLOv10 preserva a lógica geral backbone-neck-head, mas introduz atribuições duplas consistentes durante o treinamento. A estratégia combina uma ramificação one-to-many, que oferece supervisão abundante e favorece o aprendizado, com uma ramificação one-to-one, que aproxima o comportamento do modelo ao cenário de inferência sem NMS. Além disso, o desenho arquitetural incorpora cabeça de classificação mais leve, downsampling desacoplado espacial-canal, blocos guiados por rank para reduzir redundância computacional, convoluções de kernel grande e módulos parciais de autoatenção. A família é escalada em versões YOLOv10-N, S, M, B, L e X, permitindo ajustar o modelo conforme restrições de latência, memória e precisão.

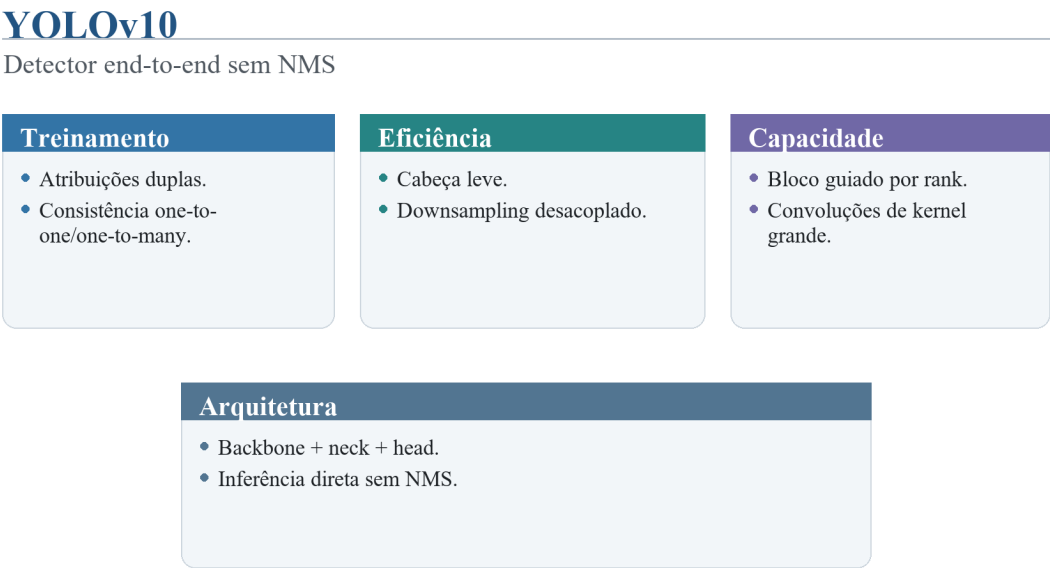


Figura 1.30. Síntese da arquitetura e das inovações do YOLO V10. Fonte: Elaborada com base em Wang et al. (2024).

1.2.5.11. YOLO V11: blocos C3k2, SPPF e C2PSA

O YOLOv11 representa uma etapa de consolidação da família Ultralytics, com foco em eficiência, extração de características e suporte a múltiplas tarefas de visão computacional. Segundo a análise arquitetural de Khanam e Hussain (2024), o modelo amplia a aplicabilidade da família YOLO para detecção, segmentação de instâncias, estimativa de pose e detecção orientada de objetos, mantendo o paradigma de detector de uma passada e a estrutura geral backbone-neck-head.

Sua arquitetura é caracterizada pela introdução do bloco C3k2, uma variação eficiente de blocos Cross Stage Partial com kernels menores, desenhada para melhorar a extração de carac-

terísticas com menor custo. O modelo também preserva o uso do SPPF (Spatial Pyramid Pooling - Fast), que amplia o campo receptivo com baixo custo computacional, e incorpora o C2PSA, bloco com atenção espacial paralela que reforça regiões relevantes do mapa de características. Desse modo, o YOLOv11 mantém a estrutura escalável em tamanhos nano, small, medium, large e extra-large, favorecendo aplicações desde dispositivos de borda até ambientes de maior capacidade computacional.

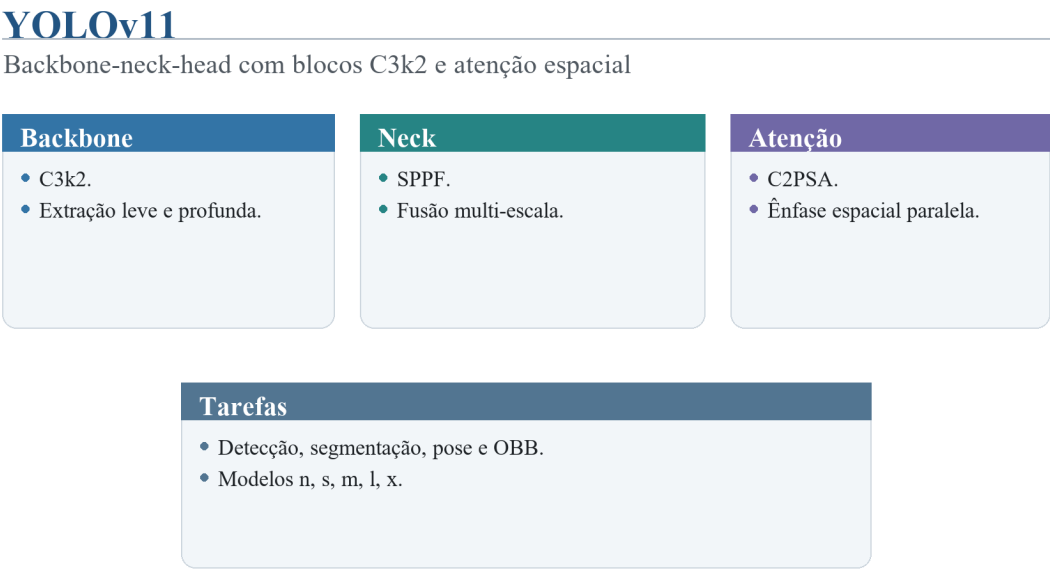


Figura 1.31. Síntese da arquitetura e dos blocos principais do YOLO V11. Fonte: Elaborada com base em Khanam e Hussain (2024).

1.2.5.12. YOLO V12: detectores em tempo real centrados em atenção

O YOLOv12, proposto por Tian, Ye e Doermann (2025), desloca o foco arquitetural da família YOLO de melhorias predominantemente convolucionais para um desenho centrado em mecanismos de atenção. A motivação do trabalho é que a atenção possui maior capacidade de modelagem contextual, mas, em arquiteturas tradicionais, tende a ser custosa para detecção em tempo real. O YOLOv12 busca preservar a velocidade típica dos detectores YOLO ao mesmo tempo em que incorpora os benefícios representacionais da atenção.

A arquitetura introduz o módulo Area Attention (A2), que reduz o custo computacional da atenção ao operar por áreas, mantendo campo receptivo amplo. Também utiliza R-ELAN (Residual Efficient Layer Aggregation Network), uma adaptação residual da lógica ELAN para enfrentar dificuldades de otimização em modelos com atenção. Outras decisões arquiteturais incluem o uso de FlashAttention para reduzir o custo de acesso à memória, remoção de codificação posicional para simplificar o fluxo, ajuste da razão MLP e uso seletivo de convoluções para preservar eficiência. Com isso, o YOLOv12 procura combinar a capacidade de modelagem

dos mecanismos de atenção com a baixa latência exigida em aplicações reais.

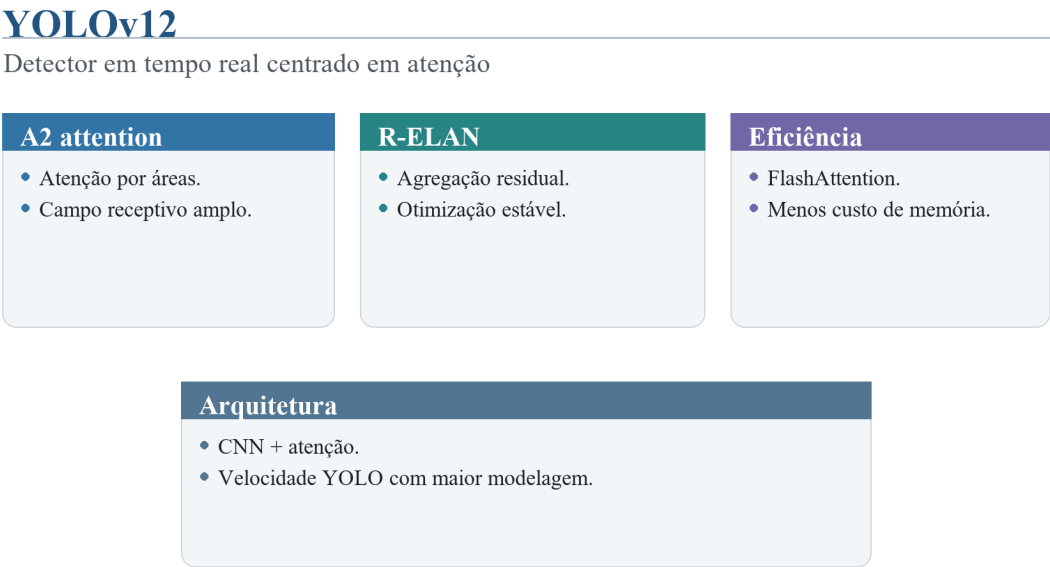


Figura 1.32. Síntese da arquitetura centrada em atenção do YOLO V12. Fonte: Elaborada com base em Tian, Ye e Doermann (2025).

1.2.5.13. YOLO26: modelos unificados end-to-end para visão em tempo real

O YOLO26, descrito por Jocher et al. (2026), é apresentado como uma família unificada de modelos de visão em tempo real, com ênfase em implantação simples, eficiência e suporte multitarefa. Diferente de versões que dependem de pós-processamento separado, o YOLO26 adota inferência end-to-end nativa sem NMS por padrão. A proposta também remove a Distribution Focal Loss (DFL), reduzindo a complexidade da cabeça de detecção e simplificando o caminho de exportação do modelo. Segundo a documentação oficial da Ultralytics e o preprint técnico de Jocher et al. (2026), a família YOLO26 opera em cinco escalas (n, s, m, l e x), alcançando, nas configurações reportadas para COCO, uma faixa de 40,9 a 57,5 mAP com latência TensorRT em GPU T4 entre 1,7 e 11,8 ms, o que reforça seu posicionamento como detector voltado a aplicações em tempo real e dispositivos de borda.

Do ponto de vista arquitetural, o YOLO26 preserva a lógica geral backbone-neck-head que caracteriza os detectores YOLO modernos, mas reorganiza a etapa final de predição. A imagem de entrada é processada por um backbone convolucional, responsável por extrair mapas de características em diferentes resoluções. Em seguida, o neck realiza a fusão multi-escala dessas características, combinando informações semânticas profundas com detalhes espaciais mais finos. Essa representação alimenta duas ramificações de detecção: uma cabeça *one-to-one*, usada por padrão na inferência end-to-end, e uma cabeça *one-to-many*, preservada para treinamento denso e para cenários em que a inferência tradicional com NMS seja preferível.

A cabeça *one-to-one* é o elemento central da proposta NMS-free. Ela é treinada para produzir uma predição principal por objeto verdadeiro, aproximando a saída do detector do comportamento desejado na inferência. Na implementação descrita por Jocher et al. (2026), essa ramificação retorna um conjunto fixo e compacto de detecções, com até 300 predições por imagem, eliminando a necessidade de uma etapa externa de supressão de caixas redundantes. A cabeça *one-to-many*, por outro lado, mantém a lógica YOLO densa, na qual múltiplas posições do mapa de características podem receber supervisão positiva para o mesmo objeto. Essa ramificação oferece sinal de treinamento mais abundante nas fases iniciais, favorecendo a aprendizagem do backbone e do neck, embora sua inferência dependa de NMS quando usada isoladamente.

Outra mudança importante é a remoção da DFL. Nas versões recentes da família YOLO, a DFL representa cada lado da caixa delimitadora como uma distribuição discreta de bins, o que melhora a localização, mas aumenta o número de parâmetros e operações na cabeça de regressão. No YOLO26, a regressão volta a ser direta, com quatro valores contínuos para a caixa, reduzindo custo computacional e evitando a limitação de alcance imposta pela discretização. Essa decisão favorece exportação para ONNX, TensorRT, CoreML, OpenVINO e outros ambientes de produção, especialmente em modelos pequenos, nos quais a cabeça de detecção pode representar parcela relevante do custo total.

O treinamento compensa a simplicidade da cabeça com três componentes coordenados. O MuSGD combina atualizações inspiradas no otimizador Muon com SGD com momento, aplicando tratamento diferente a pesos multidimensionais e parâmetros unidimensionais, como vieses e escalas de normalização. A Progressive Loss muda gradualmente o peso das perdas: no início, a cabeça *one-to-many* recebe maior ênfase por fornecer supervisão mais rica; ao longo do treinamento, o foco é deslocado para a cabeça *one-to-one*, que será usada no caminho end-to-end. O STAL (Small-Target-Aware Label Assignment) modifica a atribuição de rótulos para evitar que objetos muito pequenos fiquem sem candidatos positivos após o downsampling dos mapas de características. Essa combinação é destacada também por Chakrabarty (2026) e Sapkota et al. (2026) como um dos pontos que diferencia o YOLO26 de versões anteriores baseadas apenas em NMS e atribuição densa tradicional.

Além da detecção, o YOLO26 incorpora cabeças e perdas específicas para segmentação de instâncias, pose, classificação e detecção orientada por caixas rotacionadas (OBB). A extensão YOLOE-26 amplia a família para cenários de vocabulário aberto, permitindo inferência por texto, por referência visual ou em modo livre de prompts. Assim, o YOLO26 deve ser entendido menos como uma única arquitetura isolada e mais como uma base compartilhada para múltiplas tarefas de visão computacional, na qual o detector end-to-end sem NMS constitui o núcleo comum e as cabeças especializadas adaptam a saída ao tipo de problema.

YOLO26: arquitetura end-to-end

Backbone-neck compartilhados, cabeça dual sem NMS e treinamento MuSGD/ProgLoss/STAL

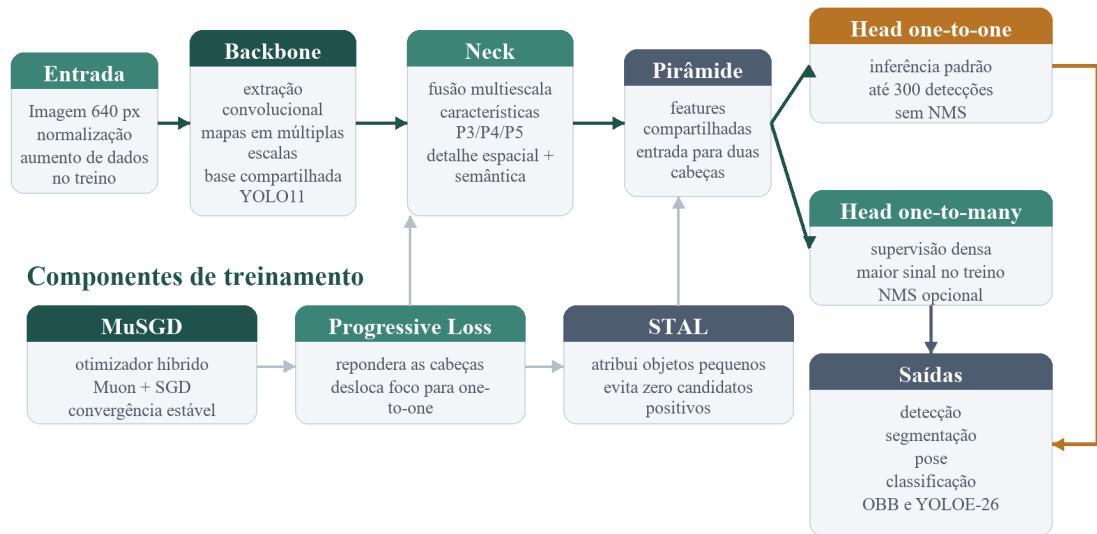


Figura 1.33. Síntese da arquitetura e do pipeline de treinamento do YOLO26. Fonte: Elaborada com base em Jocher et al. (2026).

1.2.6. Versões complementares do modelo YOLO

1.2.6.1. Scaled-YOLOV4

O Scaled-YOLOV4 foi desenvolvido por uma equipe de pesquisadores do Instituto de Ciência da Informação da Academia Sinica, em Taiwan. Uma nova abordagem para dimensionar redes neurais de detecção de objetos, com foco específico nos modelos YOLOv4. Apresenta o conceito de pesquisa de dimensionamento de modelo composto, que envolve o ajuste de seis parâmetros principais para otimizar o desempenho do modelo. Esses parâmetros incluem profundidade, largura inicial, inclinação, quantização, taxa de gargalo e largura do grupo (WANG, CHIEN-YAO; BOCHKOVSKIY; LIAO, 2020):

- **Profundidade:** Refere-se ao número de camadas na rede neural. Aumentar a profundidade geralmente melhora a precisão, mas também aumenta o tempo de processamento.
- **Largura inicial:** Determina o número de canais na primeira camada da rede. Aumentá-la também melhora a precisão, mas com maior custo computacional.
- **Inclinação:** Define a taxa de crescimento dos canais em camadas posteriores.
- **Quantização:** Refere-se ao processo de reduzir a precisão dos valores numéricos utilizados na rede, o que pode economizar memória e acelerar o processamento.
- **Taxa de gargalo:** Define a proporção entre os canais de entrada e saída de um bloco específico na rede.

- **Largura do grupo:** Determina o número de grupos de canais utilizados em uma camada convolucional.

O Scaled-YOLOV4 foi desenvolvido em PyTorch ao invés de Darknet. A principal novidade foi a introdução de técnicas para aumentar e reduzir a escala do modelo. Aumentar a escala significa criar um modelo que ganha em precisão, mas perde em velocidade. Por outro lado, reduzir a escala significa criar um modelo mais rápido, porém menos preciso. Além disso, modelos reduzidos exigem menos poder computacional e podem rodar em sistemas embarcados. A arquitetura reduzida foi chamada de YOLOV4-tiny, projetada para GPUs básicas e a arquitetura ampliada foi chamada de YOLOV4-large, que incluía três tamanhos diferentes: P5, P6 e P7. Essa arquitetura foi projetada para GPUs em nuvem e alcançou desempenho de ponta, superando todos os modelos anteriores (TERVEN; CÓRDOVA-ESPARZA; ROMERO-GONZÁLEZ, 2023).

1.2.6.2. YOLOR

O YOLOR (You Only Learn One Representation) foi desenvolvido por uma equipe de pesquisadores do Instituto de Ciência da Informação da Academia Sinica, em Taiwan, conduzido por Chien-Yao Wang, I-Hau Yeh e Hong-Yuan Mark Liao, propõe o desenvolvimento de uma rede neural unificada capaz de processar tanto o conhecimento explícito quanto o implícito, similar ao funcionamento do cérebro humano. O conhecimento explícito é aquele adquirido através do aprendizado tradicional, como estudar ou ler. Já o conhecimento implícito é adquirido inconscientemente, por meio de experiências e vivências. Ao integrar o conhecimento implícito na rede neural, os pesquisadores buscam aprimorar o desempenho em diversas tarefas. Isso se dá através da criação de uma representação unificada capaz de servir a múltiplos propósitos simultaneamente. A inspiração para essa abordagem vem da vasta base de dados de experiências humanas. Assim como nós utilizamos nossas experiências passadas para resolver novos problemas, a rede neural poderia se beneficiar de um conhecimento similar incorporado em sua estrutura. Os pesquisadores observaram que introduzir o conhecimento implícito na rede neural resultou em melhorias em tarefas como alinhamento no espaço do kernel, refinamento de previsões e aprendizado multitarefa (WANG, CHIEN-YAO; YEH; LIAO, 2021).

O YOLOR ao invés de treinar modelos separados para cada tarefa (como classificação, detecção de objetos, estimativa de pose), ele utiliza um aprendizado multitarefa. Isso significa que treina um único modelo para todas as tarefas, criando uma representação geral e usando sub-redes para adaptá-la a cada tarefa específica. O YOLOR parte da ideia de que o método tradicional de aprendizado conjunto nem sempre gera as melhores características. Por isso, ele busca codificar o conhecimento implícito da rede neural para aplicá-lo a múltiplas tarefas, similar a como humanos usam suas experiências passadas para resolver novos problemas (TERVEN; CÓRDOVA-ESPARZA; ROMERO-GONZÁLEZ, 2023).

1.2.6.3. YOLOX

Desenvolvido pela Megvii Technology, empresa líder em inteligência artificial conhecida por seu trabalho em visão computacional, aprendizado profundo e reconhecimento facial, o detector de objetos YOLOX representa um avanço significativo na série YOLO. Uma das principais inovações do YOLOX é a adoção de uma abordagem de detecção sem âncoras, rompendo com os detectores baseados em âncoras e regras de atribuição manuais utilizados anteriormente. Ao incorporar técnicas avançadas como cabeças desacopladas, detecção sem âncoras e uma estratégia avançada de atribuição de rótulos, o YOLOX alcança um melhor equilíbrio entre velocidade e precisão em comparação com outros detectores de vários tamanhos, como o YOLOV3 (GE et al., 2021).

O YOLOX se destaca por cinco mudanças principais em relação ao YOLOV3:

Arquitetura sem âncoras: YOLOX abandona o sistema de âncoras usado nas versões anteriores do YOLO, inspirando-se em detectores de objetos sem âncoras como CornerNet, CenterNet e FCOS. Isso simplifica o treinamento e a decodificação, aumentando a precisão média (AP) em 0.9 pontos.

Múltiplos positivos: Para compensar a falta de âncoras, o YOLOX usa amostragem central, onde a área central 3x3 é considerada positiva. Essa abordagem aumentou a AP em 2.1 pontos.

Cabeça desacoplada: Inspirado em pesquisas anteriores, o YOLOX separa as tarefas de classificação e localização em duas cabeças distintas. Isso melhora a AP em 1.1 pontos e acelera a convergência do modelo.

Atribuição avançada de rótulos: YOLOX se inspira em um trabalho que trata a atribuição de rótulos como um problema de Transporte Ótimo (OT) para lidar com sobreposição de objetos. Essa abordagem simplificada, chamada simOTA, aumentou a AP em 2.3 pontos.

Aumentações fortes: YOLOX utiliza técnicas de aumento de dados como MixUP e Mosaic, dispensando o pré-treinamento no ImageNet. Essas técnicas aumentaram a AP em 2.4 pontos.

1.2.6.4. DAMO-YOLO

Publicado pelo Alibaba Group, o DAMO-YOLO se destaca como um método de detecção de objetos que supera a série YOLO em termos de desempenho e eficiência. Para atender às diversas necessidades, o DAMO-YOLO oferece modelos em diferentes escalas: T/S/M/L: Indicados para aplicações industriais gerais, com equilíbrio entre velocidade e precisão e Ns/Nm/Nl: Modelos leves e eficientes para dispositivos com recursos computacionais limitados, como dispositivos móveis. Sua alta performance e baixa latência são resultado da integração de tecnologias inovadoras como (XU, XIANZHE et al., 2022):

Pesquisa de Arquitetura Neural (NAS): Otimiza a estrutura da rede neural para detecção precisa e eficiente.

FPN Generalizado Reparametrizado (RepGFPN): Aprimora a extração de características, acelerando o processamento.

Aprimoramento de Destilação: Transfere conhecimento de um modelo maior e pré-treinado, elevando ainda mais o desempenho.

1.2.6.5. PP-YOLO

PP-YOLO publicado por pesquisadores da Baidu Inc., baseia-se no YOLOV3, assim como o YOLOV4 e YOLOV5. Utilizando a plataforma de deep learning PaddlePaddle (daí o "PP" no nome), o modelo adiciona dez técnicas existentes para melhorar a precisão do detector sem alterar sua velocidade. Segundo os autores, o objetivo não era introduzir um detector totalmente novo, mas sim demonstrar como construir um detector melhor passo a passo. A maioria das técnicas utilizadas no PP-YOLO difere das do YOLOV4, e aquelas que se sobrepõem possuem implementações distintas. As principais mudanças em relação ao YOLOV3 são:

Backbone ResNet50-vd: substitui o DarkNet-53 por uma arquitetura com convoluções deformáveis no último estágio e um modelo pré-treinado por destilação, que possui maior precisão de classificação no ImageNet.

Batch size maior: para melhorar a estabilidade do treinamento, o valor passa de 64 para 192, com ajustes no cronograma de treinamento e na taxa de aprendizado.

Médias móveis: utilizadas em vez dos valores finais do treinamento para os parâmetros.

DropBlock: aplicado apenas ao FPN.

Perda de IoU: adicionada em outro ramo junto com a perda L1 para regressão de caixas delimitadoras.

Ramo de previsão de IoU: mede a precisão de localização e utiliza uma perda baseada em IoU. Durante a inferência, o YOLOV3 multiplica a probabilidade de classificação e a pontuação de objetividade para calcular a detecção final. O PP-YOLO também multiplica o IoU previsto para considerar a precisão de localização.

Abordagem sensível à grade: similar ao YOLOV4, melhora a previsão do centro da caixa delimitadora na borda da grade.

NMS de matriz: executado em paralelo, é mais rápido que o NMS tradicional.

CoordConv: utilizado na convolução 1x1 do FPN e na primeira camada de convolução da cabeça de detecção, permite à rede aprender invariância translacional, melhorando a localização da detecção.

Pooling Piramidal Espacial: utilizado apenas no mapa de características superior para aumentar o campo receptivo do backbone.

1.2.6.6. PP-YOLOV2

O desenvolvimento de PP-YOLOV2 envolveu a colaboração entre a Baidu Inc. e a Universidade de Waseda. Ele enfatiza o desafio de melhorar a eficácia do YOLOV3 sem sacrificar a velocidade para aplicações práticas. Os autores propõem refinamentos para o PP-YOLO, visando a um equilíbrio entre desempenho e eficiência, o que acaba levando ao desenvolvimento

do PP-YOLOv2 (HUANG et al., 2021). A PP-YOLOv2 aprimorou a PP-YOLO em quatro aspectos, melhorando seu desempenho. As mudanças em relação à PP-YOLO são (TERVEN; CÓRDOVA-ESPARZA; ROMERO-GONZÁLEZ, 2023):

Troca da backbone ResNet50 para ResNet101.

Rede de agregação de caminhos (PAN) em vez de FPN, similar à YOLOv4.

Função de ativação Mish. Diferente de YOLOv4 e YOLOv5, a Mish foi aplicada apenas no gargalo de detecção, mantendo a backbone com ReLU.

Maiores tamanhos de entrada para melhorar o desempenho em objetos pequenos. O maior tamanho passou de 608 para 768 e o batch size foi reduzido de 24 para 12 imagens por GPU.

1.2.6.7. PP-YOLOE

A PP-YOLOE introduziu melhorias sobre a PP-YOLOV2, incorporando um paradigma sem âncoras, uma espinha dorsal mais poderosa e um gargalo equipado com CSPRepResStage, ET-head e o algoritmo de atribuição dinâmica de rótulos TAL (XU, SHANGLIANG et al., 2022). As principais mudanças em relação à PP-YOLOV2 são:

A PP-YOLOE utiliza uma arquitetura sem âncoras.

Nova backbone e neck. Inspirados pelo TreeNet, os autores modificaram a arquitetura da backbone e neck com RepResBlocks, combinando conexões residuais e densas.

Aprendizado de alinhamento de tarefas (TAL). O YOLOX foi o primeiro a abordar o problema do desalinhamento de tarefas, onde a confiança na classificação e a precisão da localização nem sempre concordam. Para reduzir isso, a PP-YOLOE implementou o TAL proposto no TOOD, que inclui atribuição dinâmica de rótulos combinada com uma perda de alinhamento de tarefas.

Cabeça eficiente alinhada à tarefa (ET-head). Diferente do YOLOX, onde as cabeças de classificação e localização foram desacopladas, a PP-YOLOE usou uma única cabeça baseada no TOOD para melhorar a velocidade e precisão.

Varifocal (VFL) e perda focal de distribuição (DFL). A VFL pondera a perda de amostras positivas usando a pontuação do alvo, dando maior peso àquelas com alto IoU. Isso prioriza amostras de alta qualidade durante o treinamento. Ambas utilizam a pontuação de classificação baseada em IoU (IACS) como alvo, permitindo o aprendizado conjunto da classificação e da qualidade da localização, levando à consistência entre treinamento e inferência. Por outro lado, a DFL estende a perda focal de rótulos discretos para contínuos, possibilitando a otimização bem-sucedida de representações aprimoradas que combinam estimativa de qualidade e previsão de classe. Isso permite uma representação precisa da distribuição flexível em dados reais, eliminando o risco de inconsistência.

1.2.6.8. YOLO-NAS

O YOLO-NAS é uma nova arquitetura introduzida pela Deci, que visa redefinir o estado da arte em detecção de objetos. Ele foi projetado para superar modelos YOLO anteriores, aprimorando a detecção de objetos pequenos, a precisão de localização e a relação desempenho por computação. Esta arquitetura é particularmente adequada para aplicações do mundo real onde baixa latência e processamento eficiente são cruciais, como veículos autônomos e robótica. O YOLO-NAS também suporta quantização INT8 e é compatível com engines de inferência de alto desempenho como o NVIDIA® TensorRT™, tornando-o altamente prático para uso em produção. A biblioteca SuperGradients auxilia ainda mais no ajuste fino do YOLO-NAS para diversas tarefas, promovendo sua adoção em diferentes indústrias. Os avanços no YOLO-NAS são (KHVEDCHENYA; SAHOTA, 2023):

A arquitetura utiliza mecanismos de atenção, blocos com quantização aware e reparametrização no momento da inferência. Essas técnicas contribuem para seu desempenho excepcional na detecção de objetos de tamanhos e complexidades variados.

Motor AutoNAC: O algoritmo de pesquisa de arquitetura neural (NAS) proprietário da Deci, AutoNAC, foi usado para descobrir o YOLO-NAS. Ele explora com eficiência um vasto espaço de busca para encontrar arquiteturas otimizadas que equilibram precisão, velocidade e complexidade.

1.2.7. Síntese estrutural: elementos comuns às versões YOLO

Apesar da diversidade de versões oficiais e de variantes complementares apresentadas nas seções anteriores, todas as implementações da família YOLO compartilham um mesmo princípio estrutural: são detectores de estágio único que tratam a detecção como um problema de regressão e produzem, em uma única passada pela rede, as caixas delimitadoras e as probabilidades de classe (REDMON et al., 2016). Essa característica fundadora, na qual a imagem é dividida em uma grade e cada célula é responsável por prever caixas, confiança e classe, permanece reconhecível desde o YOLOv1 até o YOLOv26, ainda que os componentes internos tenham sido profundamente reformulados ao longo do tempo (TERVEN; CÓRDOVA-ESPARZA; ROMERO-GONZÁLEZ, 2023).

Do ponto de vista arquitetural, a consolidação mais importante foi a adoção do esqueleto *Backbone-Neck-Head*, que se estabeleceu firmemente a partir do YOLOv4 e passou a organizar praticamente todas as versões posteriores (BOCHKOVSKIY; WANG; LIAO, 2020). O *backbone* é responsável pela extração de características em múltiplas resoluções; o *neck* realiza a fusão multiescala dessas características, aproximando informação semântica profunda de detalhes espaciais finos; e a *head* converte essa representação em predições de caixas e classes. Essa modularidade explica por que versões tão distintas podem ser descritas sob um mesmo diagrama: o que muda entre elas é, sobretudo, a implementação de cada bloco, e não a lógica geral do fluxo de detecção.

O que diferencia as versões, portanto, é a evolução interna de cada componente. No *backbone*, a progressão vai das redes Darknet-19 e Darknet-53 às variantes com conexões parciais entre estágios (CSPDarknet), passando por blocos reparametrizáveis (EfficientRep, RepVGG), por redes de agregação eficiente de camadas (E-ELAN e GELAN) e, mais recentemente, por blocos como C2f e C3k2 e por mecanismos de atenção. No *neck*, observa-se a trajetória que parte da rede de pirâmide de características (FPN) rumo à rede de agregação de caminhos (PAN/PANet) e a esquemas mais elaborados de fusão, como BiFPN e RepGFPN, frequentemente combinados ao *pooling* piramidal espacial (SPP/SPPF). Na *head*, a tendência mais marcante é o abandono progressivo das âncoras e do pós-processamento externo: das cabeças acopladas baseadas em âncoras com Supressão de Não Máximos (NMS), passa-se às cabeças desacopladas, depois aos detectores *anchor-free* e, por fim, à inferência *end-to-end* sem NMS, presente no YOLOv10 e no YOLO26 (WANG et al., 2024; JOCHER et al., 2026).

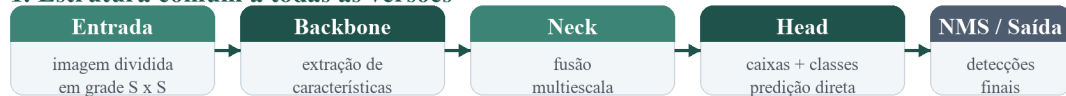
Além do esqueleto compartilhado, outros elementos são recorrentes em toda a família: a divisão da imagem em grade e a predição direta de caixas; a escalabilidade por tamanhos (n, s, m, l, x), que permite ajustar o compromisso entre latência e acurácia; o uso intensivo de técnicas de treinamento sem custo de inferência (*bag-of-freebies*), como aumento de dados e novas funções de perda (IoU, GIoU, CIoU, DFL); e o refinamento das estratégias de atribuição de rótulos (por exemplo, TAL, simOTA e STAL). Também as variantes complementares (Scaled-YOLOv4, YOLOR, YOLOX, PP-YOLO e suas evoluções, DAMO-YOLO e YOLO-NAS) mantêm o mesmo paradigma *Backbone-Neck-Head*, diferenciando-se por escolhas específicas de backbone, *neck*, cabeça ou processo de busca de arquitetura.

A Figura 1.34 sintetiza esse entendimento em três níveis: a estrutura comum a todas as versões, a evolução de cada componente e a genealogia que interliga as versões oficiais e as variantes complementares. O esquema evidencia que a família YOLO deve ser compreendida menos como uma sucessão de modelos isolados e mais como o aprimoramento contínuo de um mesmo esqueleto de detecção, cuja trajetória converge para detectores cada vez mais simples de implantar, *anchor-free* e *end-to-end*.

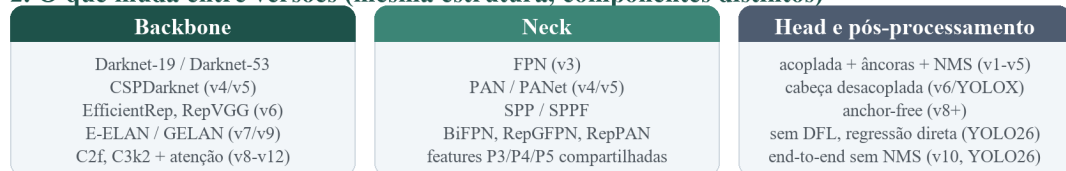
Síntese estrutural das versões YOLO

Esqueleto Backbone-Neck-Head comum e genealogia que interliga todas as versões

1. Estrutura comum a todas as versões



2. O que muda entre versões (mesma estrutura, componentes distintos)



3. Genealogia: linha principal Ultralytics e variantes complementares

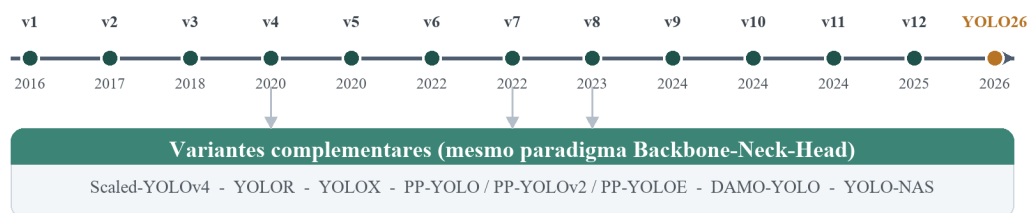


Figura 1.34. Esquema unificado da estrutura comum e da genealogia das versões YOLO, destacando o esqueleto Backbone-Neck-Head compartilhado. Fonte: Elaboração própria.

1.3. CONCLUSÕES

O capítulo estabeleceu, de forma coerente, os elementos essenciais do pipeline YOLO, como a divisão da imagem em grade, a predição conjunta de caixas e classes e a organização do tensor de saída, fornecendo a base técnica necessária para projetos que exigem rapidez e previsões unificadas. Ao articular fundamentos de redes neurais convolucionais, operação do detector e métricas de avaliação, o texto constitui um alicerce metodológico para as etapas subsequentes da tese.

A apresentação de precisão e recall, das métricas $mAP@50$ e $mAP@50:95$ e do F1-score, com sua curva de confiança, permite comparar modelos sob diferentes regimes de sobreposição e limiares, evidenciando os compromissos clássicos entre precisão e sensibilidade. Ficou explícito que elevar o limiar de confiança melhora a precisão ao custo de reduzir o recall, ao passo que reduzi-lo produz o efeito inverso, uma consciência operacional crucial em aplicações nas quais o erro de omissão (falso negativo) é tão grave quanto o de comissão (falso positivo), como na contagem de toras em pilhas densas.

A formalização da Interseção sobre União (IoU) e a inclusão do pseudocódigo da Supressão de Não Máximos (NMS) reforçaram a importância de remover detecções redundantes e priorizar as mais representativas, ponto determinante em cenários de alta sobreposição. Esses conceitos também esclarecem por que as versões mais recentes buscam substituir o NMS por inferência

end-to-end, reduzindo hiperparâmetros sensíveis e simplificando a implantação do detector.

A revisão da evolução do modelo, do YOLOv1 ao YOLO26, complementada pelas variantes Scaled-YOLOv4, YOLOR, YOLOX, PP-YOLO, DAMO-YOLO e YOLO-NAS, e sintetizada no esquema estrutural que interliga as versões, evidenciou que a família YOLO deve ser compreendida como o aprimoramento contínuo de um mesmo esqueleto Backbone-Neck-Head, cuja trajetória converge para detectores anchor-free e end-to-end cada vez mais eficientes. Esse panorama posiciona o leitor para selecionar versões conforme restrições de latência e hardware, mantendo a aderência a aplicações em tempo real.

Em conjunto, os fundamentos, as métricas e a sistematização das arquiteturas apresentados neste capítulo habilitam a escolha criteriosa de um detector YOLO e a interpretação rigorosa de seus resultados, condições indispensáveis para as contribuições dos capítulos seguintes, voltados à detecção e à contagem automática de toras de eucalipto e à consequente estimativa do volume de madeira empilhada.

1.4. REFERÊNCIAS

- ALEKSANDROV, A.; VÖLLINGER, K. **Formalizing Piecewise Affine Activation Functions of Neural Networks in Coq**. [S. l.: s. n.], 2023. p. 62-78. https://doi.org/10.1007/978-3-031-33170-1_4.
- APICELLA, A.; DONNARUMMA, F.; ISGRÒ, F.; PREVETE, R. A survey on modern trainable activation functions. **Neural Networks**, v. 138, p. 14-32, jun. 2021. <https://doi.org/10.1016/j.neunet.2021.01.026>.
- AYYADEVARA, V. K.; REDDY, Yeshwanth. **Modern Computer Vision with PyTorch: Explore deep learning concepts and implement over 50 real-world image applications**. [S. l.]: Packt Publishing Ltd., 2020.
- BHUYAN, M. K. **Computer Vision and Image Processing**. Boca Raton, FL: CRC Press/Taylor & Francis Group, 2019. <https://doi.org/10.1201/9781351248396>.
- BISHOP, C. M. **Pattern Recognition and Machine Learning**. [S. l.]: Springer Science+Business Media, LLC., 2006.
- BOCHKOVSKIY, A.; WANG, C. Y.; LIAO, H. Y. M. **Yolov4: Optimal speed and accuracy of object detection**. arXiv:2004.10934v1, 2020. <https://doi.org/10.48550/arXiv.2004.10934>.
- BOESCH, Gaudenz. **What is Intersection over Union (IoU)?** 2024. Viso.ai. Disponível em: <https://viso.ai/computer-vision/intersection-over-union-iou/>. Acessado em: 5 abr. 2024.
- CAI, S.; SHU, Y.; CHEN, G.; OOI, B. C.; WANG, W.; ZHANG, M. **Effective and Efficient Dropout for Deep Convolutional Neural Networks**. arXiv:1904.03392v5, 6 abr. 2019. Disponível em: <http://arxiv.org/abs/1904.03392>.
- CAI, Z.; RAVICHANDRAN, A.; MAJI, S.; FOWLKES, C.; TU, Z.; SOATTO, S. **Exponential Moving Average Normalization for Self-supervised and Semi-supervised Learning**. arXiv:2101.08482v2, 21 jan. 2021. DOI 10.48550/arXiv.2101.08482. Disponível em: <http://arxiv.org/abs/2101.08482>.

- CASAS, G. G.; ISMAIL, Z. H.; LIMEIRA, M. M. C.; DA SILVA, A. A. L.; LEITE, H. G. Automatic Detection and Counting of Stacked Eucalypt Timber Using the YOLOv8 Model. **Forests**, v. 14, n. 12, p. 2369, 4 dez. 2023. <https://doi.org/10.3390/f14122369>.
- CHAKRABARTY, S. **YOLO26: An Analysis of NMS-Free End to End Framework for Real-Time Object Detection**. arXiv:2601.12882v2, 18 mar. 2026. <https://doi.org/10.48550/arXiv.2601.12882>.
- CHEN, L.-C.; PAPANDREOU, G.; KOKKINOS, I.; MURPHY, K.; YUILLE, A. L. **DeepLab: Semantic Image Segmentation with Deep Convolutional Nets, Atrous Convolution, and Fully Connected CRFs**. arXiv:1606.00915v2, 2 jun. 2016. DOI 10.48550/arXiv.1606.00915. Disponível em: <http://arxiv.org/abs/1606.00915>.
- CRESWELL, A.; ARULKUMARAN, K.; BHARATH, A. A. **On denoising autoencoders trained to minimise binary cross-entropy**. arXiv:1708.08487, 28 ago. 2017. Disponível em: <http://arxiv.org/abs/1708.08487>.
- DAI, J.; LI, Y.; HE, K.; SUN, J. **R-FCN: Object Detection via Region-based Fully Convolutional Networks**. arXiv:1605.06409v3, 20 maio 2016. DOI 10.48550/arXiv.1605.06409. Disponível em: <http://arxiv.org/abs/1605.06409>.
- DAVIES, E. R. **The dramatically changing face of computer vision**. Advanced Methods and Deep Learning in Computer Vision. [S. l.]: Elsevier, 2022. p. 1-91. <https://doi.org/10.1016/B978-0-12-822109-9.00010-2>.
- DEVRIES, T.; TAYLOR, G. W. **Improved Regularization of Convolutional Neural Networks with Cutout**. arXiv:1708.04552, 2017. Disponível em: <http://arxiv.org/abs/1708.04552>.
- DIWAN, T.; ANIRUDH, G.; TEMBHURNE, J. V. Object detection using YOLO: challenges, architectural successors, datasets and applications. **Multimedia Tools and Applications**, v. 82, n. 6, p. 9243-9275, 8 mar. 2023. <https://doi.org/10.1007/s11042-022-13644-y>.
- DU, X.; LIN, T.-Y.; JIN, P.; GHIASI, G.; TAN, M.; CUI, Y.; LE, Q. V.; SONG, X. **SpineNet: Learning Scale-Permuted Backbone for Recognition and Localization**. arXiv:1912.05027v3, 10 dez. 2019. DOI 10.48550/arXiv.1912.05027. Disponível em: <http://arxiv.org/abs/1912.05027>.
- DUAN, K.; BAI, S.; XIE, L.; QI, H.; HUANG, Q.; TIAN, Q. **CenterNet: Keypoint Triplets for Object Detection**. arXiv:1904.08189v3, 17 abr. 2019. DOI 10.48550/arXiv.1904.08189. Disponível em: <http://arxiv.org/abs/1904.08189>.
- DUBEY, S. R.; SINGH, S. K.; CHAUDHURI, B. B. Activation functions in deep learning: A comprehensive survey and benchmark. **Neurocomputing**, v. 503, p. 92-108, set. 2022. <https://doi.org/10.1016/j.neucom.2022.06.111>.
- DUSMANU, M.; ROCCO, I.; PAJDLA, T.; POLLEFEYS, M.; SIVIC, J.; TORII, A.; SATTLER, T. **D2-Net: A Trainable CNN for Joint Detection and Description of Local Features**. arXiv:1905.03561, 9 maio 2019. DOI 10.48550/arXiv.1905.03561. Disponível em: <http://arxiv.org/abs/1905.03561>.
- GAO, Y.; SUN, F. Batch normalization followed by merging is powerful for phenotype prediction integrating multiple heterogeneous studies. **PLOS Computational Biology**, v. 19, n. 10, p. e1010608, 16 out. 2023. <https://doi.org/10.1371/journal.pcbi.1010608>.
- GE, Z.; LIU, S.; WANG, F.; LI, Z.; SUN, J. **YOLOX: Exceeding YOLO Series in 2021**. arXiv:2107.08430v2, 2021. DOI 10.48550/arXiv.2107.08430. Disponível em:

<https://arxiv.org/abs/2107.08430>.

GHIASI, G.; LIN, T.-Y.; LE, Q. V. **DropBlock: A regularization method for convolutional networks**. arXiv:1810.12890, 30 out. 2018. Disponível em: <http://arxiv.org/abs/1810.12890>.

GHIASI, G.; LIN, T.-Y.; PANG, R.; LE, Q. V. **NAS-FPN: Learning Scalable Feature Pyramid Architecture for Object Detection**. arXiv:1904.07392v1, 15 abr. 2019. DOI 10.48550/arXiv.1904.07392. Disponível em: <http://arxiv.org/abs/1904.07392>.

GIRSHICK, R. Fast r-cnn. 2015. **Proceedings of the IEEE international conference on computer vision [...]**. [S. l.: s. n.], 2015. p. 1440-1448.

GLOROT, X.; BORDES, A.; BENGIO, Y. Deep Sparse Rectifier Neural Networks. 2011. **Proceedings of the Fourteenth International Conference on Artificial Intelligence and Statistics**. [...]. [S. l.: s. n.], 2011. p. 315-323. Disponível em: <https://proceedings.mlr.press/v15/glorot11a.html>. Acessado em: 7 abr. 2024.

GOODALE, M. A.; MILNER, A. D. Separate visual pathways for perception and action. **Trends in Neurosciences**, v. 15, n. 1, p. 20-25, jan. 1992. [https://doi.org/10.1016/0166-2236\(92\)90344-8](https://doi.org/10.1016/0166-2236(92)90344-8).

HAO, W.; YIZHOU, W.; YAQIN, L.; ZHILI, S. The Role of Activation Function in CNN. 1 dez. 2020. **Proceedings - 2020 2nd International Conference on Information Technology and Computer Application, ITCA 2020 [...]**. [S. l.]: Institute of Electrical and Electronics Engineers Inc., 1 dez. 2020. p. 429-432. <https://doi.org/10.1109/ITCA52113.2020.00096>.

HE, K.; GKIOXARI, G.; DOLLÁR, P.; GIRSHICK, R. **Mask R-CNN**. arXiv:1703.06870v3, 20 mar. 2017. DOI 10.48550/arXiv.1703.06870. Disponível em: <http://arxiv.org/abs/1703.06870>.

HE, K.; ZHANG, X.; REN, S.; SUN, J. **Deep Residual Learning for Image Recognition**. arXiv:1512.03385v1, 10 dez. 2015a. DOI 10.48550/arXiv.1512.03385. Disponível em: <http://arxiv.org/abs/1512.03385>.

HE, K.; ZHANG, X.; REN, S.; SUN, J. Spatial Pyramid Pooling in Deep Convolutional Networks for Visual Recognition. **IEEE Transactions on Pattern Analysis and Machine Intelligence**, v. 37, n. 9, p. 1904-1916, 1 set. 2015b. <https://doi.org/10.1109/TPAMI.2015.2389824>.

HORVAT, M.; JELEČEVIĆ, L.; GLEDEC, G. A comparative study of YOLOv5 models performance for image localization and classification. 2022. **33rd Central European Conference on Information and Intelligent Systems (CECIIS 2022) [...]**. [S. l.: s. n.], 2022. Disponível em: <https://www.researchgate.net/publication/363824867>.

HUANG, X.; WANG, X.; LV, W.; BAI, X.; LONG, X.; DENG, K.; DANG, Q.; HAN, S.; LIU, Q.; HU, X.; YU, D.; MA, Y.; YOSHIE, O. **PP-YOLOv2: A Practical Object Detector**. arXiv:2104.10419v1, 21 abr. 2021. DOI 10.48550/arXiv.2104.10419. Disponível em: <http://arxiv.org/abs/2104.10419>.

HUSSAIN, M. YOLO-v1 to YOLO-v8, the Rise of YOLO and Its Complementary Nature toward Digital Manufacturing and Industrial Defect Detection. **Machines**, v. 11, n. 7, p. 677, 23 jun. 2023. <https://doi.org/10.3390/machines11070677>.

IOFFE, S.; SZEGEDY, C. **Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift**. arXiv:1502.03167, 10 fev. 2015. Disponível em: <http://arxiv.org/abs/1502.03167>.

JAIN, S.; MITTAL, R.; JAIN, A.; SINHA, A. An efficient framework for monitoring tree cover

in an area through aerial images. 6 set. 2019. **Applications of Machine Learning [...]**. [S. l.]: SPIE, 6 set. 2019. p. 41. <https://doi.org/10.1117/12.2529442>.

JANI, M.; FAYYAD, J.; AL-YOUNES, Y.; NAJJARAN, H. **Model Compression Methods for YOLOv5: A Review**. arXiv:2307.11904v1, 21 jul. 2023. DOI 10.48550/arXiv.2307.11904. Disponível em: <http://arxiv.org/abs/2307.11904>.

JOCHER, G. **YOLOv5 by Ultralytics (Version 7.0)**. 2020. <https://doi.org/10.5281/zenodo.3908559>.

JOCHER, G.; CHAURASIA, A.; QIU, J. **YOLO by Ultralytics (Version 8.0.0)**. 2023. Disponível em: <https://github.com/ultralytics/ultralytics>. Acessado em: 22 jul. 2023.

JOCHER, G.; QIU, J.; LIU, M.; LYU, S.; AKYON, F. C.; KALFAOGLU, M. E. **Ultralytics YOLO26: Unified Real-Time End-to-End Vision Models**. arXiv:2606.03748, 2026. <https://doi.org/10.48550/arXiv.2606.03748>.

KHANAM, R.; HUSSAIN, M. **YOLOv11: An Overview of the Key Architectural Enhancements**. arXiv:2410.17725, 2024. <https://doi.org/10.48550/arXiv.2410.17725>.

KHVEDCHENYA, Eugene.; SAHOTA, Harpreet. **YOLO-NAS by Deci Achieves SOTA Performance on Object Detection Using Neural Architecture Search**. 3 maio 2023. Deci. Disponível em: <https://deci.ai/blog/yolo-nas-object-detection-foundation-model/>. Acessado em: 29 abr. 2024.

KUNC, V.; KLÉMA, J. **Three Decades of Activations: A Comprehensive Survey of 400 Activation Functions for Neural Networks**. arXiv:2402.09092, 14 fev. 2024. DOI 10.48550/arXiv.2402.09092. Disponível em: <http://arxiv.org/abs/2402.09092>.

KUNDU, Rohit. **F1 Score in Machine Learning: Intro & Calculation**. 2022. V7. Disponível em: <https://www.v7labs.com/blog/f1-score-guide>. Acessado em: 5 abr. 2024.

LARSSON, G.; MAIRE, M.; SHAKHNAROVICH, G. **FractalNet: Ultra-Deep Neural Networks without Residuals**. arXiv:1605.07648v4, 24 maio 2016. Disponível em: <http://arxiv.org/abs/1605.07648>.

LAW, H.; DENG, J. **CornerNet: Detecting Objects as Paired Keypoints**. arXiv:1808.01244v2, 3 ago. 2018. DOI 10.48550/arXiv.1808.01244. Disponível em: <http://arxiv.org/abs/1808.01244>.

LECUN, Y.; BOTTOU, L.; BENGIO, Y.; HAFFNER, P. Gradient-based learning applied to document recognition. **Proceedings of the IEEE**, v. 86, n. 11, p. 2278-2324, 1998. <https://doi.org/10.1109/5.726791>.

LEDERER, J. **Activation Functions in Artificial Neural Networks: A Systematic Overview**. arXiv:2101.09957, 25 jan. 2021. Disponível em: <http://arxiv.org/abs/2101.09957>.

LI, Chuyi.; LI, Lulu.; JIANG, Hongliang.; WENG, Kaiheng.; GENG, Yifei.; LI, Liang.; KE, Zaidan.; LI, Qingyuan.; CHENG, Meng.; NIE, Weiqiang.; LI, Yiduo.; ZHANG, Bo.; LIANG, Yufei.; ZHOU, Linyuan.; XU, Xiaoming.; CHU, Xiangxiang.; WEI, Xiaoming.; WEI, Xiaolin. **YOLOv6: A single-stage object detection framework for industrial applications**. arXiv preprint arXiv:2209.02976, 2022. <https://doi.org/10.48550/arXiv.2209.02976>.

LI, X.; WANG, W.; WU, L.; CHEN, S.; HU, X.; LI, J.; TANG, J.; YANG, J. Generalized Focal Loss: Learning Qualified and Distributed Bounding Boxes for Dense Object Detection. **Advances in Neural Information Processing Systems**, v. 33, p. 21002-21012, 2020.

- MATHEW, J.; DELAVARPOUR, N.; MIRANDA, C.; STENGER, J.; ZHANG, Z.; ADUT-EYE, J.; FLORES, P. A Novel Approach to Pod Count Estimation Using a Depth Camera in Support of Soybean Breeding Applications. *Sensors*, v. 23, n. 14, p. 6506, 2023. <https://doi.org/10.3390/s23146506>.
- LIN, T.-Y.; GOYAL, P.; GIRSHICK, R.; HE, K.; DOLLÁR, P. **Focal Loss for Dense Object Detection**. arXiv:1708.02002v2, 7 ago. 2017. DOI 10.48550/arXiv.1708.02002. Disponível em: <http://arxiv.org/abs/1708.02002>.
- LIN, T.-Yi.; DOLLÁR, Piotr.; GIRSHICK, Ross.; HE, Kaiming.; HARIHARAN, Bharath.; BELONGIE, Serge. Feature Pyramid Networks for Object Detection. 2017. **Proceedings of the IEEE conference on computer vision and pattern recognition [...]**. [S. l.]: , 2017. p. 2117-2125. <https://doi.org/10.48550/arXiv.1612.03144>.
- LIU, Shu; QI, L.; QIN, H.; SHI, J.; JIA, J. **Path Aggregation Network for Instance Segmentation**. arXiv:1803.01534, 5 mar. 2018. DOI 10.48550/arXiv.1803.01534. Disponível em: <http://arxiv.org/abs/1803.01534>.
- LIU, Songtao; HUANG, D.; WANG, Y. **Learning Spatial Fusion for Single-Shot Object Detection**. arXiv:1911.09516v2, 21 nov. 2019. DOI 10.48550/arXiv.1911.09516. Disponível em: <http://arxiv.org/abs/1911.09516>.
- LIU, Songtao; HUANG, D.; WANG, Y. **Receptive Field Block Net for Accurate and Fast Object Detection**. arXiv:1711.07767v3, 21 nov. 2017. DOI 10.48550/arXiv.1711.07767. Disponível em: <http://arxiv.org/abs/1711.07767>.
- LIU, W.; ANGUELOV, D.; ERHAN, D.; SZEGEDY, C.; REED, S.; FU, C.-Y.; BERG, A. C. **SSD: Single Shot MultiBox Detector**. Computer Vision - ECCV 2016. ECCV 2016. Lecture Notes in Computer Science. Cham: Springer, 2016. vol. 9905, p. 21-37. DOI 10.1007/978-3-319-46448-0_2. Disponível em: http://link.springer.com/10.1007/978-3-319-46448-0_2.
- LONG, X.; DENG, K.; WANG, G.; ZHANG, Y.; DANG, Q.; GAO, Y.; SHEN, H.; REN, J.; HAN, S.; DING, E.; WEN, S. **PP-YOLO: An Effective and Efficient Implementation of Object Detector**. arXiv:2007.12099v3, 23 jul. 2020. DOI 10.48550/arXiv.2007.12099. Disponível em: <http://arxiv.org/abs/2007.12099>.
- MAHASIN, M.; DEWI, I. A. Comparison of CSPDarkNet53, CSPResNeXt-50, and EfficientNet-B0 Backbones on YOLO V4 as Object Detector. *International Journal of Engineering, Science & Information Technology*, v. 2, n. 3, p. 64-72, 2022. DOI 10.52088/ijesty.v1i4.291. Disponível em: <https://doi.org/10.52088/ijesty.v1i4.291>.
- MISRA, D. **Mish: A Self Regularized Non-Monotonic Activation Function**. arXiv:1908.08681, 23 ago. 2019. Disponível em: <http://arxiv.org/abs/1908.08681>.
- NAIR, V.; HINTON, G. E. Rectified Linear Units Improve Restricted Boltzmann Machines. 2010. **Proceedings of the 27th International Conference on Machine Learning (ICML-10) [...]**. [S. l.]: Omnipress, 2010. p. 807-814. Disponível em: <https://www.cs.toronto.edu/~fritz/absps/reluICML.pdf>. Acessado em: 7 abr. 2024.
- NEPAL, U.; ESLAMIAT, H. Comparing YOLOv3, YOLOv4 and YOLOv5 for Autonomous Landing Spot Detection in Faulty UAVs. *Sensors*, v. 22, n. 2, p. 464, 8 jan. 2022. <https://doi.org/10.3390/s22020464>.
- OGUNDOKUN, R. O.; MASKELIUNAS, R.; MISRA, S.; DAMAŠEVIČIUS, R. **Improved CNN Based on Batch Normalization and Adam Optimizer**. [S. l.: s. n.], 2022. p. 593-604.

https://doi.org/10.1007/978-3-031-10548-7_43.

OLORUNSHOLA, O. E.; IRHEBHUDE, M. E.; EVWIEKPAEFE, A. E. A Comparative Study of YOLOv5 and YOLOv7 Object Detection Algorithms. **Journal of Computing and Social Informatics**, v. 2, n. 1, p. 1-12, 8 fev. 2023. DOI 10.33736/jcsi.5070.2023. Disponível em: <https://publisher.unimas.my/ojs/index.php/jcsi/article/view/5070>.

PEBRIANTO, W.; MUDJIRAHARDJO, P.; HADI PRAMONO, S.; ARIEF SETYAWAN, R. **YOLOv3 with Spatial Pyramid Pooling for Object Detection with Unmanned Aerial Vehicles**. arXiv:2305.12344., 2023. <https://doi.org/10.48550/arXiv.2305.12344>.

PENG, H.; YU, Y.; YU, S. Re-Thinking the Effectiveness of Batch Normalization and Beyond. **IEEE Transactions on Pattern Analysis and Machine Intelligence**, v. 46, n. 1, p. 465-478, jan. 2024. <https://doi.org/10.1109/TPAMI.2023.3319005>.

PRINCE, S. J. D. **Computer vision: models, learning, and inference**. 1st edition. [S. l.]: Cambridge University Press., 2012.

RASHWAN, A.; KALRA, A.; POUPART, P. **Matrix Nets: A New Deep Architecture for Object Detection**. arXiv:1908.04646v2, 13 ago. 2019. DOI 10.48550/arXiv.1908.04646. Disponível em: <http://arxiv.org/abs/1908.04646>.

REDMON, J.; DIVVALA, S.; GIRSHICK, R.; FARHADI, A. You Only Look Once: Unified, Real-Time Object Detection. 2016. **Proceedings of the IEEE conference on computer vision and pattern recognition [...]**. [S. l.: s. n.], 2016. p. 779-788.

REDMON, J.; FARHADI, A. YOLO9000: better, faster, stronger. 2017. **Proceedings of the IEEE conference on computer vision and pattern recognition [...]**. [S. l.: s. n.], 2017. p. 7263-7271.

REDMON, J.; FARHADI, A. **Yolov3: An incremental improvement**. arXiv preprint arXiv:1804.02767, 2018. <https://doi.org/10.48550/arXiv.1804.02767>.

REN, S.; HE, K.; GIRSHICK, R.; SUN, J. **Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks**. arXiv:1506.01497v3, 4 jun. 2015. DOI 10.48550/arXiv.1506.01497. Disponível em: <http://arxiv.org/abs/1506.01497>.

REZATOFIGHI, H.; TSOI, N.; GWAK, J.; SADEGHIAN, A.; REID, I.; SAVARESE, S. Generalized Intersection over Union: A Metric and A Loss for Bounding Box Regression. 2019. **Proceedings of the IEEE/CVF conference on computer vision and pattern recognition [...]**. [S. l.: s. n.], 2019. p. 658-666.

RUI, L.; TANG, X.; HAO, K. DB-NMS: improving non-maximum suppression with density-based clustering. **Neural Computing and Applications**, v. 34, n. 6, p. 4747-4757, 25 mar. 2022. <https://doi.org/10.1007/s00521-021-06628-w>.

SAPKOTA, R.; CHEPPALLY, R. H.; SHARDA, A.; KARKEE, M. **YOLO26: Key Architectural Enhancements and Performance Benchmarking for Real-Time Object Detection**. arXiv:2509.25164v5, 16 mar. 2026. <https://doi.org/10.48550/arXiv.2509.25164>.

SEYREK, E. C.; UYSAL, M. A comparative analysis of various activation functions and optimizers in a convolutional neural network for hyperspectral image classification. **Multimedia Tools and Applications**, 24 nov. 2023. <https://doi.org/10.1007/s11042-023-17546-5>.

SHARMA, D.; INDU, S.; JAYANTHI, N. Lane-Line Segmentation Using YOLOv7. 5 maio 2023. **2023 2nd International Conference on Vision Towards Emerging Trends in Com-**

munication and Networking Technologies (ViTECoN) [...]. [S. l.]: IEEE, 5 maio 2023. p. 1-4. <https://doi.org/10.1109/ViTECoN58111.2023.10157772>.

SHETH, I.; RAHMAN, A. A.; HAVAEI, M.; KAHOU, S. E. **Pitfalls of Conditional Batch Normalization for Contextual Multi-Modal Learning.** arXiv:2211.15071, 28 nov. 2022. DOI 10.48550/arXiv.2211.15071. Disponível em: <http://arxiv.org/abs/2211.15071>.

SIMONYAN, K.; ZISSERMAN, A. **Very Deep Convolutional Networks for Large-Scale Image Recognition.** arXiv:1409.1556, 4 set. 2014. DOI 10.48550/arXiv.1409.1556. Disponível em: <http://arxiv.org/abs/1409.1556>.

SITAWARIN, C.; CHAKRABORTY, S.; WAGNER, D. SAT: Improving Adversarial Training via Curriculum-Based Loss Smoothing. 15 nov. 2021. **AISeC 2021 - Proceedings of the 14th ACM Workshop on Artificial Intelligence and Security, co-located with CCS 2021 [...].** [S. l.]: Association for Computing Machinery, Inc, 15 nov. 2021. p. 25-36. <https://doi.org/10.1145/3474369.3486878>.

SZELISKI, R. **Computer Vision.** Cham: Springer International Publishing, 2022. <https://doi.org/10.1007/978-3-030-34372-9>.

TAN, M.; LE, Q. V. EfficientNet: Rethinking Model Scaling for Convolutional Neural Networks. 2019. **International conference on machine learning [...].** [S. l.: s. n.], 2019. p. 6105-6114.

TAN, M.; PANG, R.; LE, Q. V. **EfficientDet: Scalable and Efficient Object Detection.** arXiv:1911.09070v7, 20 nov. 2019. DOI 10.48550/arXiv.1911.09070. Disponível em: <http://arxiv.org/abs/1911.09070>.

TERVEN, J.; CORDOVA-ESPARZA, D. **A comprehensive review of YOLO: From YOLOv1 to YOLOv8 and beyond.** arXiv preprint arXiv:2304.00501, 2023. <https://doi.org/10.48550/arXiv.2304.00501>.

TERVEN, J.; CORDOVA-ESPARZA, D.-M.; ROMERO-GONZÁLEZ, J.-A. A Comprehensive Review of YOLO Architectures in Computer Vision: From YOLOv1 to YOLOv8 and YOLO-NAS. **Machine Learning and Knowledge Extraction**, v. 5, n. 4, p. 1680-1716, 20 nov. 2023. <https://doi.org/10.3390/make5040083>.

TIAN, Y.; YE, Q.; DOERMANN, D. **YOLOv12: Attention-Centric Real-Time Object Detectors.** arXiv:2502.12524, 2025. <https://doi.org/10.48550/arXiv.2502.12524>.

TIAN, Z.; SHEN, C.; CHEN, H.; HE, T. **FCOS: Fully Convolutional One-Stage Object Detection.** arXiv:1904.01355v5, 2 abr. 2019. DOI 10.48550/arXiv.1904.01355. Disponível em: <http://arxiv.org/abs/1904.01355>.

TING, K. Precision and Recall. **Encyclopedia of Machine Learning.** Boston, MA: Springer US, 2011. p. 781-781. https://doi.org/10.1007/978-0-387-30164-8_652.

TING, Y.-S.; TENG, Y.-F.; CHIUEH, T.-D. Batch Normalization Processor Design for Convolution Neural Network Training and Inference. maio 2021. **2021 IEEE International Symposium on Circuits and Systems (ISCAS) [...].** [S. l.]: IEEE, maio 2021. p. 1-4. <https://doi.org/10.1109/ISCAS51556.2021.9401434>.

WANG, A.; CHEN, H.; LIU, L.; CHEN, K.; LIN, Z.; HAN, J.; DING, G. **YOLOv10: Real-Time End-to-End Object Detection.** arXiv:2405.14458, 2024. <https://doi.org/10.48550/arXiv.2405.14458>.

- WANG, C. Y.; BOCHKOVSKIY, A.; LIAO, H. Y. M. **YOLOv7: Trainable bag-of-freebies sets new state-of-the-art for real-time object detectors.** 2023. **Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition [...].** [S. l.: s. n.], 2023. p. 7464-7475.
- WANG, C.-Y.; BOCHKOVSKIY, A.; LIAO, H.-Y. M. **Scaled-YOLOv4: Scaling Cross Stage Partial Network.** arXiv:2011.08036v2, 16 nov. 2020. Disponível em: <http://arxiv.org/abs/2011.08036>.
- WANG, C.-Y.; LIAO, H.-Y. M.; YEH, I.-H. **Designing Network Design Strategies Through Gradient Path Analysis.** arXiv:2211.04800v1, 9 nov. 2022. DOI 10.48550/arXiv.2211.04800. Disponível em: <http://arxiv.org/abs/2211.04800>.
- WANG, C.-Y.; LIAO, H.-Y. M.; YEH, I.-H.; WU, Y.-H.; CHEN, P.-Y.; HSIEH, J.-W. **CSPNet: A New Backbone that can Enhance Learning Capability of CNN.** arXiv:1911.11929v1, 26 nov. 2019. DOI 10.48550/arXiv.1911.11929. Disponível em: <http://arxiv.org/abs/1911.11929>.
- WANG, C.-Y.; YEH, I.-H.; LIAO, H.-Y. M. **YOLOv9: Learning What You Want to Learn Using Programmable Gradient Information.** arXiv:2402.13616v2, 21 fev. 2024. DOI 10.48550/arXiv.2402.13616. Disponível em: <http://arxiv.org/abs/2402.13616>.
- WANG, C.-Y.; YEH, I.-H.; LIAO, H.-Y. M. **You Only Learn One Representation: Unified Network for Multiple Tasks.** arXiv:2105.04206v1, 10 maio 2021. Disponível em: <http://arxiv.org/abs/2105.04206>.
- WANG, Feng.; TAX, D. M. J. **Survey on the attention based RNN model and its applications in computer vision.** arXiv:1601.06823v1, 25 jan. 2016. DOI 10.48550/arXiv.1601.06823. Disponível em: <http://arxiv.org/abs/1601.06823>.
- WOO, S.; PARK, J.; LEE, J.-Y.; KWEON, I. S. **CBAM: Convolutional Block Attention Module.** arXiv:1807.06521v2, 17 jul. 2018. DOI 10.48550/arXiv.1807.06521. Disponível em: <http://arxiv.org/abs/1807.06521>.
- XU, S.; WANG, X.; LV, W.; CHANG, Q.; CUI, C.; DENG, K.; WANG, G.; DANG, Q.; WEI, S.; DU, Y.; LAI, B. **PP-YOLOE: An evolved version of YOLO.** arXiv:2203.16250v3, 30 mar. 2022. DOI 10.48550/arXiv.2203.16250. Disponível em: <http://arxiv.org/abs/2203.16250>.
- XU, X.; JIANG, Y.; CHEN, W.; HUANG, Y.; ZHANG, Y.; SUN, X. **DAMO-YOLO: A Report on Real-Time Object Detection Design.** arXiv:2211.15444v4, 23 nov. 2022. DOI 10.48550/arXiv.2211.15444. Disponível em: <http://arxiv.org/abs/2211.15444>.
- YANG, Z.; LIU, S.; HU, H.; WANG, L.; LIN, S. **RepPoints: Point Set Representation for Object Detection.** arXiv:1904.11490v2, 25 abr. 2019. DOI 10.48550/arXiv.1904.11490. Disponível em: <http://arxiv.org/abs/1904.11490>.
- YAO, Z.; CAO, Y.; ZHENG, S.; HUANG, G.; LIN, S. **Cross-Iteration Batch Normalization.** arXiv:2002.05712v3, 13 fev. 2020. Disponível em: <http://arxiv.org/abs/2002.05712>.
- YUN, S.; HAN, D.; OH, S. J.; CHUN, S.; CHOE, J.; YOO, Y. **CutMix: Regularization Strategy to Train Strong Classifiers with Localizable Features.** arXiv:1905.04899, 13 maio 2019. Disponível em: <http://arxiv.org/abs/1905.04899>.
- ZHANG, H.; CISSE, M.; DAUPHIN, Y. N.; LOPEZ-PAZ, D. **mixup: Beyond Empirical Risk Minimization.** arXiv:1710.09412, 25 out. 2017. Disponível em: <http://arxiv.org/abs/1710.09412>.

ZHANG, X.; WANG, W.; ZHAO, Y.; XIE, H. An improved YOLOv3 model based on skipping connections and spatial pyramid pooling. **Systems Science & Control Engineering**, v. 9, no sup1, p. 142-149, 1 abr. 2021. <https://doi.org/10.1080/21642583.2020.1824132>.

ZHANG, Zhi.; HE, Tong.; ZHANG, Hang.; ZHANG, Zhongyue.; XIE, Junyuan.; LI, Mu. **Bag of Freebies for Training Object Detection Neural Networks**. arXiv preprint arXiv:1902.04103, 2019. <https://doi.org/10.48550/arXiv.1902.04103>.

ZHAO, Q.; SHENG, T.; WANG, Y.; TANG, Z.; CHEN, Y.; CAI, L.; LING, H. **M2Det: A Single-Shot Object Detector based on Multi-Level Feature Pyramid Network**. arXiv:1811.04533v3, 11 nov. 2018. DOI 10.48550/arXiv.1811.04533. Disponível em: <http://arxiv.org/abs/1811.04533>.

ZHENG, Z.; WANG, P.; LIU, W.; LI, J.; YE, R.; REN, D. Distance-IoU Loss: Faster and Better Learning for Bounding Box Regression. **Proceedings of the AAAI Conference on Artificial Intelligence**, v. 34, n. 07, p. 12993-13000, 3 abr. 2020. <https://doi.org/10.1609/aaai.v34i07.6999>.

ZHU, S.-C.; WU, Y. Statistics of Natural Images. **Computer Vision**. Cham: Springer International Publishing, 2023. p. 19-35. https://doi.org/10.1007/978-3-030-96530-3_2.

CAPÍTULO II

VISÃO COMPUTACIONAL APLICADA À MADEIRA: UM MAPEAMENTO SISTEMÁTICO DA LITERATURA

RESUMO

A aplicação de técnicas de imagem ao processamento da madeira acumula mais de quatro décadas de produção científica, dispersa entre a engenharia florestal, a ciência da computação e a tecnologia da madeira, o que dificulta identificar quais problemas já dispõem de solução consolidada e quais permanecem em aberto. Este capítulo teve por objetivo mapear sistematicamente essa produção, quantificando sua distribuição por tema de aplicação, por abordagem metodológica e ao longo do tempo. Realizou-se busca por regras na base Scopus, nos campos de título, resumo e palavras-chave, combinando termos relativos à madeira e a seus produtos com termos de imageamento e visão computacional. Após remoção de duplicatas e aplicação de critérios estruturados de exclusão, o corpus final reuniu 1.019 artigos publicados entre 1983 e 2026. Cada artigo foi atribuído a uma entre nove categorias de aplicação e a uma entre três classes metodológicas, por classificação baseada em regras com pontuação ponderada, penalidades de exclusão e ordem de prioridade para empates, cuja confiabilidade foi aferida contra rotulagem manual de uma amostra. O controle de qualidade de superfície concentrou 492 artigos, ou 48,3% do total, seguido pelo imageamento tridimensional e interno, com 139 (13,6%), pela identificação de espécie e origem, com 108 (10,6%), e pela microestrutura e caracterização, com 103 (10,1%); o processamento de toras e madeira roliça respondeu por apenas 39 artigos (3,8%). Metodologicamente, as abordagens clássicas ainda predominam, somando 73,6% do corpus, contra 26,4% do aprendizado profundo. A distribuição temporal evidencia forte concentração recente: 63,7% dos artigos foram publicados a partir de 2016. O corpus acumulou 11.961 citações, com média de 11,74 e mediana de 4, e a China lidera a produção com 283 artigos. Conclui-se que a visão computacional aplicada à madeira migrou de nicho especializado para tecnologia industrial madura, com divisão de trabalho estável entre as abordagens, e que a quantificação volumétrica de madeira empilhada figura entre os temas menos explorados, o que situa e justifica os capítulos seguintes desta tese.

Palavras-chave: mapeamento sistemático; análise bibliométrica; visão computacional; processamento da madeira; aprendizado profundo.

2.1. INTRODUÇÃO

O setor de processamento da madeira figura entre os mais antigos da atividade industrial e, ainda assim, permanece em transformação contínua pela incorporação de tecnologias de automação e de controle de qualidade. Entre essas tecnologias, a análise automatizada de imagens ocupa posição particular: diferentemente de sensores que medem grandezas pontuais, uma imagem carrega simultaneamente informação sobre geometria, textura, cor e descontinuidades, o que a torna adequada a tarefas tão distintas quanto a classificação de tábuas, a identificação de espécies e a mensuração de peças (CONNERS et al., 1983; HWANG; SUGIYAMA, 2021a; URBONAS et al., 2019; CHEN et al., 2022).

O marco inicial dessa trajetória costuma ser atribuído ao trabalho de Connors et al. (1983), que demonstrou a viabilidade de localizar automaticamente defeitos superficiais em madeira serrada como parte de um sistema de processamento automatizado. Nas três décadas seguintes, a área desenvolveu-se sobre fundamentos determinísticos: limiarização, morfologia matemática, filtragem espacial e extração manual de descritores de textura e de cor, frequentemente combinados a classificadores estatísticos simples.

A transformação metodológica mais recente decorre da consolidação do aprendizado profundo em visão computacional, impulsionada por arquiteturas que aprendem as próprias representações a partir dos dados, dispensando a engenharia manual de atributos. Entre os marcos dessa transição situam-se a demonstração de que redes profundas podiam ser treinadas de forma eficaz para aprender representações compactas (HINTON; SALAKHUTDINOV, 2006), as redes convolucionais profundas para classificação em larga escala (KRIZHEVSKY et al., 2012), as redes residuais que viabilizaram o treinamento de arquiteturas muito profundas (HE et al., 2016), os detectores de estágio único que reformularam a detecção como problema de regressão (REDMON et al., 2016) e as arquiteturas totalmente convolucionais voltadas à segmentação (RONNEBERGER et al., 2015). Mais recentemente, os detectores baseados em transformadores (CARION et al., 2020; DOSOVITSKIY, 2021), os modelos de atenção (HAN et al., 2023) e os modelos de fundação para segmentação (KIRILLOV et al., 2023) ampliaram o repertório disponível.

Essas ferramentas foram progressivamente transpostas para o domínio da madeira. Na inspeção de superfícies, redes convolucionais passaram a substituir descritores manuais na detecção de nós, rachaduras e manchas (HE et al., 2020b; GAO et al., 2021; YANG et al., 2019). Na identificação de espécies e de procedência, a combinação de imagens anatômicas com aprendizado profundo alcançou acurácias antes inviáveis (RAVINDRAN et al., 2018; DA SILVA et al., 2017; LENS et al., 2020; WU et al., 2021; HERMANSON; WIEDENHOEFT, 2011). Nas operações de pátio e de colheita, a visão computacional passou a apoiar a mensuração e a manipulação robotizada de toras (WIMMER et al., 2021; LI; LIDESKOG, 2021; FORTIN et al., 2022; ANDERSSON et al., 2021). Na produção de painéis, os sistemas de inspeção em linha incorporaram detectores treináveis (GE et al., 2023; HAN et al., 2023; JIA et al., 2023; XIE

et al., 2024). E no imageamento interno, técnicas tomográficas passaram a ser processadas por métodos que combinam reconstrução geométrica e reconhecimento de padrões (NGUYEN et al., 2021; GENG et al., 2021; YE et al., 2022; WEI et al., 2011).

A esses vetores tecnológicos somaram-se, na última década, pressões regulatórias. Exigências de rastreabilidade da origem da madeira, formalizadas em instrumentos como a regulamentação europeia sobre produtos livres de desmatamento (EUROPEAN UNION, 2023), converteram a identificação automatizada de espécies e de procedência, antes tema predominantemente acadêmico, em demanda de conformidade com consequências comerciais diretas.

Apesar da extensão dessa produção, sua organização permanece fragmentada. Trabalhos sobre detecção de nós em madeira serrada convivem, na literatura, com estudos de tomografia de toras, de anatomia da madeira e de robótica em serrarias, publicados em periódicos de comunidades distintas, que raramente se citam entre si. Revisões existentes concentram-se em subdomínios específicos e adotam, em geral, critérios de seleção não explicitados, o que impede comparações quantitativas entre temas e entre abordagens.

Este capítulo tem por objetivo suprir essa lacuna por meio de um mapeamento sistemático da literatura indexada, orientado por quatro questões. Primeira: como a produção científica se distribui entre os temas de aplicação do processamento da madeira? Segunda: qual a participação relativa das abordagens clássicas e das baseadas em aprendizado profundo, e como essa participação evoluiu ao longo do tempo? Terceira: como se distribuem a produção geográfica e o impacto de citação do campo? Quarta: quais temas permanecem sub-representados e constituem, portanto, lacunas de pesquisa? A finalidade não é resenhar exaustivamente cada trabalho, mas produzir um panorama quantitativo e reproduzível que sirva de referência para situar decisões metodológicas, incluindo as adotadas nos capítulos subsequentes desta tese.

2.2. MATERIAIS E MÉTODOS

2.2.1. Fonte de dados e estratégia de busca

Os registros bibliográficos foram obtidos na base Scopus, escolhida por sua cobertura simultânea de periódicos e de anais de congresso nas áreas de engenharia, ciência da computação e ciência da madeira, condição indispensável em um campo cujas contribuições históricas foram publicadas majoritariamente em conferências (SCHMOLDT et al., 1997; FUNT; BRYANT, 1985).

A expressão de busca foi construída pela interseção de dois conjuntos de termos. O primeiro reuniu a terminologia da madeira e de seus produtos, incluindo toras, madeira roliça, madeira serrada, painéis, compensados, aglomerados e produtos engenheirados. O segundo reuniu a terminologia de imageamento e de visão computacional, abrangendo desde o processamento digital de imagens e a extração de descritores até as arquiteturas de aprendizado profundo. A busca foi executada nos campos de título, resumo e palavras-chave em 9 de dezembro de 2025,

sem restrição de ano de publicação, e reteve os itens em acesso antecipado já atribuídos ao volume de 2026.

Foram alvo preferencial os artigos de periódico revisados por pares, os trabalhos completos de congresso, as revisões e os capítulos de livro. Tipos documentais não convencionais, como notas, artigos de dados e comunicações curtas, foram mantidos quando atendiam aos critérios de conteúdo, por contribuírem para a cobertura integral do campo, e receberam tratamento idêntico ao dos demais nas análises subsequentes. Registros retratados, identificados durante a triagem, foram explicitamente excluídos para preservar a integridade dos dados.

2.2.2. Triagem inicial e critérios de elegibilidade

A busca recuperou 2.956 registros. A exportação bruta foi submetida a filtragem automatizada, iniciada pela remoção de duplicatas com base no identificador digital de objeto (DOI), no identificador eletrônico (EID) e na correspondência de títulos, de modo que cada publicação fosse contabilizada uma única vez; dois registros foram eliminados nessa etapa. A triagem automatizada por título, resumo e palavras-chave excluiu 1.888 registros por estarem fora do escopo ou por carecerem de metadados essenciais, e reteve 1.066. Destes, os 102 de maior citação foram submetidos a verificação manual individual, da qual resultou a exclusão de 45 trabalhos fora do escopo, apesar da alta citação, e de dois registros retratados. O corpus final reúne, portanto, 1.019 artigos. O relato do processo de seleção seguiu as recomendações do protocolo PRISMA (PAGE et al., 2021), e o fluxo completo, com o número de registros retidos e excluídos em cada etapa, é apresentado na Figura 2.1.

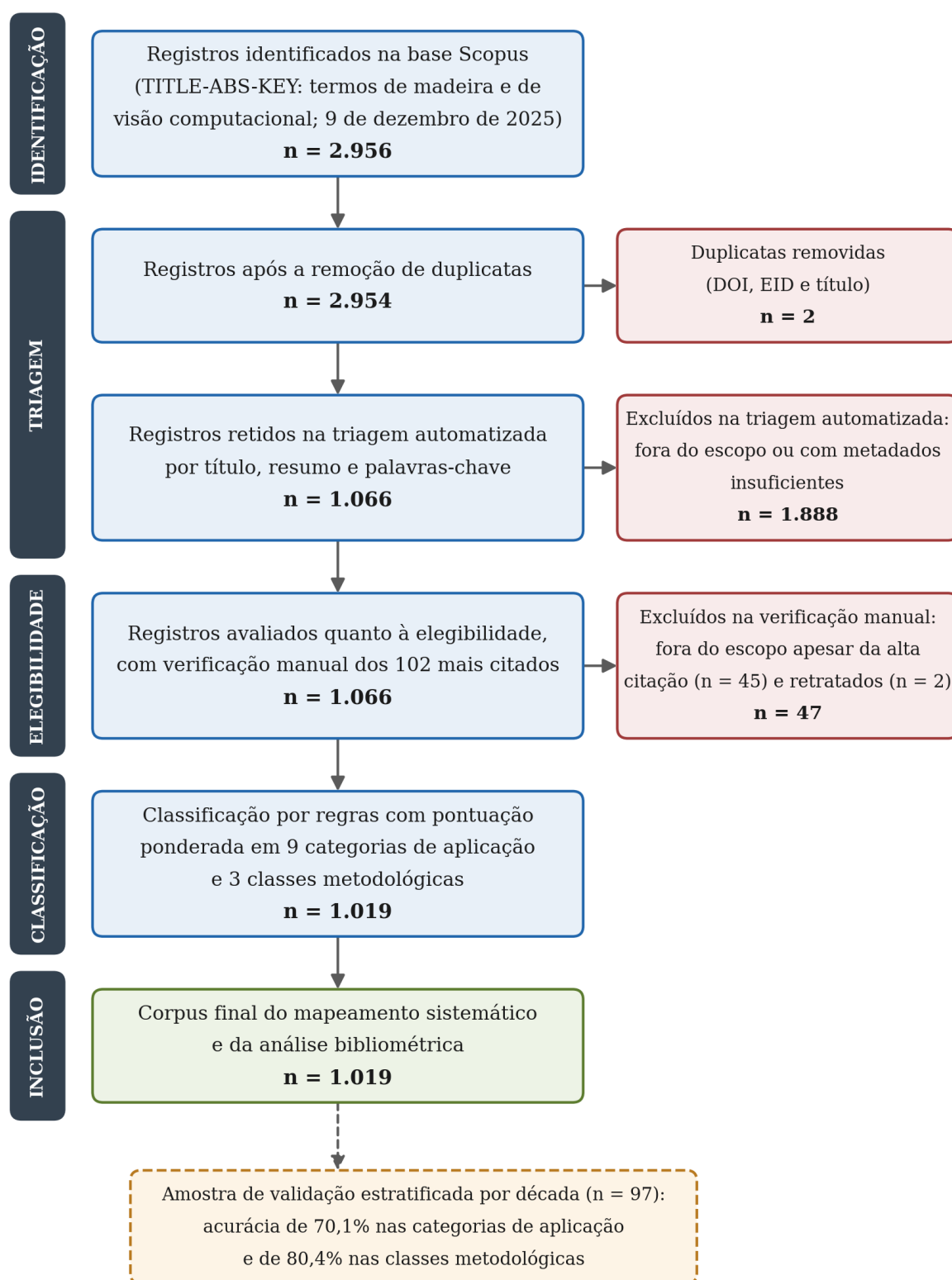


Figura 2.1. Fluxo PRISMA de seleção dos estudos, da recuperação inicial na base Scopus ao corpus final.
Fonte: Elaboração própria.

Foram retidos os registros que evidenciassem, simultaneamente, três condições: emprego de análise baseada em imagens; objeto de estudo constituído por madeira ou produto derivado,

incluindo toras, madeira serrada, painéis e microestrutura relevante ao desempenho do material; e tratamento de tarefa pertinente, como detecção de defeitos, classificação de qualidade, otimização de desdobro, medição dimensional, monitoramento de processo ou predição de propriedades.

Foram excluídos os registros que não apresentassem conteúdo identificável de visão computacional ou de análise de imagens; que se restringissem a sensoriamento remoto ou a ecologia florestal sem contexto de processamento industrial; que tratassem de imagens médicas sem relação com madeira; que descrevessem métodos genéricos de visão computacional ou de computação gráfica sem aplicação ao material; que carecessem de metadados essenciais, como autoria, título ou ano de publicação; ou cuja atribuição a uma categoria de aplicação permanecesse ambígua. Entendeu-se por ambiguidade a situação em que título, resumo e palavras-chave não forneciam informação suficiente para atribuição confiante a uma única categoria predominante.

2.2.3. Pré-processamento textual e filtragem por regras

Para cada registro foi construído um campo textual combinado, reunindo título, resumo, palavras-chave do autor e palavras-chave de indexação, convertido integralmente para caixa baixa. Sobre esse campo operaram os filtros de elegibilidade e, em seguida, os classificadores taxonômicos, de modo que todas as decisões automáticas se apoiassem na mesma representação textual.

A opção por classificação baseada em regras, e não por aprendizado supervisionado, decorreu de duas considerações. A primeira é a auditabilidade: cada atribuição pode ser rastreada até os termos que a produziram, o que é desejável em um mapeamento cuja finalidade é descritiva. A segunda é a ausência de um conjunto rotulado de tamanho suficiente para treinar um classificador confiável no domínio específico, situação em que regras calibradas por conhecimento do domínio tendem a apresentar desempenho competitivo.

2.2.4. Taxonomia de aplicações e procedimento de classificação

Cada artigo elegível foi atribuído a uma única categoria de aplicação, correspondente à tarefa predominante declarada no título e no resumo, entre nove possibilidades: controle de qualidade de superfície da madeira; imageamento tridimensional e interno; identificação de espécie e origem; produção de painéis e chapas; processamento de toras e madeira roliça; automação e robótica em serrarias; microestrutura e caracterização da madeira; predição de propriedades da madeira; e medição dimensional e geometria.

A atribuição operou por pontuação ponderada. Termos de alta especificidade, como “predição de propriedades”, “classificação de toras” ou “tomografia computadorizada”, contribuíram com escores positivos elevados, entre três e quatro pontos, ao passo que termos de menor especificidade contribuíram com um a dois pontos. Regras de exclusão aplicaram penalidades de um a dois pontos negativos quando termos conflitantes coocorriam: expressões

relativas a defeitos superficiais penalizaram a categoria de predição de propriedades, e termos relativos a toras em contexto exclusivamente superficial penalizaram a categoria de imageamento interno. Os escores foram restritos a valores não negativos.

Selecionou-se a categoria de maior escore. Nos casos de empate, aplicou-se uma ordem de prioridade previamente definida, na seguinte sequência: predição de propriedades; processamento de toras e madeira roliça; produção de painéis e chapas; controle de qualidade de superfície; identificação de espécie e origem; automação e robótica em serrarias; microestrutura e caracterização; medição dimensional e geometria; e imageamento tridimensional e interno. Os pesos e a ordem de prioridade foram estabelecidos a partir do conhecimento do domínio, calibrados em um subconjunto representativo de artigos e refinados iterativamente ao longo do processo.

Quando o foco predominante não pôde ser determinado com confiança razoável a partir da informação bibliográfica, o artigo foi excluído das análises quantitativas, em vez de atribuído arbitrariamente ou contabilizado em mais de uma categoria. Subcategorias, como a distinção entre nós e manchas no controle de superfície, ou entre compensado, aglomerado e painel de fibras na produção de painéis, foram empregadas apenas para conferir granularidade adicional dentro de cada categoria primária, sem duplicar a contagem de artigos.

2.2.5. Taxonomia metodológica

Quanto à abordagem metodológica, os artigos foram distribuídos em três classes mutuamente exclusivas, atribuídas por ordem de precedência. Classificaram-se como **aprendizado profundo** os trabalhos cujo texto indicasse o emprego de redes convolucionais, redes totalmente convolucionais, arquiteturas do tipo U-Net, detectores das famílias R-CNN e YOLO, transformadores, redes adversárias generativas ou autocodificadores. Na ausência desses termos, classificaram-se como **aprendizado de máquina clássico** os trabalhos que empregassem máquinas de vetores de suporte, árvores de decisão, florestas aleatórias, vizinhos mais próximos, regressão logística, análise discriminante linear, redes neurais rasas ou técnicas de agrupamento, tomando descritores extraídos manualmente como entrada. Os demais foram classificados como **visão computacional clássica**, categoria que reúne as técnicas determinísticas de processamento de imagens, como limiarização, morfologia matemática, filtragem, transformadas e extração manual de descritores geométricos, de textura e de cor.

Cabe registrar que essa taxonomia é metodológica, e não cronológica: trabalhos recentes podem ser corretamente classificados como visão computacional clássica quando a tarefa exige mensuração determinística, assim como redes neurais rasas empregadas nos anos 1990 (RUMELHART et al., 1986; LECUN et al., 1998) são classificadas como aprendizado de máquina clássico.

2.2.6. Aferição da confiabilidade da classificação

Para avaliar a confiabilidade do procedimento, extraiu-se uma amostra do corpus, classificada de forma independente por dois caminhos: pelo sistema automatizado de regras e por rotulagem manual conduzida segundo as mesmas definições taxonômicas. A concordância foi quantificada pela acurácia global e pelas medidas F1 em suas versões macro e ponderada, calculadas separadamente para as categorias de aplicação e para as classes metodológicas.

A medida F1 de cada classe i corresponde à média harmônica entre precisão e revocação, conforme a Equação 2.1:

$$F1_i = \frac{2 \times P_i \times R_i}{P_i + R_i} \quad (2.1)$$

Onde P_i é a precisão da classe i , isto é, a proporção de artigos atribuídos a essa classe pelo sistema de regras que a rotulagem manual confirmou; e R_i é a revocação, ou seja, a proporção de artigos que a rotulagem manual atribuiu à classe i e que o sistema de regras identificou corretamente.

A medida macro corresponde à média aritmética simples das medidas F1 das classes, atribuindo peso idêntico a categorias frequentes e raras (Equação 2.2), ao passo que a medida ponderada pondera cada classe pelo seu número de casos (Equação 2.3):

$$F1_{\text{macro}} = \frac{1}{k} \sum_{i=1}^k F1_i \quad (2.2)$$

$$F1_{\text{ponderada}} = \frac{\sum_{i=1}^k n_i F1_i}{\sum_{i=1}^k n_i} \quad (2.3)$$

Onde k é o número de classes e n_i o número de artigos da classe i segundo a rotulagem manual.

A amostra de validação excluiu deliberadamente os artigos de maior citação, já verificados individualmente, de modo a evitar que os casos mais conhecidos e mais facilmente classificáveis inflassem artificialmente a concordância observada. Selecionou-se uma amostra de 97 artigos, com estratificação por década, de modo a garantir cobertura de todo o intervalo analisado, excluídos os trabalhos previamente verificados um a um.

Para as classes metodológicas, o sistema de regras concordou com a rotulagem manual em 78 dos 97 casos, o que corresponde à acurácia de 80,4%, com F1 macro de 70,8% e F1 ponderada de 79,4%. O desempenho por classe foi de 100,0% na visão computacional clássica ($F1 = 0,835$), 65,1% no aprendizado profundo ($F1 = 0,789$) e 33,3% no aprendizado de máquina clássico ($F1 = 0,500$).

Para as categorias de aplicação, a concordância foi de 68 casos em 97, ou 70,1% de acurácia, com F1 macro de 48,3% e F1 ponderada de 64,6%. O desempenho por categoria variou de 90,0% na microestrutura e caracterização ($F1 = 0,857$), 83,3% no imageamento tridimensional

e interno ($F1 = 0,500$), 78,3% no controle de qualidade de superfície ($F1 = 0,758$) e 72,7% na identificação de espécie e origem ($F1 = 0,800$) até 12,5% na predição de propriedades ($F1 = 0,200$) e 0,0% na medição dimensional ($F1 = 0,000$).

A discrepância entre a $F1$ macro e a $F1$ ponderada, em ambas as taxonomias, decorre do desempenho baixo nas classes de poucos casos, que a média macro pondera igualmente. Trata-se de limitação conhecida da classificação com sinal escasso. Seu efeito sobre os agregados é atenuado pelo fato de as duas categorias de pior desempenho reunirem, juntas, 51 artigos, ou 5,0% do corpus, mas as discrepâncias observadas recomendam cautela na interpretação dos agregados, particularmente nas categorias de menor representação.

2.2.7. Limitações da classificação

Três limitações devem ser registradas. A primeira decorre da restrição a uma única base bibliográfica: embora a Scopus ofereça ampla cobertura, contribuições publicadas exclusivamente em veículos não indexados, em relatórios técnicos ou em idiomas de menor circulação internacional permanecem fora do alcance da busca.

A segunda decorre da atribuição de categoria única. Trabalhos que combinam, por exemplo, medição dimensional e classificação de qualidade são forçosamente alocados à tarefa predominante, o que subestima a natureza multifuncional de parte da literatura recente, na qual sistemas integrados executam simultaneamente detecção, mensuração e classificação.

A terceira decorre da própria natureza da classificação por regras, que opera sobre a informação bibliográfica e não sobre o texto completo. Artigos cujo resumo descreve o problema sem explicitar o método empregado podem ser atribuídos à classe metodológica de menor especificidade, o que tende a favorecer levemente a categoria de visão computacional clássica.

2.3. PANORAMA BIBLIOMÉTRICO

2.3.1. Tendência de publicação

O corpus final reuniu 1.019 artigos publicados entre 1983 e 2026, considerados os itens em acesso antecipado. A distribuição temporal apresenta inflexão nítida a partir de 2015, de modo que a produção do campo se concentra fortemente no período recente (Figura 2.2c). O intervalo compreendido entre 1983 e 1999, aqui denominado período de fundação, reuniu 99 artigos, ou 9,7% do total. O período de crescimento, entre 2000 e 2015, somou 271 artigos, correspondentes a 26,6%. O período subsequente, iniciado em 2016 e caracterizado pela adoção generalizada do aprendizado profundo, concentrou 649 artigos, ou 63,7% de toda a produção do campo em pouco mais de um terço do intervalo analisado.

Essa aceleração não decorre exclusivamente do crescimento geral da produção científica. Ela coincide com três condições que se estabeleceram simultaneamente: a disponibilidade de bibliotecas de aprendizado profundo de uso aberto, a redução do custo de processadores gráficos

e a difusão de câmeras digitais de alta resolução em ambiente industrial, o que reduziu substancialmente a barreira de entrada para grupos de pesquisa sem infraestrutura especializada.

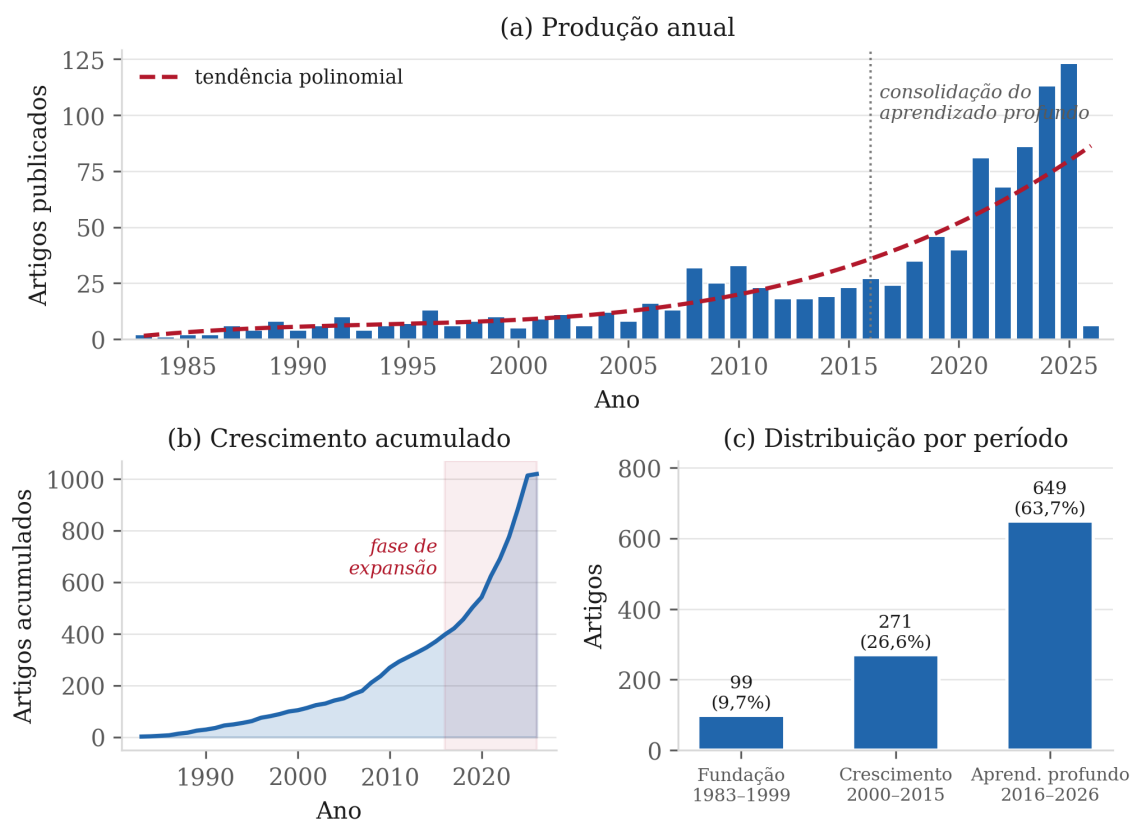


Figura 2.2. Produção anual (a), crescimento acumulado (b) e distribuição dos 1.019 artigos pelos três períodos considerados (c). Fonte: Elaboração própria, a partir do corpus de 1.019 artigos.

2.3.2. Tipos documentais e veículos de publicação

A composição do corpus por tipo documental reflete a natureza aplicada do campo. Artigos de periódico revisados por pares constituem a maioria, mas a participação de trabalhos completos de congresso é expressiva, sobretudo no período de fundação, quando os principais avanços eram divulgados em conferências de instrumentação e de visão computacional (FUNT; BRYANT, 1985; FUNT; BRYANT, 1987; BRODIE et al., 1994; SCHMOLDT et al., 1997).

A composição do corpus por tipo documental está representada na Figura 2.3a. Artigos de periódico revisados por pares somam 623 registros, ou 61,1% do total, e trabalhos completos de congresso, 359, ou 35,2%. Artigos de revisão respondem por 20 registros, ou 2,0%, e capítulos de livro, por nove, ou 0,9%. Os oito registros restantes, ou 0,8%, distribuem-se entre notas, artigos de dados, revisões curtas e itens sem tipo documental atribuído na base.

A participação de um terço de trabalhos de congresso é elevada para um campo aplicado e tem duas leituras. A primeira é de ritmo: em domínios de forte componente de engenharia, o congresso é o canal em que o resultado chega primeiro, sobretudo em fases de rápida incorporação metodológica. A segunda é de rastreabilidade: trabalhos de congresso costumam trazer

menos detalhamento de protocolo e de conjunto de dados, o que dificulta a reprodução e ajuda a explicar a heterogeneidade de relato discutida adiante.

Os veículos de publicação, apresentados na Figura 2.3b, distribuem-se por 336 títulos distintos, o que evidencia dispersão acentuada. O veículo mais frequente é a série *Proceedings of SPIE*, com 41 artigos, seguida por *Forests*, com 29, *Forest Products Journal*, com 24, *BioResources*, com 22, e *Wood Science and Technology* e *Journal of Forestry Engineering*, com 21 cada. Nenhum título concentra mais de 4% do corpus.

Essa dispersão distribui-se entre três famílias de veículos: periódicos de ciência e tecnologia da madeira, periódicos de engenharia e automação industrial e veículos de visão computacional e processamento de sinais. A tripartição é, em si, um achado relevante: indica que a comunidade não possui um veículo de referência único, o que ajuda a explicar a fragmentação da literatura e a baixa taxa de citação cruzada entre subdomínios.

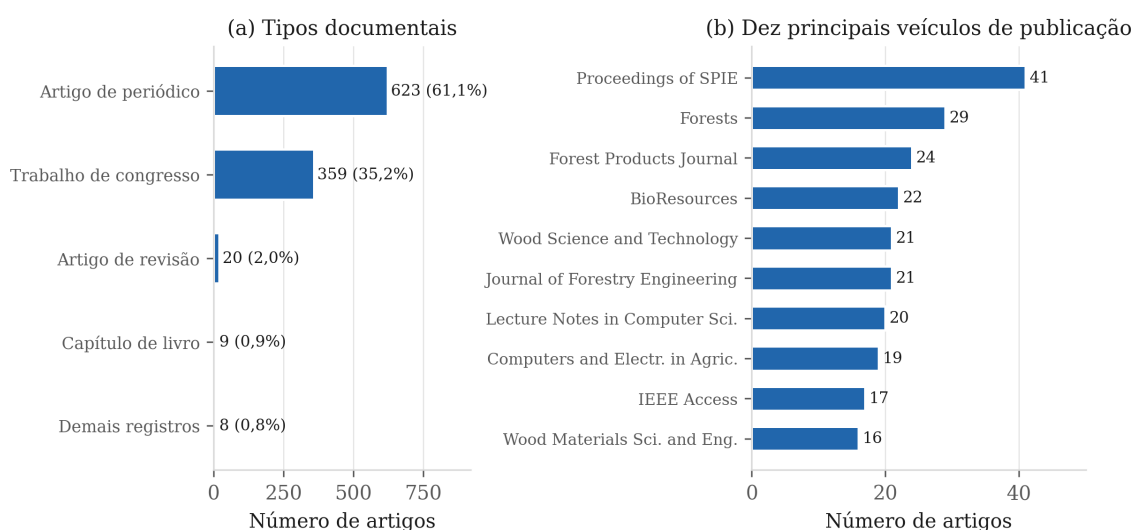


Figura 2.3. Distribuição dos artigos por tipo documental (a) e dez principais veículos de publicação (b). A barra *Demais registros* reúne notas, artigos de dados e revisões curtas, com dois registros cada, além de dois itens sem tipo documental atribuído na base. Fonte: Elaboração própria, a partir do corpus de 1.019 artigos.

2.3.3. Distribuição geográfica

Considerando os 707 registros com país do autor correspondente resolvido, a China lidera de forma destacada, com 283 artigos, seguida pelos Estados Unidos, com 45, pela Suécia, com 35, pela França, com 30, pelo Japão, com 28, pelo Canadá, com 20, pela Alemanha e pela Finlândia, com 18 cada, pelo Brasil, com 17, e pela Turquia, com 16 (Figura 2.4).

A concentração chinesa, que responde por cerca de 40% dos registros com país identificado, é fenômeno recente e concentrado no período posterior a 2016, associado à expansão da indústria de painéis e à política de fomento à automação industrial. Já a presença de países nórdicos, com Suécia e Finlândia entre os dez primeiros, reflete tradição consolidada em automação de serrarias e em mensuração de toras, com contribuições que remontam ao período de crescimento

(WIMMER et al., 2021; LI; LIDESKOG, 2021; NORELL; BORGEFORS, 2008; WIBERG; MOREN, 1999).

A posição do Brasil, com 17 artigos, merece registro. Trata-se de participação modesta diante da relevância do país na produção mundial de madeira proveniente de florestas plantadas, o que sugere a existência de espaço para contribuições nacionais, particularmente em temas ligados à mensuração e ao inventário, nos quais o setor brasileiro acumula tradição metodológica (CARMO et al., 2019; LOPES et al., 2020; CASAS et al., 2023).

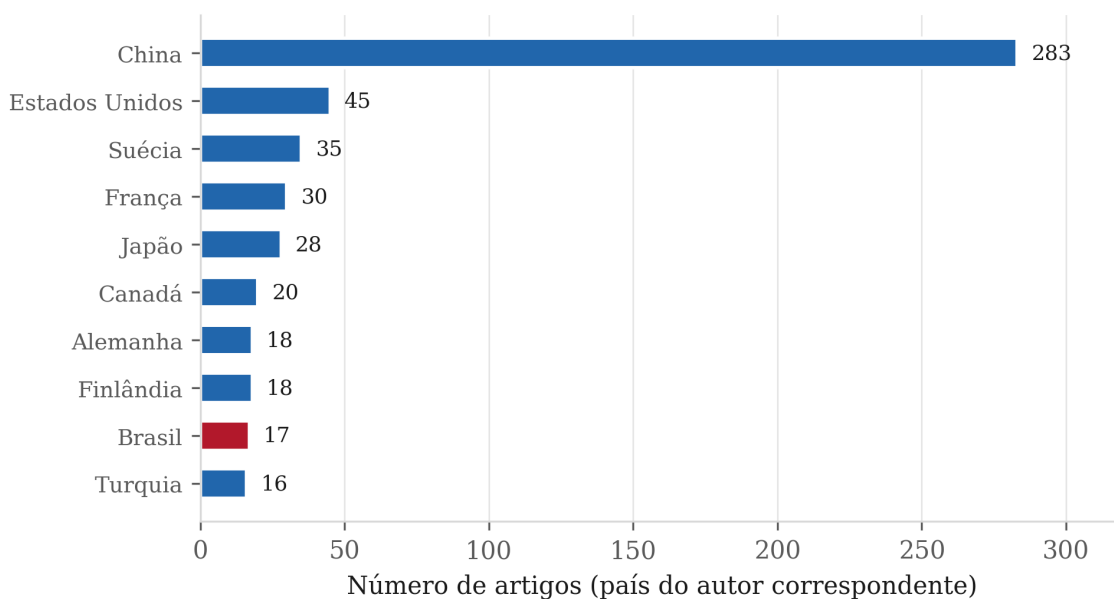


Figura 2.4. Distribuição geográfica dos dez países com maior número de artigos, segundo o país do autor correspondente. Fonte: Elaboração própria, a partir do corpus de 1.019 artigos.

2.3.4. Análise de citações

O corpus acumulou 11.961 citações, com média de 11,74 e mediana de 4 citações por artigo. A discrepância entre média e mediana indica distribuição fortemente assimétrica à direita, na qual um número reduzido de trabalhos concentra parcela expressiva do impacto, enquanto a maior parte da produção apresenta repercussão bibliométrica modesta. Esse padrão é usual em campos aplicados com forte componente de engenharia, nos quais muitos trabalhos descrevem soluções específicas de interesse local.

Entre os trabalhos de maior impacto figuram tanto contribuições fundacionais, que estabeleceram paradigmas metodológicos (CONNERS et al., 1983; ZHU et al., 1996; NYSTRÖM; KLINE, 2000), quanto aplicações recentes de aprendizado profundo que alcançaram ampla difusão (HE et al., 2020b; GAO et al., 2021; URBONAS et al., 2019; RAVINDRAN et al., 2018). A coexistência de ambos os perfis no topo da distribuição de citações reforça a leitura de um campo que preserva suas referências históricas ao mesmo tempo em que incorpora rapidamente novos métodos.

A distribuição das citações entre categorias e classes metodológicas está sintetizada na

Figura 2.5. O impacto médio mais elevado cabe à predição de propriedades da madeira, com 16,4 citações por artigo, seguida pela identificação de espécie e origem, com 15,9, e pela microestrutura e caracterização, com 15,2 (Figura 2.5a). O controle de qualidade de superfície, apesar de concentrar quase metade da produção, situa-se em 12,1, e a automação e robótica em serrarias apresenta o menor valor, 4,1.

Esse ordenamento é informativo: as categorias de maior impacto médio não são as de maior volume, e sim aquelas cujos resultados interessam simultaneamente à ciência da madeira e a domínios vizinhos, como a anatomia vegetal e a caracterização de materiais. As categorias mais próximas do chão de fábrica, ao contrário, produzem trabalhos de interesse operacional direto, cuja repercussão bibliométrica é menor.

A dispersão por classe metodológica (Figura 2.5b) mostra que o aprendizado de máquina clássico apresenta a mediana mais alta, o que se explica pela idade dos trabalhos: publicados em sua maioria antes de 2016, tiveram mais tempo para acumular citações. O aprendizado profundo, concentrado no período recente, apresenta mediana comparável à da visão computacional clássica, apesar de sua visibilidade. A mediana de citações por ano de publicação (Figura 2.5c) confirma esse efeito de maturação, com declínio sistemático a partir de 2020 que reflete a janela de acumulação, e não perda de relevância.

O histograma (Figura 2.5d) explicita a assimetria: mais da metade dos artigos acumula quatro citações ou menos, enquanto a cauda direita se estende até 209 citações no trabalho mais citado do corpus.

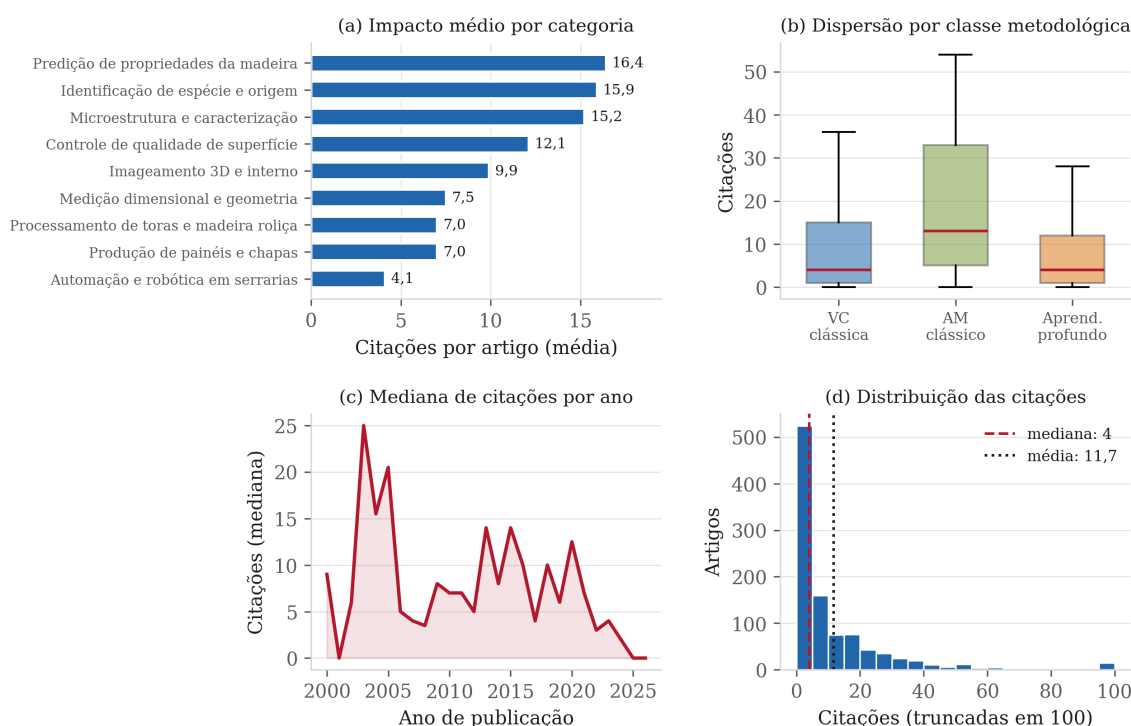


Figura 2.5. Impacto médio por categoria (a), dispersão por classe metodológica (b), mediana de citações por ano de publicação (c) e distribuição das citações (d). Fonte: Elaboração própria, a partir do corpus de 1.019 artigos.

2.3.5. Rede de coocorrência de palavras-chave

A análise de coocorrência de palavras-chave, apresentada na Figura 2.6, considerou os 30 termos mais frequentes do corpus, após reunião de variantes equivalentes, e as ligações cuja coocorrência atingiu ao menos quinze artigos, o que resultou em 141 ligações. A detecção de comunidades pelo critério de modularidade identificou quatro agrupamentos.

O primeiro e mais numeroso reúne os termos de processamento e análise de imagens, com onze integrantes, entre os quais madeira, processamento de imagens, segmentação, análise de imagens, classificação, extração de atributos, textura e tomografia computadorizada. O segundo agrupa aprendizado profundo, aprendizado de máquina, redes convolucionais, convolução, madeira serrada e setor florestal. O terceiro reúne os termos de inspeção industrial e automação, entre eles visão computacional, produtos de madeira, visão de máquina, tábuas, controle de qualidade, inspeção, automação e identificação da madeira. O quarto, o mais compacto, concentra os termos relativos a defeitos e à sua detecção.

Três leituras decorrem dessa estrutura. A primeira é a centralidade do vocabulário metodológico: as ligações mais fortes envolvem madeira e processamento de imagens, com 142 coocorrências, madeira e análise de imagens, com 104, e análise de imagens e processamento de imagens, com 90. São os termos de método, e não os de aplicação, que conectam os agrupamentos entre si.

A segunda é a separação nítida entre o agrupamento clássico e o de aprendizado profundo, que se ligam sobretudo por meio do termo madeira, e não diretamente entre si. Isso confirma, no plano do vocabulário, a divisão de trabalho identificada na distribuição metodológica.

A terceira é a ausência, entre os trinta termos mais frequentes, de qualquer palavra-chave associada à mensuração volumétrica, à contagem de peças ou ao empilhamento. Trata-se de evidência independente, obtida do vocabulário e não da classificação por categorias, de que essas tarefas permanecem marginais no campo.

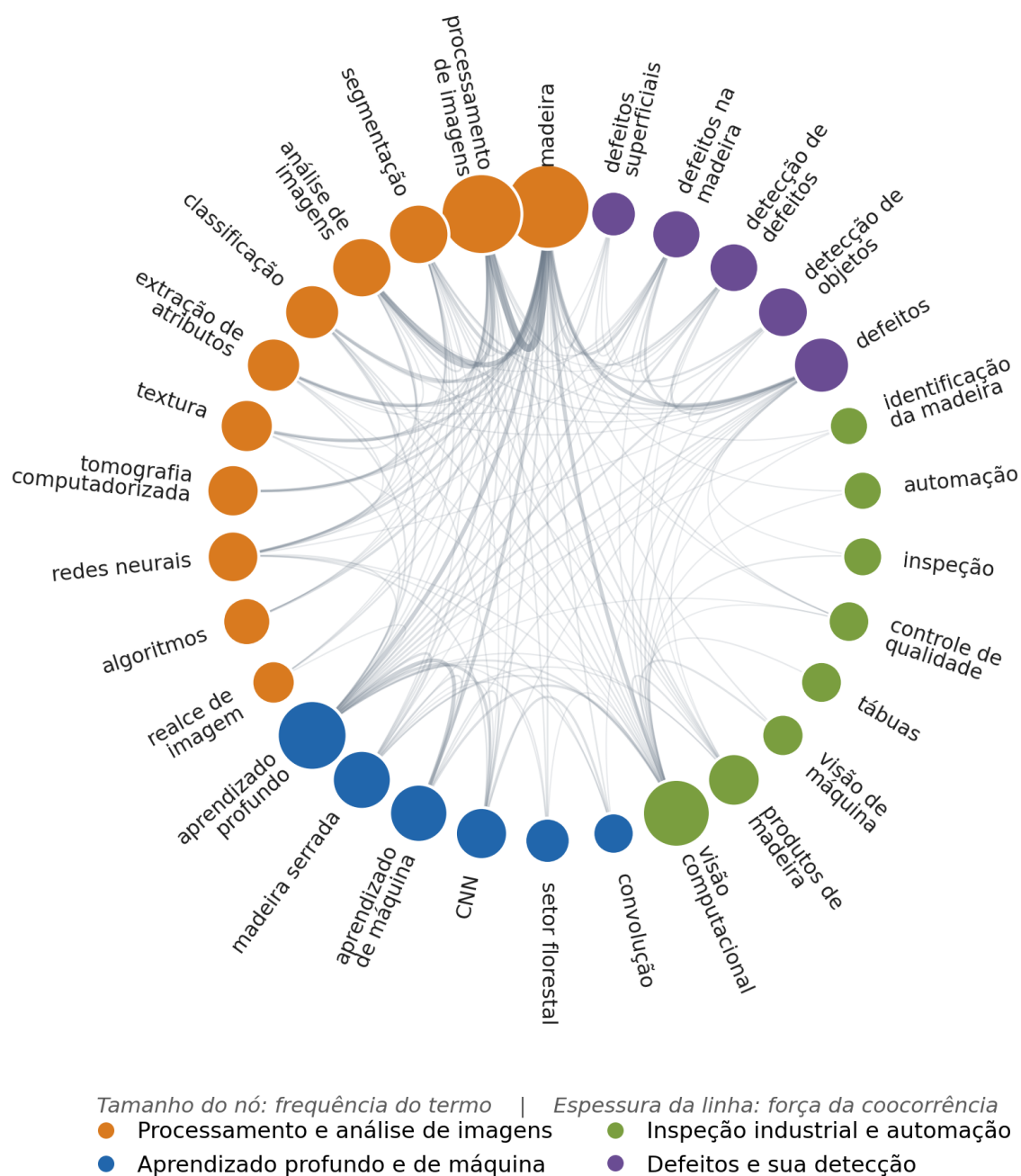


Figura 2.6. Rede de coocorrência dos trinta termos mais frequentes. O tamanho do nó indica a frequência do termo; a espessura da linha, a força da coocorrência; e a cor, o agrupamento temático detectado. Fonte: Elaboração própria, a partir do corpus de 1.019 artigos.

2.4. CATEGORIAS DE APLICAÇÃO

A distribuição dos 1.019 artigos entre as nove categorias de aplicação é fortemente desigual (Figura 2.7 e Tabela 2.1). As seções seguintes descrevem cada categoria, sua composição interna e o perfil metodológico predominante.

Tabela 2.1. Distribuição dos artigos entre as categorias de aplicação e participação da visão computacional clássica em cada uma.

Categoria de aplicação	Artigos	Participação (%)	VC clássica (%)
Controle de qualidade de superfície	492	48,3	64,6
Imageamento tridimensional e interno	139	13,6	83,5
Identificação de espécie e origem	108	10,6	53,7
Microestrutura e caracterização	103	10,1	81,6
Produção de painéis e chapas	65	6,4	83,1
Processamento de toras e madeira roliça	39	3,8	69,2
Medição dimensional e geometria	33	3,2	90,9
Automação e robótica em serrarias	22	2,2	72,7
Predição de propriedades da madeira	18	1,8	61,1
Total	1.019	100,0	70,1

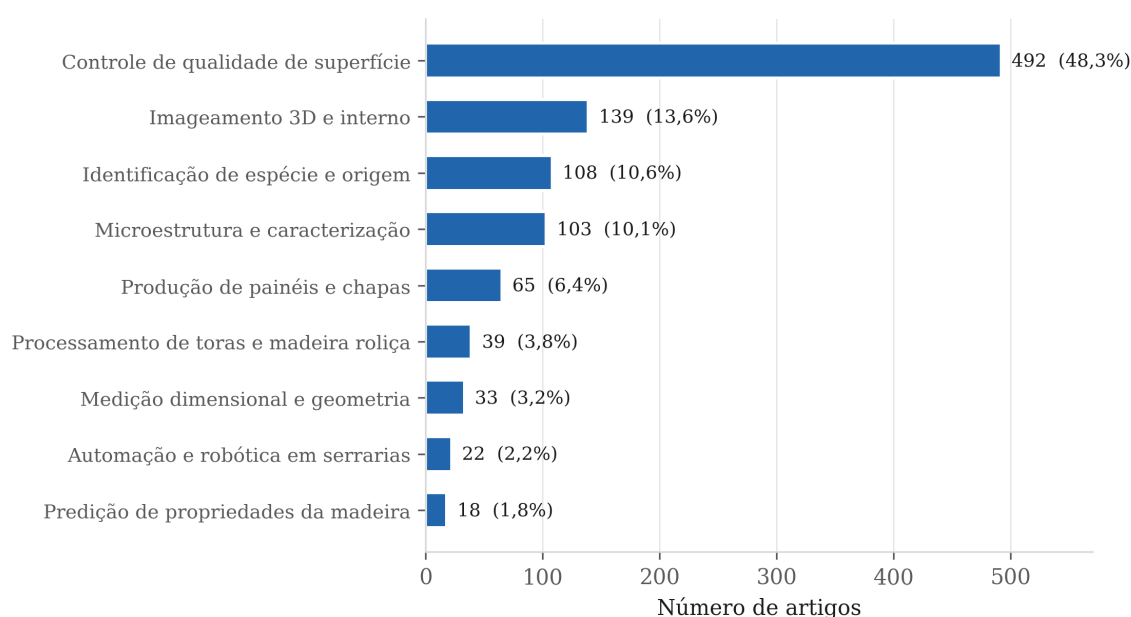


Figura 2.7. Distribuição dos 1.019 artigos entre as nove categorias de aplicação. Fonte: Elaboração própria, a partir do corpus de 1.019 artigos.

2.4.1. Controle de qualidade de superfície

Com 492 artigos, ou 48,3% do corpus, a inspeção de superfícies constitui o principal vetor de adoção industrial da visão computacional no setor. Mais de duzentos trabalhos tratam de defeitos superficiais mistos ou gerais, 111 concentram-se em manchas e alterações de coloração e 86 dedicam-se especificamente à detecção e à classificação de nós. A visão computacional clássica responde por 64,6% da categoria, participação inferior à média do corpus, o que evidencia

penetração relativamente maior do aprendizado profundo neste domínio.

A explicação para essa concentração é econômica antes de ser técnica: a classificação de tábuas incide diretamente sobre o valor comercial de cada peça processada, e o ganho marginal de uma classificação mais acurada é imediatamente mensurável. Do ponto de vista metodológico, trata-se de tarefa de natureza classificatória, com elevada variabilidade de aparência e difícil formulação por descritores manuais, condição em que o aprendizado profundo apresenta vantagem clara (HE et al., 2020b; GAO et al., 2021; YANG et al., 2019; HE et al., 2019; TU et al., 2021; KURDTHONGMEE; SUWANNARAT, 2019). Abordagens clássicas permanecem, contudo, em uso corrente, sobretudo em sistemas em linha com restrição severa de tempo de resposta (KAMAL et al., 2017; GU et al., 2010; YANG et al., 2021; SAMDANGDECH; PHIPHOBMONGKOL, 2018; HENKE et al., 2015).

2.4.2. Imageamento tridimensional e interno

Com 139 artigos, ou 13,6% do corpus, esta categoria reúne os trabalhos que buscam informação além da superfície, por tomografia computadorizada, radiografia e demais modalidades de imageamento volumétrico. A visão computacional clássica domina de forma acentuada, com 83,5% dos artigos, o que reflete a natureza da tarefa: a reconstrução tomográfica e a segmentação de estruturas internas apoiam-se em modelos físicos de atenuação e em algoritmos geométricos cujo comportamento é previsível e auditável.

Os trabalhos pioneiros da categoria remontam ao período de fundação (FUNT; BRYANT, 1985; FUNT; BRYANT, 1987; ZHU et al., 1996; WIBERG; MOREN, 1999; BHANDARKAR et al., 1998), e a linha manteve continuidade nas décadas seguintes, com aplicações voltadas à detecção de defeitos internos e à otimização do desdobro (BHANDARKAR et al., 2008; VAN DEN BULCKE et al., 2008; DEROME et al., 2011; SANDBERG; SALIN, 2012; WATANABE et al., 2012; SEDIGHI-GILANI et al., 2014; VAN DEN BULCKE et al., 2019). Contribuições recentes incorporam aprendizado profundo à segmentação volumétrica e à correção de artefatos (WEI et al., 2011; YE et al., 2022; XIE et al., 2024; FORSBERG et al., 2010; MARTINEZ-GARCIA et al., 2021).

2.4.3. Identificação de espécie e origem

Com 108 artigos, ou 10,6% do corpus, esta é a categoria mais favorável ao aprendizado profundo, que responde por 35,2% dos trabalhos, participação bem superior à média do campo, de 26,4%. É também a categoria em que a visão computacional clássica tem a menor presença, 53,7%. A razão é direta: trata-se de tarefa de classificação pura, na qual o padrão discriminante é anatômico e visualmente complexo, e para a qual existem coleções de referência que viabilizam o treinamento supervisionado.

A linha compreende estudos baseados em imagens anatômicas macroscópicas e microscópicas (HWANG; SUGIYAMA, 2021a, 2021b; HWANG et al., 2020; RAVINDRAN et al., 2018;

DA SILVA et al., 2017; LENS et al., 2020; WU et al., 2021; HE et al., 2020c; FABIJANSKA et al., 2021; ZAMRI et al., 2016; IBRAHIM et al., 2017), bem como abordagens espectrais e de imageamento além do visível (STEPPE et al., 2004; HERMANSON; WIEDENHOEFT, 2011; VAN DEN BULCKE et al., 2009). O crescimento recente da categoria está diretamente associado às exigências regulatórias de rastreabilidade da origem da madeira (EUROPEAN UNION, 2023).

2.4.4. Microestrutura e caracterização

Com 103 artigos, ou 10,1% do corpus, esta categoria reúne as investigações em escala celular e subcelular, apoiadas predominantemente em microscopia e em imageamento espectral. A visão computacional clássica responde por 81,6% dos trabalhos, coerentemente com o caráter quantitativo e metrológico das medições envolvidas, que exigem rastreabilidade dimensional (FAHLEN; SALMEN, 2003; NAUMANN et al., 2005; WALTHER et al., 2006; ARNOULD; ARINERO, 2015; KIBLEUR et al., 2022; NYSTRÖM; KLINE, 2000; ZHU et al., 1996; RESENTE et al., 2021; PERRÉ et al., 2016; FABIJANSKA; DANEK, 2018; RYDVAL et al., 2024; KATZENMAIER et al., 2023).

2.4.5. Produção de painéis e chapas

Com 65 artigos, ou 6,4% do corpus, a pesquisa sobre produtos engenheirados concentra-se na inspeção de lâminas e de compensados, no monitoramento de linhas de aglomerado e de painéis de fibras e no controle de qualidade de painéis colados. A visão computacional clássica prevalece com 83,1%, reflexo da natureza repetitiva e controlada do ambiente de linha de produção, no qual a iluminação e o posicionamento são padronizados e as técnicas determinísticas alcançam desempenho satisfatório a custo computacional reduzido (CHEN et al., 2022; GE et al., 2023; JIA et al., 2023; DING et al., 2020; SANG et al., 2020; WANG et al., 2021; LIU et al., 2020; LIN et al., 2020; BARADIT et al., 2020; FENG et al., 2019; VERHEYEN et al., 2016; CASTELLINI et al., 2008; LI et al., 2020; ABDELJABER et al., 2023).

2.4.6. Processamento de toras e madeira roliça

Com apenas 39 artigos, ou 3,8% do corpus, esta categoria reúne as tarefas de pátio e de desdobro inicial: classificação e inspeção de toras, estimativa de volume e de geometria e decisões de otimização de traçamento. A visão computacional clássica responde por 69,2% dos trabalhos.

A literatura da categoria distribui-se entre a mensuração de pilhas e de carregamentos (BRODIE et al., 1994; GUTZEIT et al., 2011; SCHRAML et al., 2015; KRUGLOV, 2016; NIȚĂ; BORZ, 2023; CASAS et al., 2023; LU et al., 2024), a detecção e a segmentação de faces de toras (NORELL; BORGEFORS, 2008; RINNHOFFER et al., 2003; WU; KILIAN, 2018) e a manipulação robotizada em operações florestais (WIMMER et al., 2021; LI; LIDESKOG,

2021; FORTIN et al., 2022; ANDERSSON et al., 2021). A modéstia numérica desta categoria, contrastada com sua relevância operacional, constitui o achado de maior consequência para esta tese e é retomada na discussão.

2.4.7. Medição dimensional e geometria

Com 33 artigos, ou 3,2% do corpus, esta categoria abrange a metrologia geométrica em sentido estrito: medição de dimensões, detecção de empenamentos e torções e caracterização de forma. A visão computacional clássica domina de maneira quase absoluta, com 90,9% dos artigos, a maior participação entre todas as categorias.

Esse é o caso em que a divisão de trabalho entre abordagens se manifesta com maior nitidez. Quando o produto da análise é uma grandeza física destinada a compor um laudo ou uma transação comercial, exige-se que o procedimento de cálculo seja explícito e que a incerteza seja rastreável, condições que os métodos geométricos determinísticos atendem e que as arquiteturas de aprendizado profundo, em sua formulação usual, não atendem (DAI; LI, 2022; MOSKALIK et al., 2022; SPREITZER et al., 2020; SHI et al., 2019; KRÄHENBÜHL et al., 2013; COUDRAY et al., 2009; MARICHAL et al., 2025).

2.4.8. Automação e robótica em serrarias

Com 22 artigos, ou 2,2% do corpus, esta é a segunda menor categoria. Reúne o controle geral de serrarias, a manipulação robotizada e o monitoramento de processo em linha, com 72,7% de participação da visão computacional clássica (XIONG; WANG, 2025; GULIRMAK; BINGÜL, 2023; AGUILERA et al., 2023; GUO; LEE, 2021; SETTIMI et al., 2025; PIERDICCA et al., 2022).

A baixa produção acadêmica nesta categoria não deve ser interpretada como baixa adoção industrial. É plausível que parte expressiva do desenvolvimento ocorra em ambiente corporativo e permaneça não publicada, situação comum em tecnologias de processo com valor competitivo direto.

2.4.9. Predição de propriedades da madeira

Com 18 artigos, ou 1,8% do corpus, esta é a menor categoria. Reúne as tentativas de estimar propriedades mecânicas, físicas ou químicas a partir de imagens, sem ensaio destrutivo. A visão computacional clássica responde por 61,1% dos trabalhos, a segunda menor participação entre todas as categorias, o que sugere abertura relativa ao aprendizado profundo (MOHAMED et al., 2021; RAHMAN et al., 2025; CHEN et al., 2023; JIANG et al., 2021; CHOLEWA et al., 2016; HIETANIEMI et al., 2014; CRUZ; SAPORITI MACHADO, 2013; GARCIA et al., 2023; SERRANO; ENQUIST, 2005; CHIDA; TAKAHASHI, 2021; HE et al., 2021; LIU et al., 2010; SUGIARTO et al., 2017).

A dimensão reduzida da categoria reflete uma dificuldade intrínseca: propriedades mecânicas dependem de características internas que a imagem de superfície não observa diretamente, de modo que a predição repousa sobre correlações indiretas, cuja generalização entre espécies e condições de crescimento permanece limitada.

2.5. EVOLUÇÃO METODOLÓGICA

2.5.1. Distribuição das classes metodológicas

A distribuição metodológica revela que, apesar da visibilidade do aprendizado profundo, as abordagens clássicas ainda respondem pela maior parte da produção (Figura 2.8). A visão computacional clássica reuniu 714 artigos, ou 70,1% do corpus, e o aprendizado de máquina clássico, 36 artigos, ou 3,5%, de modo que as abordagens clássicas somam 73,6%. O aprendizado profundo respondeu por 269 artigos, correspondentes a 26,4%.

Essa predominância não deve ser interpretada como atraso metodológico do campo, e sim como resultado de dois fatores. O primeiro é acumulativo: quatro décadas de produção incluem um extenso período anterior à viabilização prática do aprendizado profundo, e os trabalhos daquele intervalo permanecem no corpus. O segundo é funcional: tarefas de mensuração determinística continuam a ser resolvidas por métodos geométricos explícitos, cuja saída é auditável e cuja incerteza pode ser propagada analiticamente.

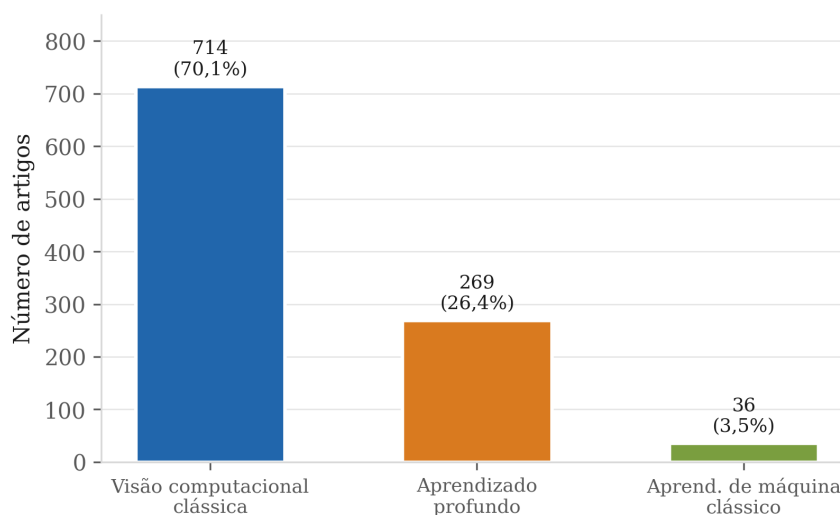


Figura 2.8. Distribuição dos 1.019 artigos entre as três classes metodológicas. Fonte: Elaboração própria, a partir do corpus de 1.019 artigos.

2.5.2. Perfil metodológico por categoria de aplicação

O cruzamento entre classes metodológicas e categorias de aplicação, apresentado na última coluna da Tabela 2.1 e sintetizado nas Figuras 2.9 e 2.10, evidencia gradiente consistente. A participação da visão computacional clássica varia de 53,7% na identificação de espécie e origem

a 90,9% na medição dimensional, e a ordenação das categorias segundo essa participação corresponde, aproximadamente, ao grau em que a tarefa exige mensuração determinística em vez de classificação.

Nas duas extremidades desse gradiente situam-se, de um lado, a metrologia geométrica, em que o produto da análise é uma dimensão física, e, de outro, a identificação de espécies e a predição de propriedades, em que o produto é uma classe ou um valor estimado por correlação. Trata-se, portanto, de divisão de trabalho fundada na natureza do problema, e não na maturidade relativa das ferramentas.

O cruzamento entre as duas taxonomias, apresentado na Figura 2.9, permite localizar onde cada abordagem se concentra. O controle de qualidade de superfície reúne, isoladamente, 318 artigos de visão computacional clássica e 157 de aprendizado profundo, o que corresponde a 58,4% de todo o aprendizado profundo do corpus. O imageamento tridimensional e interno concentra 116 artigos clássicos contra 22 de aprendizado profundo, e a microestrutura e caracterização, 84 contra 18.

Nenhuma categoria é dominada pelo aprendizado profundo em termos absolutos. A leitura por proporções, apresentada na Figura 2.10, revela porém o gradiente descrito a seguir.

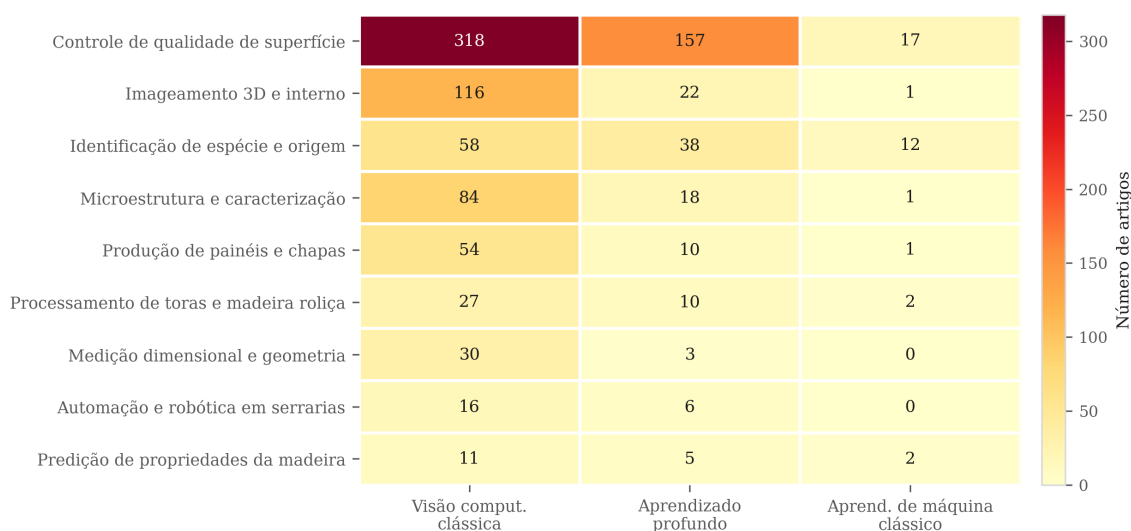


Figura 2.9. Distribuição das três classes metodológicas entre as nove categorias de aplicação. Fonte: Elaboração própria, a partir do corpus de 1.019 artigos.

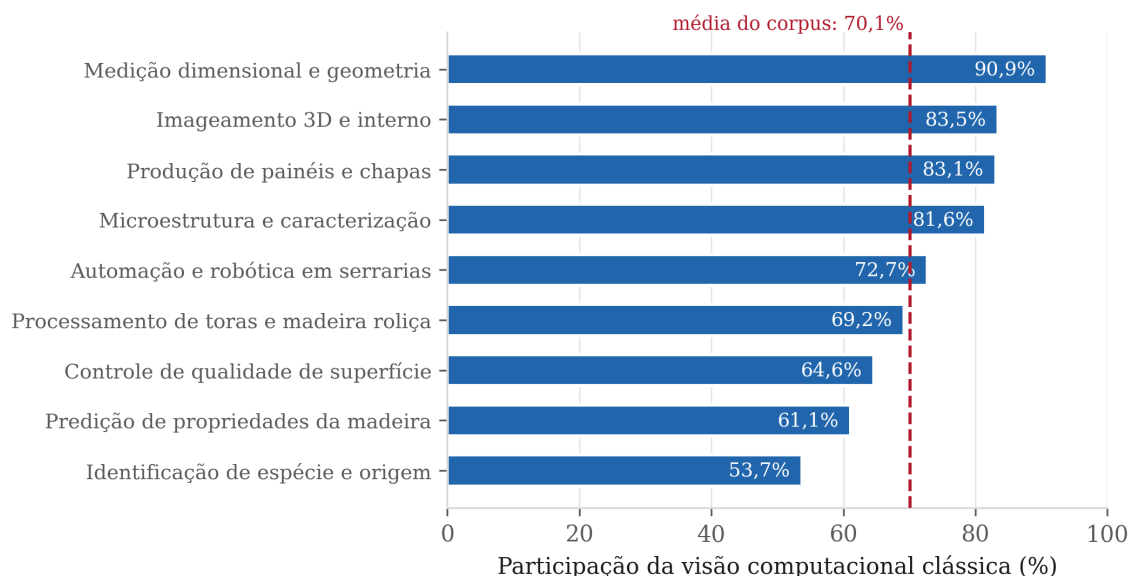


Figura 2.10. Participação da visão computacional clássica em cada categoria de aplicação. Fonte: Elaboração própria, a partir do corpus de 1.019 artigos.

2.5.3. Tendências de adoção do aprendizado profundo

A adoção do aprendizado profundo apresenta trajetória acentuadamente crescente a partir de 2016, mas heterogênea entre categorias. Progride mais rapidamente nas tarefas de classificação, como a inspeção de superfícies e a identificação de espécies, e mais lentamente nas tarefas de mensuração, nas quais a exigência de rastreabilidade opera como fator de resistência.

Cabe observar que a fronteira entre as classes vem se tornando menos nítida. Sistemas recentes combinam, em um mesmo fluxo, detecção por rede neural e cálculo geométrico determinístico, empregando o aprendizado para localizar o objeto e a geometria para medi-lo. Essa arquitetura híbrida, que preserva a auditabilidade da etapa de mensuração, é precisamente a adotada nos capítulos seguintes desta tese.

A Figura 2.11 detalha essa progressão. Por categoria (Figura 2.11a), a adoção do aprendizado profundo varia de 35,2% na identificação de espécie e origem a 9,1% na medição dimensional e geometria, com média de 26,4%. Nenhuma categoria ultrapassa o limiar de metade dos trabalhos.

A evolução anual (Figura 2.11b) é mais expressiva. Até 2016 a participação do aprendizado profundo era nula no corpus; o primeiro registro é de 2017, com 4,2%; em 2018 alcança 14,3% e, em 2019, 43,5%. A partir de 2021 estabiliza-se em torno de metade da produção anual, com 46,9% em 2021, 48,5% em 2022, 52,2% em 2024 e 48,0% em 2025.

Essa estabilização em torno de 50%, e não em valores próximos da totalidade, é o achado mais relevante desta subseção. Passada a fase de incorporação inicial, o campo não substituiu as abordagens clássicas: acomodou-se em um equilíbrio no qual cada classe atende ao tipo de tarefa para o qual é adequada.

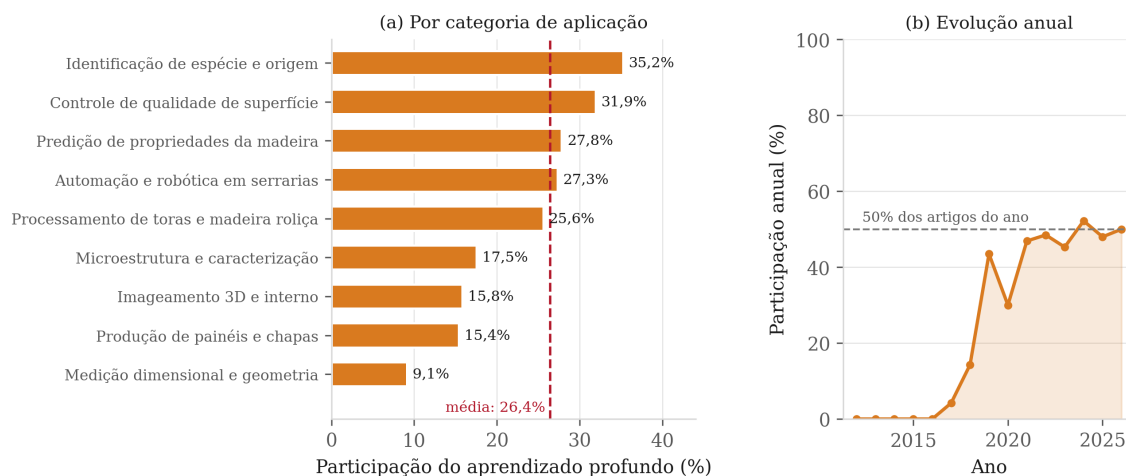


Figura 2.11. Participação do aprendizado profundo por categoria de aplicação (a) e sua evolução anual (b). Fonte: Elaboração própria, a partir do corpus de 1.019 artigos.

2.6. ANÁLISE TEMPORAL

2.6.1. Período de fundação (1983–1999)

Com 99 artigos, ou 9,7% do corpus, o período de fundação estabeleceu os paradigmas que ainda organizam o campo. O trabalho de Connors et al. (1983) inaugurou a detecção automatizada de defeitos como componente de um sistema de processamento de madeira serrada, e os anos seguintes consolidaram a tomografia de toras como linha de pesquisa autônoma (FUNT; BRYANT, 1985; FUNT; BRYANT, 1987; ZHU et al., 1996; SCHMOLDT et al., 1997; BHAN-DARKAR et al., 1998; WIBERG; MOREN, 1999; MURATA et al., 1999).

As limitações do período eram sobretudo computacionais. O custo de processamento restringia as aplicações a imagens de baixa resolução e a algoritmos de complexidade reduzida, o que orientou a pesquisa para a extração de descritores compactos e para heurísticas específicas de cada defeito.

2.6.2. Período de crescimento (2000–2015)

Com 271 artigos, ou 26,6% do corpus, este período acompanhou a difusão das câmeras digitais, o barateamento do armazenamento e o aumento da capacidade de processamento. A pesquisa diversificou-se temática e geograficamente, e o aprendizado de máquina clássico ganhou espaço, com a adoção de máquinas de vetores de suporte e de florestas aleatórias sobre descritores extraídos manualmente (NYSTRÖM; KLINE, 2000; RINNHOFFER et al., 2003; STEPPE et al., 2004; SERRANO; ENQUIST, 2005; WALTHER et al., 2006; VAN DEN BULCKE et al., 2008; CASTELLINI et al., 2008; COUDRAY et al., 2009; VAN DEN BULCKE et al., 2009; GU et al., 2010; LIU et al., 2010; FORSBERG et al., 2010; GUTZEIT et al., 2011; DEROME et al., 2011; HERMANSON; WIEDENHOEFT, 2011; SANDBERG; SALIN, 2012;

WATANABE et al., 2012; KRÄHENBÜHL et al., 2013; CRUZ; SAPORITI MACHADO, 2013; HIETANIEMI et al., 2014; SEDIGHI-GILANI et al., 2014; SCHRAML et al., 2015; HENKE et al., 2015).

2.6.3. Era do aprendizado profundo (2016–presente)

Com 649 artigos, ou 63,7% do corpus, este período responde por quase dois terços de toda a produção do campo. A transposição de arquiteturas desenvolvidas para propósitos gerais ocorreu de forma rápida e ampla, alcançando praticamente todas as categorias de aplicação (VERHEYEN et al., 2016; ZAMRI et al., 2016; CHOLEWA et al., 2016; KRUGLOV, 2016; IBRAHIM et al., 2017; KAMAL et al., 2017; DA SILVA et al., 2017; SUGIARTO et al., 2017; WU; KILIAN, 2018; RAVINDRAN et al., 2018; SAMDANGDECH; PHIPHOBMONGKOL, 2018; FABIJAŃSKA; DANEK, 2018; NORELL; BORGEFORS, 2008; FENG et al., 2019; YANG et al., 2019; HE et al., 2019; SHI et al., 2019; URBONAS et al., 2019; CARMO et al., 2019; KURDTHONGMEE; SUWANNARAT, 2019; VAN DEN BULCKE et al., 2019).

A partir de 2020, o ritmo de incorporação intensificou-se, com a difusão de detectores de estágio único, de arquiteturas de segmentação e, mais recentemente, de modelos baseados em atenção (LENS et al., 2020; HE et al., 2020a; DING et al., 2020; SANG et al., 2020; LIN et al., 2020; LIU et al., 2020; BARADIT et al., 2020; MEGID; KHAYAT, 2020; HWANG et al., 2020; SPREITZER et al., 2020; LOPES et al., 2020; LI et al., 2020; GAO et al., 2021; WU et al., 2021; TU et al., 2021; YANG et al., 2021; NGUYEN et al., 2021; GENG et al., 2021; WIMMER et al., 2021; LI; LIDESKOG, 2021; ANDERSSON et al., 2021; HWANG; SUGIYAMA, 2021b; FABIJANSKA et al., 2021; MOHAMED et al., 2021; JIANG et al., 2021; ARNOULD; ARINERO, 2015; MARTINEZ-GARCIA et al., 2021; CHIDA; TAKAHASHI, 2021; GUO; LEE, 2021; HE et al., 2021; RESENTE et al., 2021; WANG et al., 2021).

O intervalo mais recente consolida essa tendência e incorpora modelos de fundação e arquiteturas híbridas (CHEN et al., 2022; YE et al., 2022; FORTIN et al., 2022; DAI; LI, 2022; MOSKALIK et al., 2022; KIBLEUR et al., 2022; PIERDICCA et al., 2022; GE et al., 2023; JIA et al., 2023; HAN et al., 2023; ABDELJABER et al., 2023; CHEN et al., 2023; GARCIA et al., 2023; NIȚĂ; BORZ, 2023; AGUILERA et al., 2023; GULIRMAK; BINGÜL, 2023; KATZENMAIER et al., 2023; CASAS et al., 2023; XIE et al., 2024; DAS; TESFAMARIAM, 2024; LU et al., 2024; RYDVAL et al., 2024; MARICHAL et al., 2025; RAHMAN et al., 2025; XIONG; WANG, 2025; SETTIMI et al., 2025).

2.6.4. Evolução das categorias ao longo do tempo

A composição temática do corpus não permaneceu estável ao longo das quatro décadas. A Figura 2.12 apresenta a composição de cada período em termos proporcionais, o que permite comparar períodos de tamanhos muito distintos. O controle de qualidade de superfície respondia por 67,7% dos artigos no período de fundação e recuou para 48,0% no intervalo mais recente,

sem perder a posição majoritária em nenhum momento.

A identificação de espécie e origem não registra artigos antes de 2000 e alcança 13,5% no período de 2020 a 2026, o crescimento relativo mais acentuado do corpus, impulsionado pelas exigências de rastreabilidade. O imageamento tridimensional e interno atinge seu ponto mais alto entre 2010 e 2015, com 17,9%, e recua em seguida.

As categorias de mensuração seguem trajetória distinta. O processamento de toras e madeira roliça oscila entre 1,0% e 7,6% ao longo dos cinco intervalos, sem tendência de crescimento sustentado, e a medição dimensional permanece abaixo de 6% em todos eles. Sua sub-representação, portanto, não decorre de defasagem temporal que o tempo viesse a corrigir: é característica estrutural do campo, estável ao longo de quatro décadas.

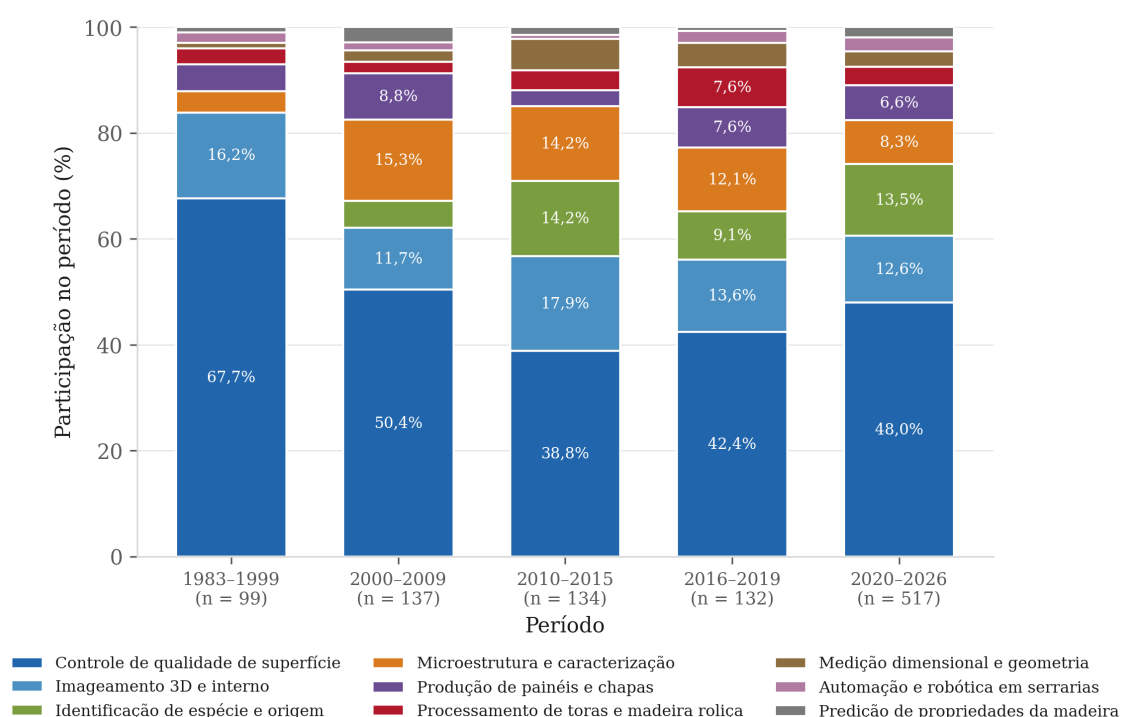


Figura 2.12. Participação de cada categoria de aplicação na produção de cada período. Fonte: Elaboração própria, a partir do corpus de 1.019 artigos.

2.7. DISCUSSÃO

O panorama obtido permite três leituras de interesse direto para esta tese.

A primeira diz respeito à divisão de trabalho entre as abordagens metodológicas. O aprendizado profundo apresenta participação relativa mais elevada nas tarefas de natureza classificatória, nas quais a variabilidade da aparência é elevada e a definição de descritores manuais é impraticável, como a detecção de defeitos superficiais e a identificação de espécies (MURATA et al., 1999; MOHAMED et al., 2021; DAI; LI, 2022; CARMO et al., 2019; LI et al., 2020). As abordagens clássicas mantêm-se nas tarefas de mensuração determinística, nas quais o que

se busca é uma grandeza física com incerteza controlada (RINNHOFFER et al., 2003). Essa distinção não é transitória: decorre da natureza dos problemas, e não da maturidade das ferramentas.

A segunda leitura refere-se à lacuna identificada nas categorias de mensuração. O processamento de toras e madeira roliça, a medição dimensional e geometria e a predição de propriedades somam, em conjunto, 90 artigos, ou 8,8% do corpus, contra 492 artigos dedicados exclusivamente ao controle de qualidade de superfície. Trata-se de desproporção expressiva quando se considera que a quantificação volumétrica da madeira empilhada é operação executada rotineiramente em todo pátio florestal e que dela dependem a comercialização e o controle de estoque (WIMMER et al., 2021; LI; LIDESKOG, 2021; FORTIN et al., 2022; ANDERSSON et al., 2021). Essa constatação, obtida de forma quantitativa e não por impressão, fundamenta a escolha temática desta tese.

A terceira leitura diz respeito ao ritmo de incorporação metodológica. A concentração de 63,7% da produção a partir de 2016 indica que o campo se encontra em fase de expansão acelerada, na qual arquiteturas desenvolvidas para propósitos gerais são rapidamente transpostas para o domínio da madeira (RAVINDRAN et al., 2018; DA SILVA et al., 2017; LENS et al., 2020; WU et al., 2021; HE et al., 2020c; FABIJANSKA et al., 2021; ZAMRI et al., 2016; IBRAHIM et al., 2017). Esse ritmo tem contrapartida: a transposição frequentemente ocorre sem adaptação às particularidades do material, como a variabilidade natural da textura, a oclusão parcial em arranjos densos e a ausência de conjuntos de dados anotados de tamanho comparável aos de domínios genéricos (URBONAS et al., 2019; HE et al., 2020b; GAO et al., 2021; HWANG; SUGIYAMA, 2021b; CASAS et al., 2023; MOHAMED et al., 2021; DING et al., 2020). Isso ajuda a explicar a distância recorrente entre os desempenhos relatados em condições controladas e os obtidos em ambiente operacional (HWANG; SUGIYAMA, 2021a; WEI et al., 2011).

2.8. DIREÇÕES FUTURAS DE PESQUISA

O mapeamento sugere cinco direções prioritárias.

A primeira é a construção de conjuntos de dados públicos e anotados voltados a tarefas de mensuração, hoje praticamente inexistentes. Enquanto a inspeção de superfícies dispõe de coleções compartilhadas, a mensuração de toras e de pilhas carece de referências comuns, o que impede a comparação direta entre métodos e obriga cada estudo a construir sua própria base (MURATA et al., 1999; MOHAMED et al., 2021; DAI; LI, 2022; CARMO et al., 2019; LI et al., 2020; RINNHOFFER et al., 2003).

A segunda é a avaliação sistemática de arquiteturas recentes em tarefas de contagem e de medição. Detectores baseados em transformadores (CARION et al., 2020; DOSOVITSKIY, 2021) e modelos de fundação para segmentação (KIRILLOV et al., 2023) foram amplamente avaliados em domínios genéricos, mas sua aplicação a objetos densamente agrupados e visualmente homogêneos, como as faces de toras em uma pilha, permanece pouco explorada.

A terceira é a integração de múltiplas modalidades de sensoriamento, combinando imagem, varredura laser e sensores de profundidade, de modo a superar as limitações intrínsecas da imagem monocular na recuperação de geometria (GENG et al., 2021; WEI et al., 2011; VAN DEN BULCKE et al., 2008; FORSBERG et al., 2010; KRÄHENBÜHL et al., 2013).

A quarta é a transferência de métodos entre subdomínios. A baixa taxa de citação cruzada identificada na análise de coocorrência sugere que soluções consolidadas em uma categoria permanecem desconhecidas nas demais, o que representa desperdício de esforço (WIMMER et al., 2021; LI; LIDESKOG, 2021; FORTIN et al., 2022; ANDERSSON et al., 2021).

A quinta é a padronização do relato de desempenho. A comparação entre estudos é hoje dificultada pela heterogeneidade das métricas, dos limiares e das condições de aquisição, de modo que ganhos relatados nem sempre são comparáveis (RAVINDRAN et al., 2018; DA SILVA et al., 2017; LENS et al., 2020; WU et al., 2021; HE et al., 2020c).

2.9. CONCLUSÕES

O mapeamento sistemático de 1.019 artigos publicados entre 1983 e 2026 mostra que a visão computacional aplicada à madeira deixou de ser um nicho especializado e consolidou-se como tecnologia industrial, com aceleração pronunciada a partir de 2016, quando se concentraram 63,7% de toda a produção do campo.

A distribuição metodológica revela divisão de trabalho estável entre as abordagens: o aprendizado profundo prevalece nas tarefas classificatórias, enquanto os métodos clássicos permanecem majoritários no conjunto do campo, com 73,6% dos artigos, sustentados pelas aplicações de mensuração determinística que exigem rastreabilidade do cálculo. A participação da visão computacional clássica varia de 53,7% na identificação de espécie e origem a 90,9% na medição dimensional, e essa ordenação corresponde ao grau em que cada tarefa exige medida física em vez de classificação.

A distribuição temática evidencia forte concentração no controle de qualidade de superfície, com 48,3% dos artigos, e correspondente sub-representação das aplicações de mensuração, que somam 8,8% do corpus. Essa lacuna, aqui quantificada, delimita o problema tratado nos capítulos seguintes e justifica o desenvolvimento de métodos específicos para a contagem e a estimativa volumétrica de toras a partir de imagens.

Por fim, a taxonomia e o procedimento de classificação estabelecidos neste capítulo constituem instrumento reprodutível, passível de reaplicação em atualizações futuras do levantamento, o que permitirá acompanhar a evolução metodológica e temática do campo com critérios estáveis.

2.10. REFERÊNCIAS

- ABDELJABER, O.; HABITE, T.; OLSSON, A. Automatic estimation of annual ring profiles in Norway spruce timber boards using optical scanning and deep learning. **Computers & Structures**, v. 275, p. 106912, 2023. <https://doi.org/10.1016/j.compstruc.2022.106912>.
- AGUILERA, C.A.; GONZÁLEZ BÖHME, L.F.; VALDES, F.; QUITRAL-ZAPATA, F.J.; RĂDUCANU, B. A Hand-Drawn Language for Human-Robot Collaboration in Wood Stereotomy. **IEEE Access**, v. 11, p. 100975-100985, 2023. <https://doi.org/10.1109/ACCESS.2023.3314337>.
- ANDERSSON, J.; BODIN, K.; LINDMARK, D.; SERVIN, M.; WALLIN, E. Reinforcement Learning Control of a Forestry Crane Manipulator. 2021. **Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)** [...]. [S. l.: s. n.], 2021. p. 2121-2128. <https://doi.org/10.1109/IROS51168.2021.9636219>.
- ARNOULD, O.; ARINERO, R. Towards a better understanding of wood cell wall characterisation with contact resonance atomic force microscopy. **Composites Part A: Applied Science and Manufacturing**, v. 74, p. 69-76, 2015. <https://doi.org/10.1016/j.compositesa.2015.03.026>.
- BARADIT, E.; GATICA, C.; YÁÑEZ, M.; FIGUEROA, J.C.; GUZMÁN, R.; CATALÁN, C. Surface roughness estimation of wood boards using speckle interferometry. **Optics and Lasers in Engineering**, v. 128, p. 106009, 2020. <https://doi.org/10.1016/j.optlaseng.2020.106009>.
- BHANDARKAR, S.M.; FAUST, T.D.; TANG, M. CATALOG: A system for detection and rendering of internal log defects using computer tomography. **Machine Vision and Applications**, v. 11, p. 171-190, 1998. <https://doi.org/10.1007/s001380050100>.
- BHANDARKAR, S.M.; LUO, X.; DANIELS, R.F.; TOLLNER, E.W. Automated planning and optimization of lumber production using machine vision and computed tomography. **IEEE Transactions on Automation Science and Engineering**, v. 5, p. 677-695, 2008. <https://doi.org/10.1109/TASE.2008.925254>.
- BRODIE, J.R.; HANSEN, A.C.; REID, J.F. Size assessment of stacked logs via the Hough transform. **Transactions of the American Society of Agricultural Engineers**, v. 37, p. 303-310, 1994. <https://doi.org/10.13031/2013.28085>.
- CARION, N.; MASSA, F.; SYNNAEVE, G.; USUNIER, N.; KIRILLOV, A.; ZAGORUYKO, S. End-to-End Object Detection with Transformers. 2020. **Proceedings of the European Conference on Computer Vision (ECCV)** [...]. [S. l.: s. n.], 2020. p. 213-229. https://doi.org/10.1007/978-3-030-58452-8_13.
- CARMO, R.N.F.; VALENCA, J.; BENCARDINO, F.; CRISTOFARO, S.; CHIERA, D. Assessment of plastic rotation and applied load in reinforced concrete, steel and timber beams using image-based analysis. **Engineering Structures**, v. 198, p. 109519, 2019. <https://doi.org/10.1016/j.engstruct.2019.109519>.
- CASAS, G.G.; ISMAIL, Z.H.; LIMEIRA, M.M.C.; DA SILVA, A.A.L.; LEITE, H.G. Automatic Detection and Counting of Stacked Eucalypt Timber Using the YOLOv8 Model. **Forests**, v. 14, p. 2369, 2023. <https://doi.org/10.3390/f14122369>.
- CASTELLINI, P.; ABASKIN, V.; ACHIMOVA, E. Portable electronic speckle interferometry device for the damages measurements in veneered wood artworks. **Journal of Cultural**

Heritage, v. 9, p. 225-233, 2008. <https://doi.org/10.1016/j.culher.2008.05.002>.

CHEN, L.-C.; PARDESHI, M.S.; LO, W.-T.; SHEU, R.-K.; PAI, K.-C.; CHEN, C.-Y.; TSAI, P.-Y.; TSAI, Y.-T. Edge-glued wooden panel defect detection using deep learning. **Wood Science and Technology**, v. 56, p. 477-507, 2022. <https://doi.org/10.1007/s00226-021-01316-3>.

CHEN, S.; SHIINA, R.; NAKAI, K.; AWANO, T.; YOSHINAGA, A.; SUGIYAMA, J. Potential of machine learning approaches for predicting mechanical properties of spruce wood in the transverse direction. **Journal of Wood Science**, v. 69, p. 22, 2023. <https://doi.org/10.1186/s10086-023-02096-z>.

CHIDA, H.; TAKAHASHI, N. Study on image diagnosis of timber houses damaged by earthquake using deep learning. **Japan Architectural Review**, v. 4, p. 420-430, 2021. <https://doi.org/10.1002/2475-8876.12221>.

CHOLEWA, N.; SUMMERS, P.T.; FEIH, S.; MOURITZ, A.P.; LATTIMER, B.Y.; CASE, S.W. A Technique for Coupled Thermomechanical Response Measurement Using Infrared Thermography and Digital Image Correlation (TDIC). **Experimental Mechanics**, v. 56, p. 145-164, 2016. <https://doi.org/10.1007/s11340-015-0086-1>.

CONNERS, R.; MCMILLIN, C.W.; LIN, K.; VÁSQUEZ-ESPINOSA, R.E. Identifying and Locating Surface Defects in Wood: Part of an Automated Lumber Processing System. **IEEE Transactions on Pattern Analysis and Machine Intelligence**, v. PAMI-5, p. 573-583, 1983. <https://doi.org/10.1109/TPAMI.1983.4767446>.

COUDRAY, N.; DIETERLEN, A.; ROTH, E.; TROUVÉ, G. Density measurement of fine aerosol fractions from wood combustion sources using ELPI distributions and image processing techniques. **Fuel**, v. 88, p. 947-954, 2009. <https://doi.org/10.1016/j.fuel.2008.12.013>.

CRUZ, H.; SAPORITI MACHADO, J.S. Effects of beetle attack on the bending and compression strength properties of pine wood. **Advanced Materials Research**, v. 778, p. 145-151, 2013. <https://doi.org/10.4028/www.scientific.net/AMR.778.145>.

DA SILVA, N.; DE RIDDER, M.; BAETENS, J.M.; VAN DEN BULCKE, J.; ROUSSEAU, M.; BRUNO, O.; BEECKMAN, H.; VAN ACKER, J.; DE BAETS, B. Automated classification of wood transverse cross-section micro-imagery from 77 commercial Central-African timber species. **Annals of Forest Science**, v. 74, p. 30, 2017. <https://doi.org/10.1007/s13595-017-0619-0>.

DAI, Y.; LI, H. Multi-Camera Digital Image Correlation in Deformation Measurement of Civil Components with Large Slenderness Ratio and Large Curvature. **Materials**, v. 15, p. 6281, 2022. <https://doi.org/10.3390/ma15186281>.

DAS, S.; TEFAMARIAM, S. Reliability assessment of stochastic dynamical systems using physics informed neural network based PDEM. **Reliability Engineering & System Safety**, v. 243, p. 109849, 2024. <https://doi.org/10.1016/j.ress.2023.109849>.

DEROME, D.; GRIFFA, M.; KOEBEL, M.; LI, J. Hysteretic swelling of wood at cellular scale probed by phase-contrast X-ray tomography. **Journal of Structural Biology**, v. 173, p. 180-190, 2011. <https://doi.org/10.1016/j.jsb.2010.08.011>.

DING, F.; ZHUANG, Z.; LIU, Y.; JIANG, D.; YAN, X.; WANG, Z. Detecting defects on solid wood panels based on an improved SSD algorithm. **Sensors**, v. 20, p. 5315, 2020. <https://doi.org/10.3390/s20185315>.

DOSOVITSKIY, A. An Image is Worth 16x16 Words: Transformers for Image Recognition at

Scale. 2021. **Proceedings of the International Conference on Learning Representations (ICLR)** [...]. [S. l.: s. n.], 2021.

EUROPEAN UNION. REGULATION (EU) 2023/1115 ON DEFORESTATION-FREE PRODUCTS. **Official Journal of the European Union**, v. L 150, p. 206-247, 2023.

FABIJANSKA, A.; DANEK, M.; BARNIAK, J. Wood species automatic identification from wood core images with a residual convolutional neural network. **Computers and Electronics in Agriculture**, v. 181, p. 105941, 2021. <https://doi.org/10.1016/j.compag.2020.105941>.

FABIJAŃSKA, A.; DANEK, M. DeepDendro—A tree rings detector based on a deep convolutional neural network. **Computers and Electronics in Agriculture**, v. 150, p. 353-363, 2018. <https://doi.org/10.1016/j.compag.2018.05.005>.

FAHLEN, J.; SALMEN, L. Cross-sectional structure of the secondary wall of wood fibers as affected by processing. **Journal of Materials Science**, v. 38, p. 119-126, 2003. <https://doi.org/10.1023/A:1021174118468>.

FENG, X.; WU, Z.; SANG, R.; WANG, F.; ZHU, Y.; WU, M. Surface design of wood-based board to imitate wood texture using 3D printing technology. **BioResources**, v. 14, p. 8196-8211, 2019. <https://doi.org/10.15376/biores.14.4.8196-8211>.

FORSBERG, F.; SJODAHL, M.; MOOSER, R.; HACK, E.; WYSS, P. Full three-dimensional strain measurements on wood exposed to three-point bending: Analysis by use of digital volume correlation applied to synchrotron radiation micro-computed tomography image data. **Strain**, v. 46, p. 47-60, 2010. <https://doi.org/10.1111/j.1475-1305.2009.00687.x>.

FORTIN, J.-M.; GAMACHE, O.; GRONDIN, V.; POMERLEAU, F.; GIGUERE, P. Instance Segmentation for Autonomous Log Grasping in Forestry Operations. 2022. **Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)** [...]. [S. l.: s. n.], 2022. p. 6064-6071. <https://doi.org/10.1109/IROS47612.2022.9982286>.

FUNT, B.V.; BRYANT, E.C. Computer vision system that analyses CT-SCANS of sawlogs. 1985. **Proceedings of the Conference on Applications of Artificial Intelligence** [...]. [S. l.: s. n.], 1985. p. 175-177.

FUNT, B.V.; BRYANT, E.C. Detection of internal log defects by automatic interpretation of computer tomography images. **Forest Products Journal**, v. 37, p. 56-62, 1987.

GAO, M.; CHEN, J.; MU, H.; QI, D. A transfer residual neural network based on ResNet-34 for detection of wood knot defects. **Forests**, v. 12, p. 212, 2021. <https://doi.org/10.3390/f12020212>.

GARCIA, R.A.; ROSERO-ALVARADO, J.; HERNÁNDEZ, R.E. Moisture-induced strains in earlywood and latewood of mature and juvenile woods in jack pine from 3D-DIC measurements. **Wood Material Science & Engineering**, v. 18, p. 570-579, 2023. <https://doi.org/10.1080/17480272.2022.2056714>.

GE, Y.; JIANG, D.; SUN, L. Wood Veneer Defect Detection Based on Multiscale DETR with Position Encoder Net. **Sensors**, v. 23, p. 4837, 2023. <https://doi.org/10.3390/s23104837>.

GENG, H.; JIN, H.; ZHOU, X.; LU, J.; LI, X.; ZHAO, J. Evaluation and Quantification of Wood 3D Pore Structure via X-CT Technology. **Journal of Forestry Engineering**, v. 37, p. 108-114, 2021. <https://doi.org/10.16270/j.cnki.slgc.2021.05.011>.

GU, I.Y.-H.; ANDERSSON, H.; VICÉN-BUENO, R. Wood defect classification based on image

analysis and support vector machines. **Wood Science and Technology**, v. 44, p. 693-704, 2010. <https://doi.org/10.1007/s00226-009-0287-9>.

GULIRMAK, E.A.; BINGÜL, Z. Determining Robot Trajectory Planning Using Image Processing for Wood Painting. 2023. **Proceedings of the 2023 7th International Symposium on Multidisciplinary Studies and Innovative Technologies (ISMSIT)** [...]. [S. l.: s. n.], 2023. <https://doi.org/10.1109/ISMSIT58785.2023.10304909>.

GUO, X.; LEE, C. Preliminary study of phase-shifting strobo-stereoscopy for cutting tool monitoring. **Journal of Manufacturing Processes**, v. 64, p. 1214-1222, 2021. <https://doi.org/10.1016/j.jmapro.2021.02.053>.

GUTZEIT, E.; OHL, S.; VOSKAMP, J.; KUIJPER, A.; URBAN, B. Automatic wood log segmentation using graph cuts. **Communications in Computer and Information Science**, v. 229, p. 96-109, 2011. https://doi.org/10.1007/978-3-642-25382-9_7.

HAN, S.; JIANG, X.; WU, Z. An Improved YOLOv5 Algorithm for Wood Defect Detection Based on Attention. **IEEE Access**, v. 11, p. 71593-71604, 2023. <https://doi.org/10.1109/ACCESS.2023.3293864>.

HE, K.; ZHANG, X.; REN, S.; SUN, J. Deep Residual Learning for Image Recognition. 2016. **Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)** [...]. [S. l.: s. n.], 2016. p. 770-778. <https://doi.org/10.1109/CVPR.2016.90>.

HE, S.; CHEN, Y.; WU, Z.; HU, Y. Research progress on wood/bamboo microscopic fluid transportation. **Journal of Forestry Engineering**, v. 5, p. 12-19, 2020a. <https://doi.org/10.13360/j.issn.2096-1359.201906043>.

HE, T.; LIU, Y.; XU, C.; ZHOU, X.; HU, Z.; FAN, J. A fully convolutional neural network for wood defect location and identification. **IEEE Access**, v. 7, p. 123453-123462, 2019. <https://doi.org/10.1109/ACCESS.2019.2937461>.

HE, T.; LIU, Y.; YU, Y.; ZHAO, Q.; HU, Z. Application of deep convolutional neural network on feature extraction and detection of wood defects. **Measurement**, v. 152, p. 107357, 2020b. <https://doi.org/10.1016/j.measurement.2019.107357>.

HE, T.; LU, Y.; JIAO, L.; ZHANG, Y.; JIANG, X.; YIN, Y. Developing deep learning models to automate rosewood tree species identification for CITES designation and implementation. **Holzforschung**, v. 74, p. 877-888, 2020c. <https://doi.org/10.1515/hf-2020-0006>.

HE, Z.; NI, F.; WANG, W.; ZHANG, J. A physics-informed deep learning method for solving direct and inverse heat conduction problems of materials. **Materials Today Communications**, v. 28, p. 102719, 2021. <https://doi.org/10.1016/j.mtcomm.2021.102719>.

HENKE, K.; PAWLOWSKI, R.; SCHREGLE, P.; WINTER, S. Use of digital image processing in the monitoring of deformations in building structures. **Journal of Civil Structural Health Monitoring**, v. 5, p. 141-152, 2015. <https://doi.org/10.1007/s13349-014-0091-6>.

HERMANSON, J.C.; WIEDENHOEFT, A.C. A brief review of machine vision in the context of automated wood identification systems. **IAWA Journal**, v. 32, p. 233-250, 2011. <https://doi.org/10.1163/22941932-90000054>.

HIETANIEMI, R.; LÓOPEZ, M.B.; HANNUKSELA, J.; SILVÉN, O. A real-time imaging system for lumber strength prediction. **Forest Products Journal**, v. 64, p. 126-133, 2014. <https://doi.org/10.13073/FPJ-D-13-00078>.

- HINTON, G.E.; SALAKHUTDINOV, R.R. Reducing the dimensionality of data with neural networks. **Science**, v. 313, p. 504-507, 2006. <https://doi.org/10.1126/science.1127647>.
- HWANG, S.-W.; KOBAYASHI, K.; SUGIYAMA, J. Detection and visualization of encoded local features as anatomical predictors in cross-sectional images of Lauraceae. **Journal of Wood Science**, v. 66, p. 16, 2020. <https://doi.org/10.1186/s10086-020-01864-5>.
- HWANG, S.-W.; SUGIYAMA, J. Computer vision-based wood identification and its expansion and contribution potentials in wood science: A review. **Plant Methods**, v. 17, p. 47, 2021a. <https://doi.org/10.1186/s13007-021-00746-1>.
- HWANG, S.-W.; SUGIYAMA, J. Evaluation of image partitioning strategies for preserving spatial information of cross-sectional micrographs in automated wood recognition of Fagaceae. **Journal of Wood Science**, v. 67, p. 19, 2021b. <https://doi.org/10.1186/s10086-021-01953-z>.
- IBRAHIM, I.; MOHD KHAIRUDDIN, A.S.M.; ABU TALIP, M.S.; AROF, H.; YUSOF, R. Tree species recognition system based on macroscopic image analysis. **Wood Science and Technology**, v. 51, p. 431-444, 2017. <https://doi.org/10.1007/s00226-016-0859-4>.
- JIA, H.; XU, H.; WANG, L.; ZHANG, J.; CHU, X.; TANG, X. Quantitative identification of surface defects in wood paneling based on improved YOLOv5. **Journal of Beijing Forestry University**, v. 45, p. 132-140, 2023. <https://doi.org/10.12171/j.1000-1522.20220419>.
- JIANG, Z.; LIANG, Y.; SU, Z.; CHEN, A.; SUN, J. Nondestructive testing of mechanical properties of bamboo-wood composite container floor by image processing. **Forests**, v. 12, p. 1535, 2021. <https://doi.org/10.3390/f12111535>.
- KAMAL, K.; QAYYUM, R.; MATHAVAN, S.; ZAFAR, T. Wood defects classification using laws texture energy measures and supervised learning approach. **Advanced Engineering Informatics**, v. 34, p. 125-135, 2017. <https://doi.org/10.1016/j.aei.2017.09.007>.
- KATZENMAIER, M.; GARNOT, V.S.F.; BJÖRKLUND, J.; SCHNEIDER, L.; WEGNER, J.D.; VON ARX, G. Towards ROXAS AI: Deep learning for faster and more accurate conifer cell analysis. **Dendrochronologia**, v. 81, p. 126126, 2023. <https://doi.org/10.1016/j.dendro.2023.126126>.
- KIBLEUR, P.; AELTERMAN, J.; BOONE, M.N.; VAN DEN BULCKE, J.; VAN ACKER, J. Deep learning segmentation of wood fiber bundles in fiberboards. **Composites Science and Technology**, v. 221, p. 109287, 2022. <https://doi.org/10.1016/j.compscitech.2022.109287>.
- KIRILLOV, A.; MINTUN, E.; RAVI, N.; MAO, H.; ROLLAND, C.; GUSTAFSON, L.; XIAO, T.; WHITEHEAD, S.; BERG, A.C.; LO, W.-Y.; DOLLÁR, P.; GIRSHICK, R. Segment Anything. 2023. **Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)** [...]. [S. l.: s. n.], 2023. p. 4015-4026. <https://doi.org/10.1109/ICCV51070.2023.00371>.
- KRIZHEVSKY, A.; SUTSKEVER, I.; HINTON, G.E. ImageNet Classification with Deep CNNs. 2012. **Proceedings of the Advances in Neural Information Processing Systems (NeurIPS)** [...]. [S. l.: s. n.], 2012. p. 1097-1105.
- KRUGLOV, A.V. The Algorithm of the Roundwood Volume Measurement via Photogrammetry. 2016. **Proceedings of the 2016 International Conference on Digital Image Computing: Techniques and Applications (DICTA)** [...]. [S. l.: s. n.], 2016. <https://doi.org/10.1109/DICTA.2016.7797088>.
- KRÄHENBÜHL, A.; KERAUTRET, B.; DEBLED-RENNESON, I. Knot segmentation in

noisy 3D images of wood. In: **Discrete Geometry for Computer Imagery**. [S. l.]: Springer, 2013. p. 383-394. https://doi.org/10.1007/978-3-642-37067-0_33.

KURDTHONGMEE, W.; SUWANNARAT, K. Locating Wood Pith in a Wood Stem Cross Sectional Image Using YOLO Object Detection. 2019. **Proceedings of the International Conference on Technologies and Applications of Artificial Intelligence (TAAI)** [...]. [S. l.: s. n.], 2019. p. 1-5. <https://doi.org/10.1109/TAAI48200.2019.8959823>.

LECUN, Y.; BOTTOU, L.; BENGIO, Y.; HAFFNER, P. Gradient-based learning applied to document recognition. **Proceedings of the IEEE**, v. 86, p. 2278-2324, 1998. <https://doi.org/10.1109/5.726791>.

LENS, F.; LIANG, C.; GUO, Y.; TANG, X.; JAHANBANIFARD, M.; DA SILVA, F.S.C.; CECCANTINI, G.; VERBEEK, F.J. Computer-assisted timber identification based on features extracted from microscopic wood sections. **IAWA Journal**, v. 41, p. 660-680, 2020. <https://doi.org/10.1163/22941932-bja10029>.

LI, S.; LIDESKOG, H. Implementation of a system for real-time detection and localization of terrain objects on harvested forest land. **Forests**, v. 12, p. 1142, 2021. <https://doi.org/10.3390/f12091142>.

LI, W.; ZHANG, Z.; ZHOU, G.; LENG, W.; MEI, C. Understanding the interaction between bonding strength and strain distribution of plywood. **International Journal of Adhesion and Adhesives**, v. 96, p. 102506, 2020. <https://doi.org/10.1016/j.ijadhadh.2019.102506>.

LIN, Y.; CHEN, D.; LIANG, S.; XU, Z.; QIU, Y.; ZHANG, J.; LIU, X. Color classification of wooden boards based on machine vision and the clustering algorithm. **Applied Sciences**, v. 10, p. 6816, 2020. <https://doi.org/10.3390/app10196816>.

LIU, C.; SHARAN, L.; ADELSON, E.H.; ROSENHOLTZ, R. Exploring features in a Bayesian framework for material recognition. 2010. **Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)** [...]. [S. l.: s. n.], 2010. p. 239-246. <https://doi.org/10.1109/CVPR.2010.5540207>.

LIU, S.; JIANG, W.; WU, L.; WEN, H.; LIU, M.; WANG, Y. Real-Time Classification of Rubber Wood Boards Using an SSR-Based CNN. **IEEE Transactions on Instrumentation and Measurement**, v. 69, p. 8725-8734, 2020. <https://doi.org/10.1109/TIM.2020.3001370>.

LOPES, D.J.V.; BURGEEEN, G.W.; BOBADILHA, G.S.; BARNES, H.M. Automated means to classify lab-scale termite damage. **Computers and Electronics in Agriculture**, v. 168, p. 105105, 2020. <https://doi.org/10.1016/j.compag.2019.105105>.

LU, Z.; YAO, H.; LYU, Y.; HE, S.; NING, H.; YU, Y.; ZHAI, L.; ZHOU, L. A Deep Learning Method for Log Diameter Measurement Using Wood Images Based on Yolov3 and DeepLabv3+. **Forests**, v. 15, p. 755, 2024. <https://doi.org/10.3390/f15050755>.

MARICHAL, H.; PASSARELLA, D.; RANDALL, G. Automatic Wood Pith Detector: Local Orientation Estimation and Robust Accumulation. In: **Pattern Recognition**. [S. l.]: Springer, 2025. p. 1-15. https://doi.org/10.1007/978-3-031-78447-7_1.

MARTINEZ-GARCIA, J.; STELZNER, I.; STELZNER, J.; GWERDER, D.; SCHÜTZ, P. Automated 3D tree-ring detection and measurement from X-ray computed tomography. **Dendrochronologia**, v. 69, p. 125877, 2021. <https://doi.org/10.1016/j.dendro.2021.125877>.

MEGID, W.A.; KHAYAT, K.H. Variations in surface quality of self-consolidation and highly workable concretes with formwork material. **Construction and Building Materials**, v. 236, p.

117638, 2020. <https://doi.org/10.1016/j.conbuildmat.2019.117638>.

MOHAMED, A.; DENG, Y.; ZHANG, H.; WONG, S.H.F.; UHEIDA, K.; ZHANG, Y.X.; ZHU, M.-C.; LEHMANN, M.; QUAN, Y. Photogrammetric evaluation of shear modulus of glulam timber using torsion test method and dual stereo vision system. **European Journal of Wood and Wood Products**, v. 79, p. 1451-1463, 2021. <https://doi.org/10.1007/s00107-021-01729-8>.

MOSKALIK, T.; TYMENDORF, Ł.; VAN DER SAAR, J.; TRZCIŃSKI, G. Methods of Wood Volume Determining and Its Implications for Forest Transport. **Sensors**, v. 22, n. 16, p. 6028, 2022. <https://doi.org/10.3390/s22166028>.

MURATA, K.; MASUDA, M.; ICHIMARU, M. Analysis of radial compression behavior of wood using digital image correlation method. **Mokuzai Gakkaishi**, v. 45, p. 375-381, 1999.

NAUMANN, A.; NAVARRO-GONZALEZ, M.; PEDDIREDDI, S.; KUES, U.; POLLE, A. Fourier transform infrared microscopy and imaging: Detection of fungi in wood. **Fungal Genetics and Biology**, v. 42, p. 829-835, 2005. <https://doi.org/10.1016/j.fgb.2005.06.003>.

NGUYEN, D.M.; ALMEIDA, G.; NGUYEN, T.M.L.; ZHANG, J.; LU, P.; COLIN, J.; PERRE, P. A Critical Review of Current Imaging Techniques to Investigate Water Transfers in Wood and Biosourced Materials. **Transport in Porous Media**, v. 137, p. 21-61, 2021. <https://doi.org/10.1007/s11242-020-01538-2>.

NIȚĂ, M.; BORZ, S. Accuracy of a Smartphone-based freeware solution and two shape reconstruction algorithms in log volume measurements. **Computers and Electronics in Agriculture**, v. 205, p. 107653, 2023. <https://doi.org/10.1016/j.compag.2023.107653>.

NORELL, K.; BORGEFORS, G. Estimation of pith position in untreated log ends in sawmill environments. **Computers and Electronics in Agriculture**, v. 63, p. 155-162, 2008. <https://doi.org/10.1016/j.compag.2008.02.006>.

NYSTRÖM, J.; KLINE, D.E. Automatic classification of compression wood in green southern yellow pine. **Wood and Fiber Science**, v. 32, p. 301-310, 2000.

PAGE, M.J.; MCKENZIE, J.E.; BOSSUYT, P.M.; BOUTRON, I.; HOFFMANN, T.C.; MULROW, C.D.; SHAMSEER, L.; TETZLAFF, J.M.; AKL, E.A.; BRENNAN, S.E.; ET AL. The PRISMA 2020 statement: An updated guideline for reporting systematic reviews and meta-analyses. **BMJ**, v. 372, p. n71, 2021.

PERRÉ, P.; ALMEIDA, G.; AYOUEZ, M.; FRANK, X. New modelling approaches to predict wood properties from its cellular structure: Image-based representation and meshless methods. **Annals of Forest Science**, v. 73, p. 147-162, 2016. <https://doi.org/10.1007/s13595-015-0519-0>.

PIERDICCA, R.; BALESTRA, M.; MICHELETTI, G.; FELICETTI, A.; TOSCANO, G. Semi-automatic detection and segmentation of wooden pellet size exploiting a deep learning approach. **Renewable Energy**, v. 197, p. 406-416, 2022. <https://doi.org/10.1016/j.renene.2022.07.109>.

RAHMAN, A.; STREET, J.T.; WOOTEN, J.; MARUFUZZAMAN, M.; GUDE, V.G.; BUCHANAN, R.; WANG, H. MoistNet: Machine vision-based deep learning models for wood chip moisture content measurement. **Expert Systems with Applications**, v. 259, p. 125363, 2025. <https://doi.org/10.1016/j.eswa.2024.125363>.

RAVINDRAN, P.; COSTA, A.; SOARES, R.; WIEDENHOEFT, A.C. Classification of CITES-listed and other neotropical Meliaceae wood images using convolutional neural networks. **Plant Methods**, v. 14, p. 25, 2018. <https://doi.org/10.1186/s13007-018-0292-9>.

- REDMON, J.; DIVVALA, S.; GIRSHICK, R.; FARHADI, A. You Only Look Once: Unified, Real-Time Object Detection. 2016. **Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)** [...]. [S. l.: s. n.], 2016. p. 779-788.
- RESENTE, G.; GILLERT, A.; TROUILLIER, M.; ANADON-ROSELL, A.; PETERS, R.L.; VON ARX, G.; FREIHERR VON LUKAS, U.; WILMKING, M. Mask, Train, Repeat! Artificial Intelligence for Quantitative Wood Anatomy. **Frontiers in Plant Science**, v. 12, p. 767400, 2021. <https://doi.org/10.3389/fpls.2021.767400>.
- RINNHOFFER, A.; PETUTSCHNIGG, A.; ANDREU, J.-P. Internal log scanning for optimizing breakdown. **Computers and Electronics in Agriculture**, v. 41, p. 7-21, 2003. [https://doi.org/10.1016/S0168-1699\(03\)00039-5](https://doi.org/10.1016/S0168-1699(03)00039-5).
- RONNEBERGER, O.; FISCHER, P.; BROX, T. U-Net: Convolutional Networks for Biomedical Image Segmentation. 2015. **Proceedings of the International Conference on Medical Image Computing and Computer-Assisted Intervention (MICCAI)** [...]. [S. l.: s. n.], 2015. p. 234-241. https://doi.org/10.1007/978-3-319-24574-4_28.
- RUMELHART, D.E.; HINTON, G.E.; WILLIAMS, R.J. Learning representations by back-propagating errors. **Nature**, v. 323, p. 533-536, 1986. <https://doi.org/10.1038/323533a0>.
- RYDVAL, M.; BJÖRKLUND, J.; VON ARX, G.; BEGOVIĆ, K.; LEXA, M.; NOGUEIRA, J.; SCHURMAN, J.S.; JIANG, Y. Ultra-high-resolution reflected-light imaging for dendrochronology. **Dendrochronologia**, v. 83, p. 126160, 2024. <https://doi.org/10.1016/j.dendro.2023.126160>.
- SAMDANGDECH, N.; PHIPHOBMONGKOL, S. Log-End Cut-Area Detection in Images Taken from Rear End of Eucalyptus Timber Trucks. 2018. **Proceedings of the 2018 15th International Joint Conference on Computer Science and Software Engineering (JCSSE)** [...]. [S. l.: s. n.], 2018. <https://doi.org/10.1109/JCSSE.2018.8457388>.
- SANDBERG, K.; SALIN, J.-G. Liquid water absorption in dried Norway spruce timber measured with CT scanning and viewed as a percolation process. **Wood Science and Technology**, v. 46, p. 207-219, 2012. <https://doi.org/10.1007/s00226-010-0371-1>.
- SANG, R.; MANLEY, A.J.; WU, Z.; FENG, X. Digital 3D wood texture: UV-curable inkjet printing on board surface. **Coatings**, v. 10, p. 1144, 2020. <https://doi.org/10.3390/coatings10121144>.
- SCHMOLDT, D.L.; LI, P.; ABBOTT, A. Machine vision using artificial neural networks with local 3D neighborhoods. **Computers and Electronics in Agriculture**, v. 16, p. 255-271, 1997. [https://doi.org/10.1016/s0168-1699\(97\)00002-1](https://doi.org/10.1016/s0168-1699(97)00002-1).
- SCHRAML, R.; CHARWAT-PESSLER, J.; PETUTSCHNIGG, A.; UHL, A. Towards the applicability of biometric wood log traceability using digital log end images. **Computers and Electronics in Agriculture**, v. 119, p. 112-122, 2015. <https://doi.org/10.1016/j.compag.2015.10.003>.
- SEDIGHI-GILANI, M.; BOONE, M.N.; MADER, K.; SCHWARZE, F.W.M.R. Synchrotron X-ray micro-tomography imaging and analysis of wood degraded by *Physiporus vitreus* and *Xylaria longipes*. **Journal of Structural Biology**, v. 187, p. 149-157, 2014. <https://doi.org/10.1016/j.jsb.2014.06.003>.
- SERRANO, E.; ENQUIST, B. Contact-free measurement and non-linear finite element analyses of strain distribution along wood adhesive bonds. **Holzforschung**, v. 59, p. 641-646, 2005.

<https://doi.org/10.1515/HF.2005.103>.

SETTIMI, A.; GAMERRO, J.; WEINAND, Y. Augmented Carpentry: Computer vision-assisted framework for manual fabrication. **Automation in Construction**, v. 179, p. 106433, 2025. <https://doi.org/10.1016/j.autcon.2025.106433>.

SHI, J.; XIANG, W.; LIU, Q.; SHAH, S. MtreeRing: An R package with graphical user interface for automatic measurement of tree ring widths using image processing techniques. **Dendrochronologia**, v. 58, p. 125644, 2019. <https://doi.org/10.1016/j.dendro.2019.125644>.

SPREITZER, G.; TUNNICLIFFE, J.; FRIEDRICH, H. Porosity and volume assessments of large wood (LW) accumulations. **Geomorphology**, v. 358, p. 107122, 2020. <https://doi.org/10.1016/j.geomorph.2020.107122>.

STEPPE, K.; CNUUDE, V.; GIRARD, C.; LEMEUR, R.; CNUUDE, J.-P.; JACOBS, P. Use of X-ray computed microtomography for non-invasive determination of wood anatomical characteristics. **Journal of Structural Biology**, v. 148, p. 11-21, 2004. <https://doi.org/10.1016/j.jsb.2004.05.001>.

SUGIARTO, B.; PRAKASA, E.; WARDOYO, R.; DAMAYANTI, R.; KRISDIANTO, K.; DEWI, L.M.; PARDEDE, H.F.; RIAN TO, Y. Wood identification based on histogram of oriented gradient (HOG) feature and support vector machine (SVM) classifier. 2017. **Proceedings of the International Conference on Information Technology and Electrical Engineering (ICITISEE)** [...]. [S. l.: s. n.], 2017. p. 1-6. <https://doi.org/10.1109/ICITISEE.2017.8285523>.

TU, Y.; LING, Z.; GUO, S.; WEN, H. An Accurate and Real-Time Surface Defects Detection Method for Sawn Lumber. **IEEE Transactions on Instrumentation and Measurement**, v. 70, p. 3024431, 2021. <https://doi.org/10.1109/TIM.2020.3024431>.

URBONAS, A.; RAUDONIS, V.; MASKELIŪNAS, R.; DAMASEVICIUS, R. Automated identification of wood veneer surface defects using faster region-based convolutional neural network with data augmentation and transfer learning. **Applied Sciences**, v. 9, p. 4898, 2019. <https://doi.org/10.3390/app9224898>.

VAN DEN BULCKE, J.; BOONE, M.A.; DHAENE, J.; VAN LOO, D.; VAN HOOREBEKE, L.; BOONE, M.N.; WYFFELS, F.; BEECKMAN, H.; VAN ACKER, J.; DE MIL, T. Advanced X-ray CT scanning can boost tree ring research for earth system sciences. **Annals of Botany**, v. 124, p. 837-847, 2019. <https://doi.org/10.1093/aob/mcz126>.

VAN DEN BULCKE, J.; BOONE, M.N.; VAN ACKER, J.; STEVENS, M.; VAN HOOREBEKE, L. X-ray tomography as a tool for detailed anatomical analysis. **Annals of Forest Science**, v. 66, p. 508, 2009. <https://doi.org/10.1051/forest/2009033>.

VAN DEN BULCKE, J.; MASSCHAELE, B.; DIERICK, M.; VAN ACKER, J.; STEVENS, M.; VAN HOOREBEKE, L. Three-dimensional imaging and analysis of infested coated wood with X-ray submicron CT. **International Biodeterioration & Biodegradation**, v. 61, p. 278-286, 2008. <https://doi.org/10.1016/j.ibiod.2007.09.004>.

VERHEYEN, M.; BECKERS, W.; CLAESEN, E.; MOONEN, G.; DEMEESTER, E. Vision-based sorting of medium density fibreboard and grade A wood waste. 2016. **Proceedings of the 2016 IEEE 21st International Conference on Emerging Technologies and Factory Automation (ETFA)** [...]. [S. l.: s. n.], 2016. <https://doi.org/10.1109/ETFA.2016.7733546>.

WALTHER, T.; TERZIĆ, K.; DONATH, T.; MEINE, H.; BECKMANN, F.; THOEMEN, H. Microstructural analysis of lignocellulosic fiber networks. 2006. **Proceedings**

of the Developments in X-Ray Tomography V [...]. [S. l.]: SPIE, 2006. v. 6318. <https://doi.org/10.1117/12.679050>.

WANG, Z.; ZHUANG, Z.; LIU, Y.; DING, F.; TANG, M. Color classification and texture recognition system of solid wood panels. **Forests**, v. 12, p. 1154, 2021. <https://doi.org/10.3390/f12091154>.

WATANABE, K.; LĂZĂRESCU, C.; SHIDA, S.; AVRAMIDIS, S. A Novel Method of Measuring Moisture Content Distribution in Timber During Drying Using CT Scanning and Image Processing Techniques. **Drying Technology**, v. 30, p. 256-262, 2012. <https://doi.org/10.1080/07373937.2011.634977>.

WEI, Q.; LEBLON, B.; LA ROCQUE, A. On the use of X-ray computed tomography for determining wood properties: A review 1. **Canadian Journal of Forest Research**, v. 41, p. 2120-2140, 2011. <https://doi.org/10.1139/x11-111>.

WIBERG, P.; MOREN, T.J. Moisture flux determination in wood during drying above fibre saturation point using CT-scanning and digital image processing. **European Journal of Wood and Wood Products**, v. 57, p. 137-144, 1999. <https://doi.org/10.1007/s001070050029>.

WIMMER, G.; SCHRAML, R.; HOFBAUER, H.; PETUTSCHNIGG, A.; UHL, A. Two-Stage CNN-Based Wood Log Recognition. 2021. **Proceedings of the International Conference on Advanced Concepts for Intelligent Vision Systems (ACIVS) [...]**. [S. l.: s. n.], 2021. p. 115-125. https://doi.org/10.1007/978-3-030-87007-2_9.

WU, F.; GAZO, R.; HAVIAROVA, E.; BENES, B. Wood identification based on longitudinal section images by using deep learning. **Wood Science and Technology**, v. 55, p. 553-580, 2021. <https://doi.org/10.1007/s00226-021-01261-1>.

WU, K.; KILIAN, A. Designing Natural Wood Log Structures with Stochastic Assembly and Deep Learning. In: **Robotic Fabrication in Architecture, Art and Design**. [S. l.]: Springer, 2018. p. 16-30. https://doi.org/10.1007/978-3-319-92294-2_2.

XIE, G.; WANG, L.; WILLIAMS, R.A.; LI, Y.; ZHANG, P.; GU, S. Segmentation of wood CT images for internal defects detection based on CNN: A comparative study. **Computers and Electronics in Agriculture**, v. 216, p. 109244, 2024. <https://doi.org/10.1016/j.compag.2024.109244>.

XIONG, Y.; WANG, Y. Placing Nature: Interactive form-finding using computer vision and robotic arm collaboration for natural timber structure. 2025. **Proceedings of the International Conference on Computer-Aided Architectural Design Research in Asia [...]**. [S. l.: s. n.], 2025. p. 295-304. <https://doi.org/10.52842/conf.caadria.2025.2.295>.

YANG, H.; CHEN, Y.; SONG, K.; YIN, Z. Multiscale Feature-Clustering-Based Fully Convolutional Autoencoder for Fast Accurate Visual Inspection of Texture Surface Defects. **IEEE Transactions on Automation Science and Engineering**, v. 16, p. 1450-1467, 2019. <https://doi.org/10.1109/TASE.2018.2886031>.

YANG, Y.; WANG, H.; JIANG, D.; HU, Z. Surface detection of solid wood defects based on SSD improved with ResNet. **Forests**, v. 12, p. 1419, 2021. <https://doi.org/10.3390/f12101419>.

YE, R.; PEI, Y.; WANG, W.; ZHOU, H. Scientific Computational Visual Analysis of Wood Internal Defects Detection in View of Tomography Image Reconstruction Algorithm. **Mobile Information Systems**, v. 2022, p. 6091352, 2022. <https://doi.org/10.1155/2022/6091352>.

ZAMRI, M.I.P.; CORDOVA, F.; MOHD KHAIRUDDIN, A.S.M.; MOKHTAR, N.; YU-

SOF, R. Tree species classification based on image analysis using Improved-Basic Gray Level Aura Matrix. **Computers and Electronics in Agriculture**, v. 124, p. 227-233, 2016. <https://doi.org/10.1016/j.compag.2016.04.004>.

ZHU, D.; CONNERS, R.; SCHMOLDT, D.L.; ARAMAN, P.A. A prototype vision system for analyzing CT imagery of hardwood logs. **IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics**, v. 26, p. 522-532, 1996. <https://doi.org/10.1109/3477.517028>.

CAPÍTULO III

DETECÇÃO E CONTAGEM AUTOMÁTICA DE MADEIRA DE EUCALIPTO EMPILHADA USANDO O MODELO YOLOv8

RESUMO

A quantificação do número de toras em pilhas de madeira é operação demorada, onerosa e sujeita a erros, ainda realizada de forma predominantemente manual na indústria florestal, o que limita a frequência e a abrangência dos controles de estoque. Diante disso, este estudo teve por objetivo avaliar a viabilidade de automatizar a detecção e a contagem de toras empilhadas de eucalipto (*Eucalyptus urophylla* × *Eucalyptus grandis*) a partir de imagens, comparando o desempenho obtido com fotografias e com vídeo. Para tanto, empregou-se o modelo YOLOv8. O conjunto de dados incluiu 230 imagens diversas de madeira roliça de eucalipto, abrangendo toras separadas em trilhos e madeira empilhada, anotadas com o LabelImg para uma delimitação precisa das extremidades das toras. O YOLOv8 foi personalizado com uma arquitetura CSPDarknet53, módulo C2f e camada SPPF para maior eficiência computacional. O treino utilizou o otimizador AdamW no Ultralytics YOLOv8.0.137 (Python 3.10.12, Torch 2.0.1 + CUDA 11.8) numa GPU NVIDIA T1000 (4096 MiB). O desempenho do modelo foi avaliado através da precisão, recall e média da precisão média a 50 % IoU (mAP50). Os melhores resultados foram obtidos na época 261, com precisão/recall/mAP50 de 0,814/0,812/0,844 no treino e 0,778/0,798/0,839 na validação. Os testes de generalização em imagens não vistas confirmaram detecção robusta em toras espaçadas, dispersas e sobrepostas. No entanto, a contagem automática a partir de imagens fixas mostrou-se menos confiável para madeira empilhada na configuração padrão de inferência, enquanto a contagem baseada em vídeo apresentou erro percentual relativo de +1,561%, sem viés estatisticamente detectável. Uma análise de sensibilidade posterior mostrou que essa diferença decorre em grande parte do limiar de confiança: reduzindo-o de 0,25 para 0,10 e fundindo as caixas coincidentes, a revocação em fotografia sobe de 59,9% para 90,4% sem introduzir detecções fora da madeira. Estes resultados indicam que o vídeo é preferível para a quantificação do total de uma pilha, enquanto as fotografias, com configuração ajustada, são adequadas à mensuração de trechos delimitados e à contagem de peças individuais de madeira roliça.

Palavras-chave: visão computacional; aprendizagem profunda; detecção de objetos; indústria florestal; contagem automática

3.1. INTRODUÇÃO

A quantificação de madeira é essencial na silvicultura e na indústria madeireira. O volume de madeira é um fator-chave na precificação, sendo importante dispor de um sistema de medição que atenda às condições exigidas e que também seja eficiente (PÁSZTORY; HEINZMANN; BARBU, 2018). A estimativa precisa do volume de madeira é crucial no setor, especialmente para madeira industrial. O volume é frequentemente estimado em metros cúbicos empilhados, e a precisão dessas estimativas afeta vários aspectos do manejo e utilização da madeira (CREMER et al., 2020), incluindo faturamento e estimativa de custos. Estimativas imprecisas podem levar a consequências econômicas significativas (PURFÜRST et al., 2023).

Vários métodos são empregados para quantificar o volume de pilhas. A técnica de medição manual envolve medir o comprimento, a altura e a profundidade da madeira empilhada para determinar o volume estéreo (HUSCH; MILLER; BEERS, 1993). Outra abordagem consiste em imergir toras em água para aproximar as irregularidades da superfície da pilha; esse método é comumente chamado de xilometria (ANTUNES SANTANA; IMAÑA ENCINAS; RIESCO MUÑOZ, 2023). A medição do Peso Absoluto Total Residual Seco em Estufa (ATRO) é amplamente utilizada na indústria madeireira para determinar o teor de umidade de uma amostra, expresso como percentual do peso seco final da amostra. Essa medição é fundamental para avaliar e comparar com precisão a densidade de vários tipos de madeira, e possui aplicações práticas no setor florestal (GLASS; ZELINKA; JOHNSON, 2014; KOMAN; FEHER, 2015; WATANABE; KOBAYASHI; KURODA, 2012).

Além disso, as medições podem ser obtidas pesando a carga em caminhões (CARVALHO; CAMARGO, 1996); digitalizando cargas de caminhões com lasers (KUNICKAYA et al., 2022; NYLINDER; KUBÉNKA; HULTNÄS, 2008); utilizando softwares para avaliação digital de imagens (CAMPOS; LEITE, 2017); empregando expressões matemáticas; e utilizando fatores de conversão (SOARES; PAULA NETO. F.; SOUZA, 2011). O uso da tecnologia de câmeras estéreo, especificamente o LogStackPro, mostrou-se promissor como uma alternativa precisa aos métodos manuais tradicionais para medir volumes de madeira em transações comerciais. Embora comparações estatísticas não tenham encontrado diferenças significativas entre as técnicas de medição avaliadas, considerações práticas enfatizam o potencial impacto financeiro das discrepâncias de volume (MEDERSKI et al., 2023). Além disso, métodos baseados em LiDAR são uma tendência atualmente; é possível que esses métodos complementem ou até substituam as técnicas tradicionais de medição no futuro (LEITE et al., 2020; PU et al., 2023; PURFÜRST et al., 2023).

Um dos maiores desafios no setor industrial é quantificar com precisão a madeira empilhada, particularmente no caso da madeira de eucalipto usada para produção de celulose. Softwares e aplicativos móveis específicos foram desenvolvidos para lidar com esses desafios na medição do volume de madeira. Esses aplicativos utilizam diversas técnicas para obter resultados precisos e consistentes. O Digtora (BERTOLA et al., 2003) emprega o método dos pontos equidistantes

para garantir precisão e consistência nas medições. Esse método envolve a colocação de uma série de pontos igualmente espaçados sobre a madeira e a medição das distâncias entre eles, calculando-se então o volume com base nessas medições.

O NeuroDIC (SILVEIRA, 2014) utiliza modelos de redes neurais artificiais (RNA) para classificar imagens de madeira e seu espaçamento. Uma vez treinada, a RNA pode ser usada para obter um fator de conversão para estimar o volume de madeira empilhada. O Trestima Stack (KÄRHÄ et al., 2019) emprega visão computacional para determinar com precisão o volume da madeira com base em imagens capturadas por smartphone ou tablet. O volume real é calculado com base na área da superfície da pilha, no comprimento da tora e em um fator de coeficiente gerado automaticamente.

Além disso, softwares como o IFOVEA (<https://fovea.eu/index.php>, acesso em 12 de julho de 2023) e o Timbeter (<https://timbeter.com/>, acesso em 12 de julho de 2023) (MOSKALIK et al., 2022) foram desenvolvidos, permitindo que os usuários obtenham vistas panorâmicas de múltiplas imagens e quantifiquem a madeira empilhada por meio de análise de imagens. Apesar da existência dessa tecnologia, e da necessidade de quantificar o volume sólido da madeira em pilhas, ou em caminhões de transporte para pequenos produtores e empresas florestais de médio e grande porte, a automação total ainda é um objetivo pendente.

Quando as toras de uma pilha têm o mesmo comprimento, ou variação relativamente pequena, a distribuição dos seus diâmetros pode ser usada para calcular o volume sólido. Para isso, é necessário conhecer o número de toras na pilha. A contagem manual usando a própria pilha ou uma imagem dela não é uma boa opção em escala comercial, exceto para pilhas muito pequenas. A alternativa é automatizar o processo de contagem de toras usando uma solução de visão computacional, que exige um modelo de detecção de objetos, neste caso, para toras de madeira.

Existem, entretanto, soluções tecnológicas para lidar com esse problema. Uma dessas abordagens envolve aproveitar o poder do aprendizado profundo (deep learning) com modelos YOLO (You Only Look Once) (REDMON et al., 2016), projetados especificamente para detecção de objetos em tempo real e que utilizam CNNs profundas em diversas aplicações, como robótica, veículos autônomos e vigilância por vídeo. Atualmente, existem oito versões do modelo YOLO. O YOLOv8 (JOCHER; CHAURASIA; QIU, 2023) representa o estado da arte em modelos de visão computacional e foi desenvolvido pela Ultralytics em janeiro de 2023. Esse modelo oferece suporte pronto para diversas tarefas, incluindo detecção de objetos, classificação e segmentação, sendo acessível tanto por pacote Python quanto por interface de linha de comando.

O modelo YOLO mostrou-se altamente eficaz na detecção precisa e automatizada de madeira. Ele já foi aplicado com sucesso em tarefas como detecção de nós superficiais em madeira serrada (FANG et al., 2021); extração de características de contorno em texturas complexas, como trincas em estruturas de madeira (LIU, Y. et al., 2020; MA et al., 2022); e identificação de defeitos na madeira com notável precisão (CUI; LU; LIU, 2023;

KURDTHONGMEE, 2023).

A visão computacional é um campo da ciência da computação que trata da extração de informações significativas a partir de imagens ou vídeos digitais. É utilizada em uma ampla variedade de áreas, incluindo reconhecimento de objetos, vigilância, controle robótico, imagens médicas e manufatura (DAVIES, 2022).

A inteligência artificial (IA) aplicada à visão computacional é inspirada no funcionamento do cérebro humano. Um dos principais conhecimentos advindos da neurociência é que o cérebro humano possui duas vias distintas de processamento visual: a via ventral e a via dorsal. A via ventral é responsável pela identificação e reconhecimento de objetos. Ela começa no lobo occipital e se projeta para o lobo temporal, onde atua no processamento das características dos objetos, como forma, cor e textura. A via dorsal é responsável por processar a localização espacial dos objetos. Ela começa no lobo occipital e se projeta para o lobo parietal, onde atua no rastreamento do movimento dos objetos e de sua relação com o corpo (GOODALE; MILNER, 1992).

A visão humana é frequentemente influenciada pela tarefa específica que estamos tentando realizar. É, portanto, importante que os modelos visuais sejam capazes de interpretar estímulos visuais de forma relevante para a tarefa em questão. Uma maneira de alcançar isso é treinar os modelos com dados anotados especificamente para a tarefa que se espera que o modelo execute. Ao considerar a tarefa específica, podemos treinar modelos visuais para serem mais eficientes e precisos (ZHU; WU, 2023).

As redes neurais convolucionais profundas (CNNs) revolucionaram a classificação de imagens e levaram à ampla adoção de redes neurais profundas em aplicações de visão computacional por empresas como Google, Facebook, Microsoft e Baidu (KRIZHEVSKY; SUTSKEVER; HINTON, 2017). Essas redes são capazes de aprender características hierárquicas das imagens, desde características de baixo nível, como bordas e formas, até características de alto nível, como objetos e cenas (ZEILER; FERGUS, 2014). A classificação de imagens é uma tarefa de visão computacional que visa identificar e categorizar objetos em uma imagem. Isso é feito analisando as propriedades numéricas de várias características da imagem, como cor, textura e forma, que são então organizadas em categorias (GAVALI; BANU, 2019).

A visão computacional emergiu como uma tecnologia de destaque com várias aplicações nas ciências, oferecendo grande potencial na automação de processos tanto no ambiente industrial quanto na pesquisa básica (VOULODIMOS et al., 2018). Pesquisadores florestais têm utilizado IA na silvicultura para detectar defeitos e texturas na superfície da madeira. Uma dessas técnicas é a projeção integral, que pode ser usada para traçar e localizar a posição de uma placa de madeira. Essa técnica foi testada em cinco tipos de amostras e apresentou alta precisão média de classificação (ZHANG et al., 2015). Outro estudo desenvolveu um algoritmo robusto e de baixo custo para detectar defeitos em madeira. Esse algoritmo pode ser aplicado em uma máquina de otimização automática de madeira, ajudando a melhorar a qualidade e eficiência do processamento (CAVALIN et al., 2006).

Nosso principal objetivo foi aproveitar técnicas avançadas de visão computacional, em especial o modelo de última geração YOLOv8, para automatizar o processo de detecção e contagem de madeira de eucalipto empilhada (híbrido *Eucalyptus urophylla* × *Eucalyptus grandis*). Esse avanço na visão computacional não apenas beneficia o setor florestal, mas também abre caminho para uma integração tecnológica mais ampla na indústria. Esta pesquisa oferece uma aplicação de detecção automática que pode ser usada como solução alternativa para a indústria florestal, otimizando processos de produção e enfrentando os desafios da quantificação do volume sólido da madeira empilhada. Além disso, esses avanços abrem caminho para a integração de tecnologias de visão computacional de ponta em diversas áreas de pesquisa florestal, prometendo um futuro tecnologicamente avançado para o setor.

3.2. MATERIAIS E MÉTODOS

3.2.1. Conjunto de dados

O conjunto de dados é composto por fotografias obtidas de uma organização florestal em Minas Gerais, Brasil. Para o treinamento do modelo YOLOv8, foram utilizadas dois tipos de imagens: aquelas de toras de eucalipto separadas e instaladas sobre um trilho, e aquelas de madeira de eucalipto empilhada (Figura 3.1). Ao todo, foram utilizadas 230 imagens. A madeira empilhada apresenta diâmetro variando de 4 a 31 cm e é proveniente de um inventário pré-corte com idade entre 6 e 7 anos.



Figura 3.1. O banco de dados é composto por um conjunto de dois tipos de imagens utilizadas para treinar o modelo YOLOv8.

O equipamento fotográfico utilizado foi um smartphone iPhone 13 (fabricado pela Apple na

planta da Foxconn em Tamil Nadu, Índia), com abertura $f/1.6$, tempo de exposição de $1/173$ s, ISO 50, distância focal de 5 mm, resolução de 72 dpi e dimensões de 1536×2048 pixels. As fotografias foram capturadas com um ângulo de visão de 60° em condições ambientais favoráveis, especificamente sem ocorrência de chuva.

Durante o processo de anotação, foi dada atenção especial à delimitação precisa dos objetos-alvo nas superfícies das toras, garantindo a eficácia do modelo na detecção e contagem desses objetos em cenários reais. A inclusão dessas imagens diversificadas de toras aumenta a capacidade de generalização do modelo e melhora seu desempenho no reconhecimento de diferentes configurações de toras na madeira de eucalipto empilhada.

3.2.2. Aquisição e contagem em vídeo

Além das fotografias, foi utilizado um registro em vídeo para a contagem de toras em uma pilha industrial completa, com 47,4 m de comprimento. A aquisição foi realizada por meio de uma varredura contínua ao longo da pilha: o operador deslocou-se paralelamente à face da pilha, mantendo a câmera voltada para as faces das toras, de modo que a filmagem percorresse toda a extensão da pilha de uma extremidade à outra. Esse procedimento faz com que cada tora seja registrada em vários quadros consecutivos e sob diferentes ângulos de visada, o que amplia o conjunto de faces efetivamente visíveis em relação ao que se obtém em uma única fotografia estática.

Como consequência direta desse procedimento, a contagem em vídeo não pode ser obtida pela simples soma das detecções realizadas em cada quadro, uma vez que a mesma tora aparece em dezenas de quadros sucessivos. Para evitar a duplicação, a detecção foi acoplada ao algoritmo de rastreamento BoT-SORT (AHARON; ORFAIG; BOBROVSKY, 2022), que associa as detecções entre quadros consecutivos e atribui a cada tora um identificador único, mantido enquanto o objeto permanece visível na cena. A contagem total do vídeo corresponde, portanto, ao número de identificadores únicos gerados ao longo de toda a filmagem, e não ao número de detecções em um quadro isolado. Esse mesmo mecanismo explica por que o número máximo de detecções por imagem não restringe a contagem obtida em vídeo, conforme discutido na análise de sensibilidade.

O parâmetro determinante nesse procedimento é o tempo durante o qual o rastreador mantém ativo um identificador cujo objeto deixou de ser detectado (`track_buffer`). Seu dimensionamento deve corresponder ao tempo em que cada tora permanece no campo de visão. A velocidade da varredura foi medida por correlação de fase entre quadros consecutivos e resultou em 0,10 largura de quadro por segundo, de modo que uma mesma tora permanece visível por aproximadamente 10 s, ou 300 quadros a 30 quadros por segundo. O valor padrão das implementações, de 30 quadros, equivale a apenas 1 s e é, portanto, insuficiente para esta aquisição: qualquer oclusão momentânea provoca a perda do identificador e a criação de um novo, inflando a contagem. Adotou-se, por essa razão, `track_buffer` igual a 250 quadros, da mesma ordem de grandeza do tempo de permanência medido. Os demais parâmetros foram mantidos nos val-

ores empregados para as fotografias, isto é, limiar de confiança de 0,25, IoU de 0,7 e resolução de entrada de 640 pixels. A sensibilidade da contagem a essa escolha é apresentada na seção de resultados.

3.2.3. Formato de Anotação e Divisão do Conjunto de Dados

A anotação é registrada em um arquivo de texto no qual cada linha descreve uma caixa por meio de cinco valores. O primeiro (Object-Class) identifica a classe do objeto; o segundo e o terceiro (x_n e y_n , Equações 3.1 e 3.2) são as coordenadas do centro da caixa, normalizadas pela largura e pela altura da imagem; o quarto e o quinto (w_n e h_n , Equações 3.3 e 3.4) são a largura e a altura da caixa, normalizadas da mesma forma. Cabe esclarecer que esses valores são *extraídos* da caixa desenhada pelo anotador e gravados no arquivo de rótulos, e não parâmetros informados para produzir a anotação. O processo de anotação foi realizado utilizando o software LabelImg (TZUTALIN, 2015). A anotação foi feita por meio de caixas delimitadoras retangulares, alinhadas aos eixos da imagem, traçadas de modo a circunscrever exclusivamente a face, isto é, a seção transversal exposta de cada tora. Não foram anotados o corpo longitudinal da tora nem regiões adjacentes, de forma que cada caixa corresponde a uma única face. Como a face é aproximadamente circular e a caixa é retangular, os quatro cantos da caixa incluem necessariamente pequenas porções de fundo ou de faces vizinhas; trata-se de uma característica inerente ao uso de caixas delimitadoras, e não de imprecisão na anotação. Adotou-se ainda o critério de anotar somente as faces efetivamente visíveis na fotografia: faces ocultas por outras toras, por galhos ou por resíduos não foram anotadas, ainda que a tora estivesse fisicamente presente na pilha. Em consequência, o modelo é treinado para detectar as faces visíveis na imagem, e não o número total de toras que compõem a pilha, o que deve ser considerado na interpretação das contagens obtidas em pilhas com elevado grau de oclusão.

O conjunto de dados foi dividido em subconjuntos de treinamento, validação e teste, com distribuição de 80%, 10% e 10%, respectivamente. A divisão foi realizada por fotografia, e não por anotação: das 230 imagens, 184 foram destinadas ao treinamento, 23 à validação e 23 ao teste, de modo que todas as anotações de uma mesma fotografia permanecessem em um único subconjunto. Esse critério evita que anotações de uma mesma fotografia sejam distribuídas entre subconjuntos distintos, o que superestimaria o desempenho avaliado. Cada imagem de madeira empilhada possuía entre 50 e 150 anotações, e todas as faces das toras foram anotadas nas imagens de toras separadas. No total, foram adicionadas 6880 anotações (box labels). A largura das *bounding boxes*, expressa em valores normalizados pela largura da imagem, variou de 0,008464 a 0,121745, com média de 0,035087, o que corresponde, em imagens de 1536 pixels de largura, a caixas de 13,0 a 187,0 pixels, com média de 53,9 pixels. A altura, normalizada pela altura da imagem, variou de 0,006348 a 0,142333, com média de 0,037198, equivalente a caixas de 13,0 a 291,5 pixels, com média de 76,2 pixels, em imagens de 2048 pixels de altura.

Cada imagem foi movida para o subconjunto correspondente junto com o seu arquivo de rótulos, garantindo a correspondência entre imagem e anotação. A Figura 3.2 apresenta um

exemplo ilustrativo das anotações utilizadas neste estudo.

$$x_n = \frac{x_c}{W_{\text{imagem}}} \quad (3.1)$$

$$y_n = \frac{y_c}{H_{\text{imagem}}} \quad (3.2)$$

$$w_n = \frac{W_{\text{caixa}}}{W_{\text{imagem}}} \quad (3.3)$$

$$h_n = \frac{H_{\text{caixa}}}{H_{\text{imagem}}} \quad (3.4)$$

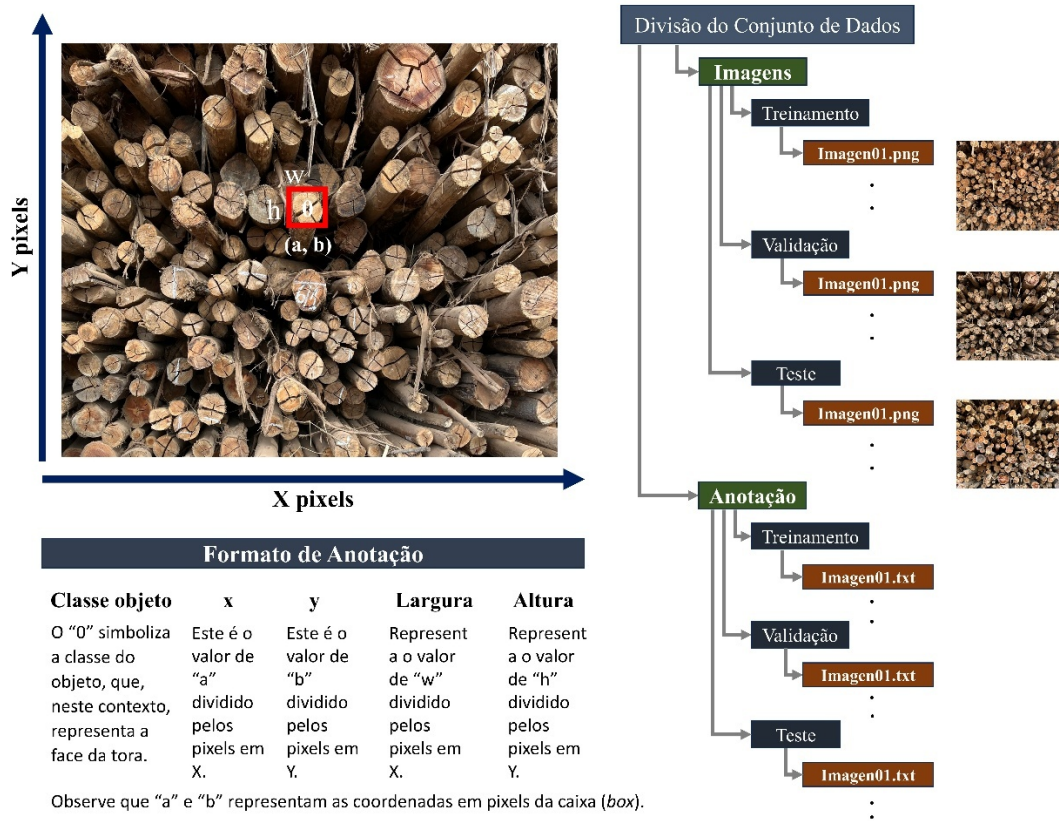


Figura 3.2. Formato de anotação e divisão do conjunto de dados em uma imagem de madeira de eucalipto empilhada.

3.2.4. Arquitetura e Configuração do Modelo

O projeto incorpora um backbone CSPDarknet53 personalizado em sua arquitetura. Também utiliza o módulo C2f (cross-stage partial bottleneck with two convolutions). Para acelerar o processamento, é empregada uma camada spatial pyramid pooling fast (SPPF), que permite o agrupamento de recursos em um mapa de tamanho fixo. Além disso, cada convolução é acompanhada por normalização em lote (batch normalization) e ativação SiLU (Sigmoid Linear Unit).

Notavelmente, a head do modelo é desacoplada para lidar de forma independente com tarefas orientadas a objetos. O design da arquitetura é descrito em “Comprehensive Review of YOLO: From YOLOv1 to YOLOv8 and Beyond” (TERVEN; CORDOVA-ESPARZA, 2023).

O YOLOv8 oferece cinco versões escalonadas: YOLOv8n (nano), YOLOv8s (small), YOLOv8m (medium), YOLOv8l (large) e YOLOv8x (extra-large). Utilizamos o YOLOv8n, que é ideal para personalizar um novo modelo e detectar objetos muito pequenos.

Foi utilizado o otimizador AdamW com grupos de parâmetros específicos para o treinamento do modelo YOLOv8. A configuração empregada foi AdamW ($\text{lr} = 0,002$, $\text{momentum} = 0,9$), com os seguintes valores de weight decay para os respectivos parâmetros: 57 weight ($\text{decay} = 0,0$), 64 weight ($\text{decay} = 0,0005$) e 63 bias ($\text{decay} = 0,0$). As imagens originais, com 1536×2048 pixels, foram redimensionadas para o treinamento e a validação de modo que o maior lado passasse a 640 pixels, preservando-se a razão de aspecto; o preenchimento das bordas (*letterbox*) completou a imagem até a resolução quadrada de 640×640 pixels adotada pela rede. Em decorrência desse redimensionamento, a face de uma tora, que ocupa em média $53,9 \times 76,2$ pixels na imagem original, passa a ocupar cerca de $16,8 \times 23,8$ pixels na imagem de entrada, o que caracteriza a tarefa como detecção de objetos pequenos e ajuda a explicar as limitações de revocação observadas em pilhas densas.

O modelo YOLOv8 foi implementado utilizando Ultralytics YOLOv8.0.137, Python-3.10.12 e torch-2.0.1 + cu118 com suporte a CUDA em NVIDIA T1000 (4096 MiB). O treinamento foi configurado com limite máximo de 3000 épocas e tolerância de 300 épocas para a parada antecipada, encerrando-se antes desse limite. Cada batch size foi definido como 2. O sistema também foi configurado com o BoT-SORT (Robust Associations Multi-Pedestrian Tracking) (AHARON; ORFAIG; BOBROVSKY, 2022), um algoritmo de rastreamento baseado em regras projetado para ser robusto frente a colisões e obstruções, tornando-o ideal para aplicações de contagem em vídeo.

Na etapa de generalização, a inferência sobre as fotografias foi executada com os valores padrão da implementação utilizada: limiar de confiança de 0,25, limiar de *intersection over union* (IoU) de 0,7, resolução de entrada de 640 pixels no maior lado e número máximo de detecções por imagem (max_det) igual a 300. Como as pilhas industriais avaliadas contêm mais de 300 toras por fotografia, esse último parâmetro trunca a contagem; as contagens apresentadas na Tabela 3.2 foram, por isso, obtidas sem essa restrição. A sensibilidade da contagem aos três parâmetros é avaliada na seção de resultados.

3.2.5. Procedimento de generalização

A generalização do modelo foi validada utilizando fotografias totalmente distintas dos conjuntos de treinamento, validação e teste. O melhor modelo obtido foi então aplicado a dois tipos distintos de imagens: toras dispostas sobre um trilho e madeira empilhada. Para representar claramente as toras dispostas no trilho, as imagens foram segmentadas de forma a exibir apenas a superfície frontal de cada tora.

Em seguida, aplicamos o modelo à madeira empilhada na indústria florestal. É essencial considerar que, em um negócio voltado para a produção de celulose, a madeira empilhada não apresenta um arranjo definido e é composta por uma mistura de diâmetros grandes, médios e pequenos. Para avaliar o desempenho do modelo, ele foi aplicado a dez fotografias de uma mesma pilha industrial, tomadas em seções contíguas e não sobrepostas ao longo de toda a sua extensão, de modo a cobrir a pilha integralmente sem repetir trechos já registrados. Para cada imagem, realizamos uma contagem manual cuidadosa para obter a contagem observada, que foi posteriormente comparada com a contagem automática gerada pelo modelo.

3.2.6. Medidas de desempenho

A avaliação do desempenho foi realizada utilizando três métricas: Precisão (P) (Equação (3.5)), Revocação (*recall*, R) (Equação (3.6)) e *mean Average Precision* a um limiar de *intersection over union* (IoU) de 50% (mAP50). Essas métricas são essenciais para avaliar o desempenho do modelo e sua capacidade de identificar e classificar corretamente objetos ou instâncias no conjunto de dados fornecido.

O cálculo do mAP50 envolve obter a curva de precisão-revocação considerando como detecção correta toda predição cuja sobreposição com a caixa de referência seja de pelo menos 50% de IoU e, em seguida, calcular a média das precisões médias entre as diferentes categorias ou classes de objetos. O valor final do mAP50 representa a precisão média em todas as classes ao limiar de IoU de 50%.

$$\text{Precisão} = \frac{TP}{TP + FP} \quad (3.5)$$

$$\text{Revocação} = \frac{TP}{TP + FN} \quad (3.6)$$

Em que:

TP representa os True Positives, o número de previsões positivas corretas feitas pelo modelo; FP representa os False Positives, o número de previsões positivas incorretas feitas pelo modelo; FN representa os False Negatives, o número de instâncias positivas que não foram corretamente previstas pelo modelo.

O modelo foi generalizado utilizando imagens e vídeos. Cada imagem e vídeo testado foi contado manualmente para obter a contagem observada. A contagem manual seguiu o mesmo critério adotado na anotação: foram contabilizadas apenas as faces de toras efetivamente visíveis no arquivo avaliado, e não a totalidade das toras que compõem a pilha. Dessa forma, a contagem observada e a contagem estimada referem-se à mesma grandeza, e as diferenças entre elas expressam exclusivamente falhas de detecção do modelo, sem qualquer componente atribuível a divergência de definição entre os dois métodos de contagem. O erro (E) foi calculado usando o valor absoluto da diferença entre a contagem estimada e a contagem observada. A porcentagem de erro relativo (RE%) também foi calculada dividindo-se a diferença entre a contagem

estimada e a contagem observada pela contagem observada.

3.3. RESULTADOS

3.3.1. Desempenho do Modelo

O processo de treinamento foi interrompido antecipadamente, pois não houve melhorias observáveis nas últimas 300 épocas. Os melhores resultados foram obtidos na época 261. Ao longo do treinamento, um total de 561 épocas foi concluído, levando aproximadamente 12,35 horas para finalizar. O modelo YOLOv8 utilizado no processo de treinamento é composto por 168 camadas e um total de 3.005.843 parâmetros.

As métricas de desempenho do modelo treinado para precisão, revocação (recall) e mean average precision (mAP50) foram avaliadas (Tabela 3.1). No conjunto de treinamento, o modelo alcançou precisão de 0,814, revocação de 0,812 e mAP50 de 0,844. No conjunto de validação, o modelo alcançou precisão de 0,778, revocação de 0,798 e mAP50 de 0,839. No conjunto de teste, o modelo alcançou precisão de 0,741, revocação de 0,779 e mAP50 de 0,799. Os maiores valores foram observados no conjunto de treinamento, seguidos pelos da validação e, por fim, pelos do teste, ordenação esperada e compatível com a ausência de sobreajuste acentuado.

Tabela 3.1. Métricas de desempenho avaliadas na divisão dos dados de madeira de eucalipto empilhada.

Divisão dos Dados	Precisão	Revocação	mAP50
Treinamento	0,814	0,812	0,844
Validação	0,778	0,798	0,839
Teste	0,741	0,779	0,799

A Figura 3.3 ilustra os resultados de quatro curvas de desempenho distintas para um modelo de visão computacional: a curva F1-confiança, a curva precisão-confiança, a curva precisão-revocação (precision-recall) e a curva revocação-confiança (recall-confidence). Cada curva representa como diferentes métricas de desempenho se relacionam com a confiança da predição do modelo nos conjuntos de treinamento, validação e teste.

Os resultados mostram que o modelo apresenta bom desempenho tanto em precisão quanto em revocação, com um valor F1 de 0,81 no conjunto de treinamento. Entretanto, o valor F1 é ligeiramente menor nos conjuntos de validação (0,79) e teste (0,76).

A curva precisão-confiança mostra que o modelo é muito preciso em limiares de confiança elevados; entretanto, a precisão é ligeiramente menor em limiares mais baixos. A curva precisão-revocação indica que o modelo está alcançando uma alta taxa de revocação, o que significa que está identificando uma alta proporção de verdadeiros positivos. A curva revocação-confiança mostra o comportamento oposto: a revocação é mais alta nos limiares baixos e diminui à medida que o limiar aumenta, uma vez que um número crescente de detecções passa a ser

descartado.

No geral, os resultados indicam que o modelo apresenta bom desempenho nos conjuntos de treinamento e validação; contudo, o desempenho no conjunto de teste é ligeiramente inferior.

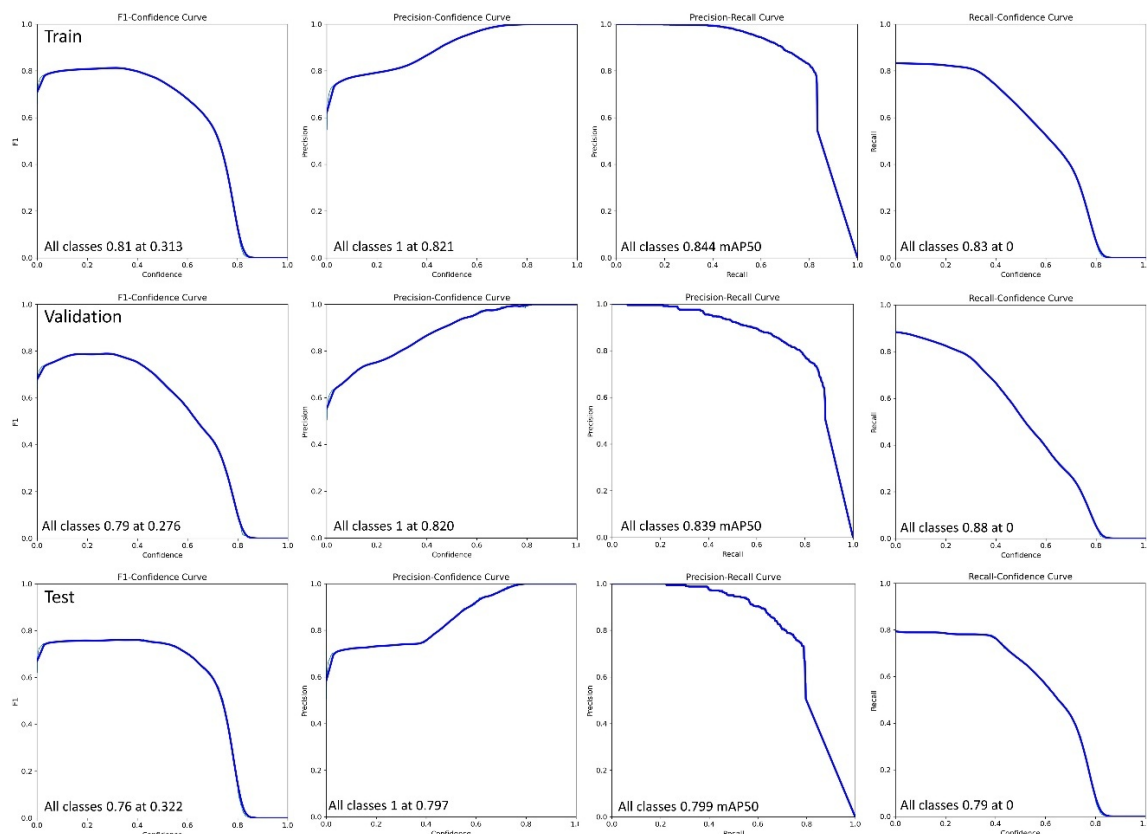


Figura 3.3. Avaliação das diferentes curvas de desempenho do modelo treinado, incluindo a curva F1-confiança, a curva precisão-confiança, a curva precisão-revocação e a curva revocação-confiança.

3.3.2. Generalização do Modelo

A aplicação do modelo às fotografias de toras dispostas sobre o trilho gerou, para cada imagem, um relatório com as anotações de todos os objetos detectados, do qual resulta a contagem automática. Os cenários englobaram uma variedade de configurações de toras, incluindo toras amplamente espaçadas com diâmetros grandes (Figura 3.4A) e diâmetros médios (Figura 3.4E), toras dispersas com diâmetros pequenos (Figura 3.4B), bem como toras espaçadas e sobrepostas com diâmetros menores (Figura 3.4C,D).

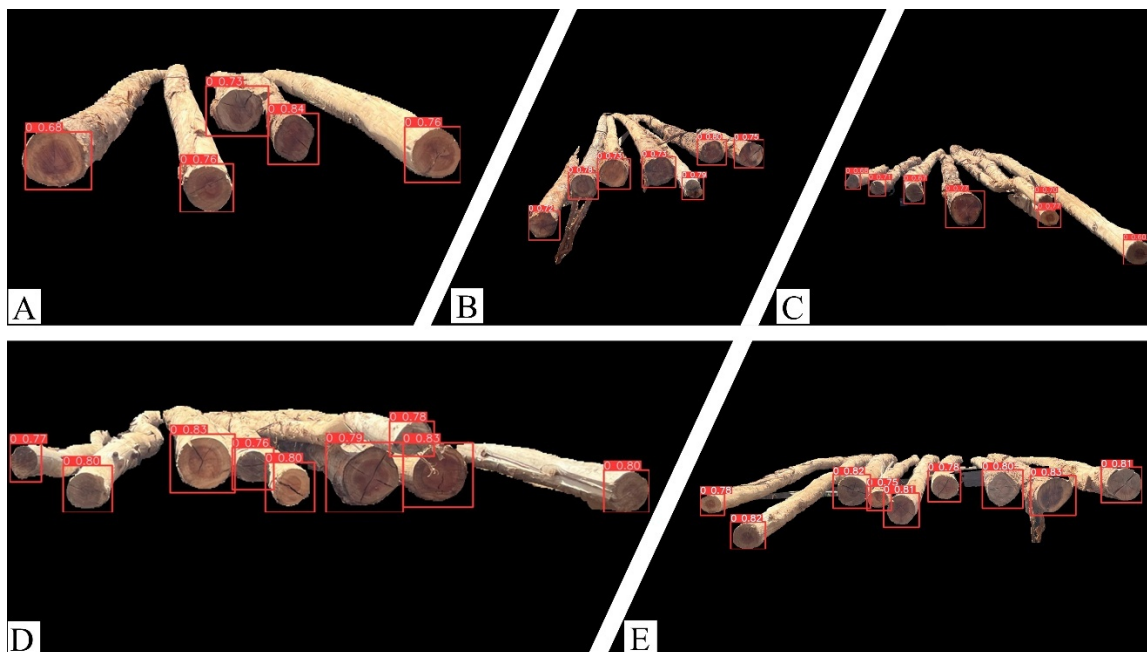


Figura 3.4. Imagens utilizadas na generalização do modelo, classificadas como toras espaçadas com diâmetros grandes (A) e diâmetros médios (E), toras dispersas com diâmetros pequenos (B), bem como toras espaçadas e sobrepostas com diâmetros menores (C,D).

Nesses arranjos, em que as toras se encontram separadas e com as faces integralmente visíveis, a contagem automática coincidiu com a contagem manual, sem detecções omitidas ou excedentes. Cabe registrar que se trata de uma avaliação qualitativa sobre um pequeno número de arranjos, cuja finalidade é verificar o comportamento do modelo em condições favoráveis; a avaliação quantitativa do desempenho é apresentada a seguir, para as fotografias de madeira empilhada.

As dez fotografias da pilha industrial, com as detecções obtidas pelo modelo, são apresentadas na Figura 3.5.

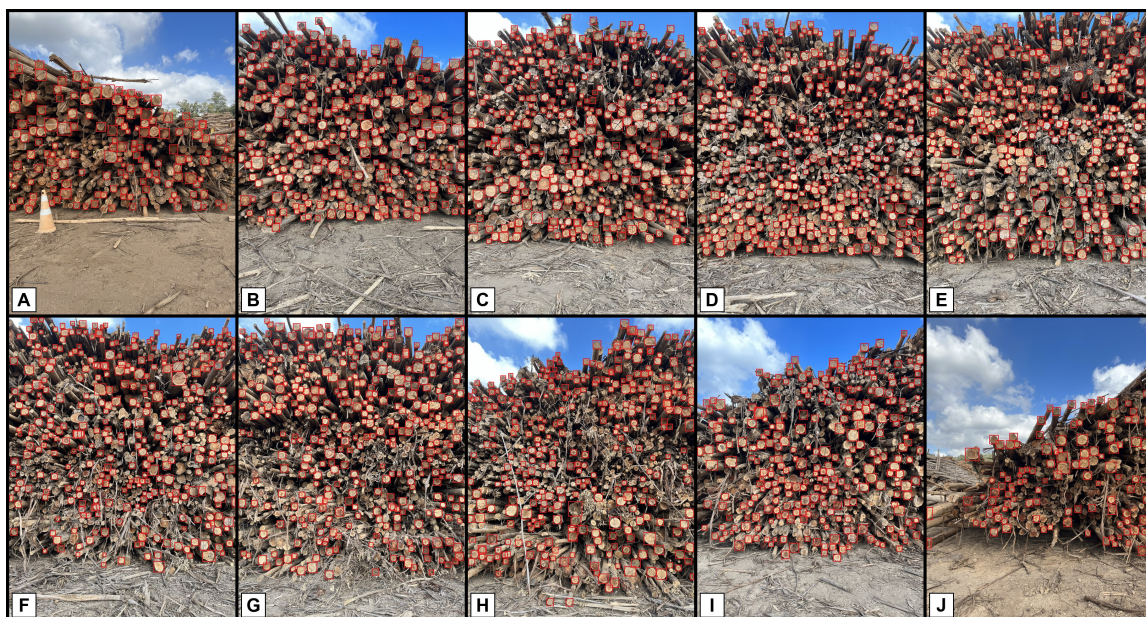


Figura 3.5. Imagens (A-J) utilizadas na generalização do modelo para a contagem e detecção de madeira empilhada na indústria de produção de celulose.

Verificou-se que a contagem observada foi consistentemente superior à contagem estimada, indicando que o modelo subestimou o valor real. O erro relativo (RE%) apresentou valores elevados para todos os arquivos de imagem, variando de $-32,252\%$ a $-47,465\%$, com média de $-39,972\%$, enquanto, para o arquivo de vídeo, o RE% foi de apenas $+1,561\%$ (Tabela 3.2). No conjunto das dez fotografias, o modelo detectou 2.819 das 4.644 faces de toras contadas manualmente, ou seja, $60,7\%$ do total, ao passo que, no vídeo, foram contabilizadas 4.816 toras contra 4.742 contadas manualmente.

Tabela 3.2. Dados dos arquivos e análises de erro para imagens e vídeo testados.

Nome do Arquivo	Tipo	Formato	Contagem Observada	Contagem Estimada	E	RE%
A	Imagem	.jpg	278	147	131	-47,122
B	Imagem	.jpg	387	260	127	-32,817
C	Imagem	.jpg	540	317	223	-41,296
D	Imagem	.jpg	580	380	200	-34,483
E	Imagem	.jpg	534	314	220	-41,199
F	Imagem	.jpg	586	334	252	-43,003
G	Imagem	.jpg	524	355	169	-32,252
H	Imagem	.jpg	546	325	221	-40,476
I	Imagem	.jpg	452	273	179	-39,602
J	Imagem	.jpg	217	114	103	-47,465
Vídeo S1	Vídeo	.mp4	4742	4816	74	1,561

O vídeo (Vídeo S1), no qual foi registrada uma pilha completa, com 47,4 m de comprimento, apresentou contagem estimada de 4816 toras contra 4742 contadas manualmente, um erro de 74 toras, equivalente a +1,561%. Diferentemente do que ocorre com as fotografias, em que o modelo subestima de forma sistemática, a contagem em vídeo não apresenta viés apreciável no total da pilha. Esses resultados fornecem informações essenciais sobre a precisão e a eficácia do modelo na automatização do processo de contagem de madeira empilhada por meio de filmagens em vídeo.

A Tabela 3.2 apresenta em detalhe o erro (E) e o erro relativo, em formato percentual (RE%), de cada uma das seções fotografadas e do vídeo.

3.3.3. Análise de sensibilidade dos parâmetros de inferência

A contagem automática não depende apenas do desempenho intrínseco do detector: ela é condicionada também pelos parâmetros adotados na etapa de inferência. Para avaliar essa dependência, realizou-se uma análise de sensibilidade sobre as mesmas dez fotografias, variando isoladamente o número máximo de detecções por imagem (*max_det*), o limiar de confiança e a resolução de entrada (Tabela 3.3). A análise partiu dos mesmos pesos treinados e da mesma configuração descrita nos Materiais e Métodos, e reproduz exatamente os dez valores publicados na Tabela 3.2 quando se adota um limite não restritivo de detecções por imagem, correspondente a *max_det* igual ou superior a 400 nas configurações avaliadas, o que assegura sua comparabilidade direta com os resultados apresentados.

Tabela 3.3. Análise de sensibilidade da contagem automática aos parâmetros de inferência, nas dez fotografias de madeira empilhada (contagem manual total: 4.644 faces de toras).

Parâmetro	Valor	Contagem total	RE% médio	RE% médio
<i>Número máximo de detecções por imagem (max_det)</i>				
<i>max_det</i>	100	1.000	-76,10	76,10
<i>max_det</i>	200	1.861	-58,08	58,08
<i>max_det</i>	300	2.594	-44,02	44,02
<i>max_det</i>	400	2.819	-39,97	39,97
<i>max_det</i>	500	2.819	-39,97	39,97
<i>max_det</i>	1000	2.819	-39,97	39,97
<i>Limiar de confiança</i>				
Confiança	0,05	5.767	21,63	21,63
Confiança	0,08	4.865	3,01	7,54
Confiança	0,10	4.489	-4,73	5,99
Confiança	0,12	4.174	-11,51	11,51
Confiança	0,15	3.798	-19,27	19,27
Confiança	0,20	3.274	-30,41	30,41

Confiança	0,25	2.819	-39,97	39,97
Confiança	0,30	2.418	-48,46	48,46
Confiança	0,40	1.654	-64,44	64,44
Confiança	0,50	1.008	-78,01	78,01
<i>Resolução de entrada (pixels no maior lado)</i>				
Resolução	640	2.819	-39,97	39,97
Resolução	960	3.288	-30,01	30,01
Resolução	1280	3.656	-22,22	22,22
Resolução	1920	3.735	-20,08	20,08
Resolução	2560	3.736	-19,37	19,37
<i>Configuração do rastreador, na contagem em vídeo (contagem manual: 4.742 toras)</i>				
Rastreador	BoT-SORT, buffer 30	6.532	37,75	37,75
Rastreador	ByteTrack, buffer 250	5.569	17,44	17,44
Rastreador	BoT-SORT, buffer 250	4.816	1,56	1,56

Os três blocos da Tabela 3.3 foram obtidos variando um parâmetro por vez, mantidos os demais nos valores adotados no estudo. O número máximo de detecções condiciona a contagem enquanto for inferior ao número de faces que o modelo é capaz de detectar em cada fotografia. Com o valor de 300, seis das dez imagens atingem esse limite, uma vez que as pilhas avaliadas contêm entre 217 e 586 faces visíveis. A partir de 400 detecções por imagem o resultado estabiliza, e valores superiores não alteram a contagem, o que indica ser esse o patamar a partir do qual o parâmetro deixa de ser restritivo para o material analisado. O efeito do limite, porém, é modesto: sua remoção acrescenta 225 detecções e reduz o erro relativo médio de $-44,02\%$ para $-39,97\%$, ganho de 4,05 pontos percentuais.

O limiar de confiança exerce efeito consideravelmente maior. A redução de 0,25 para 0,10 eleva a contagem total de 2.819 para 4.489 faces, aproximando-a das 4.644 contadas manualmente, e o erro relativo médio passa a $-4,73\%$; o total detectado iguala o manual em um limiar de 0,090. Para verificar a natureza dessas detecções adicionais, as dez fotografias foram anotadas integralmente nos dois limiares e inspecionadas visualmente. Adotou-se como falso positivo a detecção cuja caixa não contém madeira, isto é, aquela que recai sobre o céu, sobre o solo ou sobre qualquer outro elemento da cena que não seja a própria pilha.

A inspeção não identificou nenhum falso positivo, em nenhum dos dois limiares. Duas verificações automatizadas sobre a totalidade das 4.489 detecções obtidas em 0,10 corroboram esse resultado. A primeira mediu a proporção de pixels de céu no interior de cada caixa: o valor máximo observado é de 65,9%, nenhuma caixa alcança 70% e mesmo a mais extrema contém a silhueta de uma tora. A segunda examinou a porção inferior das imagens, onde o chão e os

resíduos de colheita ocupam aproximadamente o quinto inferior de todas as fotografias: das 72 detecções cujo centro está abaixo de 80% da altura da imagem, todas recaem sobre faces de toras soltas caídas entre os resíduos, e nenhuma sobre solo descoberto, galhos ou lascas.

Não há, portanto, falso positivo de fundo em nenhum dos dois limiares: toda caixa recai sobre madeira. Verificou-se, contudo, um segundo tipo de erro, que não decorre de confundir madeira com fundo, mas de marcar a mesma face mais de uma vez. A Figura 3.6 mostra as detecções de uma das fotografias nos dois limiares: as 226 caixas acrescentadas pela redução distribuem-se sobre faces sombreadas, parcialmente ocluídas ou situadas ao fundo da pilha, e nenhuma sobre o plano de fundo da cena, mas parte delas repete uma face já marcada.

a) Limiar 0,25 — 380 detecções



b) Limiar 0,10 — 606 detecções



c) Detalhe — 58 detecções



d) Detalhe — 103 detecções

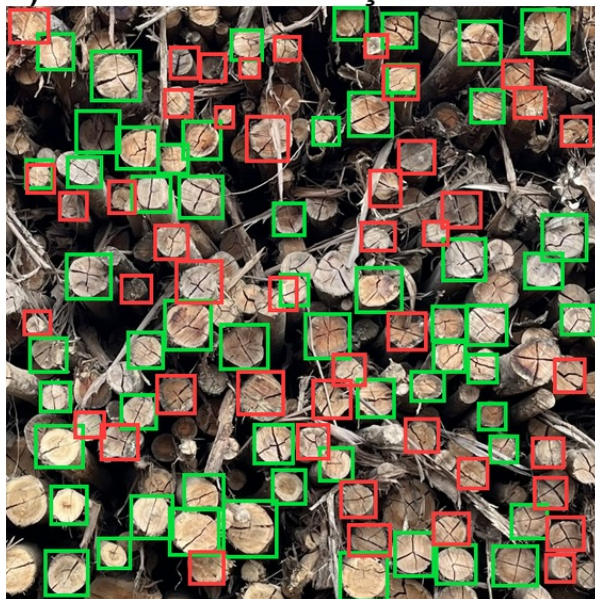


Figura 3.6. Detecções na fotografia D nos limiares de confiança de 0,25 (a, c) e de 0,10 (b, d), em vista geral e em recorte ampliado da mesma região. Em verde, as detecções com confiança igual ou superior a 0,25; em vermelho, as acrescentadas pela redução do limiar. Todas recaem sobre madeira, mas parte delas repete uma face já marcada, como se observa nas caixas concêntricas do recorte (d). Fonte: Elaboração própria.

A supressão de não máximos foi aplicada com limiar de sobreposição de 0,7, de modo que pares de caixas com sobreposição inferior a esse valor, bem como caixas inteiramente contidas em outras, sobrevivem ao pós-processamento. Para quantificá-las, agruparam-se as caixas cuja

razão de interseção sobre união atinge 0,45, ou cuja interseção cobre ao menos 70% da menor delas, e tomou-se cada grupo como uma única face. Observaram-se grupos de duas a quatro caixas concêntricas sobre a mesma face, com confianças distintas, em todas as dez fotografias (Figura 3.7). O padrão é recorrente: uma caixa de confiança mais alta, compatível com o limiar de 0,25, acompanhada de uma ou mais caixas de confiança baixa, ligeiramente maiores ou menores, que só aparecem quando o limiar é reduzido.

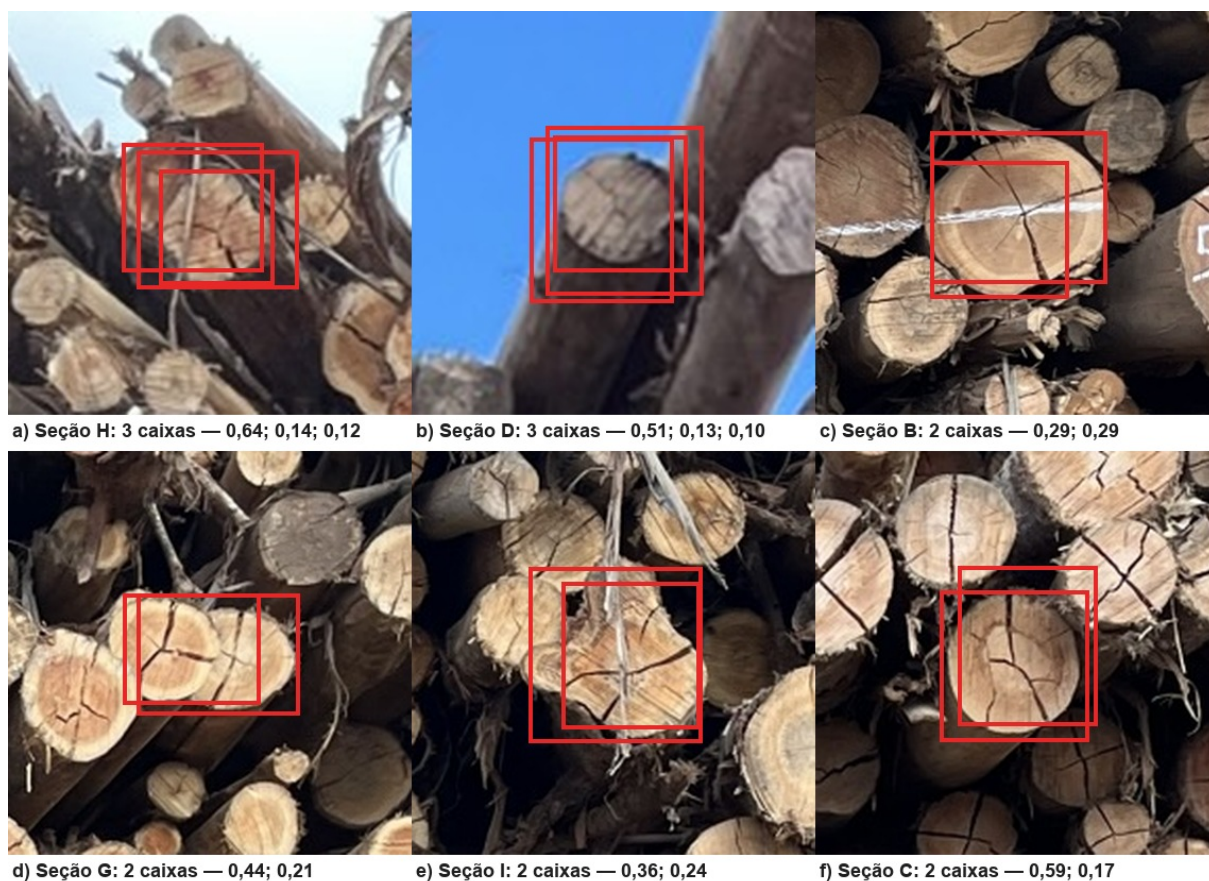


Figura 3.7. Caixas sobrepostas sobre a mesma face de tora, no limiar de confiança de 0,10, em seis das dez fotografias. Cada recorte indica a seção de origem, o número de caixas e as respectivas confianças. Todas as caixas de um mesmo grupo apresentam razão de interseção sobre união de ao menos 0,50 entre si, o que as caracteriza como registros repetidos de uma única face. Fonte: Elaboração própria.

O efeito é fortemente dependente do limiar. Em 0,25, apenas 35 das 2.819 caixas são repetições, ou 1,2%, e a contagem bruta praticamente coincide com o número de faces distintas, 2.784. Em 0,10, são 293 das 4.489, ou 6,5%, e o número de faces distintas cai para 4.196. Variando os critérios de agrupamento entre 0,35 e 0,65 para a interseção sobre união e entre 0,60 e 0,80 para a contenção, a contagem de repetições permanece entre 199 e 331, de modo que a ordem de grandeza do efeito não depende da escolha dos limites.

Corrigida a duplicação, a revocação passa de 59,9% em 0,25 para 90,4% em 0,10, e o erro relativo do total, de -40,05% para -9,65%. O ganho permanece expressivo, mas é bem menor do que a contagem bruta sugere: os -3,34% obtidos sem correção resultam, em parte, de faces contadas duas vezes. Cabe registrar que, uma vez fundidas as caixas repetidas, nenhuma das

dez fotografias supera a contagem manual, o que reforça a validade da referência adotada.

Conclui-se que o limiar de 0,25, herdado da configuração padrão da biblioteca, é conservador demais para fotografias de pilhas densas, nas quais as faces ocupam poucos pixels, e que sua redução é vantajosa desde que acompanhada de uma etapa de fusão de caixas coincidentes. Ressalva-se que o valor de 0,10 foi selecionado sobre as mesmas dez fotografias usadas na avaliação, de modo que sua confirmação exige um conjunto independente de pilhas.

A resolução de entrada revelou-se o parâmetro de maior efeito entre os avaliados sem introduzir superestimativa. A elevação de 640 para 1.280 pixels aumenta a contagem de 2.819 para 3.656 faces, reduzindo o erro relativo médio de $-39,97\%$ para $-22,22\%$, um ganho de quase 18 pontos percentuais; acima de 1.920 pixels o resultado estabiliza em torno de 3.735 faces, ou $80,4\%$ do total manual. Diferentemente do que ocorre ao reduzir o limiar de confiança, nenhuma fotografia passa a superestimar a contagem em qualquer das resoluções testadas, o que indica tratar-se de recuperação de faces efetivamente presentes. Esse resultado é coerente com a natureza da tarefa: as fotografias originais possuem 1536×2048 pixels e, ao serem redimensionadas para 640 pixels no maior lado, as faces das toras passam a ocupar poucos pixels na entrada da rede.

O último bloco da Tabela 3.3 refere-se à contagem em vídeo e confirma o dimensionamento do `track_buffer` discutido nos Materiais e Métodos. Com o valor padrão de 30 quadros, equivalente a 1 s, a contagem alcança 6.532 toras, ou $37,75\%$ acima da contagem manual, em razão da perda e recriação de identificadores a cada oclusão momentânea. Elevando-se o parâmetro para 250 quadros, valor da mesma ordem do tempo de permanência de uma tora no campo de visão, a contagem passa a 4.816 toras, ou $1,56\%$ acima da contagem manual. A substituição do BoT-SORT pelo ByteTrack, mantido o mesmo buffer, conduz a 5.569 toras. Cabe destacar que o dimensionamento do buffer decorre da medição da velocidade de varredura, e não de ajuste ao valor de referência.

3.3.4. Comparação estatística entre fotografia e vídeo

As comparações apresentadas até aqui contrapõem o conjunto das dez fotografias a um único arquivo de vídeo, o que não permite avaliar se a diferença observada entre os dois modos de aquisição excede a variação natural entre trechos da pilha. Para viabilizar essa avaliação, explorou-se o fato de que as dez fotografias e o vídeo registram a mesma pilha.

A correspondência entre as fotografias e o vídeo foi estabelecida por casamento de pontos-chave, com verificação geométrica por homografia: as dez fotografias foram localizadas na linha do tempo do vídeo, em posições estritamente crescentes e correspondentes à ordem em que aparecem na Figura 3.5. A varredura da câmera foi caracterizada por correlação de fase entre quadros consecutivos, mostrando-se contínua e monotônica. Com base nessas posições, o vídeo foi dividido em dez trechos delimitados pelos pontos médios entre fotografias consecutivas, de modo que cada trecho corresponde à mesma porção física da pilha registrada na fotografia respectiva. Cada tora rastreada foi atribuída ao trecho em que seu identificador surgiu pela

primeira vez, o que evita a dupla contagem nas fronteiras. A contagem manual de cada seção, já apresentada na Tabela 3.2, serve de referência comum aos dois modos de aquisição.

Tabela 3.4. Contagem manual, por fotografia nos limiares de confiança de 0,25 e de 0,10, e por vídeo, nas dez seções contíguas e não sobrepostas de uma mesma pilha. As contagens em 0,10 são apresentadas após a fusão das caixas coincidentes.

Seção	Manual	Foto 0,25	RE%	Foto 0,10	RE%	Vídeo	RE%
A	278	147	-47,1	216	-22,3	136	-51,1
B	387	260	-32,8	366	-5,4	261	-32,6
C	540	317	-41,3	461	-14,6	473	-12,4
D	580	380	-34,5	569	-1,9	623	7,4
E	534	314	-41,2	472	-11,6	680	27,3
F	586	334	-43,0	536	-8,5	710	21,2
G	524	355	-32,3	504	-3,8	693	32,3
H	546	325	-40,5	491	-10,1	563	3,1
I	452	273	-39,6	406	-10,2	388	-14,2
J	217	114	-47,5	175	-19,4	289	33,2
Total	4.644	2.819	-39,3	4.196	-9,6	4.816	3,7

A contagem por fotografia apresenta viés sistemático inequívoco: o erro relativo médio das dez seções é de $-39,97\%$, com desvio-padrão de 5,39 pontos percentuais, e o teste t pareado rejeita a hipótese de ausência de viés com $t = -23,47$ e $p < 0,001$ (intervalo de confiança de 95% de $-43,82\%$ a $-36,12\%$); o teste de Wilcoxon conduz à mesma conclusão ($p = 0,002$). A contagem por vídeo, ao contrário, não apresenta viés detectável: o erro relativo médio é de $+1,43\%$, com $t = 0,16$ e $p = 0,879$, e o intervalo de confiança de 95%, de $-19,09\%$ a $+21,94\%$, contém o zero.

A comparação direta entre os dois modos de aquisição, pareada por seção, indica diferença estatisticamente significativa em favor do vídeo ($t = -4,54$; $p = 0,001$; Wilcoxon $p = 0,006$). A regressão da contagem automática sobre a contagem manual reforça esse resultado: para as fotografias, a inclinação é de 0,655, com intervalo de confiança de 95% de 0,543 a 0,767, que **exclui** a unidade, e coeficiente de determinação de 0,943; para o vídeo, a inclinação é de 1,370, com intervalo de 0,824 a 1,915, que **contém** a unidade, e coeficiente de determinação de 0,752.

Duas ressalvas devem acompanhar esses resultados. A primeira é que a ausência de viés no vídeo não equivale a precisão: o erro por seção varia de $-51,1\%$ a $+33,2\%$, com erro absoluto médio de 23,47%, de modo que o excelente resultado do total decorre, em parte, de compensação entre seções que superestimam e seções que subestimam. A fotografia, embora fortemente enviesada, é consideravelmente mais consistente, com erro absoluto médio de 39,97% e desvio-padrão de apenas 5,39 pontos percentuais; o elevado coeficiente de determinação de sua regressão indica que a subestimativa é proporcional e previsível, e portanto passível de correção por um fator de calibração.

Repetiu-se a comparação com a fotografia no limiar de 0,10, após a fusão das caixas coincidentes, cujos valores constam da Tabela 3.4. O viés de subestimação persiste, mas cai de $-39,97\%$ para $-10,78\%$, com desvio-padrão de 6,53 pontos percentuais ($t = -5,22$; $p < 0,001$; intervalo de confiança de 95% de $-15,45\%$ a $-6,11\%$). O erro absoluto médio passa a $10,78\%$, menos da metade dos $23,47\%$ do vídeo, e a comparação pareada dos erros absolutos por seção favorece agora a fotografia ($t = -3,15$; $p = 0,012$; Wilcoxon $p = 0,020$). A regressão também se altera de forma expressiva: a inclinação passa de 0,655 para 1,005, com intervalo de confiança de 95% de 0,869 a 1,142, que contém a unidade, e o coeficiente de determinação sobe para 0,973. O que resta do erro deixa de ser proporcional e converte-se em deslocamento aproximadamente constante, da ordem de 47 faces por seção.

A leitura conjunta dos três conjuntos de resultados é a seguinte: na configuração adotada no estudo, o vídeo é superior à fotografia, e essa superioridade é estatisticamente significativa; corrigido o limiar de confiança e a duplicação de caixas, a fotografia torna-se mais acurada por seção, embora conserve viés de subestimação, ao passo que o vídeo permanece o único dos três modos sem viés detectável no total. Cabe reiterar, contudo, que o limiar de 0,10 foi escolhido a partir dessas mesmas dez seções, de modo que a vantagem da fotografia sob essa configuração não constitui evidência independente.

A segunda ressalva é que as dez seções provêm de uma única pilha. Os testes aqui apresentados avaliam, portanto, se a diferença entre os modos de aquisição excede a variação entre trechos dessa pilha, e não permitem generalizar a magnitude do erro para outras pilhas, espécies ou padrões de empilhamento. A validação da acurácia do método em sentido amplo requer delineamento com repetições em pilhas distintas, cada uma registrada por fotografia e por vídeo, o que constitui desdobramento necessário deste trabalho.

3.4. DISCUSSÃO

A evolução das versões do modelo YOLO e as características arquiteturais que as distinguem foram tratadas no Capítulo 1, ao qual se remete o leitor; a discussão a seguir concentra-se na interpretação dos resultados obtidos neste capítulo.

Nossa pesquisa baseia-se nos impressionantes resultados de estudos anteriores que utilizaram com sucesso o modelo YOLO em diversas tarefas relacionadas à madeira, como detecção de nós superficiais em madeira serrada, extração de características de contorno em estruturas complexas e detecção de defeitos em madeira (CUI; LU; LIU, 2023; FANG et al., 2021; KURDTHONGMEE, 2023; MA et al., 2022). Em avaliações dedicadas à segmentação de troncos e galhos, o YOLOv8 obteve precisão, revocação e F1 de 0,888, 0,974 e 0,922 para troncos, e 0,739, 0,789 e 0,742 para galhos, respectivamente (AHMED et al., 2023). Esses resultados demonstram a eficácia do modelo na segmentação de troncos e galhos. Sua versatilidade também foi demonstrada na detecção de palmeiras, especialmente quando combinada com imagens de veículos aéreos não tripulados (VANT) e mapeamento remoto

(RPW), alcançando precisão de 0,923, revocação de 0,951 e F1 de 0,937 (HAJJAJI et al., 2023). Em outro estudo, variantes do YOLOv8 foram avaliadas na classificação de espécies arbóreas a partir de imagens aéreas, sendo que o YOLOv8-l superou as demais variantes, alcançando médias ponderada e micro de 71,55% e 72,70%, respectivamente (BAYRAK; ERDEM; UZAR, 2023). Pesquisas também foram realizadas para detecção de ervas daninhas em gramados usando detectores YOLO. O YOLOv8l apresentou o melhor desempenho no conjunto de teste inicial, com precisão de 0,9476, mAP_0.5 de 0,9795 e mAP_0.5:0.95 de 0,8123; contudo, seu desempenho em conjuntos de teste adicionais não atingiu o padrão exigido para uso profissional (SPORTELLI et al., 2023).

A robustez do método em cenários diversos não apenas comprova sua eficácia, mas também amplia suas aplicações potenciais em contextos ambientais e ecológicos. Pesquisas indicam que a detecção de objetos pode ser aplicada a uma variedade de variáveis, permitindo a classificação precisa de cenas a partir de dados de sensoriamento remoto, incluindo tarefas como detecção de árvores (PU et al., 2023) e árvores mortas no manejo florestal (WANG, XUEWEN et al., 2022), contribuindo para o entendimento dos impactos humanos sobre ecossistemas naturais e biodiversidade.

No entanto, nosso foco foi especificamente na detecção e contagem de madeira de eucalipto, um componente crítico na produção de celulose e no manejo florestal, que exige automação precisa e eficiente. Até o momento, não foram encontrados estudos aplicados para essa tarefa. Os resultados indicaram que a detecção de faces de toras em pilhas densas é tarefa de difícil execução. Nos conjuntos de treinamento, validação e teste, compostos por imagens com as mesmas características do material de treinamento, o modelo alcançou precisão entre 0,741 e 0,814 e revocação entre 0,779 e 0,812. Nas pilhas industriais, contudo, a fotografia detectou 60,7% das faces visíveis, com subestimativa sistemática de -39,97% em média. Não se pode, portanto, qualificar o desempenho em fotografias de pilhas densas como satisfatório para fins de inventário sem correção prévia, ainda que a regressão entre contagem automática e manual apresente coeficiente de determinação de 0,943, o que indica subestimativa proporcional e passível de calibração. Verificou-se que a contagem em vídeo foi mais acurada que a obtida por fotografia na mesma pilha, diferença estatisticamente significativa ($p = 0,001$), o que se atribui ao deslocamento da câmera, que registra cada tora em vários quadros e sob diferentes ângulos, tornando visíveis faces que permanecem ocultas em uma tomada estática. Além disso, o vídeo apresenta a vantagem de melhorar a precisão e a eficiência da detecção e contagem, aproveitando múltiplos quadros e rastreando objetos ao longo deles, evitando contagens duplicadas. Fotografias devem ser utilizadas apenas para detecção e contagem de toras separadas.

A margem de erro aceitável para sistemas de detecção e contagem varia de acordo com a aplicação específica. Em setores onde o valor do produto é elevado, como na indústria de celulose, mesmo pequenos erros podem levar a perdas financeiras significativas. No experimento de detecção de madeira empilhada por vídeo, obtivemos erro de contagem de +1,561% no total da pilha, sem viés estatisticamente detectável, valor plenamente adequado a fins de contagem; a

variação entre seções, contudo, ainda requer pesquisas adicionais para maior precisão local. Em contrapartida, o uso de fotografias resultou em erros entre 32% e 47%, limitando fortemente sua aplicabilidade, especialmente em contextos industriais. Isso reforça a importância de alcançar alta precisão em sistemas de detecção e contagem.

A análise de sensibilidade indica que a principal limitação do método não reside nos parâmetros de inferência, mas na composição do conjunto de treinamento. O modelo foi treinado com 230 fotografias, nas quais cada imagem de madeira empilhada continha entre 50 e 150 anotações. As pilhas industriais utilizadas na etapa de generalização, por sua vez, apresentam entre 217 e 586 toras por fotografia, ou seja, densidades de 1,5 a 4 vezes superiores ao máximo observado durante o treinamento. O modelo é, portanto, aplicado em condição de extrapolação quanto à densidade de objetos por imagem, o que ajuda a explicar tanto a subestimativa sistemática observada nas fotografias quanto o fato de que o limite padrão de 300 detecções por imagem, folgado em relação ao conjunto de treinamento, se mostre restritivo diante das pilhas industriais. Nessa condição, reduzir o limiar de confiança recupera parte expressiva das faces perdidas, e a verificação das 4.489 detecções obtidas em 0,10 mostrou que todas recaem sobre madeira. Ainda assim, a redução não constitui solução completa, pois faz o total perder a ancoragem na contagem manual de referência.

A subestimativa observada pode ser decomposta em duas parcelas. A primeira corresponde ao próprio teto de revocação do modelo: no conjunto de teste, a revocação foi de 0,779 (Tabela 3.1), de modo que, mesmo em imagens com as características do conjunto de treinamento, cerca de 22% das faces anotadas não são detectadas. Aplicada às 4.644 faces contadas manualmente nas dez fotografias, essa revocação corresponderia a 3.618 faces detectadas. O modelo detectou 2.819, ou 60,7% do total, o que revela uma perda adicional de 799 faces, equivalente a 17,2 pontos percentuais, atribuível especificamente às condições das pilhas industriais.

A análise de sensibilidade indica que essa perda adicional está associada, em grande medida, à resolução de entrada adotada. As fotografias possuem 1536×2048 pixels e as faces detectadas apresentam lado mediano de 44,3 pixels nessa resolução, valor 1,45 vez menor que o lado geométrico médio das faces anotadas no treinamento, de 64,1 pixels. Ao serem redimensionadas para 640 pixels no maior lado, essas faces passam a ocupar cerca de 14 pixels na entrada da rede, e as faces menores, próximas do limite inferior anotado no treinamento, reduzem-se a poucos pixels. Elevar a resolução de entrada para 1.280 pixels recupera 837 faces adicionais, sem que qualquer fotografia passe a superestimar a contagem, o que confirma tratar-se de faces efetivamente presentes e não detectadas por limitação de escala.

Outras hipóteses foram avaliadas e descartadas. A oclusão não contribui para a diferença, uma vez que tanto a anotação quanto a contagem manual consideraram exclusivamente as faces visíveis. O limite máximo de detecções por imagem responde por apenas 4,05 pontos percentuais, conforme quantificado na análise de sensibilidade. A supressão de não máximos também foi testada: o afrouxamento do limiar de IoU de 0,7 para 0,9 acrescenta detecções, porém todas elas se sobrepõem a caixas já existentes com IoU superior a 0,7, e 86% com IoU superior a 0,8, o que

as caracteriza como duplicatas sobre a mesma face, e não como faces recuperadas. Conclui-se, portanto, que a subestimativa decorre da não detecção de faces de pequena dimensão aparente, e não de parâmetros de pós-processamento.

Essa constatação orienta as estratégias de aprimoramento. Ganhos expressivos podem ser obtidos simplesmente elevando a resolução de inferência, medida que não exige novo treinamento; ganhos adicionais dependem da ampliação do conjunto de treinamento com fotografias de pilhas densas obtidas em escala semelhante à de aplicação e da adoção de esquemas de inferência por mosaicos, nos quais a imagem é dividida em recortes processados individualmente e cujas detecções são posteriormente reunidas, preservando a resolução original das faces em vez de reduzi-la pelo redimensionamento global da imagem (AKYON; ALTINUC; TEMIZEL, 2022). Cabe destacar que a contagem em vídeo, cujo erro relativo no total da pilha foi de +1,561%, não é afetada por essa limitação na mesma medida, pois o deslocamento da câmera ao longo da pilha aproxima progressivamente as faces do observador e o rastreamento consolida em um único identificador as detecções obtidas em diferentes quadros e distâncias.

Vale destacar que, para que o modelo possa operar em larga escala ou ser comercializado, como no desenvolvimento de um aplicativo, é necessário um volume substancial de fotografias, superior ao recebido do setor industrial que forneceu os dados para este estudo.

Os resultados obtidos contribuem para a aplicação da visão computacional nas ciências florestais ao delimitar, com base em evidência quantitativa, em que condições a contagem automática de madeira empilhada é viável: por vídeo, sem viés detectável na pilha avaliada; por fotografia, apenas mediante calibração da subestimativa ou elevação da resolução de inferência. A consolidação do método em ferramenta de uso industrial depende ainda da validação em pilhas distintas e da ampliação do conjunto de treinamento.

Além disso, este estudo abre possibilidades para integrar a tecnologia de visão computacional em outras áreas de pesquisa e prática florestal, impulsionando o setor para um futuro tecnologicamente avançado. A contagem precisa de toras em pilhas pode ajudar a melhorar a automação de aplicações já existentes ou auxiliar no desenvolvimento de novas estratégias para quantificar o volume sólido de madeira empilhada, abordando um desafio crucial na indústria florestal.

3.5. CONCLUSÕES

O modelo YOLOv8 mostrou-se capaz de automatizar a contagem de madeira de eucalipto empilhada, com desempenho que depende fortemente do modo de aquisição e, como se demonstrou, também da configuração de inferência: adequado quando a contagem é feita em vídeo e insuficiente, sem correção, quando feita em fotografias de pilhas densas na configuração padrão da biblioteca. Nas condições em que se mostrou adequado, o procedimento pode agilizar o processo manual e reduzir tempo e recursos.

Embora seja difícil alcançar alta precisão na detecção, essa limitação não afetou a contagem

automática. O modelo apresentou excelente desempenho na identificação de toras em diferentes arranjos, atingindo alta precisão. Contudo, subestimou de forma consistente o número de toras em imagens estáticas, com erros variando de $-32,252\%$ a $-47,465\%$.

Enquanto as imagens estáticas apresentaram desafios, o desempenho do modelo melhorou significativamente na contagem de toras em vídeo, cujo erro no total da pilha foi de $+1,561\%$, sem viés estatisticamente detectável. A comparação pareada entre os dois modos de aquisição, realizada em dez seções contíguas de uma mesma pilha, confirmou que a superioridade do vídeo é estatisticamente significativa ($p = 0,001$) na configuração adotada. Essa superioridade manifesta-se no total da pilha, e não seção a seção: o erro do vídeo varia de $-51,1\%$ a $+33,2\%$ entre trechos, com desvio-padrão de 28,68 pontos percentuais, de modo que o excelente resultado agregado decorre em parte de compensação entre seções.

A análise de sensibilidade mostrou que a insuficiência da fotografia decorre em grande medida do limiar de confiança herdado da configuração padrão, e não do modo de aquisição em si. A inspeção integral das detecções não identificou nenhum falso positivo de fundo em nenhum dos limiares avaliados, de modo que o limiar governa apenas quantas faces são recuperadas. Reduzindo-o de 0,25 para 0,10 e fundindo as caixas coincidentes, que respondem por 6,5% dos registros nesse limiar, a revocação sobe de 59,9% para 90,4%, o erro do total cai de $-40,05\%$ para $-9,65\%$ e a inclinação da regressão sobre a contagem manual passa de 0,655 para 1,005, contendo a unidade. Sob essa configuração, a fotografia torna-se mais acurada que o vídeo seção a seção, com erro absoluto médio de 10,78% contra 23,47% ($p = 0,012$).

Depreende-se daí uma recomendação operacional de duas partes: o vídeo é preferível para a quantificação do total de uma pilha, por não apresentar viés detectável, ao passo que a fotografia, com limiar reduzido e fusão de caixas coincidentes, é preferível para a mensuração de trechos delimitados, por ser consideravelmente mais estável. Ressalva-se que o valor de 0,10 foi selecionado sobre as mesmas seções usadas na avaliação, de modo que sua confirmação, bem como a do procedimento de fusão, exige um conjunto independente de pilhas. Reitera-se, ainda, que esses resultados provêm de uma única pilha e que a generalização da acurácia do método requer delineamento com repetições em pilhas distintas. Pesquisas e melhorias adicionais podem superar as limitações atuais, tornando-o uma ferramenta ainda mais robusta e confiável para operações florestais.

3.6. REFERÊNCIAS

AHARON, Nir; ORFAIG, Roy; BOBROVSKY, Ben-Zion. **BoT-SORT: Robust Associations Multi-Pedestrian Tracking**. arXiv preprint arXiv:2206.14651, 2022. <https://doi.org/10.48550/arXiv.2206.14651>.

AKYON, Fatih Cagatay; ALTINUC, Sinan Onur; TEMIZEL, Alptekin. Slicing Aided Hyper Inference and Fine-Tuning for Small Object Detection. In: **2022 IEEE International Conference on Image Processing (ICIP)**. Bordeaux: IEEE, 2022. p. 966-970.

<https://doi.org/10.1109/ICIP46576.2022.9897990>.

AHMED, Dawood; SAPKOTA, Ranjan; CHURUVIJA, Martin; KARKEE, Manoj. **Machine Vision-Based Crop-Load Estimation Using YOLOv8**. arXiv preprint arXiv:2304.13282, 2023. <https://doi.org/10.48550/arXiv.2304.13282>.

ANTUNES SANTANA, Otacílio; IMAÑA ENCINAS, José; RIESCO MUÑOZ, Guillermo. Stacking factor in transporting firewood produced from a mixture of Caatinga biome species in Brazil. **International Journal of Forest Engineering**, v. 34, n. 1, p. 54-63, 2023. <https://doi.org/10.1080/14942119.2022.2084674>.

BAYRAK, O. C.; ERDEM, F.; UZAR, M. Deep learning based aerial imagery classification for tree species identification. **Int. Arch. Photogramm. Remote Sens. Spatial Inf. Sci.**, XLVIII-M-1-2023, p. 471-476, 2023. <https://doi.org/10.5194/isprs-archives-XLVIII-M-1-2023-471-2023>, 2023.

BERTOLA, Alexandre; SOARES, Carlos Pedro Boechat; RIBEIRO, José Carlos; LEITE, Helio Garcia; SOUZA, Agostinho Lopes. Determination of piling factors through Digitora software. **Revista Árvore**, v. 27, n. 6, p. 837-844, 2003. <https://doi.org/10.1590/S0100-67622003000600010>.

BOCHKOVSKIY, A.; WANG, C. Y.; LIAO, H. Y. M. **Yolov4: Optimal speed and accuracy of object detection**. arXiv:2004.10934v1, 2020. <https://doi.org/10.48550/arXiv.2004.10934>.

CAMPOS, João Carlos Chagas; LEITE, Helio Garcia. **Forest measurement: Questions and answers**. Viçosa: UFV, 2017.

CARVALHO, A. M.; CAMARGO, F. R. A. Avaliação do método de recebimento de madeira por estere [Evaluation of the method of receiving wood by stere]. **Revista O Papel**, v. 57, p. 65-68, 1996.

CAVALIN, P.; OLIVEIRA, L. S.; KOERICH, A. L.; BRITTO, A. S. Wood Defect Detection using Grayscale Images and an Optimized Feature Set. In: **IECON 2006 - 32nd Annual Conference on IEEE Industrial Electronics**. IEEE, 2006. p. 3408-3412. <https://doi.org/10.1109/IECON.2006.347618>.

CREMER, Tobias; BERENDT, Ferreol; DIEZ, Felipe de Miguel; WOLFGRAMM, Felix; BLASKO, Lubomir. Accuracy of Photo-Optical Measurement of Wood Piles. In: **The 1st International Electronic Conference on Forests-For a Better Future: Sustainability, Innovation, Interdisciplinarity**. Basel Switzerland: MDPI, 2020. p. 90. <https://doi.org/10.3390/IECF2020-08192>.

CUI, Yuming; LU, Shuochen; LIU, Songyong. Real-time detection of wood defects based on SPP-improved YOLO algorithm. **Multimedia Tools and Applications**, v. 82, n. 14, p. 21031-21044, 2023. <https://doi.org/10.1007/s11042-023-14588-7>.

DAVIES, E.R. The dramatically changing face of computer vision. In: **Advanced Methods and Deep Learning in Computer Vision**. Elsevier, 2022. p. 1-91. <https://doi.org/10.1016/B978-0-12-822109-9.00010-2>.

FANG, Yiming; GUO, Xianxin; CHEN, Kun; ZHOU, Zhu; YE, Qing. Accurate and automated detection of surface knots on sawn timbers using YOLO-V5 model. **BioResources**, v. 16, n. 3, p. 5390-5406, 2021. <https://doi.org/10.15376/biores.16.3.5390-5406>.

GAVALI, Pralhad; BANU, J. Saira. Deep Convolutional Neural Network for Image Classification on CUDA Platform. In: **Deep Learning and Parallel Computing Environment for Bio-**

engineering Systems. Elsevier, 2019. p. 99-122. <https://doi.org/10.1016/B978-0-12-816718-2.00013-0>.

GLASS, Samuel; ZELINKA, Samuel L.; JOHNSON, Jay A. **Investigation of Historic Equilibrium Moisture Content Data from the Forest Products Laboratory**. 2014. <https://doi.org/10.2737/FPL-GTR-229>.

GOODALE, Melvyn A.; MILNER, A. David. Separate visual pathways for perception and action. **Trends in Neurosciences**, v. 15, n. 1, p. 20-25, 1992. [https://doi.org/10.1016/0166-2236\(92\)90344-8](https://doi.org/10.1016/0166-2236(92)90344-8).

HAJJAJI, Yosra; ALZAHM, Ayyub; BOULILA, Wadii; FARAH, Imed Riadh; KOUBAA, Anis. **Sustainable Palm Tree Farming: Leveraging IoT and Multi-Modal Data for Early Detection and Mapping of Red Palm Weevil**. arXiv preprint arXiv:2306.16862, 2023. <https://doi.org/10.48550/arXiv.2306.16862>.

HUSCH, B.; MILLER, C. I.; BEERS, T. W. **Forest Mensuration**. Krieger Publishing Company., 1993.

JOCHER, G. **YOLOv5 by Ultralytics (Version 7.0)**, 2020. <https://doi.org/10.5281/zenodo.3908559>.

JOCHER, G.; CHAURASIA, A.; QIU, J. **YOLO by Ultralytics (Version 8.0.0)**, 2023. Disponível em: <https://github.com/ultralytics/ultralytics>.

KOMAN, Szabolcs; FEHER, Sandor. Basic density of hardwoods depending on age and site. **Wood Research**, v. 60, n. 6, p. 907-912, 2015.

KRIZHEVSKY, Alex; SUTSKEVER, Ilya; HINTON, Geoffrey E. ImageNet classification with deep convolutional neural networks. **Communications of the ACM**, v. 60, n. 6, p. 84-90, 2017. <https://doi.org/10.1145/3065386>.

KUNICKAYA, Olga; POMIGUEV, Aleksandr; KRUCHININ, Igor; STORODUBTSEVA, Tamara; VORONOVA, Anna; LEVUSHKIN, Dmitry; BORISOV, Vyacheslav; IVANOV, Viktor. Analysis of modern wood processing techniques in timber terminals. **Central European Forestry Journal**, v. 68, n. 1, p. 51-59, 2022. <https://doi.org/10.2478/forj-2021-0017>.

KURDTHONGMEE, Wattanapong. **Improving Wood Defect Detection Accuracy with Yolo V3 by Incorporating Out-of-Defect Area Annotations**. 2023. <https://doi.org/10.2139/ssrn.4395580>.

KÄRHÄ, Kalle; NURMELA, Sari; KARVONEN, Heikki; KIVINEN, Veli-Pekka; MELKAS, Timo; NIEMINEN, Miikka. Estimating the accuracy and time consumption of a mobile machine vision application in measuring timber stacks. **Computers and Electronics in Agriculture**, v. 158, p. 167-182, 2019. <https://doi.org/10.1016/j.compag.2019.01.040>.

LEITE, Rodrigo Vieira; DO AMARAL, Cibele Hummel; DE PAULA PIRES, Raul; SILVA, Carlos Alberto; SOARES, Carlos Pedro Boechat; MACEDO, Renata Paulo; DA SILVA, Antonilmar Araújo Lopes; BROADBENT, Eben North; MOHAN, Midhun; LEITE, Hélio Garcia. Estimating stem volume in eucalyptus plantations using airborne LiDAR: A comparison of area- and individual tree-based approaches. **Remote Sensing**, v. 12, n. 9, 2020. <https://doi.org/10.3390/RS12091513>.

LI, Chuyi.; LI, Lulu.; JIANG, Hongliang.; WENG, Kaiheng.; GENG, Yifei.; LI, Liang.; KE, Zaidan.; LI, Qingyuan.; CHENG, Meng.; NIE, Weiqiang.; LI, Yiduo.; ZHANG, Bo.; LIANG, Yufei.; ZHOU, Linyuan.; XU, Xiaoming.; CHU, Xiangxiang.; WEI, Xiaoming.; WEI, Xiaolin.

YOLOv6: A single-stage object detection framework for industrial applications. arXiv preprint arXiv:2209.02976, 2022. <https://doi.org/10.48550/arXiv.2209.02976>.

LI, Xiang; WANG, Wenhai; WU, Lijun; CHEN, Shuo; HU, Xiaolin; LI, Jun; TANG, Jinhui; YANG, Jian. Generalized Focal Loss: Learning Qualified and Distributed Bounding Boxes for Dense Object Detection. **Advances in Neural Information Processing Systems**, v. 33, p. 21002-21012, 2020.

LIU, Wei; ANGUELOV, Dragomir; ERHAN, Dumitru; SZEGEDY, Christian; REED, Scott; FU, Cheng-Yang; BERG, Alexander C. SSD: Single Shot MultiBox Detector. In: **Computer Vision - ECCV 2016**. ECCV 2016. Lecture Notes in Computer Science. Cham: Springer, 2016. p. 21-37. https://doi.org/10.1007/978-3-319-46448-0_2.

LIU, Y.; HOU, M.; LI, A.; DONG, Y.; XIE, L.; JI, Y. Automatic detection of timber-cracks in wooden architectural heritage using YOLOv3 algorithm. **The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences**, v. XLIII-B2-2020, p. 1471-1476, 2020. <https://doi.org/10.5194/isprs-archives-XLIII-B2-2020-1471-2020>.

MA, Jian; YAN, Weidong; LIU, Guoqi; XING, Shiyu; NIU, Siqi; WEI, Tong. Complex Texture Contour Feature Extraction of Cracks in Timber Structures of Ancient Architecture Based on YOLO Algorithm. **Advances in Civil Engineering**, v. 2022, p. 1-13, 2022. <https://doi.org/10.1155/2022/7879302>.

MEDERSKI, Piotr S.; NASKRENT, Bartłomiej; TOMCZAK, Arkadiusz; TOMCZAK, Karol. Accuracy of Photo-Optical Timber Measurement Using a Stereo Camera Technology. **Croatian journal of forest engineering**, v. 45, n. 1, 2023. <https://doi.org/10.5552/crojfe.2024.2268>.

MOSKALIK, Tadeusz; TYMENDORF, Łukasz; VAN DER SAAR, Jan; TRZCIŃSKI, Grzegorz. Methods of Wood Volume Determining and Its Implications for Forest Transport. **Sensors**, v. 22, n. 16, p. 6028, 2022. <https://doi.org/10.3390/s22166028>.

NYLINDER, M.; KUBÉNKA, T.; HULTNÄS, M. **Roundwood measurement of truck loads by laser scanning**. Field study at Arauco pulp mill Nueva Aldea, p. 1-9, 2008.

PU, Yihan; XU, Dandan; WANG, Haobin; LI, Xin; XU, Xia. A New Strategy for Individual Tree Detection and Segmentation from Leaf-on and Leaf-off UAV-LiDAR Point Clouds Based on Automatic Detection of Seed Points. **Remote Sensing**, v. 15, n. 6, p. 1619, 2023. <https://doi.org/10.3390/rs15061619>.

PURFÜRST, T; DE MIGUEL-DÍEZ, F; BERENDT, F; ENGLER, B; CREMER, T. Comparison of wood stack volume determination between manual, photo-optical, iPad-LiDAR and handheld-LiDAR based measurement methods. **iForest - Biogeosciences and Forestry**, v. 16, n. 4, p. 243-252, 2023. <https://doi.org/10.3832/ifor4153-016>.

PÁSZTORY, Z.; HEINZMANN, B.; BARBU, M. C. Manual and Automatic Volume Measuring Methods for Industrial Timber. In: **IOP Conference Series: Earth and Environmental Science**. Institute of Physics Publishing, 2018. <https://doi.org/10.1088/1755-1315/159/1/012019>.

REDMON, J.; DIVVALA, S.; GIRSHICK, R.; FARHADI, A. You Only Look Once: Unified, Real-Time Object Detection. In: **Proceedings of the IEEE conference on computer vision and pattern recognition**. 2016. p. 779-788.

REDMON, J.; FARHADI, A. YOLO9000: better, faster, stronger. In: **Proceedings of the IEEE conference on computer vision and pattern recognition**. 2017. p. 7263-7271.

REDMON, J.; FARHADI, A. **Yolov3: An incremental improvement**. arXiv preprint

arXiv:1804.02767, 2018. <https://doi.org/10.48550/arXiv.1804.02767>.

SILVEIRA, D. D. P. **Estimation of the volume wooden stacked using digital images and neural networks**. 2014. Master's Dissertation, Universidade Federal de Viçosa. Disponível em: <http://locus.ufv.br/handle/123456789/3169>.

SOARES, C. P. B.; PAULA NETO, F.; SOUZA, A. L. **Dendrometria e inventário florestal**. 2. ed. Viçosa: Editora UFV, 2011. 272 p.

SPORTELLI, Mino; APOLO-APOLO, Orly Enrique; FONTANELLI, Marco; FRASCONI, Christian; RAFFAELLI, Michele; PERUZZI, Andrea; PEREZ-RUIZ, Manuel. Evaluation of YOLO Object Detectors for Weed Detection in Different Turfgrass Scenarios. **Applied Sciences**, v. 13, n. 14, p. 8502, 2023. <https://doi.org/10.3390/app13148502>.

TERVEN, Juan; CORDOVA-ESPARZA, Diana. **A comprehensive review of YOLO: From YOLOv1 to YOLOv8 and beyond**. arXiv preprint arXiv:2304.00501, 2023. <https://doi.org/10.48550/arXiv.2304.00501>.

TZUTALIN, D. **LabelImg**, 2015. Disponível em: <https://github.com/tzutalin/labelImg>.

VOULODIMOS, Athanasios; DOULAMIS, Nikolaos; DOULAMIS, Anastasios; PROTOPADAKIS, Eftychios. Deep Learning for Computer Vision: A Brief Review. **Computational Intelligence and Neuroscience**, v. 2018, p. 1-13, 2018. <https://doi.org/10.1155/2018/7068349>.

WANG, C. Y.; BOCHKOVSKIY, A.; LIAO, H. Y. M. YOLOv7: Trainable bag-of-freebies sets new state-of-the-art for real-time object detectors. In: **Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition**. 2023. p. 7464-7475.

WANG, Xuewen; ZHAO, Qingzhan; JIANG, Ping; ZHENG, Yuchen; YUAN, Limengzi; YUAN, Panli. LDS-YOLO: A lightweight small object detection method for dead trees from shelter forest. **Computers and Electronics in Agriculture**, v. 198, p. 107035, 2022. <https://doi.org/10.1016/j.compag.2022.107035>.

WATANABE, Ken; KOBAYASHI, Isao; KURODA, Naohiro. Investigation of wood properties that influence the final moisture content of air-dried sugi (*Cryptomeria japonica*) using principal component regression analysis. **Journal of Wood Science**, v. 58, n. 6, p. 487-492, 2012. <https://doi.org/10.1007/s10086-012-1283-5>.

ZEILER, Matthew D.; FERGUS, Rob. Visualizing and Understanding Convolutional Networks. In: **Computer Vision - ECCV 2014**. Lecture Notes in Computer Science. Cham: Springer, 2014. p. 818-833. https://doi.org/10.1007/978-3-319-10590-1_53.

ZHANG, Yizhuo; LIU, Sijia; TU, Wenjun; YU, Huiling; LI, Chao. Using computer vision and compressed sensing for wood plate surface detection. **Optical Engineering**, v. 54, n. 10, p. 103102, 2015. <https://doi.org/10.1117/1.OE.54.10.103102>.

ZHENG, Zhaohui; WANG, Ping; LIU, Wei; LI, Jinze; YE, Rongguang; REN, Dongwei. Distance-IoU Loss: Faster and Better Learning for Bounding Box Regression. **Proceedings of the AAAI Conference on Artificial Intelligence**, v. 34, n. 07, p. 12993-13000, 2020. <https://doi.org/10.1609/aaai.v34i07.6999>.

ZHU, Song-Chun; WU, Ying. Statistics of Natural Images. In: **Computer Vision**. Cham: Springer International Publishing, 2023. p. 19-35. https://doi.org/10.1007/978-3-030-96530-3_2.

CAPÍTULO IV

QUANTIFICAÇÃO DO VOLUME SÓLIDO DE MADEIRA DE EUCALIPTO EMPILHADA UTILIZANDO MODELOS DE DETECÇÃO-SEGMENTAÇÃO E DE DISTRIBUIÇÃO DE DIÂMETROS

RESUMO

A quantificação precisa de madeira é essencial na silvicultura e na indústria madeireira, impactando o planeamento da colheita, o processamento, a precificação e toda a gestão da cadeia de abastecimento. Os métodos tradicionais para estimar o volume de madeira empilhada, frequentemente baseados em medições manuais, são morosos e suscetíveis a erros. Este estudo teve como objetivo desenvolver um procedimento preciso para estimar o volume de madeira de eucalipto empilhada em pátios. O procedimento proposto combina detecção automática de toras com modelos de distribuição de diâmetros. A detecção automática foi realizada com técnicas avançadas de visão computacional, especificamente o modelo You Only Look Once versão 9 (YOLOv9), que automatiza a identificação e contagem individual das toras numa pilha. Utilizaram-se modelos de distribuição do diâmetro do fuste para estimar o volume total da pilha com base na contagem das toras e nas densidades de probabilidade. Esta abordagem garante elevada precisão e eficiência, reduzindo significativamente o tempo e o esforço necessários para a estimativa de volume. O conjunto de dados inclui medições de diâmetro de um inventário pré-corte de eucaliptos com 6 e 7 anos, juntamente com vídeos de madeira empilhada. O modelo YOLOv9 foi treinado para detectar toras a partir desses vídeos, obtendo alta precisão nas tarefas de detecção e segmentação. Métricas como Box Precision, Box Recall e mean Average Precision (mAP) foram utilizadas para avaliar a eficácia do modelo. Os resultados indicam que o modelo generaliza bem para novos dados, com elevada precisão tanto nos conjuntos de validação quanto de teste. Entre os modelos de distribuição avaliados, a distribuição de valor extremo generalizada (Generalized Extreme Value, GEV) apresentou o melhor ajuste para os dados de diâmetro do fuste, permitindo previsões de volume precisas. Este procedimento, que integra detecção automática com modelos de distribuição de diâmetro, oferece uma solução escalável aplicável a pilhas grandes e complexas. Por fim, foi criado um repositório para permitir que utilizadores testem o método proposto. Trabalhos futuros irão focar no refinamento da precisão do modelo e na expansão da sua aplicabilidade a diferentes espécies, condições de produção florestal e características das toras.

Palavras-chave: estimativa de volume de madeira; inventário florestal; silvicultura de precisão; visão computacional; YOLOv9

4.1. INTRODUÇÃO

A quantificação precisa do volume sólido da madeira é fundamental em toda a cadeia produtiva florestal e madeireira, influenciando desde o planejamento da colheita e o processamento até a precificação e a comercialização do produto final. Medições exatas são essenciais para evitar perdas financeiras, pois o volume de madeira impacta diretamente nos custos e lucros (DE MIGUEL-DÍEZ et al., 2022; LÖWE et al., 2019), especialmente em contextos industriais, onde a estimativa da madeira em metros cúbicos é crucial para faturamento e planejamento de custos. A quantificação da madeira é também um elemento-chave para o planejamento e a contabilidade na cadeia de suprimento de madeira, determinando a carga máxima para veículos de transporte e garantindo o cumprimento de regulamentos e normas (CREMER et al., 2020).

Tradicionalmente, a estimativa do volume de madeira empilhada é feita utilizando o fator médio de empilhamento, que pode ser influenciado por variáveis como diâmetro e comprimento das toras, método de empilhamento e tempo de armazenamento no pátio. No entanto, avanços tecnológicos e novas metodologias têm melhorado significativamente a precisão da quantificação de recursos madeireiros utilizando essa abordagem (DE BARROS et al., 2024; DE MIGUEL-DÍEZ et al., 2022; VICENTE GOUVEIA FILHO et al., 2022).

A revolução tecnológica na medição de volume de madeira atingiu um novo patamar com o advento de técnicas avançadas. Técnicas como a varredura a laser (PANAGIOTIDIS; ABDOLLAHNEJAD, 2021), na qual uma carga de madeira é escaneada para criar um modelo 3D detalhado, permitem avaliações volumétricas altamente precisas (NYLINDER; KUBÉNKA; HULTNÄS, 2008). Métodos foto-ópticos utilizam luz e sensores para medir diferentes aspectos de uma pilha de madeira, possibilitando um cálculo mais preciso do volume total (MOSKALIK et al., 2022). Técnicas de classificação de imagens empregam redes neurais artificiais para analisar imagens digitais de pilhas de madeira (CAMPOS; LEITE, 2017; SILVEIRA, 2014; SOARES; PAULA NETO. F.; SOUZA, 2011). Softwares de modelagem 3D também podem ser usados para criar representações digitais com base em diversos pontos de dados, permitindo estimativas volumétricas de alta precisão (HERBON et al., 2015).

O desenvolvimento de equipamentos ainda mais sofisticados, como sistemas ópticos e LiDAR (Light Detection and Ranging), potencializa esses avanços. A tecnologia LiDAR utiliza lasers para medir distâncias, criando nuvens de pontos 3D detalhadas do objeto medido, o que é extremamente preciso para cálculos de volume (PURFÜRST et al., 2023), e pode ser aplicada a tarefas florestais autônomas, como navegação, detecção de árvores e classificação de espécies (GRONDIN et al., 2023). Uma técnica inovadora adicional é a detecção e medição de diâmetros utilizando visão computacional (LU et al., 2024; YU et al., 2023).

O uso de visão computacional e técnicas de aprendizado de máquina na detecção de madeira tem demonstrado elevado potencial, alcançando taxas de acerto superiores a 90% na identificação de espécies. Esses sistemas automatizados oferecem tempos de processamento mais rápidos que os métodos tradicionais, tornando-se economicamente viáveis. As técnicas têm ampla

aplicabilidade em áreas como biometria florestal e são escaláveis, uma vez que modelos de aprendizado de máquina podem ser treinados com extensos conjuntos de dados (SILVA et al., 2022).

Esse progresso é especialmente relevante para a indústria de celulose e papel, que depende de um fornecimento contínuo de madeira para produzir seus produtos e promover a sustentabilidade por meio do manejo responsável dos recursos. O avanço nas tecnologias de quantificação de madeira é crucial para essa indústria, pois garante um fornecimento constante e confiável de matéria-prima. Tempos de processamento mais rápidos na quantificação da madeira traduzem-se diretamente em maiores retornos financeiros para as empresas (DURIGAN et al., 2022). Além disso, essa capacidade de processamento aprimorada é fortalecida por técnicas avançadas de modelagem em biometria florestal.

Pesquisadores têm utilizado modelos de distribuição, como Weibull, log-normal, gamma, beta e Johnson, para modelar a variabilidade nos diâmetros das árvores em povoamentos de *Eucalyptus* e *Pinus*. Essa abordagem envolve ajustar a distribuição aos dados coletados durante inventários florestais. A eficácia dessa técnica está na capacidade de representar com precisão as variações naturais nos diâmetros das árvores observadas nessas plantações específicas (MATEUS; TOMÉ, 2011; PALAHÍ; PUKKALA; TRASOBARES, 2006; SA et al., 2023; SCHMIDT et al., 2020).

Nosso objetivo principal foi desenvolver um procedimento preciso para estimar o volume de madeira de eucalipto empilhada no pátio, eliminando a necessidade de medições manuais demoradas e suscetíveis a erros. Esse procedimento combina a detecção automática de toras com modelos de distribuição de diâmetros. A detecção automática utiliza tecnologia de visão computacional, envolvendo algoritmos de análise de imagens para identificar e contar automaticamente toras individuais dentro de uma pilha, independentemente do tamanho (CASAS et al., 2023). Uma vez identificadas e contadas as toras, entram em ação os modelos de distribuição de diâmetros, que consideram a análise de afilamento obtida no inventário pré-colheita. Utilizando a densidade de probabilidade e o número de toras, procedemos à estimativa do volume sólido total da pilha.

Essa abordagem oferece vantagens significativas. Primeiro, reduz drasticamente o tempo e o esforço necessários para a estimativa de volume, em comparação aos métodos manuais. Segundo, ao automatizar a detecção de toras, o processo torna-se menos suscetível a erros humanos, especialmente em pilhas grandes e complexas. Por fim, a escalabilidade é uma força central desse procedimento: os mesmos princípios básicos de detecção automática e modelos de distribuição de diâmetros podem ser aplicados para estimar, de forma eficiente, o volume de pilhas de madeira muito grandes.

Na indústria florestal, a medição precisa do volume de madeira empilhada é essencial para o gerenciamento eficiente dos recursos e para a avaliação correta do valor comercial. Os métodos tradicionais de medição, que envolvem o uso de trenas ou ferramentas manuais, costumam ser trabalhosos, demorados e sujeitos a erros humanos (PANAGIOTIDIS; ABDOLLAHNEJAD,

2021). Em resposta a essas limitações, pesquisadores têm explorado ativamente novas abordagens para calcular o volume de madeira empilhada. Uma dessas abordagens promissoras é o uso de técnicas de análise de imagens, que aproveitam o poder da inteligência artificial e do aprendizado de máquina para automatizar o processo de medição.

4.2. MATERIAIS E MÉTODOS

4.2.1. Conjunto de dados

O processo envolveu três conjuntos de dados. O primeiro conjunto foi obtido de uma indústria florestal em Minas Gerais, Brasil. Ele inclui medições de diâmetro à altura do peito (DAP) provenientes de inventários pré-colheita de híbridos *Eucalyptus urophylla* × *Eucalyptus grandis*. Essas medições foram realizadas aos 66 e 86 meses de idade, em parcelas retangulares de 400 m², com árvores plantadas no espaçamento de 3 × 3 m. As medições de DAP foram utilizadas para estimar o perfil do fuste das árvores de eucalipto, estabelecendo os diâmetros ao longo do tronco. No total, 3350 árvores foram medidas aleatoriamente.

O segundo conjunto de dados incluiu imagens e vídeos de madeira de eucalipto empilhada, estabelecidos conforme a metodologia descrita por Casas et al. (2023), cujo mesmo conjunto foi adotado em nossa pesquisa. Os vídeos foram utilizados para contar automaticamente o número de toras empilhadas.

Para validar o método proposto de estimativa de volume, foi empregado um terceiro conjunto de dados. Esse conjunto consistiu em dados de toras medidos manualmente, cujos volumes foram calculados utilizando a fórmula de Smalian. Os resultados dessa fórmula foram então comparados aos volumes estimados obtidos pelo processo de estimativa aqui proposto.

A Figura 4.1A mostra a distribuição das classes diamétricas (DAP) do inventário pré-colheita. O eixo x representa a classe de diâmetro e o eixo y representa o número de árvores em cada classe. A maioria das árvores apresenta classe de diâmetro de 18 cm. A Figura 4.1B é um exemplo da imagem de frame utilizada para a contagem automática das toras. A imagem de frame mostra madeira empilhada com uma variedade de diâmetros.

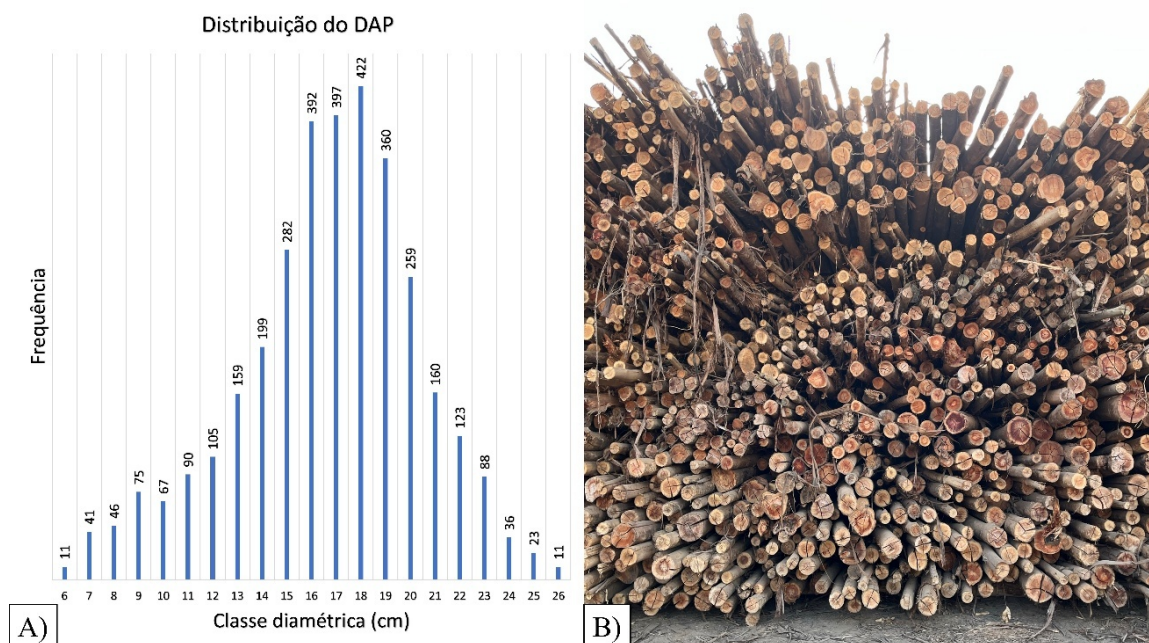


Figura 4.1. A) Distribuição das classes diamétricas (DAP) do inventário pré-colheita e B) exemplo da imagem de frame utilizada para a contagem automática das toras.

4.2.2. Volume sólido por meio de distribuição de probabilidade

O modelo de afilamento de Garay (GARAY L., 1979), obtido por Gomide (GOMIDE, 2016) (Equação 4.1), foi aplicado para determinar o diâmetro ao longo do fuste utilizando o primeiro conjunto de dados. Essa equação caracteriza o formato do fuste de árvores de eucalipto. Os diâmetros (d) dos fustes foram medidos em diversas alturas (h) ao longo do tronco, especificamente em 0,3 m, 6,3 m, 12,3 m, 18,3 m, 24,3 m e 30,3 m. Essas medições foram realizadas a cada 6 m, uma vez que as peças de madeira empilhada possuíam cada uma 6 m de comprimento.

$$\frac{d}{DAP} = 1,171261 + 0,30096 \ln \left[1 - 0,96233 \left(\frac{h}{H} \right)^{0,30445} \right] \quad (4.1)$$

Onde d representa o diâmetro estimado (cm) na altura h , H representa a altura total da árvore (m) e DAP representa o diâmetro (cm) a 1,3 m de altura.

Utilizamos código em Python, importando o módulo stats da biblioteca scipy, o que nos permitiu ajustar 32 diferentes tipos de distribuições de probabilidade aos dados de diâmetro (d). Essa abordagem possibilitou explorar uma ampla gama de pressupostos estatísticos e obter uma compreensão mais abrangente dos padrões subjacentes aos dados.

Os modelos de distribuição foram avaliados utilizando os critérios de log-verossimilhança, Critério de Informação de Akaike (AIC) e Critério de Informação Bayesiano (BIC). O modelo com o melhor desempenho estatístico nos dados foi selecionado. A avaliação considerou a capacidade dos modelos de ajustarem-se com precisão aos dados: o critério de log-verossimilhança mede a probabilidade de o modelo ter gerado os dados observados, enquanto os critérios AIC e BIC são medidas mais complexas que também levam em conta o número de parâmetros do

modelo.

Posteriormente, o volume sólido das toras sem casca foi determinado para as madeiras empilhadas. Para isso, utilizamos a fórmula de Huber (HUSCH; MILLER; BEERS, 1993), com uma modificação no posicionamento da área seccional. Em vez de usar a área seccional basal no meio da tora, utilizamos a face da tora como área seccional basal. O volume sólido resultante da tora foi então multiplicado pela probabilidade do número de toras dentro de cada classe diamétrica (Equação 4.2), derivada dos resultados do segundo conjunto de dados.

$$V = 0,00007854 \times L \times D^2 \times Z \quad (4.2)$$

Onde V é o volume sólido da tora (m^3), L é o comprimento da tora (m), que no nosso estudo foi de 6 m, D é a classe diamétrica (cm) da face da tora, e Z é a probabilidade do número de toras por classe diamétrica.

Para demonstrar a confiabilidade desse processo, foi realizada uma série de simulações. Cada simulação consistiu em uma pilha hipotética de 5000 toras, cujos diâmetros foram obtidos por amostragem aleatória do banco de dados de diâmetros gerado pelo modelo de afilamento a partir do inventário pré-corte. Para cada pilha simulada, o volume de referência foi calculado somando-se o volume individual das 5000 toras sorteadas, ao passo que o volume estimado foi obtido pelo procedimento aqui proposto, isto é, pela combinação da distribuição de probabilidade ajustada com a contagem total de toras. Foram realizadas 30 simulações independentes, o que permitiu avaliar a estabilidade da estimativa frente à variação amostral dos diâmetros.

Além disso, foi realizada uma análise comparativa entre os resultados obtidos por esse processo e aqueles derivados do método tradicional de Smalian (terceiro conjunto de dados), que se baseia em dados coletados manualmente.

Foram estabelecidas 35 classes de diâmetro, nas quais calculamos tanto a área basal quanto o volume, utilizando a fórmula de Huber adaptada. Em seguida, aplicamos o modelo de distribuição Generalized Extreme Value (GEV) para determinar a distribuição de probabilidade. O exemplo apresentado na Tabela 4.3 ilustra esse processo.

Em seguida, estimamos o volume para cada classe diamétrica multiplicando o volume observado (m^3) pela densidade de probabilidade. Além disso, esse volume estimado passou por uma correção baseada na diferença da densidade de probabilidade.

Para validar o processo, foram realizadas duas medições manuais de diâmetro em ambas as extremidades de uma amostra de toras. Foram medidas 200 toras, todas com 6 m de comprimento. O volume sólido foi determinado utilizando o método de Smalian. A Tabela 4.5 apresenta os valores médios, máximos e mínimos para as variáveis analisadas.

4.2.3. Detecção e segmentação automática de toras

A contagem automática de toras foi realizada utilizando o modelo YOLO, versão 9. Esse processo seguiu a metodologia e utilizou o mesmo conjunto de dados estabelecido por Casas et

al. (2023), que propõe o uso de técnicas de visão computacional para contar com precisão toras em pilhas de madeira por meio de vídeo.

O treinamento do modelo YOLOv9 foi conduzido em um ambiente Python personalizado, configurado para aproveitar a aceleração por GPU. O script de treinamento foi parametrizado com informações específicas, como caminho do conjunto de dados, tamanho da imagem, dispositivo, tamanho do batch, configuração do modelo e hiperparâmetros, visando treinar o modelo por até 300 épocas, com paciência de early stopping de 50 épocas.

A arquitetura do YOLOv9 aborda a perda de informações em modelos profundos de detecção de objetos introduzindo o Programmable Gradient Information (PGI). O PGI utiliza um Auxiliary Reversible Branch, que preserva todos os detalhes da entrada, e um componente de Multi-level Auxiliary Information, que coleta gradientes de diferentes pontos de predição. Essa abordagem combinada garante que informações de treinamento confiáveis cheguem ao núcleo da rede (GELAN), uma arquitetura leve que combina fluxo de gradiente eficiente com convoluções padrão para uso otimizado de parâmetros e, conseqüentemente, maior precisão na detecção de objetos.

As anotações utilizadas no treinamento seguiram o mesmo critério adotado no Capítulo 3: para a tarefa de detecção, caixas delimitadoras retangulares circunscrevendo exclusivamente a face exposta de cada tora; para a tarefa de segmentação, polígonos delimitando o contorno dessa mesma face, sem incluir o corpo longitudinal da tora nem o fundo da cena. Dessa forma, tanto a caixa quanto a máscara referem-se à seção transversal da tora, que é a grandeza de interesse para a estimativa da área basal.

Para detectar e segmentar toras de forma eficiente, foi adotada uma abordagem em duas etapas. O YOLOv9-c, versão compacta do detector YOLOv9, realiza a detecção de toras com alta precisão, aproveitando o conceito PGI, que minimiza a perda de informação e preserva características essenciais para uma identificação precisa. Já o Gelan-c-seg, variante compacta da arquitetura GELAN do YOLOv9, realiza a segmentação das toras. Esse design flexível permite selecionar blocos computacionais ideais conforme a capacidade de processamento do dispositivo, equilibrando precisão, velocidade de inferência e uso de recursos.

As métricas de desempenho fornecem informações sobre o desempenho do modelo nos conjuntos de treinamento, validação e teste. Essas métricas incluem: Precisão de Caixa (P), Revocação de Caixa (R), mAP50 de Caixa (mean Average Precision a 50% de intersection-over-union), mAP50-95 de Caixa (mean Average Precision em múltiplos limiares), Precisão de Máscara, Revocação de Máscara, mAP50 de Máscara e mAP50-95 de Máscara.

O sistema utilizado para o treinamento possui processador Intel(R) Core(TM) i7-13700H de 13ª geração, com frequência de 2,40 GHz, 8,00 GB de RAM instalada e sistema operacional de 64 bits em arquitetura x64. Conta com uma GPU NVIDIA GeForce RTX 4060, equipada com 8 GB de memória GDDR6 e largura de barramento de 128 bits.

Para este exemplo de aplicação, optou-se pela detecção devido à sua eficiência computacional. A segmentação, embora potencialmente mais informativa, exige recursos computacionais.

cionais significativamente maiores, muitas vezes necessitando a redução do tamanho do lote (batch size). A detecção de objetos foi empregada para estimar o número de toras de madeira na pilha utilizando um vídeo. O modelo YOLOv9-c (WANG; YEH; LIAO, 2024) treinado foi utilizado para este fim, e as anotações foram posteriormente estabelecidas utilizando o Roboflow Supervision, que é um conjunto de ferramentas reutilizáveis de visão computacional projetadas para facilitar diversas tarefas, como carregamento de conjuntos de dados, desenho de detecções em imagens ou vídeos e contagem de detecções.

A Figura 4.4 apresenta um guia passo a passo para estimar o volume total de madeira empilhada de eucalipto. O processo envolveu quatro etapas:

4.3. RESULTADOS

4.3.1. Volume sólido por meio de distribuição de probabilidade

As métricas de desempenho do modelo YOLOv9 treinado neste capítulo foram avaliadas para precisão, revocação (*recall*) e *mean average precision* (mAP50). Na tarefa de detecção, o modelo alcançou precisão de 0,796, revocação de 0,846 e mAP50 de 0,856 no conjunto de treinamento; precisão de 0,798, revocação de 0,848 e mAP50 de 0,860 no conjunto de validação; e precisão de 0,764, revocação de 0,812 e mAP50 de 0,818 no conjunto de teste. Essas métricas fornecem informações sobre o desempenho do modelo na detecção precisa de objetos, sendo que o conjunto de validação apresentou o melhor desempenho geral. O detalhamento completo das métricas de detecção e de segmentação, incluindo os valores de mAP50-95, é apresentado na subseção *Análise abrangente do desempenho do modelo e técnica de processamento de imagens*.

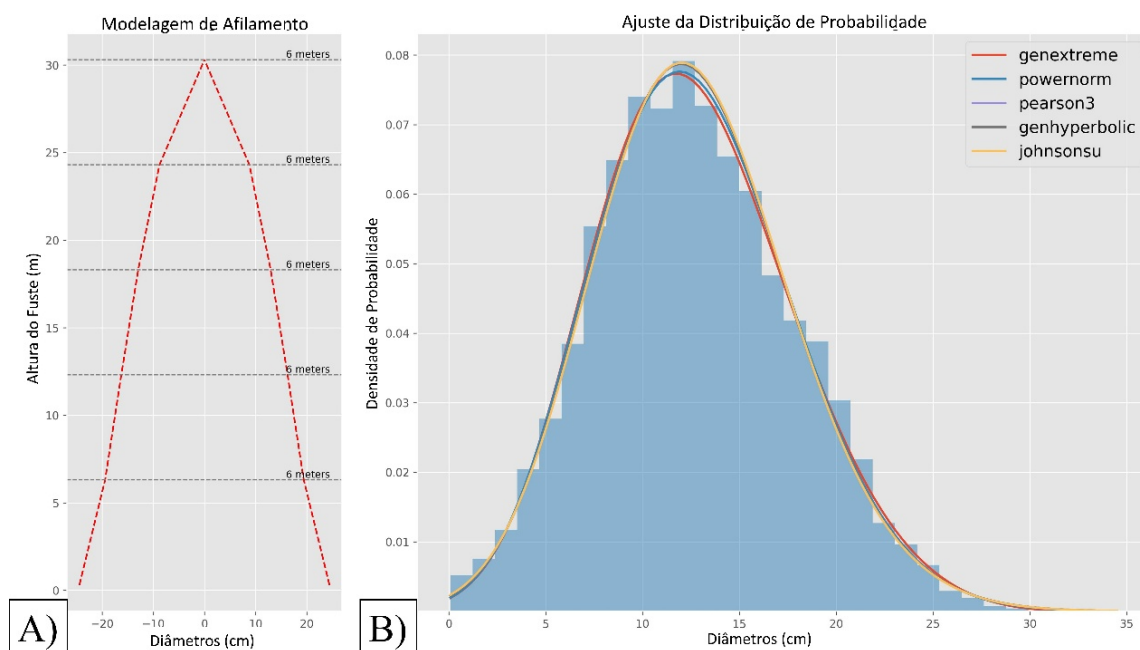


Figura 4.2. A) Modelagem de afilamento de eucalipto aos 66 e 86 meses, com alturas do fuste a cada 6 m, e B) ajuste da distribuição de probabilidade dos 5 melhores modelos de distribuição.

As probabilidades de distribuição foram ajustadas aos dados de diâmetro gerados pelo modelo de afilamento (Taper model) (Tabela 4.1). A Figura 4.2B apresenta as 5 distribuições com melhor ajuste. Entre todos os modelos de distribuição ajustados, a distribuição Generalized Extreme Value (GEV) apresentou o melhor ajuste aos dados (Tabela 4.2). Essa distribuição apresentou log-verossimilhança de $-38.865,48235$, AIC de $77.736,9647$ e BIC de $77.759,33114$.

Os parâmetros foram:

$$c=0,1987076665765512$$

$$\mu=10,686786238491923$$

$$\sigma=4,858693770711352$$

A função densidade de probabilidade da distribuição GEV ajustada é apresentada na Equação 4.3, escrita na mesma parametrização adotada pela biblioteca SciPy (VIRTANEN et al., 2020), que foi a utilizada na estimação dos parâmetros por máxima verossimilhança:

$$f(x; c, \mu, \sigma) = \frac{1}{\sigma} \left[1 - c \left(\frac{x - \mu}{\sigma} \right) \right]^{\frac{1}{c} - 1} \exp \left\{ - \left[1 - c \left(\frac{x - \mu}{\sigma} \right) \right]^{\frac{1}{c}} \right\}, \quad 1 - c \left(\frac{x - \mu}{\sigma} \right) > 0 \quad (4.3)$$

Onde:

x = valores aleatórios de diâmetro (cm);

μ = parâmetro de localização (location parameter), valor em torno do qual a distribuição é centrada;

σ = parâmetro de escala (scale parameter), que determina a largura da distribuição;

c = parâmetro de forma (shape parameter), que define o peso das caudas da distribuição.

Cabe destacar que a biblioteca SciPy adota, para o parâmetro de forma, convenção de sinal oposta à da formulação clássica da família GEV proposta por Jenkinson (1955) e consolidada por Coles (2001), na qual a densidade é escrita com o termo $[1 + \xi(x - \mu)/\sigma]$. As duas parametrizações são equivalentes, bastando fazer $\xi = -c$. Assim, o valor estimado $c = 0,1987$ corresponde a $\xi = -0,1987 < 0$, o que caracteriza a distribuição do tipo III (Weibull) da família GEV, isto é, uma distribuição assimétrica à direita, porém com cauda superior limitada, e não do tipo II (Fréchet), de cauda pesada e ilimitada (COLES, 2001; VIRTANEN et al., 2020).

O limite superior do suporte da distribuição ajustada é dado por $x_{\max} = \mu + \sigma/c = 10,6868 + 4,8587/0,1987 = 35,14$ cm. Esse resultado é coerente, do ponto de vista biológico e operacional, com um povoamento clonal de eucalipto destinado à produção de celulose, cujos diâmetros são naturalmente limitados superiormente, e justifica a adoção das 35 classes diamétricas empregadas neste estudo (Tabela 4.3), uma vez que a probabilidade associada a diâmetros superiores a 35 cm é nula por construção.

Tabela 4.1. Parâmetros de diferentes modelos de distribuições de diâmetro ajustadas.

Distribuições	Número de parâmetros	Parâmetros
---------------	----------------------	------------

genextreme	3	c = 0,1987; loc = 10,6868; scale = 4,8587
powernorm	3	c = 0,2435; loc = 7,4837; scale = 3,1517
pearson3	3	skew = 0,2707; loc = 12,6826; scale = 5,1059
genhyperbolic	5	p = 53,381; a = 10,8625; b = 9,1421; loc = -21,384; scale = 1,1975
johnsonsu	4	a = -17,6552; b = 10,8512; loc = -38,4944; scale = 20,8317
skewnorm	3	a = 1,383; loc = 8,3465; scale = 6,6933
nct	4	df = 91,7495; nc = 8,3284; loc = -23,212; scale = 4,2744
gennorm	3	beta = 2,3139; loc = 12,7533; scale = 7,7057
exponnorm	3	K = 0,4677; loc = 10,5195; scale = 4,6249
tukeylambda	3	lam = 0,161; loc = 12,7187; scale = 3,6258
norm	2	loc = 12,6825; scale = 5,0989
t	3	df = 626.108.445,3755579; loc = 12,6825; scale = 5,0989
crystalball	4	beta = 1632,97; m = 1567,37; loc = 12,6825; scale = 5,0989
norminvgauss	4	a = 706,821; b = -58,852; loc = 23,9557; scale = 134,9308
loggamma	3	c = 982,7692; loc = -1099,26; scale = 161,3869
genlogistic	3	c = 1,9187; loc = 9,4165; scale = 3,5025
dweibull	3	c = 1,2962; loc = 12,747; scale = 4,4377
logistic	2	loc = 12,5613; scale = 2,9444
hypsecant	2	loc = 12,4959; scale = 3,5308
dgamma	3	a = 1,4013; loc = 12,7469; scale = 2,9404
gumbel_r	2	loc = 10,1828; scale = 4,7257
laplace_asymmetric	3	kappa = 0,8559; loc = 11,4197; scale = 4,042

laplace	2	loc = 12,4124; scale = 4,111
moyal	2	loc = 9,184; scale = 3,1994
gumbel_1	2	loc = 15,2768; scale = 5,2202
skewcauchy	3	a = 0,1771; loc = 11,2397; scale = 3,1697
cauchy	2	loc = 12,2922; scale = 3,1824
truncnorm	4	a = -0,1347; b = 2,4404; loc = 1,8882; scale = 13,3663
truncweibull_min	5	c = 1,4639; a = 0,0021; b = 1,5466; loc = 0,0384; scale = 22,2831
kappa4	4	h = 1,2223; k = 0,6382; loc = -4,6158; scale = 24,9656
loguniform	4	a = 0,1589; b = 34,5716; loc = -0,0709; scale = 1
reciprocal	4	a = 0,1589; b = 34,5716; loc = -0,0709; scale = 1

Tabela 4.2. Comparação dos resultados dos pressupostos estatísticos das distribuições diamétricas ajustadas.

Distribuições	Log-verossimilhança	AIC	BIC
genextreme	-38.865,48235	77.736,96	77.759,33
powernorm	-38.872,69188	77.751,38	77.773,75
pearson3	-38.885,94178	77.777,88	77.800,25
genhyperbolic	-38.886,11281	77.782,23	77.819,50
johnsonsu	-38.889,94327	77.787,89	77.817,71
skewnorm	-38.895,68449	77.797,37	77.819,74
nct	-38.898,2981	77.804,60	77.834,42
gennorm	-38.924,37054	77.854,74	77.877,11
exponnorm	-38.926,5468	77.859,09	77.881,46
tukeylambda	-38.937,80938	77.881,62	77.903,99
norm	-38.946,884	77.897,77	77.912,68
t	-38.946,88401	77.899,77	77.922,13
crystalball	-38.946,884	77.901,77	77.931,59
norminvgauss	-38.951,72346	77.911,45	77.941,27
loggamma	-38.962,83237	77.931,66	77.954,03
genlogistic	-39.043,44935	78.092,90	78.115,27
dweibull	-39.145,93283	78.297,87	78.320,23
logistic	-39.148,66907	78.301,34	78.316,25
hypsecant	-39.322,84643	78.649,69	78.664,60

dgamma	-39.323,03436	78.652,07	78.674,44
gumbel_r	-39.381,48912	78.766,98	78.781,89
laplace_asymmetric	-39.636,56074	79.279,12	79.301,49
laplace	-39.698,96806	79.401,94	79.416,85
moyal	-39.978,00866	79.960,02	79.974,93
gumbel_l	-40.244,35274	80.492,71	80.507,62
skewcauchy	-41.401,42085	82.808,84	82.831,21
cauchy	-41.465,57412	82.935,15	82.950,06
truncnorm	-42.241,77091	84.491,54	84.521,36
truncweibull_min	-42.589,18839	85.188,38	85.225,65
kappa4	-43.738,56063	87.485,12	87.514,94
loguniform	-52.708,57524	105.425,15	105.454,97
reciprocal	-52.708,57524	105.425,15	105.454,97

4.3.2. Simulação da estimativa de volume sólido por densidade de probabilidade

Tabela 4.3. Estimativa do volume sólido por classe diamétrica utilizando densidade de probabilidade.

Classe diamétrica (cm)	Área basal (m ²)	Volume (m ³)	Probabilidade da classe	Número esperado de toras	Volume estimado (m ³)	Volume estimado corrigido (m ³)
1	0,0001	0,0005	0,003706	18,5292	0,0087	0,0088
2	0,0003	0,0019	0,006926	34,6292	0,0653	0,0655
3	0,0007	0,0042	0,011841	59,2046	0,2511	0,2519
4	0,0013	0,0075	0,018656	93,2791	0,7033	0,7054
5	0,002	0,0118	0,027273	136,3658	1,6065	1,6114
6	0,0028	0,017	0,037232	186,1611	3,1581	3,1678
7	0,0038	0,0231	0,047746	238,7283	5,5124	5,5292
8	0,005	0,0302	0,05783	289,1498	8,7206	8,7471
9	0,0064	0,0382	0,06649	332,4494	12,6897	12,7283
10	0,0079	0,0471	0,072901	364,5044	17,1769	17,2291
11	0,0095	0,057	0,07654	382,7017	21,8216	21,888
12	0,0113	0,0679	0,077242	386,2079	26,2075	26,2872
13	0,0133	0,0796	0,075172	375,8605	29,9333	30,0243
14	0,0154	0,0924	0,070757	353,784	32,6765	32,7759
15	0,0177	0,106	0,064575	322,876	34,2341	34,3383
16	0,0201	0,1206	0,057259	286,2962	34,538	34,643
17	0,0227	0,1362	0,04941	247,0503	33,6453	33,7476
18	0,0254	0,1527	0,041542	207,709	31,7133	31,8098
19	0,0284	0,1701	0,034053	170,2653	28,9651	29,0532
20	0,0314	0,1885	0,027221	136,1037	25,6549	25,733
21	0,0346	0,2078	0,02121	106,0482	22,0385	22,1056
22	0,038	0,2281	0,016091	80,4538	18,3499	18,4057
23	0,0415	0,2493	0,011863	59,3153	14,7864	14,8314

24	0,0452	0,2714	0,008475	42,3732	11,5015	11,5365
25	0,0491	0,2945	0,005841	29,2064	8,602	8,6282
26	0,0531	0,3186	0,003861	19,3071	6,1504	6,1691
27	0,0573	0,3435	0,002427	12,1369	4,1694	4,1821
28	0,0616	0,3695	0,001433	7,1668	2,6478	2,6558
29	0,0661	0,3963	0,000781	3,9037	1,5471	1,5518
30	0,0707	0,4241	0,000381	1,9067	0,8087	0,8111
31	0,0755	0,4529	0,000159	0,7968	0,3608	0,3619
32	0,0804	0,4825	0,000052	0,2612	0,126	0,1264
33	0,0855	0,5132	0,000011	0,0556	0,0285	0,0286
34	0,0908	0,5448	0,000001	0,0044	0,0024	0,0024
35	0,0962	0,5773	0	0	0	0
SOMA	1,171	7,0262	0,997	4984,791	440,4017	441,7413

Aplicando-se a função densidade de probabilidade ajustada a uma pilha hipotética de 5000 toras, a soma do número esperado de toras nas 35 classes diamétricas resultou em 4984,79 toras, e não em 5000. Essa diferença de 15,21 toras decorre de a soma das densidades pontuais avaliadas nos centros de classe não totalizar exatamente a unidade, e foi utilizada como fator de correção dos volumes estimados.

O volume total estimado foi de 441,7413 m³ para a pilha hipotética de 5000 toras. A Tabela 4.4 apresenta os volumes de referência e estimados para as 30 simulações.

Tabela 4.4. Simulação para a estimativa do volume sólido de 30 pilhas de 5000 toras.

Amostra (pilha)	Contagem (toras)	Volume total simulado (m ³)	Volume total estimado pela distribuição GEV (m ³)	RE%
1	5000	455,483	441,741	-3,017
2	5000	460,828	441,741	-4,142
3	5000	461,776	441,741	-4,339
4	5000	460,066	441,741	-3,983
5	5000	460,477	441,741	-4,069
6	5000	468,204	441,741	-5,652
7	5000	462,269	441,741	-4,441
8	5000	462,278	441,741	-4,442
9	5000	462,725	441,741	-4,535
10	5000	464,596	441,741	-4,919
11	5000	460,624	441,741	-4,099
12	5000	461,415	441,741	-4,264
13	5000	467,299	441,741	-5,469
14	5000	465,83	441,741	-5,171
15	5000	466,656	441,741	-5,339
16	5000	468,722	441,741	-5,756
17	5000	456,072	441,741	-3,142
18	5000	462,603	441,741	-4,510
19	5000	464,564	441,741	-4,913
20	5000	457,898	441,741	-3,528
21	5000	457,86	441,741	-3,520
22	5000	466,922	441,741	-5,393
23	5000	457,56	441,741	-3,457
24	5000	447,967	441,741	-1,390
25	5000	464,493	441,741	-4,898
26	5000	467,186	441,741	-5,446
27	5000	466,392	441,741	-5,285

28	5000	473,405	441,741	-6,688
29	5000	461,615	441,741	-4,305
30	5000	469,534	441,741	-5,919

Verificamos que a metodologia de estimativa de volume subestimou consistentemente o volume observado, apresentando um erro relativo médio (RE%) de -4,535% nas 30 simulações. Esse valor é integralmente explicado pela diferença entre o segundo momento da distribuição ajustada e o das amostras simuladas, detalhada na subseção seguinte (187,50 cm² contra 196,41 cm², ou -4,54%), o que confirma a coerência interna do procedimento: o desvio observado é atribuível ao ajuste da cauda superior da GEV, e não a erros de discretização, de acumulação de probabilidades ou de cálculo volumétrico. Esse processo de validação demonstra a robustez e a confiabilidade da nossa metodologia de estimativa de volume, uma vez que produz, de forma consistente, estimativas precisas.

4.3.3. Verificação por meio da distribuição acumulada

A coerência do procedimento descrito foi verificada por meio da função de distribuição acumulada (FDA) da GEV, apresentada na Equação 4.4 na mesma parametrização da Equação 4.3 (COLES, 2001; VIRTANEN et al., 2020):

$$F(x; c, \mu, \sigma) = \exp \left\{ - \left[1 - c \left(\frac{x - \mu}{\sigma} \right) \right]^{\frac{1}{c}} \right\} \quad (4.4)$$

Rigorosamente, a probabilidade associada a uma classe diamétrica de amplitude 1 cm, centrada em D_j , corresponde à diferença da distribuição acumulada avaliada nos limites da classe, e não ao valor pontual da função densidade (Equação 4.5):

$$p_j = F(D_j + 0,5) - F(D_j - 0,5) \quad (4.5)$$

Onde: p_j é a probabilidade da classe diamétrica j ; D_j é o centro de classe (cm); e $F(\cdot)$ é a função de distribuição acumulada definida na Equação 4.4.

A aplicação da Equação 4.5 às 35 classes resultou em $\sum p_j = 0,99688$. A fração remanescente de 0,312% concentra-se integralmente na cauda inferior, correspondente a diâmetros menores que 0,5 cm, sem significado físico para toras comerciais; na cauda superior a probabilidade é nula, pois o suporte da distribuição é limitado em 35,14 cm, conforme discutido na Equação 4.3. Normalizando as probabilidades para o suporte efetivamente utilizado, de modo que $\sum p_j = 1$, o volume estimado da pilha hipotética de 5000 toras passa a 441,98 m³, contra os 441,74 m³ obtidos pelo procedimento de correção baseado na diferença de 15,21 toras, o que representa discrepância de apenas 0,05%. Um valor próximo, 441,78 m³, é recuperado pelo cálculo exato da esperança $E[D^2]$ sob a distribuição ajustada. Conclui-se, portanto, que o uso da densidade pontual e a correção adotada não introduzem viés relevante nas estimativas, embora

a formulação por meio da distribuição acumulada (Equação 4.5) seja preferível, por dispensar o fator de correção baseado na diferença de contagem: após a normalização das probabilidades para o intervalo diamétrico considerado, sua soma é igual à unidade.

Essa verificação permite ainda localizar com precisão a origem do erro relativo médio de $-4,535\%$ observado nas 30 simulações da Tabela 4.4. Como o volume é proporcional ao quadrado do diâmetro, o valor esperado $E[D^2]$ sob a GEV ajustada, igual a $187,50 \text{ cm}^2$, é $4,54\%$ inferior ao segundo momento amostral dos diâmetros das pilhas simuladas, igual a $196,41 \text{ cm}^2$. Ou seja, a subestimativa sistemática decorre do ajuste da cauda superior da distribuição, justamente a região que concentra as classes de maior contribuição volumétrica, e não da discretização em classes diamétricas nem do fator de correção empregado.

4.3.4. Validação de um método de estimativa de volume de toras por meio de comparação com o método de Smalian

Tabela 4.5. Relatório dos valores médios, máximos e mínimos das variáveis medidas.

Amostra	Contagem	Variável	Média	Máximo	Mínimo
Amostra	Contagem	Variável	(cm)	(cm)	(cm)
Toras	200	Diâmetro da primeira face da tora	12,99	28	2,4
Toras	200	Diâmetro da segunda face da tora	12,81	29	2,7

A Tabela 4.6 apresenta uma análise comparativa entre o volume observado pelo método de Smalian, considerado como volume observado, e o procedimento proposto neste estudo, considerado como volume estimado. A diferença volumétrica foi de $-0,1045 \text{ m}^3$, correspondente a um erro relativo de $-0,5898\%$. Os resultados demonstram a confiabilidade do procedimento de estimativa proposto.

Tabela 4.6. Comparação do volume estimado pelo processo proposto com o volume observado obtido pelo método de Smalian.

Amostra	Contagem	Volume observado (m^3)	Volume estimado (m^3)	Diferença (m^3)	Erro (%)
Toras	200	17,72	17,6161	$-0,1045$	$-0,5898$

4.3.5. Análise abrangente do desempenho do modelo e técnica de processamento de imagens

4.3.5.1. Segmentação

O treinamento foi interrompido antecipadamente, pois não houve melhora nos últimos 50 epochs. Os melhores resultados foram obtidos no epoch 137, e o melhor modelo foi salvo.

No total, foram completados 188 epochs, levando 41,988 h. O modelo alcançou:

- Box Precision: 0,851 - 85,1% das bounding boxes previstas estavam corretas;
- Box Recall: 0,852 - 85,2% das bounding boxes reais foram corretamente identificadas;
- Box mAP50: 0,877 - alta precisão na detecção com overlap de 50%;
- Box mAP50-95: 0,696 - medida mais rigorosa considerando múltiplos limiares.

Para as máscaras de segmentação:

- Mask Precision: 0,830
- Mask Recall: 0,834
- Mask mAP50: 0,856
- Mask mAP50-95: 0,514

Na validação, o desempenho foi ligeiramente melhor, sugerindo boa generalização:

- Box Precision: 0,851
- Box Recall: 0,854
- Box mAP50: 0,878
- Box mAP50-95: 0,702
- Mask Precision: 0,820
- Mask Recall: 0,827
- Mask mAP50: 0,847
- Mask mAP50-95: 0,525
- No teste, o desempenho variou:
- Box Precision: 0,928 - maior precisão;

- Box Recall: 0,649 - mais falhas na detecção;
- Box mAP50: 0,771
- Box mAP50-95: 0,545
- Mask Precision: 0,875
- Mask Recall: 0,611
- Mask mAP50: 0,731
- Mask mAP50-95: 0,445

Velocidade de processamento:

- Pré-processamento: 4,9 ms/imagem
- Inferência: 42,3 ms/imagem
- Non-Maximum Suppression (NMS): 6,8 ms/imagem
- Resolução: 640 × 640 px

4.3.5.2. Detecção

O treinamento durou 35,658 h, com conclusão no epoch 193. O early stopping interrompeu o processo após 50 epochs sem melhorias, com o melhor desempenho no epoch 142.

Treinamento:

- Precisão: 0,796
- Revocação: 0,846
- mAP50: 0,856
- mAP50-95: 0,477

Validação:

- Precisão: 0,798
- Revocação: 0,848
- mAP50: 0,860
- mAP50-95: 0,480

Teste:

- Precisão: 0,764
- Revocação: 0,812
- mAP50: 0,818
- mAP50-95: 0,467

Velocidade de processamento:

- Pré-processamento: 2,0-2,6 ms/imagem
- Inferência: 159,7-166,0 ms/imagem
- NMS: 21,7-36,0 ms/imagem
- Resolução: 640×640 px

Figura 4.3 apresenta uma comparação lado a lado de duas técnicas de processamento de imagem: segmentação (à esquerda) e detecção (à direita). Na imagem de segmentação, as toras estão destacadas com máscaras vermelhas translúcidas, cobrindo a seção transversal de cada tora para ilustrar a tentativa do modelo de segmentar e isolar individualmente cada peça da madeira do fundo. Na imagem de detecção, as toras são delimitadas por caixas vermelhas, marcando a posição da extremidade de cada tora.

Essa comparação destaca as abordagens distintas dos dois métodos. A segmentação fornece um contorno detalhado de toda a região ocupada pelo objeto, produzindo máscaras precisas que o separam do seu entorno. A detecção, por outro lado, localiza os objetos desenhando caixas delimitadoras ao seu redor, o que fornece a localização e a contagem dos objetos, embora sem os detalhes de forma que a segmentação proporciona.

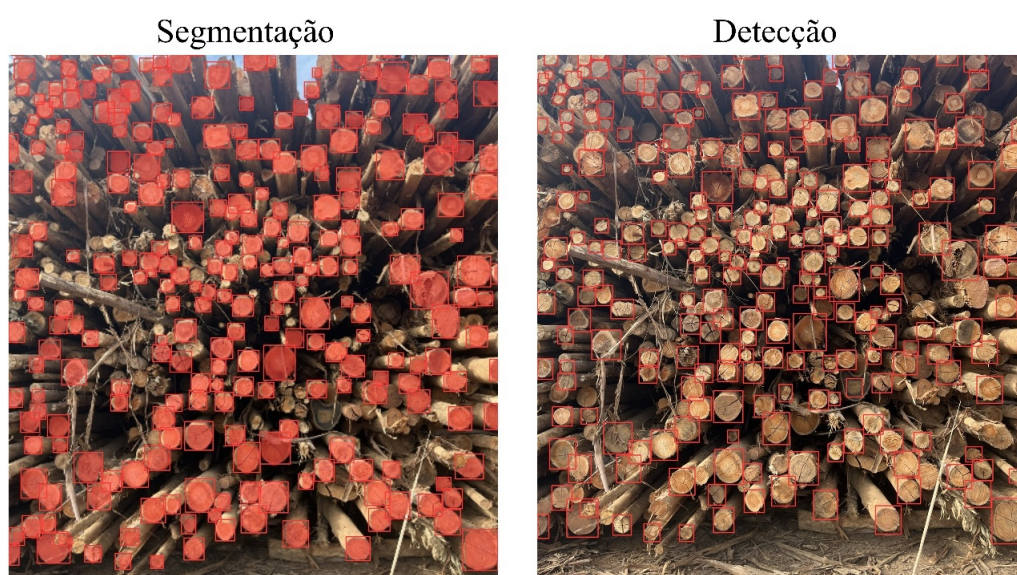


Figura 4.3. Comparação das técnicas de segmentação e detecção no processamento de imagens.

4.3.6. Aplicação da estimativa do volume de madeira empilhada por meio de contagem automática

A contagem total estimada de toras na pilha foi de 2.698. Considerando este valor, o volume sólido de madeira na pilha foi estimado pela metodologia proposta (Equação 4.2), aplicando-se o volume médio por tora derivado da Tabela 4.3, de 441,7413 m³ para 5.000 toras, ou seja, 0,088348 m³ por tora. O resultado é de 238,36 m³.

O Vídeo S1 apresenta um caso real de pilhas em uma indústria florestal de eucalipto para produção de celulose. É evidente que existe o desafio de melhorar a detecção de toras de formato irregular e toras ocultas ao longo da pilha.

Base de dados: Dados de diâmetro em alturas específicas derivados da análise do afilamento do tronco (taper analysis) do inventário pré-corte.

Distribuição de probabilidade: A distribuição de probabilidade dos diâmetros do tronco.

Contagem automática: Técnicas de visão computacional, como detecção ou segmentação, foram utilizadas.

Volume total da pilha: O volume total da madeira empilhada foi estimado utilizando a Equação 4.2.



Figura 4.4. Guia passo a passo para estimar o volume total de madeira de eucalipto empilhada.

4.4. DISCUSSÃO

Um estudo testou medições de diâmetro de toras em ambiente florestal utilizando os modelos YOLOv3 e DeepLabv3+ para identificação e segmentação de toras, mesmo em ambientes com sobreposição de extremidades. O estudo obteve uma precisão média (mAP) de 97,28 % para detecção do diâmetro e uma média de interseção sobre união (IoU) de 92,22 % para segmentação, indicando alta precisão do modelo. A metodologia atingiu uma precisão de medição de volume com erro de 1,2 % e erro médio de diâmetro de 0,73 %, ambos dentro dos padrões da indústria (LU et al., 2024).

Utilizando o Mask R-CNN para detecção precisa das extremidades das toras (taxa de de-

tecção de 98,2 %, IoU da máscara de 0,943 e precisão média de 0,796), o sistema empregou uma câmara estéreo binocular para calcular o tamanho da extremidade da tora. Um algoritmo de busca rotacional de diâmetro identificou os diâmetros curto e longo. Comparado às medições manuais, o método proposto apresentou erro médio de apenas 5,3 mm, com erros individuais de 5,7 mm para o diâmetro curto e 7,19 mm para o longo (YU et al., 2023).

Outro estudo buscou melhorar a detecção das extremidades de toras de eucalipto empilhadas utilizando um modelo YOLOv4-Tiny aprimorado. As principais modificações incluíram o uso de caixas âncora em múltiplas escalas agrupadas por K-median, pooling piramidal espacial (SPP) para lidar com diferentes escalas e limiarização suave com squeeze and excitation (SE) para melhor alinhamento das bounding boxes. Essas melhorias resultaram em desempenho superior ao YOLOv4-Tiny e suas variantes, atingindo alta precisão (93,97 %), recall (95,34 %) e F1-score (0,95) no conjunto de teste. O modelo também apresentou menor perda CIOU e alta precisão média (AP), indicando sua eficácia para aplicações práticas na área florestal (LIN et al., 2022).

Nossos resultados demonstram que o modelo YOLOv9 apresenta bom desempenho tanto nas tarefas de detecção quanto de segmentação, especialmente em termos de precisão e recall. As altas precisões de caixas e máscaras no conjunto de teste (0,928 e 0,875, respectivamente) indicam forte desempenho na identificação e segmentação precisa das extremidades das toras. Comparando com os testes realizados por Lu et al. (2024), Yu et al. (2023) e Lin et al. 2022, as métricas de nossa segmentação e detecção são competitivas, embora ligeiramente inferiores em alguns aspectos. Essas diferenças podem estar relacionadas à complexidade do conjunto de dados, como ilustrado no Vídeo S1.

Outro estudo comparou os métodos iFOVEA (<https://fovea.eu>) e Timbeter (<https://timbeter.com>), que se baseiam em técnicas foto-ópticas e envolvem a análise de imagens de pilhas de madeira capturadas, utilizando algoritmos. O método iFOVEA apresentou erro médio de 1,34 % para pilhas de madeira de folhosas e -12,82 % para coníferas, enquanto o método Timbeter apresentou erros de -1,83 % e 2,41 %, respectivamente (MOSKALIK et al., 2022). Esses métodos requerem pilhas devidamente preparadas e produzem resultados comparáveis às medições manuais. Outro estudo comparou um método manual, baseado no acordo-quadro alemão para comércio de madeira (RVR), com um aplicativo foto-óptico (iFOVEA) para medição do volume de pilhas. Para pilhas com volume inferior a 20 m³, a variância entre as medições manual e pelo aplicativo foi maior ($\pm 4,02$ %). Para pilhas maiores que 100 m³, a variância foi menor ($\pm 1,82$ %). No geral, o aplicativo apresentou bom desempenho, sugerindo ser uma alternativa confiável, especialmente para medições precisas de grandes pilhas de madeira de boa qualidade. Resultados semelhantes foram encontrados para desvios médios de -2,09 % (RVR esquerda) e -3,66 % (RVR direita) quando comparados à estimativa manual, com melhor precisão para pilhas maiores (BERENDT; WOLFGRAMM; CREMER, 2021).

Além disso, um estudo comparando diferentes métodos de medição de volume de pilhas de madeira encontrou variações significativas entre eles, com desvios relativos de até 9 %. Méto-

dos manuais e ópticos 2D tiveram desvios comparáveis a estudos anteriores, enquanto métodos 3D com LiDAR móvel e RGB apresentaram desvios mais elevados (PURFÜRST et al., 2023), indicando a necessidade de calibração e validação precisas antes da adoção em larga escala.

Outra metodologia para levantamento automático de pilhas de madeira com dispositivos móveis combina técnicas de estereoscopia multivista e detecção 3D existente para obter as localizações 3D das extremidades das toras e calcular o volume de contorno e o volume sólido. Essa abordagem trata a pilha de madeira como um metaobjeto, que inclui toras, espaços entre elas e parâmetros do ambiente. Avaliada no banco de dados público HAWKwood, mostrou-se adequada para aplicações reais, com diferença média de 2,2 % no volume de contorno e -5,6 % no volume sólido. As possíveis aplicações incluem medição eficiente e precisa usando apenas um smartphone ou câmera digital (HERBON et al., 2015).

A classificação de imagens com redes neurais artificiais (RNA) também desempenha papel relevante na quantificação volumétrica da madeira empilhada. Pesquisadores usaram esse método para derivar um fator de conversão que auxilia na estimativa do volume (CAMPOS; LEITE, 2017; SILVEIRA, 2014).

Nosso estudo apresenta um procedimento inédito que combina detecção automática de toras com modelos de distribuição diamétrica para estimar o volume sólido de madeira de eucalipto empilhada. O processo reduz significativamente o tempo e o esforço em comparação aos métodos manuais, minimizando erros humanos. O volume sólido estimado apresentou erro relativo médio (RE %) de -4,535 % em 30 simulações. A comparação entre o volume estimado e o observado pelo método de Smalian apresentou erro relativo de apenas -0,5898 % para a amostra de teste (200 toras), evidenciando alta precisão da metodologia. Esses resultados validam a robustez e confiabilidade do processo, que pode ser aplicado a pilhas grandes e complexas, tornando-o uma ferramenta valiosa para inventários florestais, planejamento de colheita e processamento de madeira.

Embora o estudo se concentre na madeira de eucalipto para produção de celulose, pesquisas futuras devem aprimorar a precisão do modelo e ampliar sua aplicação para outras espécies e condições florestais. A incorporação de variáveis adicionais, como conicidade da tora e densidade da madeira, pode melhorar a precisão das estimativas. Além disso, a integração de tecnologias de sensoriamento avançadas, como LiDAR, com modelos de visão computacional, pode fornecer avaliações ainda mais detalhadas e precisas.

A busca por novos métodos para calcular o volume de madeira empilhada é motivada pela necessidade de aumentar a eficiência, precisão e confiabilidade das medições. A análise de imagens oferece uma alternativa promissora, com potencial para automatizar o processo e reduzir significativamente o tempo e o esforço necessários. É importante reconhecer que a precisão desses métodos depende fortemente da eficácia dos algoritmos de análise de imagens. A pesquisa e o desenvolvimento contínuos de técnicas robustas são cruciais para garantir a viabilidade a longo prazo dessas abordagens, de forma que atendam aos requisitos de precisão para comercialização da madeira, especialmente em indústrias de pequeno e médio porte (PÁSZ-

TORY; HEINZMANN; BARBU, 2018). Esses métodos oferecem uma alternativa para estimar o volume, mas sua precisão pode ser afetada por fatores como espécie da árvore, tamanho da tora, método de empilhamento, teor de umidade, temperatura e até dimensões da pilha, fatores que ainda precisam ser investigados.

4.5. CONCLUSÕES

O uso de modelos de distribuição mostrou-se eficaz na análise dos diâmetros do fuste para estimar o volume sólido total. Essa abordagem estatística forneceu um procedimento alternativo para estimativa de volume, acomodando variações no tamanho das toras e nas configurações das pilhas.

O procedimento proposto é escalável e pode ser aplicado a pilhas grandes e complexas, tornando-se uma ferramenta valiosa para a indústria florestal. Ao automatizar o processo de estimativa de volume, o método favorece práticas mais eficientes de inventário florestal e promove a equidade comercial, garantindo a quantificação precisa da madeira.

Embora o presente estudo demonstre avanços significativos na estimativa do volume de madeira, pesquisas futuras devem concentrar-se no refinamento da precisão do modelo e na ampliação de sua aplicabilidade a diferentes tipos de produção florestal e condições das toras. Isso inclui a adaptação da metodologia para diferentes espécies arbóreas, arranjos de pilha e condições ambientais variadas, visando aumentar ainda mais sua utilidade e robustez.

4.6. REFERÊNCIAS

- BERENDT, Ferréol; WOLFGRAMM, Felix; CREMER, Tobias. Reliability of photo-optical measurements of log stack gross volume. **Silva Fennica**, v. 55, n. 3, 2021. <https://doi.org/10.14214/sf.10555>.
- CAMPOS, João Carlos Chagas; LEITE, Helio Garcia. **Forest measurement: Questions and answers**. Viçosa: UFV, 2017.
- CASAS, Gianmarco Goycochea; ISMAIL, Zool Hilmi; LIMEIRA, Mathaus Messias Coimbra; DA SILVA, Antonilmar Araújo Lopes; LEITE, Helio Garcia. Automatic Detection and Counting of Stacked Eucalypt Timber Using the YOLOv8 Model. **Forests**, v. 14, n. 12, p. 2369, 2023. <https://doi.org/10.3390/f14122369>.
- COLES, Stuart. **An Introduction to Statistical Modeling of Extreme Values**. London: Springer, 2001. 208 p. <https://doi.org/10.1007/978-1-4471-3675-0>.
- CREMER, Tobias; BERENDT, Ferréol; DIEZ, Felipe de Miguel; WOLFGRAMM, Felix; BLASKO, Lubomir. Accuracy of Photo-Optical Measurement of Wood Piles. In: **The 1st International Electronic Conference on Forests-For a Better Future: Sustainability, Innovation, Interdisciplinarity**. Basel Switzerland: MDPI, 2020. p. 90. <https://doi.org/10.3390/IECF2020-08192>.

DE BARROS, Vinicius Andrade; BOECHAT SOARES, Carlos Pedro; DA SILVA, Gilson Fernandes; GOYCOCHEA CASAS, Gianmarco; GARCIA LEITE, Helio. Conversion Factor Estimation of Stacked Eucalypt Timber Using Supervised Image Classification with Artificial Neural Networks. **Pertanika Journal of Science and Technology**, v. 32, n. 4, p. 1527-1543, 2024. <https://doi.org/10.47836/pjst.32.4.05>.

DE MIGUEL-DÍEZ, Felipe; PETTENKOFER, Tim; WALLOR, Evelyn; TOLOSANA-ESTEBAN, Eduardo; PURFÜRST, Thomas; CREMER, Tobias. Influence of log and stack parameters on the conversion factors for Norway Spruce wood stacks using a 3D simulation model. **International Journal of Forest Engineering**, v. 33, n. 2, p. 119-128, 2022. <https://doi.org/10.1080/14942119.2022.2038023>.

DURIGAN, Aline; SILVA OLIVEIRA, Rudson; MISCHKA SGARBIERO, Ivan; AGOSTINHO DA SILVA, Dimas. Gestão operacional do pátio de madeira na produção de cavacos para indústria de polpa e papel. **Advances in Forestry Science**, v. 9, n. 3, p. 1799-1809, 2022. <https://doi.org/10.34062/afs.v9i3.13409>.

GARAY L. **Tropical forest utilization system**. VIII A taper model for the entire stem profile, including buttressing. Seattle: 1979.

GOMIDE, Cíntia Ribeiro. **Determinação do número de toras comerciais destinadas à produção de celulose aplicando um modelo de distribuição diamétrica e equação de taper**. 2016. Universidade Federal de Viçosa.

GRONDIN, Vincent; FORTIN, Jean-Michel; POMERLEAU, François; GIGUÈRE, Philippe. Tree detection and diameter estimation based on deep learning. **Forestry: An International Journal of Forest Research**, v. 96, n. 2, p. 264-276, 2023. <https://doi.org/10.1093/forestry/cpac043>.

HERBON, Christopher; TONNIES, Klaus-Dietz; OTTE, Benjamin; STOCK, Bernd. Mobile 3D wood pile surveying. In: **2015 14th IAPR International Conference on Machine Vision Applications (MVA)**. IEEE, 2015. p. 422-425. <https://doi.org/10.1109/MVA.2015.7153101>.

HUSCH, B.; MILLER, C. I.; BEERS, T. W. **Forest Mensuration**. Krieger Publishing Company., 1993.

JENKINSON, Arthur F. The frequency distribution of the annual maximum (or minimum) values of meteorological elements. **Quarterly Journal of the Royal Meteorological Society**, v. 81, n. 348, p. 158-171, 1955. <https://doi.org/10.1002/qj.49708134804>.

LIN, Yaohai; CAI, Ruixing; LIN, Peijie; CHENG, Shuying. A detection approach for bundled log ends using K-median clustering and improved YOLOv4-Tiny network. **Computers and Electronics in Agriculture**, v. 194, 2022. <https://doi.org/10.1016/j.compag.2022.106700>.

LU, Zhenglan; YAO, Huilu; LYU, Yubiao; HE, Sheng; NING, Heng; YU, Yuhui; ZHAI, Lixia; ZHOU, Lin. A Deep Learning Method for Log Diameter Measurement Using Wood Images Based on Yolov3 and DeepLabv3+. **Forests**, v. 15, n. 5, p. 755, 2024. <https://doi.org/10.3390/f15050755>.

LÖWE, Radim; SEDMÍKOVÁ, Monika; NATOV, Pavel; JANKOVSKÝ, Martin; HEJCMANOVÁ, Pavla; DVOŘÁK, Jiří. Differences in timber volume estimates using various algorithms available in the control and information systems of harvesters. **Forests**, v. 10, n. 5, 2019. <https://doi.org/10.3390/f10050388>.

MATEUS, Ayana; TOMÉ, Margarida. Modelling the diameter distribution of Eucalyptus plan-

tations with Johnson's S B probability density function: Parameters recovery from a compatible system of equations to predict stand variables. **Annals of Forest Science**, v. 68, n. 2, p. 325-335, 2011. <https://doi.org/10.1007/s13595-011-0037-7>.

MOSKALIK, Tadeusz; TYMENDORF, Łukasz; VAN DER SAAR, Jan; TRZCIŃSKI, Grzegorz. Methods of Wood Volume Determining and Its Implications for Forest Transport. **Sensors**, v. 22, n. 16, p. 6028, 2022. <https://doi.org/10.3390/s22166028>.

NYLINDER, M.; KUBÉNKA, T.; HULTNÄS, M. **Roundwood measurement of truck loads by laser scanning**. Field study at Arauco pulp mill Nueva Aldea, p. 1-9, 2008.

PALAHÍ, Marc; PUKKALA, Timo; TRASOBARES, Antoni. Modelling the diameter distribution of *Pinus sylvestris*, *Pinus nigra* and *Pinus halepensis* forest stands in Catalonia using the truncated Weibull function. **Forestry**, v. 79, n. 5, p. 553-562, 2006. <https://doi.org/10.1093/forestry/cpl037>.

PANAGIOTIDIS, Dimitrios; ABDOLLAHNEJAD, Azadeh. Reliable Estimates of Merchantable Timber Volume from Terrestrial Laser Scanning. **Remote Sensing**, v. 13, n. 18, p. 3610, 2021. <https://doi.org/10.3390/rs13183610>.

PURFÜRST, T; DE MIGUEL-DÍEZ, F; BERENDT, F; ENGLER, B; CREMER, T. Comparison of wood stack volume determination between manual, photo-optical, iPad-LiDAR and handheld-LiDAR based measurement methods. **iForest - Biogeosciences and Forestry**, v. 16, n. 4, p. 243-252, 2023. <https://doi.org/10.3832/ifor4153-016>.

PÁSZTORY, Z.; HEINZMANN, B.; BARBU, M. C. Manual and Automatic Volume Measuring Methods for Industrial Timber. In: **IOP Conference Series: Earth and Environmental Science**. Institute of Physics Publishing, 2018. <https://doi.org/10.1088/1755-1315/159/1/012019>.

SA, Qila; JIN, Xingji; PUKKALA, Timo; LI, Fengri. Developing Weibull-based diameter distributions for the major coniferous species in Heilongjiang Province, China. **Journal of Forestry Research**, v. 34, n. 6, p. 1803-1815, 2023. <https://doi.org/10.1007/s11676-023-01610-9>.

SCHMIDT, Luciane Naimeke; SANQUETTA, Mateus Niroh Inoue; MCTAGUE, John Paul; DA SILVA, Gilson Fernandes; FRAGA FILHO, Clayton Vieira; SANQUETTA, Carlos Roberto; SOARES SCOLFORO, José Roberto. On the use of the Weibull distribution in modeling and describing diameter distributions of clonal eucalypt stands. **Canadian Journal of Forest Research**, v. 50, n. 10, p. 1050-1063, 2020. <https://doi.org/10.1139/cjfr-2020-0051>.

SILVA, José Luís; BORDALO, Rui; PISSARRA, José; DE PALACIOS, Paloma. Computer Vision-Based Wood Identification: A Review. **Forests**, v. 13, n. 12, 2022. <https://doi.org/10.3390/f13122041>.

SILVEIRA, D. D. P. **Estimation of the volume wooden stacked using digital images and neural networks**. 2014. Master's Dissertation, Universidade Federal de Viçosa. Disponível em: <http://locus.ufv.br/handle/123456789/3169>.

SOARES, C. P. B.; PAULA NETO, F.; SOUZA, A. L. **Dendrometria e inventário florestal**. 2. ed. Viçosa: Editora UFV, 2011. 272 p.

VICENTE GOUVEIA FILHO, Kleber; SHIRLEN SOARES, Thelma; SANTOS CRUZ, Edmilson; AUGUSTO MIRANDA MATHIAS, Renan. Determinação de fatores de empilhamento e volume de madeira empilhada por meio do processamento de imagens digitais. **Advances in Forestry Science**, v. 9, n. 3, p. 1851-1858, 2022. <https://doi.org/10.34062/afs.v9i3.13470>.

VIRTANEN, Pauli; GOMMERS, Ralf; OLIPHANT, Travis E.; HABERLAND, Matt; REDDY,

Tyler; COURNAPEAU, David; BUROVSKI, Evgeni; PETERSON, Pearu; WECKESSER, Warren; BRIGHT, Jonathan et al. SciPy 1.0: fundamental algorithms for scientific computing in Python. **Nature Methods**, v. 17, n. 3, p. 261-272, 2020. <https://doi.org/10.1038/s41592-019-0686-2>.

WANG, Chien-Yao; YEH, I-Hau; LIAO, Hong-Yuan Mark. **YOLOv9: Learning What You Want to Learn Using Programmable Gradient Information**. arXiv:2402.13616v2, 2024. <https://doi.org/10.48550/arXiv.2402.13616>.

YU, Chunjiang; SUN, Yongke; CAO, Yong; HE, Jie; FU, Yixing; ZHOU, Xiaotao. A Novel Wood Log Measurement Combined Mask R-CNN and Stereo Vision Camera. **Forests**, v. 14, n. 2, p. 285, 2023. <https://doi.org/10.3390/f14020285>.

CAPÍTULO V

NEUROTIMBER: ESTIMATIVA DE VOLUME DE MADEIRA EMPILHADA USANDO DETECÇÃO POR IA E MODELOS DE DISTRIBUIÇÃO DIAMÉTRICA

RESUMO

O NeuroTimber é uma solução integrada para estimativa do volume sólido de madeira empilhada que combina visão computacional e modelagem estatística diamétrica, disponibilizada tanto como aplicação web (Hugging Face Spaces, construída em Gradio) quanto como repositório aberto no GitHub. Na interface web, acessível diretamente pelo navegador, o usuário realiza o upload de um vídeo na aba Detecção em Vídeo, aciona o processamento e obtém a contagem total de toras. Essa etapa utiliza um detector YOLOv9 (com backend de inferência, AutoShape e NMS) acoplado ao ecossistema Supervision e ao rastreador ByteTrack, buscando manter identificadores persistentes por tora ao longo do vídeo e reduzir contagens duplicadas. A aba Diâmetros & Volume recebe um arquivo .xlsx/.xls/.csv com medições de diâmetro (cm), permite definir o intervalo de classes (mín.-máx.) e o comprimento da tora (m), ajusta automaticamente distribuições de probabilidade (via SciPy) e apresenta as Top-5 segundo o critério AIC. A aplicação gera um gráfico (histograma + curvas ajustadas) e, após o usuário escolher a distribuição, calcula o volume estimado combinando (i) contagem total de toras, (ii) probabilidade por classe e (iii) volume cilíndrico por tora, somado sobre as classes.

O repositório GitHub concentra o código-fonte, notebooks e instruções de execução local, incluindo configuração de dependências, pesos do modelo, parâmetros de inferência (confiança/IoU/classes) e scripts para ajuste de distribuições. Esse arranjo garante reprodutibilidade, transparência metodológica e facilidade de customização. A versão Gradio oferece uma experiência “point-and-click”, útil para adoção operacional rápida, enquanto o código aberto viabiliza auditoria acadêmica, reuso e extensão.

Os principais benefícios do NeuroTimber são agilidade, padronização e rastreabilidade do processo de quantificação, reduzindo esforço manual e variabilidade entre avaliadores. Em termos operacionais, a padronização do fluxo reduz erros de transcrição, facilita treinamento de equipes, abre caminho para escalabilidade e possibilita análises de sensibilidade quanto à escolha da distribuição e ao intervalo de classes diamétricas. Em síntese, o NeuroTimber integra, em um único fluxo acessível via web, a contagem automatizada de toras e a estimativa estatística de volume, entregando eficiência, consistência, auditabilidade e capacidade de extensão, qualidades-chave para decisões ágeis e robustas no manejo e na logística florestal.

Palavras-chave: quantificação volumétrica; pilhas de toras; visão computacional aplicada; YOLOv9; Gradio; Hugging Face Spaces; GitHub; inventário florestal

5.1. INTRODUÇÃO

O NeuroTimber é um sistema inovador concebido para estimar de forma precisa e eficiente o volume sólido de toras de madeira empilhadas, unindo avanços recentes em visão computacional, modelagem estatística e interfaces interativas baseadas na Web. A aplicação, desenvolvida pelo autor, possui duas vertentes complementares: um repositório no GitHub, que concentra todo o código-fonte, modelos de detecção e documentação técnica; e uma versão funcional hospedada na plataforma Hugging Face Spaces, construída com a biblioteca Gradio, permitindo o uso direto pelo navegador, sem necessidade de instalação local.

A concepção do NeuroTimber surgiu da necessidade de superar as limitações dos métodos tradicionais de quantificação de madeira, que geralmente se baseiam em medições manuais, demoram a ser executados e estão sujeitos a erros humanos significativos. Nesse contexto, o sistema propõe uma abordagem automatizada que combina a detecção individual de toras, por meio de modelos de deep learning, com a aplicação de modelos de distribuição diamétrica para calcular o volume total de uma pilha. Esta integração entre visão computacional e modelagem estatística proporciona não apenas maior rapidez no processamento, mas também maior consistência e reprodutibilidade dos resultados.

No núcleo do NeuroTimber encontra-se um modelo YOLOv9 treinado especificamente para a detecção e contagem de toras em vídeos de pilhas de madeira. A utilização desta arquitetura de rede neural, reconhecida pela sua velocidade e precisão, permite identificar cada tora individualmente, mesmo em cenários complexos com sobreposição de objetos, variações de iluminação ou heterogeneidade no tamanho das peças. Após a detecção, o sistema relaciona o número de toras encontradas com dados de diâmetro de fuste obtidos previamente em medições de campo, os quais são organizados em classes de diâmetro. Sobre esses dados, são ajustadas distribuições de probabilidade usando métodos estatísticos implementados na biblioteca SciPy, permitindo estimar a frequência relativa de cada classe diamétrica na pilha observada.

O cálculo do volume total segue princípios geométricos adaptados à indústria madeireira, considerando o diâmetro médio de cada classe, a probabilidade estimada de ocorrência e o comprimento padrão das toras fornecido pelo usuário. Essa abordagem resulta em uma estimativa confiável do volume sólido, compatível com os padrões de inventário florestal e aplicável a diferentes espécies e condições de empilhamento.

Do ponto de vista da implementação, o NeuroTimber é projetado para ser flexível e acessível. O repositório no GitHub contém scripts e notebooks que permitem ao usuário executar todo o fluxo de análise em ambiente local ou na nuvem, modificar parâmetros de detecção, treinar novos modelos e personalizar as distribuições estatísticas utilizadas. Já a versão em Hugging Face Spaces, desenvolvida em Gradio, disponibiliza uma interface gráfica simples e intuitiva, na qual o usuário pode enviar vídeos, fornecer informações básicas como comprimento das toras e obter como resultado a visualização das detecções, gráficos de distribuição de diâmetros e o volume total estimado.

Essa dupla presença, código aberto e aplicação interativa, confere ao NeuroTimber um caráter versátil, atendendo tanto pesquisadores e desenvolvedores que desejam aprofundar-se no funcionamento interno do sistema, quanto profissionais da indústria florestal que necessitam de uma solução prática e imediata para estimativa volumétrica. Ao consolidar práticas modernas de inteligência artificial com métodos tradicionais de mensuração florestal, o NeuroTimber representa um avanço significativo em direção à automação e à digitalização de processos na cadeia produtiva da madeira, alinhando-se às tendências globais de floresta 4.0 e silvicultura de precisão.

5.2. VISÃO GERAL DO PIPELINE DE IMPLEMENTAÇÃO

Acesso via navegador:

Abra: <https://github.com/Casas846/NeuroTimber>

O NeuroTimber estima o volume sólido de toras empilhadas combinando:

Detecção, rastreamento e contagem de toras em vídeo (YOLOv9 + ByteTrack via Supervision) → produz o número total de toras.

Modelagem de distribuição diamétrica (SciPy) a partir de dados reais de diâmetro em Excel → produz uma probabilidade por classe de diâmetro.

Estimativa de volume ao multiplicar, por classe, o volume geométrico unitário pelo número esperado de toras da classe, que já incorpora a probabilidade, somando as classes.

5.2.1. Preparação do ambiente

5.2.1.1. Instalação e versões

Instala pacotes fundamentais: numpy, opencv-python-headless, torch, supervision, IPython, pandas, scipy, matplotlib.

Desinstala qualquer PyTorch pré-existente e reinstala PyTorch 2.5.1 com CUDA 12.1 (+ torchvision, torchaudio) pelos wheels oficiais.

Motivo: garantir compatibilidade e desempenho com GPU.

5.2.1.2. Repositório YOLOv9

Clona SkalskiP/yolov9 e instala os requisitos.

Motivo: usar a implementação do YOLOv9, utilitários de inferência (DetectMultiBackend, AutoShape) e helpers (NMS, device selection, etc.).

5.2.1.3. Imports e supressão de avisos

Importa bibliotecas científicas, viz. numpy, pandas, scipy.stats, e ferramentas de visualização (matplotlib).

Desativa warnings para manter o notebook limpo.

5.2.2. Configuração de caminhos e artefatos

O usuário define os caminhos de entrada/saída:

weights: pesos do detector YOLOv9 treinado para toras.

data: arquivo .yaml com nomes de classes (mapeamento id→rótulo).

SOURCE_VIDEO_PATH: vídeo de entrada (pilha de madeira).

TARGET_VIDEO_PATH: vídeo de saída anotado.

data_path: planilha Excel com diâmetros observados no campo.

5.2.3. Detecção com YOLOv9 e integração com Supervision

5.2.3.1. Classe Extended Detections

```
class ExtendedDetections(BaseDetections):
    @classmethod
    def from_yolov9(cls, yolov9_results) -> 'ExtendedDetections':
        ...
```

Converte a saída crua do YOLOv9 (results.pred) em um objeto Detections do Supervision, contendo:

xy-xy: caixas [x1,y1,x2,y2],

confidence: confiabilidade por detecção,

class_id e CLASS_NAME_DATA_FIELD (nomes de classe).

Caso não haja detecções, retorna cls.empty().

Benefício: padroniza a interface de detecções para anotar quadros e alimentar o rastreador.

5.2.3.2. Carregamento e post-processing

```
device = select_device('cuda:0')
model = DetectMultiBackend(weights=weights, device=device, data=data, fuse=True)
model = AutoShape(model)
```

- Seleciona GPU (cuda:0) se disponível; caso contrário, deve-se usar 'cpu'.
- DetectMultiBackend: carrega pesos, configura backend e faz fusion de camadas para ganho de velocidade.
- AutoShape: encapsula pré/pós-processamento (redimensionar, NMS) para invocação simples.

5.2.3.3. Parâmetros de inferência/NMS

```
def prepare_yolov9(model, conf=0.2, iou=0.7, classes=None, agnostic_nms=False, max_det=1000):
    model.conf = conf      # limiar de confiança
    model.iou = iou        # limiar de NMS
    model.classes = classes # filtra classes (ex.: [0] = "log")
    model.agnostic = agnostic_nms
    model.max_det = max_det
    return model
```

- conf controla a sensibilidade (menor $\Rightarrow$ mais detecções, porém mais falsos positivos).
- iou regula a supressão de sobreposição.
- classes permite filtrar a(s) classe(s) de interesse (no seu uso final, [0]).

5.2.3.4. Anotação simples de vídeo

prepare_model_and_video_info lê metadados (FPS, dimensões).

setup_annotator cria um anotador de bounding boxes.

simple_annotate_frame faz:

conversão BGR $\rightarrow$ RGB,

inferência (size=640),

conversão para ExtendedDetections,

desenho das caixas.

simple_process_video aplica o callback por quadro e salva TARGET_VIDEO_PATH.

5.2.4. Rastreamento (ByteTrack) e contagem global de toras

5.2.4.1. Contador global

```
class GlobalCounter:
    def __init__(self):
        self.total_count = 0
        self.tracker_ids = set()
    def update(self, detections):
        for tracker_id in detections.tracker_id:
            if tracker_id not in self.tracker_ids:
                self.tracker_ids.add(tracker_id)
                self.total_count += 1
```

Mantém um conjunto de IDs únicos atribuídos pelo rastreador e acumula a contagem. Cada ID único é tomado como uma tora ao longo do vídeo, na medida em que o rastreador consiga mantê-lo estável.

5.2.4.2. ByteTrack e anotadores

```
byte_tracker = sv.ByteTrack(
    track_activation_threshold=0.25,
    lost_track_buffer=250,           # tolera longas perdas temporárias
    minimum_matching_threshold=0.95,
    frame_rate=video_info.fps
)
```

ByteTrack associa detecções quadro a quadro, dando ID persistente.

Parâmetros altos de buffer e matching ajudam quando a câmera é fixa e objetos são quase estáticos..

Anotadores:

RoundBoxAnnotator (caixas coloridas por ID),

LabelAnnotator (texto com #id, classe e confiança).

5.2.4.3. Loop de anotação com contagem

annotate_frame:

atualiza o rastreador,

desenha caixas,

(opcional) desenha rótulos,

chama global_counter.update(...),

desenha “Total Count: X” no canto superior direito.

process_video:

roda inferência (aqui, size=608), atualiza o ByteTrack e escreve o vídeo final,

ao terminar, imprime Total logs counted: {global_counter.total_count} e retorna esse valor.

Observação prática: ByteTrack é ótimo para objetos em movimento; para pilhas (estáticas), funciona porque cada tora é re-detectada e o tracker mantém o ID. Os parâmetros de buffer altos ajudam a não “esquecer” IDs. O procedimento busca manter um identificador persistente por tora e reduzir a contagem duplicada, cuja magnitude depende do rastreador e de seus parâmetros, conforme demonstrado no Capítulo 3.

5.2.5. Modelagem de distribuição diamétrica (SciPy)

5.2.5.1. Leitura dos dados e seleção da coluna

```
data = pd.read_excel(data_path)
data_column = data.columns[0]
data_values = data[data_column].dropna().values
```

- Lê a primeira coluna de um Excel com diâmetros observados no campo (em cm, tipicamente).
- Remove NaN e guarda em `data_values`.

5.2.5.2. Lista de distribuições candidatas

```
distributions = [
    'genextreme', 'powernorm', 'pearson3', 'genhyperbolic', 'johnsonsu', 'truncnorm',
    'skewnorm', 'nct', 'gennorm', 'exponnorm', 'norm', 'dweibull', 'dgamma', 't'
]
```

Conjunto eclético com distribuições assimétricas e pesadas na cauda (útil para diâmetros reais, que raramente são perfeitamente normais).

5.2.5.3. Ajuste e seleção via AIC

```
def best_fit_distributions(data, distributions, top_n=5):
    for distribution in distributions:
        dist = getattr(stats, distribution)
        params = dist.fit(data)          # estima parâmetros (inclui loc, scale)
        log_likelihood = np.sum(dist.logpdf(data, *params[:-2], loc=params[-2], scale=params[-1]))
        k = len(params)                  # nº de parâmetros
        aic = 2 * k - 2 * log_likelihood # AIC = 2k - 2 ln(L)
        ...
```

- Para cada distribuição:

ajusta parâmetros por máxima verossimilhança,

calcula log-verossimilhança,

computa o AIC (quanto menor, melhor equilíbrio ajuste/complexidade).

- Ordena por AIC e retorna as Top-5.

5.2.5.4. Probabilidades por classe de diâmetro

```
x_values = np.arange(1, 36) # classes 1..35 (cm)
results = pd.DataFrame(x_values, columns=['Value'])

for i, (dist, params, aic) in enumerate(top_distributions):
    p = dist.pdf(x_values, *params[:-2], loc=params[-2], scale=params[-1])
    results[f'Rank {i+1}': {dist.name}] = p
```

- Para cada distribuição Top-5, avalia a densidade nos diâmetros inteiros de 1 a 35 cm e grava no `results`.

- Também plota um histograma dos dados e as curvas das distribuições mais bem ranqueadas, salvando:

top_5_best_fit_distributions.png (gráfico),
probabilities_top_5.csv (tabela de densidades).

5.2.6. Estimativa do volume empilhado

5.2.6.1. Fórmula geométrica e unidades

Para cada classe de diâmetro d (em cm) e comprimento de tora L (em m), o volume individual idealizado como cilindro é:

$$V_{tora} = \frac{\pi}{4} \left(\frac{d}{100} \right)^2 L = \frac{\pi}{40000} d^2 L \quad (5.1)$$

No código:

```
volume_observed = (np.pi / 40000) * (class_number ** 2) * log_length
```

- class_number corresponde a d em cm,
- log_length corresponde a L em m.

5.2.6.2. Combinação com probabilidades e contagem

A ideia é obter o número esperado de toras em cada classe:

$$E[\text{contagem na classe } d] \approx p(d) \times \text{total_logs} \quad (5.2)$$

e então:

$$V_{classe} = E[\text{contagem na classe } d] \times V_{tora} \quad (5.3)$$

No código:

```
log_length = 6
total_volume_estimated = 0
for _, row in results.iterrows():
    class_number = row['value']
    probability = row['Rank 1: genextreme'] # escolher a coluna da distribuição vencedora
    log_count = total_logs # contagem total + proxy para contagem esperada por classe
    volume_observed = (np.pi / 40000) * (class_number ** 2) * log_length
    volume_estimated = probability * log_count * volume_observed
    total_volume_estimated += volume_estimated
```

- Probabilidade por classe $\times$ contagem total $\Rightarrow$ contagem esperada por classe.

- Volume por tora $\times$ contagem esperada $\Rightarrow$ volume por classe.
- Soma sobre todas as classes $\Rightarrow$ volume sólido total estimado.

5.3. TUTORIAL DE USO DO NEUROTIMBER (VERSÃO WEB - HUGGING FACE SPACES)

Acesso via navegador (sem instalação):

Abra: <https://huggingface.co/spaces/Casas846/NeuroTimber>. Verifique no canto superior que o status esteja Running. Caso apareça Building ou Queued, aguarde a inicialização do espaço.

5.3.1. Visão geral da interface

O NeuroTimber possui duas abas principais:

Detecção em Vídeo - Faz o upload de um vídeo da pilha de madeira, executa a detecção automática (YOLOv9) e retorna a contagem total de toras.

Diâmetros & Volume - Recebe dados reais de diâmetro (arquivo .xlsx/.xls/.csv), ajusta distribuições estatísticas e estima o volume sólido total considerando o comprimento informado da tora.

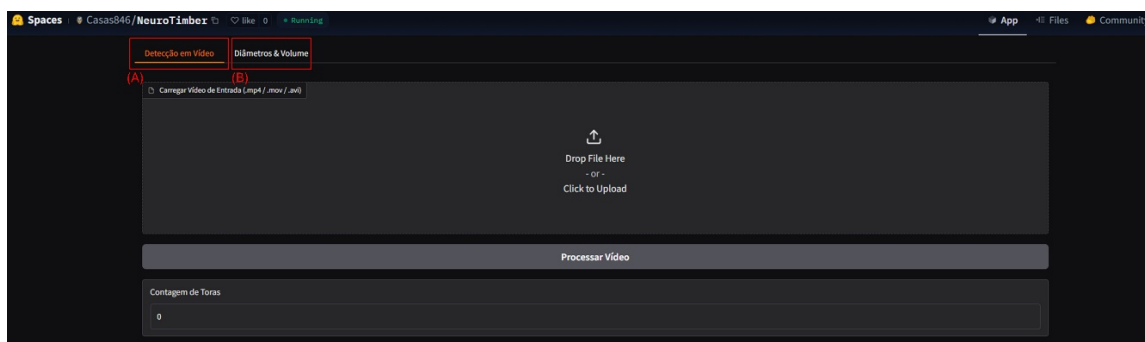


Figura 5.1. Aba Detecção em Vídeo do NeuroTimber (versão web). A interface contém: (A) área de upload do vídeo e (B) Diâmetros & Volume

5.3.2. Detecção de toras em vídeo (aba “Detecção em Vídeo”)

5.3.2.1. Carregar o vídeo

Clique em Click to Upload (ou arraste e solte o arquivo) na caixa central.

Formatos aceitos: .mp4, .mov, .avi.

Boas práticas para melhor detecção:

Prefira vídeos com iluminação uniforme e pouco brilho/contraste extremo.

Enquadre toda a pilha sempre que possível; evite cortes laterais.

Evite movimentos bruscos da câmera; filmagens estáveis geram detecções mais consistentes.

5.3.2.2. Processar o vídeo

Clique em Processar Vídeo.

O sistema executa a detecção com YOLOv9 e realiza rastreamento para evitar contagens duplicadas ao longo dos quadros.

5.3.2.3. Ler o resultado

Após o processamento, verifique o campo Contagem de Toras.

Esse valor representa a contagem total de toras detectadas no vídeo analisado.

Observação: se você pretende usar essa contagem para estimar volume, siga para a aba Diâmetros & Volume e utilize a contagem como entrada conceitual (o app utiliza internamente a contagem total e a reparte por classes via distribuição diamétrica ajustada).

5.3.3. Estimativa de volume (aba “Diâmetros & Volume”)

5.3.3.1. Carregar os dados de diâmetro

Clique em Click to Upload e selecione sua planilha .xlsx/.xls (ou .csv) contendo diâmetros medidos em campo (em cm).

Recomenda-se que a primeira coluna contenha as medidas; valores ausentes serão ignorados.

5.3.3.2. Definir intervalos e comprimento

Diâmetro Mínimo (cm) e Diâmetro Máximo (cm): definem o range de classes considerado (ex.: 1 a 36 cm).

Comprimento da Tora (m): informe o L utilizado na sua realidade operacional (ex.: 6 m).

5.3.3.3. Ajustar distribuições

Clique em Ajustar Distribuições.

A aplicação calcula as Top-5 distribuições e exibe:

Uma tabela com Rank, Distribuição, AIC e Parâmetros;

O gráfico combinando o histograma empírico com as curvas ajustadas.

5.3.3.4. Escolher a distribuição e calcular volume

No seletor Escolher Distribuição, selecione a distribuição preferida (por exemplo, a de Rank 1).

Clique em Calcular Volume.

O app computa o Volume Estimado (m^3), combinando:

Contagem total de toras (obtida na aba de detecção),

Probabilidade por classe de diâmetro (vinda do ajuste escolhido),

Volume por tora idealizado como cilindro,
O resultado aparece no campo Volume Estimado.

The screenshot displays the 'Diâmetros & Volume' tab of the NeuroTimber application. It features a dark-themed interface with the following elements:

- (A)** A file upload area with the text 'Carregar Dados de Diâmetro (.xlsx/.xls/.csv)' and a 'Drop File Here' instruction.
- (B)** Input fields for 'Diâmetro Mínimo (cm)' (value: 1) and 'Diâmetro Máximo (cm)' (value: 36).
- (C)** An input field for 'Comprimento da Tora (m)' (value: 6).
- (D)** A button labeled 'Ajustar Distribuições'.
- (E)** A plot area showing a histogram and 'Top-5 Ajustes' with columns for Rank, Distribuição, AIC, and Parâmetros.
- (F)** A button labeled 'Escolher Distribuição'.
- (G)** A button labeled 'Calcular Volume'.
- (H)** An output field for 'Volume Estimado'.

Figura 5.2. Parte superior da aba Diâmetros & Volume. Elementos: (A) upload dos dados de diâmetro (.xlsx/.xls/.csv); (B) campos para Diâmetro Mínimo e Diâmetro Máximo (cm); (C) Comprimento da Tora (m); (D) botão Ajustar Distribuições que calcula os melhores ajustes segundo AIC; (E) Gráfico com o histograma e as Top-5 distribuições; (F) seletor Escolher Distribuição; (G) botão Calcular Volume; e (H) saída Volume Estimado.

5.4. CONCLUSÕES

O NeuroTimber apresenta uma solução integrada e acessível para a quantificação de volume de madeira empilhada, unindo visão computacional e modelagem estatística em um mesmo fluxo de trabalho. A aplicação web no ambiente do Hugging Face Spaces construída em Gradio permite que o usuário interaja de forma direta pelo navegador, desde o envio do vídeo até a obtenção da contagem total de toras. Em paralelo, o repositório aberto no GitHub disponibiliza todo o código necessário para execução reprodutível, auditoria e customização, fortalecendo a transparência metodológica e a continuidade do desenvolvimento.

A principal contribuição do sistema está na padronização do processo de medição, reduzindo a dependência de procedimentos manuais demorados e sujeitos a grande variabilidade entre avaliadores. Ao automatizar a detecção e o rastreamento das toras no vídeo, o NeuroTimber eleva a consistência e a repetibilidade dos resultados, favorecendo decisões mais ágeis e confiáveis em contextos operacionais. A possibilidade de operar inteiramente pela web elimina

barreiras de instalação e facilita o uso em diferentes ambientes, inclusive por equipes com perfis técnicos distintos.

Outro mérito relevante é a integração da contagem automática com a estimativa de volume baseada em distribuições diamétricas ajustadas a dados reais. O usuário fornece medições de diâmetro e o sistema recomenda modelos estatísticos com base em critérios de qualidade de ajuste, oferecendo visualizações que ajudam na interpretação e na escolha final. Esse desenho metodológico cria um rastro verificável, com artefatos como vídeos anotados, tabelas e gráficos, que podem ser incorporados a relatórios técnicos e a estudos acadêmicos com clareza e rastreabilidade.

Em síntese, o NeuroTimber conecta rigor científico e aplicabilidade prática ao oferecer, em um único ecossistema, uma aplicação web pronta para uso e um repositório aberto para pesquisa e extensão. O resultado é um fluxo de trabalho padronizado, auditável e escalável, que promove ganhos reais de eficiência, consistência e governança de dados na mensuração de madeira empilhada, contribuindo para a modernização do manejo e da logística florestal.

CAPÍTULO VI

NEUROTIMBER MOBILE: ESTIMATIVA DE VOLUME SÓLIDO DE MADEIRA EMPILHADA POR SEGMENTAÇÃO DE IMAGENS E CALIBRAÇÃO PELA LARGURA DA PILHA EM APLICATIVO DE MENSAGERIA

RESUMO

A quantificação do volume sólido de madeira empilhada continua sendo uma etapa crítica e onerosa das operações florestais, tradicionalmente dependente de medições manuais ou de fatores de empilhamento sujeitos a elevada variabilidade. Este capítulo apresenta o *NeuroTimber Mobile*, uma solução leve de estimativa volumétrica que opera diretamente em um aplicativo de mensageria (Telegram, acessível como @NeuroTimber_bot) e que combina três componentes de visão computacional e mensuração florestal: (i) o modelo *Segment Anything Model 3* (SAM 3), empregado para segmentar e recortar automaticamente a pilha, removendo o fundo da fotografia; (ii) o modelo YOLO26-Large de segmentação de instâncias, que delimita a face circular de cada tora individualmente; e (iii) uma calibração métrica direta, na qual a razão entre a largura física da face da pilha, informada pelo usuário, e a largura do recorte em pixels converte a área de madeira em metros quadrados, aplicando-se em seguida a fórmula de Smalian para obter o volume. O sistema foi avaliado em um estudo de caso com 30 pilhas de *Eucalyptus grandis* empilhadas manualmente, cujos volumes sólidos de referência foram determinados por cubagem rigorosa. Com a largura correta da face da pilha (3 m) e a contagem dos pixels de madeira obtidos por segmentação, o método alcançou erro médio percentual absoluto de 4,66%, viés médio de +1,66%, erro máximo de 8,75% e coeficiente de correlação de 0,795 com os volumes de referência. A largura ótima implícita, obtida por retroajuste, foi de 2,98 m, praticamente coincidente com a largura real do quadro experimental (3 m), o que confirma a coerência física do procedimento. O aplicativo dialoga com o usuário para coletar a largura da face, o comprimento da tora e o número de faces disponíveis (uma ou duas); no modo de uma face, assume-se a igualdade entre as duas faces da pilha. Os resultados demonstram que uma calibração simples fornecida pelo operador, aliada a modelos de segmentação de ponta, é capaz de produzir estimativas acuradas de volume sem a necessidade de estruturas laboratoriais, viabilizando o uso em campo por meio de um dispositivo móvel comum.

Palavras-chave: madeira empilhada; volume sólido; segmentação de instâncias; SAM 3; YOLO; fórmula de Smalian; aplicativo móvel; Telegram; inventário florestal

6.1. INTRODUÇÃO

A mensuração de madeira empilhada é uma atividade recorrente e economicamente relevante em toda a cadeia produtiva florestal, sendo empregada na comercialização, no controle de estoques em pátios de fábricas e no planejamento logístico do transporte (DE MIGUEL-DÍEZ; PURFÜRST et al., 2023). Historicamente, o volume de pilhas é expresso em estéreo (volume aparente) e convertido em volume sólido por meio de fatores de empilhamento, cuja aplicação, embora prática, introduz erros consideráveis em razão da variação no comprimento, no diâmetro e na forma das toras, no método de empilhamento e na presença de defeitos (BARROS, 2017; SOARES; PAULA NETO; SOUZA, 2011). Estudos clássicos relatam que o uso de fatores de empilhamento inadequados pode gerar erros de até 65% na estimativa do volume sólido, o que compromete a confiança nas transações comerciais e a acurácia dos inventários (BARROS, 2017).

Diante dessas limitações, a análise de imagens digitais consolidou-se como alternativa de baixo custo para a quantificação de madeira empilhada, reduzindo a participação do operador e aumentando a repetibilidade das medições. A revisão de De Miguel-Díez, Purfürst et al. (2023) evidencia a diversidade de métodos fotoópticos e de escaneamento a laser desenvolvidos para estimar fatores de conversão em pátios e carregamentos, ressaltando, ao mesmo tempo, a necessidade de soluções que sejam simultaneamente acuradas e operacionalmente simples. No Brasil, Barros (2017) demonstrou que a classificação de imagens por redes neurais artificiais permite estimar fatores de empilhamento em pilhas de *Eucalyptus* com diferenças médias inferiores a 1% em relação ao valor observado em campo, resultado que reforça o potencial da abordagem baseada em imagens.

Os avanços recentes em aprendizado profundo ampliaram substancialmente esse potencial. A família de detectores YOLO (*You Only Look Once*), introduzida por Redmon et al. (2016), reformulou a detecção de objetos como um problema de regressão em uma única passada de rede, alcançando desempenho em tempo real com alta acurácia. No domínio florestal, Casas et al. (2023) treinaram um modelo YOLOv8 para detectar e contar toras de eucalipto empilhadas, obtendo contagens automáticas confiáveis mesmo em cenários de elevada sobreposição. Paralelamente, o *Segment Anything Model* (SAM), proposto por Kirillov et al. (2023) e posteriormente aprimorado nas versões SAM 2 (RAVI et al., 2024) e SAM 3 (CARION et al., 2025), esta última empregada neste capítulo, estabeleceu um novo paradigma de segmentação por *prompts*, capaz de isolar objetos arbitrários em imagens e vídeos sem treinamento específico (*zero-shot*). A combinação de segmentação de instâncias, que delimita com precisão a face de cada tora, com uma segmentação de contexto, que isola a pilha do fundo, cria as condições para transformar a contagem de pixels de madeira em uma medida física de área basal.

Ao mesmo tempo, observa-se uma tendência crescente de aproximar essas tecnologias do usuário final por meio de dispositivos móveis. Aplicativos de inventário florestal baseados em *smartphones*, como o ForestScanner (TATSUMI; YAMAGUCHI; FURUYA, 2023), demon-

straram que sensores e câmeras de dispositivos comuns podem substituir instrumentos dedicados na obtenção de variáveis dendrométricas em campo. De modo análogo, assistentes conversacionais têm sido adotados como interface de baixa barreira de entrada para serviços agrícolas: sistemas como o AgronomoBot, construído sobre a API do Telegram (MOSTAÇO et al., 2018), mostram que a mensageria instantânea é um canal eficaz para entregar informação técnica a produtores, sem exigir instalação de programas nem conhecimento especializado.

Neste contexto, o presente capítulo propõe e avalia o *NeuroTimber Mobile*, uma versão do sistema NeuroTimber concebida para operar inteiramente dentro de um aplicativo de mensageria. Diferentemente da abordagem baseada em catálogo de diâmetros descrita nos capítulos anteriores, o *NeuroTimber Mobile* adota uma calibração métrica direta, na qual o próprio operador informa a largura física da face da pilha; essa informação, combinada à segmentação da pilha (SAM 3) e das toras (YOLO26), converte a área de madeira observada em volume sólido pela fórmula de Smalian. O objetivo deste estudo é descrever detalhadamente essa metodologia e avaliar sua acurácia em um conjunto de 30 pilhas de *Eucalyptus grandis* com volumes de referência conhecidos, além de apresentar a implementação do sistema como um agente conversacional acessível pelo Telegram.

6.2. MATERIAIS E MÉTODOS

6.2.1. Conjunto de dados e volumes de referência

Utilizaram-se as fotografias e os volumes de referência de 30 pilhas de madeira de *Eucalyptus grandis* empilhadas manualmente, originalmente obtidos por Barros (2017) no setor de Silvicultura do Departamento de Engenharia Florestal da Universidade Federal de Viçosa. As pilhas foram montadas com rodízio das peças disponíveis, de modo que apresentassem volumes, tamanhos e conjuntos de toras distintos. Um quadro de madeira padronizou o comprimento da face da pilha (C) em 3 m e a altura máxima (H) em 2 m, enquanto a largura da pilha (L), correspondente ao comprimento das toras, foi fixada em 1 m. Cabe destacar a distinção geométrica adotada neste capítulo: a **largura da face** da pilha, dimensão horizontal visível na fotografia e empregada na calibração, corresponde ao comprimento $C = 3$ m do quadro; já o valor de 1 m refere-se à **profundidade** da pilha, isto é, ao comprimento da tora, utilizado como L na fórmula de Smalian. A confusão entre essas duas grandezas é uma fonte potencial de erro grosseiro na calibração, conforme discutido adiante.

As fotografias de ambas as faces de cada pilha foram tomadas com câmera digital a uma distância de 3 m de cada face, com resolução de 1152×864 pixels. O volume sólido de referência (V_{cc}) de cada pilha foi determinado por Barros (2017) pelo somatório dos volumes individuais das toras, calculados pela fórmula de Smalian a partir da cubagem rigorosa, constituindo o padrão-ouro contra o qual as estimativas automáticas foram comparadas.

6.2.2. Segmentação e recorte da pilha com SAM 3

A primeira etapa do processamento consiste em isolar a pilha do restante da cena (solo, vegetação, estruturas de sustentação e céu). Para tanto, empregou-se o *Segment Anything Model 3* (SAM 3) (CARION et al., 2025) no modo de segmentação por *prompt* textual, utilizando a instrução “stacked wood pile”. As máscaras retornadas pelo modelo foram filtradas por um limiar de confiança de 0,20 e combinadas; em seguida, reteve-se apenas o maior componente conexo, correspondente ao corpo principal da pilha, aplicando-se uma operação morfológica de fechamento com elemento estruturante elíptico de 15×15 pixels para suavizar bordas e preencher pequenas discontinuidades. A máscara resultante foi usada para gerar a imagem limpa, na qual todos os pixels externos à pilha foram zerados (fundo preto). A extensão horizontal da máscara, isto é, a diferença entre a maior e a menor coluna ocupada, definiu a **largura do recorte em pixels** (w_{px}), grandeza fundamental para a calibração métrica descrita na Seção 6.2.4.

6.2.3. Segmentação individual das toras com YOLO26

Sobre a imagem limpa, aplicou-se o modelo YOLO26-Large de segmentação de instâncias, treinado para reconhecer a face transversal de toras de eucalipto. Diferentemente de um detector, que apenas delimita cada objeto por uma caixa retangular, o modelo de segmentação produz uma **máscara por pixel** para cada tora, delineando a forma aproximadamente circular da face e excluindo os espaços vazios entre as peças. Adotou-se limiar de confiança de 0,25 e número máximo de detecções de 1500 por imagem, valor suficiente para acomodar pilhas com centenas de toras. A união das máscaras de todas as toras detectadas define o conjunto de pixels efetivamente ocupados por madeira, cuja cardinalidade (N_{mad}) representa a área basal projetada da face, em pixels. A opção por um modelo de segmentação, em vez de um mero detector de caixas, é determinante para a acurácia do método: a área basal a ser convertida em volume corresponde à seção circular da tora, e não ao quadrado que a circunscreve, de modo que apenas a máscara fornece diretamente a medida fisicamente correta.

6.2.4. Calibração métrica pela largura da pilha

A conversão da área de madeira, medida em pixels, para unidades físicas fundamenta-se em uma relação de escala direta. Sendo W a largura física da face da pilha (em metros), informada pelo usuário, e w_{px} a largura do recorte da pilha (em pixels) obtida na etapa de segmentação de contexto, define-se o fator de escala:

$$m_{px} = \frac{W}{w_{px}} \quad [\text{m} \cdot \text{px}^{-1}] \quad (6.1)$$

A área basal de madeira de uma face, em metros quadrados, é obtida multiplicando-se o número de pixels de madeira pelo quadrado do fator de escala:

$$G = N_{mad} \cdot m_{px}^2 = N_{mad} \left(\frac{W}{w_{px}} \right)^2 \quad [\text{m}^2] \quad (6.2)$$

Esse procedimento dispensa qualquer referência a catálogos de diâmetro ou a modelos de distribuição probabilística, apoiando-se exclusivamente em uma medida física fornecida pelo operador em campo.

6.2.5. Cálculo do volume pela fórmula de Smalian

O volume sólido da pilha é estimado tratando-a como uma seção de comprimento L (a profundidade da pilha, igual ao comprimento das toras) delimitada por duas faces de áreas basais G_A e G_B . Aplica-se, então, a fórmula de Smalian (MACHADO; FIGUEIREDO FILHO, 2003):

$$V = \frac{G_A + G_B}{2} L \quad [\text{m}^3] \quad (6.3)$$

O sistema opera em dois modos. No **modo de duas faces**, o usuário fotografa ambas as faces da pilha, e G_A e G_B são calculados independentemente. No **modo de uma face**, quando o operador dispõe de apenas uma fotografia, assume-se a igualdade entre as faces ($G_B = G_A$), de modo que a Equação (6.3) se reduz a $V = G_A L$. Essa flexibilidade adequa-se às condições reais de campo, nas quais nem sempre é possível acessar as duas extremidades de uma pilha.

6.2.6. Implementação móvel: NeuroTimber Mobile

Toda a cadeia de processamento foi encapsulada em um agente conversacional executado sobre a API de *bots* do Telegram, denominado *NeuroTimber Mobile* e acessível pelo identificador @NeuroTimber_bot. A arquitetura integra um modelo de linguagem, responsável pelo diálogo em linguagem natural e pela orquestração das ferramentas, aos módulos de visão computacional descritos anteriormente. O fluxo de interação foi projetado para coletar, de forma conversacional, os três parâmetros indispensáveis à estimativa:

1. a **largura da face da pilha** (em metros), obrigatória, por constituir a referência de calibração;
2. o **comprimento da tora** (em metros), correspondente à profundidade L , com valor padrão de 1 m;
3. o **número de faces** disponíveis (uma ou duas).

À medida que o usuário fornece esses valores, o agente os armazena e confirma; somente após o registro da largura o sistema autoriza o envio das fotografias. Internamente, o agente mantém, para cada usuário, um estado de conversa que preserva os parâmetros informados e os caminhos das últimas fotografias, o que viabiliza tanto a coleta incremental de informações

quanto o reproprocessamento posterior. A orquestração é realizada por meio de ferramentas específicas: uma função registra os parâmetros da pilha (largura, comprimento e número de faces); outra dispara a estimativa de volume sobre a(s) fotografia(s) recebida(s); e uma terceira permite recalcular o volume corrigindo qualquer parâmetro sem a necessidade de reenviar as imagens, reduzindo o esforço do operador em campo.

Recebidas as imagens, o agente executa automaticamente a sequência SAM 3 → recorte → YOLO26 → calibração → Smalian e devolve ao usuário o volume estimado, acompanhado da contagem de toras segmentadas e das imagens anotadas com as máscaras coloridas. No modo de duas faces, o sistema solicita as duas fotografias em sequência (face A e face B) e processa cada uma independentemente; no modo de uma face, o processamento é disparado imediatamente após o recebimento da única fotografia, assumindo-se a igualdade entre as faces. A Figura 6.1 sintetiza, na forma de fluxograma, todas as etapas do processamento executadas pelo aplicativo, desde a coleta dos parâmetros até a devolução do resultado.



Figura 6.1. Fluxograma de funcionamento do NeuroTimber Mobile: coleta dos parâmetros (largura da face, comprimento da tora e número de faces), envio da(s) fotografia(s), segmentação e recorte da pilha por SAM 3, segmentação das toras por YOLO26, calibração métrica, cálculo da área basal, aplicação da fórmula de Smalian e devolução do resultado ao usuário.

6.2.7. Avaliação estatística

A acurácia das estimativas foi avaliada comparando-se o volume estimado (V_{est}) ao volume de referência (V_{cc}) das 30 pilhas. Calcularam-se o erro relativo por pilha, o erro médio percentual absoluto (EMPA), o viés médio, o erro máximo, a raiz do erro quadrático médio (RMSE) e o coeficiente de correlação de Pearson (r). O erro relativo de cada pilha i foi definido como:

$$E_i = \frac{V_{est,i} - V_{cc,i}}{V_{cc,i}} \times 100 \quad [\%] \quad (6.4)$$

Adicionalmente, procedeu-se a um retroajuste da largura, buscando o valor de W que minimiza o erro global, com o intuito de verificar se a largura física ótima recuperada pelo método corresponde à largura real do quadro experimental (3 m), o que constituiria evidência da con-

sistência física do procedimento.

6.3. RESULTADOS

6.3.1. Segmentação da pilha e das toras

O modelo SAM 3 isolou de forma consistente o corpo da pilha em todas as imagens, removendo vegetação, solo, parede e as estruturas de sustentação do quadro. A Figura 6.2 ilustra, para uma pilha representativa, as três etapas do processamento: a imagem original (à esquerda), o recorte obtido por SAM 3 com fundo preto e a largura da pilha demarcada (ao centro), e a segmentação individual das toras pelo YOLO26 (à direita). A largura do recorte, expressa em pixels, é a grandeza que a calibração converte na medida física de 3 m.

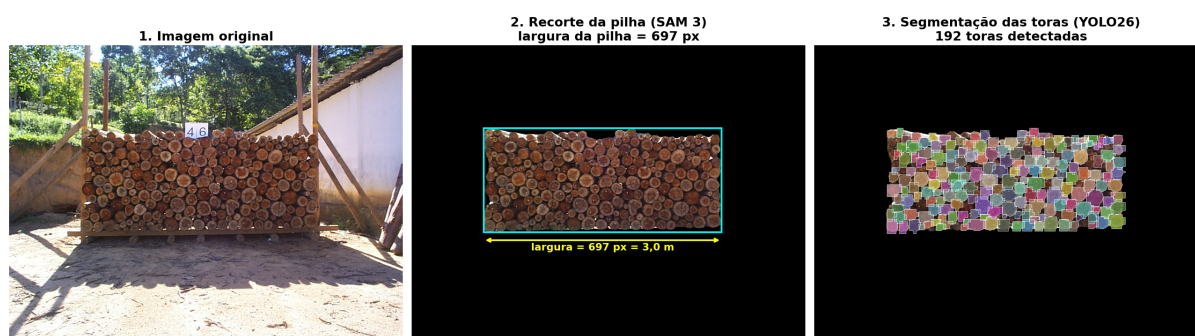


Figura 6.2. Etapas do processamento de uma pilha (P24, face A): (1) imagem original; (2) recorte da pilha por SAM 3, com fundo preto e demarcação da largura empregada na calibração; (3) segmentação individual das toras pelo modelo YOLO26, com 192 faces detectadas.

A Figura 6.3 apresenta uma segunda pilha (P16, face A) submetida ao mesmo procedimento, evidenciando a robustez da segmentação frente a variações de iluminação, sombreamento e heterogeneidade no tamanho das toras.



Figura 6.3. Segundo exemplo de processamento (P16, face A): imagem original, recorte da pilha por SAM 3 e segmentação individual das toras (223 faces detectadas).

6.3.2. Estimativas de volume

Aplicando-se a calibração com a largura correta da face (3 m) e a fórmula de Smalian, obtiveram-se as estimativas de volume apresentadas na Tabela 6.1, com o erro relativo de cada uma das 30 pilhas. As estimativas concentraram-se em torno dos valores de referência, com predominância de pequenos erros positivos e negativos.

Tabela 6.1. Volume de referência (V_{cc}), volume estimado pelo NeuroTimber Mobile, erro relativo e número de toras segmentadas em cada face (A e B) para as 30 pilhas de *Eucalyptus grandis*.

Pilha	V_{cc} (m ³)	V_{est} (m ³)	Erro (%)	n_A	n_B
1	2,4632	2,6737	+8,55	288	262
2	2,7099	2,9116	+7,44	253	269
3	2,6269	2,4803	-5,58	208	231
4	2,7641	2,6181	-5,28	201	248
5	2,7999	2,8653	+2,34	236	256
6	2,8999	2,6540	-8,48	250	286
7	2,8103	2,6652	-5,16	224	259
8	2,8320	2,8066	-0,90	233	268
9	2,9010	2,9220	+0,72	249	281
10	3,1397	3,0692	-2,24	272	300
11	2,9347	2,8256	-3,72	244	257
12	2,8442	2,8756	+1,10	227	261
13	2,8675	2,6166	-8,75	240	277
14	2,8227	2,9391	+4,12	250	268
15	2,8290	2,6917	-4,85	255	295
16	2,9254	3,0120	+2,96	267	288
17	2,9122	2,9859	+2,53	254	282
18	3,2176	3,3791	+5,02	239	255
19	3,0476	3,2108	+5,35	228	255
20	2,9609	3,1484	+6,33	212	226
21	2,9093	3,1225	+7,33	211	230
22	3,0000	3,1513	+5,04	212	230
23	2,9291	3,0508	+4,15	212	230
24	2,9686	3,0886	+4,04	210	228
25	2,9740	3,0325	+1,97	211	238
26	2,9630	3,1099	+4,96	239	237
27	2,9775	3,1335	+5,24	239	242
28	2,8941	2,9976	+3,57	225	223

continua na próxima página

Tabela 6.1 - continuação

Pilha	V_{cc} (m ³)	V_{est} (m ³)	Erro (%)	n_A	n_B
29	2,9404	3,1298	+6,44	246	242
30	3,2176	3,3960	+5,55	265	270

Considerando as 30 pilhas, o método apresentou erro médio percentual absoluto (EMPA) de 4,66%, viés médio de +1,66%, erro máximo de 8,75% e raiz do erro quadrático médio de 0,148 m³. O coeficiente de correlação de Pearson entre estimativas e referências foi de 0,795, indicando que o volume estimado acompanha de forma consistente a variação real entre as pilhas. A Figura 6.4 sintetiza esses resultados: no gráfico de dispersão (a), os pontos distribuem-se próximos à linha de identidade 1:1; no histograma (b), a maioria dos erros concentra-se entre -6% e +6%, com leve tendência à superestimação.

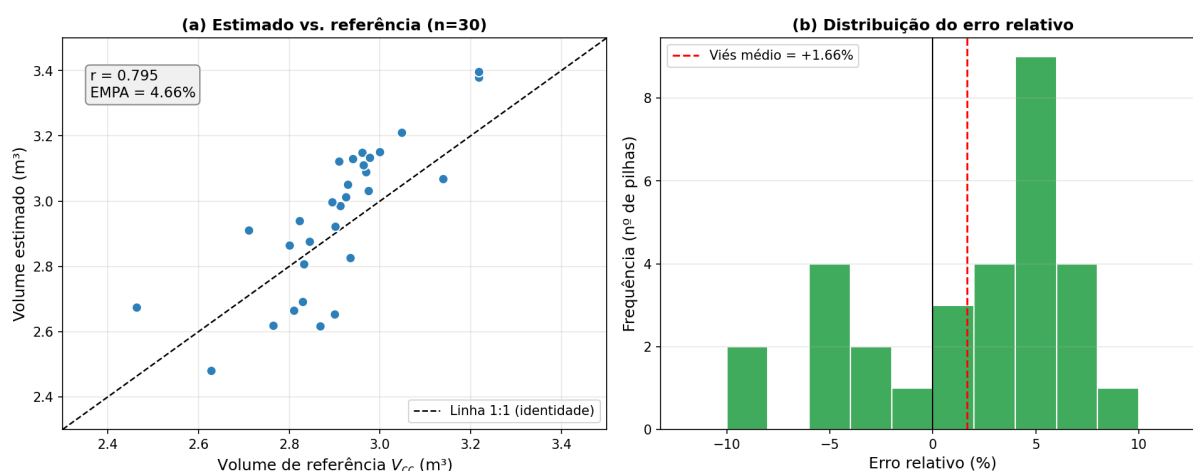


Figura 6.4. Avaliação do NeuroTimber Mobile nas 30 pilhas: (a) volume estimado versus volume de referência, com a linha de identidade 1:1; (b) distribuição do erro relativo, com o viés médio destacado. EMPA = erro médio percentual absoluto.

6.3.3. Consistência física da calibração

O retroajuste da largura, isto é, a busca do valor de W que minimiza o erro global das 30 pilhas, resultou em uma largura ótima implícita de 2,98 m, praticamente coincidente com a largura real do quadro experimental (3 m). Esse resultado é uma evidência forte da consistência física do procedimento: a área de madeira medida por segmentação, quando corretamente escalonada, reproduz a área basal real das pilhas. Em contraposição, quando se adotou erroneamente a profundidade da pilha (1 m) como se fosse a largura da face, o volume foi subestimado em cerca de 89%, uma vez que a estimativa é proporcional ao quadrado da largura informada e, portanto, reduz-se a um nono do valor obtido com 3 m, o que ilustra a criticidade de fornecer a dimensão correta na calibração e reforça a distinção geométrica estabelecida na Seção 6.2.1.

6.3.4. O aplicativo NeuroTimber Mobile em operação

O sistema foi disponibilizado como agente conversacional no Telegram, sob o identificador @NeuroTimber_bot, e testado em uma aplicação de campo real. As Figuras 6.5, 6.6 e 6.7 reproduzem as três etapas de um diálogo típico, correspondentes a um caso concreto de estimativa em modo de uma face.

Na primeira etapa (Figura 6.5), o aplicativo se apresenta e solicita, em linguagem natural, os três parâmetros necessários: a largura da face da pilha, o comprimento da tora e o número de faces disponíveis. O usuário responde de forma compacta (“1. 3 m, 2. 1 m, 3. 1 foto”), e o agente interpreta a mensagem, registra os parâmetros (largura de 3 m, tora de 1 m, modo de uma face) e confirma que está pronto para receber a fotografia, orientando quanto ao enquadramento e à iluminação adequados.

Na segunda etapa (Figura 6.6), o usuário envia a fotografia da face da pilha. O agente informa que a imagem foi recebida em modo de uma face e inicia o processamento; ao concluir, devolve a imagem segmentada, na qual cada tora aparece delimitada por uma máscara colorida sobre o fundo preto do recorte, acompanhada da contagem: 243 toras segmentadas. Essa devolução visual permite ao operador inspecionar a qualidade da segmentação antes de aceitar o resultado.

Na terceira etapa (Figura 6.7), o aplicativo apresenta o resultado final. Para essa pilha, operando em modo de uma face com largura de 3 m e comprimento de tora de 1 m, o sistema estimou um volume total de $2,67 \text{ m}^3$, correspondente a uma área basal de face de $2,67 \text{ m}^2$ e a 243 toras detectadas, explicitando o método empregado (fórmula de Smalian) e o modo de operação (uma face, com as duas faces assumidas iguais). O agente encerra oferecendo-se para corrigir qualquer parâmetro, hipótese em que o volume é recalculado sobre a mesma fotografia. A interface conversacional elimina a necessidade de instalação de programas e de conhecimento técnico especializado, permitindo que operadores de campo obtenham estimativas volumétricas diretamente de um dispositivo móvel comum.

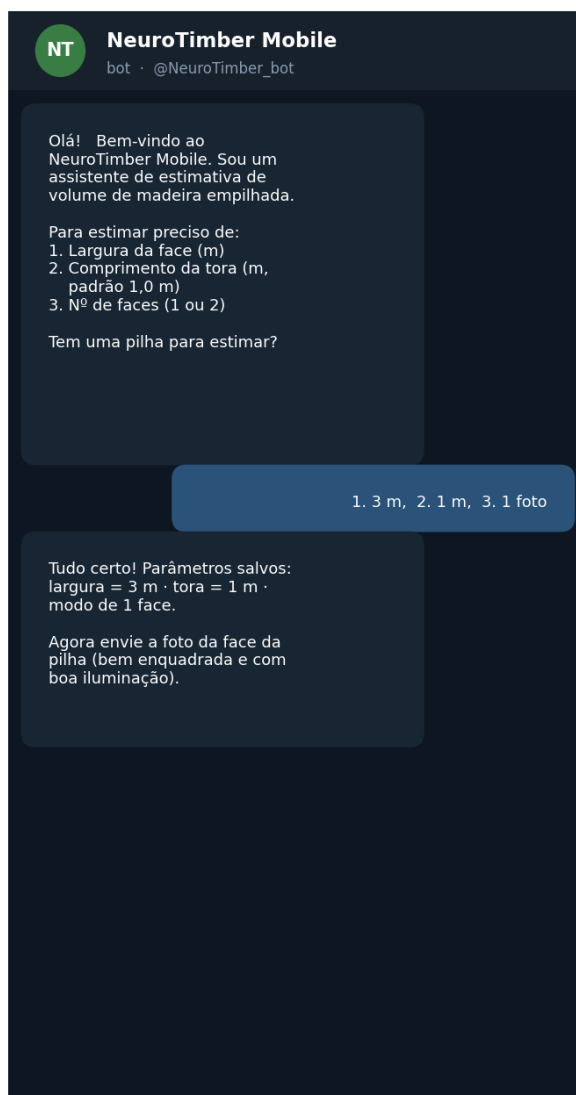


Figura 6.5. NeuroTimber Mobile (@NeuroTimber_bot) - etapa 1: apresentação, solicitação dos parâmetros (largura da face, comprimento da tora e número de faces) e confirmação dos valores informados pelo usuário.

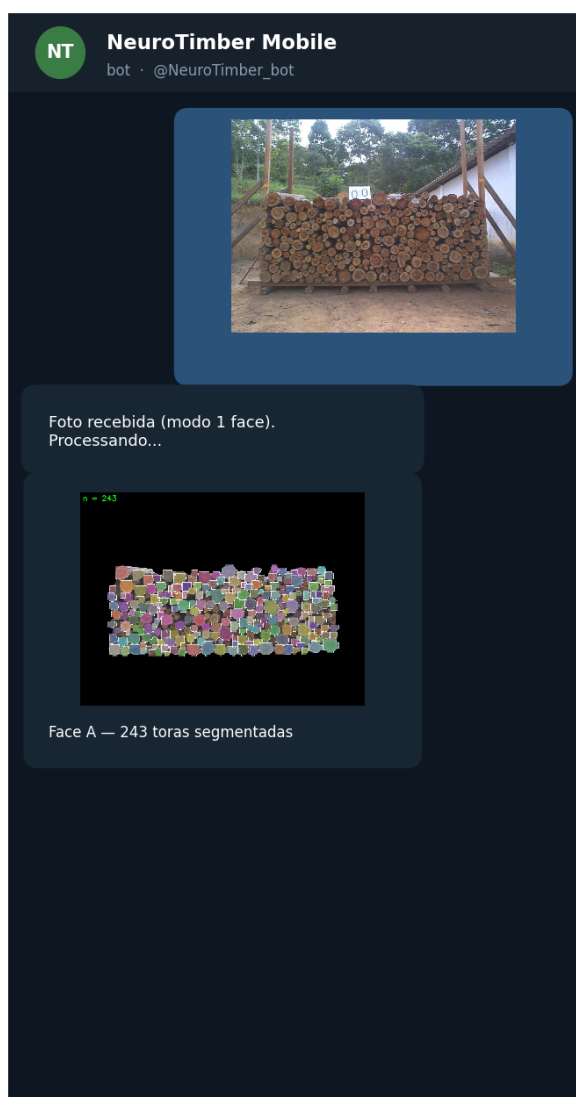


Figura 6.6. NeuroTimber Mobile - etapa 2: envio da fotografia da face da pilha e devolução da segmentação das toras (243 toras segmentadas, em modo de uma face).

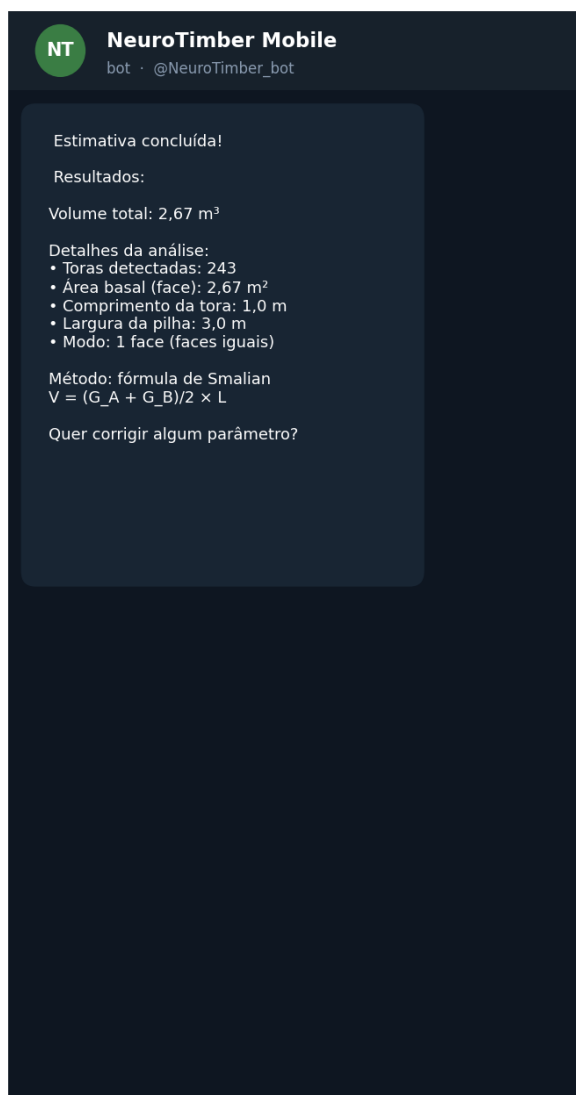


Figura 6.7. NeuroTimber Mobile - etapa 3: apresentação do resultado final (volume total de 2,67 m³, área basal de 2,67 m², 243 toras, comprimento de tora de 1 m, largura de 3 m e modo de uma face), com o método de cálculo explicitado.

6.4. APLICAÇÃO EM MODO OPERACIONAL: PROPOSTA

Os resultados apresentados foram obtidos em condições de estudo de caso, com processamento executado em uma estação de trabalho equipada com unidade de processamento gráfico (GPU). Para viabilizar o uso do NeuroTimber Mobile em escala e em condições reais de campo, propõe-se um modelo operacional de arquitetura cliente-servidor, no qual o dispositivo móvel atua apenas como cliente leve de captura e envio de imagens, enquanto todo o processamento de visão computacional é realizado remotamente em um servidor dedicado. A Figura 6.8 sintetiza esse fluxo operacional, organizado em quatro etapas: coleta de dados em campo, envio de imagens, processamento com inteligência artificial no servidor e obtenção dos resultados.

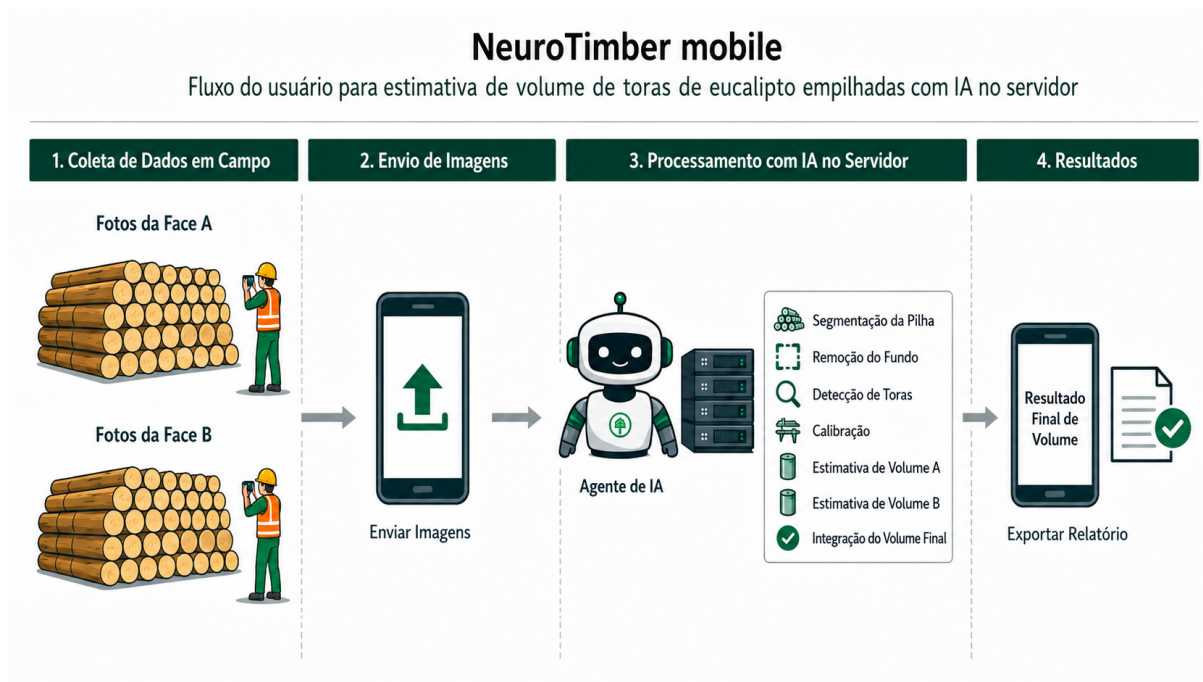


Figura 6.8. Proposta de fluxo operacional do NeuroTimber Mobile: (1) coleta de dados em campo, com a captura das fotografias da face A e da face B da pilha pelo operador; (2) envio das imagens a partir do dispositivo móvel; (3) processamento com IA no servidor, executado pelo agente sobre a sequência de segmentação da pilha, remoção do fundo, detecção das toras, calibração e estimativa de volume de cada face; e (4) apresentação e exportação do relatório com o volume final.

Na **primeira etapa**, o operador realiza a coleta de dados diretamente no campo ou no pátio, fotografando uma ou ambas as faces da pilha com a câmera de um dispositivo móvel comum. Essa etapa não exige instrumentos dedicados nem calibração prévia do equipamento: bastam fotografias bem enquadradas da face da pilha, o que reduz drasticamente o custo e o tempo de aquisição em relação aos métodos tradicionais de cubagem ou de medição por fatores de empilhamento.

Na **segunda etapa**, as imagens são enviadas ao servidor por meio do aplicativo de mensageria. Como o cliente é o próprio Telegram, dispensa-se a instalação de qualquer programa específico, e o envio pode ser feito assim que houver conectividade, mesmo que a captura tenha ocorrido em área sem sinal. Essa dissociação entre captura e envio é particularmente vantajosa em ambientes florestais, nos quais a conectividade é frequentemente intermitente.

A **terceira etapa** concentra a carga computacional no servidor e constitui o núcleo da arquitetura proposta. Os modelos empregados (SAM 3 para a segmentação e o recorte da pilha e YOLO26 para a segmentação das toras) possuem requisitos de memória e de processamento incompatíveis com a maioria dos dispositivos móveis, demandando aceleração por GPU para operar em tempo hábil. Ao centralizar a inferência em um servidor, garante-se desempenho consistente independentemente do aparelho do usuário, além de permitir a atualização dos modelos e dos parâmetros de forma centralizada, sem necessidade de reinstalação por parte dos operadores. O agente de IA hospedado no servidor orquestra toda a cadeia: recebe as imagens,

executa a segmentação da pilha, a remoção do fundo, a detecção e a segmentação das toras, a calibração métrica e a estimativa da área basal de cada face, integrando-as no volume final pela fórmula de Smalian. Essa centralização também favorece a governança dos dados, uma vez que as imagens, os resultados e os registros de processamento podem ser armazenados de forma padronizada e auditável.

Na **quarta etapa**, o resultado é devolvido ao usuário no próprio diálogo, acompanhado das imagens anotadas, e pode ser exportado como relatório para incorporação a sistemas de inventário e a controles de estoque. A interação com o usuário, mediada por linguagem natural, mantém-se simples em todas as etapas: o operador informa os parâmetros da pilha, envia as fotografias e recebe o volume estimado, sem contato direto com a complexidade dos modelos subjacentes. Do ponto de vista de escalabilidade, a arquitetura cliente-servidor permite atender simultaneamente a múltiplos operadores, cada um com seu estado de conversa independente, e viabiliza a evolução do sistema (novos modelos, novas espécies, novos modos de calibração) de maneira transparente ao usuário final. Cabe ressaltar que se trata de uma proposta de operacionalização; sua validação em ambiente produtivo, incluindo aspectos de latência de rede, disponibilidade do servidor, consumo de dados e integração com sistemas corporativos, constitui um desdobramento natural deste trabalho.

6.4.1. Recomendações para a aquisição das fotografias

A acurácia do NeuroTimber Mobile depende diretamente da qualidade geométrica e radiométrica das imagens enviadas, uma vez que toda a cadeia de estimativa (segmentação da pilha, segmentação das faces das toras e conversão de pixels em área) parte da fotografia fornecida pelo operador. A literatura sobre mensuração fotoóptica de madeira empilhada é convergente ao apontar o alinhamento da câmera em relação à pilha, a distância de tomada, as condições de iluminação e o fundo da cena como os principais fatores que condicionam o erro das estimativas (CREMER et al., 2020; BERENDT; WOLFGRAMM; CREMER, 2021; PURFÜRST et al., 2023; DE MIGUEL-DÍEZ; PURFÜRST et al., 2023). Com base nessas referências e nos ensaios conduzidos neste capítulo, recomenda-se o seguinte protocolo de aquisição.

- **Posicionamento frontal.** O observador deve posicionar-se de frente para a pilha, de modo que o eixo óptico da câmera fique perpendicular ao plano da face das toras. Tomadas em diagonal ou oblíquas devem ser evitadas, pois a projeção em perspectiva reduz a área aparente das faces circulares, transformando-as em elipses, e faz com que a escala varie ao longo da imagem: a porção mais próxima da câmera é registrada com mais pixels por metro do que a porção mais distante. Como a calibração métrica adotada assume uma escala única para toda a face, qualquer obliquidade introduz erro sistemático na estimativa da área basal e, conseqüentemente, do volume.
- **Centralização e altura da câmera.** A câmera deve ser mantida na metade da altura da

pilha e alinhada ao centro da face, de forma que o centro da imagem coincida com o centro geométrico da face fotografada. Isso distribui simetricamente a distorção residual da lente e evita a convergência das linhas verticais causada pela inclinação da câmera para cima ou para baixo.

- **Nivelamento.** Recomenda-se acionar a grade e o indicador de nível do aplicativo de câmera do dispositivo, mantendo o aparelho paralelo à face da pilha nos dois eixos, sem rotação lateral nem inclinação vertical.
- **Distância e enquadramento.** A fotografia deve ser tomada a uma distância aproximada de 2 a 4 m da pilha, ajustada de modo que a pilha ocupe cerca de 85% da altura do quadro (BERENDT; WOLFGRAMM; CREMER, 2021). A face inteira, incluindo suas duas extremidades laterais, deve estar contida na imagem, pois a largura total da face é a referência métrica utilizada na calibração; o corte de qualquer das bordas invalida a escala.
- **Óptica.** Deve-se utilizar a lente principal do dispositivo, evitando-se a lente grande-angular ou ultra-angular, cuja distorção de barril curva as bordas da imagem, bem como o zoom digital, que degrada a resolução efetiva das faces das toras. Quando for necessário ampliar o campo de visão, o operador deve afastar-se fisicamente da pilha em vez de recorrer a esses recursos.
- **Campo de visão desobstruído.** A face da pilha deve estar livre de obstruções, tais como vegetação, resíduos de colheita, equipamentos, veículos ou pessoas posicionadas entre o operador e a pilha, uma vez que oclusões parciais reduzem a área segmentada e subestimam o volume.
- **Iluminação.** Prefere-se iluminação difusa e uniforme, característica de dias nublados ou de áreas sombreadas. Devem ser evitados o sol direto e rasante, que projeta sombras acentuadas entre as toras e pode ser interpretado como espaço vazio pelo modelo de segmentação, bem como as tomadas em contraluz, que reduzem o contraste entre a madeira e o fundo. O uso de flash é desaconselhado por gerar reflexos especulares em faces úmidas.
- **Fundo da cena.** Um fundo limpo e contrastante em relação à madeira favorece a segmentação de contexto realizada pelo SAM 3. Sempre que possível, deve-se evitar que outras pilhas, troncos soltos ou tocos apareçam imediatamente atrás ou ao lado da pilha de interesse, o que poderia levar à inclusão indevida de área no recorte.
- **Nitidez.** A imagem deve estar em foco sobre a face das toras e livre de tremor. Recomenda-se resolução mínima de 1920 pixels no lado maior e apoio do aparelho, ou uso do temporizador, em condições de baixa luminosidade.
- **Preparo da pilha.** As faces devem estar razoavelmente alinhadas em um mesmo plano e limpas, com remoção de terra, serragem e casca solta que possam mascarar o contorno das

seções transversais. Pilhas muito irregulares, com toras salientes ou recuadas em relação ao plano da face, aumentam a dispersão das estimativas (BERENDT; WOLFGRAMM; CREMER, 2021).

- **Referência métrica.** A dimensão informada na calibração deve corresponder à largura da face fotografada, e não à profundidade da pilha, que equivale ao comprimento das toras. A confusão entre essas duas medidas foi a principal fonte de erro grosseiro identificada neste estudo, com subestimação de cerca de 89%. Quando a largura da face for de difícil medição, recomenda-se posicionar no plano da face um objeto de dimensão conhecida, como uma baliza ou uma trena estendida, para servir de referência de escala.
- **Repetição e modo de duas faces.** Recomenda-se registrar ao menos duas fotografias por face, repetindo a tomada caso a imagem anotada devolvida pelo aplicativo indique segmentação incompleta. No modo de duas faces, ambas as extremidades da pilha devem ser fotografadas seguindo integralmente o mesmo protocolo, de modo que a média das áreas empregada na fórmula de Smalian não seja afetada por diferenças de aquisição entre as duas tomadas.
- **Envio da imagem original.** A fotografia deve ser enviada sem recorte, rotação ou edição prévia, preservando a proporção original do quadro, uma vez que qualquer alteração posterior à captura compromete a relação entre a largura informada e a largura em pixels utilizada na calibração.

A adoção sistemática desse protocolo é o principal fator sob controle direto do operador e constitui condição necessária para reproduzir, em condições operacionais, a acurácia relatada neste capítulo.

6.5. DISCUSSÃO

Os resultados obtidos demonstram que uma calibração métrica simples, fornecida pelo próprio operador, é suficiente para converter a segmentação de imagens em estimativas acuradas de volume sólido de madeira empilhada. O erro médio percentual absoluto de 4,66% situa-se dentro da faixa de acurácia considerada adequada para fins de inventário e comercialização, sendo comparável ou superior a diversos métodos fotoópticos revisados por De Miguel-Díez, Purfürst et al. (2023). O elevado coeficiente de correlação (0,795) indica que o método captura de forma consistente a variação de volume entre pilhas, o que é particularmente relevante em contextos operacionais nos quais se busca ordenar e comparar lotes.

Um aspecto metodológico central deste estudo é a demonstração de que a escolha do modelo de visão computacional deve ser guiada pela grandeza física de interesse. Enquanto detectores baseados em caixas delimitadoras, como as versões clássicas do YOLO (REDMON et al., 2016; CASAS et al., 2023), são adequados para a contagem de toras, a estimativa de área basal exige

a delimitação da seção circular da madeira, tarefa para a qual os modelos de segmentação de instâncias são intrinsecamente mais apropriados. A adoção do SAM 3 (CARION et al., 2025) para o recorte da pilha e do YOLO26 para a segmentação das toras materializa essa complementaridade: a segmentação de contexto fornece a referência de escala (a largura do recorte), e a segmentação de instâncias fornece a área a ser escalonada. A quase coincidência entre a largura ótima recuperada (2,98 m) e a largura real (3 m) confirma que o encadeamento desses dois modelos preserva a fidelidade geométrica necessária à mensuração.

A dependência de uma medida fornecida pelo usuário constitui, simultaneamente, a principal força e a principal limitação do método. Por um lado, dispensar catálogos de diâmetro e modelos probabilísticos torna o procedimento transparente, auditável e independente de dados prévios da região, além de reduzir a incerteza associada à modelagem estatística. Por outro lado, a acurácia fica condicionada à qualidade da medida de largura informada: o experimento de retroajuste evidenciou que a adoção equivocada da profundidade da pilha como largura conduziu a subestimação de cerca de 89%, correspondente à razão entre os quadrados das larguras. Esse achado alinha-se às observações de Barros (2017) quanto à sensibilidade dos métodos baseados em imagem às condições de aquisição e às definições geométricas, e reforça a importância de instruções claras ao operador. A interface conversacional do NeuroTimber Mobile mitiga esse risco ao solicitar explicitamente cada parâmetro e ao distinguir, no diálogo, a largura da face do comprimento da tora.

A entrega da solução por meio de um aplicativo de mensageria insere-se em uma tendência consolidada de aproximar tecnologias de inteligência artificial dos usuários de campo. Assim como o ForestScanner (TATSUMI; YAMAGUCHI; FURUYA, 2023) tornou acessível a mensuração dendrométrica em *smartphones*, e assistentes como o AgronomoBot (MOSTAÇO et al., 2018) demonstraram a eficácia do Telegram como canal de serviços técnicos agrícolas, o NeuroTimber Mobile utiliza a mensageria instantânea para eliminar barreiras de instalação e de capacitação. Essa escolha de projeto favorece a adoção por equipes com perfis técnicos heterogêneos e viabiliza o uso em condições de campo com conectividade limitada, nas quais um aplicativo leve e conversacional é preferível a soluções que exijam *hardware* dedicado ou processamento local intensivo.

Entre as limitações do estudo, destaca-se a avaliação restrita a pilhas de *Eucalyptus grandis* montadas sob condições controladas, com largura de face padronizada. Em situações operacionais, a largura da face pode variar ao longo da pilha e ser de difícil medição, o que demanda estudos futuros com pilhas de dimensões irregulares e com estratégias de referência de escala alternativas, como marcadores de dimensão conhecida posicionados na cena. Recomenda-se, ainda, ampliar o conjunto de espécies e de condições de empilhamento e avaliar a robustez da segmentação sob variações mais extremas de iluminação e oclusão. Apesar dessas ressalvas, os resultados sustentam a viabilidade do NeuroTimber Mobile como ferramenta prática de estimativa volumétrica, conciliando acurácia, simplicidade operacional e acessibilidade.

6.6. CONCLUSÕES

O NeuroTimber Mobile demonstrou que a integração de segmentação de contexto (SAM 3) e de segmentação de instâncias (YOLO26) com uma calibração métrica fornecida pelo operador permite estimar o volume sólido de madeira empilhada com acurácia adequada a fins de inventário. No estudo de caso com 30 pilhas de *Eucalyptus grandis*, o método alcançou erro médio percentual absoluto de 4,66%, viés reduzido (+1,66%) e alta correlação (0,795) com os volumes de referência, sem recorrer a catálogos de diâmetro ou a modelos de distribuição probabilística.

A recuperação de uma largura ótima implícita de 2,98 m, praticamente idêntica à largura real do quadro experimental, confirmou a consistência física do procedimento e evidenciou que a contagem dos pixels da face circular das toras, corretamente escalonada, reproduz a área basal real das pilhas. O estudo também alertou para a criticidade de fornecer a dimensão correta na calibração, uma vez que a confusão entre a largura da face e a profundidade da pilha conduz a erros grosseiros. Em decorrência disso, foi estabelecido um protocolo de aquisição das fotografias, que prescreve o posicionamento frontal do observador, com o eixo óptico da câmera perpendicular à face da pilha, o enquadramento integral da face a uma distância de 2 a 4 m, a iluminação difusa e o envio da imagem original, sem edição. O cumprimento desse protocolo é condição necessária para reproduzir em campo a acurácia aqui relatada.

Ao disponibilizar todo o fluxo em um agente conversacional no Telegram (@NeuroTimber_bot), o sistema torna a estimativa volumétrica acessível diretamente de um dispositivo móvel comum, sem instalação de programas nem conhecimento técnico especializado, e com suporte tanto ao modo de duas faces quanto ao modo de uma face. Dessa forma, o NeuroTimber Mobile contribui para a modernização e a democratização da mensuração de madeira empilhada, oferecendo uma alternativa transparente, auditável e de baixo custo aos métodos tradicionais baseados em fatores de empilhamento.

6.7. REFERÊNCIAS

- BARROS, V. A. de. **Determinação de fatores de empilhamento por meio de classificação de imagem utilizando redes neurais**. 2017. Dissertação (Mestrado em Ciência Florestal) - Universidade Federal de Viçosa, Viçosa, 2017.
- CARION, N.; GUSTAFSON, L.; HU, Y.-T.; DEBNATH, S.; HU, R.; SURIS, D.; RYALI, C.; ALWALA, K. V.; KHEDR, H.; HUANG, A.; LEI, J.; MA, T.; GUO, B.; KALLA, A.; MARKS, M.; GREER, J.; WANG, M.; SUN, P.; RÄDLE, R.; AFOURAS, T.; MAVROUDI, E.; XU, K.; WU, T.-H.; ZHOU, Y.; MOMENI, L.; HAZRA, R.; DING, S.; VAZE, S.; PORCHER, F.; LI, F.; LI, S.; KAMATH, A.; CHENG, H. K.; DOLLÁR, P.; RAVI, N.; SAENKO, K.; ZHANG, P.; FEICHTENHOFER, C. **SAM 3: Segment Anything with Concepts**. arXiv:2511.16719, 20 nov. 2025. <https://doi.org/10.48550/arXiv.2511.16719>.

- CASAS, G. G.; ISMAIL, Z. H.; LIMEIRA, M. M. C.; DA SILVA, A. A. L.; LEITE, H. G. Automatic Detection and Counting of Stacked Eucalypt Timber Using the YOLOv8 Model. **Forests**, v. 14, n. 12, p. 2369, 2023.
- BERENDT, F.; WOLFGRAMM, F.; CREMER, T. Reliability of photo-optical measurements of log stack gross volume. **Silva Fennica**, v. 55, n. 3, article id 10555, 2021. <https://doi.org/10.14214/sf.10555>.
- CREMER, T.; BERENDT, F.; DE MIGUEL-DÍEZ, F.; WOLFGRAMM, F.; BLASKO, L. Accuracy of Photo-Optical Measurement of Wood Piles. In: **The 1st International Electronic Conference on Forests**. Basel: MDPI, 2020. p. 90. <https://doi.org/10.3390/IECF2020-08192>.
- DE MIGUEL-DÍEZ, F.; PURFÜRST, T.; TOLOSANA-ESTEBAN, E.; MULLER, M.; CREMER, T. Estimation of conversion factors for wood stacks in landings and their influencing parameters: a comprehensive literature review for America and Europe. **Silva Fennica**, v. 57, n. 1, article id 22018, 2023.
- JOCHER, G.; QIU, J.; LIU, M.; LYU, S.; AKYON, F. C.; KALFAOGLU, M. E. **Ultralytics YOLO26: Unified Real-Time End-to-End Vision Models**. arXiv:2606.03748, 2026. <https://doi.org/10.48550/arXiv.2606.03748>.
- KIRILLOV, A.; MINTUN, E.; RAVI, N.; MAO, H.; ROLLAND, C.; GUSTAFSON, L.; XIAO, T.; WHITEHEAD, S.; BERG, A. C.; LO, W.-Y.; DOLLÁR, P.; GIRSHICK, R. Segment Anything. In: **Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)**, 2023, p. 4015-4026.
- MACHADO, S. A.; FIGUEIREDO FILHO, A. **Dendrometria**. Curitiba: [s.n.], 2003. 309 p.
- MOSTAÇO, G. M.; SOUZA, Í. R. C. de; CAMPOS, L. B.; CUGNASCA, C. E. AgronomoBot: a smart answering chatbot applied to agricultural sensor networks. In: **14th International Conference on Precision Agriculture**, Montreal, 2018.
- PURFÜRST, T.; DE MIGUEL-DÍEZ, F.; BERENDT, F.; ENGLER, B.; CREMER, T. Comparison of wood stack volume determination between manual, photo-optical, iPad-LiDAR and handheld-LiDAR based measurement methods. **iForest - Biogeosciences and Forestry**, v. 16, n. 4, p. 243-252, 2023. <https://doi.org/10.3832/ifor4153-016>.
- RAVI, N.; GABEUR, V.; HU, Y.-T.; HU, R.; RYALI, C.; MA, T.; KHEDR, H.; RÄDLE, R.; ROLLAND, C.; GUSTAFSON, L.; MINTUN, E.; PAN, J.; ALWALA, K. V.; CARION, N.; WU, C.-Y.; GIRSHICK, R.; DOLLÁR, P.; FEICHTENHOFER, C. SAM 2: Segment Anything in Images and Videos. **arXiv preprint** arXiv:2408.00714, 2024.
- REDMON, J.; DIVVALA, S.; GIRSHICK, R.; FARHADI, A. You Only Look Once: Unified, Real-Time Object Detection. In: **Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)**, 2016, p. 779-788.

SOARES, C. P. B.; PAULA NETO, F.; SOUZA, A. L. **Dendrometria e inventário florestal**. 2. ed. Viçosa: Editora UFV, 2011. 272 p.

TATSUMI, S.; YAMAGUCHI, K.; FURUYA, N. ForestScanner: A mobile application for measuring and mapping trees with LiDAR-equipped iPhone and iPad. **Methods in Ecology and Evolution**, v. 14, n. 7, p. 1603-1609, 2023.

CONCLUSÕES GERAIS

Os seis capítulos do trabalho convergem para um fluxo coeso que vai do fundamento à aplicação. Os conceitos de detecção de uma passada com CNNs e o papel de métricas como precisão, recall, mAP e F1, aliados ao NMS, forneceram o arcabouço para projetar e avaliar modelos YOLO em cenários com alta sobreposição e objetos pequenos. Sobre essa base, a evidência mostrou que a contagem automática é robusta para toras isoladas em fotografias e que, em pilhas industriais, o uso de vídeo com múltiplos quadros e rastreamento reduz substancialmente a subcontagem observada em imagens estáticas. Nos testes, fotografias apresentaram erros relativos da ordem de $-32,3\%$ a $-47,5\%$, enquanto o vídeo apresentou erro de $+1,6\%$ no total da pilha, sem viés estatisticamente detectável, sustentando a recomendação operacional de priorizar vídeo para pilhas e reservar fotos para casos de toras separadas.

A ponte estatística entre contagem e volume sólido demonstrou viabilidade prática ao combinar a contagem global com a distribuição de diâmetros ajustada por máxima verossimilhança e selecionada por AIC, com a GEV se destacando no ajuste aos dados disponíveis. O procedimento resultou em estimativas consistentes no estudo de caso, com exemplo de 2.698 toras e $238,36 \text{ m}^3$, erro médio de $-4,535\%$ em 30 simulações e diferença de $-0,5898\%$ frente a Smalian em amostra independente, reforçando acurácia e estabilidade. A materialização em sistema reprodutível com YOLOv9 e ByteTrack para contagem em vídeo e SciPy para ajuste e cálculo volumétrico viabiliza uso real com transparência e rastreabilidade. Persistem desafios como oclusões, variação de textura e necessidade de ampliar e diversificar o conjunto de imagens, mas o pipeline está pronto para escalar, com diretrizes claras de coleta de dados, rastreamento e reporte de métricas que sustentam comparabilidade ao longo de novas versões do modelo e de futuras aplicações em silvicultura de precisão.

Por fim, o NeuroTimber Mobile amplia essa trajetória ao levar a estimativa volumétrica para um aplicativo de mensageria, substituindo o catálogo de diâmetros por uma calibração métrica direta fornecida pelo operador e combinando a segmentação de contexto (SAM 3) com a segmentação de instâncias das toras (YOLO26). No estudo de caso com 30 pilhas de *Eucalyptus grandis*, o método alcançou erro médio percentual absoluto de $4,66\%$ e correlação de $0,795$ com os volumes de referência, e a recuperação de uma largura ótima de $2,98 \text{ m}$, praticamente idêntica à largura real do quadro experimental, confirmou a consistência física do procedimento. Em conjunto, os seis capítulos consolidam um percurso que parte dos fundamentos da visão computacional e culmina em ferramentas acessíveis e auditáveis para a mensuração de madeira empilhada, conciliando rigor científico, reprodutibilidade e aplicabilidade prática em campo.