



Izaquiel de Queiroz Ferreira

# **Detecção Monocular de Cilindros Pressurizados com Aplicações em Segurança Industrial**

Recife

2023

Izaquiel de Queiroz Ferreira

# **Detecção Monocular de Cilindros Pressurizados com Aplicações em Segurança Industrial**

Dissertação submetida ao Programa de Pós-Graduação em Informática Aplicada da Universidade Federal Rural de Pernambuco, como requisito parcial para obtenção do título de Mestre em Informática Aplicada

Universidade Federal Rural de Pernambuco – UFRPE

Departamento de Estatística e Informática

Programa de Pós-Graduação em Informática Aplicada - PPGIA

Orientador: João Paulo Silva do Monte Lima

Recife

2023

Dados Internacionais de Catalogação na Publicação  
Universidade Federal Rural de Pernambuco  
Sistema Integrado de Bibliotecas  
Gerada automaticamente, mediante os dados fornecidos pelo(a) autor(a)

---

- F383d Ferreira, Izaquiel de Queiroz  
Detecção Monocular de Cilindros Pressurizados com Aplicações em Segurança Industrial / Izaquiel de Queiroz Ferreira. - 2023.  
78 f. : il.
- Orientador: Joao Paulo Silva do Monte Lima.  
Inclui referências e anexo(s).
- Dissertação (Mestrado) - Universidade Federal Rural de Pernambuco, Programa de Pós-Graduação em Informática Aplicada, Recife, 2023.
1. YOLOv5. 2. YOLOv6. 3. YOLOv7. 4. YOLOv8. 5. Detecção de objetos. I. Lima, Joao Paulo Silva do Monte, orient. II. Título

*Ao meu pai e meu irmão,  
onde estiverem.*

# Agradecimentos

Primeiramente, quero expressar minha mais profunda gratidão a Deus, cuja constante inspiração e orientação me acompanharam ao longo desta jornada. Sua luz iluminou meu caminho e me deu forças para superar os desafios que surgiram durante a elaboração desta dissertação.

À minha amada família, que sempre esteve ao meu lado, oferecendo apoio incondicional, amor e encorajamento. Suas palavras de incentivo foram meu combustível nos momentos de dúvida, e sua presença constante foi meu porto seguro.

À Coordenação de Aperfeiçoamento de Pessoal de Nível Superior (Capes), pelo valioso financiamento concedido, que possibilitou a concretização deste mestrado. Sua contribuição foi essencial para que eu pudesse me dedicar aos estudos e à pesquisa.

Aos meus amigos, cujo apoio e camaradagem fizeram desta jornada acadêmica uma experiência verdadeiramente enriquecedora. Suas conversas, risadas e gestos de amizade me lembraram da importância de manter um equilíbrio saudável entre o trabalho e o convívio social.

Ao meu dedicado orientador, João Paulo, que não apenas compartilhou seu vasto conhecimento, mas também demonstrou um compromisso inabalável em me guiar rumo ao sucesso. Sua paciência, sabedoria e motivação foram cruciais para que eu não desistisse diante dos obstáculos que surgiram.

Aos estimados membros da banca, João Marcelo e Valmir Macario, que aceitaram o convite para avaliar este trabalho. Suas contribuições e *feedbacks* construtivos enriquecerão sobremaneira esta dissertação, e sou grato por dedicarem seu tempo e expertise à sua avaliação.

E, por último mas não menos importante, à minha querida esposa, Joana. Sua compreensão, apoio e amor incondicional foram fundamentais para minha perseverança durante todo esse processo. Sua presença ao meu lado trouxe alegria e serenidade, tornando cada passo deste percurso ainda mais significativo.

O presente trabalho foi realizado com apoio da Coordenação de Aperfeiçoamento de Pessoal de Nível Superior - Brasil (CAPES) - Código de Financiamento 001.

*“...A mão de Deus abençoa tudo o quanto o homem faz,  
sabendo que ele é capaz de ter atitude boa,  
até os erros perdoa, lhe conduzindo pra trilha.  
Por isso meu mundo brilha, eu sei do lugar que venho.  
Toda vitória que tenho, dedico a minha família.  
(Poeta Marquinhos da Serrinha)*

# Resumo

A gestão eficaz de resíduos industriais é uma preocupação global crescente devido aos potenciais riscos ambientais e de segurança associados a esses materiais. Entre os resíduos industriais, os cilindros pressurizados, utilizados para armazenar gases tóxicos e inflamáveis, destacam-se devido ao seu potencial para causar acidentes e danos ambientais. A detecção manual de tais cilindros em pilhas de sucatas é ineficiente e propensa a erros. A visão computacional oferece uma solução poderosa e eficiente para a detecção de objetos em imagens. Nessa dissertação, explorou-se a aplicação do algoritmo YOLO (*You Only Look Once*) na detecção de cilindros pressurizados em pilhas de sucatas. Foi criado um conjunto de dados com 1799 imagens de cilindros coletadas da internet, rotuladas e divididas em conjuntos de treinamento (80%) e validação (20%). Quatro versões atuais do YOLO foram utilizadas e comparadas no treinamento, resultando em uma taxa de precisão média (*mAP*) de 78% na detecção de cilindros no conjunto de validação. A pesquisa contribui para a gestão de resíduos industriais ao fornecer uma ferramenta automatizada para detectar cilindros pressurizados, aumentando a segurança e reduzindo riscos ambientais e de segurança. Embora os resultados tenham sido promissores, novos testes em ambientes reais e a expansão do conjunto de dados são recomendados para melhorar a variabilidade e a precisão da detecção. Também se sugere investigar a detecção de objetos em oclusão parcial usando o YOLO. Essa dissertação oferece uma contribuição significativa à área de visão computacional aplicada à gestão de resíduos industriais ao realizar o treinamento e análise comparativa com diferentes arquiteturas do YOLO, ampliando o conhecimento científico e tecnológico nesse campo específico. Ela destaca o potencial da visão computacional e do algoritmo YOLO na detecção de objetos críticos em ambientes desafiadores, como pilhas de sucatas. No entanto, é fundamental reconhecer que, sem testes em um ambiente real, é necessário realizar essas avaliações para afirmar com segurança a eficácia, especialmente considerando a complexidade intrínseca de detectar objetos em pilhas de sucatas e ambientes ruidosos.

**Palavras-chave:** YOLOv5, YOLOv6, YOLOv7, YOLOv8, Detecção de objetos, Cilindros.

# Abstract

The effective management of industrial waste is a growing global concern due to the potential environmental and safety risks associated with these materials. Among industrial wastes, pressurized cylinders used to store toxic and flammable gases stand out due to their potential to cause accidents and environmental damage. Manual detection of such cylinders in scrap piles is inefficient and error-prone. Computer vision provides a powerful and efficient solution for object detection in images. In this dissertation, the application of the YOLO (You Only Look Once) algorithm in detecting pressurized cylinders in scrap piles was explored. A dataset of 1799 cylinder images collected from the internet was created, labeled, and divided into training (80%) and validation (20%) sets. Four current versions of YOLO were used and compared in training, resulting in an average precision rate (mAP) of 78% in cylinder detection on the validation set. The research contributes to industrial waste management by providing an automated tool for detecting pressurized cylinders, enhancing safety, and reducing environmental and safety risks. While the results were promising, further tests in real environments and dataset expansion are recommended to improve detection variability and accuracy. Investigating object detection in partial occlusion using YOLO is also suggested. This dissertation provides a significant contribution to the field of computer vision applied to industrial waste management by conducting training and comparative analysis with different YOLO architectures, expanding scientific and technological knowledge in this specific field. However, it is crucial to acknowledge that, without tests in a real-world environment, it is necessary to conduct these assessments to confidently assert the effectiveness, particularly considering the intrinsic complexity of detecting objects in scrap heaps and noisy environments.

**Keywords:** YOLOv5, YOLOv6, YOLOv7, YOLOv8, Object Detection, Cylinders.

# Lista de ilustrações

Figura 1 – Resíduos industriais em pilhas de sucatas. . . . .	14
Figura 2 – Classificação, localização, detecção e segmentação em imagem de objetos únicos e múltiplos. . . . .	16
Figura 3 – Fluxo da abordagem GS-IST, onde as caixas possuem as etapas do método e abaixo delas estão a saída de cada etapa. . . . .	19
Figura 4 – Experimentos na reconstrução de plantas de dutos a partir de nuvens de pontos de grande escala. Primeira linha: nuvem de pontos. Segunda linha: dutos detectados pelo método proposto. Terceira linha: plantas de dutos reconstruídas. . . . .	20
Figura 5 – Exemplo de detecção automática de cilindro usando a técnica proposta. (esquerda) Nuvem de pontos de uma planta petroquímica real. (à direita) Cilindros detectados mostrados como malhas poligonais realçadas. . . . .	21
Figura 6 – Dados da nuvem de pontos do cilindro escalonado. . . . .	22
Figura 7 – Uma visão aproximada do STD (esquerda) contra <i>outliers</i> em comparação com o método baseado em área de Patil et al. (2017) (direita). Os pontos azuis representam amostras de cilindros detectados. O STD é robusto contra outliers, enquanto o método baseado em área é sensível a <i>outliers</i> . . . . .	23
Figura 8 – Exemplos de classificações corretas utilizando o MHS comparando-se com classificação de outra CNN. . . . .	25
Figura 9 – Avaliação qualitativa da estrutura. . . . .	26
Figura 10 – A primeira coluna: imagem RGB, a segunda coluna: imagem de profundidade, a terceira coluna: MSERs detectados, a quarta coluna: resultados finais da extração do cilindro. Para o método proposto, todos os cilindros são detectados e extraídos simultaneamente. . . . .	28
Figura 11 – Exemplos qualitativos obtidos em diferentes tipos de cenas, gerados a partir de dados de varreduras reais. . . . .	29
Figura 12 – Exemplos de resultados obtidos a partir das imagens incluídas nos dados de teste. . . . .	31
Figura 13 – Modelo gráfico de uma <i>LeNet</i> . . . . .	37
Figura 14 – Estrutura das camadas de uma Rede Neural Convolutiva. . . . .	38
Figura 15 – Ilustração da convolução entre um filtro 3x3 e o volume de entrada. . . . .	40
Figura 16 – Aplicação de <i>max pooling</i> em uma imagem 4x4 utilizando um filtro 2x2. . . . .	42

Figura 17 – Ilustração da extração de características de uma imagem por uma CNN e sua posterior classificação. . . . .	42
Figura 18 – Estrutura do YOLOv5. . . . .	46
Figura 19 – Análise de desempenho de mAP de versões do YOLO sobre o dataset COCO. . . . .	47
Figura 20 – Fluxograma de desenvolvimento da pesquisa. . . . .	48
Figura 21 – Interface do <i>software</i> de rotulação e categorização <i>LabelImg</i> . . . . .	50
Figura 22 – Exemplo de mistura de imagens por meio de corte e aplicação de <i>patches</i> aleatórios com IDA. . . . .	52
Figura 23 – <i>Batch</i> de imagens gerado com YOLOv5. . . . .	55
Figura 24 – <i>Batch</i> de imagens gerado com YOLOv6. . . . .	56
Figura 25 – <i>Batch</i> de imagens gerado com YOLOv7. . . . .	57
Figura 26 – <i>Batch</i> de imagens gerado com YOLOv8. . . . .	58
Figura 27 – Análise comparativa de convergência de mAP entre os modelos. . . . .	59
Figura 28 – Resultados gráficos do treinamento com YOLOv5 . . . . .	60
Figura 29 – Resultados gráficos do treinamento com YOLOv6 . . . . .	61
Figura 30 – Resultados gráficos do treinamento com YOLOv7 . . . . .	61
Figura 31 – Resultados gráficos do treinamento com YOLOv8 . . . . .	62
Figura 32 – Predição de conjunto de validação com YOLOv5 . . . . .	63
Figura 33 – Predição de conjunto de validação com YOLOv6. . . . .	64
Figura 34 – Predição de conjunto de validação com YOLOv7. . . . .	65
Figura 35 – Predição de conjunto de validação com YOLOv8. . . . .	66
Figura 36 – Testes em imagens reais de pilhas de sucatas de uma siderúrgica. . . . .	67
Figura 37 – Análise comparativa de convergência de mAP entre os modelos sem aumento de dados. . . . .	68

# Lista de tabelas

Tabela 1 – Trecho do arquivo com rotulações para YOLO gerado para a base de imagens . . . . .	50
Tabela 2 – Resultados de validação de cada treinamento . . . . .	59
Tabela 3 – Matriz de confusão do treinamento com YOLOv5 . . . . .	60
Tabela 4 – Matriz de confusão do treinamento com YOLOv6 . . . . .	60
Tabela 5 – Matriz de confusão do treinamento com YOLOv7 . . . . .	60
Tabela 6 – Matriz de confusão do treinamento com YOLOv8 . . . . .	60

# Lista de abreviaturas e siglas

AP	<i>Average Precision</i>
CNN	<i>Convolutional Neural Network</i>
CPU	<i>Central Processing Unit</i>
FP	<i>False Positive</i>
FN	<i>False Negative</i>
GS-IST	<i>Geometric and Statistical Incremental Semantic Tracking</i>
IDA	<i>Image Data Augmentation</i>
mAP	<i>Mean Average Precision</i>
MDP	<i>Markov Decision Process</i>
MHS	<i>Multilayer Hybrid Deep-learning System</i>
MLP	<i>Multi Layer Perceptron</i>
MOT	<i>Multi object tracking</i>
MSER	<i>Maximally Stable Extremal Regions</i>
P	Precisão
PR	<i>Precision Recall</i>
R	<i>Recall</i>
ReLU	<i>Rectified Linear Units</i>
SLAM	<i>Simultaneous Localization and Mapping algorithm</i>
STD	<i>Slicing-tracking-detection</i>
TP	<i>True Positive</i>
VC	Visão computacional
YOLO	<i>You Only Look Once</i>

# Sumário

	<b>Lista de ilustrações</b> . . . . .	<b>8</b>
<b>1</b>	<b>INTRODUÇÃO</b> . . . . .	<b>14</b>
1.1	Objetivos do Trabalho . . . . .	17
1.2	Visão Geral . . . . .	17
<b>2</b>	<b>TRABALHOS RELACIONADOS</b> . . . . .	<b>18</b>
2.1	Nuvens de pontos . . . . .	18
2.2	Multilayer Hybrid Deep-learning System . . . . .	24
2.3	RGB-D . . . . .	25
2.4	MSER . . . . .	27
2.5	Teoria dos Jogos . . . . .	29
2.6	YOLOv5 . . . . .	30
<b>3</b>	<b>FUNDAMENTAÇÃO TEÓRICA</b> . . . . .	<b>33</b>
3.1	Gestão de Resíduos Industriais . . . . .	33
3.2	Visão Computacional . . . . .	34
3.2.1	Reconhecimento de Padrões . . . . .	35
3.2.2	Redes Neurais Convolucionais . . . . .	37
3.2.2.1	Camada Convolucional . . . . .	39
3.2.2.2	Camada de <i>Pooling</i> . . . . .	41
3.2.2.3	Camada Totalmente Conectada . . . . .	42
3.2.3	Deteção de Objetos . . . . .	43
3.2.3.1	YOLO . . . . .	45
<b>4</b>	<b>METODOLOGIA</b> . . . . .	<b>48</b>
4.1	Fluxograma . . . . .	48
4.2	Criação do Dataset de Imagens . . . . .	49
4.3	Treinamento . . . . .	51
4.4	Métricas de Avaliação . . . . .	52
<b>5</b>	<b>RESULTADOS E DISCUSSÃO</b> . . . . .	<b>54</b>
5.1	Discussões . . . . .	66
<b>6</b>	<b>CONCLUSÕES E TRABALHOS FUTUROS</b> . . . . .	<b>70</b>
6.1	Conclusões . . . . .	70
6.2	Contribuições . . . . .	71

<b>6.3</b>	<b>Trabalhos Futuros</b> . . . . .	<b>71</b>
	<b>REFERÊNCIAS</b> . . . . .	<b>73</b>
	<b>ANEXO A – HYP.VOC.YAML</b> . . . . .	<b>80</b>

# 1 Introdução

A gestão adequada de resíduos industriais é uma preocupação crescente em todo o mundo ([HOVEIDI et al., 2013](#)), devido aos potenciais riscos ambientais e de segurança que esses materiais podem representar. Entre os diversos tipos de resíduos gerados, os cilindros pressurizados, como aqueles utilizados para armazenar gases inflamáveis e tóxicos, merecem uma atenção especial devido ao seu potencial de causar acidentes graves e danos ao meio ambiente. A Figura 1 ilustra a complexidade para detectar objetos em uma pilha de sucatas.

Figura 1 – Resíduos industriais em pilhas de sucatas.



Fonte: [Allonda \(2023\)](#)

Diferentes países e regiões desenvolveram regulamentações e diretrizes para regular a gestão de resíduos industriais ([AZZAHRA; SAPTARINI, 2021](#)). Essas regulamentações têm como objetivo garantir que as indústrias cumpram práticas adequadas de gestão de resíduos e minimizem seu impacto ambiental ([AZZAHRA; SAPTARINI, 2021](#)). Elas podem incluir requisitos para segregação, armazenamento, transporte, tratamento e disposição de resíduos ([AZZAHRA; SAPTARINI, 2021](#)).

A gestão de resíduos industriais, especialmente no que se refere a metais e cilindros pressurizados, é um aspecto crítico para garantir a sustentabilidade ambiental e minimizar os impactos negativos das atividades industriais. Vários estudos têm sido conduzidos para avaliar e melhorar as práticas de gestão de resíduos industriais em diferentes regiões.

Um estudo realizado na zona industrial de Savojbolagh, no Irã, examinou as práticas de gestão de resíduos industriais ([FARZADKIA et al., 2020](#)). O estudo constatou que os tipos e proporções de resíduos gerados variavam dependendo do número e categoria das indústrias, dos processos de fabricação e das atividades de gestão de

resíduos. Neste caso específico, os principais materiais de resíduos perigosos gerados nas indústrias cosméticas e farmacêuticas incluíam medicamentos vencidos, soluções laboratoriais e tambores e sacos contaminados.

Em termos de composição de resíduos, um estudo realizado na região sul do estado do Rio de Janeiro, no Brasil, identificou papel e papelão, plástico, madeira e metais como os tipos mais comuns de resíduos gerados a partir de embalagens e recipientes de produtos (PEREIRA et al., 2007). O estudo também destacou a necessidade de um sistema integrado de gestão de resíduos industriais para abordar questões como oportunidades de energia a partir de resíduos e o descarte irregular.

Da mesma forma, um estudo realizado no Kuwait analisou as práticas de gestão de resíduos sólidos industriais no país (ALHUMOUD; AL-KANDARI, 2008). O estudo constatou que papel e papelão, plástico, madeira e metais eram os tipos mais comuns de resíduos, gerados principalmente a partir de embalagens e envoltórios de produtos. O estudo também destacou a falta de práticas de reutilização, reciclagem e valorização na gestão de resíduos sólidos industriais.

Esses estudos citados enfatizam a importância de práticas eficazes de gestão de resíduos industriais, incluindo o manejo adequado e a disposição de resíduos perigosos, a otimização de materiais de barreira para a disposição de resíduos e a necessidade de sistemas integrados de gestão de resíduos. A implementação dessas práticas pode contribuir para minimizar o impacto ambiental das atividades industriais e promover a sustentabilidade.

Na indústria, a detecção precisa e eficiente de cilindros pressurizados em pilhas de sucatas é um desafio significativo para os gestores de resíduos e os profissionais de segurança. A detecção visual manual é uma abordagem ineficiente e propensa a erros, pois depende da capacidade humana de identificar esses cilindros entre uma grande quantidade de resíduos diversos. Dentre os possíveis erros, destacam-se:

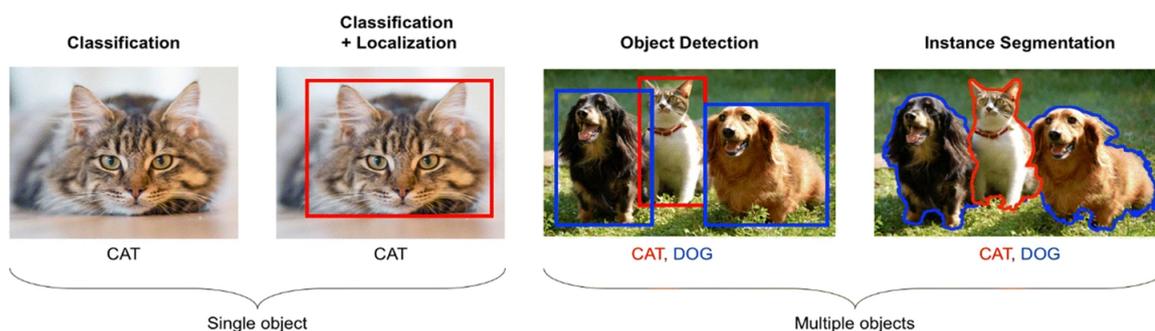
- **Omissões:** A capacidade limitada do olho humano pode resultar na falta de identificação de alguns cilindros, levando a omissões na detecção manual;
- **Fadiga Visual:** A fadiga visual dos profissionais encarregados da detecção pode levar a lapsos na atenção, aumentando o risco de não identificação adequada dos cilindros;
- **Interpretação Subjetiva:** A detecção visual manual está sujeita à interpretação subjetiva dos operadores, e diferentes níveis de habilidade entre eles podem resultar em inconsistências nos resultados;
- **Inconsistências Operacionais:** A variação na habilidade e experiência dos operadores pode contribuir para inconsistências na identificação dos cilindros, intro-

duzindo imprecisões no processo; e

- **Erro Humano:** Em geral, a dependência da capacidade humana na detecção visual manual torna o processo propenso a erros humanos, comprometendo a eficiência e confiabilidade da identificação de cilindros em pilhas de sucatas.

Nesse contexto, a visão computacional tem desempenhado um papel fundamental no desenvolvimento de soluções automatizadas para a detecção de objetos em imagens e vídeos. O algoritmo YOLO (You Only Look Once) tem se destacado como uma ferramenta poderosa e eficiente para a detecção em tempo real de objetos em imagens, combinando alta precisão e velocidade de processamento (ALAM; ISLAM, 2019; YANG, 2023; WANG et al., 2021; HOSSAIN; ISLAM; ALMUTAIRI, 2022; YANG; ZHANG, 2023; XUE; ZHANG, 2022; GUO et al., 2022; KUMAR et al., 2023). A Figura 2 ilustra de forma clara uma compreensão de classificação, localização, detecção e segmentação de objetos únicos ou múltiplos em uma imagem.

Figura 2 – Classificação, localização, detecção e segmentação em imagem de objetos únicos e múltiplos.



Fonte: O'Reilly (2023)

A detecção proposta neste estudo envolverá modelos treinados com diferentes instâncias do algoritmo YOLO, direcionado especificamente para a identificação de cilindros pressurizados em pilhas de sucatas. Essa abordagem busca proporcionar uma solução flexível e aplicável em diversos cenários, incluindo a possibilidade de implementação em ambientes industriais.

Para o treinamento do modelo, serão utilizados conjuntos de dados representativos que abrangem a diversidade de formas, tamanhos e condições dos cilindros pressurizados encontrados em ambientes variados, a fim de simular toda complexidade de uma pilha de sucata. A preparação e processamento desses dados de treinamento serão abordados de forma detalhada, visando garantir a robustez e a generalização do modelo para diferentes contextos de aplicação.

## 1.1 Objetivos do Trabalho

Esta dissertação tem como objetivo aprofundar a aplicação do algoritmo YOLO na detecção de cilindros pressurizados em pilhas de sucatas, destacando a relevância dessa abordagem para a gestão de resíduos industriais. O estudo incluirá uma análise comparativa de diferentes modelos do YOLO, como YOLOv5, YOLOv6, YOLOv7 e YOLOv8, visando identificar suas vantagens e limitações específicas no contexto desta aplicação.

A pesquisa abrangerá uma investigação minuciosa das técnicas existentes e suas limitações, com o propósito de propor uma abordagem otimizada para enfrentar os desafios específicos associados à detecção de cilindros pressurizados em pilhas de sucatas. Além disso, serão discutidas questões relacionadas à preparação e processamento dos dados de treinamento, levando em consideração a diversidade de formas, tamanhos e condições dos cilindros encontrados em ambientes industriais reais.

Antecipa-se que os resultados obtidos contribuirão substancialmente para a melhoria das práticas de gestão de resíduos industriais, ao oferecer uma ferramenta automatizada e confiável para a detecção precisa de cilindros pressurizados. A aplicação de algoritmos de visão computacional nesse contexto não apenas promoverá a eficiência na identificação desses objetos, mas também potencializará a segurança, reduzindo o risco de acidentes e protegendo o meio ambiente, bem como os profissionais envolvidos na gestão desses resíduos.

Considerando a relevância do tema, esta dissertação almeja proporcionar uma contribuição substancial à área de visão computacional aplicada à gestão de resíduos industriais, ampliando o conhecimento científico e tecnológico nesse domínio específico. A análise comparativa dos diferentes modelos do YOLO adiciona uma camada de complexidade e sofisticação ao estudo, oferecendo *insights* valiosos para a seleção do modelo mais adequado às demandas específicas da detecção de cilindros pressurizados em pilhas de sucatas.

## 1.2 Visão Geral

Os próximos capítulos desse documento estão divididos na estrutura a seguir. No Capítulo 2 os trabalhos relacionados são detalhados. No Capítulo 3 alguns conceitos teóricos são explanados. No Capítulo 4 a metodologia da pesquisa é especificada. No Capítulo 5 os resultados obtidos são relatados e discutidos. Por fim, no Capítulo 6 o trabalho é concluído com os principais pontos e sugerindo trabalhos futuros.

## 2 Trabalhos Relacionados

Este capítulo conduzirá uma análise abrangente dos trabalhos relacionados voltados à detecção e modelagem de cilindros, um domínio crucial na área de visão computacional e processamento de imagens. A detecção precisa e a representação eficaz de cilindros desempenham um papel fundamental em aplicações diversas, desde automação industrial até reconstrução tridimensional.

Ao explorar a literatura, é possível identificar categorias distintas de técnicas comumente empregadas nestes estudos. Inicialmente, encontramos abordagens baseadas em análises geométricas e estatísticas, onde os pesquisadores se concentram na extração de características relevantes dos cilindros, muitas vezes explorando relações geométricas em ambientes específicos.

Outro conjunto de trabalhos destaca o uso de métodos hierárquicos, que transformam a detecção de cilindros 3D em problemas mais simples de detecção de círculos 2D. Essas estratégias se mostram eficazes para superar desafios computacionais, mantendo a precisão na identificação de cilindros em ambientes complexos.

Além disso, há uma categoria que adota algoritmos de aprendizado profundo, como o YOLOv5, para abordar a diversidade de tamanhos, arranjos e características distintas de cilindros. Essas técnicas destacam-se pela capacidade de aprendizado e generalização, proporcionando soluções robustas para cenários desafiadores.

Outras abordagens inovadoras incluem o emprego de Teoria dos Jogos, *frameworks* específicos (como o *slicing-tracking-detection*), e métodos que exploram a relação estrutural entre candidatos a eixo de cilindros. Essas técnicas, embora diversas, convergem para aprimorar a detecção de cilindros por meio de estratégias especializadas.

Ao categorizar e examinar essas diferentes abordagens, esta revisão busca identificar padrões, lacunas e oportunidades, proporcionando uma base sólida para contextualizar nosso trabalho proposto. Este estudo se destaca ao oferecer uma contribuição única para a detecção e modelagem de cilindros, capitalizando sobre as lições aprendidas e as inovações destacadas por essas categorias de técnicas.

### 2.1 Nuvens de pontos

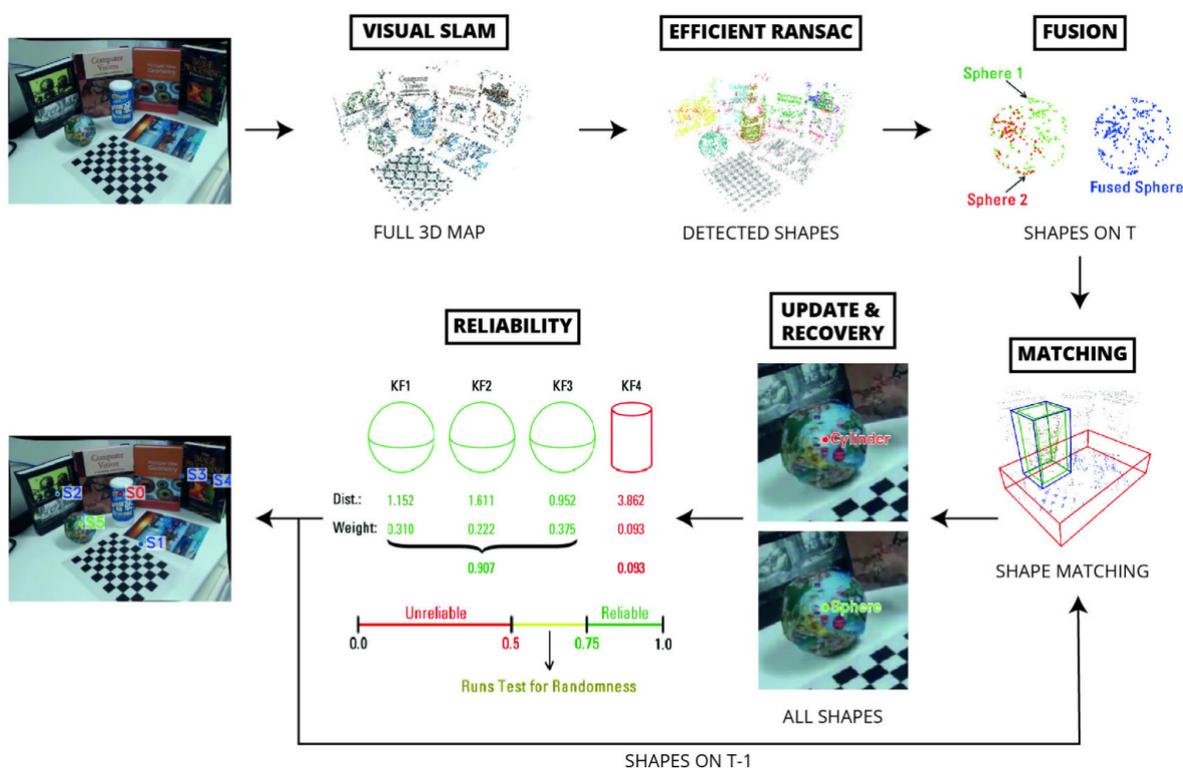
[Roberto et al. \(2019\)](#) propuseram um método de modelagem primitiva usando análises geométricas e estatísticas para nuvens de pontos esparsos que podem ser executadas em dispositivos móveis. A ideia central foi aproveitar o mapeamento incre-

mental de sistemas SLAM para analisar a relação geométrica entre a nuvem de pontos e as formas estimadas ao longo do tempo, selecionando apenas formas confiáveis.

Os autores também incorporaram uma avaliação estatística para filtrar primitivas erroneamente detectadas em estimativas instáveis. Ou seja, eles buscaram corrigir características do ambiente que foram inicialmente identificadas de maneira imprecisa ou incerta. Os resultados mostram que o método proposto melhora a precisão e consistência da detecção correta em comparação com técnicas existentes.

Embora a execução na versão móvel seja mais lenta do que na versão de *desktop*, o método teve baixo consumo de CPU, permitindo sua execução em paralelo com a técnica de SLAM visual. Além disso, foi avaliado que o uso de CPU, consumo de energia e memória RAM em dispositivos móveis não são preocupações significativas ao executar o método proposto. Este trabalho se relaciona através da modelagem de formas primitivas, podendo estas também serem de formas cilíndricas. A Figura 3 demonstra o fluxo do método proposto.

Figura 3 – Fluxo da abordagem GS-IST, onde as caixas possuem as etapas do método e abaixo delas estão a saída de cada etapa.



Fonte: Roberto et al. (2019)

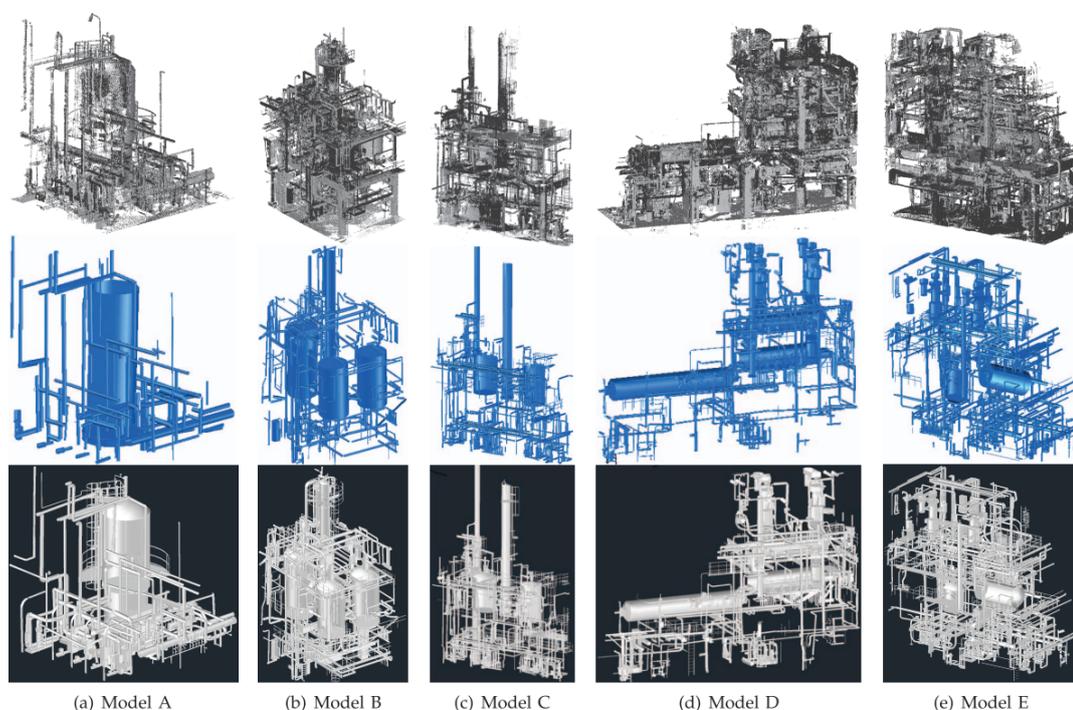
Liu et al. (2013) propuseram um método hierárquico para detectar e reconstruir cilindros em nuvens de pontos de plantas de dutos industriais. Eles superaram a complexidade computacional usando uma característica estrutural para transformar a

detecção de cilindros 3D em problemas mais simples de detecção de círculos 2D. O método é eficiente e foi testado com sucesso em aplicações industriais.

Os autores enfrentaram o desafio de determinar parâmetros de cilindros a partir de normais de pontos ruidosos, resolvendo-o ao coletar vizinhos próximos e estimar normais usando um método de ajuste de plano tangente. Eles utilizaram a transformada de *Hough* para detecção de formas geométricas e otimizaram o método RAN-SAC para melhor avaliação de hipóteses. Comparando com outros algoritmos, o método proposto foi mais rápido e preciso, graças à consideração das informações estruturais das plantas de dutos.

Nesta pesquisa foi introduzido um método eficiente para detectar e reconstruir cilindros em nuvens de pontos de plantas de dutos industriais, com melhorias nas técnicas de detecção e uma abordagem mais eficaz para lidar com dados ruidosos, resultando em maior eficiência e precisão em comparação com algoritmos anteriores. A Figura 4 ilustra cinco conjuntos de dados de pontos de grande escala escaneados e plantas de dutos reconstruídas.

Figura 4 – Experimentos na reconstrução de plantas de dutos a partir de nuvens de pontos de grande escala. Primeira linha: nuvem de pontos. Segunda linha: dutos detectados pelo método proposto. Terceira linha: plantas de dutos reconstruídas.



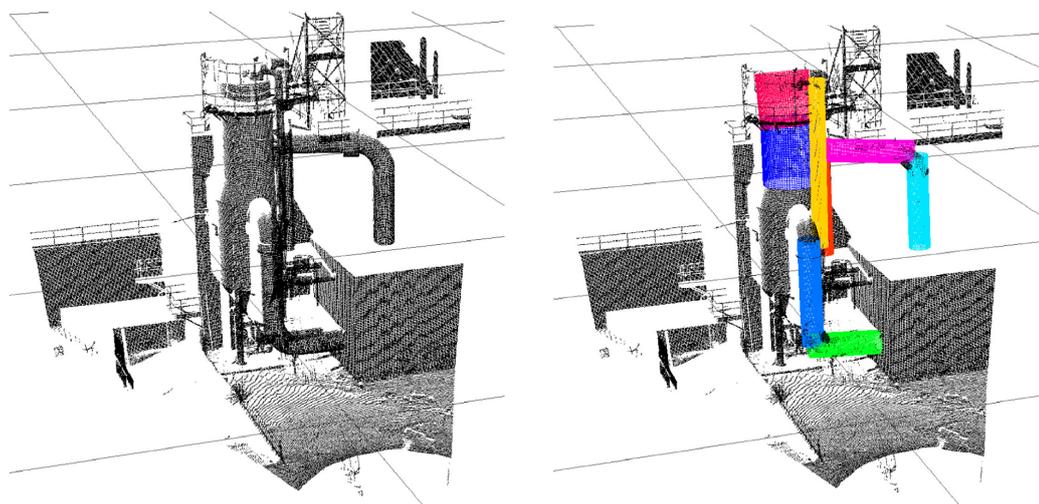
Fonte: [Liu et al. \(2013\)](#)

[Araújo e Oliveira \(2020\)](#) discutiram a detecção de cilindros baseada em conectividade em nuvens de pontos não organizadas. Os autores apresentaram um método

para detectar cilindros em dados de nuvem de pontos aproveitando informações de conectividade. A abordagem proposta utiliza uma representação baseada em gráfico da nuvem de pontos, onde os cilindros são identificados como componentes conectados no gráfico.

O método leva em consideração propriedades geométricas como normais locais de superfície e distâncias entre pontos para determinar candidatos a cilindros. Resultados experimentais demonstram a eficácia da abordagem proposta na detecção de cilindros em nuvens de pontos não organizadas. A Figura 4 ilustra o uso da técnica proposta aplicada a uma nuvem de pontos de uma planta petroquímica contendo cilindros com várias orientações, comprimentos, diâmetros e posições.

Figura 5 – Exemplo de detecção automática de cilindro usando a técnica proposta. (esquerda) Nuvem de pontos de uma planta petroquímica real. (à direita) Cilindros detectados mostrados como malhas poligonais realçadas.



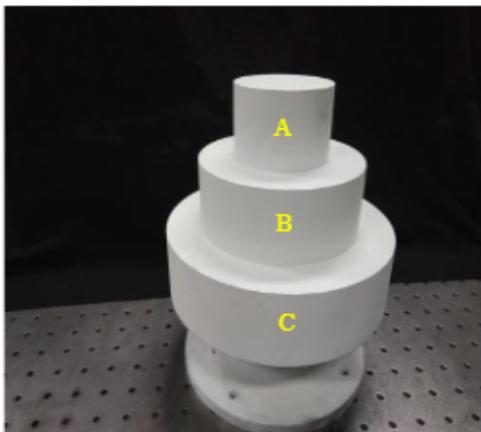
Fonte: Araújo e Oliveira (2020)

Yu, Ji e Xue (2020) apresentaram um método para detectar a pose de um cilindro usando tecnologia de medição de projeção de padrões digitais em um ambiente tridimensional. Devido a problemas como oclusão e ruído, as informações de nuvem de pontos obtidas podem ser incompletas e imprecisas. Para lidar com isso, o método proposto utiliza um processo de filtragem bilateral, conforme ilustrado na Figura 6a, para reduzir pontos discrepantes e ruídos.

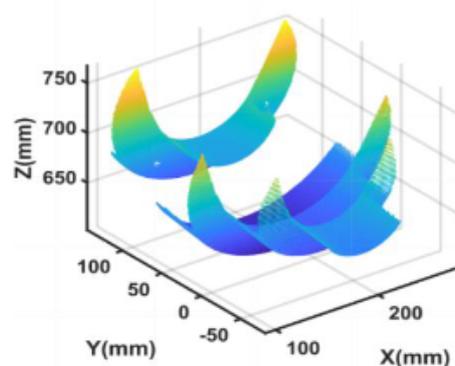
Em seguida, com base no princípio de formação de elipses em cortes do cilindro, múltiplos planos de corte equidistantes são selecionados. O algoritmo de consistência de amostragem aleatória, evidenciado na Figura 6b, é empregado para ajustar elipses em cada plano de corte, obtendo seus parâmetros. Posteriormente, esse mesmo algoritmo é utilizado para ajustar o eixo do cilindro a partir dos centros das elipses nos diferentes planos de corte, permitindo a obtenção dos parâmetros precisos do cilindro detectado. Os experimentos realizados com cilindros de diferentes materiais confirma-

ram a eficácia e alta precisão desse método na detecção de cilindros em ambientes complexos.

Figura 6 – Dados da nuvem de pontos do cilindro escalonado.



(a) Cilindro escalonado com reflexão difusa.



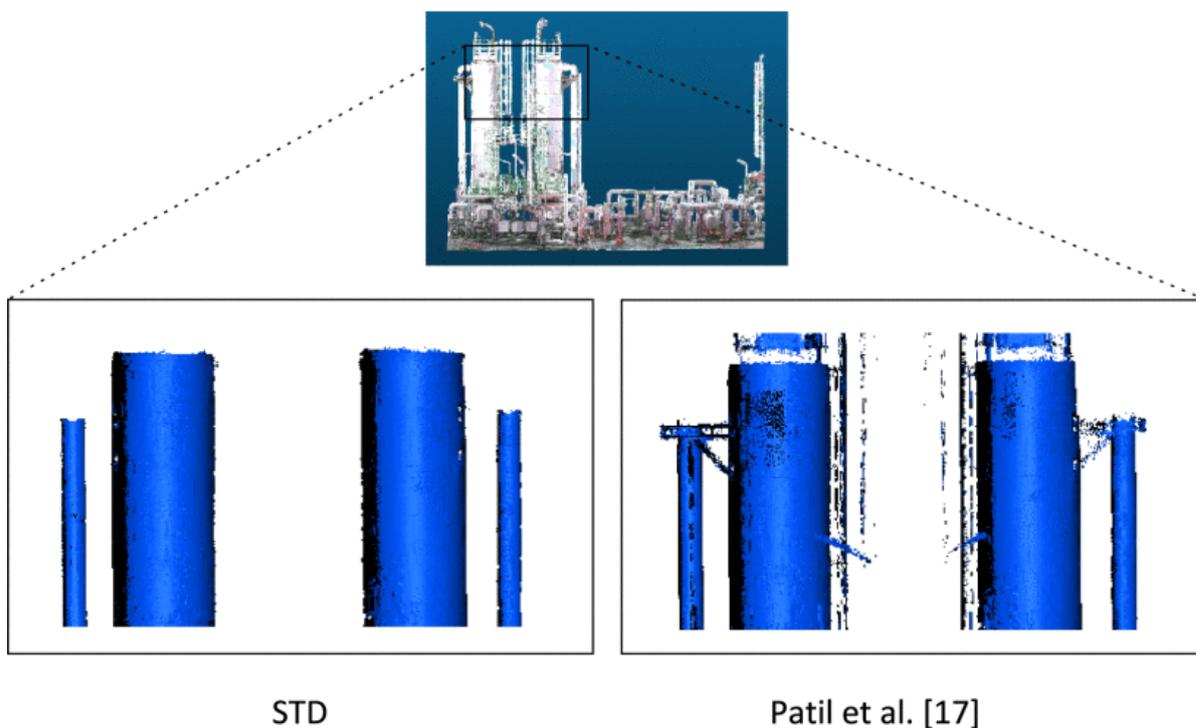
(b) Dados da nuvem de pontos 3D do cilindro escalonado com uma contagem de dados igual a 497783.

Fonte: Yu, Ji e Xue (2020)

Lu et al. (2021) desenvolveram um novo *framework* chamado *slicing-tracking-detection* (STD). O *framework* detecta vários cilindros com precisão e simultaneamente a partir de nuvens de pontos de plantas de processo complexas e de grande escala. Nele, o problema de detecção de cilindros 3D é reformulado como uma tarefa de rastreamento de elementos de cilindros baseada em um *Multi Object Tracking* (MOT). Os autores geraram fatias da nuvem de pontos de entrada e renderizaram suas sequências. Em seguida, a estrutura de um cilindro é modelada com um *Markov Decision Process* (MDP), onde o elemento é rastreado com um e o *miss tracking* é associado a propostas de elementos deste cilindro por meio de aprendizado por reforço.

Finalmente, aplicando MDP para cada cilindro, vários cilindros podem ser detectados simultaneamente e com precisão. Experimentos extensivos mostram que a estrutura STD proposta pode superar significativamente as abordagens de ponta em eficiência, precisão e robustez. A Figura 7 mostra um exemplo da execução, onde os métodos propostos superam o método baseado em área (PATIL et al., 2017) ao enfrentar os *outliers*.

Figura 7 – Uma visão aproximada do STD (esquerda) contra *outliers* em comparação com o método baseado em área de Patil et al. (2017) (direita). Os pontos azuis representam amostras de cilindros detectados. O STD é robusto contra *outliers*, enquanto o método baseado em área é sensível a *outliers*.



Fonte: Lu et al. (2021)

Ao comparar a pesquisa atual com os trabalhos relacionados que lidam com nuvens de pontos, observamos nuances distintas nas abordagens. Enquanto Roberto et al. (2019) propõem um método de modelagem primitiva em nuvens de pontos esparsos, a pesquisa atual diferencia-se ao focar na detecção específica de cilindros pressurizados em ambientes desafiadores.

Por outro lado, Liu et al. (2013) introduz uma abordagem hierárquica para detectar e reconstruir cilindros em nuvens de pontos de plantas de dutos, com ênfase em estruturas industriais. Já a pesquisa atual concentra-se na detecção de cilindros pressurizados, abordando a complexidade da oclusão parcial e arranjos específicos.

Por fim, Araújo e Oliveira (2020) discute a detecção de cilindros baseada em conectividade em nuvens de pontos não organizadas, enquanto a pesquisa atual se destaca ao aplicar diferentes versões do YOLO para detectar cilindros pressurizados em ambientes simulados de pilhas de sucatas.

Cada abordagem oferece contribuições distintas para a detecção de objetos em nuvens de pontos, destacando a diversidade de técnicas e aplicabilidades na pesquisa em visão computacional.

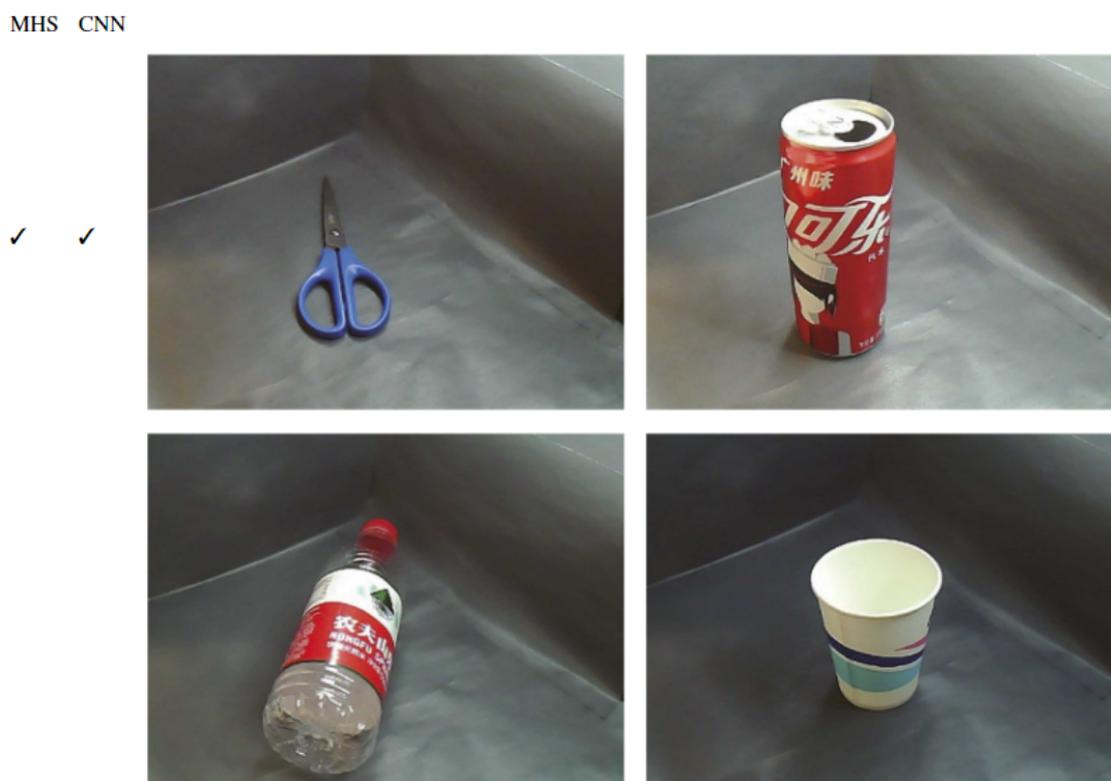
Ao comparar a pesquisa atual com o trabalho que apresenta o STD proposto por

Lu et al. (2021), observam-se diferenças significativas nas estratégias adotadas para a detecção de cilindros. Enquanto o STD reformula a detecção de cilindros como uma tarefa de rastreamento de elementos de cilindros, a pesquisa atual utiliza diferentes versões do framework YOLO para abordar a detecção específica de cilindros pressurizados em ambientes desafiadores. O STD emprega um MOT e modela a estrutura de um cilindro com um MDP, associando o *miss tracking* a propostas de elementos de cilindros. Em contraste, a abordagem atual destaca-se por sua simplicidade e eficácia na detecção de cilindros pressurizados, sem a necessidade de rastreamento complexo. Ambas as abordagens compartilham o objetivo de detecção de cilindros, mas diferem nas estratégias adotadas, refletindo a diversidade de métodos disponíveis na pesquisa em visão computacional.

## 2.2 Multilayer Hybrid Deep-learning System

Huang et al. (2018) propuseram um *Multilayer Hybrid Deep-learning System* (MHS) para classificação e reciclagem de resíduos. O sistema utilizava uma câmera de alta resolução para capturar imagens de resíduos e sensores para coletar informações adicionais de características. O MHS empregava um algoritmo baseado em CNN para extrair características de imagem e um método de perceptron de múltiplas camadas (MLP) para consolidar características de imagem e outras informações para a classificação de resíduos. O MHS é treinado e validado em comparação com itens rotulados manualmente, alcançando uma precisão geral de classificação superior a 90% em dois cenários de teste diferentes, o que supera significativamente um método baseado em CNN de referência que depende apenas de entradas de imagem. A Figura 8 ilustra objetos classificados corretamente com o MHS proposto.

Figura 8 – Exemplos de classificações corretas utilizando o MHS comparando-se com classificação de outra CNN.



Fonte: [Huang et al. \(2018\)](#)

Comparando a pesquisa atual com o *Multilayer Hybrid Deep-learning System* (MHS) proposto por [Huang et al. \(2018\)](#), percebe-se distinções nas abordagens para lidar com detecção de cilindros. Enquanto o MHS concentra-se na classificação e reciclagem de resíduos utilizando uma combinação de CNN e MLP, a pesquisa atual destaca-se pela aplicação de diferentes versões do framework YOLO para a detecção específica de cilindros pressurizados em ambientes desafiadores. O MHS utiliza câmeras de alta resolução e sensores para extrair características de imagem e informações adicionais de resíduos. Em contraste, a abordagem atual enfatiza a detecção de cilindros pressurizados em cenários complexos, abordando a oclusão parcial e arranjos específicos por meio do YOLO. Ambas as abordagens empregam técnicas de aprendizado profundo, mas são direcionadas a contextos e desafios distintos, refletindo a diversidade de métodos disponíveis na pesquisa em visão computacional.

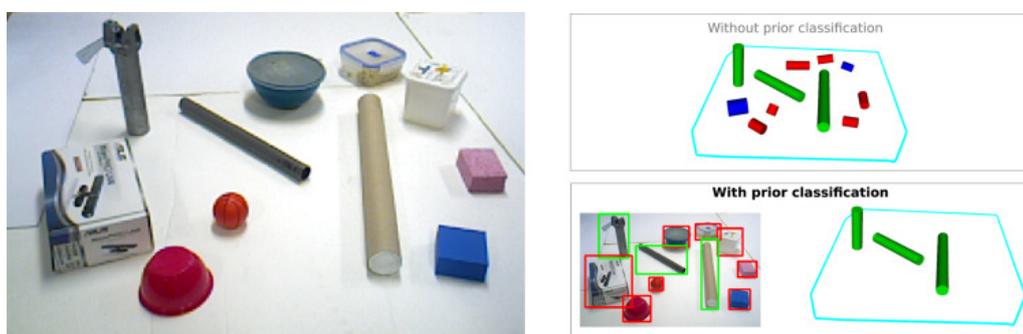
## 2.3 RGB-D

[Figueiredo et al. \(2019\)](#) desenvolveram uma solução abrangente para detectar e identificar formas cilíndricas usando câmeras RGB-D de consumo. Conforme os autores, abordagens convencionais frequentemente sofrem com *outliers* e não aprovei-

tam restrições ambientais. Para contornar isso, foram introduzidas três inovações: um esquema de votação suave com informações de curvatura, amostragem seletiva de orientações conhecidas e um classificador de cilindros baseado em aprendizado profundo para filtragem 2D. Experimentos com dados sintéticos mostram que incorporar a curvatura principal melhora a estimativa dos parâmetros dos cilindros. A combinação do classificador 2D e da votação 3D acelera e aprimora a identificação. Resultados qualitativos e quantitativos com dados reais confirmam as vantagens do método proposto. A avaliação qualitativa pode ser vista na Figura 9, onde:

- Detecção: as classificações boas e ruins estão em verde e vermelho, respectivamente;
- Identificação do parâmetro: verde representa a estimativa correta do parâmetro;
- Azul representa objetos de forma não cilíndrica corretos identificados pela qualidade da linha de base do critério de ajuste; e
- Vermelho representa estimativas erradas sem o classificador.

Figura 9 – Avaliação qualitativa da estrutura.



Fonte: [Figueiredo et al. \(2019\)](#)

Ao comparar a pesquisa atual com o trabalho de [Figueiredo et al. \(2019\)](#), que desenvolveram uma solução abrangente para detectar e identificar formas cilíndricas usando câmeras RGB-D, observam-se diferenças nas estratégias de detecção de cilindros. Enquanto a abordagem de Figueiredo et al. se destaca na utilização de câmeras RGB-D de consumo para detectar formas cilíndricas, a pesquisa atual foca-se na detecção específica de cilindros pressurizados em ambientes desafiadores, empregando diferentes versões do framework YOLO. O método de [Figueiredo et al. \(2019\)](#) incorpora informações de curvatura, amostragem seletiva de orientações conhecidas e um classificador de cilindros baseado em aprendizado profundo para a detecção 2D de formas cilíndricas. Por outro lado, a abordagem atual lida com o desafio da oclusão parcial e arranjos específicos de cilindros pressurizados, utilizando técnicas avançadas de Image Data Augmentation e refinamentos de hiperparâmetros com o YOLO.

Ambas as abordagens compartilham o objetivo de detecção de cilindros, mas diferem nas estratégias e contextos específicos em que são aplicadas, evidenciando a diversidade de métodos na pesquisa em visão computacional.

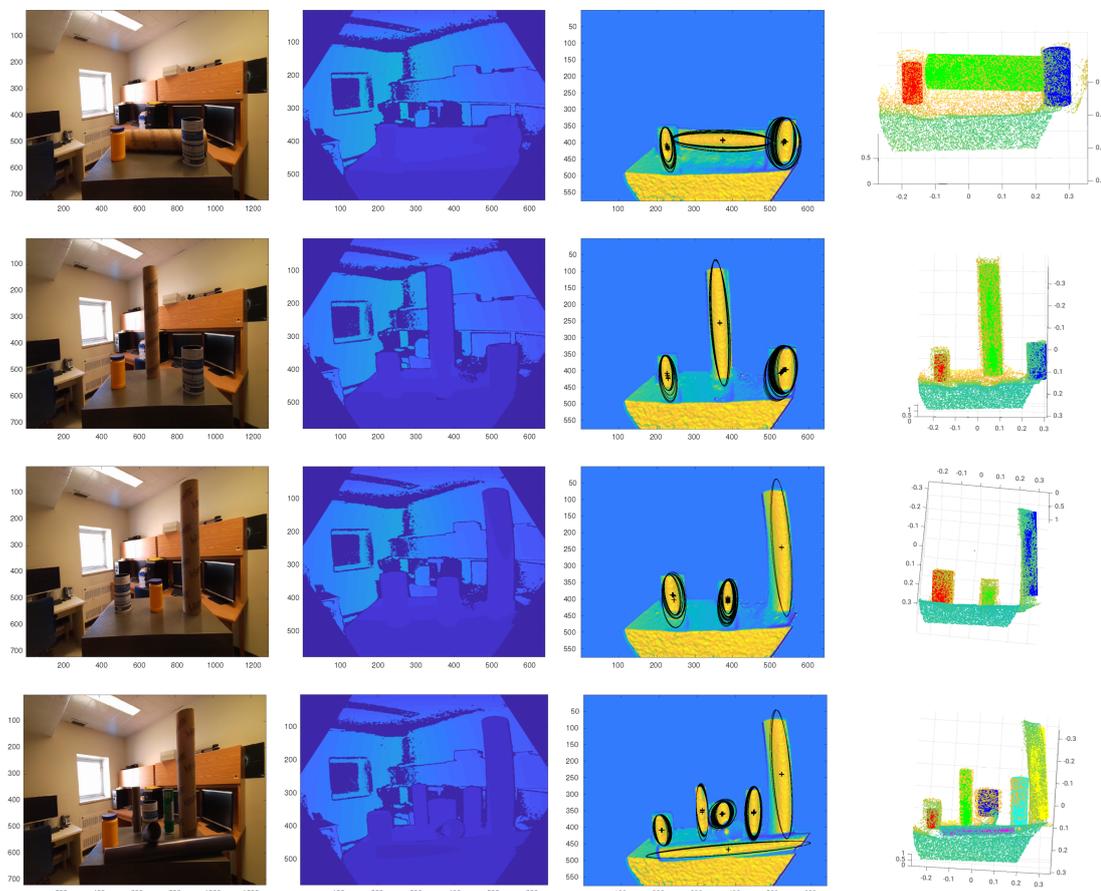
## 2.4 MSER

Moradi, Laurendeau e Gosselin (2021) propuseram um extrator de cilindros que se torna uma solução robusta dividida em três sub-módulos:

1. Um algoritmo de estimativa de vetor normal rápido e preciso a partir de imagens de profundidade bruta. Com o método de estimativa, é fornecida uma solução em forma fechada para calcular o vetor normal em cada ponto.
2. O segundo submódulo se beneficia do detector de recursos de *Maximally Stable Extremal Regions* (MSEER) para detectar simultaneamente os cilindros presentes na cena.
3. Por fim, os cilindros detectados são extraídos usando o algoritmo de extração de cilindros proposto.

Resultados quantitativos e qualitativos mostram que o algoritmo proposto supera os algoritmos de referência em cada uma das seguintes áreas: estimativa de vetor normal, detecção de cilindro e extração de cilindro. O processo completo de extração da solução é ilustrada na Figura 10.

Figura 10 – A primeira coluna: imagem RGB, a segunda coluna: imagem de profundidade, a terceira coluna: MSERs detectados, a quarta coluna: resultados finais da extração do cilindro. Para o método proposto, todos os cilindros são detectados e extraídos simultaneamente.



Fonte: Moradi, Laurendeau e Gosselin (2021)

Comparando a pesquisa atual com o trabalho de Moradi, Laurendeau e Gosselin (2021), que propôs um extrator de cilindros utilizando o algoritmo MSER, observam-se divergências nas abordagens para a detecção de cilindros. Enquanto a pesquisa de Moradi et al. divide o processo em três sub-módulos, incluindo um algoritmo de estimativa de vetor normal, detecção utilizando MSER, e um algoritmo de extração de cilindros, a pesquisa atual destaca-se pela aplicação de diferentes versões do *framework* YOLO para detectar cilindros pressurizados em ambientes desafiadores. O MSER utiliza imagens de profundidade para estimar vetores normais e detectar cilindros, enquanto a abordagem YOLO lida com o desafio da oclusão parcial e arranjos específicos de cilindros pressurizados. Ambas as abordagens buscam a detecção de cilindros, mas diferem nos métodos e contextos específicos, destacando a diversidade de técnicas na pesquisa em visão computacional.

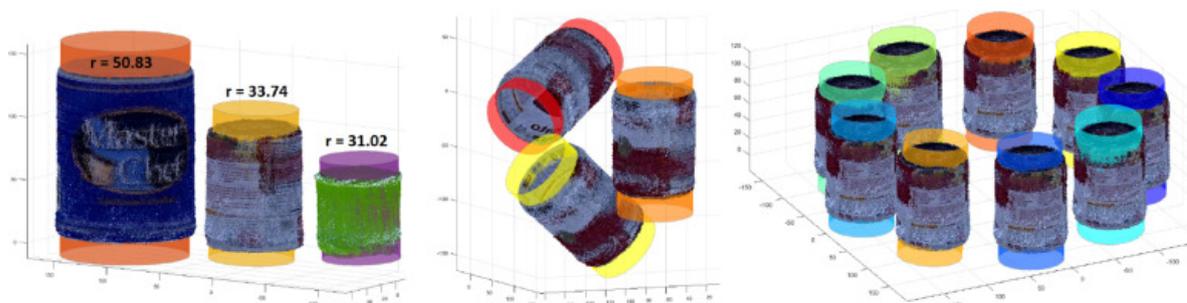
## 2.5 Teoria dos Jogos

[Bergamasco et al. \(2020\)](#) exploraram a extração de vários cilindros em uma cena, utilizando um procedimento interno baseado na Teoria dos Jogos para analisar as relações geométricas entre diferentes candidatos a eixo. A abordagem consiste em quantificar a semelhança entre dois cilindros potenciais, considerando um movimento rígido, ou seja, uma transformação geométrica que preserva as distâncias entre os pontos, alinhando os eixos desses cilindros em uma única linha reta. Esse movimento rígido é representado por um quatérnio dual unitário, um conceito matemático que descreve a orientação espacial e o deslocamento associado. Ele permite medir a distância entre os cilindros através do caminho mais curto em um espaço euclidiano tridimensional estendido (SE(3)).

O procedimento teórico fundamentado na Teoria dos Jogos é então empregado para explorar essa medida de similaridade, visando identificar grupos de características elementares (primitivas) que maximizem o acordo mútuo entre elas. O resultado desse processo evolutivo é uma distribuição de probabilidade que caracteriza conjuntos de candidatos (ou seja, eixos), sendo essa distribuição utilizada para estimar diretamente os parâmetros finais dos cilindros em questão.

A eficácia do algoritmo proposto foi demonstrada por meio de uma série de experimentos abrangentes, destacando sua capacidade de lidar robustamente com ruídos, uma vez que o processo naturalmente elimina dados inconsistentes. Em comparação com outras abordagens, este método dispensa a necessidade de calcular normais de pontos e não exige um ajuste detalhado de múltiplos parâmetros. A Figura 11 ilustra uma seleção de nuvens de pontos geradas com as formas cilíndricas extraídas, evidenciando a eficiência do método na identificação e modelagem de cilindros em cenas complexas.

Figura 11 – Exemplos qualitativos obtidos em diferentes tipos de cenas, gerados a partir de dados de varreduras reais.



Fonte: [Bergamasco et al. \(2020\)](#)

Comparando a pesquisa atual com o trabalho de [Bergamasco et al. \(2020\)](#), que explora a extração de vários cilindros utilizando a Teoria dos Jogos para analisar rela-

ções geométricas, percebem-se diferenças nas estratégias para detecção de cilindros. Enquanto o método de Bergamasco et al. emprega um procedimento interno baseado na Teoria dos Jogos para analisar a semelhança entre cilindros potenciais, a pesquisa atual destaca-se pela aplicação de diferentes versões do *framework* YOLO para detectar cilindros pressurizados em ambientes desafiadores.

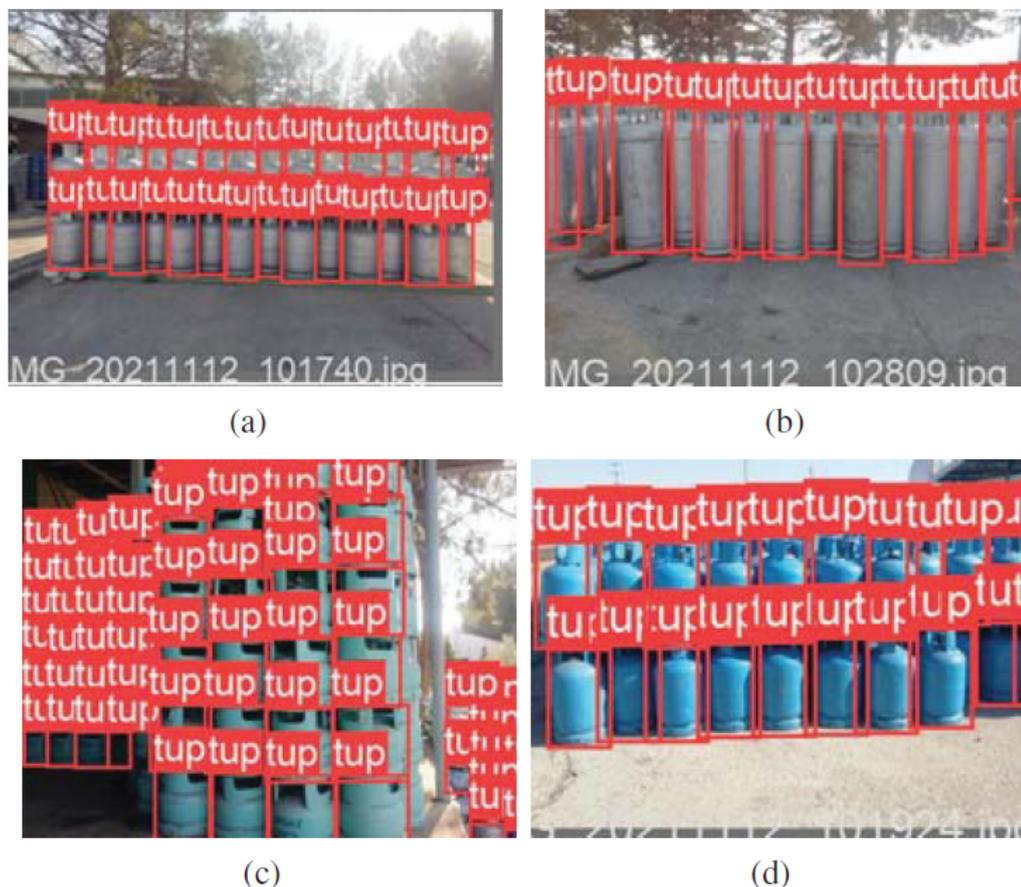
A abordagem de Bergamasco et al. (2020) utiliza um quatérnio dual unitário para medir a distância entre cilindros em um espaço tridimensional estendido, enquanto a pesquisa atual lida com o desafio da oclusão parcial e arranjos específicos de cilindros pressurizados através de técnicas avançadas de *Image Data Augmentation* e refinamentos de hiperparâmetros com o YOLO. Ambas as abordagens têm como objetivo a detecção de cilindros, mas divergem nos métodos e contextos específicos, ressaltando a diversidade de técnicas na pesquisa em visão computacional.

## 2.6 YOLOv5

Albayrak e Ozerdem (2022) propuseram uma investigação para abordar a detecção de cilindros de gás com características distintas de cor e forma. Para este fim, adotou-se o método de aprendizado profundo YOLOv5. Em locais de armazenamento ou instalações de envase, a contagem precisa de cilindros frequentemente enfrenta dificuldades devido à diversidade de tamanhos, arranjos e à grande quantidade de cilindros presentes. Visando preencher essa lacuna, o estudo envolveu a construção de um conjunto de dados que engloba diferentes tipos de cilindros encontrados em instalações de envase de gás.

Os resultados obtidos revelaram que o algoritmo YOLOv5 exibe uma notável taxa de sucesso de 96,16% na detecção de cilindros de gás com variações significativas em termos de cor e forma. Além da eficiência na detecção, a metodologia também demonstrou sólido desempenho em diversas métricas de avaliação, como precisão, sensibilidade e interseção de caixas delimitadoras. Este trabalho utilizou apenas 46 imagens, dividindo o conjunto de treinamento em 80% para treino e 20% para validação. O que levaria à validação com cerca de 9 imagens apenas. A Figura 12 mostra o teste de detecções de cilindros com o modelo treinado.

Figura 12 – Exemplos de resultados obtidos a partir das imagens incluídas nos dados de teste.



Fonte: Albayrak e Ozerdem (2022)

Comparando a pesquisa atual com o trabalho de [Albayrak e Ozerdem \(2022\)](#), que investigou a detecção de cilindros de gás utilizando o método de aprendizado profundo YOLOv5, destaca-se que ambos os estudos utilizam o YOLOv5 para a detecção de cilindros, evidenciando a eficácia dessa arquitetura em tarefas específicas. No entanto, uma contribuição única da pesquisa atual é a análise comparativa realizada com treinamentos em diferentes versões do YOLO, incluindo o YOLOv5, YOLOv6, YOLOv7 e YOLOv8.

Essa abordagem permitiu uma compreensão mais aprofundada das nuances de desempenho entre as diferentes iterações do YOLO, fornecendo insights valiosos sobre a evolução e as melhorias ao longo dessas versões. Enquanto o estudo de [Albayrak e Ozerdem \(2022\)](#) concentra-se na detecção de cilindros de gás com características distintas de cor e forma, a pesquisa atual direciona seu foco para a detecção específica de cilindros pressurizados em ambientes desafiadores, simulando pilhas de sucatas. Ambos os estudos demonstram a versatilidade do YOLOv5, mas diferem nos contextos e objetivos específicos de detecção de cilindros, evidenciando a adaptabili-

dade desta arquitetura em diversas aplicações de visão computacional.

## 3 Fundamentação Teórica

Neste capítulo é apresentada uma explanação sobre os conteúdos que serão abordados: Gestão de Resíduos Industriais, Visão Computacional, Reconhecimento de Padrões, Detecção de Objetos e YOLO. Na Seção 3.1, são apresentados os conceitos sobre a gestão dos resíduos industriais. Na Seção 3.2, define-se visão computacional, e demonstra-se o conceito de reconhecimento de padrões, detecção de objetos, redes neurais convolucionais e informações acerca do *framework* YOLO.

### 3.1 Gestão de Resíduos Industriais

A gestão de resíduos industriais refere-se ao manejo sistemático e responsável, tratamento e disposição dos resíduos gerados pelas atividades industriais. Envolve a implementação de estratégias e práticas para minimizar os impactos ambientais e de saúde negativos dos resíduos industriais (HOVEIDI et al., 2013).

Os resíduos industriais podem incluir vários tipos de materiais, como resíduos sólidos, resíduos líquidos e resíduos perigosos, que são produzidos pelas indústrias durante seus processos de fabricação (HOVEIDI et al., 2013). Os resíduos podem vir de uma ampla variedade de indústrias, incluindo manufatura, mineração, produção de energia e farmacêutica (HOVEIDI et al., 2013; LIS; NOWACKI, 2022; AZZAHRA; SAPTARINI, 2021).

A gestão adequada de resíduos industriais é crucial para o desenvolvimento sustentável e a proteção ambiental. Ela ajuda a prevenir a poluição, reduzir a depleção de recursos e minimizar os riscos para a saúde humana e os ecossistemas (HOVEIDI et al., 2013). Práticas eficazes de gestão de resíduos incluem redução e prevenção de resíduos, reciclagem e reutilização, tratamento e disposição, e conformidade regulatória (HOVEIDI et al., 2013; OMARA et al., 2019).

Um dos principais desafios na gestão de resíduos industriais é a caracterização e classificação adequadas dos materiais de resíduos. Isso envolve identificar a composição, quantidade e potenciais riscos dos resíduos para determinar os métodos de gestão mais adequados (OMARA et al., 2019). A caracterização de resíduos auxilia no desenvolvimento de planos de gestão de resíduos, criação de instalações de tratamento e garantia de conformidade com regulamentações (OMARA et al., 2019).

A implementação de sistemas eficazes de gestão de resíduos industriais pode trazer benefícios tanto ambientais quanto econômicos. Práticas adequadas de gestão de resíduos podem reduzir os custos associados à disposição de resíduos, melhorar

a eficiência de recursos e criar oportunidades para a reciclagem e recuperação de materiais valiosos (OMARA et al., 2019). Além disso, pode melhorar a reputação das indústrias, demonstrando seu compromisso com a sustentabilidade e a responsabilidade ambiental (HASSAN; DILAWAR; AKBAR, 2021).

Em conclusão, a gestão de resíduos industriais é um aspecto crucial do desenvolvimento sustentável e da proteção ambiental. Ela envolve o manejo adequado, tratamento e disposição de resíduos gerados pelas atividades industriais. Práticas eficazes de gestão de resíduos ajudam a prevenir a poluição, conservar recursos e proteger a saúde humana e os ecossistemas. Regulamentações e diretrizes desempenham um papel crucial em garantir que as indústrias cumpram práticas adequadas de gestão de resíduos. Ao implementar sistemas eficazes de gestão de resíduos, as indústrias podem obter benefícios tanto ambientais quanto econômicos.

## 3.2 Visão Computacional

Visão computacional (VC) é um campo de estudo que se concentra em capacitar computadores a compreender e interpretar informações visuais de imagens ou vídeos. Envolve o desenvolvimento de algoritmos e técnicas que permitem que os computadores extraiam informações significativas a partir de dados visuais e tomem decisões ou ações com base nessas informações (SZELISKI, 2011).

Ela tem uma ampla gama de aplicações em diversos domínios. No campo da computação gráfica, técnicas de visão computacional são usadas para criar modelos baseados em imagens de objetos do mundo real, gerar efeitos visuais e mesclar imagens do mundo real usando técnicas de fotografia computacional (SZELISKI, 2011). Na indústria de manufatura, ela é aplicada em tarefas como inspeção industrial e montagem, onde desempenha um papel crucial na promoção da informatização, digitalização e inteligência de sistemas de manufatura (ZHOU; ZHANG; KONZ, 2022). A VC também é usada em sistemas de transporte, como sistemas de pesagem em movimento de veículos, onde é empregada para medir parâmetros de deformação de pneus e reconhecer marcações na lateral dos pneus (FENG; LEUNG, 2023). Além disso, existem aplicações no campo médico, especialmente em pesquisas sobre câncer, onde é usada para análise computacional de imagens e desenvolvimento de ferramentas de diagnóstico (ARAFAT; REYES-ALDASORO, 2022).

A aprendizagem contínua é um aspecto importante da pesquisa em VC. As técnicas de aprendizado contínuo são usadas para adaptar modelos de VC a novos dados e tarefas ao longo do tempo, sem esquecer informações previamente aprendidas (QU et al., 2021). Essas técnicas incluem regularização, destilação de conhecimento, métodos baseados em memória, repetição generativa e isolamento de parâmetros (QU et

al., 2021). A aprendizagem contínua em VC ainda é uma área ativa de pesquisa, com investigações em andamento sobre suas potenciais aplicações em diferentes subáreas de VC (QU et al., 2021).

Ballard e Brown (1982) definem VC como um campo multidisciplinar que tem como objetivo pesquisar a capacidade de computadores de ganhar um nível humano de entendimento de imagens digitais ou vídeos.

VC é a área de estudo interdisciplinar que simula e tenta reproduzir a incrível capacidade da visão humana nas máquinas, consistindo em captar as imagens, melhorá-las, segmentar, detectar pontos de interesse, extrair informações e finalmente realizar relações entre imagens (BACKES; JUNIOR, 2016).

Conforme Santos et al. (2018), o que reforça o aspecto multidisciplinar da VC é a ligação íntima com várias outras ciências, algumas destas sendo:

- Processamento de Sinais;
- Reconhecimento de Padrões;
- Processamento de Imagens;
- Inteligência Artificial;
- Matemática; e
- Física.

Em geral, a visão computacional é um campo em rápida evolução, com diversas aplicações e esforços contínuos de pesquisa. Ela desempenha um papel crucial ao capacitar computadores a entender e interpretar informações visuais, e seu impacto pode ser observado em diversos domínios, como computação gráfica, manufatura, transporte e saúde.

### 3.2.1 Reconhecimento de Padrões

Duda, Hart e Stork (2012) descrevem reconhecimento de padrões como o ato de receber dados brutos e tomar uma ação baseada na “categoria” do padrão.

O reconhecimento de padrões em visão computacional refere-se ao processo de identificar e categorizar padrões ou objetos em dados visuais, como imagens ou vídeos. Envolve o desenvolvimento de algoritmos e técnicas que permitem que os computadores reconheçam e classifiquem automaticamente padrões com base em suas características visuais (JAIN; DUIN; MAO, 2000).

O objetivo principal do reconhecimento de padrões é realizar classificação supervisionada ou não supervisionada, onde os padrões são atribuídos a classes predefinidas ou agrupados com base em suas similaridades (JAIN; DUIN; MAO, 2000). A abordagem estatística tem sido amplamente utilizada no reconhecimento de padrões,

com extensa pesquisa e aplicações práticas (JAIN; DUIN; MAO, 2000). Nos últimos anos, as técnicas de redes neurais e os métodos da teoria do aprendizado estatístico também têm chamado a atenção no campo (JAIN; DUIN; MAO, 2000).

Um sistema de reconhecimento de padrões envolve várias etapas, incluindo a definição de classes de padrões, o ambiente de percepção, a representação de padrões, a extração e seleção de características, a análise de agrupamento, o projeto e aprendizado de classificadores, a seleção de amostras de treinamento e teste e a avaliação de desempenho (JAIN; DUIN; MAO, 2000). Cada etapa desempenha um papel crucial no processo geral de reconhecimento e requer consideração e projeto cuidadosos.

Apesar de décadas de pesquisa e desenvolvimento em reconhecimento de padrões, o desafio de reconhecer padrões complexos com orientação, localização e escala arbitrárias continua sem solução (JAIN; DUIN; MAO, 2000). No entanto, as técnicas de reconhecimento de padrões continuam a evoluir e encontrar aplicações em diversos domínios. Por exemplo, o reconhecimento de padrões é usado em mineração de dados, busca na web, recuperação de dados multimídia, reconhecimento facial e reconhecimento de escrita cursiva (JAIN; DUIN; MAO, 2000).

No campo da VC, as técnicas de reconhecimento de padrões são essenciais para tarefas como detecção de objetos, reconhecimento de objetos, segmentação de imagens e compreensão de cenas. Essas técnicas permitem que os computadores analisem dados visuais e tomem decisões ou ações com base nos padrões reconhecidos (SZELISKI, 2011).

Castro e Prado (2002) enfatizam que o reconhecimento de padrões é uma habilidade desenvolvida naturalmente nos seres humanos e em alguns animais para possibilitar as tomadas de decisões, pois os padrões são os meios de interpretação que elaboram ações e decisões. A exposição a estímulos e exemplos facilita o indivíduo a incorporar tal aprendizado que passa e evoluir com as experiências, proporcionando facilidade no reconhecimento de padrões.

Segundo Marengoni e Stringhini (2009), reconhecer significa conhecer de novo e para isso é preciso algum conhecimento prévio sobre o objeto a ser reconhecido, implicando então no armazenamento deste conhecimento. Sendo uma das principais áreas de VC, o reconhecimento de objetos está relacionado diretamente com o reconhecimento de padrões.

No geral, o reconhecimento de padrões em visão computacional é um campo dinâmico e desafiador que visa desenvolver técnicas robustas e eficientes para reconhecer e classificar automaticamente padrões em dados visuais. Abrange várias abordagens, incluindo métodos estatísticos e técnicas de redes neurais, e encontra

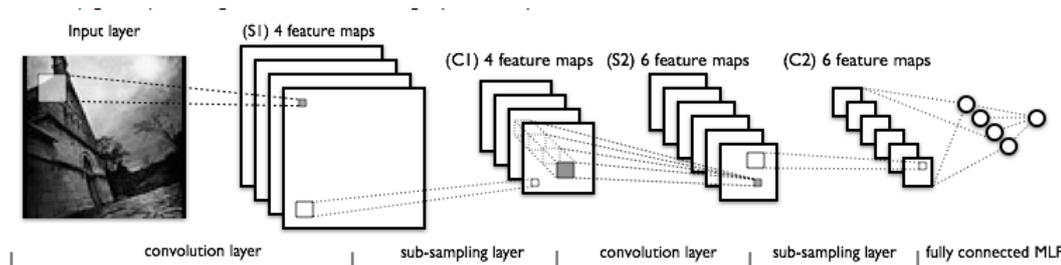
aplicações em diversos domínios.

### 3.2.2 Redes Neurais Convolucionais

A primeira *Convolutional Neural Network* (CNN) foi proposta por [LeCun et al. \(1990\)](#), intitulada como LeNet, era um método de reconhecimento de letras manuscritas em classificação invariante para códigos de endereços postais (CEP) em escaneamento de envelopes nos correios dos Estados Unidos, sendo assim uma das primeiras CNNs utilizadas em aplicação comercial.

A CNN pode executar tarefas de classificação e é composta de múltiplas camadas, cada uma calcula transformações convolucionais ([LECUN et al., 1990](#)). A Figura 13 ilustra o modelo gráfico de uma LeNet, sendo as camadas iniciais responsáveis pelas convoluções e reduções e as últimas camadas são totalmente conectadas, correspondendo a uma rede neural tradicional.

Figura 13 – Modelo gráfico de uma *LeNet*.



Fonte: [Lab \(2018\)](#)

Segundo [Upreti \(2022\)](#), as redes neurais convolucionais (CNNs) são uma classe de redes neurais artificiais que revolucionaram o campo da visão computacional. Elas são especificamente projetadas para processar e analisar dados visuais, como imagens e vídeos.

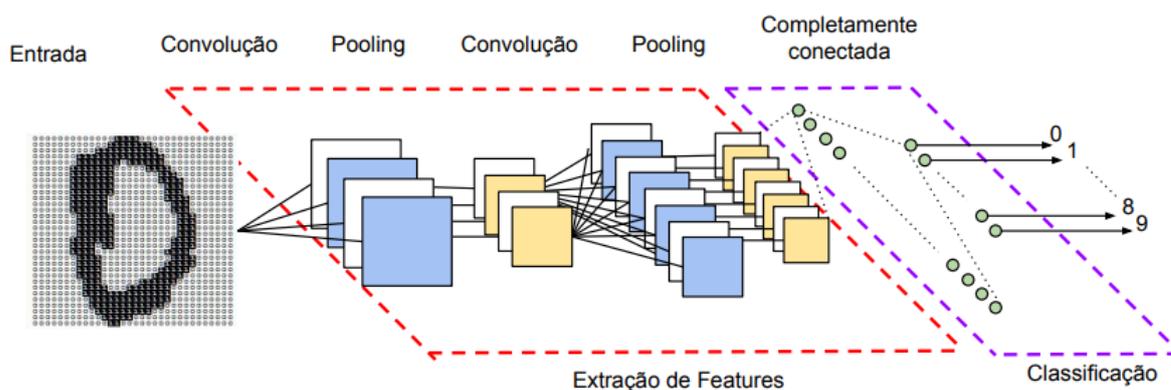
A principal característica das CNNs é sua capacidade de aprender automaticamente representações hierárquicas de características visuais a partir de dados de entrada brutos. Isso é conseguido através do uso de camadas convolucionais, que aplicam filtros ou kernels aos dados de entrada para extrair características locais ([SZEGEDY et al., 2015](#)). Esses filtros capturam diferentes padrões e características em várias escalas espaciais, permitindo que a rede aprenda representações complexas dos dados de entrada ([SZEGEDY et al., 2015](#)).

As CNNs também incorporam outras camadas, como camadas de pooling e camadas totalmente conectadas. As camadas de pooling reduzem o tamanho dos mapas de recursos, reduzindo suas dimensões espaciais, mas mantendo as informações mais relevantes ([MUSTAFAYEV; AZIMOV, 2021](#)). As camadas totalmente conectadas

tadas conectam todos os neurônios da camada anterior à camada seguinte, permitindo que a rede faça previsões ou classificações com base nas características aprendidas (UPRETI, 2022).

Vargas, Paes e Vasconcelos (2016) definem que a CNN foi inspirada no processo biológico de processamento de dados visuais, como uma variação de Perceptrons de Múltiplas Camadas (MLP, do inglês *Multi Layer Perceptrons*), uma Rede Neural de múltiplas camadas, tendo a capacidade de aplicar filtros em dados visuais, mantendo a relação de vizinhança entre os *pixels* de uma imagem ao longo do processamento da rede. Os autores ainda afirmam a existência de múltiplas partes em uma CNN, cada uma com funções diferentes. A Figura 14 demonstra as conexões e funções entre as partes de uma CNN.

Figura 14 – Estrutura das camadas de uma Rede Neural Convolutacional.



Fonte: Vargas, Paes e Vasconcelos (2016)

Na Figura 14 pode-se observar a existência de várias camadas em uma CNN, cada uma responsável por uma função específica na propagação do sinal de entrada. Segundo Araújo et al. (2017), em uma CNN existem três tipos de camadas: convolucionais, de *pooling* e totalmente conectadas. As camadas convolucionais extraem os atributos dos volumes de entrada. As camadas de *pooling* reduzem a dimensionalidade do volume restante após as camadas convolucionais e ajudam a tornar a representação invariante a pequenas translações na entrada. A última camada, denominada totalmente conectadas, propaga o sinal por meio da multiplicação ponto a ponto e o uso de uma função de ativação. Por fim, a saída de uma CNN são as probabilidades de um dado de entrada pertencer a alguma das classes, nas quais a rede foi treinada.

Uma das rupturas na arquitetura CNN foi a introdução do modelo AlexNet, que obteve melhorias significativas na precisão da classificação de imagens no Image-

Net Large-Scale Visual Recognition Challenge (ILSVRC)<sup>1</sup> em 2012 (MUNTASA et al., 2022). Desde então, várias arquiteturas CNN foram desenvolvidas, incluindo VGGNet, GoogLeNet e ResNet, cada uma com seu próprio design e características de desempenho (HE et al., 2016; SZEGEDY et al., 2015).

As CNNs têm sido amplamente aplicadas em várias tarefas de visão computacional, como classificação de imagens, detecção de objetos, segmentação de imagens e geração de imagens. Elas demonstraram um desempenho notável nessas tarefas, superando os métodos tradicionais de visão computacional em termos de precisão e eficiência (YANG; ANJIE, 2021).

O sucesso das CNNs pode ser atribuído à sua capacidade de aprender e extrair automaticamente características relevantes de dados visuais, eliminando a necessidade de engenharia de recursos manuais. Ao aproveitar grandes conjuntos de dados e recursos computacionais poderosos, as CNNs podem aprender representações complexas e generalizar bem para dados não vistos (SIMONYAN; ZISSERMAN, 2014).

Em resumo, CNNs são uma classe de redes neurais artificiais que se destacam no processamento e análise de dados visuais. Elas revolucionaram as tarefas de visão computacional ao aprender automaticamente representações hierárquicas de características visuais. As CNNs alcançaram um desempenho de ponta em diversas aplicações e continuam sendo um foco de pesquisa e desenvolvimento no campo da visão computacional.

### 3.2.2.1 Camada Convolutiva

Araújo et al. (2017) descrevem a camada convolutiva como um conjunto de filtros que recebem um arranjo 3D, também chamado de volume. Cada filtro possui dimensão reduzida, porém ele se estende por toda a profundidade do volume de entrada. Por exemplo, se a imagem for colorida, então ela possui 3 canais e o filtro da primeira camada convolutiva terá tamanho  $5 \times 5 \times 3$  (5 pixels de altura e largura e profundidade igual a 3).

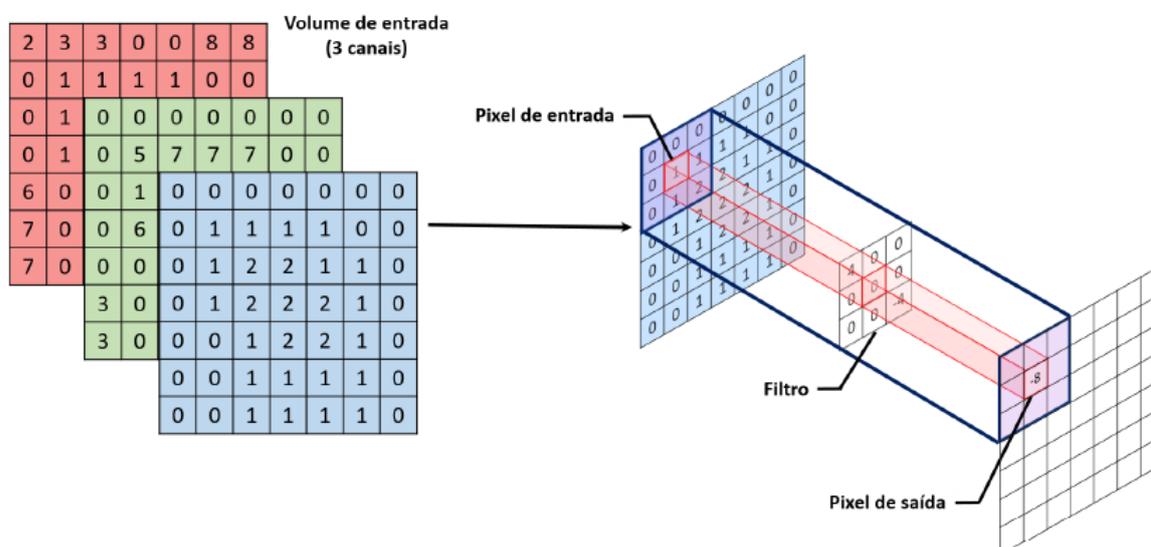
Karpathy (2017) completa que, de forma automática, os filtros são ajustados durante o processo de treinamento da rede para que sejam ativados durante a presença de características relevantes identificadas no volume de entrada, como orientação de bordas ou manchas de cores. A relevância é avaliada de tal forma que os resultados sejam otimizados em função de um conjunto de amostras previamente classificadas (ARAÚJO et al., 2017).

A Figura 15 demonstra a operação conhecida como convolução, onde acontece o somatório do produto ponto a ponto entre os valores de um filtro e cada posição do

<sup>1</sup> ILSVRC2012: <https://www.image-net.org/challenges/LSVRC/2012/>

volume de entrada.

Figura 15 – Ilustração da convolução entre um filtro 3x3 e o volume de entrada.



Fonte: Araújo et al. (2017)

Existem três parâmetros que controlam o tamanho do volume resultante da camada convolucional: profundidade (*depth*), passo (*stride*) e *zero-padding* (KARN, 2016). Araújo et al. (2017) descrevem estes parâmetros da seguinte forma:

- A **profundidade** do volume resultante é igual ao número de filtros utilizados. Cada um desses filtros será responsável por extrair características diferentes no volume de entrada;
- O parâmetro **passo** especifica o tamanho do salto na operação de convolução. Este salto define quantos *pixels* o filtro se desloca para o lado nos *pixels* da imagem; e
- A operação de **zero-padding** consiste em preencher com zeros a borda do volume de entrada, sendo vantajoso pelo poder de controlar a altura e largura do volume resultante.

Os valores resultantes após a operação de convolução passam por uma função de ativação, sendo a mais comum a função *ReLU* (*Rectified Linear Units*) (KRIZHEVSKY; SUTSKEVER; HINTON, 2012).

As funções de ativação determinam se uma informação pode ser propagada ou não pela rede, com base na informação gerada na camada anterior e multiplicada pelos pesos definidos durante o treinamento (OLIVEIRA, 2016). Geralmente esses pesos são definidos em algoritmos de *backpropagation* e representam o conhecimento obtido pela rede no decorrer do treinamento (ALPAYDIN, 2014).

Já as funções de ativação *ReLU* aplicam funções de ativação de não-saturação (OLIVEIRA, 2016). Ou seja, aumentam a não-linearidade da função de decisão da rede, sem afetar os campos receptivos. A Equação 3.1 demonstra a função *ReLU*, onde  $x$  é o valor de entrada e o valor propagado será  $x$  se ele for maior que 0, ou 0 caso contrário (SPOLTI, 2018).

$$f(x) = \max(0, x) \quad (3.1)$$

### 3.2.2.2 Camada de *Pooling*

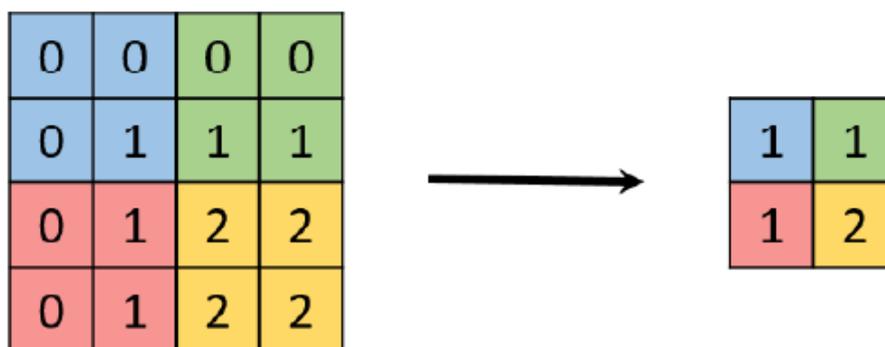
Geralmente, após uma camada convolucional existe uma camada de *pooling* que tem como objetivo reduzir progressivamente a dimensão do volume de entrada, consequentemente a redução diminui o custo computacional da rede e evita *overfitting* (KARPATHY, 2017).

O *overfitting* ocorre quando a rede obtém bons resultados apenas para o conjunto de treino e perde a capacidade de generalizar esse desempenho no conjunto de validação (MARIM, 2019), ou seja, a rede fica especializada em classificar os objetos contidos nos dados utilizados no treinamento da rede, mas mostra-se ineficaz para prever novos resultados (SPOLTI, 2018).

Nesta operação, os valores pertencentes a uma determinada região de atributos, gerados pelas camadas convolucionais, são substituídos por alguma métrica dessa região. Goodfellow, Bengio e Courville (2016) destacam que a forma mais comum de *pooling* consiste em substituir os valores de uma região pelo valor máximo, também conhecida como *max pooling*, sendo útil para eliminar valores desprezíveis para diminuir a dimensão e acelerar a computação (ARAÚJO et al., 2017).

A Figura 16 ilustra a aplicação de *max pooling* em uma imagem 4x4 utilizando um filtro 2x2. Além de reduzir o tamanho da imagem, reduz o processamento para as próximas camadas, consequentemente (ARAÚJO et al., 2017).

Figura 16 – Aplicação de *max pooling* em uma imagem 4x4 utilizando um filtro 2x2.



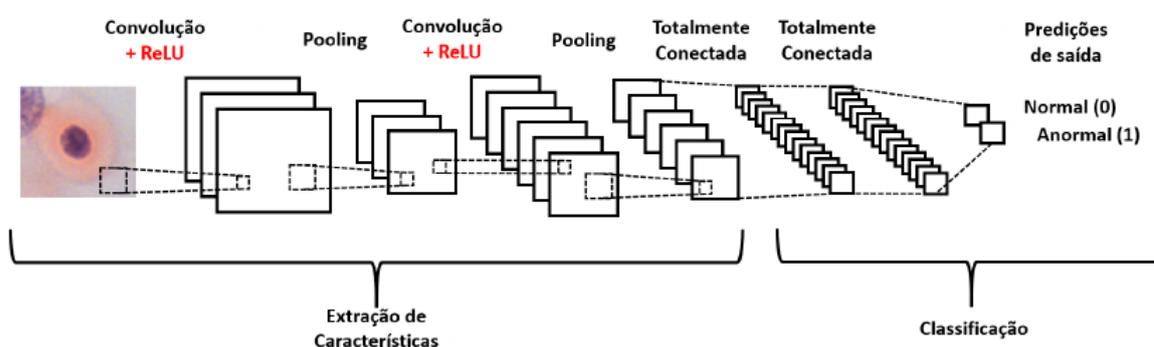
Fonte: Araújo et al. (2017)

### 3.2.2.3 Camada Totalmente Conectada

Conforme Araújo et al. (2017), a saída das camadas convolucionais e de *pooling* representam as características extraídas da imagem de entrada, tendo como objetivo serem utilizadas para classificar a imagem em uma classe pré-determinada.

As camadas totalmente conectadas são exatamente iguais a uma rede neural artificial convencional (HAYKIN et al., 2009) que usa a função de ativação *softmax* (BISHOP, 2006) na última camada (de saída). A Figura 17 ilustra a extração de características de uma imagem por uma CNN e sua posterior classificação.

Figura 17 – Ilustração da extração de características de uma imagem por uma CNN e sua posterior classificação.



Fonte: Araújo et al. (2017)

Pode-se observar na Figura 17, as saídas das etapas aplicadas inicialmente (Convolução e *Pooling*) sendo aplicadas na camada Totalmente Conectada com o objetivo de utilizar essas saídas (características) para classificar a imagem em uma classe pré-determinada (ARAÚJO et al., 2017).

A função de ativação *softmax* é utilizada para calcular a probabilidade de uma

classe num problema multivariado (PIASSA, 2019), ou seja, a rede tem como saída valores arbitrários onde o maior valor representa a classe a qual o objeto tende a pertencer (ARAÚJO, 2019). A Equação 3.2 representa a fórmula para obter-se os resultados da *softmax*, onde  $y$  é o valor de saída.

$$\text{Softmax}(y_i) = \frac{e^{y_i}}{\sum_j e^{y_j}} \quad (3.2)$$

### 3.2.3 Detecção de Objetos

A detecção de objetos em visão computacional refere-se à tarefa de identificar e localizar objetos de interesse dentro de uma imagem ou vídeo. Envolve o desenvolvimento de algoritmos e técnicas que possam detectar e classificar automaticamente objetos em dados visuais (REN et al., 2017).

É uma tecnologia importante que possui diversas aplicações em diferentes áreas, como reconhecimento de vestuário de moda, recomendação de moda, busca online, vigilância e recuperação de imagens (LEE; LIN, 2021).

A detecção de objetos desempenha um papel crucial em várias aplicações de visão computacional, incluindo condução autônoma, vigilância, robótica e compreensão de imagens. Isso permite que as máquinas percebam e entendam seu ambiente visual, permitindo que tomem decisões informadas e ajam adequadamente com base nos objetos detectados (REN et al., 2017).

Segundo Kevin et al. (2019), o objetivo da detecção de objetos é identificar com precisão e localizar objetos de interesse em uma imagem ou vídeo. Isso geralmente envolve gerar previsões de caixas delimitadoras ao redor dos objetos e atribuir rótulos de classe a eles. Métodos de detecção de objetos frequentemente empregam técnicas de proposta de região para gerar localizações potenciais de objetos, que são posteriormente refinadas e classificadas pela rede de detecção.

A detecção de objetos é uma tarefa fundamental em visão computacional e serve como um bloco de construção para várias aplicações de nível superior. Por exemplo, na condução autônoma, a detecção de objetos é crucial para identificar e rastrear pedestres, veículos e outros obstáculos na estrada (OSMAN; SHEHATA, 2022). Em sistemas de vigilância, a detecção de objetos permite a detecção e rastreamento de atividades ou objetos suspeitos (GARRIDO-VALENZUELA; CRANENBURGH; CATS, 2022). Na recuperação de imagens, a detecção de objetos pode ser usada para indexar e pesquisar imagens com base na presença de objetos específicos (FARHANZA, 2020).

Para Kevin et al. (2019), a detecção de objetos envolve o desenvolvimento de algoritmos e técnicas que podem detectar e classificar automaticamente objetos em da-

dos visuais, sendo uma tarefa desafiadora devido à variabilidade na aparência, escala, orientação e interferência do fundo dos objetos.

Tradicionalmente, os algoritmos de detecção de objetos dependiam de características e classificadores criados manualmente para identificar objetos. No entanto, com os avanços na aprendizagem profunda, as redes neurais convolucionais se tornaram a abordagem dominante para a detecção de objetos (REN et al., 2017). Métodos de detecção de objetos baseados em CNN alcançaram um desempenho notável ao aproveitar a capacidade das redes profundas de aprender representações hierárquicas de dados visuais.

Uma abordagem popular para a detecção de objetos é o framework Faster R-CNN (Rede Convolutacional Baseada em Regiões) (REN et al., 2017). O Faster R-CNN combina uma rede de proposta de região (RPN) com um detector Fast R-CNN para obter uma detecção de objetos precisa e eficiente. A RPN gera propostas de região, que são caixas delimitadoras potenciais que podem conter objetos de interesse. Essas propostas são então classificadas e refinadas pelo detector Fast R-CNN (REN et al., 2017). Os componentes RPN e Fast R-CNN são treinados juntos de forma integrada, permitindo que a rede aprenda a gerar propostas de região de alta qualidade e classificar objetos com precisão (REN et al., 2017).

Abordagens baseadas em aprendizado profundo, como o Faster R-CNN, mostraram avanços significativos na detecção de objetos, utilizando CNNs para aprender representações hierárquicas de dados visuais (KEVIN et al., 2019).

A RPN no Faster R-CNN é uma rede totalmente convolutacional que prevê limites de objetos e pontuações de objetividade em cada posição da imagem (REN et al., 2017). Ela gera propostas de região identificando regiões que provavelmente contêm objetos. Essas propostas são então usadas pelo detector Fast R-CNN para classificação e localização de objetos (REN et al., 2017). Ao compartilhar características convolucionais entre a RPN e o Fast R-CNN, a rede pode determinar efetivamente onde concentrar sua atenção para a detecção de objetos (REN et al., 2017).

Outro algoritmo popular de detecção de objetos é o YOLO. Ele trata a detecção de objetos como um problema de regressão e utiliza uma única rede neural para prever caixas delimitadoras e probabilidades de classe diretamente a partir da imagem completa em uma única avaliação (ALAM; ISLAM, 2019). O YOLO é conhecido por sua rapidez e precisão, tornando-o um método comumente usado para detecção de objetos (WANG et al., 2021). Ele tem sido aplicado em diferentes áreas, como identificação e contagem de células sanguíneas na área da saúde (ALAM; ISLAM, 2019), detecção de pedestres em vídeos de vigilância (YANG, 2023) e detecção em tempo real em dispositivos móveis (WANG et al., 2021).

O YOLO possui várias vantagens em relação a outros algoritmos de detecção de objetos. É mais rápido em comparação com métodos como o Faster R-CNN (ALAM; ISLAM, 2019). A capacidade de detecção em tempo real do YOLO o torna adequado para aplicações que requerem respostas rápidas, como sistemas de vídeo inteligentes (LEE; HWANG, 2021). Além disso, o YOLO leva em consideração tanto a velocidade quanto a precisão, tornando-o uma escolha versátil para tarefas de detecção de objetos (WANG et al., 2021).

O algoritmo YOLO tem sido amplamente estudado e aprimorado. Pesquisadores propuseram variações e melhorias no *framework* original do YOLO, como o TRC-YOLO, que é um método de detecção em tempo real para alvos leves em dispositivos móveis (WANG et al., 2021). Esses avanços visam melhorar o desempenho e a eficiência da detecção de objetos em diferentes cenários.

Em resumo, a detecção de objetos em visão computacional envolve o desenvolvimento de algoritmos e técnicas para identificar e localizar automaticamente objetos em dados visuais. Abordagens baseadas em CNN, como o Faster R-CNN e o YOLO, avançaram significativamente o campo, alcançando alta precisão e eficiência em tarefas de detecção de objetos. A detecção de objetos tem aplicações diversas e é essencial para permitir que as máquinas compreendam e interajam com o mundo visual.

### 3.2.3.1 YOLO

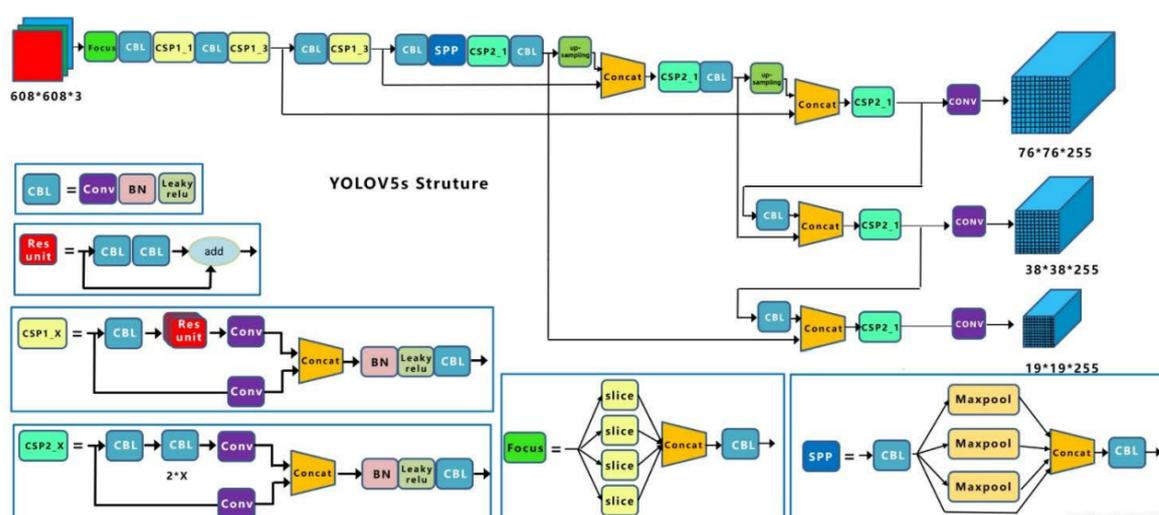
A arquitetura YOLO é um modelo de detecção de objetos em tempo real que emprega uma única rede neural convolucional para dividir uma imagem em grades, sendo que cada grade faz previsões sobre caixas delimitadoras e pontuações de confiança (GÜNDÜZ; IŞIK, 2023a). O YOLO pode detectar mais de 9000 categorias diferentes de objetos, tornando-se um sistema de detecção versátil e abrangente (REDMON; FARHADI, 2016). Ele opera com a imagem inteira de uma vez, em vez de suas partes individuais, e estuda representações generalizadas de objetos, contribuindo para sua alta velocidade de trabalho, permitindo que reconheça até 35 quadros por segundo (KALININA; NIKOLAEV, 2020). O YOLOv3<sup>2</sup>, um dos seus *frameworks* comuns de detecção de objetos em uma etapa, utiliza uma única CNN para prever várias caixas delimitadoras e probabilidades de classe (ZHONG; CHEN; MIAN, 2023). Além disso, a alta precisão e velocidade do YOLO o tornam mais adequado para aplicações em tempo real do que outros algoritmos (REN et al., 2020).

A arquitetura YOLO tem visto melhorias incrementais ao longo dos anos, sendo o YOLOv3 um exemplo dessas melhorias (REDMON; FARHADI, 2018). O YOLOv3 trouxe aprimoramentos, como a adoção da estrutura de rede *darknet-19* como a rede

<sup>2</sup> <https://github.com/ultralytics/yolov3>

de extração de características, a combinação de características finas da imagem e a conexão de características rasas com características profundas, o que tem sido útil para a detecção de alvos de pequeno tamanho (ZHANG et al., 2020). Além disso, o YOLOv5<sup>3</sup> viu melhorias significativas, incluindo o uso de uma rede de agrupamento espacial de pirâmide baseada no método de agrupamento estocástico para modificar a estrutura básica da rede, a aplicação de uma rede de pirâmide de características bidirecionais para fusão de características em várias escalas, e a melhoria da função de perda, adotando a função EIoU para otimizar o modelo de treinamento (LIU, 2022). A Figura 18 ilustra a arquitetura de camadas de CNN do YOLOv5.

Figura 18 – Estrutura do YOLOv5.



Fonte: Liu (2022)

Outras versões do YOLO disponíveis para utilização:

- YOLOv6<sup>4</sup> foi disponibilizado pela Meituan em 2022 e está em uso em muitos dos robôs de entrega autônomos da empresa;
- YOLOv7<sup>5</sup> adicionou tarefas adicionais, como estimativa de pose no conjunto de dados de pontos-chave COCO.
- YOLOv8<sup>6</sup> é a versão mais recente do YOLO da *Ultralytics*. Como um modelo de última geração (SOTA), o YOLOv8 se baseia no sucesso das versões anteriores, introduzindo novos recursos e melhorias para melhorar o desempenho, a flexibilidade e a eficiência. O YOLOv8 oferece suporte a uma gama completa de tarefas

<sup>3</sup> <https://github.com/ultralytics/yolov5>

<sup>4</sup> <https://github.com/meituan/YOLOv6>

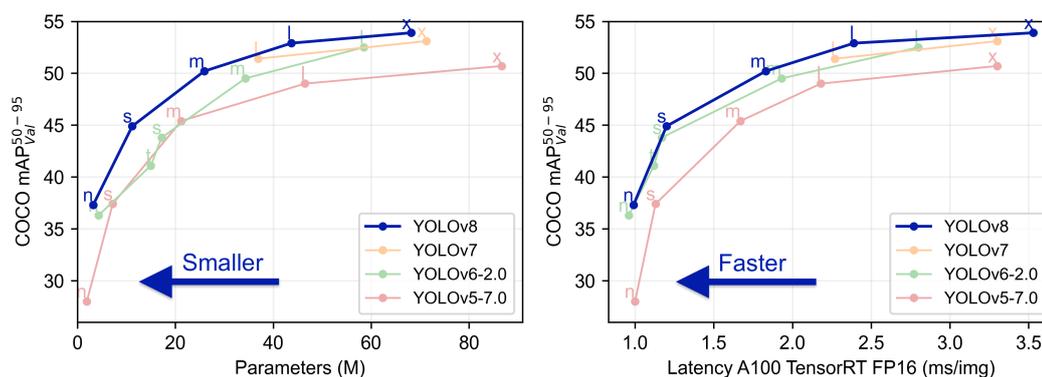
<sup>5</sup> <https://github.com/WongKinYiu/yolov7>

<sup>6</sup> <https://github.com/ultralytics/ultralytics>

de IA de visão, incluindo detecção, segmentação, estimativa de pose, rastreamento e classificação. Essa versatilidade permite que os usuários aproveitem os recursos do YOLOv8 em diversos aplicativos e domínios.

A Figura 19 ilustra a diferença de desempenho dessas versões treinadas com o dataset COCO(LIN et al., 2014).

Figura 19 – Análise de desempenho de mAP de versões do YOLO sobre o dataset COCO.



Fonte: Repositório do YOLOv8

Em conclusão, a arquitetura YOLO é um modelo de detecção de objetos em tempo real poderoso e versátil que passou por várias iterações e melhorias, tornando-o adequado para uma ampla gama de aplicações, incluindo, mas não se limitando à detecção de placas de trânsito, classificação de defeitos em pavimentos e até mesmo detecção e contagem de leveduras.

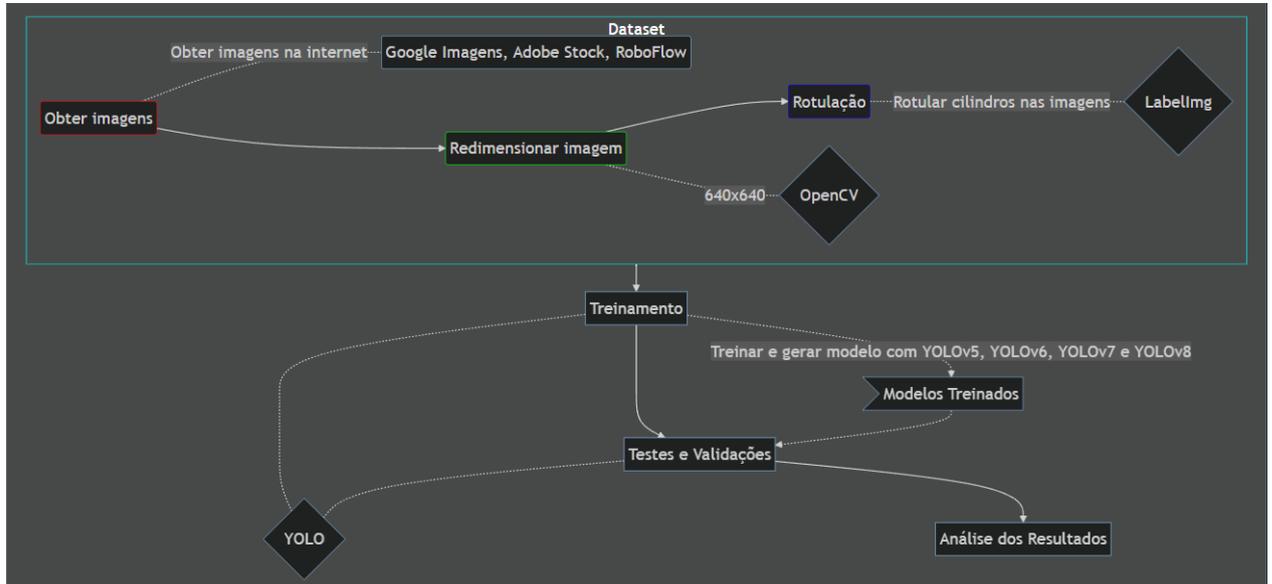
## 4 Metodologia

Neste Capítulo será apresentada em detalhe a abordagem utilizada para detectar cilindros pressurizados com versões do YOLO. Na Seção 4.1 é demonstrado o fluxo de desenvolvimento da pesquisa. A Seção 4.2 detalha a obtenção de imagens, rotulação dos cilindros e montagem do *dataset*. Já a Seção 4.3 explana a configuração utilizada para o treinamento dos modelos YOLO com o *dataset* de imagens. Por fim, a Seção 4.4 demonstra as métricas utilizadas para avaliar o modelo treinado.

### 4.1 Fluxograma

O desenvolvimento desta pesquisa baseia-se na preparação do dataset, treinamento e análise dos resultados com métricas propostas por Jocher et al. (2022) com versões atuais do *framework* YOLO para modelos personalizados de detecção. A Figura 20 ilustra todos os passos da metodologia proposta.

Figura 20 – Fluxograma de desenvolvimento da pesquisa.



Fonte: Elaborada pelo autor

O fluxo inicia-se pela obtenção de imagens em repositórios na internet. Sendo essas redimensionadas e rotuladas para preparação do *dataset* que será utilizado para treinar e avaliar os modelos no YOLO. Por fim, os dados gerados no treinamento serão analisados.

## 4.2 Criação do Dataset de Imagens

Para a criação do dataset<sup>1</sup>, foram coletadas 1799 imagens na internet contendo cilindros dos mais variados tipos e ambientes. As imagens foram pesquisadas através do *Google Imagens*<sup>2</sup> sob Licença *Creative Commons*<sup>3</sup>, *RoboFlow*<sup>4</sup> que permitem que essas imagens sejam usadas gratuitamente com os devidos créditos. Além disso, outras imagens foram adquiridas também através do *Adobe Stock*<sup>5</sup> sob licença gratuita.

Alguns dos principais termos utilizados em português e inglês nos mecanismos de buscas nas plataformas foram:

- Cilindros pressurizados;
- Cilindros pressurizados em pilhas de sucatas;
- Cilindros de oxigênio;
- Cilindros de CO<sub>2</sub>;
- Cilindros descartados;
- Cilindros de gás; e
- Extintores.

Todas as imagens foram redimensionadas para 640x640 *pixels* através do *framework* OpenCV<sup>6</sup>.

Após esse processamento, os cilindros contidos nas imagens foram rotulados com a ferramenta Labelling (TZUTALIN, 2015), conforme mostra a Figura 21.

---

<sup>1</sup> Dataset disponível em: [https://drive.google.com/drive/folders/17xnue1uzC3\\_jl2t8fJU5ggjOH2OeOgT](https://drive.google.com/drive/folders/17xnue1uzC3_jl2t8fJU5ggjOH2OeOgT)

<sup>2</sup> <https://images.google.com.br/>

<sup>3</sup> <https://creativecommons.org/tag/google-image-search/>

<sup>4</sup> <https://app.roboflow.com/>

<sup>5</sup> <https://stock.adobe.com/>

<sup>6</sup> <https://opencv.org/>

Figura 21 – Interface do software de rotulação e categorização *LabelImg*.



Fonte: Tzutalin (2015)

Esta rotulação gera os arquivos parametrizados para o treinamento no YOLO, conforme demonstra o Quadro 1. Neste dataset foram rotulados 4249 cilindros. Após a rotulação, o dataset foi dividido em dois grupos:

- Treinamento (80%) - 1436 imagens com 3408 cilindros rotulados; e
- Validação (20%) - 363 imagens com 841 cilindros rotulados.

classe	x-centro	y-centro	largura	altura
15	0.479687	0.582262	0.225000	0.670951
15	0.711719	0.579692	0.220312	0.696658
15	0.257812	0.592545	0.221875	0.686375
15	0.073438	0.579692	0.143750	0.670951
15	0.922656	0.595116	0.154688	0.696658
15	0.714844	0.287918	0.207813	0.246787
15	0.509375	0.313625	0.190625	0.267352
15	0.334375	0.302057	0.153125	0.254499
15	0.121875	0.302057	0.165625	0.275064
15	0.181250	0.206941	0.143750	0.223650
15	0.364063	0.213368	0.140625	0.246787
15	0.545312	0.219794	0.156250	0.249357
15	0.714063	0.200514	0.184375	0.231362
15	0.900000	0.208226	0.184375	0.236504

Tabela 1 – Trecho do arquivo com rotulações para YOLO gerado para a base de imagens

### 4.3 Treinamento

Para o treinamento foram selecionadas as arquiteturas do YOLOv5, YOLOv6, YOLOv7 e YOLOv8.

Todas as versões disponibilizam modelos pré-treinados para servirem de base na transferência de aprendizado, onde uma rede é pré ajustada a partir de um outro modelo treinado com outros dados (PAN; YANG, 2010). O modelo selecionado para esta pesquisa foram os modelos médios (m) por conta da limitação dos recursos de processamento para o treinamento. Pois ele tenta um equilíbrio entre desempenho e precisão.

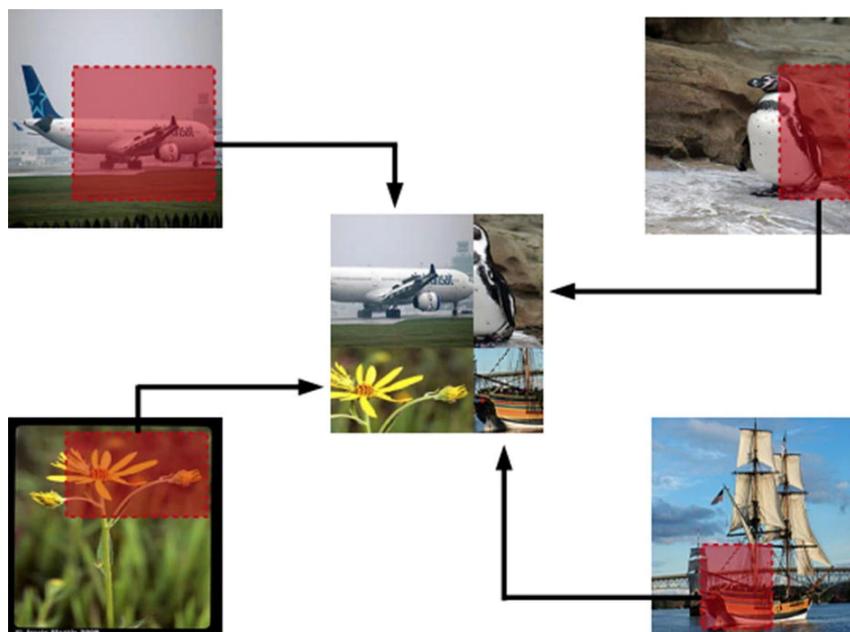
O treinamento foi configurado para executar em até 1000 épocas em um computador com processador AMD Ryzen 7 5800H de 3.20 GHz, 16GB de memória RAM e uma placa de vídeo GeForce RTX 3060 com 6GB de RAM.

O YOLO utiliza a técnica de *Imagem Data Augmentation* (IDA), pois tem integrado ao seu código o *framework Albumentations*<sup>7</sup>. Ela é uma técnica utilizada em aprendizado de máquina e aprendizado profundo para aumentar o tamanho e a diversidade de um conjunto de dados de treinamento, criando novos pontos de dados sintéticos. Isso envolve a aplicação de várias transformações ou modificações nos dados existentes para gerar amostras adicionais que sejam similares, mas não idênticas aos dados originais (SHORTEN; KHOSHGOFTAAR, 2019; PARK et al., 2019; WU et al., 2022). A Figura 22 ilustra uma das formas de aumento de *dataset* com IDA, neste caso novas imagens geradas com misturas aleatórias de outras imagens cortadas.

---

<sup>7</sup> <https://github.com/albumentations-team/albumentations>

Figura 22 – Exemplo de mistura de imagens por meio de corte e aplicação de *patches* aleatórios com IDA.



Fonte: Takahashi, Matsubara e Uehara (2019)

Por padrão, o YOLO disponibiliza alguns hiperparâmetros para IDA testadas nos *datasets* COCO (LIN et al., 2014), *Objects365* (SHAO et al., 2019) e VOC (EVERINGHAM et al., 2010).

Nesta pesquisa, utilizaremos os hiperparâmetros testados com o YOLOv5 no dataset VOC. Os valores desses hiperparâmetros e o que cada um ajusta estão disponíveis no Anexo A. Essa escolha se deu, pois com esse conjunto de valores, os treinamentos demonstraram melhorias nos resultados em cerca de 4 % em comparação com os treinamentos sem aumento de dados.

#### 4.4 Métricas de Avaliação

A **matriz de confusão** é uma tabela usada para avaliar o desempenho de um modelo de classificação. Ele fornece um resumo das previsões feitas pelo modelo em comparação com os rótulos reais dos dados. A matriz é tipicamente quadrada e possui linhas e colunas representando as classes preditas e reais, respectivamente (BRATU; CZIBULA, 2021; MIAO et al., 2022). Ela é dividida em quatro quadrantes: Verdadeiros Positivos (TP), Verdadeiros Negativos (TN), Falsos Positivos (FP) e Falsos Negativos (FN). Cada célula na matriz representa a contagem ou proporção de pontos de dados que se enquadram em uma categoria específica (BRATU; CZIBULA, 2021; MIAO et al., 2022).

A **Mean Average Precision (mAP)**, definida na Equação 4.4, é um indicador

amplamente utilizado para avaliar modelos de detecção de objetos (MAHAUR; SINGH; MISHRA, 2022; REN et al., 2017), incluindo o YOLO. Ele mede a precisão e a recuperação do modelo ao calcular a média da precisão em diferentes categorias de objetos e diferentes níveis de limiares de confiança. Um  $mAP$  mais alto indica um melhor desempenho de detecção (GÜNDÜZ; IŞIK, 2023b).

$mAP$  considera Precisão (P) e Recall (R), definidos nas Equações 4.1 e 4.2, respectivamente.

$$(P) = \frac{TP}{TP + FP} \quad (4.1)$$

$$(R) = \frac{TP}{TP + FN} \quad (4.2)$$

*Average Precision* (AP) é a média de precisão em diferentes valores de Recall obtidos pela curva *Precision Recall* (PR) sob um determinado limite, definida na Equação 4.3.

$$(AP) = \sum_k (R_{k+1} - R_k) \max_{R: \bar{R} \geq R_{k+1}} P(\bar{R}) \quad (4.3)$$

onde  $P(\bar{R})$  é o P medida em  $\bar{R}$ .

$$(mAP) = \frac{1}{n} \sum_{k=1}^n AP_i \quad (4.4)$$

onde  $n$  é o número de classes e  $AP_i$  é a  $AP$  na classe  $i$ .

## 5 Resultados e Discussão

Foram realizados quatro treinamentos com o intuito de analisar a arquitetura e resultados das diferentes versões do YOLO no conjunto de dados.

Devido a limitações de desempenho relacionadas aos recursos disponíveis, o treinamento foi conduzido utilizando uma configuração de lote (*batch*) que processava 12 imagens por iteração. Nas Figuras 23, 24, 25 e 26 é possível visualizar esses lotes, com a representação das imagens geradas juntamente aos cilindros devidamente rotulados. No contexto dessas ilustrações, a marcação numérica no retângulo delimitador indica a classe treinada, no caso, a classe "cilindro". Essas figuras foram geradas empregando os hiperparâmetros mencionados.

Figura 23 – Batch de imagens gerado com YOLOv5.



Fonte: Gerada com YOLOv5

Figura 24 – Batch de imagens gerado com YOLOv6.



Fonte: Gerada com YOLOv6

Figura 25 – Batch de imagens gerado com YOLOv7.



Fonte: Gerada com YOLOv7

Figura 26 – Batch de imagens gerado com YOLOv8.

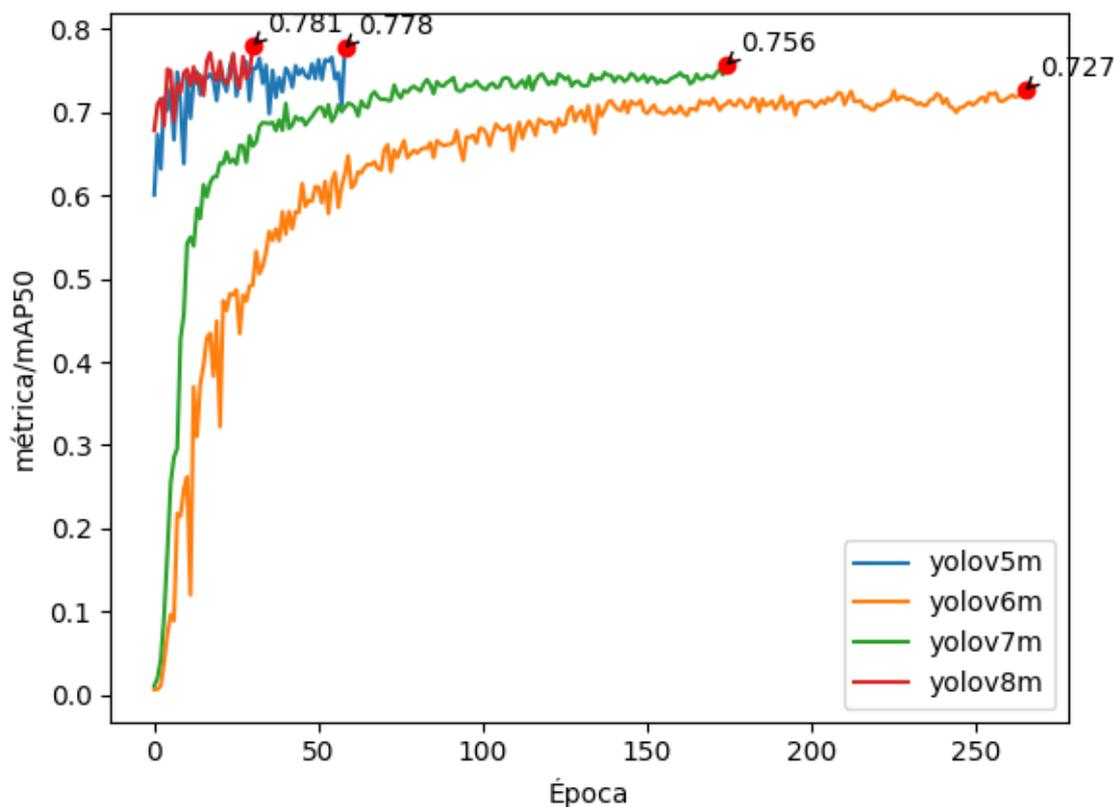


Fonte: Gerada com YOLOv8

Cada ciclo de treinamento foi definido para abranger até 1000 épocas, embora o YOLO esteja equipado com um parâmetro (*patience*) para identificar a convergência dos resultados e, por padrão, este parâmetro seja estabelecido em 100 épocas. Em outras palavras, caso não ocorram alterações substanciais nos resultados após 100 épocas, o processo de treinamento é automaticamente interrompido.

A Figura 27 ilustra os resultados máximos de mAPs para cada treinamento e em quais épocas esses valores atingiram convergência.

Figura 27 – Análise comparativa de convergência de mAP entre os modelos.



Fonte: Gerada pelo autor

O Quadro 2 mostra detalhadamente os valores das demais métricas alcançada por cada treinamento.

Hiperparam.	Conv.	P	R	mAP.5	mAP.5:.95
YOLOv5m	58	0.788	0.707	0.778	0.419
YOLOv6m	265	0.804	0.668	0.727	0.386
YOLOv7m	174	0.773	0.73	0.756	0.381
YOLOv8m	30	0.806	0.733	0.781	0.42

Tabela 2 – Resultados de validação de cada treinamento

Com esses resultados, é possível verificar um melhor resultado de YOLOv8 em comparação com as outras versões. Atingindo sua convergência logo na trigésima época.

Os Quadros 3, 4, 5 e 6 demonstram a matrizes de confusão geradas por cada treinamento, onde o melhor resultado foi obtido com treinamento feito com o YOLOv8 com 78% de TP e apenas 22% de FN.

Predito/Real	Cilindro	Não-cilindro
<b>Cilindro</b>	74%	
<b>Não-cilindro</b>	26%	100%

Tabela 3 – Matriz de confusão do treinamento com YOLOv5

Predito/Real	Cilindro	Não-cilindro
<b>Cilindro</b>	72%	
<b>Não-cilindro</b>	28%	100%

Tabela 4 – Matriz de confusão do treinamento com YOLOv6

Predito/Real	Cilindro	Não-cilindro
<b>Cilindro</b>	68%	
<b>Não-cilindro</b>	32%	100%

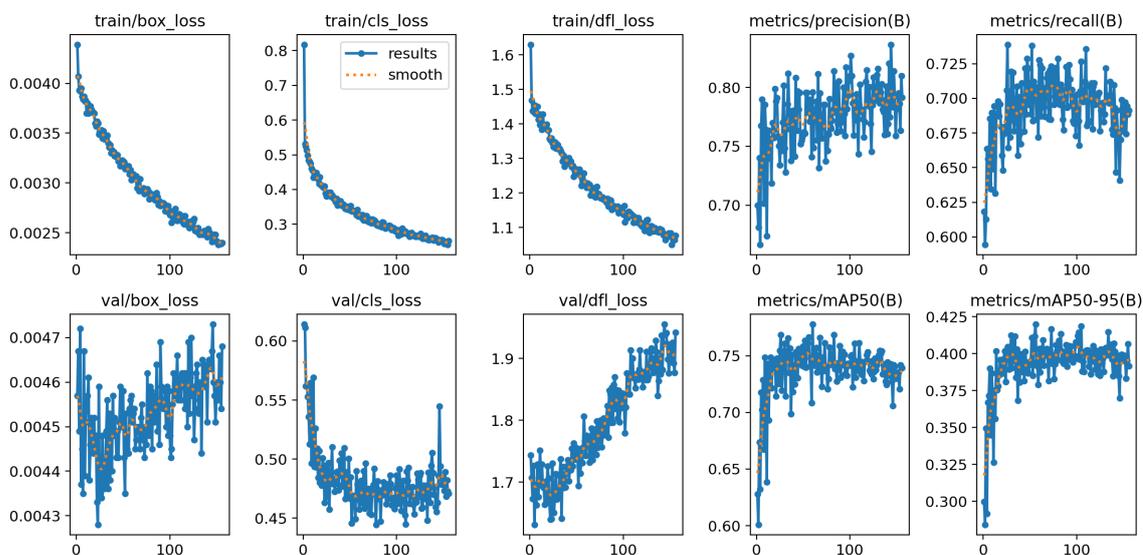
Tabela 5 – Matriz de confusão do treinamento com YOLOv7

Predito/Real	Cilindro	Não-cilindro
<b>Cilindro</b>	78%	
<b>Não-cilindro</b>	22%	100%

Tabela 6 – Matriz de confusão do treinamento com YOLOv8

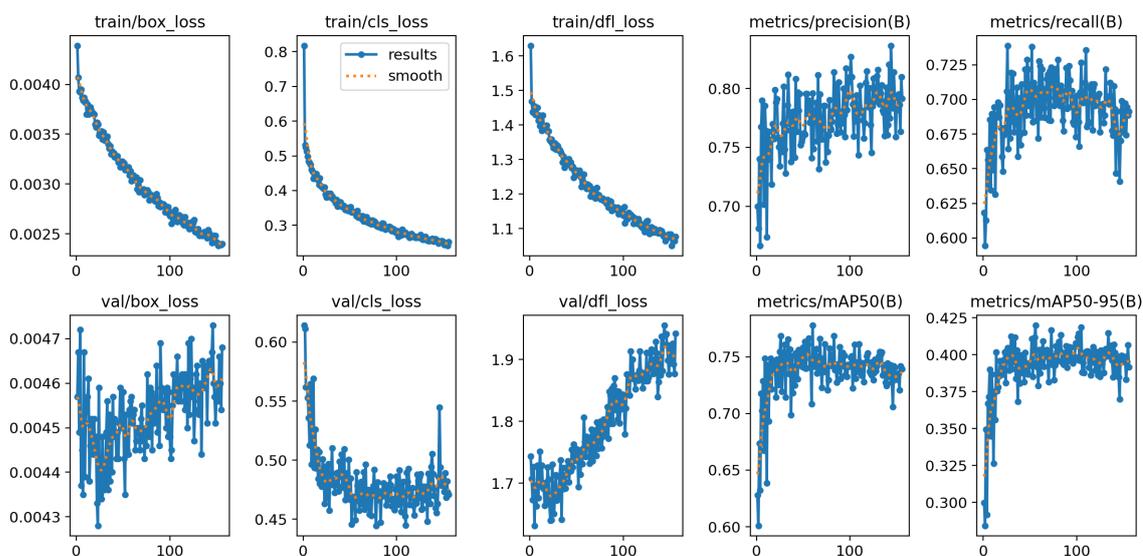
As Figuras 28,29, 30, 31 ilustram as demais métricas disponibilizadas por cada versão do YOLO.

Figura 28 – Resultados gráficos do treinamento com YOLOv5



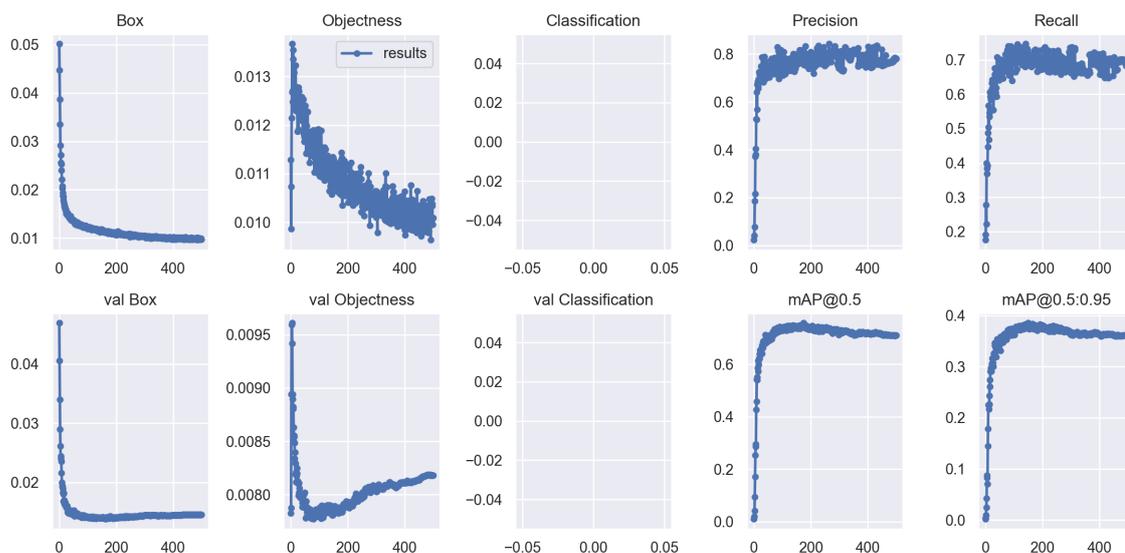
Fonte: Gerada com YOLOv5

Figura 29 – Resultados gráficos do treinamento com YOLOv6



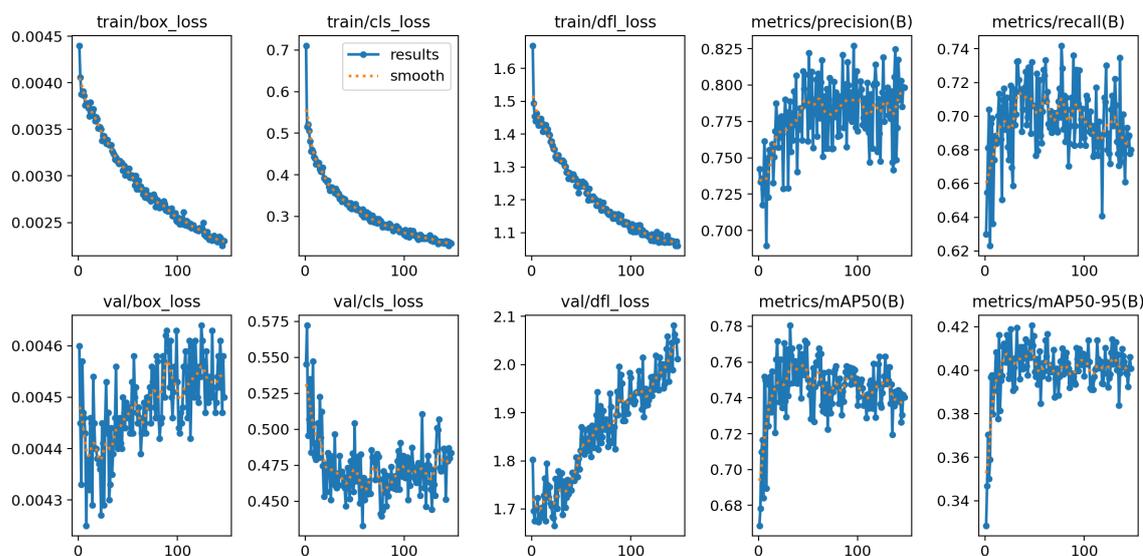
Fonte: Gerada com YOLOv6

Figura 30 – Resultados gráficos do treinamento com YOLOv7



Fonte: Gerada com YOLOv7

Figura 31 – Resultados gráficos do treinamento com YOLOv8



Fonte: Gerada com YOLOv8

Neste momento, é relevante abordar os valores do mAP, os quais foram calculados considerando diferentes limiares de detecção, conhecidos como threshold. O mAP<sub>0.5</sub> corresponde ao limiar mínimo de confiança necessário para classificar uma detecção como positiva, enquanto o mAP<sub>0.95</sub> refere-se a um limiar de confiança mais elevado. Esse contexto implica que, para uma detecção ser validamente positiva, a pontuação de confiança associada precisa ser igual ou superior a 0,95. A adoção do segundo valor torna-se pertinente quando se almeja um critério de detecção mais rigoroso.

Os modelos treinados obtiveram resultados de mAP acima de 72% para o limiar de 0,5 e 38% para o limite máximo de 0,95. Esses resultados foram obtidos a partir da análise de um conjunto de validação composto por 363 imagens contendo 841 cilindros rotulados. As Figuras 32, 33, 34 e 35 proporcionam uma representação visual das previsões geradas para algumas das imagens contidas nesse conjunto de dados. Junto com a designação da classe do objeto, também é apresentada a probabilidade estimada de esse objeto ser identificado como um cilindro, de acordo com o modelo em questão.

Figura 32 – Predição de conjunto de validação com YOLOv5 .



Fonte: Gerada com YOLOv5

Figura 33 – Predição de conjunto de validação com YOLOv6.



Fonte: Gerada com YOLOv6

Figura 34 – Predição de conjunto de validação com YOLOv7.



Fonte: Gerada com YOLOv7

Figura 35 – Predição de conjunto de validação com YOLOv8.



Fonte: Gerada com YOLOv8

Em destaque, os melhores resultados foram obtidos pelos modelos YOLOv5 e YOLOv8, ambos com mAP acima de 76% e 78%, respectivamente.

## 5.1 Discussões

A presente pesquisa se dedicou à tarefa desafiadora de detectar cilindros em ambientes diversos, um domínio específico que revelou sua complexidade devido à escassez de conjuntos de dados adequados. A construção de um conjunto de dados representativo, contendo imagens que retratam cilindros em variados ambientes e estados, demandou esforços substanciais.

A montagem do conjunto de dados foi guiada por critérios rigorosos, priorizando

imagens de licenças gratuitas para garantir conformidade legal e disponibilidade pública. Essa abordagem adicionou complexidade à fase de coleta, buscando equilibrar a qualidade e diversidade das imagens com as restrições de licenciamento.

Adicionalmente, a colaboração valiosa de uma empresa siderúrgica proporcionou um conjunto de imagens relevantes. Estas imagens, semelhantes ao propósito deste estudo, ofereceram uma oportunidade única de validação e comparação.

Além disso, conduziram-se testes em imagens não incluídas no conjunto de treinamento, representando o ambiente de pilhas de sucatas. Esses testes visaram avaliar o desempenho do modelo em detecções fora do escopo do treinamento, fornecendo *insights* sobre sua robustez e generalização.

As Figuras 36a e 36b ilustram os testes feitos com imagens reais em pilhas de sucatas disponibilizadas por uma empresa siderúrgica.

Figura 36 – Testes em imagens reais de pilhas de sucatas de uma siderúrgica.



(a) Testes com modelo treinado com YOLOv8



(b) Testes com modelo treinado com YOLOv8

Fonte: Gerada com YOLOv8

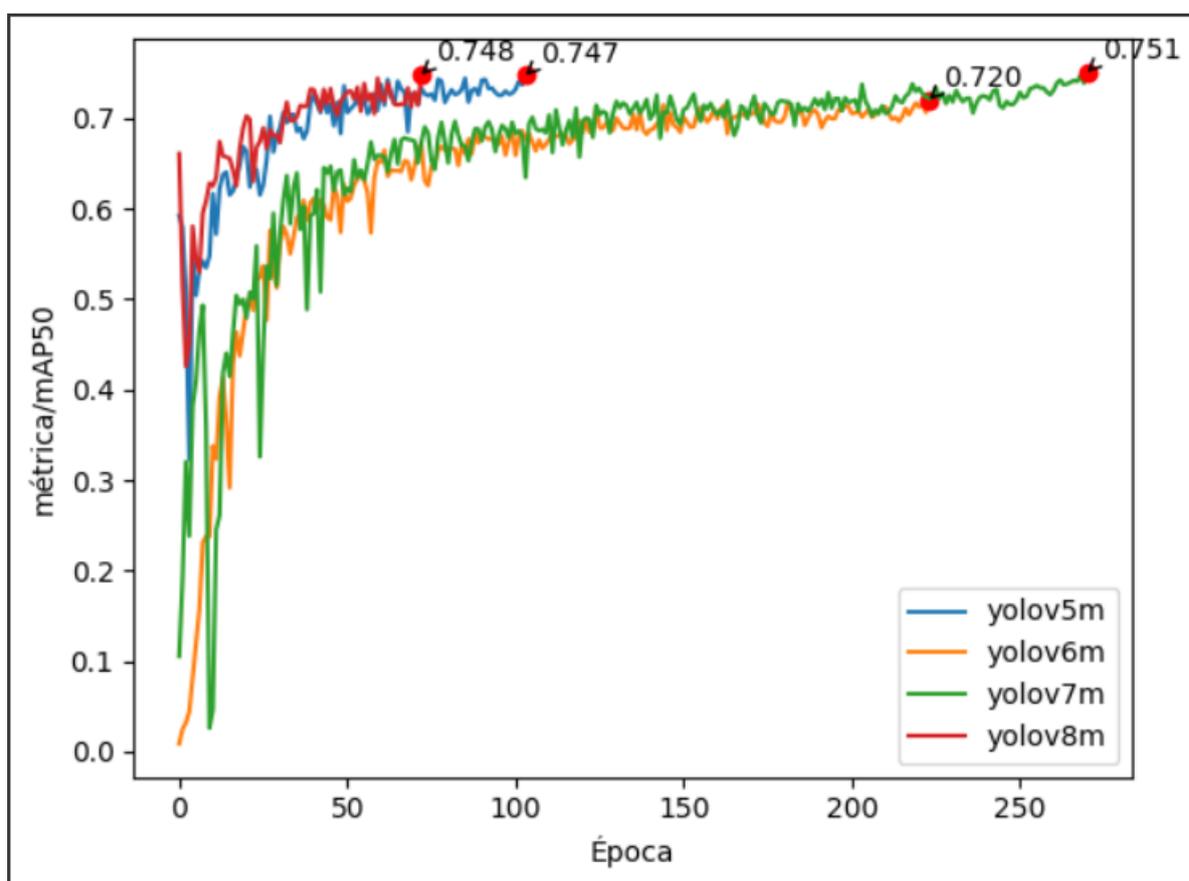
Essa abordagem investigativa e aprofundada proporcionou uma compreensão mais clara do desempenho dos modelos de detecção propostos, reforçando a rele-

vância da pesquisa no contexto da detecção de objetos em ambientes complexos e desafiadores.

A análise dos resultados desses testes evidencia um desempenho sólido na detecção de cilindros em cenários de pilhas de sucatas. No entanto, é essencial conduzir testes abrangentes adicionais com diversas imagens de ambientes distintos e avaliar a capacidade de detecção em contextos de vídeo para obter *insights* valiosos sobre sua aplicabilidade dinâmica.

Entre os modelos treinados, aqueles derivados do Anexo A demonstraram uma clara melhora na detecção precisa dos cilindros. Essa observação sugere a relevância dos hiperparâmetros ajustados para as técnicas de *Image Data Augmentation* empregadas pelo YOLO. A Figura 37 demonstra o treinamento sem aumento de dados, o que confirma a melhoria nos resultados com um aumento de até 4% de mAP.

Figura 37 – Análise comparativa de convergência de mAP entre os modelos sem aumento de dados.



Fonte: Gerada pelo autor

Existe a possibilidade de otimizar esses hiperparâmetros para IDA por meio de técnicas de algoritmo generativo. No entanto, neste trabalho, não houve tempo e recursos computacionais suficientes para aplicar essas ferramentas. Como resultado, os valores de mAP obtidos foram baixos e sem relevância.

A perspectiva de ampliar o conjunto de dados com imagens provenientes do mesmo domínio proporciona oportunidades para um retreinamento refinado do modelo proposto, beneficiando-se da transferência de aprendizado. Esse processo pode conduzir a resultados ainda mais aprimorados, consolidando a capacidade do modelo de generalizar e desempenhar com excelência em diferentes contextos.

Diante disso, a continuação deste estudo sugere uma exploração mais profunda dessas áreas de interesse, incluindo a expansão do conjunto de dados, o refinamento dos modelos e a investigação do comportamento em vídeos, representando passos cruciais para uma compreensão abrangente e eficaz da detecção de cilindros em pilhas de sucatas.

## 6 Conclusões e Trabalhos Futuros

### 6.1 Conclusões

Em síntese, esta pesquisa empreendeu uma análise abrangente da aplicação de diferentes versões do *framework* YOLO para a detecção de cilindros pressurizados. O estudo se destacou ao enriquecer o modelo com técnicas avançadas de *Image Data Augmentation* (IDA) e refinamentos de hiperparâmetros, expandindo as fronteiras de seu desempenho tradicional, além de uma análise comparativa dos resultados obtidos com treinamentos do YOLOv5, YOLOv6, YOLOv7 e YOLOv8. Os resultados foram minuciosamente examinados, levando em conta os diferentes conjuntos de hiperparâmetros disponibilizados no *framework* YOLOv5, conforme documentado nos Anexos [A](#).

O procedimento de treinamento foi conduzido com zelo, curando um conjunto de dados composto por 1799 imagens, abrangendo um total de 4249 cilindros pressurizados. Esse conjunto foi estrategicamente dividido em duas parcelas distintas:

- Conjunto de Treinamento (80%): 1436 imagens contendo 3408 cilindros; e
- Conjunto de Validação (20%): 363 imagens contendo 841 cilindros.

Os modelos resultantes do treinamento com YOLOv8 demonstraram um desempenho notável, alcançando uma média de Precisão Média de Anotações (*mAP*) superior a 78% na detecção de cilindros, conforme avaliado no conjunto de validação. Além disso, o modelo treinado com o YOLOv5 ainda exibiu um desempenho respeitável, com *mAP* superior a 76%.

Ao testar os modelos em cenários que simulam pilhas de sucatas, os resultados foram promissores, embora algumas instâncias de falsos positivos e negativos tenham sido observadas.

Desta forma, esta pesquisa revelou um avanço substancial na detecção de cilindros pressurizados, evidenciando uma taxa *mAP* superior a 77% no conjunto de validação. Não obstante, é imperativo ressaltar a necessidade de investigações adicionais em ambientes reais, assim como a expansão do conjunto de dados mediante a incorporação de mais imagens para enriquecer a diversidade das amostras. Dado o desafio subjacente à detecção de objetos parcialmente obscurecidos em arranjos complexos, é recomendável prosseguir com uma investigação mais profunda sobre a detecção de objetos em condições de oclusão parcial utilizando a estratégia YOLO. O

caminho trilhado por este estudo abre portas para uma compreensão mais sólida e eficaz da detecção de objetos em cenários desafiadores, contribuindo significativamente para a pesquisa em visão computacional e aplicações práticas afins.

## 6.2 Contribuições

- **Aplicação Avançada do YOLO:** Demonstração eficaz e comparativa da aplicação aprimorada de versões do framework YOLO para detecção de cilindros pressurizados.
- **Conjunto de Dados Curado:** Criação de um conjunto de dados valioso, contendo imagens diversificadas de cilindros pressurizados para treinamento e validação.
- **Avaliação em Cenários Complexos:** Avaliação bem-sucedida da detecção em situações desafiadoras, como pilhas de sucatas, evidenciando a aplicabilidade prática.
- **Direcionamento para Pesquisas Futuras:** Identificação de áreas promissoras para investigações subsequentes, como detecção de objetos em oclusão parcial.

## 6.3 Trabalhos Futuros

### 1. Detecção em Cenários Desafiadores:

- Desenvolver estratégias específicas para melhorar a detecção de cilindros pressurizados em situações de oclusão parcial e quando os cilindros estão misturados com outros objetos.
- Explorar métodos avançados de fusão de informações para aprimorar a precisão da detecção em pilhas de objetos.

### 2. Ampliação e Diversificação do Conjunto de Dados:

- Expandir o conjunto de dados de treinamento para incluir uma variedade ainda maior de cenários, tais como diferentes ambientes industriais, ângulos de visão e variações de iluminação.
- Introduzir cilindros de diferentes tamanhos e formas para aumentar a diversidade do conjunto de treinamento.

### 3. Otimização de Hiperparâmetros Específicos:

- Investigar a otimização de hiperparâmetros específicos para a detecção de cilindros pressurizados, levando em consideração suas características geométricas e aparência visual.
- Explorar a influência de diferentes combinações de hiperparâmetros no desempenho da detecção.

#### **4. Avaliação em Ambientes Industriais Reais:**

- Realizar testes de campo em ambientes industriais reais onde a detecção de cilindros pressurizados é essencial, avaliando a eficácia e a aplicabilidade da abordagem desenvolvida.

#### **5. Técnicas de Aumento de Dados Avançadas:**

- Explorar técnicas avançadas de *Image Data Augmentation* (IDA) específicas para cilindros pressurizados, considerando suas propriedades físicas e características de superfície.

## Referências

ALAM, M. K.; ISLAM, M. T. Machine learning approach of automatic identification and counting of blood cells. *Healthcare Technology Letters*, v. 6, p. 103–108, 2019. Citado 3 vezes nas páginas 16, 44 e 45.

ALBAYRAK, A.; OZERDEM, M. S. Gas cylinder detection using deep learning based yolov5 object detection method. *2022 7th International Conference on Computer Science and Engineering (UBMK)*, 2022. Citado 2 vezes nas páginas 30 e 31.

ALHUMOUD, J. M.; AL-KANDARI, F. A. Analysis and overview of industrial solid waste management in kuwait. *Management of Environmental Quality: an International Journal*, Emerald Group Publishing Limited, v. 19, n. 5, p. 520–532, 2008. Citado na página 15.

ALLONDA. *O que é gestão de resíduos industriais?* 2023. Disponível em: <<https://allonda.com/blog/gestao-de-residuo/o-que-e-gestao-de-residuos-industriais/>>. Citado na página 14.

ALPAYDIN, E. *Introduction to machine learning*. [S.I.]: MIT press, 2014. Citado na página 40.

ARAFAT, Y.; REYES-ALDASORO, C. C. Computational image analysis techniques, programming languages and software platforms used in cancer research: A scoping review. In: SPRINGER. *Annual Conference on Medical Image Understanding and Analysis*. [S.I.], 2022. p. 833–847. Citado na página 34.

ARAÚJO, A. M. Classificação e detecção de pessoas em ambientes não controlados utilizando redes neurais convolucionais. Universidade Federal da Paraíba, 2019. Citado na página 43.

ARAÚJO, A. M. C.; OLIVEIRA, M. M. Connectivity-based cylinder detection in unorganized point clouds. *Pattern Recognition*, v. 100, p. 107161, 2020. Citado 3 vezes nas páginas 20, 21 e 23.

ARAÚJO, F. H. et al. Redes neurais convolucionais com tensorflow: Teoria e prática. *SOCIEDADE BRASILEIRA DE COMPUTAÇÃO. III Escola Regional de Informática do Piauí. Livro Anais-Artigos e Minicursos*, Sociedade Brasileira de Computação, v. 1, 2017. Citado 5 vezes nas páginas 38, 39, 40, 41 e 42.

AZZAHRA, L.; SAPTARINI, N. M. Pharmaceutical industrial waste regulation in five countries in asia. *Scholar*, v. 863, p. 15, 2021. Citado 2 vezes nas páginas 14 e 33.

BACKES, A. R.; JUNIOR, J. J. M. S. *Introdução à visão computacional usando Matlab*. [S.I.]: Alta Books Editora, 2016. Citado na página 35.

BALLARD, D. H.; BROWN, C. M. Computer vision. englewood cliffs. *J: Prentice Hall*, 1982. Citado na página 35.

- BERGAMASCO, F. et al. Cylinders extraction in non-oriented point clouds as a clustering problem. *Pattern Recognition*, Elsevier, v. 107, p. 107443, 2020. Citado 2 vezes nas páginas 29 e 30.
- BISHOP, C. M. *Pattern recognition and machine learning*. [S.l.]: Springer Science+ Business Media, 2006. Citado na página 42.
- BRATU, A.; CZIBULA, G. Dagan: An approach for augmenting time series imbalanced datasets via latent space sampling using adversarial techniques. *Scientific Programming*, Hindawi Limited, v. 2021, p. 1–13, 2021. Citado na página 52.
- CASTRO, A. A. M.; PRADO, P. P. L. Algoritmos para reconhecimento de padrões. *Revista Ciências Exatas*, v. 8, n. 2002, 2002. Citado na página 36.
- DUDA, R. O.; HART, P. E.; STORK, D. G. *Pattern classification*. [S.l.]: John Wiley & Sons, 2012. Citado na página 35.
- EVERINGHAM, M. et al. The pascal visual object classes (voc) challenge. *International Journal of Computer Vision*, v. 88, n. 2, p. 303–338, jun. 2010. Citado na página 52.
- FARHANZA, F. Information systems using object counting and background subtraction from video in supermarket. *International Journal of Advanced Trends in Computer Science and Engineering*, v. 9, p. 6362–6369, 2020. Citado na página 43.
- FARZADKIA, M. et al. Evaluation of industrial wastes management practices: Case study of the savojbolagh industrial zone, iran. *Waste Management & Research*, SAGE Publications Sage UK: London, England, v. 38, n. 1, p. 44–58, 2020. Citado na página 14.
- FENG, M. Q.; LEUNG, R. Y. Computer vision-based vehicle weigh-in-motion system. In: SPIE. *Health Monitoring of Structural and Biological Systems XVII*. [S.l.], 2023. v. 12488, p. 203–212. Citado na página 34.
- FIGUEIREDO, R. et al. A robust and efficient framework for fast cylinder detection. *Robotics and Autonomous Systems*, Elsevier, v. 117, p. 17–28, 2019. Citado 2 vezes nas páginas 25 e 26.
- GARRIDO-VALENZUELA, F.; CRANENBURGH, S. v.; CATS, O. Enriching geospatial data with computer vision to identify urban environment determinants of social interactions. *AGILE: GIScience Series*, v. 3, p. 1–5, 2022. Citado na página 43.
- GOODFELLOW, I.; BENGIO, Y.; COURVILLE, A. *Deep learning, ser. Adaptive computation and machine learning series*. [S.l.]: Cambridge, MA: The MIT Press, 2016. Citado na página 41.
- GUO, K. et al. A pavement distresses identification method optimized for yolov5s. *Scientific Reports*, v. 12, 2022. Citado na página 16.
- GÜNDÜZ, M.; IŞIK, G. A new yolo-based method for social distancing from real-time videos. *Neural Computing and Applications*, v. 35, p. 15261–15271, 2023. Citado na página 45.

- GÜNDÜZ, M. □.; IŞIK, G. A new yolo-based method for real-time crowd detection from video and performance analysis of yolo models. *Journal of Real-Time Image Processing*, v. 20, 2023. Citado na página 53.
- HASSAN, M.; DILAWAR, S.; AKBAR, M. Industrial waste management in pakistan: Problems and prospects. *Global Management Sciences Review*, VI, v. 6, p. 39–53, 2021. Citado na página 34.
- HAYKIN, S. et al. Neural networks and learning machines. vol. 3 pearson. *Upper Saddle River, NJ, USA*, 2009. Citado na página 42.
- HE, K. et al. Deep residual learning for image recognition. *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016. Citado na página 39.
- HOSSAIN, A.; ISLAM, M. T.; ALMUTAIRI, A. F. A deep learning model to classify and detect brain abnormalities in portable microwave based imaging system. *Scientific Reports*, v. 12, 2022. Citado na página 16.
- HOVEIDI, H. et al. Industrial waste management with application of riam environmental assessment: a case study on toos industrial state, mashhad. *Iranica Journal of Energy Amp; Environment*, 2013. Citado 2 vezes nas páginas 14 e 33.
- HUANG, C. et al. Multilayer hybrid deep-learning method for waste classification and recycling. *Computational Intelligence and Neuroscience*, v. 2018, p. 1–9, 2018. Citado 2 vezes nas páginas 24 e 25.
- JAIN, A. K.; DUIN, R. P. W.; MAO, J. Statistical pattern recognition: A review. *IEEE Transactions on pattern analysis and machine intelligence*, Ieee, v. 22, n. 1, p. 4–37, 2000. Citado 2 vezes nas páginas 35 e 36.
- JOCHER, G. et al. ultralytics/yolov5: v7. 0-yolov5 sota realtime instance segmentation. *Zenodo*, 2022. Citado na página 48.
- KALININA, M.; NIKOLAEV, P. Research of yolo architecture models in book detection. *Proceedings of the 8th Scientific Conference on Information Technologies for Intelligent Decision Making Support (ITIDS 2020)*, 2020. Citado na página 45.
- KARN, U. An intuitive explanation of convolutional neural networks. *the data science blog*, 2016. Citado na página 40.
- KARPATHY, A. Cs231n convolutional neural networks for visual recognition. *URL <http://cs231n.github.io>*, v. 50, 2017. Citado 2 vezes nas páginas 39 e 41.
- KEVIN, K. et al. Comparison of cellular video quality for object detection using neural network convolution. *SinkrOn*, v. 4, p. 260, 2019. Citado 2 vezes nas páginas 43 e 44.
- KRIZHEVSKY, A.; SUTSKEVER, I.; HINTON, G. E. Imagenet classification with deep convolutional neural networks. In: *Advances in neural information processing systems*. [S.l.: s.n.], 2012. Citado na página 40.
- KUMAR, S. et al. Robust vehicle detection based on improved you look only once. *Computers, Materials Amp; Continua*, v. 74, p. 3561–3577, 2023. Citado na página 16.

- LAB, L. *Convolutional Neural Networks (LeNet)*. 2018. Citado na página 37.
- LECUN, Y. et al. Handwritten digit recognition with a back-propagation network. In: *Advances in neural information processing systems*. [S.l.: s.n.], 1990. p. 396–404. Citado na página 37.
- LEE, C.; LIN, C. A two-phase fashion apparel detection method based on yolov4. *Applied Sciences*, v. 11, p. 3782, 2021. Citado na página 43.
- LEE, J.; HWANG, K. Yolo with adaptive frame control for real-time object detection applications. *Multimedia Tools and Applications*, v. 81, p. 36375–36396, 2021. Citado na página 45.
- LIN, T.-Y. et al. Microsoft coco: Common objects in context. In: SPRINGER. *Computer Vision–ECCV 2014: 13th European Conference, Zurich, Switzerland, September 6-12, 2014, Proceedings, Part V 13*. [S.l.], 2014. p. 740–755. Citado 2 vezes nas páginas 47 e 52.
- LIS, T.; NOWACKI, K. Pro-ecological possibilities of using metallurgical waste in the production of aggregates. *Production Engineering Archives*, v. 28, n. 3, p. 252–256, 2022. Citado na página 33.
- LIU, K. Stbi-yolo: A real-time object detection method for lung nodule recognition. *IEEE Access*, IEEE, v. 10, p. 75385–75394, 2022. Citado na página 46.
- LIU, Y.-J. et al. Cylinder detection in large-scale point cloud of pipeline plant. *IEEE transactions on visualization and computer graphics*, IEEE, v. 19, n. 10, p. 1700–1707, 2013. Citado 3 vezes nas páginas 19, 20 e 23.
- LU, Z. et al. Slicing-tracking-detection: Simultaneous multi-cylinder detection from large-scale and complex point clouds. *IEEE Transactions on Visualization and Computer Graphics*, IEEE, v. 28, n. 12, p. 4172–4185, 2021. Citado 3 vezes nas páginas 22, 23 e 24.
- MAHAUR, B.; SINGH, N.; MISHRA, K. Road object detection: a comparative study of deep learning-based algorithms. *Multimedia Tools and Applications*, Springer, v. 81, n. 10, p. 14247–14282, 2022. Citado na página 53.
- MARENGONI, M.; STRINGHINI, S. Tutorial: Introdução à visão computacional usando opencv. *Revista de Informática Teórica e Aplicada*, v. 16, n. 1, 2009. Citado na página 36.
- MARIM, Y. V. R. Detecção de objetos: Estudo e aplicação da arquitetura r-cnn. 2019. Citado na página 41.
- MIAO, N. et al. Instance-specific augmentation: Capturing local invariances. 2022. Citado na página 52.
- MORADI, S.; LAURENDEAU, D.; GOSSELIN, C. Multiple cylinder extraction from organized point clouds. *Sensors*, MDPI, v. 21, n. 22, p. 7630, 2021. Citado 2 vezes nas páginas 27 e 28.

- MUNTASA, A. et al. A pyramid model of convolutional neural network to classify acute lymphoblastic leukemia images. *International Journal of Intelligent Engineering & Systems*, v. 15, n. 6, 2022. Citado na página 39.
- MUSTAFAYEV, E.; AZIMOV, R. Comparative analysis of the application of multilayer and convolutional neural networks for recognition of handwritten letters of the azerbaijani alphabet. *Cybernetics and Computer Technologies*, p. 65–73, 2021. Citado na página 37.
- OLIVEIRA, D. C. *Uma abordagem para detecção de pessoas em imagens de veículos aéreos não-tripulados*. Dissertação (Mestrado) — Universidade Tecnológica Federal do Paraná, 2016. Citado 2 vezes nas páginas 40 e 41.
- OMARA, T. et al. Characterization and prognostication of wastes generated by industries in kampala industrial and business park—namanve. *Open Access Library Journal*, Scientific Research Publishing, v. 6, n. 4, p. 1–15, 2019. Citado 2 vezes nas páginas 33 e 34.
- OSMAN, I. I.; SHEHATA, M. S. Few-shot learning network for moving object detection using exemplar-based attention map. *2022 IEEE International Conference on Image Processing (ICIP)*, 2022. Citado na página 43.
- O'REILLY. *Detection or localization and segmentation*. 2023. Disponível em: <<https://www.oreilly.com/library/view/deep-learning-for/9781788295628/4fe36c40-7612-44b8-8846-43c0c4e64157.xhtml>>. Citado na página 16.
- PAN, S. J.; YANG, Q. A survey on transfer learning. *IEEE Transactions on Knowledge and Data Engineering*, v. 22, p. 1345–1359, 2010. Citado na página 51.
- PARK, D. S. et al. SpecAugment: A Simple Data Augmentation Method for Automatic Speech Recognition. In: *Proc. Interspeech 2019*. [S.l.: s.n.], 2019. p. 2613–2617. Citado na página 51.
- PATIL, A. K. et al. An adaptive approach for the reconstruction and modeling of as-built 3d pipelines from point clouds. *Automation in construction*, Elsevier, v. 75, p. 65–78, 2017. Citado 3 vezes nas páginas 8, 22 e 23.
- PEREIRA, D. et al. Industrial solid waste strategies and policies for the south region of rio de janeiro state. *Industrial Solid Waste Strategies and Policies for the South Region of Rio de Janeiro State*, 2007. Citado na página 15.
- PIASSA, P. V. Utilização de deep learning para detecção de anomalias em redes. 2019. Citado na página 43.
- QU, H. et al. Recent advances of continual learning in computer vision: An overview. *arXiv preprint arXiv:2109.11369*, 2021. Citado 2 vezes nas páginas 34 e 35.
- REDMON, J.; FARHADI, A. *YOLO9000: Better, Faster, Stronger*. 2016. Citado na página 45.
- REDMON, J.; FARHADI, A. *Yolov3: An incremental improvement*. 2018. Citado na página 45.

- REN, P. et al. A novel squeeze yolo-based real-time people counting approach. *International Journal of Bio-Inspired Computation*, v. 16, p. 94, 2020. Citado na página 45.
- REN, S. et al. Faster r-cnn: Towards real-time object detection with region proposal networks. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, v. 39, p. 1137–1149, 2017. Citado 3 vezes nas páginas 43, 44 e 53.
- ROBERTO, R. et al. Geometrical and statistical incremental semantic modeling on mobile devices. *Computers & Graphics*, Elsevier, v. 84, p. 199–211, 2019. Citado 3 vezes nas páginas 18, 19 e 23.
- SANTOS, P. V. et al. Metodologia para análise de imagens de baixa resolução, para definição de mub (mapa urbano básico) para apoio às concessionárias de distribuição. Universidade Federal de Goiás, 2018. Citado na página 35.
- SHAO, S. et al. Objects365: A large-scale, high-quality dataset for object detection. In: *Proceedings of the IEEE/CVF international conference on computer vision*. [S.l.: s.n.], 2019. p. 8430–8439. Citado na página 52.
- SHORTEN, C.; KHOSHGOFTAAR, T. M. A survey on image data augmentation for deep learning. *Journal of big data*, SpringerOpen, v. 6, n. 1, p. 1–48, 2019. Citado na página 51.
- SIMONYAN, K.; ZISSERMAN, A. Very deep convolutional networks for large-scale image recognition. 2014. Citado na página 39.
- SPOITI, A. C. Classificação de vias através de imagens aéreas usando deep learning. Universidade Federal de Uberlândia, 2018. Citado na página 41.
- SZEGEDY, C. et al. Going deeper with convolutions. *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2015. Citado 2 vezes nas páginas 37 e 39.
- SZELISKI, R. Computer vision: Algorithms and applications. *Choice Reviews Online*, v. 48, p. 48–5140–48–5140, 2011. Citado 2 vezes nas páginas 34 e 36.
- TAKAHASHI, R.; MATSUBARA, T.; UEHARA, K. Data augmentation using random image cropping and patching for deep cnns. *IEEE Transactions on Circuits and Systems for Video Technology*, IEEE, v. 30, n. 9, p. 2917–2931, 2019. Citado na página 52.
- TZUTALIN. *Labellmg. Git code*. 2015. Disponível em: <<https://github.com/tzutalin/labellmg>>. Citado 2 vezes nas páginas 49 e 50.
- UPRETI, A. Convolutional neural network (cnn): a comprehensive overview. *International Journal of Multidisciplinary Research and Growth Evaluation*, p. 488–493, 2022. Citado 2 vezes nas páginas 37 e 38.
- VARGAS, A. C. G.; PAES, A.; VASCONCELOS, C. N. Um estudo sobre redes neurais convolucionais e sua aplicação em detecção de pedestres. In: *Proceedings of the XXIX Conference on Graphics, Patterns and Images*. [S.l.: s.n.], 2016. p. 1–4. Citado na página 38.

WANG, G. et al. Trc□yolo: a real□time detection method for lightweight targets based on mobile devices. *IET Computer Vision*, v. 16, p. 126–142, 2021. Citado 3 vezes nas páginas 16, 44 e 45.

WU, X. et al. Virtual data augmentation method for reaction prediction. *Scientific Reports*, Nature Publishing Group UK London, v. 12, n. 1, p. 17098, 2022. Citado na página 51.

XUE, Z.; ZHANG, L. Multiscale object detection method for track construction safety based on improved yolov5. *Mathematical Problems in Engineering*, v. 2022, p. 1–10, 2022. Citado na página 16.

YANG, C. Application of improved target detection algorithm in sports robot motion behavior. *International Conference on Mathematics, Modeling, and Computer Science (MMCS2022)*, 2023. Citado 2 vezes nas páginas 16 e 44.

YANG, S.; ANJIE, J. Recognition of oil and gas reservoir space based on deep learning. *E3S Web of Conferences*, v. 267, p. 01038, 2021. Citado na página 39.

YANG, x.; ZHANG, h. Foreign object debris detection based on improved yolov5 algorithm. *International Conference on Image, Signal Processing, and Pattern Recognition (ISPP 2023)*, 2023. Citado na página 16.

YU, C.; JI, F.; XUE, J. Cutting plane based cylinder fitting method with incomplete point cloud data for digital fringe projection. *IEEE Access*, IEEE, v. 8, p. 149385–149401, 2020. Citado 2 vezes nas páginas 21 e 22.

ZHANG, M. et al. Application of adversarial sample attack in aerial photo identification of transport vehicle. *DEStech Transactions on Engineering and Technology Research*, 2020. Citado na página 46.

ZHONG, J.; CHEN, J.; MIAN, A. Dualconv: dual convolutional kernels for lightweight deep neural networks. *IEEE Transactions on Neural Networks and Learning Systems*, v. 34, p. 9528–9535, 2023. Citado na página 45.

ZHOU, L.; ZHANG, L.; KONZ, N. Computer vision techniques in manufacturing. *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, IEEE, v. 53, n. 1, p. 105–117, 2022. Citado na página 34.

# ANEXO A – hyp.VOC.yaml

```
lr0: 0.00334 # Taxa de aprendizado inicial
lrf: 0.15135 # Taxa de aprendizado final
momentum: 0.74832 # Momento utilizado no otimizador
weight_decay: 0.00025 # Peso da penalização L2 (regularização)
warmup_epochs: 3.3835 # Número de épocas para o aquecimento inicial do
treinamento
warmup_momentum: 0.59462 # Momento durante o aquecimento
warmup_bias_lr: 0.18657 # Taxa de aprendizado do viés durante o aquecimento
box: 0.02 # Peso da coordenada da caixa no cálculo da perda
cls: 0.21638 # Peso da classificação no cálculo da perda
cls_pw: 0.5 # Peso da classificação positiva nas caixas (positivo fraco)
obj: 0.51728 # Peso da existência do objeto na caixa no cálculo da perda
obj_pw: 0.67198 # Peso da existência positiva do objeto nas caixas (positivo
fraco)
iou_t: 0.2 # Limiar de interseção sobre união (IoU) para considerar uma
detecção verdadeira
anchor_t: 3.3744 # Limiar de IoU para ajustar automaticamente o tamanho das
âncoras
fl_gamma: 0.0 # Parâmetro gamma para a função Focal Loss
hsv_h: 0.01041 # Alteração máxima no canal de matiz na transformação de cores
HSV
hsv_s: 0.54703 # Alteração máxima no canal de saturação na transformação de
cores HSV
hsv_v: 0.27739 # Alteração máxima no canal de valor na transformação de cores
HSV
degrees: 0.0 # Alteração máxima na rotação da imagem
translate: 0.04591 # Alteração máxima na translação da imagem
scale: 0.75544 # Alteração máxima na escala da imagem
shear: 0.0 # Alteração máxima na distorção angular da imagem
perspective: 0.0 # Alteração máxima na perspectiva da imagem
flipud: 0.0 # Probabilidade de virar a imagem verticalmente
fliplr: 0.5 # Probabilidade de virar a imagem horizontalmente
mosaic: 0.85834 # Probabilidade de aplicar a técnica de mosaico nas imagens de
entrada
mixup: 0.04266 # Probabilidade de aplicar a técnica de mixup no treinamento
copy_paste: 0.0 # Probabilidade de aplicar a técnica de copy-paste no
treinamento
anchors: 3.412 # Número de âncoras no modelo YOLO
```