

MONITORAMENTO REMOTO PERVASIVO DA SAÚDE UTILIZANDO DISPOSITIVO MÓVEL

Clairton Siebra¹, Bruno Adonis¹, Joanacelle Caldas¹ e Raniere Medeiros¹

¹Centro de Informática, Universidade Federal da Paraíba, João Pessoa – PB - Brasil

Resumo: Este trabalho aplica o método Sliding Window (SW) associado a uma Rede Neural Artificial (RNA) para consolidação de dados advindos de um acelerômetro para o monitoramento de movimentos humanos. A partir desses dados foi extraído um vetor de entrada, para o classificador, com quatro características. Foram feitas várias combinações entre os parâmetros da SW, otimizando a média de acertos, atingindo então 92,63%. Conclui-se que a Sliding Window associada a RNA é apropriada para detecção dos seis movimentos aqui estudados. Esta técnica pode ser amplamente utilizada no monitoramento remoto de pacientes de forma menos invasiva, onde uma central remota realiza o processamento offline dos dados recebidos através do dispositivo móvel.

Palavras-chave: Monitoramento Pervasivo, RNA e Acelerômetro.

Abstract: This paper applies the method Sliding Window (SW) in association with Artificial Neural Network (ANN) for consolidation of data that is generated by an accelerometer, which monitors human movements. From the extracted data was created an input vector with four features to the classifier. Several combinations were made regarding the SW parameters, optimizing the mean hit, which reached 92.63%. It is concluded that a sliding window associated with ANN is appropriate to the detection of the six studied movements. This technique can be widely used in remote monitoring of patients in a less invasive way, while a remote central carries out an offline processing of the received data from a mobile device.

Keywords: Pervasive Monitoring, ANN and Accelerometer.

Introdução

O crescente aumento populacional vem gerando preocupações sobre os serviços de assistência médica oferecidos em todo o mundo. Várias estratégias são desenvolvidas a fim de atender a demanda dos indivíduos, em especial os que necessitam de cuidados contínuos, como por exemplo, os que apresentam doenças crônicas¹. Por muitas vezes a falta de infraestrutura hospitalar, ou mesmo a distância em que se encontram dos centros médicos, limita o acompanhamento necessário para essas pessoas². É importante ressaltar que os idosos também fazem parte desse cenário, pois estão fadados a uma série de problemas físicos e de saúde, como por exemplo fraturas e insuficiência cardiovascular. Uma alternativa viável como solução para esses problemas é o monitoramento remoto do paciente. Essa técnica pode ser aplicada para obtenção de diversos tipos de sinais, como frequência cardíaca e ECG.

Nosso trabalho foca no monitoramento do nível de atividade de pacientes. Neste contexto, pesquisas anteriores, como os de Mathie (2010), Giacomossi (2011), Khan (2010), mostram as vantagens do acelerômetro como sensor de movimentos, sendo ele menos invasivo, de fácil instalação, de pequena dimensão e baixo custo. Anteriormente foi realizado um trabalho sobre detecção de movimentos humanos, tomando como base informações capturadas de um acelerômetro tri-axial integrado a um moderno

telefone móvel (*Smartphone*). Para isso, dados foram coletados previamente e utilizados na classificação dos seguintes movimentos: sentar, levantar, deitar, levantar após deitar, andar e correr. Essa classificação foi possível por meio de uma Rede Neural Artificial (RNA) de três camadas. Nove características foram extraídas desses dados e usadas como entrada para a RNA. As características foram baseadas no cálculo do Desvio Padrão, Média Aritmética e Transformada Rápida de Fourier (FFT). Após esse processo, os dados advindos do acelerômetro foram tratados em tempo real com o intuito de classificar a posição em que o paciente se encontrava em determinado instante⁶. Um segundo trabalho também foi realizado seguindo essa mesma linha de raciocínio, porém focado no consumo energético da bateria do telefone para o processamento das informações obtidas pelo acelerômetro⁷.

Dando continuidade a pesquisa, o presente trabalho concentra-se em dois pontos importantes que foram levantados durante a realização dos trabalhos citados anteriormente. O primeiro ponto refere-se a eficiência energética, pois foi constatado um alto consumo energético devido ao processamento dos dados em tempo real e o segundo ponto destaca o ruído detectado durante a classificação dos dados. O ruído foi responsável por falsos positivos na detecção dos movimentos (entende-se por falso positivo como a detecção errônea de um movimento qualquer). Com o intuito de corrigir os problemas aqui retratados, esse trabalho apresenta como solução para o primeiro ponto o processamento *offline* dos dados. Sendo assim, o aparelho telefônico fica responsável apenas por capturar e guardar em uma base os valores emitidos pelo acelerômetro, para posteriormente serem enviados a uma central onde serão processados e classificados. Como solução para o segundo ponto propõe-se a utilização do método *Sliding Window* (Janela Deslizante) associado ao classificador⁸. Esse método é baseado na segmentação de dados, utilizando para isso duas constantes: largura da janela e tamanho do deslocamento.

Métodos

A seguir serão descritos os métodos utilizados no decorrer desta pesquisa. Inicialmente foi realizado a captura do sinal utilizando o acelerômetro interno ao aparelho celular. Em seguida foram levantadas nove características para a sua caracterização. Após o levantamento das características foi realizado a análise de componente principal, reduzindo de nove para quatro características. Com as características selecionadas foi desenvolvida a arquitetura da RNA, treinamento e validação da mesma utilizando o método de validação cruzada. Com a rede neural validada foi utilizado a técnica de janela deslizante para processamento *offline*.

Módulo Captura

O módulo sensor utilizado para captura dos dados foi o acelerômetro triaxial interno do celular da família Samsung Galaxy. O acelerômetro presente em telefones móveis pode ser usado para detectar a orientação e aceleração, a partir de uma vibração ou movimento.

Os eixos x, y e z, capturados pelo acelerômetro durante a execução de movimentos previamente selecionados, foram armazenados em arquivo texto na memória do celular. Em seguida foram organizados e salvos no computador. O método para captura do conjunto de dados é descrito a seguir:

- O módulo sensor foi preso na cintura de cada voluntário (Figura 1);
- Os voluntários realizaram 6 movimentos: sentar, levantar, deitar, levantar do deitar, andar e correr. Não houve controle sobre amplitude e intensidade na realização dos movimentos;
- Foram coletados 70 padrões de cada movimento, totalizando 420 padrões de sete voluntários com idades entre 20 e 50 anos.

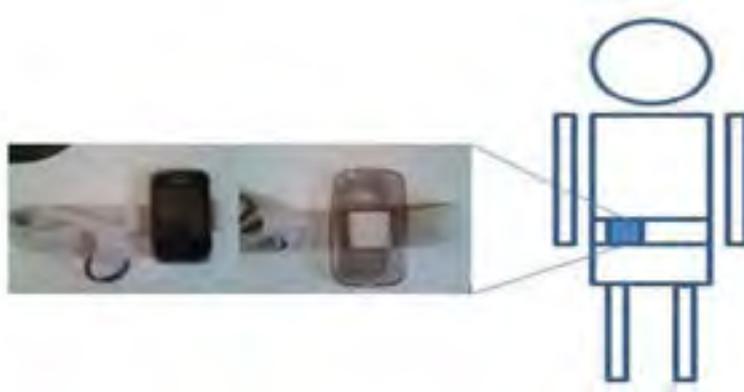


Figura 1: Esboço do módulo sensor preso a cintura do voluntário

Caracterização do Sinal

A etapa seguinte, após a obtenção dos dados, foi a caracterização de cada um dos 420 padrões. Diversas características podem ser utilizadas neste processo. Para facilitar o entendimento, cada padrão pode ser visto como um conjunto de três curvas ao longo do tempo, cada uma delas em um dos eixos cartesianos do acelerômetro. Deste modo, exemplos de características que poderiam ser utilizadas são: maior amplitude da curva, sua integral (área total), distância entre o maior e menor valor (pico e vale)⁴, entre outras.

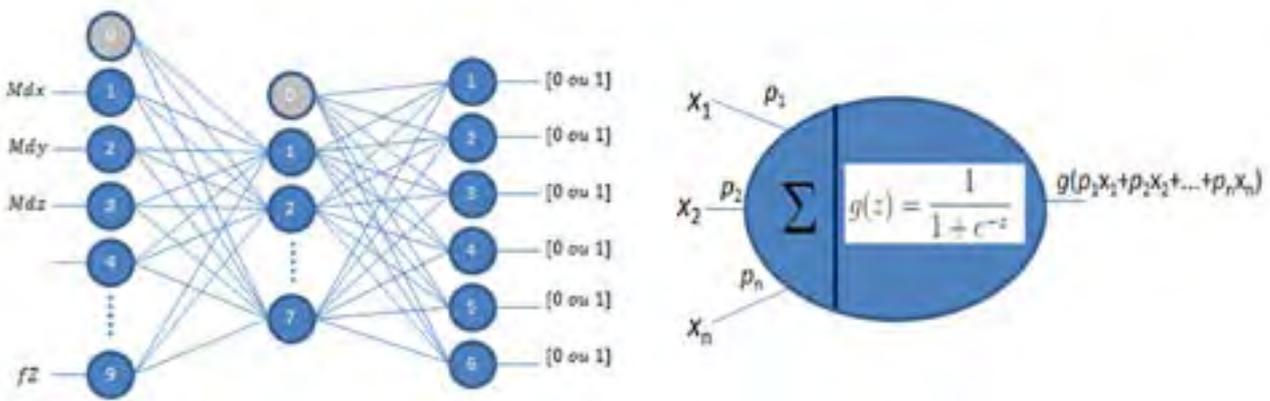


Figura 3: Arquitetura da rede neural e detalhamento do neurônio

O processo de caracterização das curvas foi baseado inicialmente no cálculo de doze características. As três primeiras características, denotadas por Mdx, Mdy e Mdz, referem-se a parâmetros posturais ou valores de offset, onde são calculadas as médias dos vetores⁹ x, y e z.

$$Mdx = 1/n \sum_{i=1}^n x(i) \quad (1)$$

$$Mdy = 1/n \sum_{i=1}^n y(i) \quad (2)$$

$$Mdz = 1/n \sum_{i=1}^n z(i) \quad (3)$$

A quarta, quinta e sexta característica, denotadas por stdX, stdY e stdZ, referem-se ao desvio padrão de cada eixo, onde são calculados o desvio padrão dos vetores¹⁴ x, y e z.

$$stdX = \left(\frac{1}{n-1} \sum_{i=1}^n (x(i) - Mdx)^2 \right)^{\frac{1}{2}} \quad (4)$$

$$stdY = \left(\frac{1}{n-1} \sum_{i=1}^n (y(i) - Mdy)^2 \right)^{\frac{1}{2}} \quad (5)$$

$$stdZ = \left(\frac{1}{n-1} \sum_{i=1}^n (z(i) - Mdz)^2 \right)^{\frac{1}{2}} \quad (6)$$

A sétima, oitava e nona característica, denotadas de fX , fY e fZ , são obtidas através da Transformada Rápida de Fourier (FFT). A partir do cálculo da FFT foi extraída a frequência que apresentou a maior amplitude de cada um dos vetores x , y e z . Essas frequências extraídas representam três características⁶.

As três últimas características, dez, onze e doze, denotadas Ex , Ey , Ez , são obtidas a partir da energia do sinal¹⁴. Todas as doze características foram calculadas utilizando um programa desenvolvido na plataforma Octave¹⁰ e armazenadas em arquivo texto na forma de uma matriz de dados.

Análise de Componentes Principais

Análise de Componentes Principais (PCA) é uma técnica padrão normalmente utilizada para redução de dimensionalidade dos dados em reconhecimento estatístico de padrões e processamento de sinais. Assim para realizarmos a redução da dimensionalidade em determinados dados de entrada, é então obtido os autovalores e autovetores da matriz de correlação do vetor de dados de entrada e então projetamos os dados ortogonalmente sobre um subespaço abrangido pelos autovetores pertencentes aos autovalores dominantes¹². A idéia é reter aqueles que possuem alta variância, enquanto os de menor variância são descartados.

Foi então aplicado PCA nas doze características inicialmente selecionadas e constatado que quatro das doze características concentravam a maior variância, sendo elas Mdx , Mdy , Mdz e $stdX$. Desta forma, foi reduzida a dimensionalidade de doze para quatro características, as quais foram utilizadas como entrada do classificador. A redução de dimensionalidade resulta em uma otimização de custos computacionais, pois precisamos calcular apenas quatro características das doze inicialmente propostas.

Módulo Classificador

O elemento de classificação dos movimentos foi definido como uma Rede Neural Artificial (RNA), a qual implementa o modelo de aprendizagem supervisionada, também conhecida como aprendizagem com o professor¹¹. Figura 2 mostra o diagrama da aprendizagem supervisionada.

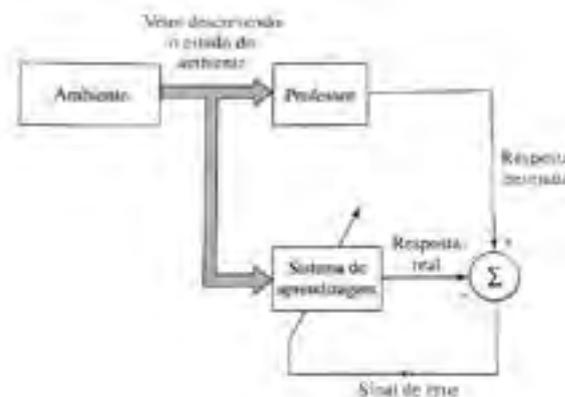
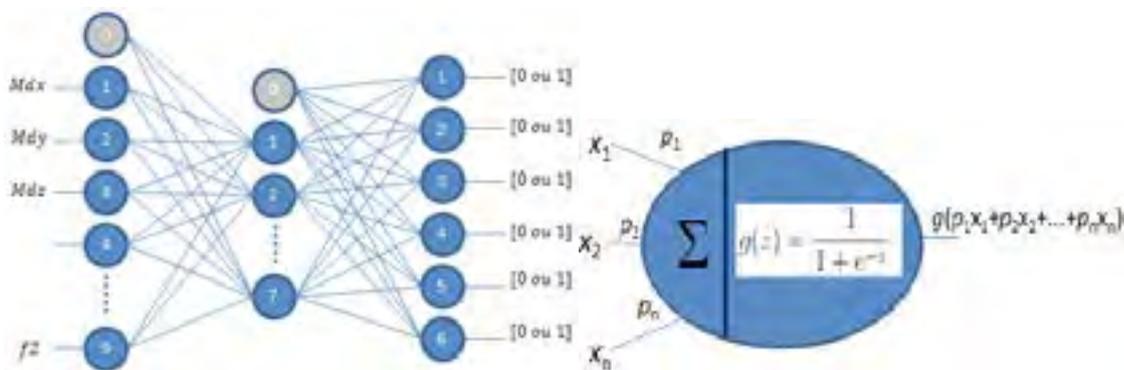


Figura 2: Aprendizagem supervisionada, imagem retirada do livro Haykin S.¹¹

Neste esquema é fornecido ao modelo a resposta ótima esperada e o treinamento do modelo consiste na redução do erro, que é dado pela a diferença da resposta esperada com a resposta real. Este erro é então utilizado para realimentar o sistema de aprendizagem.

A RNA utilizada trata-se de um modelo conhecido como Perceptron de Múltiplas Camadas (MLP-multilayer perceptron). No modelo utilizado, os sinais de entrada são propagados para camadas adiante enquanto o erro calculado é propagado da camada de saída para a camada de entrada, denotada retropropagação¹⁵. Uma RNA é composta por várias unidades de processamento, chamadas neurônios. Estes neurônios são conectados por canais de comunicação que estão associados a determinado peso. Logo, toda conexão entre dois neurônios possui um peso p_i . Os neurônios fazem operações apenas sobre seus dados locais, os quais são recebidos pelas conexões de entrada. O comportamento inteligente de uma Rede Neural Artificial vem das interações entre as unidades de processamento da rede (neurônios). A Figura 3 mostra a arquitetura da RNA utilizada no experimento anterior⁶ com nove características e o detalhamento do neurônio com sua função de ativação, a Sigmóide.



Essa função foi utilizada com a variável z sendo instanciada com o somatório da ponderação dos valores de entrada do neurônio, denotada $g(z)$.

$$g(x) = \frac{1}{1 + e^{-x}} \text{ onde } z = p_1x_1 + p_2x_2 + \dots + p_nx_n \quad (7)$$

A arquitetura utilizada no experimento atual é uma simplificação da Figura 3. Tal arquitetura também possui três camadas. Porém, a primeira delas possui apenas cinco neurônios e é chamada de Camada de Entrada. O primeiro neurônio (bias) tem valor constante e igual a 1, sendo utilizado para aumentar o grau de liberdade e adaptação da rede. Cada um dos demais quatro neurônios corresponde a entrada de uma das quatro características selecionadas.

A camada intermediária é chamada de camada escondida (hidden layer) e não existe uma regra geral para o número de neurônios. Para determinar a quantidade de neurônios desta camada foram feitos testes empíricos avaliando o decréscimo da função de custo, podendo ser visualizada adiante na equação 8, ao longo de 100-1000 iterações, avaliando também o crescimento da média de acertos sobre o conjunto de testes (conjunto que não participa do treinamento). Chegamos ao valor de 11 neurônios, sendo um deles o bias. Um número muito maior que o selecionado poderia levar a RNA a apresentar problemas de overfit, onde a rede teria uma ótima precisão para a identificação dos elementos do conjunto de treinamento, mas não generalizaria bem para novos exemplos do conjunto de teste.

Por fim, a camada de saída possui 6 neurônios e ela funciona como uma porta lógica OU-Exclusivo, de modo que apenas uma das saídas pode ser ativada a cada conjunto de entrada. Cada uma das saídas representa um movimento do usuário, sentar, levantar, levantar do deitar, andar, correr, deitar.

Treinamento da Rede Neural

No treinamento da RNA foram utilizadas as quatro características selecionadas em 420 padrões. O vetor característica, com as características e sua classe ficou representado como:

Mdx	Mdy	Mdz	stdX	C
-----	-----	-----	------	---

Onde C é um número inteiro ($C \in [1..6]$) que representa cada um dos seis movimentos. O erro calculado para a RNA é dado por uma função de custo J, de modo que o treinamento da rede tem o objetivo de reduzir este custo. Então, em cada iteração é calculado o valor de J, que deve ser decrescente até que se estabilize. Foram utilizadas 10000 iterações. A função de custo J utilizada é dada por:

$$J(\theta) = \frac{1}{m} \sum_{i=1}^m \sum_{k=1}^k (-y_k^{(i)} \log((g_{\theta}(x^{(i)}))_k) - (1 - y_k^{(i)}) \log(1 - (g_{\theta}(x^{(i)}))_k)) + FR(\theta) \quad (8)$$

$$FR(\theta) = \frac{\lambda}{2m} [\sum_{j=1}^m \sum_{k=1}^k (\theta_{jk}^{(1)})^2 + \sum_{j=1}^m \sum_{k=1}^k (\theta_{jk}^{(2)})^2] \quad (9)$$

Onde:

- θ representa o conjunto de todos os pesos da rede, sendo dividido em duas matrizes: $\theta(1)$ de dimensionalidade 10x5 e $\theta(2)$ de dimensionalidade 6x11;
- m representa o número de exemplos de treinamento, $m = 420$;
- k representa o número de classes utilizadas na classificação, $k = 6$;
- x é a matriz de características, a qual contém 420 linhas (exemplos de treinamento) e 4 colunas (uma para cada característica);
- y é o vetor de respostas de tamanho 420 que indica qual movimento está representado em cada uma das linhas da matriz de características;
- g_{θ} representa a função de ativação do neurônio (Sigmoide), Eq.(7);
- FR é um fator de regularização utilizado para que a rede tenha um melhor controle sobre a velocidade de aprendizagem;
- λ é a taxa de aprendizado que deve ser um valor entre 0 e 1. Foi utilizado $\lambda = 0,02$.

O treinamento da rede neural artificial se resume a minimizar a função de custo J. Porém, apenas monitorar o decaimento do custo J não é o suficiente para saber se o modelo está classificando corretamente para novos padrões desconhecidos da rede. É também necessário monitorar percentual de acerto da RNA para novos exemplos (conjunto de testes). Pensando nisso e baseado na literatura existente a validação do modelo será descrita no tópico seguinte.

Validações de Aprendizagem

O método utilizado para treinamento e validação dos dados foi a Validação Cruzada (Cross Validation), onde foi explorada uma variante deste método, denotada *K-fold Cross Validation*. A amostra original é subdividida aleatoriamente em K partições de tamanho igual, e a subamostra K_i é retirada do conjunto para ser utilizada na validação, enquanto os demais K-1 (k menos um) subamostras são utilizadas como dados do treino. O processo de validação cruzada é então repetido K vezes, até que cada uma das subamostras K_i seja utilizada apenas uma vez como subamostra de validação. Foi utilizado $K = 10$, ou seja, as amostras foram aleatoriamente divididas em 10 subamostras. O processo é ilustrado na Figura 4.

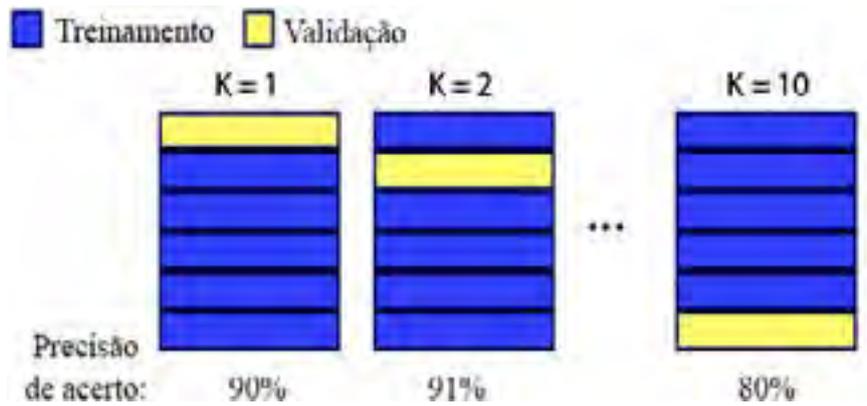


Figura 4: ilustração 10-fold validação cruzada, autoria própria.

Onde a cada K é obtido um percentual de acerto sobre os padrões da partição K . Ao final da validação é realizada então a média das K médias de acerto obtidas, tendo assim um percentual médio de acerto da RNA. Com o classificador treinado e validado, na etapa seguinte iremos utilizar este classificador para classificar novos exemplos de entradas que serão segmentados utilizando a janela deslizante.

Processamento *Offline* do Sinal

Para o processamento offline foi utilizado o método de Janela Deslizante (Sliding Window). Essa técnica é muito usada na extração de subamostras de um conjunto de dados, a partir de um tamanho predefinido para janela e deslocamento. O sinal é analisado de forma que os padrões possam ser identificados e classificados. Este processo é similar ao de detecção de faces humanas em fotos. Ou seja, o padrão de uma face deve ser procurado em todas as partes que compõem a foto, assim como, por exemplo, um padrão de sentar deve ser procurado em todo o sinal gravado. A Figura 5 representa um esboço do método.

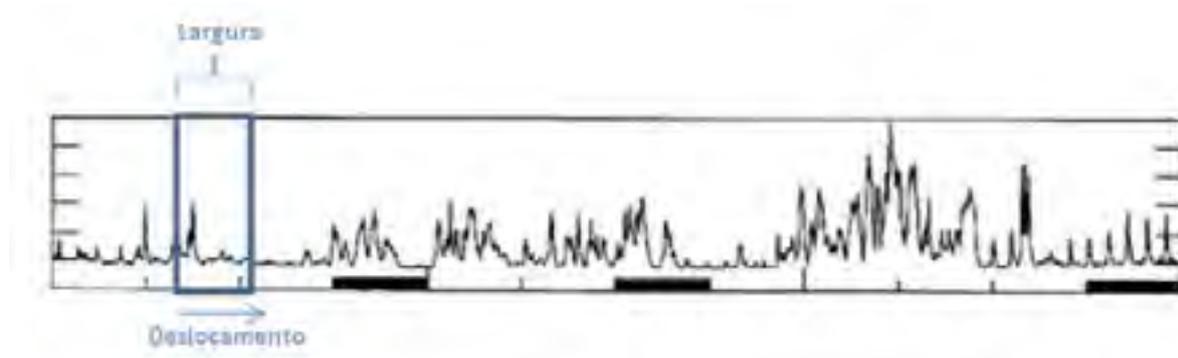


Figura 5 Abstração do método *Sliding Window*

O maior desafio para este método é encontrar o tamanho ideal para o deslocamento e largura da janela. Diversas combinações podem ser utilizadas para estes parâmetros. Não é recomendado que o deslocamento seja igual à largura (Figura 6a) porque alguns movimentos que deveriam ser classificados podem ficar divididos sobre duas janelas. Assim, o classificador pode não identificar tal movimento. Por outro lado, onde o deslocamento é um valor muito inferior à largura da janela, resulta em um longo tempo de processamento (Figura 6b).

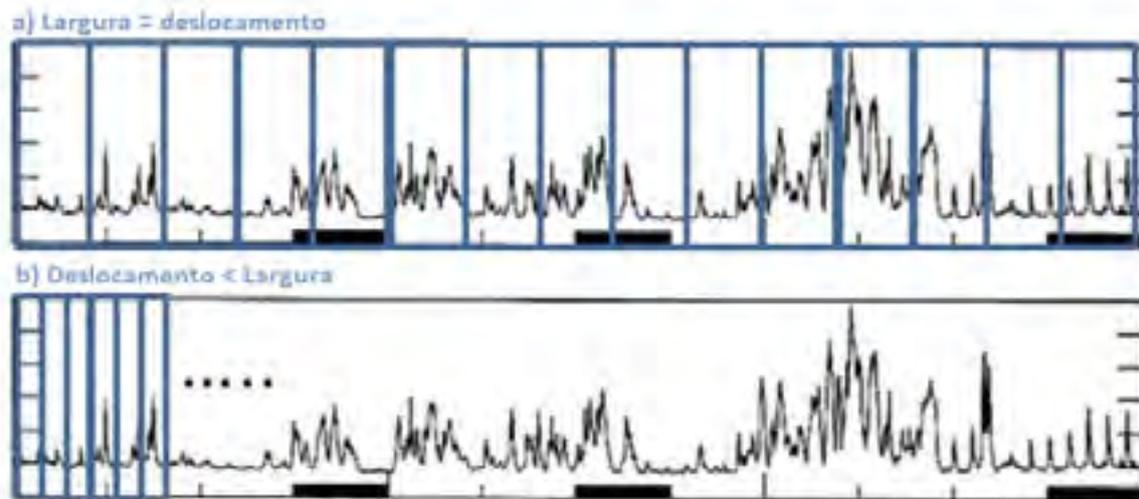


Figura 6: Abstração do método *Sliding Window*. a) quando a largura tem o mesmo valor do deslocamento, b) quando o deslocamento é muito inferior a largura da janela.

O método *sliding Window* foi realizado ao longo das três curvas utilizando o mesmo tamanho de deslocamento e largura da janela para cada um dos eixos, pois as características utilizadas abrangem os três eixos do acelerômetro. O intervalo entre o movimento mais curto e o movimento mais longo foi utilizado como limite inferior e limite superior respectivamente. Foram testadas e avaliadas todas as combinações de deslocamento e largura dentro desse intervalo.

Foram realizadas várias combinação de tamanho de janela e deslocamento. O acerto de novos padrões desconhecidos pela RNA obtido para cada combinação foi armazenado para ser comparado ao final. Os dados em tempo real foram analisados por um sistema composto por uma aplicação mobile na plataforma Android, que ficou responsável em capturar os dados do acelerômetro e uma API encarregada de processar os dados e informar os resultados ao administrador. O método descrito nesse tópico foi implementado na API.

A aplicação mobile em desenvolvimento captura os dados do acelerômetro e encapsula o conjunto de dados em um intervalo definido. Esses dados são enviados ao serviço API REST¹³ através do protocolo HTTP (Hypertext Transfer Protocol) e, caso ocorra erro no envio, a aplicação envia os dados em outro momento. Esse mecanismo impossibilita a perda de informações essenciais para tomada de decisões. Na API, os dados são encaminhados para uma fila de processamento. O serviço identifica o indivíduo, segmenta os dados com a janela deslizante, encaminha para uma rede neural, que classifica os dados e, por fim, entrega as informações para o administrador em diversos formatos como gráfico, fluxos, etc.

Resultados

Visando maximizar o percentual de acerto por movimento detectado, inicialmente foi realizado um experimento com o intuito de encontrar o melhor valor para a largura da janela e tamanho do deslocamento. As combinações consideradas foram variações de 50 a 210 pontos para o deslocamento e 90 a 330 pontos para a largura da janela. Os incrementos foram de 10 e 20 pontos para deslocamento e largura da janela, respectivamente. A Tabela 1 representa alguns percentuais obtidos para o movimento andar.

Tabela 1: Alguns percentuais de acertos do movimento andar, obtidos a cada combinação de largura da janela 'L' e deslocamento 'D'.

<i>D</i> \ <i>L</i>	130	150	170	190	210	230	250	270
50	83,33	83,33	95,83	100	100	100	100	100
60	85	85	95	100	100	100	100	100
70	94,28	91,17	94,11	100	97,05	100	100	100
80	90	86,67	96,67	96,67	100	100	100	100

Após uma análise entre os resultados obtidos para todos os movimentos, a melhor combinação encontrada foi 80 pontos para o deslocamento e 190 pontos para a largura da janela. Essa combinação maximiza o percentual de acerto para os seis movimentos e não somente para o movimento andar. Os seguintes percentuais dos movimentos foram obtidos, Tabela 2.

Tabela 2: Percentual de acerto por movimento e média geral.

Sentar	82,44
Levantar	97,7
Andar	96,67
Correr	100
Deitar	87,32
Levantar do Deitar	91,67
Média	92,63

A Tabela 2 demonstra que a média de acertos para os movimentos foi bastante satisfatório, alcançando um percentual de 92,63%. A solução de falsos positivos foi minimizada com a inserção de um limiar na classificação dos movimentos, onde o movimento era classificado como um dos 6 tipos apenas se o percentual de acerto fosse maior que 54%, caso contrário o classificador retornava zero, que foi interpretado como ruído pelo sistema.

Conclusão

Os resultados demonstraram que o método Janela Deslizante quando associado a um classificador, maximiza o percentual de acerto para cada movimento. É válido salientar que os limites para a largura da janela e tamanho do deslocamento devem ser respeitados, para que os resultados sejam satisfatórios. A aplicação desenvolvida comportou-se de maneira esperada, suavizando os problemas que foram identificados nos trabalhos realizados anteriormente. Além disso mostrou-se eficiente no monitoramento dos indivíduos, podendo facilitar o acompanhamento médico-paciente e possibilitando outros benefícios como: assistência médica remota, avaliação das atividades executadas pelos pacientes etc. Em trabalhos futuros pretende-se estender essa aplicação para outras plataformas como IOS, por exemplo, desenvolver um embarcado e implementar o sistema no mesmo, deixando-o mais transparente no caso de pacientes com dificuldades específicas e também deixando o celular livre para atividades diárias.

Referências

- [1] Copetti, A. (2010) “Monitoramento Inteligente e Sensível ao Contexto na Assistência Domiciliar Telemonitorada”, Tese de Doutorado em Computação, UFF, Niterói, RJ. p.135.
- [2] Nourizadeh S., C. Deroussent, Y. Q. Song, J. P. Thomesse, “Medical and Home automation Sensor Networks for Senior Citizens Telehomecare”, LORIA research laboratory - INPL, France MEDETIC – Colmar, France 2009.
- [3] M. J. Mathie, B.G. Celler, N.H. Lovell.; A.C.F. Coster. (2010) “Classification of basic daily movements using a triaxial accelerometer”, *Medical & Biological Engineering & Computing*, Vol. 42. p. 679-687.
- [4] Giacomossi L. Carlos. (2011) “Método não Invasivo Utilizando o Acelerômetro para Classificar Movimentos Normais e Anormais de Humanos”, Dissertação (Mestrado), Universidade Tecnológica Federal do Paraná, Curitiba, 96f.
- [5] Khan AM, Lee YK, Lee E, Kim TS. (2010) “Accelerometer’s position independent physical activity recognition system for long-term activity monitoring in the elderly”, *Medical Biological Engineering computing*, International Federation for Medical and Biological Engineering, p. 1271-1279. Nov.
- [6] Bruno A Sá; Siebra, C. Extração De Características Das Curvas De Um Acelerômetro Triaxial Para O Processo De Classificação De Movimentos. Anais do XIV Congresso Brasileiro de Informática em Saúde, 2014.
- [7] Bruno A Sá, Siebra, C., GOUVEIA, T. B., SILVA, F. Q., SANTOS, A. L. . A Neural Network Based Application for Remote Monitoring of Human Behaviour. International Conference on Artificial Intelligence, 2015, Roma. Proceedings of the World Symposium on Computer Applications & Research, 2015.
- [8] Oliveira, Rodrigo Fonseca de Santa Cruz, “Detector De Humanos em Imagens Digitais Utilizando Extratores De Características Indicadoras De Movimento”, 2014.
- [9] Winter R. (1999) “Estudo Sobre a Medição do Tremor Humano Utilizando Acelerômetro Piezoelétrico e Processamento de Sinais Digitais”, Dissertação (Mestrado em Engenharia Biomédica), Universidade Tecnológica Federal do Paraná, Curitiba, 151f.
- [10] Eaton John W. (2016) “GNU Octave”, <https://www.gnu.org/software/octave/index.html>.
- [11] Haykin S. (2001). *Redes Neurais: Princípios e Práticas*, 2ª ed. Porto Alegre, p.88-89.
- [12] Haykin S. (2001). *Redes Neurais: Princípios e Práticas*. 2ª ed. Porto Alegre, p.430-442.
- [13] Fielding, Roy Thomas(2000) “Architectural styles and the design of network-based software architectures”, University of California, Irvine.
- [14] Nishkam Ravi , Nikhil Dandekar , Preetham Mysore , Michael L. Littman, Activity recognition from accelerometer data, Proceedings of the 17th conference on Innovative applications of artificial intelligence, p.1541-1546, July 09-13, 2005, Pittsburgh, Pennsylvania.
- [15] Haykin S. (2001). *Redes Neurais: Princípios e Práticas*. 2ª ed. Porto Alegre, p.183-188.
- [3] M. J. Mathie, B.G. Celler, N.H. Lovell.; A.C.F. Coster. (2010) “Classification of basic daily movements using a triaxial accelerometer”, *Medical & Biological Engineering & Computing*, Vol. 42. p. 679-687.
- [4] Giacomossi L. Carlos. (2011) “Método não Invasivo Utilizando o Acelerômetro para Classificar Movimentos Normais e Anormais de Humanos”, Dissertação (Mestrado), Universidade Tecnológica Federal do Paraná, Curitiba, 96f.
- [5] Khan AM, Lee YK, Lee E, Kim TS. (2010) “Accelerometer’s position independent physical activity recognition system for long-term activity monitoring in the elderly”, *Medical Biological Engineering computing*, International Federation for Medical and Biological Engineering, p. 1271-1279. Nov.

- [6] Bruno A Sá; Siebra, C. Extração De Características Das Curvas De Um Acelerômetro Triaxial Para O Processo De Classificação De Movimentos. Anais do XIV Congresso Brasileiro de Informática em Saúde, 2014.
- [7] Bruno A Sá, Siebra, C., GOUVEIA, T. B., SILVA, F. Q., SANTOS, A. L. . A Neural Network Based Application for Remote Monitoring of Human Behaviour. International Conference on Artificial Intelligence, 2015, Roma. Proceedings of the World Symposium on Computer Applications & Research, 2015.
- [8] Oliveira, Rodrigo Fonseca de Santa Cruz, “Detector De Humanos em Imagens Digitais Utilizando Extratores De Características Indicadoras De Movimento”, 2014.
- [9] Winter R. (1999) “Estudo Sobre a Medição do Tremor Humano Utilizando Acelerômetro Piezoelétrico e Processamento de Sinais Digitais”, Dissertação (Mestrado em Engenharia Biomédica), Universidade Tecnológica Federal do Paraná, Curitiba, 151f.
- [10] Eaton John W. (2016) “GNU Octave”, <https://www.gnu.org/software/octave/index.html>.
- [11] Haykin S. (2001). Redes Neurais: Princípios e Práticas, 2ª ed. Porto Alegre, p.88-89.
- [12] Haykin S. (2001). Redes Neurais: Princípios e Práticas. 2ª ed. Porto Alegre, p.430-442.
- [13] Fielding, Roy Thomas(2000) “Architectural styles and the design of network-based software architectures”, University of California, Irvine.
- [14] Nishkam Ravi , Nikhil Dandekar , Preetham Mysore , Michael L. Littman, Activity recognition from accelerometer data, Proceedings of the 17th conference on Innovative applications of artificial intelligence, p.1541-1546, July 09-13, 2005, Pittsburgh, Pennsylvania.
- [15] Haykin S. (2001). Redes Neurais: Princípios e Práticas. 2ª ed. Porto Alegre, p.183-188.

