REVISTA CIENTÍFICA - RECIMA21 ISSN 2675-6218

IDENTIFICAÇÃO DE MOVIMENTAÇÕES HUMANAS UTILIZANDO WI-FI COM *CHANNEL STATE INFORMATION*

HUMAN MOVEMENT IDENTIFICATION USING WI-FI WITH CHANNEL STATE INFORMATION

IDENTIFICACIÓN DE MOVIMIENTOS HUMANOS MEDIANTE WI-FI CON INFORMACIÓN DEL ESTADO DEL CANAL

Maria Fernanda Cerne Magalhães[1], André Luiz da Silva[2], Fabiana Florian[3]

e6116992
https://doi.org/10.47820/recima21.v6i11.6992

PUBLICADO: 11/2025

RESUMO

Esta pesquisa descreve o desenvolvimento de um sistema para a identificação de movimentações através de informações físicas do sinal WiFi como amplitude e fase, denominada *Channel State Information* (CSI). Com a captura de dados em um ambiente controlado utilizando o microcomputador Raspberry Pi 4, com um *firmware* customizado e com componentes para a emissão e transmissão de sinais Wi-Fi. Logo, os dados foram classificados por modelos de *Machine Learning* como SVM, Random Forest e KNN. Os resultados do experimento demonstraram a capacidade de detecção, concluindo-se que o CSI é uma alternativa não intrusiva para a identificação de movimentações de métodos tradicionais de monitoramento por imagens ou áudio.

PALAVRAS-CHAVE: *Channel State Information*. Inteligência Artificial. Processamento.

ABSTRACT

This research describes the development of a system for the identification of movements using physical information from the Wi-Fi signal, such as amplitude and phase, known as Channel State Information (CSI). Data is captured in a controlled environment using a Raspberry Pi 4 microcomputer with custom firmware and components for the emission and transmission of Wi-Fi signals. Subsequently, the data is classified using Machine Learning models, such as SVM, Random Forest, and KNN. The experimental results demonstrate the capability of the system to detect movements, indicating that CSI is a non-intrusive alternative for movement identification, superior to traditional monitoring methods using images or audio.

KEYWORDS: *Channel State Information. Artificial Intelligence. Processing.*

RESUMEN

Esta investigación describe el desarrollo de un sistema para la identificación de movimientos mediante información física proveniente de señales Wi-Fi, como la amplitud y la fase, conocida como Channel State Information (CSI). Los datos se capturaron en un entorno controlado utilizando una microcomputadora Raspberry Pi 4 con firmware y componentes personalizados para la emisión y transmisión de señales Wi-Fi. Posteriormente, los datos se clasificaron mediante modelos de aprendizaje automático como SVM, Random Forest y KNN. Los resultados del experimento demostraron la capacidad del sistema para detectar movimiento, concluyendo que la CSI es una alternativa no intrusiva a los métodos tradicionales de monitorización de imágenes o audio para la identificación de movimientos.

PALABRAS CLAVE*: Channel State Information. Inteligencia artificial. Procesamiento.*

[1] Graduanda em Engenharia da Computação pela Universidade de Araraquara - UNIARA.
[2] Prof. Dr. na Universidade de Araraquara - UNIARA.
[3] Docente na Universidade de Araraquara - UNIARA. Doutora em Alimentos e Nutrição pela UNESP (FCFAR).

ISSN: 2675-6218 - RECIMA21

Este artigo é publicado em acesso aberto (Open Access) sob a licença Creative Commons Atribuição 4.0 Internacional (CC-BY), que permite uso, distribuição e reprodução irrestritos em qualquer meio, desde que o autor original e a fonte sejam creditados.

INTRODUÇÃO

Os avanços em tecnologia computacional impulsionaram o desenvolvimento de sistemas de monitoramento complexos, aplicados desde a segurança patrimonial até o cuidado com idosos e crianças. Contudo, soluções tradicionais baseadas em câmeras e microfones são intrusivas e levantam sérias preocupações sobre a privacidade. Esses métodos são suscetíveis a ataques maliciosos, expondo a intimidade dos indivíduos e criando vulnerabilidades significativas.

Como alternativa, propõe-se um monitoramento não intrusivo que permita identificar movimentações de pessoas por meio da análise de uma propriedade do sinal Wi-Fi: o *Channel State Information* (CSI). Esta técnica captura informações físicas das ondas eletromagnéticas, como amplitude e fase. Ao analisar as variações causadas pela movimentação humana, é possível determinar a presença de pessoas ou identificar atividades, preservando integralmente a privacidade, pois não há coleta de dados pessoais sensíveis como áudio ou vídeo.

A literatura recente validou essa migração. Damodaran *et al.,* (2020) destacam que soluções visuais são limitadas pela linha de visão (LOS) e iluminação, além de serem intrusivas, o que impulsiona a pesquisa em sensoriamento não intrusivo como o Wi-Fi. No entanto, um desafio significativo persiste: a interferência causada por múltiplos usuários. Wang, Wu e Ni (2016) apontam a dificuldade em identificar atividades distintas quando várias pessoas estão presentes, pois os movimentos combinados geram interferências complexas no sinal. Segundo os autores, construir um banco de dados treinado para cobrir essa variação é uma tarefa de custo elevado.

Diante dessa lacuna, o objetivo deste trabalho é desenvolver e validar um sistema de identificação de movimentações em ambiente interno, utilizando as propriedades físicas do sinal Wi-Fi (CSI). Para garantir alta acurácia e confiabilidade frente aos desafios de ruído e interferência, os dados coletados são submetidos a filtragem e processados por algoritmos de *Machine Learning*. Desta forma, busca-se entregar um sistema de monitoramento eficiente que, por operar sem a captura de dados intrusivos, soluciona o conflito entre monitoramento e privacidade.

1. REFERENCIAL TEÓRICO

1. Wi-Fi (IEEE 802.11)

Segundo Tanenbaum, Feamster e Wetherall (2021), como existem muitos fornecedores e fabricantes com ideias próprias sobre redes, isso poderia causar confusão e dificultar a comunicação entre dispositivos. Para evitar esse problema, a indústria criou padrões de redes, permitindo a interoperabilidade entre diferentes computadores. O IEEE (*Institute of Electrical and Electronics Engineers*), a maior organização profissional do mundo, desenvolve padrões nas áreas de informática e engenharia elétrica. Seu comitê 802 foi responsável por padronizar diversos tipos de redes.

Este artigo é publicado em acesso aberto (Open Access) sob a licença Creative Commons Atribuição 4.0 Internacional (CC-BY), que permite uso, distribuição e reprodução irrestritos em qualquer meio, desde que o autor original e a fonte sejam creditados.

Segundo Kurose e Ross (2021), os diversos padrões da família IEEE 802.11, como os padrões b, g, n, ac e ax, foram criados sucessivamente ao longo do tempo para redes sem fio. Esses padrões operam em duas principais faixas de frequência: de 2,4 a 2,485 GHz e de 5,1 a 5,8 GHz. A principal diferença entre eles é que novas tecnologias são implementadas com o decorrer do tempo e adicionadas a novos padrões.

A faixa de 5 GHz oferece menor alcance de transmissão para a mesma potência, além de sofrer maior impacto da propagação por múltiplos caminhos. Já a faixa de 2,4 GHz não requer licença para uso, porém pode sofrer interferência por compartilhar o espectro com outros dispositivos eletrônicos, como telefones sem fio e micro-ondas (Kurose; Ross, 2021).

1.1.1. Técnicas de modulação e transmissão

A multiplexação por divisão ortogonal de frequência (*Orthogonal Frequency Division Multiplexing* – OFDM) é uma técnica de modulação empregada em redes IEEE 802.11, além de ser amplamente utilizada em redes Wi-Fi, sistemas de comunicação por linhas de energia elétrica, redes a cabo e sistemas celulares. Segundo Tanenbaum, Feamster e Wetherall (2021), o OFDM divide a largura de banda do canal em múltiplas subportadoras, que transmitem dados de forma paralela e independente. Para preservar essa ortogonalidade e evitar interferências causadas pela dispersão temporal do canal, é inserido um intervalo de guarda – uma repetição parcial do símbolo no tempo. Essa estrutura reduz a degradação do canal e torna o sistema mais robusto a efeitos como desvanecimento seletivo e interferências. Além disso, o OFDM proporciona alta eficiência espectral e maior resistência à deterioração do sinal causada por múltiplos caminhos de propagação.

O MIMO (*Multiple Input, Multiple Output* – múltiplas entradas e múltiplas saídas) foi introduzido no padrão IEEE 802.11n, sendo uma das principais inovações nas redes Wi-Fi atuais. De acordo com Tanenbaum, Feamster e Wetherall (2021), essa tecnologia permite o uso simultâneo de até quatro antenas transmissoras e quatro receptoras, possibilitando a transmissão paralela de múltiplos fluxos independentes de dados. Apesar de esses sinais se combinarem no receptor, técnicas avançadas de processamento de sinal permitem sua separação eficaz. O uso de múltiplas antenas proporciona não apenas o aumento da taxa de transmissão, mas também maior confiabilidade e alcance do sinal, mesmo em ambientes com interferência ou obstáculos.

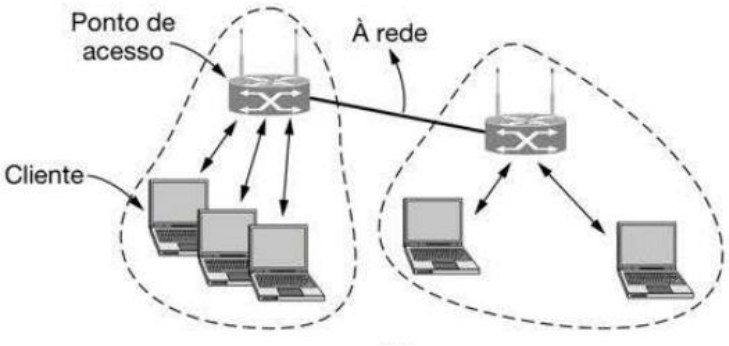
1.1.2. Arquitetura de uma rede Wi-Fi

Como apontado por Tanenbaum, Feamster e Wetherall (2021), as redes 802.11, conhecidas como o padrão Wi-Fi, possuem uma arquitetura padrão, conforme a Figura 1, na qual clientes, como celulares, computadores que se conectam a outra rede, como a Internet. A infraestrutura deste modo permite que cada cliente se associe a um ponto de acesso (PA) que está conectado a outra rede. Isso faz com que o cliente possa transmitir e receber pacotes por meio do ponto de acesso.

ISSN: 2675-6218 - RECIMA21
Este artigo é publicado em acesso aberto (Open Access) sob a licença Creative Commons Atribuição 4.0 Internacional (CC-BY), que permite uso, distribuição e reprodução irrestritos em qualquer meio, desde que o autor original e a fonte sejam creditados.

Assim, os PAs podem se conectar, através de uma rede com fio, formando uma rede 802.11 estendida. Neste modo, permite que clientes possam enviar quadros para outros clientes por via de seus PAs.

Figura 1. Arquitetura 802.11 _ modo de infraestrutura



Fonte: TANENBAUM; FEAMSTER; WETHERALL, 2021, p. 200

1.1.3. Largura de banda e canais

Segundo Tanenbaum, Feamster e Wetherall (2021), a largura de banda é uma medida física da capacidade de transporte de um meio, normalmente ela é medida em Hz. Existem filtros que são utilizados para limitar o volume de largura de banda de um sinal. Em canais sem fio, como 802.11 possui permissão para usar até 20MHz, assim os rádios 802.11 conseguem filtrar a largura de banda do sinal para essa faixa, existem também bandas de 80MHz. Como sempre há ruído aleatório (térmico), o volume do ruído é medido pela relação da potência do sinal e a do ruído, chamada relação sinal/ruído, também conhecida SNR (*Signal-to-Noise Ratio*).

1.1.4. Fenômenos de propagação

Um dos principais efeitos que impactam a propagação dos sinais é o fenômeno de propagação multivias, podendo ser chamado também de multicaminho. Como é descrito por Tanenbaum, Feamster e Wetherall (2021), esse fenômeno ocorre quando as ondas eletromagnéticas refletem em objetos e no solo, fazendo com que as ondas tomem caminhos diferentes e comprimentos entre o emissor e o receptor. Podendo causar um embaralhamento do sinal recebido. Além disso, o movimento de objetos entre o receptor e o emissor gera uma propagação multivias que mudam no decorrer do tempo, o que é de extrema importância para técnicas de detecção baseadas em *Channel State Information*.

ISSN: 2675-6218 - RECIMA21
Este artigo é publicado em acesso aberto (Open Access) sob a licença Creative Commons Atribuição 4.0 Internacional (CC-BY), que permite uso, distribuição e reprodução irrestritos em qualquer meio, desde que o autor original e a fonte sejam creditados.

1.1.5. Interferências, ruídos e atenuação em ambientes Wi-fi

Conforme descrito por Tanenbaum, Feamster e Wetherall (2021), outro fator é quando muitas fontes de rádio utilizam a mesma banda de frequência e acabam gerando interferência umas nas outras. Também há interferência através de fontes que transmitem, criando um ruído eletromagnético no ambiente gerando interferências, fontes que normalmente operam em 2,4 GHz. O que torna os padrões 802.11 mais recentes preferíveis, pois operam em 5GHz de frequência de banda.

Como descrito por Tanenbaum, Feamster e Wetherall (2021), para redes sem fio qualquer radiação eletromagnética pode dispersar um sinal, reduzindo a força, podendo ser chamada de atenuação de percurso para alguns casos, que também ocorre com o aumento da distância do receptor e emissor.

1.2. Detalhes *Channel State Information* (CSI)

O *Channel State Information* (CSI) é um conjunto de informações sobre como as ondas eletromagnéticas se propagam entre o receptor e o transmissor Wi-Fi. Essas informações permitem extrair dados pormenorizados sobre o ambiente no qual o sinal se propaga, revelando interações sutis com objetos e pessoas.

Conforme Trindade, Almeida e Pereira Junior (2024, p. 6):

> O CSI do Wi-Fi registra informações sobre como os sinais sem fio se propagam do transmissor para o receptor, descrevendo o comportamento das ondas eletromagnéticas ao longo das frequências. Essas informações incluem a amplitude e a fase do sinal, que podem ser alteradas por reflexões, obstruções e outros fatores no ambiente. Cada elemento do CSI mostra como o ambiente impacta a propagação do sinal, formando uma função chamada Resposta em Frequência do Canal (CFR). Em sistemas Wi-Fi que usam múltiplas antenas, a tecnologia divide o espectro em várias subportadoras através da técnica chamada OFDM (*Multiplexação por Divisão de Frequência Ortogonal*). O transmissor envia sinais de treinamento especiais no início da transmissão, que o receptor usa para estimar como o canal Wi-Fi afeta cada subportadora. Essencialmente, isso ajuda o receptor a entender o comportamento do sinal em diferentes condições, ajustando-se para garantir uma melhor transmissão e recepção dos dados.

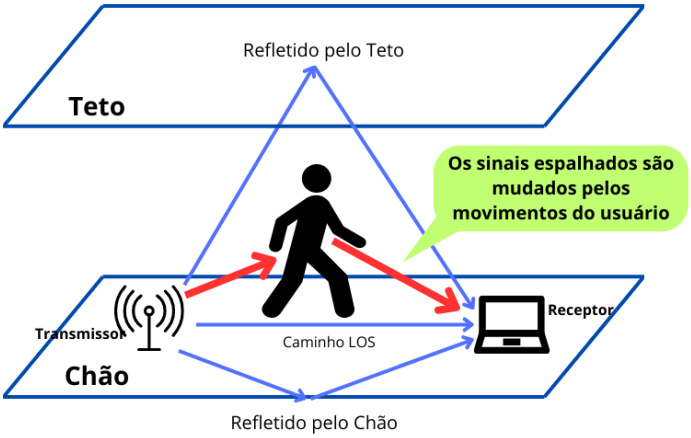
1.2.1. Coleta e identificação de CSI

A obtenção do *Channel State Information* inicia-se com a transmissão de símbolos pré-definidos para cada subportadora, denominados *Long Training Symbols* (LTFs), pelo dispositivo transmissor. Ao receber os símbolos, o receptor é capaz de estimar a matriz CSI, comparando o sinal recebido com a sequência original LTFs. Para cada portadora, o Wi-Fi pode ser descrito como: $y = Hx + n$, onde y é o sinal recebido, x é o sinal transmitido, H é a matriz CSI e n é o vetor de ruído. O receptor estima a matriz CSI H utilizando um sinal pré-definido sinal x e um sinal recebido y, assim removendo o prefixo cíclico, *demapping e a* demodulação OFDM (Ma; Zhou; Wang, 2019).

Este artigo é publicado em acesso aberto (Open Access) sob a licença Creative Commons Atribuição 4.0 Internacional (CC-BY), que permite uso, distribuição e reprodução irrestritos em qualquer meio, desde que o autor original e a fonte sejam creditados.

Como descrito por Wang *et al.,* (2018), em um ambiente fechado com um receptor e um transmissor, o sinal de Wi-Fi irá se propagar por múltiplos caminhos e ficará estável se não houver movimentações nem pessoas. No entanto, quando uma pessoa se move os sinais que estão espalhados se alteram, causando perturbações no canal e causando a atenuação da amplitude e a distorção de fase. Portanto, diferentes efeitos de multicaminhos podem ser obtidos se uma pessoa se mover, conforme a Figura 2, o que resulta em uma diferença nos dados CSI coletados pelo receptor. Essas variações podem ser utilizadas para reconhecer diferentes comportamentos ao compará-los com padrões de distorção do canal correspondentes.

Figura 2. Sinal Wi-Fi propagado em ambientes internos



Fonte: Adaptado de Wang *et al.,* (2018, p. 2)

1.3. Inteligência Artificial

Para identificar padrões nas movimentações registradas nos conjuntos de dados CSI, utiliza-se a Inteligência Artificial, que permite às máquinas reconhecerem e classificarem diferentes tipos de movimentos em um ambiente. Segundo a IBM ([s.d]-b), a Inteligência Artificial possibilita que computadores aprendam, compreendam, solucionem problemas e tomem decisões de maneira autônoma, atuando de forma independente e complementando a inteligência humana.

1.3.1. *Machine Learning*

O *Machine Learning* é uma subárea da Inteligência Artificial voltada para o desenvolvimento de modelos capazes de aprender com dados. De acordo com a IBM ([s.d]-b), essa abordagem permite a criação de modelos treinados por algoritmos que tomam decisões e realizam previsões com base nos dados disponíveis. Entre os algoritmos mais comuns, destacam-se a regressão linear, regressão logística, *decision trees*, máquinas de vetores de suporte (SVMs), k-vizinhos mais próximos (KNN), Naive Bayes e técnicas de agrupamento (*clustering*).

Este artigo é publicado em acesso aberto (Open Access) sob a licença Creative Commons Atribuição 4.0 Internacional (CC-BY), que permite uso, distribuição e reprodução irrestritos em qualquer meio, desde que o autor original e a fonte sejam creditados.

No contexto do aprendizado supervisionado, os modelos são treinados a partir de dados de entrada e saída previamente rotulados por humanos, permitindo que aprendam relações subjacentes e realizem previsões corretas sobre novos dados não rotulados (IBM, [s.d]-a). Essa técnica foi utilizada no desenvolvimento desta pesquisa para a classificação de movimentos a partir de sinais CSI.

2. MÉTODO

Esta seção aborda o desenvolvimento do sistema de detecção de movimentos através do *Channel State Information* (CSI). Nesta seção, são detalhados os componentes de *hardware* e *software* utilizados, a arquitetura do sistema e a configuração do ambiente experimental para a coleta e processamento dos dados.

Trata-se de um estudo experimental de delineamento quantitativo, focado no desenvolvimento e na validação de um sistema de classificação de atividades humanas. A metodologia foi dividida em três etapas principais: configuração do ambiente e coleta de dados brutos de CSI; pré-processamento, filtragem e extração de características; e treinamento, otimização e avaliação de modelos de *Machine Learning*.

3. DESENVOLVIMENTO

3.1. Arquitetura do Sistema e Componentes

O sistema para a detecção de movimentações é configurado com três elementos principais. As configurações gerais do sistema são:

1. Roteador Wi-Fi (Ponto de Acesso e Receptor): É o ponto de acesso (AP) da rede Wi-Fi. Também serve como destinatário dos pacotes de dados enviados pelo laptop, funcionando como receptor do tráfego monitorado.
2. *Laptop Dell Inspirion* (Transmissor de Tráfego): Gera tráfego na rede, como cliente do Ponto de Acesso, envia pacotes contínuos ao roteador (como *pings*), o fluxo destes pacotes é a base fundamental para a extração do CSI.
3. Raspberry Pi 4 Modelo B (Monitor e Coletor CSI): Configurado com o *firmware* Nexmon CSI, monitora passivamente a comunicação entre o laptop e o roteador, extraindo os dados CSI dos pacotes que trafegam pelo ar, sem interferir na comunicação original.

3.2. Configurações dos componentes de *hardware* e *software*

Nesta seção são abordadas as configurações dos componentes para iniciar a coleta dos dados para a identificação de movimentações.

Este artigo é publicado em acesso aberto (Open Access) sob a licença Creative Commons Atribuição 4.0 Internacional (CC-BY), que permite uso, distribuição e reprodução irrestritos em qualquer meio, desde que o autor original e a fonte sejam creditados.

3.3. Raspberry PI 4 Modelo B - Configuração do nexmon CSI

O *firmware* Nexmon CSI foi instalado e configurado no Raspberry Pi 4 Modelo B. O processo de configuração para a coleta dos dados CSI envolveu as seguintes etapas:

A primeira etapa consiste na execução da ferramenta *makecsiparams*, incluída no Nexmon CSI. Esta ferramenta gera uma sequência de caracteres alfanuméricos com base em parâmetros específicos definidos pelo usuário. Esta sequência de letras geradas torna-se um 'código de ativação', conforme Figura 3. Esses parâmetros definem como a coleta será realizada. O parâmetro *-c* 157/80 indica que o canal Wi-Fi selecionado é o 157 e a largura de banda configurada é de 80 MHz. Os parâmetros *-C* 1 e *-N* 1 definem, respectivamente, o número de cadeias de recepção (uma antena) e fluxos espaciais (o caminho que o sinal percorre o espaço) que serão capturados. O parâmetro -m é utilizado para filtrar os dados coletados, pois considera apenas os pacotes que têm o endereço MAC de um dispositivo específico, no caso o *laptop* Dell Inspirion, pois ele enviou os pacotes. Por fim, -b 0x88 define um filtro que seleciona pacotes que começam com o *byte* 0x88, este valor é definido para que seja possível a depuração do sistema.

Figura 3. Nexmon CSI: Geração de Parâmetros

```
root@raspberrypi:/home/
                      # ./makecsiparams -c 157/80 -C 1 -N 1 -m
 -b 0x88
```

Fonte: Autoria Própria

Antes de ativar o extrator CSI, a interface de rede sem fio precisa ser configurada para o modo de monitoramento. Isso envolve encerrar processos conflitantes, como o *wpa_supplicant,* que é responsável por gerenciar conexões Wi-Fi, conforme Figura 4.

Figura 4. Nexmon CSI: Encerrando Processo de gerenciamento de Conexões

```
root@raspberrypi:/home
                      # pkill wpa_supplicant
```

Fonte: Autoria Própria

Em seguida, é criada uma nova interface de rede virtual, denominada `*mon0*`, do tipo monitor. Essa interface é dedicada exclusivamente à captura passiva de tráfego de rede, permitindo que o Raspberry Pi "escute" todos os pacotes Wi-Fi no alcance sem a necessidade de se associar a uma rede, conforme a Figura 5.

Figura 5. Criação da Interface Monitor

```
root@raspberrypi:/home/                                  # nexutil -Iwlan0
-s500 -b -l34 -vm+IBEQGIAQBQhJJ9KhIAAAAAAAAAAAAAAAAAAAAAAAAAAA==
```

Fonte: Autoria Própria

ISSN: 2675-6218 - RECIMA21

Este artigo é publicado em acesso aberto (Open Access) sob a licença Creative Commons Atribuição 4.0 Internacional (CC-BY), que permite uso, distribuição e reprodução irrestritos em qualquer meio, desde que o autor original e a fonte sejam creditados.

Após a criação da interface de monitoramento é necessário ativá-la, conforme demonstrado na Figura 6.

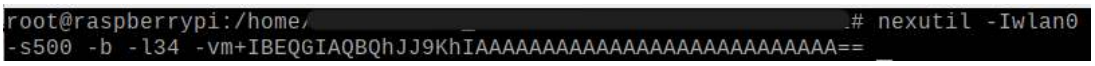
Figura 6. Ativando Interface



Fonte: Autoria Própria

Este é o comando principal que ativa o *firmware* Nexmon CSI na interface -I wlan0, que é a interface física do chip de rede do Raspberry Pi. O parâmetro -s 500 é utilizado para enviar um comando específico de configuração ao chip. O parâmetro -b indica que o valor seguinte deve ser interpretado como um conjunto de caracteres codificados em Base64, e -l 34 especifica o comprimento do *buffer* (armazenador) customizado em 34 *bytes*, que correspondem ao tamanho do conjunto de letras da configuração CSI gerada anteriormente. Finalmente, o parâmetro -v, seguido do conjunto de caracteres gerados anteriormente, passa a *string* de parâmetro gerada por *makecsiparams*, a qual contém todas as configurações necessárias para a extração do CSI. A Figura 8 ilustra a execução deste comando.

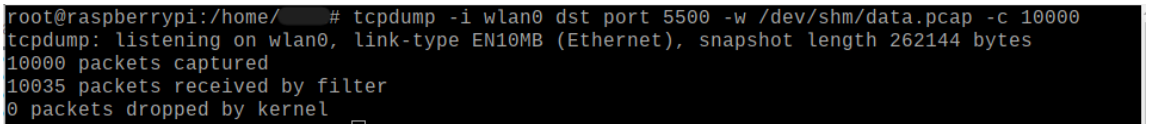
Figura 7. Ativando o extrator



Fonte: Autoria Própria

Para a coleta dos dados, o comando *tcpdump* é utilizado para capturar os pacotes que contêm CSI. O parâmetro -i wlan0 especifica a interface de rede para a captura. A opção *dst port* 5500 filtra o tráfego, capturando exclusivamente pacotes UDP, o *firmware* Nexmon CSI, por padrão, envia os dados CSI para essa porta específica. O parâmetro *-w /dev/shm/data.pcap* direciona o salvamento dos pacotes para o arquivo data.pcap no diretório /dev/shm, que é um sistema de arquivos baseado em memória RAM, otimizando a velocidade de gravação. Por fim, *-c* 1000 define o limite de 1000 pacotes a serem capturados, controlando a duração da coleta, sendo salvos em um arquivo .pcap. A execução deste comando é demonstrada na Figura 8.

Figura 8. Coletando *Channel Information State*



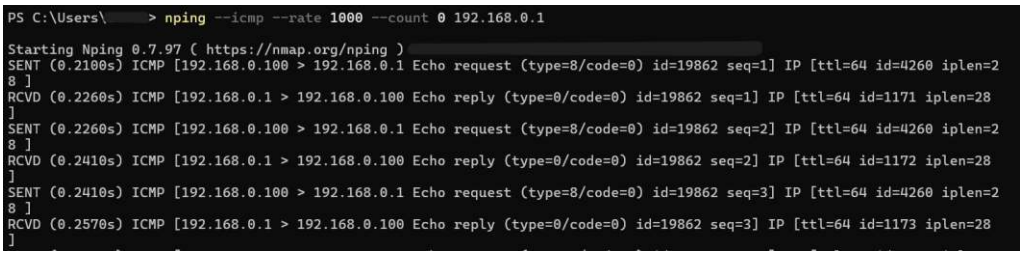
Fonte: Autoria Própria

Este artigo é publicado em acesso aberto (Open Access) sob a licença Creative Commons Atribuição 4.0 Internacional (CC-BY), que permite uso, distribuição e reprodução irrestritos em qualquer meio, desde que o autor original e a fonte sejam creditados.

3.4. *Laptop* Dell Inspirion - Transmissor

O *laptop* Dell Inspirion foi configurado para enviar 1000 pacotes ICMP por segundo ao roteador, garantindo um fluxo contínuo de dados e robustez nos conjuntos de dados. Permite que o Raspberry Pi consiga registrar informações sem lacunas e inconsistências. Utilizando o *nping* foram emitidos 1000 pacotes por segundo do tipo *ping* (protocolo do tipo ICMP) para o IP do roteador que é 192.168.0.1. A Figura 9 demonstra como foi o envio de pacotes ao roteador.

Figura 9. Envio de pacotes no *Laptop*



Fonte: Autoria Própria

3.5. Roteador Wi-Fi - Configuração de Frequência

A banda de frequência de 5 GHz foi selecionada para a configuração do roteador, com o objetivo de mitigar interferências, uma vez que a banda oferece mais canais não sobrepostos e menor congestionamento. No entanto, o seu alcance mais curto impõe uma limitação na área de detecção. A largura de banda foi definida em 80 MHz e o canal em 157 para estabilizar a comunicação. Essa combinação possibilitou o uso de 256 subportadoras, das quais algumas são pilotos, usadas para correção de erros e sincronização e outras nulas que não transportam dados e são utilizadas para evitar interferências entre as outras subportadoras, resultando assim em 234 subportadoras úteis. Proporciona-se dados CSI mais granulares e detalhados, essenciais para a precisão da detecção, conforme nos resultados.

3.6. Ambiente Experimental

Os experimentos foram conduzidos em um ambiente controlado para minimizar interferências eletromagnéticas externas e garantir a repetibilidade dos testes. O local consistia em uma sala fechada com dimensões de 2,68 m de largura, 2,86 m de comprimento e 2,91 m de altura. A escolha de um espaço de dimensões reduzidas permitiu um controle mais rigoroso sobre a propagação dos sinais Wi-Fi, um fator crucial para a análise do CSI.

A disposição dos equipamentos no interior da sala foi estrategicamente definida para otimizar a aquisição dos dados. O roteador (transmissor) e um *notebook* (receptor) foram posicionados a uma distância de 1,60 m entre si. O Raspberry Pi 4 Modelo B, atuando como dispositivo coletor, foi instalado em uma posição perpendicular à linha de visada direta (Line-of-Sight

Este artigo é publicado em acesso aberto (Open Access) sob a licença Creative Commons Atribuição 4.0 Internacional (CC-BY), que permite uso, distribuição e reprodução irrestritos em qualquer meio, desde que o autor original e a fonte sejam creditados.

- LoS) entre o transmissor e o receptor, a uma distância de 1,20 m. Esta configuração triangular foi projetada para maximizar a sensibilidade do sistema às perturbações no sinal causadas por movimentos.

Para garantir a integridade das coletas que exigiam a ausência de pessoas, como a classe "Vazio", o Raspberry Pi foi operado remotamente por um segundo *notebook* localizado fora do ambiente experimental. Todas as coletas de atividades foram realizadas com a presença de apenas um indivíduo por vez, a fim de isolar os padrões de movimento.

3.7. Composição do Conjunto de Dados

O conjunto de dados utilizado neste estudo foi estruturado em quatro classes de atividades: Caindo, Movimentação, Parado e Vazio. A classe Vazio corresponde a medições do ambiente sem pessoas. A classe Parado contém amostras de indivíduos estáticos. A categoria Movimentação agrega dados de atividades como andar e correr, enquanto a classe Caindo consiste em simulações de quedas.

Para a composição do dataset, foram coletados 800 arquivos no total, sendo 200 por classe. A fim de promover a variabilidade e a capacidade de generalização do modelo, os dados foram obtidos de quatro participantes diferentes, no qual cada um gerou 50 amostras para cada uma das quatro classes. A duração de cada coleta variou, resultando em arquivos com 300 a 1.000 pacotes de dados.

3.8. Pré-Processamento dos Dados

Os dados obtidos de CSI apresentam ruídos e variações naturais que podem comprometer o desempenho dos modelos de *machine learning*. Desta forma, é essencial realizar um processamento com o objetivo de tornar os sinais coletados em sinais robustos e representativos.

3.9. Extração dos dados brutos

Inicialmente, o pré-processamento consistiu na extração dos dados brutos de CSI a partir dos arquivos .pcap. Para esta tarefa, foi empregada a biblioteca *nexcsi* em Python, que interpreta o formato dos pacotes e fornece as informações de estado de canal como uma série temporal de números complexos para cada subportadora.

O sistema, operando em 80 MHz de largura de banda, gera dados para 256 subportadoras. Destas, 14 são subportadoras nulas, que atuam como bandas de guarda para evitar interferência, e 8 são subportadoras piloto, utilizadas para sincronização e correção de erros. Desta forma, restaram 234 subportadoras úteis, das quais as informações foram efetivamente aproveitadas.

Para cada série temporal de CSI, os valores complexos foram decompostos em seus componentes de magnitude e fase. A amplitude, obtida pela função *numpy.abs*, quantifica a

Este artigo é publicado em acesso aberto (Open Access) sob a licença Creative Commons Atribuição 4.0 Internacional (CC-BY), que permite uso, distribuição e reprodução irrestritos em qualquer meio, desde que o autor original e a fonte sejam creditados.

atenuação do sinal no canal. A fase, extraída com *numpy.angle*, mede a rotação de fase imposta pelo percurso do sinal, porém esta retorna um intervalo de 180 graus, gerando saltos que não condizem com a realidade. Para corrigir e representar a fase de forma real é preciso aplicar um processo de *unwrapping* utilizando a função *np.unwrap*. A Figura 10 exibe o trecho de código em Python responsável por automatizar este processo, mostrando o laço de repetição *(loop)* que percorre as pastas de cada atividade e extrair as métricas de amplitude e fase.

Figura 10. Extração dos dados brutos CSI

```
for atividade in atividades:
    folder = os.path.join(base_path, atividade)
    if not os.path.exists(folder):
        continue

    for file in os.listdir(folder):
        file_path = os.path.join(folder, file)
        try:
            samples = decoder(device).read_pcap(file_path)
            csi = decoder(device).unpack(samples['csi'], zero_nulls=True, zero_pilots=True)

            amp = np.abs(csi)
            pha_deg = np.angle(csi, deg=True)

            pha_unwrapped = np.unwrap(np.deg2rad(pha_deg), axis=0)
```

Fonte: Autoria Própria

3.10. Filtragem

Os dados podem ter ruídos de diversas fontes, sendo inevitáveis, prejudicando a performance dos modelos. Para isso, foram aplicados filtros tanto na fase quanto na amplitude dos sinais extraídos. Um dos filtros foi o de Hampel para retirar picos anômalos no sinal, este filtro usa uma janela sobre o sinal e substitui pontos que desviam da mediana local pela própria mediana, preservando assim o sinal e removendo picos pontuais. Também foi utilizado o filtro passa-baixa (Butterworth) de terceira ordem, para suavizar o sinal e atenuar componentes indesejados, assim reduz a instabilidade sem comprometer com informações essenciais. Conforme Figura 11.

Figura 11. Filtro de Hampel e de passa-baixa

```
    def hampel_filter(data, window_size=7, n_sigmas=4):
        median_filtered = medfilt(data, kernel_size=window_size)
        diff = np.abs(data - median_filtered)
        sigma = 1.4826 * np.median(diff)
        mask = diff > n_sigmas * sigma
        data[mask] = median_filtered[mask]
        return data

    def lowpass_filter(data, cutoff_freq=12.0, fs=50.0, order=3):
        nyquist = 0.5 * fs
        normal_cutoff = cutoff_freq / nyquist
        b, a = butter(order, normal_cutoff, btype='low', analog=False)
        y = filtfilt(b, a, data)
        return y
[2]  ✓ 0.0s
```

Fonte: Autoria Própria

ISSN: 2675-6218 - RECIMA21
Este artigo é publicado em acesso aberto (Open Access) sob a licença Creative Commons Atribuição 4.0 Internacional (CC-BY), que permite uso, distribuição e reprodução irrestritos em qualquer meio, desde que o autor original e a fonte sejam creditados.

3.11. Segmentação Temporal

Para transformar as séries temporais contínuas de CSI em amostras discretas, adequadas para o treinamento de modelos de *machine learning*, foi empregada a técnica de segmentação por janelas deslizantes (*sliding windows*). Esta abordagem foi aplicada às classes de atividades contínuas: "Movimentação" e "Parado".

Foi definida uma janela com tamanho de 150 amostras. Este intervalo foi determinado como suficiente para capturar a assinatura característica de um ciclo de movimento, como uma passada. As janelas foram deslizadas com um passo de 100 amostras, resultando em uma sobreposição de 50 amostras entre janelas consecutivas. Essa sobreposição é fundamental para garantir que eventos transitórios que ocorrem nas bordas das janelas não sejam perdidos, além de aumentar artificialmente o volume de dados de treinamento.

A classe "Caindo" e “Vazio”, por sua vez, recebeu um tratamento distinto. Por “Caindo” tratar de um evento impulsivo, de curta duração, enquanto “vazio” como um estado estático para referência, as amostras destas classes não foram segmentadas. Em vez disso, cada arquivo de queda foi tratado como uma única amostra, utilizando-se a série temporal completa para a extração de características. Esta abordagem garante a captura integral da assinatura do evento, evitando que seu padrão seja fragmentado pelas janelas. A Figura 12 detalha a lógica de implementação para a segmentação condicional das atividades.

ISSN: 2675-6218 - RECIMA21
Este artigo é publicado em acesso aberto (Open Access) sob a licença Creative Commons Atribuição 4.0 Internacional (CC-BY), que permite uso, distribuição e reprodução irrestritos em qualquer meio, desde que o autor original e a fonte sejam creditados.

Figura 12. Janelas deslizantes

```python
def getData(apply_filter=True):
    window_size=150
    step=100
    features, labels = [], []

    for atividade in atividades:
        folder = os.path.join(base_path, atividade)
        if not os.path.exists(folder):
            continue

        for file in os.listdir(folder):
            file_path = os.path.join(folder, file)

            samples = decoder(device).read_pcap(file_path)
            csi = decoder(device).unpack(samples['csi'], zero_nulls=True, zero_pilots=True)

            amp = np.abs(csi)
            pha_deg = np.angle(csi, deg=True)

            pha_unwrapped = np.unwrap(np.deg2rad(pha_deg), axis=0)

            if apply_filter:
                for i in range(amp.shape[1]):
                    amp[:, i] = lowpass_filter(hampel_filter(amp[:, i]))

                for i in range(pha_unwrapped.shape[1]):
                    pha_unwrapped[:, i] = lowpass_filter(hampel_filter(pha_unwrapped[:, i]))

            if atividade == "caindo" or atividade == "vazio":
                f = extract_features(amp, pha_unwrapped)
                features.append(f)
                labels.append(atividade)
            else:
                for start in range(0, amp.shape[0] - window_size + 1, step):
                    end = start + window_size
                    win_amp = amp[start:end, :]
                    win_pha = pha_unwrapped[start:end, :]
                    f = extract_features(win_amp, win_pha)
                    features.append(f)
                    labels.append(atividade)

    return features, labels
[5]  ✓  0.0s
```

Fonte: Autoria Própria

4. ANÁLISE E DISCUSSÃO

4.1. Análise de Componentes Principais (PCA)

Um dos maiores desafios no processamento do CSI é a elevada dimensionalidade dos dados e a presença de redundâncias, já que as 234 subportadoras úteis apresentam forte correlação entre si. Para mitigar esse problema, foi empregada a Análise de Componentes Principais (PCA), uma técnica de redução de dimensionalidade que transforma o conjunto original de variáveis correlacionadas em um novo conjunto de variáveis linearmente independentes.

A técnica de PCA foi aplicada para mitigar a redundância entre as subportadoras e reduzir o volume dos dados que não impactam a identificação de movimentações. A abordagem consistiu em utilizar o primeiro componente principal que contém informações gerais, mas também combinado com componentes secundários, que juntos capturam 95% da variância total do sinal, garantindo assim a preservação das informações essenciais.

Antes da aplicação do PCA, os dados foram normalizados pela subtração da média de cada subportadora. E com base em uma análise de variância aplicada, foram definidos 4 componentes para a amplitude e 15 para a fase.

Este artigo é publicado em acesso aberto (Open Access) sob a licença Creative Commons Atribuição 4.0 Internacional (CC-BY), que permite uso, distribuição e reprodução irrestritos em qualquer meio, desde que o autor original e a fonte sejam creditados.

4.2. Extração de Características

Ao final do processo foi gerado um vetor com 46 características, com componente principal e os demais componentes, resultando em uma representação com informações para gerar um ótimo treinamento aos classificadores. Conforme Quadro 1, Quadro 2, Quadro 3 e Quadro 4.

Quadro 1. Características extraídas no Componente Principal para a Amplitude

Característica	**Descrição**
Média	Valor médio do componente, indicando a linha de base do sinal.
Variância	Medida da dispersão do sinal em torno da média.
Desvio Padrão	Raiz quadrada da variância; indica a volatilidade do sinal.
Valor de Pico-a-Pico	Diferença entre o valor máximo e mínimo; indica a amplitude máxima da perturbação.
Energia do Sinal	Soma dos quadrados das amostras; representa a intensidade total do sinal.
Desvio Absoluto Mediano (MAD)	Medida robusta da variabilidade, menos sensível a *outliers*.
Assimetria *(Skewness)*	Mede a assimetria da distribuição de probabilidade do sinal.
Curtose *(Kurtosis)*	Mede o "achatamento" ou a presença de caudas pesadas na distribuição.
Entropia	Mede a complexidade ou a imprevisibilidade do sinal.
Domínio do Tempo - Cruzamentos por Zero	Número de vezes que o sinal muda de sinal (positivo/negativo).

ISSN: 2675-6218 - RECIMA21
Este artigo é publicado em acesso aberto (Open Access) sob a licença Creative Commons Atribuição 4.0 Internacional (CC-BY), que permite uso, distribuição e reprodução irrestritos em qualquer meio, desde que o autor original e a fonte sejam creditados.

REVISTA CIENTÍFICA - RECIMA21 ISSN 2675-6218

Domínio do Tempo - Autocorrelação (lag 1)	Mede a correlação do sinal com ele mesmo em um instante anterior.
Domínio do Tempo - Número de Picos	Contagem de picos proeminentes detectados no sinal.
Domínio da Frequência - Frequência Dominante	Frequência com a maior potência na banda de 0.5-4.0 Hz.
Domínio da Frequência - Potência em Banda 1	Potência média do sinal na banda de frequência de 0.5-2.0 Hz.
Domínio da Frequência - Potência em Banda 2	Potência média do sinal na banda de frequência de 2.0-4.0 Hz.

Fonte: Autoria Própria

Quadro 2. Características extraídas no Componente Principal para a Fase

Característica	**Descrição**
Média	Medida da dispersão do sinal de fase.
Variância	Medida da dispersão do sinal do PC da fase.
Desvio Padrão	Medida da dispersão do sinal de fase.
Valor Pico-a-Pico (PTP)	Diferença entre o valor máximo e mínimo do sinal de fase.
Desvio Absoluto Mediano (MAD)	Medida robusta da variabilidade do sinal de fase.
Assimetria *(Skewness)*	Mede a assimetria da distribuição de probabilidade da fase.
Curtose *(Kurtosis)*	Mede o "achatamento" da distribuição do sinal de fase.

Fonte: Autoria Própria

ISSN: 2675-6218 - RECIMA21
Este artigo é publicado em acesso aberto (Open Access) sob a licença Creative Commons Atribuição 4.0 Internacional (CC-BY), que permite uso, distribuição e reprodução irrestritos em qualquer meio, desde que o autor original e a fonte sejam creditados.

Quadro 3. Características extraídas no Componente Secundários para a Amplitude

Característica	**Descrição**
Variância	Medida da dispersão de cada componente secundário de amplitude.
Energia do Sinal	Intensidade total de cada componente secundário de amplitude.
Valor de Pico-a-Pico	Amplitude máxima da perturbação em cada componente secundário.

Fonte: Autoria Própria

Quadro 4. Características extraídas no Componente Secundários para a Fase

Característica	**Descrição**
Variância	Medida da dispersão de cada componente secundário de fase.

Fonte: Autoria Própria

4.3. Modelagem e Avaliação

Após a coleta e o processamento dos dados CSI, foi preciso treinar modelos supervisionados de *machine learning* para identificar os tipos de movimentações no ambiente, permitindo distinguir através de diferentes modelos e avaliar a precisão de cada um com base nas classes e características do CSI extraído.

4.4. Preparação dos dados para Treinamento

Para a classificação dos modelos, os conjuntos de dados foram divididos em 30% para teste e 70% para treinamento, utilizando uma amostragem estratificada para manter a proporção original.

Antes do treinamento, foi necessário passar por duas etapas de pré-processamento. Dimensionamento de Características, que normaliza cada característica dos dados em uma média de zero e um desvio padrão unitário, garantindo assim que todas as informações contribuam de forma equitativa, utilizando a biblioteca *Scikit-learn StandardScaler.*

A outra etapa foi o balanceamento de classes. Como, com o janelamento, as classes de “Movimentação” e “Parado” geraram mais informações, para equalizar o número utilizou-se o SMOTE, contido na biblioteca *imbalanced-learn*, que cria novas amostras sintéticas para as classes que possuem menos dados, em vez de duplicar os que já existem. Essa técnica é aplicada apenas

Este artigo é publicado em acesso aberto (Open Access) sob a licença Creative Commons Atribuição 4.0 Internacional (CC-BY), que permite uso, distribuição e reprodução irrestritos em qualquer meio, desde que o autor original e a fonte sejam creditados.

para o treinamento, sendo também uma prática utilizada para evitar vazamento de dados e garantir que a avaliação do modelo seja realista. Conforme a Figura 13.

Figura 13. Preparação para Treinamento

```
def train_and_evaluate(model, param_grid, features, labels, class_names, model_name="Modelo"):
    x_train, x_test, y_train, y_test = train_test_split(features, labels, test_size=0.3, stratify=labels, random_state=42)

    smote = SMOTE(random_state=42)
    x_train_res, y_train_res = smote.fit_resample(x_train, y_train)

    scaler = StandardScaler()
    x_train_scaled = scaler.fit_transform(x_train_res)
    x_test_scaled = scaler.transform(x_test)

    grid_search = GridSearchCV(model, param_grid, cv=5, n_jobs=-1, verbose=1, scoring='accuracy')
    grid_search.fit(x_train_scaled, y_train_res)

    best_model = grid_search.best_estimator_

    metrics = evaluate_model(best_model, x_test_scaled, y_test, class_names, model_name)
    return best_model, scaler, metrics
[16]  ✓ 0.0s
```

Fonte: Autoria Própria

4.5. Seleção e Otimização dos Classificadores

Foram escolhidos quatro algoritmos de *machine learning* supervisionados diferentes, como Máquina de Vetores de Suporte (SVM), *Random Forest*, *K-Nearest Neighbors* (KNN), *Decision Tree*.

Para cada modelo foi realizada uma otimização de hiperparâmetros por meio da técnica Busca em Grade com Validação Cruzada, utilizando a biblioteca *Scikit-learn* (*Grid Search* CV). Isso permite fazer um teste sistematicamente em uma grade de combinação de hiperparâmetros previamente definidos para avaliar as combinações. Logo, o processo seleciona a combinação que resulta em um desempenho melhor.

Figura 14. Modelos

```
models = {
    "SVM": {
        "model": SVC(class_weight='balanced', random_state=42, probability=True),
        "params": {
            'C': [0.1, 1, 10, 100, 1000],
            'gamma': [1, 0.1, 0.01, 0.001, 'scale', 'auto'],
            'kernel': ['rbf']
        }
    },
    "Random Forest": {
        "model": RandomForestClassifier(random_state=42, class_weight='balanced'),
        "params": {'n_estimators': [200, 500], 'max_depth': [10, 20, None], 'min_samples_split': [2, 5]}
    },
    "KNN": {
        "model": KNeighborsClassifier(),
        "params": {
            'n_neighbors': [3, 5, 7, 9, 11, 13],
            'weights': ['uniform', 'distance'],
            'metric': ['euclidean', 'manhattan', 'minkowski']
        }
    },
    "Decision Tree": {
        "model": DecisionTreeClassifier(random_state=42, class_weight='balanced'),
        "params": {
            'criterion': ['gini', 'entropy'],
            'max_depth': [10, 20, 30, None],
            'min_samples_split': [2, 5, 10],
            'min_samples_leaf': [1, 2, 4],
            'max_features': [None, 'sqrt', 'log2']
        }
    }
}
[17]  ✓ 0.0s
```

Fonte: Autoria Própria

Este artigo é publicado em acesso aberto (Open Access) sob a licença Creative Commons Atribuição 4.0 Internacional (CC-BY), que permite uso, distribuição e reprodução irrestritos em qualquer meio, desde que o autor original e a fonte sejam creditados.

4.6. Métricas de Avaliação

Para a avaliação do desempenho foram utilizadas métricas como acurácia, que representa a proporção de predições corretas em relação ao total de amostras. Precisão, que mede a proporção de predições positivas que estavam corretas. *Recall*, que mede a proporção de amostras positivas que foram corretamente identificadas, e F1-Score, que é a média entre a Precisão e o *Recall*, oferecendo uma métrica balanceada quando há um desequilíbrio entre as classes.

5. RESULTADOS E DISCUSSÃO

Nesta seção são apresentados e analisados os resultados dos quatro modelos de *Machine Learning* (SVM, *Random Forest*, KNN e *Decision Tree*) para a classificação de atividades humanas. As métricas de Acurácia, Precisão e F1-Score foram calculadas para cada modelo no conjunto de teste, que corresponde a 30% do total de amostras. O Quadro 5 demonstra os resultados obtidos.

Quadro 5. Resultados de Desempenho dos Modelos com Dados Pós-Filtragem

Modelo	Acurácia	Precisão	F1-Score
SVM	94,86%	94,91%	94,86%
Random Forest	95,46%	95,49%	95,46%
KNN	93,34%	93,40%	93,35%
Decision Tree	91,23%	91,27%	91,23%

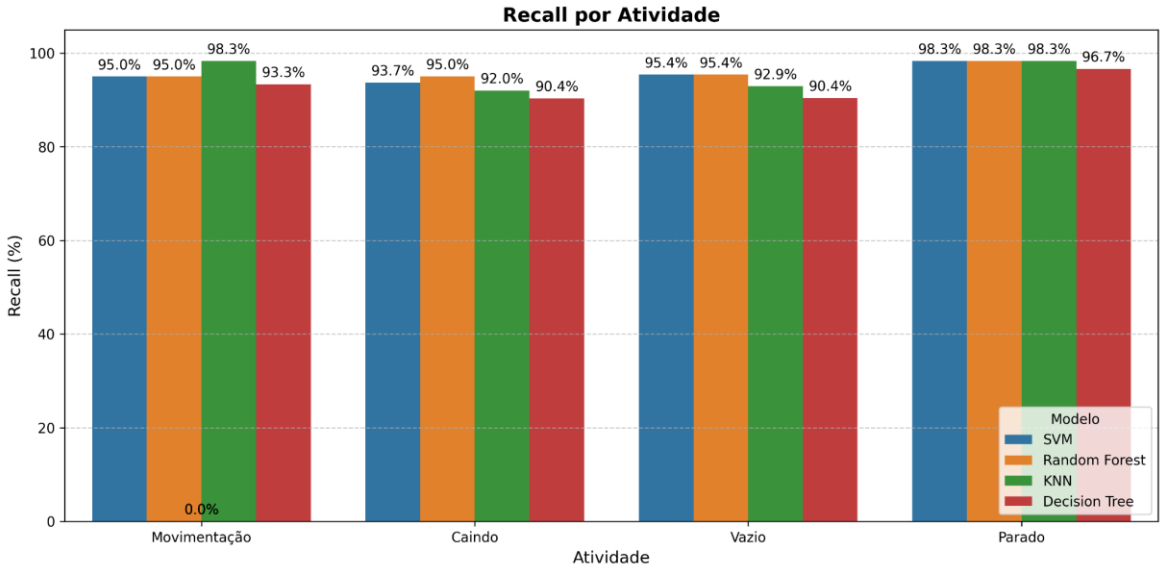
Fonte: Autoria Própria

Os resultados indicam que todos os modelos obtiveram acurácia superior a 91%, evidenciando que o conjunto de características extraído do CSI é informativo e adequado para distinguir os padrões de movimentação humana. O *Random Forest* apresentou o melhor desempenho geral, alcançando acurácia de 95,46%, confirmando sua eficácia em cenários com variáveis correlacionadas e dados reduzidos por técnicas de PCA. O KNN e o SVM também demonstraram desempenho robusto, enquanto o *Decision Tree* obteve métricas inferiores, porém ainda satisfatórias dentro do contexto do experimento.

Para complementar a análise, a Figura 15 apresenta o *Recall* por atividade, permitindo verificar a sensibilidade dos modelos em cada classe específica.

ISSN: 2675-6218 - RECIMA21
Este artigo é publicado em acesso aberto (Open Access) sob a licença Creative Commons Atribuição 4.0 Internacional (CC-BY), que permite uso, distribuição e reprodução irrestritos em qualquer meio, desde que o autor original e a fonte sejam creditados.

Figura 15. *Recall* por Atividade



Fonte: Autoria Própria

A Figura 15 evidencia que o Random Forest e o KNN apresentaram melhor estabilidade entre as quatro classes, alcançando elevados valores de *Recall*. Esse comportamento indica boa capacidade de generalização dos padrões característicos de cada tipo de movimentação. O SVM apresentou desempenho estável, mas com pequenas variações nas classes mais dinâmicas. Já o *Decision Tree* demonstrou menor sensibilidade em atividades rápidas, como “Movimentação” e “Caindo”, o que era esperado devido à sua menor capacidade de modelar transições abruptas no padrão do sinal.

A qualidade dos resultados está vinculada às 46 características extraídas nos quadros anteriores, que representam informações relevantes de amplitude e fase. A aplicação do PCA, reduzindo redundâncias entre subportadoras, foi determinante para melhorar a separabilidade das classes, favorecendo principalmente o *Random Forest* e o SVM.

6. CONSIDERAÇÕES

Este estudo apresentou o desenvolvimento e a validação de um sistema para identificação de movimentações humanas utilizando o *Channel State Information* (CSI) extraído de sinais Wi-Fi. Por meio de um pipeline estruturado envolvendo coleta, filtragem, segmentação, extração de características e classificação supervisionada, demonstrou-se que o CSI é uma alternativa tecnicamente viável e não intrusiva para monitoramento de atividades em ambientes internos.

A principal contribuição deste trabalho consiste na proposta de uma arquitetura de baixo custo, reprodutível e baseada em *hardware* amplamente acessível, capaz de capturar variações físicas do canal sem recorrer a sensores intrusivos, como câmeras ou microfones.

Este artigo é publicado em acesso aberto (Open Access) sob a licença Creative Commons Atribuição 4.0 Internacional (CC-BY), que permite uso, distribuição e reprodução irrestritos em qualquer meio, desde que o autor original e a fonte sejam creditados.

O uso de 46 características extraídas a partir das componentes principais de amplitude e fase possibilitou diferenciar quatro classes de movimentos com alta precisão. Entre os modelos testados, o *Random Forest* apresentou melhor desempenho, alcançando 95,46% de acurácia, enquanto os demais modelos também obtiveram resultados superiores a 91%.

Apesar dos resultados promissores, algumas limitações devem ser reconhecidas. As coletas foram realizadas em ambiente controlado e com apenas um indivíduo por vez, o que reduz a complexidade da propagação multivias e não representa cenários com múltiplas pessoas em movimento simultâneo. Além disso, a disposição fixa dos dispositivos e o uso de um único modelo de roteador podem influenciar o comportamento do canal, limitando a generalização dos resultados para ambientes mais dinâmicos.

Com base nas limitações observadas, recomenda-se que futuros trabalhos explorem ambientes maiores e não controlados, investiguem a identificação de múltiplos indivíduos simultâneos, ampliem o conjunto de classes de movimentação e avaliem a aplicação de técnicas de aprendizagem profunda. Outra direção promissora envolve a otimização da execução em tempo real diretamente no Raspberry Pi, bem como a comparação detalhada entre custos e consumo energético de abordagens baseadas em CSI e sensores tradicionais, como PIR e câmeras.

Os resultados deste estudo demonstram que o monitoramento baseado em CSI tem potencial significativo para aplicações domésticas, clínicas e de segurança, especialmente em cenários onde a privacidade é essencial. A validação apresentada reforça a viabilidade dessa abordagem e abre caminho para sistemas mais completos e robustos que utilizem informações físicas do Wi-Fi como ferramenta de sensoriamento inteligente.

REFERÊNCIAS

DAMODARAN, Neena et al. Device free human activity and fall recognition using WiFi channel state information (CSI). **CCF Transactions on Pervasive Computing and Interaction**, v. 2, p. 1–17, 2020. Disponível em: https://link.springer.com/article/10.1007/s42486-020-00027-1. Acesso em: 23 fev. 2025.

GRINGOLI, Francesco et al. Free your CSI: A channel state information extraction platform for modern Wi-Fi chipsets. *In:* **Proceedings of the 13th International Workshop on Wireless Network Testbeds, Experimental Evaluation & Characterization**. 2019. p. 21-28. Disponível em: https://dl.acm.org/doi/10.1145/3349623.3355477. Acesso em: 20 mar. 2025.

IBM. **O que é aprendizado supervisionado?** [S. l.]: IBM, s. d. Disponível em: https://www.ibm.com/br-pt/think/topics/supervised-learning. Acesso em: 8 set. 2025.

IBM. **O que é inteligência artificial (IA)?** [S. l.]: IBM, s. d.(b). Disponível em: https://www.ibm.com/br-pt/think/topics/artificial-intelligence. Acesso em: 27 ago. 2025.

IMBALANCED-LEARN DEVELOPERS. **Imbalanced-learn documentation**, *[S. l.],* 14 ago. 2025. Disponível em: https://imbalanced-learn.org/stable/. Acesso em: 28 set. 2025.

ISSN: 2675-6218 - RECIMA21

Este artigo é publicado em acesso aberto (Open Access) sob a licença Creative Commons Atribuição 4.0 Internacional (CC-BY), que permite uso, distribuição e reprodução irrestritos em qualquer meio, desde que o autor original e a fonte sejam creditados.

REVISTA CIENTÍFICA - RECIMA21 ISSN 2675-6218

KUROSE, James F.; ROSS, Keith W. **Redes de computadores e a internet**: uma abordagem top-down. 8. ed. Porto Alegre: Pearson, 2021.

MA, Yongsen; ZHOU, Gang; WANG, Shuangquan. WiFi sensing with channel state information: A survey. **ACM Computing Surveys (CSUR)**, v. 52, n. 3, p. 1–36, 2019. Disponível em: https://dl.acm.org/doi/abs/10.1145/3310194. Acesso em: 25 maio 2025.

SCHULZ, Matthias; WEGEMER, Daniel; HOLLICK, Matthias. **Nexmon**: The C-based Firmware Patching Framework. Disponível em: https://nexmon.org. Acesso em: 20 mar. 2025.

TANENBAUM, Andrew S.; FEAMSTER, Nick; WETHERALL, David. **Redes de computadores**. 6. ed. Porto Alegre: AMGH, 2021.

THE NUMPY DEVELOPERS. **NumPy v2.0 Manual**. [S. l.]: The Numpy Developers, 2024. Disponível em: https://numpy.org/doc/stable/. Acesso em: 8 set. 2025.

THE PANDAS DEVELOPMENT TEAM. **pandas 2.2.2 documentation**. [S. l.]: The Pandas Development Team, 2024. Disponível em: https://pandas.pydata.org/docs/. Acesso em: 8 set. 2025.

THE SCIKIT-LEARN DEVELOPERS. **scikit-learn 1.5 documentation**. [S. l.]: The Scikit-Learn Developers, 2024. Disponível em: https://scikit-learn.org/stable/. Acesso em: 8 set. 2025.

TRINDADE, Eduardo Fabrício Gomes; DE ALMEIDA, Felipe Silveira; PEREIRA JUNIOR, Lourenço Alves Pereira. Além do Sinal: Autenticação Biofísica com Wi-Fi CSI e Raspberry Pi. *In:* **Simpósio Brasileiro de Segurança da Informação e de Sistemas Computacionais (SBSeg)**. SBC, 2024. p. 32-47. Disponível em: https://sol.sbc.org.br/index.php/sbseg/article/download/30015/30014/. Acesso em: 05 fev. 2025.

WANG, Yuxi; WU, Kaishun; NI, Lionel M. Wifall: Device-free fall detection by wireless networks. **IEEE Transactions on Mobile Computing**, v. 16, n. 2, p. 581–594, 2016. Disponível em: https://ieeexplore.ieee.org/abstract/document/7458186. Acesso em: 11 mar. 2025.

WANG, Zhu et al. Wi-Fi CSI-based behavior recognition: From signals and actions to activities. **IEEE Communications Magazine**, v. 56, n. 5, p. 109–115, 2018. Disponível em: https://ieeexplore.ieee.org/abstract/document/8360860. Acesso em: 5 fev. 2025.

ISSN: 2675-6218 - RECIMA21

Este artigo é publicado em acesso aberto (Open Access) sob a licença Creative Commons Atribuição 4.0 Internacional (CC-BY), que permite uso, distribuição e reprodução irrestritos em qualquer meio, desde que o autor original e a fonte sejam creditados.