

**UNIVERSIDADE FEDERAL DE ITAJUBÁ
PROGRAMA DE PÓS-GRADUAÇÃO EM
ENGENHARIA DE PRODUÇÃO**

**Modelo baseado em gêmeos digitais e visão computacional para
dar suporte ao gerenciamento de inventários no setor varejista**

Renzo Paranaíba Mesquita

Itajubá, dezembro de 2025

**UNIVERSIDADE FEDERAL DE ITAJUBÁ
PROGRAMA DE PÓS-GRADUAÇÃO EM
ENGENHARIA DE PRODUÇÃO**

Renzo Paranaíba Mesquita

**Modelo baseado em gêmeos digitais e visão computacional para
dar suporte ao gerenciamento de inventários no setor varejista**

Tese submetida ao Programa de Pós-Graduação
em Engenharia de Produção como parte dos
requisitos para obtenção do Título de Doutor em
Ciências em Engenharia de Produção.

Área de Concentração: Engenharia de
Produção

Orientador: Prof. Dr. Fabiano Leal

Coorientador: Prof. Dr. José Antonio de
Queiroz

**Dezembro de 2025
Itajubá**

**UNIVERSIDADE FEDERAL DE ITAJUBÁ
PROGRAMA DE PÓS-GRADUAÇÃO EM
ENGENHARIA DE PRODUÇÃO**

Renzo Paranaíba Mesquita

**Modelo baseado em gêmeos digitais e visão computacional para
dar suporte ao gerenciamento de inventários no setor varejista**

Tese submetida à banca examinadora em 10 de
dezembro de 2025.

Banca examinadora:

Prof. Dr. Jorge Muniz Junior (UNESP)

Prof. Dr. Carlos Henrique dos Santos (UNIFAL)

Prof. Dr. José Arnaldo Barra Montevechi (UNIFEI)

Prof. Dr. Matheus Brendon Francisco (UNIFEI)

Prof. Dr. Fabiano Leal (orientador)

Prof. Dr. José Antonio de Queiroz (coorientador)

**Itajubá
2025**

DEDICATÓRIA

Aos meus queridos pais, Ediêta e Raul, à minha querida irmã, Hanna, e à minha amada esposa, Frâneli Eduarda

AGRADECIMENTOS

De todos os desafios enfrentados ao longo do desenvolvimento deste trabalho, posso dizer que escrever esta seção foi uma das partes mais desafiadoras. É neste momento que facilmente cometemos o equívoco de esquecer de agradecer a algum amigo que, direta ou indiretamente, foi de grande ajuda nesta jornada. Considero-me o pivô dela, mas sem o apoio, conhecimento e o incentivo de muitos, nada teria acontecido. Por isso, já gostaria de iniciar agradecendo a todos aqueles que, em algum momento, estiveram ao meu lado ao longo desta trajetória.

Não poderia, contudo, deixar de explicitar meus agradecimentos àqueles que claramente formaram a minha base até aqui. Começo destacando meus estimados mestres, Guilherme Marcondes (graduação), Enzo Seraphim (mestrado) e Fabiano Leal (doutorado). Que privilégio tê-los tido por perto! Vocês me ensinaram, não apenas por palavras, mas também por meio de ações, o verdadeiro significado do que é ser um professor. Como disse Sir Isaac Newton: “Se me vi mais longe, foi por estar de pé sobre os ombros de gigantes”. Minha mais profunda gratidão aos senhores. Sinto-me honrado em poder compartilhar o que me ensinaram com as futuras gerações.

Aos meus queridos pais, Ediêta e Raul, à minha querida irmã, Hanna, e à minha amada esposa, Frâneli Eduarda, todo meu amor e carinho. Ao longo desta jornada, não houve um único momento em que eu não tenha pensado em vocês. Nos períodos mais desafiadores, foi o amor e apoio incondicional de vocês que me fizeram seguir em frente.

Meu muito obrigado às queridas instituições INATEL e UNIFEI, aos órgãos de fomento CNPq, CAPES e FAPEMIG, pelo suporte, pelo incentivo e pelo ambiente propício à pesquisa e à inovação, que contribuíram para o desenvolvimento deste trabalho e que seguem contribuindo para a contínua busca pelo conhecimento, em prol de uma sociedade mais justa e comprometida com o avanço científico.

Por fim, não poderia deixar de agradecer àquele que olha por todos nós. Talvez ainda não tenhamos (e talvez nunca teremos) a capacidade de compreendê-lo plenamente, mas sabemos que sua bondade e seu amor infinitos estão sempre nos confortando, principalmente nos momentos mais difíceis. Obrigado, senhor Deus, pelo dom da vida, por iluminar meus passos e por me permitir chegar até aqui. Que minha formação se torne instrumento da tua obra, para ajudar, sobretudo, aqueles que mais precisam.

EPÍGRAFE

“Se me vi mais longe, foi por estar de pé sobre os ombros de gigantes”

Isaac Newton

RESUMO

A Modelagem e Simulação é um dos métodos mais largamente empregados na área de Pesquisa Operacional. Ela vem evoluindo a cada dia, saltando de modelos com tempo de vida limitado e que não realizam a troca de dados de forma recorrente com o mundo real, para abordagens mais dinâmicas, de uso constante, e que permite a troca de dados em tempo real ou quase tempo real entre sistemas reais e virtuais. Este novo tipo de modelo, de uso contínuo, e de intensa troca de dados entre o mundo real e virtual é chamado de Gêmeo Digital. Esta abordagem tem sido usada mais intensamente no setor manufatureiro, entretanto, seu uso tem se expandido para outros setores de grande relevância na economia, como é o caso do setor varejista. Existem várias operações presentes em estabelecimentos deste setor que podem ser beneficiadas pela aplicação dos Gêmeos Digitais, principalmente quando aliada a técnicas baseadas em Inteligência Artificial, como é o caso da Visão Computacional. O controle de inventários, por exemplo, se destaca como uma das operações mais fundamentais e estratégicas deste setor, mas que pouco tem usufruído de abordagens modernas como Gêmeos Digitais e Visão Computacional para prover um suporte mais automatizado e orientado a dados aos seus *stakeholders*. De modo a contribuir com a literatura e tal lacuna existente, este trabalho visa apresentar um modelo computacional para apoiar o gerenciamento de inventários em gôndolas presentes em estabelecimentos do setor varejista, integrando técnicas modernas de Visão Computacional com o dinamismo provido pelos Gêmeos Digitais. Ele busca realizar o mapeamento, acompanhamento e previsão do consumo de produtos presentes em estabelecimentos por meio do cálculo de um índice de esvaziamento que leva em consideração espaços vazios (*gaps*) encontrados nas suas prateleiras. Além disso, visa integrar ferramentas de *software* gratuitas e dispositivos (câmeras) de fácil acesso, ser modular, expansível e adaptável para produtos específicos de cada estabelecimento. Testes práticos realizados buscaram evidenciar a robustez do modelo em diferentes cenários com variações de *zoom*, iluminação e *layouts*, sua capacidade de detectar oclusões humanas em diversos níveis e movimentos, e sua utilidade prática em cenários de consumo e reposição ao oferecer tanto *feedback* descritivo quanto preditivo em tempo real, atuando assim como uma alternativa orientada a dados para auxílio na tomada de decisões.

Palavras-chave: Modelagem, Simulação, Gêmeos Digitais, Visão Computacional, Varejo, Controle de Inventários, Prateleiras.

ABSTRACT

Modeling and Simulation is one of the most widely used methods in the field of Operations Research. It has been evolving daily, moving from models with a limited lifespan and that do not exchange data regularly with the real world, to more dynamic, constantly used approaches that allow real-time or near real-time data exchange between real and virtual systems. This new type of model, with continuous use and intense data exchange between the real and virtual worlds, is called a Digital Twin. This approach has been used most extensively in the manufacturing sector, however, its use has expanded to other highly relevant sectors of the economy, such as the retail sector. Several operations within this sector can benefit from the application of Digital Twins, especially when combined with Artificial Intelligence-based techniques, such as Computer Vision. Inventory control, for example, stands out as one of the most fundamental and strategic operations in this sector, but it has rarely benefited from modern approaches such as Digital Twins and Computer Vision to provide more automated and data-driven support to its stakeholders. To contribute to the literature and fill this gap, this work aims to present a computational model to support shelf inventory management in retail establishments, integrating modern Computer Vision techniques with the dynamism provided by Digital Twins. It seeks to map, monitor, and predict product consumption in establishments by calculating an emptying rate that takes into account empty spaces (gaps) found on their shelves. Furthermore, it aims to integrate free software tools and easily accessible devices (cameras), being modular, expandable, and adaptable to the specific products of each establishment. Practical tests conducted sought to demonstrate the model's robustness in different scenarios with variations in zoom, lighting, and layouts, its ability to detect human occlusions at different levels and movements, and its practical utility in consumption and replenishment scenarios by offering both descriptive and predictive feedback in near-real time, thus acting as a data-driven alternative to aid decision-making.

Keywords: *Modeling, Simulation, Digital Twin, Computer Vision, Retail, Inventory Management, Shelves.*

LISTA DE ABREVIATURAS

| | |
|---------|---|
| ANN | <i>Artificial Neural Networks</i> |
| AP | <i>Average Precision</i> |
| ARIMA | <i>Autoregressive Integrated Moving Average</i> |
| BD | Banco de Dados |
| BDA | <i>Big Data Analytics</i> |
| BIM | <i>Building Information Modeling</i> |
| BPMN | <i>Business Process Model Notation</i> |
| CN | Computação em Nuvem |
| CNN | <i>Convolutional Neural Network</i> |
| COCO | <i>Common Objects in Context</i> |
| CPS | <i>Cyber-physical Systems</i> |
| CSV | <i>Comma Separated Values</i> |
| Cobot | <i>Collaborative Robot</i> |
| DBM | <i>Deep Boltzmann Machines</i> |
| DBN | <i>Deep Belief Networks</i> |
| EER | <i>Enhanced Entity-Relationship Diagram</i> |
| EXIF | <i>Exchangeable Image File Format</i> |
| FL | Fuzzy Logic |
| GD | Gêmeo Digital |
| GDs | Gêmeos Digitais |
| GPS | <i>Global Positioning System</i> |
| I4.0 | Indústria 4.0 |
| IA | Inteligência Artificial |
| IoT | <i>Internet of Things</i> |
| IoU | <i>Intersection over Union</i> |
| KNNs | <i>K-nearest Neighbors</i> |
| LIDAR | <i>Light Detection and Ranging Sensor</i> |
| M&S | Modelagem e Simulação |
| MAE | <i>Mean Absolute Error</i> |
| mAP | <i>Mean Average Precision</i> |
| MAPE | <i>Mean Absolute Percentage Error</i> |
| MD | Modelo Digital |
| ML | <i>Machine Learning</i> |
| MP | Métodos de Previsão |
| NumPy | <i>Numerical Python</i> |
| OpenCV | <i>Open-Source Computer Vision Library</i> |
| PE | Pesquisa Exploratória |
| POS | <i>Point of Sales</i> |
| QR Code | <i>Quick Response Code</i> |
| RBM | <i>Restricted Boltzmann Machines</i> |
| RFID | <i>Radio-frequency Identification</i> |
| RL | Revisão da Literatura |
| RMSE | <i>Root Mean Square Error</i> |
| RNA | Redes Neurais Artificiais |
| RQs | <i>Research Questions</i> |

| | |
|-----------|---|
| RSL | Revisão Sistemática da Literatura |
| SARIMA | <i>Seasonal Autoregressive Integrated Moving Average</i> |
| SARIMAX | <i>Seasonal Autoregressive Integrated Moving Average with Exogenous Factors</i> |
| SBA | Simulação Baseada em Agentes |
| SCDT | <i>Supply Chain Digital Twins</i> |
| SD | Sombra Digital |
| SED | Simulação de Eventos Discretos |
| SGBDR | Sistema Gerenciador de Bancos de Dados Relacionais |
| shelf-OOS | <i>Shelf Out-of-stock</i> |
| SVM | <i>Support Vector Machines</i> |
| VC | Visão Computacional |
| YOLO | <i>You Only Look Once</i> |

LISTA DE FIGURAS

| | |
|---|----|
| Figura 1 - Conceito de um possível GD para dar suporte à tomada de decisões em uma loja de conveniências inspirado nas três dimensões propostas por Qi e Tao (2018) | 11 |
| Figura 2 - Classificação, Localização e Segmentação em objeto único ou múltiplos objetos . | 14 |
| Figura 3 - Arquitetura da versão clássica do modelo YOLO | 17 |
| Figura 4 - Evolução do modelo YOLO ao longo dos anos | 18 |
| Figura 5 - Passos realizados nas etapas de Planejamento e Busca/Triagem | 23 |
| Figura 6 - Tipos de estudos encontrados na RSL | 24 |
| Figura 7 - Aplicação direta ou indireta dos estudos encontrados no setor varejista..... | 25 |
| Figura 8 - Atividades mais populares do setor varejista que já buscaram explorar do uso dos GDs..... | 26 |
| Figura 9 - Conceitos mais populares relacionados à implementação de GDs no setor varejista | 27 |
| Figura 10 - Dispositivos de <i>hardware</i> mais populares relacionados a implementação de GDs no setor varejista..... | 28 |
| Figura 11 - Softwares mais populares relacionados a implementação de GDs no setor varejista | 29 |
| Figura 12 - Tipos de dados usados nos estudos de caso..... | 30 |
| Figura 13 - Classificação desta pesquisa | 39 |
| Figura 14 - Etapas de um projeto de M&S..... | 41 |
| Figura 15 - Problemática deste trabalho | 43 |
| Figura 16 – Exemplo ilustrativo da falta de produtos em prateleiras com fundo aparente | 45 |
| Figura 17 - Modelo Conceitual Geral do GD proposto | 46 |
| Figura 18 – Diagrama BPMN do processo de captura, apresentação e previsão do índice de esvaziamento de produtos em prateleiras | 49 |
| Figura 19 - Diagrama BPMN do processo de calibração | 51 |
| Figura 20 - Diagrama BPMN do processo de reconhecimento de elementos..... | 53 |
| Figura 21 – <i>Mockup</i> do <i>Dashboard</i> proposto com painéis <i>Ranking</i> , Visão Geral e Previsão Padrão dos Índices de Esvaziamento..... | 55 |
| Figura 22 – Painéis disponíveis atualmente no <i>dashboard</i> do GD e possíveis painéis futuros que poderão ser adicionados..... | 56 |

| | |
|--|----|
| Figura 23 - Gôndola e câmara utilizadas no experimento e a interação entre elas..... | 58 |
| Figura 24 - Fusão Hierárquica dos Modelos utilizados neste trabalho..... | 63 |
| Figura 25 - Exemplo de imagens presentes no treinamento dos modelos (a) Generalista e (b) Especialista | 65 |
| Figura 26 - Localização dos pontos P1 e P2 nas caixas delimitadoras de produtos localizados nos extremos da gôndola | 70 |
| Figura 27 - Organização do BD Relacional do GD proposto..... | 74 |
| Figura 28 - Cálculo do índice de esvaziamento individual de cada <i>gap</i> detectado..... | 75 |
| Figura 29 - Exemplo do painel Ranking - Índice de Esvaziamento | 77 |
| Figura 30 - Exemplo de painel Visão Geral - Índice de Esvaziamento..... | 78 |
| Figura 31 - Painel de Previsão Padrão aguardando dados da Janela de Observação | 78 |
| Figura 32 - Painel de Previsão Padrão após executar uma previsão em um determinado momento | 79 |
| Figura 33 - Exemplo de informações presentes no painel Acompanhamento Individual - Índice de Esvaziamento | 80 |
| Figura 34 - Tela de configurações do <i>dashboard</i> (símbolo de engrenagem) | 80 |
| Figura 35 - Gráficos apresentando métricas de treinamento/validação do Modelo Generalista ao longo das épocas | 83 |
| Figura 36 - Gráficos apresentando métricas do treinamento/validação do Modelo Especialista ao longo das épocas | 83 |
| Figura 37 - Matriz de confusão das classes do Modelo Generalista | 85 |
| Figura 38 - Matriz de confusão das classes do Modelo Especialista | 86 |
| Figura 39 - Organização do Teste 1 | 87 |
| Figura 40 - Combinação Z1,I2 com exemplos de <i>layouts</i> pequenos, médios, grandes e mistos, respectivamente | 88 |
| Figura 41 - Combinação Z2,I2 com exemplos de layouts pequenos (canto superior esquerdo), médios (canto superior direito), grandes (canto inferior esquerdo) e mistos (canto inferior direito), respectivamente | 89 |
| Figura 42 – TE Média de detecção para as combinações de <i>zoom</i> e iluminação | 91 |
| Figura 43 - Sinal de oclusão apresentado pelo GD no painel de Acompanhamento Individual | 93 |
| Figura 44 - Exemplos de oclusões causadas no Teste 2..... | 94 |
| Figura 45 - Ações tomadas em cada momento T do Teste 3..... | 96 |

| | |
|--|----|
| Figura 46 - Tendência de consumo crescente até o momento T6 (30min). | 98 |
| Figura 47 – Abastecimento ilustrado no momento T10. | 99 |

LISTA DE QUADROS

| | |
|--|----|
| Quadro 1 - Resumo de benefícios e oportunidades relacionados ao uso de GDs do setor varejista | 31 |
| Quadro 2 - Resumo dos principais desafios relacionados ao uso de GDs do setor varejista ... | 32 |
| Quadro 3 - Resultado da PE sobre a integração de GDs e técnicas VC no setor varejista. Os tipos de estudos podem ser Conceito (C), Revisão da Literatura (RL) ou Estudo de Caso (EC) | 34 |
| Quadro 4 - Resultado da PE sobre a integração de GDs e o algoritmo YOLO, independente da área de aplicação. Os tipos de estudos podem ser Conceito (C), Revisão da Literatura (RL) ou Estudo de Caso | 35 |
| Quadro 5 - Principais componentes lógicos utilizados na implementação do GD..... | 62 |
| Quadro 6 – Bases de dados externas de produtos de onde foram extraídas imagens para treinamento do modelo Generalista..... | 64 |

LISTA DE TABELAS

| | |
|--|----|
| Tabela 1 - Parâmetros de treinamentos dos modelos de VC Generalista e Especialista..... | 67 |
| Tabela 2 - Tempo de treinamento e número de épocas em que os modelos atingiram os melhores resultados de detecção | 68 |
| Tabela 3 - Dados presentes no arquivo de calibração do GD com valores de exemplo..... | 72 |
| Tabela 4 - Métricas de desempenho dos Modelos Generalista e Especialista sobre os seus respectivos conjuntos de teste..... | 84 |
| Tabela 5 - Métricas de desempenho das classes individuais de ambos os modelos para os dados de teste | 84 |
| Tabela 6 - Resultados do Teste 1 | 90 |
| Tabela 7 - Métricas da classe <i>empty_space</i> para cada combinação de <i>zoom</i> e iluminação..... | 91 |
| Tabela 8 - Resultados do Teste 2..... | 94 |
| Tabela 9 - Cronograma de eventos do Teste 3 e resultados. | 96 |
| Tabela 10 - Parâmetros do <i>auto_arima</i> utilizado no painel de Previsão Padrão..... | 97 |

SUMÁRIO

| | |
|--|-----------|
| 1. INTRODUÇÃO | 1 |
| 1.1 Contexto..... | 1 |
| 1.2 Justificativas..... | 2 |
| 1.3 Problema de pesquisa..... | 3 |
| 1.4 Objetivos geral e específicos | 4 |
| 1.5 Contribuições | 5 |
| 1.6 Condições de contorno da pesquisa | 6 |
| 1.7 Estrutura da tese..... | 7 |
| 2. REFERENCIAL TEÓRICO | 9 |
| 2.1 Considerações iniciais..... | 9 |
| 2.2 Gêmeos Digitais (GDs)..... | 9 |
| 2.3 Visão Computacional (VC) | 13 |
| 2.4 Métodos de Previsão (MP) | 18 |
| 2.4.1 O método ARIMA | 20 |
| 2.5 Revisão Sistemática da Literatura (RSL): Gêmeos Digitais no Setor Varejista..... | 20 |
| 2.5.1 Planejamento | 21 |
| 2.5.2 Busca/Triagem..... | 22 |
| 2.5.3 Análise/Síntese | 23 |
| 2.5.4 Apresentação | 24 |
| 2.6 Pesquisa Exploratória (PE): Integração Visão Computacional e Gêmeos Digitais... | 34 |
| 3. MÉTODO DE PESQUISA | 38 |
| 3.1 Considerações iniciais..... | 38 |
| 3.2 Classificação da pesquisa..... | 38 |
| 3.3 Definição do método: Modelagem e Simulação (M&S) | 39 |
| 3.3.1 Descrição do método | 40 |
| 4. APLICAÇÃO | 43 |
| 4.1. Considerações iniciais | 43 |
| 4.2 Concepção | 43 |
| 4.2.1 BPMN geral do GD | 47 |
| 4.2.2 Subprocesso de calibração do GD | 50 |

| | | |
|-----------|--|------------|
| 4.2.3 | Subprocesso de reconhecimento de elementos..... | 52 |
| 4.2.4 | Proposta de <i>Front-end (Dashboard)</i> | 54 |
| 4.3 | Implementação..... | 57 |
| 4.3.1 | Materiais..... | 58 |
| 4.3.2 | Métodos..... | 62 |
| 4.3.2.1 | Preparação e Treinamento dos Modelos de Visão Computacional..... | 62 |
| 4.3.2.2 | Processo de Calibração do GD..... | 69 |
| 4.3.2.3 | Processo de Reconhecimento de Elementos..... | 72 |
| 4.3.2.4 | Implementação e acesso ao <i>Dashboard</i> | 76 |
| 4.4 | Análise..... | 81 |
| 4.4.1 | Métricas dos Modelos de Visão Computacional..... | 81 |
| 4.4.2 | Testes Práticos e Resultados..... | 86 |
| 4.4.2.1 | Teste Prático 1..... | 86 |
| 4.4.2.2 | Teste Prático 2..... | 92 |
| 4.4.2.3 | Teste Prático 3..... | 95 |
| 5. | CONCLUSÕES..... | 100 |
| 5.1 | Síntese dos Resultados..... | 100 |
| 5.2 | Limitações da Pesquisa..... | 103 |
| 5.3 | Recomendações para trabalhos futuros..... | 105 |
| | REFERÊNCIAS..... | 108 |
| | APÊNDICE A - Publicações..... | 125 |
| i. | Periódicos..... | 125 |
| ii. | Congressos..... | 125 |
| | APÊNDICE B – Manual de Adaptação e Treinamento de Novos Produtos no GD..... | 126 |
| | APÊNDICE C – Organização do Banco de Dados Relacional do GD Proposto..... | 152 |

1. INTRODUÇÃO

1.1 Contexto

Desde o seu advento nos anos 60, o uso da Modelagem e Simulação (M&S) computacional vem se popularizando, partindo de um método inicialmente acessível a alguns poucos *experts*, para uma abordagem padrão no acervo de ferramentas dos tomadores de decisão. Ela é usada principalmente para ajudá-los a entender melhor a dinâmica de diferentes processos e facilitar a resolução de problemas diários encontrados em diferentes negócios. A M&S é um dos métodos mais largamente empregados na área de Pesquisa Operacional e tem se tornado ainda mais acessível nos últimos anos, uma vez que os dispositivos de *hardware* necessários para utilizá-la têm se tornado mais rápidos e baratos, e as aplicações de *software* têm oferecido cada vez mais recursos necessários para se representar virtualmente as complexidades de objetos e ambientes físicos (Nance; Sargent, 2002; Law, 2014).

Além disso, a própria forma de se realizar M&S em si também vem se modificando, saltando de modelos com tempo de vida limitado e que não realizam qualquer troca de dados frequente entre os mundos real e virtual, para abordagens mais dinâmicas e de uso constante, em que os dados fluem entre os mundos real e virtual em tempo real ou quase real (Kritzinger *et al.*, 2018). Esta nova forma de simulação está associada ao conceito de Gêmeo Digital (GD), e de acordo com Gartner (2016), ela é considerada uma das tendências tecnológicas que podem trazer ainda mais precisão para os tomadores de decisão.

Existe uma grande quantidade de áreas que podem se beneficiar dos Gêmeos Digitais (GDs) (Augustine, 2020; Fatorachian; Kazemi, 2020; Santos *et al.*, 2021). O varejo, por exemplo, é um setor que graças à ascensão da Internet das Coisas (IoT, do inglês *Internet of Things*), agora está mais apto a tirar vantagem de tecnologias como câmeras e sensores de diferentes naturezas capazes de gerar dados úteis para se alimentar ferramentas analíticas e modelos mais dinâmicos. Isso consequentemente facilita a criação dos GDs, que podem estrategicamente auxiliar os varejistas a melhorarem a performance de suas operações diárias de uma forma mais constante e realista (Augustine, 2020; Maïzi; Bendavid, 2019).

O GD é uma abordagem relativamente nova e apenas nesta última década pesquisas voltadas a este assunto vieram à tona. É importante destacar que a maioria das soluções baseadas em GDs propostas até o momento são voltadas ao setor manufatureiro (Maïzi; Bendavid, 2019; Santos *et al.*, 2021). Elas aparecem de forma mais frequente neste setor pois

as tecnologias da Indústria 4.0 (I4.0) formam a base necessária para que isso aconteça de forma mais natural. O varejo é um modelo de negócio já bem conhecido, mas que se encontra em constante expansão (Mou; Robb; Dehoratius, 2018). Porém, a literatura ainda carece de pesquisas que ilustrem como estas tecnologias e os GDs podem impactar positivamente outros contextos, como é o caso de várias operações presentes no setor varejista (Mesquita; Leal; Queiroz, 2024).

A fim de contribuir com novas abordagens que buscam a melhoria de processos do setor varejista, este trabalho propõe um modelo para gerenciamento de inventários no setor varejista baseado em GDs e Visão Computacional (VC) para melhor dar suporte a processos logísticos dentro destes locais, mais particularmente, no que tange ao monitoramento e abastecimento de gôndolas, estruturas compostas por múltiplas prateleiras e fundamentais em estabelecimentos deste setor.

Além de representar o estado atual do sistema físico por meio da VC, o GD proposto também incorpora métodos de previsão que utilizam o histórico de consumo detectado para estimar estados futuros do inventário, tornando o modelo não apenas descritivo, mas também preditivo. Maiores detalhes sobre a justificativa, problema de pesquisa, objetivos e outras informações relevantes podem ser encontradas nos subtópicos seguintes.

1.2 Justificativas

Como mencionado, grande parte das aplicações de GDs até o momento se concentram no escopo da indústria manufatureira (Maïzi; Bendavid, 2019; Santos *et al.*, 2021). Isto conseqüentemente abre um leque de oportunidades para realização de pesquisas que ilustrem como esta abordagem pode impactar positivamente outros setores da economia, como é o caso deste trabalho, que visa ilustrar seus impactos no controle de inventários do varejo, uma operação carente de pesquisas no setor (Mesquita; Leal; Queiroz, 2024).

Devido ao maior acesso e popularização de algoritmos e dispositivos sensíveis de diversas naturezas, a compreensão dos estados das estruturas presentes dentro dos estabelecimentos varejistas já é uma realidade. A VC, por exemplo, é uma técnica que, por meio do uso de algoritmos de Inteligência Artificial (IA) e dispositivos populares como computadores pessoais e câmeras, tem avançado consideravelmente nos últimos anos, e possibilitado a geração de dados estratégicos sobre o que acontece dentro destes estabelecimentos (Ramalho *et al.*, 2019; Schwartz *et al.*, 2021).

Porém, a academia ainda carece de pesquisas que ilustrem de forma clara qual o impacto do uso de dados produzidos por técnicas de VC ao se alimentar modelos de simulação baseados em GDs e como isso poderia apoiar certas operações do varejo, que em sua grande parte, ainda são realizadas utilizando abordagens não sistematizadas e/ou não orientada a dados, como é o caso da gerência de inventários em gôndolas.

Além disso, apesar de gôndolas serem estruturas fundamentais e estratégicas no varejo para organização e disponibilização de produtos aos clientes, nenhum estudo que envolve o uso de GDs e que lida com a falta de produtos nelas foi encontrado, conforme ilustra os resultados das pesquisas apresentadas nos subtópicos 2.5 e 2.6 deste trabalho. A falta de produtos nas prateleiras de uma gôndola pode impactar nos custos operacionais logísticos de um estabelecimento, e principalmente, causar insatisfação do cliente (Chuang, 2018).

A justificativa para realização deste estudo é buscar compreender como um modelo baseado em GD e VC poderia proporcionar um suporte diferenciado orientado a dados aos estabelecimentos, buscando principalmente compreender quais prateleiras precisam ser carregadas em determinado momento se baseando em um índice de esvaziamento atual e futuro delas, proposto por este trabalho.

1.3 Problema de pesquisa

Com o advento da IoT, maior popularização de *tags*/sensores/câmeras de todos os tipos e a chegada das redes de alta velocidade (5G), a capacidade dos estabelecimentos de coletarem dados e “sentirem” em tempo real ou quase tempo real o que está acontecendo dentro de seus espaços está se tornando cada vez maior. Enquanto várias pesquisas têm examinado como pontos de venda (POS, do inglês *Point of Sales*), caixas automáticos e a Internet têm mudado o varejo, pouco ainda se sabe sobre os impactos que tecnologias emergentes podem contribuir para atividades deste setor (Mou; Robb; Dehoratius, 2018; Chuang, 2018).

A logística aplicada ao varejo é uma área largamente estudada. Uma atividade importante dentro deste contexto, mas que ainda carece de pesquisas que melhor compreendam seus impactos dentro de um estabelecimento, é o problema de falta de produtos nas prateleiras (*shelf-OOS* - do inglês *Shelf Out-of-stock problem*) (Chuang, 2018). Por exemplo, o custo operacional associado com a busca de produtos no estoque para recarregá-las é impactante, e pode ser responsável por até 48% dos custos operacionais logísticos em uma cadeia de suprimentos do varejo (Mou; Robb; Dehoratius, 2018). O Walmart, gigante do setor, admitiu que possui problemas de shelf-OOS e já sofreu uma perda de em média 3 bilhões de dólares

por conta de ineficiência no processo logístico de recarregamento de suas prateleiras (Mou; Robb; Dehoratius, 2018).

Em supermercados e lojas de departamentos, apesar de serem modelos de negócio muito populares deste setor, são praticamente nulas as pesquisas que buscam compreender como abordagens emergentes, tais como os GDs, poderiam impactar positivamente seus processos logísticos. Em muitos estabelecimentos, por exemplo, colaboradores ainda são responsáveis por, de tempo em tempo, checarem se produtos estão faltando em prateleiras. Já foi identificado que este tempo de checagem tem grande influência na disponibilidade de produtos, satisfação do cliente e desempenho nas vendas (Chuang, 2018). Em dias de estabelecimentos movimentados, pode acontecer de faltar produtos nas prateleiras, e os funcionários não saberem que isso aconteceu. Geralmente acreditam em suas experiências ou intuições para definirem os critérios e realizarem o procedimento de reabastecimento.

O conceito de GD, integrado à VC, assume papel central no enfrentamento do problema de *shelf out-of-stock* neste trabalho, ao buscar viabilizar uma representação digital dinâmica das prateleiras baseada em dados visuais coletados continuamente no ambiente físico. Essa abordagem busca prover um monitoramento em tempo real, realizar a identificação automática de lacunas (*gaps*) nas prateleiras e oferecer suporte proativo ao reabastecimento, buscando reduzir intervenções desnecessárias e priorizando situações críticas. Além disso, ao incorporar dados históricos e modelos preditivos, o GD busca ampliar sua atuação para além do caráter reativo, buscando antecipar faltas e promovendo maior disponibilidade de produtos.

Ademais, em contraste com soluções complexas e proprietárias para automatização dos estabelecimentos varejistas, como as da Amazon Go e similares citadas por exemplo por Szabó-Szentgróti, Rámháp e Kézai (2023), o GD proposto oferece uma alternativa mais acessível, escalável e de uso geral para auxílio no controle de inventários, pois busca utilizar câmeras convencionais, *software* aberto e arquitetura modular, reduzindo custos e facilitando adaptação a diferentes produtos e contextos, sem dependência de tecnologias fechadas.

1.4 Objetivos geral e específicos

O objetivo geral deste trabalho é propor um modelo baseado em GD e VC, fazendo uso de tecnologias acessíveis e que visa dar suporte em tempo real ao gerenciamento de inventários em prateleiras presentes em estabelecimentos varejistas. O modelo busca orientar os *stakeholders* sobre quais prateleiras merecem maior atenção no processo de recarregamento.

Para isso, se baseia em um índice de esvaziamento que leva em consideração os espaços vazios (*gaps*) identificados nas prateleiras dos produtos monitorados.

Este objetivo geral se desdobra nos seguintes objetivos específicos:

- Identificar o potencial uso dos GDs dentro do varejo, por meio da realização de uma Revisão Sistemática da Literatura (RSL), a fim de justificar a sua aplicabilidade neste setor.
- Propor uma abordagem capaz de quantificar os espaços vazios (*gaps*) detectados pelo GD em cada um dos produtos monitorados, no caso, o cálculo de seus índices de esvaziamento.
- Realizar testes práticos para avaliar a abordagem proposta, incluindo detecção sob diferentes condições de *zoom*, iluminação e *layout*, análises de oclusão e verificação da operação do algoritmo de previsão.
- Propor um guia que permita pesquisadores e varejistas adaptarem o modelo para detecção, monitoramento e previsão de consumo de diferentes tipos de produtos presentes em prateleiras, buscando promover maior flexibilidade e escalabilidade da proposta.

1.5 Contribuições

O modelo proposto visa apresentar contribuições para os estabelecimentos varejistas como também para a literatura. Primeiramente, durante a pesquisa bibliográfica, foi identificado que a aplicação de GDs para melhoria de processos ligados ao controle de inventários ainda são minoria (Mesquita; Leal; Queiroz, 2024). Isso abre um leque de oportunidades para aplicação e avaliação desta abordagem nesta operação comumente presente nos estabelecimentos varejistas.

Trabalhos propostos por Fernández-Caramés e Fraga-Lamas (2018), Kaur e Kaur (2018), Schwartz *et al.* (2021), identificados na RSL sobre o potencial uso dos GDs no setor varejista (subtópico 2.5 deste trabalho), ilustram como o advento e popularização da IoT tem permitido aos estabelecimentos varejistas elevarem seus controles de inventários a outro nível, facilitando a criação de sistemas cyber-físicos que possam contribuir na melhoria destas operações. Porém, nenhuma aplicação prática que ilustre claramente os benefícios providos pelos GDs para controle de inventários foi apresentada por eles ou por outros autores.

Trabalhos que integram os conceitos de GD e VC também são minoria. Resultados da Pesquisa Exploratória (PE) sobre a integração de GDs com algoritmos de VC (subtópico 2.6 deste trabalho), independente da aplicabilidade, ilustram que já houve a tentativa de se integrar os dois conceitos. Porém, nenhuma delas está relacionada ao controle de inventários ou oferece um guia de como adaptar a solução para outros contextos semelhantes.

Sendo assim, o ineditismo deste trabalho se encontra em: 1) Identificar o potencial uso dos GDs dentro do setor varejista por meio da realização e publicação de uma RSL, uma vez que nenhum trabalho que aborde este tema de forma ampla foi encontrado na literatura; 2) propor um modelo computacional baseado em GDs e VC, de caráter prático, modular e expansível, utilizando *hardware* e *software* acessíveis, que opera em tempo real e utiliza técnicas de previsão, para dar suporte ao gerenciamento de inventários em prateleiras presentes em estabelecimentos varejistas, que possa ser adaptado para reconhecimento de diferentes tipos de produtos presentes nas gôndolas dos estabelecimentos, e que possa estimar o consumo destes produtos por meio do cálculo de um índice de esvaziamento que leva em consideração os espaços vazios (*gaps*) encontrados nas prateleiras.

As etapas do modelo serão detalhadas para que os pesquisadores possam utilizá-lo sem maiores dificuldades, colaborando para a expansão de estudos acadêmicos sobre a aplicação dos GDs no controle de inventários. Além disso, especialistas em M&S e gestores de estabelecimentos poderão compreender melhor suas etapas, utilizá-lo e aperfeiçoá-lo em estudos futuros.

1.6 Condições de contorno da pesquisa

São condições de contorno desta pesquisa:

- O modelo será capaz de trabalhar com gôndolas tradicionais que apresentem um processo organizado de oferta de produtos, não levando em consideração: *displays* que não apresentem esta característica, como por exemplo, cabides, caixas, baús etc ou gôndolas tradicionais, mas completamente desorganizadas.
- Os testes do modelo foram realizados em laboratório, em ambiente controlado, mas fazendo uso de uma gôndola profissional comumente empregada em estabelecimentos varejistas para exposição de produtos.
- O modelo no momento será limitado à identificação de um único formato e tipo de produto por prateleira.

- A solução proposta busca integrar recursos de *hardware* e *software* populares para se realizar o monitoramento do índice de esvaziamento dos produtos em prateleiras em tempo real.
- A investigação restringe-se inicialmente ao uso de câmeras com uma angulação capaz de captar a frente das gôndolas a serem monitoradas.
- O GD não leva em consideração o processo de contagem de produtos individuais, mas a identificação dos fundos (*gaps*) encontrados nas prateleiras para calcular um índice de esvaziamento dos produtos monitorados, e conseqüentemente, estimar a disponibilidade dos produtos;
- Este estudo não prevê a realização de um estudo de viabilidade financeira para o investimento de uma possível implantação do sistema (dispositivos necessários + modelo) em um ambiente real.

1.7 Estrutura da tese

Esta tese encontra-se estruturada em cinco capítulos, da seguinte forma:

- Neste capítulo (Capítulo 1) é apresentado o contexto no qual esta pesquisa está inserida e as suas justificativas. São apresentados também o problema de pesquisa, os objetivos gerais e específicos a serem alcançados, as contribuições e condições de contorno do trabalho.
- O Capítulo 2 abordará a revisão da literatura necessária para a elaboração desta tese, sendo apresentado mais em detalhes o conceito de GD, VC, Previsão e os resultados da RSL e PE realizadas. Para a RSL e PE, foram utilizados como fontes, principalmente, artigos de periódicos e artigos de congressos internacionais de bases de dados conceituadas como Scopus e Web of Science, e livros referências na área.
- O Capítulo 3 mostrará detalhes do método de pesquisa empregado, descrevendo as etapas que compõem este método.
- O Capítulo 4 apresenta detalhes da Concepção, Implementação e Análise dos resultados do GD proposto;
- O Capítulo 5 contém as conclusões do autor, que destaca a síntese dos resultados, as limitações da pesquisa e apresenta recomendações de trabalhos futuros.
- Em seguida, são apresentadas as referências utilizadas na pesquisa.
- Por fim, encontram-se os apêndices da tese, que apresentam em sequência: as publicações em periódicos e congressos durante a execução deste trabalho, um manual

de adaptação e treinamento de novos produtos no GD e maiores detalhes da organização do Banco de Dados (BD) Relacional utilizado na implementação do GD proposto.

2. REFERENCIAL TEÓRICO

2.1 Considerações iniciais

Este capítulo apresenta a fundamentação teórica para elaboração desta tese. Na seção 2.2 são apresentados conceitos importantes a respeito de GDs, desde a origem do termo até os requisitos necessários para construí-los. Na seção 2.3 são apresentados conceitos importantes sobre VC, como o princípio de funcionamento desta técnica, possibilidades de aplicação e algoritmos populares. Na seção 2.4 são discutidos conceitos fundamentais ligados a métodos de previsões, com destaque para o método *Autoregressive Integrated Moving Average* (ARIMA). Na seção 2.5 são apresentados os passos executados e resultados encontrados em uma RSL que ilustra a potencial aplicação dos GDs no setor varejista. Por fim, na seção 2.6, são apresentados os passos executados e resultados encontrados em pesquisas exploratórias que buscaram melhor ilustrar como vem sendo feita a integração entre GDs e algoritmos de VC, como o *You Only Look Once* (YOLO).

2.2 Gêmeos Digitais (GDs)

O termo GD foi proposto pela National Aeronautics and Space Administration (NASA) para se referir a “uma simulação integrada multifísica, multiescala e probabilística de um veículo ou sistema construído que utiliza os melhores modelos físicos disponíveis, atualizações de sensores, histórico de frota, etc., para espelhar a vida de seu gêmeo físico correspondente em operação” (Glaessgen; Stargel, 2012, p. 7). No entanto, graças à popularização de tecnologias disruptivas na última década, como IoT, Computação em Nuvem (CN), *Big Data Analytics* (BDA) e IA, recentemente este conceito ganhou força e permitiu que muitos tipos de negócios além da indústria aeroespacial incrementassem seus modelos a fim de criar representações digitais de objetos e sistemas do mundo real (Tao *et al.*, 2019; Vijayakumar, 2020).

De acordo com Wright e Davidson (2020), existem três partes importantes que devem ser levadas em consideração para se propor um GD: um modelo de objeto ou sistema físico real, um conjunto evolutivo de dados relacionados a ele e um meio de atualizar ou ajustar dinamicamente um modelo virtual de acordo com os dados. Diferente de uma abordagem que fornece uma representação momentânea de um objeto ou sistema em um momento específico com base em dados históricos, o GD pode estender a vida útil de um modelo, uma vez que os

dados mais recentes são usados para atualizar seus parâmetros a fim de fornecer uma nova forma de se fazer previsões (Beregi; Szaller; Kádár, 2018).

Isso não significa que o intervalo de tempo em que o GD é atualizado deva ser em tempo real. Depende do contexto da aplicação e, em alguns casos, o gêmeo deve ser atualizado em horas, em vez de segundos. O ponto mais importante desta abordagem é permitir a criação de modelos dinâmicos que descrevem o comportamento de objetos ou sistemas que mudam ao longo do tempo (Wright; Davidson, 2020).

Kritzinger *et al.* (2018) destacam que a caracterização de um GD é baseada no nível de integração entre os mundos físico e virtual que é colocada em prática. Existem três definições diferentes: Modelo Digital (MD), Sombra Digital (SD) e o GD. Um MD é simplesmente uma representação digital de um objeto ou sistema físico que não realiza troca alguma de dados com o mundo físico. Os dados necessários para se criar um modelo virtual são inseridos manualmente, e uma mudança no estado do objeto real não tem efeito direto no seu equivalente virtual, e vice-versa.

Diferente do MD, que utiliza uma abordagem estática, na SD existe um fluxo de dados unidirecional automatizado entre os mundos físico e virtual. A SD atua como uma atribuição pura de dados de estado de um ativo físico, representando suas condições e alterações em tempo real ou quase real. Essa estrutura ainda não busca compreender relações de causa e efeito nem gerar conhecimento, mas sim refletir o comportamento observado de forma contínua. Seu papel essencial é fornecer a base de dados estruturada sobre a qual o conhecimento organizacional pode ser construído, funcionando como o primeiro estágio do desenvolvimento de um GD, ao realizar a digitalização de processos reais que posteriormente serão interpretados e modelados (Schuh *et al.*, 2019; Bergs *et al.*, 2021).

Por fim, o GD é o modelo mais sofisticado dentre os citados anteriormente. É um sistema que possibilita fluxos de dados bidirecionais entre o mundo físico e virtual, ou seja, alterações no ambiente virtual também podem impactar o estado do seu equivalente físico. Os GDs podem operar sob duas possibilidades principais: de forma autônoma ou semiautônoma. No primeiro caso, ele é capaz de perceber, decidir e agir de modo independente, sem intervenção humana, ou seja, ele interpreta dados do ambiente físico, executa simulações no mundo virtual e implementa ações corretivas no mundo real de forma automática (Santos *et al.*, 2020; Agrawal *et al.*, 2023).

Já o GD semiautônomo caracteriza-se pela colaboração entre o mundo virtual e o operador humano, em que o GD atualiza-se em tempo real ou quase real, analisa dados e recomenda decisões, mas mantém o humano no ciclo decisório para validar ou executar as ações no ambiente físico. Assim, a diferença essencial entre ambos está no grau de autonomia: enquanto o autônomo atua de maneira independente, o semiautônomo funciona como um sistema de apoio inteligente, combinando automação e julgamento humano para garantir controle e confiabilidade nas decisões (Santos *et al.*, 2020; Agrawal *et al.*, 2023).

Em sinergia com Wright e Davidson (2020), Qi e Tao (2018) também afirmam que um GD é composto por três componentes, que são as entidades físicas no mundo físico, os modelos virtuais no mundo virtual e os dados conectados que ligam os dois mundos. Com base nesta afirmação, a Figura 1 retrata a ideia de uma possível aplicação de GDs em um tipo popular de varejo, como um supermercado.

O sistema real ou mundo físico compreende a própria loja no mundo real, que é formada por entidades físicas como clientes, funcionários, produtos e estruturas interligadas/sensoriais; o modelo virtual ou mundo virtual é responsável por modelar o comportamento real do sistema analisando, descrevendo, prevendo e otimizando-o em meios virtuais; e por fim, uma camada de dados conecta os dois mundos, permitindo a transmissão de dados em constante atualização do mundo físico para o mundo virtual e a transmissão de informações estratégicas (dados processados) na forma oposta para orientar a gestão na tomada de decisões.

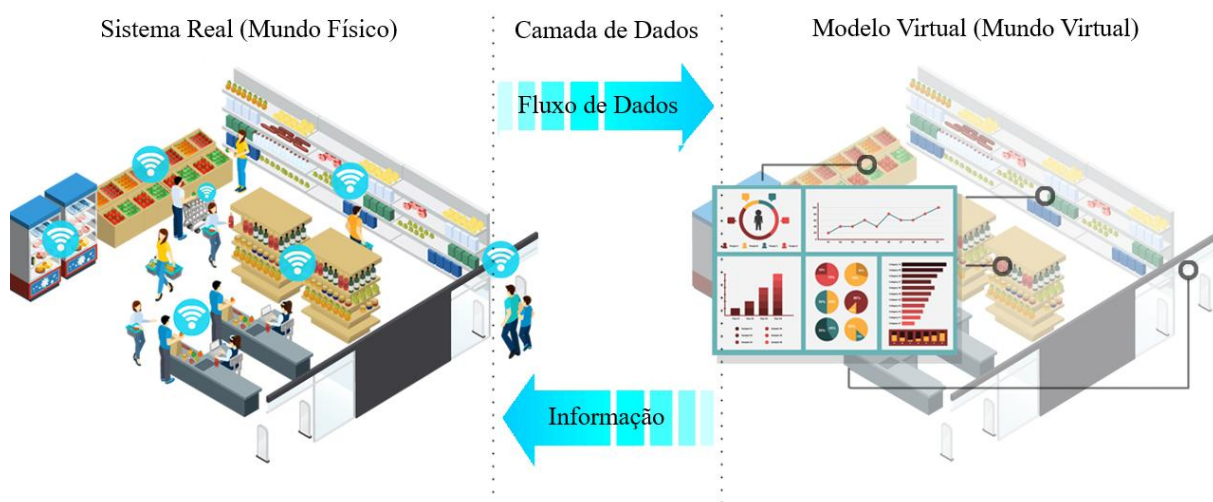


Figura 1 - Conceito de um possível GD para dar suporte à tomada de decisões em uma loja de conveniências inspirado nas três dimensões propostas por Qi e Tao (2018)
Fonte: elaborado pelo próprio autor

Os GDs consolidaram-se como um dos pilares da I4.0, pois materializam a integração entre o mundo físico e o digital por meio de modelos virtuais continuamente alimentados por dados reais oriundos de sistemas e processos. Essa conexão permite monitoramento em tempo real ou quase real, simulação de cenários, análise preditiva e otimização operacional ao longo de todo o ciclo de vida de produtos e sistemas. Ao viabilizar decisões baseadas em dados, maior automação, flexibilidade produtiva e integração entre tecnologias como IoT, IA e análise avançada de dados, os GDs passaram a desempenhar um papel central na transformação de sistemas em ambientes inteligentes, conectados e adaptativos, características fundamentais da I4.0 (Park *et al.*, 2024).

As tecnologias oferecidas pela I4.0 têm permitido que acadêmicos e empresas explorem o conceito de GD, especialmente na indústria manufatureira (Santos *et al.*, 2021). A literatura apresenta aplicações importantes deste conceito neste campo nos últimos anos (Vachálek *et al.*, 2017; Rabah *et al.*, 2018; Ding *et al.*, 2019; Ait-Alla *et al.*, 2020; Wang; Lee; Angelica, 2020), e apesar dos trabalhos focados em operações de manufatura ainda serem a maioria, existem novas áreas significativas que podem se beneficiar dos GDs, como agricultura, saúde e cidades inteligentes, demonstrando seu caráter transversal na I4.0 (Park *et al.*, 2024).

Na agricultura, são usados para monitoramento de culturas e otimização de recursos; na saúde, viabilizam modelos virtuais de pacientes para apoiar medicina personalizada e gestão hospitalar; e, em ambientes urbanos, permitem simular e gerenciar sistemas complexos como transporte e infraestrutura. Essas aplicações mostram que os GDs apoiam a tomada de decisão baseada em dados, otimização de recursos e planejamento de sistemas complexos em diferentes domínios além da produção industrial (Park *et al.*, 2024).

Segundo Augustine (2020), o varejo é outro exemplo de setor promissor onde o GD pode desempenhar um papel fundamental, uma vez que pode complementar a experiência do cliente, apoiar na manutenção de inventários e na administração de recursos humanos. Shankar *et al.* (2021) afirmam que embora muitas pesquisas tenham examinado como tecnologias como os pontos de venda, os caixas eletrônicos e a Internet mudaram o varejo, não se sabe muito sobre os impactos das tecnologias emergentes.

A I4.0 e as suas tecnologias podem ter um grande impacto nos processos do varejo. Uma combinação de *beacons*, sensores, câmeras, ferramentas de *analytics* e outras tecnologias habilitadoras de IoT podem permitir a “smartização” desta indústria (Fatorachian; Kazemi,

2021). Este processo forma a base para a criação dos GDs, consideradas por Gartner (2016) como uma das dez maiores inovações significativas da atualidade.

2.3 Visão Computacional (VC)

Os seres humanos possuem uma notável capacidade de perceber, sem esforço, aspectos tridimensionais em seu entorno, como discernir a forma dos objetos e identificar emoções a partir de expressões faciais em imagens. Investigadores em VC tiveram avanços significativos em técnicas matemáticas nos últimos anos para também permitir que as máquinas identifiquem formas e aparências a partir de imagens. Porém, alcançar uma compreensão semelhante à humana ainda continua sendo um desafio (Szeliski, 2022).

A VC aborda a complexa tarefa de descrever e reconstruir o mundo visual representado em imagens, incluindo forma, iluminação e distribuições de cores. Este esforço é complicado pela ambiguidade inerente aos dados visuais e pela necessidade de eliminar a ambiguidade de potenciais soluções utilizando modelos baseados na física, métodos probabilísticos ou aprendizagem automática. Embora a computação gráfica seja excelente na criação de cenas realistas, a VC enfrenta o desafio inverso, no caso, o de interpretar imagens do mundo real (Szeliski, 2022).

Os domínios da VC abrangem o reconhecimento de cenas ou objetos, detecção de objetos, rastreamento de vídeos, segmentação de objetos, estimativa de poses e movimentos, modelagem de cenas e restauração de imagens. Isso só é possível graças ao uso de modelos de IA voltados ao aprendizado profundo. Alguns modelos populares que podem ser utilizados em tarefas de VC são as Redes Neurais Convolucionais (CNNs, do inglês *Convolutional Neural Network*), Rede Profunda de Crenças (DBN, do inglês *Deep Belief Networks*), Máquina de Boltzmann (DBM, do inglês *Deep Boltzmann Machines*), Máquina de Boltzmann Restritas (RBM, do inglês *Restricted Boltzmann Machines*) e Autoencoders Empilhados (Murthy *et al.*, 2020; Diwan; Anirudh; Tembhrne, 2023).

Uma das tarefas mais importantes e populares da VC é a detecção de objetos. Ela pode ser considerada como a combinação de três subtarefas que são: a classificação, localização e a segmentação de uma imagem, conforme ilustrado pela Figura 2. A classificação de imagens envolve categorizar uma imagem ou objetos dentro dela em grupos predefinidos. Normalmente, esta tarefa é cumprida por meio do uso de técnicas de aprendizado de máquina supervisionado

¹ ou técnicas de aprendizado profundo², em que o modelo aprende a partir de um conjunto de dados rotulado³ e de tamanho considerável (Diwan; Anirudh; Tembhrne, 2023).

Abordagens comuns de aprendizado de máquina para essa finalidade abrangem as Redes Neurais Artificiais (RNAs), Máquinas de Vetores de Suporte (SVMs, do inglês *Support Vector Machines*), árvores de decisão e K-vizinhos mais próximos (KNNs, do inglês *K-nearest Neighbors*). No entanto, no domínio da aprendizagem profunda, as CNNs e seus derivados destacam-se como as arquiteturas predominantes para classificação de imagens e tarefas relacionadas (Diwan; Anirudh; Tembhrne, 2023).

Já a localização de objetos envolve a identificação da posição de um ou mais objetos dentro de uma imagem, colocando-os em caixas delimitadoras retangulares (no inglês, conhecidas como *Bounding Boxes*). Por outro lado, a segmentação de imagens envolve a divisão de uma imagem em vários segmentos, onde cada segmento pode abranger um objeto inteiro ou uma parte dele. A segmentação de imagens é frequentemente empregada para identificar objetos, linhas e curvas, como os limites de segmentos dentro de uma imagem. O objetivo principal da segmentação de imagens é fornecer uma representação significativa de uma imagem (Diwan; Anirudh; Tembhrne, 2023).



Figura 2 - Classificação, Localização e Segmentação em objeto único ou múltiplos objetos
Fonte: Adaptado de Shanmugamani (2017)

Muitas soluções de VC que empregam algoritmos de aprendizado de máquina e aprendizado profundo enfrentam vários desafios comuns. Em primeiro lugar, eles muitas vezes enfrentam dificuldades com o treinamento de imagens em múltiplas escalas, onde os detectores

¹ É quando um modelo é treinado com dados já rotulados (entrada e resposta correta) para aprender a prever saídas para novos dados.

² É uma área do ML que usa redes neurais com várias camadas para aprender automaticamente padrões complexos a partir de grandes volumes de dados.

³ Em ML, os dados rotulados são aqueles que vêm acompanhados de etiquetas ou rótulos que especificam a categoria ou classe à qual pertencem. Essas marcações são fundamentais para o treinamento de modelos de aprendizado supervisionado, pois oferecem ao algoritmo exemplos concretos de como os dados devem ser categorizados ou compreendidos.

treinados para resoluções de entrada específicas tendem a ter um desempenho inferior quando confrontados com imagens de diferentes escalas ou resoluções (Diwan; Anirudh; Tembhrne, 2023).

Além disso, desequilíbrios entre classes de imagens de primeiro e segundo plano podem impactar significativamente o desempenho do modelo, distorcendo a representação de diferentes categorias. Ademais, a detecção de objetos relativamente menores representa um desafio, pois os modelos treinados fazendo uso de objetos maiores tendem a ter um desempenho ruim quando aplicados em objetos menores (Diwan; Anirudh; Tembhrne, 2023).

A necessidade de grandes conjuntos de dados e poder computacional também se apresentam como uma barreira, uma vez que a detecção de objetos baseada em aprendizagem profunda requer extensos conjuntos de dados para treinamento, processos de anotação⁴ e recursos computacionais poderosos para processamento. Além disso, embora os modelos de aprendizagem profunda sejam superiores aos métodos tradicionais de aprendizagem automática, eles apresentam dificuldades quando avaliados em conjuntos de dados menores, levando a um desempenho inferior. Ademais, a localização imprecisa durante as previsões, causada pela inclusão de pixels de fundo e pela detecção de objetos semelhantes, introduz erros que comprometem a precisão do algoritmo (Liu *et al.*, 2020; Wang *et al.*, 2020; Diwan; Anirudh; Tembhrne, 2023).

Apesar dos desafios, a VC encontra ampla aplicação em vários cenários do mundo real, como reconhecimento facial, inspeção de máquinas, robótica, automação do varejo, avaliação de imagens médicas, veículos autônomos, construção de modelos 3D, vigilância, reconhecimento biométrico, entre outros. Essas aplicações demonstram o impacto prático desta técnica em diversos campos, mesmo enquanto os pesquisadores continuam a lidar com suas complexidades (Szeliski, 2022; Diwan; Anirudh; Tembhrne, 2023).

Dentre os modelos disponíveis para serem utilizados em tarefas de VC, aqueles baseados em CNNs têm se destacado como os mais populares nos últimos anos. A ascensão da AlexNet (Krizhevsky; Sutskever; Hinton, 2012), em 2012, serviu como um catalisador para pesquisadores em vários domínios da VC, incluindo classificação de imagens, localização, detecção de objetos, segmentação e muito mais. Essa ascensão desencadeou o desenvolvimento

⁴ Em ML, a anotação de dados refere-se à prática de associar etiquetas, categorias ou indicadores a conjuntos de dados, destacando características particulares ou dados relevantes para o treinamento de modelos supervisionados. Este processo pode envolver, por exemplo, identificar e marcar objetos em imagens, permitindo que os algoritmos aprendam a reconhecê-los com precisão.

de vários modelos de classificação e detecção de objetos de ponta, permitindo que o campo avançasse significativamente (Sultana; Sufian; Dutta, 2020; Murthy *et al.*, 2020).

Entre os modelos baseados em CNNs, o YOLO (Redmon *et al.*, 2016) se tornou um dos modelos mais largamente utilizados em VC. Grande parte disso se deve ao fato de ser um modelo de uso gratuito e por apresentar excelentes capacidades de detecção para aplicações em tempo real (Jiang *et al.*, 2022; Terven; Córdova-Esparza; Romero-González, 2023). O YOLO opera como uma rede de estágio único⁵, empregando uma CNN simples para prever diretamente as probabilidades de classe⁶ e as caixas delimitadoras das imagens de entrada. Em seu processo, o modelo particiona a imagem de entrada em um determinado número de grades.

Dentro de cada célula da grade, o modelo prevê um número predefinido de caixas delimitadoras junto com uma pontuação de confiança. Essa pontuação de confiança é determinada pela multiplicação da probabilidade de detecção de objetos e pela métrica Intersecção sobre União (IoU, do inglês *Intersection over Union*)⁷ entre as caixas delimitadoras previstas e reais. Caixas delimitadoras com probabilidades de classe que ultrapassam um limite predeterminado são então escolhidas para identificar a localização do objeto na imagem (Redmon *et al.*, 2016). A Figura 3 ilustra este processo.

Em 2023, a Ultralytics, criadora do YOLOv5, revelou o YOLOv8. Esta nova iteração da série YOLO introduziu cinco variações adaptadas a diferentes escalas: YOLOv8n (nano), YOLOv8s (pequeno), YOLOv8m (médio), YOLOv8l (grande) e YOLOv8x (extragrande). As principais diferenças entre estas variações estão em seu tamanho e complexidade. Embora modelos maiores e mais complexos ofereçam maior precisão, eles operam em um ritmo mais lento. Por outro lado, modelos menores e mais simples sacrificam alguma precisão para um desempenho mais rápido (Jocher, 2023).

Quando se trata de fazer uso de modelos de aprendizado de máquina, existem três metodologias principais: aproveitar um modelo pré-treinado, conduzir o treinamento desde o início ou realizar um aprendizado incremental. A utilização de um modelo pré-treinado implica na utilização de um modelo que já passou por treinamento em um conjunto de dados substancial adaptado a uma tarefa específica, como por exemplo, a classificação de certas classes de

⁵ Uma rede de estágio único realiza todas as etapas de detecção e classificação em um único passo, tornando-a mais rápida e eficiente em termos de processamento, embora possa sacrificar uma certa precisão em comparação com redes de estágio duplo.

⁶ Uma classe representa uma categoria particular de entidade visual, objeto ou fenômeno que um sistema de visão computacional é ensinado a discernir, reconhecer ou categorizar dentro de uma imagem ou vídeo.

⁷ É uma medida que avalia a precisão da sobreposição entre duas áreas delimitadas, geralmente caixas delimitadoras usadas para identificar objetos em uma imagem.

imagens. O modelo pré-treinado já assimilou a capacidade de identificar padrões e características distintas nos dados, que podem posteriormente ser aplicados a um novo conjunto de dados para se identificar classes de objetos semelhantes (Zhuang *et al*, 2021).

Já treinar um modelo do zero exige o seu treinamento em um conjunto de dados específico e adaptado, por exemplo, para identificar novos tipos de objetos. Este processo pode ser demorado, mas necessário em muitos casos. Por fim, pode-se realizar o processo de aprendizado incremental, ou seja, adicionar novas classes em um modelo já treinado, mas sem esquecer as classes antigas (Zhuang *et al*, 2021).

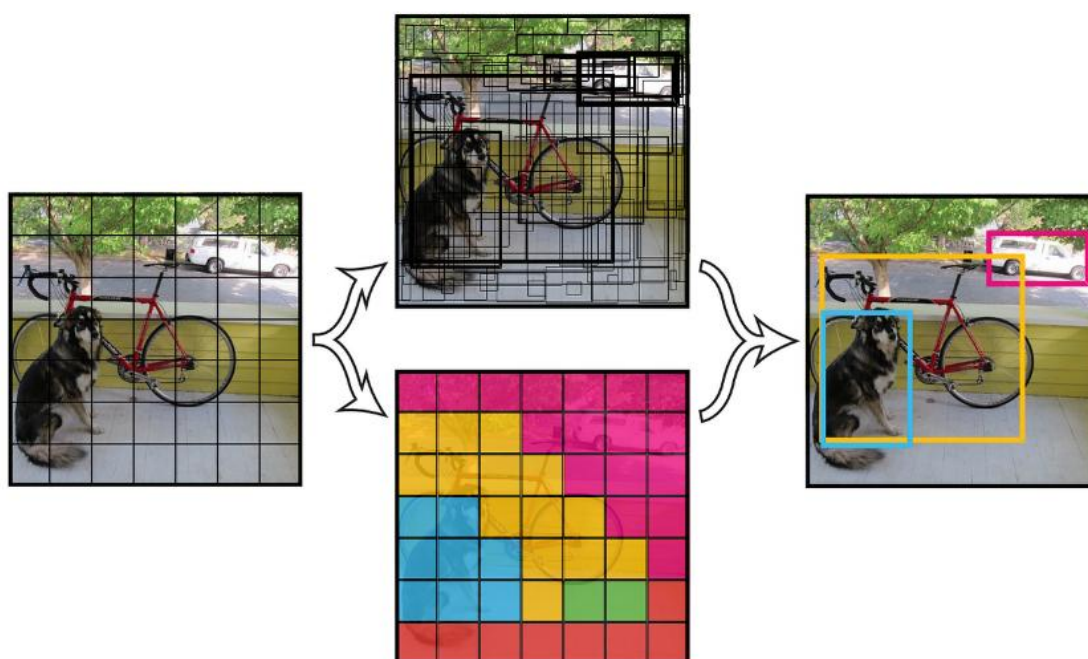


Figura 3 - Arquitetura da versão clássica do modelo YOLO
Fonte: Redmon *et al*. (2016)

O YOLOv8 já disponibiliza modelos pré-treinados para detecção, segmentação, identificação de poses e classificação de uma grande quantidade de objetos fazendo uso de conjunto de dados muito populares em VC, como por exemplo o *Common Objects in Context* (COCO) e o ImageNet. O COCO é conjunto de dados em grande escala projetado para detecção, segmentação e detecção de poses de objetos com mais de 200 mil imagens rotuladas.

Já o ImageNet é um conjunto de dados em grande escala para detecção de objetos e classificação de imagens com mais de 14 milhões de imagens e 20.000 categorias. Apesar da disponibilidade de modelos pré-treinados, é importante ressaltar que o YOLOv8 também pode

ser treinado com grandes conjuntos de dados customizados e ser executado em uma variedade de plataformas de *hardware*, de CPUs a GPUs. (Jocher; Chaurasia, 2023).

Outro grande motivo da popularidade do YOLO se deve a sua constante evolução, que, a cada dois anos ou menos, costuma apresentar uma nova versão com aperfeiçoamentos no processo de detecção em relação à versão anterior. A Figura 4 ilustra a evolução do YOLO desde a sua versão base em 2015 até o ano de 2023, ano em que foi lançada a versão utilizada neste trabalho, no caso, o YOLOv8. Detalhes a respeito das melhorias ocorridas entre as versões ao longo dos anos até a versão YOLOv8 podem ser encontradas em Hussain (2023).

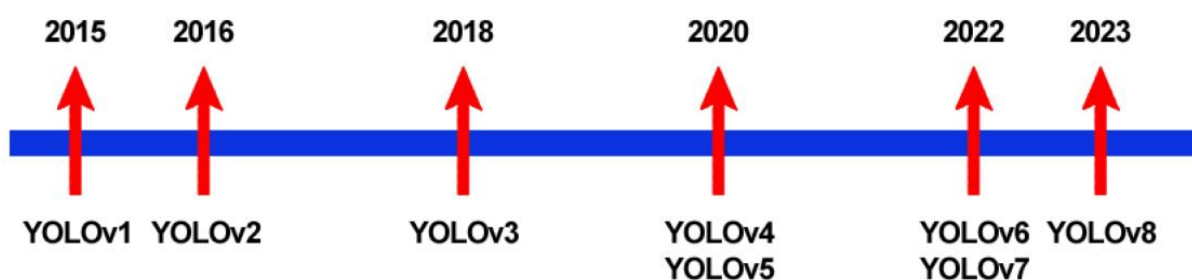


Figura 4 - Evolução do modelo YOLO ao longo dos anos
Fonte: Adaptado de Hussain (2023)

As diferentes versões do YOLO podem ser utilizadas como recursos estratégicos de VC para serem aplicadas em diversos setores da economia (Rane, 2023). No setor varejista, mais particularmente, o uso do YOLO já é uma realidade para dar suporte a diversas operações presentes neste setor, como por exemplo, na melhoria de sistemas de *self-checkout* (Shoman *et al.*, 2022; Vats; Anastasiu, 2023; Pham *et al.*, 2023), gerência de inventários (Tonioni; Serra; Di Stefano, 2018; Kumar; Simha; Agarwal, 2023; Oliveira *et al.*, 2024), comportamento do cliente (Nagnath *et al.*, 2020; Erlina; Fikri, 2023), entre outros.

2.4 Métodos de Previsão (MP)

Os Métodos de Previsão (MP) têm como principal objetivo reconhecer padrões de comportamento em séries históricas e projetar o comportamento futuro, ou ainda, identificar fatores causais que influenciam esse comportamento e extrapolá-lo. Eles são fundamentais para auxiliar no processo de tomada de decisão sobre o futuro (Petropoulos; Kourentzes; Nikolopoulos; Siemen, 2018).

O tema da previsão de demanda é atual e relevante para instituições de diversos segmentos de atuação. Os métodos e os modelos para prever a demanda de produtos, serviços, materiais e otimizar a utilização dos recursos organizacionais são relevantes para as empresas e o comércio. A gestão de estoques, por exemplo, é uma operação que pode ser bastante beneficiada por meio de tais previsões, principalmente fazendo-se uso de séries temporais (Ackermann; Sellitto, 2022).

Nas últimas décadas, ferramentas e recursos computacionais têm sido fortes aliados no processo de previsão. Aplicações como o Microsoft Excel, por exemplo, facilitam cálculos que envolvam vários valores, manipulando grandes volumes de dados de forma organizada. Além disso, várias linguagens de programação estão atualmente capacitadas para a realização de tais análises. A linguagem R, por exemplo, possui uma variedade de recursos destinados à preparação de dados, visualizações e modelagem de séries temporais (Krispin, 2019).

Outra linguagem que vem se destacando para análise de dados e previsões é o Python. Ela é uma linguagem de fácil entendimento, além de possuir uma extensa documentação. Suas bibliotecas científicas, como SciPy e NumPy, permitem a realização de operações vetoriais e matriciais de forma eficiente, além de serem compiladas em linguagens de baixo nível como C, proporcionando um desempenho superior em relação a outras linguagens (Subasi, 2020).

Os MP são basicamente divididos em dois tipos, sendo eles, qualitativos ou quantitativos. O primeiro tipo é considerado subjetivo, pois se baseia em estimativas e opiniões. Por outro lado, o segundo tipo faz uso de cálculos matemáticos para identificar padrões nos dados históricos e transpor este padrão para o futuro. Dentro dos modelos quantitativos existem os modelos causais, que consideram a relação entre variáveis diferentes (causa e efeito), e os modelos de séries temporais, que se baseiam em dados históricos da própria variável (Arvan; Fahimnia; Reisi; Siemsen, 2019).

Mais particularmente, para as séries temporais, existem diversos MP conhecidos. Dentre eles, o ARIMA se destaca pelo seu equilíbrio entre precisão, interpretabilidade e facilidade de implementação, especialmente para problemas de previsão univariados, lineares e de curto prazo, o que o torna uma ferramenta valiosa em várias aplicações (Kontopoulou; Panagopoulos; Kakkos; Matsopoulos, 2023). O subtópico seguinte discutirá mais em detalhes o princípio de funcionamento deste método.

2.4.1 O método ARIMA

A abreviação ARIMA significa *Autoregressive (AR) Integrated (I) Moving Average (MA)*. Também conhecido como modelo de Box-Jenkins fazendo uma alusão aos seus criadores, este método busca fazer uso de dados do passado para prever o futuro por meio da integração de duas técnicas: autocorrelação e médias móveis (Silva *et al.*, 2021).

Um modelo ARIMA tradicional ou não sazonal é classificado como um modelo ARIMA (p,d,q) , em que p significa a ordem do modelo autorregressivo, d o grau de diferenciação para deixar a série estacionária (característica fundamental para se aplicar um modelo ARIMA) e q é a ordem do modelo de média móvel (Şahinli, 2020).

Modelos ARIMA também são capazes de modelar séries que apresentam um componente sazonal. Neste caso, temos uma variação chamada de *Seasonal Autoregressive Integrated Moving Average (SARIMA)* ou ARIMA $(p, d, q) \times (P, D, Q)S$ em que o segundo componente faz referência à parte sazonal e S significa a duração de um ciclo sazonal (Jain; Sukhdeve; Gadia; Sahu; Verma, 2021).

Os modelos ARIMA e SARIMA podem ser aplicados apenas em séries temporais univariadas. Porém, muitos problemas podem apresentar algum tipo de relacionamento entre variáveis. Neste caso, deve-se usar uma variação do ARIMA mais adequada para esta situação, no caso, o modelo *Seasonal Auto-Regressive Integrated Moving Average with exogenous factors (SARIMAX)*.

Em suma, este modelo pode levar em consideração variáveis exógenas, ou seja, usar dados externos na previsão (Jain; Sukhdeve; Gadia; Sahu; Verma, 2021). O ARIMA é o método de previsão utilizado em um dos painéis presentes no *dashboard* do GD proposto. Maiores detalhes de implementação deste método fazendo-se uso da linguagem Python serão apresentados no subtópico 4.3.1 deste trabalho.

2.5 Revisão Sistemática da Literatura (RSL): Gêmeos Digitais no Setor Varejista

A fim de melhor identificar como vem sendo a aplicação do conceito de GDs no setor varejista, uma RSL inédita foi realizada e publicada sobre o assunto (Mesquita; Leal; Queiroz, 2024). Segundo Denyer e Tranfield (2009), uma RSL consiste em uma abordagem metodológica que busca localizar estudos já existentes, selecionar e avaliar suas contribuições,

analisar e sintetizar seus dados e relatar as evidências de forma a possibilitar conclusões claras sobre o que já é conhecido e o que ainda não foi explorado.

Diferente de uma Revisão de Literatura tradicional que permite o desenvolvimento de uma pesquisa sem seguir um padrão metodológico, a RSL exige o uso de uma metodologia clara, ou seja, o uso de diretrizes para se realizar todas as etapas da pesquisa, tornando o estudo replicável, científico e mais transparente (Denyer; Tranfield, 2009; Oliveira; Lima; Montevechi, 2016).

Portanto, a fim de avaliar o potencial uso dos GDs para dar suporte a operações do setor varejista, esta RSL foi estruturada seguindo os 4 estágios sugeridos por Oliveira, Lima e Montevechi (2016) para se conduzir este tipo de revisão:

- Planejamento: desenvolvimento de um protocolo de pesquisa;
- Busca/Triagem: seleção e avaliação da qualidade dos estudos;
- Análise/Síntese: tratamento e análise dos dados;
- Apresentação: relato e destaque das descobertas;

2.5.1 Planejamento

O primeiro passo foi formar uma equipe para desenvolver o estudo. A equipe envolvida na revisão consistiu em dois professores com conhecimentos em GDs e um estudante de doutorado. Em seguida, uma busca exploratória sobre o tema de pesquisa foi conduzida usando a base de dados Scopus.

De acordo com Baas *et al.* (2020), o Scopus é uma das maiores bases de dados científicas uma vez que ela certifica que apenas revistas, artigos de congresso e livros de alta qualidade estejam indexados nela, pois possui um rigoroso processo de seleção de conteúdo. A pesquisa foi conduzida no final de junho de 2024 e incluiu a palavra-chave “*Digital Twin*” seguido pelo operador lógico “AND” e finalizando com a palavra-chave “*Retail*”, respectivamente. O título, resumo e introdução foram analisados pelos autores.

Apesar das contribuições encontradas nos trabalhos desta análise exploratória, foi identificado uma carência de estudos que discutem de forma mais ampla o uso dos GDs no setor varejista, não apresentando aspectos como: categorização dos trabalhos explorando este conceito até o momento; principais operações do varejo que têm utilizado deste conceito; tecnologias que têm permitido a implantação deste conceito neste setor; quais as fontes de dados, janela de atualização e grau de autonomia dos trabalhos práticos; quais os benefícios,

oportunidades e desafios do uso de GDs para os varejistas. O propósito da RSL foi analisar melhor as perspectivas sobre a aplicação dos GDs no setor varejista a fim de contribuir com um estudo teórico atualizado sobre o assunto e que possa auxiliar acadêmicos e profissionais.

Baseado nas lacunas mencionadas, as seguintes perguntas de pesquisa (RQ, do inglês *Research Questions (RQ)*) foram formuladas:

RQ1: Quais os tipos de trabalhos foram encontrados até o momento sobre este assunto? (Conceitos, Revisões ou Casos de Estudo)

RQ2: Quais operações do varejo têm tirado vantagem desta abordagem?

RQ3: Quais tecnologias-chave tem permitido a aplicação de GDs no setor varejista?

RQ4: Quais os dados de entrada, janela de atualização e o grau de autonomia das soluções práticas encontradas?

RQ5: Quais são os principais benefícios, oportunidades e desafios de se aplicar GDs no setor varejista?

2.5.2 Busca/Triagem

Nesta etapa o primeiro passo foi identificar e selecionar as bases de dados da pesquisa. Três bases de dados estratégicas foram incluídas neste estudo: Scopus, Web of Science e IEEE Xplore. De acordo com um estudo realizado por Santos *et al.* (2021), outras diferentes nomenclaturas podem ser utilizadas para se referir a abordagens muito similares aos GDs, tais como “*Cyber-physical System*”, “*Real-time Simulation*”, “*Near Real-time Simulation*”, “*Symbiotic Simulation*”, “*Online Simulation*”, “*Data-driven Simulation*” e “*Semi-physical Simulation*”. É importante ressaltar que estas nomenclaturas estão ligadas a GDs baseados em simulação.

Portanto, o operador booleano “OU” foi usado para ser obter todas as possíveis combinações entre o termo “*Digital Twin*” e os termos mencionados, seguido pelo operador “AND” e a palavra-chave “*retail**”. O símbolo “*” é um coringa que permite a busca por palavras-chave que variam em uma palavra raiz. Por exemplo, palavras-chave como “*retailer*”, “*retailing*” e assim por diante também serão levadas em consideração nesta pesquisa graças ao uso do coringa “*”.

Além disso, os seguintes critérios também foram incluídos na pesquisa dos artigos nas bases de dados:

- Os termos devem estar presentes no título, resumo ou palavras-chave dos artigos.
- Os artigos devem estar escritos em inglês.
- Devem ser revistas ou artigos de congresso que passaram por uma revisão em pares.
- Os artigos devem ter sido publicados nos últimos 10 anos, considerando a data final como junho de 2024.

Após a primeira busca, 160 artigos foram encontrados, mas 43 eram redundantes. Logo, 117 artigos foram levados para a fase de triagem a fim de que seus resumos pudessem ser avaliados em mais detalhes buscando identificar aqueles mais alinhados com os objetivos deste trabalho. Esta pesquisa foca em artigos que direto ou indiretamente discutem ou aplicam o conceito de GDs em lojas físicas ou virtuais. Por isso, após a fase de triagem, 65 artigos únicos foram selecionados para que pudessem ser lidos na íntegra. A Figura 5 resume os passos realizados na etapa de Planejamento seguida pelas etapas de Busca/Triagem.

Pesquisa Exploratória

Base Utilizada: Scopus;
Objetivo: Visão geral do uso de GDs no setor varejista;
Crêterios: Título, Resumo ou Palavras-chave que contenham: ("Digital Twin") AND ("Retail");
Resultados: carência de estudos que discutem o estado da arte sobre o uso de GDs voltados ao setor varejista de forma mais ampla;

Triagem

Bases Utilizadas: Scopus, Web of Science e IEEE Xplore;
Objetivo: Analisar artigos que direta ou indiretamente discutam ou apliquem o conceito GD em lojas físicas ou online de varejo;
Crêterios: Ler atentamente o título e o resumo de todos os artigos encontrados na etapa anterior;
Resultados: 65 artigos únicos foram selecionados para leitura na íntegra.

Busca na Literatura

Bases Utilizadas: Scopus, Web of Science e IEEE Xplore;
Objetivo: Analisar publicações que de alguma forma discutem ou apliquem o conceito de GD no setor varejista;
Crêterios: Título, Resumo ou Palavras-chave que contenham: ("Digital Twin" OR "Cyber-physical System" OR "Symbiotic Simulation" OR "Online Simulation" OR "Data-driven Simulation" OR "Real-time Simulation" OR "Near real-time simulation" OR "Semi-physical simulation") AND ("retail");
Crêterios Adicionais: os artigos devem ser escritos em inglês; devem ser periódicos ou anais de conferências revisados por pares; e devem ter sido publicados nos últimos dez anos, considerando a data final como 21 de junho de 2024;
Resultados: 160 artigos foram encontrados (incluindo redundâncias);

Figura 5 - Passos realizados nas etapas de Planejamento e Busca/Triagem

Fonte: elaborado pelo próprio autor

2.5.3 Análise/Síntese

A análise e síntese das descobertas foram realizadas utilizando uma planilha do Microsoft Excel. Isso facilitou a extração de informações dos artigos lidos, com cada artigo sistematicamente sendo registrado na planilha juntamente com as respostas às RQs. Posteriormente, foi empregada estatística descritiva para se analisar os dados.

2.5.4 Apresentação

Sobre os tipos de trabalhos já publicados sobre o assunto (RQ1), 37 deles ou por volta de 57% já tentaram de alguma forma aplicar o conceito de GD de forma prática no setor varejista. Dentre os trabalhos, 28 deles ou 43% são revisões da literatura ou estudos teóricos, como mostrado na Figura 6. Este estudo busca apresentar uma visão geral da potencial aplicação dos GDs no setor varejista, enquanto as revisões de literatura encontradas discutem seu uso em operações específicas do varejo.

A maioria dos artigos (46 ou 71% do total) discutem ou falam sobre soluções que impactam diretamente estabelecimentos varejistas, sejam estabelecimentos físicos ou virtuais. Outros 19 artigos ou 29% discutem como suas propostas podem impactar indiretamente o setor, conforme apresentado na Figura 7. Uma vez que o varejo é um dos últimos passos em uma cadeia de suprimentos do fornecedor ao consumidor, a aplicação dos GDs em fases que precedem o setor pode também impactá-lo positivamente.

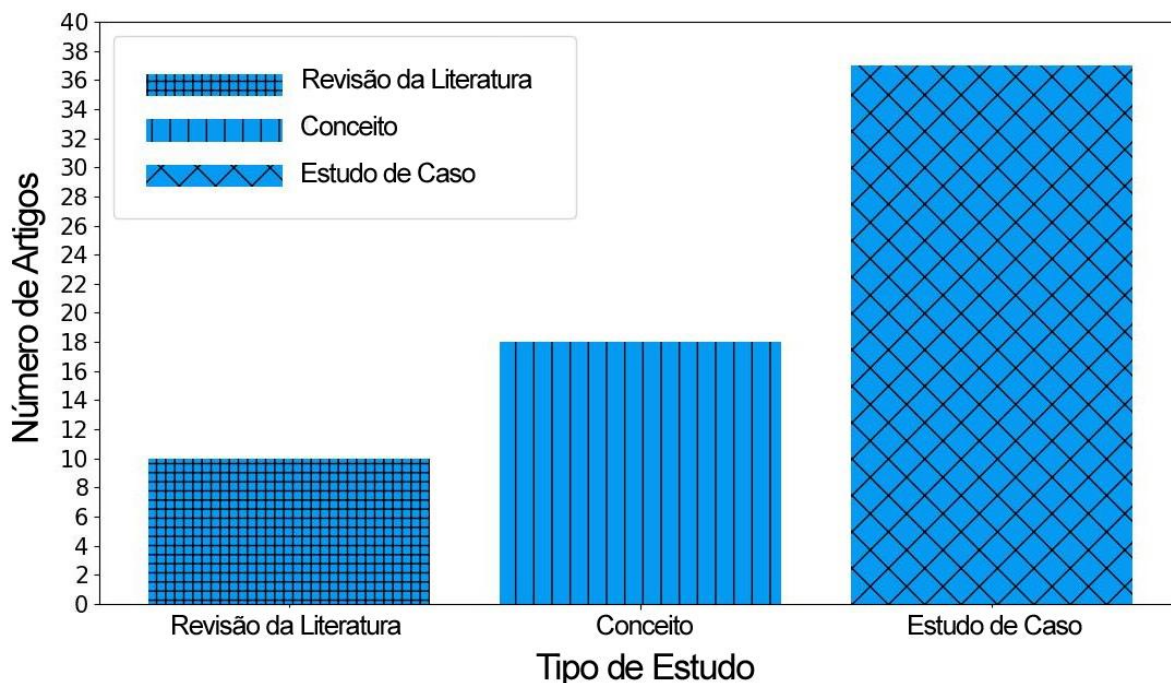


Figura 6 - Tipos de estudos encontrados na RSL

Fonte: elaborado pelo próprio autor

Entre os artigos analisados, os autores identificaram 8 diferentes operações na indústria varejista em que os GDs já têm sido discutidos ou aplicados, conforme ilustra a Figura 8. É possível notar que operações voltadas à Transformação Digital dos estabelecimentos, Previsão do Comportamento dos Clientes e Cadeia de Suprimentos são as que mais têm tirando proveito desta abordagem até o momento.

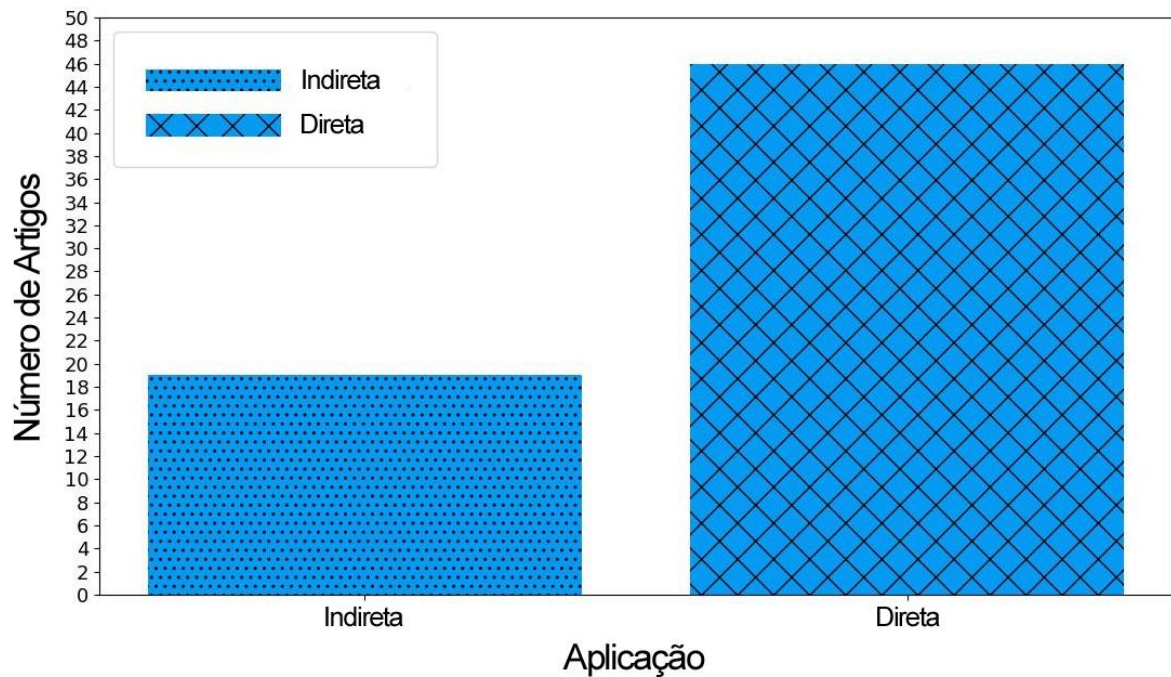


Figura 7 - Aplicação direta ou indireta dos estudos encontrados no setor varejista
 Fonte: elaborado pelo próprio autor

Apesar da manutenção de equipamentos ser uma das operações menos exploradas, o trabalho proposto por Rossoni, Fumagalli e Colombo (2020) abre portas para as possibilidades que os GDs podem trazer para auxiliar no diagnóstico e prognóstico de vários equipamentos presentes em estabelecimentos varejistas, poupando-os de danos desnecessários e consequentemente evitando qualquer tipo de inconveniência que possa atrapalhar a experiência do cliente.

Em relação a atividades voltadas a operações de *checkout* (Antczak; Weron; Zabawa, 2020; Hauser *et al.*, 2017; Mykoniatis *et al.*, 2020), segurança (Gupta *et al.*, 2020; Rachana Harish *et al.*, 2021, Sasikumar *et al.*, 2024), gerência de inventários (Schwartz *et al.*, 2021; Fernández-Caramés; Fraga-Lamas, 2018; Kaur; Kaur, 2018, Liu *et al.*, 2024; Pous; Chindemi; Alajami, 2023) e movimentação de materiais (Culler; Long, 2016; Knoch *et al.*, 2022; Elbouzidi *et al.*, 2022, Dennis *et al.*, 2024; Rzydzik; Kroczeck, 2024; Ashrafian; Pedersen, 2023; Elbouzidi *et al.*, 2023), apesar de serem atividades fundamentais presentes em estabelecimentos varejistas, ainda são poucas as pesquisas de GDs que têm buscado aperfeiçoá-las.

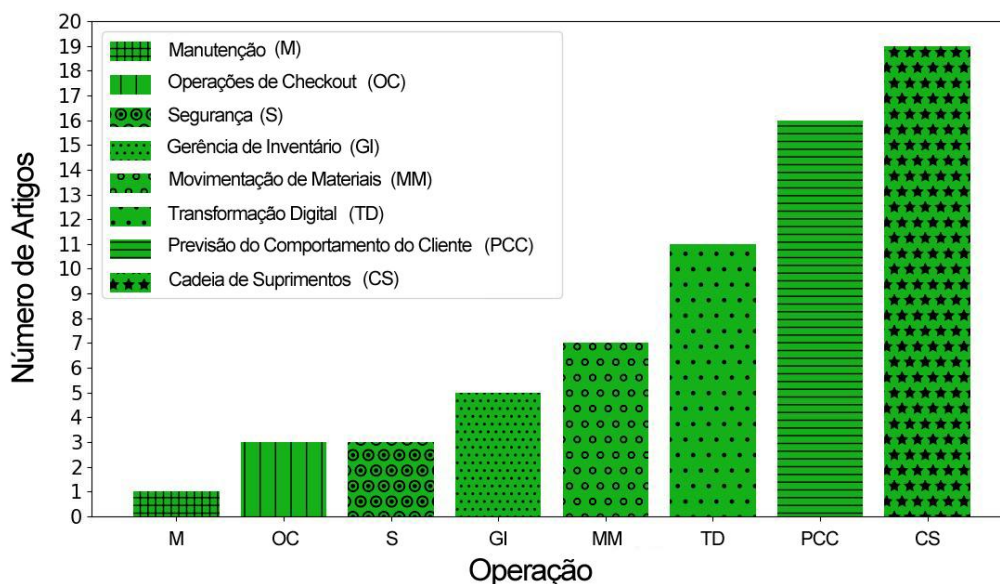


Figura 8 - Atividades mais populares do setor varejista que já buscaram explorar o uso dos GDs
 Fonte: elaborado pelo próprio autor

A maioria das pesquisas estão concentradas nas operações de transformação digital dos estabelecimentos (Hashmi; Mo; Beckett, 2021; Bakharev *et al.*, 2021; Janhofer *et al.*, 2020, Trappey *et al.*, 2018, White; Schmidt; Golparvar-Fard, 2014; Singh *et al.*, 2022; Schopf; Dresse; Matthes, 2022; Leusmann *et al.*, 2023; Kumpel; Beetz, 2023; Kumpel *et al.*, 2023), previsão do comportamento do cliente (Kaur *et al.*, 2020; Mao *et al.*, 2018; Yensabai *et al.*, 2023; Hanke *et al.*, 2018; Lee *et al.*, 2017; Hata *et al.*, 2018; Barnes, 2022; Durana; Musova; Cuțitoi, 2022; Shinkar; Bogiri, 2019; Adikari *et al.*, 2020; Hung; Wang; Chang, 2020; Chan; Lau; Fan, 2020; Maïzi; Bendavid, 2019; Maïzi; Bendavid, 2021; Kim *et al.*, 2024; Shams *et al.*, 2024) e principalmente em operações voltadas ao aperfeiçoamento das cadeias de suprimentos (Fatorachian; Kazemi, 2020; Karaköse; Yetiş, 2017; Sun *et al.*, 2019; Maheshwari; Kamble, 2022; Matsuda *et al.*, 2019; Matsuda *et al.*, 2020; Nishi *et al.*, 2020; Matsuda *et al.*, 2021; Kamiebisu *et al.*, 2022; Pal; Kant, 2020; Defraeye *et al.*, 2021; Shoji *et al.*, 2022a; Shoji *et al.*, 2022b; Onwude *et al.*, 2022; Shrivastava *et al.*, 2023; Gutierrez-Franco; Mejia-Argueta; Rabelo, 2021; Burgos; Ivanov, 2021; Lv *et al.*, 2024; Sengupta; Dreyer, 2023).

Por meio da leitura dos 37 estudos de caso encontrados nesta pesquisa, os principais conceitos, recursos de *hardware* e *software* diretamente usados pelos autores para proporem soluções baseadas em GDs foram listados a fim de se ter uma visão geral sobre quais tecnologias-chave têm permitido a aplicação do conceito de GDs no setor varejista até o momento (RQ3). As palavras/expressões usadas por cada estudo de caso foram organizadas em

três colunas (conceitos, *hardware* e *software*) e após a leitura dos artigos, o pacote *wordcloud* da linguagem de programação Python foi utilizado para se gerar as nuvens de palavras.

A Figura 9 ilustra os principais conceitos tecnológicos usados pelos autores. Como pode ser notado, Análise de Dados (do inglês, *Data Analysis*), Análise Estatística (do inglês, *Statistical Analysis*), Simulação, (do inglês, *Simulation*) e IA (no inglês, *Artificial Intelligence*) se destacam como os conceitos mais citados ao longo dos artigos. Foi possível notar que praticamente todos os recursos ou aplicações dos GDs são voltados ao uso intenso de dados.

Isso é justificado pelo fato que quase todos os trabalhos práticos aplicaram conceitos de Análise de Dados e Análise Estatística para se extrair *insights* importantes dos dados utilizados. A grande quantidade de dados disponíveis para se alimentar soluções baseadas em GDs tem também estimulado o uso de técnicas de IA.

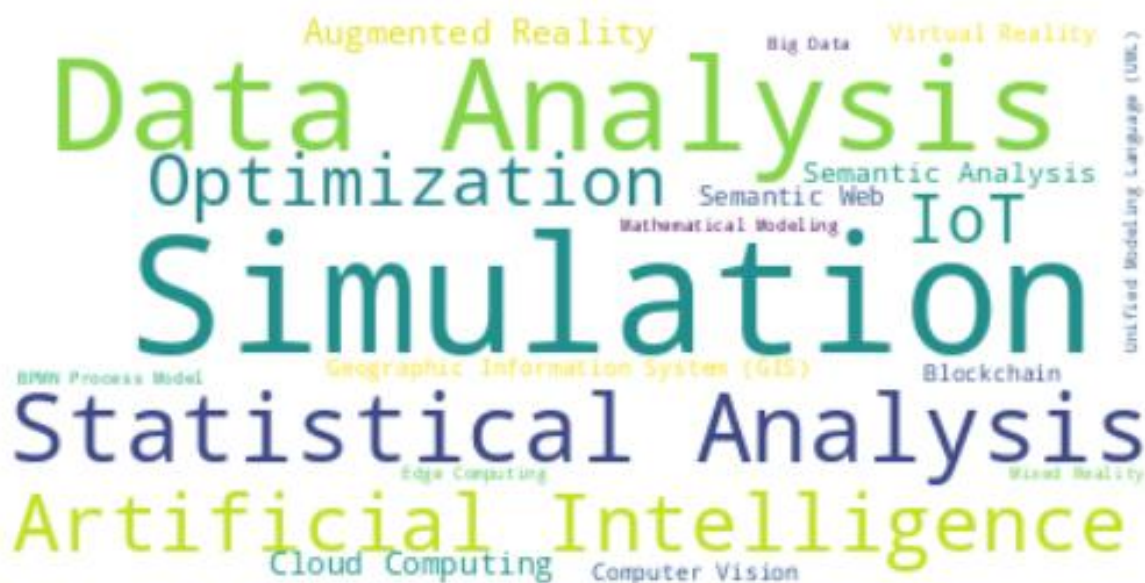


Figura 9 - Conceitos mais populares relacionados à implementação de GDs no setor varejista
Fonte: elaborado pelo próprio autor

A Figura 10 ilustra os dispositivos de *hardware* mais populares empregados pelos autores. Entre eles, sensores de diferentes tipos assumem o protagonismo em gerar dados úteis para os sistemas. Alguns autores também mencionam o uso intenso de sensores para dar suporte a soluções baseadas em GDs, porém, sem especificá-los.

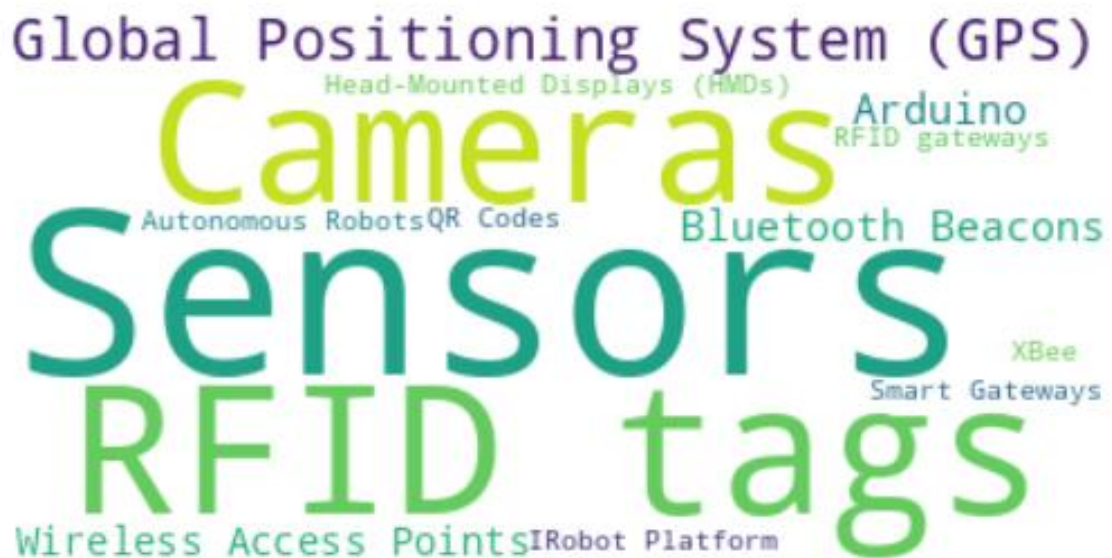


Figura 10 - Dispositivos de *hardware* mais populares relacionados a implementação de GDs no setor varejista
 Fonte: elaborado pelo próprio autor

Vários estabelecimentos têm implementado mecanismos de etiquetagem modernos em seus ativos físicos. Entre os mecanismos de identificação mais populares, os ligados à identificação por radiofrequência (RFID, do inglês *Radio-frequency Identification*) se destacam. Devido à popularização de câmeras e grande melhora das técnicas de VC atualmente, a detecção, classificação e segmentação de objetos também vem se tornando de uso estratégico para varejistas.

Em relação às soluções de *software* utilizadas pelos autores que apresentaram aplicações práticas, os resultados são bastante heterogêneos, conforme ilustrado na Figura 11. Foi possível observar a presença de ferramentas de simulação clássicas como o COMSOL Multiphysics, AnyLogic e o NetLogo. Ferramentas matemáticas também clássicas, como Excel e o MATLAB, também estão presentes. Java e R foram as linguagens de programação mais largamente utilizadas.

Este estudo também buscou compreender quais foram os dados de entrada, janelas de atualização e o grau de autonomia das soluções encontradas (RQ4). Um recurso fundamental para construção dos GDs são os dados de entrada. A Figura 12 apresenta os diferentes tipos de fontes de dados usados pelos estudos de caso encontrados nesta pesquisa. A maioria dos dados empregados pelos autores são secundários. Estes dados são principalmente bases de dados que guardam dados passados de sensores, nível de inventários, transações dos clientes ou dados de navegações dos clientes.



Figura 11 - Softwares mais populares relacionados a implementação de GDs no setor varejista
 Fonte: elaborado pelo próprio autor

Dados em tempo real são oriundos de sensores, câmeras, Bancos de Dados (BDs) ou plataformas. Dados simulados são os menos utilizados. Eles foram propostos pelos próprios autores ou fazendo uso de ferramentas como MATLAB, Excel e a linguagem de programação R. Como pode ser notado, GDs são soluções puramente orientadas a dados que podem fazer uso de diferentes fontes de dados para representar entidades ou processos do mundo real dentro do mundo virtual.

É importante destacar que a janela de atualização em que um GD é atualizado não precisa ser em tempo real. Isso depende do contexto da aplicação, e o ponto mais importante desta abordagem é permitir a criação de modelos dinâmicos que descrevem sistemas ou objetos que mudam de comportamento ao longo do tempo e que seja de uso constante (Wright; Davidson, 2020).

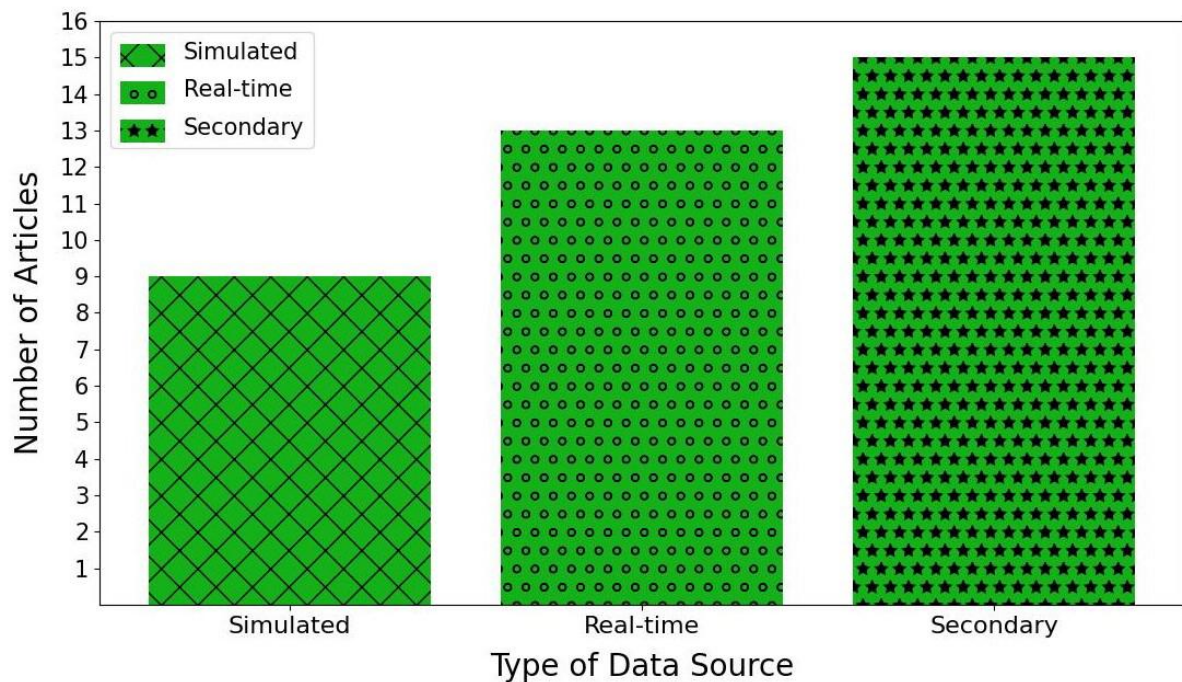


Figura 12 - Tipos de dados usados nos estudos de caso
 Fonte: elaborado pelo próprio autor

Entre os estudos de caso analisados, 49% não usaram atualização em tempo real. Isso varia de acordo com a necessidade de se tomar decisões. Nos outros 51% dos estudos de caso, o intervalo de tempo utilizado foi em tempo real ou quase real. O nível de autonomia de um GD está ligado com sua capacidade de comandar ações diretas dentro dos sistemas físicos (resultando em uma autonomia total, sem intervenção humana) ou fornecer sugestões de ações (atuando como uma ferramenta de apoio inteligente ao humano) (Santos *et al.*, 2021).

Entre os casos de estudo analisados, apenas 16% apresentam soluções totalmente autônomas. Isso mostra que embora os modelos virtuais estejam se tornando mais precisos a cada dia no setor varejista, a grande maioria dos sistemas ainda precisa da participação humana para a tomada de decisões no mundo real.

Enquanto a adoção dos GDs no setor varejista ainda se encontra em um estado inicial comparado a outros setores, como o manufatureiro, mesmo assim foi possível identificar vários benefícios, oportunidades e desafios de se aplicar GDs neste setor. O Quadro 1 e o Quadro 2 fornecem um resumo das descobertas (RQ5).

Quadro 1 - Resumo de benefícios e oportunidades relacionados ao uso de GDs do setor varejista

| Benefícios e Oportunidades |
|--|
| <p>- A magnitude dos dados gerados por plataformas e sensores são estrategicamente úteis para se alimentar plataformas de simulação e de <i>analytics</i> a fim de se obter informações úteis para se prever o comportamento do cliente e auxiliar a gerência em processos logísticos; (Maïzi; Bendavid, 2019; Hung; Wang; Chang, 2020; Kaur <i>et al.</i>, 2020; Chan; Lau; Fan, 2020; Maïzi; Bendavid, 2021; Singh <i>et al.</i>, 2022; Yensabai <i>et al.</i>, 2023);</p> <p>- Permite a identificação de onde e como alimentos frescos perdem sua qualidade em uma cadeia de suprimentos. Isso também facilita o cálculo do tempo de vida de prateleira destes alimentos no lado do varejista (Pal; Kant, 2020; Shoji <i>et al.</i>, 2022a; Shoji <i>et al.</i>, 2022b; Onwude <i>et al.</i>, 2022; Defraeye <i>et al.</i>, 2021; Shrivastava <i>et al.</i>, 2023);</p> <p>- Facilita a criação de Cadeias de Fornecimento Virtuais, onde varejistas, fabricantes e fornecedores possam estar altamente conectados como agentes de <i>software</i>. Ao utilizar desta abordagem, o comportamento e o efeito de cada empresa e o comportamento total da cadeia de suprimentos pode ser simulada ao mesmo tempo quando um pedido é feito no varejista (Matsuda <i>et al.</i>, 2019; Matsuda <i>et al.</i>, 2020; Nishi <i>et al.</i>, 2020; Matsuda <i>et al.</i>, 2021; Kamiebisu <i>et al.</i>, 2022);</p> <p>- Oferece um modo de tentar diferentes opções para se melhorar as operações do varejo, como as operações de <i>checkout</i> (Hauser <i>et al.</i>, 2017; Antczak, Weron; Zabawa, 2020; Mykoniatis <i>et al.</i>, 2020);</p> <p>- Permite que os tomadores de decisão criem sistemas de alerta e façam uma melhor utilização dos recursos em operações de entregas, como reduzir o número de veículos utilizados, diminuir os custos e obter melhor acurácia no tempo estimado de chegada do produto ao cliente (Gutierrez-Franco, Mejia-Argueta; Rabelo, 2021);</p> <p>- Criar sistemas que tirem vantagem dos procedimentos de aprendizado ao se utilizar de IA para se autoajustarem, buscando atender os objetivos dos <i>stakeholders</i> de forma benéfica a todos (Hanke <i>et al.</i>, 2018; Gutierrez-Franco; Mejia-Argueta; Rabelo, 2021; Elbouzidi <i>et al.</i>, 2022; Mykoniatis <i>et al.</i>, 2020; Chan; Lau; Fan, 2020);</p> <p>- Rastrear o engajamento dos clientes dentro das lojas tem sido pouco explorado. Por meio da utilização de plataformas IoT e sensores, estas ações podem ser gravadas, reportadas e previstas (Chan; Lau; Fan, 2020; Kaur <i>et al.</i>, 2020; Maizi; Bendavid, 2019; Maizi; Bendavid, 2021);</p> <p>- Estabelecimentos físicos estão encontrando uma crescente competição do comércio eletrônico. Sistemas ubíquos têm o potencial de melhorar a representatividade das lojas físicas e diminuir as desvantagens associadas em se manter estes estabelecimentos (Hauser <i>et al.</i>, 2017; Hanke <i>et al.</i>, 2018);</p> <p>- Os GDs para as Cadeias de Suprimentos (SCDT, do inglês <i>Supply Chain Digital Twins</i>) representam um tópico de pesquisa emergente na modelagem, gerência e controle das cadeias de suprimentos, uma vez que ajuda a avaliar, prever e otimizar o comportamento do sistema tanto a níveis físicos e virtuais (Burgos; Ivanov, 2021; Maheshwari; Kamble, 2022);</p> |

- Lojas varejistas tradicionais podem ser revitalizadas para se integrar tecnologias IoT, criando efetivamente experiências de compras multicanais (*omnichannel*). Isso significa alavancar vários canais para se lidar com as necessidades dos clientes, provendo assim experiências de compras mais integradas (Janhofer *et al.*, 2020; Chan; Lau; Fan, 2020);

Fonte: o próprio autor

Quadro 2 - Resumo dos principais desafios relacionados ao uso de GDs do setor varejista

| Desafios |
|---|
| <p>- Conforme a interação entre o mundo físico e digital aumenta, o mundo físico fica mais sujeito a vulnerabilidades de segurança (White; Schmidt; Golparvar-Fard, 2014; Karaköse; Yetiş, 2017; Trappey <i>et al.</i>, 2018; Hung; Wang; Chang, 2020; Gupta <i>et al.</i>, 2020; Rachana Harish <i>et al.</i>, 2021);</p> <p>- Analisar grandes quantidades de dados pode ser um processo complexo e demorado. É importante empregar técnicas estatísticas sofisticadas e ML para se extrair <i>insights</i> significativos dos dados (Gutierrez-Franco; Mejia-Argueta; Rabelo, 2021; Yensabai <i>et al.</i>, 2023);</p> <p>- Em termos de autonomia, a maioria das soluções baseadas em GDs ainda precisam do envolvimento humano para efetivamente fazer mudanças do lado dos sistemas físicos (Elbouzidi <i>et al.</i>, 2022);</p> <p>- Empresas são geralmente relutantes para adaptar tecnologias digitais por conta do alto grau de heterogeneidade (Maheshwari; Kamble, 2022);</p> <p>- Operações autônomas de sistemas CPS/IoT exigem uma abordagem interdisciplinar e cruzada de tecnologias e conceitos para se realizar a conexão entre os mundos real e virtual (Junaid Farooq; Zhu, 2020);</p> <p>- Desenvolver estudos que lidam com múltiplas decisões de operações, múltiplos temas e produtos (Maizi; Bendavid, 2019);</p> <p>- Desenvolver soluções adaptativas uma vez que cada cadeia de suprimento apresenta uma combinação diferente de operações unitárias, e cada carregamento apresenta condições logísticas e ambientais diferentes (Shrivastava <i>et al.</i>, 2023);</p> |

Fonte: o próprio autor

Neste estudo foi observado que, apesar dos GDs ainda estarem em seus primeiros anos, a maioria dos trabalhos encontrados nesta pesquisa são estudos de caso que de alguma forma tentaram aplicar este conceito para buscar alcançar melhores resultados em uma série de operações presentes no varejo. Dentre essas operações, a previsão do comportamento do cliente e a cadeia de suprimentos se destacam como as mais estabelecidas na literatura, ilustrando claramente a necessidade de entender cada vez mais em detalhes as ações dos clientes e a importância de se criar cadeias de suprimentos cada vez mais precisas e previsíveis. No entanto,

a pesquisa também identificou que a aplicação do conceito GD em outras operações importantes ainda é limitada, abrindo assim uma gama de oportunidades de pesquisa.

Também foi possível observar que quase todos os recursos ou aplicações de GDs são dependentes de dados, ou seja, os dados são a base e o combustível deles. Isso se justifica pelo fato de que Análise de Dados, Análise Estatística, Simulação e IA foram os conceitos mais utilizados pelos estudos de caso. Sensores de diferentes naturezas, câmeras, *tags* RFID e GPS são as principais fontes de dados entre os estudos de caso. Isso mostra claramente como o uso desses dispositivos também se tornou popular e estratégico no setor varejista.

Os recursos de *software* utilizados pelos autores em aplicações práticas são bastante heterogêneos. Isso ilustra que não há aplicações específicas para se criar um GD. O mais importante é fornecer um modelo de uso constante, que conecte firmemente os mundos físico e virtual, e que represente o sistema físico em sua contraparte virtual.

A conexão entre esses mundos não precisa ser necessariamente em tempo real. Praticamente metade dos estudos de caso não usam atualização em tempo real, e a maioria deles se baseia em fontes de dados secundários. A maioria das soluções práticas também não é totalmente autônoma, ou seja, ainda requer a participação humana para auxiliar na tomada das decisões finais.

Muitos benefícios, oportunidades e desafios também foram apresentados. Entre os benefícios e oportunidades, destaca-se a alta disponibilidade de dados produzidos atualmente, uma vez que isso tem facilitado cada vez mais a criação de soluções baseadas em GDs. O uso crescente de técnicas de IA em soluções baseadas em GD também gerou grande interesse, pois têm ajudado a criar sistemas cada vez mais independentes e autônomos. A vulnerabilidade de segurança e a análise de grande quantidade de dados foram os principais desafios levantados pelos autores quando se trata de criar soluções de GD para o varejo.

A contribuição desta RSL foi fornecer uma visão geral significativa que pode dar suporte a acadêmicos e profissionais em pesquisas futuras sobre a aplicação de GDs em diversas operações presentes no setor varejista. Para trabalhos futuros, para ter um panorama ainda mais amplo do uso potencial de GDs no setor varejista, recomenda-se adicionalmente examinar fontes alternativas de pesquisa, como patentes e produtos registrados.

Nesta RSL, também foi possível observar que, dentre as diversas operações do setor que têm explorado o uso dos GDs de alguma forma, a Gerência de Inventários se apresenta como uma das operações mais carentes de pesquisas. Apesar dos trabalhos encontrados já

apresentarem iniciativas práticas para suporte nesta atividade, nenhuma propõe um modelo de uso contínuo e que faça uso de técnicas modernas e acessíveis de VC e Previsão para dar suporte em tempo real ao gerenciamento de inventários em prateleiras, estruturas estas fundamentais presentes em estabelecimentos varejistas para organização e exposição dos produtos.

2.6 Pesquisa Exploratória (PE): Integração Visão Computacional e Gêmeos Digitais

A integração entre técnicas de VC e o conceito de GDs ainda é uma junção pouco explorada na literatura, principalmente com aplicações voltadas ao setor varejista. A fim de entender melhor este cenário, uma PE foi realizada na base de dados Scopus em dezembro de 2023, utilizando da seguinte consulta *TITLE-ABS-KEY ("digital twin" AND "computer vision" AND "retail*") AND (LIMIT-TO (LANGUAGE , "English"))*. Esta consulta buscou identificar artigos de todos os anos desta base que possuem em seus títulos, resumos ou palavras-chaves as expressões “*digital twin*”, “*computer vision*” e “*retail**” e estejam na língua inglesa. Apenas dois resultados foram retornados, conforme ilustrado no Quadro 3.

Quadro 3 - Resultado da PE sobre a integração de GDs e técnicas VC no setor varejista. Os tipos de estudos podem ser Conceito (C), Revisão da Literatura (RL) ou Estudo de Caso (EC)

| Autor (Ano) | C | RL | EC |
|-------------------------------|----------|-----------|-----------|
| Ramalho <i>et al.</i> (2019) | | | X |
| Schwartz <i>et al.</i> (2021) | | | X |

Fonte: o próprio autor

Com o objetivo de melhorar a experiência de varejo e facilitar a gestão de lojas físicas, o trabalho de Ramalho *et al.* (2019) apresenta uma ferramenta desenvolvida para analisar o tráfego de pedestres dentro dos estabelecimentos, fornecendo aos varejistas *insights* sobre o volume de clientes e dados demográficos, auxiliando assim no processo de vendas. Para atingir este objetivo, foram explorados métodos e algoritmos de VC, seguidos de conceituação teórica e implementação de ferramentas. Apesar de fazer uso extenso da VC, inclusive com aplicação do algoritmo YOLOv2, a abordagem não atua como um GD pois não há integração contínua e bidirecional de dados entre o sistema físico real e seu modelo digital ao longo do tempo.

Já o trabalho de Schwartz *et al.* (2021) propõe vincular ativos digitais criados por meio de modelagem de informações de construção (BIM, do inglês *Building Information Modeling*) com suas contrapartes físicas usando marcadores fiduciais com padrões definidos por refletores

esféricos colestéricos. Estes marcadores apresentam uma forma inovadora de codificar informações que são facilmente detectadas e lidas com VC, mas que são praticamente invisíveis ao olho humano. Os autores ressaltam como esta abordagem poderia potencializar a criação de GDs, mas não apresentam uma abordagem prática que permita integrar estes dois conceitos, visto que o trabalho foca em ilustrar uma proposta isolada de vinculação segura e automática entre objetos físicos e seus modelos digitais por meio dos marcadores, buscando fortalecer a confiança, rastreabilidade e sincronização entre os mundos real e virtual.

Após análise dos resultados da primeira consulta, uma segunda consulta foi realizada também na base de dados Scopus, em dezembro de 2023, utilizando da *query TITLE-ABS-KEY ("digital twin" AND "YOLO") AND (LIMIT-TO (LANGUAGE , "English"))*. O objetivo foi identificar como o algoritmo YOLO vem sendo integrado em soluções baseadas em GDs, independente do setor de aplicação. Doze resultados foram retornados. Dois dos resultados foram descartados, uma vez que seus títulos, resumos e palavras-chave se referem a uma coletânea de artigos (*Proceedings*), e não a um artigo específico. Por isso, dez artigos desta busca foram analisados, conforme ilustra a Quadro 4.

Quadro 4 - Resultado da PE sobre a integração de GDs e o algoritmo YOLO, independente da área de aplicação. Os tipos de estudos podem ser Conceito (C), Revisão da Literatura (RL) ou Estudo de Caso

| Autor (Ano) | C | RL | EC |
|------------------------------------|----------|-----------|-----------|
| Amarnath, N e V (2023) | | | X |
| Dawid, Buchwald e Pawlak (2023) | | | X |
| Fernandés, Delrieux e Muñoz (2022) | | | X |
| Liu <i>et al.</i> (2023) | | | X |
| Oriti <i>et al.</i> (2022) | | | X |
| Pokuciński e Mrozek (2023) | | | X |
| Udoy <i>et al.</i> (2023) | | | X |
| Yun e Park (2021a) | | | X |
| Yun e Park (2021b) | | | X |
| Zhou <i>et al.</i> (2022) | | | X |

Fonte: o próprio autor

O trabalho de Amarnath, N e V (2023) propõe o GD de um braço robótico capaz de prever com precisão sua temperatura, quantidade de corrente retida e padrões de comportamento esperados em determinadas situações. Os autores utilizaram o apoio do algoritmo YOLOv5 embarcado em um minicomputador Raspberry Pi para auxiliar na detecção de possíveis objetos que podem ser manipulados pelo braço robótico, aumentando assim as capacidades da solução proposta.

Dawid, Buchwald e Pawlak (2023), por meio da junção do algoritmo YOLOv7 e da plataforma Unity3D, propuseram uma solução baseada em GD para detectar com precisão o *status* dos faróis reais de veículos. O objetivo dos autores foi fornecer uma solução prática e automatizada para avaliar o estado operacional dos faróis dos veículos, o que tem implicações significativas para a segurança rodoviária e a manutenção geral deles.

Já Yun e Park (2021a) e Yun e Park (2021b) propuseram um método para se implementar um GD para um sistema de carro autônomo. Na camada virtual, o YOLOv4 juntamente com funções de detecção de obstáculos aproveitadas do famoso jogo ‘GTA 5’ foram utilizadas. Já na camada física, um ‘DonkeyCar’, ou seja, um veículo real programável por meio da linguagem Python, foi utilizado. A proposta dos autores foi propor uma plataforma para se testar e aperfeiçoar recursos de carros autônomos por meio de um GD. O trabalho de Liu *et al.* (2023) também propõe uma solução de GD integrada a algoritmos de VC, como o YOLOv5, para se analisar os riscos de tráfego em rodovias, buscando assim fornecer um suporte eficiente no projeto e gestão destes locais.

Fernandés, Delrieux e Munõz (2022) apresentam uma solução baseada em GD para virtualização do ambiente de trabalho, mais particularmente do mapeamento em tempo real do movimento das pessoas que se encontram nele. O objetivo dos autores é explorar possibilidades de soluções que possam contribuir com uma melhor gestão do trabalho dos funcionários da construção civil. A solução é dividida em duas etapas: detecção de pessoas seguida da estimativa de pose delas. Na etapa de detecção, mais em particular, uma variação do algoritmo YOLO é comparada com outras soluções de VC para se realizar esta tarefa.

Ainda na área de construção civil, Zhou *et al.* (2022) propõem um estudo piloto de GD para monitoramento e manutenção de pontes. O YOLOv5 foi utilizado pelos autores juntamente com outro algoritmo de VC chamado *Simple Online and Realtime Tracking* para se extrair a trajetória dos veículos que trafegam pela ponte.

Pokuciński e Mrozek (2023) não apresentam uma solução que integre os conceitos de GD e VC, porém, ressaltam como o uso do YOLO poderia ser estratégico para auxiliar na criação de GDs para ambientes *indoor*, como apartamentos. Oriti *et al.* (2022) descrevem o desenvolvimento e implantação de um GD para ser utilizado em uma linha de produção de *skates*. Neste sistema, o YOLO é utilizado como suporte a um Cobot (do inglês, *Collaborative Robot*) para se detectar a posição das rodas antes de se realizar a inserção delas no *skate*.

Já Udoy *et al.* (2023) propõem uma versão aprimorada do conhecido sistema de código de QR Code (do inglês, *Quick Response Codes*) visando aumentar a capacidade de armazenamento de dados de objetos que possam ser alvo de soluções baseadas em GDs. O YOLOv5 é utilizado pelos autores como mecanismo de leitura deste novo tipo de QR Code proposto.

Como pode ser observado nos resultados encontrados, todas as soluções apresentadas pelos autores são estudos de caso que em algum momento buscaram utilizar o suporte do algoritmo YOLO para aumentar as capacidades dos GDs propostos para solucionar diversos tipos de problemas. Porém, nenhuma solução de caráter prático que claramente integre GDs e VC e seja aplicada ao setor varejista foi encontrada. Isso ilustra claramente a carência de soluções baseadas em GDs para este setor de alto impacto na economia. Além disso, em nenhum dos trabalhos também foi identificada uma solução que integre as duas técnicas para suporte ao controle de inventários, independente da área de atuação.

3. MÉTODO DE PESQUISA

3.1 Considerações iniciais

Esta seção apresenta a classificação da pesquisa científica realizada quanto à sua natureza, aos objetivos, a abordagem e ao método utilizado. Em seguida, será apresentada mais em detalhes a definição do método utilizado neste trabalho, no caso, a M&S.

3.2 Classificação da pesquisa

Segundo Azevedo e Ensslin (2020) esta pesquisa classifica-se, quanto à sua natureza, como aplicada, uma vez que objetiva a produção de conhecimentos que possam ser prontamente utilizados na prática ou direcionados à resolução de problemas concretos. Já quanto aos objetivos, ela pode ser classificada como exploratória e descritiva. É exploratória por buscar compreender um problema real, ainda pouco estruturado, por meio da exploração do estado da arte e por propor a construção de uma ferramenta para investigação de cenários, identificação de variáveis relevantes e análise preliminar de comportamentos da proposta.

Ao mesmo tempo, assume um caráter descritivo ao representar e detalhar o funcionamento do sistema real, fornecendo uma caracterização objetiva dos dados, processos e parâmetros envolvidos, o que possibilita a análise de seu desempenho em diferentes condições. Essa abordagem combinada permite tanto a geração de novos conhecimentos quanto a sistematização de informações relevantes para a tomada de decisão.

Esta pesquisa trará uma abordagem quantitativa, pois visa trabalhar com variáveis mensuráveis, o que significa traduzir em números as informações para classificá-las e analisá-las (Miguel *et al.*, 2018). Por fim, o método a ser utilizado é a M&S, que segundo Miguel *et al.* (2018), é empregada quando se deseja experimentar, por meio de um modelo, um sistema real, determinando como este sistema responderá a modificações que lhe são propostas. A Figura 13 resume a classificação desta pesquisa.

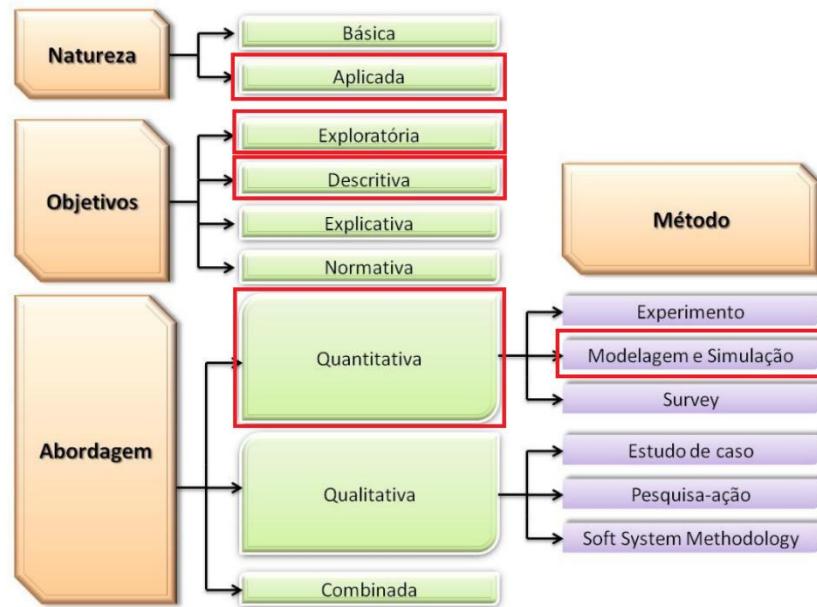


Figura 13 - Classificação desta pesquisa
 Fonte: Adaptado de Miguel *et al.* (2018)

A pesquisa aplicada tem um foco prático, produzindo resultados que podem ser diretamente utilizados para resolver problemas reais (Azevedo; Ensslin, 2020). Assim, esta tese irá apresentar um modelo para gerenciamento de inventários no setor varejista baseado em GD e VC, cujo resultado poderá possivelmente auxiliar os *stakeholders* na tomada de decisões e, conseqüentemente, melhorar seus processos de controle de inventários.

A característica mais marcante de pesquisas com objetivo descritivo é a coleta de dados, que pode ser feita via questionário ou observação sistemática (Azevedo; Ensslin, 2020). Logo, este trabalho visa coletar os dados de forma sistemática por meio da execução do modelo, descrever sua aplicação e apresentar seus resultados.

Segundo Miguel *et al.* (2018) a pesquisa quantitativa parte do princípio de que tudo pode ser medido numericamente, transformando opiniões e informações em dados numéricos para classificá-los e analisá-los, exigindo assim a utilização de recursos e técnicas estatísticas. Sobre o método para condução do trabalho, que será a M&S, deseja-se experimentar, por meio de um modelo dinâmico e determinístico, um sistema real, determinando-se como este sistema responderá a modificações que lhe serão propostos.

3.3 Definição do método: Modelagem e Simulação (M&S)

A M&S é um dos métodos mais largamente empregados na área de PO. Por muito tempo, e principalmente após a popularização dos computadores pessoais, ela tem sido utilizada

para auxiliar as pessoas a construir modelos virtuais de situações que acontecem em diferentes locais e processos que existem no mundo real. Se os relacionamentos entre os elementos que compõem um modelo são simplificados, métodos matemáticos podem ser suficientes para se obter informações importantes de uma situação. Porém, a grande maioria dos sistemas presentes no mundo real são complexos para serem analisados apenas analiticamente, por isso, se utiliza da técnica de simulação computacional (Law, 2015).

Segundo Law (2015), as simulações computacionais podem ser classificadas em três diferentes dimensões, são elas:

- Estática vs. Dinâmica: a simulação do tipo estática é a representação de um sistema em um momento em particular, em que o tempo dos acontecimentos não é levado em consideração. Um exemplo clássico deste tipo de simulação é a de Monte Carlo. Por outro lado, um modelo de simulação dinâmico é capaz de representar um sistema que se modifica ao longo do tempo.
- Determinística vs Estocástica: se a simulação não trabalha com qualquer componente ou valor aleatório, ela é chamada de determinística. Porém, se possui componentes com valores aleatórios, é considerada estocástica.
- Contínua vs Discreta: se um sistema se altera a cada fração de tempo, a simulação é dita contínua. Porém, se o estado de um sistema se altera apenas após a ocorrência de eventos, a simulação é dita discreta.

O sistema proposto caracteriza-se como uma simulação dinâmica, determinística e discreta, pois o estado do GD evolui ao longo do tempo, porém apenas em instantes de atualização (eventos de detecção e previsão), e, dadas as mesmas entradas, produz os mesmos resultados. Além disso, dentro de M&S, esta pesquisa pode ser considerada como empírica, pois procura alinhar o comportamento do modelo a um sistema real, contribuindo assim para o aprimoramento da tomada de decisão.

3.3.1 Descrição do método

Este estudo é de natureza aplicada, com abordagem quantitativa e que utiliza de M&S. A fim de melhor definir as etapas de desenvolvimento deste projeto, o autor utiliza da sequência de passos que um projeto de M&S deve seguir segundo Montevechi *et al.* (2010). Em suma, os autores apresentam um *framework* para condução de projetos que se divide em 3 etapas fundamentais conforme ilustra a Figura 14, sendo elas: Concepção, Implementação e Análise dos Dados.

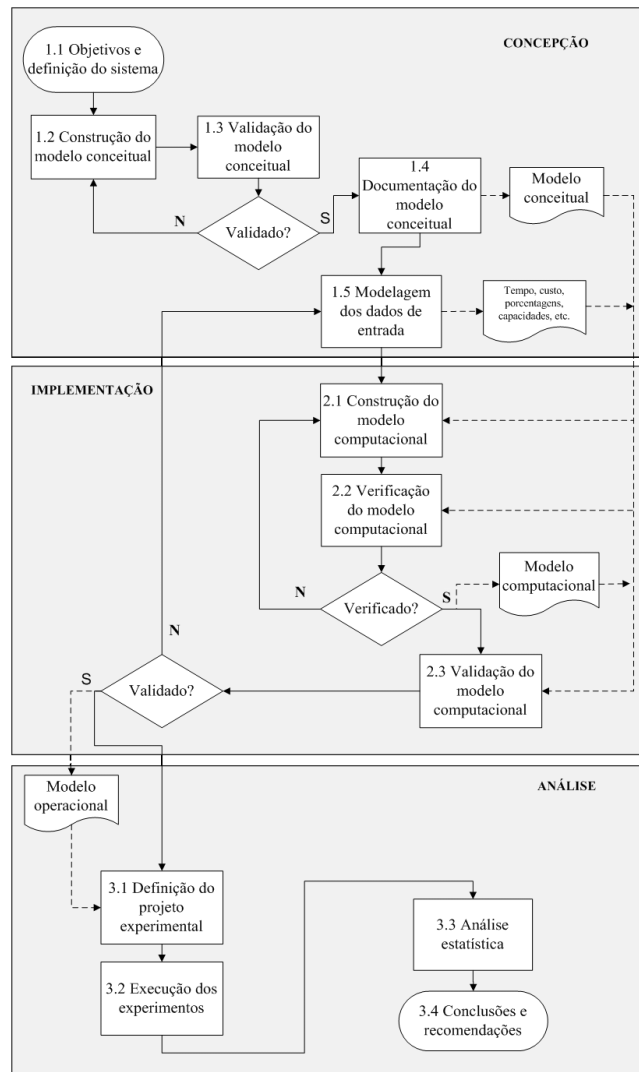


Figura 14 - Etapas de um projeto de M&S
 Fonte: Montevechi *et al.* (2010)

A finalidade de cada uma das etapas presentes no *framework* é a seguinte:

- **Concepção:** fase inicial do projeto, que busca descrever de forma clara os objetivos do problema, quais os principais elementos envolvidos no sistema por meio da construção, validação e documentação de um modelo conceitual, e quais os principais dados de entrada que podem ser utilizados. Esta é uma fase primordial e que consome uma porcentagem de tempo significativa.
- **Implementação:** uma vez que o modelo conceitual esteja validado e os dados de entrada estejam definidos, realiza-se a construção do modelo computacional. Uma vez construído, é importante verificar e validar se o modelo computacional permitirá a realização de experimentações para auxiliar na tomada de decisões em um problema.

- Análise: uma vez que o modelo esteja validado, ele se torna um modelo operacional, ou seja, experimentos e análises estatísticas já podem ser definidas e realizadas sobre ele, a fim de se tirar conclusões e sugerir recomendações para tomadas de decisões no mundo real.

A etapa de Concepção será melhor detalhada no subtópico 4.2 desta tese, seguida da etapa de Implementação no subtópico 4.3, e por fim o subtópico de Análise no subtópico 4.4.

4. APLICAÇÃO

4.1. Considerações iniciais

Esta seção apresenta detalhes das três etapas de desenvolvimento da proposta, no caso, as etapas de Concepção, Implementação e Análise. Na etapa de Concepção, será descrita a problemática do trabalho e como o GD foi projetado para permitir o monitoramento em tempo real do estado das prateleiras, visando auxiliar os *stakeholders* na tomada de decisão. Na etapa de Implementação serão descritos os materiais e métodos utilizados na implementação do GD proposto na etapa de concepção. Já na etapa de Análise, serão apresentadas e discutidas métricas dos modelos de VC utilizados e resultados de testes práticos de operação do GD.

4.2 Concepção

Seguindo a ideia do *framework* proposto por Montevechi *et al.* (2010), a primeira etapa para condução de projetos de M&S é a Concepção. Inicialmente, a Figura 15 ilustra um diagrama minimalista da problemática deste trabalho, construído fazendo-se uso da ferramenta IDEF-SIM, também proposta por Montevechi *et al.* (2010).

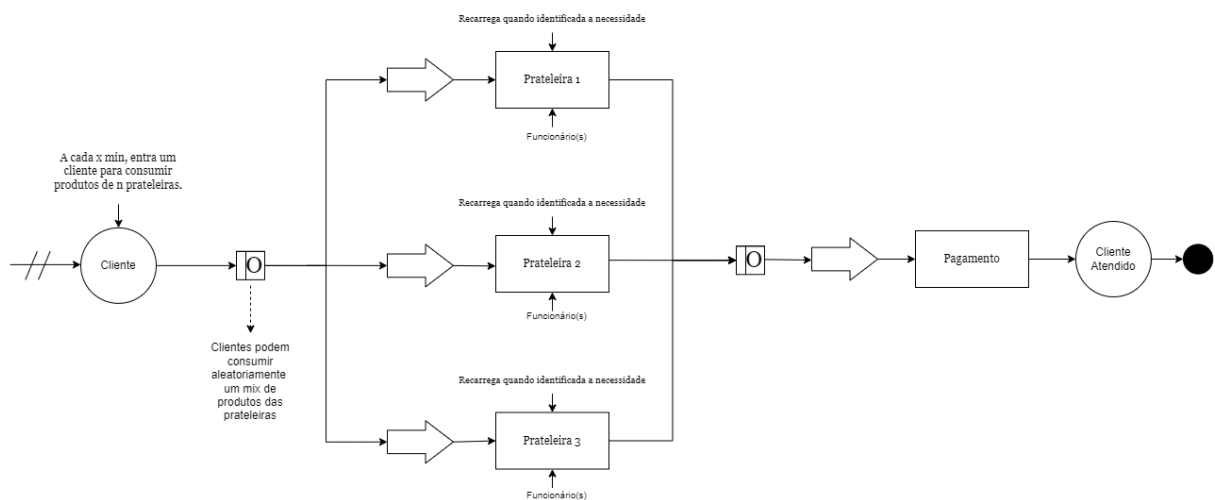


Figura 15 - Problemática deste trabalho
Fonte: elaborado pelo próprio autor

Em um estabelecimento varejista físico, clientes chegam a cada x minutos para consumirem um mix de produtos de n prateleiras diferentes. Neste diagrama, a fim de simplificar a descrição e representar os testes práticos descritos na seção 4.4.2, 3 prateleiras foram representadas. No estabelecimento, clientes se deslocam entre as prateleiras para

consumirem seus respectivos produtos, até que eles possam seguir finalmente para a realização do pagamento.

No caso da falta de produtos nas prateleiras, funcionários de um estabelecimento varejista (ex: um supermercado) são responsáveis pelo recarregamento delas, mas muitas vezes, nem sequer sabem que isso aconteceu, pois estão envolvidos em outras atividades. O trabalho de Mesquita, Montevechi e Leal (2022), por meio de uma simulação, ilustra como o monitoramento automatizado de ativos como prateleiras podem trazer ganhos significativos na quantidade de produtos presente nelas, minimizando problemas de *shelf-OOS*.

A Figura 16 ilustra situações vivenciadas ao visitar estabelecimentos varejistas. Como pode ser notado, após o consumo dos produtos pelos clientes, algumas prateleiras se apresentam visivelmente vazias, inclusive com seus fundos (*gaps*) à mostra, um indicador claro de esvaziamento. Santra e Mukherjee (2019) ressaltam que, do ponto de vista do varejista, o reconhecimento do produto e sua localização são importantes. Porém, a identificação de *gaps* é igualmente importante e merece maior atenção em pesquisas.

Kümpel, Mueller e Beetz (2021), Santra, Ghosh e Mukherjee (2022), Šikić *et al.* (2024) e Mesquita *et al.* (2024) também reforçam o impacto dos *gaps* no controle de inventários, uma vez que propuseram soluções que de alguma forma buscaram atacar o problema de *shelf-OOS* por meio da identificação deles. Porém, seus estudos ainda são de caráter teórico, não associados a GDs, não quantificadores (ou seja, não apresentam um método para quantificar de alguma forma os *gaps* detectados) e/ou não trabalham com retorno aos *stakeholders* em tempo real.

Este trabalho propõe um modelo para gerenciamento de inventários no setor varejista baseado em GD e VC. O objetivo do modelo é que ele sirva de suporte a processos logísticos dentro dos estabelecimentos, mais particularmente, no que tange ao monitoramento e orientações aos *stakeholders* quanto ao abastecimento de produtos nas prateleiras, estruturas estas fundamentais em estabelecimentos deste setor, mas que ainda carecem de uma abordagem orientada a dados, de caráter prático e quantificável para monitoramento em tempo real de seus estados.



Figura 16 – Exemplo ilustrativo da falta de produtos em prateleiras com fundo aparente
Fonte: Hendler, 2019

O diagrama em blocos ilustrado na Figura 17 apresenta o modelo conceitual geral proposto para este trabalho. A integração dos blocos dentro da área pontilhada corresponde ao GD, que atua como um GD semiautônomo, ou seja, tem a função de apoiar os *stakeholders* na tomada de decisões de recarregamento por meio de sugestões baseadas em análises prescritivas e preditivas.

Ele foi proposto buscando atender as três características fundamentais para se ter um GD segundo Qi e Tao (2018) e Wright e Davidson (2020), que são: o monitoramento de um objeto ou sistema físico real (prateleiras), um modelo virtual que é ajustado dinamicamente de acordo com os dados captados (*dashboard*) e um fluxo evolutivo de dados que conectam os dois mundos (provido pelos modelos de VC com suporte às lógicas de *Back-end* e estruturas de armazenamento de dados). Ou seja, ele se comporta como um modelo dinâmico, de uso constante, e que permite a troca de dados em tempo real entre os mundos real e virtual.

Todo o processo se inicia com a câmera realizando a captura de imagens das prateleiras com produtos. As imagens são recebidas em tempo real pelo GD, pois atuam como *inputs* fundamentais para a sua devida operação. Por meio do processamento de dados e regras de negócio específicas (aqui, chamado de *Back-ends*), o modelo é capaz de se calibrar, reconhecer elementos e integrar rotinas internas a fim de prover os *outputs* necessários para orientação dos *stakeholders*.

Os modelos de VC YOLO Padrão, Generalista e Especialista dão suporte aos *Back-ends* de Calibração e Reconhecimento. Estes *Back-ends* são utilizados para ajustarem o GD para operarem adequadamente sobre as estruturas das prateleiras detectadas e para realizar a

detecção de alterações no estado destas prateleiras, respectivamente. Recursos de armazenamento como um arquivo de configuração e um BD relacional são utilizados como suporte para armazenarem os *outputs* destas rotinas, que servirão como *input* de outras sub-rotinas e ao *Front-end (dashboard)* proposto, que atua como uma aplicação *web* (ou seja, pode ser acessado por múltiplos usuários e qualquer dispositivo, desde que se tenha um navegador *web* compatível disponível).

O modelo conceitual geral proposto na Figura 17 foi validado e aprovado pelos professores orientadores com conhecimentos em GDs antes de seguir para as próximas etapas. Este modelo conceitual deu início à documentação da proposta deste trabalho, que em seguida, foi melhor detalhada por meio de modelos diagramáticos na notação *Business Process Model Notation* (BPMN) para representar as responsabilidades e interações entre os blocos internos do GD, assim como os dados de entrada e saídas esperadas do modelo.

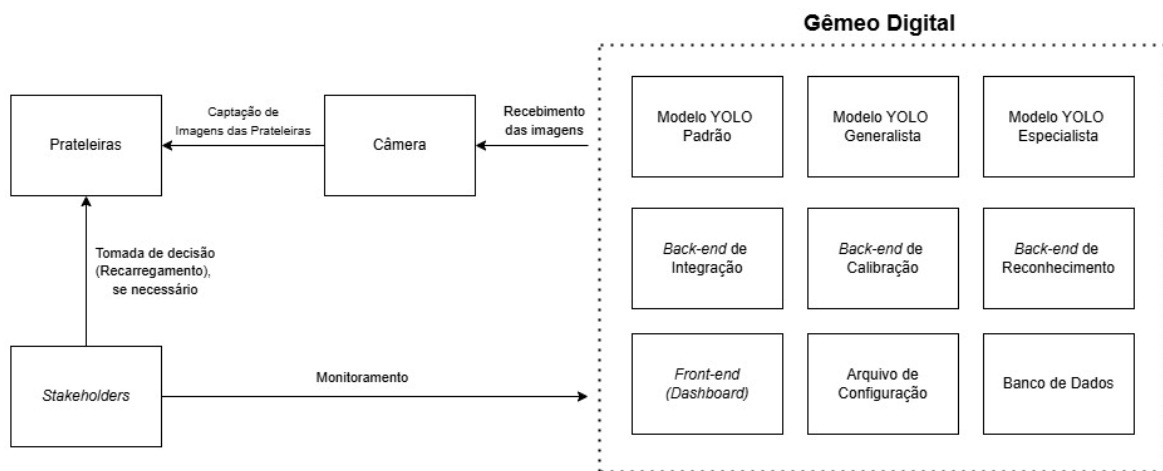


Figura 17 - Modelo Conceitual Geral do GD proposto
Fonte: elaborado pelo próprio autor

A BPMN é um padrão gráfico projetado para fornecer uma maneira formal, porém amigável, de representar processos de negócios em diversos domínios. Seu principal objetivo é ser facilmente compreendido tanto por usuários técnicos quanto de negócios, como analistas e demais participantes de um processo. Seus elementos gráficos como atividades, *gateways*, eventos e fluxos de conexão permitem uma representação clara da sequência lógica, responsabilidades e interações dentro de um processo (Chinosi; Trombetta, 2012).

Os subtópicos seguintes discutirão mais em detalhes o diagrama BPMN mencionado, detalhando os passos do GD proposto para captura, apresentação e previsão do índice de

esvaziamento de produtos em prateleiras e seus subprocessos, que são a calibração do GD e o reconhecimento de elementos.

4.2.1 BPMN geral do GD

A Figura 18 ilustra a *pool* principal do diagrama BPMN para realizar a captura, apresentação e previsão do índice de esvaziamento de produtos em prateleiras. Uma *pool* em um diagrama BPMN simboliza uma parte do processo, estabelecendo o escopo de suas atividades e como ela se comunica com os demais envolvidos. De forma geral, esta *pool* é subdividida em 3 *lanes*, no caso a Câmera, o *Back-end* de Integração e o *dashboard*. Uma *lane* em um diagrama BPMN indica as responsabilidades de um participante dentro de uma *pool*, estruturando visualmente as tarefas que ele realiza no processo.

Todo processo se inicia com a Câmera realizando a captura da imagem de uma prateleira. Uma vez que a imagem é capturada, ela é enviada ao *Back-end* de Integração para que seja realizado o processamento sobre ela. No *Back-end* de Integração, antes de mais nada, verifica-se se o sistema já se encontra calibrado. A Calibração do GD é uma parte fundamental para seu correto funcionamento, pois como *output* desta etapa é produzido um arquivo *Comma Separated Values* (CSV) de configuração do sistema, essencial para a etapa de Reconhecimento de Elementos na prateleira. Maiores detalhes sobre o processo de calibração do GD são abordados no subtópico 4.2.2.

Uma vez que o sistema esteja calibrado, no *Back-end* de Integração ele segue para a atividade de reconhecimento de elementos, atividade fundamental para detecção de entidades importantes neste sistema como pessoas, produtos e espaços vazios presentes nas prateleiras de cada produto. Como *output* desta etapa são calculados os índices de esvaziamento dos produtos presentes nas prateleiras, que devem ser devidamente armazenados em um BD relacional para serem futuramente consultados e também servirem de dados históricos para realização de previsões futuras. Maiores detalhes sobre o processo de reconhecimento de elementos são abordados no subtópico 4.2.3.

O processo de captura de imagens, calibração e reconhecimento dos elementos no GD podem acontecer de forma independente do *dashboard* do sistema, ou seja, sem que ele necessariamente esteja em execução. Isso foi pensado para prover maior independência ao *Back-end* e levando em consideração que o *dashboard* pode nem sempre estar em execução pelos *stakeholders*. Porém, conforme as etapas do *Back-end* vão sendo realizadas, os *stakeholders* também podem fazer uso do *dashboard* para consultarem em tempo real

informações importantes para auxílio na tomada de decisão, como a visualização dos índices de esvaziamento dos produtos presentes nas prateleiras e a previsão de índices futuros. Maiores detalhes sobre os recursos presentes no *dashboard* inicialmente proposto neste trabalho são abordados no subtópico 4.2.4.

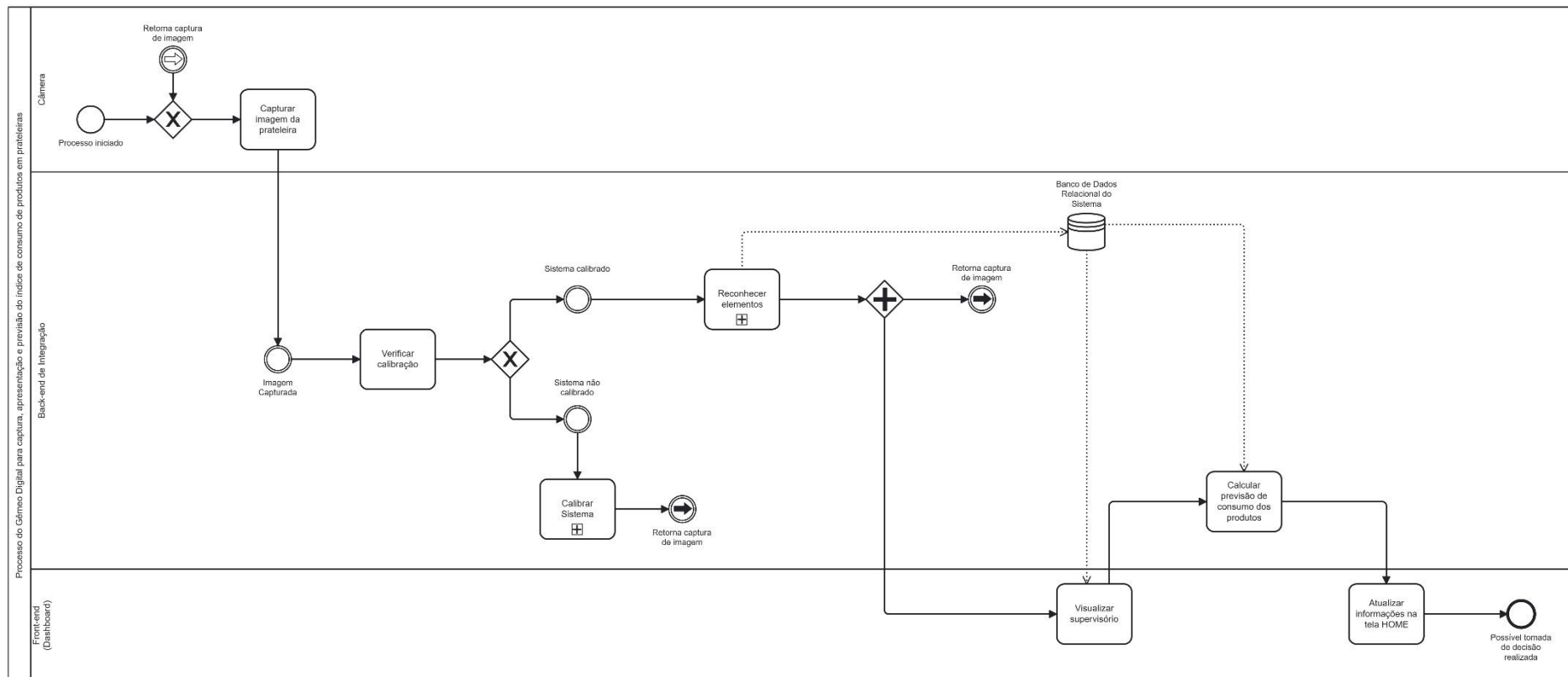


Figura 18 – Diagrama BPMN do processo de captura, apresentação e previsão do índice de esvaziamento de produtos em prateleiras
 Fonte: elaborado pelo próprio autor.

4.2.2 Subprocesso de calibração do GD

Como mencionado, a calibração do GD é uma parte mandatória para seu correto funcionamento neste momento. Na Figura 18 a atividade “Calibrar Sistema” é representada com um símbolo “+” internamente, indicando que ela é um subprocesso, ou seja, ela contém um processo interno detalhado que pode ser visualizado ou modelado em outro nível. Logo, a Figura 19 ilustra uma *pool* exclusiva para este subprocesso e que trabalha com três novas *lanes*: o Modelo YOLO Generalista, o Modelo YOLO Especialista e com o *Back-end* de Calibração.

Os Modelos YOLO Generalista e Especialista são os algoritmos de VC baseados no YOLO, já discutido no subtópico 2.3 desta tese. O Modelo YOLO Generalista é um modelo treinado de forma customizada pelo autor para focar no processo de detecção de dois elementos: formato (embalagem) de produtos no geral e espaços vazios. Já o Modelo YOLO Especialista também é um modelo treinado e customizado pelo autor, mas para focar no processo de classificação de produtos mais específicos, buscando dar suporte ao Modelo YOLO Generalista.

A combinação de dois ou mais modelos de VC para se realizar um processo de detecção se chama fusão hierárquica de modelos. Este tipo de abordagem oferece duas vantagens: desacoplamento funcional entre os modelos e maior acurácia nas detecções (Kinasih; Machbub; Yulianti; Rohman, 2023; Melek; Sonmez; Ayril; Varli, 2023; Pan; Xu; Cheng; Lian, 2024). O desacoplamento funcional busca modularizar as responsabilidades dos modelos. Isso permite treinar cada modelo com bases de dados específicas e atualizar partes do GD sem recomeçar do zero, evitando criar assim um modelo de detecção monolítico.

Além disso, a fusão hierárquica tende a melhorar a acurácia pois funciona como uma triagem em duas etapas. Primeiro, um modelo faz uma varredura ampla, extraindo objetos de interesse de forma mais genérica. Em seguida, outro modelo olha apenas para regiões de interesse detectadas e faz uma análise mais detalhada, corrigindo erros e diferenciando objetos parecidos. Por exemplo, neste trabalho, o Modelo YOLO Generalista detecta um produto do tipo lata (de forma mais genérica), e o Modelo YOLO Especialista classifica esta lata como sendo de Coca-Cola.

No processo de calibração, inicialmente o Modelo YOLO Generalista busca identificar a embalagem de um produto. Em seguida, passa a responsabilidade ao Modelo YOLO

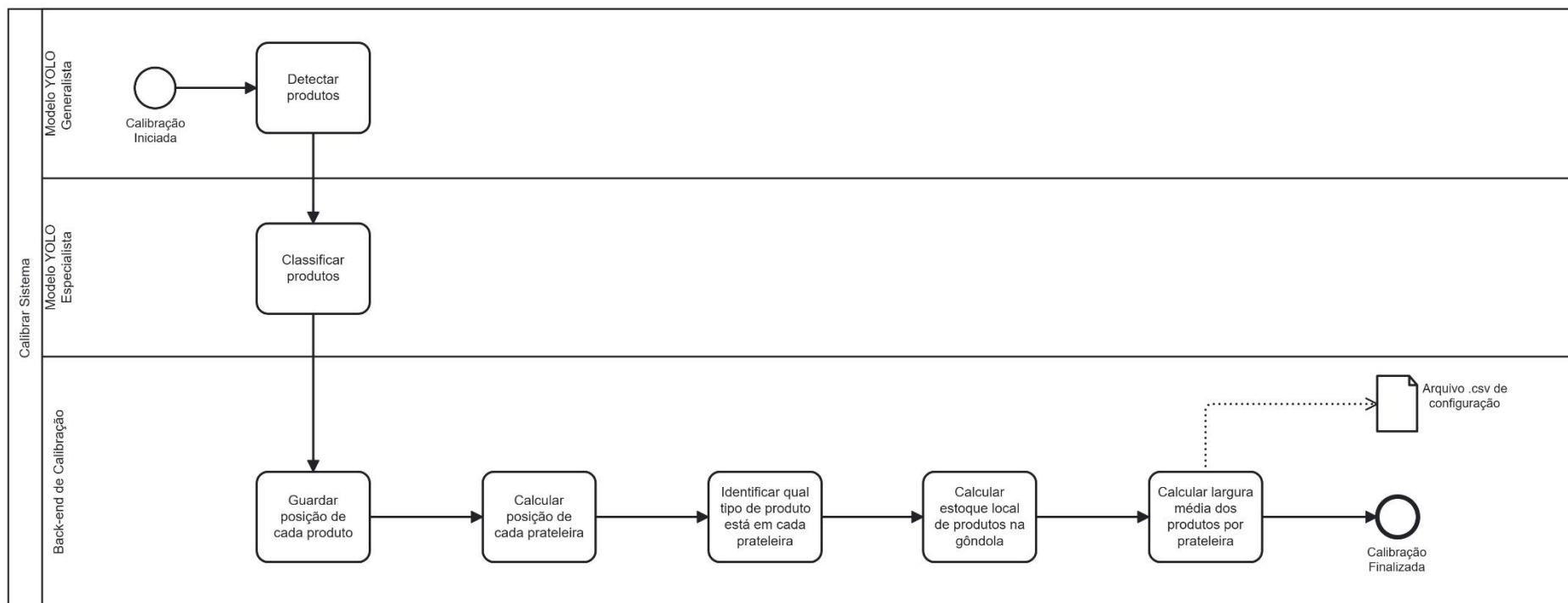


Figura 19 - Diagrama BPMN do processo de calibração
 Fonte: elaborado pelo próprio autor

Especialista, que é responsável em classificar os produtos mais em detalhes. Uma vez que os produtos tenham sido identificados na imagem, o *Back-end* de Calibração estará apto a extrair dados importantes como: a posição de cada produto específico nas prateleiras; a posição de cada uma das prateleiras em uma gôndola; realizar a associação e armazenamento de cada tipo específico de produto a uma respectiva prateleira; calcular o estoque local de produtos na gôndola; e calcular e armazenar a largura média de cada tipo de produto específico presente em uma prateleira. Este processo gera como *output* um arquivo .csv de configuração do sistema, que funciona como um *input* essencial para a atividade de reconhecimento de elementos.

4.2.3 Subprocesso de reconhecimento de elementos

A atividade de reconhecimento de elementos é uma etapa fundamental para detecção de entidades importantes neste sistema, uma vez que ele já esteja calibrado. A Figura 20 ilustra uma *pool* exclusiva para este subprocesso, que trabalha com quatro *lanes*: um modelo YOLO Padrão, o Modelo YOLO Generalista, o Modelo YOLO Especialista e com o *Back-end* de Reconhecimento. O modelo YOLO Padrão nada mais é que um modelo pré-treinado disponibilizado pela Ultralytics (mantenedora do YOLO) para detecção de objetos comuns do cotidiano, incluindo pessoa, uma entidade importante para ser detectada neste sistema (Jocher, 2023).

No início do processo de reconhecimento, o arquivo .csv de calibração do sistema inicialmente é carregado pelo *Back-end* de Reconhecimento a fim de que possa ser usado em etapas subsequentes deste processo. Em seguida, a responsabilidade é passada para o Modelo YOLO Padrão, que busca verificar se existem pessoas presentes na imagem. Caso exista, a responsabilidade é retornada ao *Back-end* de Reconhecimento, que interrompe o processo de detecção antes de reiniciar o processo de reconhecimento de elementos. O reconhecimento de pessoas no sistema é feito pois, em um processo de consumo de produtos, é natural que pessoas passem ou parem em frente à gôndola monitorada para realizar o consumo de produtos.

Caso pessoas não sejam detectadas em frente à gôndola, a responsabilidade é repassada ao Modelo YOLO Generalista, que busca detectar os espaços vazios presentes em cada prateleira. Este modelo também detecta os tipos de recipientes dos produtos, antes de passar a responsabilidade ao Modelo YOLO Especialista, que como já mencionado, realiza a classificação dos produtos mais em detalhes. Em seguida, a responsabilidade retorna ao *Back-end* de Reconhecimento, que associa cada espaço vazio detectado a sua respectiva prateleira.

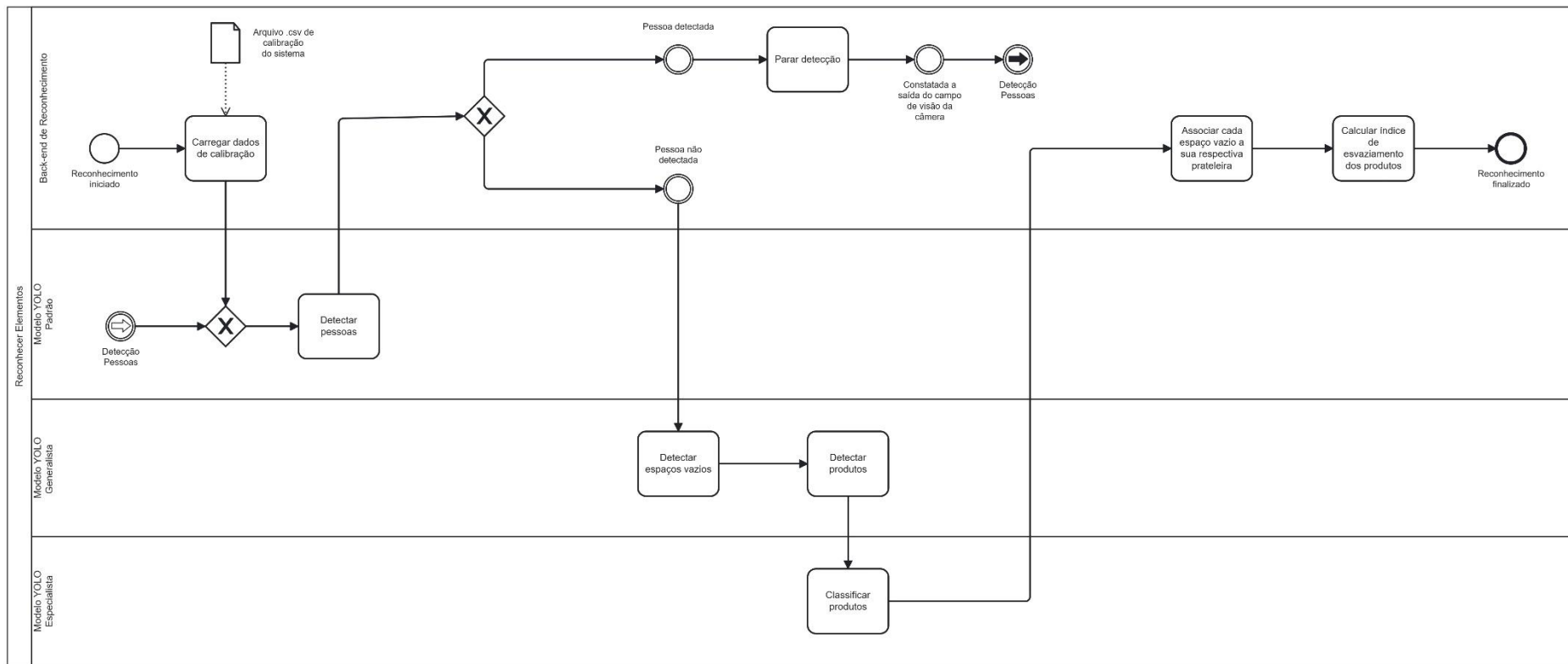


Figura 20 - Diagrama BPMN do processo de reconhecimento de elementos
 Fonte: elaborado pelo próprio autor.

Uma vez que esta etapa é realizada, o *Back-end* de reconhecimento será capaz de calcular o índice de esvaziamento dos diferentes produtos presentes nas prateleiras. Como *output* deste processo, os índices são armazenados em um BD Relacional, para que possam servir de *inputs* fundamentais de recursos presentes no *dashboard* proposto.

4.2.4 Proposta de *Front-end* (*Dashboard*)

Vale ressaltar que a parte do modelo geral que engloba as lógicas do *Back-end* juntamente com os modelos YOLO são capazes de realizar todo o processo de calibração e reconhecimento de elementos de forma independente de um supervisor. Porém, a fim de orientar os *stakeholders* sobre o consumo de produtos se baseando no índice de esvaziamento deles, um *dashboard* foi proposto para o GD.

De acordo com Shao, Kibira e Frechette (2024), por meio de fluxos de dados em tempo real ou quase real, suporte a BDs e painéis, um GD pode prover aos *stakeholders* uma forma de compreenderem o que vem acontecendo com seu correspondente físico por meio de textos, tabelas e/ou gráficos informativos (GD Descritivo) e até mesmo por meio de relatórios de previsão e tendências (GD Preditivo).

Por isso, ainda na etapa de Concepção, *mockups*⁸ de telas foram propostas pelo autor fazendo-se uso da ferramenta NinjaMock⁹, conforme ilustra a Figura 21. O objetivo foi propor um *dashboard* que, na sua tela principal (tela *HOME*, representado pelo ícone de uma casa, no topo da tela), os *stakeholders* sejam capazes de acessar de forma objetiva informações pertinentes à tomada de decisão.

Nesta versão do *dashboard*, 4 painéis estão sendo disponibilizados pelo autor, 3 painéis descritivos (capazes de ilustrar valores do estado das prateleiras em tempo real) e 1 painel preditivo (capaz de ilustrar cenários futuros após a execução de um algoritmo de previsão). Esta arquitetura em painéis modulares foi proposta para permitir que em trabalhos futuros novos painéis com informações pertinentes à tomada de decisão como *Key Performance Indicators* (KPIs), Simulações e Previsões específicas possam ser adicionados sob demanda, conforme ilustra a Figura 22.

Os 4 painéis disponibilizados nesta versão do *dashboard* são: um *Ranking*, uma Visão Geral, a Previsão Padrão e o Acompanhamento Individual dos Índices de Esvaziamento dos

⁸ *mockups* são representações visuais estáticas de um produto ou interface, usadas para demonstrar a aparência das aplicações antes de serem devidamente implementadas via código.

⁹ <https://ninjamock.com/>

produtos monitorados. Apesar de todos os painéis estarem presentes na tela *HOME*, os *stakeholders* têm a possibilidade de selecionar aquele ou aqueles que mais gostariam de visualizar por meio de uma configuração disponível na tela de Configurações (representado no topo da tela pelo ícone de uma engrenagem). Além disso, o ícone com símbolo de “+” no topo do *dashboard* permite adicionar sob demanda novos produtos para serem monitorados.

O painel de “*Ranking*” tem como objetivo apresentar de forma objetiva e ordenada os produtos que estão com os índices mais altos em um determinado momento. Por meio de uma tabela dinâmica, é possível identificar informações como: prioridade (quanto maior o valor de um índice, maior sua prioridade), valor do índice de esvaziamento do produto, nome do produto, sua localização dentro do estabelecimento e uma estimativa de disponibilidade do produto em sua respectiva prateleira. A estimativa de disponibilidade (em %) de um produto é representado pela equação 1.

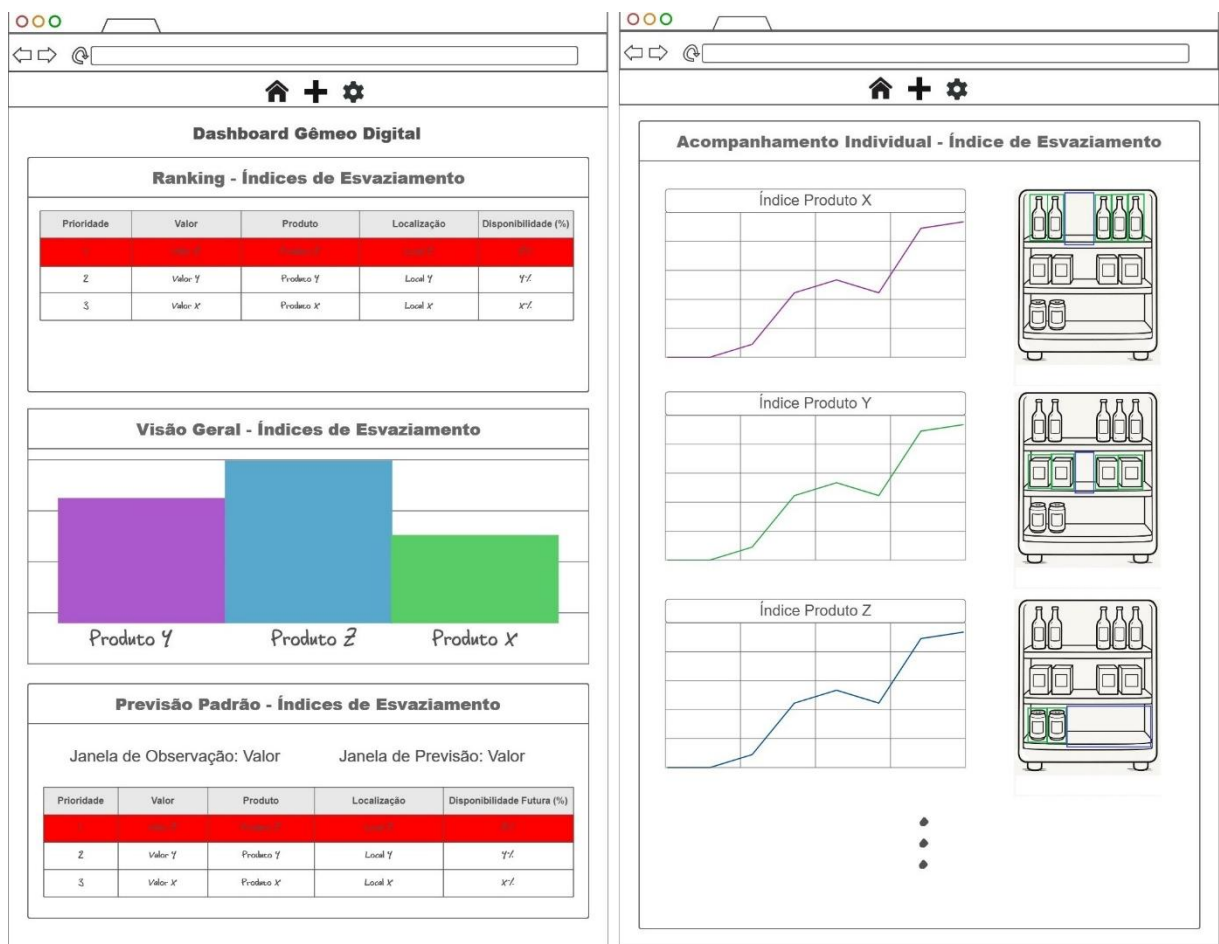


Figura 21 – *Mockup* do *Dashboard* proposto com painéis *Ranking*, *Visão Geral* e *Previsão Padrão* dos Índices de Esvaziamento

Fonte: elaborado pelo próprio autor

O produto com maior índice em um determinado momento terá sua linha destacada na cor vermelha. Produtos que se encontram abaixo da disponibilidade mínima configurada terão suas linhas pintadas na cor roxa. Na versão definitiva, legendas com os significados das cores serão disponibilizadas no próprio painel para serem sempre consultadas pelos *stakeholders* quando necessário. Além disso, neste painel também será possível ordenar os produtos monitorados pelas suas disponibilidades estimadas ao invés do índice de esvaziamento. Neste caso, itens de menor disponibilidade estimada aparecerão com maior prioridade na tabela. Maiores detalhes do cálculo do Índice de Esvaziamento de um produto serão apresentados na equação (7).

$$Disponibilidade\ Estimada\ (\%) = \left(1 - \left(\frac{\text{Índice de Esvaziamento}}{\text{Estoque total na prateleira}} \right) \right) * 100 \quad (1)$$

O painel de “Visão Geral” apresenta de forma gráfica (gráfico em barras) o índice de todos os produtos em monitoramento, permitindo que os *stakeholders* tenham uma visão geral deles, e que possam, também de forma objetiva, fazer comparações entre eles quando necessário. Já o painel de “Previsão Padrão” tem como finalidade apresentar informações pertinentes aos produtos monitorados após seus dados históricos serem automaticamente submetidos a um algoritmo de previsão, que é chamado a cada 1min utilizando de uma janela deslizante¹⁰ de dados durante a execução do *dashboard* e após atingir o tempo configurado da Janela de Observação.

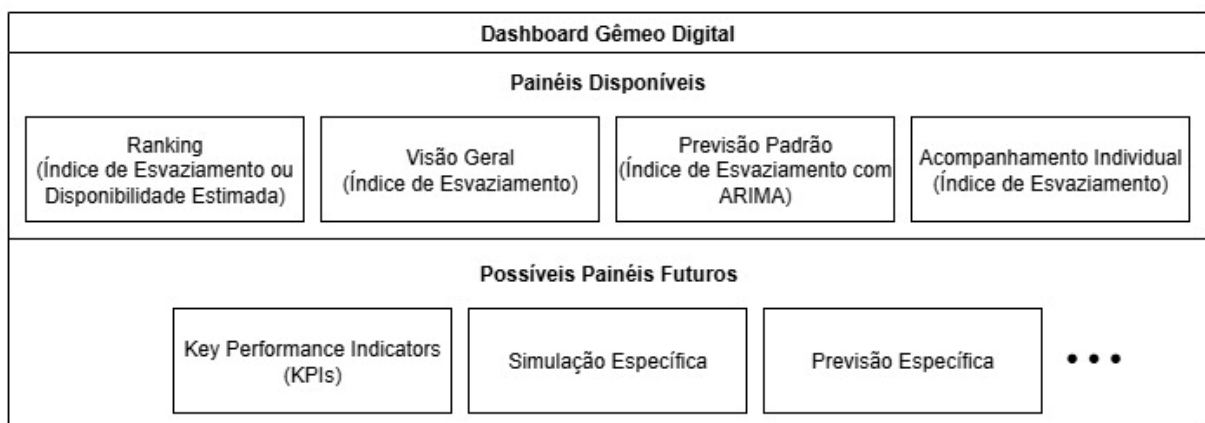


Figura 22 – Painéis disponíveis atualmente no *dashboard* do GD e possíveis painéis futuros que poderão ser adicionados

Fonte: elaborado pelo próprio autor

Este painel de previsão é chamado de “Padrão” pois pode ficar disponível na tela *HOME* uma vez que as Janelas de Observação e Previsão sejam configuradas pelos *stakeholders* na tela

¹⁰ Trata-se de uma técnica de análise temporal que mantém uma janela móvel de dados recentes dentro de uma Janela de Observação configurada.

de configurações. A Janela de Observação corresponde ao conjunto de dados históricos que serão utilizados para alimentar o(s) algoritmo(s) de previsão (Ex: dados dos últimos 40 min). Já a Janela de Previsão corresponde ao período futuro, ou seja, quanto tempo à frente os *stakeholders* gostariam de prever o estado dos produtos (Ex: próximos 15 min).

Uma vez que a “Previsão Padrão” é executada pela primeira vez, uma tabela dinâmica aparece dentro do painel e será possível identificar informações como: prioridade, valor do índice de esvaziamento futuro do produto, nome do produto, sua localização dentro do estabelecimento e qual a disponibilidade estimada futura de cada produto após a previsão. A coloração das linhas (vermelho ou roxo) segue a mesma ideia das cores apresentadas no painel de “*Ranking*”.

Já o painel de “Acompanhamento Individual” permitirá que os *stakeholders* visualizem em tempo real, e para cada produto monitorado, sua respectiva série temporal ilustrando a evolução do seu índice de esvaziamento ao longo do tempo. O objetivo é permitir uma investigação mais detalhada sobre o comportamento de consumo de um produto específico quando necessário. Além disso, ao lado de cada gráfico, é apresentada uma SD da prateleira onde o produto se encontra, buscando também prover um *feedback* visual constante aos *stakeholders* sobre o estado da prateleira no momento que o último índice do respectivo produto foi calculado.

4.3 Implementação

Nos subtópicos a seguir serão apresentados maiores detalhes dos materiais e métodos utilizados para implementação do GD proposto. Nos materiais, detalhes de componentes fundamentais tanto físicos quanto lógicos são apresentados. Dentre os componentes físicos, destacam-se a gôndola, a câmera e os computadores utilizados para treinamento dos modelos de VC e para execução do GD. Já para os componentes lógicos, destacam-se os recursos da linguagem Python utilizados para implementação do *Back-end* e *dashboard*, a versão e plataforma utilizadas para treinamento dos modelos YOLO e o BD Relacional utilizado.

Nos métodos, são discutidos maiores detalhes técnicos: do processo de captura e tratamento das imagens das prateleiras; dos *datasets* e das configurações de treinamento dos modelos de VC, além de uma discussão das métricas produzidas por eles; da implementação, dos cálculos e *outputs* produzidos tanto no processo de calibração quanto do processo de reconhecimento de elementos; da organização e utilização do BD Relacional; da implementação e do acesso ao *dashboard* para auxílio dos *stakeholders* na tomada de decisões.

4.3.1 Materiais

Em relação aos componentes físicos utilizados na implementação do GD, a gôndola utilizada no experimento é confeccionada e distribuída pela empresa SA MOVEIS. Ela é uma estrutura profissional utilizada em estabelecimentos varejistas para organização e exposição de produtos. Ela é do tipo Parede (estante usada em estabelecimentos que é posicionada encostada diretamente na parede de um ambiente), é feita de aço, tem as dimensões 170cm de altura, 80cm de largura e 40cm de profundidade e possui suporte para prateleiras capazes de suportarem até 35Kg cada.

A câmera utilizada é a Logitech C920, um dispositivo *Full HD* de 1080p a 30fps. Ela possui lente de vidro, foco automático e campo de visão de 78°¹¹. Possui dois microfones estéreo e correção automática de luz (RightLight2¹²), garantindo boa imagem mesmo em ambientes com pouca iluminação, segundo o fabricante. Possui conexão USB 2.0 e é compatível com os sistemas operacionais Windows, macOS, ChromeOS e Android. Ela é ideal para videochamadas e aplicações em tempo real. Seu preço gira em torno de US\$75 (consulta realizada em julho de 2025). A Figura 23 ilustra a gôndola e a câmera utilizadas no experimento, juntamente com a interação entre elas, que possui um espaçamento de 2,25 metros.



Figura 23 - Gôndola e câmera utilizadas no experimento e a interação entre elas
Fonte: elaborado pelo próprio autor

¹¹ A câmera "enxerga" o que está dentro de um cone de 78° partindo da lente.

¹² O RightLight 2 é uma tecnologia da Logitech que ajusta automaticamente brilho e contraste para melhorar a imagem em ambientes com pouca luz.

Para treinamento dos modelos de VC, foi utilizada uma máquina AMD Ryzen 9 5950X com 16 núcleos de 3.4GHz, 128 GB de memória RAM, unidade SSD de 2 TB, placa de vídeo NVIDIA RTX 3090 com 24GB de memória dedicada e sistema operacional Windows 11 Education. Já para execução do GD, composto pelos códigos do *back-end*, *front-end* e modelos de VC treinados, foi utilizado um computador *desktop* com processador Intel Core i7 de 7ª geração e 4.2GHz, 32 GB de memória RAM, unidade SSD de 512 GB, placa de vídeo NVIDIA GeForce GTX 1050Ti com 4GB de memória dedicada e sistema operacional Windows 10.

Em relação aos componentes lógicos utilizados na implementação do GD, as lógicas do *back-end* utilizam da linguagem Python v3.10, juntamente com suas bibliotecas *Numerical Python* (NumPy) v1.23.2 e *Pandas* v2.2.3. Outras ferramentas importantes utilizadas são a Plataforma Roboflow, o *Open Source Computer Vision Library* (OpenCV) v4.10.0.84, o YOLOv8.3.4m, o Pmdarima v2.0.4 e o Banco de Dados MySQL v8.0.37 com MySQL Connector v9.1.0 para Python. Já para as lógicas do *dashboard* é utilizada a biblioteca Dash. Maiores detalhes das implementações tanto do *back* quanto do *front-end* serão apresentados no subtópico 4.3.2.

O NumPy é utilizado para oferecer alto desempenho em operações numéricas que envolvam *arrays* multidimensionais como vetores e matrizes, estruturas estas fundamentais para armazenamento e tratamento de dados em tempo de execução. Já o Pandas simplifica o tratamento, análise e limpeza dos dados por meio do oferecimento de recursos profissionais para manipulação de dados. Ambas são muito úteis quando se trabalha com grandes volumes de dados em Python (Sapre; Vartak, 2020).

O Roboflow é uma plataforma *online* que oferece recursos importantes para gerenciamento de *datasets* e anotação de imagens em tarefas voltadas a VC. Ela permite importar, visualizar e organizar conjuntos de imagens que serão fundamentais no treinamento de um modelo VC. Ao oferecer recursos de anotação de imagens assistida por IA e automação automática de dados, ela se torna uma ferramenta indispensável para desenvolvedores e pesquisadores que queiram ter máxima produtividade na organização de seus *datasets*. Além disso, após o pré-processamento dos *datasets*, a plataforma permite exportá-los automaticamente para o formato exigido por versões específicas do YOLO, como o YOLOv8, versão utilizada neste trabalho (Kale *et al.*, 2024).

O OpenCV, mais particularmente a biblioteca cv2 (OpenCV para Python), é uma biblioteca de código aberto para VC usada para processamento de imagens e vídeos. Seus

recursos são fundamentais para integrar modelos como o YOLOv8 em aplicações práticas. Neste projeto, embora o cv2 não seja utilizado para detecção de objetos, ele dá suporte a todo o fluxo de entrada e saída de dados do processo de detecção, como a captura de imagens ou vídeos em tempo real, pré-processamento dos dados, visualização e salvamento dos resultados com marcações gráficas (como caixas delimitadoras, rótulos das classes, pontuações de confiança, etc). Essa capacidade de manipular imagens e vídeos com eficiência torna o OpenCV uma ferramenta indispensável no desenvolvimento de soluções baseadas em VC (Gaikwad, 2023).

Como já mencionado e melhor detalhado no subtópico 2.3 deste trabalho, a versão do YOLO utilizada para dar suporte aos recursos de VC do GD proposto é o YOLOv8. Decidiu-se pela utilização desta versão por se tratar de uma versão ainda recente, bem documentada e com resultados de detecção competitivos em diversas tarefas de VC comparada a outras versões do modelo (Ultralytics, 2024).

Dentre as variações do YOLOv8 disponíveis, foi utilizada a versão YOLOv8m, tanto para o modelo pré-treinado quanto para os modelos customizáveis. A principal vantagem desta variação é oferecer um equilíbrio entre acurácia e desempenho, tornando-o adequado para aplicações em tempo real que exigem precisão, mas sem demandar máquinas de alto desempenho (Ultralytics, 2023).

Para previsão de valores futuros nesta versão do GD, mais precisamente envolvendo séries temporais, o modelo ARIMA inicialmente foi utilizado. Na linguagem Python, a escolha dos melhores parâmetros e a melhor variação do modelo ARIMA a ser utilizado pode ser feito por meio do uso da função *auto_arima*, disponível no pacote pmdarima. De acordo com Mesquita *et al.* (2023), o uso desta função poupa o desenvolvedor de executar e analisar manualmente testes implícitos à execução do modelo ARIMA, como a execução de funções de autocorrelação, autocorrelação parcial, testes de estacionaridade, etc.

Al-Qazzaz (2022) também ressalta que, ao minimizar a intervenção humana nas etapas de análise e ajuste do melhor modelo, o *auto_arima* reduz o risco de erros por interpretação incorreta dos dados, além de acelerar consideravelmente o processo de modelagem, sendo ideal para conjuntos de dados grandes ou aplicações em tempo real.

O MySQL é um Sistema Gerenciador de Banco de Dados Relacional (SGBDR) de código aberto, amplamente utilizado para armazenar, consultar e gerenciar dados estruturados em sistemas. Ele organiza os dados em tabelas compostas por linhas e colunas, permitindo não só o armazenamento de dados em massa nestas tabelas como também o relacionamento entre

elas. Entre suas principais vantagens, destacam-se a alta performance, escalabilidade, confiabilidade, facilidade de uso e boa documentação (Šušter; Ranisavljević, 2023).

No GD proposto, o BD MySQL funciona como uma peça estratégica, atuando como um *middleware* capaz de armazenar dados relacionados ao processo de detecção (como espaços vazios detectados), dados resultantes dos cálculos realizados pelo *back-end* (como cálculo do índice de esvaziamento dos produtos monitorados) e dados históricos úteis que alimentam os algoritmos de previsão.

Já o Dash é um *framework* em Python voltado para criação de aplicações interativas voltadas à visualização de dados analíticos. Ela facilita a construção de *dashboards* interativos com gráficos dinâmicos, componentes visuais personalizáveis e interface responsiva. Além disso, possui grande compatibilidade com bibliotecas utilizadas neste projeto como o NumPy e o Pandas (Jin *et al.*, 2023).

Essas características tornam o Dash uma escolha estratégica para aplicações de ciência de dados e engenharia em Python, como é o caso deste projeto, que visa apresentar visualizações dinâmicas e oferecer sugestões de tomada de decisão objetivas aos *stakeholders* por meio do *dashboard* criado fazendo uso deste *framework*.

Como pôde ser visto, a *stack* de desenvolvimento¹³ deste trabalho se baseia na linguagem Python. Isso se justifica pelo fato de a linguagem prover um ecossistema único e rico em recursos voltados à conectividade e interatividade em IoT, IA, M&S, análise estatística, armazenamento e análise de dados, VC, otimização, entre muitos outros recursos que facilitam a criação de soluções baseadas em GDs (Agarwal; Calloway; Niederhut, 2023). O objetivo do autor é reduzir atritos no uso de recursos heterogêneos para se propor a solução, e principalmente, facilitar a reprodutibilidade e escalabilidade da solução proposta.

A Quadro 5 resume as principais características dos componentes lógicos explicados e utilizados no desenvolvimento do GD proposto, destacando seus nomes, versões, licenças de uso e *links* de acesso oficiais. Todos podem ser usados de forma gratuita para se criar, atualizar e operar o GD proposto, seja para experimentos acadêmicos ou para uso comercial, desde que se atenda às exigências das licenças de uso.

¹³ É o conjunto de recursos utilizados para se construir, testar, implantar e manter um *software*

Quadro 5 - Principais componentes lógicos utilizados na implementação do GD

| Nome | Versão | Licença | Link |
|--------------------|-----------|-----------------------|---|
| Python | 3.10 | PSFL | https://www.python.org/downloads/release/python-3100/ |
| NumPy | 1.23.2 | BSD | https://numpy.org/ |
| Pandas | 2.2.3 | BSD | https://pandas.pydata.org/ |
| Roboflow | 0.40.0 | Plano Público | https://roboflow.com/ |
| OpenCV | 4.10.0.84 | Apache License 2.0 | https://opencv.org/ |
| YOLO | 8.3.4 | AGPL-3.0 | https://yolov8.com/ |
| pmdarima | 2.0.4 | MIT License | https://alkaline-ml.com/pmdarima/ |
| MySQL | 8.0.37 | GPL-v2 | https://dev.mysql.com/ |
| MySQL Connector | 9.1.0 | GPL-v2 | https://dev.mysql.com/doc/connector-python/en/ |
| Dash | 2.18.2 | MIT License | https://dash.plotly.com/ |

Fonte: elaborado pelo próprio autor

4.3.2 Métodos

Este subtópico discute maiores detalhes: da preparação dos *datasets* dos modelos de VC, do treinamento customizado dos modelos YOLO Generalista e Especialista e do uso do modelo YOLO Padrão; do tratamento das imagens capturadas e do sistema de calibração do GD; do processo de detecção de espaços vazios e produtos, da associação produto-prateleira e do cálculo dos índices de esvaziamento; da organização do BD e do seu uso estratégico para armazenamento de dados; do *dashboard* proposto para auxiliar os *stakeholders* na tomada de decisão.

4.3.2.1 Preparação e Treinamento dos Modelos de Visão Computacional

A fim de que o GD possa ser capaz de detectar pessoas, espaços vazios, diferentes formatos de produtos e produtos específicos, é necessário que um modelo YOLO Padrão seja utilizado e que os modelos YOLO Generalista e Especialista sejam inicialmente treinados para que em seguida também possam ser utilizados. Como já mencionado, utilizou-se a abordagem de fusão hierárquica de modelos neste trabalho visando modularizar as responsabilidades dos modelos e alcançar melhores resultados na detecção dos elementos. A Figura 24 ilustra a fusão hierárquica destes modelos.

Como mencionado, o GD é capaz de detectar o trânsito de pessoas em frente às prateleiras. Quando isso acontece, ele deve interromper a detecção de produtos e espaços vazios momentaneamente até que a visualização da gôndola pelo GD volte a ser nítida. Para a detecção de pessoas, o modelo Padrão YOLOv8m pré-treinado com o *dataset* COCO foi utilizado (Ultralytics, 2023). Apenas para esta classe, 250000 anotações foram utilizadas pelos desenvolvedores para realizar o treinamento. Apesar deste modelo Padrão ser capaz de detectar 80 classes diferentes de objetos do cotidiano, neste experimento ele foi configurado para detectar apenas a classe pessoa (*person*).

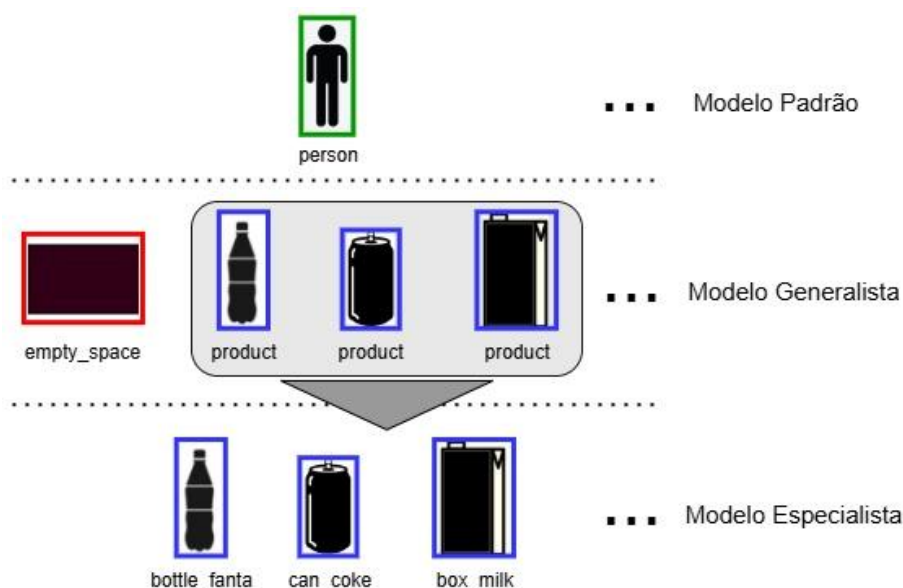


Figura 24 - Fusão Hierárquica dos Modelos utilizados neste trabalho
 Fonte: elaborado pelo próprio autor

Antes mesmo da realização do treinamento dos modelos YOLO Generalista e Especialista, é necessário que *datasets* com imagens dos espaços vazios, dos diferentes formatos de produtos e dos produtos específicos que se deseja detectar sejam levantadas, organizadas, anotadas e exportadas para o padrão compreensível pelo YOLOv8. No modelo Generalista, imagens de 2 classes foram levantadas: espaço vazio (no modelo, descrita como a classe *empty_space*) e formato do produto (classe *product*). No modelo Especialista, imagens de 3 tipos específicos de produtos foram levantadas: caixa de leite (classe *box_milk*), lata de Coca-Cola (classe *can_coke*) e garrafa de Fanta (classe *bottle_fanta*). O padrão “embalagem_producto” nas classes do modelo Especialista busca representar de forma objetiva o formato da embalagem e o tipo específico do produto monitorado, respectivamente.

O *dataset* levantado para detecção do formato dos produtos no modelo Generalista foi composto por 2.475 imagens. Deste número, 1.275 foram adicionadas pelo autor e 1.200 foram

extraídas de *datasets* públicos disponíveis na plataforma Roboflow. O Quadro 6 ilustra os nomes e respectivos *links* de acesso aos *datasets* externos de onde as imagens foram extraídas. Para este experimento, o modelo Generalista foi treinado inicialmente para identificar produtos que possuam o formato de latas, garrafas e caixas do tipo tetra pak. Porém, é importante ressaltar que futuramente poderá ser treinado para identificar novos formatos de embalagens.

Quadro 6 – Bases de dados externas de produtos de onde foram extraídas imagens para treinamento do modelo Generalista

| Nome | Link de acesso |
|---|---|
| <i>Can Drink Detection</i> | https://universe.roboflow.com/philogun/can-drink-detection |
| <i>Bottle Computer Vision Model</i> | https://universe.roboflow.com/ml-fwwom/bottle-cqnb1 |
| <i>Tetrapacks Computer Vision Dataset</i> | https://universe.roboflow.com/lemon-dataset-h2ime/tetrapacks |

Fonte: elaborado pelo próprio autor

Também no modelo Generalista, o *dataset* utilizado para detecção de espaços vazios foi composto por 650 imagens levantadas pelo autor, todas retiradas da gôndola de onde os testes foram realizados. Já para o modelo Especialista, cada produto teve um *dataset* composto inicialmente por 250 imagens levantadas pelo autor. Estas quantidades foram definidas se baseando em testes empíricos realizados pelo autor, em que após os treinamentos foi possível verificar uma boa capacidade de detecção.

A fim de otimizar o tempo de geração das imagens, vídeos com duração de em média 2 minutos foram gravados para se extrair imagens para cada classe. Durante a gravação dos vídeos, os produtos tiveram suas disposições modificadas, a fim de se gerar imagens em diferentes escalas, ângulos, oclusões e condições de iluminação, enriquecendo assim o nível de detalhes dos *datasets*. A Figura 25 ilustra um mosaico contendo exemplos de imagens presentes nestes *datasets*.

Uma vez que os *datasets* foram gerados e organizados em pastas, o próximo passo foi utilizar da plataforma Roboflow para se realizar o processo de anotação de suas imagens. Nesta plataforma, a anotação das imagens presentes nos *datasets* podem ser realizadas de forma manual, automática ou semiautomática. Por meio do recurso *Auto Label*, lançado em março de 2024 na plataforma, o processo de anotação de imagens ficou mais simplificado, podendo diminuir o tempo de anotação em até 50% ao se fazer uso deste recurso (Roboflow, 2025).

O *Auto Label* utiliza modelos grandes e pré-treinados em vastos conjuntos de dados, como o Grounding Dino, para localizar padrões de objetos dentro em uma imagem com base em descrições textuais (Roboflow, 2025). Neste trabalho, o *Auto Label* foi utilizado de forma semiautomática, uma vez que ao submeter os *datasets* a este recurso, nem todos os produtos e espaços vazios foram devidamente anotados. Para contornar este problema, após a passagem do *Auto Label* a plataforma permite que revisões sejam feitas sobre as figuras anotadas, permitindo que objetos não anotados automaticamente sejam anotados manualmente.



(a) Exemplos de imagens de treinamento do Modelo Generalista

(b) Exemplos de imagens de treinamento do Modelo Especialista

Figura 25 - Exemplo de imagens presentes no treinamento dos modelos (a) Generalista e (b) Especialista
Fonte: elaborado pelo próprio autor

Após a anotação das imagens presentes em cada *dataset*, a plataforma permite dividir os conjuntos de dados em treino, validação e testes. Neste projeto, todos o *datasets* preparados utilizaram inicialmente a divisão padrão indicada pelo Roboflow: 70% das imagens para treinamento, 20% das imagens para validação e 10% para testes. No treinamento de um modelo de VC, os dados de treinamento são usados ao longo das épocas para ajustar os pesos do modelo. Os dados de validação são usados no final de cada época para monitorar o desempenho do treinamento e evitar *overfitting*¹⁴. Já os dados de teste medem o desempenho final do modelo quando submetido a dados inéditos.

Em seguida, a plataforma oferece recursos de pré-processamento nas imagens, como o *Auto-Orient* e o *Resize*, utilizados neste projeto. O *Auto-Orient* garante que todas as imagens

¹⁴ *Overfitting* é quando o modelo aprende demais os detalhes e ruídos do conjunto de treino, perdendo capacidade de generalizar para novas imagens.

estejam rotacionadas corretamente com base nos metadados *Exchangeable Image File Format* (EXIF). Já o *Resize* padroniza as dimensões das imagens dos *datasets*, que neste projeto, são de 640 x 640 pixels. Estes ajustes são fundamentais para reduzirem inconsistências no processo de treinamento e se adequarem às exigências do YOLOv8 para que possa oferecer um bom equilíbrio em velocidade de inferência e precisão (Ultralytics, 2023).

Outro recurso importante oferecido pela plataforma Roboflow e utilizado na preparação dos *datasets* deste projeto é o aumento artificial de dados (no inglês, *data augmentation*). Esta é uma técnica que aplica transformações sobre as imagens de um *dataset* como brilho, corte e rotação, aumentando a diversidade do conjunto de treinamento e buscando melhorar a generalização de um modelo de VC (Ultralytics, 2025a).

São inúmeras as formas de aumento artificial de dados oferecidas pela plataforma Roboflow. Neste projeto, foram utilizadas as configurações *Flip*, *90° rotate*, *Brightness* e *Noise*, que criam respectivamente variações das imagens invertidas verticalmente, horizontalmente, com variações na iluminação e com ruídos (imperfeições) (Ultralytics, 2025a). Após a aplicação do *data augmentation*, o número de imagens de treino de cada *dataset* foi duplicado e a nova proporção entre treino, validação e teste dos *datasets* foi alterada para 82%, 12% e 6%, respectivamente.

Uma vez que os *datasets* foram organizados e pré-processados, seguiu-se para a etapa de treinamento dos modelos de VC. Mesmo para o treinamento de um modelo de VC customizado, é recomendável que um modelo base do YOLOv8 seja carregado para se realizar o *transfer learning* para os novos modelos. O *transfer learning* é uma técnica de aprendizado de máquina em que um modelo desenvolvido para uma tarefa mais genérica é reutilizado como ponto de partida para outro modelo mais específico, aproveitando padrões, pesos e características aprendidas para economizar tempo de treinamento e recursos computacionais (Ultralytics, 2025b). Como já mencionado, o YOLOv8m foi utilizado como modelo base neste trabalho.

Para o treinamento tanto do modelo Generalista como Especialista, os seguintes parâmetros e seus respectivos valores apresentados no Tabela 1 foram utilizados. O arquivo *data.yaml* contém os caminhos para os conjuntos de treino e validação, e informações dos nomes e quantidade de classes de cada modelo. As épocas (do inglês, *epochs*) são o número de vezes que todo o conjunto de dados de treino é processado, buscando melhorar a precisão do modelo ao repetir os aprendizados sobre os dados. Decidiu-se por utilizar um valor alto de

épocas (3000) a fim de buscar alcançar uma boa otimização dos modelos (o valor padrão é 100 para *datasets* grandes, com milhares de imagens).

Tabela 1 - Parâmetros de treinamentos dos modelos de VC Generalista e Especialista

| Parâmetro | Valor |
|-----------------|--|
| <i>data</i> | Localização do arquivo <i>data.yaml</i> na máquina de treinamento. |
| <i>epochs</i> | 3000 |
| <i>imgsz</i> | 640 |
| <i>device</i> | 0 |
| <i>batch</i> | -1 |
| <i>plots</i> | <i>True</i> |
| <i>patience</i> | 200 |
| <i>save</i> | <i>True</i> |

Fonte: elaborado pelo próprio autor

O *imgsz* é o valor padrão do tamanho das imagens (em *pixels*) para o treinamento e validação dos modelos, que neste trabalho, é 640x640px. O *device=0* significa que a GPU principal da máquina de treinamento será utilizada para realização do treinamento dos modelos. O *batch* significa o número de imagens processadas por vez antes de atualizar os pesos dos modelos. Neste caso, o *batch=-1* significa um ajuste automático pelo próprio YOLO para usar o máximo de imagens dentro da disponibilidade de memória da GPU. O *plots=True* permite gerar gráficos com métricas importantes para avaliação do modelo e que serão discutidas no subtópico 4.4 deste trabalho.

O *patience* é um parâmetro do tipo *early stopping*, ou seja, ele define quantas épocas consecutivas o treinamento pode prosseguir sem apresentar melhoras significativas nas métricas de validação. Ele evita desperdício de tempo e ajuda prevenir o *overfitting*, ou seja, impede que o modelo continue aprendendo padrões irrelevantes ou fique viciado apenas nos dados de treino.

No treinamento dos modelos deste trabalho, o YOLO interrompe o treinamento antes das 3000 épocas se não houver melhora por 200 épocas consecutivas. Por fim, o parâmetro *save=True* permite que o modelo ao longo do treinamento vá criando *checkpoints* dos resultados parciais para que, caso aconteça algum problema no treinamento, seja possível

retornar de onde parou. Mais detalhes sobre os parâmetros utilizados e demais parâmetros podem ser encontrados em Ultralytics (2024b).

Após o treinamento dos modelos Generalista e Especialista, foram coletados os tempos totais de treinamento e o número de épocas em que os modelos atingiram os melhores resultados de detecção. Estes valores são apresentados na Tabela 2. É importante ressaltar que, apesar de inicialmente os treinamentos terem sido configurados para trabalharem com 3000 épocas, é possível observar que os modelos atingiram os melhores resultados de detecção bem antes de atingirem o número total de épocas configurado.

O Modelo Generalista executou 468 épocas, mas a partir de 268 épocas não houve melhoras significativas. Já o Modelo Especialista executou 724 épocas, mas a partir de 524 épocas também não houve melhoras significativas. Lembrando que isso acontece pois os treinamentos dos modelos foram configurados para atuarem com *patience* = 200. Maiores detalhes das métricas produzidas pelos modelos serão discutidos no subtópico 4.4.

Tabela 2 - Tempo de treinamento e número de épocas em que os modelos atingiram os melhores resultados de detecção

| | Modelo Generalista | Modelo Especialista |
|-----------------------------|---------------------------|----------------------------|
| Tempo de Treinamento | 2h 59min 30seg | 2h 50min 38seg |
| Número de Épocas | 268/468 | 524/724 |

Fonte: elaborado pelo próprio autor

Além dos detalhes apresentados acima para preparação e treinamento dos modelos de VC, é importante ressaltar a importância do manual técnico presente no Apêndice B neste trabalho. Mais do que um material de apoio, ele busca estruturar de forma clara, sequencial e ilustrada todo o processo de reconfiguração dos submodelos de VC, contemplando a organização dos dados, a parametrização, o retreinamento e a validação deles.

Ao sistematizar esse conhecimento, o manual transforma a adaptação do GD em um procedimento replicável, metodologicamente orientado e acessível, permitindo sua aplicação em estabelecimentos distintos sem comprometer a coerência conceitual do modelo. Dessa forma, ele não apenas complementa a proposta, mas viabiliza, na prática, seu caráter adaptativo e escalável, contribuindo de maneira decisiva para a disseminação e democratização do uso do GD em diferentes realidades do varejo.

4.3.2.2 Processo de Calibração do GD

Antes do GD operar de forma efetiva, é necessário que ele seja calibrado. O arquivo de calibração é um *output* essencial desta etapa pois associa os produtos monitorados às suas respectivas prateleiras. Em suma, a calibração identifica as extremidades horizontais e verticais da gôndola, pois por meio destas medidas é possível calcular a altura de cada prateleira, identificar os produtos que se encontram em cada uma delas (realizar associação produto-prateleira) e estimar seus estoques locais (ou seja, a quantidade de um produto específico que está sendo monitorado em uma gôndola).

O processo de calibração se inicia com a filmagem em tempo real da gôndola e seus respectivos produtos devidamente organizados nas prateleiras. Este vídeo, antes de ser devidamente analisado, é redimensionado para a resolução 640x640 pixels, resolução na qual os modelos de VC deste trabalho foram treinados para realizarem as detecções.

Para a realização das detecções, é importante ressaltar mais uma vez que dois modelos diferentes estão sendo utilizados: o primeiro modelo (Generalista) é responsável por detectar o formato dos produtos e espaços vazios. Já o segundo modelo (Especialista), a partir do recorte de um produto detectado, verifica se o produto é uma garrafa de Fanta, uma lata de Coca-Cola ou uma caixa de leite.

Uma vez que a detecção foi iniciada e os tipos de produtos foram detectados pelo modelo Especialista de VC, o usuário é questionado sobre as seguintes informações: qual o número de prateleiras presentes na gôndola, qual a profundidade (em unidades) de cada tipo de produto presente nas prateleiras e um texto indicando a localização do produto (ex: Corredor 7, Gôndola 1).

A profundidade fornecida de cada produto é multiplicada pela quantidade de produtos detectados horizontalmente pelo modelo de VC em cada prateleira, pois assim é possível calcular a quantidade de produtos em cada uma delas. Com os dados inseridos e a quantidade dos produtos nas prateleiras calculados, todas as posições dos dois pontos $P1(x_1, y_1)$ e $P2(x_2, y_2)$ presentes nas caixas delimitadoras dos produtos detectados são salvas a fim de dar suporte ao cálculo da posição de cada uma das prateleiras na gôndola.

Com os pontos P1 e P2 das caixas delimitadoras armazenados, encontra-se os valores de x que estão mais à esquerda e mais à direita da gôndola, permitindo assim calcular a largura de cada prateleira. Da mesma forma, este processo é feito para os valores de y , para assim ter-se uma estimativa da localização da prateleira mais baixa e da mais ao topo.

A Figura 26 ilustra um exemplo da localização dos pontos P1 e P2 presentes em produtos de uma gôndola hipotética. Como pode ser visto, eles se localizam respectivamente no canto superior esquerdo e canto inferior direito de cada caixa delimitadora. Os cálculos da largura e altura, ilustradas na Figura 26, são apresentados nas equações (2) e (3), respectivamente. Vale ressaltar que os valores y da base da gôndola são numericamente maiores que os valores do topo.

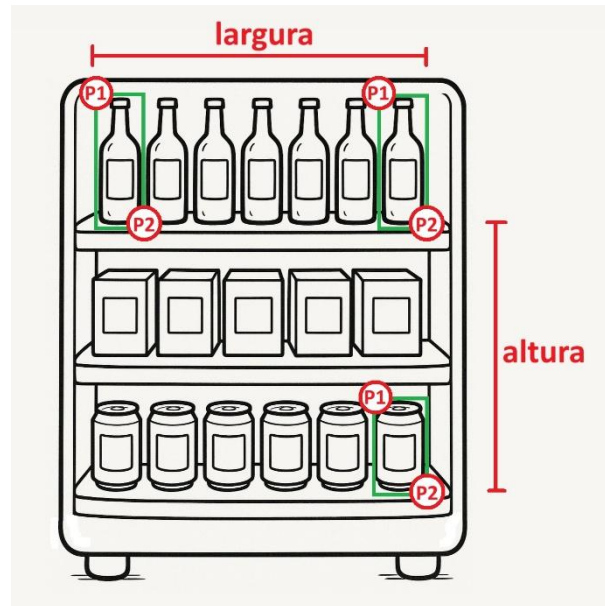


Figura 26 - Localização dos pontos P1 e P2 nas caixas delimitadoras de produtos localizados nos extremos da gôndola

Fonte: elaborado pelo próprio autor

$$largura = x_{2 \text{ direita}} - x_{1 \text{ esquerda}} \quad (2)$$

Onde:

- $x_{2 \text{ direita}}$ se encontra dentro do ponto P2 presente na caixa delimitadora do produto mais à direita da gôndola;
- $x_{1 \text{ esquerda}}$ se encontra dentro do ponto P1 presente na caixa delimitadora do produto mais à esquerda da gôndola;

$$altura = y_{2 \text{ baixo}} - y_{2 \text{ topo}} \quad (3)$$

Onde:

- $y_{2 \text{ baixo}}$ se encontra dentro do ponto P2 presente na caixa delimitadora do produto mais abaixo da gôndola;

- $y_{2_{topo}}$ se encontra dentro do ponto P2 presente na caixa delimitadora do produto mais ao topo da gôndola;

Uma vez que a altura entre a prateleira da base e do topo é calculada, em seguida uma estimativa é feita para se determinar o espaçamento entre elas, levando em consideração que este espaçamento é equidistante. A equação (4) ilustra este cálculo. Com o número total de prateleiras já fornecido pelo usuário anteriormente, a altura já calculada é dividida pelo número de prateleiras subtraído de 1. Com o cálculo do espaçamento realizado, agora é possível calcular a localização de cada prateleira, que é ilustrado pela equação (5).

A altura de uma prateleira x (a primeira prateleira é a de número 0) é definida como o espaçamento multiplicado pelo número da prateleira, somada à altura da prateleira mais ao topo. Estes valores são salvos, pois serão essenciais para posterior associação de cada produto à sua respectiva prateleira.

$$espaçamento = \frac{altura}{(n_{prateleiras} - 1)} \quad (4)$$

$$\sum_{x=0}^{num_{prateleira}-1} altura_{prateleira\ x} = altura_{base} - (espaçamento * x) \quad (5)$$

Uma vez que os cálculos das alturas de cada prateleira são realizados, novas detecções são feitas nos produtos presentes na gôndola para que as coordenadas de suas caixas delimitadoras sejam novamente extraídas. Uma iteração é feita a fim de comparar a altura y_2 da caixa delimitadora de cada produto com o vetor de alturas das prateleiras.

Se a diferença entre os valores da altura y_2 de uma caixa delimitadora e a altura de uma prateleira for de no máximo 10%, significa que, a partir da estimativa, aquele produto pertence à prateleira correspondente. Esta etapa é crucial para se realizar a associação produto-prateleira e também para dimensionar corretamente os espaços vazios de cada prateleira futuramente.

Uma vez que a associação produto-prateleira foi realizada, é calculado o estoque local de cada um dos produtos presentes na gôndola. Este cálculo é feito realizando a multiplicação entre o estoque de um produto em uma prateleira e o total de prateleiras que possuem este respectivo produto. Uma média da largura de cada um dos produtos também é calculada, pois estes valores auxiliam no cálculo dos índices de esvaziamento de cada produto em etapas futuras.

Por fim, os dados que serão utilizados posteriormente como suporte nas etapas de detecção de produtos e espaços vazios são organizados verticalmente e salvos em um arquivo .csv. A descrição dos dados armazenados no arquivo de configuração com exemplo de valores produzidos é apresentada na Tabela 3.

Tabela 3 - Dados presentes no arquivo de calibração do GD com valores de exemplo

| Dado | Exemplo |
|--|--|
| Número de prateleiras monitoradas da gôndola | 3 |
| Posição y (altura) de cada prateleira (da base ao topo, em pixels) | 490 353 216 |
| A média de largura dos produtos de cada prateleira | 34 36 51 |
| Nome do produto associado a cada prateleira | bottle_fanta can_coke box_milk |
| Posição mais à esquerda da gôndola (valor de x, em pixels) | 97 |
| Posição mais à direita da gôndola (valor de x, em pixels) | 557 |
| Lista de dicionários contendo dados dos produtos no seguinte padrão: “classe do produto”:estoque total do produto | [{'box_milk': '36'}, {'bottle_fanta': '84'}, {'can_coke': '60'}] |
| Lista de dicionários contendo dados da localização dos produtos no seguinte padrão: “classe do produto”: “localização” | [{'box_milk': 'Gondola1'}, {'bottle_fanta': 'Gondola1'}, {'can_coke': 'Gondola1'}] |

Fonte: elaborado pelo próprio autor

4.3.2.3 Processo de Reconhecimento de Elementos

Uma vez calibrado, o GD estará apto a realizar o processo de reconhecimento de elementos (pessoas, produtos e espaços vazios nas prateleiras) de forma adequada. Esta etapa é fundamental para se produzir dados que possibilitam calcular o índice de esvaziamento dos produtos que estão sendo monitorados.

Este índice leva em consideração os *gaps* detectados nas prateleiras para se ter uma estimativa geral de como estes produtos vem sendo consumidos em tempo real localmente nestas estruturas. Esta abordagem é uma extensão do trabalho proposto e publicado por Mesquita *et al.* (2024).

No processo de reconhecimento de elementos, inicialmente o arquivo de calibração é carregado a fim de ser utilizado pelo *Back-end* de Reconhecimento e por armazenar os primeiros dados no BD Relacional do GD. O BD é uma estrutura fundamental para armazenar dados importantes de configuração do *dashboard*, das detecções dos espaços vazios em tempo real, dos cálculos realizados e de dados históricos.

A Figura 27 ilustra um *Enhanced Entity-Relationship Diagram* (EER) da organização deste BD, que neste experimento é formado por 5 tabelas: *product*, *empty_spaces*, *consumption*, *consumption_forecasting* e *settings*. A tabela *product* registra dados básicos dos produtos específicos que estão sendo monitorados dentro da gôndola. Ela possui um relacionamento 1 para N¹⁵ com a tabela *empty_spaces*, outro relacionamento 1 para N com a tabela *consumption* e também um relacionamento 1 para 1¹⁶ com a tabela *consumption_forecasting*.

A tabela *empty_spaces* é responsável por registrar todas as ocorrências de espaços vazios detectados pelo *Back-end* de Reconhecimento. Seus dados servem de base para se calcular o índice de esvaziamento de cada produto e não permitir que ocorram duplicatas das caixas delimitadoras dos espaços vazios já detectados. A tabela *consumption* é responsável por armazenar o histórico do índice de esvaziamento já calculado para cada produto que está sendo monitorado. Seus dados servem de base para se traçar as séries temporais dos índices de esvaziamento dos produtos ao longo do tempo e para a realização de previsões futuras pelo algoritmo de previsão.

Já a tabela *consumption_forecasting* é utilizada para armazenar a previsão do índice de esvaziamento e do estoque estimado futuro dos produtos após a execução do módulo de previsão presente no *dashboard* do GD. Por fim, a tabela *settings* armazena as preferências do usuário dentro do *dashboard*, como quais painéis deseja visualizar e quais as configurações de

¹⁵ Em um relacionamento 1 para N, um registro da tabela A pode se relacionar com muitos da tabela B, mas cada registro da tabela B se relaciona com apenas um da tabela A.

¹⁶ Em um relacionamento 1 para 1, um registro da tabela A se relaciona com no máximo um registro da tabela B, e vice-versa.

previsão. Maiores detalhes das colunas presentes dentro de cada uma das tabelas do BD podem ser encontrados no Apêndice C deste trabalho.

No processo de reconhecimento de elementos, caso uma pessoa não seja detectada, o GD faz uso inicialmente do Modelo Generalista para realizar a detecção dos *gaps* nas prateleiras. Caso haja a detecção de um espaço, inicialmente é necessário associá-lo a uma prateleira. Isso é feito dividindo a posição y_2 da caixa delimitadora pertencente ao espaço vazio pelos valores das alturas de cada prateleira previamente estimados. Se o resultado de uma das divisões estiver entre 0,9 e 1,1, significa que o respectivo espaço vazio pertence à respectiva prateleira.

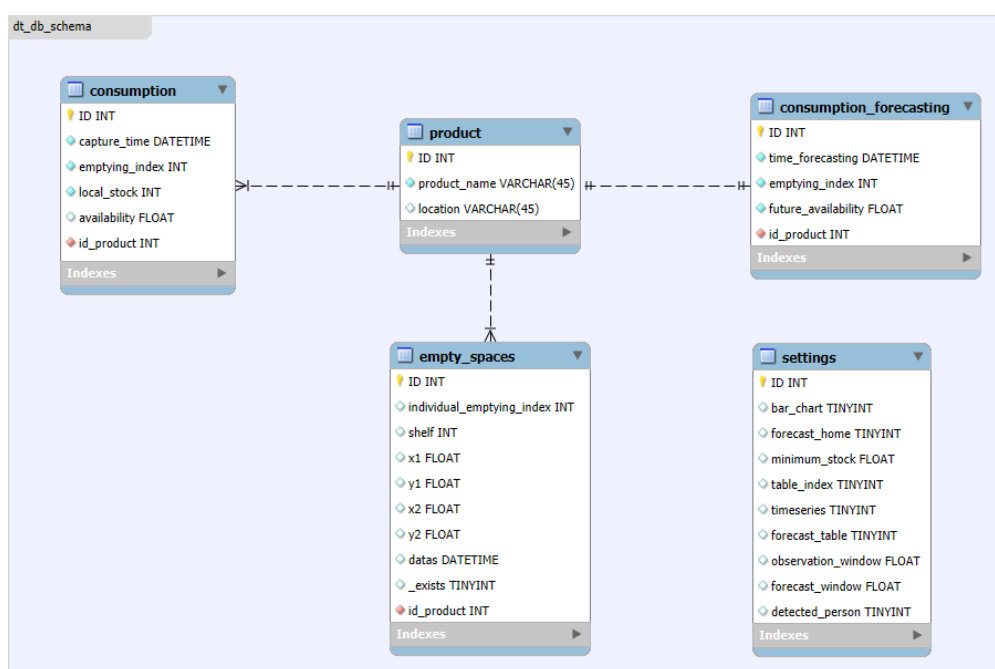


Figura 27 - Organização do BD Relacional do GD proposto
Fonte: elaborado pelo próprio autor

Conhecida qual prateleira pertence o espaço vazio detectado, a largura média dos produtos que se encontram nela, a profundidade (em unidades) de quantos produtos cabem nela e a classe desses produtos são extraídos do arquivo de calibração. A equação 6 apresenta o cálculo do índice de esvaziamento de um espaço vazio pertencente a um produto específico fazendo uso destes dados.

A Figura 28 apresenta um exemplo visual produzido pelo *backend* do GD do cálculo de cada um dos *gaps* detectados nas prateleiras. Porém, é importante ressaltar que o índice de esvaziamento total de um produto (ou simplesmente, o índice de esvaziamento de um produto) é calculado pela soma de todos os seus *gaps* detectados. A equação 7 ilustra este cálculo.

$$\text{indice_espaço} = \left(\frac{x_2 - x_1}{\text{largura_media_produto}} \right) * \text{profundidade_produto} \quad (6)$$

Onde:

- x_2 representa a coordenada horizontal mais à direita do respectivo espaço vazio;
- x_1 representa a coordenada horizontal mais à esquerda do respectivo espaço vazio;
- $\text{largura_media_produto}$ representa a largura média do produto que se encontra na mesma prateleira do respectivo espaço vazio;
- $\text{profundidade_produto}$ representa a profundidade (em unidades) do produto que se encontra na mesma prateleira do respectivo espaço vazio;

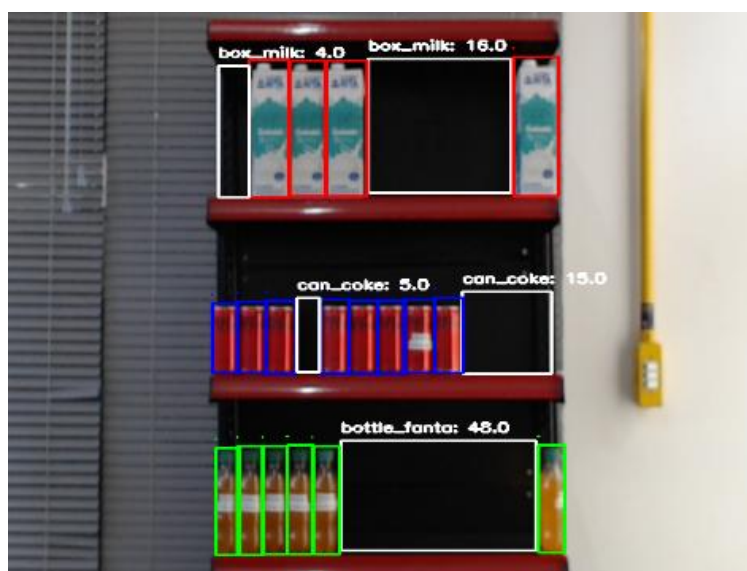


Figura 28 - Cálculo do índice de esvaziamento individual de cada *gap* detectado
Fonte: elaborado pelo próprio autor

$$\text{indice_de_esvaziamento}_p = \sum_{i=0}^{\text{numero_espaços}_p} \left(\frac{x_2^{(i,p)} - x_1^{(i,p)}}{\text{largura_media_produto}_p} \right) * \text{profundidade_produto}_p \quad (7)$$

Onde:

- $\text{indice_de_esvaziamento}_p$ representa a soma dos índices dos espaços vazios pertencentes a um produto p ;
- numero_espaços_p representa a quantidade de espaços vazios detectados para um produto p ;
- $x_2^{(i,p)}$ representa a coordenada horizontal mais à direita do espaço vazio i pertencente ao produto p ;

- $x_1^{(i,p)}$ representa a coordenada horizontal mais à esquerda do espaço vazio i pertencente ao produto p ;
- $largura_media_produto_p$ representa a largura média do produto p ;
- $profundidade_produto_p$ representa a profundidade (em unidades) do produto;

4.3.2.4 Implementação e acesso ao *Dashboard*

No *dashboard* proposto, a dinâmica de atualização é controlada por *callbacks*, ou seja, funções implementadas em Python que reagem a alterações em componentes previamente definidos e atualizam, de forma automática, os elementos da interface. Na tela *HOME*, composta por painéis descritivos e preditivos que requerem atualização contínua, essas *callbacks* podem ser acionadas tanto por interações do usuário quanto por gatilhos temporizados que executam consultas periódicas ao *backend* para obtenção de novos dados.

A cada execução, as informações são processadas e os resultados são retornados a gráficos, tabelas e indicadores, sem necessidade de recarregamento da página. Esse mecanismo assegura que os painéis permaneçam sincronizados em tempo real, refletindo de forma contínua e responsiva as variações dos dados monitorados. O acesso ao *dashboard* é realizado por meio de um navegador *web*, utilizando o endereço e a porta onde o servidor da aplicação está em execução.

Em ambiente de desenvolvimento, é comum que a aplicação seja executada localmente (por exemplo, em <http://127.0.0.1:8050/home>). Porém, pelo fato do *dashboard* ser uma aplicação *web*, ele oferece vantagens relevantes, como acessibilidade multiplataforma sem necessidade de instalação local, facilidade de integração com *Application Programming Interfaces* (APIs) e serviços em nuvem, além de permitir escalabilidade para múltiplos acessos simultâneos.

As figuras a seguir ilustram cada um dos painéis propostos no *dashboard* com exemplos de valores. A Figura 29 ilustra o painel descritivo *Ranking – Índice de Esvaziamento*. Como pode ser visto, os valores apresentados na tabela podem ser filtrados pelo valor do índice de esvaziamento (quanto maior, maior a prioridade) ou pela disponibilidade estimada (quanto menor, maior a prioridade). Além disso, apresenta em uma legenda o significado de cada uma das cores propostas para representar situações distintas. É importante ressaltar que o padrão das

cores pode ser alterado no código fonte do *dashboard* proposto, caso o *stakeholder* deseje utilizar outras cores ao invés das inicialmente propostas.

No exemplo ilustrado neste painel, é possível verificar que o produto *bottle_fanta* se encontra com um índice baixo (cor amarela) e disponibilidade estável. Já o produto *can_coke* aparece com o índice mais alto (cor vermelha), mas ainda dentro da disponibilidade mínima. O produto *box_milk* se encontra no estado mais crítico, pois sua disponibilidade estimada se encontra abaixo dos 30% (cor roxa), disponibilidade mínima configurada na tela de configurações do *dashboard*. Os valores produzidos por este painel podem ser de grande valia para se traçar estratégias de reposição de suprimentos pelos *stakeholders*.

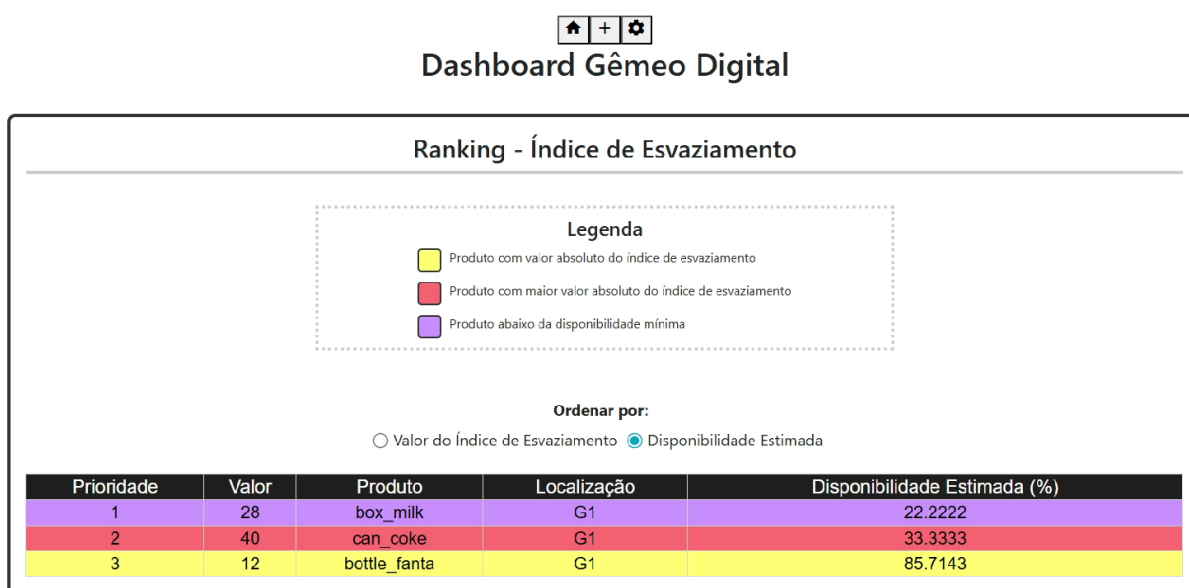


Figura 29 - Exemplo do painel Ranking - Índice de Esvaziamento
 Fonte: elaborado pelo próprio autor

A Figura 30 ilustra o painel descritivo Visão Geral – Índice de Esvaziamento. Como pode ser visto, ele apresenta de forma objetiva e gráfica os valores dos índices calculados para cada produto. Em implementações futuras, novos tipos de gráficos e filtros poderão ser adicionados.

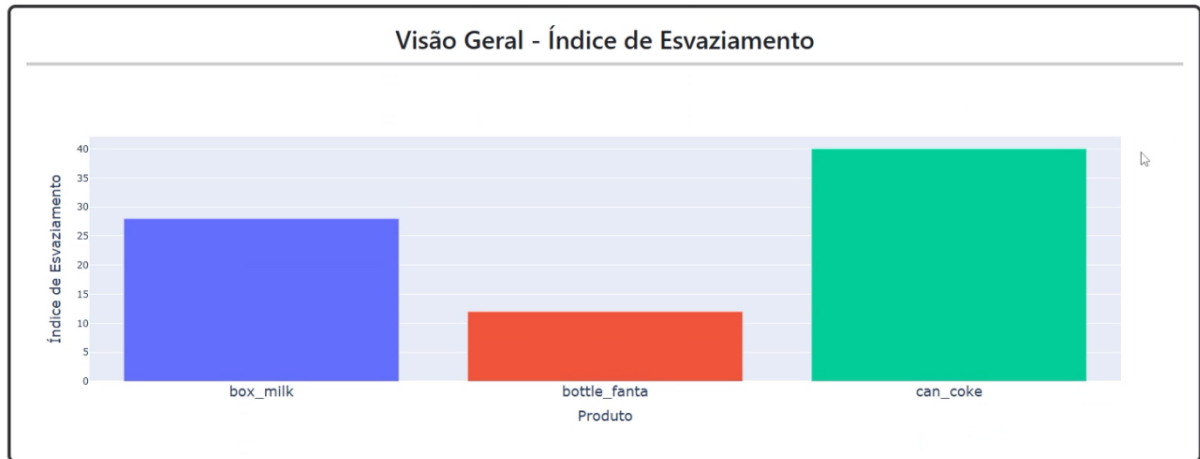


Figura 30 - Exemplo de painel Visão Geral - Índice de Esvaziamento
 Fonte: elaborado pelo próprio autor

A Figura 31 ilustra o painel de Previsão Padrão quando a previsão ainda não se encontra em execução. Para realização da previsão, é importante que os dados coletados sejam do tamanho da Janela de Observação (dados passados) configurada pelos *stakeholders* na tela de configurações. Além disso, a previsão trabalha com dados no formato de minutos. Por isso, antes de serem enviados ao algoritmo de previsão, no *back-end* eles passam por um *resample*¹⁷ de dados.



Figura 31 - Painel de Previsão Padrão aguardando dados da Janela de Observação
 Fonte: elaborado pelo próprio autor

Já a Figura 32 ilustra o mesmo painel após realizar uma previsão 10min à frente. Como pode ser visto, este painel apresenta o mesmo padrão de cores do painel *Ranking*, porém,

¹⁷ Um *resample* de dados é o processo de alterar a frequência temporal de uma série de dados, agregando, interpolando ou redistribuindo valores para um novo intervalo de tempo (por exemplo, de segundos para minutos ou de diário para mensal).

descrevendo possíveis valores futuros produzidos pelo algoritmo de previsão. No caso do exemplo, dois produtos se encontram em situação crítica, ilustrando a necessidade de reposição imediata.



Figura 32 - Painel de Previsão Padrão após executar uma previsão em um determinado momento
 Fonte: elaborado pelo próprio autor

É importante ressaltar que, para fins de ganhos de performance, é sugerido que painéis que realizam tarefas específicas e não aproveitam dados de outros painéis sejam executados por meio do uso de *threads*. *Threads* são unidades de execução dentro de um processo que podem rodar em paralelo, permitindo dividir tarefas independentes e assim aproveitar melhor múltiplos núcleos do processador para ganhar desempenho. O painel de Previsão Padrão, por exemplo, faz uso desta abordagem.

A Figura 33 ilustra o painel de Acompanhamento Individual – Índice de Esvaziamento. Como mencionado, ele permite acompanhar de forma individual o padrão de consumo de cada produto em monitoramento ao longo tempo. Além disso, cada vez que é atualizado com o último valor, apresenta ao lado do gráfico de consumo um *snapshot* (imagem atualizada da respectiva prateleira) a fim de também fornecer um *feedback* visual de cada prateleira em monitoramento aos *stakeholders*. Isso pode ser importante para detecção de *layouts* desorganizados ou caso haja desconfiança de valores dos índices de esvaziamento.

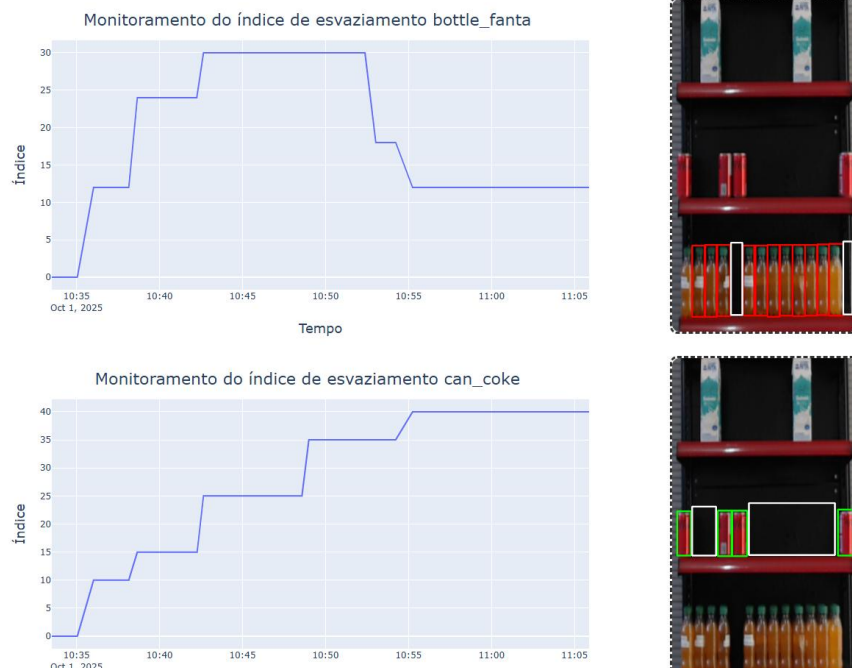


Figura 33 - Exemplo de informações presentes no painel Acompanhamento Individual - Índice de Esvaziamento
 Fonte: elaborado pelo próprio autor

Por fim, a Figura 34 ilustra a tela de configurações do *dashboard*. Nele é possível selecionar quais painéis se deseja visualizar na tela *HOME*, realizar configurações do painel de Previsão Padrão e da Disponibilidade mínima que se deseja configurar para identificar necessidade urgente de reabastecimento (representado pelas linhas na cor roxa).

Configurações

Visualização

- Mostrar gráfico de barras
- Mostrar: tabela de ranking
- Mostrar: histórico e snapshot
- Mostrar: tabela de previsão

Previsão Padrão

Janela de observação (mínimo 30)

Janela de previsão

Disponibilidade mínima

%

Salvar

Figura 34 - Tela de configurações do *dashboard* (símbolo de engrenagem)
 Fonte: elaborado pelo próprio autor

Ao longo da construção do modelo computacional por meio dos materiais e métodos descritos, o GD foi sendo verificado de forma progressiva pelos autores a fim de avaliar se sua

construção seguia o mais fielmente possível a documentação definida na etapa de concepção. *Backups* do modelo computacional de cada uma das etapas verificadas foram sendo armazenados conforme a implementação avançava. Uma vez que todas as etapas foram verificadas, partiu-se para a etapa de validação do modelo.

Na etapa de validação, testes empíricos foram realizados pelo autor a fim de avaliar o quão próximo o GD proposto se aproximava do sistema real, a fim de avaliar se o modelo serve ao propósito pelo qual foi criado. Ao identificar a operação do modelo de forma plausível, produzindo saídas muito próximas das esperadas, partiu-se para a etapa de Análise, a fim de que experimentos e análises estatísticas já pudessem ser definidas e realizadas sobre ele, a fim de se tirar conclusões e sugerir recomendações para tomadas de decisões no mundo real.

4.4 Análise

Nos subtópicos a seguir, primeiramente serão apresentados e discutidos maiores detalhes das métricas gerais e por classe produzidas pelos modelos de VC, recursos estes fundamentais para detecção de elementos de interesse e geração de dados. Em seguida, serão apresentados e discutidos resultados de três testes práticos de operação do GD proposto.

4.4.1 Métricas dos Modelos de Visão Computacional

Inicialmente este subcapítulo apresenta uma análise a respeito do desempenho dos modelos YOLO treinados nas tarefas de detecção dos formatos dos produtos e espaços vazios (Modelo Generalista) e produtos específicos (Modelo Especialista). Como já mencionado, na etapa de treinamento ambos os modelos foram configurados para treinarem com até 3000 épocas. Porém, o Modelo Generalista atingiu seu melhor resultado com 268 épocas e o Especialista com 524 épocas, respectivamente.

A Figura 35 e a Figura 36 apresentam gráficos com métricas importantes geradas após o treinamento de ambos os modelos. De forma geral, eles apresentam os valores das seguintes métricas ao longo das épocas: *box_loss*, *cls_loss*, *dfl_loss*, *precision*, *recall*, mAP50 e mAP50-95. As chamadas curvas de perda (métricas terminadas com o sufixo *_loss*) são obtidas com base no conjunto de treinamento. Elas são extraídas, apresentadas e discutidas principalmente para se avaliar a convergência dos modelos.

Já as demais métricas de desempenho são calculadas sobre o conjunto de validação. Mais particularmente, as métricas *precision*, *recall*, *mAP50* e *mAP50-95* são fundamentais de serem extraídas, apresentadas e discutidas quando se busca avaliar modelos de VC baseados

em CNN (Ultralytics, 2023b). A seguir são apresentados mais em detalhes o significado de cada uma destas métricas:

- *box_loss*: mede o erro entre a caixa delimitadora prevista pelo modelo e a caixa real do objeto (anotada no *dataset*). Quanto menor, melhor;
- *cls_loss*: mede o erro de classificação, ou seja, o modelo está atribuindo a classe certa ao elemento detectado. Quanto menor, melhor;
- *dfl_loss*: ajuda a melhorar a localização das caixas delimitadoras ao refinar cada vez mais as suas bordas. Quanto menor, melhor;
- *precision*: proporção de verdadeiros positivos dentre todas as predições positivas, ou seja, das detecções realizadas pelo modelo, quantas estavam certas. Quanto maior, melhor;
- *recall*: proporção de verdadeiros positivos dentre todos os alvos positivos, ou seja, dos objetos que realmente existiam, quantos o modelo encontrou. Quanto maior, melhor;
- *mAP50*: a *Average Precision (AP)* é um único número que resume o desempenho combinado entre *precision* e *recall*. Já o *Mean Average Precision (mAP)* é a média AP entre todas as classes do modelo, fornecendo uma visão geral do desempenho do modelo quando ele possui múltiplas classes. É uma métrica mais tolerante, bastando que uma caixa delimitadora sobreponha metade (50%) da área de uma classe para contar como acerto. Quanto maior, melhor;
- *mAP50-95*: Parecido com o *mAP50*, mas é uma métrica mais rígida, pois seu valor é calculado a partir da média de 10 limiares de detecção diferentes, que vão de 0,5 até 0,95. Por conta da maior rigidez, seus valores são menores que o do *mAP50*. Quanto maior, melhor;

No treinamento de ambos os modelos se observa uma queda nos valores de *box_loss*, *cls_loss* e *dfl_loss* conforme o número de épocas aumenta. Além disso, é possível notar uma maior estabilização do Modelo Generalista após 200 épocas e do Modelo Especialista após 500 épocas. O inverso acontece com as métricas *precision*, *recall*, *mAP50* e *mAP50-95*, ou seja, como desejável, estas métricas aumentaram conforme o número de épocas avançaram, além de também apresentarem maior estabilidade após 200 épocas e 500 épocas para os Modelos Generalista e Especialista, respectivamente.

Além disso, para os dados de teste, pode-se concluir que os modelos alcançaram resultados satisfatórios (próximos de 1) para *precision*, *recall*, pontuações *mAP* diferentes e *F1 score*, conforme apresentado na Tabela 4. O *F1-Score* é outra métrica importante formado pela média harmônica entre *precision* e *recall*. Ela oferece uma avaliação equilibrada do desempenho de um modelo, considerando falsos positivos e falsos negativos. Para esta métrica, também quanto maior o valor, melhor.

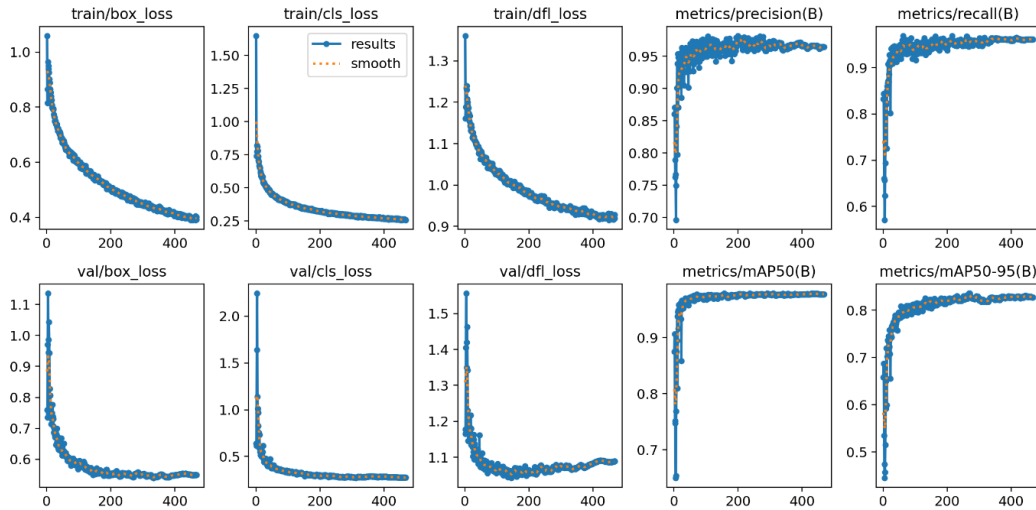


Figura 35 - Gráficos apresentando métricas de treinamento/validação do Modelo Generalista ao longo das épocas
Fonte: elaborado pelo próprio autor

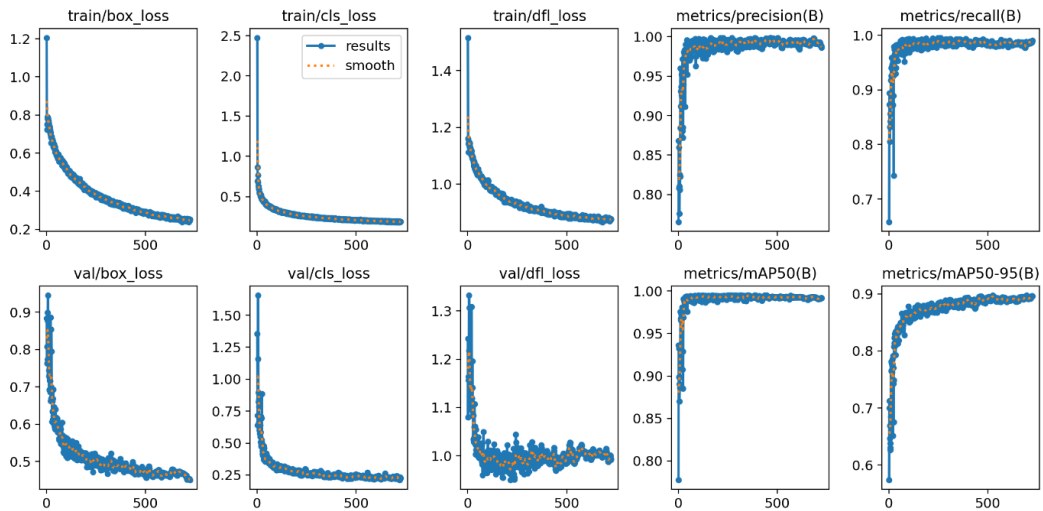


Figura 36 - Gráficos apresentando métricas do treinamento/validação do Modelo Especialista ao longo das épocas
Fonte: elaborado pelo próprio autor

Tabela 4 - Métricas de desempenho dos Modelos Generalista e Especialista sobre os seus respectivos conjuntos de teste

| Métrica | Modelo Generalista | Modelo Especialista |
|------------------|---------------------------|----------------------------|
| <i>precision</i> | 0,974 | 0,995 |
| <i>recall</i> | 0,950 | 0,996 |
| <i>mAP50</i> | 0,986 | 0,995 |
| <i>mAP50-95</i> | 0,854 | 0,878 |
| <i>F1-Score</i> | 0,962 | 0,9955 |

Fonte: elaborado pelo próprio autor

A Tabela 5 apresenta as métricas por classe, o que representa o desempenho dos modelos nas detecções individuais para os dados de teste. Como pode ser visto, os resultados individuais das classes também apresentam resultados satisfatórios de detecção (próximos de 1). Dentre todas elas, a classe para detecção de latas de Coca-Cola (*can_coke*) foi a que apresentou menores resultados na métrica mais rígida, no caso, a *mAP50-95*. Apesar de ainda ser um resultado satisfatório, isso ilustra que o GD em algum momento pode ter dificuldades para classificar uma lata de Coca-Cola de forma precisa.

Tabela 5 - Métricas de desempenho das classes individuais de ambos os modelos para os dados de teste

| | Modelo Generalista | | Modelo Especialista | | |
|------------------|---------------------------|----------------|----------------------------|-----------------|-----------------|
| | Classe | | Classe | | |
| Métrica | empty_space | product | bottle_fanta | box_milk | can_coke |
| <i>precision</i> | 0,965 | 0,982 | 0,991 | 1 | 0,994 |
| <i>recall</i> | 0,919 | 0,981 | 1 | 0,998 | 0,991 |
| <i>mAP50</i> | 0,980 | 0,992 | 0,995 | 0,995 | 0,995 |
| <i>mAP50-95</i> | 0,809 | 0,899 | 0,899 | 0,934 | 0,802 |
| <i>F1-Score</i> | 0,9412 | 0,9818 | 0,995 | 0,998 | 0,992 |

Fonte: elaborado pelo próprio autor

A Figura 37 e a Figura 38 apresentam as matrizes de confusão das classes de ambos os modelos na fase de validação. Cada linha da matriz representa a instância de uma classe predita pelo modelo, enquanto cada coluna representa a instância da classe atual. Ela é chamada de

matriz de confusão pois ajuda a evidenciar possíveis “confusões” feitas pelos modelos ao se classificar elementos. A diagonal principal de ambas as matrizes representa os verdadeiros positivos (do inglês, *True Positive*), ou seja, representam as classes que esperavam ser detectadas e foram detectadas corretamente.

Os valores que aparecem abaixo e acima da diagonal principal indicam erros de classificação entre classes, ou seja, objetos reais de uma classe previstos como outra. Para o modelo Generalista, na coluna *Background*, observa-se valores de falsos positivos (do inglês, *False Positive*). Isso ilustra a detecção incorreta de produtos e espaços vazios em locais onde não deveriam aparecer, ou seja, o aparecimento de ruídos ou detecções fantasmas.

Visando diminuir este tipo de detecção equivocada no GD, o autor aumentou o limiar de confiança das detecções para 60%, definindo assim um valor de corte para aceitar uma detecção como válida. Esta configuração é feita diretamente no código do modelo ao ser chamado no *Back-end*. O limiar de confiança (*conf*) é um parâmetro de operação que controla o ponto de corte entre detecções aceitas e rejeitadas (como detecções fantasmas).

Sua definição é tipicamente feita de forma empírica para alcançar o melhor compromisso entre erro e acerto conforme os requisitos da aplicação (Wenkel *et al*, 2021). É importante ressaltar que o Background, apesar de aparecer na matriz de confusão, não é uma classe treinada. O YOLO cria esta categoria apenas para contabilizar erros de detecção em áreas sem objetos anotados.

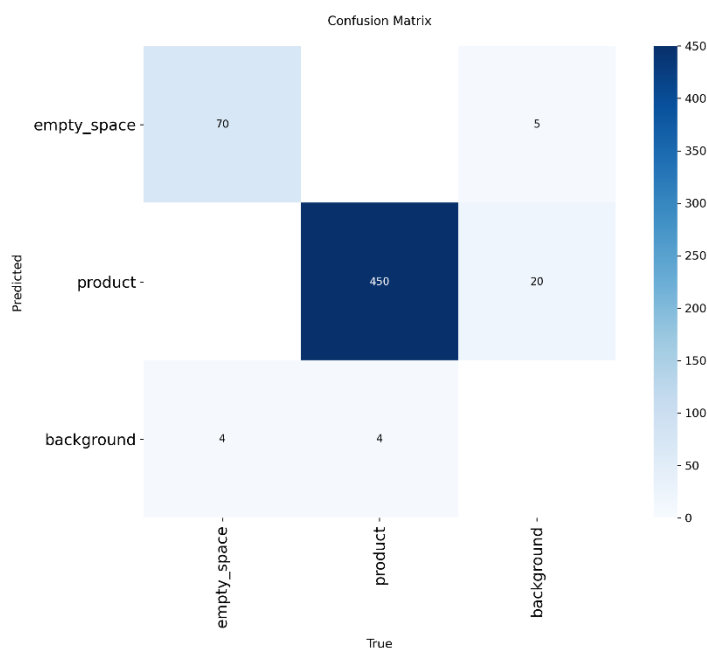


Figura 37 - Matriz de confusão das classes do Modelo Generalista

Fonte: elaborado pelo próprio autor

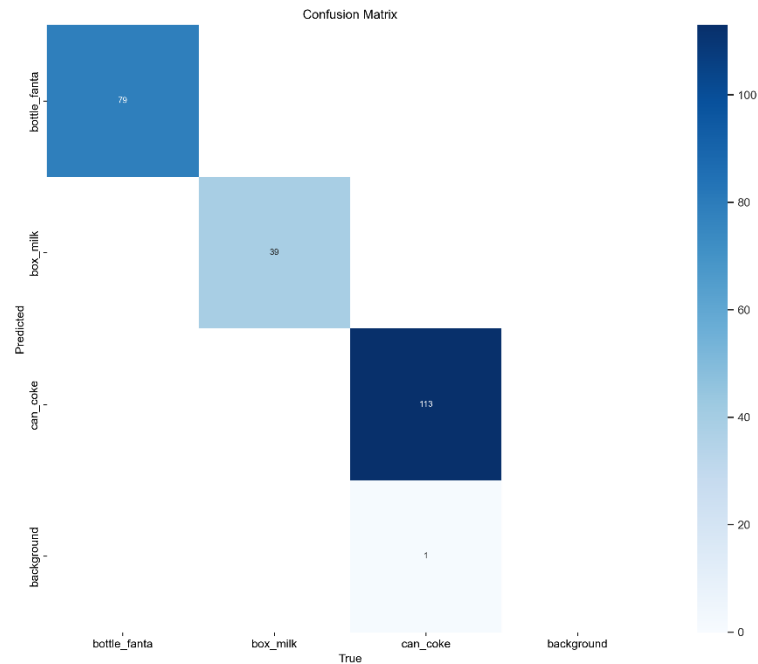


Figura 38 - Matriz de confusão das classes do Modelo Especialista
Fonte: elaborado pelo próprio autor

4.4.2 Testes Práticos e Resultados

A fim de avaliar na prática a operação do GD proposto, três testes foram preparados, executados e analisados. No primeiro, foram definidos cenários de operação que trabalham com variações de três variáveis: *zoom*, iluminação e *layout*. No segundo, testes específicos de oclusão de pessoas (tipo de oclusão que o GD foi inicialmente preparado para detectar) também foram realizados sobre a melhor variação de *zoom* e iluminação detectadas no primeiro teste. Por fim, um teste de consumo de um produto ao longo do tempo também sobre a melhor variação de *zoom* e iluminação foi realizado a fim de se avaliar os resultados dos índices capturados e das previsões realizadas. A escolha destas condições para os testes reflete fatores relevantes para a operação prática da proposta, respeitando restrições inerentes a ela.

4.4.2.1 Teste Prático 1

O primeiro conjunto de testes levou em consideração a operação do GD sob diferentes condições de *zoom*, iluminação e *layout*, buscando verificar sua operação quando estas condições são alteradas. A variação do *zoom* permite avaliar a robustez do modelo em diferentes distâncias do foco em relação às prateleiras monitoradas. O objetivo é avaliar o quanto o GD

mantém um desempenho consistente nos diferentes níveis de aproximação dos produtos monitorados.

A iluminação também é considerada um fator crítico em soluções que envolvam VC, pois variações de intensidade luminosa, sombras ou reflexos podem afetar a acurácia da detecção. Ambas as variáveis são mencionadas por Liu *et al.* (2019) e Santra e Mukherjee (2019) como variáveis importantes que podem afetar a detecção de elementos em condições reais. Já as variações no *layout* permitem verificar como o GD se comporta quando os *gaps* encontrados são de tamanhos variados. A Figura 39 ilustra a organização geral deste teste.

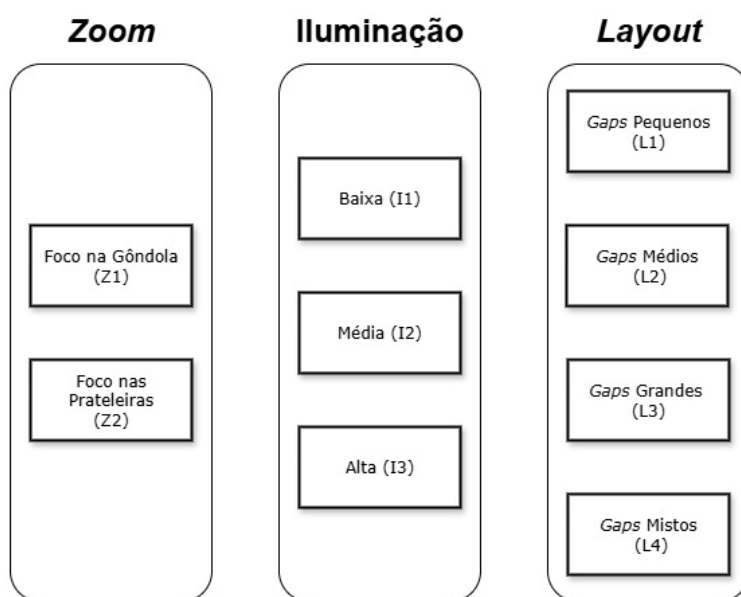


Figura 39 - Organização do Teste 1
Fonte: elaborado pelo próprio autor

Em relação ao *zoom*, 2 tipos diferentes foram definidos: foco na gôndola (*zoom* mais distante, denominado, Z1) e foco nas prateleiras onde se encontram os produtos em monitoramento (Z2). Já para definição dos níveis de iluminação, o trabalho de Lin e Yoon (2015) foi utilizado como base para se propor 3 níveis: baixo (I1), normal (I2) e alto (I3). Para baixo, foi definida uma iluminação $< 100 \text{ lux}$ ¹⁸, para normal $\approx 650 \text{ lux}$ e para alto uma iluminação $> 1000 \text{ lux}$. A iluminação do ambiente onde a gôndola se encontra foi alterada e a medição dos níveis foi realizada à frente das prateleiras por meio do aplicativo Light Meter LM-3000¹⁹.

¹⁸ Lux é a unidade que mede a intensidade da iluminação percebida em uma superfície, correspondendo a um lúmen por metro quadrado.

¹⁹ <https://lightray.io/lightmeter/>

Já para a definição dos *layouts*, *gaps* pequenos (denominado, L1), médios (L2), grandes (L3) e mistos (L4) foram organizados aleatoriamente nas prateleiras dos três produtos pelos quais o GD foi preparado para monitorar. *Gaps* pequenos foram formados por espaços da largura de um único produto, mas que se permitia ver em profundidade o fundo da prateleira. *Gaps* médios são da largura de 2 ou 3 produtos, *gaps* grandes são espaços formados por 4 produtos ou mais e *gaps* mistos envolvem os 3 tipos de *gaps* anteriores.

A Figura 40 e a Figura 41 ilustram exemplos de cada um dos *layouts* sobre a combinação Z1,I2 e Z2,I2, respectivamente. É importante ressaltar que, neste momento, o cálculo do índice de esvaziamento ainda não é capaz de levar em consideração *layouts* completamente desorganizados, mas *layouts* que apresentem *gaps* capazes de acomodarem, em largura, cada uma das quantidades de produtos propostas. Maiores detalhes de como mitigar este problema serão apresentados no tópico 5.2.

Para cada teste realizado com variações de *zoom* e iluminação, algumas métricas foram coletadas a fim de compreender a robustez do modelo em cada cenário. Como a proposta se trata de um GD, é fundamental que ao menos dois atributos sejam medidos: a Fidelidade de Representação e o Tempo de Resposta (Su *et al.*, 2023; Wu *et al.*, 2023).

A Fidelidade de Representação busca avaliar o quanto o GD está descrevendo corretamente o mundo físico conforme esperado. Neste trabalho, ela será mensurada por meio da Taxa de Erro (TE) dos índices de esvaziamento, se comparando os valores reais (averiguado pelo autor em cada teste) com os apresentados no *dashboard*.



Figura 40 - Combinação Z1,I2 com exemplos de *layouts* pequenos (primeiro *layout*, da esquerda para a direita), médios (segundo *layout*), grandes (terceiro *layout*) e mistos (quarto *layout*), respectivamente

Fonte: elaborado pelo próprio autor



Figura 41 - Combinação Z2,I2 com exemplos de layouts pequenos (canto superior esquerdo), médios (canto superior direito), grandes (canto inferior esquerdo) e mistos (canto inferior direito), respectivamente
Fonte: elaborado pelo próprio autor

Já o Tempo de Resposta (TR) busca avaliar se o GD está descrevendo o comportamento do mundo físico em tempo útil para auxiliar em uma possível tomada de decisão. Neste trabalho, ela será mensurada por meio do cálculo do tempo fim a fim de uma ação, ou seja, ao se consumir produtos nas prateleiras em monitoramento e for detectados espaços vazios, mediu-se por meio de cronoanálise quanto tempo o GD levou para apresentar o resultado no *dashboard*.

Além disso, métricas de VC como *Precision*, *Recall*, *mAP50-95* e *F1-Score* também foram extraídas do modelo generalista, mais precisamente da classe *empty_space*, a fim de verificar o quanto o GD perde acurácia de detecção desta classe (fundamental para o cálculo dos índices) quando submetido às variações propostas. A fim de se levantar as métricas apresentadas para cada variação de *zoom*, iluminação e *layout*, cenas reais das prateleiras organizadas com os *gaps* pequenos, médios, grandes e mistos foram gravadas em vídeo.

Foram coletadas 40 cenas (10 cenas aleatórias obedecendo cada tipo de *layout*) em cada variação de Z e I. Além disso, ao mesmo tempo que as cenas eram coletadas, a tela do *dashboard* do GD também foi gravada a fim de que, após as gravações, as auditorias pudessem ser realizadas. A Tabela 6 apresenta os resultados levantados após as auditorias.

Tabela 6 - Resultados do Teste 1

| Zoom | Iluminação | Layout (Gap) | TE Média (% <i>, por Gap</i>) | TE Média (%) | DP TE (%) | TE IC95% | TR Médio (s) | DP TR (s) | TR IC95% |
|------|------------|--------------|--------------------------------|--------------|-----------|------------------|--------------|-----------|----------------|
| Z1 | I1 | L1 | 27% | 19,86% | 10,78% | [16,37%; 23,35%] | 13,92s | 0,85s | [13,64; 14,19] |
| | | L2 | 19% | | | | | | |
| | | L3 | 14% | | | | | | |
| | | L4 | 19% | | | | | | |
| | I2 | L1 | 23% | 16,47% | 13,26% | [12,18%; 20,76%] | 13,96s | 1,28s | [13,55; 14,38] |
| | | L2 | 13% | | | | | | |
| | | L3 | 13% | | | | | | |
| | | L4 | 17% | | | | | | |
| | I3 | L1 | 41% | 28,11% | 12,31% | [24,12%; 32,09%] | 14,15s | 1,55s | [13,65; 14,66] |
| | | L2 | 27% | | | | | | |
| | | L3 | 20% | | | | | | |
| | | L4 | 25% | | | | | | |
| Z2 | I1 | L1 | 0% | 4,38% | 4,59% | [2,89%; 5,87%] | 13,89 | 0,82s | [13,62; 14,16] |
| | | L2 | 3% | | | | | | |
| | | L3 | 9% | | | | | | |
| | | L4 | 5% | | | | | | |
| | I2 | L1 | 0% | 5,24% | 5,29% | [3,52%; 6,95%] | 13,92s | 1,03s | [13,59; 14,26] |
| | | L2 | 3% | | | | | | |
| | | L3 | 7% | | | | | | |
| | | L4 | 11% | | | | | | |
| | I3 | L1 | 0% | 3,99% | 5,22% | [2,30%; 5,68%] | 13,69s | 1,23s | [13,29; 14,09] |
| | | L2 | 5% | | | | | | |
| | | L3 | 9% | | | | | | |
| | | L4 | 2% | | | | | | |

Fonte: elaborado pelo próprio autor

Como pode ser observado, os testes com *zoom* focado na gôndola (Z1) produziram TEs bem maiores se comparado aos testes de detecção de *gaps* realizados com o *zoom* focado nos produtos em monitoramento (Z2). Nos testes com Z1, *gaps* pequenos foram os que tiveram maior dificuldade de serem detectados, podendo chegar a produzir uma TE de até 41% com iluminação alta.

Já nos testes com Z2, a situação contrária foi observada: os *gaps* pequenos foram praticamente todos detectados nos testes realizados, independente da qualidade de iluminação. A Figura 42 ilustra de forma objetiva a TE Média de detecção para todas as combinações de *zoom* e iluminação com um Intervalo de Confiança (IC) de 95%. Como pode ser visto, detecções realizadas com Z2 podem produzir TEs até 7x menores se comparadas às detecções com Z1.

A Tabela 7 apresenta os resultados das métricas de VC para cada uma das combinações de *zoom* e iluminação. Como pode ser observado, observa-se métricas mAP50-95 (principal

métrica e a mais exigente para reportar a acurácia de uma detecção com YOLO) de Z1 menores que de Z2. Uma análise realizada por meio do teste de correlação de Pearson mostrou uma correlação (r) negativa moderada a forte entre a TE Média (Tabela 6) e mAP50-95 (Tabela 7) igual a $-0,74$. Isso é um indicador de quanto maior for TE, menor tende a ser o mAP50-95 para cada uma das combinações de *zoom* e iluminação. Porém, esta afirmação ainda não é estatisticamente significativa ao nível de 5%, pois $p\text{-value}=0,09$, indicando apenas uma tendência deste indicador, mas que ainda não pode ser confirmada.

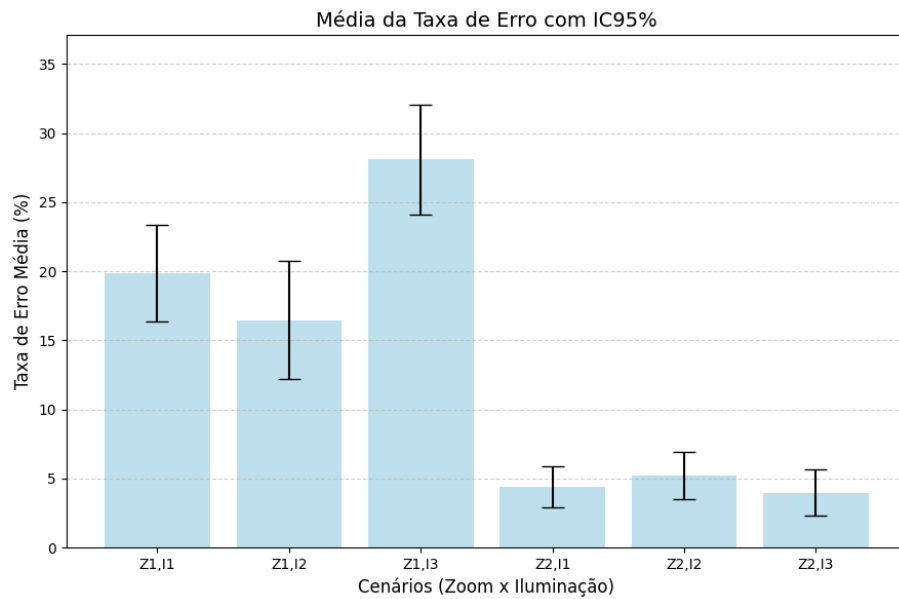


Figura 42 – TE Média de detecção para as combinações de *zoom* e iluminação
 Fonte: elaborado pelo próprio autor

Tabela 7 - Métricas da classe *empty_space* para cada combinação de *zoom* e iluminação

| Zoom | Iluminação | Precision | Recall | mAP50-95 | F1-Score |
|------|------------|-----------|--------|----------|----------|
| Z1 | I1 | 0,970 | 0,992 | 0,912 | 0,980 |
| | I2 | 0,889 | 0,948 | 0,694 | 0,917 |
| | I3 | 0,888 | 0,913 | 0,738 | 0,900 |

| | | | | | |
|----|----|-------|-------|-------|-------|
| | | | | | |
| | I1 | 0,997 | 0,996 | 0,949 | 0,996 |
| Z2 | I2 | 0,992 | 1 | 0,962 | 0,995 |
| | I3 | 0,985 | 1 | 0,955 | 0,992 |

Fonte: elaborado pelo próprio autor

Já em relação ao TR do GD ao se realizar alterações nas prateleiras, não foi possível observar variações consideráveis, independente da combinação de *zoom*, iluminação e *gaps* realizadas. Ele foi medido levando em consideração o tempo de atualização do painel com maior número de elementos gráficos do *dashboard*, no caso, o painel de Acompanhamento Individual.

Em todos os testes, após a realização das alterações, o TR foi por volta de 14s (para monitoramento de 3 produtos). Este tempo inclui ações como: *buffer* de captura da câmera, pré-processamento das imagens capturadas, latência das inferências realizadas pela fusão hierárquica dos modelos de VC, *cooldown* de oclusão por pessoa, cálculo dos índices de esvaziamento, operações de busca e inserção de dados no BD e renderização dos elementos no *dashboard*.

4.4.2.2 Teste Prático 2

Liu *et al.* (2019) e Santra e Mukherjee (2019) também destacam a oclusão como uma variável importante de ser levada em consideração em testes que envolvam VC. Porém, testes gerais de oclusão (ou seja, oclusões por qualquer tipo de elemento que não seja pessoa) não foram levados em consideração neste trabalho, já que a proposta na configuração atual consegue detectar apenas oclusões por pessoas.

Por isso, foram realizados testes específicos de oclusão humana contemplando quatro variações: oclusão por membro (denominado, O1), por parte do corpo (O2), por corpo inteiro (O3) e por passagem (*motion*, O4). É importante ressaltar que, quando uma oclusão por pessoa é detectada em frente a uma gôndola em monitoramento, as detecções de produtos e *gaps* são interrompidas momentaneamente e no painel de acompanhamento individual do produto é apresentado um ícone indicador da oclusão, conforme ilustrado na Figura 43.



Figura 43 - Sinal de oclusão apresentado pelo GD no painel de Acompanhamento Individual
 Fonte: elaborado pelo próprio autor

A oclusão por membro levou em consideração a aparição de dedos, mãos e braços em frente às prateleiras monitoradas. A oclusão parcial levou em consideração a aparição de uma pessoa pela metade, de frente e de costas. A oclusão total levou em consideração pessoas aparecendo de corpo inteiro, de frente e de costas. Por fim, o *motion* levou em consideração uma pessoa passando em frente à gôndola em 3 tipos de velocidades: lenta, moderada e rápida. A Figura 44 ilustra um mosaico com exemplos de imagens extraídas dos testes de oclusão.

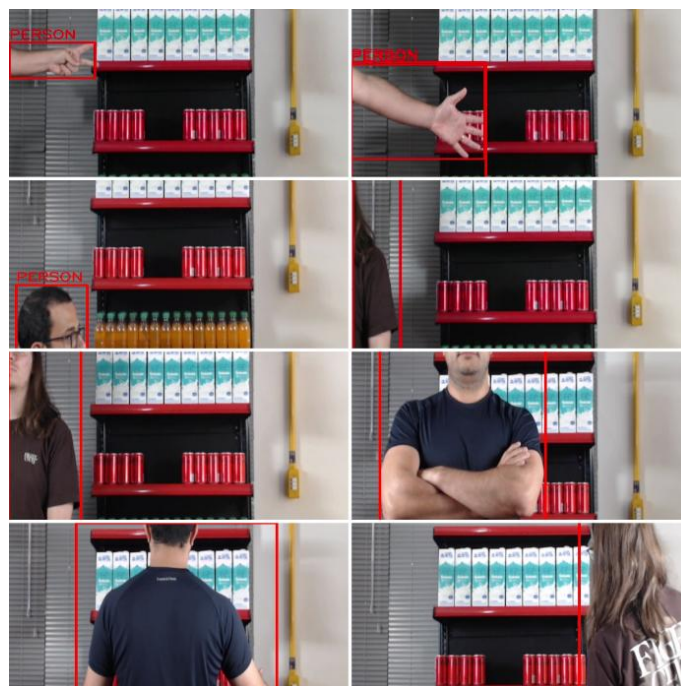


Figura 44 - Exemplos de oclusões causadas no Teste 2
 Fonte: elaborado pelo próprio autor

Os testes de oclusão foram realizados sobre a configuração Z2, I3, apresentada no Teste Prático 1, e que obteve a menor taxa de erro entre as demais variações testadas. Para cada variação de Z2, I3 e Ox (onde x representa as diferentes variações de oclusão mencionadas), executaram-se 10 tentativas de oclusão. A Tabela 8 ilustra os resultados levantados após os testes.

Tabela 8 - Resultados do Teste 2

| Zoom | Iluminação | Oclusão | TD Média (%, por Oclusão) | TD Média (%) | DP TD (%) | TD IC95% | TR Médio (s) | DP TR (s) | TR IC95% |
|------|------------|---------|------------------------------|-----------------|--------------|-----------------|-----------------|--------------|-----------------|
| Z2 | I3 | O1 | 100% | 92,5% | 15% | [0,88; 0,97] | 7,03s | 1s | [6,71; 7,36] |
| | | O2 | 100% | | | | | | |
| | | O3 | 100% | | | | | | |
| | | O4 | 70% | | | | | | |

Fonte: elaborado pelo próprio autor

O alto valor da Taxa de Detecção (TD) é um indicador da grande capacidade do modelo YOLOv8m padrão para detecção de oclusão por pessoas, seja parcialmente ou integralmente, principalmente em detecções estáticas (O1, O2 e O3). O movimento rápido (*motion*) é o único cenário que apresentou perda relevante, o que confirma que a mudança rápida de pose prejudica detectores *frame a frame*.

Porém, mesmo que a pessoa não seja detectada por *motion*, o processo de detecção de *gaps* e produtos não foi afetado, pois o *blur* dura poucos *frames* e não apresenta a geometria

nem a textura dos elementos presentes nas áreas delimitadas da gôndola (definidas no processo de calibração).

4.4.2.3 Teste Prático 3

O presente teste teve como finalidade avaliar em um cenário de consumo otimista (ou seja, com *gaps* de fáceis detecções) a capacidade do GD em responder, de forma progressiva e consistente, às alterações reais no estado de uma única prateleira em monitoramento, bem como verificar as previsões de consumo. Para tal, foi conduzido um experimento controlado com duração total de 60 minutos, no qual se simulou um processo progressivo de consumo de latas de Coca-Cola.

Durante a execução, foram registradas de maneira sistemática, a cada 5min, tanto as ações de retirada planejadas quanto o tempo decorrido até que as mudanças fossem refletidas no *dashboard*. Adicionalmente, a partir do minuto 30 (momento T6), buscou-se analisar a resposta do painel de Previsão Padrão, que está configurado com uma janela de observação deslizante de 30min e janela de previsão de 10min.

Ou seja, a partir do minuto 30, uma nova previsão é calculada minuto a minuto por meio do *auto_arima*, buscando sempre avaliar a possível situação de consumo 10min à frente. O protocolo também contemplou a reposição do item quando a disponibilidade projetada no painel descritivo de *Ranking* caísse abaixo de 20% (momento T9) da capacidade total da prateleira, que é de 60 unidades.

A Tabela 9 detalha o cronograma de eventos de T0 a T12 e a Figura 45 apresenta visualmente as alterações realizadas ao longo do processo. Os valores do Índice de Esvaziamento e Disponibilidade Estimada foram extraídos do painel descritivo de *Ranking*. Já os valores de Previsão do Índice e Previsão de Disponibilidade foram extraídos do painel preditivo de Previsão Padrão. Observou-se que o TR médio para monitoramento de um único produto durante o tempo do experimento, levando em consideração os TRs de quando houve consumo, foi de 12,17s com um DP de 0,42s.

Tabela 9 - Cronograma de eventos do Teste 3 e resultados.

| T | Tempo (min) | Ação (un) | Índice de Esvaziamento Reportado no GD | Disp. Reportada no GD (%) | TR (s) | Previsão do Índice no GD (T + 10) | Previsão de Disp. no GD (T + 10) | Obs. |
|-----|-------------|-----------|--|---------------------------|--------|-----------------------------------|----------------------------------|---|
| T0 | 0 | 0 | 0 | 100 | - | - | - | Início do Teste |
| T1 | 5 | -5 | 5 | 91,67 | 12,17 | - | - | Consumo |
| T2 | 10 | 0 | 5 | 91,67 | - | - | - | Sem ação |
| T3 | 15 | -10 | 15 | 75 | 12,02 | - | - | Consumo |
| T4 | 20 | 0 | 15 | 75 | - | - | - | Sem ação |
| T5 | 25 | -15 | 30 | 50 | 12,14 | - | - | Consumo |
| T6 | 30 | 0 | 30 | 50 | - | 40 | 33,33 | Primeira previsão observada |
| T7 | 35 | -10 | 40 | 33,33 | 11,49 | 52 | 13,89 | Previsão indicando baixa disponibilidade nos próximos 10min |
| T8 | 40 | -5 | 45 | 25 | 11,99 | 55 | 8,33 | Previsão indicando baixa disponibilidade nos próximos 10min |
| T9 | 45 | -10 | 55 | 8,33 | 11,77 | 60 | 0 | Baixa disponibilidade indicada e reposição realizada |
| T10 | 50 | 0 | 0 | 100 | 12,82 | - | - | Estoque repost |
| T11 | 55 | -10 | 10 | 83,33 | 12,91 | - | - | Consumo |
| T12 | 60 | -20 | 30 | 50 | 12,22 | - | - | Consumo |

Fonte: elaborado pelo próprio autor.

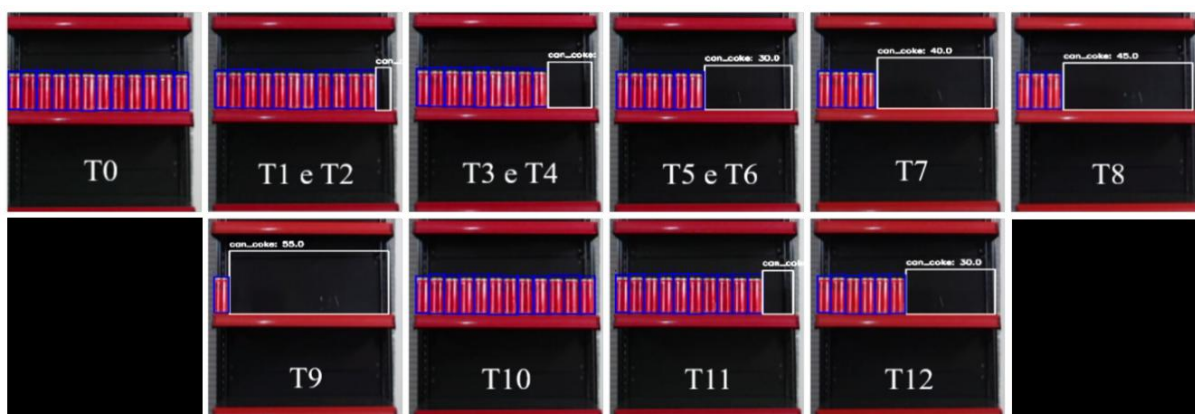


Figura 45 - Ações tomadas em cada momento T do Teste 3.

Fonte: elaborado pelo próprio autor.

É importante também destacar as configurações utilizadas na função `auto_arima` para se realizar as previsões no painel de Previsão Padrão após os 30 primeiros minutos. A configuração dos parâmetros selecionados privilegia rapidez e robustez na escolha de modelos ARIMA que lidam com janelas curtas. A Tabela 10 ilustra estes parâmetros juntamente com uma breve explicação sobre cada um deles.

Tabela 10 - Parâmetros do `auto_arima` utilizado no painel de Previsão Padrão

| Parâmetro | Valor | Explicação |
|------------------------------|--------------|---|
| <i>stepwise</i> | <i>True</i> | Vantajoso em janelas curtas, pois utiliza de uma heurística que reduz o número de modelos testados (muitas combinações de parâmetros (p, d, q)) |
| <i>seasonal</i> | False | Evita estimativas instáveis de sazonalidade, pois em janelas curtas geralmente poucos ciclos são observáveis |
| <i>max_p</i> | 2 | Limita a ordem AR para dependências de curto prazo sem superestimar a janela |
| <i>max_q</i> | 2 | Limita a ordem MA para ruído de curto alcance |
| <i>max_d</i> | 1 | Permite uma única diferenciação, suficiente na maioria dos valores com degraus curtos |
| <i>d</i> | <i>None</i> | Permite que o modelo determine automaticamente o número de diferenciações para tornar a série estacionária |
| <i>test</i> | “kpss” | Detecta com sensibilidade a não-estacionaridade em series curtas, ajudando d |
| <i>information_criterion</i> | “aicc” | O aicc penaliza modelos complexos para amostras pequenas |
| <i>error_action</i> | “ignore” | Evita que falhas no ajuste de modelos candidatos interrompam a busca dos melhores parâmetros (p, d, q) |

Fonte: elaborado pelo próprio autor

Na Tabela 9 é possível observar o valor do índice de esvaziamento e da disponibilidade estimada no momento T6, que são de 30 e 50%, respectivamente. Além disso, como já se passaram 30min, a previsão foi executada pela primeira vez. Como pode ser observado, no momento T6, o `auto_arima` conseguiu detectar a tendência de consumo crescente e calcular o

possível valor de índice e disponibilidade futuros (dos próximos 10min), que são de 40 e 33,33%, respectivamente. A tendência de consumo do momento T6 é ilustrada na Figura 46.



Figura 46 - Tendência de consumo crescente até o momento T6 (30min).
Fonte: elaborado pelo próprio autor.

Logo após o consumo realizado no momento T7, apesar do painel descritivo de *Ranking* indicar uma disponibilidade estimada de 33,33% (ou seja, ainda acima do limite configurado para disponibilidade), o painel de Previsão Padrão já adianta uma informação importante. Segundo os dados apresentados por ele, baseado nos padrões de consumo da última janela deslizante, nos próximos 10 minutos poderá haver falta de produtos, pois a disponibilidade futura poderá chegar a 13,89%. Este é um sinal importante aos *stakeholders*, que poderiam possivelmente tomarem uma ação imediata ou se prepararem para uma reposição próxima.

Neste cronograma de eventos, a reposição foi realizada logo após o momento T9, em que após um consumo, o painel descritivo de *Ranking* apontou claramente um valor de disponibilidade abaixo dos 20% configurados. No momento da reposição, foi possível observar a capacidade do GD de detectá-la, conforme ilustrado na Figura 47. Além disso, é importante ressaltar que após um recarregamento, o GD foi configurado para executar uma pausa preditiva do respectivo produto reposto.

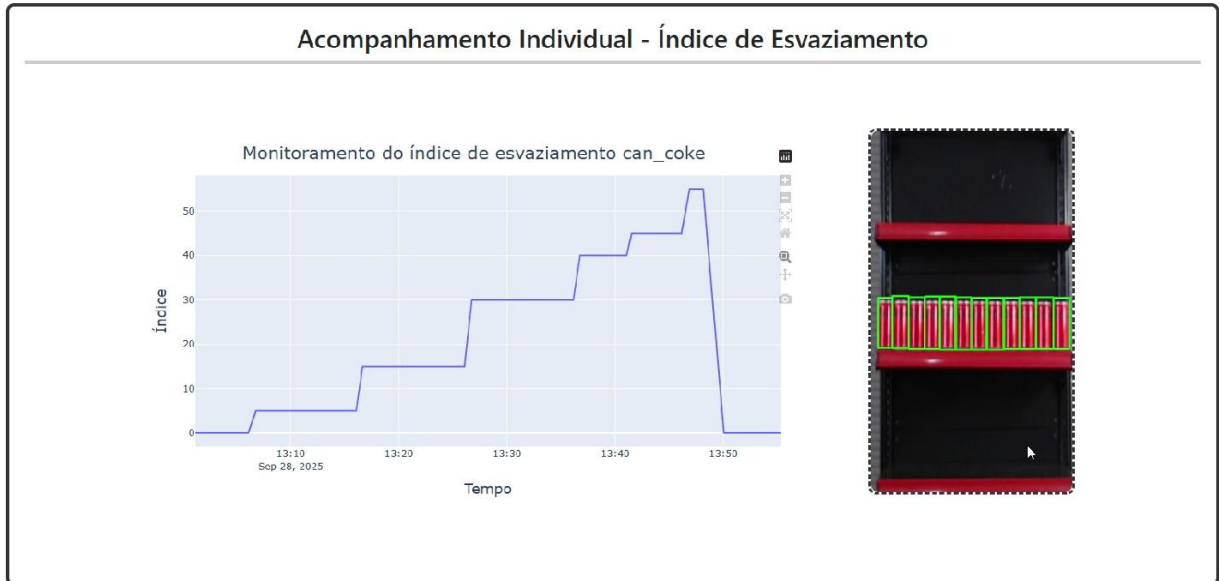


Figura 47 – Abastecimento ilustrado no momento T10.

Fonte: elaborado pelo próprio autor.

Ao interromper temporariamente as previsões até que a nova janela seja preenchida com dados coerentes com a situação atual, evita-se que o modelo produza previsões instáveis, enviesadas ou baseadas em padrões do passado que não representam mais a realidade (neste caso, uma diminuição drástica do Índice de Esvaziamento), garantindo maior confiabilidade e acurácia nos resultados (Schaffer; Dobbins; Pearson, 2021).

Embora o uso do `auto_arima` tenha se mostrado eficaz para capturar tendências de consumo e antecipar possíveis rupturas no teste realizado, em trabalhos futuros é importante compará-lo com outros algoritmos de previsão para garantir maior robustez nas previsões. Essa comparação pode usar métricas de erro, como *Mean Absolute Error* (MAE), *Mean Absolute Percentage Error* (MAPE) e *Root Mean Square Error* (RMSE), além de avaliar tempo de resposta e adaptação a mudanças bruscas, permitindo escolher o modelo mais preciso e eficiente para diferentes situações.

5. CONCLUSÕES

5.1 Síntese dos Resultados

Este trabalho buscou ilustrar uma proposta de abordagem prática de GD que pode atuar como uma ferramenta de apoio na tomada de decisões em estabelecimentos varejistas, mais precisamente no suporte ao gerenciamento de inventários locais em gôndolas. Porém, inicialmente buscou-se compreender o potencial uso dos GDs de forma ampla no setor varejista, considerado de grande relevância na economia.

Por meio da realização de uma RSL inédita na literatura, foi possível evidenciar o potencial do uso de GDs no setor ao possibilitar se combinar técnicas como Análise de Dados, IA, M&S e Análise Estatística com uso de IoT, câmeras e sensores para se monitorar ambientes físicos e gerar representações virtuais de uso constante.

Ao utilizar dados históricos, simulados, em tempo real ou quase real, os GDs têm permitido principalmente melhorar operações presentes nas cadeias de suprimentos dos estabelecimentos, prever o comportamento dos consumidores e deixar as tomadas de decisões em estabelecimentos cada vez mais orientadas a dados.

Apesar do avanço principalmente nas atividades citadas, o estudo evidenciou que muitas soluções ainda operam com baixa autonomia, atualização limitada em tempo real ou quase real, e alta dependência de dados secundários. Além disso, aponta áreas que ainda são pouco exploradas, representando oportunidades importantes para pesquisas e para a adoção prática dos GDs no varejo, como é o caso da gerência de inventários, área explorada neste trabalho.

A junção de abordagens como GD e VC representa um passo importante para criação de um varejo mais conectado, orientado a dados e preditivo nas atividades de gerência de inventários. A partir da integração de um modelo computacional a um ambiente ou ativo físico real monitorado por câmeras, é possível espelhar processos por meio de uma cópia virtual inteligente e de uso constante.

A VC é estratégica para criação de GDs pois transforma imagens e vídeos do ambiente físico em um fluxo contínuo de dados estruturados, permitindo monitorar e atualizar o estado real do ativo monitorado em tempo real ou quase real. Essa abordagem reduz a necessidade de sensores físicos adicionais, viabiliza a coleta de dados em grande escala e garante que o GD

permaneça sincronizado ao ambiente físico, produzindo dados descritivos e preditivos para auxílio na tomada de decisões.

A proposta desenvolvida neste trabalho se caracteriza como um GD pois mantém uma representação digital contínua de um ativo físico, no caso, a gôndola, um tipo de estrutura estratégica e popular em estabelecimentos varejistas. Ele é atualizado de forma sincronizada em tempo real a partir de dados capturados e tratados por uma fusão hierárquica de modelos de VC baseada em YOLO, um algoritmo que vem ganhando grande destaque e popularidade entre os algoritmos de VC, capaz de realizar a detecção rápida e precisa de produtos e *gaps* em tempo real quando obedecida as limitações do sistema (que serão melhores descritas no subtópicos de limitações da pesquisa).

A detecção automática de *gaps* pelo GD produz dados fundamentais para se calcular um índice de esvaziamento dos produtos monitorados e, conseqüentemente, calcular uma estimativa de disponibilidade destes produtos. O índice proposto busca oferecer uma medida objetiva e quantitativa do grau de esvaziamento de um produto, que amplia a análise para além da detecção somente visual. Ao traduzir *gaps* em estimativas numéricas de consumo, o índice acrescenta uma dimensão quantitativa e operacional ao monitoramento, buscando oferecer uma base analítica mais sólida para suporte à decisão sobre disponibilidade e reposição.

Ademais, além de prover estas informações de forma descritiva em quase tempo real por meio de painéis presentes em um *dashboard* objetivo, a proposta incorpora uma camada preditiva com análise de séries temporais em tempo real baseadas em *auto_arima*, buscando assim estimar o comportamento futuro de consumo, possibilitando antecipação de cenários e apoio à tomada de decisões de recarregamento.

Na RSL, também foi identificado que um dos maiores desafios de se utilizar GDs se encontra no alto grau de heterogeneidade das tecnologias envolvidas. Por isso, buscando tornar a proposta mais acessível e melhor replicável, optou-se por uma arquitetura baseada em tecnologias de baixo custo e amplamente disponíveis, mesmo reconhecendo as limitações do sistema.

A implementação foi desenvolvida majoritariamente em Python, utilizando de recursos reconhecidos para ciência de dados e VC, como NumPy e Pandas para manipulação e análise de dados, OpenCV para pré-processamento de imagens e suporte a operações de visão, e o YOLOv8 integrado ao Roboflow para treinamento e gestão do conjunto de dados de detecção de produtos e *gaps*.

Para previsão de consumo em curto prazo, empregou-se o *auto_arima*, enquanto a persistência dos dados históricos foi estruturada em MySQL, com comunicação via MySQL Connector. A apresentação dos resultados e indicadores de apoio à decisão foi realizada por meio de um *dashboard* interativo desenvolvido com Dash, permitindo visualizações em tempo real.

Além disso, para os experimentos e testes de prova de conceito, foi utilizada a câmera Logitech C920, um modelo de boa relação custo-benefício, compatível com ambientes *indoor* e suportado nativamente pelas bibliotecas utilizadas. Essa combinação de ferramentas visa demonstrar que é possível construir um GD semiautônomo funcional e útil para apoio à decisão no varejo sem depender de soluções proprietárias complexas ou equipamentos de alto custo, favorecendo a reprodutibilidade e adoção em diferentes contextos.

Adicionalmente, foi elaborado um manual técnico de caráter didático voltado à adaptação do GD proposto, com o intuito de orientar, de forma clara e estruturada, o processo de reconfiguração do sistema. O material apresenta, de maneira sequencial e ilustrada, os procedimentos necessários à organização, parametrização, ao retreinamento e à validação dos modelos de VC do sistema. Essa abordagem visa facilitar a transferência de conhecimento técnico e permitir que o sistema seja replicado e ajustado a diferentes realidades operacionais sem comprometer sua coerência conceitual. Assim, o manual contribui tanto para o caráter adaptativo e escalável do GD quanto para a democratização de sua aplicação em diferentes contextos.

É importante ressaltar que a validação da proposta foi conduzida com base em um processo de modelagem e validação experimental, incluindo testes controlados com variações de *zoom*, iluminação e *layouts*, oclusões por pessoas e simulação de consumo dinâmico com previsões também em tempo real por meio de janela deslizante. Desta forma, a proposta transcende a função de monitoramento ao integrar uma representação digital contínua de um ativo físico, cálculo de índices derivados da análise de *gaps*, análise de dados em fluxo e previsão de comportamento futuro.

No primeiro teste realizado sobre a proposta, resultados indicaram que, ao utilizar filmagem com *zoom* focado e iluminação adequada, o GD opera com TE muito baixas, próximas de 5%, reforçando a importância de boas condições visuais para maximizar a acurácia de detecção dos modelos de VC treinados para este teste.

É importante também ressaltar que, como os resultados derivam de um modelo de VC treinado de forma customizada, treinos com conjuntos de imagens que representem diferentes condições de *zoom* e iluminação usadas inicialmente podem alterar as TEs, assim, reduzindo-as quando há melhor cobertura das diferentes situações. Já o TR necessário para monitoramento dos 3 produtos presentes no experimento, independente das condições de *zoom* e iluminação, o GD apresentou latência média estável, indicando que o TR é controlado principalmente pelo processamento interno da proposta, não pela qualidade visual das imagens capturadas.

No segundo teste foi possível observar a alta acurácia do modelo YOLOv8m padrão para detecção de oclusões de pessoas em diferentes formatos. Em implementações futuras, este modelo poderá ser incrementado ou substituído por outro modelo, padrão ou customizado, contendo vários outros tipos de oclusões que possam ser pertinentes de serem tratadas. Já no terceiro teste, foi possível observar a capacidade do *auto_arima* de detectar a tendência de consumo e antecipar possíveis tomadas de decisões ao apresentar valores futuros no painel de Previsão Padrão.

Em suma, a proposta desenvolvida demonstra que é possível construir um GD para monitoramento das prateleiras de varejo com baixo custo e alta aplicabilidade prática, integrando VC baseada em YOLO, BDs relacionais e análise de dados fazendo uso de recursos poderosos da linguagem Python. Ao oferecer um painel dinâmico que monitora *gaps*, estima disponibilidade de produtos e antecipa tendências de consumo com base em séries temporais, o GD busca ampliar a capacidade de tomada de decisão dos *stakeholders* e reforça a viabilidade de soluções acessíveis para gestão de inventários e reposições mais assertivas.

Essa abordagem contribui para a área ao demonstrar como técnicas modernas de IA, análise de dados e IoT podem ser aplicadas de forma integrada em GDs voltados ao varejo, abrindo um leque de oportunidades para futuras pesquisas em predição de demanda, simulação de cenários e melhorias em processos de reposição.

5.2 Limitações da Pesquisa

Apesar da contribuição trazida por este trabalho levando em consideração os pontos levantados na Síntese dos Resultados, é importante ressaltar que a proposta possui limitações que, neste momento, deverão ser obedecidas para que ela opere da melhor forma possível.

Por ser uma solução voltada ao uso de câmeras 2-D de custo-benefício, a intenção da proposta desde o início não seria realizar a detecção precisa da falta de produtos por meio da

contagem deles, mas por meio da detecção de *gaps* que dariam “pistas” da falta de produtos e de quantos produtos possivelmente poderiam caber dentro deles. Em suma, quanto maior o *gap* detectado, maior a estimativa de consumo do respectivo produto.

A Figura 16 deste trabalho, por exemplo, ilustra um caso em que a proposta neste momento seria de grande apoio. Porém, a abordagem baseada na identificação de *gaps* apresenta sensibilidade a *layouts* desorganizados, o que pode degradar a fidelidade do índice de esvaziamento. Para mitigar essa limitação, propõe-se padronizações no planograma das prateleiras por meio da boa organização das filas (produtos em profundidade) ou inserção de divisores discretos entre elas, mantendo os produtos alinhados.

Por meio de testes realizados, foi possível observar que variações fotométricas (iluminação) e de enquadramento (*zoom*) impactam consideravelmente a TE do GD. Por isso, ao utilizar a abordagem, é importante que os modelos de VC sejam treinados com imagens em condições que representem bem o ambiente onde será aplicado, buscando diminuir ao máximo a TE de detecção. Além disso, os fundos das prateleiras podem possuir padrões diferentes em cada estabelecimento, por isso, também devem ser levados em consideração ao se treinar o modelo Generalista de VC para cada local.

Apesar das baixas TEs e TRs quando utilizado em condições favoráveis, a proposta foi validada por meio de uma prova de conceito (POC, do inglês *Proof of Concept*) de uma única gôndola levando 3 tipos de produtos em consideração. O aumento significativo de produtos em uma ou múltiplas gôndolas poderia trazer resultados inesperados, como uma sobrecarga de processamento, consequentemente prejudicando a qualidade das detecções e aumentando as TEs e TRs.

O ARIMA é considerado um dos métodos mais relevantes para previsão de séries temporais. Por meio do `auto_arima`, é possível automatizar a seleção dos melhores parâmetros do modelo ARIMA, economizando tempo e aumentando a chance de obter previsões estatísticas precisas sem testes manuais extensivos, mesmo com séries curtas, e adaptar-se rapidamente a mudanças nos dados em tempo real. Porém, a fim de validação do uso de previsões em tempo real por meio da técnica de janela deslizante, ele foi o único algoritmo utilizado e não se sabe como poderá se comportar em situações adversas ainda não observadas.

E claro, a proposta no momento foi testada sob uma gôndola profissional e com produtos reais, porém, resultados obtidos em cenários controlados podem não levar em consideração outras variáveis presentes em um ambiente real de operação. Em laboratório, as condições

tendem a ser mais previsíveis em comparação ao contexto real de operação e não se permite avaliar plenamente a usabilidade da proposta com os *stakeholders*. Sugestões de trabalhos que explorem as três últimas limitações mencionadas são sugeridos na seção de Recomendações de trabalhos futuros.

5.3 Recomendações para trabalhos futuros

Algumas oportunidades de pesquisa podem ser destacadas, tanto para aprimorar a abordagem apresentada quanto para expandir seu campo de atuação. Recomendações em 8 diferentes perspectivas são listadas a seguir:

- **Visão Computacional:** nesta perspectiva, sugerem-se os seguintes direcionamentos de pesquisa:
 - Os modelos de VC podem ser incrementados com a capacidade de detectar *layouts* desorganizados, permitindo assim avisar e auxiliar os *stakeholders* em questões voltadas ao planograma das prateleiras.
 - O modelo para detecção de oclusões pode ser incrementado para permitir a detecção de novas possíveis oclusões.
 - Por meio de informações pré-cadastradas no BD sobre as dimensões dos produtos, detecções dos produtos específicos e das prateleiras, uma heurística pode ser proposta para permitir o GD ser calibrado automaticamente.
 - Novas fusões hierárquicas de modelos fazendo uso das demais variações do YOLO podem ser criadas e comparadas a fim de avaliar ganhos de performance.
- **Melhorias no cálculo do Índice de Esvaziamento:** neste momento, a largura média dos produtos detectados em uma prateleira pelo modelo Especialista de VC é um fator importante que determina quantos produtos cabem dentro de um *gap* detectado nela. Testes com pequenas variações na largura do produto ou junção de vários *gaps* bem pequenos poderiam ser realizados a fim de verificar uma possível diminuição nas TEs das detecções.
- **Adição de novos painéis:** novos painéis com KPIs, simulações e previsões específicas poderiam ser adicionados. Número de rupturas diárias, *lead time* de recarregamento, tempo médio de recarregamento, tempo que um cliente permanece em frente uma prateleira, entre outros, poderiam aproveitar das capacidades oferecidas pela VC e dos dados coletados para

aumentar a quantidade de informações úteis aos *stakeholders* para auxílio na tomada de decisão.

- **Adição de novos algoritmos de previsão:** apesar da popularidade do ARIMA, método de previsão utilizado neste trabalho, é importante que outros métodos de previsão sejam adicionados e comparados a fim de não só buscar previsões cada vez mais precisas, mas também avaliar quais métodos operam melhor em diferentes situações que possam acontecer durante o consumo de produtos nas prateleiras.
- **Escalabilidade e testes de performance:** o POC proposto neste trabalho não levou em consideração o aumento significativo da quantidade de produtos em monitoramento. Por isso, é importante que testes de performance sejam realizados a fim de avaliar a robustez da proposta nesta perspectiva. É importante também que o sistema seja escalado para se trabalhar com um conjunto de gôndolas, capturando dados de consumo de múltiplas câmeras. Ainda nesta perspectiva, é importante medir a generalização do GD ao adequá-lo a novos tipos de produtos por meio do uso do manual de adaptação proposto.
- **Aplicação em ambientes reais:** a capacidade dos estabelecimentos de monitorar seus ativos por meio de VC está se tornando uma realidade cada vez maior com o barateamento de câmeras 2-D no mercado. Porém, é importante ressaltar que nem todas as gôndolas possam precisar de monitoramento, mas inicialmente, o foco pode ficar naquelas mais estratégicas, como por exemplo, gôndolas de produtos em promoção. É importante que futuramente a proposta seja validada em gôndolas estratégicas reais presentes em estabelecimentos. Além disso, é importante que dados relacionados à percepção dos *stakeholders* sobre a operação e uso da proposta sejam coletados para possíveis ajustes e melhorias na proposta.
- **Viabilidade financeira de implementação:** propõe-se também realizar um estudo de análise financeira para avaliar custos e benefícios da implementação da proposta, considerando *hardware* e *software* necessários, a fim de viabilizar uma implantação custo-efetiva e escalável em diferentes perfis de varejo.
- **Adaptação para novos contextos no suporte ao controle de inventários:** apesar do foco do GD proposto ser inicialmente voltado ao monitoramento de gôndolas presentes no setor varejista, ele é aplicável a qualquer ambiente onde o estoque é fisicamente visível, o

consumo gera esvaziamento progressivo e a ruptura tem custo operacional relevante. Por isso, pode ser facilmente adaptado para novos contextos como a manufatura (Ex: prevenção de parada de linha por falta de componentes), hospitais (Ex: acompanhamento constante da disponibilidade de insumos e materiais críticos) e centros de distribuição (Ex: otimização do reabastecimento de áreas de *picking*).

REFERÊNCIAS

ACKERMANN, Andres E. F.; SELBITTO, Miguel A. Métodos de previsão de demanda: uma revisão da literatura. **Innovar**, [S.L.], v. 32, n. 85, p. 83-99, 1 jan. 2022. Universidad Nacional de Colombia. <http://dx.doi.org/10.15446/innovar.v32n85.100979>.

ADIKARI, Sasadara B.; GANEGODA, Naleen C.; MEEGAMA, Ravinda G. N.; WANNIARACHCHI, Indika L. Applicability of a Single Depth Sensor in Real-Time 3D Clothes Simulation: augmented reality virtual dressing room using kinect sensor. **Advances In Human-Computer Interaction**, [S.L.], v. 2020, p. 1-10, 18 mai. 2020. Hindawi Limited. <http://dx.doi.org/10.1155/2020/1314598>.

AGARWAL, Meghann; CALLOWAY, Chris; NIEDERHUT, Dillon (eds.). **Proceedings of the 22nd Python in Science Conference (SciPy 2023)**, Austin, TX, 10–16 jul. 2023. SciPy. <http://dx.doi.org/10.25080/gerudo-f2bc6f59-039>.

AGRAWAL, Ashwin; THIEL, Robert; JAIN, Pooja; SINGH, Vishal; FISCHER, Martin. Digital Twin: Where do humans fit in?. **Automation In Construction**, [S.L.], v. 148, p. 104749, abr. 2023. Elsevier BV. <http://dx.doi.org/10.1016/j.autcon.2023.104749>.

AIT-ALLA, Abderrahim; KREUTZ, Markus; RIPPEL, Daniel; LÜTJEN, Michael; FREITAG, Michael. Simulated-based methodology for the interface configuration of cyber-physical production systems. **International Journal of Production Research**, [S.L.], v. 59, n. 17, p. 5388-5403, 17 jun. 2020. Informa UK Limited. <http://dx.doi.org/10.1080/00207543.2020.1778209>.

AL-QAZZAZ, Redha Ali; YOUSIF, Suhad A.. High performance time series models using auto autoregressive integrated moving average. **Indonesian Journal Of Electrical Engineering And Computer Science**, [S.L.], v. 27, n. 1, p. 422, 1 jul. 2022. Institute of Advanced Engineering and Science. <http://dx.doi.org/10.11591/ijeecs.v27.i1.pp422-430>.

AMARNATH, Aakash; N, Amudhan A.; V, Manoj K. Empowering Digital Twin Development: a profound exploration with altair tools. **2023 International Conference on Self Sustainable Artificial Intelligence Systems (ICSSAS)**, [S.L.], p. 1146-1153, 18 out. 2023. IEEE. <http://dx.doi.org/10.1109/icssas57918.2023.10331675>.

ANTCZAK, Tomasz; WERON, Rafal; ZABAWA, Jacek. Data-Driven Simulation Modeling of the Checkout Process in Supermarkets: insights for decision support in retail operations. **IEEE Access**, [S.L.], v. 8, p. 228841-228852, 2020. Institute of Electrical and Electronics Engineers (IEEE). <http://dx.doi.org/10.1109/access.2020.3045919>.

ARVAN, Meysam; FAHIMNIA, Behnam; REISI, Mohsen; SIEMSEN, Enno. Integrating human judgement into quantitative forecasting methods: a review. **Omega**, [S.L.], v. 86, p. 237-252, jul. 2019. Elsevier BV. <http://dx.doi.org/10.1016/j.omega.2018.07.012>.

ASHRAFIAN, Alireza; PEDERSEN, Sigmund. Digital Twin for Complex Logistics Systems: the case study of a large-scale automated order picking and fulfillment system. **IFAC-Papersonline**, [S.L.], v. 56, n. 2, p. 11056-11061, 2023. Elsevier BV. <http://dx.doi.org/10.1016/j.ifacol.2023.10.808>.

AUGUSTINE, Peter. The industry use cases for the Digital Twin idea. **Advances In Computers**, [S.L.], p. 79-105, 2020. Elsevier.
<http://dx.doi.org/10.1016/bs.adcom.2019.10.008>.

AZEVEDO, Rogério Cabral de; ENSSLIN, Leonardo. **Metodologia da Pesquisa para Engenharias**. Belo Horizonte: CEFET-Mg, 2020. 196 p.

BAAS, Jeroen; SCHOTTEN, Michiel; PLUME, Andrew; CÔTÉ, Grégoire; KARIMI, Reza. Scopus as a curated, high-quality bibliometric data source for academic research in quantitative science studies. **Quantitative Science Studies**, [S.L.], v. 1, n. 1, p. 377-386, fev. 2020. MIT Press - Journals. http://dx.doi.org/10.1162/qss_a_00019.

BAKHAREV, Vladimir; MITYASHIN, Gleb; KATRASHOVA, Yuliya; STRELNIKOV, Andrey; BUGAENKO, Alexey; KARACHEV, Vyacheslav. The Impact of Industry 4.0 Technologies on Retail Development. **Proceedings of the International Scientific Conference - Digital Transformation on Manufacturing, Infrastructure and Service**, [S.L.], p. 1-7, 18 nov. 2020. ACM. <http://dx.doi.org/10.1145/3446434.3446548>.

BARNES, Robert. Ambient Sound Recognition and Processing Tools, Object Perception and Motion Control Algorithms, and Behavioral Predictive Analytics in the Virtual Economy of the Metaverse. **Analysis and Metaphysics**, [S.L.], v. 21, p. 159-175, 2022. Addleton Academic Publishers. <http://dx.doi.org/10.22381/am21202210>.

BEETZ, Michael; STELTER, Simon; BEßLER, Daniel; DHANABALACHANDRAN, Kaviya; NEUMANN, Michael; MANIA, Patrick; HAIDU, Andrei. Robots Collecting Data: modelling stores. **Springer Tracts in Advanced Robotics**, [S.L.], p. 41-64, 2022. Springer International Publishing. http://dx.doi.org/10.1007/978-3-031-06078-6_2.

BEREGI, Richárd; SZALLER, Ádám; KÁDÁR, Botond. Synergy of multi-modelling for process control. **IFAC-Papersonline**, [S.L.], v. 51, n. 11, p. 1023-1028, 2018. Elsevier BV. <http://dx.doi.org/10.1016/j.ifacol.2018.08.473>.

BERGS, Thomas; GIERLINGS, Sascha; AUERBACH, Thomas; KLINK, Andreas; SCHRAKNEPPER, Daniel; AUGSPURGER, Thorsten. The Concept of Digital Twin and Digital Shadow in Manufacturing. **Procedia Cirp**, [S.L.], v. 101, p. 81-84, 2021. Elsevier BV. <http://dx.doi.org/10.1016/j.procir.2021.02.010>.

BURGOS, Diana; IVANOV, Dmitry. Food retail supply chain resilience and the COVID-19 pandemic: a digital twin-based impact analysis and improvement directions. **Transportation Research Part e: Logistics and Transportation Review**, [S.L.], v. 152, p. 102412, ago. 2021. Elsevier BV. <http://dx.doi.org/10.1016/j.tre.2021.102412>.

CHAN, Chi On; LAU, H. C.W.; FAN, Youqing. Implementing IoT-Adaptive Fuzzy Neural Network Model Enabling Service for Supporting Fashion Retail. **Proceedings of the 4Th International Conference on Machine Learning and Soft Computing**, [S.L.], p. 19-24, 17 jan. 2020. ACM. <http://dx.doi.org/10.1145/3380688.3380692>.

CHINOSI, Michele; TROMBETTA, Alberto. BPMN: an introduction to the standard. **Computer Standards & Interfaces**, [S.L.], v. 34, n. 1, p. 124-134, jan. 2012. Elsevier BV. <http://dx.doi.org/10.1016/j.csi.2011.06.002>.

CHUANG, Howard Hao-Chun. Fixing shelf out-of-stock with signals in point-of-sale data. **European Journal of Operational Research**, [S.L.], v. 270, n. 3, p. 862-872, nov. 2018. Elsevier BV. <http://dx.doi.org/10.1016/j.ejor.2017.10.059>.

CULLER, David; LONG, James. A Prototype Smart Materials Warehouse Application Implemented Using Custom Mobile Robots and Open Source Vision Technology Developed Using EmguCV. **Procedia Manufacturing**, [S.L.], v. 5, p. 1092-1106, 2016. Elsevier BV. <http://dx.doi.org/10.1016/j.promfg.2016.08.080>.

DA VEIGA, Claudimar Pereira; DA VEIGA, Cássia Rita Pereira; CATAPAN, Anderson; TORTATO, Ubiratã; DA SILVA, Wesley Vieira. Demand forecasting in food retail: A comparison between the Holt-Winters and ARIMA models. **WSEAS transactions on business and economics**, v.11, p. 608-614, 2014. <https://wseas.com/journals/articles.php?id=5004>

DAWID, Aleksander; BUCHWALD, Paweł; PAWLAK, Bartłomiej. The Digital Twin to Train a Neural Network Detecting Headlamps Failure of Motor Vehicles. **Lecture Notes in Networks and Systems**, [S.L.], p. 29-38, 2023. Springer Nature Switzerland. http://dx.doi.org/10.1007/978-3-031-37720-4_3.

DEFRAEYE, Thijs; SHRIVASTAVA, Chandrima; BERRY, Tarl; VERBOVEN, Pieter; ONWUDE, Daniel; SCHUDEL, Seraina; BÜHLMANN, Andreas; CRONJE, Paul; ROSSI, René M.. Digital twins are coming: will we need them in supply chains of fresh horticultural produce?. **Trends in Food Science & Technology**, [S.L.], v. 109, p. 245-258, mar. 2021. Elsevier BV. <http://dx.doi.org/10.1016/j.tifs.2021.01.025>.

DENYER, David; TRANFIELD, David. Producing a systematic review. In: BUCHANAN, David; BRYMAN, Alan. **The SAGE Handbook of Organizational Research Methods**. Sage, 2009. p. 671–689.

DING, Kai; CHAN, Felix T.s.; ZHANG, Xudong; ZHOU, Guanghui; ZHANG, Fuqiang. Defining a Digital Twin-based Cyber-Physical Production System for autonomous manufacturing in smart shop floors. **International Journal of Production Research**, [S.L.], v. 57, n. 20, p. 6315-6334, 16 jan. 2019. Informa UK Limited. <http://dx.doi.org/10.1080/00207543.2019.1566661>.

DIWAN, Tausif; ANIRUDH, G.; TEMBHURNE, Jitendra V.. Object detection using YOLO: challenges, architectural successors, datasets and applications. **Multimedia Tools and Applications**, [S.L.], v. 82, n. 6, p. 9243-9275, 8 ago. 2022. Springer Science and Business Media LLC. <http://dx.doi.org/10.1007/s11042-022-13644-y>.

DURANA, Pavol; MUSOVA, Zdenka; CUȚITOI, Adela-Claudia. Digital Twin Modeling and Spatial Awareness Tools, Acoustic Environment Recognition and Visual Tracking Algorithms, and Deep Neural Network and Vision Sensing Technologies in Blockchain-based Virtual Worlds. **Analysis and Metaphysics**, [S.L.], v. 21, p. 261-277, dez. 2022. Addleton Academic Publishers. <https://dx.doi.org/10.22381/am21202216>.

ELBOUZIDI, Adnane Drissi; BÉLANGER, Marie-Jane; CADI, Abdessamad Ait El; PELLERIN, Robert; LAMOURI, Samir; VALENCIA, Estefania Tobon. The Role of AI in

Warehouse Digital Twins. **Proceedings of the European Modeling & Simulation Symposium, EMSS**, [S.L.], p. 1-8, 2022. CAL-TEK srl.
<http://dx.doi.org/10.46354/i3m.2022.emss.024>.

ELBOUZIDI, Adnane Drissi; CADI, Abdessamad Ait El; PELLERIN, Robert; LAMOURI, Samir; VALENCIA, Estefania Tobon; BÉLANGER, Marie-Jane. The Role of AI in Warehouse Digital Twins: literature review. **Applied Sciences**, [S.L.], v. 13, n. 11, p. 6746, 1 jun. 2023. MDPI AG. <http://dx.doi.org/10.3390/app13116746>.

ERLINA, Tati; FIKRI, Muhammad. A YOLO Algorithm-based Visitor Detection System for Small Retail Stores using Single Board Computer. **Journal of Applied Engineering and Technological Science (JAETS)**, [S.L.], v. 4, n. 2, p. 908-920, 5 jun. 2023. Yayasan Riset dan Pengembangan Intelektual. <http://dx.doi.org/10.37385/jaets.v4i2.1872>.

FATORACHIAN, Hajar; KAZEMI, Hadi. Impact of Industry 4.0 on supply chain performance. **Production Planning & Control**, [S.L.], v. 32, n. 1, p. 63-81, 14 jan. 2020. Informa UK Limited. <http://dx.doi.org/10.1080/09537287.2020.1712487>.

FERNÁNDEZ, Manlio Massiris; DELRIEUX, Claudio; MUNOZ, J. Alvaro Fernandez. Automated Personnel Digital Twinning in Industrial Workplaces. **2022 International Conference on Electrical, Computer and Energy Technologies (ICECET)**, [S.L.], p. 1-6, 20 jul. 2022. IEEE. <http://dx.doi.org/10.1109/icecet55527.2022.9872882>.

FERNÁNDEZ-CARAMÉS, Tiago M.; FRAGA-LAMAS, Paula. A Review on Human-Centered IoT-Connected Smart Labels for the Industry 4.0. **IEEE Access**, [S.L.], v. 6, p. 25939-25957, 2018. Institute of Electrical and Electronics Engineers (IEEE).
<http://dx.doi.org/10.1109/access.2018.2833501>.

GAIKWAD, Vrushab. OpenCV: A Comprehensive Review and Applications in Computer Vision. **International Journal of Advanced Research in Science, Communication and Technology (IJARSCT)**, v. 3, n. 2, jul. 2023. <https://doi.org/10.48175/568>

GARTNER. **Gartner Identifies the Top 10 Strategic Technology Trends for 2017**. 2016. Disponível em: <https://www.gartner.com/en/newsroom/press-releases/2016-10-18-gartner-identifies-the-top-10-strategic-technology-trends-for-2017>. Acesso em: 22 mar. 2024.

GLAESSGEN, Edward; STARGEL, David. The Digital Twin Paradigm for Future NASA and U.S. Air Force Vehicles. **53rd AIAA/ASME/ASCE/AHS/ASC Structures, Structural Dynamics and Materials Conference<
>20Th AIAA/ASME/AHS Adaptive Structures Conference<
>14Th AIAA**, [S.L.], p. 1-14, 23 abr. 2012. American Institute of Aeronautics and Astronautics. <http://dx.doi.org/10.2514/6.2012-1818>.

GUPTA, Rajesh; TANWAR, Sudeep; AL-TURJMAN, Fadi; ITALIYA, Prit; NAUMAN, Ali; KIM, Sung Won. Smart Contract Privacy Protection Using AI in Cyber-Physical Systems: tools, techniques and challenges. **IEEE Access**, [S.L.], v. 8, p. 24746-24772, 2020. Institute of Electrical and Electronics Engineers (IEEE).
<http://dx.doi.org/10.1109/access.2020.2970576>.

GUTIERREZ-FRANCO, Edgar; MEJIA-ARGUETA, Christopher; RABELO, Luis. Data-Driven Methodology to Support Long-Lasting Logistics and Decision Making for Urban

Last-Mile Operations. **Sustainability**, [S.L.], v. 13, n. 11, p. 6230, 1 jun. 2021. MDPI AG. <http://dx.doi.org/10.3390/su13116230>.

HANKE, Jannis; HAUSER, Matthias; DÜRR, Alexander; THIESSE, Frederic. Redefining the Offline Retail Experience: designing product recommendation systems for fashion stores. **26Th European Conference on Information Systems**, Portsmouth, United Kingdom, p. 1-14, jun. 2018. https://aisel.aisnet.org/ecis2018_rp/1

HASHMI, Muhammad A.; MO, John P.T.; BECKETT, Ronald C.. Transdisciplinary systems approach to realization of digital transformation. **Advanced Engineering Informatics**, [S.L.], v. 49, p. 101316, ago. 2021. Elsevier BV. <http://dx.doi.org/10.1016/j.aei.2021.101316>.

HUSSAIN, Muhammad. YOLO-v1 to YOLO-v8, the Rise of YOLO and Its Complementary Nature toward Digital Manufacturing and Industrial Defect Detection. **Machines**, [S.L.], v. 11, n. 7, p. 677, 23 jun. 2023. MDPI AG. <http://dx.doi.org/10.3390/machines11070677>.

HATA, Nozomi; NAKAYAMA, Takashi; TANAKA, Akira; WAKAMATSU, Takashi; YOSHIDA, Akihiro; TATEIWA, Nariaki; NISHIKAWA, Yuri; OZAWA, Jun; FUJISAWA, Katsuki. Mobility Optimization on Cyber Physical System via Multiple Object Tracking and Mathematical Programming. **2018 IEEE International Conference on Big Data (Big Data)**, [S.L.], p. 4026-4035, dez. 2018. IEEE. <http://dx.doi.org/10.1109/bigdata.2018.8622146>.

HAUSER, Matthias, GÜNTHER, Sebastian A.; FLATH, Christoph Michael; THIESSE, Frederic. Designing Pervasive Information Systems: a fashion retail case study. **38Th International Conference on Information Systems (ICIS 2017)**, Seoul, South Korea, p. 1-16, dez. 2017. <https://aisel.aisnet.org/icis2017/HCI/Presentations/13>

HENDLER, Daniela. **Descubra como se prevenir da ruptura de gôndola**. 2019. Disponível em: <https://gestaodeclientes.com.br/ruptura-de-gondola/>. Acesso em: 18 jul. 2025.

HUNG, Yu-Hsin; WANG, Yi-Jie; CHANG, Ray-I. Investigation of the Effective Use of Ensemble Learning Algorithms for Cyber Data Analytics –The Prediction of the Customer Revenue on the Google Merchandise Store (GStore). **2020 The 4Th International Conference on E-Society, E-Education And E-Technology**, [S.L.], p. 76-82, 15 ago. 2020. ACM. <http://dx.doi.org/10.1145/3421682.3421690>.

JAIN, Aayush; SUKHDEVE, Tanay; GADIA, Himanshu; SAHU, Satya Prakash; VERMA, Satya. COVID19 prediction using time series analysis. **International Conference on Artificial Intelligence and Smart Systems (ICAIS)**. p. 1599-1606, 2021. <http://dx.doi.org/10.1109/ICAIS50930.2021.9395877>

JANHOFER, Dustin; BARANN, Benjamin; CORDES, Ann-Kristin; BECKER, Jörg. Mastering Omni-Channel Retailing Challenges with Industry 4.0 Concepts. **Proceedings of the Annual Hawaii International Conference on System Sciences**, [S.L.], p. 5389-5398, 2020. Hawaii International Conference on System Sciences. <http://dx.doi.org/10.24251/hicss.2020.662>.

JIANG, Peiyuan; ERGU, Daji; LIU, Fangyao; CAI, Ying; MA, Bo. A Review of Yolo Algorithm Developments. **Procedia Computer Science**, [S.L.], v. 199, p. 1066-1073, 2022. Elsevier BV. <http://dx.doi.org/10.1016/j.procs.2022.01.135>.

JIN, Yuchen; XU, Chicheng; LIN, Tao; LI, Weichang; ZEGHLACHE, Mohamed Larbi. Python Dash for Well Data Validation, Visualization, and Processing. **Petrophysics – The SPWLA Journal of Formation Evaluation and Reservoir Description**, v. 64, n. 4, p. 568–573, ago. 2023. <https://doi.org/10.30632/PJV64N4-2023a6>

JOCHER, Glenn; CHAURASIA, Ayush. **YOLO by Ultralytics**. 2023. Disponível em: <https://github.com/ultralytics/ultralytics?tab=security-ov-file>. Acesso em: 28 nov. 2023.

JUNAID FAROOQ, Muhammad; ZHU, Quanyan. PhD Forum: Enabling Autonomic IoT for Smart Urban Services. **2020 IEEE 6Th World Forum on Internet of Things (WF-IoT)**, [S.L.], p. 1-2, jun. 2020. IEEE. <http://dx.doi.org/10.1109/wf-iot48130.2020.9221230>.

KALE, P. D.; TANVI, Mahajan; ANKITA, Kanawade; SAMRUDHI, Shete; SHWETA, Jadhav. Comparative Analysis of Image Annotation Tools: LabelImg, VGG Annotator, Label Studio, and Roboflow. **Journal of Emerging Technologies and Innovative Research (JETIR)**, v. 11, n. 5, mai. 2024. <https://www.jetir.org/view?paper=JETIR2405D59>

KAMIEBISU, Ryuichi; SASO, Taiki; NAKAO, Jun; LIU, Ziang; NISHI, Tatsushi; MATSUDA, Michiko. Use cases of the platform for structuring a smart supply chain in discrete manufacturing. **Procedia Cirp**, [S.L.], v. 107, p. 687-692, 2022. Elsevier BV. <http://dx.doi.org/10.1016/j.procir.2022.05.046>.

KARAKÖSE, Mehmet; YETİŞ, Hasan. A Cyberphysical System Based Mass-Customization Approach with Integration of Industry 4.0 and Smart City. **Wireless Communications and Mobile Computing**, [S.L.], v. 2017, p. 1-9, 2017. Hindawi Limited. <http://dx.doi.org/10.1155/2017/1058081>.

KAUR, Jasleen; KAUR, Pankaj Deep. CE-GMS: A Cloud IoT-Enabled Grocery Management System. **Electronic Commerce Research and Applications**, [S.L.], v. 28, p. 63-72, mar. 2018. Elsevier BV. <http://dx.doi.org/10.1016/j.elerap.2018.01.005>.

KAUR, Manpreet; SALIM, Flora D.; REN, Yongli; CHAN, Jeffrey; TOMKO, Martin; SANDERSON, Mark. Joint Modelling of Cyber Activities and Physical Context to Improve Prediction of Visitor Behaviors. **ACM Transactions on Sensor Networks**, [S.L.], v. 16, n. 3, p. 1-25, 13 ago. 2020. Association for Computing Machinery (ACM). <http://dx.doi.org/10.1145/3393692>.

KIM, Sunyeong; NISHIMOTO, Keita; MIZUSHIMA, Masahide; YAMADA, Setsuo; TOMITA, Junji. Using Smart-store Behavior Data to Optimize Sales Promotion. **NTT Technical Review**, [S.L.], v. 22, n. 5, p. 71-77, maio 2024. NTT Science and Core Technology Laboratory Group. <http://dx.doi.org/10.53829/ntr202405fa9>.

KINASIH, Fabiola Maria Teresa Retno; MACHBUB, Carmadi; YULIANTI, Lenni; ROHMAN, Arief Syaichu. Two-stage multiple object detection using CNN and correlative filter for accuracy improvement. **Heliyon**, [S.L.], v. 9, n. 1, p. 12716, jan. 2023. Elsevier BV. <http://dx.doi.org/10.1016/j.heliyon.2022.e12716>.

KNOCH, Sonke; HUSTER, Marco; JOST, Matthias; MOHR, Jonas; MERKEL, Dieter; LUXENBURGER, Andreas; SCHWARTZ, Tim; KIM, Jeong-Jung; KOH, Doo-Yeol; PARK,

Jinseong. WALL-ET: assistance in supermarkets and warehouses through social cognitive robots. **2022 IEEE International Conference on Artificial Intelligence and Virtual Reality (AIVR)**, [S.L.], p. 219-220, dez. 2022. IEEE.
<http://dx.doi.org/10.1109/aivr56993.2022.00045>.

KONTOPOULOU, Vaia I.; PANAGOPOULOS, Athanasios D.; KAKKOS, Ioannis; MATSOPOULOS, George K.. A Review of ARIMA vs. Machine Learning Approaches for Time Series Forecasting in Data Driven Networks. **Future Internet**, [S.L.], v. 15, n. 8, p. 255, 30 jul. 2023. MDPI AG. <http://dx.doi.org/10.3390/fi15080255>.

KRISPIN, Rami. **Hands-on Time Series Analysis with R**. Birmingham: Packty Publishing Ltd., 2019. 448 p.

KRITZINGER, Werner; KARNER, Matthias; TRAAR, Georg; HENJES, Jan; SIHN, Wilfried. Digital Twin in Manufacturing: A Categorical Literature Review and Classification. **IFAC-Papersonline**, [S.L.], v. 51, n. 11, p. 1016-1022, 2018. Elsevier BV.
<http://dx.doi.org/10.1016/j.ifacol.2018.08.474>.

KRIZHEVSKY, Alex; SUTSKEVER, Ilya; HINTON, Geoffrey E. ImageNet Classification with Deep Convolutional Neural Networks. **Advances in Neural Information Processing Systems**, **25**, [S. L], p. 1-9, 2012. <http://papers.nips.cc/paper/4824-imagenetclassification-with-deep-convolutional-neural-networks.pdf>.

KUMAR, Krishna; SIMHA, Jay; AGARWAL, Rashmi. Retail Shelf Planning Using YOLO V8. **2023 Global Conference on Information Technologies and Communications (GCITC)**, [S.L.], p. 1-8, 1 dez. 2023. IEEE.
<http://dx.doi.org/10.1109/gcitic60406.2023.10426304>.

KÜMPEL, Michaela; MUELLER, Christian A.; BEETZ, Michael. Semantic Digital Twins for Retail Logistics. **Dynamics In Logistics**, [S.L.], p. 129-153, 2021. Springer International Publishing. http://dx.doi.org/10.1007/978-3-030-88662-2_7

KÜMPEL, Michaela; DECH, Jonas; HAWKIN, Alina; BEETZ, Michael. Robotic Shopping Assistance for Everyone: dynamic query generation on a semantic digital twin as a basis for autonomous shopping assistance. In: **AAMAS '23: PROCEEDINGS OF THE 2023 INTERNATIONAL CONFERENCE ON AUTONOMOUS AGENTS AND MULTIAGENT SYSTEMS**, 2023, London, UK. Poster Session II. London, UK: AAMAS '23, 2023. p. 2523-2525. <https://dl.acm.org/doi/abs/10.5555/3545946.3598989>

KÜMPEL, Michaela; BEETZ, Michael. ProductKG: a product knowledge graph for user assistance in daily activities. In: **9TH JOINT ONTOLOGY WORKSHOPS (JOWO 2023)**, 9., 2023, Québec, Canada. Ontology Showcase and Demonstrations Track. Québec, Canada.: Fois 2023, 2023. p. 1-14. <https://ceur-ws.org/Vol-3637/paper52.pdf>

LAW, Averill M.. **Simulation Modeling & Analysis**. 5. ed. Boston: McGraw-Hill, 2014. 800 p.

LEUSMANN, Nils; MIR, Glareh; NGUYEN, Hoang Giang; STELTER, Simon; DHANABALACHANDRAN, Kaviya; ODABASI, Cagatay; MALKI, Maik; HAWKIN, Alina; BAZLEN, Felix; BEETZ, Michael. Retail SemDT Collection Knowledge-base, A

Platform Architecture. **2023 IEEE Smart World Congress (SWC)**, [S.L.], p. 1-6, 28 ago. 2023. IEEE. <http://dx.doi.org/10.1109/swc57546.2023.10449259>.

LIN, Yu-Fong; YOON, So-Yeon. Exploring the Effects of Lighting on Consumer Responses in a Retail Environment using 3D Walk-Through Animation. **Archives Of Design Research**, [S.L.], v. 28, n. 2, p. 5, 31 maio 2015. Korean Society of Design Science. <http://dx.doi.org/10.15187/adr.2015.05.28.2.5>.

LIU, Jing; TANG, Zhongze; LIU, Changhe; ZHAO, Yijing; PAN, Xiazhuo. Highway Driving Safety Analysis and Management Based on Digital Twin. **2023 IEEE 3Rd International Conference on Digital Twins and Parallel Intelligence (DTPI)**, [S.L.], p. 1-7, 7 nov. 2023. IEEE. <http://dx.doi.org/10.1109/dtpi59677.2023.10365430>.

LIU, Li; OUYANG, Wanli; WANG, Xiaogang; FIEGUTH, Paul; CHEN, Jie; LIU, Xinwang; PIETIKÄINEN, Matti. Deep Learning for Generic Object Detection: a survey. **International Journal Of Computer Vision**, [S.L.], v. 128, n. 2, p. 261-318, 31 out. 2019. Springer Science and Business Media LLC. <http://dx.doi.org/10.1007/s11263-019-01247-4>.

LIU, Jingya; HUANG, Isaac; ANAND, Aishwarya; CHANG, Po-Hao; HUANG, Yufang. Digital Twin in Retail: an ai-driven multi-modal approach for real-time product recognition and 3d store reconstruction. **2024 IEEE Conference on Virtual Reality and 3D User Interfaces Abstracts and Workshops (VRW)**, [S.L.], p. 368-373, 16 mar. 2024. IEEE. <http://dx.doi.org/10.1109/vrw62533.2024.00072>.

LV, Zhihan; QIAO, Liang; MARDANI; Abbas; LV, Haibin. Digital Twins on the Resilience of Supply Chain Under COVID-19 Pandemic. **IEEE Transactions on Engineering Management**, [S.L.], v. 71, p. 10522-10533, 2024. Institute of Electrical and Electronics Engineers (IEEE). <http://dx.doi.org/10.1109/tem.2022.3195903>.

MAHESHWARI, Pratik; KAMBLE, Suchet. The Application of Supply Chain Digital Twin to Measure Optimal Inventory Policy. **IFAC-Papersonline**, [S.L.], v. 55, n. 10, p. 2324-2329, 2022. Elsevier BV. <http://dx.doi.org/10.1016/j.ifacol.2022.10.055>.

MAÏZI, Yasmina; BENDAVID, Ygal. Leveraging on the Digital Twin for improving retail store daily operations management. **The 18Th International Conference on Modelling and Applied Simulation**, [S.L.], p. 92-100, set. 2019. CAL-TEK SRL. <http://dx.doi.org/10.46354/i3m.2019.mas.013>.

MAÏZI, Yasmina; BENDAVID, Ygal. Building a digital twin for IoT smart stores: a case in retail and apparel industry. **International Journal of Simulation and Process Modelling**, [S.L.], v. 16, n. 2, p. 147, 2021. Inderscience Publishers. <http://dx.doi.org/10.1504/ijspm.2021.115868>.

MAO, Haixia; FAN, Xiaopeng; GUAN, Jinping; CHEN, Yeh-Cheng; SU, Haoran; SHI, Wenzhong; ZHAO, Yubin; WANG, Yang; XU, Chengzhong. Customer attractiveness evaluation and classification of urban commercial centers by crowd intelligence. **Computers In Human Behavior**, [S.L.], v. 100, p. 218-230, nov. 2019. Elsevier BV. <http://dx.doi.org/10.1016/j.chb.2018.08.019>.

MATSUDA, Michiko; NISHI, Tatsushi; HASEGAWA, Mao; MATSUMOTO, Sota. Virtualization of a supply chain from the manufacturing enterprise view using e-catalogues. **Procedia Cirp**, [S.L.], v. 81, p. 932-937, 2019. Elsevier BV. <http://dx.doi.org/10.1016/j.procir.2019.03.230>.

MATSUDA, Michiko; NISHI, Tatsushi; HASEGAWA, Mao; TERUNUMA, Takuto. Construction of a virtual supply chain using enterprise e-catalogues. **Procedia Cirp**, [S.L.], v. 93, p. 688-693, 2020. Elsevier BV. <http://dx.doi.org/10.1016/j.procir.2020.04.093>.

MATSUDA, Michiko; NISHI, Tatsushi; KAMIEBISU, Ryuichi; HASEGAWA, Mao; ALIZADEH, Roghayyeh; LIU, Ziang. Use of virtual supply chain constructed by cyber-physical systems concept. **Procedia Cirp**, [S.L.], v. 104, p. 351-356, 2021. Elsevier BV. <http://dx.doi.org/10.1016/j.procir.2021.11.059>.

MELEK, Ceren Gulra; SONMEZ, Elena Battini; AYRAL, Hakan; VARLI, Songul. Development of a Hybrid Method for Multi-Stage End-to-End Recognition of Grocery Products in Shelf Images. **Electronics**, [S.L.], v. 12, n. 17, p. 3640, 29 ago. 2023. MDPI AG. <http://dx.doi.org/10.3390/electronics12173640>.

MESQUITA, Renzo P.; MONTEVECHI, José Arnaldo B.; LEAL, Fabiano; Simulação Computacional Aplicada em um Problema de Logística Interna do Varejo. In: **ENEGEP – Encontro Nacional de Engenharia de Produção**, 42., 2022, Foz do Iguaçu – PR. [10.14488/ENEGEP2022_TN_ST_384_1900_43175](https://doi.org/10.14488/ENEGEP2022_TN_ST_384_1900_43175)

MESQUITA, Renzo P.; COSTA, Frâneli Eduarda R.; BALESTRASSI, Pedro Paulo; LEAL, Fabiano; Previsão do Índice de Vendas do Setor Varejista Brasileiro Utilizando os Métodos Holt-Winters e Arima: Um Estudo com Python. In: **ENEGEP – Encontro Nacional de Engenharia de Produção**, 43., 2023, Fortaleza - CE. [10.14488/ENEGEP2023_TN_ST_401_1974_45297](https://doi.org/10.14488/ENEGEP2023_TN_ST_401_1974_45297)

MESQUITA, Renzo P.; LEAL, Fabiano.; QUEIROZ, José Antônio de. Digital Twins in the Retail Industry: a systematic literature review. **International Journal Of Simulation Modelling**, [S.L.], v. 23, n. 3, p. 424-434, 31 ago. 2024. DAAAM International. <http://dx.doi.org/10.2507/ijssimm23-3-690>.

MESQUITA, Renzo P.; TEIXEIRA, Eduardo H.; RIBEIRO, Avner J. G.; GANDOLPHO, Bernardo D.; PONTES, Fábio Luiz F.. Computer Vision Applied to Smart Markets: a case study for empty shelf detection. **Anais do XXI Encontro Nacional de Inteligência Artificial e Computacional (Eniac 2024)**, [S.L.], p. 460-471, 17 nov. 2024. Sociedade Brasileira de Computação - SBC. <http://dx.doi.org/10.5753/eniac.2024.245034>.

MIGUEL, Paulo Augusto Cauchick; FLEURY, Afonso; MELLO, Carlos Henrique Pereira; NAKANO, Davi Noboru; LIMA, Edson Pinheiro de; TURRIONI, João Batista; HO, Linda Lee; MORABITO, Reinaldo; COSTA, Sérgio E. Gouvêa da.; MARTINS, Roberto Antônio; SOUSA, Rui; PUREZA, Vitória. **Metodologia de pesquisa em Engenharia de Produção e Gestão de operações**. 3. ed. Rio de Janeiro: Elsevier, 2018.

MONTEVECHI, Jose Arnaldo Barra; LEAL, Fabiano; PINHO, Alexandre Ferreira de; COSTA, Rafael Florencio da Silva; OLIVEIRA, Mona Liza Moura de; SILVA, Andre Luis Faustino da. Conceptual modeling in simulation projects by mean adapted IDEF: an

application in a brazilian tech company. **Proceedings of the 2010 Winter Simulation Conference**, [S.L.], p. 1624-1635, dez. 2010. IEEE.
<http://dx.doi.org/10.1109/wsc.2010.5678908>.

MOU, Shandong; ROBB, David J.; DEHORATIUS, Nicole. Retail store operations: literature review and research directions. **European Journal of Operational Research**, [S.L.], v. 265, n. 2, p. 399-422, mar. 2018. Elsevier BV. <http://dx.doi.org/10.1016/j.ejor.2017.07.003>.
MURTHY, Chinthakindi Balam; HASHMI, Mohammad Farukh; BOKDE, Neeraj Dhanraj; GEEM, Zong Woo. Investigations of Object Detection in Images/Videos Using Various Deep Learning Techniques and Embedded Platforms—A Comprehensive Review. **Applied Sciences**, [S.L.], v. 10, n. 9, p. 3280, 8 maio 2020. MDPI AG.
<http://dx.doi.org/10.3390/app10093280>.

MYKONIATIS, K.; SHIRZAEI, S.; KATSIGIANNIS, M.; PANAGOPOULOS, A.A.; DEB, S.; POTTER, T.; ANGELOPOULOU, A. Society 5.0: a simulation study of self checkout operations in a grocery store. **Proceedings of the 32Nd European Modeling & Simulation Symposium (EMSS 2020)**, [S.L.], p. 78-83, 2020. CAL-TEK srl.
<http://dx.doi.org/10.46354/i3m.2020.emss.011>.

NAGNATH, Yerule Sanket; KAO, Chih-Chung; SUN, Wei-Cheng; LIN, Chia-Hui; HSIEH, Chung-Wei. Realtime Customer Merchandise Engagement Detection and Customer Attribute Estimation with Edge Device. **2020 IEEE International Conference on Consumer Electronics - Taiwan (ICCE-Taiwan)**, [S.L.], p. 1-2, 28 set. 2020. IEEE.
<http://dx.doi.org/10.1109/icce-taiwan49838.2020.9258278>.

NANCE, Richard E.; SARGENT, Robert G. Perspectives on the Evolution of Simulation. **Operations Research**, [S.L.], v. 50, n. 1, p. 161-172, fev. 2002. Institute for Operations Research and the Management Sciences (INFORMS).
<http://dx.doi.org/10.1287/opre.50.1.161.17790>.

NISHI, Tatsushi; MATSUDA, Michiko; HASEGAWA, Mao; ALIZADEH, Roghayyeh; LIU, Ziang; TERUNUMA, Takuto. Automatic Construction of Virtual Supply Chain as Multi-Agent System Using Enterprise E-Catalogues. **International Journal of Automation Technology**, [S.L.], v. 14, n. 5, p. 713-722, 5 set. 2020. Fuji Technology Press Ltd..
<http://dx.doi.org/10.20965/ijat.2020.p0713>.

OLIVEIRA, Josenildo Brito; LIMA, Renato Silva; MONTEVECHI, José Arnaldo Barra. Perspectives and relationships in Supply Chain Simulation: a systematic literature review. **Simulation Modelling Practice and Theory**, [S.L.], v. 62, p. 166-191, mar. 2016. Elsevier BV. <http://dx.doi.org/10.1016/j.simpat.2016.02.001>.

OLIVEIRA, Pedro Guilherme F.; COSTA, Frâneli Eduarda R.; MESQUITA, Renzo P.; LEAL, Fabiano; Uma Proposta de Sistema Baseado em Visão Computacional para dar Suporte ao Controle de Inventários em Smart Markets. In: **ENEGEP – Encontro Nacional de Engenharia de Produção**, 44., 2024, Porto Alegre - RS.
[10.14488/ENEGEP2024_TN_ST_412_2021_47360](http://dx.doi.org/10.14488/ENEGEP2024_TN_ST_412_2021_47360)

ONWUDE, Daniel; BAHRAMI, Flora; SHRIVASTAVA, Chandrima; BERRY, Tarl; CRONJE, Paul; NORTH, Jade; KIRSTEN, Nicola; SCHUDEL, Seraina; CRENNNA, Eleonora; SHOJI, Kanaha. Physics-driven digital twins to quantify the impact of pre- and

postharvest variability on the end quality evolution of orange fruit. **Resources, Conservation and Recycling**, [S.L.], v. 186, p. 106585, nov. 2022. Elsevier BV. <http://dx.doi.org/10.1016/j.resconrec.2022.106585>.

PAL, Amitangshu; KANT, Krishna. Smart Sensing, Communication, and Control in Perishable Food Supply Chain. **ACM Transactions on Sensor Networks**, [S.L.], v. 16, n. 1, p. 1-41, 17 jan. 2020. Association for Computing Machinery (ACM). <http://dx.doi.org/10.1145/3360726>.

PAN, Jingmin; XU, Sukui; CHENG, Zhiyuan; LIAN, Shun. C2F-YOLO: a coarse-to-fine object detection framework based on yolo. **Proceedings Of The 2024 3Rd Asia Conference On Algorithms, Computing And Machine Learning**, [S.L.], p. 150-157, 22 mar. 2024. ACM. <http://dx.doi.org/10.1145/3654823.3654851>.

PARK, Sangchan; MALIPHOL, Sira; WOO, Jiyoun; FAN, Liu. Digital Twins in Industry 4.0. **Electronics**, [S.L.], v. 13, n. 12, p. 2258, 8 jun. 2024. MDPI AG. <http://dx.doi.org/10.3390/electronics13122258>

PETROPOULOS, Fotios; KOURENTZES, Nikolaos; NIKOLOPOULOS, Konstantinos; SIEMSEN, Enno. Judgmental selection of forecasting models. **Journal Of Operations Management**, [S.L.], v. 60, n. 1, p. 34-46, maio 2018. Wiley. <http://dx.doi.org/10.1016/j.jom.2018.05.005>.

PHAM, Long Hoang; TRAN, Duong Nguyen-Ngoc; NGUYEN, Huy-Hung; JEON, Hyung-Joon; TRAN, Tai Huu-Phuong; JEON, Hyung-Min; JEON, Jae Wook. Improving Deep Learning-based Automatic Checkout System Using Image Enhancement Techniques. **2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)**, [S.L.], p. 5333-5340, jun. 2023. IEEE. <http://dx.doi.org/10.1109/cvprw59228.2023.00562>.

POKUCIŃSKI, Sebastian; MROZEK, Dariusz. Object Detection with YOLOv5 in Indoor Equirectangular Panoramas. **Procedia Computer Science**, [S.L.], v. 225, p. 2420-2428, 2023. Elsevier BV. <http://dx.doi.org/10.1016/j.procs.2023.10.233>.

POUS, Rafael; CHINDEMI, Marcos; ALAJAMI, Abdussalam. Showing products in a retail store digital twin with item location captured by an RFID robot. **2023 IEEE 13Th International Conference on RFID Technology and Applications (RFID-TA)**, [S.L.], p. 189-192, 4 set. 2023. IEEE. <http://dx.doi.org/10.1109/rfid-ta58140.2023.10290509>.

QI, Qinglin; TAO, Fei. Digital Twin and Big Data Towards Smart Manufacturing and Industry 4.0: 360 degree comparison. **IEEE Access**, [S.L.], v. 6, p. 3585-3593, 2018. Institute of Electrical and Electronics Engineers (IEEE). <http://dx.doi.org/10.1109/access.2018.2793265>.

RABAH, Souad; ASSILA, Ahlem; KHOURI, Elio; MAIER, Florian; ABABSA, Fakreddine; BOURNY, Valéry; MAIER, Paul; MÉRIENNE, Frédéric. Towards improving the future of manufacturing through digital twin and augmented reality technologies. **Procedia Manufacturing**, [S.L.], v. 17, p. 460-467, 2018. Elsevier BV. <http://dx.doi.org/10.1016/j.promfg.2018.10.070>.

RACHANA HARISH, Arjun; LIU, X.L.; ZHONG, Ray Y.; HUANG, George Q.. Log-flock: a blockchain-enabled platform for digital asset valuation and risk assessment in e-commerce logistics financing. **Computers & Industrial Engineering**, [S.L.], v. 151, p. 107001, jan. 2021. Elsevier BV. <http://dx.doi.org/10.1016/j.cie.2020.107001>.

RAMALHO, Vinícius da Silva; SZEJKA, Anderson Luis; RUDEK, Marcelo; CANGIOLIERI JUNIOR, Osiris. Computer Vision with Cognitive Learning to Improve the Decision-Making During the Sales Process in Physical Stores. **IFIP Advances in Information and Communication Technology**, [S.L.], p. 260-269, 2019. Springer International Publishing. http://dx.doi.org/10.1007/978-3-030-42250-9_25.

RANE, Nitin. YOLO and Faster R-CNN object detection for smart Industry 4.0 and Industry 5.0: applications, challenges, and opportunities. **SSRN Electronic Journal**, [S.L.], p. 1-28, 2023. Elsevier BV. <http://dx.doi.org/10.2139/ssrn.4624206>.

REDMON, Joseph; DIVVALA, Santosh; GIRSHICK, Ross; FARHADI, Ali. You Only Look Once: unified, real-time object detection. **2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)**, [S.L.], p. 779-788, jun. 2016. IEEE. <http://dx.doi.org/10.1109/cvpr.2016.91>.

ROBOFLOW. **Auto Label**. 2024. Disponível em: <https://roboflow.com/auto-label>. Acesso em: 3 jun. 2025.

ROSSONI, Marco; FUMAGALLI, Andrea; COLOMBO, Giorgio. Machine Health State Recognition Through Images Classification with Neural Network for Condition-Based Maintenance. **Lecture Notes in Mechanical Engineering**, [S.L.], p. 432-443, 20 set. 2019. Springer International Publishing. http://dx.doi.org/10.1007/978-3-030-31154-4_37.

RZYDZIK, Sebastian; KROCZEK, Piotr. Time parameters verification of a numerical simulator of an automated store warehouse. **Bulletin of the Polish Academy of Sciences Technical Sciences**, [S.L.], p. 149816-149816, 18 mar. 2024. Polish Academy of Sciences Chancellery. <http://dx.doi.org/10.24425/bpasts.2024.149816>.

ŞAHINLI, Mehmet Arif. Potato Price Forecasting with Holt-Winters and ARIMA Methods: a case study. **American Journal Of Potato Research**, [S.L.], v. 97, n. 4, p. 336-346, 17 jul. 2020. Springer Science and Business Media LLC. <http://dx.doi.org/10.1007/s12230-020-09788-y>.

SANTOS, Carlos Henrique dos; MONTEVECHI, José Arnaldo Barra; QUEIROZ, José Antônio de; MIRANDA, Rafael de Carvalho; LEAL, Fabiano. Decision support in productive processes through DES and ABS in the Digital Twin era: a systematic literature review. **International Journal of Production Research**, [S.L.], v. 60, n. 8, p. 2662-2681, 13 mar. 2021. Informa UK Limited. <http://dx.doi.org/10.1080/00207543.2021.1898691>.

SANTOS, Carlos Henrique dos; QUEIROZ, José Antônio de; LEAL, Fabiano; MONTEVECHI, José Arnaldo Barra. Use of simulation in the industry 4.0 context: Creation of a Digital Twin to optimise decision making on non-automated process. **Journal Of Simulation**, [S.L.], v. 16, n. 3, p. 284-297, 6 set. 2020. Informa UK Limited. <http://dx.doi.org/10.1080/17477778.2020.1811172>.

SANTRA, Bikash; GHOSH, Udita; MUKHERJEE, Dipti Prasad. Graph-based modelling of superpixels for automatic identification of empty shelves in supermarkets. **Pattern Recognition**, [S.L.], v. 127, p. 108627, jul. 2022. Elsevier BV. <http://dx.doi.org/10.1016/j.patcog.2022.108627>.

SANTRA, Bikash; MUKHERJEE, Dipti Prasad. A comprehensive survey on computer vision based approaches for automatic identification of products in retail store. **Image and Vision Computing**, v. 86, p. 45-63, 2019. <https://doi.org/10.1016/j.imavis.2019.03.005>

SAPRE, Shubham; VARTAK, Aditya. Scientific Computing and Data Analysis using NumPy and Pandas. **International Research Journal of Engineering and Technology**, [S. l.], v. 7, n. 12, p. 1530–1535, dez. 2020. <http://irjet.net/archives/V7/i12/IRJET-V7I12236.pdf>

SASIKUMAR, A; RAVI, Logesh; DEVARAJAN, Malathi; VAIRAVASUNDARAM, Subramaniaswamy; KOTECHA, Ketan; HERENC SAR, Norbert. Sustainable Electronics: a blockchain-empowered digital twin-based governance system for consumer electronic products. **IEEE Transactions on Consumer Electronics**, [S.L.], v. 70, n. 2, p. 4968-4975, maio 2024. Institute of Electrical and Electronics Engineers (IEEE). <http://dx.doi.org/10.1109/tce.2024.3394512>.

SCHAFFER, Andrea L.; DOBBINS, Timothy A.; PEARSON, Sallie-Anne. Interrupted time series analysis using autoregressive integrated moving average (ARIMA) models: a guide for evaluating large-scale health interventions. **Bmc Medical Research Methodology**, [S.L.], v. 21, n. 1, p. 0-0, 22 mar. 2021. Springer Science and Business Media LLC. <http://dx.doi.org/10.1186/s12874-021-01235-8>.

SCHOPF, Tim; DRESSE, Kilian; MATTHES, Florian. Towards AI Platforms for Stationary Retail. **2022 5Th International Conference on Artificial Intelligence for Industries (Ai4I)**, [S.L.], p. 22-22, set. 2022. IEEE. <http://dx.doi.org/10.1109/ai4i54798.2022.00012>.

SCHUH, Günther; KELZENBERG, Christoph; WIESE, Jan; OCHEL, Tim. Data Structure of the Digital Shadow for Systematic Knowledge Management Systems in Single and Small Batch Production. **Procedia Cirp**, [S.L.], v. 84, p. 1094-1100, 2019. Elsevier BV. <http://dx.doi.org/10.1016/j.procir.2019.04.210>.

SCHWARTZ, Mathew; GENG, Yong; AGHA, Hakam; KIZHAKIDATHAZHATH, Rijeesh; LIU, Danqing; LENZINI, Gabriele; LAGERWALL, Jan P F. Linking physical objects to their digital twins via fiducial markers designed for invisibility to humans. **Multifunctional Materials**, [S.L.], v. 4, n. 2, p. 022002, 1 jun. 2021. IOP Publishing. <http://dx.doi.org/10.1088/2399-7532/ac0060>.

SENGUPTA, Sourav; DREYER, Heidi. Realizing zero-waste value chains through digital twin-driven S&OP: a case of grocery retail. **Computers In Industry**, [S.L.], v. 148, p. 103890, jun. 2023. Elsevier BV. <http://dx.doi.org/10.1016/j.compind.2023.103890>.

SHAMS, Abdullah; NUGROHO, Dimas Fajrian; HUMMEL, Maximilian; QUATTELBAUM, Bastian; BRECKENFELDER, Christof; JULIDAYANI, Suchi. 5GARderobe: a scalable real-time virtual wardrobe & e-commerce solution. **2024 International Conference on Artificial Intelligence, Computer, Data Sciences and**

Applications (ACDSA), [S.L.], p. 1-6, 1 fev. 2024. IEEE.
<http://dx.doi.org/10.1109/acdsa59508.2024.10467374>.

SHANKAR, Venkatesh; KALYANAM, Kirthi; SETIA, Pankaj; GOLMOHAMMADI, Alireza; TIRUNILLAI, Seshadri; DOUGLASS, Tom; HENNESSEY, John; BULL, J.s.; WADDOUPS, Rand. How Technology is Changing Retail. **Journal of Retailing**, [S.L.], v. 97, n. 1, p. 13-27, mar. 2021. Elsevier BV. <http://dx.doi.org/10.1016/j.jretai.2020.10.006>.

SHANMUGAMANI, Rajalingappaa. **Deep Learning for Computer Vision: Expert Techniques to Train Advanced Neural Networks Using Tensorflow and Keras**. Birmingham, UK: Packt Publishing, 2018. 310 p.

SHAO, Guodong; KIBIRA, Deogratias; FRECHETTE, Simon. Digital Twins for Advanced Manufacturing: the standardized approach. **Simulation Foundations, Methods And Applications**, [S.L.], p. 145-169, 2024. Springer Nature Switzerland.
http://dx.doi.org/10.1007/978-3-031-69107-2_7.

SHINKAR, Rshami S.; BOGIRI, Nagaraju. A Real Time Virtual Dressing Room Application using Opencv. **International Journal of Innovative Technology and Exploring Engineering**, [S.L.], v. 8, n. 11, p. 987-992, 30 set. 2019. Blue Eyes Intelligence Engineering and Sciences Engineering and Sciences Publication - BEIESP.
<http://dx.doi.org/10.35940/ijitee.h7396.0981119>.

SHOJI, Kanaha; SCHUDEL, Seraina; ONWUDE, Daniel; SHRIVASTAVA, Chandrima; DEFRAEYE, Thijs. Mapping the postharvest life of imported fruits from packhouse to retail stores using physics-based digital twins. **Resources, Conservation and Recycling**, [S.L.], v. 176, p. 105914, 2022a. Elsevier BV. <http://dx.doi.org/10.1016/j.resconrec.2021.105914>.

SHOJI, Kanaha; SCHUDEL, Seraina; SHRIVASTAVA, Chandrima; ONWUDE, Daniel; DEFRAEYE, Thijs. Optimizing the postharvest supply chain of imported fresh produce with physics-based digital twins. **Journal Of Food Engineering**, [S.L.], v. 329, p. 111077, 2022b. Elsevier BV. <http://dx.doi.org/10.1016/j.jfoodeng.2022.111077>.

SHOMAN, Maged; ABOAH, Armstrong; MOREHEAD, Alex; DUAN, Ye; DAUD, Abdulateef; ADU-GYAMFI, Yaw. A Region-Based Deep Learning Approach to Automated Retail Checkout. **2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)**, [S.L.], p. 3210-3215, jun. 2022. IEEE.
<http://dx.doi.org/10.1109/cvprw56347.2022.00362>.

SHRIVASTAVA, Chandrima; SCHUDEL, Seraina; SHOJI, Kanaha; ONWUDE, Daniel; SILVA, Fátima Pereira da; TURAN, Deniz; PAILLART, Maxence; DEFRAEYE, Thijs. Digital twins for selecting the optimal ventilated strawberry packaging based on the unique hygrothermal conditions of a shipment from farm to retailer. **Postharvest Biology and Technology**, [S.L.], v. 199, p. 112283, maio 2023. Elsevier BV.
<http://dx.doi.org/10.1016/j.postharvbio.2023.112283>.

SILVA, Aline Beatriz dos Santos; ARAÚJO, Ana Catarina de Melo; FRIAS, Paulo Germano de; VILELA, Mirella Bezerra Rodrigues Vilela; BONFIM, Cristine Vieira do. Modelo Autorregressivo Integrado de Médias Móveis (ARIMA): aspectos conceituais e metodológicos e sua aplicabilidade na mortalidade infantil. **Revista Brasileira de Saúde**

Materno Infantil, v. 21, p. 657-666, abr-jun 2021. <http://dx.doi.org/10.1590/1806-93042021000200016>

SINGH, Maulshree; SRIVASTAVA, Rupal; FUENMAYOR, Evert; KUTS, Vladimir; QIAO, Yuansong; MURRAY, Niall; DEVINE, Declan. Applications of Digital Twin across Industries: a review. **Applied Sciences**, [S.L.], v. 12, n. 11, p. 5727, 4 jun. 2022. MDPI AG. <http://dx.doi.org/10.3390/app12115727>.

SU, Shuo; NASSEHI, Aydin; HICKS, Ben; ROSS, Joel. Characterisation and evaluation of identity for digital twins for the manufacturing domain. **Journal Of Manufacturing Systems**, [S.L.], v. 71, p. 224-237, dez. 2023. Elsevier BV. <http://dx.doi.org/10.1016/j.jmsy.2023.09.004>.

SUBASI, Abdulhamit. **Practical Machine Learning for Data Analysis Using Python**. London: Academic Press, 2020. 521 p.

SULTANA, F.; SUFIAN, A.; DUTTA, P.. A Review of Object Detection Models Based on Convolutional Neural Network. **Advances in Intelligent Systems and Computing**, [S.L.], p. 1-16, 2020. Springer Singapore. http://dx.doi.org/10.1007/978-981-15-4288-6_1.

SUN, Cunwu; ZHENG, Meimei; DU, Shichang; WU, Kan; ZHOU, Hao; XIA, Tangbin; LIN, Jie. Ordering Problem with Known Capacity in Smart Factory. **2019 IEEE International Conference on Industrial Cyber Physical Systems (ICPS)**, [S.L.], p. 592-596, maio 2019. IEEE. <http://dx.doi.org/10.1109/icphys.2019.8780380>.

ŠUŠTER, Ivan; RANISAVLJEVIĆ, Tamara. Optimization of MySQL Database. **Journal of Process Management and New Technologies**, v. 11, n. 1–2, p. 141–151, 2023. <http://dx.doi.org/10.5937/jouproman2301141Q>

SZABÓ-SZENTGRÓTI, Eszter; RÁMHÁP, Szabolcs; KÉZAI, Petra Kinga. Systematic Review of Cashierless Stores (Just Walk Out Stores) Revolutionizing The Retail. **Management & Marketing**, [S.L.], v. 18, n. 1, p. 427-448, 1 dez. 2023. Walter de Gruyter GmbH. <http://dx.doi.org/10.2478/mmcks-2023-0023>

SZELISKI, Richard. **Computer Vision: algorithms and applications**. 2. ed. Washington, EUA: Springer, 2022. 947 p.

TAO, Fei; QI, Qinglin; WANG, Lihui; NEE, A.y.C.. Digital Twins and Cyber-Physical Systems toward Smart Manufacturing and Industry 4.0: correlation and comparison. **Engineering**, [S.L.], v. 5, n. 4, p. 653-661, ago. 2019. Elsevier BV. <http://dx.doi.org/10.1016/j.eng.2019.01.014>.

TERVEN, Juan; CORDOVA-ESPARZA, Diana-Margarita; ROMERO-GONZÁLEZ, Julio-Alejandro. A Comprehensive Review of YOLO Architectures in Computer Vision: from yolov1 to yolov8 and yolo-nas. **Machine Learning and Knowledge Extraction**, [S.L.], v. 5, n. 4, p. 1680-1716, 20 nov. 2023. MDPI AG. <http://dx.doi.org/10.3390/make5040083>.

TONIONI, Alessio; SERRA, Eugenio; STEFANO, Luigi di. A deep learning pipeline for product recognition on store shelves. **2018 IEEE International Conference on Image**

Processing, Applications and Systems (IPAS), [S.L.], p. 25-31, dez. 2018. IEEE.
<http://dx.doi.org/10.1109/ipas.2018.8708890>.

TRAPPEY, Amy J.C.; TRAPPEY, Charles; WANG, W.C.; HSIEH, H. I. Patent Analysis of Key Technologies for Smart Retailing and their Projected Economic Impact. **2018 IEEE 22Nd International Conference on Computer Supported Cooperative Work in Design ((CSCWD))**, [S.L.], p. 582-587, maio 2018. IEEE.
<http://dx.doi.org/10.1109/cscwd.2018.8465265>.

ULTRALYTICS. **Explore Ultralytics YOLOv8**. 2023. Disponível em:
<https://docs.ultralytics.com/pt/models/yolov8/>. Acesso em: 29 mar. 2025.

ULTRALYTICS. **Métricas de Desempenho do YOLO**. 2023b. Disponível em:
<https://docs.ultralytics.com/pt/guides/yolo-performance-metrics/>. Acesso em: 24 jul. 2025.

ULTRALYTICS. **Comparações de Modelos: Escolha o Melhor Modelo de Detecção de Objetos para o Seu Projeto**. 2024. Disponível em:
<https://docs.ultralytics.com/pt/compare/#yolov8-vs>. Acesso em: 29 mar. 2025.

ULTRALYTICS. **Train settings**. 2024b. Disponível em:
<https://docs.ultralytics.com/pt/usage/cfg/#train-settings>. Acesso em: 16 ago. 2025.

ULTRALYTICS. **Aumento de Dados usando Ultralytics YOLO**. 2025a. Disponível em:
<https://docs.ultralytics.com/pt/guides/yolo-data-augmentation/>. Acesso em: 14 jun. 2025.

ULTRALYTICS. **Transfer Learning**. 2025b. Disponível em:
<https://www.ultralytics.com/glossary/transfer-learning>. Acesso em: 16 ago. 2025.

VACHÁLEK, Jan; BARTALSKY, Lukas; ROVNY, Oliver; SISMISOVA, Dana; MORHAC, Martin; LOKSIK, Milan. The digital twin of an industrial production line within the industry 4.0 concept. **2017 21St International Conference on Process Control (PC)**, [S.L.], p. 258-262, jun. 2017. IEEE. <http://dx.doi.org/10.1109/pc.2017.7976223>.

VATS, Arpita; ANASTASIU, David C.. Enhancing Retail Checkout through Video Inpainting, YOLOv8 Detection, and DeepSort Tracking. **2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)**, [S.L.], p. 5529-5536, jun. 2023. IEEE. <http://dx.doi.org/10.1109/cvprw59228.2023.00585>.

VIJAYAKUMAR, D. Sudaroli. Digital twin in consumer choice modeling. **Advances In Computers**, [S.L.], p. 265-284, 2020. Elsevier.
<http://dx.doi.org/10.1016/bs.adcom.2019.09.010>.

WANG, Chien-Yao; LIAO, Hong-Yuan Mark; WU, Yueh-Hua; CHEN, Ping-Yang; HSIEH, Jun-Wei; YEH, I-Hau. CSPNet: a new backbone that can enhance learning capability of cnn. **2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)**, [S.L.], p. 1571-1580, jun. 2020. IEEE.
<http://dx.doi.org/10.1109/cvprw50498.2020.00203>.

WANG, Kung-Jeng; LEE, Ying-Hao; ANGELICA, Septianda. Digital twin design for real-time monitoring – a case study of die cutting machine. **International Journal of Production**

Research, [S.L.], v. 59, n. 21, p. 6471-6485, 14 set. 2020. Informa UK Limited.
<http://dx.doi.org/10.1080/00207543.2020.1817999>.

WENKEL, Simon; ALHAZMI, Khaled; LIIV, Tanel; ALRSHOUD, Saud; SIMON, Martin. Confidence Score: the forgotten dimension of object detection performance evaluation. **Sensors**, [S.L.], v. 21, n. 13, p. 4350, 25 jun. 2021. MDPI AG.
<http://dx.doi.org/10.3390/s21134350>.

WHITE, Jules; SCHMIDT, Douglas C.; GOLPARVAR-FARD, Mani. Applications of Augmented Reality [Scanning the Issue]. **Proceedings of the IEEE**, [S.L.], v. 102, n. 2, p. 120-123, fev. 2014. Institute of Electrical and Electronics Engineers (IEEE).
<http://dx.doi.org/10.1109/jproc.2013.2295873>.

WRIGHT, Louise; DAVIDSON, Stuart. How to tell the difference between a model and a digital twin. **Advanced Modeling and Simulation in Engineering Sciences**, [S.L.], v. 7, n. 1, p. 1, 11 mar. 2020. Springer Science and Business Media LLC.
<http://dx.doi.org/10.1186/s40323-020-00147-4>.

WU, Honghai; JI, Pengwei; MA, Huahong; XING, Ling. A Comprehensive Review of Digital Twin from the Perspective of Total Process: data, models, networks and applications. **Sensors**, [S.L.], v. 23, n. 19, p. 8306, 8 out. 2023. MDPI AG.
<http://dx.doi.org/10.3390/s23198306>

YENSABAI, Chavapol; NGOENTHAI, Waranyu; LEANGARUN, Teema; KOOLPIRUCK, Diew. Digital Retail Shop Services in Cyber-Physical Retail System: a case study of food business. **2023 Third International Symposium on Instrumentation, Control, Artificial Intelligence, and Robotics (ICA-SYMP)**, [S.L.], p. 61-64, 18 jan. 2023. IEEE.
<http://dx.doi.org/10.1109/ica-symp56348.2023.10044743>.

YUN, Heuijee; PARK, Daejin. Simulation of Self-driving System by implementing Digital Twin with GTA5. **2021 International Conference on Electronics, Information, and Communication (ICEIC)**, [S.L.], p. 1-2, 2021a. IEEE.
<http://dx.doi.org/10.1109/iceic51217.2021.9369807>.

YUN, Heuijee; PARK, Daejin. Virtualization of Self-Driving Algorithms by Interoperating Embedded Controllers on a Game Engine for a Digital Twinning Autonomous Vehicle. **Electronics**, [S.L.], v. 10, n. 17, p. 2102, 2021b. MDPI AG.
<http://dx.doi.org/10.3390/electronics10172102>.

ZHUANG, Fuzhen; QI, Zhiyuan; DUAN, Keyu; XI, Dongbo; ZHU, Yongchun; ZHU, Hengshu; XIONG, Hui; HE, Qing. A Comprehensive Survey on Transfer Learning. **Proceedings of the IEEE**, [S.L.], v. 109, n. 1, p. 43-76, jan. 2021. Institute of Electrical and Electronics Engineers (IEEE). <http://dx.doi.org/10.1109/jproc.2020.3004555>.

APÊNDICE A - Publicações

i. Periódicos

MESQUITA, Renzo P.; LEAL, Fabiano.; QUEIROZ, José Antônio de. Digital Twins in the Retail Industry: a systematic literature review. **International Journal Of Simulation Modelling**, [S.L.], v. 23, n. 3, p. 424-434, 31 ago. 2024. DAAAM International. <http://dx.doi.org/10.2507/ijssimm23-3-690>

ii. Congressos

MESQUITA, Renzo P.; MONTEVECHI, José Arnaldo B.; LEAL, Fabiano; Simulação Computacional Aplicada em um Problema de Logística Interna do Varejo. In: **ENEGEP – Encontro Nacional de Engenharia de Produção**, 42., 2022, Foz do Iguaçu – PR. https://dx.doi.org/10.14488/ENEGEP2022_TN_ST_384_1900_43175

MESQUITA, Renzo P.; COSTA, Frâneli Eduarda R.; BALESTRASSI, Pedro Paulo; LEAL, Fabiano; Previsão do Índice de Vendas do Setor Varejista Brasileiro Utilizando os Métodos Holt-Winters e Arima: Um Estudo com Python. In: **ENEGEP – Encontro Nacional de Engenharia de Produção**, 43., 2023, Fortaleza - CE. https://dx.doi.org/10.14488/ENEGEP2023_TN_ST_401_1974_45297

MARCONDES, Aline P.; BARBOZA, Gabriel Ilian F.; LUGLI, Alexandre B.; MESQUITA, Renzo P.; Leveraging the Prototyping of Digital Shadows – A Smart Shelf Case Study. In: **INDUSCON – IEEE/IAS International Conference on Industry Applications**, 15., 2023, São Bernardo do Campo - SP. <https://dx.doi.org/10.1109/INDUSCON58041.2023.10374929>

OLIVEIRA, Pedro Guilherme F.; COSTA, Frâneli Eduarda R.; MESQUITA, Renzo P.; LEAL, Fabiano; Uma Proposta de Sistema Baseado em Visão Computacional para dar Suporte ao Controle de Inventários em Smart Markets. In: **ENEGEP – Encontro Nacional de Engenharia de Produção**, 44., 2024, Porto Alegre - RS. https://dx.doi.org/10.14488/ENEGEP2024_TN_ST_412_2021_47360

MESQUITA, Renzo P.; TEIXEIRA, Eduardo H.; RIBEIRO, Avner J. G.; GANDOLPHO, Bernardo D.; PONTES, Fábio Luiz F.; Computer Vision Applied to Smart Markets: a Case Study for Empty Shelf Detection. In: **ENIAC – Encontro Nacional de Inteligência Artificial e Computacional**, 21., 2024, Belém - PA. <https://doi.org/10.5753/eniac.2024.245034>

MESQUITA, Renzo P.; PIZZONI, Henrique; TEIXEIRA, Eduardo H.; LEAL, Fabiano.; A Digital Shadow Proposal based on Computer Vision to Support Retail Shelf Management. In: **INDUSCON – IEEE/IAS International Conference on Industry Applications**, 2025, São Sebastião – SP. <https://dx.doi.org/10.1109/INDUSCON66435.2025.11241726>

APÊNDICE B – Manual de Adaptação e Treinamento de Novos Produtos no GD

1. OBJETIVO

O objetivo deste documento é apresentar um manual de adaptação do sistema de Visão Computacional (VC) baseado na biblioteca YOLOv8 para a detecção de novos produtos e espaços vazios em prateleiras.

2. PROCEDIMENTOS DE ADAPTAÇÃO

A fim de que novos produtos sejam inseridos e detectados pelo sistema de VC, é necessário seguir os passos que serão detalhados nos tópicos a seguir. Para fins práticos e conforme demonstrado na Figura 1, latas de Guaraná serão utilizadas para demonstrar como o treinamento do modelo de VC YOLOv8 deverá ser realizado.



Figura 1 – Novo produto a ser inserido: lata de Guaraná

2.1 PREPARAÇÃO DOS DADOS

Neste subtópico serão apresentados os passos necessários para se preparar os dados para a correta configuração do sistema.

2.1.1 Organização das Imagens e/ou Vídeos

Para que o modelo de VC possa detectar as classes (classes são objetos ou entidades que se deseja detectar), ele precisa ser treinado em um conjunto de imagens que contenham essas classes devidamente marcadas.

Para organizar as imagens, é indicado gravar vídeos ou tirar fotos com os produtos nas prateleiras, variando a disposição, a profundidade e o lado do produto que está aparecendo. É importante adicionar aleatoriedade e dispor o produto de diferentes maneiras na prateleira, respeitando o espaço entre um e outro e tentando se aproximar do que seria o ideal.

No caso dessa pesquisa, foi escolhido o método de gravação de vídeo, pois como o produto foi variado diversas vezes, um único vídeo foi gravado e depois foi realizado o recorte das imagens desejadas ao longo dele. Exemplos de algumas imagens utilizadas são demonstrados na Figura 2.



Figura 2 – Exemplo de imagens utilizadas no dataset de Guaraná.

Foi utilizado, para cada tipo de produto, 250 imagens devidamente anotadas (com as marcações de onde existe o produto) e todas foram salvas em uma mesma pasta. A Figura 2 demonstra a diversidade de imagens que compõem o *dataset*. O acervo abrange desde fotos de um único produto até cenários complexos, como prateleiras inteiras, na qual o item aparece junto a outros produtos, intercalado por espaços de diferentes tamanhos.

O conjunto de dados também contempla situações desafiadoras, como a oclusão (um produto parcialmente escondido) e alterações na disposição dos itens, além de vistas que capturam a gôndola por completo. Outras variações podem ser exploradas a depender do

contexto e das possibilidades que o produto pode ser organizado. É importante levar em consideração que todas as fotos têm a sua importância, mas é preciso priorizar ter mais fotos focadas em um único produto e com ele em sua organização ideal.

2.1.2 Anotação de Classes (Roboflow, Labellmg)

A partir do momento em que as 250 imagens foram capturadas, o próximo passo é realizar as anotações (*annotations*) dessas imagens. As *annotations* consistem em demarcar caixas que indicam onde estão os produtos e qual é o tipo de produto. Para isso, será usada a plataforma Roboflow²⁰, que é dinâmica e intuitiva.

2.1.2.1 Utilizando o Roboflow

O primeiro passo é acessar a plataforma Roboflow e realizar o cadastro de uma nova conta. Uma vez realizado o cadastro, o usuário será redirecionado para uma tela com o objetivo de criar a sua área de trabalho, como acontece na Figura 3. Os passos seguintes são:

1. Defina o nome no campo “*Name Your Workspace*”
2. Na opção “*Select Plan*” escolha o plano “*Public Plan*”, o qual não é pago.
3. Em seguida aperte o botão de continuar.

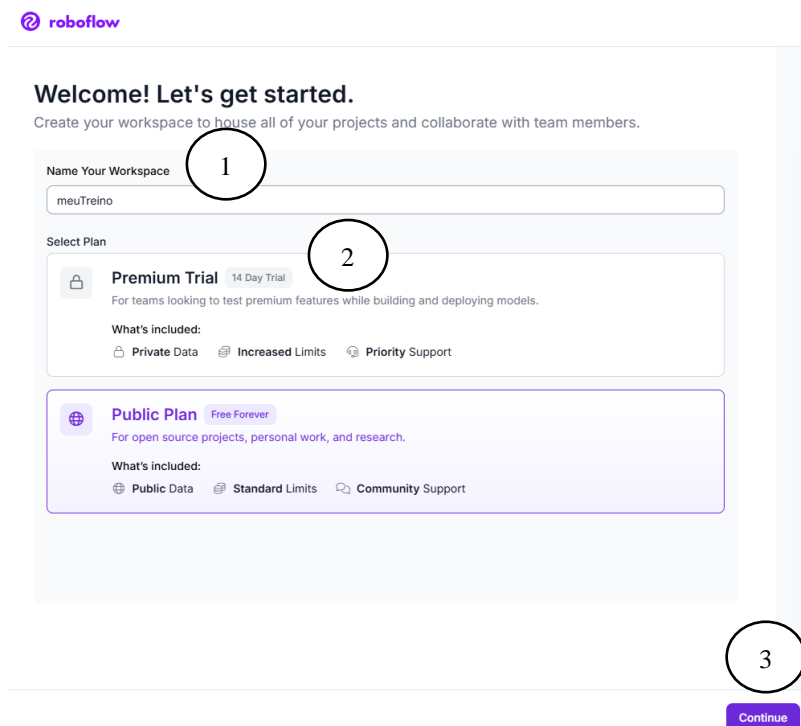


Figura 3 – Criando a área de trabalho no Roboflow

²⁰ Acesso ao Roboflow: <https://roboflow.com/>

Após clicar em continuar, a aba de “*Invite your team*” aparecerá. O usuário poderá seguir direto caso não haja mais pessoas que irão participar do treinamento do modelo. Caso tenha, o usuário poderá adicionar o *e-mail* delas para que se juntem ao mesmo *workspace* que está sendo criado. Feito isso, clique em “*Create workspace*”. Agora, nessa área de trabalho, o usuário poderá criar diferentes projetos e fazer o *upload* de suas imagens para treinar modelos de VC.

2.1.2.2 Criando o Projeto de VC

A partir da criação da área de trabalho, os projetos poderão ser adicionados seguindo os passos presentes na Figura 4:

1. Na aba à esquerda, clique em “*Projects*”
2. Após isso, no centro clique em “*New Project*”

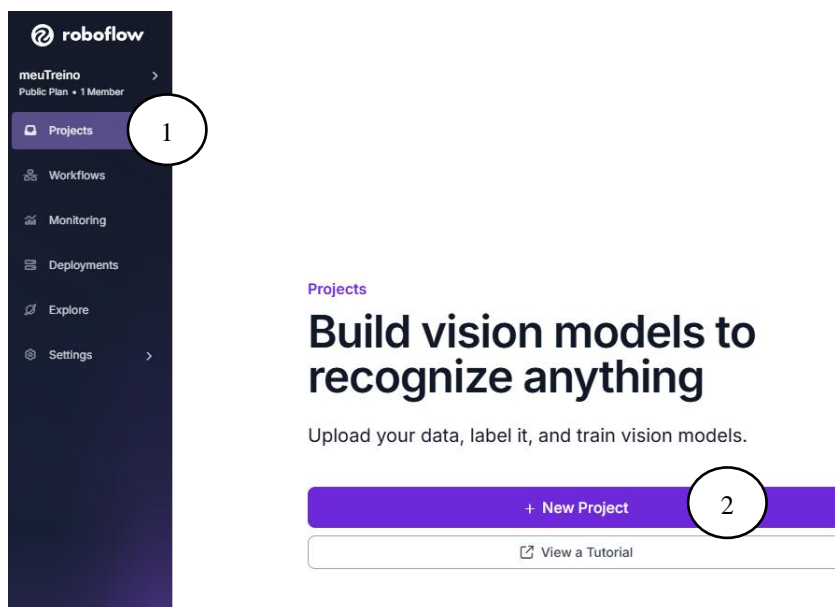


Figura 4 - Criando um projeto no Roboflow

De acordo com a Figura 5, no cabeçalho “*Let’s create your project*”, preencha com o nome do seu projeto e o tipo de objeto que você está detectando em “*Annotation Group*”. Deixe as demais opções como já estão no padrão do sistema (*Visibility* marcado em *Public* e *Project Type* para *Object Detection*). Para finalizar, clique em “*Create Public Project*”.

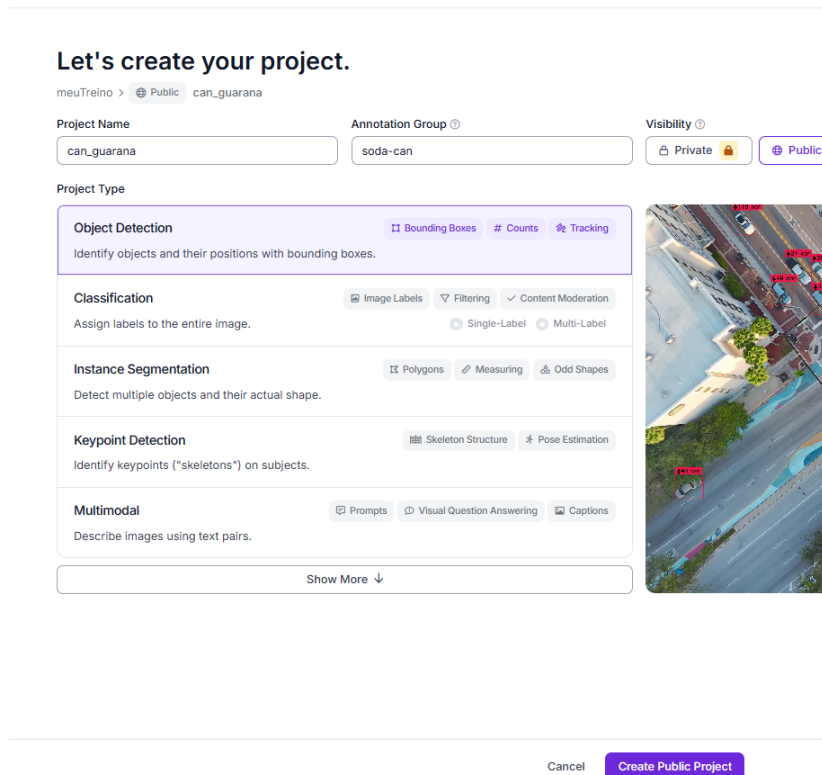


Figura 5 - Criando um projeto no Roboflow

Após realizar os passos anteriores, a aba de *upload* dos arquivos será aberta. Para enviar os arquivos, de acordo com a Figura 6, clique em “*Select Folder*” e selecione a pasta que contém todas as imagens preparadas anteriormente.

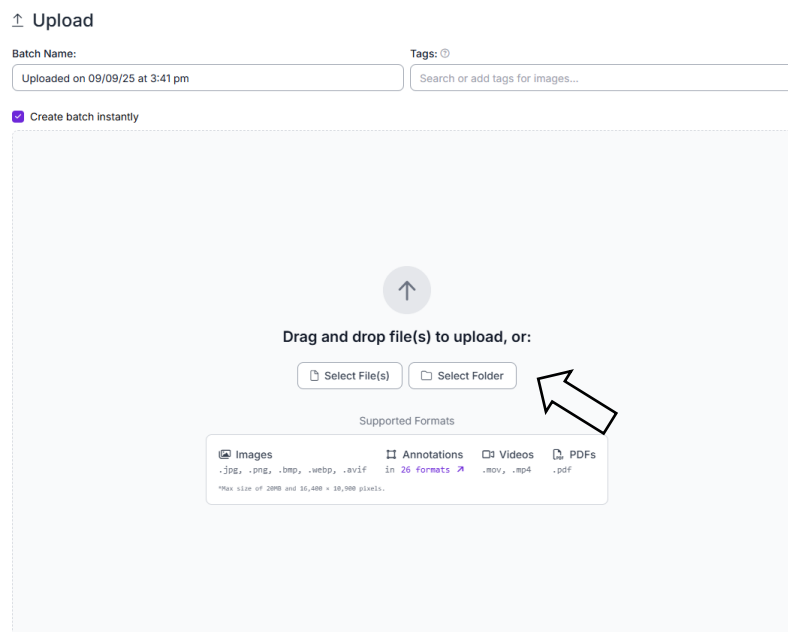


Figura 6 – Fazendo upload de imagens no Roboflow

Após o *upload* ser realizado, aparecerá uma tela com todas as imagens enviadas e no canto direito da tela terá as opções demonstradas na Figura 7. Com isso, começará o processo

de rotulagem das imagens, isso é, realizar as *annotations* que foram mencionadas anteriormente.

O usuário tem a escolha fazer tudo manualmente e demarcar sozinho utilizando a opção “*Label Myself*”. Porém, para maior praticidade do processo, o Roboflow disponibiliza o uso do modelo de VC Grounding-Dino²¹ para que ele já tente rotular o máximo de imagens para o usuário. Uma vez feito isso, o usuário só precisará revisar as imagens para verificar se estão conforme o esperado. Para seguir essa última etapa, a qual é consideravelmente indicada, clique em “*Auto-Label Entire Batch*”.

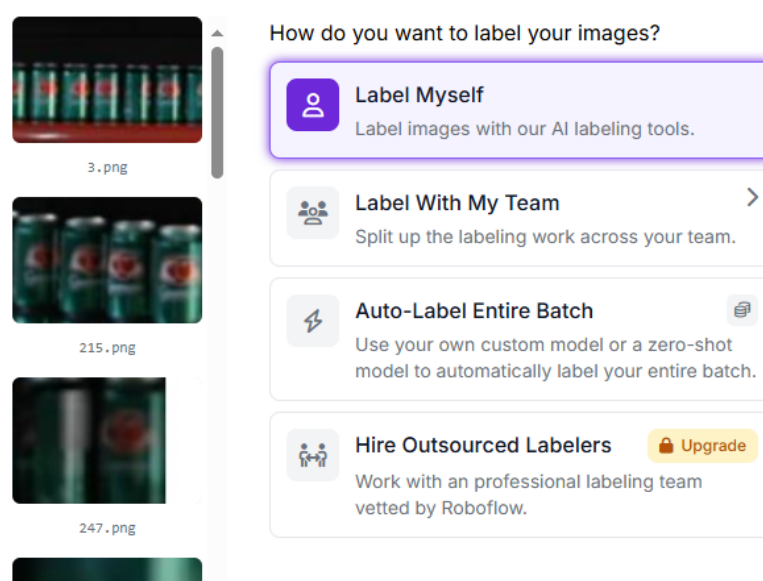


Figura 7 - Opções de rotulagem das imagens

Ao abrir a tela demonstrada na Figura 8, o usuário deverá seguir o tutorial indicado para obter os melhores resultados e entender como usar a ferramenta. Ao finalizar a etapa do tutorial, que é bem curta, o próximo passo é clicar “*Get Started*”. Feito isso, logo o usuário poderá dar início ao processo.

²¹ Disponível em: <https://roboflow.com/model/groundingdino>

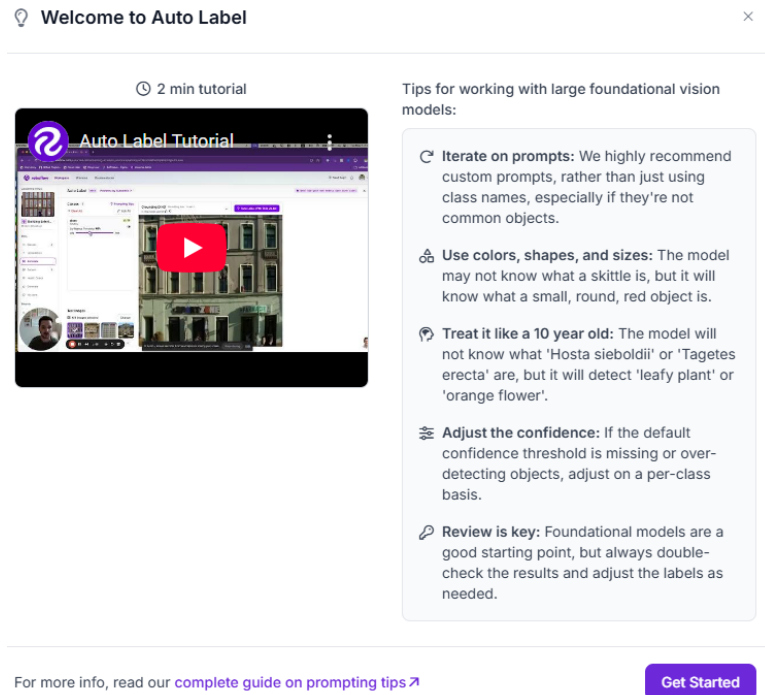


Figura 8 – Rotulação automática disponível no Roboflow

No canto esquerdo haverá a aba “Classes” (conforme demonstrado na Figura 9) e, dentro dela, o usuário deverá indicar:

1. O nome da classe. Nesse caso, “can_guarana”;
2. A descrição visual dela logo abaixo. Portanto, “green soda can” (lata de refrigerante verde). Isso permitirá ao Grounding-Dino tentar identificar um padrão nas imagens das latas, facilitando o processo de anotação.

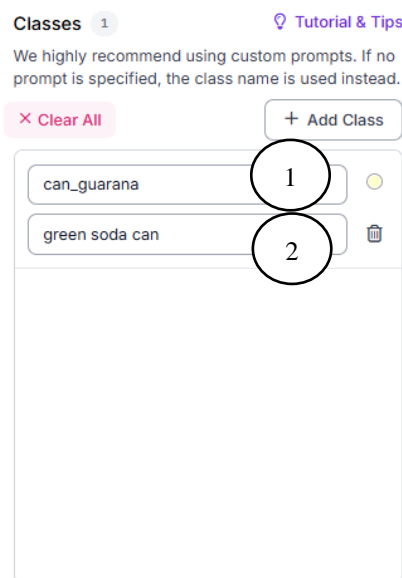


Figura 9 - Nomeação das classes

Após a nomeação da classe e descrição, será necessário clicar na opção “*Generate Test Results*”, conforme demonstrado na Figura 10.

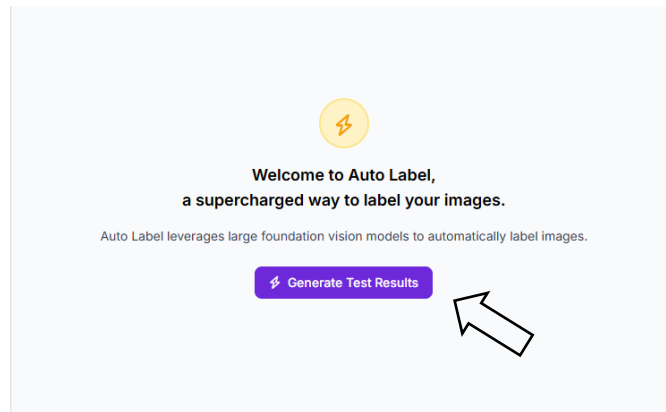


Figura 10 - Geração dos resultados do teste

Na próxima tela, conforme ilustra a Figura 11, o usuário deverá ajustar o parâmetro *Confidence Threshold* (Limiar de confiança) a fim de que o modelo possa buscar detectar o produto desejado. Para escolher o melhor valor, é recomendado que o usuário faça testes e veja qual situação será mais benéfica.

Na Figura 11 é demonstrado como, ao utilizado o limiar de confiança em 90%, o sistema não reconheceu o produto devidamente. Porém, ao abaixar o limiar de confiança para 80%, conforme apresentado na Figura 12, o produto foi reconhecido. Esse valor pode variar para cada imagem e dataset.

Após a definição do limiar de confiança mais adequado ao seu dataset, o usuário deverá clicar no botão “Auto Label With This Model” (Figura 12, canto superior direito).

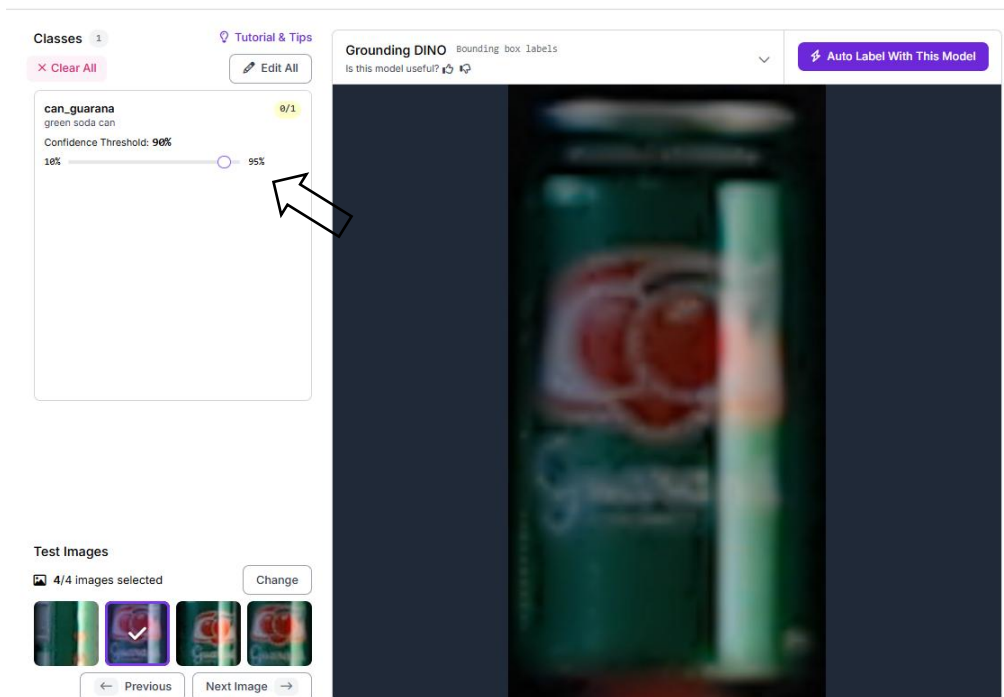


Figura 11 - Definição do limiar de confiança

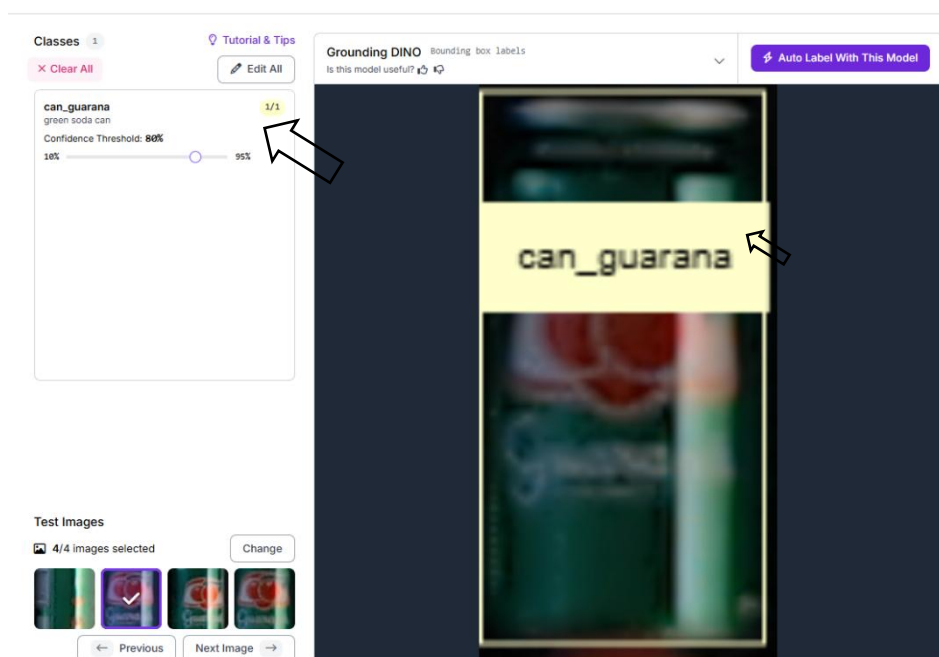


Figura 12 - Definição do limiar de confiança

Para realizar o treinamento do modelo de VC, será consumido créditos os quais são inclusos em cada plano disponível pelo Roboflow²². O usuário utilizando do plano gratuito possui um número menor de créditos, porém, caso seja necessário o uso de mais créditos, o usuário poderá assinar um plano extra para esse fim. Após a verificação dos créditos, o usuário

²² Comparação dos planos disponíveis no Roboflow: <https://roboflow.com/pricing>

deverá clicar em “*Start Auto Label*”. Nesse momento, será necessário aguardar até que o processo seja concluído.

2.1.2.3 Processo de Revisão

Após iniciar o Auto Label, o usuário deverá mudar da aba de “Annotating” para “Review”. Assim que estiver na aba “Review”, clique nela para observar o dataset já com as anotações. Em seguida, clique em uma das fotos e vá passando uma por uma para realizar o processo de revisão. Antes disso, o Roboflow disponibiliza um rápido tutorial alertando o usuário sobre boas práticas de como anotar imagens não detectadas manualmente, conforme ilustra a Figura 13.

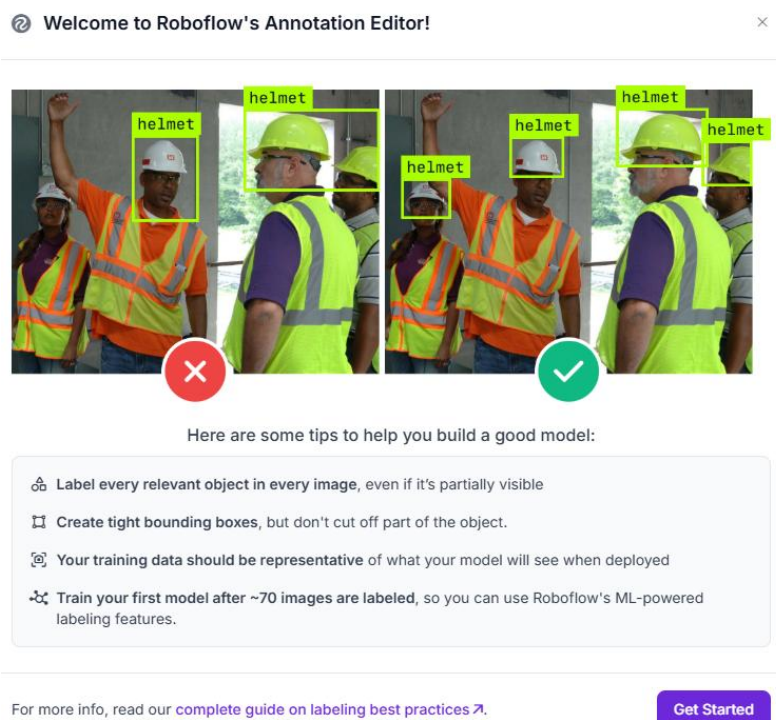


Figura 13 - Imagem tutorial do Roboflow na etapa de anotações

Na execução do projeto de reconhecimento das latas, é possível observar em uma das imagens (representada na Figura 14) que o modelo não anotou nenhum produto automaticamente, portanto, será necessário fazer o procedimento manualmente. Ao invés de clicar em “Reject”, o usuário pode marcar as latas de guaraná e, então depois, apertar em “Approve”.

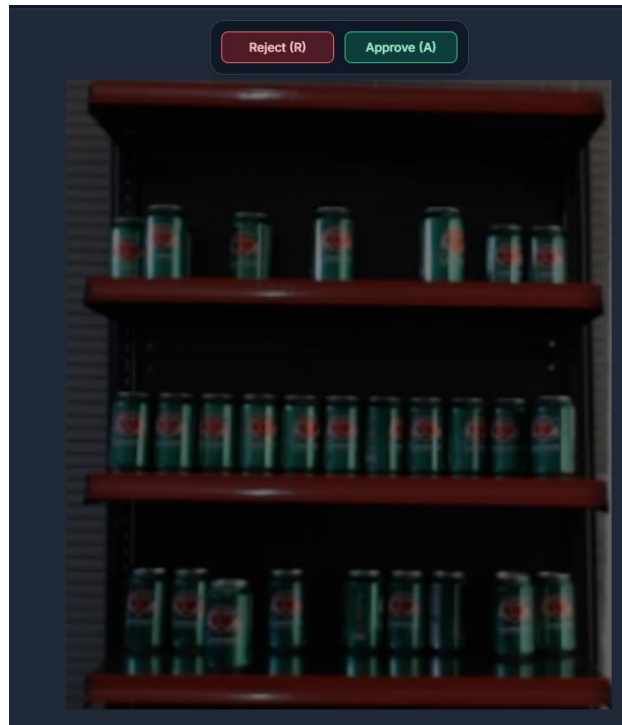


Figura 14 - Não reconhecimento automático das latas pelo Roboflow

Para marcar as fotos, é necessário apertar a tecla “B” no teclado ou clicar diretamente na ferramenta bounding box tool (barra lateral direita, conforme Figura 15). Após a seleção da ferramenta bounding box tool, o usuário poderá realizar as marcações. Um exemplo de como a imagem deverá ficar pós o procedimento das marcações é demonstrada na Figura 16.

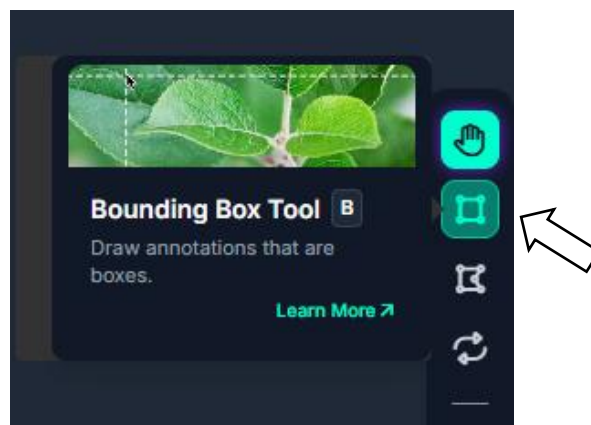


Figura 15 - Opção *Bounding Box Tool*



Figura 16 - Marcação manual utilizando a ferramenta *bounding box tool*

Note agora que todas as latas foram devidamente marcadas e respeitando seus limites. Quando o usuário for realizar as marcações, é preciso que as anotações sejam feitas de maneira bem rente ao produto. É possível notar que haverá produtos que serão bem demarcados, mas outros não, como foi o caso apresentado na Figura 16.

Caso seja necessário refazer ou só ajustar alguma anotação, o usuário poderá excluir a anotação e fazer outra com a ferramenta *bounding box tool*, ou apenas ajustar a caixa de marcação utilizando a ferramenta “*Drag Tool*” (Tecla D), conforme apresentado na Figura 17.

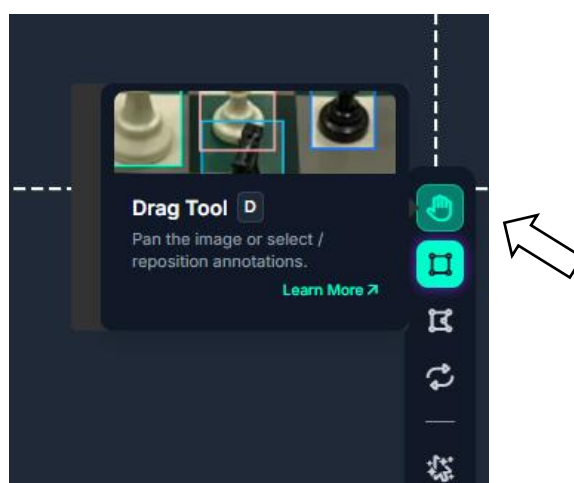


Figura 17 - Opção *Drag Tool*

Após a aprovação de todas as imagens que foram revisadas, o usuário deverá clicar no botão “*Add approved to dataset*” o qual é demonstrado na Figura 18 e, dessa forma, adicionar as imagens anotadas ao *dataset*.

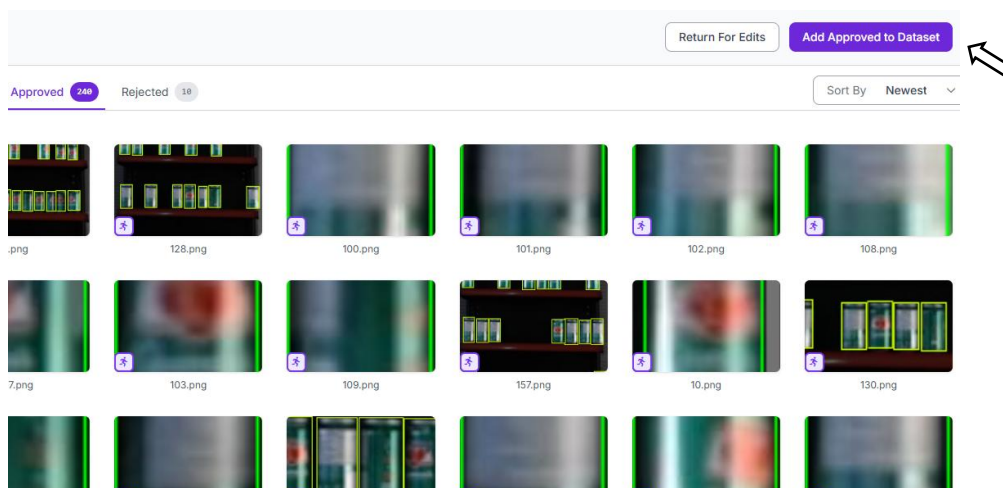


Figura 18 – Adicionando as imagens ao *dataset*

Quando o usuário clicar na opção, verá que terá dentro do campo “*Method*” a possibilidade de definir como suas imagens serão utilizadas. Para treinar o modelo de YOLOv8, é necessário imagens para treino e imagens para validação. As imagens de treino são utilizadas para o modelo ser treinado e aprender como deverá fazer a detecção, enquanto as imagens de validação vão servir como uma correção para o modelo sempre estar olhando para uma imagem e conferir se está fazendo a detecção correta.

Também há a possibilidade do uso de imagens de teste, as quais usam o modelo para conferir métricas de desempenho após o treinamento, usando como base o *dataset* composto por imagens não conhecidas do modelo.

Portanto, dentro do “*Method*”, conforme a Figura 19, o usuário deverá:

1. Selecionar a opção “*Split Images Between Train/Valid/Test*” e manter a distribuição de 70% pra Treino, 20% pra Validação e 10% pra Teste;
2. Clicar em “*Add 250 Images*”;

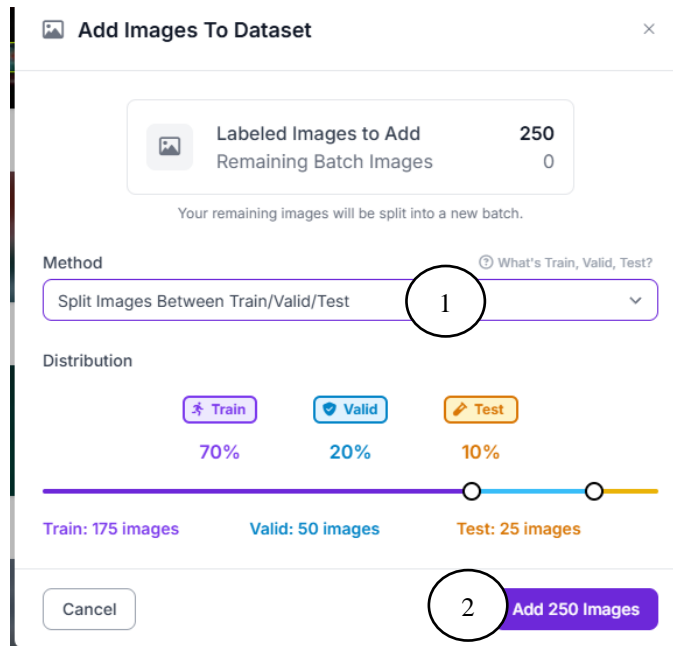


Figura 19 - Adicionando as imagens ao *dataset*

2.1.2.4 Criando a versão para fazer *download*

Depois de finalizada a etapa de revisão, na aba esquerda, o usuário deverá clicar em “*Versions*” para terminar de montar a versão e prosseguir com o *download*. Esse *download* será necessário para gerar o arquivo que será enviado posteriormente ao YOLOv8 para treinamento. Para isso, conforme ilustra a Figura 20, os passos a seguir deverão ser executados:

1. Dar um nome a versão do *dataset*;
2. Na aba de *Preprocessing* adicionar as opções:
 - a. “*Auto-Orient*”;
 - b. “*Resize*” com “*Stretch to*” de 640x640 (resolução padrão utilizada pelo YOLOv8);
3. Clicar em “*Continue*”.

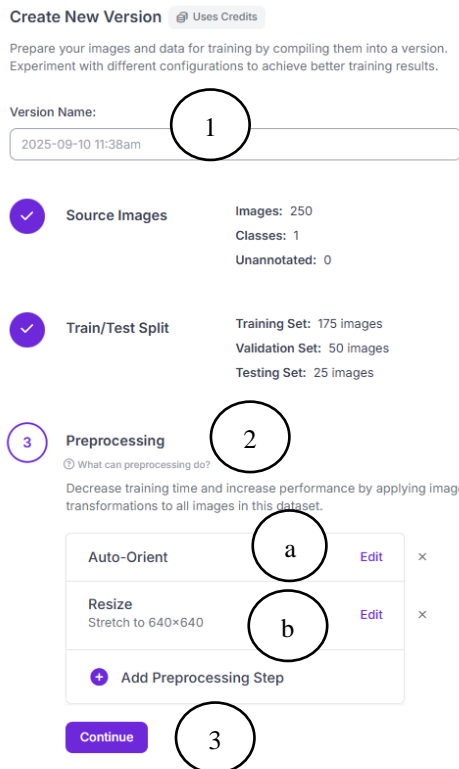


Figura 20 - Criando uma nova versão

Após isso, a aba “*Augmentation*” aparecerá. O *Data Augmentation* serve para adicionar variação às imagens a fim de que o modelo possa funcionar bem em ambientes com variação de iluminação e ruído visual, como é o caso do cenário aqui demonstrado. Nessa aba, é sugerido que o usuário coloque ao menos as seguintes configurações (vide Figura 21):

1. Em “*Brightness*” deixar com uma variação de 20% (tanto positiva, quanto negativa);
2. Em “*Noise*”, a variação deverá ser de 1.65%;
3. Clicar em “*Continue*”.

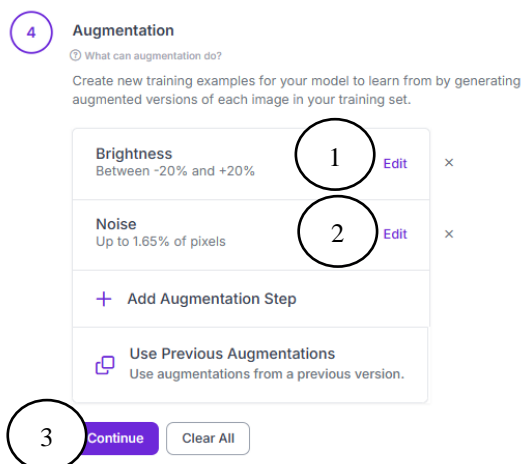


Figura 21 - Configurações da tela “*Augmentation*”

O último passo antes da criação da versão é definir o tamanho máximo da versão final. Para essa etapa, de acordo com a Figura 22, é sugerido que o usuário configure a opção “*Maximum Version Size*” para ao menos “2x” e clicar em “*Create*”.

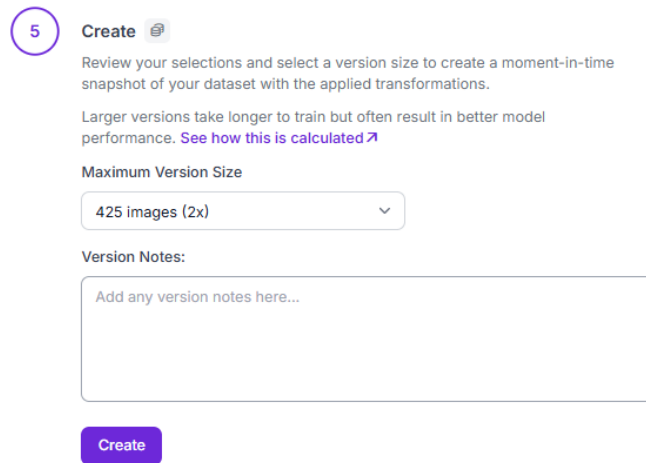


Figura 22 - Configuração do “*Maximum Version Size*”

Após estipular as métricas demonstradas nos passos anteriores, o usuário estará em uma tela, na qual em seu canto superior direito tem a opção “*Download Dataset*”. Após clicar nessa opção, a tela evidenciada na Figura 23 aparecerá, assim o usuário deverá:

1. Selecionar YOLOv8 no campo “*Image and Annotation Format*”;
2. Marcar a opção “*Download Zip to Computer*” no campo “*Download Options*”;
3. Clicar em “*Continue*”;

O *download* iniciará e, assim que finalizado, o usuário estará com o *dataset* do produto de seu interesse devidamente organizado e configurado.

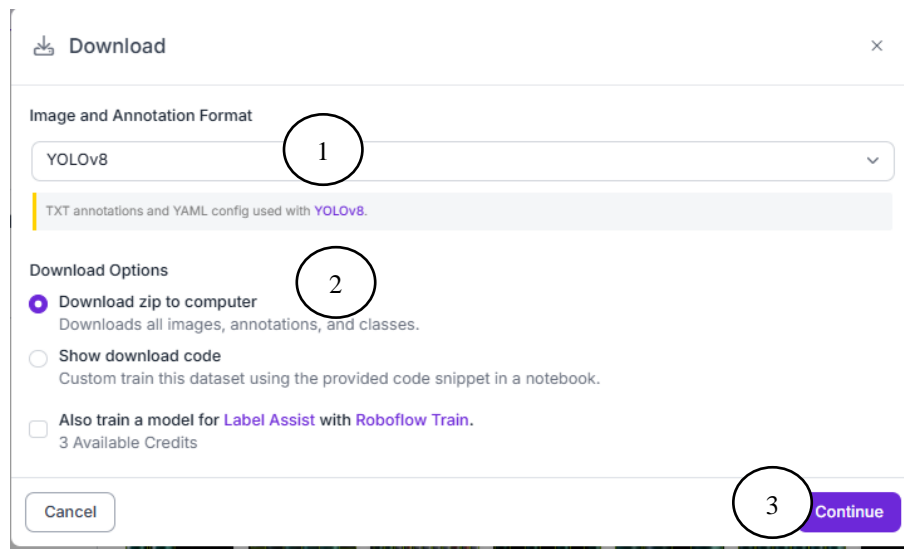


Figura 23 - Download do *dataset*

2.1.3 Treinamento do Modelo

Ao longo deste subcapítulo, serão expostos os próximos passos após a criação do novo *dataset* para que ele possa ser incluso no *dataset* já existente e treinado no YOLOv8.

2.1.3.1 Mesclando os *Datasets*

A partir do momento que o *dataset* foi preparado e baixado, em seguida ele deverá ser mesclado ao original para que seja capaz de detectar os produtos já existentes no modelo. Na Figura 24 está apresentada como a divisão dos diretórios contendo o conjunto de novas imagens e o modelo original deverão estar.

```

meuDataset/
├── test/
│   ├── images
│   └── labels
├── train/
│   ├── images
│   └── labels
├── valid/
│   ├── images
│   └── labels
└── data.yaml

```

Figura 24 – Divisão dos diretórios para mesclar os datasets

Com a divisão de diretórios feita, o usuário agora possui a pasta de testes, treino e validação, além de um arquivo com dados para o modelo chamado de `data.yaml`. Dentro de cada uma das três pastas mencionadas, há uma pasta chamada “*images*” e outra “*labels*”. As *images* são propriamente as imagens do *dataset*, enquanto as *labels* representam um arquivo de texto de cada uma das imagens, contendo a representação em coordenadas de todas as anotações (*bounding boxes*) que aquela imagem possui e que foram realizadas pelo Roboflow.

Cada arquivo `.txt` possui dentro dele uma ou várias linhas, e cada linha representa uma *bounding box*. Para essa linha, temos os seguintes dados, os quais são apresentados em sequência:

- id da classe (objeto);
- x e y do centro (geralmente informado em pixels);
- Largura e altura (geralmente informado em pixels).

Os ID’s das classes podem ser muitos, tudo de acordo com o número de classes. Todavia, eles devem seguir um padrão de sequência, iniciando do zero e acrescentando de 1 em 1 por cada classe. No *dataset* criado neste manual, por exemplo, como só temos uma classe, todo arquivo `.txt` começa com o número 0, indicando que a primeira e única classe é a classe 0. Se houvesse mais produtos, como no caso do *dataset* com latas de guaraná, garrafas de fanta e caixas de leite, teríamos as classes 0, 1 e 2.

Se os dois *datasets* forem combinados sem antes fazer nenhum tratamento, é possível notar que existirá a classe 0 representando tanto a lata de guaraná quanto outro produto no outro *dataset*. Para resolver isso, é preciso então trocar o número da classe que representa a lata de guaraná para não colidir com um número de classe que já está sendo utilizado no *dataset* original.

Para fazer esse ajuste, é preciso conferir os dois arquivos `data.yaml` (o usuário precisará abrir utilizando de algum editor de texto) e realizar os procedimentos que serão descritos. Ao abrir, o usuário notará a presença da estrutura apresentada na Figura 25.

```
1  train: ../train/images
2  val: ../valid/images
3  test: ../test/images
4
5  nc: 1
6  names: ['can_guarana']
```

Figura 25 – Estrutura exemplo do arquivo `data.yaml`

O arquivo indica os caminhos para os modelos onde estão cada um dos diretórios, o número de classes (*nc*), e quais são os nomes das classes em ordem (*names*). No *dataset* original há uma estrutura similar, mas que se difere no número de classes e nos nomes, conforme ilustra a Figura 26.

```
train: ../train/images
val: ../valid/images
test: ../test/images

nc: 3
names: ['bottle_fanta', 'box_milk', 'can_coke']
```

Figura 26 - Estrutura do arquivo data.yaml com mais de uma classe

Nesse caso, temos que *bottle_fanta* possui o ID 0, *box_milk* o ID 1 e *can_coke* o ID 2. Se o usuário quiser adicionar mais um produto, deverá então mudar o campo *nc* somando mais 1, e acrescentar o nome da classe que deseja. Um exemplo prático é demonstrado na Figura 27.

```
train: ../train/images
val: ../valid/images
test: ../test/images

nc: 4
names: ['bottle_fanta', 'box_milk', 'can_coke', 'can_guarana']
```

Figura 27 - Adição de mais uma classe no arquivo data.yaml

Como o *nc* foi alterado para 4, significa que no *dataset* há 4 classes, e adicionou-se a *name* *can_guarana*, o usuário deverá salvar o arquivo e fechar. Agora, por dedução, entendemos que *can_guarana* deverá ter o ID 3 dentro desse *dataset*. Para isso, a partir do *dataset* criado, o usuário terá que trocar todas as *labels* *txt* para que, ao invés de iniciar com 0, iniciem com o novo id que o *can_guarana* irá assumir, que nesse caso, é o ID 3.

Um ponto de atenção antes de executar a próxima atividade: antes de fazer as alterações no arquivo *.txt*, é recomendado que o usuário crie uma cópia do *dataset* já criado, pois se o código estiver errado e alterar as *labels* e as imagens de forma incorreta, será necessário baixar o *dataset* novamente no Roboflow.

Por fins de praticidade, para que o usuário não precise fazer isso indo de arquivo em arquivo, ele pode criar um *script* em Python que troque as classes das *labels* txt de um diretório de arquivos para YOLOv8 de forma automatizada.

É importante informar no comando que o arquivo contém os diretórios *train*, *valid* e *test* e, que dentro de cada diretório, há imagens e *labels*. Indique que a classe é o primeiro valor que é informado dentro do começo de cada linha do arquivo .txt, e que é para trocar do valor atual para o valor que é desejado (nesse caso, de 0 para 3).

Após realizar a edição do arquivo .txt, a última etapa para montar o *dataset* final será o usuário ter que passar as imagens e *labels* do *dataset* gerado no Roboflow e editado no Python para o novo *dataset* que contém os demais produtos. Para isso, será necessário abrir cada pasta (*train*, *test* e *valid*) e suas subpastas (imagens e *labels*), uma por vez, e transferir os arquivos para o *dataset* final de mesmo caminho.

Um exemplo do último passo será apresentado na sequência: O usuário abre o *dataset*, vai em *train*, abre a pasta de imagens, seleciona todas as imagens e cópia. Agora vai até o *dataset* final, vai em *train*, abre a pasta de imagens e cola todas as minhas imagens. O usuário deverá fazer o mesmo passo para *labels*, selecionado todas as *labels* e colando no *dataset* final na mesma pasta. Isso deverá ser realizado em cada pasta e subpasta, gerando um total de 6 repetições:

- *Train* – imagens e *labels*.
- *Valid* – imagens e *labels*.
- *Test* – imagens e *labels*.

2.2 CONFIGURAÇÃO DO MODELO E COMANDO DE TREINAMENTO

2.2.1 Treinamento

Uma vez que o *dataset* esteja pronto, o próximo passo é partir para o treinamento do modelo de VC. Para isso, o usuário deverá ter instalado em sua máquina o Python versão 3.8 ou superior. A partir disso, abra um arquivo .py e comece instalando a biblioteca Ultralytics. Para isso, dentro da IDE ou no terminal, será necessário executar o comando “pip install Ultralytics”. Assim que instalado, as próximas etapas consistem em escrever o código.

Além disso, é importante se certificar de estar com o modelo de VC original dos produtos individuais baixado (arquivo .pt) e no mesmo diretório de seus arquivos, pois será necessário usar esse modelo pré treinado para treinar o novo modelo (O arquivo best.pt consiste

no modelo que já foi treinado no sistema original). A organização do novo diretório deverá seguir a estrutura apresentada na Figura 28.

```

train/
├─ images/
└─ labels/
valid/
├─ images/
└─ labels /
test/
├─ images/
└─ labels/
data.yaml
best.pt

```

Figura 28 - Estrutura do diretório para treinamento do modelo

Para o treinamento, o usuário irá utilizar o comando *from Ultralytics import YOLO* (comando de importação da biblioteca). Logo após, é preciso passar a variável *model=YOLO('best.pt')*, responsável por carregar o modelo. E, por último, utilizar o comando *model.train()*, responsável pelo treinamento.

O método *train()* é responsável por treinar o modelo, e para que seja realizado da melhor forma, será necessário preencher o código com os parâmetros apresentados na Tabela 1 e na Figura 29. Tendo passado os parâmetros, o usuário poderá rodar o programa em Python para começar o seu treinamento.

Tabela 1 - Parâmetros para o método train()

| Parâmetro | Informações do Parâmetro | Descrição |
|---------------|--------------------------|---|
| <i>data</i> | data.yaml | Aponta para o arquivo de configuração do <i>dataset</i> , que contém os caminhos para as imagens de treino/validação e as informações sobre as classes. |
| <i>batch</i> | -1 | Define o número de imagens processadas simultaneamente em cada passo. O valor -1 ativa o " <i>auto-batch</i> ", que ajusta automaticamente o maior tamanho de lote possível com base na memória da GPU. |
| <i>epochs</i> | 3000 | Indica o número total de vezes que o algoritmo de treinamento irá percorrer todo o conjunto de dados para realizar o treinamento |

| Parâmetro | Informações do Parâmetro | Descrição |
|--------------------|--------------------------|--|
| (conclusão) | | |
| <i>device</i> | 0 | Especifica o dispositivo para o treinamento. 0 geralmente se refere à primeira GPU disponível, enquanto o texto “cpu” seria usado para treinar na CPU. |
| <i>patience</i> | 40 | É o número de épocas que o treinamento continuará sem registrar melhora nas métricas de validação antes de parar automaticamente (mecanismo de <i>early stopping</i>). |
| <i>imgsz</i> | 640 | Define a resolução (em pixels) para a qual todas as imagens serão redimensionadas antes de serem enviadas para o modelo. |
| <i>plots</i> | <i>True</i> | Se <i>True</i> , gera e salva gráficos de visualização ao final do treinamento, como curvas de perda e matriz de confusão. |
| <i>save</i> | <i>True</i> | Se <i>True</i> , salva os <i>checkpoints</i> do modelo (os pesos) durante e ao final do treinamento, evitando recomeçar do início caso algum problema aconteça no meio do treinamento. |

```

from ultralytics import YOLO

modelo = YOLO("best.pt")

results = modelo.train(data="C:\Desktop\meuDataset\data.yaml",
                       epochs = 3000,
                       imgsz = 640,
                       device = 0,
                       batch = -1,
                       plots = True,
                       patience = 40,
                       save = True,
                       )

```

Figura 29 - Parâmetros para o método train()

2.2.2 Métricas do Treinamento

Na Figura 30 estão disponíveis as métricas do treinamento realizado no tópico 2.2.1. É possível interpretar que:

1. Após 165 épocas, o modelo foi devidamente treinado atingindo 0.748 horas de treinamento;
2. Não executou todas as 3.000 épocas, pois o uso do *patience* atuou após perceber que não houve melhoras tendo passada 40 épocas desde a melhor época.

- a. O usuário poderá ajustar o *patience* conforme a necessidade, pois pode ser que o seu modelo precise de mais tempo para atingir as melhores épocas.

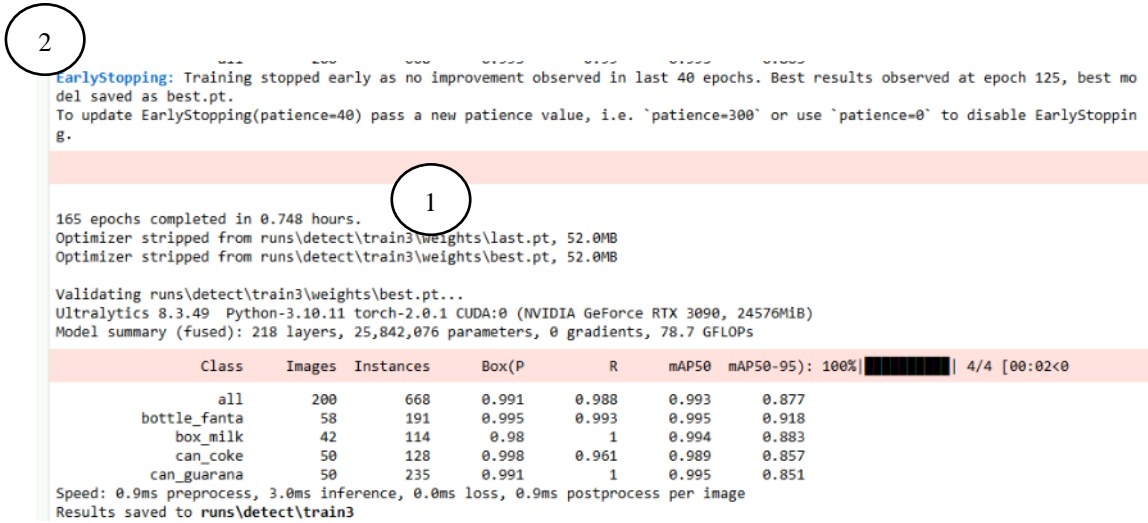


Figura 30 - Métricas do treinamento

As demais métricas do treinamento são demonstradas na Figura 31, considerando a precisão atingida, mAP50-95, recall, dentre outros.

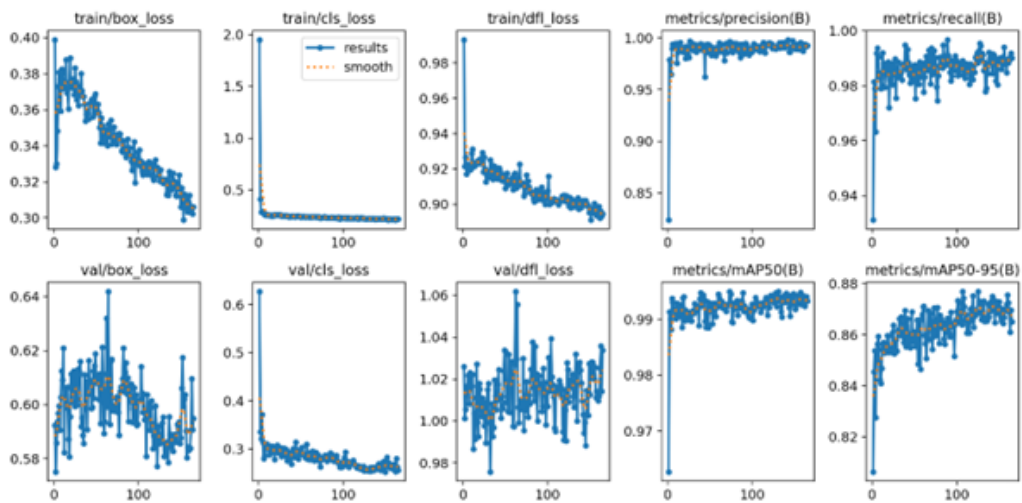


Figura 31 - Métricas do treinamento

2.3 VALIDAÇÃO

Depois de finalizado o treinamento, a validação consistirá em rodar o código do sistema, mas agora levando em consideração a inserção do novo produto. Portanto, para isso, é necessária a manutenção mínima no código. É necessário realizar a troca do modelo antigo pelo novo modelo.

2.3.1 Procedimentos para Testar a Acurácia em Novos Cenários e Testes Mínimos Recomendados

Para testes do novo modelo, inicialmente serão utilizados testes visuais empíricos (logo, utilizando um cenário físico). Em seguida, será feito o uso do método de validação do YOLOv8 para se visualizar métricas. Neste último, o teste irá passar sobre um conjunto de imagens nunca vistas antes pelo modelo, o qual é o conjunto de teste, a fim de avaliar sua acurácia em fotos do produto nunca vistas antes.

2.3.1.1 VALIDAÇÃO EMPÍRICA

Após rodar o algoritmo de VC com uma confiabilidade de 70% (conforme demonstrado na Figura 32), o resultado apresentado na Figura 33 será obtido.

```
from ultralytics import YOLO

model = YOLO('best.pt')

model.predict(source=0, imgsiz=640, conf=0.7, show=True)
```

Figura 32 - Nível de Confiabilidade

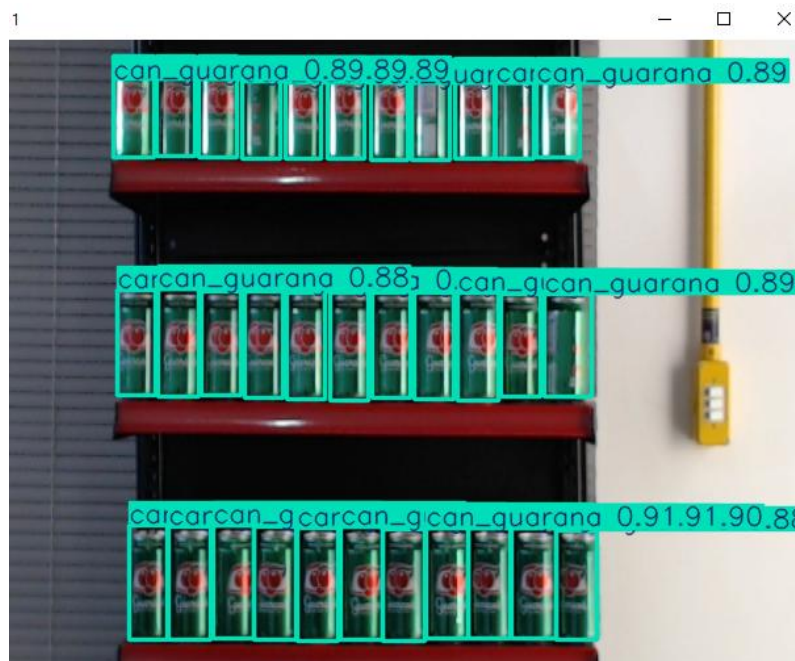


Figura 33 - Detecção no teste com confiabilidade 0.7

Visualmente é possível validar que o modelo está operando da forma esperada e detectando tudo que precisava ser detectado. Para mais validações, o usuário pode utilizar diferentes disposições desse produto, inclusive misturando com as outras classes de produtos.

2.3.1.2 VALIDAÇÃO ATRAVÉS DE MÉTRICAS

Para obter dados ainda mais técnicos, o usuário pode utilizar o método `val()` do YOLOv8. Para utilizá-lo, o código disponível na Figura 34 poderá ser aplicado.

```
from ultralytics import YOLO

modelo = YOLO("best.pt")

modelo.val(data="C:\Desktop\meuDataset\data.yaml")
```

Figura 34 - Aplicação do método `val()`

A grande diferença agora é que, ao invés de utilizar o caminho da pasta `valid` dentro do `val` do `.yaml`, o sistema usará o caminho da pasta de `test` (a qual foi criada no `dataset` anteriormente), já que o objetivo é obter métricas sobre imagens nunca vistas pelo modelo, o que garante que o modelo possa generalizar bem sobre novas situações. Para isso, é necessário realizar a alteração disponível na Figura 35 dentro do arquivo `data.yaml`.

```
train: ../train/images
val: ../test/images
test: ../valid/images
```

Figura 35 - Ajuste no arquivo `data.yaml`

Após realizar o ajuste e executar o código novamente, os resultados presentes na Figura 36 serão obtidos.

| Class | Images | Instances | Box(P) | R | mAP50 | mAP50-95): 100% | 7/7 [00:03<0 |
|--------------|--------|-----------|--------|-------|-------|-----------------|--------------|
| all | 100 | 312 | 0.99 | 0.989 | 0.995 | 0.874 | |
| bottle_fanta | 32 | 79 | 0.998 | 1 | 0.995 | 0.886 | |
| box_milk | 18 | 39 | 0.974 | 0.978 | 0.994 | 0.901 | |
| can_coke | 25 | 114 | 1 | 0.98 | 0.995 | 0.802 | |
| can_guarana | 25 | 80 | 0.987 | 1 | 0.995 | 0.906 | |

Figura 36 - Resultados de validação do modelo

A análise dos resultados de validação do modelo aponta para uma performance geral de alta confiabilidade. Com uma precisão de 0.99 e *recall* de 0.989, o sistema demonstra um equilíbrio notável entre a minimização de falsos positivos e falsos negativos.

A excelência é reforçada pelo mAP50 de 0.995, e a robustez da localização das caixas delimitadoras é confirmada pelo sólido mAP50-95 de 0.874, indicando um desempenho consistente mesmo sob critérios de avaliação rigorosos.

Especificamente para a classe "can_guarana", os resultados são notáveis. O modelo alcançou um *recall* perfeito de 1.0, o que significa que foi capaz de identificar 100% das instâncias de "can_guarana" presentes nas imagens de validação, não deixando nenhum objeto dessa classe passar despercebido.

Além disso, a precisão de 0.987 mostra que as detecções foram corretas na grande maioria dos casos. Notavelmente, o mAP50-95 para esta classe foi de 0.906, o mais alto entre todas as categorias, sugerindo que as predições de caixas delimitadoras para "can_guarana" são as mais precisas e bem ajustadas do modelo. Maiores detalhes do significado de cada uma das métricas podem ser visualizadas no tópico 4.4.1 desta tese.

2.4 MODELO GENERALISTA

Todo o processo realizado nas etapas anteriores foi para o treinamento e personalização do modelo especialista, o qual é capaz de identificar produtos específicos, desde que sejam garrafas, latas ou caixas tetrapak. Porém, novos formatos de produtos podem ser adicionados no modelo generalista (Ex: pacote) seguindo as mesmas etapas descritas no treinamento do modelo especialista, porém, levando em consideração 2 classes: *empty_space* (para espaços vazios) e *product* (para formato de produtos de forma geral).

Em suma, é repetir as etapas aqui descritas no manual, mas agora utilizando de modelo pré-treinado o modelo generalista. Lembrando que novas fotos de espaços vazios também podem ser adicionadas para buscar melhorar a detecção deles ou adicionar novos padrões de espaços vazios. Após treinado o modelo, se faz por necessário a troca do arquivo .pt do modelo generalista no código fonte *main_backend* do projeto *gemeodigital_backend* disponível no seguinte link: https://github.com/renzokuken31/Projeto_GD_VC.git

APÊNDICE C – Organização do Banco de Dados Relacional do GD Proposto

Este apêndice apresenta detalhes técnicos referentes às tabelas do BD Relacional para dar suporte ao armazenamento de dados produzidos pelo GD proposto. O objetivo é apresentar suas finalidades e documentar a organização interna dos seus dados a fim de garantir transparência, reprodutibilidade e possibilidade de adaptações futuras.

Tabela **product**

Finalidade: registra das básicos dos produtos que estão sendo monitorados dentro da gôndola

| Coluna | Tipo de Dado | Descrição |
|--------------|--------------|--|
| ID | INT | Identificador único de um produto |
| Product_name | VARCHAR(45) | Nome do produto |
| Location | VARCHAR(45) | Localizacao do produto dentro do estabelecimento |

Tabela **empty_spaces**

Finalidade: registra todas as ocorrências de espaços vazios detectados pelo *Back-end* de Reconhecimento

| Coluna | Tipo de Dado | Descrição |
|---------------------------|--------------|---|
| ID | INT | Identificador único de um espaço vazio |
| individual_emptying_index | INT | Índice de esvaziamento para o respectivo espaço vazio |
| shelf | INT | Número da prateleira ao qual o espaço vazio pertence |
| x1,y1,x2,y2 | FLOAT | Coordenadas da caixa delimitadora do respectivo espaço vazio |
| datas | DATETIME | Data e hora que ocorreu a inserção dos dados do espaço vazio no BD |
| _exists | TINYINT | <i>Flag</i> que identifica se a caixa delimitadora do respectivo espaço vazio ainda está sendo detectada ou não |
| id_product | INT | Identificador do produto a qual pertence o espaço vazio |

Tabela **consumption**

Finalidade: responsável por armazenar o índice de esvaziamento já calculado para cada produto que está sendo monitorado

| Coluna | Tipo de Dado | Descrição |
|----------------|--------------|--|
| ID | INT | Identificador único de um índice de esvaziamento |
| capture_time | DATETIME | Data e hora do cálculo do respectivo índice |
| emptying_index | INT | Valor do índice de esvaziamento de cada produto |

| | | |
|--------------|-------|--|
| local_stock | INT | Estoque máximo do produto na gondola |
| availability | FLOAT | Estimativa da porcentagem do respectivo produto sendo monitorado |
| id_product | INT | Identificador do produto a qual pertence o espaço vazio |

Tabela **consumption_forecasting**

Finalidade: armazenar a previsão do índice de esvaziamento e do estoque estimado futuro dos produtos

| Coluna | Tipo de Dado | Descrição |
|---------------------|--------------|--|
| ID | INT | Identificador único da previsão do índice de esvaziamento futuro de um produto |
| time_forecasting | DATETIME | Data e hora que foi armazenada a previsão calculada |
| emptying_index | INT | Valor do índice de esvaziamento future previsto para um determinado produto |
| future_availability | FLOAT | Qual a estimativa da porcentagem futura do respectivo produto sendo monitorado |
| id_product | INT | Identificador do produto a qual pertence a previsão |

Tabela **settings**

Finalidade: armazena as preferências do usuário dentro do *dashboard*

| Coluna | Tipo de Dado | Descrição |
|--------------------|--------------|--|
| ID | INT | Identificador de uma configuração |
| bar_chart | TINYINT | Determina se o painel “Visão Geral – Índices de Esvaziamento” será exibido (1) ou não (0) |
| minimum_stock | FLOAT | Estoque mínimo (em %) que os produtos podem atingir sem gerar alertas |
| table_index | TINYINT | Determina se o painel “Ranking – Índice de Esvaziamento” será exibido (1) ou não (0) |
| timeseries | TINYINT | Determina se o painel “Acompanhamento Individual – Índice de Esvaziamento” será exibido (1) ou não (0) |
| forecast_table | TINYINT | Determina se o painel “Previsão Padrão – Índices de Esvaziamento” será exibido (1) ou não (0) |
| observation_window | INT | Valor da janela de observação para o painel “Previsão Padrão – Índices de Esvaziamento” |
| forecast_window | INT | Valor da janela de previsão para o painel “Previsão Padrão – Índices de Esvaziamento” |
| detected_person | TINYINT | Determina se uma pessoa foi detectada (1) ou não (0) na frente da gondola monitorada |