



DESENVOLVIMENTO DE ALGORITMOS DE IA PARA DISPOSITIVOS  
VESTÍVEIS UTILIZANDO COMPUTAÇÃO DE BORDA

Jonathan Cristovão Ferreira da Silva

Orientadores: Ricardo Augusto Rabelo Oliveira  
Mateus Coelho Silva

Ouro Preto  
Julho de 2023

DESENVOLVIMENTO DE ALGORITMOS DE IA PARA DISPOSITIVOS  
VESTÍVEIS UTILIZANDO COMPUTAÇÃO DE BORDA

Jonathan Cristovão Ferreira da Silva

Dissertação de Mestrado apresentada ao Programa de Pós-graduação em Ciência da Computação, da Universidade Federal de Ouro Preto, como parte dos requisitos necessários à obtenção do título de Mestre em Ciência da Computação.

Orientadores: Ricardo Augusto Rabelo  
Oliveira  
Mateus Coelho Silva

Ouro Preto  
Julho de 2023

## SISBIN - SISTEMA DE BIBLIOTECAS E INFORMAÇÃO

S586d Silva, Jonathan Cristovão Ferreira da.  
Desenvolvimento de algoritmos de IA para dispositivos vestíveis  
utilizando computação de borda. [manuscrito] / Jonathan Cristovão  
Ferreira da Silva. Jonathan Cristovão Ferreira da Silva. - 2023.  
95 f.: il.: color., gráf., tab..

Orientador: Prof. Dr. Ricardo Augusto Rabelo Oliveira.

Coorientador: Me. Mateus Coelho Silva.

Dissertação (Mestrado Acadêmico). Universidade Federal de Ouro  
Preto. Departamento de Computação. Programa de Pós-Graduação em  
Ciência da Computação.

Área de Concentração: Ciência da Computação.

1. Inteligência Artificial. 2. Agricultura. 3. Laranja. I. Silva, Jonathan  
Cristovão Ferreira da. II. Oliveira, Ricardo Augusto Rabelo. III. Silva,  
Mateus Coelho. IV. Universidade Federal de Ouro Preto. V. Título.

CDU 004

Bibliotecário(a) Responsável: Luciana De Oliveira - SIAPE: 1.937.800



MINISTÉRIO DA EDUCAÇÃO  
UNIVERSIDADE FEDERAL DE OURO PRETO  
REITORIA  
INSTITUTO DE CIÊNCIAS EXATAS E BIOLÓGICAS  
DEPARTAMENTO DE COMPUTAÇÃO  
PROGRAMA DE PÓS-GRADUAÇÃO EM CIÊNCIA DA  
COMPUTAÇÃO



## FOLHA DE APROVAÇÃO

**Jonathan Cristovão Ferreira da Silva**

Desenvolvimento de algoritmos de IA para dispositivos vestíveis utilizando computação de borda

Dissertação apresentada ao Programa de Pós-Graduação em Ciência da Computação da Universidade Federal de Ouro Preto como requisito parcial para obtenção do título de Mestre em em Ciência da Computação

Aprovada em 15 de junho de 2023

### Membros da banca

Prof. Dr. Ricardo Augusto Rabelo Oliveira - Orientador - Universidade Federal de Ouro Preto  
Prof. Dr. Saul Emanuel Delabrida Silva - Universidade Federal de Ouro Preto  
Prof. Dr. José Augusto Nacif - Universidade Federal de Viçosa  
Dr. Vicente José Peixoto de Amorim - Dell Technologies

Prof. Dr. Ricardo Augusto Rabelo Oliveira, orientador do trabalho, aprovou a versão final e autorizou seu depósito no Repositório Institucional da UFOP em 29/06/2023



Documento assinado eletronicamente por **Ricardo Augusto Rabelo Oliveira, PROFESSOR DE MAGISTERIO SUPERIOR**, em 01/07/2023, às 12:39, conforme horário oficial de Brasília, com fundamento no art. 6º, § 1º, do [Decreto nº 8.539, de 8 de outubro de 2015](#).



A autenticidade deste documento pode ser conferida no site [http://sei.ufop.br/sei/controlador\\_externo.php?acao=documento\\_conferir&id\\_orgao\\_acesso\\_externo=0](http://sei.ufop.br/sei/controlador_externo.php?acao=documento_conferir&id_orgao_acesso_externo=0), informando o código verificador **0550032** e o código CRC **326BE85E**.

# Agradecimentos

Encontrar motivação nas pessoas ao nosso redor é um poderoso impulso na busca constante pelo conhecimento. Sou grato por ter exemplos inspiradores na minha vida, especialmente minha mãe, Maria das Neves. Sua dedicação e persistência são uma fonte inesgotável de inspiração. Ao meu pai, João Santana, por sempre me incentivar a seguir em frente. Além disso, tenho a sorte de contar com o amor e o carinho da minha companheira, Juliana. Ela está ao meu lado, me ouvindo sempre que preciso desabafar, e comemorando comigo todas as minhas conquistas pessoais e acadêmicas. Aos meus sogros, Carlos e Tânia, por todo o apoio que têm me dado ao longo do tempo. Sua presença e suporte são inestimáveis, e sou grato por tê-los em minha vida. Ao meu cunhado, Paulo, por sua amizade sincera e pelos ensinamentos que compartilha comigo. Além disso, sou extremamente grato aos meus amigos Dionízio e Mauricio. A amizade é um tesouro valioso que valorizo profundamente. É através das conexões significativas com pessoas tão especiais que encontramos inspiração, crescimento e força para alcançar nossos objetivos. Vocês são os pilares da minha trajetória e a razão do meu sucesso. Amo vocês!

Não posso deixar de mencionar a importância dos meus orientadores e professores nessa jornada de conhecimento. Ricardo Rabelo, Mateus Coelho e Vicente Amorim, agradeço por compartilharem seus valiosos ensinamentos comigo, que têm sido fundamentais para o meu crescimento acadêmico e profissional. Ao laboratório iMobilis, que tem me fornecido suporte essencial ao longo dessa jornada. Sua infraestrutura e equipe têm contribuído significativamente para o meu desenvolvimento e aprendizado.

Gostaria de agradecer à FAPEMIG, CAPES, CNPq e UFOP pelo fomento ao projeto de pesquisa apresentado. O presente trabalho foi realizado com apoio da Coordenação de Aperfeiçoamento de Pessoal de Nível Superior - Brasil (CAPES) - Código de Financiamento 001.

Resumo da Dissertação apresentada à UFOP como parte dos requisitos necessários para a obtenção do grau de Mestre em Ciências (M.Sc.)

## DESENVOLVIMENTO DE ALGORITMOS DE IA PARA DISPOSITIVOS VESTÍVEIS UTILIZANDO COMPUTAÇÃO DE BORDA

Jonathan Cristovão Ferreira da Silva

Julho/2023

Orientadores: Ricardo Augusto Rabelo Oliveira  
Mateus Coelho Silva

Programa: Ciência da Computação

Os dispositivos vestíveis estão cada vez mais presentes em nossas vidas. Além disso, os algoritmos de inteligência artificial vêm se tornando essenciais para compor estes dispositivos. Como os dispositivos vestíveis são restritos de recursos, tecnologias que exigem grande capacidade computacional podem ser inviáveis para aplicações neste contexto, principalmente quando se trata da computação de borda. Visto isso, o trabalho propõe o desenvolvimento de algoritmos de inteligência artificial para integração nestes dispositivos com o processamento dos dados na borda, sem utilizar recursos em nuvem. Esta proposta é validada com base em dois estudos de casos. O primeiro estudo de caso é a aplicação de técnicas de *Machine Learning* e *Deep Learning* na agricultura, com o objetivo de desenvolver um capacete inteligente para realizar inspeção de doenças em laranjas. No segundo estudo de caso é desenvolvida uma nova solução vestível para o reconhecimento de atividade de caminhada. Com o auxílio de três algoritmos de IA, este estudo de caso apresentou novas perspectivas para autoavaliação do usuário a partir dos dados coletados na atividade realizada. Dessa maneira, esse trabalho apresenta uma análise de aspectos do desenvolvimento de algoritmos de IA para integração em dois dispositivos vestíveis através da computação de borda.

Abstract of Dissertation presented to UFOP as a partial fulfillment of the requirements for the degree of Master of Science (M.Sc.)

DESENVOLVIMENTO DE ALGORITMOS DE IA PARA DISPOSITIVOS  
VESTÍVEIS UTILIZANDO COMPUTAÇÃO DE BORDA

Jonathan Cristovão Ferreira da Silva

July/2023

Advisors: Ricardo Augusto Rabelo Oliveira  
Mateus Coelho Silva

Department: Computer Science

Wearable devices are increasingly present in our lives. In addition, artificial intelligence (AI) algorithms are becoming essential to compose these devices. As wearable devices are resource-constrained, technologies that require large computational capacity may be impractical for applications in this context, especially regarding edge computing. Given this, the work proposes the development of AI algorithms for integration in these devices with data processing at the edge without using cloud resources. This proposal is validated based on two case studies. The first case study is the application of Machine Learning and Deep Learning techniques in agriculture, intending to develop an intelligent helmet to carry out disease inspection in oranges. The second case study developed a new wearable solution for walking activity recognition. With the help of three AI algorithms, this case study presented new perspectives for the user's self-assessment based on the data collected in the activity performed. Thus, this work analyzes aspects of developing AI algorithms for integration in two wearable devices through edge computing.

# Sumário

## Lista de Figuras

## Lista de Tabelas

<b>1</b>	<b>Introdução</b>	<b>1</b>
1.1	Motivação . . . . .	2
1.2	Objetivos . . . . .	2
1.3	Contribuições . . . . .	3
1.4	Instituições Envolvidas . . . . .	3
1.5	Organização do Texto . . . . .	3
<b>2</b>	<b>Descrição do problema</b>	<b>4</b>
2.1	Estudo de caso 1: Inspeção de laranjas . . . . .	5
2.1.1	A Laranja . . . . .	6
2.2	Estudo de caso 2: Reconhecimento de Atividades Humanas . . . . .	8
<b>3</b>	<b>Referencial Teórico</b>	<b>10</b>
3.1	Inteligência Artificial . . . . .	11
3.1.1	Machine Learning . . . . .	11
3.1.2	Deep Learning . . . . .	12
3.2	Computação de Borda . . . . .	14
3.2.1	Computação de Borda: Vantagens . . . . .	14
3.3	Dispositivos Vestíveis . . . . .	15
3.3.1	Componentes dos sistemas vestíveis . . . . .	16
3.3.2	Requisitos de um Sistema Vestível . . . . .	16
3.3.3	Dispositivos vestíveis IA . . . . .	18
3.3.4	Visão Computacional . . . . .	18
3.4	Estudo de caso 1: Inspeção de laranjas . . . . .	19
3.4.1	Algoritmos de IA na agricultura . . . . .	20
3.4.2	Dispostivos e aplicações . . . . .	21
3.5	Estudo de caso 2: HAR . . . . .	22
3.5.1	HAR baseado em visão . . . . .	22

## SUMÁRIO

3.5.2	HAR baseado em sensores . . . . .	23
3.5.3	HAR baseado em IA . . . . .	23
3.5.4	HAR nos esportes . . . . .	24
3.5.5	Considerações . . . . .	24
<b>4</b>	<b>Metodologia</b>	<b>26</b>
4.1	Definição de Requisitos para a construção de um dispositivo vestível .	26
4.2	Estudo de caso 1: Inspeção de laranjas . . . . .	26
4.2.1	Algoritmo de Deep Learning . . . . .	29
4.2.2	Dispositivo vestível: Capacete inteligente . . . . .	31
4.3	Estudo de caso 2: HAR . . . . .	35
4.3.1	Sensores vestíveis para o HAR . . . . .	36
4.3.2	HAR algoritmos de IA . . . . .	37
4.3.3	Criação da base de dados . . . . .	38
<b>5</b>	<b>Resultados</b>	<b>42</b>
5.1	Resultados para o Estudo de Caso 1 . . . . .	42
5.1.1	Algoritmos de Machine Learning . . . . .	42
5.1.2	Algoritmos de Deep Learning . . . . .	45
5.1.3	Resultados com os hardwares do dispositivo vestível para o estudo de caso 1 . . . . .	48
5.2	Resultados para o Estudo de Caso 2 . . . . .	53
5.2.1	Algoritmos de IA para o HAR . . . . .	55
<b>6</b>	<b>Discussões</b>	<b>58</b>
6.1	Estudo de caso 1 . . . . .	58
6.1.1	Os algoritmos de machine learning . . . . .	58
6.1.2	Os algoritmos de Deep learning . . . . .	59
6.1.3	O dispositivo vestível . . . . .	60
6.2	Estudo de caso 2 - Reconhecimento de atividades Humanas . . . . .	60
6.2.1	Os algoritmos de IA para o HAR . . . . .	61
6.2.2	O dispositivo vestível . . . . .	61
<b>7</b>	<b>Conclusões e Trabalhos Futuros</b>	<b>63</b>
	<b>Referências Bibliográficas</b>	<b>65</b>

# Lista de Figuras

2.1	Três principais doenças da laranja [1]. . . . .	6
2.2	Inspeção manual de laranjas [1]. . . . .	7
2.3	Dispositivos para o monitoramento e o reconhecimento de atividades humanas - Adaptado [2]. . . . .	8
3.1	Estrutura Machine Learning - Adaptado [3]. . . . .	10
3.2	Distribuição da localização de dispositivos vestíveis no corpo humano (dados de 2008 a abril de 2017) [4]. . . . .	15
3.3	Diagrama para a construção de um dispositivo vestível - Adaptado - [5]. . . . .	17
3.4	Histograma HSV - Autoria própria. . . . .	19
4.1	A) Frescas; B) Cancro; C) Pinta preta; D) Greening - Autoria própria.	27
4.2	Processo de criação modelos de <i>machine learning</i> baseado em visão computacional - Autoria propria. . . . .	28
4.3	Ilustração da estrutura da rede de classificação - Adaptado [6]. . . . .	30
4.4	Fluxo de dados para o protótipo - Autoria própria. . . . .	31
4.5	Visão geral da arquitetura de solução proposta - Autoria própria. . .	32
4.6	O protótipo - Autoria própria. . . . .	33
4.7	Arquiteturas propostas para o estudo de caso - Autoria própria. . . .	34
4.8	Proposta do dispositivo vestível para o HAR - Autoria própria. . . . .	35
4.9	Dispositivo vestível usado para coletar dados de movimento do indivíduo. As áreas destacadas indicam o local onde cada dispositivo está posicionado - Autoria própria. . . . .	36
4.10	Ciclo completo de caminhada [7]. . . . .	37
4.11	<i>Data augmentation: Noise; Convolution; Quantize</i> - Autoria própria.	40
4.12	<i>Data augmentation: Drift</i> - Autoria própria. . . . .	41
5.1	Avaliação dos valores de acurácia e perda dos conjuntos de treinamento e validação. . . . .	45
5.2	Classificação com o modelo MobileNetV2 integrado em uma aplicação mobile. . . . .	48

## LISTA DE FIGURAS

5.3	Consumo de CPU do sistema. . . . .	49
5.4	Consumo de CPU do Raspberry Pi Zero W. . . . .	49
5.5	Frequência de processamento do sistema. . . . .	50
5.6	Frequência de processamento do sistema para o Raspberry Pi Zero W. . . . .	50
5.7	Consumo de memória do sistema. . . . .	51
5.8	Consumo de memória do sistema para o Raspberry Pi Zero W. . . . .	52
5.9	Consumo de energia do sistema. . . . .	52
5.10	Consumo de energia do sistema para o Raspberry Pi Zero W. . . . .	53
5.11	A) Stand; B) Seated. . . . .	54
5.12	C) Walk outdoor; D) Indoor uphill walk. . . . .	55
5.13	Resultado do treinamento dos algoritmos. . . . .	56
5.14	<i>Confusion Matrix</i> . . . . .	56

# Lista de Tabelas

4.1	Descrição da base de dados para o estudo de caso 1. . . . .	27
4.2	Descrição da base de dados para o estudo de caso 2. . . . .	38
4.3	Descrição dos dados enviados pelos sensores SPU. . . . .	39
5.1	Teste com o modelo para classificação de múltiplas classes . . . . .	43
5.2	<i>Confusion Matrix</i> - SVM . . . . .	44
5.3	<i>Confusion Matrix</i> - KNN. . . . .	44
5.4	<i>Confusion Matrix</i> - <i>Random Forest</i> . . . . .	44
5.5	<i>Confusion Matrix</i> - MLP. . . . .	44
5.6	Teste com modelo para detecção de laranja. . . . .	46
5.7	<i>Confusion Matrix</i> - MobileNetV2. . . . .	47
5.8	<i>Confusion Matrix</i> - ResNet50. . . . .	47
5.9	Resultados da análise temporal entre os modelos MobileNetV2 e Nas- NetMobile. Os testes apresentam separação estatística significativa para os resultados de cada par de algoritmos. . . . .	47
5.10	Métricas para os modelos LSTM, GRU e Simple RNN . . . . .	56

# Capítulo 1

## Introdução

Com o decorrer do tempo, novas tecnologias surgem e transformam a maneira como as pessoas se integram com o mundo ao seu redor, como o avanço da internet das coisas (*Internet of Things* - IoT) e a inteligência artificial (IA). Essas tecnologias permitem que sistemas de computação tenham novas aplicações para tarefas essenciais no nosso cotidiano, como, por exemplo, os dispositivos vestíveis, os quais vêm ganhando espaço em nossas vidas [8] [9]. Os dispositivos vestíveis possuem ampla aplicabilidade, como, por exemplo, no reconhecimento de atividades humanas (*Human Activities Recognition* — HAR). Além desta, podem ser usados para o monitoramento de caminhada [10] e outras atividades em diferentes contextos: industrial, saúde e esportes [11] [12].

Embora os dispositivos vestíveis sejam funcionais, as restrições de recursos, por exemplo, baixa memória RAM, impõem limitações na implementação de algoritmos de IA em sua estrutura. Além disso, a capacidade de processamento desses dispositivos são menores em comparação com outros dispositivos mais robustos. Assim, para a integração dos modelos de IA neste contexto de limitações, torna-se necessário o estudo dos aspectos construtivos dos algoritmos, de forma que cumpram os requisitos mínimos para a execução nos dispositivos vestíveis.

Diante disso, este trabalho aborda o desenvolvimento de algoritmos de IA para dispositivos vestíveis, de forma que executem o processamento na borda do dispositivo sem utilizar recursos em nuvem. Além de preservar os recursos desses dispositivos, esse tipo de processamento pode, também, diminuir a latência das informações geradas [13]. No estudo de caso 1 é desenvolvido um protótipo de um capacete inteligente que realiza uma inspeção de doenças em laranjas. Esse dispositivo pode auxiliar o citricultor na tomada de decisão durante o manejo dos frutos de forma rápida e eficiente. Por fim, o estudo de caso 2 aborda a construção de uma calça inteligente a partir de sensores para o HAR. Esse dispositivo coleta os dados do usuário para que um algoritmo realize a classificação dessas informações com base na fusão dos dados coletados pelos sensores vestíveis.

## 1.1 Motivação

Os dispositivos vestíveis, por serem hardwares compactos, possuem limitações computacionais, refletindo em dificuldades de executar algoritmos de IA no seu sistema operacional [14]. Além disso, as restrições de comunicação podem ser outro fator limitante na utilização de técnicas mais avançadas nestes dispositivos. Porém, já se tem conhecimento de que esses obstáculos possam ser mitigados com a inteligência baseada na computação de borda [13]. O processamento na borda é uma solução para superar eventuais problemas de latência e retira, em parte, a dependência de serviços em nuvem, ou seja, esse modo de processamento não depende de recursos externos. De posse dessa informação, nota-se a importância em estudar os modelos de IA para aplicação nesse cenário, de modo que não utilize todo o recurso do *hardware*.

## 1.2 Objetivos

Conforme mencionado nas subseções anteriores, os dispositivos vestíveis apresentam grande potencial de inovação tecnológica com aplicações de algoritmos de IA em diferentes contextos. Sendo assim, esse trabalho tem o seguinte objetivo principal:

- Analisar a perspectiva do uso de algoritmos IA nos dispositivos vestíveis através da computação de borda com base nos estudos de caso de inspeção de laranjas e reconhecimento de atividades humanas.

Os tópicos, análises e contribuições que esse projeto busca contemplar são:

- Validação dos algoritmos de IA utilizados para dois estudos de caso, pelas métricas padrão de avaliação, precisão, *recall* e F1-score;
- Estudar os algoritmos de IA para aplicações em diferentes *hardwares* embarcados para computação vestível;
- Estudo de caso 1: Propor e desenvolver um protótipo de capacete inteligente, com uma arquitetura descentralizada, para auxiliar citricultores na inspeção de laranjas e analisar a viabilidade da execução do método em plataforma embarcada;
- Estudo de caso 2: Propor e desenvolver um protótipo de uma calça inteligente para reconhecimento de atividades humanas, além de uma metodologia de classificação dos dados recebidos por estes sensores neste contexto.

## 1.3 Contribuições

- DA SILVA, Jonathan Cristovão Ferreira; SILVA, Mateus Coelho; OLIVEIRA, Ricardo AR. Towards a novel wearable solution for citrus inspection using Edge AI. In: **2022 IEEE 46th Annual Computers, Software, and Applications Conference (COMPSAC)**. IEEE, 2022. p. 966-971.
- DA SILVA, Jonathan CF et al. A novel intelligent mobile application using human-centered AR: A case study in orange inspection. In: **Anais Estendidos do XXI Simpósio Brasileiro de Fatores Humanos em Sistemas Computacionais**. SBC, 2022. p. 72-75.
- DA SILVA, Jonathan CF et al. Using Mobile Edge AI to Detect and Map Diseases in Citrus Orchards. **Sensors**, v. 23, n. 4, p. 2165, 2023.
- DA SILVA, Jonathan Cristovao Ferreira et al. Towards novel smart wearable sensors to classify subject-specific human walking activities. In: **Anais Estendidos do XII Simpósio Brasileiro de Engenharia de Sistemas Computacionais**. SBC, 2022. p. 68-73.
- DA SILVA, Jonathan CF. Dataset for Recognition of human activity. 2023. Disponível em: <https://www.kaggle.com/jonathansilva2020/datasets>.
- DA SILVA, Jonathan CF. Dataset for Classification of Citrus Diseases. 2022. Disponível em: <https://www.kaggle.com/jonathansilva2020/datasets>.

## 1.4 Instituições Envolvidas

O trabalho foi realizado no laboratório iMobilis, localizado no Departamento de Computação (DECOM) da Universidade Federal de Ouro Preto. A pesquisa foi fomentada com bolsa de estudos da Coordenação de Aperfeiçoamento de Pessoal de Nível Superior (CAPES).

## 1.5 Organização do Texto

O Capítulo 2 apresenta a descrição do problema. O Capítulo 3 mostra uma visão geral do referencial teórico para o assunto. No Capítulo 4 é abordada a metodologia e os requisitos para a construção dos dispositivos vestíveis inteligentes, além da definição dos testes de validação. Os resultados se encontram no Capítulo 5, com as conclusões preliminares alcançadas pelos estudos de caso. No Capítulo 6 são apresentadas as discussões obtidas através dos resultados da pesquisa. Por fim, o Capítulo 7 apresenta as conclusões e os trabalhos futuros.

# Capítulo 2

## Descrição do problema

Dispositivos vestíveis geralmente possuem recursos limitados de *hardware* em comparação com os computadores tradicionais. Executar modelos de IA nesses dispositivos tem sido, de certo modo, difícil. Além disso, o treinamento de certos algoritmos de IA no próprio *hardware* pode ser inviável. Estas dificuldades estão relacionadas à diferentes fatores, como:

- Limitações de processamento: dispositivos vestíveis geralmente têm processadores de menor potência em comparação com computadores tradicionais, o que pode limitar sua capacidade de lidar com algoritmos de IA que exigem grande capacidade computacional.
- Limitações de armazenamento: dispositivos vestíveis geralmente têm menor espaço de armazenamento quando comparados com os computadores tradicionais, o que pode limitar a quantidade de dados que podem ser armazenados localmente para uso em modelos de IA.
- Limitações de energia: dispositivos vestíveis geralmente têm baterias menores em relação aos computadores tradicionais, o que pode limitar o tempo de execução de modelos de IA que consomem muita energia.
- Limitações de conectividade: muitos dispositivos vestíveis dependem de conexões sem fio, como Wi-Fi ou Bluetooth, para se conectar a outros dispositivos ou serviços em nuvem para executar os modelos de IA. Essas conexões podem ser limitadas em termos de velocidade, largura de banda e confiabilidade, o que pode afetar a capacidade do dispositivo em executar modelos de IA em tempo real.
- Limitações de interface: dispositivos vestíveis geralmente têm telas pequenas e interfaces limitadas em comparação com computadores tradicionais, o que pode dificultar a visualização e a interação com modelos de IA complexos.

É notório que existem diferentes limitações relacionadas à IA. No entanto, apesar destas, a IA têm apresentado resultados promissores para uso nos dispositivos vestíveis, como a aplicação de modelos de *Machine Learning* (ML) para monitoramento e reconhecimento de pessoas. Na documentação do Keras [15] é possível observar que os algoritmos de IA, como modelos de *Deep Learning* (DL), são geralmente baseados em vários milhões de parâmetros, com muitas camadas em sua arquitetura, o que requer grande poder de processamento para sua implantação.

Antes de sua implantação, é importante considerar cuidadosamente as limitações do dispositivo e estudar os modelos de IA a fim de garantir que sejam executados com eficácia e eficiência. Diante deste gargalo relacionado aos modelos de IA, este trabalho busca desenvolver algoritmos de IA aplicados no contexto dos dispositivos vestíveis, com base em dois estudos de casos. Para isso, será utilizado o processamento na borda dos dispositivos para a preservação dos recursos utilizados e entrega de informações em baixa latência [13].

## 2.1 Estudo de caso 1: Inspeção de laranjas

Os avanços da Agricultura 4.0 [16], Agricultura Digital e Agricultura de Precisão [17] estão proporcionando um aumento na produção de alimentos de forma sustentável, otimizando os processos produtivos e reduzindo perdas com o controle de pragas e doenças [18]. Uma das tecnologias que auxiliam este desenvolvimento é a visão computacional, que, associada à computação embarcada, torna-se uma ferramenta muito importante neste contexto [19]. Um exemplo do uso da tecnologia no campo é a classificação de imagens de frutas usando DL. Essas técnicas auxiliam no desenvolvimento da indústria, ao mesmo tempo que contribuem para a otimização de setores tradicionais da economia, como, os exportadores de frutas [20]. Assim, o desenvolvimento de um algoritmo inteligente possibilita a identificação de doenças em laranjas em larga escala, de modo a se reduzir as perdas na citricultura decorrentes da instalação de patógenos [21].

Na literatura estão disponíveis exemplos do uso da visão computacional no contexto da agricultura, como a pulverização de plantas daninhas em tempo real [22] e a inspeção e controle de qualidade de frutas [23]. Além disso, há ainda outras técnicas inovadoras aplicadas neste contexto da agricultura, como a realidade aumentada (RA), que permite sobrepor elementos virtuais à nossa visão da realidade [24]. Esta técnica pode ser replicada em atividades agrícolas, por exemplo, em fazendas de gado [25] e gestão de qualidade de água em fazendas [26].

As doenças que acometem as laranjas apresentam aspectos visíveis em suas superfícies, possibilitando a aplicação de técnicas de visão computacional para pré processar as imagens. Com o auxílio de um algoritmo de IA torna-se possível re-

alizer a identificação do tipo de doença presente na fruta. Assim, a aplicação de técnicas inteligentes neste contexto permite que os citricultores melhorem a qualidade de seus produtos, sem contar a redução significativa na perda das colheitas. Com uma inspeção rápida e eficiente, o manejo dessas frutas pode ser feito de forma efetiva evitando perda ou diminuição da qualidade do fruto causado pelas infestações de doenças e pragas nas plantações.

### 2.1.1 A Laranja

Conforme a Associação Nacional dos Exportadores de Sucos Cítricos (CitrusBR), a laranja é uma das frutas mais cultivadas em todo o mundo, sendo um produto com forte liquidez. O tipo e a qualidade nutricional são determinados pelo tamanho, cor, quantidade de açúcares, acidez, vitaminas, minerais, pectinas e pigmentos, dentre outros componentes presentes no suco, os quais são variáveis a depender do local de cultivo [27]. O Brasil, país com uma das maiores participações na produção mundial de laranja, deteve cerca de 22,0% na produção do fruto em 2018 [28]. Além do suco, existem alguns componentes que pode ser aproveitados pela indústria como subprodutos, inclusive, também podem ser exportados. Os principais subprodutos do processo industrial são:

- O *Comminuted Citrus*, base resultante da moagem da fruta inteira que é utilizado como ingrediente para bebidas à base de frutas [27];
- A polpa, utilizada em bebidas como fonte de açúcar [27].



Cancro cítrico



Pinta preta



Greening cítrico

Figura 2.1: Três principais doenças da laranja [1].

Há duas fases de processamentos industriais com importantes matérias-primas extraídas da laranja. Em uma dessas fases é produzido o óleo da casca, sendo que um de seus principais componentes, o limoneno, é utilizado nas indústrias de produtos de limpeza doméstica e industrial, como desinfetantes, limpadores de superfície e

removedores de manchas. Além disso, são utilizados na produção de compostos cosméticos e produtos químicos [27]. Pelo fato da laranja estar atrelada à diferentes setores industriais, o controle fitossanitário deve ser de alta eficiência, haja vista que um dos problemas que podem impactar a produção dos produtos e subprodutos da laranja são as infestações de doenças. Nesse contexto, torna-se importante conhecer as principais doenças que afetam a laranja, conforme demonstrado na Figura 2.1:

- Cancro cítrico: é outra doença que afeta todas as espécies e variedades de laranjas de importância comercial, causado pela bactéria *Xanthomonas citri subsp. citri* [29];
- Pinta preta: também conhecida como mancha preta, pode ser identificada por lesões nas folhas e frutos, e ocorre em todas as variedades de laranja. Segundo o Fundo de Defesa da Citricultura [30] a pinta preta é considerada uma das doenças mais importantes das citriculturas brasileira e mundial;
- *Greening* cítrico: é uma das doenças que mais afetam as plantações de citros do mundo, causada pelo inseto “psilídeo cítrico asiático”. Essa doença apresenta uma grande ameaça para as espécies de laranjas, pois as árvores infectadas produzem frutas verdes, deformadas e amargas, tornando-as inadequadas para a comercialização como frutas frescas ou para a produção de suco [31].



Figura 2.2: Inspeção manual de laranjas [1].

Os métodos de inspeção comumente aplicados no cultivo da laranja, são feitos de forma manual, como ilustra a Figura 2.2, onde citricultores realizam uma inspeção nas laranjeiras. A inspeção manual é um processo demorado e laborioso, além de existir a possibilidade de ser ineficaz, a depender do período de contaminação da planta. Se as medidas de combate forem tomadas tardiamente o avanço das doenças pode impactar a qualidade do fruto e todo o sistema produtivo. Assim, há uma

necessidade de desenvolver novas tecnologias para a citricultura, visando otimizar o processo de inspeção. Dessa forma, para o estudo de caso 1 é proposto uma solução de dispositivo vestível inteligente para auxiliar os citricultores na inspeção de laranjas através da computação de borda. Além disso, essa solução é desenvolvida com um fluxo de dados distribuído para preservar os recursos de processamento e energia do dispositivo.

## 2.2 Estudo de caso 2: Reconhecimento de Atividades Humanas

Hoje a corrida é um dos tipos de atividade física mais populares realizados globalmente [32]. A adoção pela corrida por atletas profissionais ou amadores pode estar diretamente relacionada à facilidade de prática desse esporte, uma vez que, qualquer espaço público pode ser utilizado para este fim, além do baixo investimento financeiro necessário. Em trabalhos anteriores foi demonstrado que, além da alimentação balanceada, a metodologia correta e o posicionamento dos membros na execução do exercício influenciam diretamente no desempenho do atleta para este tipo de esporte. Desta forma, avaliar a biomecânica do corpo do indivíduo pode não só prevenir lesões futuras, como também melhorar seu desempenho geral [33][34].



Figura 2.3: Dispositivos para o monitoramento e o reconhecimento de atividades humanas - Adaptado [2].

A Figura 2.3 apresenta exemplos de dispositivos que podem entregar informações importantes para o usuário dentro de uma atividade realizada, por exemplo, corrida.

Além dessas soluções, câmeras e marcadores corporais podem ser utilizados para avaliar diferentes aspectos de corredores, como marcha, ângulo de inclinação do pé, flexão do joelho, queda pélvica, entre outros [35]. Apesar de sua validade e resultados, a abordagem de vídeo para avaliar a corrida carece de flexibilidade e validade, pois é necessário um ambiente muito bem controlado. Por exemplo, os pesquisadores frequentemente usam vídeos gravados de atletas correndo em uma esteira como referência para avaliar os pontos de referência anteriores.

Sistemas destinados ao HAR geralmente usam métodos de DL para identificar tarefas com base em dados enviados por sensores [36]. Isso se deve à eficiência desses modelos na fase de aprendizado, já que não dependem inteiramente de dados pré-processados como os modelos de ML, portanto, seu desempenho é maior na execução dessas tarefas. Os *smartwatches* e os smartphones são dispositivos comumente usados nesse contexto, pois possuem diversos sensores integrados, como acelerômetro e giroscópio [36]. Além destes dispositivos possuem diversos sensores em seu sistema físico, fazem também interface com aplicativos em tempo real [37]. Os dados coletados e armazenados por eles são usados por algoritmos inteligentes, com o intuito de classificar e apresentar informações valiosas ao usuário.

Sensores únicos, como dos dispositivos citados anteriormente (acelerômetro e giroscópio), possuem desvantagens no uso dos dados gerados em comparação com outros dispositivos vestíveis para o HAR [38]. Uma das desvantagens está associada à imprecisão nas análises dos dados. Para suprir esta desvantagem, há um método que utiliza a fusão de dados, que combina informações de múltiplos sensores para aumentar a confiabilidade dos sistemas para o HAR, abordando possíveis problemas no reconhecimento de atividades pelos dados gerados [39]. No entanto, para que o sistema realize um reconhecimento de atividade de forma eficiente, é necessário integrar outros sensores ao sistema de reconhecimento [40].

A proposta para este estudo de caso é desenvolvida a partir dos dados recebidos por múltiplos sensores. Os sensores vestíveis acoplados ao corpo do indivíduo detectam e coletam informações da atividade de caminhada por meio do uso de Unidade de Medidas Inerciais (IMUs). Isso se torna possível devido ao uso de métodos de DL, que realizam aprendizado de dados pela técnica de fusão de dados [41]. Ademais, esta abordagem oferece flexibilidade para outras atividades, por exemplo, os atletas são capazes de coletar dados enquanto praticam sua atividade em qualquer ambiente. Essa particularidade permite criar um cenário mais realista para recuperar e analisar os dados posteriormente. Além do aumento da precisão dos dados proporcionados pelas IMUs, os sensores não possuem cabos/fios aparentes permitindo, também, maior conforto durante a execução da corrida. A inexistência dos fios permite que os dispositivos sejam posicionados em praticamente qualquer parte do corpo sem interferir nos movimentos naturais do atleta.

# Capítulo 3

## Referencial Teórico

Os Capítulos 1 e 2 foram delineados com o objetivo de apresentar a motivação e os principais aspectos para o desenvolvimento do presente trabalho, destacando os elementos que demonstram a importância da criação de soluções inteligentes no contexto dos dispositivos vestíveis, especialmente para estudos realizados dentro da citricultura e no HAR. Neste capítulo apresentamos uma revisão embasada na literatura sobre conceitos relacionados aos algoritmos de IA, com o ML e DL, bem como algumas de suas aplicações para a classificação de imagens e o HAR. Além disso, apresentamos ainda os elementos necessários para a compreensão e construção de um dispositivo vestível, conforme os requisitos necessários para aplicação da IA destes *hardwares*.

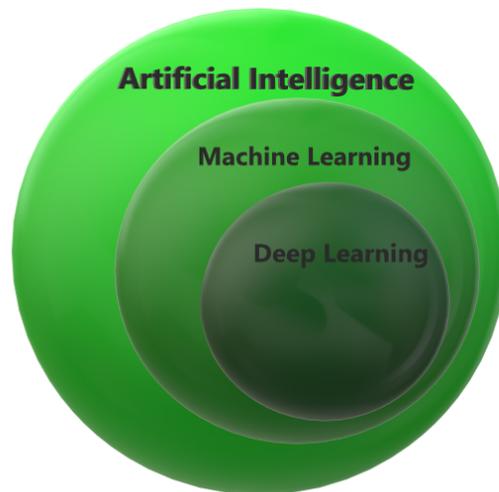


Figura 3.1: Estrutura Machine Learning - Adaptado [3].

## 3.1 Inteligência Artificial

A inteligência artificial pode ser definida como o método ou técnica necessária para construir programas inteligentes de computadores [3]. Ainda segundo o mesmo autor, a inteligência computacional, por sua vez, é a capacidade do computador aprender uma tarefa do mundo real e conseguir solucioná-la. Dentro da IA existe uma divisão nos campos de aprendizado, como mostrada na Figura 3.1, explorada nas subseções posteriores.

### 3.1.1 Machine Learning

Aprendizado de máquina ou *Machine Learning* é um campo da IA com o foco na construção de algoritmos e modelos inteligentes [42]. Essas técnicas permitem que máquinas aprendam, a partir de experiências anteriores, a otimizar tarefas específicas. Dessa forma, tornam-se essenciais para tomada de decisão e previsão de novos dados, já que com o uso de técnicas estatísticas e de análise de dados é possível encontrar padrões entre as informações sem precisarem que os mesmos sejam reprogramados. Como não há um modelo padrão que funcione em todos os casos, a escolha da melhor técnica de aprendizado depende do tipo do problema aplicado. Em geral, os algoritmos de ML podem ser classificados em quatro grupos:

- **Aprendizado supervisionado:** Os dados de entrada e saída são rotulados por supervisão humana para o treinamento do algoritmo. O modelo é treinado por um conjunto de dados que contém as informações corretas de cada classe [43]. Um exemplo básico é a busca de imagens do Google, na qual o algoritmo localiza a origem da imagem e outras semelhantes [42].
- **Aprendizado não-supervisionado:** Neste tipo de aprendizado, como não há rótulos nos dados de entrada, o algoritmo busca um determinado padrão de forma que faça um agrupamento dos dados segundo a proximidade das características de cada um [43]. Por exemplo, em uma pesquisa sobre hábitos de consumo, é preciso agrupar informações como registros e frequência de compras juntamente com, o perfil do cliente para encontrar padrões [42].
- **Aprendizado semi-supervisionado:** Abordagem que combina os dois tipos de aprendizado mencionados anteriormente. Assim, há uma pequena quantidade de respostas definidas entre as incertezas, que ajudam a direcionar as descobertas do modelo de aprendizado [44].
- **Aprendizado por reforço:** Não possui nenhum conjunto prévio de dados. Neste tipo de aprendizado o agente aprende com os erros [45][46], sendo classificado como um algoritmo de aprendizado por reforço [47], onde o agente recebe

recompensas positivas ou negativas de acordo com o progresso na execução da tarefa dentro do ambiente de aprendizado.

Cada um desses tipos de aprendizados tem suas próprias técnicas e aplicações específicas, contudo, todos se baseiam na ideia de que as máquinas podem aprender e melhorar com o tempo, tornando-se cada vez mais eficientes em tarefas específicas.

## Detecção e Classificação das imagens

Na literatura existem diversos modelos de ML para a classificação de imagens [48], como o *Support Vector Machine* (SVM), que separa os dados por um hiperplano entre classes diferentes [49]. Outro algoritmo é o *K-Nearest Neighbors* (KNN), que tem como critério de avaliação os dados dos vizinhos, classificando os dados baseando-se em um parâmetro de distância euclidiana [50] [51]. Também há o *Random Forest*, que gera várias árvores de decisão e combina o resultado da classificação de todas elas [52]. No ramo das árvores geradas, há condições para verificação e classificação dos dados de acordo com a classe em análise.

Na tarefa de classificação de imagens também há o uso das redes neurais artificiais [53]. Essas redes imitam o funcionamento de neurônios do sistema nervoso humano, como o perceptron de várias camadas (*Multi-Layer Perceptron* — MLP) [54][55]. Essa técnica faz uma combinação linear de dados de entrada através de uma função de ativação para assim realizar a classificação dos dados de entrada, como por exemplo, imagens de exames de mamografia [56].

Contudo, os algoritmos citados recebem aleatoriamente os dados para fazer a classificação das imagens. O SVM classifica os dados de forma mais precisa com o kernel RBF, contrário a um kernel linear ou polinomial. O KNN possui uma adaptação rápida aos dados comparando os vizinhos mais próximos. As árvores geradas pelo *Random Forest* realizam uma classificação eficiente sem dificuldades. O MLP, por sua vez, atinge um resultado expressivo na classificação, com o ajuste dos pesos entre suas camadas[57]. A literatura reforça a eficiência das análises por esses modelos na classificação de imagens em doenças de laranjas [58].

### 3.1.2 Deep Learning

Aprendizado de profundo ou DL são redes neurais com grande quantidade de camadas adicionais em sua arquitetura, que otimizam os resultados e melhoram a aprendizagem do modelo [59] [60]. A estrutura dessas redes segue por várias camadas interconectadas, com cada camada baseando-se nos resultados das camadas anteriores, chamado de propagação direta [61].

O *backpropagation* faz o caminho reverso da propagação direta. Ele utiliza métodos iterativos, como o gradiente descendente, para calcular erros nas previsões

e ajustar os pesos da função das últimas camadas para a primeira, na fase de treinamento [62]. O *backpropagation* em conjunto com a propagação direta, permite que a rede neural faça as previsões e a correção dos erros no processo de aprendizagem. Assim, com um número específico de interações a rede se torna mais precisa e eficiente [61].

O conceito mencionado acima descreve o funcionamento simplificado das redes neurais profundas. Há duas variações dessas redes comumente utilizadas para a construção de modelos de IA, sendo elas:

- **Redes neurais convolucionais (CNNs):** redes profundas que possuem várias camadas em sua estrutura, como camada convolucional, camada de não linearidade, camada de agrupamento e camada totalmente conectada [63]. A CNN tem um excelente desempenho em problemas de ML, como na classificação de imagens, detecção ou reconhecimento de objetos, utilizando técnicas de visão computacional [64].
- **Redes neurais recorrentes (RNNs):** redes neurais artificiais comumente utilizadas para resolução de problemas com dados sequenciais ou séries temporais, como processamento de linguagem natural [65]. Do mesmo modo que as CNNs, as RNNs utilizam dados de treinamento para aprender. Elas se distinguem por sua “memória”, pois obtêm informações de entradas anteriores para influenciar a entrada e a saída atuais. Enquanto as redes neurais profundas tradicionais assumem que as entradas e saídas são independentes umas das outras, a saída das RNNs depende dos elementos anteriores dentro da sequência. Embora eventos futuros também sejam úteis para determinar a saída de uma determinada sequência, as redes neurais recorrentes unidirecionais não podem explicar esses eventos em suas previsões [66].

As redes profundas são construídas com muitas camadas em sua arquitetura, o que demanda um poder computacional maior para executá-las. Atualmente, as unidades de processamento gráfico (*Graphical Processing Units* — GPUs) são as melhores opções para o treinamento dessas redes, uma vez que são capazes de processar um grande volume de dados de forma paralela [60]. No contexto dos dispositivos vestíveis, é importante destacar que nem todos possuem esse poder de processamento. Dessa forma, uma alternativa a esses modelos são as redes DL compactas, como as MobileNets [67] e as redes EfficientNets, que possuem menos camadas e uma menor quantidade de parâmetros [68].

## 3.2 Computação de Borda

Computação de borda ou *Edge computing* é um paradigma da computação que tem como ideia central executar as tarefas de computação mais próximo da fonte dos dados, ou seja, na borda do dispositivo ou rede [69]. Esse conceito é difundido por diferentes pesquisadores: Satyanarayanan [69] afirma que é um novo modelo que implementa os recursos de computação e armazenamento mais próximo dos dispositivos móveis ou sensores, como micro *data centers*; por outro lado, Zha [70] descreve a computação de borda como um modelo que unifica os recursos que estão próximos do usuário, fornecendo serviços de computação, armazenamento de dados e recursos de rede. Em outras palavras, computação de borda migra os serviços de computação, capacidades de armazenamento e recursos para a borda de forma que forneça esses serviços de maneira eficiente. Essa abordagem é muito importante pois busca atender às necessidades críticas da indústria, como a conexão, requisito de baixa latência e alta largura de banda na rede, otimização de dados, segurança e privacidade [71].

### 3.2.1 Computação de Borda: Vantagens

Tendo em vista que o modelo da computação de borda realiza o armazenamento e processamento dos dados na borda dos dispositivos, sem realizar o *upload* dos dados para a nuvem, percebe-se que este modo de computação possui importantes vantagens quando comparado com a computação em nuvem. Em relação às vantagens, podemos citar os seguintes aspectos:

- **Processamento e análise em tempo real:** o crescimento rápido do volume de dados e a pressão da largura de banda da rede são desvantagens da computação em nuvem [72] e pode aumentar a latência dos dados. Diferentemente da computação de borda, como as tarefas são realizadas no nó da borda, o tempo de resposta para a transmissão das informações é reduzido. Além disso, como esse modo de computação é comumente aplicado à uma pequena escala de dados, o custo no processamento e o consumo de energia dos dispositivos são reduzidos quando comparados aos dispositivos que enviam informações para a nuvem.
- **Segurança e privacidade:** a computação em nuvem requer que os dados sejam carregados na nuvem para processamento. Nesse processo, há riscos de segurança e privacidade, como perda e vazamento de informações [73]. Na computação de borda esses riscos podem ser evitados, tendo em vista que o processamento de dados é realizado no próprio dispositivo, sem a necessidade de fazer *upload* para a nuvem.

Como pode-se observar, existem algumas semelhanças e diferenças da computação de borda com a computação em nuvem. No entanto, cabe salientar que a computação de borda é uma extensão da computação em nuvem, porém, com suas próprias características. A computação em nuvem tem como principal característica a possibilidade de compreender o todo, capacidade de processar uma grande quantidade de dados e realizar análises aprofundadas. Por outro lado, a computação de borda se concentra no processamento local e pode desempenhar um papel melhor para uma análise em pequena escala e em tempo real [74]. Portanto, a computação de borda torna-se a melhor opção para análises de pequena escala e construções de soluções inteligentes para resolução de serviços locais.

### 3.3 Dispositivos Vestíveis

Os dispositivos ou sistemas vestíveis podem ser definidos como um tipo de tecnologia que o usuário pode vestir ou utilizar como acessório junto ao corpo, por exemplo, *smartwatches*, *smartbands* e *smartclothing* [75]. Essa tecnologia está em crescente expansão no mercado e em áreas de pesquisa, portanto, provavelmente estará presente em diversos nichos nos próximos anos [4]. Esses autores afirmam que a pesquisa com os vestíveis está preenchendo lacunas entre muitos projetos acadêmicos e profissionais, como eletrônica, moda e sistemas de energia. A Figura 3.2 apresenta o crescimento desses dispositivos a partir de 2008 até o início de 2017.

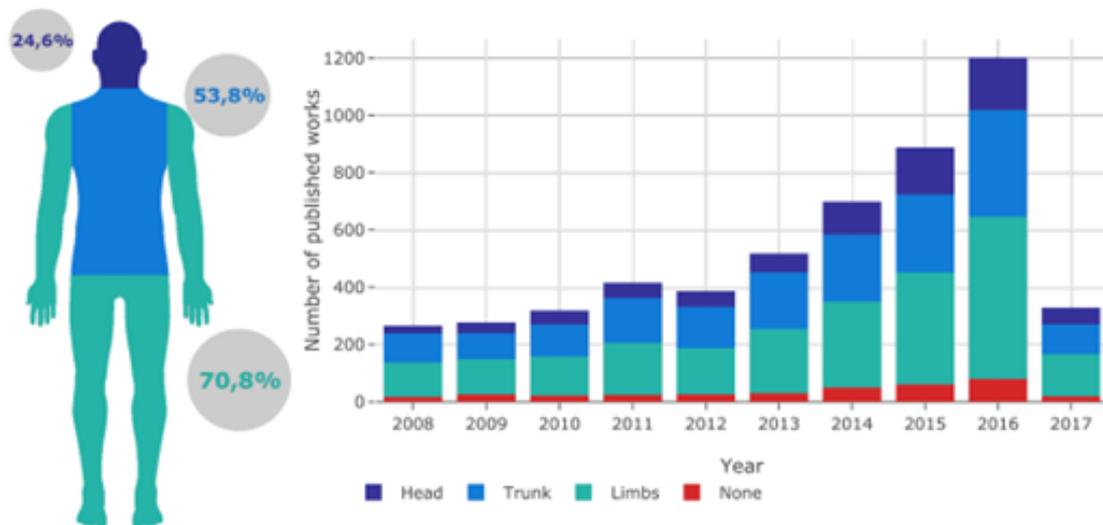


Figura 3.2: Distribuição da localização de dispositivos vestíveis no corpo humano (dados de 2008 a abril de 2017) [4].

Atualmente, os dispositivos vestíveis podem ser relativamente simples, incluindo pequenos sensores para a captura de dados brutos, como acelerômetros e giroscópios

[76]. As possibilidades do uso dessa tecnologia são ampliadas com a integração dos modelos de IA [77]. Com auxílio desses algoritmos, os dispositivos vestíveis se tornam uma ferramenta importante para diversas áreas, como no monitoramento do quadro clínico de pacientes na área médica [78] e nos esportes, com a medição da frequência cardíaca ou oxigenação sanguínea para melhorar o desempenho físico de atletas [79].

### 3.3.1 Componentes dos sistemas vestíveis

Nesta subseção, são discutidos os principais componentes que compõem os sistemas vestíveis. Estes componentes são:

- **Microcontroladores:** usados para processar e controlar informações. Por exemplo, o Arduino [80].
- **Sensores e atuadores:** coletam informações sobre o ambiente, o usuário ou ambos. Exemplos: captura de imagem (câmeras), localização (GPS), orientação do corpo (IMU), distância (LIDAR) e sensores de frequência cardíaca. No que se referem aos atuadores, podemos usar como exemplo LEDs ou alto-falantes, que emitem respostas sobre as informações processadas pelo sistema após serem coletadas pelo sensor [81].
- **Software/Firmware no dispositivo vestível:** Integra o aplicativo que realiza a tomada de decisão, a coleta de informações pelos sensores e define como serão processadas no Hardware [82].

A comunicação entre os dispositivo vestíveis e sensores pode ser feita por condutores têxteis, flexíveis e utilizados para a construção de circuitos. Para a conectividade sem fio há o Wi-Fi ou Bluetooth BLE [83]. A energia elétrica utilizada para operar todo o sistema é também um fator importante, uma vez que essa energia pode ser fornecida por baterias portáteis, sendo independente para cada componente ou centralizada para todo o sistema vestível.

### 3.3.2 Requisitos de um Sistema Vestível

Conhecendo os dispositivos vestíveis e seus componentes, é essencial definir o design e a integração entre todo o sistema para otimizar os recursos utilizados, como memória e consumo de energia [84]. Desse modo, nesta seção são apresentados aspectos importantes para a construção desses dispositivos. A construção dos dispositivos vestíveis se inicia com a definição dos requisitos que possuem algumas restrições, como energia, robustez e comunicação [85]. Dessa forma, ao definir um

design para esses dispositivos, um dos primeiros aspectos a ser considerado é o requisito de energia [86]. Além disso, estes dispositivos possuem limitações para seu desenvolvimento, como peso e tamanho, tendo em vista que devem ser confortáveis ao usuário, portanto, o requisito de robustez é um dos mais importantes no desenvolvimento desse tipo de dispositivo [87].

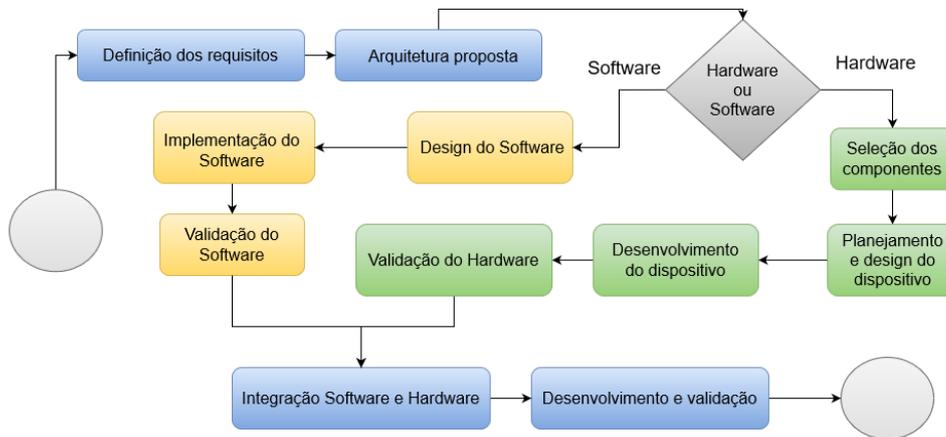


Figura 3.3: Diagrama para a construção de um dispositivo vestível - Adaptado - [5].

A maneira como os componentes do sistema vestível são integrados ao seu sistema define o tipo de sua arquitetura. Uma arquitetura descentralizada utiliza comunicação entre o *hardware* e os sensores via *Wi-Fi* ou *Bluetooth* [88]. Por outro lado, em uma arquitetura centralizada os componentes são conectados diretamente por cabos ao *hardware* principal [81]. Esse aspecto pode influenciar no requisito de comunicação do dispositivo, que segundo Chen (2016) [89] é um fator crítico para um sistema que tem como requisito a comunicação em tempo real. Assim, a arquitetura do dispositivo pode influenciar no desempenho energético e na comunicação [90].

Na literatura encontramos aplicações dos dispositivos vestíveis em campo, como exemplo, com sensores integrados nas vestimentas de um técnico agrícola para monitorar suas atividades nas plantações [91]. Neste contexto, devem atender os requisitos apresentados nesta seção. A Figura 3.3 é uma visão geral das etapas comuns geralmente consideradas ao desenvolver um novo dispositivo vestível em campo [5].

Outra possível aplicação em campo é no contexto da mineração [82]. Os autores apresentam um sistema vestível com a proposta de analisar o impacto operacional neste ambiente. Este equipamento pode se tornar uma solução trivial em diversas áreas essenciais para a atividade humana, tanto no monitoramento do usuário quanto na coleta de informações externas, como na citricultura e ambientes de difícil acesso.

Nestas aplicações dos dispositivos vestíveis em campo há algumas desvantagens,

como sua autonomia energética para operar por um período prolongado. Portanto, para atender os requisitos mínimos necessários ao desenvolver um dispositivo vestível inteligente que seja eficiente, primeiro temos que atender os aspectos revisados nesta seção, afim de garantir um bom desempenho e autonomia energética do sistema.

### 3.3.3 Dispositivos vestíveis IA

Há muitos desafios para integração de IA nos dispositivos vestíveis [92], em especial quando se trata da integração de algoritmos de DL. Grande parte dos algoritmos de DL possuem um número consideravelmente alto de camadas em sua arquitetura, sendo uma combinação entre um grande número de neurônios. Além disso, ainda existem problemas, como privacidade, gerenciamento de recursos e latência do sistema. Deng et al, (2019) [93] propuseram o designer de *framework* com aceleração do processador, como adaptação do modelo, para resolver um problema de restrição de recursos nesses dispositivos. Tal feito reforça a importância de novas arquiteturas para melhorar o desempenho da IA nos dispositivos.

Liu et al, (2019) [94] descrevem a dificuldade em executar a IA em um dispositivo vestível, com o processamento na borda, para detectar e classificar objetos, devido aos requisitos do sistema. Eles aplicam técnicas de descarregamento de baixa latência para resolver esse problema em aberto e obtém uma melhora na precisão das tarefas para detecção de objetos e de pontos-chave em humanos. A computação de borda, neste cenário, está sendo empregada para diversos dispositivos, fornecendo informações precisas aos usuários através de algoritmos de IA [95].

### 3.3.4 Visão Computacional

Um dispositivo vestível pode ter parte do sistema de sensoriamento baseado em visão computacional. A técnica em questão tem por finalidade replicar o funcionamento da visão humana. Baseando-se nesse conceito, a visão computacional objetiva permitir que os computadores identifiquem e processem objetos em imagens e vídeos da mesma maneira que os humanos [96].

Para os computadores, as imagens são conjuntos de *pixels*, que geralmente variam entre 0 a 255, os quais indicam sua intensidade. Para cada cor em cada *pixel* uma informação é armazenada usando 8 *bits* (1 *byte*). Para imagens coloridas, cada *pixel* pode ser representado por 3 valores (vermelho, verde e azul) na mesma escala de 0 a 255. Esse sistema de cores se chama RGB (do inglês, *red*, *green* e *blue*). Nessa representação, as três cores são combinadas de várias formas de modo a reproduzir um espectro cromático, com as cores captáveis pelos olhos humanos. Desse modo, com combinações diferentes dos canais RGB pode-se apresentar quaisquer outras cores delimitadas por esse espaço de cor.

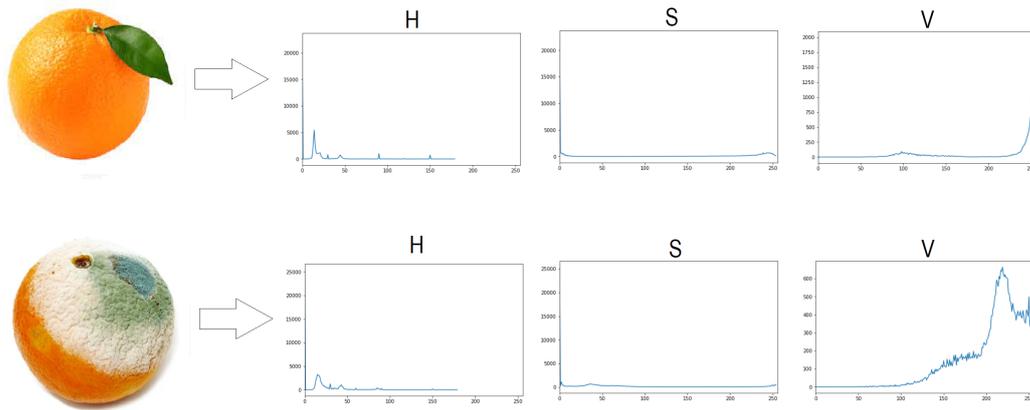


Figura 3.4: Histograma HSV - Autoria própria.

Outro sistema de cores muito usado para representar imagens é o HSV. Nesse sistema cada cor é representada por uma informação de matiz, saturação e brilho distintos. O uso da componente (matiz) pode tornar o método menos sensível às variações de iluminação, podendo ser útil para diferentes tipos de aplicações, como na Figura 3.4, que demonstra a distribuição da intensidade de pixels em cada um dos canais HSV de duas laranjas. Com a análise em um histograma de uma imagem tem-se uma percepção da quantidade de brilho, saturação e intensidade da cor presente na imagem. Assim, torna-se possível extrair as características presentes em qualquer objeto, a partir da manipulação desses três canais, juntos ou individuais, representados pelo histograma HSV. Contudo, as informações sobre texturas, formatos e quantidade de cor, são preciosas para alimentar algoritmos na construção de modelos de IA para a classificação de imagens.

### 3.4 Estudo de caso 1: Inspeção de laranjas

A inspeção de laranjas é uma técnica usada tanto para o controle de qualidade como também para a detecção precoce de doenças na fruta [97]. Na etapa de manejo pré e pós-colheita, a inspeção pode ser feita por meio de imagens hiperespectrais, permitindo a identificação de possíveis doenças através de anormalidades na casca da laranja. Nesse contexto, se torna viável a aplicação das técnicas de visão computacional, as quais estão se tornando uma ferramenta científica e comercial de grande importância para o segmento da citricultura [97].

As técnicas de visão computacional são comumente usadas para a extração de características de imagens, como a segmentação [98]. Yin et al,(2017) [99] faz a extração dos dados com técnicas hiperespectrais para o processamento das imagens

de laranjas, por meio da segmentação das regiões afetadas por fungos. Além disso, usam um modelo de ML para a classificação das imagens. A diferença para o estudo de caso 1 com esta literatura, é que os autores não realizam um detalhamento do desempenho dos algoritmos aplicados para a classificação das imagens

Na literatura, alguns trabalhos utilizam Veículos Aéreos Não Tripulados (VANT) para realizar inspeções em plantações [100], como detectar [101] e mapear doenças [102].

No contexto da citricultura, esses métodos podem conter falhas inerentes à metodologia, pois com o uso do VANT a fruta não é inspecionada totalmente, portanto, se as doenças estiverem em um lado diferente da câmera do VANT, não serão detectadas. Dessa forma, para informações mais fidedignas é imprescindível que o usuário movimente os VANT entre as laranjeiras, o que torna-se inviável por riscos de danos no equipamento. Com o dispositivo vestível o usuário poderá realizar a tarefa de inspeção interagindo com os frutos, por exemplo, rotacionando o objeto para verificar toda a sua superfície, permitindo a detecção acurada de doenças que acometam os frutos. O estudo de caso 1 procura contemplar este aspecto de mobilidade entre as plantações, além disso, usar um modelo de IA por uma análise hiperespectral das imagens para a identificação das doenças nas laranjas.

### 3.4.1 Algoritmos de IA na agricultura

Algoritmos de IA possuem uma função importante para a inspeção de frutos na agricultura. Como exemplo podemos citar, o KNN, algoritmo de ML usado para a classificação de imagens de laranjas que não requer um alto poder computacional para implementação [103]. Modelos de DL também estão sendo incorporados na agricultura [104]. O setor pode se beneficiar dessas técnicas para processos mais modernos, econômicos e seguros. Os algoritmos de IA associados aos sistemas de borda tornam-se uma ferramenta valiosa nesse contexto, por exemplo, para a detecção de doenças de frutas através de imagens coletadas no ambiente [105].

Too et al, (2019) [106] utilizaram algoritmos de DL para o reconhecimento de doenças em plantas. Os autores compararam os modelos VGG 16, Inception V4, ResNet com 50, 101 e 152 camadas e DenseNets com 121 camadas. Os modelos foram treinados com a base de dados *plantVillage* com 38 classes diferentes, incluindo imagens doentes e saudáveis de folhas de 14 plantas. Observaram que as DenseNets tendem a melhorar a precisão com o número crescente de épocas, atingindo uma precisão de 99,75%, sendo superior as outras arquiteturas, isso sem apresentar sinais de *overfitting* durante o treinamento. Um modelo sem *overfitting* pode ser utilizado para prever dados nunca vistos anteriormente, uma vez que se torna genérico após a etapa de treinamento. Além disso, as DenseNets requerem um número menor

de parâmetros e um tempo de computação razoável para alcançar desempenhos de última geração.

Em outra análise comparativa, Subetha et al, (2021) [107] usaram alguns modelos tradicionais de DL para identificação de doenças em folhas de macieira. O conjunto de dados utilizado contém 3.651 imagens e quatro categorias, sendo sarna, saudável, múltiplas doenças e ferrugem na macieira. Nos testes finais alcançaram 87,7% de precisão na classificação dessas doenças com as redes ResNet50 e VGG19. Outros autores realizaram uma análise semelhante, também com alguns modelos de DL, no entanto, para detectar doenças da cultura do arroz [108]. Estes autores compararam os desempenhos de cinco modelos para as imagens coletadas: VGG16, VGG19, ResNet50, ResNet50V2 e ResNet101V2. O modelo com melhor desempenho foi o ResNet101V2, com uma precisão de 86.79%.

Na agricultura, há diversas aplicações com os algoritmos de IA para a detecção e identificação de doenças. Entretanto, é necessário um estudo detalhado dos modelos de IA para definir um modelo mais viável com base nos recursos de *hardware* disponíveis. Portanto, o estudo de caso 1 busca utilizar um modelo que seja eficiente e não exija o limite de processamento do *hardware*.

### 3.4.2 Dispositivos e aplicações

Com a agricultura 4.0, dispositivos e aplicativos inteligentes estão cada vez mais inseridos para solucionar problemas nas plantações, como a identificação de doenças [109], problema este semelhante à tarefa relacionada ao estudo de caso 1. Neste cenário, os dispositivos *mobile* com RA e IA podem ajudar os agricultores de várias maneiras durante o manejo e cultivo de plantações, utilizando menos recursos [110]. Os avanços nessas tecnologias estão se tornando tendências para o desenvolvimento da agricultura inteligente [111].

Katsaros et al,(2017) [112] afirmam que ferramentas de RA podem fornecer informações assistivas para detecção de pragas em plantações por meio de dispositivos inteligentes. Uma ferramenta de RA em campo pode auxiliar os agrônomos na resolução de problemas práticos em áreas de baixo desempenho [113], melhorando a qualidade da produção [114].

Com integração de novas tecnologias, como IA e RA, os dispositivos baseados na computação de borda aplicados no contexto da agricultura melhoram a segurança e privacidade dos dados coletados pelo ambiente [115, 116]. Kara et al, (2021) [117] desenvolveram uma aplicação utilizando IA visando o reconhecimento de doenças em plantações, com auxílio da computação em nuvem. A aplicação pode enfrentar problemas de latência, uma vez que utiliza a computação em nuvem. Diferentemente, o estudo de caso 1 busca estabelecer o processamento de dados na borda, o

que faz com que os recursos de privacidade, segurança e estabilidade dos serviços sejam preservados para a inspeção das laranjas. Assim, evita problemas de comunicação comparados aos dispositivos e aplicações semelhantes que usam serviços em nuvem.

## 3.5 Estudo de caso 2: HAR

O reconhecimento de atividade humana, ou HAR, é a tarefa de perceber computacionalmente o comportamento de humanos. Os comportamentos em questão podem ser divididos em gestos, ações atômicas, atividades em grupo, interações, comportamentos e eventos [118]. Nesse contexto, os gestos são considerados movimentos primitivos das partes do corpo de uma pessoa. Ações atômicas correspondem a movimentos complexos. As atividades em grupo são um conjunto de tarefas realizadas por um grupo de indivíduos. Por sua vez, interações são atividades que envolvem dois ou mais indivíduos. Comportamentos são atividades influenciadas por emoções, personalidade ou estados psicológicos. Por fim, eventos são atividades que descrevem ações sociais e exibem intenções ou estruturas sociais.

As atividades humanas podem ser classificadas em duas categorias:

- Unimodais: na caracterização unimodal, todos os dados coletados correspondem à mesma modalidade de sensor, por exemplo, imagens ou áudio. O HAR unimodal é dividido entre os métodos de espaço-tempo, estocástico, e é baseado em regras e formas [118].
- Multimodais: o reconhecimento multimodal emprega dados de diferentes modalidades para realizar a tarefa de reconhecimento. O HAR multimodal é geralmente dividido entre os métodos afetivo, comportamental e social [118].

A solução proposta para o estudo de caso 2 utiliza as técnicas unimodais para a construção do dispositivo vestível para o HAR. As técnicas unimodais são geralmente baseadas em fluxo óptico ou séries temporais de sensores. Nas próximas seções, são explorados os aspectos do HAR e como eles permitem a criação de novas soluções inteligentes relacionadas aos contexto de esportes. Os principais conjuntos são o HAR baseado em visual e baseado em sensor, o que abrange novas tecnologias estimulando a criação e melhoria de métodos aplicados ao HAR [119].

### 3.5.1 HAR baseado em visão

O HAR baseado em visão se tornou um tópico importante na área de visão computacional [120]. Este conjunto de técnicas pode ser dividido em duas principais áreas:

métodos baseados em contato e métodos remotos. As técnicas baseadas em contato requerem contato físico com o dispositivo de aquisição [120]. Os autores avaliam que a maioria das soluções desenvolvidas são remotas, principalmente em aplicações mais recentes.

Beddiar et al, (2020) [120] afirmam que a HAR baseado em visão pode ser dividido em duas categorias principais. A primeira é por meio de recursos artesanais, e conta com extração de recursos pré-definidos e métodos de extração de segundo plano baseados em visão computacional. O treinamento desses modelos requer menos dados rotulados e geralmente é computacionalmente eficiente. A segunda categoria é a abordagem de aprendizado de recursos. Esses algoritmos exibem *pipelines* de ponta a ponta, desde o nível do *pixel* até as classes de atuação. Já existem algumas aplicações nesse contexto, como o monitoramento de pessoas em casas inteligentes [121] e a redução do consumo de energia elétrica em edifícios [122], mostrando o potencial do uso dessas soluções em diferentes setores, incluindo os esportes.

### 3.5.2 HAR baseado em sensores

O avanço tecnológico de novos sensores e a disseminação do IoT permitiram a criação de novas soluções para o HAR [123]. Segundo os autores existem vários problemas no HAR baseado em sensores, incluindo o custo computacional e privacidade dos dados. Além disso, afirmam que os algoritmos de DL têm um papel significativo no desenvolvimento de soluções, especialmente usando CNNs e RNNs. Uma abordagem baseada em sensores adiciona mobilidade, discrição e facilidade de uso, conforme afirmado por Nweke et al, (2018) [124]. Neste trabalho, os autores reforçam a importância das técnicas de DL, que podem executar uma etapa automática de extração de recursos, onde vários métodos são aplicados e os dividem em modo generativo, discriminativo e híbrido.

Atualmente, já são conhecidas várias aplicações para HAR baseado em sensor. Por exemplo, pode ser usado para classificar atividades do cotidiano (AVDs) [125], monitoramento de saúde [126], esportes [127], entre outros.

### 3.5.3 HAR baseado em IA

O tópico central no desenvolvimento de novas soluções neste contexto é o HAR baseado em IA. Desse modo, os algoritmos de DL se tornam uma importante ferramenta para o desenvolvimento de novas soluções HAR [123], tendo como um dos principais modelos usados em aplicações HAR as RNNs e CNNs.

Ramasamy et al, (2018) [128] afirmam que essas atividades começam com a premissa de extrair conhecimentos de dados brutos. Ademais, defendem que o uso de dados brutos no HAR é útil para monitoramento de saúde funcional e comporta-

mental, rastreamento de condicionamento físico e análise de esportes. Os autores mostram que várias iniciativas criaram conjuntos de dados contendo amostras de atividades humanas, além de imporem alguns dos modelos populares de DL em aparelhos como redes neurais profundas (DNNs), CNNs e RNNs, incluindo memória de longo prazo (LSTM).

Vários autores adotam essas técnicas para resolver tarefas destinadas ao HAR. Wan et al, (2020) [129] aplicaram e compararam vários algoritmos no desenvolvimento de uma proposta de aplicativo HAR baseado em *smartphone*. Zhou et al, (2020) [130] avaliaram como criar modelos de DL para HAR usando dados rotulados. Bianchi et al, (2019) [131] propuseram o uso de computação em nuvem para processar dados de IMU para HAR de vida ambiente assistida. Várias dessas aplicações empregam HAR baseado em sensores na perspectiva espaço-temporal. Como muitos desses autores mostram, a IA exerce um papel central no desenvolvimento de novos sistemas HAR.

### 3.5.4 HAR nos esportes

Finalmente, um último aspecto crucial na presente discussão é o uso de HAR nos esportes. Alguns autores empregam essas tecnologias na criação de aplicativos esportivos. Hsu et al, (2018) [132] propuseram o uso da tecnologia vestível e IA para os esportes e atividades diárias. Já Imram et al, (2022) [133] utilizaram uma rede convolucional rasa com o HAR baseado em sensores para o reconhecimento de atividades esportivas. Outros, Steels et al, (2020) [134] abordaram uma proposta semelhante para o *badminton* recreativo e profissional.

Também é possível encontrar alguns exemplos de dispositivos inteligentes para o HAR, por exemplo, relacionado ao reconhecimento de atividade de caminhada [135][136]. Além disso, os dispositivos destinados ao HAR possuem diversas aplicações na área da saúde [137], como a identificação de um distúrbio neurológico no usuário acometido pelo mal de Parkinson [138].

### 3.5.5 Considerações

O conceito de HAR é dividido em métodos unimodais e multimodais. Unimodais usam apenas um tipo de dado sensorial para realizar a classificação das atividades, como acelerômetro ou giroscópio. Por exemplo, pode-se usar apenas dados do acelerômetro para classificar atividades como caminhar. Essas técnicas podem ser mais simples e computacionalmente mais eficientes do que as multimodais, contudo, podem ter uma precisão mais limitada. Por outro lado, as multimodais usam vários tipos de dados sensoriais. Por exemplo, pode-se usar dados do acelerômetro, giroscópio e magnetômetro para identificar atividades como dirigir um carro. Essas

técnicas podem ser mais precisas em relação às unimodais, mas podem ser mais complexas e exigir mais recursos computacionais.

A escolha entre unimodal e multimodal para o HAR depende do conjunto de dados e das atividades que estão sendo reconhecidas, bem como dos recursos disponíveis para a computação. Além disso, os dois paradigmas principais do HAR são: baseados em sensores ou em visão. Os sensores mais comuns são acelerômetros, giroscópios e magnetômetros, capazes de coletar informações sobre o movimento e a posição do corpo humano. No entanto, a visão computacional também pode ser usada como uma fonte de dados para o HAR, embora seja menos comum.

A IA tem um papel central no desenvolvimento de novas soluções HAR. Nas abordagens clássicas, a IA depende de estágios manuais de extração de recursos, enquanto os aplicativos modernos já incorporam essa etapa no treinamento do modelo. As redes CNNs e RNNs são os dois tipos mais relevantes empregados ao HAR. As CNNs são capazes de aprender características hierárquicas, o que é útil para o HAR, dado que as atividades humanas podem ser compostas de vários movimentos ou gestos simultâneos. Já as redes RNNs são úteis para modelar sequências de dados, como séries temporais de sensores. Elas são capazes de levar em consideração informações do passado ao processar os dados de entrada, o que é importante para o HAR, pois as atividades humanas geralmente envolvem sequências de movimentos que se desenvolvem ao longo do tempo.

Bijalwan et al. [139] usam a fusão dos dados enviados por dois sensores: microsoft kinect V2; IMUs. Ainda assim, este sensor não tem a precisão e exatidão do BNO080, um sensor IMU 9DoF (9 graus de liberdade). Além disso, quatro BNO080 podem detectar com mais precisão qualquer inconsistência na marcha realizada por diferentes usuários. Sandro et al. [140] realizaram um *streaming* profundo para o HAR usando dados de pacientes na área da saúde. Apesar de algumas semelhanças com o nosso trabalho, eles se diferem na forma de entregar as informações. Em nosso trabalho o processamento acontece na borda do dispositivo evitando problemas de latência.

Em geral, muitas vezes essas ferramentas são utilizadas no desenvolvimento de novas aplicações voltadas para o esporte. Com este conjunto de funcionalidades, uma solução inovadora foi desenvolvida para o reconhecimento de atividade de caminhada, sendo um dispositivo com HAR unimodal baseado em sensores vestíveis. Além disso, o dispositivo utiliza quatro sensores, conforme ilustrado na Figura 4.9, que distingue de trabalhos normalmente encontrados na literatura, em que utilizam sensores simples para o HAR, como os sensores únicos [141]. Pode-se concluir que o HAR possui uma ampla gama de possibilidade de usos, podendo ser aplicado em processos industriais e fabris, área da saúde e no setor esportivo, o qual, inclusive, é o foco do texto.

# Capítulo 4

## Metodologia

No Capítulo 3 foram apresentados os conceitos necessários para a construção de um dispositivo vestível, assim como os algoritmos de IA aplicados neste contexto. Nas próximas seções deste capítulo são apresentados as propostas dos algoritmos de IA para integração em dois estudos de casos. O primeiro estudo apresenta um capacete inteligente para a inspeção de laranjas. Já o segundo, discorre sobre um dispositivo com sensores vestíveis para HAR. Desse modo, com base nos dois estudos de casos, são definidos os modelos de IA para a integração nestes sistemas, bem como os requisitos mínimos para o funcionamento.

### **4.1 Definição de Requisitos para a construção de um dispositivo vestível**

A revisão no Capítulo 3 apresentou os requisitos mínimos para o funcionamento de um sistema vestível, como a comunicação, robustez e autonomia energética [142]. A conservação desses recursos torna-se necessário, tendo em vista que serão utilizados em um ambiente externo. Além disso, os recursos energéticos desses dispositivos, como já discutido, são limitados, o que destaca a importância em definir melhores estratégias para a preservação da energia do dispositivo. Por fim, o dispositivo deve ser confortável para o usuário, para não afetar sua mobilidade ou condição física, e dessa forma, não comprometa as medições dos sensores e toda solução.

### **4.2 Estudo de caso 1: Inspeção de laranjas**

Nesta seção são apresentados os primeiros passos para construção do modelo de IA objetivando a classificação de imagens das laranjas. Primeiramente foi proposto um algoritmo de ML, utilizando a técnica de visão computacional para extrair as características das imagens. A metodologia de análise utilizada foi a denominada

pseudoespectral [143], na qual retira-se as informações do canal H de uma imagem no formato HSV. Os dados são então armazenados em um vetor de recursos para o treinamento do algoritmo. Existem abordagens semelhantes à do trabalho proposto, como o uso de câmeras multiespectrais para realizar triagem de frutas [144]. As câmeras multiespectrais contam com diversos sensores e captam ondas de diferentes comprimentos. Entretanto, o método pseudoespectral, o qual utiliza apenas um dos canais de cores, pode ser mais vantajoso para a aplicação nestes dispositivos, pois diminuem a quantidade de informações para análise e exige menos recursos do dispositivo.

A Figura 4.1 apresenta as quatro classes utilizadas no treinamento dos algoritmos para a inspeção de laranjas. Além disso, a base de dados utilizada [145] contém a mesma quantidade de imagens, conforme demonstrado na Tabela 4.1.

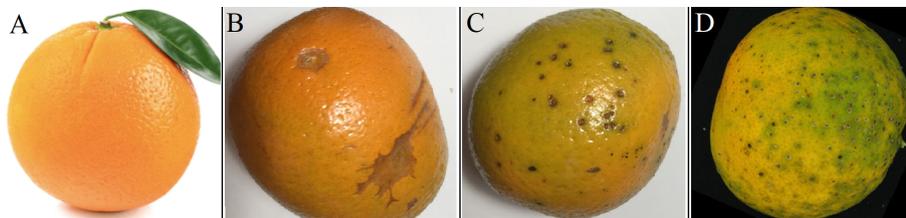


Figura 4.1: A) Frescas; B) Cancro; C) Pinta preta; D) Greening - Autoria própria.

Tabela 4.1: Descrição da base de dados para o estudo de caso 1.

Classe	Quantidade de imagens
Cancro	1002
Fresca	1446
Pinta preta	1030
Greening	1845

A Figura 4.2 apresenta o fluxo dos modelos algoritmos no treino e no teste para a classificação das imagens. No pré-processamento das imagens é realizado a conversão das imagens RGB para o espaço de cores HSV. Em seguida, é realizado a extração das informações presentes no canal H. As informações extraídas são armazenadas no vetor de características para a construção dos modelos. Com ajuda da biblioteca “scikit-learn”, quatro modelos de ML, baseando-se no aprendizado supervisionado, foram desenvolvidos para a realização dos testes de desempenho, dos algoritmos e dos dispositivos utilizados. Os modelos SVM, KNN, *Random Forest* e MLP foram escolhidos para as análises.

O algoritmo possui três etapas principais: (i) carregar a imagem na memória; (ii) extrair o vetor de características da imagem; (iii) classificar a imagem. Essas tarefas são repetidas por inúmeras vezes. Para realizar a análise de desempenho entre os classificadores dos modelos de ML foi utilizado três métricas padrão para ava-

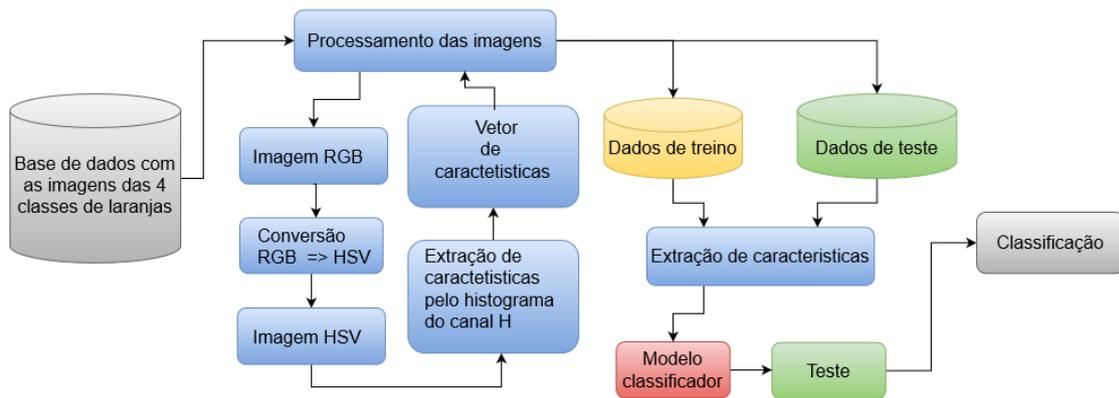


Figura 4.2: Processo de criação modelos de *machine learning* baseado em visão computacional - Autoria propria.

liação: Precisão, *Recall* e *F1-Score*. Tais parâmetros avaliam os seguintes aspectos do algoritmo:

- Precisão: mostra a quantidade de dados rotulados como positivos e que deveriam ser positivos. Em outras palavras, é um parâmetro de desempenho para o verdadeiro positivo (*TP*) rotulado corretamente pelo modelo. É representado pela equação 4.1

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

- O *Recall*: avalia a capacidade do sistema em encontrar todas as amostras positivas do conjunto. É representado pela equação 4.2, onde *FN* é o número de falsos negativos;

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

- *F1-score*: é calculado como a média harmônica ponderada entre precisão e *recall* dado pela equação 4.3:

$$F1 - Score = 2 \times \frac{Precisão \times Recall}{Precisão + Recall} \quad (4.3)$$

A métrica de avaliação final para o conjunto teste é denominada de *confusion matrix*. Esta matriz mostra a distribuição de acertos e erros do modelo após o treinamento. Para a integração do modelo de IA no *hardware* foi realizado uma padronização, essencial nesses dispositivos com baixa capacidade computacional. Para isso foi usado o PMML, um *framework* de persistência de modelo, de forma a evitar problemas de execução no *hardware* [146]. A biblioteca Nyoka<sup>1</sup> e o pypmml<sup>2</sup> auxiliaram a construção dos modelos de IA para a integração no dispositivo.

### 4.2.1 Algoritmo de Deep Learning

As redes profundas podem possuir muitas camadas, exigindo mais recursos de processamento dos sistemas para serem implementadas. Assim, no contexto dos dispositivos vestíveis, precisamos escolher modelos menos profundos e que preserve seus recursos. Diante disso, foi avaliado três possíveis candidatos para compor o sistema e funcionar como *backbones*. Os modelos utilizados, disponíveis no Keras [147], foram:

- **MobileNet-V2**: rede neural convolucional (CNN) comumente usada para resolver problemas de classificação [148]. Em suas camadas superiores, este modelo possui 14 MB.
- **NasNet-Mobile**: rede CNN leve usada também para problemas de classificação [149]. No entanto, possui 23 MB em suas camadas superiores.
- **ResNet50**: rede CNN comumente utilizada para o reconhecimento de imagens [150]. Este modelo tem 98 MB em suas camadas superiores.

As redes MobileNets diferem das redes de convolução tradicionais por serem pequenas e rápidas [67]. Por isso, o modelo DL torna-se uma opção poderosa para integração em aplicativos *mobile*, por exemplo, em tarefas de detecção de imagens como em doenças de frutas [151]. MobileNets menores são mais rápidas, pois usam multiplicadores de largura e resolução, negociando uma quantidade razoável de precisão para reduzir o tamanho e a latência [67].

A base da arquitetura NASNet está na estrutura *Neural Architecture Search* (NAS) [152]. Esse modelo de DL é flexível e escalável em diferentes aplicações. Na detecção de objetos, uma versão pequena do NASNet pode atingir 74% de precisão

<sup>1</sup><https://github.com/nyoka-pmml/nyoka>

<sup>2</sup><https://pypi.org/project/pypmml/>

top-1, que é superior a modelos de última geração de tamanho equivalente para plataformas móveis [153].

As redes ResNet [150] possuem uma estrutura de aprendizado residual, a qual facilita o treinamento das redes. Os autores concluíram que o empilhamento de camadas não degrada o desempenho da rede. Portanto, o modelo mais profundo não deve produzir um erro maior no treinamento comparado com os modelos mais rasos. Nesse caso, há a suposição de que deixar as camadas empilhadas se encaixarem em um mapeamento residual é mais fácil do que deixá-las se encaixarem diretamente no mapeamento de subjacência desejado. As ResNets possuem um alto desempenho, porém de acordo com a documentação do keras [15], essa rede possui uma grande quantidade de parâmetros, podendo sobrecarregar sistemas com baixo poder de processamento, por exemplo, um dispositivo vestível, sendo, portanto, uma limitação.

Os três modelos de DL citados foram utilizados para fins de comparação das diferentes arquiteturas. Na tarefa de classificação de imagens, em diferentes *hardwares*. A estrutura do modelo de previsão se inicia com uma camada de normalização para uma imagem de três canais. Em seguida os dados fluem pelo *backbone* do modelo. Por fim, a saída do *backbone* é achatada e passa por uma camada densa com 512 neurônios. A saída é uma camada densa com função de ativação *softmax*. A Figura 4.3 ilustra como esses *backbones* são integrados na construção do modelo.

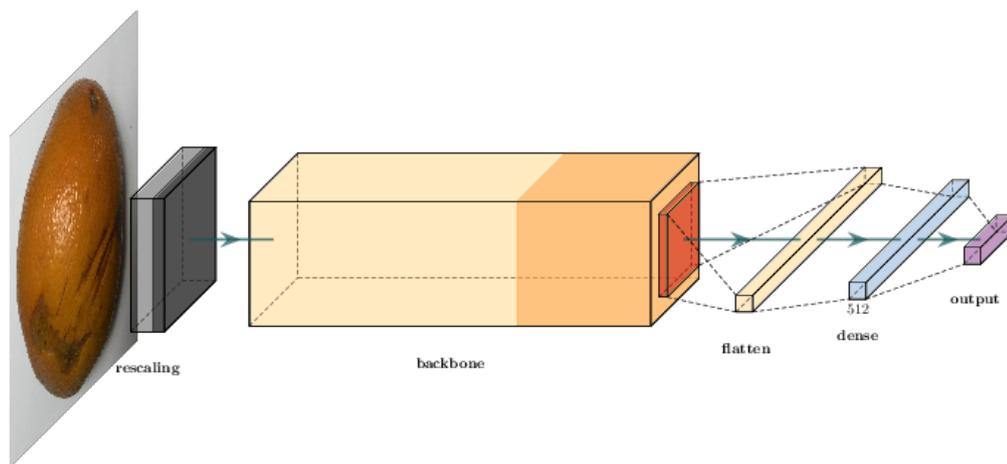


Figura 4.3: Ilustração da estrutura da rede de classificação - Adaptado [6].

O desempenho dos modelos de DL inicialmente é avaliado de acordo com as métricas utilizadas para os algoritmos de ML, precisão, *recall* e *F1-Score*. Posteriormente a avaliação é feita com base nas restrições de tempo, dos modelos com poucos parâmetros para as inferências das imagens. Também é realizada a medição do tempo médio de previsão de todo o conjunto de teste para cada modelo, comparando cada par de amostras, com uma análise de teste T.

O desempenho entre os modelos de ML e DL podem ser semelhantes, para a tarefa de classificação das imagens. Em razão disso, a restrição de tempo de inferência das imagens pode ser um fator diferencial na escolha do modelo que será integrado no dispositivo vestível. Com os testes mencionados torna-se possível a implementação destes algoritmos no dispositivo vestível, de forma que realizem o processamento na borda e executem a tarefa de classificação de laranjas sem eventuais problemas de processamento nos sistemas.

#### 4.2.2 Dispositivo vestível: Capacete inteligente

Soini et al, (2019) [21] desenvolveram um modelo de DL para detecção do *greening* em laranjas, utilizando técnicas de visão computacional. Os autores integraram o modelo e câmeras em veículos para realizar o monitoramento em pomares de laranjas. Apesar de ser um algoritmo eficiente para inspeção, a forma com que é aplicado pode não ser tão eficaz, uma vez que deve-se considerar que os veículos não circulam entre as árvores de laranjas, impedindo que a inspeção seja feita em toda a plantação. Dessa forma, na iminência de solucionar o problema de circulação, a decisão para a criação desse capacete inteligente foi em virtude da flexibilidade do usuário, o qual passa a poder circular de uma maneira mais precisa entre as laranjeiras e realizar um monitoramento com maior eficiência. Além disso, o capacete protege o citricultor contra possíveis quedas de frutos durante a realização da tarefa.

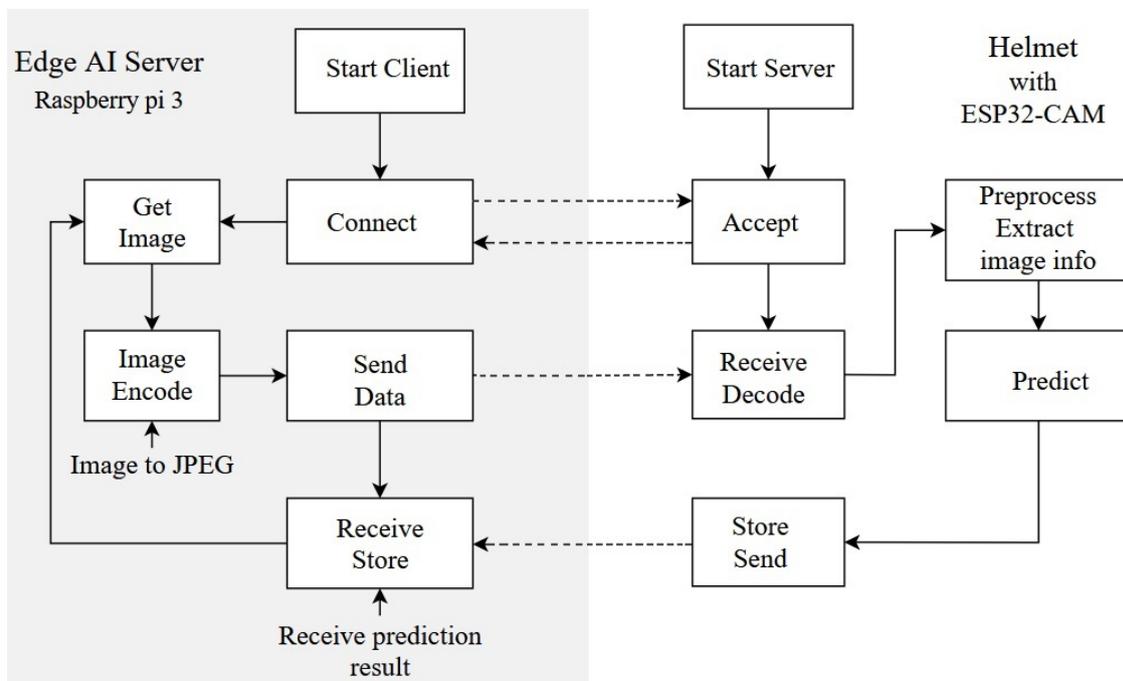


Figura 4.4: Fluxo de dados para o protótipo - Autoria própria.

O capacete inteligente foi projetado de acordo com os requisitos para a construção de um dispositivo vestível, com o objetivo de fornecer as informações necessárias ao

usuário de forma mais eficiente e com maior precisão. Essas vantagens tornam-se possíveis devido ao processamento de borda do sistema, que evita problemas de latência de comunicação. Para a preservação dos recursos do dispositivo foi proposto um sistema distribuído, conforme ilustrado na Figura 4.4, onde apresenta a comunicação de ambos os *hardwares* e o fluxo de dados do sistema.

### Visão geral do protótipo do dispositivo vestível

Neste trabalho foi proposto elaborar um sistema vestível distribuído com a transmissão de dados via *Wi-Fi*, projetado para realizar o processamento de dados na borda, do dispositivo, por meio de informações coletadas do ambiente. O módulo ESP-32 CAM coleta as imagens, com auxílio de técnicas de visão computacional. A Figura 4.5 apresenta uma proposta de arquitetura para esse sistema. O protótipo foi construído com base em equipamentos de segurança utilizados em locais de trabalho e pesquisas. Assim, para atender aos requisitos do sistema vestível, o capacete é apresentado como sistema físico.

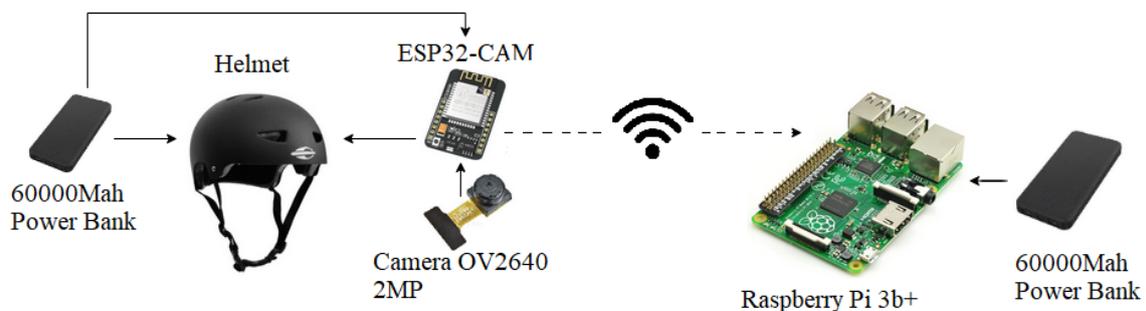


Figura 4.5: Visão geral da arquitetura de solução proposta - Autoria própria.

O módulo para processamento dos dados na borda, o Raspberry Pi 3B+ (Pi3), deve ler os dados transmitidos pelo módulo ESP-32, estabelecendo uma conexão estável. Esses dispositivos baseados na arquitetura ARM são soluções viáveis para esta proposta, pois apresentaram baixo consumo de energia. Como visto no Capítulo 3, os sistemas vestíveis têm restrições de recursos, portanto, precisam ser eficientes em termos de energia. Além disso, o padrão de rede de conexão *Wi-Fi* (IEEE 802.11) é utilizado para realizar a comunicação entre os dois módulos. Assim como o recurso de banda larga, a rede *Wi-Fi* garante qualidade ideal entre dispositivos, especialmente para *streaming* de câmera.

As imagens coletadas pelo ESP-32 CAM são enviadas para uma IA na borda do Pi3. O processamento acontece totalmente *offline*, sem utilizar recursos na nuvem, preservando os recursos de comunicação e de energia do dispositivo.

### Desenvolvimento e avaliação do Protótipo

Para testar as funcionalidades requeridas pela arquitetura proposta, foi construído um protótipo da solução, apresentado na Figura 4.6. Este protótipo contém

os elementos mencionados na seção anterior para inspeção das laranjas, por meio das informações coletadas pelo usuário. As informações coletadas alimentam um algoritmo de IA desenvolvido para o processamento na borda do dispositivo.

O módulo ESP32-CAM utilizado é um microcontrolador integrado com câmera OV2640 com resolução de 2 *megapixels*. O dispositivo tem a particularidade de ainda poder ser integrado a aplicativos IoT que permitem atividades avançadas, como rastreamento e reconhecimento de imagens. Este módulo usa uma CPU de 32 bits de núcleo único de baixo consumo e memória SRAM de 520 KB, com uma frequência de *clock* de até 160 MHz. Sua placa de rede suporta conexão sem fio 802.11b/g/n e Bluetooth 4.2 BLE. A taxa de transferência de imagem é de 15 a 60 fps. Possui uma entrada de fonte micro USB DC de 3,3V a 5V e 500mA. O ESP-32 foi integrado ao sistema para realizar apenas streaming de dados sem fio, assim, preservando a autonomia energética do sistema e melhorando o processamento de IA no dispositivo de borda.



Figura 4.6: O protótipo - Autoria própria.

O Pi3 foi usado para processar os dados coletados. A CPU deste módulo é uma Broadcom BCM2837B0, contendo um processador Cortex-A53 (ARMv8) Quad-Core, com frequência de *clock* de até 1,4 GHz, 1 GB de RAM, com código e dados armazenados em cartão SD. Sua placa de rede suporta conexão sem fio 802.11e *Bluetooth* 4,2, BLE e alimentação de 5V e 2,5A.

A Figura 4.6 mostra o protótipo, um capacete utilizado em atividades perigosas na natureza. Este equipamento pode ser substituído por um modelo padrão de capacete, de segurança, utilizado em áreas onde há risco de queda de objetos sobre o usuário, como queda de galhos ou laranjas.

#### **Testes de validação de hardware para a construção do protótipo**

Como visto nas subseções anteriores e na revisão da literatura, os *hardwares* uti-

lizados para a construção dos dispositivos vestíveis possuem limitações operacionais para realizar tarefas específicas. Portanto, é necessário validar o protótipo para se ter conhecimento das limitações inerentes ao mesmo. Para a validação, além do Pi3, outras duas plataformas são e comparadas:

- **Sipeed M1:** Com um chip IA K210, Cpu: RISC-V Dual Core 64bit, FPU, 400Mhz, 8M SRAM, OV2640 câmera, LCD de 2,4 polegadas com resolução de  $320 \times 240$  *pixels*;
- **Raspberry Pi Zero W:** 1GHz Single Core ARMv6 CPU (BCM2835) Video Core IV GPU, 512MB RAM.

A plataforma Sipeed é avaliada com o modelo DL integrado, comparando o consumo de energia com o sistema proposto. No Raspberry Pi Zero W, além do consumo energético, o fator de utilização com dois módulos de captura de imagem, a câmera USB e uma câmera Pi com resolução de  $3280 \times 2464$  *pixels*, são também comparados.

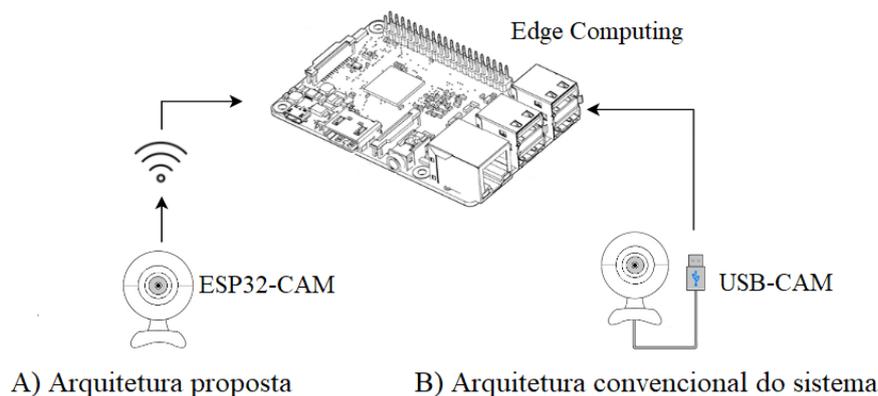


Figura 4.7: Arquiteturas propostas para o estudo de caso - Autoria própria.

A Figura 4.7 apresenta duas arquiteturas para validação do sistema com base na execução dos modelos de IA. Na Figura 4.7-A os dados são coletados pelo ESP32-CAM e enviados via *Wi-Fi* para o Pi3. Já na Figura 4.7-B a câmera USB de 2 *megapixels* é conectada ao Pi3W, caracterizando um sistema não distribuído.

A ferramenta psutil (*python system and process utilities*) foi escolhida para criar um perfil do sistema e um multímetro para verificar o fluxo de corrente elétrica ao executar a IA no dispositivo. Com isso, pode-se realizar uma análise do perfil do dispositivo, monitorar o consumo de processamento da CPU, gasto de memória e o consumo energético. Além disso, a fim de definir se há diferenças entre as amostras de dados, foi realizado uma análise estatística nas médias das duas amostras. A partir desses dados, foi estabelecido as seguintes hipóteses:

- H0: As amostras provavelmente são retiradas das mesmas distribuições. Se verdadeiro, falha ao rejeitar H0.
- H1: As amostras provavelmente são retiradas de diferentes distribuições. Se H1 for aceita, rejeita-se H0.

Como não sabemos se a distribuição dos dados é normal, foi utilizado um teste não paramétrico, *Wilcoxon Signed-Rank Test*, para comparar estatisticamente nossos dados. Esperamos que a arquitetura proposta na Figura 4.5 obtenha um desempenho superior à arquitetura com a câmera USB integrada ao sistema. Assim, por ser uma arquitetura distribuída, reduziria o custo de processamento e melhoraria a autonomia energética do sistema primário onde os dados são processados com IA integrada.

### 4.3 Estudo de caso 2: HAR

Nas próximas subseções são apresentados os sensores, *hardwares* e os aspectos para o desenvolvimento do dispositivo vestível para o HAR. A Figura 4.8-A ilustra a etapa de aquisição de dados por meio de sensores vestíveis. A Figura 4.8-B apresenta o pré-processamento dos dados, como o formato entregue para o treinamento e validação dos algoritmos de IA.

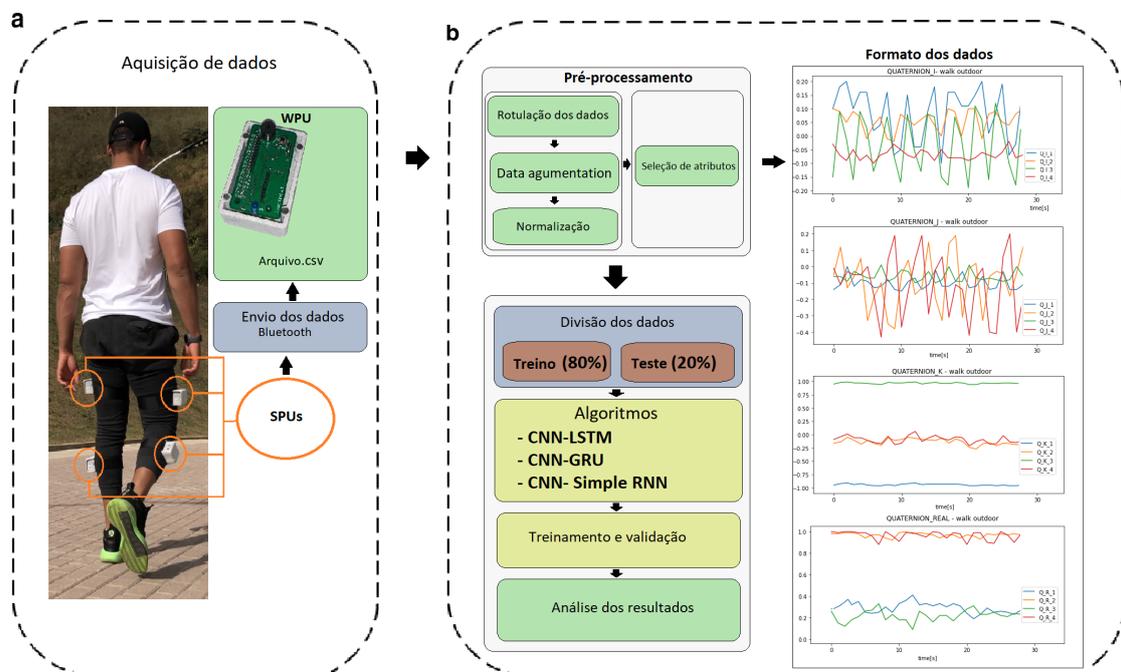


Figura 4.8: Proposta do dispositivo vestível para o HAR - Autoria própria.

### 4.3.1 Sensores vestíveis para o HAR

Os dados usados ao longo deste projeto foram coletados e pré-processados usando uma solução vestível específica anexada a parte inferior do corpo do indivíduo. O dispositivo é composto por um conjunto de IMUs de alto desempenho para coletar o movimento físico das pernas do usuário.

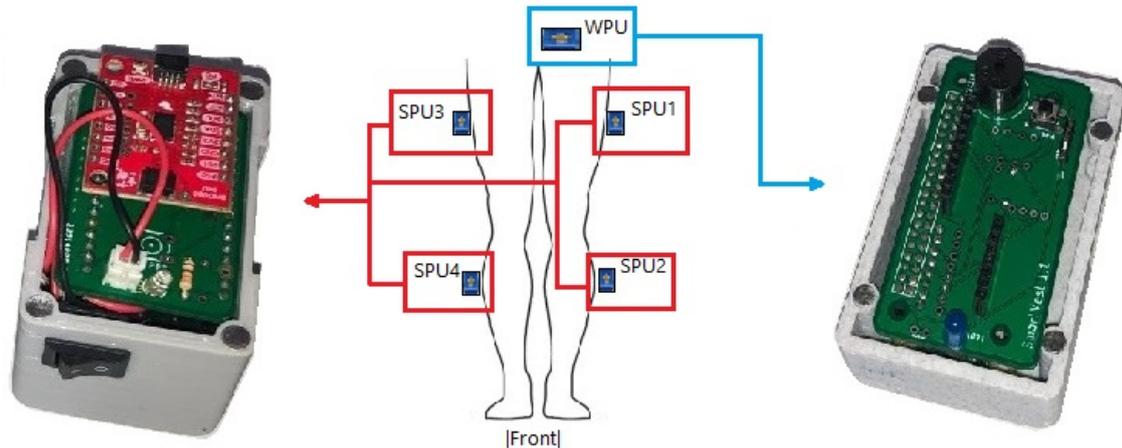


Figura 4.9: Dispositivo vestível usado para coletar dados de movimento do indivíduo. As áreas destacadas indicam o local onde cada dispositivo está posicionado - Autoria própria.

Dentro deste escopo de projeto, duas classes distintas de *hardware* foram utilizadas para compor a solução final: 1) Unidade de Processamento Sensor (SPU) e a Unidade de Processamento Vestível (WPU). A Figura 4.9 representa a posição de cada elemento quando fixado ao corpo do indivíduo. Resumidamente, os dados são coletados de forma distribuída por meio das SPU e, em seguida, encaminhados para a WPU. Na WPU, as informações recebidas podem ser pré-processadas e enviadas para um servidor externo ou armazenadas localmente em um cartão SD (memória *flash*).

#### SPU

A SPU usa quatro unidades diferentes fixas às pernas do usuário com o objetivo de coletar as informações brutas em tempo real das atividades realizadas pelo indivíduo. Cada uma dessas unidades incorpora o seguinte *hardware*:

- BNO080 IMU: Sensor inercial de 9 graus compreendendo leituras de acelerômetro, giroscópio e magnetômetro. É usado para recuperar a orientação física das partes do corpo;
- Bateria de íon de lítio;

- NodeMCU ESP-32: Plataforma de *hardware* baseada na solução Espressif ESP-32. É responsável por ler os dados detectados pela IMU e encaminhá-los continuamente para o hardware WPU usando a interface *Bluetooth*.

## WPU

A WPU é composta por um *hardware* o qual é responsável por algumas funções específicas, sendo elas: receber os dados coletados pelas SPUs; sequenciar o *timestamp* de cada pacote recebido; armazenar os dados localmente para posterior análise ou enviá-los para um servidor/serviço remoto utilizando o padrão IEEE 802.11, por uma interface sem fio. A WPU incorpora os seguintes componentes:

- Raspberry Pi Zero W:
- Sensores de umidade e temperatura:
- BNO080 IMU: Sensor inercial de 9 graus compreendendo leituras de acelerômetro, giroscópio e magnetômetro. Tem a particularidade de recuperar a orientação física da parte superior do corpo, temperatura e umidade.

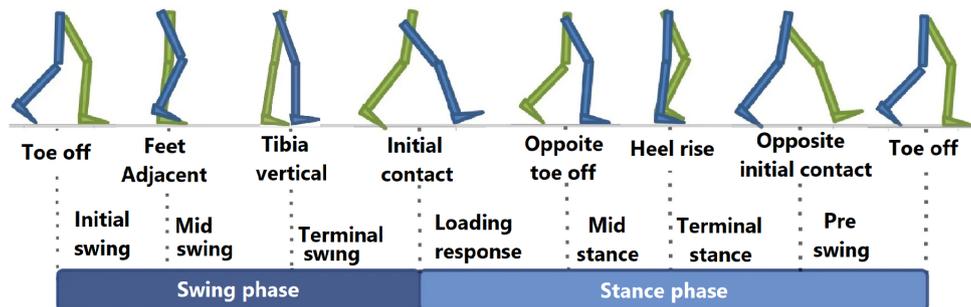


Figura 4.10: Ciclo completo de caminhada [7].

### 4.3.2 HAR algoritmos de IA

Existem diversos algoritmos de IA aplicados para o HAR. Entretanto, neste estudo de caso, foi utilizado apenas três modelos, sendo estes comumente empregados para análise de séries temporais. Os algoritmos são:

- *Long short-term memory* (LSTM): redes neurais recorrentes (RNN) capazes de classificar dados sequenciais, devido às suas características de armazenamento, memória e aprendizagem [154] [155]. Estas redes são comumente usadas para reconhecimento de eventos, por análise de séries temporais, como em HAR [135].

- *Gated Recurrent Unit* (GRU): Introduzido por Cho et al, (2014), o GRU visa resolver o problema de dissipação de gradiente, cujo é comum em uma rede neural recorrente padrão. O GRU também pode ser considerado uma variação do LSTM porque ambos são projetados de forma semelhante [156].
- *Simple RNN*: A *Simple RNN* usa dados sequenciais ou dados de séries temporais. Portanto, o RNN trabalha com o princípio de salvar a saída de uma camada específica e realimentar a entrada para prever a saída da camada [157].

Os dados das SPUs são enviados para o WPU com o decorrer do tempo. Com isso, a fusão dos dados dos quatro sensores forma um evento específico ao longo do tempo. Essa condição permite que os algoritmos de IA classifique os dados dentro de um intervalo da série temporal, indicando assim a atividade realizada pelo usuário. Para a avaliação do algoritmos de IA neste estudo de caso, as métricas, como Precisão, *Recall* e *F1-Score*, anteriormente descritas na seção 4.2, também são utilizadas.

### 4.3.3 Criação da base de dados

A base de dados [158] foi criada para avaliar o uso do dispositivo vestível e treinar modelos de IA. A Tabela 4.2 apresenta as quatro classes coletadas para esta base de dados.

Tabela 4.2: Descrição da base de dados para o estudo de caso 2.

Classe	Descrição
seated	Dados coletados em uma posição sentada fixa em uma cadeira comum.
stand	Dados coletados na posição em pé.
walking	Dados coletados na rua com velocidade média de 3km/h.
iwuphill	Dados coletados em ambiente interno, equipamento de caminhada com inclinação e velocidade de 5 km/h.

As quatro SPUs são responsáveis pelo envio de dados para a WPU. Os dados recebidos pela WPU são um sinal composto com as informações espaciais correspondentes aos quatérnions, sendo I, J, K, REAL e Radianos. Uma observação importante é que a frequência de amostragem de dados pode ser inconstante, apesar de os quatro sensores terem os mesmos aspectos construtivos. A variação na amostragem pode ser devido a aspectos de cada um dos componente eletrônicos que compõem o *hardware*.

Segundo Ravi et al, (2005) [159], para modelar problemas relacionados a atividades humanas, a taxa de amostragem deve estar na faixa de 50Hz. Essa faixa significa que não temos problemas com a aplicação de modelos de IA, porque assim garantimos a normalização dos dados de entrada, para treinamento dos algoritmos. Além disso, para este estudo, os dados foram coletados por um único usuário, para que possamos garantir a homogeneidade das informações.

Os quatro sensores das SPUs foram posicionados na perna do usuário, conforme mostra a Figura 4.9. Os dados coletados pelo WPU têm quatro componentes, I, J, K e R (REAL). A Tabela 4.3 mostra os nomes destas componentes para análise, respectivamente.

Tabela 4.3: Descrição dos dados enviados pelos sensores SPU.

Quatérnion	ID	Componente
Q-I	ID	Q-I-ID
Q-J	ID	Q-J-ID
Q-K	ID	Q-K-ID
Q-R	ID	Q-R-ID

### Base de dados - *Data augmentation*

Os dados foram coletados por uma única pessoa para garantir homogeneidade amostral. Entre uma coleta de dados e outra, houve uma pausa intencional de alguns dias para recuperação do usuário e a coleta das informações da forma mais homogênea possível. Nesse contexto, os dados coletados foram em baixa quantidade. Contudo, para resolver o problema da quantidade, operações matemáticas foram feitas no banco de dados original para aumentar o número de instâncias de cada classe.

A Figura 4.11 e a Figura 4.12 demonstram as operações realizadas no quatérnion J de uma instância da classe *walk*. Este procedimento foi realizado em todas as classes do conjunto de dados original. Quatro operações foram realizadas em todos os quatérnions das quatro classes do conjunto de dados original:

- **Noise:** Adiciona ruído aleatório à série temporal de cada quatérnion . O ruído adicionado em cada ponto de tempo é distribuído de forma independente e idêntica;
- **Convolve:** Convolve séries temporais com uma janela de kernel;
- **Quantize:** Quantize séries temporais para um conjunto de níveis. Os valores são arredondados para o nível mais próximo no conjunto de níveis;
- **Drift:** Desvio do valor da série temporal de forma aleatória e suave. A extensão do Drift é controlada pelo Drift máximo e pelo número de pontos de Drift.

Foi utilizado a biblioteca TSaug [160] para aumentar os dados. A primeira operação foi feita com a adição de duas variações, 0,1 e 0,5. A segunda aplicação foi a operação de convolução, com variação de 10 e 5. Terceira, a quantização, com duas variações de 10 e 20. Por fim, foi adicionado o *Drift* com quatro variações:

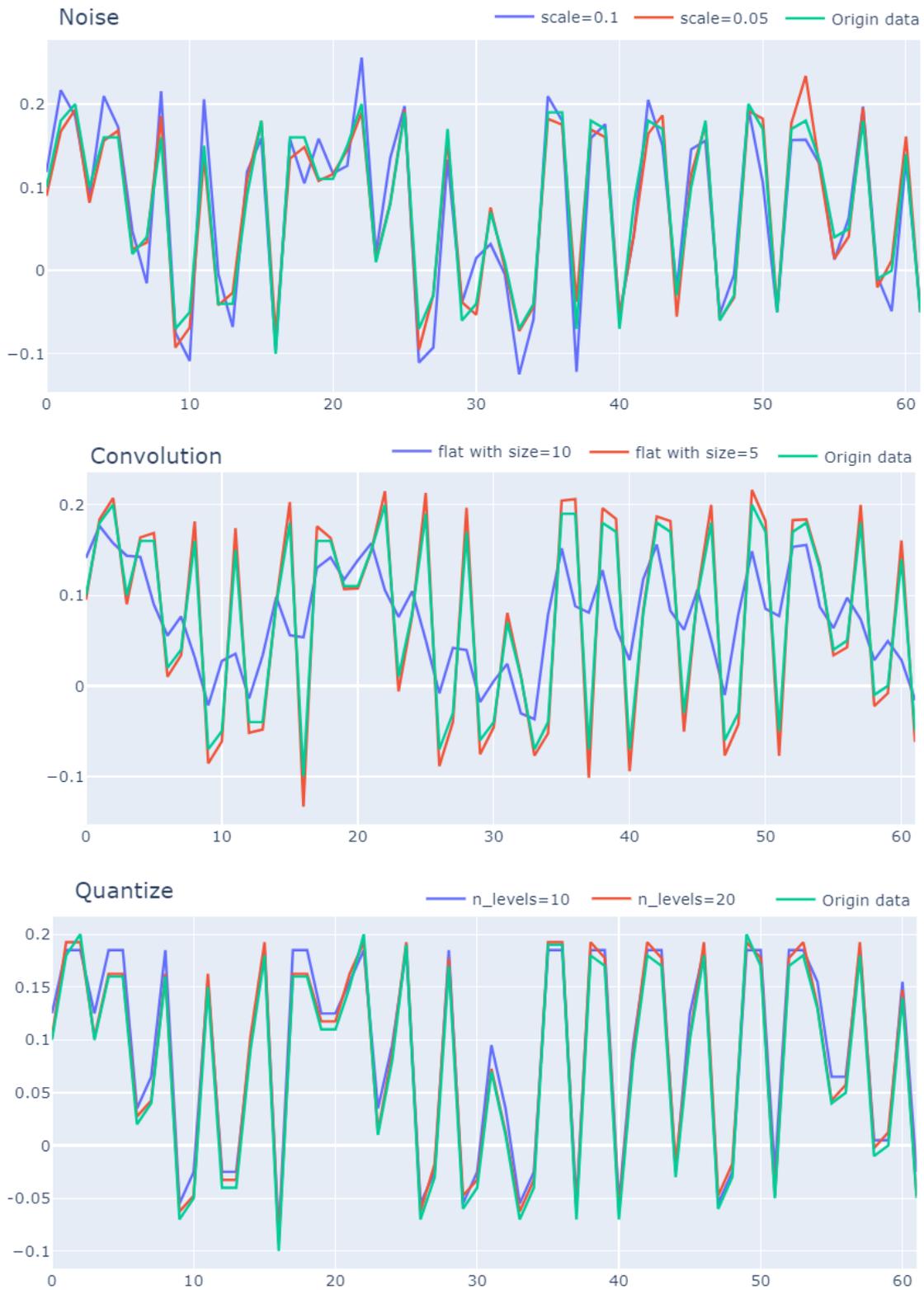


Figura 4.11: *Data augmentation: Noise; Convolution; Quantize* - Autoria própria.

- *max drift* = 0,2 e *number drift points* = 5;
- *max-drift* = 0,5 e *number drift points* = 5;

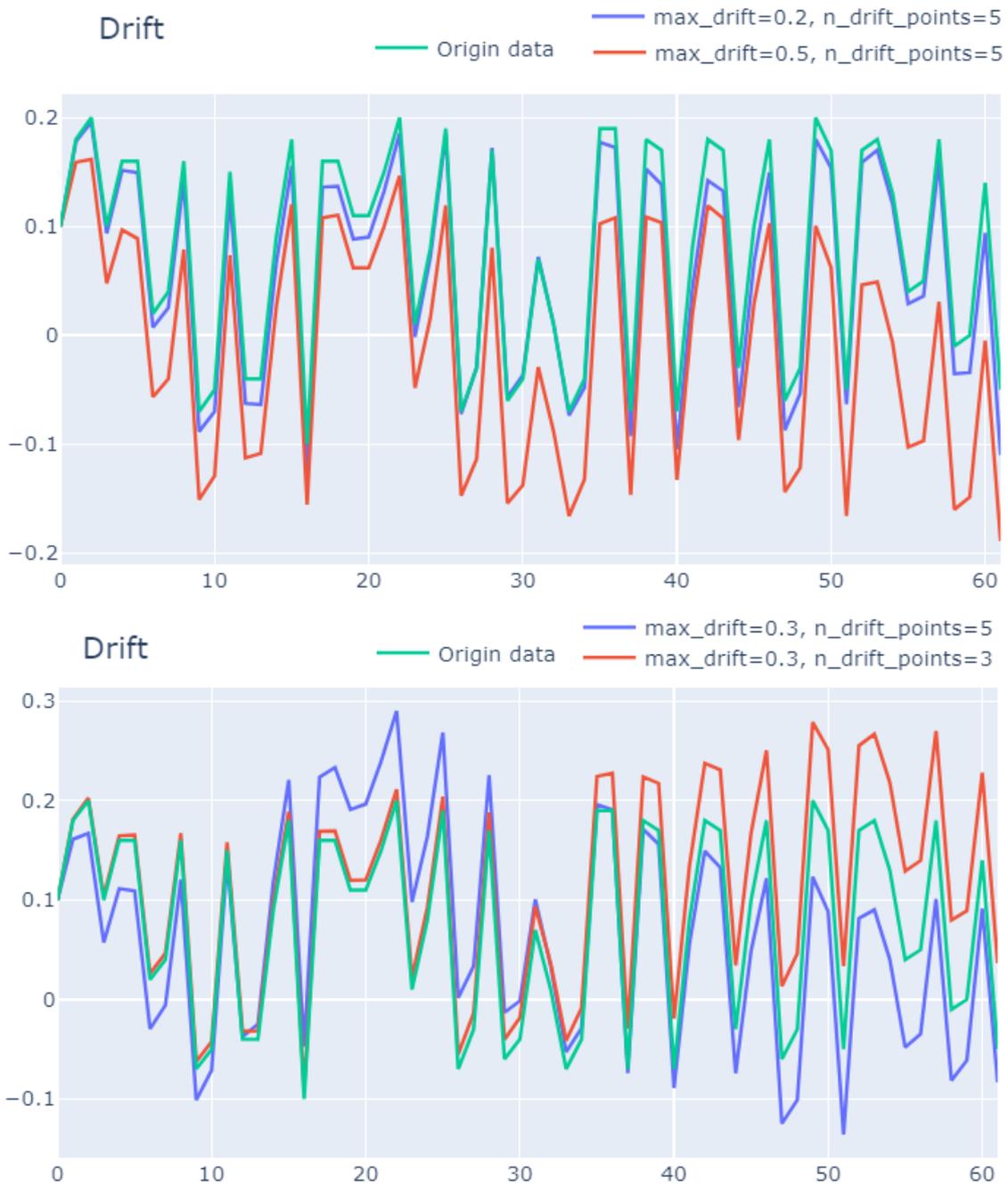


Figura 4.12: *Data augmentation: Drift* - Autoria própria.

- $max\text{-}drift = 0,3$  e  $number\ drift\ points = 5$ ;
- $max\text{-}drift = 0,3$  e  $number\ drift\ points = 3$ .

No *dataset* com o *data augmentation* [161], dez novos dados foram gerados, a partir de cada instância do conjunto de dados original. Desse modo, a ampliação da base de dados faz com que o algoritmo evite possíveis problemas no treinamento, por exemplo, sobreajuste *overfitting*.

# Capítulo 5

## Resultados

Este capítulo apresenta os resultados dos algoritmos de IA desenvolvidos para a implementação nos dispositivos vestíveis, com base nos estudos de casos descritos nos capítulos anteriores. Primeiramente são apresentados os resultados dos testes de validação para o estudo de caso 1. Em seguida, as conclusões preliminares nas etapas construtivas dos algoritmos, bem como o funcionamento do protótipo para a inspeção de laranjas, com uma proposta de arquitetura descentralizada. Além disso, apresentamos os resultados dos dados coletados e treinados pelo dispositivo vestível destinado ao HAR.

### 5.1 Resultados para o Estudo de Caso 1

Nas próximas seções são apresentados os resultados dos testes dos algoritmos de IA para a inspeção de laranjas, além do protótipo do capacete inteligente desenvolvido neste contexto. Nesta etapa foram estudados modelos de ML e DL para classificação das imagens. O uso dos dois modelos foi necessário para fornecer uma linha base para realizar a classificação das laranjas, bem como realizar as análises a fim de verificar qual modelo possui maior viabilidade de integração no dispositivo vestível, a partir dos requisitos dos *hardwares* utilizados para testes. Os algoritmos de ML foram treinados em um computador pessoal, com processador Intel i7 de 9<sup>a</sup> geração, placa de vídeo GTX 1660TI, 16 GB de RAM e os dados armazenados em um SSD. Já os algoritmos de DL foram treinados no google colab (ambiente de desenvolvimento em nuvem do google).

#### 5.1.1 Algoritmos de Machine Learning

Os primeiros testes aplicados foram as métricas de avaliação para os modelos de ML e DL candidatos a integrar o dispositivo. Conforme já mencionado no Capítulo 4, três métricas foram avaliadas: Precisão, *Recall* e *F1-Score*. Além disso, a acurácia global

também foi analisada. A Tabela 5.1 exibe os resultados do processo de treinamento. Todos os algoritmos submetidos à avaliação apresentaram acurácia geral acima de 90%.

Tabela 5.1: Teste com o modelo para classificação de múltiplas classes

	<b>Precision</b>	<b>Recall</b>	<b>F1-Score</b>	<b>Support</b>
<b>SVM:</b>				
Frescas	1,00	1,00	1,00	149
Cancro	0,91	0,88	0,89	98
Greening	1,00	1,00	1,00	179
Pinta Preta	0,89	0,92	0,91	110
Acurácia = 96.08%				536
<b>KNN:</b>				
Frescas	1,00	0,98	0,99	141
Cancro	0,88	0,92	0,90	92
Greening	1,00	1,00	1,00	208
Pinta Preta	0,90	0,88	0,89	95
Acurácia = 96.0%				536
<b>Random Forest:</b>				
Frescas	1,00	1,00	1,00	127
Cancro	0,94	0,87	0,90	105
Greening	1,00	1,00	1,00	191
Pinta Preta	0,88	0,95	0,91	113
Acurácia = 96.26%				536
<b>MLP:</b>				
Frescas	1,00	1,00	1,00	154
Cancro	0,94	0,96	0,95	96
Greening	1,00	1,00	1,00	188
Pinta Preta	0,96	0,94	0,95	98
Acurácia = 98.13%				536

Analisando a Tabela 5.1, observa-se que os modelos apresentaram dificuldades em identificar as classes cancro e pinta preta. O KNN teve a menor precisão comparado aos outros três modelos para classificar a classe cancro. O *Random Forest* obteve a segunda melhor acurácia, sendo de 96,26%. No entanto, em relação ao *Recall*, a classe cancro obteve o pior desempenho entre as classes dos outros modelos.

O MLP foi o modelo com melhor desempenho, com uma precisão geral de 98,13%. Diferente dos outros algoritmos, o MLP usa a técnica de rede neural de múltiplas camadas, sendo o grande diferencial deste algoritmo classificador. Além disso, apresenta um equilíbrio entre os parâmetros, Precisão, *Recall* e *F1-Score*, entre as quatro classes de laranjas, as quais foram usadas para o treino e teste neste contexto.

Nas Tabelas 5.2, 5.3, 5.4 e 5.5 são apresentadas as *confusion matrix* de cada modelo de classificação que complementam as conclusões anteriores. Na classificação da classe cancro percebe-se que os modelos apresentaram uma maior dificuldade, o

que pode estar relacionado à menor quantidade de amostras disponíveis para a realização dos testes nesta classe.

Tabela 5.2: *Confusion Matrix* - SVM

	Frescas	Cancro	Greening	Pinta preta
Fresca	97,0 %	1,0 %	1,0 %	1,0 %
Cancro	1,0 %	88,6 %	1,0 %	10,4 %
Greening	1,0 %	1,0 %	98,3 %	1,0 %
Pinta preta	1,0 %	9,27 %	1,0 %	87,8 %

Tabela 5.3: *Confusion Matrix* - KNN.

	Frescas	Cancro	Greening	Pinta preta
Fresca	97,8 %	1,0 %	1,0 %	2,1 %
Cancro	1,0 %	86,7 %	1,0 %	7,4 %
Greening	1,0 %	1,0 %	98,5 %	1,0 %
Pinta preta	1,0 %	11,2 %	1,0 %	89,4 %

Tabela 5.4: *Confusion Matrix* - *Random Forest*.

	Frescas	Cancro	Greening	Pinta preta
Fresca	97,7 %	1,0 %	1,0 %	1,0 %
Cancro	1,0 %	88,7 %	1,0 %	11,38 %
Greening	1,0 %	1,0 %	98,5 %	1,0 %
Pinta preta	1,0 %	5,7 %	1,0 %	86,9 %

Tabela 5.5: *Confusion Matrix* - MLP.

	Frescas	Cancro	Greening	Pinta preta
Fresca	98,0 %	1,0 %	1,0 %	1,0 %
Cancro	1,0 %	92,0 %	1,0 %	4,0 %
Greening	1,0 %	1,0 %	98,4 %	1,0 %
Pinta preta	1,0 %	6,0 %	1,0 %	93,8 %

Os modelos de ML também foram analisados com base nas restrições de tempo para realizar as três etapas principais do algoritmo: aquisição da imagem, extração para o vetor de características e previsão das classes [162]. Os autores usaram um Raspberry Pi3 B+ para a análise de desempenho. Dentre as análises realizadas, o modelo com melhor desempenho foi o *Random Forest*, com 7,63 previsões/segundos, (p/s), em seguida o MLP e o SVM, com 6,17 p/s e 4,95 p/s respectivamente. O KNN, dentre todos, teve o pior desempenho de tempo médio para realizar esta tarefa, com 1,78 p/s. O resultado reforça que os principais candidatos para integrar o dispositivo vestível são o MLP e o *Random Forest*, com base no desempenho para a tarefa de classificação de imagens de laranjas.

## 5.1.2 Algoritmos de Deep Learning

Nesta seção apresentamos os resultados obtidos no treinamento dos modelos de DL para detecção das doenças presentes na laranja, *frame a frame*, classificando uma única laranja por vez. Além disso, os resultados do MobileNetV2 são comparados com o ResNet50 e a NasNetMobile.

### Treinamento

Inicialmente as redes foram treinadas com auxílio da API do Keras como linha de base para criar os algoritmos. Os parâmetros de treinamento usam a função de perda de entropia cruzada categórica e o otimizador Adam. Como o conjunto de teste é usado apenas como indicador, o mesmo foi empregado nas etapas de validação. Na Figura 5.1 estão representados os resultados das sessões de treinamento para cada modelo. Nesta figura nota-se que os resultados não apresentam sinais de *overfitting*, com uma convergência satisfatória para cada modelo.

Na segunda etapa da construção dos modelos de DL foram utilizadas as métricas de classificação de ML para avaliação, sendo a Precisão, *Recall* e o *F1-score*. Além disso, foi avaliada a média global para cada caso. A Tabela 5.6 exibe as métricas para o MobileNetV2, ResNet50 e o NasNetMobile. Os resultados indicam que o NasNet-Mobile teve um desempenho um pouco pior, quando comparado com o MobilenetV2 e o ResNet50.

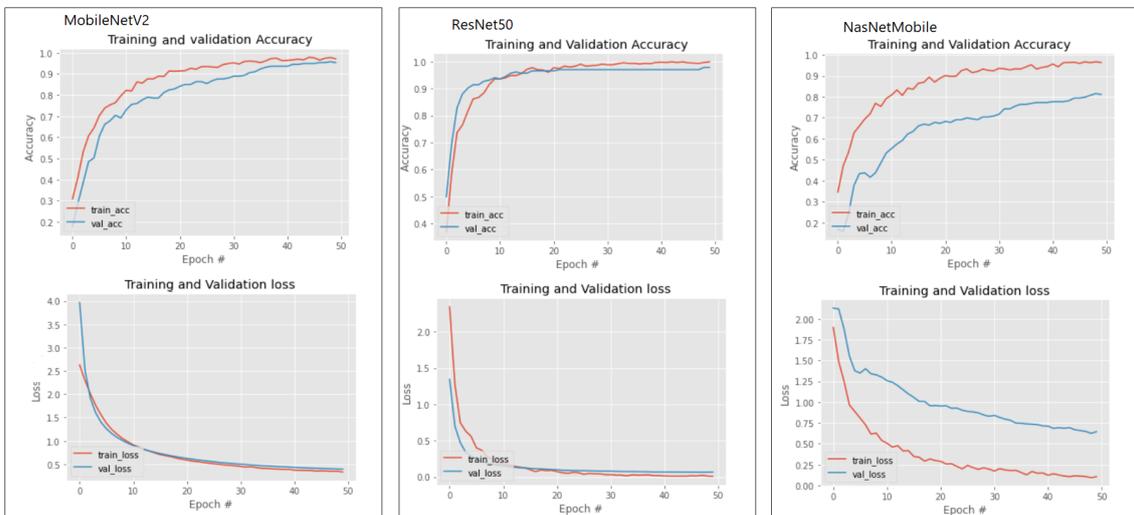


Figura 5.1: Avaliação dos valores de acurácia e perda dos conjuntos de treinamento e validação.

Os modelos de DL foram treinados com 50 épocas para compor o conjunto de dados utilizado, já apresentado no Capítulo 4. Os modelos utilizam uma taxa inicial de treinamento de  $1e-4$  e *batch size* igual a 32. Além destes, foi utilizado pesos da *imagenet* em cada um dos modelos. A estrutura dos modelos segue a seguinte base: primeira camada *AveragePooling2D(pool-size=(7, 7))*; segunda camada *Flat-*

*ten(name="flatten");* uma camada *Dense(128, activation="relu");* camada de *Dropout(0.5)* e a camada de ativação *Dense(2, activation="softmax")*.

Tabela 5.6: Teste com modelo para detecção de laranja.

	<b>Precision</b>	<b>Recall</b>	<b>F1-Score</b>	<b>Support</b>
<b>MobileNetV2:</b>				
Black Spot	0,84	0,90	0,87	41
Citrus Canker	0,89	0,85	0,87	40
Fresh Oranges	1,00	0,99	0,99	78
Greening	1,00	1,00	1,00	74
<b>Macro average</b>	0,93	0,93	0,93	233
<b>Weighted average</b>	0,95	0,95	0,95	233
<b>Global Accuracy:</b>	95%			
<b>ResNet50:</b>				
Black Spot	0,97	0,93	0,95	41
Citrus Canker	0,97	0,97	0,94	40
Fresh Oranges	1,00	1,00	0,99	78
Greening	1,00	1,00	1,00	74
<b>Macro average</b>	0,97	0,97	0,97	233
<b>Weighted average</b>	0,98	0,98	0,98	233
<b>Global Accuracy:</b>	98%			
<b>NasNetMobile:</b>				
Black Spot	0,65	0,85	0,74	41
Citrus Canker	0,50	0,03	0,05	40
Fresh Oranges	0,78	1,00	0,88	78
Greening	0,96	1,00	0,98	74
<b>Macro average</b>	0,72	0,72	0,66	233
<b>Weighted average</b>	0,77	0,81	0,74	233
<b>Global Accuracy:</b>	81%			

Na Tabela 5.6, vemos que os modelos treinados alcançaram uma alta precisão nos testes para a classificação das imagens de laranjas. As métricas mostram índices acima de 80% para Precisão, *Recall* e o *F1-Score* de cada um dos modelos. Entre os três modelos treinados, o ResNet50 alcançou a maior acurácia global, de 98%, no entanto, é importante destacar que possui a maior quantidade de parâmetros. Desse modo, o modelo candidato para integração no dispositivo vestível é o MobileNetV2, uma vez que além de possuir menor quantidade de parâmetros comparados aos outros dois modelos, alcançou uma acurácia global próxima ao ResNet50, que é modelo mais profundo entre os analisados.

#### **MobileNetV2 X ResNet50**

Ambos os modelos foram treinados visando realizar a comparação entre uma rede com poucos parâmetros (3,5 milhões) e outra com muitos parâmetros (25,6 milhões). A rede ResNet50 possui aproximadamente 7x mais parâmetros que a MobileNetV2. Os dois modelos de classificação usam uma estrutura baseada em Keras, axelerate

para IA na borda [163], que auxilia na velocidade de construção do modelo, na etapa de treinamento.

As Tabelas 5.7 e 5.8 apresentam os resultados obtidos nos testes com os modelos MobileNetV2 e ResNet50. Os dois modelos tiveram bons resultados, com poucos erros para classificar as instâncias.

Tabela 5.7: *Confusion Matrix* - MobileNetV2.

	Fresh	Canker	Greening	Black Spot
Fresh	98,0 %	0,0 %	0,0 %	2,0 %
Canker	0,0 %	100,0 %	0,0 %	0,0 %
Greening	0,0 %	0,0 %	100,0 %	0,0 %
Black Spot	2,2 %	0,0 %	0,0 %	98,8 %

Tabela 5.8: *Confusion Matrix* - ResNet50.

	Fresh	Canker	Greening	Black Spot
Fresh	94,0 %	6,0 %	0,0 %	0,0 %
Canker	2,0 %	98,0 %	0,0 %	0,0 %
Greening	0,0 %	0,0 %	100,0 %	0,0 %
Black Spot	0,0 %	0,0 %	0,0 %	100,0 %

### MobileNetV2 X NasNetMobile

Estas duas redes foram escolhidas por serem menos profundas e possuírem menos parâmetros que outras disponíveis na documentação do Keras, como a ResNet50. Com base neste estudo de caso, observou-se que as redes menos profundas fornecem soluções adequadas dentro do contexto dos dispositivos vestíveis e no processamento na borda do dispositivo. Em relação ao tempo, entre os dois modelos foi realizado uma análise das restrições de tempo. Para isso foi medido o tempo necessário para prever as classes de todas as imagens entre o conjunto de dados. Como a diferença entre cada modelo é apenas o *backbone*, todas as demais diferenças de tempo estão relacionadas a este aspecto. A Tabela 5.9 exhibe os resultados deste conjunto de testes.

Tabela 5.9: Resultados da análise temporal entre os modelos MobileNetV2 e NasNet-Mobile. Os testes apresentam separação estatística significativa para os resultados de cada par de algoritmos.

	MobileNetV2	NasNetMobile	<i>p-value</i>
times (ms)	30 ± 3	44 ± 2	$p < 2.2 \times 10^{-16}$

Os resultados encontrados indicam que existe uma diferença significativa estatisticamente entre os tempos de cada conjunto amostral. O modelo com melhor

desempenho de temporização foi o MobileNetV2. A diferença relativa entre os modelos foi de cerca de 10%. Considerando as métricas de ML, o resultado indica que o MobileNetV2 deve possuir um desempenho melhor no âmbito desta aplicação. Dessa forma, o MobileNetV2 foi integrado em uma aplicação *mobile* para os testes do algoritmo em um ambiente não controlado, como mostra a Figura 5.2.

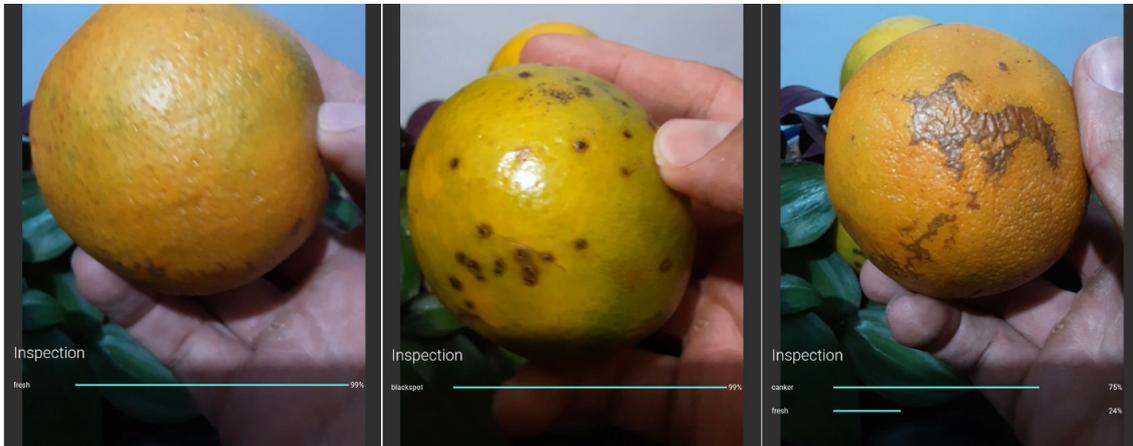


Figura 5.2: Classificação com o modelo MobileNetV2 integrado em uma aplicação mobile.

### 5.1.3 Resultados com os hardwares do dispositivo vestível para o estudo de caso 1

As Tabelas 5.5 e 5.7 anteriormente citadas, mostram resultados satisfatórios com os testes realizados para o modelo de ML e o modelo de DL, respectivamente. Com isso, o MLP e o MobileNetV2 foram escolhidos para os testes no protótipo.

Os testes realizados no monitoramento dos sistemas (CPU, Memória e Energia) também são avaliados. Para cada uma das configurações, ilustradas nas Figuras 5.3 e 5.4, foram realizadas 30 execuções. Após cada execução o equipamento foi desligado e reiniciado, para garantir que o sistema funcionasse sempre da mesma forma durante os testes. Na arquitetura em que a câmera é conectada via USB, o *Wi-Fi* foi desligado para impedir um consumo desnecessário de recurso computacional. Por fim, vale ressaltar que os gráficos referentes ao monitoramento, apresentam a média das 30 execuções.

#### CPU e Frequência de Processamento

Uma parte essencial do sistema a ser monitorado é a CPU, onde ocorre o processo de classificação das imagens das laranjas. Nesta análise, o protótipo melhorou ligeiramente o desempenho da tarefa distribuída, reduzindo o gasto de CPU do dispositivo vestível. O custo de CPU obtido é em média 6,45% comparado ao Pi3 com a câmera

integrada, o qual possui média de 9,57% para utilização de CPU. Para análise estatística, adotamos  $T = 2,06$  e o valor de  $p = 0,04$ , denotando um intervalo de confiança acima de 95%.

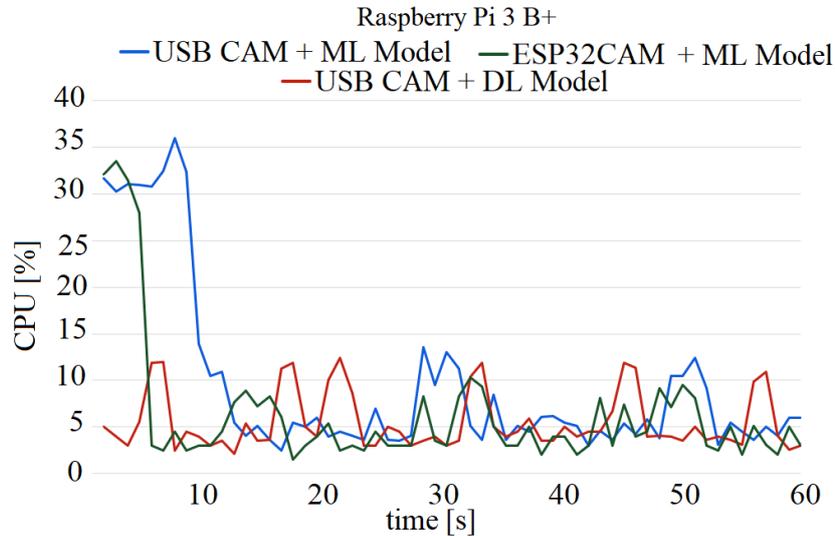


Figura 5.3: Consumo de CPU do sistema.

A Figura 5.3 mostra os valores do consumo. O sistema que contém a câmera USB e o algoritmo de ML tem valor máximo de 36% e mínimo de 2,45%. Com o algoritmo DL, o máximo é de 12,4% e o mínimo de 2,1%. O protótipo apresentou valores máximos e mínimos de 33,5% e 1,5%, respectivamente. Nessa circunstância o *download* das imagens via *wireless* torna-se um fator importante na definição do desempenho, uma vez que a inferência das imagens deve ser realizada instantaneamente. Visando manter boa latência, os tempos de espera dos dispositivos vestíveis devem ser reduzidos para transferência de imagens.

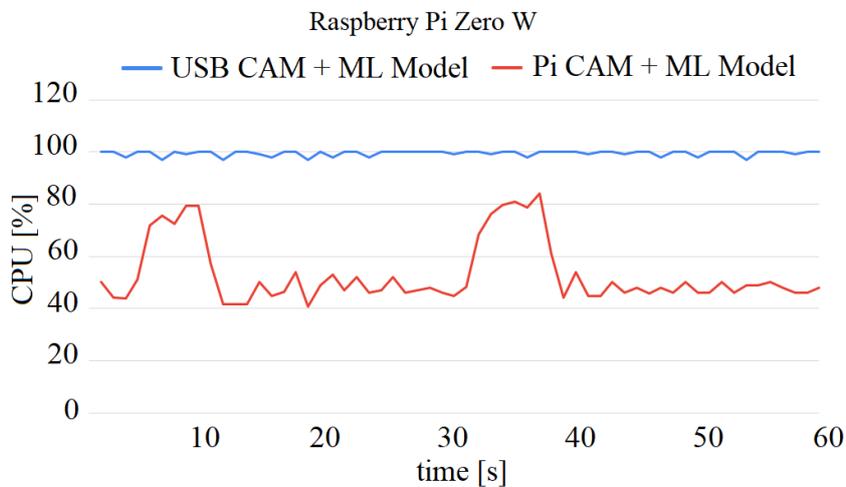


Figura 5.4: Consumo de CPU do Raspberry Pi Zero W.

A Figura 5.4 mostra os valores de consumo de CPU no Raspberry Pi Zero W.

O sistema com a câmera USB tem valor máximo de 100% e mínimo de 97%. O sistema com a câmera Pi tem valor máximo de 84% e mínimo de 40,8%. Assim, a execução fica próxima do limite do sistema, preservando pequenos recursos de CPU com a câmera Pi.

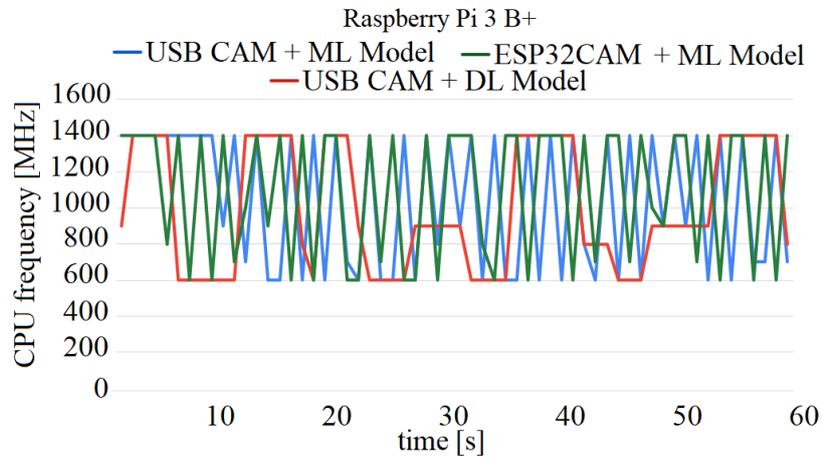


Figura 5.5: Frequência de processamento do sistema.

A Figura 5.5 mostra o comportamento da frequência em três cenários, sendo eles USB CAM + ML model; ESP32CAM + ML model; USB CAM + DL model. Os picos e a frequência mínima de processamento dos sistemas são equivalentes, sendo respectivamente 1400MHz e 600MHz. A tarefa de *stream* é um pouco mais exigente no dispositivo, com média de 1048,33 MHz, e o Edge com a câmera USB embutida tem média de 1091,61 MHz. Para análise estatística temos  $T = -0,64$ , e o valor  $p = 0,52$ , denotando um intervalo de confiança de 48%. O resultado mostra que essas amostras não são separáveis neste teste.

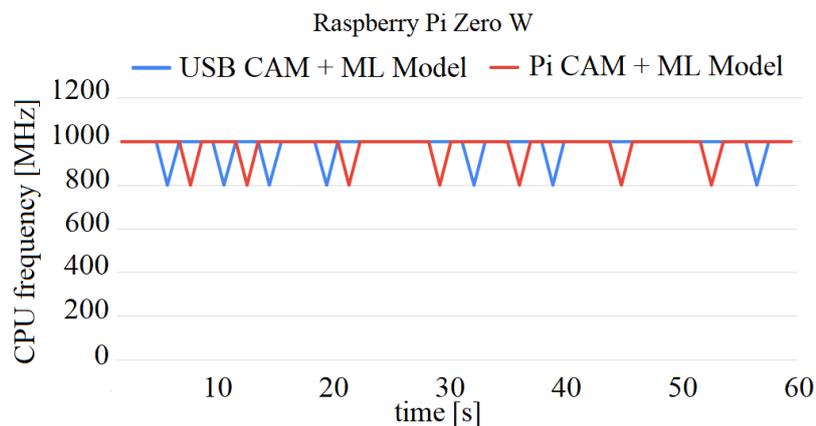


Figura 5.6: Frequência de processamento do sistema para o Raspberry Pi Zero W.

A Figura 5.6 mostra o comportamento da frequência no Raspberry Pi Zero W em dois cenários: USB CAM + ML Model e Pi CAM + ML Model. Os valores máximo e mínimo são 1000 e 800 MHz, respectivamente.

## Consumo de memória

A Figura 5.7 mostra o consumo de memória dos sistemas comparados, os quais são:

- Pi3 (ML Model) com a câmera USB: máximo = 105,26 MB; médio = 104,38 MB; mínimo = 80,57 MB.
- Pi3 (DL Model) com a câmera USB: máximo = 229,88 MB; médio = 229,11 MB; mínimo = 228,13 MB.
- O protótipo (Pi3 com *stream* pelo ESP 32-CAM): máximo = 112,88 MB; médio = 103,91 MB; mínimo = 83,05 MB.

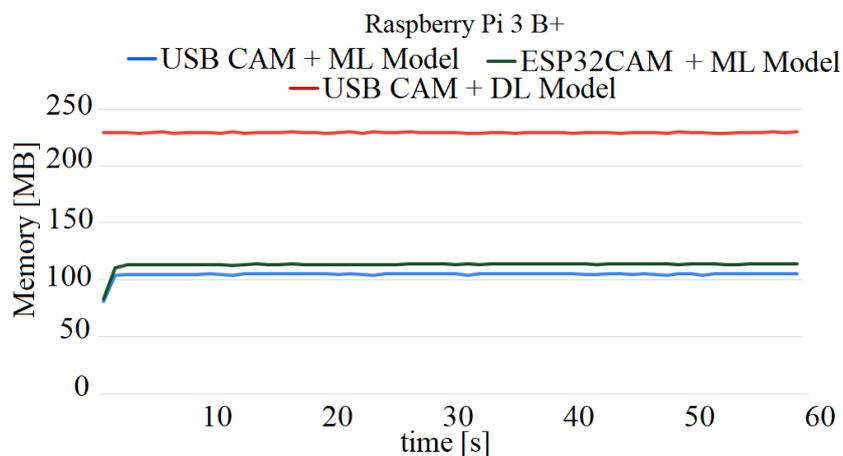


Figura 5.7: Consumo de memória do sistema.

O consumo de RAM tem relação direta com o desempenho do dispositivo. Os dois cenários se comportaram de forma semelhante, utilizando aproximadamente 10% da memória disponível para todo o sistema. Portanto, não há alta demanda para o recurso de memória do sistema. Porém, com o algoritmo DL e a câmera USB, o consumo é aproximadamente duas vezes maior que os outros dois comparados.

A Figura 5.8 mostra o Consumo do Raspberry Pi Zero W. Os valores máximo e mínimo foram 87,5 Mb com a câmera USB e 84,2 Mb para a câmera Pi.

## Consumo de energia

A Figura 5.9 mostra a comparação da quantidade de corrente elétrica requerida pelo sistema no processamento dos dados. Com a câmera USB integrada e o algoritmo ML, o fluxo de corrente tem um pico de 1211,43 mA, uma média de 1059,15 mA e um mínimo de 821,23 mA. Já com o algoritmo DL o consumo tem uma pequena redução, um pico de 980,23 mA, uma média de 805,33 mA e um mínimo de 683,23 mA.

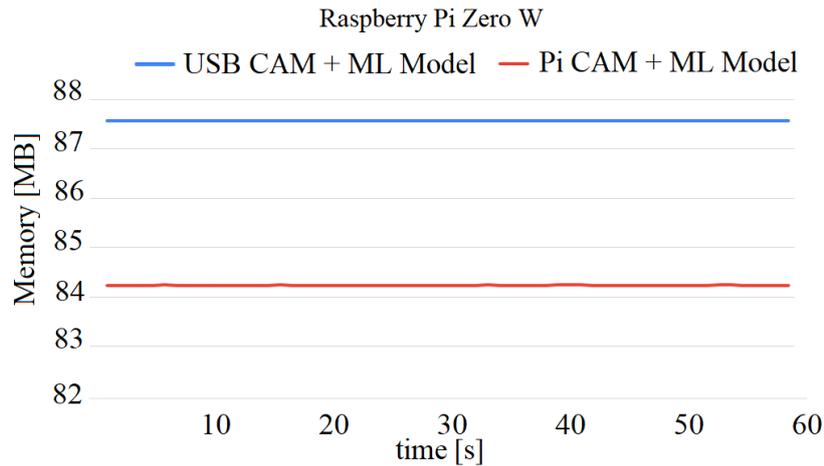


Figura 5.8: Consumo de memória do sistema para o Raspberry Pi Zero W.

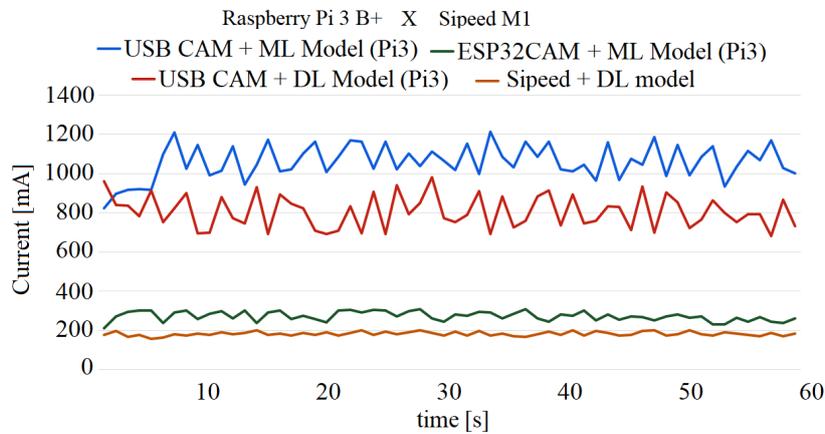


Figura 5.9: Consumo de energia do sistema.

Outra arquitetura que realiza o fluxo de dados do ESP32-CAM reduziu o consumo de energia elétrica. Mesmo com a conexão sem fio, a corrente de pico do dispositivo é de 911,72 mA, com média de 757,15 mA e mínima de 680,02 mA. Este resultado mostra como a distribuição de tarefas impacta positivamente no consumo de energia elétrica, necessária no processamento. Considerando que o ESP32-CAM tem um baixo consumo de energia, chegando a 308,22 mA, com média de 273,06 mA e mínimo de 210,02 mA, o protótipo pode ser utilizado por aproximadamente 11 horas com uma bateria de 3000mAh sem necessidade de recarga. Para análise estatística, temos  $T = 20,25$ , e o valor  $p < 0,0001$ , sendo um intervalo de confiança acima de  $>99,9\%$ .

O Sipeed com o algoritmo DL tem o consumo máximo de corrente elétrica de 201,14 mA e mínimo de 158,22 mA, eficiente para a tarefa de classificação de imagens, conforme mostrado neste trabalho. No entanto, torna-se limitado a realização de tarefas mais complexas. Por fim, a Figura 5.10 ilustra o consumo de energia na Raspberry Pi Zero W, onde é maior com câmera USB em relação a Pi câmera.

Enquanto a Pi câmera é alimentada diretamente pela placa Raspberry Pi Zero W, a câmera USB geralmente requer alimentação própria, isso faz com que demande mais energia do sistema.

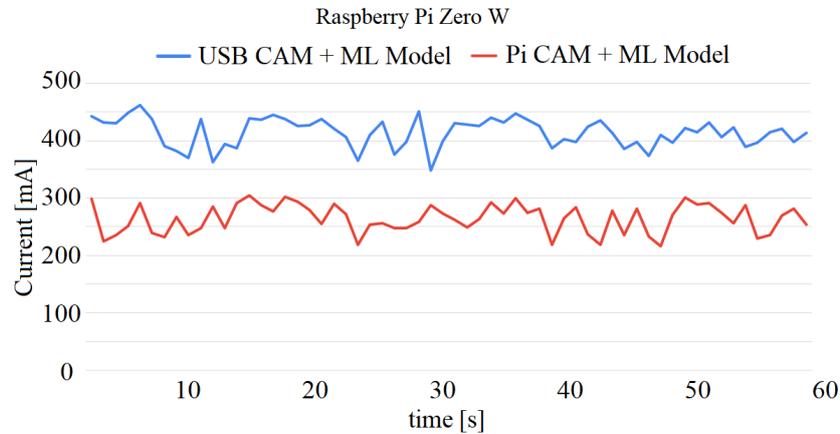


Figura 5.10: Consumo de energia do sistema para o Raspberry Pi Zero W.

O Raspberry Pi Zero W com o algoritmo ML tem o seguinte consumo de corrente nos dois cenários de referência:

- Raspberry Pi Zero W com a câmera USB: máximo = 462 mA; mínimo 348 mA e médio = 414 mA.
- Raspberry Pi Zero W com a Pi câmera: máximo = 304 mA; mínimo 216 mA e médio = 264 mA.

## 5.2 Resultados para o Estudo de Caso 2

Nesta seção são apresentados os resultados obtidos nos testes com os dados coletados para o estudo de caso 2, juntamente com os resultados com três algoritmos de IA aplicados.

A Figura 5.11-A e B mostram os dados entre duas posições usadas para calibração dos sensores em aproximadamente 60 segundos. Pequenas oscilações são perceptíveis na Figura 5.11-A, a classe stand, o que pode ser decorrente de pequenos movimentos das pernas do usuário ou mudanças dos pés durante a coleta dos dados. Na posição da Figura 5.11-B (sentado) não são observadas muitas variações, o que significa que o usuário permaneceu a maior parte do tempo sem se mover. Em ambas as leituras, as informações dos quatérnios K permaneceram praticamente constantes.

A Figura 5.12-C e D mostram os dados coletados pelos sensores para as duas atividades propostas neste estudo de caso, sendo uma a caminhada ao ar livre e outra caminhada em uma esteira com inclinação. Em ambas as atividades nota-

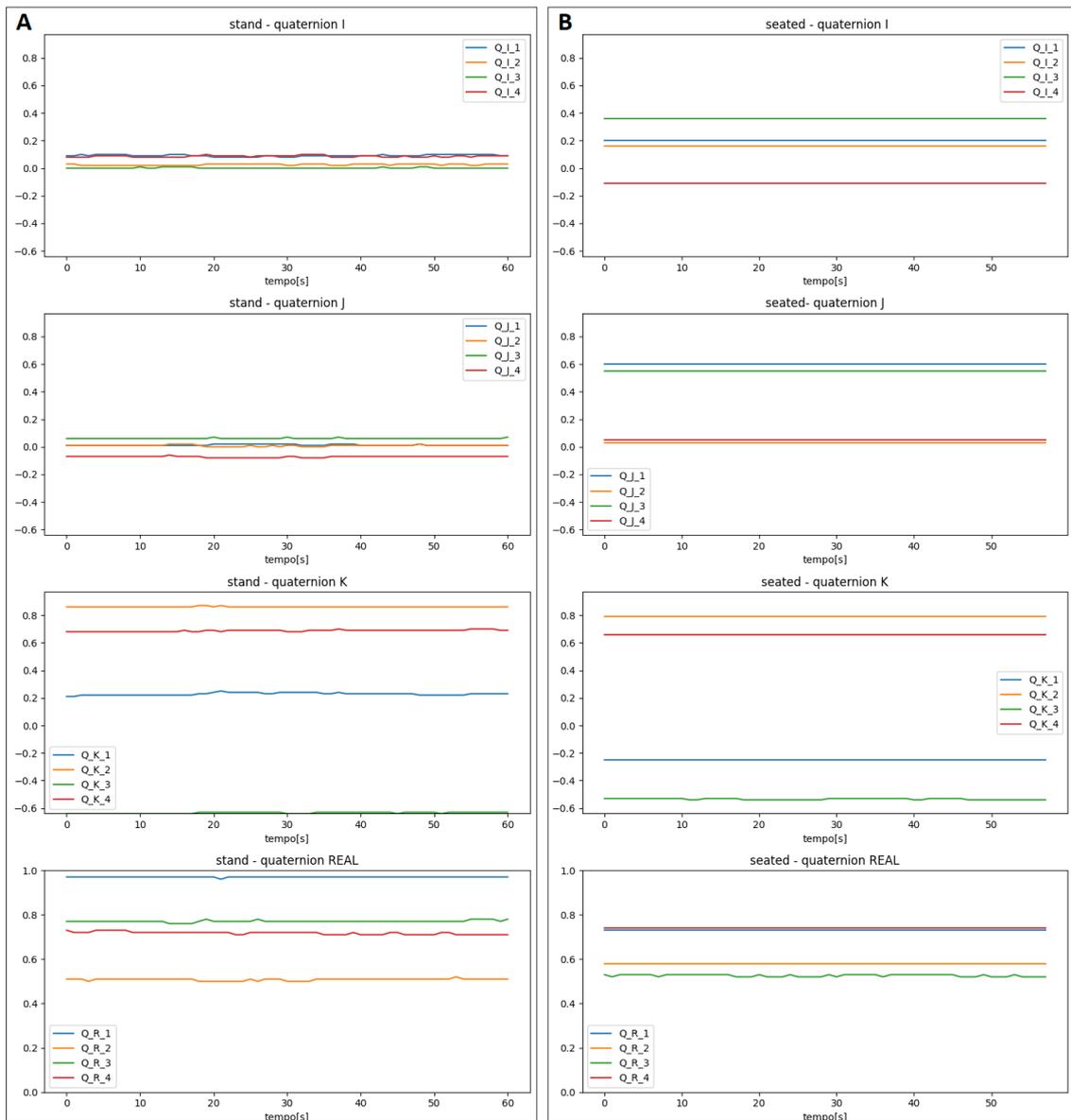


Figura 5.11: A) Stand; B) Seated.

se que há uma variação maior em relação às posições de calibração, o que já era esperado.

Na atividade de caminhada com inclinação, é observado que nenhuma das componentes dos quatérnions permanece constante. As componentes J dos sensores SPU1 e SP3 e as componentes I dos sensores SPU2 e SPU4 sofrem maiores oscilações quando comparadas aos valores relacionados à caminhada ao ar livre. Tais oscilações são devido às maiores amplitudes do movimento para realização da atividade e, por na caminhada ao ar livre não ter as amplitudes, tem-se menores oscilações. Dessa maneira, com futuras análises e modelagens de algoritmos de IA podemos identificar, por exemplo, o esforço físico de um atleta para realizar determinada atividade ou esporte.

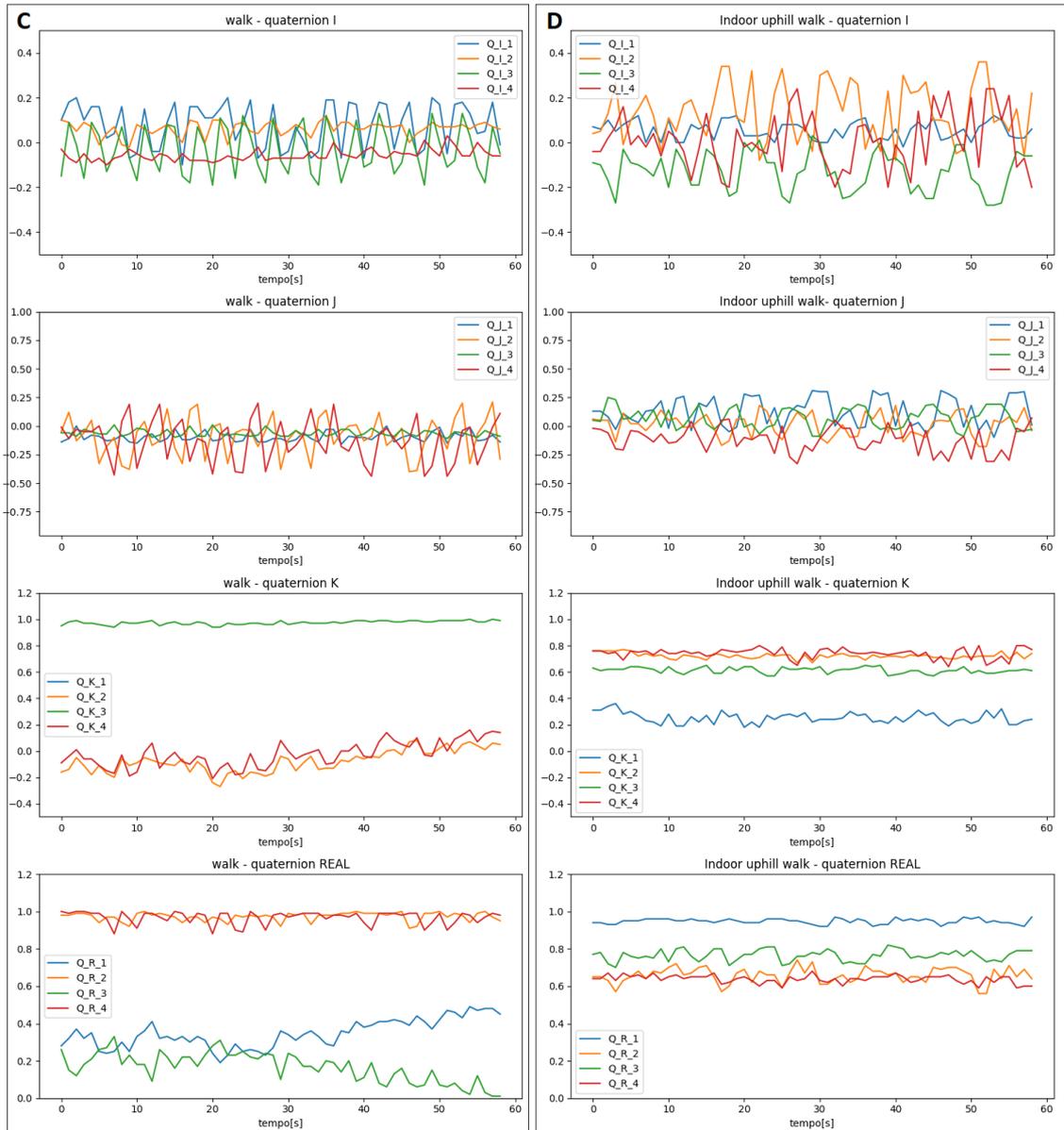


Figura 5.12: C) Walk outdoor; D) Indoor uphill walk.

### 5.2.1 Algoritmos de IA para o HAR

A Figura 5.13 mostra os resultados do treinamento para três modelos de IA. Os gráficos mostram uma tendência zero apesar das oscilações de treinamento em cada época. Além disso, esses resultados não apresentam *overfitting*, mostrando convergência satisfatória para os modelos. Após 15 épocas, o modelo GRU obteve uma precisão de validação de 98%. A Figura 5.14 exibe os resultados do teste para os três modelos AI. Neste teste, vemos que o modelo classificou os dados com precisão entre as quatro classes adotadas.

Os resultados dos testes de treinamento dos três algoritmos de IA usados para este estudo de caso podem ser observados na Tabela 5.10. A precisão, o *recall* e o *F1-score* foram avaliados para cada uma das classes. Os resultados indicam que

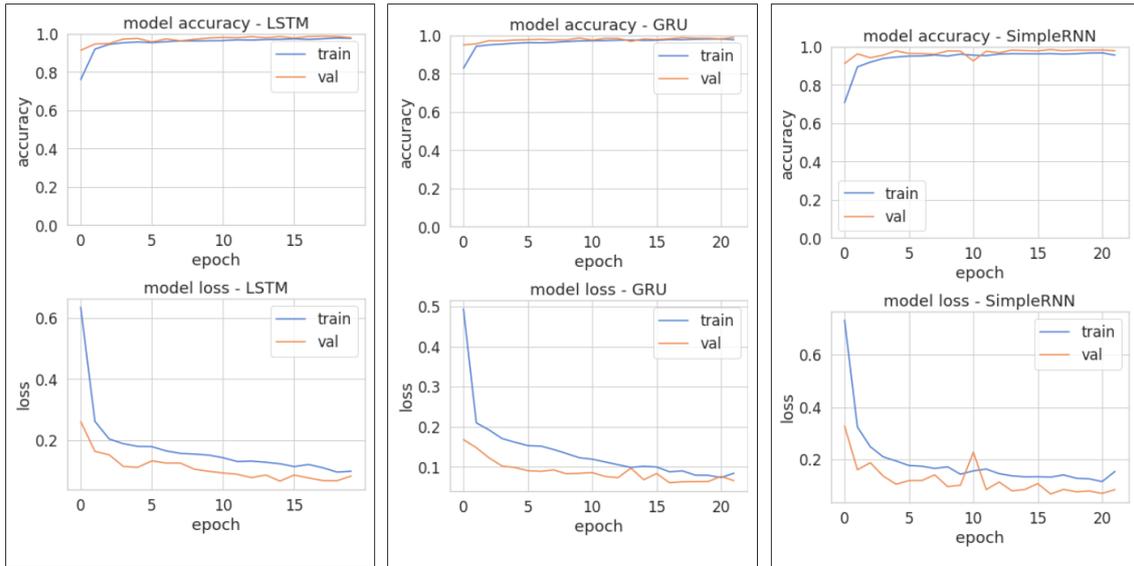


Figura 5.13: Resultado do treinamento dos algoritmos.

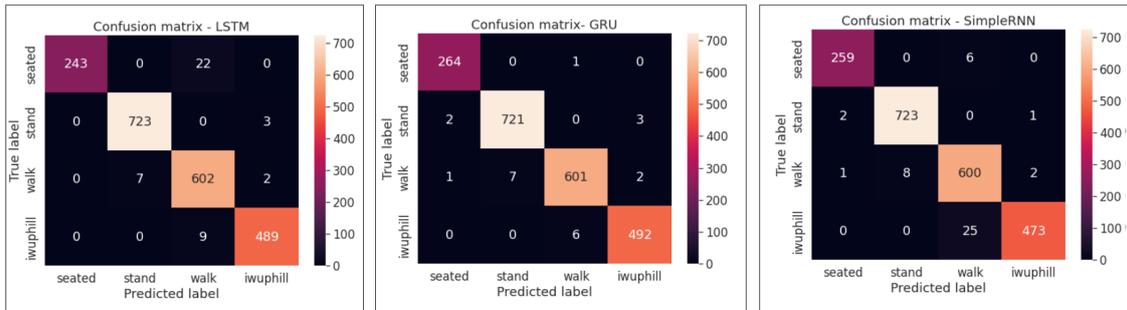


Figura 5.14: *Confusion Matrix*.

o GRU teve um desempenho um pouco melhor quando comparado com as duas outras opções. As métricas de LSTM e *Simple RNN* indicam que tem o mesmo desempenho, no entanto, o LSTM tem um *Recall* melhor na classificação da classe iwuphill. Para uma visão geral, os algoritmos estudados atingiram uma precisão global superior a 98%.

Semwall et al.[164] alcançaram 92% de precisão usando um algoritmo CNN para classificar atividade de caminhada com os dados coletados com sensores IMU, no entanto, os resultados dos sensores com 9DoF e a fusão dos dados mostraram a

Tabela 5.10: Métricas para os modelos LSTM, GRU e Simple RNN

	Precisão			Recall			F1-Score			Support
	LSTM	GRU	Simple RNN	LSTM	GRU	Simple RNN	LSTM	GRU	Simple RNN	
Seated	1.00	0.99	0.99	0.92	1.00	0.98	0.96	0.99	0.98	265
Stand	0.99	0.99	0.99	1.00	0.99	1.00	0.99	0.99	0.99	726
Walk	0.95	0.99	0.95	0.99	0.98	0.98	0.97	0.99	0.97	611
Iwuphill	0.99	0.99	0.99	0.97	0.99	0.93	0.98	0.99	0.98	498
<b>Macro average</b>	0.98	0.99	0.98	0.99	0.99	0.98	0.98	0.99	0.98	2100
<b>Weighted average</b>	0.98	0.99	0.98	0.97	0.99	0.98	0.98	0.99	0.98	2100
<b>Global Accuracy</b>	<b>98 %</b>	<b>99 %</b>	<b>98 %</b>							

eficiência dessa metodologia para o treinamento e teste dos algoritmos atingindo uma precisão de 98%. Bijalwan et al. [165] alcançaram a precisão de 98% usando CNN-LSTM para o HAR baseado em visão. Com HAR baseado em sensor, alcançamos 99% de precisão com o GRU. Além disso, alcançamos uma precisão superior a 96,37%, obtida por Sravan et al. [166].

O dispositivo vestível utilizado neste estudo de caso foi capaz de fornecer informações sobre a atividade de caminhada realizada pelo usuário devido a fusão dos dados dos sensores vestíveis. Como as atividades exploradas são semelhantes, foi essencial o cruzamento de dados entre os quatro sensores, na perna do usuário. O uso de quatro sensores foi um grande diferencial dessa abordagem, comparado aos dispositivos que utilizam sensores únicos destinados ao HAR. Os sensores únicos provavelmente teriam dificuldades em classificar tarefas com movimentos semelhantes. Assim, o método abordado nesse estudo de caso mostra uma boa eficiência para os algoritmos de IA alcançarem uma alta acurácia global após o treinamento.

# Capítulo 6

## Discussões

Este capítulo apresenta as discussões dos resultados apresentados no Capítulo 5. Primeiramente, serão abordadas as discussões relativas ao estudo de caso 1, com as considerações sobre os algoritmos utilizados. Em seguida, as perspectivas do dispositivo vestível utilizado para a inspeção de laranjas. Por fim, as considerações do estudo de caso 2, com as análises dos algoritmos e do dispositivo desenvolvido para o reconhecimento da atividade de caminhada.

### 6.1 Estudo de caso 1

Para o desenvolvimento do dispositivo vestível responsável pela inspeção em laranjas, previamente foi apresentado o algoritmo de ML para a classificação de imagens utilizando a técnica de aprendizado supervisionado. Com intuito de realizar uma comparação entre os modelos de IA, que melhor enquadram-se no dispositivo, foram estudados também as técnicas envolvendo modelos de DL. Desse modo, foi possível integrar os métodos mais viáveis, em um novo dispositivo vestível, com finalidade de realizar a inspeção de laranjas.

#### 6.1.1 Os algoritmos de machine learning

Para a classificação de imagens, foram utilizados quatro abordagens tradicionais de ML. Os modelos escolhidos foram o SVM, KNN, *Decision Tree* e o *Random Forest*. Antes de treinar o modelo com a base de dados contendo as quatro classes, as três doenças de laranjas e as laranjas frescas, foi realizado a extração das características de cada imagem para um vetor. O método adotado extrai as informações do canal H de uma imagem HSV, processo denominado de análise “pseudoespectral”.

Nestes tipos de análises em específico, os métodos de ML alcançaram mais de 92% de precisão na classificação dos dados, sendo, os algoritmos *Random Forest* e MLP apresentaram uma acurácia maior entre os modelos treinados. Dentre os dois,

o modelo MLP obteve uma acurácia de 98%, apresentou um alto desempenho na execução no dispositivo, tanto no *hardware* com maior processamento (Pi3), quando no de menor (Raspberry pi zero W), além da excelente eficácia para o reconhecimento das doenças nas laranjas. Portanto, foi o escolhido para integração no dispositivo vestível.

### 6.1.2 Os algoritmos de Deep learning

Na revisão de literatura foi possível compreender melhor os algoritmos de DL, os quais possuem milhares de parâmetros. Estes algoritmos estudados permitem a integração no dispositivo vestível, de forma que realize todo o processamento na borda, sem precisar de um alto poder computacional do sistema para detectar as doenças em laranjas. Dentro deste contexto, os resultados apresentados destacam os benefícios do uso do modelo MobileNetV2 nos dispositivos vestíveis.

Altos valores nas métricas de avaliação dos algoritmos de DL nem sempre são determinantes para a escolha do melhor modelo e implementação no dispositivo vestível de interesse. Nas informações obtidas pelo levantamento bibliográfico destaca-se que os modelos tradicionais alcançam grandes resultados. No entanto, no contexto dos vestíveis, algoritmos com grande número de parâmetros, como várias camadas na arquitetura do modelo, podem exigir maior poder de processamento do *hardware*, sendo inviável para aplicação de modelos muito profundos nos dispositivos vestíveis. De posse dessa informação, a melhor opção foi integrar um modelo que, apesar de não ter a melhor precisão que modelos mais profundos, alcança uma resposta em um curto intervalo de tempo para a inferência das imagens, além de não demandar *hardware* com alto poder de processamento.

Na avaliação dos modelos de DL para classificação de imagens de laranja, pode-se concluir que o MobileNetV2 apresentou um bom equilíbrio entre precisão e tempo para realizar a tarefa em questão um dispositivo vestível.

Quando comparado com o NasNetMobile, apresentou um desempenho de temporização menor e maior acurácia global. A acurácia global de 95% do MobileNetV2 mostrou-se efetiva nos testes com o Sipeed e o Pi3, além disso, uma integração em uma aplicação *mobile*, como mostra a Figura 5.2. Contudo, o modelo de DL treinado consegue fazer a classificação de doenças em laranjas de forma eficiente. Assim, ser integrado no dispositivo vestível para o processamento na borda.

Um dos principais motivos da escolha do MobileNetV2 para integrar o dispositivo vestível, foi pela menor quantidade de parâmetros exigidos comparando-se com os outros modelos de DL treinados, acarretando em menor poder de processamento do *hardware*, podendo funcionar sem problemas em dispositivos restritos de recursos. Apesar dessa vantagem, há de se destacar que os modelos mais profundos podem

ou não funcionar nestes dispositivos, como o Raspberry pi zero W, pois exigem do dispositivo uma quantidade expressiva de memória do sistema, sobretudo se tratando da computação de borda.

### 6.1.3 O dispositivo vestível

Os resultados apresentados mostram as vantagens e desvantagens em aplicar algoritmos de ML ou DL nos dispositivos vestíveis. O algoritmo mais efetivo para esse estudo de caso dependerá do *hardware* utilizado. Para o Raspberry Pi Zero W os modelos de DL não são viáveis, uma vez que é o dispositivo com menor poder computacional entre os utilizados. Apesar do modelo DL obter ótimos resultados, para o Pi3 o modelo ML consome menor quantidade de memória do sistema, condição que pode ser determinante na escolha do modelo, haja vista que simultaneamente outras tarefas podem ser adicionadas. Por exemplo, o armazenamento de dados, para a geolocalização das doenças nas plantações ou o armazenamento das imagens das laranjas doentes. Contudo, o modelo de ML foi a melhor escolha para integrar o dispositivo, pois ajuda na preservação dos recursos de memória e energia, nesse contexto.

A proposta de distribuir as tarefas do sistema foi de suma importância para preservação dos recursos do *hardware*. Diferentemente do Sipeed e do Raspberry Pi Zero W, que funcionam no limite de seus sistemas, a abordagem usada no desenvolvimento do capacete inteligente diminui o uso de memória e energia na realização da tarefa avaliada. Dessa forma, o dispositivo vestível com um algoritmo de IA processado na borda, com uma arquitetura distribuída, tem seus recursos preservados, mantendo a autonomia energética e a comunicação em baixa latência. Por fim, este dispositivo, ao ser aplicado em campo, pode auxiliar citricultores na tarefa de inspeção das laranjas, realizando inspeções eficientes, o que, por consequência reduziria perdas de frutos por infestações de doenças nos pomares e, logo, os danos econômicos para os produtores.

## 6.2 Estudo de caso 2 - Reconhecimento de atividades Humanas

O desenvolvimento do dispositivo vestível para o HAR foi direcionado para atividades de caminhada. Primeiramente, para criação do dispositivo, foi estudado os principais aspectos relacionados ao campo, como o HAR baseado em visão, sensores, IA e esportes, possibilitando o entendimento mais apurado das necessidades de construir uma possível nova solução, caso seja viável em diferentes aspectos. Não

obstante, a ideia é também propor três soluções de algoritmos de IA para classificar os dados coletados pelo dispositivo.

A revisão de literatura enfatizou que o HAR tem aplicações em diferentes áreas, como, por exemplo, nos esportes. A tarefa para o HAR é descrever o conjunto de técnicas usadas para perceber o comportamento humano de forma computacional. Até então, já existem várias abordagens para esta tarefa, com sensores de modalidade única ou multimodal. O HAR pode ser separado, a depender dos autores do trabalhos, em visual ou sensor. Em ambos os casos, é trivial o papel que a IA exerce estando presente na maioria das propostas de novas soluções.

### 6.2.1 Os algoritmos de IA para o HAR

A revisão de literatura possibilitou o entendimento dos algoritmos de IA aplicados para o HAR. Em várias abordagens esportivas, os autores propõem o uso de HAR baseado em sensores usando unidades vestíveis, com a utilização, pela maioria dos autores, de abordagens que empregam o uso das CNNs ou RNNs para executar tarefas HAR. Desse modo, para este estudo de caso foi apresentado três modelos de RNN usualmente utilizados para análise de dados por séries temporais, sendo as redes LSTM, GRU E *Simple* RNN, que se mostraram eficientes na classificação dos dados.

A integração de quatro sensores denominados SPUs foi essencial para a aplicação deste estudo de caso. Ao mesclar os dados, os três algoritmos de IA apresentaram uma precisão geral de mais de 98%. A precisão observada mostra que o dispositivo pode ser aplicado a outros estudos de caso, por exemplo, para ajudar atletas (ex: corredores), fornecendo informações preciosas para melhorar o desempenho físico.

Assim, algoritmos de IA para o HAR possuem uma abordagem interessante para aplicações com sensores vestíveis, conforme apresentado anteriormente para a classificação dos dados coletados. Tal feito se deve aos avanços dos dispositivos mais compactos com a possibilidade de implementar algoritmos de IA em suas estruturas, assim como devido ao desenvolvimento desses algoritmos que não requerem alto poder computacional para o processamento dos dados.

### 6.2.2 O dispositivo vestível

A revisão bibliográfica mostrou que os esportes de caminhada estão entre as atividades físicas mais populares realizadas em todo o mundo. Esta atividade tem um amplo mercado, fomentando o interesse em diversas áreas, como fisiologia e a criação de novas soluções para autoavaliação, sendo, portanto, com alvos de várias pesquisas. O dispositivo desenvolvido neste estudo vai de encontro aos interesses de muitas pesquisas, sobretudo por ter apresentado boas perspectivas para aplicação. O *hard-*

*ware*, juntamente com os algoritmos de IA, conseguiram realizar uma classificação de forma eficiente para a atividades de caminhada.

A evolução do dispositivo vestível desenvolvido, uma calça inteligente, pode ser usada futuramente para o reconhecimento de outras atividades, como corrida. Devido à fusão dos dados, os algoritmos podem ser mais eficientes neste contexto, ampliando as possibilidades de esportes aplicados para a construção de novas bases de dados, como também a classificação de outros esportes com algoritmos de IA.

# Capítulo 7

## Conclusões e Trabalhos Futuros

O levantamento bibliográfico mostrou que os algoritmos de IA estão se tornando uma tecnologia imprescindível em nossas vidas, estando presentes em diversos ramos e ferramentas do nosso cotidiano. Além disso, outra tecnologia com grande potencial de inovação e desenvolvimento são os dispositivos vestíveis, os quais estão ganhando cada vez mais visibilidade no mercado e no meio acadêmico. Esses dispositivos, em conjunto com os algoritmos de IA, entregam informações cada vez mais eficientes para o usuário, principalmente quando o processamento dos dados é realizado na borda do dispositivo.

Entretanto, esses dispositivos são limitados de recursos computacionais devido às características físicas, como dimensões reduzidas, o que até o momento impede a integração de melhores componentes para aumentar seu poder computacional. Diante disso, nesse estudo foram apresentadas possíveis soluções para implementação dos algoritmos de IA nos dispositivos vestíveis para dois estudos de caso: A inspeção de laranjas e o reconhecimento de atividade de caminhada. Desse modo, o trabalho contempla os seguintes aspectos e lições aprendidas:

- A revisão da literatura apresentou os principais requisitos para o projeto e prototipagem dos dispositivos vestíveis, bem como os aspectos dos algoritmos de IA para a integração nestes sistemas, além de mostrar as vantagens de realizar o processamento na borda, com *hardwares* de baixo poder computacional;
- Foi possível integrar os algoritmos de IA nos dispositivos vestíveis com baixa capacidade computacional, usando o processamento dos dados totalmente na borda. Além disso, em ambientes externos, esses dispositivos apresentam uma solução de baixa latência e com alta eficiência energética;
- O estudo de caso 1, com a utilização de técnicas de visão computacional, bem como a compreensão dos aspectos construtivos dos algoritmos e do dispositivo, foram essenciais para a validação dos algoritmos de IA. Nos modelos de ML,

a análise pseudoespectral, estratégia utilizada para o pré-processamento das imagens, foi importante para a construção dos algoritmos, para processar os dados na borda, sem limitar os recursos computacionais do dispositivo;

- Ainda sobre o estudo de caso 1, para os algoritmos de DL, os modelos estudados com menor quantidade de parâmetros podem ser executados no dispositivo, de forma eficiente, sem limitar os recursos do mesmo. Por fim o capacete inteligente criado nesse estudo de caso permite a conservação dos recursos do dispositivo, com a proposta de uma arquitetura distribuída, possibilitando aplicações como modelos de IA mais complexos. Assim, a solução vestível pode ser empregada na inspeção de laranjas, para auxiliar citricultores em atividades que são atualmente realizadas manualmente e que são extremamente laboriosas e passíveis de erros;
- O estudo de caso 2, envolvendo o reconhecimento de atividade humana, foi possível validar os algoritmos de IA, isso com a construção de uma base sólida para o treinamento dos modelos. O *data augmentation* usado para a ampliação da base de dados mostrou eficiente para os algoritmos, especialmente com ajuda das RNNs para classificação das atividades. Assim, a solução construída para este estudo de caso, futuramente pode auxiliar usuários na realização dessas atividades.

Nos trabalhos futuros há pretensão de continuar o desenvolvimento dos algoritmos de IA aplicados para os dispositivos vestíveis, especificamente para o dispositivo destinado ao HAR. Neste estudo de caso existe uma ampla aplicabilidade para a continuidade da pesquisa, por exemplo, a construção de aplicações *mobile* a partir de dados gerados em tempo real para o usuário. Também há possibilidade de construção de novas bases de dados, que incorporam outras atividades, como corrida, *trekking*, ciclismo, entre outros, o que mostra a quantidade significativa de possibilidades para novas pesquisas e projetos.

# Referências Bibliográficas

- [1] DE DEFESA DA CITRICULTURA, F. “Fundecitrus: Ciência e Sustentabilidade para a Citricultura”. 2021. Disponível em: <<https://www.fundecitrus.com.br>>.
- [2] PEAKE, J. M., KERR, G., SULLIVAN, J. P. “A critical review of consumer wearables, mobile applications, and equipment for providing biofeedback, monitoring stress, and sleep in physically active populations”, *Frontiers in physiology*, v. 9, pp. 743, 2018.
- [3] MCCARTHY, J. “What is artificial intelligence”. <http://jmc.stanford.edu/articles/whatisai.html>. Accessed: 2022-06-15.
- [4] AMORIM, V. J. P., OLIVEIRA, R. A. R., DA SILVA, M. J. “Recent Trends in Wearable Computing Research: A Systematic Review”, 2020.
- [5] SILVA, M. C., AMORIM, V. J. P., RIBEIRO, S. P., et al. “Field Research Cooperative Wearable Systems: Challenges in Requirements, Design and Validation”, October 2019.
- [6] HOESER, T., KUENZER, C. “Object Detection and Image Segmentation with Deep Learning on Earth Observation Data: A Review-Part I: Evolution and Recent Trends”, *Remote Sensing*, v. 12, n. 10, 2020. ISSN: 2072-4292. doi: 10.3390/rs12101667. Disponível em: <<https://www.mdpi.com/2072-4292/12/10/1667>>.
- [7] SANTOS, G., TAVARES, T., ROCHA, A. “Reliability and generalization of gait biometrics using 3D inertial sensor data and 3D optical system trajectories”, *Scientific Reports*, v. 12, n. 1, pp. 1–15, 2022.
- [8] SHAFIQUE, K., KHAWAJA, B. A., SABIR, F., et al. “Internet of things (IoT) for next-generation smart systems: A review of current challenges, future trends and prospects for emerging 5G-IoT scenarios”, *Ieee Access*, v. 8, pp. 23022–23040, 2020.

- [9] HEIKENFELD, J., JAJACK, A., ROGERS, J., et al. “Wearable sensors: modalities, challenges, and prospects”, *Lab on a Chip*, v. 18, n. 2, pp. 217–248, 2018.
- [10] YURUR, O., LIU, C. H. *Generic and energy-efficient context-aware mobile sensing*. CRC Press, 2015.
- [11] PATEL, M. S., ASCH, D. A., VOLPP, K. G. “Wearable devices as facilitators, not drivers, of health behavior change”, *Jama*, v. 313, n. 5, pp. 459–460, 2015.
- [12] KONG, X. T., LUO, H., HUANG, G. Q., et al. “Industrial wearable system: the human-centric empowering technology in Industry 4.0”, *Journal of Intelligent Manufacturing*, v. 30, n. 8, pp. 2853–2869, 2019.
- [13] CHEN, Z., HU, W., WANG, J., et al. “An Empirical Study of Latency in an Emerging Class of Edge Computing Applications for Wearable Cognitive Assistance”. In: *Proceedings of the Second ACM/IEEE Symposium on Edge Computing*, New York, NY, USA, 2017. Association for Computing Machinery. ISBN: 9781450350877. doi: 10.1145/3132211.3134458. Disponível em: <<https://doi.org/10.1145/3132211.3134458>>.
- [14] CHAWLA, N., DALAL, S. “Edge AI with Wearable IoT: A Review on Leveraging Edge Intelligence in Wearables for Smart Healthcare”, *Green Internet of Things for Smart Cities*, pp. 205–231, 2021.
- [15] “Keras Applications”. <https://keras.io/api/applications/>. Accessed: 2022-05-23.
- [16] KLERKX, L., JAKKU, E., LABARTHE, P. “A review of social science on digital agriculture, smart farming and agriculture 4.0: New contributions and a future research agenda”, *NJAS - Wageningen Journal of Life Sciences*, v. 90-91, pp. 100315, 2019. ISSN: 1573-5214. doi: <https://doi.org/10.1016/j.njas.2019.100315>. Disponível em: <<https://www.sciencedirect.com/science/article/pii/S1573521419301769>>.
- [17] SISHODIA, R. P., RAY, R. L., SINGH, S. K. “Applications of Remote Sensing in Precision Agriculture: A Review”, *Remote Sensing*, v. 12, n. 19, 2020. ISSN: 2072-4292. doi: 10.3390/rs12193136. Disponível em: <<https://www.mdpi.com/2072-4292/12/19/3136>>.
- [18] DE OLIVEIRA, M. E., CORRÊA, C. G. “Virtual Reality and Augmented reality applications in agriculture: a literature review”. In: *2020 22nd*

*Symposium on Virtual and Augmented Reality (SVR)*, pp. 1–9, 2020. doi: 10.1109/SVR51698.2020.00017.

- [19] 4VISION. “Como a visão computacional está revolucionando a gestão de qualidade na indústria”. 2019. Disponível em: <<https://www.4vision.com.br/blog>>.
- [20] ROSA, A. L. “Classificação de Imagens de Frutas utilizando Aprendizado de Máquina”, 2019. Disponível em: <<https://repositorio.ufsc.br/handle/123456789/197598>>.
- [21] SOINI, C. T., FELLAH, S., ABID, M. R. “Greening Infection Detection (Ci-GID) by Computer Vision and Deep Learning”, 2019. Disponível em: <<https://dl.acm.org/doi/10.1145/3325917.3325936>>.
- [22] FILHO, W. S. S. “Plano estratégico da equipe técnica de citros da Embrapa Mandioca e Fruticultura 2017-2021”, 2019. Disponível em: <<http://www.infoteca.cnptia.embrapa.br/infoteca/handle/doc/1108838>>.
- [23] SALDAÑA, E., SICHE, R., LUJÁN, M., et al. “visão computacional aplicada à inspeção e ao controle da qualidade de frutas e verduras”, 2013. Disponível em: <[https://www.scielo.br/scielo.php?pid=S1981-67232013000400002&script=sci\\_abstract&tlng=pt](https://www.scielo.br/scielo.php?pid=S1981-67232013000400002&script=sci_abstract&tlng=pt)>.
- [24] BOTTANI, E., VIGNALI, G. “Augmented reality technology in the manufacturing industry: A review of the last decade”, *IISE Transactions*, v. 51, n. 3, pp. 284–310, 2019. doi: 10.1080/24725854.2018.1493244. Disponível em: <<https://doi.org/10.1080/24725854.2018.1493244>>.
- [25] CARIA, M., SARA, G., TODDE, G., et al. “Exploring Smart Glasses for Augmented Reality: A Valuable and Integrative Tool in Precision Livestock Farming”, *Animals*, v. 9, n. 11, 2019. ISSN: 2076-2615. doi: 10.3390/ani9110903. Disponível em: <<https://www.mdpi.com/2076-2615/9/11/903>>.
- [26] XI, M., ADCOCK, M., MCCULLOCH, J. “Future Agriculture Farm Management using Augmented Reality”. In: *2018 IEEE Workshop on Augmented and Virtual Realities for Good (VAR4Good)*, pp. 1–3, 2018. doi: 10.1109/VAR4GOOD.2018.8576887.
- [27] CITRUSBR. “Laranja e suco Produtos e Subprodutos”. 2020. Disponível em: <<http://www.citrusbr.com>>.

- [28] EMBRAPA. “Laranja no Mundo, Produção Mundial”, 2018. Disponível em: <[http://www.cnpmf.embrapa.br/Base\\_de\\_Dados/index\\_pdf/mundo/laranja/laranja\\_mundo\\_desempenho.htm](http://www.cnpmf.embrapa.br/Base_de_Dados/index_pdf/mundo/laranja/laranja_mundo_desempenho.htm)>.
- [29] FUNDECITRUS. “Cancro Cítrico”. 2016. Disponível em: <<https://www.fundecitrus.com.br/doencas/cancro>>.
- [30] FUNDECITRUS. “Pinta Preta”. 2016. Disponível em: <<https://www.fundecitrus.com.br/doencas/pinta-preta>>.
- [31] OF AGRICULTURE ANIMAL, U. S. D., SERVICE, P. H. I. “Citrus Greening”. Outubro 2018. Disponível em: <<https://www.aphis.usda.gov/aphis/resources/pests-diseases/hungry-pests/the-threat/citrus-greening/citrus-greening-hp>>.
- [32] “Running Jogging - Statistics Facts”. <https://www.statista.com/topics/1743/running-and-jogging/#dossierKeyfigures>.
- [33] FOLLAND, J. P., ALLEN, S. J., BLACK, M. I., et al. “Running Technique is an Important Component of Running Economy and Performance”, *Med Sci Sports Exerc*, v. 49, n. 7, pp. 1412–1423, 07 2017.
- [34] FRANCIS, P., WHATMAN, C., SHEERIN, K., et al. “The Proportion of Lower Limb Running Injuries by Gender, Anatomical Location and Specific Pathology: A Systematic Review”, *J Sports Sci Med*, v. 18, n. 1, pp. 21–31, 03 2019.
- [35] SOUZA, R. B. “An Evidence-Based Videotaped Running Biomechanics Analysis”, *Phys Med Rehabil Clin N Am*, v. 27, n. 1, pp. 217–236, Feb 2016.
- [36] NWEKE, H. F., TEH, Y. W., AL-GARADI, M. A., et al. “Deep learning algorithms for human activity recognition using mobile and wearable sensor networks: State of the art and research challenges”, *Expert Systems with Applications*, v. 105, pp. 233–261, 2018. ISSN: 0957-4174. doi: <https://doi.org/10.1016/j.eswa.2018.03.056>.
- [37] MASOUD, M., JARADAT, Y., MANASRAH, A., et al. “Sensors of smart devices in the internet of everything (IoE) era: big opportunities and massive doubts”, *Journal of Sensors*, v. 2019, 2019.
- [38] CORNACCHIA, M., OZCAN, K., ZHENG, Y., et al. “A Survey on Activity Detection and Classification Using Wearable Sensors”, *IEEE Sensors Journal*, v. 17, n. 2, pp. 386–403, 2017. doi: 10.1109/JSEN.2016.2628346.

- [39] PIRES, I. M., GARCIA, N. M., POMBO, N., et al. “From data acquisition to data fusion: a comprehensive review and a roadmap for the identification of activities of daily living using mobile devices”, *Sensors*, v. 16, n. 2, pp. 184, 2016.
- [40] GRAVINA, R., ALINIA, P., GHASEMZADEH, H., et al. “Multi-sensor fusion in body sensor networks: State-of-the-art and research challenges”, *Information Fusion*, v. 35, pp. 68–80, 2017.
- [41] SHOAI, M., BOSCH, S., INCEL, O. D., et al. “Fusion of smartphone motion sensors for physical activity recognition”, *Sensors*, v. 14, n. 6, pp. 10146–10176, 2014.
- [42] MULLER, A. C., GUIDO, S. *Introduction to Machine Learning with Python*, 2016.
- [43] HASTIE, T., TIBSHIRANI, R., FRIEDMAN, J. *The elements of statistical learning: data mining, inference and prediction*. Springer, 2009. Disponível em: <<http://www-stat.stanford.edu/~tibs/ElemStatLearn/>>.
- [44] FACURE, M. “Aprendizado Semi-Supervisionado para Detecção de Fraudes Parte 1”, 2017. Disponível em: <<https://lamfo-unb.github.io/>>.
- [45] NETO, A. R. P., MURTA, A. H. F., MENDONÇA, V. I., et al. “Programação dinâmica e aprendizado por reforço”, 2020. doi: [1http://www.opl.ufc.br/pt/project/dp-rl/](http://www.opl.ufc.br/pt/project/dp-rl/).
- [46] MNIH, V., KAVUKCUOGLU, K., SILVER, D., et al. “Playing atari with deep reinforcement learning”, *arXiv preprint arXiv:1312.5602*, 2013.
- [47] FRANÇOIS-LAVET, V., HENDERSON, P., ISLAM, R., et al. “An introduction to deep reinforcement learning”, *Foundations and Trends® in Machine Learning*, v. 11, n. 3-4, pp. 219–354, 2018.
- [48] WANG, P., FAN, E., WANG, P. “Comparative analysis of image classification algorithms based on traditional machine learning and deep learning”, *Pattern Recognition Letters*, v. 141, pp. 61–67, 2021. ISSN: 0167-8655. doi: <https://doi.org/10.1016/j.patrec.2020.07.042>. Disponível em: <<https://www.sciencedirect.com/science/article/pii/S0167865520302981>>.
- [49] HUANG, S., CAI, N., PACHECO, P. P., et al. “Applications of support vector machine (SVM) learning in cancer genomics”, *Cancer genomics & proteomics*, v. 15, n. 1, pp. 41–51, 2018.

- [50] WASULE, V., SONAR, P. “Classification of brain MRI using SVM and KNN classifier”. In: *2017 Third International Conference on Sensing, Signal Processing and Security (ICSSS)*, pp. 218–223. IEEE, 2017.
- [51] BHAVANI, R., WISELIN JIJI, G. “Image registration for varicose ulcer classification using KNN classifier”, *International Journal of Computers and Applications*, v. 40, n. 2, pp. 88–97, 2018.
- [52] PEDREGOSA, F., VAROQUAUX, G., GRAMFORT, A., et al. “Scikit-learn: Machine Learning in Python”, *Journal of Machine Learning Research*, v. 12, pp. 2825–2830, 2011.
- [53] BALA, R., KUMAR, D. “Classification using ANN: A review”, *Int. J. Comput. Intell. Res.*, v. 13, n. 7, pp. 1811–1820, 2017.
- [54] HUSH. “Classification with neural networks: a performance analysis”. In: *IEEE 1989 International Conference on Systems Engineering*, pp. 277–280, 1989. doi: 10.1109/ICSYSE.1989.48672.
- [55] ZHANG, C., PAN, X., LI, H., et al. “A hybrid MLP-CNN classifier for very fine resolution remotely sensed image classification”, *ISPRS Journal of Photogrammetry and Remote Sensing*, v. 140, pp. 133–144, 2018.
- [56] VALARMATHIE1, P., SIVAKRITHIKA, V., DINAKARAN, K. “Classification of mammogram masses using selected texture, shape and margin features with multilayer perceptron classifier”, 2016.
- [57] GUIDO, S., MULLER, A. *Introduction to Machine Learning with Python*. O’Reilly Media, 2017.
- [58] SILVA, M. C., DA SILVA, J. C. F., OLIVEIRA, R. A. R. “IDiSSC: Edge-computing-based Intelligent Diagnosis Support System for Citrus Inspection”. In: *Proceedings of the 23rd International Conference on Enterprise Information Systems - Volume 1: ICEIS,,* pp. 685–692. INSTICC, SciTePress, 2021. ISBN: 978-989-758-509-8. doi: 10.5220/0010444106850692.
- [59] LECUN, Y., BENGIO, Y., HINTON, G. “Deep learning”, *nature*, v. 521, n. 7553, pp. 436–444, 2015.
- [60] GULLI, A., PAL, S. *Deep learning with Keras*. Packt Publishing Ltd, 2017.
- [61] ZHANG, A., LIPTON, Z. C., LI, M., et al. “Dive into deep learning”, *arXiv preprint arXiv:2106.11342*, 2021.

- [62] GOODFELLOW, I., BENGIO, Y., COURVILLE, A. *Deep learning*. MIT press, 2016.
- [63] ALBAWI, S., MOHAMMED, T. A., AL-ZAWI, S. “Understanding of a convolutional neural network”. In: *2017 International Conference on Engineering and Technology (ICET)*, pp. 1–6, 2017. doi: 10.1109/ICEngTechnol.2017.8308186.
- [64] HUANG, G., LIU, Z., VAN DER MAATEN, L., et al. “Densely connected convolutional networks”. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 4700–4708, 2017.
- [65] SHERSTINSKY, A. “Fundamentals of recurrent neural network (RNN) and long short-term memory (LSTM) network”, *Physica D: Nonlinear Phenomena*, v. 404, pp. 132306, 2020.
- [66] PASCANU, R., GULCEHRE, C., CHO, K., et al. “How to construct deep recurrent neural networks”, *arXiv preprint arXiv:1312.6026*, 2013.
- [67] HOWARD, A. G., ZHU, M., CHEN, B., et al. “Mobilenets: Efficient convolutional neural networks for mobile vision applications”, *arXiv preprint arXiv:1704.04861*, 2017.
- [68] TAN, M., LE, Q. “Efficientnet: Rethinking model scaling for convolutional neural networks”. In: *International conference on machine learning*, pp. 6105–6114. PMLR, 2019.
- [69] SATYANARAYANAN, M. “The emergence of edge computing”, *Computer*, v. 50, n. 1, pp. 30–39, 2017.
- [70] ZHAO, Z., LIU, F., CAI, Z., et al. “Edge computing: platforms, applications and challenges”, *J. Comput. Res. Dev.*, v. 55, n. 2, pp. 327–337, 2018.
- [71] HONG, X., WANG, Y. “Edge computing technology: development and countermeasures”, *Strategic Study of Chinese Academy of Engineering*, v. 20, n. 2, pp. 20–26, 2018.
- [72] FOX, A., GRIFFITH, R., JOSEPH, A., et al. “Above the clouds: A berkeley view of cloud computing”, *Dept. Electrical Eng. and Comput. Sciences, University of California, Berkeley, Rep. UCB/EECS*, v. 28, n. 13, pp. 2009, 2009.
- [73] LI, X., CHEN, T., CHENG, Q., et al. “Smart applications in edge computing: Overview on authentication and data security”, *IEEE Internet of Things Journal*, v. 8, n. 6, pp. 4063–4080, 2020.

- [74] JARARWEH, Y., DOULAT, A., ALQUDAH, O., et al. “The future of mobile cloud computing: integrating cloudlets and mobile edge computing”. In: *2016 23rd International conference on telecommunications (ICT)*, pp. 1–5. IEEE, 2016.
- [75] SILINA, Y., HADDADI, H. “New Directions in Jewelry: A Close Look at Emerging Trends & Developments in Jewelry-like Wearable Devices”, p. 49–56, 2015. doi: 10.1145/2802083.2808410. Disponível em: <<https://doi.org/10.1145/2802083.2808410>>.
- [76] YOUNES, R. “Improving the Accuracy of Wearable Activity Classifiers”, p. 509–514, 2015. doi: 10.1145/2800835.2801656. Disponível em: <<https://doi.org/10.1145/2800835.2801656>>.
- [77] BERGLUND, M. E., DUVALL, J., DUNNE, L. E. “A Survey of the Historical Scope and Current Trends of Wearable Technology Applications”, p. 40–43, 2016. Disponível em: <<https://doi.org/10.1145/2971763.2971796>>.
- [78] PWC. “The wearable life 2.0 - connected living in a wearable”. 2018. Disponível em: <<https://www.pwc.se/sv/pdf-reports/the-wearable-life-2-0.pdf>>.
- [79] SILVA, M. C., OLIVEIRA, R. A. R. “Analyzing the Effect of Increased Distribution on a Wearable Appliance”, 2020.
- [80] MCROBERTS, M. *Arduino básico*. Novatec Editora, 2018.
- [81] GUBBI, J., BUYYA, R., MARUSIC, S., et al. “Internet of Things (IoT): A vision, architectural elements, and future directions”, *Future Generation Computer Systems*, v. 29, n. 7, pp. 1645–1660, 2013. ISSN: 0167-739X. doi: <https://doi.org/10.1016/j.future.2013.01.010>. Disponível em: <<https://www.sciencedirect.com/science/article/pii/S0167739X13000241>>.
- [82] AMORIM, V. J. P., SILVA, M. C., OLIVEIRA, R. A. R. “Software and Hardware Requirements and Trade-Offs in Operating Systems for Wearables: A Tool to Improve Devices’ Performance”, April 2019.
- [83] KIRBY, B. P., MOSLEY, B. “The Architecture of Wearable Technology”, 2015. doi: 10.13140/RG.2.1.1480.7205.
- [84] ISASA, J. A. E., LARSEN, P. G., HANSEN, F. O. “Energy-Aware Model-Driven Development of a Wearable Healthcare Device”, pp. 44–63, 2017.

- [85] ISASA, J. A. E., LARSEN, P. G., HANSEN, F. O. “Energy-aware model-driven development of a wearable healthcare device”. In: *Software Engineering in Health Care*, Springer, pp. 44–63, 2014.
- [86] RAAD, H. *Fundamentals of IoT and Wearable Technology Design*. Wiley-IEEE Press, 2021.
- [87] NIU, L., HAN, Q., WANG, T., et al. “Reliability-aware energy management for embedded real-time systems with (m,k)-hard timing constraint”, *Journal of Signal Processing Systems*, v. 90, n. 4, pp. 515–536, 2018.
- [88] HIREMATH, S., YANG, G., MANKODIYA, K. “Wearable Internet of Things: Concept, architectural components and promises for person-centered healthcare”, pp. 304–307, 2014. doi: 10.1109/MOBIHEALTH.2014.7015971.
- [89] CHEN, B., HU, G., HO, D. W., et al. “Distributed covariance intersection fusion estimation for cyber-physical systems with communication constraints”, *IEEE Transactions on Automatic Control*, v. 61, n. 12, pp. 4020–4026, 2016.
- [90] FIANDRINO, C., ALLIO, N., KLIAZOVICH, D., et al. “Profiling Performance of Application Partitioning for Wearable Devices in Mobile Cloud and Fog Computing”, *IEEE Access*, v. 7, pp. 12156–12166, 2019. doi: 10.1109/ACCESS.2019.2892508.
- [91] PATILT, P. A., JAGYASIT, B. G., RAVALT, J., et al. “Design and Development of Wearable Sensor Textile for Precision Agriculture”, 2015.
- [92] CHEN, J., RAN, X. “Deep Learning With Edge Computing: A Review”, *Proceedings of the IEEE*, v. 107, n. 8, pp. 1655–1674, 2019. doi: 10.1109/JPROC.2019.2921977.
- [93] DENG, S., ZHAO, H., FANG, W., et al. “Edge Intelligence: The Confluence of Edge Computing and Artificial Intelligence”, 09 2019.
- [94] LIU, L., LI, H., GRUTESER, M. “Edge Assisted Real-time Object Detection for Mobile Augmented Reality”, *The 25th Annual International Conference on Mobile Computing and Networking*, 2019.
- [95] KUMARI, P., LÓPEZ-BENÍTEZ, M., LEE, G., et al. “Wearable Internet of Things-from human activity tracking to clinical integration”, p. 2361–2364, 2017.

- [96] KAKANI, V., NGUYEN, V. H., KUMAR, B. P., et al. “A critical review on computer vision and artificial intelligence in food industry”, *Journal of Agriculture and Food Research*, v. 2, pp. 100033, 2020.
- [97] CUBERO, S., LEE, W. S., ALEIXOS, N., et al. “Automated systems based on machine vision for inspecting citrus fruits from the field to postharvest—a review”, *Food and Bioprocess Technology*, v. 9, pp. 1623–1639, 2016.
- [98] PUTRA, K. T., HARIADI, T. K., RIYADI, S., et al. “Feature Extraction for Quality Modeling of Malang Oranges on an Automatic Fruit Sorting System”. In: *2018 2nd International Conference on Imaging, Signal Processing and Communication (ICISPC)*, pp. 74–78. IEEE, 2018.
- [99] YIN, S., BI, X., NIU, Y., et al. “Hyperspectral classification for identifying decayed oranges infected by fungi”, *Emirates Journal of Food and Agriculture*, pp. 601–609, 2017.
- [100] VELUSAMY, P., RAJENDRAN, S., MAHENDRAN, R. K., et al. “Unmanned Aerial Vehicles (UAV) in Precision Agriculture: Applications and Challenges”, *Energies*, v. 15, n. 1, 2022. ISSN: 1996-1073. doi: 10.3390/en15010217. Disponível em: <<https://www.mdpi.com/1996-1073/15/1/217>>.
- [101] HARI SHANKAR, R., VEERARAGHAVAN, A., UVAIS, et al. “Application of UAV for Pest, Weeds and Disease Detection using Open Computer Vision”, pp. 287–292, 2018. doi: 10.1109/ICSSIT.2018.8748404.
- [102] ÉRIKA AKEMI SAITO MORIYA, IMAI, N. N., TOMMASELLI, A. M. G., et al. “Detection and mapping of trees infected with citrus gummosis using UAV hyperspectral data”, *Computers and Electronics in Agriculture*, v. 188, pp. 106298, 2021. ISSN: 0168-1699. doi: <https://doi.org/10.1016/j.compag.2021.106298>. Disponível em: <<https://www.sciencedirect.com/science/article/pii/S016816992100315X>>.
- [103] ROTONDO, T., FARINELLA, G. M., CHILLEMI, A., et al. “A Digital Countryside Notebook for Smart Agriculture and Oranges Classification.” In: *ICETE (1)*, pp. 547–551, 2018.
- [104] KAMILARIS, A., PRENAFETA-BOLDÚ, F. X. “Deep learning in agriculture: A survey”, *Computers and Electronics in Agriculture*, v. 147, pp. 70–90, 2018. ISSN: 0168-1699. doi: <https://doi.org/10.1016/j.compag.2018.02.016>. Disponível em: <<https://www.sciencedirect.com/science/article/pii/S0168169917308803>>.

- [105] ZHAO, S., PENG, Y., LIU, J., et al. “Tomato Leaf Disease Diagnosis Based on Improved Convolution Neural Network by Attention Module”, *Agriculture*, v. 11, n. 7, 2021. ISSN: 2077-0472. doi: 10.3390/agriculture11070651. Disponível em: <<https://www.mdpi.com/2077-0472/11/7/651>>.
- [106] TOO, E. C., YUJIAN, L., NJUKI, S., et al. “A comparative study of fine-tuning deep learning models for plant disease identification”, *Computers and Electronics in Agriculture*, v. 161, pp. 272–279, 2019. ISSN: 0168-1699. doi: <https://doi.org/10.1016/j.compag.2018.03.032>. Disponível em: <<https://www.sciencedirect.com/science/article/pii/S0168169917313303>>. BigData and DSS in Agriculture.
- [107] T., S., KHILAR, R., SUBAJA CHRISTO, M. “A comparative analysis on plant pathology classification using deep learning architecture – Resnet and VGG19”, *Materials Today: Proceedings*, 2021. ISSN: 2214-7853. doi: <https://doi.org/10.1016/j.matpr.2020.11.993>. Disponível em: <<https://www.sciencedirect.com/science/article/pii/S221478532039670X>>.
- [108] BURHAN, S. A., MINHAS, S., TARIQ, A., et al. “Comparative Study Of Deep Learning Algorithms For Disease And Pest Detection In Rice Crops”, pp. 1–5, 2020. doi: 10.1109/ECAI50035.2020.9223239.
- [109] MOHAMETH, F., BINGCAI, C., SADA, K. A. “Plant disease detection with deep learning and feature extraction using plant village”, *Journal of Computer and Communications*, v. 8, n. 6, pp. 10–22, 2020.
- [110] GARG, S., SINHA, P., SINGH, A. “Overview of Augmented Reality and Its Trends in Agriculture Industry”. In: Senjyu, T., Mahalle, P., Perumal, T., et al. (Eds.), *IOT with Smart Systems*, pp. 627–636, Singapore, 2022. Springer Nature Singapore.
- [111] YANG, X., SHU, L., CHEN, J., et al. “A Survey on Smart Agriculture: Development Modes, Technologies, and Security and Privacy Challenges”, *IEEE/CAA Journal of Automatica Sinica*, v. 8, n. 2, pp. 273–302, fev. 2021. doi: 10.1109/jas.2020.1003536. Disponível em: <<https://doi.org/10.1109/jas.2020.1003536>>.
- [112] KATSAROS, A., KERAMOPOULOS, E. “FarmAR, a farmer’s augmented reality application based on semantic web”. In: *South Eastern European Design Automation, Computer Engineering, Computer Networks and Social Media Conference (SEEDA-CECNSM)*, pp. 1–6, 2017. doi: 10.23919/SEEDA-CECNSM.2017.8088230.

- [113] CARVALHO, A., O'MAHONY, N., KRPALKOVA, L., et al. "Farming on the edge: Architectural Goals". In: *2019 8th International Conference on Agro-Geoinformatics (Agro-Geoinformatics)*, pp. 1–6, 2019. doi: 10.1109/Agro-Geoinformatics.2019.8820424.
- [114] KUMARI, S., RAGHURAM, P., VENKATESH, V., et al. "Future perspectives on progressive farming with adoption of virtual reality technology for sustainable quality in agriculture". v. 34, pp. 250–279. Emerald, ago. 2021. doi: 10.1108/tqm-06-2021-0191. Disponível em: <<https://doi.org/10.1108/tqm-06-2021-0191>>.
- [115] ZHANG, X., CAO, Z., DONG, W. "Overview of Edge Computing in the Agricultural Internet of Things: Key Technologies, Applications, Challenges", *IEEE Access*, v. 8, pp. 141748–141761, 2020. doi: 10.1109/ACCESS.2020.3013005.
- [116] NALINAKSH, K., RISHIWAL, V., GANZHA, M., et al. "An Internet of Things (IoT) solution framework for agriculture in India and other Third World countries". In: *2018 Fourth International Conference on Advances in Computing, Communication Automation (ICACCA)*, pp. 1–6, 2018. doi: 10.1109/ICACCAF.2018.8776792.
- [117] KARAR, M. E., ALSUNAYDI, F., ALBUSAYMI, S., et al. "A new mobile application of agricultural pests recognition using deep learning in cloud computing system", *Alexandria Engineering Journal*, v. 60, n. 5, pp. 4423–4432, 2021. ISSN: 1110-0168. doi: <https://doi.org/10.1016/j.aej.2021.03.009>. Disponível em: <<https://www.sciencedirect.com/science/article/pii/S1110016821001642>>.
- [118] VRIGKAS, M., NIKOU, C., KAKADIARIS, I. A. "A review of human activity recognition methods", *Frontiers in Robotics and AI*, v. 2, pp. 28, 2015.
- [119] DANG, L. M., MIN, K., WANG, H., et al. "Sensor-based and vision-based human activity recognition: A comprehensive survey", *Pattern Recognition*, v. 108, pp. 107561, 2020.
- [120] BEDDIAR, D. R., NINI, B., SABOKROU, M., et al. "Vision-based human activity recognition: a survey", *Multimedia Tools and Applications*, v. 79, n. 41, pp. 30509–30555, 2020.
- [121] KIM, K., JALAL, A., MAHMOOD, M. "Vision-based human activity recognition system using depth silhouettes: A smart home system for monitoring

- the residents”, *Journal of Electrical Engineering & Technology*, v. 14, n. 6, pp. 2567–2573, 2019.
- [122] TIEN, P. W., WEI, S., CALAUTIT, J. K., et al. “Vision-based human activity recognition for reducing building energy demand”, *Building Services Engineering Research and Technology*, v. 42, n. 6, pp. 691–713, 2021.
- [123] CHEN, K., ZHANG, D., YAO, L., et al. “Deep learning for sensor-based human activity recognition: Overview, challenges, and opportunities”, *ACM Computing Surveys (CSUR)*, v. 54, n. 4, pp. 1–40, 2021.
- [124] NWEKE, H. F., TEH, Y. W., AL-GARADI, M. A., et al. “Deep learning algorithms for human activity recognition using mobile and wearable sensor networks: State of the art and research challenges”, *Expert Systems with Applications*, v. 105, pp. 233–261, 2018.
- [125] GU, T., WANG, L., WU, Z., et al. “A pattern mining approach to sensor-based human activity recognition”, *IEEE Transactions on Knowledge and Data Engineering*, v. 23, n. 9, pp. 1359–1372, 2010.
- [126] NWEKE, H. F., TEH, Y. W., MUJTABA, G., et al. “Data fusion and multiple classifier systems for human activity detection and health monitoring: Review and open research directions”, *Information Fusion*, v. 46, pp. 147–170, 2019.
- [127] UD DIN TAHIR, S. B., JALAL, A., BATOOL, M. “Wearable sensors for activity analysis using SMO-based random forest over smart home and sports datasets”. In: *2020 3rd International Conference on Advancements in Computational Sciences (ICACS)*, pp. 1–6. IEEE, 2020.
- [128] RAMASAMY RAMAMURTHY, S., ROY, N. “Recent trends in machine learning for human activity recognition—A survey”, *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, v. 8, n. 4, pp. e1254, 2018.
- [129] WAN, S., QI, L., XU, X., et al. “Deep learning models for real-time human activity recognition with smartphones”, *Mobile Networks and Applications*, v. 25, n. 2, pp. 743–755, 2020.
- [130] ZHOU, X., LIANG, W., KEVIN, I., et al. “Deep-learning-enhanced human activity recognition for Internet of healthcare things”, *IEEE Internet of Things Journal*, v. 7, n. 7, pp. 6429–6438, 2020.

- [131] BIANCHI, V., BASSOLI, M., LOMBARDO, G., et al. “IoT wearable sensor and deep learning: An integrated approach for personalized human activity recognition in a smart home environment”, *IEEE Internet of Things Journal*, v. 6, n. 5, pp. 8553–8562, 2019.
- [132] HSU, Y.-L., YANG, S.-C., CHANG, H.-C., et al. “Human daily and sport activity recognition using a wearable inertial sensor network”, *IEEE Access*, v. 6, pp. 31715–31728, 2018.
- [133] IMRAN, H. A. “Khail-Net: A Shallow Convolutional Neural Network for Recognizing Sports Activities Using Wearable Inertial Sensors”, *IEEE Sensors Letters*, v. 6, n. 9, pp. 1–4, 2022.
- [134] STEELS, T., VAN HERBRUGGEN, B., FONTAINE, J., et al. “Badminton activity recognition using accelerometer data”, *Sensors*, v. 20, n. 17, pp. 4685, 2020.
- [135] MEKRUKSAVANICH, S., JITPATTANAKUL, A. “Biometric User Identification Based on Human Activity Recognition Using Wearable Sensors: An Experiment Using Deep Learning Models”, *Electronics*, v. 10, n. 3, 2021. ISSN: 2079-9292. doi: 10.3390/electronics10030308. Disponível em: <<https://www.mdpi.com/2079-9292/10/3/308>>.
- [136] ROMIJNDERS, R., WARMERDAM, E., HANSEN, C., et al. “A Deep Learning Approach for Gait Event Detection from a Single Shank-Worn IMU: Validation in Healthy and Neurological Cohorts”, *Sensors*, v. 22, n. 10, 2022. ISSN: 1424-8220. doi: 10.3390/s22103859. Disponível em: <<https://www.mdpi.com/1424-8220/22/10/3859>>.
- [137] WANG, Y., CANG, S., YU, H. “A survey on wearable sensor modality centred human activity recognition in health care”, *Expert Systems with Applications*, v. 137, pp. 167–190, 2019. ISSN: 0957-4174. doi: <https://doi.org/10.1016/j.eswa.2019.04.057>.
- [138] CELIK, Y., STUART, S., WOO, W. L., et al. “Wearable Inertial Gait Algorithms: Impact of Wear Location and Environment in Healthy and Parkinson’s Populations”, *Sensors*, v. 21, n. 19, 2021. ISSN: 1424-8220. doi: 10.3390/s21196476.
- [139] BIJALWAN, V., SEMWAL, V. B., MANDAL, T. “Fusion of multi-sensor-based biomechanical gait analysis using vision and wearable sensor”, *IEEE Sensors Journal*, v. 21, n. 13, pp. 14213–14220, 2021.

- [140] HURTADO, S., GARCÍA-NIETO, J., POPOV, A., et al. “Human Activity Recognition From Sensorised Patient’s Data in Healthcare: A Streaming Deep Learning-Based Approach”, 2023.
- [141] LU, J., ZHENG, X., SHENG, M., et al. “Efficient Human Activity Recognition Using a Single Wearable Sensor”, *IEEE Internet of Things Journal*, v. 7, n. 11, pp. 11137–11146, 2020. doi: 10.1109/JIOT.2020.2995940.
- [142] HUHNS, M., WILLIAMS, L. *Software Engineering in Health Care: 4th International Symposium, FHIES 2014, and 6th International Workshop, SEHC 2014, Washington, DC, USA, July 17-18, 2014, Revised Selected Papers*. 01 2017. ISBN: 978-3-319-63193-6. doi: 10.1007/978-3-319-63194-3.
- [143] PIPATNORASETH, T., PHOGNSUPHAP, S., WIRATKAPUN, C., et al. “Breast Microcalcification Visualization Using Pseudo-Color Image Processing”. In: *2019 12th Biomedical Engineering International Conference (BMEiCON)*, pp. 1–5. IEEE, 2019.
- [144] JAI. “Prism-based multispectral cameras empower high speed fruit sorting”, 2020. Disponível em: <<https://news.jai.com/blog>>.
- [145] DA SILVA, J. C. F. “Orange diseases dataset”, 2021. Disponível em: <<https://www.kaggle.com/datasets/jonathansilva2020/orange-diseases-dataset>>.
- [146] GUAZZELLI, A., ZELLER, M., LIN, W.-C., et al. “PMML: An open standard for sharing models”, *The R Journal*, v. 1, n. 1, pp. 60–65, 2009.
- [147] CHOLLET, F., OTHERS. “Keras”. <https://keras.io>, 2015.
- [148] XIANG, Q., WANG, X., LI, R., et al. “Fruit image classification based on Mobilenetv2 with transfer learning technique”. In: *Proceedings of the 3rd International Conference on Computer Science and Application Engineering*, pp. 1–7, 2019.
- [149] ÇAKMAK, M., TENEKECI, M. E. “Melanoma detection from dermoscopy images using Nasnet Mobile with Transfer Learning”. In: *2021 29th Signal Processing and Communications Applications Conference (SIU)*, pp. 1–4. IEEE, 2021.
- [150] HE, K., ZHANG, X., REN, S., et al. “Deep residual learning for image recognition”. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 770–778, 2016.

- [151] BI, C., WANG, J., DUAN, Y., et al. “MobileNet based apple leaf diseases identification”, *Mobile Networks and Applications*, pp. 1–9, 2020.
- [152] ZOPH, B., LE, Q. V. “Neural architecture search with reinforcement learning”, *arXiv preprint arXiv:1611.01578*, 2016.
- [153] ZOPH, B., VASUDEVAN, V., SHLENS, J., et al. “Learning transferable architectures for scalable image recognition”. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 8697–8710, 2018.
- [154] HOCHREITER, S., SCHMIDHUBER, J. “Long short-term memory”, *Neural computation*, v. 9, n. 8, pp. 1735–1780, 1997.
- [155] GRAVES, A. “Long short-term memory”, *Supervised sequence labelling with recurrent neural networks*, pp. 37–45, 2012.
- [156] CHUNG, J., GULCEHRE, C., CHO, K., et al. “Empirical evaluation of gated recurrent neural networks on sequence modeling”, *arXiv preprint arXiv:1412.3555*, 2014.
- [157] KOUTNIK, J., GREFF, K., GOMEZ, F., et al. “A clockwork rnn”. In: *International Conference on Machine Learning*, pp. 1863–1871. PMLR, 2014.
- [158] “Dataset for human activity recognition HAR”. <https://www.kaggle.com/datasets/jonathansilva2020/recognition-of-human-activity-har-four-classes>, .
- [159] RAVI, N., DANDEKAR, N., MYSORE, P., et al. “Activity recognition from accelerometer data”. In: *Aaai*, v. 5, pp. 1541–1546. Pittsburgh, PA, 2005.
- [160] “tsaug documentation”. <https://tsaug.readthedocs.io/en/stable/>. Accessed: 2022-12-10.
- [161] “Recognition of human activity with Data Agumentation”. <https://www.kaggle.com/datasets/jonathansilva2020/recognition-of-human-activity-data-agumentation>, .
- [162] SILVA, M., DA SILVA, J. F., OLIVEIRA, R. “IDiSSC: Edge-computing-based Intelligent Diagnosis Support System for Citrus Inspection”. In: *Proceedings of the 23rd International Conference on Enterprise Information Systems*. SCITEPRESS - Science and Technology Publications, 2021. doi: 10.5220/0010444106850692. Disponível em: <<https://doi.org/10.5220/0010444106850692>>.

- [163] SILVA. “Multiclass image classification mobilenet”. 2021. Disponível em: <https://github.com/JonathanCristovao/Multi-class-image-classification-mobilenet.git>.
- [164] SEMWAL, V. B., GAUD, N., LALWANI, P., et al. “Pattern identification of different human joints for different human walking styles using inertial measurement unit (IMU) sensor”, *Artificial Intelligence Review*, v. 55, n. 2, pp. 1149–1169, 2022.
- [165] BIJALWAN, V., SEMWAL, V. B., SINGH, G., et al. “Heterogeneous computing model for post-injury walking pattern restoration and postural stability rehabilitation exercise recognition”, *Expert Systems*, v. 39, n. 6, pp. e12706, 2022.
- [166] CHALLA, S. K., KUMAR, A., SEMWAL, V. B. “A multibranch CNN-BiLSTM model for human activity recognition using wearable sensor data”, *The Visual Computer*, v. 38, n. 12, pp. 4095–4109, 2022.