



Desenvolvimento de uma técnica de detecção e identificação de sinais produzidos por sensores de RF com aplicações no cuidado de idosos.

Palavras-Chave: Identificação de movimentos com RF, Efeito *Doppler*, *Support Vector Machine*

Autor: Gabriel São Martinho da Silva, FT – Unicamp

Orientador: Prof. Dr. Leonardo Lorenzo Bravo Roger, FT - Unicamp

INTRODUÇÃO:

Nos últimos anos, um dos grandes desafios enfrentados por diversas sociedades é o envelhecimento populacional. Com o aumento da expectativa de vida surge a necessidade de garantir a segurança e o bem-estar de pessoas idosas, principalmente daquelas que vivem sozinhas. Dessa forma, o uso de tecnologias para promover o cuidado dessas pessoas, caso ocorra algum acidente, acaba sendo de suma importância para fornecer respostas rápidas para qualquer tipo de emergência.

Uma abordagem para o monitoramento de pessoas é a utilização de sensores de radiofrequência (RF), que podem trazer algumas vantagens em relação a outros métodos de monitoramento. A primeira vantagem é a redução da necessidade de uma pessoa física para tutorar constantemente os idosos, o que permite uma maior independência para essas pessoas.

Além disso, a utilização de RF possui vantagens em relação às câmeras de vídeo, já que garante uma maior privacidade, pois como as ondas de RF são utilizadas para identificar apenas determinados tipos de movimentos, isso evita com que os idosos sejam gravados em situações constrangedoras com câmeras, o que vai possibilitar uma maior sensação de privacidade e intimidade.

Outra vantagem significativa em comparação com as câmeras é a operação efetiva em ambientes escuros. Diferente das câmeras, que dependem de uma iluminação adequada para funcionar corretamente, os sensores de RF podem detectar os movimentos mesmo em ambientes com pouca iluminação ou até mesmo com a ausência de luz, o que possibilitaria um monitoramento contínuo e ininterrupto durante o período do dia como no da noite, sendo [1], um dos mais importantes trabalhos já publicados neste campo.

O presente projeto de IC se inspira nesse trabalho, mas melhora seu desempenho em vários aspectos: utilização de um *hardware* mais simples e barato, proposta de um *software* com menor exigências de processamento e o mais importante, o sistema de sensoriamento aqui proposto consegue realizar o processamento em tempo real.

METODOLOGIA:

Para a realização desse projeto foram utilizados os seguintes componentes: um radar de efeito *Doppler* (mais especificamente o HB100), um módulo amplificador (LM386), um Arduino UNO e um computador. O HB100 [2] é um radar Doppler que não possui um custo elevado, foi adquirido por apenas R\$35,00. Ele é utilizado para detectar movimentos. Esse radar se baseia no efeito *Doppler*, que é um fenômeno que explica a mudança na frequência de um sinal para um alvo que está se movimentando.

Basicamente, esse radar emite uma frequência equivalente a 10.525 GHz (banda X), caso exista um objeto que esteja em sua zona de detecção, esse sinal vai ser refletido e recebido novamente pelo módulo HB100. Devido ao efeito *Doppler*, se algum objeto estiver se movimentando, a frequência recebida será diferente da frequência originalmente emitida pelo radar. Caso o objeto esteja se movimentando em direção ao radar, essa frequência aumenta; caso esteja se afastando, a frequência diminui. Por fim, o módulo HB100 fornece a variação das frequências na forma de um sinal analógico proporcional a essa variação. Esse sinal é amplificado pelo módulo LM386 [3] em até 200 vezes (13 dB).

O sinal amplificado é conectado a uma placa Arduino UNO [4] que tem como objetivo realizar a Conversão Analógica para Digital, o sinal é convertido no formato digital de 10 bits, mostrando que o Arduino UNO possui um total de 1024 possibilidades para representar tensões entre 0V até 5V.

Depois que o Arduino realiza essa conversão, os valores são enviados para um computador com o objetivo de ler esses sinais transmitidos. Posteriormente, uma inteligência artificial (IA) vai analisar as formas de onda (tensão x tempo) e determinar qual movimento está sendo realizado [5].



Figura 1: Representação de todo o circuito montado para ler e processar os sinais transmitido pelo módulo HB100

Depois da montagem do sistema, foi criado um código em *Python* para processar esses sinais e representá-los graficamente. Além disso, ao pressionar a tecla “espaço”, o código salva um arquivo do tipo CSV com os últimos 4000 valores que o Arduino transmitiu para o computador e, também, uma imagem do tipo “.png” para mostrar graficamente esse arquivo CSV.

Com esse *software* desenvolvido, foi possível passar para a segunda parte do projeto que seria a criação de um banco de dados com diversos arquivos CSV para representar um total de 4 movimentos,

sendo eles: “movimentação da mão para frente e para trás”, “movimentação da mão para cima e para baixo”, “andar para frente” e “estar parado”. A tabela 1 mostra graficamente essas formas de ondas:

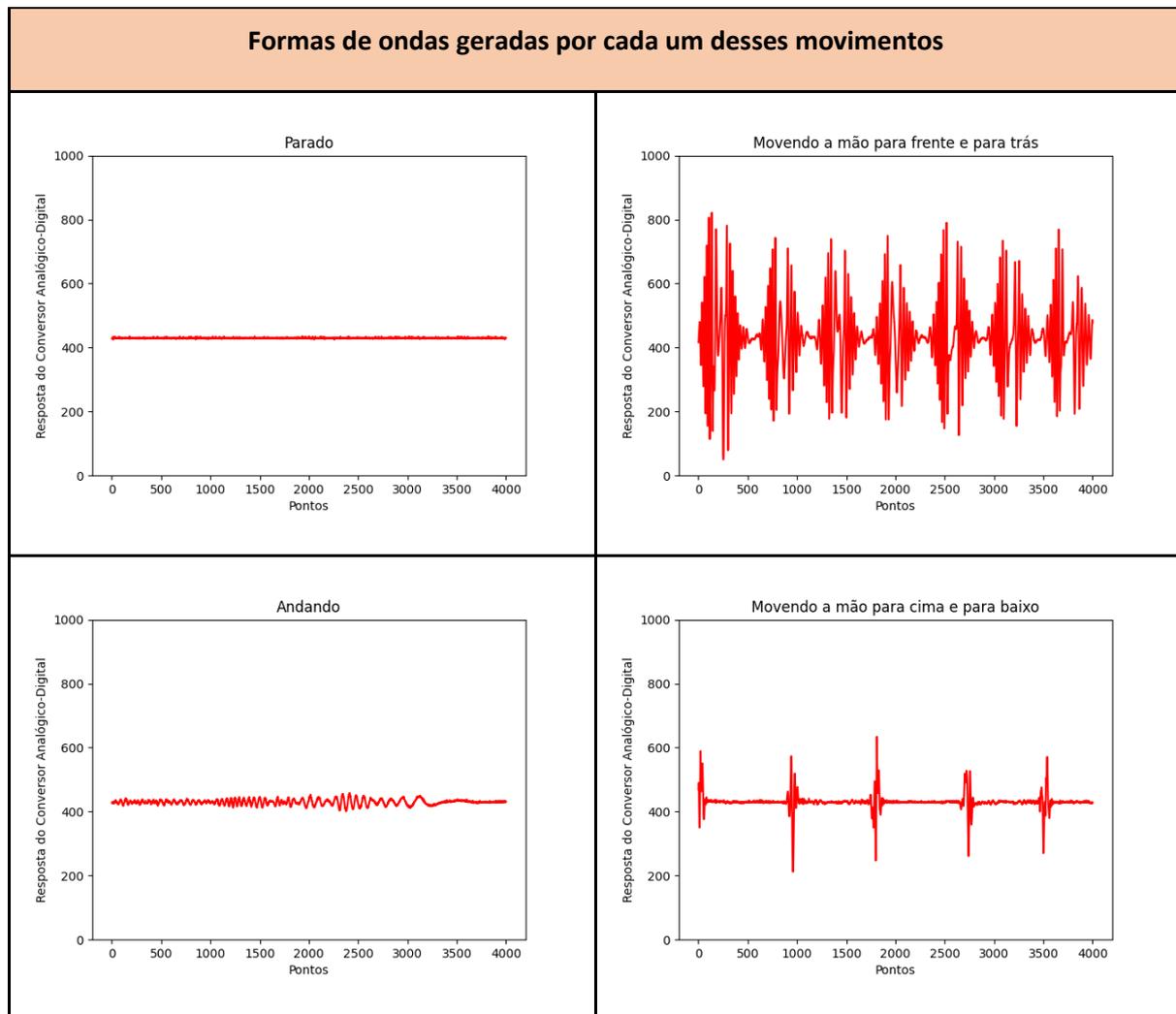


Tabela 1: Representação dos 4 tipos de movimentos realizados e suas respectivas formas de ondas

Para a criação desse banco de dados foram salvos 1250 arquivos CSV, com 4000 pontos para cada um desses 4 movimentos estudados.

Depois que o banco de dados foi concluído, deu-se início a terceira e última parte do projeto que consiste na criação de um algoritmo com inteligência artificial para classificar cada um desses 4 tipos de movimentos. A inteligência artificial escolhida para realizar essa classificação foi o *support vector machine (SVM) com kernel rbf* (esse é um modelo de aprendizado de máquina usado para classificação e regressão, basicamente ele funciona encontrando a linha ou plano que melhor separa os dados fornecidos em diferentes classes) da biblioteca *scikit-learn do python* [6].

Dessa forma, quando o modelo de inteligência artificial fosse concluído, ele seria passado para um outro programa em *python*, que tem a capacidade de analisar em tempo real quais movimentos estão sendo realizados e exibir em uma tela.

RESULTADOS E DISCUSSÃO:

Ao longo do desenvolvimento do processo os resultados foram os esperados, sendo possível analisar esses 4 movimentos em tempo real devido à alta precisão dos modelos gerados. Diversos modelos preditivos foram criados para determinar a maior precisão possível dos movimentos.

De todas as 1250 amostras de movimentos que estavam disponíveis, 250 foram separadas para realizar testes dos modelos treinados. Assim, sempre que um modelo era finalizado, ele respondia um teste com 1000 perguntas (250 amostras separadas de cada um desses 4 movimentos, totalizando essas 1000 perguntas). Dessa forma, foi possível verificar a acurácia do modelo e perceber em quais pontos ele estava com mais dificuldade para identificar esses movimentos.

Foram treinados modelos com 200, 400, 600, 800 e 1000 amostras do banco de dados disponível e para cada uma dessas quantidades, o parâmetro C (que é um valor que determina o equilíbrio entre, permitir um modelo com uma margem de decisão mais ampla, que pode generalizar melhor para novos dados, mas aceita mais erros de classificação e um modelo mais complexo que se ajusta bem aos dados de treinamento, mas pode ter dificuldade em generalizar novos dados (o que pode levar ao *overfitting*)), foi mudado de 1 até 100 com um acréscimo de 1. Com isso, foi criado um gráfico, Fig. 2 e identificado o melhor modelo para a classificação dos movimentos.

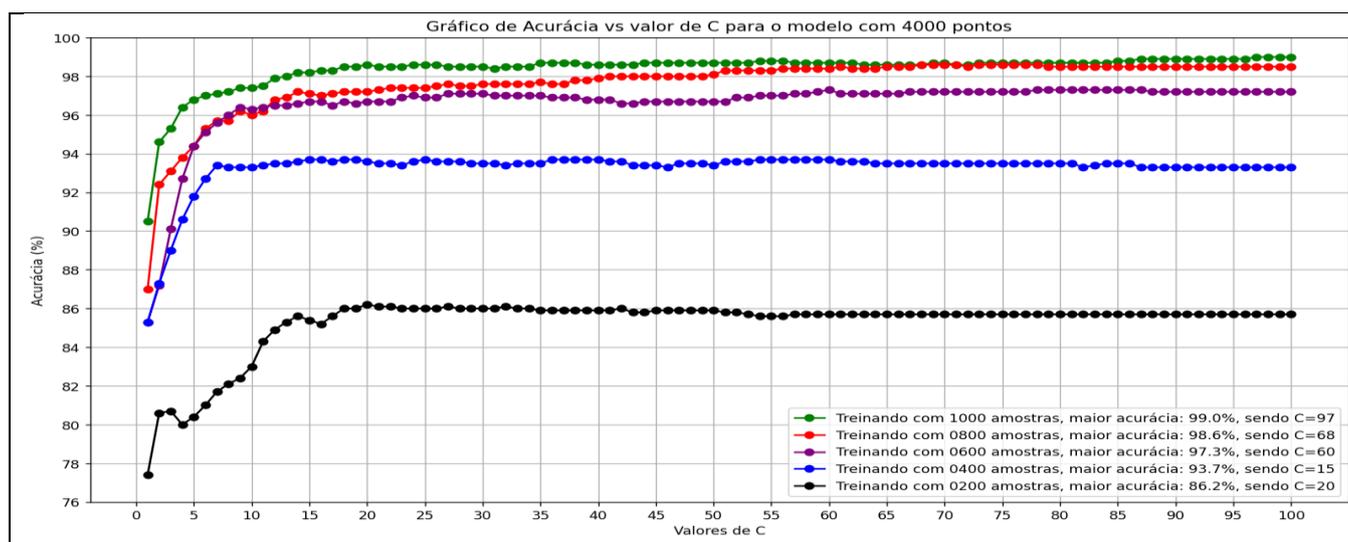


Fig. 2. Representação do valor da acurácia do modelo conforme é mudado o parâmetro C

Nesse gráfico da Fig. 2 é possível ver que ocorre uma melhora na acurácia conforme mais amostras são utilizadas para treinar esse modelo. Outra situação identificada foi o excelente desempenho do melhor modelo encontrado, observando-se um total de 99% de acurácia. Isso significa que, das 1000 perguntas que o modelo deveria classificar qual tipo de movimento estava acontecendo, ele acertou um total de 990 perguntas, mostrando uma alta precisão na classificação dos movimentos que estavam ocorrendo.

Observe na Fig. 3 a acurácia do melhor modelo treinado e quais confusões ocorreram.

```
( Movendo a mão para cima e para baixo ): 96.80% de acertos e 3.20% de erros
→ Confundi com Parado: 0 vezes
→ Confundi com Andando: 8 vezes
→ Confundi com Movendo a mão para frente e para trás: 0 vezes
→ Quantidade de acertos: 242/250
→ Total de suposições: 244

( Parado ): 100.00% de acertos e 0.00% de erros
→ Confundi com Movendo a mão para cima e para baixo: 0 vezes
→ Confundi com Andando: 0 vezes
→ Confundi com Movendo a mão para frente e para trás: 0 vezes
→ Quantidade de acertos: 250/250
→ Total de suposições: 250

( Andando ): 99.20% de acertos e 0.80% de erros
→ Confundi com Movendo a mão para cima e para baixo: 2 vezes
→ Confundi com Parado: 0 vezes
→ Confundi com Movendo a mão para frente e para trás: 0 vezes
→ Quantidade de acertos: 248/250
→ Total de suposições: 256

( Movendo a mão para frente e para trás ): 100.00% de acertos e 0.00% de erros
→ Confundi com Movendo a mão para cima e para baixo: 0 vezes
→ Confundi com Parado: 0 vezes
→ Confundi com Andando: 0 vezes
→ Quantidade de acertos: 250/250
→ Total de suposições: 250

Acurácia do modelo: 99.00%
```

Fig. 3. Imagem que representa a acurácia do melhor modelo treinado e quais confusões estão acontecendo

CONCLUSÕES:

O sistema de sensoriamento por RF desenvolvido neste projeto de IC está funcionando com uma alta precisão e realiza as previsões em tempo real. Essa é uma contribuição importante, se comparado com trabalhos prévios já publicados na literatura, os quais realizam o processamento de forma *offline*. A simplicidade do *hardware* utilizado, seu baixo custo, assim como o modelo SVM, que foi escolhido por seu menor consumo computacional em comparação a outras soluções existentes no estado da arte, como por exemplo redes convulsionais [1] para realizar a identificação dos movimentos dos idosos, tornam a solução apresentada aqui como uma forte candidata a virar um produto com possibilidades de comercialização. Essa ideia cobra maior força se leva-se em conta que o sistema de sensoriamento proposto respeita a privacidade dos idosos monitorados. No entanto, o trabalho pode ser aprimorado, serão estudadas algumas modificações no projeto como por exemplo, a otimização do *software*, para possibilitar que o programa consiga funcionar ainda com menos recursos computacionais.

BIBLIOGRAFIA

- [1] CHUMA, Euclides L. "Human Movement Recognition System Using CW Doppler Radar Sensor with FFT and Convolutional Neural Network", IEEE MTT-S Latin America Microwave Conference, LAMC, Cali, Colombia, 2020.
- [2] HB100 Microwave Sensor Module Datasheet, Disponível em: https://www.mantech.co.za/Datasheets/Products/HB100_RADAR.pdf Acesso em: 29 jul. 2023.
- [3] Texas Instruments. LM386 Low Voltage Audio Power Amplifier. Disponível em: https://www.ti.com/lit/ds/symlink/lm386.pdf?ts=1690700087289&ref_url=https%253A%252F%252Fwww.google.com%252F. Acesso em: 30 jul. 2023
- [4] MARGOLIS, Michael. Arduino: Guia para colocar suas ideias em prática. São Paulo, Novatec Editora, 2013.
- [5] GÉRON, Aurélien. Aprendizado de Máquina com Scikit-Learn, Keras, TensorFlow: Conceitos, ferramentas e técnicas para a construção de sistemas inteligentes. São Paulo, Alta Books, 2020.
- [6] MCKINNEY, Wes. Python para Análise de Dados. São Paulo, Novatec Editora, 2018.