

Rafael de Moura Moreira

Classificação de Batidas do Coração Usando Algoritmo de Baixo Custo

Itajubá – MG

Julho de 2019

Rafael de Moura Moreira

Classificação de Batidas do Coração Usando Algoritmo de Baixo Custo

Dissertação submetida ao Programa de Pós-Graduação em Ciência e Tecnologia da Computação como parte dos requisitos para obtenção do Título de Mestre em Ciência e Tecnologia da Computação.

UNIVERSIDADE FEDERAL DE ITAJUBÁ
PROGRAMA DE PÓS-GRADUAÇÃO EM CIÊNCIA E TECNOLOGIA DA
COMPUTAÇÃO

Orientador: Prof. Dr. Robson Luiz Moreno

Itajubá – MG

Julho de 2019

Agradecimentos

Ao meu orientador, Prof. Robson Moreno, por sua orientação, seus conselhos e pela coragem e sabedoria para tomar decisões difíceis.

Aos meus pais, Márcia e Paulo, por toda uma vida de apoio e de investimento em minha educação.

À minha noiva Brenda, parceira para todas as horas, pela paciência e pelo suporte emocional nos momentos mais difíceis.

Ao Prof. Rodrigo “Max” e aos colegas Adriano e Henrique, amigos e meus primeiros parceiros na carreira científica.

Ao Prof. Lenarth, meu primeiro chefe, pelo excelente exemplo profissional, aconselhamento de carreira e por todo o apoio para que seguisse com meus estudos durante minha passagem pelo IESTI.

Aos professores Tales, Evaldo e João Paulo por suas valiosas contribuições para o trabalho.

A todos os meus professores, cada qual responsável à sua maneira por um pedacinho da minha formação.

A todos os meus amigos, pelos momentos de apoio, pela compreensão com a minha indisponibilidade frequente e, principalmente, pelos momentos de descontração e alegria.

Ao meu padrinho Fernando “Dedão” (*in memoriam*), responsável pelas minhas primeiras linhas de código e primeiro contato com um voltímetro – e possivelmente meu primeiro professor de engenharia de computação, anos antes do início dos meus estudos na universidade.

À CAPES, pela bolsa de estudos.

*“Our future is a race between the growing power of
our technology and the wisdom with which we use it.*

Let’s make sure that wisdom wins.”

(Stephen Hawking)

Resumo

O eletrocardiograma é uma poderosa ferramenta na detecção de patologias. Diversas doenças podem se manifestar através de alterações na forma do sinal elétrico produzido pelo coração. O desenvolvimento de dispositivos portáteis e de fácil uso capazes de analisar o sinal automaticamente permitiria o acompanhamento da evolução de patologias em casa, além de mutirões de exames em comunidades carentes, com pouco acesso a médicos e hospitais bem equipados. Neste trabalho é proposto um sistema completo combinando diferentes algoritmos para realizar classificação em tempo real de batimentos cardíacos para detecção de doenças utilizando hardware dedicado, com recursos limitados. Algumas técnicas utilizadas, como o algoritmo de Pan Tompkins para detecção de complexos QRS, já foram exaustivamente testados e utilizados em diversos sistemas de classificação de batimentos cardíacos, enquanto outros, como a segmentação dinâmica e os parâmetros de Hjorth, foram propostos recentemente para esse tipo de problema e seu funcionamento com sinais armazenados para pós-processamento foi demonstrado, além de seu baixo consumo de recursos computacionais. O sistema proposto foi testado verificando a integração das diferentes técnicas sem nenhum tipo de conhecimento prévio sobre o sinal, mediu seu desempenho utilizando a base de dados MIT-BIH Arrhythmia e seu tempo de execução. Dentro da proposta de detectar doenças, o modelo obteve índices de acerto comparáveis a outros trabalhos influentes nesta área de pesquisa, mas ele foi capaz de cumprir os requisitos de tempo de execução apenas em um laptop convencional, não se mostrando ainda viável para classificação em tempo real em sistemas embarcados mais acessíveis.

Palavras-chave: ECG. Classificação de batimentos cardíacos. Classificação online. Parâmetros de Hjorth. Microcontrolador. Sistemas embarcados.

Abstract

The electrocardiogram is a powerful tool for heart disease diagnosis. Several conditions manifest themselves as artifacts on the heart's electric signal waveform. The development of user-friendly portable devices able to analyse an electrocardiogram signal automatically could allow patients to monitor their own conditions at home, and allow for large scale examinations on low income communities without doctors or well equipped hospitals. A full system combining different algorithms is proposed to perform online heartbeat classification using dedicated hardware with limited resources. Some of the techniques utilized, such as the Pan Tompkins QRS detection algorithm, have been extensively tested and used in different heartbeat classification systems, while others, such as dynamic segmentation and Hjorth parameters, have been previously shown to work for offline classification and use few computational resources. The proposed model tests how the different techniques integrate and work with no previous information about the signal, verifies their accuracy using the MIT-BIH Arrhythmia dataset and checks its execution time. Although it had good accuracy and was able to perform online classification on a conventional laptop, on a microcontroller it exceeded the execution time required for online classification.

Keywords: ECG. Heartbeat classification. Online classification. Hjorth parameters. Microcontroller. Embedded systems.

Lista de ilustrações

Figura 1 – Alterações em um sinal de ECG.	9
Figura 2 – Elementos de um sistema de classificação automática de batimentos. . .	11
Figura 3 – Diagrama de blocos do sistema proposto.	14
Figura 4 – Efeito da componente DC no filtro passa-faixa.	17
Figura 5 – Resultado da combinação dos filtros passa-faixa e DC.	18
Figura 6 – Ondas presentes em um ECG normal.	19
Figura 7 – Resultado da segmentação dinâmica.	20
Figura 8 – Relação entre a integral e o complexo QRS. QS é a largura do complexo QRS, e W é a largura da janela deslizante da integral.	21
Figura 9 – Tempo de aprendizado do algoritmo de Pan-Tompkins.	22
Figura 10 – Formas de onda geradas pelo algoritmo de Pan-Tompkins: A) Sinal original. b) Sinal filtrado. c) Derivada. d) Derivada ao quadrado. e) Integral por janela deslizante. f) Sinal original com o atraso provocado pelos filtros. g) Saída do algoritmo.	23
Figura 11 – Formas de onda geradas pela implementação proposta: Sinal original; Sinal filtrado; Derivada; Derivada ao quadrado; Integral por janela deslizante; Saída do algoritmo.	24
Figura 12 – Comparação de dimensões entre uma Raspberry Pi e um cartão de crédito.	28

Lista de tabelas

Tabela 1 – Reagrupamento proposto das classes da AAMI e dos tipos originais da base MIT-BIH Arrhythmia.	15
Tabela 2 – Matriz de confusão do sistema.	29
Tabela 3 – Índices de performance.	30

Lista de abreviaturas e siglas

AAMI	<i>Association for the Advanced of Medical Instrumentation</i> (Associação para o Avanço da Instrumentação Médica)
DC	<i>Direct Current</i> (Corrente contínua)
ECG	Eletrocardiograma
EEG	Eletroencefalograma
RBF	<i>Radial Basis Function</i> (Função de base radial)
SVM	<i>Support Vector Machine</i> (Máquina de vetores de suporte)

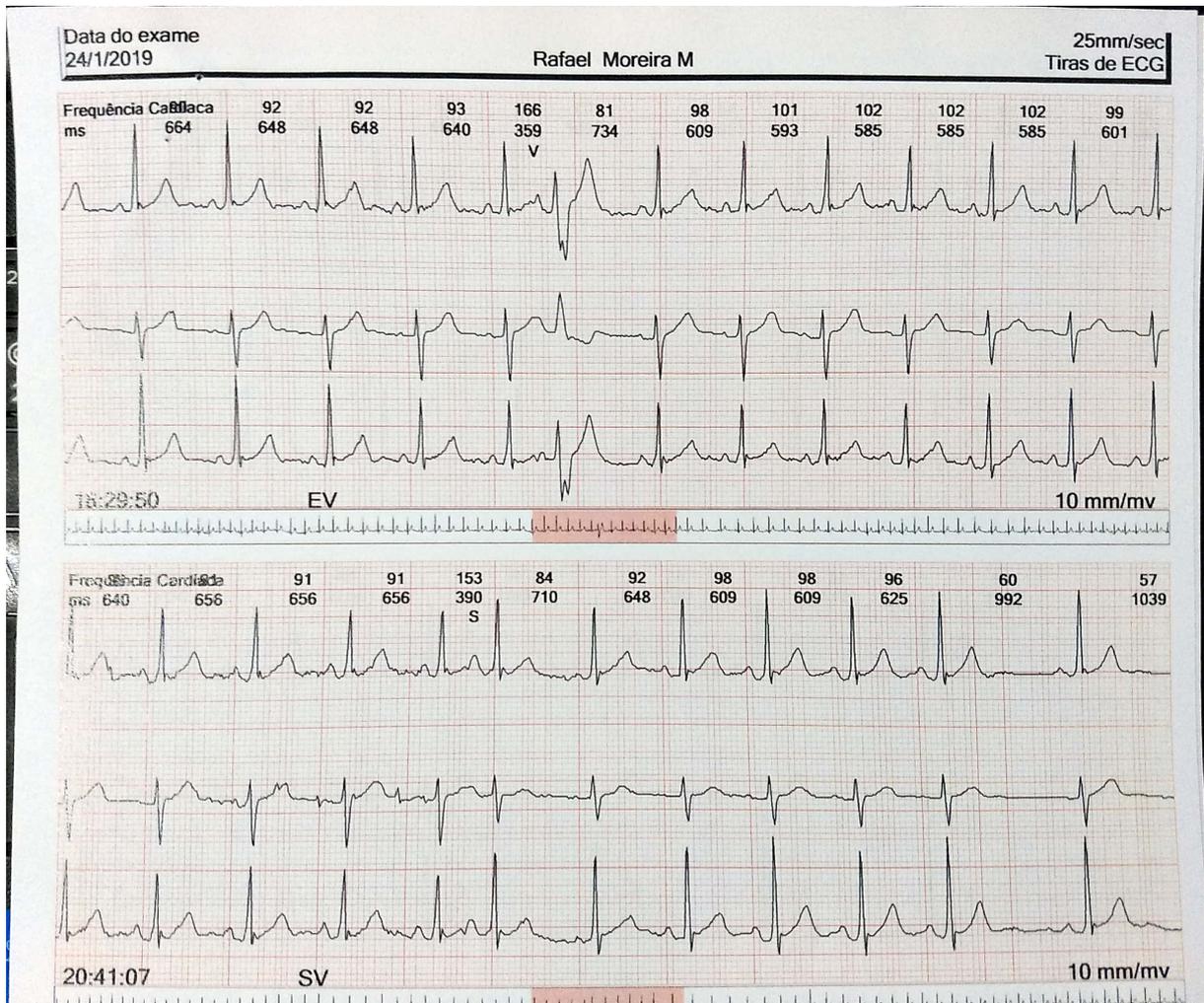
Sumário

1	INTRODUÇÃO	9
1.1	Sistemas Embarcados	10
1.2	Classificação Automática de Batimentos Cardíacos	11
1.3	Objetivo	12
1.4	Organização do Trabalho	12
2	SISTEMA PROPOSTO	13
2.1	Programação	13
2.2	Base de dados	14
2.3	Pré-Processamento	15
2.3.1	Filtro Passa-Faixa	16
2.3.2	Filtro DC	16
2.4	Segmentação	18
2.4.1	Detecção de Pontos Fiduciais	19
2.4.1.1	Período de aprendizado	22
2.4.1.2	Atrasos na detecção	22
2.5	Extração de características	22
2.5.1	Intervalos R-R	24
2.5.2	Características de amplitude	25
2.5.3	Parâmetros de Hjorth	25
2.6	Classificação (SVM)	26
2.6.1	Otimização dos parâmetros da SVM.	26
2.6.2	Parametrização dos dados de entrada	27
2.6.3	Balanceando a base de dados	27
3	RESULTADOS	28
3.1	Classificação	29
3.1.1	Métricas adotadas	29
3.1.1.1	Resultados da classificação	29
3.2	Tempo de execução	30
4	CONCLUSÃO	31
	REFERÊNCIAS	32

1 Introdução

Os exames de eletrocardiografia (ECG) são ferramentas importantes no diagnóstico de doenças do coração. Alterações na forma de onda do sinal elétrico produzido pelo órgão – conhecidas como arritmias cardíacas – podem indicar diferentes problemas. A Figura 1 ilustra a manifestação de problemas em sinais de ECG. Como muitas vezes as anomalias podem aparecer ocasionalmente, são usados exames de longa duração, como o Holter, que utiliza um dispositivo portátil para capturar os sinais ECG de um paciente durante o seu dia-a-dia, por horas, ou mesmo dias consecutivos (HOLTER, 1961).

Figura 1 – Alterações em um sinal de ECG.



Fonte: Autor

A gravação final pode conter dezenas de milhares de batimentos, tornando sua análise custosa em termos de horas de trabalho do especialista, além de aumentar a

propensão a erros devido ao cansaço do mesmo.

Para solucionar este problema, recorre-se a programas de computador para detectar automaticamente as anomalias: o sinal capturado é fornecido ao programa, e ele pode responder quantos batimentos anômalos foram encontrados, quais as classes de anomalia observadas, além de diversas informações estatísticas sobre frequência cardíaca, atrasos etc.

Vários desses programas, porém, são de *alto custo computacional*: devido à complexidade dos cálculos executados, grandes quantidades de memória e/ou ciclos de processamento são necessários, mesmo em sistemas computacionais robustos (MAZOMENOS et al., 2013).

Essas exigências – horas de análise de um médico especialista ou disponibilidade de computadores robustos – são impossíveis de atender em diversos locais carentes, onde não há recursos financeiros suficientes para a contratação de médicos ou a aquisição de equipamentos (SILVA; MOODY; CELI, 2011).

Surge, então, a necessidade de pesquisar meios de detectar anomalias em sinais cardíacos com *baixo custo computacional* - ou seja, consumindo poucos recursos de processamento e memória, viabilizando então o desenvolvimento de dispositivos dedicados e mais acessíveis. Estes poderiam ser utilizados por técnicos e enfermeiros e informem com clareza a suspeita de problemas cardíacos em pouco tempo, permitindo a sua aquisição para triagem de pacientes potencialmente doentes para encaminhamento para locais com melhor estrutura para realizar seu tratamento.

1.1 Sistemas Embarcados

Sistemas embarcados são sistemas microprocessados cujo computador é completamente dedicado ao sistema que ele controla. Diferentemente dos computadores de propósito geral, como laptops ou desktops, um sistema embarcado realiza apenas um conjunto de tarefas predefinidas, geralmente com pré-requisitos específicos. Já que este é um sistema dedicado a tarefas específicas, pode-se otimizar o projeto reduzindo tamanho, recursos computacionais e custo do produto (OLIVEIRA; ANDRADE, 2006).

O desenvolvimento de sistemas embarcados é uma área fundamental dentro da eletrônica, pois o custo de sistemas programados já atingiu o mesmo patamar de custo de sistemas eletrônicos dedicados (ROWE; ROSENBERG; NOURBAKHS, 2002). Deste modo é possível desenvolver produtos com alto grau de personalização e baixo custo. Uma das maiores vantagens trazidas pelo software embarcado é a facilidade de desenvolvimento, atualização e correção de erros, reduzindo o tempo de projeto e aumentando a qualidade e a quantidade de subprodutos gerados (BARR, 1999).

Entre os sistemas mais simples, estão os controladores embutidos em utilidades domésticas, tais como fornos micro-ondas, sistemas de entretenimento, sistemas automotivos etc. Na outra extremidade estão os sistemas militares de defesa, os sistemas de controle de plantas industriais (químicas e nucleares) e o controle de tráfego aéreo e ferroviário.

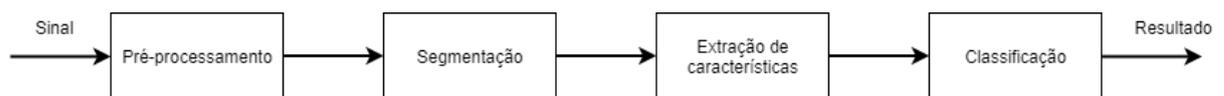
Atualmente, há uma demanda crescente por dispositivos que permitam um controle confiável e individualizado de dieta, sinais vitais e evolução de doenças crônicas (FAYN; RUBEL, 2009). Complementarmente a isso, a tecnologia tem se desenvolvido rumo a uma maior miniaturização de dispositivos, maior eficiência energética e melhor desempenho computacional, permitindo o desenvolvimento de dispositivos portáteis (como implantes ou os chamados *wearables* (dispositivos que podem ser vestidos como relógios e pulseiras, ou embutidos em roupas) para monitoramento contínuo de sinais vitais, aplicação de medicamentos, marca-passos, entre outros (SRIDHARA et al., 2011).

O desenvolvimento de dispositivos embarcados para detecção de problemas cardíacos tem potencial para facilitar o acesso a cuidados médicos em regiões carentes de recursos e permitir que pacientes possam, em casa, realizar um controle mais fino de suas condições de saúde e buscar auxílio médico quando necessário.

1.2 Classificação Automática de Batimentos Cardíacos

Um sistema típico de classificação automática de batimentos cardíacos é constituído por diferentes partes, ilustradas no diagrama da Figura 2.

Figura 2 – Elementos de um sistema de classificação automática de batimentos.



Fonte: Autor (baseado nas informações de Luz et al. (2016))

Cada uma dessas etapas possui uma função específica (LUZ et al., 2016):

- **Pré-processamento:** Nessa etapa são aplicados filtros e amplificadores, de modo a aumentar a relação sinal/ruído.
- **Segmentação:** Serve para detectar o início e o fim de cada batimento em um sinal longo. Frequentemente envolve a detecção de uma parte específica de um sinal cardíaco (como o complexo QRS) e cálculos a partir deste para obter os limites correspondentes à batida.
- **Extração de características:** Como um batimento cardíaco pode possuir de centenas a milhares de amostras, classificá-los utilizando diretamente os valores das

amostras pode ser ineficiente, lento ou mesmo inviável. Portanto é comum que seja criado um vetor de características utilizando informações estatísticas, informações sobre as frequências presentes, entre outros.

- **Classificação:** Utiliza técnicas estatísticas ou de aprendizado de máquina para reconhecer certas características como pertencentes a uma ou outra classe de sinais.

1.3 Objetivo

O objetivo deste trabalho é verificar a viabilidade de implementar uma solução computacional embarcada dedicada a analisar sinais de ECG e apontar rapidamente quando anomalias forem detectadas. Para isso, foram escolhidos algoritmos de baixo custo computacional: uma técnica de segmentação dinâmica dos batimentos cardíacos combinada com a extração dos parâmetros de Hjorth (HJORTH, 1970) dos segmentos para utilização por um algoritmo de classificação, *Support Vector Machine* (SVM), conforme proposto por Leite e Moreno (2017). O foco do trabalho está na análise do desempenho das técnicas de segmentação e extração de características. A SVM já é um algoritmo consolidado na área, amplamente testado em conjunto com outras técnicas de extração de características (LUZ et al., 2016).

Modificações ao trabalho original foram necessárias para verificar a possibilidade de uso real da solução proposta: a análise do sinal previamente armazenado foi substituída por uma análise em tempo real, coletando as amostras individualmente e salvando-as em um *buffer*. Foi utilizado um algoritmo clássico de detecção de picos de sinais cardíacos (algoritmo de Pan-Tompkins) para auxiliar na segmentação dos sinais, visto que, enquanto no trabalho original foi utilizado um arquivo com marcações indicando os picos, em um caso real o sinal é previamente desconhecido e imprevisível, e a grande maioria dos módulos foi implementado em linguagem C visando alto desempenho computacional e portabilidade.

Em seguida foi analisada a eficácia da solução utilizando métricas já consolidadas, o seu tempo de execução em um hardware portátil de baixo custo e feitas considerações sobre o compromisso entre sua eficiência comparada com algoritmos mais caros e a possibilidade de execução em sistemas portáteis como meio de triagem de pacientes.

1.4 Organização do Trabalho

O Capítulo 2 fornece revisões de literatura sobre cada técnica utilizada neste trabalho e explica sua implementação.

O Capítulo 3 explica as métricas adotadas e apresenta os resultados obtidos

Por fim, o Capítulo 4 traz conclusões e sugestões para trabalhos futuros.

2 Sistema proposto

Conforme descrito em (LUZ et al., 2016) e mencionado no capítulo introdutório, um sistema para classificação automática de batimentos possui algumas etapas comuns. Neste capítulo serão descritos os algoritmos utilizados em cada uma dessas etapas, além de particularidades específicas dessa implementação visando a integração com as outras etapas e alguns outros benefícios, como alto desempenho e portabilidade.

2.1 Programação

Exceto pela etapa de classificação, todos os algoritmos foram implementados em linguagem C, evitando ao máximo o uso de bibliotecas e funções prontas. A única exceção foi a função *sqrt()*, disponível na biblioteca *math.h*, que é padrão e fornecida com a maioria dos compiladores – porém, existe diversos algoritmos de fácil implementação que calculam uma boa aproximação para a raiz quadrada (VETTERLING et al., 1989) e poderiam ser utilizados em uma eventual versão para microcontroladores mais limitados, para evitar utilizar bibliotecas, reduzindo o tamanho do programa gerado.

A linguagem C possui diversas características que a tornam uma boa escolha para o desenvolvimento de software embarcado (SILVA, 2006). Apesar de ser uma linguagem de alto nível, permite ao programador um acesso direto aos dispositivos de hardware (PEREIRA, 2007).

Para a etapa de classificação foi utilizada a biblioteca de SVM de alto desempenho *Libsvm*, também desenvolvida em linguagem C e consolidada no meio acadêmico (CHANG; LIN, 2011).

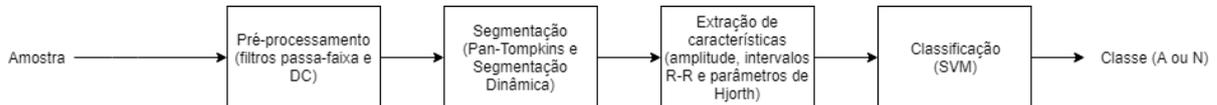
Os algoritmos utilizados em cada etapa foram:

- **Pré-processamento:** Foi implementada a mesma cadeia de filtros passa-faixa utilizados no algoritmo de detecção de picos de Pan-Tompkins (PAN; TOMPKINS, 1985), além de um filtro DC.
- **Segmentação:** Foi utilizado o algoritmo de Pan e Tompkins (1985) para detectar os picos R no sinal cardíaco, e a partir desses picos, a segmentação dinâmica proposta por Leite e Moreno (2017).
- **Extração de características:** Conforme proposto por Leite e Moreno (2017), foram utilizados alguns parâmetros estatísticos simples dos segmentos, além dos parâmetros de Hjorth (HJORTH, 1970).

- **Classificação:** Conforme descrito no item 2.1, foi utilizada uma implementação pronta de *Support Vector Machine* (CHANG; LIN, 2011).

A Figura 3 apresenta o diagrama de blocos do sistema proposto.

Figura 3 – Diagrama de blocos do sistema proposto.



Fonte: Autor

Para o treinamento e os testes foi utilizada a base de dados MIT-BIH Arrhythmia (MOODY; MARK, 2001). A base de dados e cada um dos algoritmos serão descritos a seguir.

2.2 Base de dados

O experimento utilizou sinais da base de dados MIT-BIH Arrhythmia, que contém 48 gravações de meia hora de exames ECG de diversos pacientes com batimentos normais, além de alguns com diferentes tipos de arritmia (MOODY; MARK, 2001). É uma base de dados padrão para testar sistemas de classificação de batimentos cardíacos, e por conta disso pode ser usada para comparar diretamente o desempenho de diferentes trabalhos. Todos os algoritmos individuais implementados neste trabalho já foram anteriormente validados utilizando essa base de dados.

A base de dados separa cada batimento entre 15 tipos diferentes. Porém, a *Association for the Advanced of Medical Instrumentation (AAMI)* definiu 5 classes mais gerais, reagrupando as classes originais (INSTRUMENTATION et al., 1998). Este é o padrão adotado na maioria dos trabalhos na área.

Como o objetivo deste trabalho não é diagnosticar doenças específicas, e sim identificar pacientes potencialmente doentes, um reagrupamento ainda mais geral é proposto, conforme Tabela 1.

Uma limitação da base MIT-BIH, muito frequente em bases de dados médicas em geral, é o fato de que as classes não são balanceadas. Mais de 80% das amostras dessa base pertencem a uma mesma classe, mesmo após reagrupar. Isso pode induzir os classificadores a erros: um classificador que simplesmente responde sempre a mesma classe poderia obter uma taxa de acerto de 80%, mas seria extremamente perigoso, por identificar pacientes doentes como saudáveis, podendo até mesmo levar à sua morte por falta de tratamento médico. Um dos motivos para o reagrupamento proposto é justamente facilitar

a classificação dos pacientes entre saudáveis e não-saudáveis. Posteriormente neste capítulo serão dados mais detalhes sobre estratégias para reduzir o impacto do desbalanceamento das classes, e no capítulo seguinte será explicado como as métricas levam esse problema em consideração para separar um bom classificador de um classificador “viciado”.

Tabela 1 – Reagrupamento proposto das classes da AAMI e dos tipos originais da base MIT-BIH Arrhythmia.

Classes propostas	Classes AAMI	Tipos MIT-BIH
N Normal	N Normal	Batida normal (NOR)
		Bloqueio de ramo de feixes esquerdo (LBBB)
		Bloqueio de ramo de feixes direito (RBBB)
		Batida de escape atrial (AE)
		Batida de escape nodal (ou juncional) (NE)
A Batida anormal	S Ectópica supraventricular	Batida prematura atrial (AP)
		Batida prematura atrial anômala (aAP)
		Batida premadura nodal (ou juncional) (NP)
		Batida prematura supraventricular (SP)
	V Ectópica ventricular	Batida prematura ventricular (PVC)
		Batida de escape ventricular (VE)
	F Batida de fusão	Fusão de batida normal e ventricular (fVN)
		Q Batida desconhecida
	Fusão de batida normal e ritmada (fPN)	
	Batida não-classificada (U)	

Todos os testes utilizaram apenas a derivação *Lead-I* da base de dados. Há técnicas que utilizam mais de uma derivação (LUZ et al., 2016). Porém, algoritmos que trabalham com múltiplas derivações possuem maior complexidade computacional, frequentemente utilizando cálculos matriciais.

Os arquivos originais da base de dados foram pré-processados para descartar todas as informações redundantes ou desnecessárias, deixando apenas uma sequência numérica contendo os valores do sinal para cada amostra da *Lead-I*. Desta maneira, as informações de entrada do algoritmo se assemelham a um sinal obtido diretamente de um sensor, sem nenhum dado extra para que possa ser verificado se dada amostra é ou não um ponto fiducial, a qual classe ela pertence ou se alguma amostra foi perdida.

2.3 Pré-Processamento

O principal objetivo do pré-processamento em um sistema que trabalha com sinais em geral é melhorar a qualidade dos dados. Em sinais biológicos, isso tipicamente consiste em remover o excesso de ruído e ampliar o sinal, de modo que a relação sinal/ruído cresça e haja menos interferência que possa prejudicar o desempenho das etapas posteriores.

2.3.1 Filtro Passa-Faixa

Um dos algoritmos utilizados na etapa seguinte, responsável pela detecção de picos, já implementa um filtro passa-faixa, composto por um filtro passa-baixa e um filtro passa-alta (PAN; TOMPKINS, 1985), com ganhos 36 e 32, respectivamente, e frequências de corte 11 Hz e 5Hz, cujas funções de transferência são dadas pelas equações 2.1 e 2.2. Suas respectivas equações de diferenças para implementação em software são dadas pelas equações 2.3 e 2.4.

$$H(z) = \frac{(1 - z^{-6})^2}{(1 - z^{-1})^2} \quad (2.1)$$

Função de transferência do filtro passa-baixa

$$H(z) = \frac{(-1 + 32z^{-16} + z^{-32})}{(1 + z^{-1})} \quad (2.2)$$

Função de transferência do filtro passa-alta

$$y(nT) = 2y(nT - T) - y(nT - 2T) + x(nT) - 2x(nT - 6T) + x(nT - 12T) \quad (2.3)$$

Equação de diferenças do filtro passa-baixa

$$y(nT) = 32x(nT - 16T) - [y(nT - T) + x(nT) - x(nT - 32T)] \quad (2.4)$$

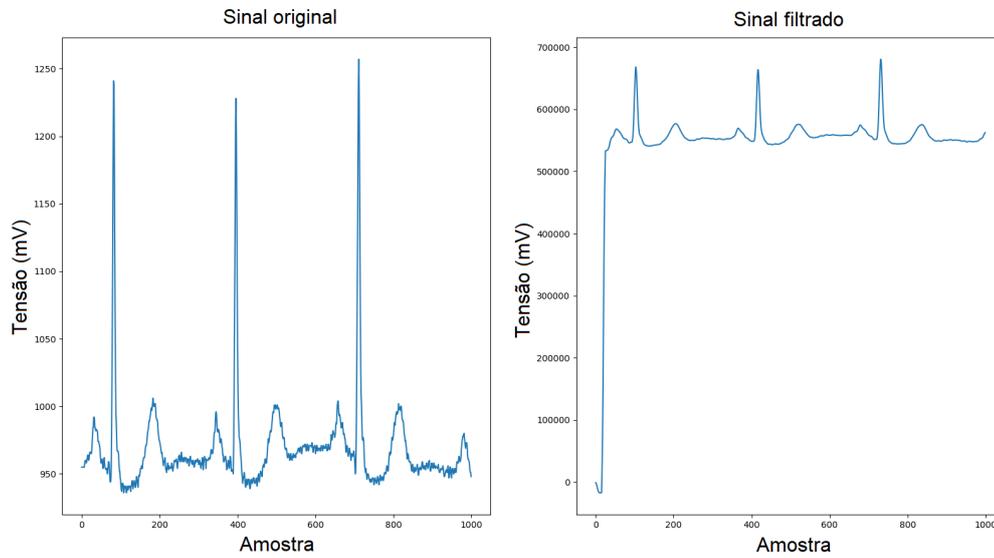
Equação de diferenças do filtro passa-alta

2.3.2 Filtro DC

Apesar da implementação deste trabalho ter utilizado os mesmos sinais de teste – a base de dados MIT-BIH (MOODY; MARK, 2001), disponibilizada na plataforma PhysioNet (GOLDBERGER et al., 2000), descrita no item 2.2, o trabalho original fez a sua própria conversão A/D a partir das fitas originais, enquanto este trabalho utilizou uma versão já digitalizada amplamente divulgada. Devido a diferenças no processo de digitalização e nas características dos conversores utilizados, os filtros apresentaram um problema por conta de uma componente DC que não estava presente no trabalho original, como ilustrado na Figura 4.

Para resolver o problema, foi implementado um filtro DC simples. Sua função de transferência é dada pela equação 2.5, e sua equação de diferenças pela equação 2.6.

Figura 4 – Efeito da componente DC no filtro passa-faixa.



Fonte: Autor

$$H(z) = \frac{(1 - z^{-1})}{(1 - 0.995z^{-1})} \quad (2.5)$$

Função de transferência do filtro DC

$$y(nT) = x(nT) - x(nT - T) + 0.995y(nT - T) \quad (2.6)$$

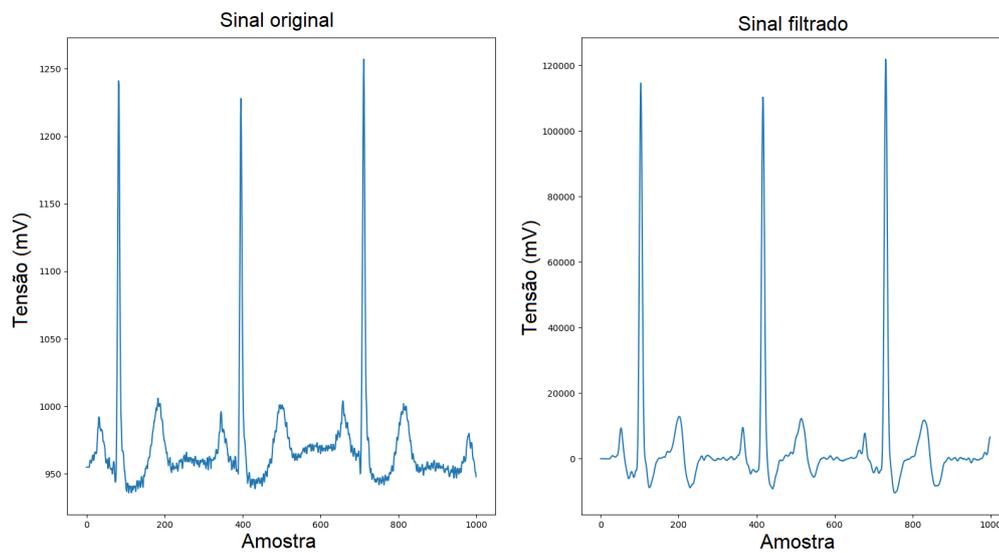
Equação de diferenças do filtro DC

Após a implementação do filtro DC, o sinal filtrado torna-se muito mais próximo do sinal original e não provoca nenhum tipo de problema nos módulos seguintes do sistema. A Figura 5 ilustra o resultado.

Através de inspeção visual das formas de onda em diferentes estágios do pré-processamento é possível observar que há um certo atraso de algumas amostras provocado pelos filtros. Esse atraso será compensado na etapa seguinte.

Tanto para a aquisição do sinal original quanto para o processo de filtragem foram utilizados *buffers* circulares: dois vetores, um para o sinal original, e outro para o sinal filtrado. Sempre que uma nova amostra é obtida, todo o conteúdo do vetor é deslocado uma posição para trás, e a nova amostra é salva em sua última posição. Os filtros então são aplicados, o seu vetor também é deslocado em uma posição e a última posição é utilizada pela nova amostra do sinal filtrado.

Figura 5 – Resultado da combinação dos filtros passa-faixa e DC.



Fonte: Autor

Os *buffers* são mantidos tanto para o cálculo do sinal filtrado (que utiliza diversas amostras do sinal original) quanto para uso de dados anteriores para obter novas informações (como detectar um possível pico perdido, na etapa do algoritmo de Pan-Tompkins quanto para extrair dados de um batimento completo já segmentado, na etapa de extração de características).

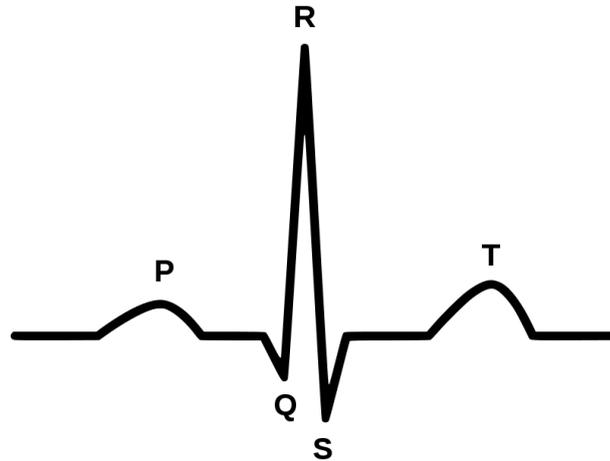
2.4 Segmentação

Um sinal de ECG correspondente a um batimento cardíaco tipicamente contém 3 partes visualmente identificáveis: a onda P, o complexo QRS (cujo ponto fiducial é o ponto R) e a onda T (SPACH; KOOTSEY, 1983). A Figura 6 foi extraída de um ECG real e destaca o ponto fiducial.

Para a etapa de segmentação, foi adotada a técnica de segmentação dinâmica proposta por Leite e Moreno (2017). Ela consiste nos seguintes passos:

1. Sempre que um novo pico R é detectado, guarda-se o intervalo entre ele e o pico anterior em um vetor.
2. Calcula-se a média entre últimos 7 intervalos para descobrir a duração média considerando a frequência cardíaca atual.
3. Considera-se a duração do sinal como a duração média, e utiliza-se as seguintes proporções:

Figura 6 – Ondas presentes em um ECG normal.



Fonte: Wikimedia Commons (COMMONS, 2019)

- Onda P: primeiros 25% do sinal.
- Complexo QRS: os próximos 25% do sinal.
- Onda T: todas as amostras restantes.

Sendo que 35% das amostras se situam antes do ponto fiducial, e o restante após o mesmo.

A técnica proposta é computacionalmente simples, possuindo número constante de operações, mas ao mesmo tempo é mais confiável do que técnicas amplamente utilizadas (como adotar comprimento fixo), visto que ela se adapta rapidamente a variações do ritmo cardíaco e consegue segmentar corretamente tanto batidas normais quanto diversas classes de arritmia.

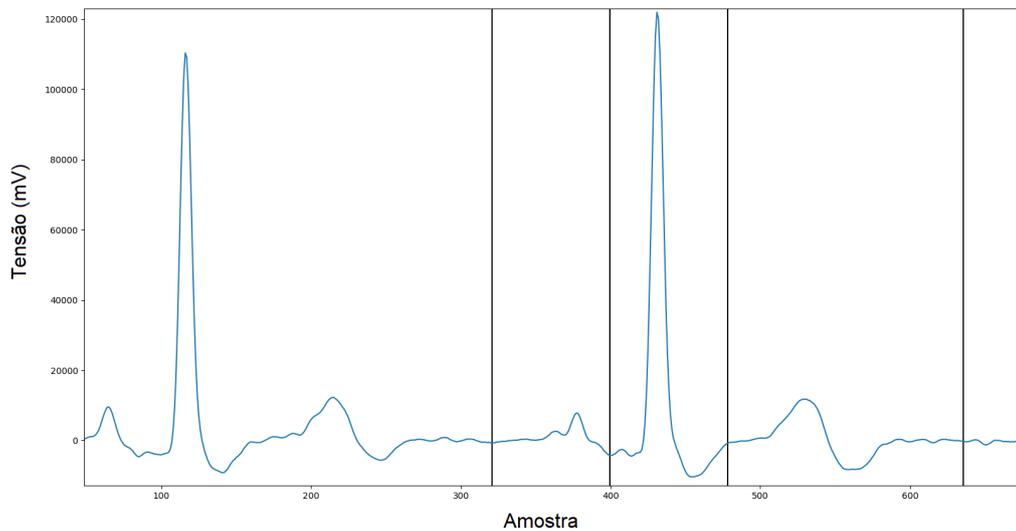
A Figura 7 mostra os pontos onde o batimento se inicia, as transições da onda P para o complexo QRS e do complexo QRS para a onda T, e o final do batimento.

Porém, a técnica proposta supõe conhecimento dos pontos fiduciais. Para isso é necessário utilizar um algoritmo detector de picos.

2.4.1 Detecção de Pontos Fiduciais

Para detectar os picos cardíacos, foi escolhido o algoritmo de Pan e Tompkins (1985). Ele é um algoritmo de baixo custo computacional desenvolvido para detecção em tempo real de pontos fiduciais em sinais de eletrocardiograma. Além disso, ele é robusto, sendo capaz de se adequar a dificuldades diversas, como variações na frequência cardíaca, ondas T com amplitude elevada e alguns tipos de arritmia. Ele é um dos algoritmos com maior precisão, menor custo computacional e é amplamente utilizado até hoje em sistemas

Figura 7 – Resultado da segmentação dinâmica.



Fonte: Autor

de classificação de batidas (LUZ et al., 2016), o que o torna uma excelente escolha para um sistema embarcado de tempo real.

Além do sinal original (devidamente filtrado), o algoritmo utiliza também as funções integral (utilizando a técnica de janela deslizante) e derivada para obter informações adicionais sobre o sinal. A fórmula original da derivada utilizada pelo artigo foi substituída pela implementação de Leite e Moreno (2017), que resulta em uma simples diferença entre duas amostras consecutivas, sem prejuízo para a forma de onda.

A integral traz informações importantes sobre a amplitude e a largura do complexo QRS, demonstradas na Figura 8. É sugerida uma janela de largura equivalente a 150ms.

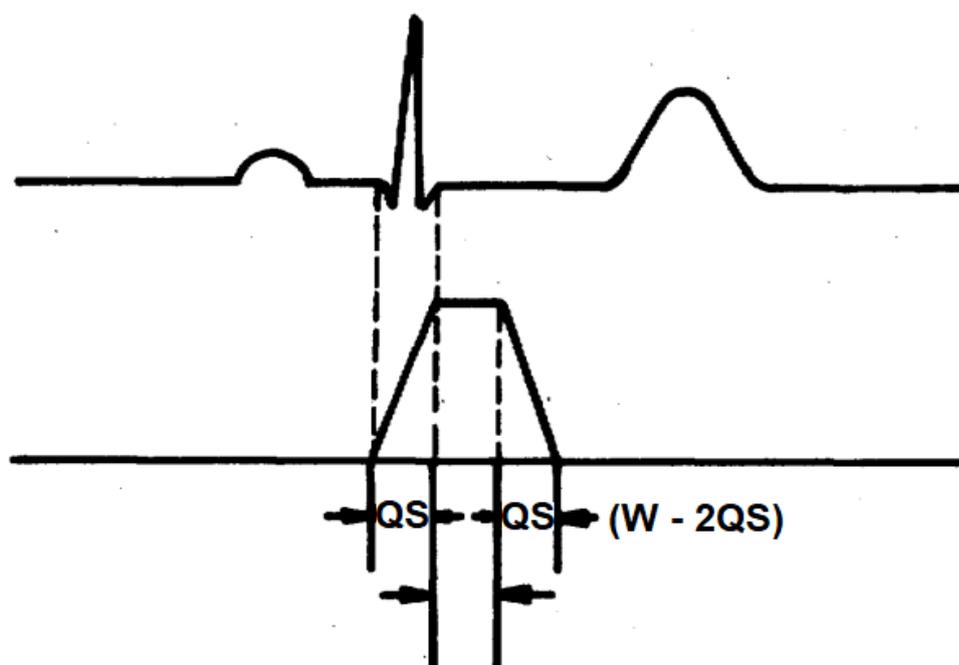
Assim como o sinal original e o sinal filtrado, os valores de ambas as derivadas e da integral são mantidos em *buffers* circulares para possível uso posterior.

Para verificar se uma amostra é ou não um pico, são mantidos alguns valores de limiar: um para o sinal original, um para a integral e outros dois que correspondem à metade dos dois primeiros. O cálculo deles utiliza os valores de ruído (possíveis "picos" que não correspondem a complexos QRS) e os valores dos picos anteriormente detectados.

Quando o valor de uma amostra e o valor correspondente da integral são simultaneamente maiores do que seus respectivos limiares, automaticamente considera-se que a amostra é um ponto fiducial. Os limiares são todos reajustados.

Porém, se um possível pico é encontrado em menos de 360ms após a última detecção, deve-se considerar a hipótese de ser apenas a onda T. Verifica-se então o valor de sua

Figura 8 – Relação entre a integral e o complexo QRS. QS é a largura do complexo QRS, e W é a largura da janela deslizante da integral.



Fonte: Adaptado de Pan e Tompkins (1985)

derivada ao quadrado – o quadrado elimina a parte negativa do sinal e amplifica a onda de forma não-linear, “exagerando” o valor dos picos. Caso esse valor seja superior à metade do quadrado da derivada do último pico, ele é considerado um ponto fiducial e todos os limiares são recalculados. Caso contrário, ele é considerado como uma onda T.

A duração dos últimos 7 intervalos entre pontos fiduciais, que serão referenciados por *intervalos R-R*, são armazenados. Considera-se como um intervalo “regular” se estiver entre 92% e 116% da média dos últimos 7 intervalos. Um segundo vetor armazena os valores dos últimos 7 intervalos regulares.

Quando é observado que o ritmo cardíaco está irregular (o último batimento fugiu da janela considerada regular), os valores de todos os limiares são divididos pela metade, para tornar o algoritmo mais sensível.

Caso o algoritmo leve mais do que 166% do intervalo médio para detectar um novo ponto fiducial, considera-se que ele pode ter deixado de detectar um pico. É refeita a análise das amostras anteriores utilizando a segunda versão dos limiares, correspondentes à metade das primeiras.

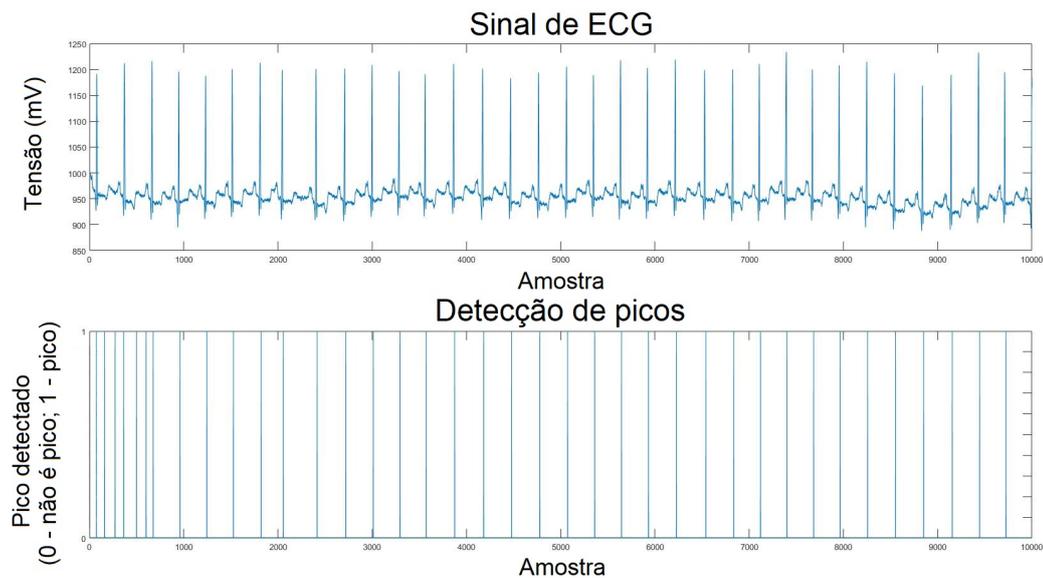
Sempre que um pico é detectado, considera-se um período refratário de 200ms durante o qual nenhuma nova detecção pode ocorrer, pois é fisiologicamente impossível que 2 complexos QRS ocorram neste intervalo. Qualquer pico em potencial neste intervalo

é uma onda T.

2.4.1.1 Período de aprendizado

Segundo os autores, o algoritmo leva aproximadamente 2 segundos para ajustar adequadamente os seus pesos. Até que isso ocorra, diversas detecções errôneas ocorrerão, e por conta disso esses sinais devem ser descartados. A Figura 9 ilustra esse problema.

Figura 9 – Tempo de aprendizado do algoritmo de Pan-Tompkins.



Fonte: Autor

2.4.1.2 Atrasos na detecção

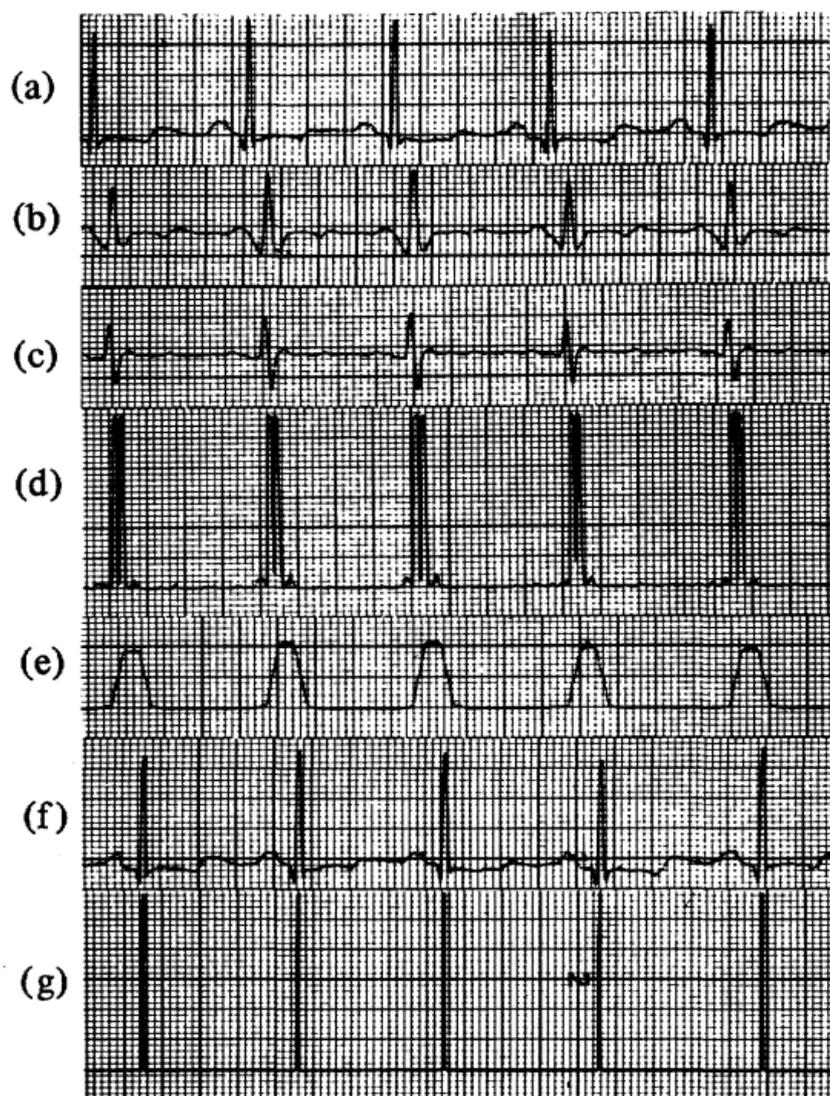
A cascata de filtros provoca um atraso na saída do algoritmo, observado pelos autores originais na Figura 10.

Neste trabalho, para resolver tanto a questão do tempo de ajuste do algoritmo quanto do atraso adicionado pelo pré-processamento, as primeiras amostras da saída são descartadas. A Figura 11 realiza comparação semelhante à da Figura 10 considerando as formas de ondas gerada pela implementação proposta.

2.5 Extração de características

A extração de características é fundamental no sucesso da classificação dos sinais. Mesmo um bom classificador apresentará maus resultados se receber dados pouco significativos (LUZ et al., 2016).

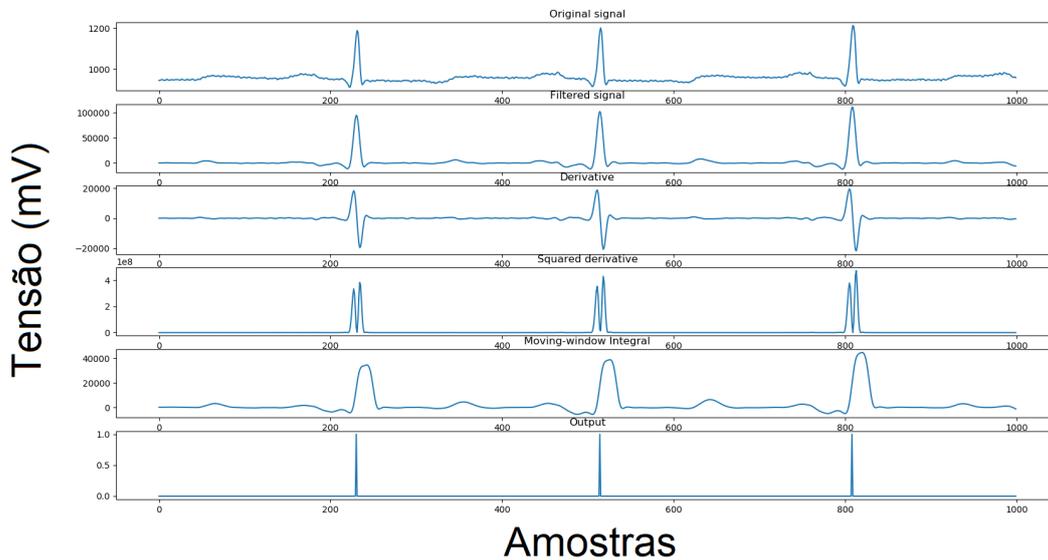
Figura 10 – Formas de onda geradas pelo algoritmo de Pan-Tompkins: A) Sinal original. b) Sinal filtrado. c) Derivada. d) Derivada ao quadrado. e) Integral por janela deslizante. f) Sinal original com o atraso provocado pelos filtros. g) Saída do algoritmo.



Fonte: Pan e Tompkins (1985)

Essa etapa é necessária para resolver um problema conhecido como a *maldição da dimensionalidade*: mesmo amostrado a frequências relativamente baixas, o sinal elétrico correspondente a um único batimento poderá ter de centenas a milhares de amostras. Isso aumentaria consideravelmente a complexidade do classificador, gerando diversos possíveis problemas, como o *overfitting* do classificador (fenômeno onde o classificador se ajusta especificamente para as amostras utilizadas em seu treinamento, perdendo a capacidade de generalizar), tempo inviável para conclusão do treinamento, e até mesmo um aumento significativo do tempo da classificação, que poderá impactar no desempenho global do sistema (LUZ et al., 2016).

Figura 11 – Formas de onda geradas pela implementação proposta: Sinal original; Sinal filtrado; Derivada; Derivada ao quadrado; Integral por janela deslizante; Saída do algoritmo.



Fonte: Autor

Um conjunto de 21 características de cada batimento foi utilizado neste trabalho: 3 características relacionadas a um intervalo entre 2 picos R, 3 características relacionadas à amplitude da cada segmento (onda P, complexo QRS e onda T) e 3 parâmetros de Hjorth para cada segmento.

2.5.1 Intervalos R-R

Três características relacionadas ao intervalo R-R foram utilizadas:

- **Pre-RR**: o intervalo entre o ponto fiducial atual e o seu antecessor.
- **Pos-RR**: o intervalo entre o ponto fiducial atual e o seu sucessor.
- **Recent-RR**: a média dos 10 intervalos R-R mais recentes.

A duração de intervalos R-R não é suficiente para indicar problemas: apesar de certos batimentos defeituosos apresentarem durações atípicas, eles também podem simplesmente indicar variações no ritmo cardíaco (LEITE; MORENO, 2017). Por esse motivo, versões normalizadas de Pre-RR e Pos-RR foram utilizadas, dividindo seus valores por Recent-RR. Sendo assim, as informações sobre intervalos R-R utilizadas não são absolutas, mas relativas à taxa de batimentos cardíacos.

2.5.2 Características de amplitude

Devido à gama de formas de onda possíveis para um sinal cardíaco normal (inclusive para um mesmo paciente), dados estatísticos sobre cada segmento de um batimento podem trazer algumas pistas importantes, como a ausência de uma das ondas, ou uma amplitude exagerada para uma delas (LEITE; MORENO, 2017).

Três valores foram usados para cada segmento: seus valores máximo e mínimo, e também a média de todas as amostras. Todos esses valores podem ser obtidos em uma mesma malha de repetição, tornando seu cálculo computacionalmente simples.

2.5.3 Parâmetros de Hjorth

Ao estudar sinais de eletroencefalograma (EEG), Hjorth propôs um método para obter informações quantitativas no domínio do tempo sobre esses sinais (HJORTH, 1970). Essas informações, hoje conhecidas como parâmetros de Hjorth, se provaram úteis para caracterizar diversos tipos diferentes de sinais biológicos, como no design de Interfaces Cérebro-Máquina (VIDAURRE et al., 2009), eletromiografia (MOUZÉ-AMADY; HORWAT, 1996) e a caracterização de sons de pulmão (RIZAL; HIDAYAT; NUGROHO, 2015).

Esses parâmetros não são números puramente matemáticos: eles possuem uma interpretação física relacionada ao comportamento do sinal, e podem ser utilizados para obter informações sobre potência e frequências presentes. Porém, seu cálculo é mais simples e exige menos recursos computacionais do que técnicas tradicionais de análise em frequência, como a transformada de Fourier ou wavelets.

Já foi demonstrado que esses parâmetros são uma alternativa confiável e computacionalmente mais eficiente do que outros métodos de extração de características comumente utilizados para sinais de ECG, mas seu potencial ainda foi pouco explorado na área (LEITE; MORENO, 2017).

Esses parâmetros são:

- **Atividade:** Mede a potência média no domínio do tempo. É calculada por:

$$Atividade(y(t)) = var(y(t)) = \frac{1}{N} \sum_{i=1}^N (y(i) - \mu)^2. \quad (2.7)$$

- **Mobilidade:** Mede a taxa de variação média, trazendo informação sobre o formato da curva. É dada por:

$$Mobilidade(y(t)) = \sqrt{\frac{Atividade\left(\frac{dy(t)}{dt}\right)}{Atividade(y(t))}}. \quad (2.8)$$

- **Complexidade:** É uma medida do quanto a frequência do sinal varia em relação a uma onda senoidal pura. Aproxima-se de 1 conforme o sinal se aproxima da senóide. É calculada por:

$$Complexidade(y(t)) = \frac{Mobilidade(\frac{dy(t)}{dt})}{Mobilidade(y(t))}. \quad (2.9)$$

Cada parâmetro é calculado para cada segmento (onda P, complexo QRS e onda T) do batimento cardíaco.

2.6 Classificação (SVM)

Máquinas de Vetores de Suporte (*Support Vector Machines* – SVM) são algoritmos de aprendizado supervisionado com grande capacidade de generalização frequentemente utilizados em problemas de classificação de padrões. Eles utilizam um subconjunto representativo das características apresentadas pelo conjunto de treinamento para criar vetores de suporte.

Esses classificadores utilizam um kernel – uma função matemática – para mapear vetores de entrada para um novo espaço vetorial com dimensionalidade superior. Assim, classes que não eram facilmente separadas em seu espaço original passam a ser separáveis por um hiperplano. Neste trabalho, foi utilizada a função de base radial (*Radial Basis Function* – RBF), definida por:

$$K(\vec{u}, \vec{v}) = \exp(-g \|\vec{u} - \vec{v}\|^2), g > 0. \quad (2.10)$$

Três passos importantes foram necessários durante o treinamento para melhorar a precisão do classificador: encontrar os parâmetros ótimos, normalizar os dados e balancear a base de dados.

2.6.1 Otimização dos parâmetros da SVM.

Além do parâmetro “g” do kernel RBF, existe um parâmetro “C” (custo) utilizado pela SVM para determinar a margem de separação entre as classes. Valores altos tornam o classificador menos tolerante a erros de classificação, mas reduzem sua capacidade de generalização (*overfitting*).

Os valores ótimos foram obtidos utilizando um algoritmo de *grid search*: foram testadas todas as combinações possíveis entre diversas faixas de valores para ambas as variáveis utilizando verificação cruzada *5-fold*, na qual o conjunto de treinamento é subdividido em 5 partes. São feitas 5 rodadas de treinamento e teste para cada valor, em cada é utilizado uma das frações para treinamento e as outras 4 para teste. Foram obtidos os valores de $C = 2000$ e $g = 0,01$.

2.6.2 Parametrização dos dados de entrada

A grande faixa de valores possíveis no vetor de entrada, desde pequenas frações a números extremamente altos, torna o treinamento mais lento e mais propenso a erros. Dessa maneira, todos os valores dos vetores de entrada foram normalizados para o intervalo (-1, 1). Essa etapa pode diminuir um pouco a precisão do sistema, mas reduz consideravelmente o tempo de treinamento e evita alguns tipos de erro.

2.6.3 Balanceando a base de dados

Bases de dados desbalanceadas são aquelas nas quais algumas classes possuem muito mais representantes do que outras. Esse é um problema comum em problemas médicos: a maioria das amostras tende a representar dados “saudáveis” ou relacionados a doenças comuns, e poucas amostras representam doenças mais raras. No caso da base MIT-BIH Arrhythmia, cerca de 80% das amostras pertencem à classe N (segundo a classificação da AAMI), enquanto os outros 20% se distribuem entre outras 4 classes.

Utilizar uma base de dados desbalanceada no treinamento pode ser problemática por induzir os mecanismos de pontuação do algoritmo de treinamento ao erro: no caso da base de dados utilizada, por exemplo, o classificador poderia responder “N” para todas as amostras e ainda obteria 80% de acerto.

Foram feitos dois ajustes para resolver isso: a redução do número de classes e a atribuição para cada classe de pesos, utilizando pesos maiores para classes com menos representantes. A redução do número de classes nem sempre é possível de ser realizada. Neste trabalho, foi feita uma concessão devido ao escopo do próprio projeto: como o objetivo é apenas realizar uma triagem entre pacientes saudáveis e pacientes com algum tipo de anomalia, e não o diagnóstico preciso da doença, as 4 classes diferentes de “N” foram agrupadas em uma única classe “A”.

Para os valores de C e g obtidos, foram utilizados, respectivamente, pesos 1 e 2,5 para as classes N e A. Estes valores foram obtidos empiricamente de maneira similar à *grid search*: partindo de pesos inversamente proporcionais ao número de classes, foram testadas diferentes combinações aumentando e diminuindo cada um dos pesos. A implementação da SVM utiliza esses pesos junto do custo “ C ” para criar custos individuais para cada classe.

A implementação da SVM utilizada é a *Libsvm* (CHANG; LIN, 2011).

3 Resultados

Todos os testes foram executados em um laptop com um processador Intel i5-8520U e 16GB de memória RAM. Para verificar a execução em um sistema embarcado foi utilizada uma Raspberry Pi 3 Model B+, exibida na Figura 12 junto de um cartão de crédito para referência de tamanho. Suas especificações são:

- Cortex-A53 (ARMv8) 64-bit quadcore CPU @ 1.4GHz,
- 1GB LPDDR2 SDRAM,
- Raspbian com kernel Linux 4.14.

Figura 12 – Comparação de dimensões entre uma Raspberry Pi e um cartão de crédito.



Fonte: Autor

Para o experimento, os sinais foram lidos uma amostra por vez, simulando a aquisição online de um sinal através de sensores, como ocorreria em um ambiente real de uso do equipamento. A frequência de amostragem não foi fixada: ao invés disso, o tempo para classificação do sinal foi medido para verificar se ele seria capaz de realizar a classificação em tempo real na amostragem original da base de dados.

3.1 Classificação

3.1.1 Métricas adotadas

Os resultados da classificação são exibidos na forma de uma matriz de confusão: as linhas representam a classe real de cada amostra, enquanto as colunas mostram como elas foram classificadas. Isso permite visualizar mais facilmente quais tipos de erro de classificação são mais comuns.

Conforme explicado anteriormente, meramente observar a taxa de acerto pode induzir ao erro: a falta de balanceamento das classes permitiria que um classificador sempre classificasse todas as amostras como pertencendo à classe mais comum e ainda assim fosse considerado altamente preciso. Portanto, dois parâmetros estatísticos foram utilizados na avaliação: Sensitividade (Se) e Valor Preditivo Positivo (+P).

A Sensitividade relaciona o número de falsos negativos com o total de positivos verdadeiros. É dada pela equação 3.1, onde TP é o número de positivos verdadeiros para uma classe e FN é o número de falsos negativos. Ela mostra quão relevantes são os resultados retornados pelo classificador.

$$Se = 100\left(\frac{TP}{TP + FN}\right). \quad (3.1)$$

O Valor Preditivo Positivo, também conhecido como "precisão", relaciona o número de falsos negativos e o número de positivos verdadeiros para uma classe. É dado pela equação 3.2, onde TP é o número de positivos verdadeiros e FP é o número de falsos negativos. Ele indica a probabilidade de um valor anômalo corresponder a uma doença real.

$$+P = 100\left(\frac{TP}{TP + FP}\right). \quad (3.2)$$

3.1.1.1 Resultados da classificação

A matriz de confusão do sistema é dada na Tabela 2. Os valores de S e +P calculados são exibidos na Tabela 3.

Tabela 2 – Matriz de confusão do sistema.

Real/Detectado	N	A
N	36145	7768
A	2612	4375

Como todos os trabalhos utilizados na comparação adotaram as classes recomendadas pela AAMI, novos índices foram calculados a partir de suas respectivas matrizes de

confusão, reagrupando-os conforme proposto na Tabela 1. Os resultados são apresentados na Tabela 3.

Tabela 3 – Índices de performance.

Trabalho	N Se	N +P	A Se	A +P
De Chazal (CHAZAL; O'DWYER; REILLY, 2004)	86.86	99.16	94.00	46.86
Chen (CHEN et al., 2017)	98.42	95.42	62.58	83.38
Llamedo (LLAMEDO; MARTÍNEZ, 2010)	77.55	99.47	96.67	34.65
Ye (YE; KUMAR; COIMBRA, 2012)	89.99	98.15	86.44	51.93
Zhang (ZHANG et al., 2014)	88.94	98.97	92.83	51.87
Leite (LEITE; MORENO, 2017)	96.68	95.42	62.46	69.88
Proposto	82.31	93.26	62.62	36.03

3.2 Tempo de execução

Cada sinal de 30 minutos de duração foi lido uma amostra por vez. Os testes foram executados no laptop e repetidos na Raspberry Pi. O critério para verificar o sucesso ou fracasso da classificação em tempo real foi o tempo de execução: se o sistema foi capaz de ler e classificar todo o sinal em um período de tempo menor do que a duração do mesmo, ele seria capaz de realizar a tarefa em tempo real, conforme o sinal é amostrado. Caso contrário, ele não seria capaz, pois o *buffer* acumularia cada vez mais amostras e acabaria sendo completamente preenchido, levando à perda de amostras ou ao travamento do sistema.

No laptop, todos os sinais foram devidamente amostrados, classificados e seus resultados exportados em arquivos CSV em menos de 20 minutos cada, indicando que o sistema seria capaz de realizar a tarefa e ainda haveria uma margem de segurança, com tempo ocioso da CPU.

Porém, na Raspberry Pi, nenhum teste levou menos de duas horas, o que significa que em um caso real muitas amostras seriam perdidas, prejudicando a qualidade da classificação ou mesmo causando travamentos no sistema. Apesar das etapas de segmentação de batimentos e extração de características terem sido extremamente rápidas – levando poucos minutos para gerar um CSV com os dados correspondentes a horas de sinal – o módulo classificador foi um gargalo. Apesar de todas as precauções tomadas para simplificar os dados e a execução dos testes (como exibir a saída de cada classificação simplesmente como um sinal binário em um pino de GPIO), os mesmos ainda se provaram excessivamente complexos, resultando em modelos de SVM com quantidades de vetores de suporte na ordem de 10^5 . Classificar um sinal utilizando esses modelos implica em realizar múltiplos de 10^5 cálculos vetoriais, o que se provou ser muito complexo mesmo para um microcontrolador de alta performance (para o padrão de computação embarcada).

4 Conclusão

Um sistema completo para a classificação automática em tempo real de batimentos cardíacos foi proposto utilizando técnicas computacionalmente simples: filtros IIR, detecção de complexos de QRS de Pan-Tompkins, segmentação dinâmica de sinais cardíacos, parâmetros de Hjorth e um classificador SVM.

O sistema foi treinado com dados da base de dados MIT-BIH Arrhythmia utilizando diferentes estratégias visando tanto a melhoria da qualidade da classificação quanto o ganho de desempenho computacional, buscando viabilizar seu uso em sistemas computacionalmente simples, para que no futuro soluções embarcadas de baixo custo pudessem ser utilizadas em áreas de difícil acesso e/ou pouco desenvolvimento financeiro.

Diferentes testes foram realizados buscando aferir tanto a viabilidade da integração de todas as técnicas (cuja eficácia havia sido comprovada separadamente e, na maioria dos casos, apenas em análises com sinais previamente armazenados e condições ideais) quanto o seu desempenho computacional em diferentes equipamentos.

O trabalho obteve indicadores (Sensitividade e Valor Preditivo Positivo) um pouco inferiores quando comparado com outros trabalhos recentes, mas seus índices se situam na mesma faixa de valores de alguns destes trabalhos. Uma queda nos indicadores de acerto era esperado devido ao fato da análise ter sido realizada online, introduzindo novas fontes de erro (ex: na detecção dos picos, etapa não realizada por muitos outros trabalhos que focam apenas no desempenho da etapa de extração de características e/ou classificação).

Ficou demonstrado o potencial de uso das técnicas propostas, bem como o baixo custo computacional das etapas de segmentação e extração de características testadas pela primeira vez em um ambiente em tempo real para classificação de batimentos cardíacos.

Porém, não foi obtido sucesso em simplificar o suficiente a etapa de classificação para viabilizar sua implementação real em um sistema embarcado de baixo custo. Desta maneira, convém que trabalhos futuros investiguem o desempenho das etapas de segmentação e extração de características junto a outros algoritmos de classificação. Outras frentes que poderiam ser investigadas seriam a implementação do sistema em FPGA, em hardware novo que venha a ser desenvolvido nos próximos anos e também uma abordagem *baremetal*, para verificar a sobrecarga de processamento provocada pelo uso de um sistema operacional no sistema.

Referências

- BARR, M. Programming embedded systems in c/c++. *Inc., Sebastopol, California*, O Reilly & Associates, 1999. Citado na página 10.
- CHANG, C.-C.; LIN, C.-J. Libsvm: A library for support vector machines. *ACM transactions on intelligent systems and technology (TIST)*, Acm, v. 2, n. 3, p. 27, 2011. Citado 3 vezes nas páginas 13, 14 e 27.
- CHAZAL, P. D.; O'DWYER, M.; REILLY, R. B. Automatic classification of heartbeats using ecg morphology and heartbeat interval features. *IEEE transactions on biomedical engineering*, IEEE, v. 51, n. 7, p. 1196–1206, 2004. Citado na página 30.
- CHEN, S. et al. Heartbeat classification using projected and dynamic features of ecg signal. *Biomedical Signal Processing and Control*, Elsevier, v. 31, p. 165–173, 2017. Citado na página 30.
- COMMONS, W. *Sinus rhythm without labels*. 2019. Disponível em: <https://commons.wikimedia.org/wiki/File:SinusRhythmLabels.svg#/media/File:SinusRhythm_withoutLabels.svg>. Citado na página 19.
- FAYN, J.; RUBEL, P. Toward a personal health society in cardiology. *IEEE Transactions on Information technology in Biomedicine*, IEEE, v. 14, n. 2, p. 401–409, 2009. Citado na página 11.
- GOLDBERGER, A. L. et al. Physiobank, physiotoolkit, and physionet: components of a new research resource for complex physiologic signals. *Circulation*, Am Heart Assoc, v. 101, n. 23, p. e215–e220, 2000. Citado na página 16.
- HJORTH, B. Eeg analysis based on time domain properties. *Electroencephalography and clinical neurophysiology*, Elsevier, v. 29, n. 3, p. 306–310, 1970. Citado 3 vezes nas páginas 12, 13 e 25.
- HOLTER, N. J. New method for heart studies: continuous electrocardiography of active subjects over long periods is now practical. *Science*, American Association for the Advancement of Science, v. 134, n. 3486, p. 1214–1220, 1961. Citado na página 9.
- INSTRUMENTATION, A. for the Advancement of M. et al. Testing and reporting performance results of cardiac rhythm and st segment measurement algorithms. *ANSI/AAMI EC38*, v. 1998, 1998. Citado na página 14.
- LEITE, J. P. R.; MORENO, R. L. Heartbeat classification with low computational cost using hjorth parameters. *IET Signal Processing*, IET, v. 12, n. 4, p. 431–438, 2017. Citado 7 vezes nas páginas 12, 13, 18, 20, 24, 25 e 30.
- LLAMEDO, M.; MARTÍNEZ, J. P. Heartbeat classification using feature selection driven by database generalization criteria. *IEEE Transactions on Biomedical Engineering*, IEEE, v. 58, n. 3, p. 616–625, 2010. Citado na página 30.

- LUZ, E. J. d. S. et al. Ecg-based heartbeat classification for arrhythmia detection: A survey. *Computer methods and programs in biomedicine*, Elsevier, v. 127, p. 144–164, 2016. Citado 7 vezes nas páginas 11, 12, 13, 15, 20, 22 e 23.
- MAZOMENOS, E. B. et al. A low-complexity ecg feature extraction algorithm for mobile healthcare applications. *IEEE journal of biomedical and health informatics*, IEEE, v. 17, n. 2, p. 459–469, 2013. Citado na página 10.
- MOODY, G. B.; MARK, R. G. The impact of the mit-bih arrhythmia database. *IEEE Engineering in Medicine and Biology Magazine*, IEEE, v. 20, n. 3, p. 45–50, 2001. Citado 2 vezes nas páginas 14 e 16.
- MOUZÉ-AMADY, M.; HORWAT, F. Evaluation of hjorth parameters in forearm surface emg analysis during an occupational repetitive task. *Electroencephalography and Clinical Neurophysiology/Electromyography and Motor Control*, Elsevier, v. 101, n. 2, p. 181–183, 1996. Citado na página 25.
- OLIVEIRA, A. S.; ANDRADE, F. S. *Sistemas Embarcados - Hardware e Firmware na Prática*. [S.l.]: Ed. Érica, 2006. ISBN 8536501057. Citado na página 10.
- PAN, J.; TOMPKINS, W. J. A real-time qrs detection algorithm. *IEEE Trans. Biomed. Eng.*, v. 32, n. 3, p. 230–236, 1985. Citado 5 vezes nas páginas 13, 16, 19, 21 e 23.
- PEREIRA, F. *Microcontroladores PIC: programação em C*. [S.l.]: Érica, 2007. Citado na página 13.
- RIZAL, A.; HIDAYAT, R.; NUGROHO, H. A. Determining lung sound characterization using hjorth descriptor. In: IEEE. *2015 International Conference on Control, Electronics, Renewable Energy and Communications (ICCEREC)*. [S.l.], 2015. p. 54–57. Citado na página 25.
- ROWE, A.; ROSENBERG, C.; NOURBAKHSI, I. A low cost embedded color vision system. In: IEEE. *Intelligent Robots and Systems, 2002. IEEE/RSJ International Conference on*. [S.l.], 2002. v. 1, p. 208–213. Citado na página 10.
- SILVA, I.; MOODY, G. B.; CELI, L. Improving the quality of ecgs collected using mobile phones: The physionet/computing in cardiology challenge 2011. In: IEEE. *2011 Computing in Cardiology*. [S.l.], 2011. p. 273–276. Citado na página 10.
- SILVA, J. D. S. Uma plataforma para sistemas embarcados: Desenvolvimento e avaliação de desempenho de um processador rtl e uma cache l1 usando systemc. *Relatório Final de Graduação (Bacharel em Ciências da Computação)-Curso de Ciências da Computação, Universidade Federal do Rio Grande do Norte, Natal, 2006*. Citado na página 13.
- SPACH, M. S.; KOOTSEY, J. M. The nature of electrical propagation in cardiac muscle. *American Journal of Physiology-Heart and Circulatory Physiology*, American Physiological Society Bethesda, MD, v. 244, n. 1, p. H3–H22, 1983. Citado na página 18.
- SRIDHARA, S. R. et al. Microwatt embedded processor platform for medical system-on-chip applications. *IEEE Journal of Solid-State Circuits*, IEEE, v. 46, n. 4, p. 721–730, 2011. Citado na página 11.
- VETTERLING, W. T. et al. *Numerical recipes*. [S.l.]: University Press, 1989. Citado na página 13.

VIDAURRE, C. et al. Time domain parameters as a feature for eeg-based brain–computer interfaces. *Neural Networks*, Elsevier, v. 22, n. 9, p. 1313–1319, 2009. Citado na página 25.

YE, C.; KUMAR, B. V.; COIMBRA, M. T. Heartbeat classification using morphological and dynamic features of ecg signals. *IEEE Transactions on Biomedical Engineering*, IEEE, v. 59, n. 10, p. 2930–2941, 2012. Citado na página 30.

ZHANG, Z. et al. Heartbeat classification using disease-specific feature selection. *Computers in biology and medicine*, Elsevier, v. 46, p. 79–89, 2014. Citado na página 30.