

Uso de técnicas de subtração de quadros para segmentação de sinais em eletrocardiogramas

Alexei Manso Correa Machado¹, Paulo Junio Reis Rodrigues¹

¹Instituto de Ciências Exatas e Informática
Pontifícia Universidade Católica de Minas Gerais – Belo Horizonte, MG – Brasil

alexeimcmachado@gmail.com, pjrrodrigues@sga.pucminas.br

1. Introdução

O Eletrocardiograma (ECG) tem como objetivo apresentar graficamente as atividades do coração, medindo as variações do potencial elétrico do músculo cardíaco em atividade [Davey and Sharman 2018]. Os dados adquiridos nestas amostragem são reproduzidos em um monitor, para que o médico possa analisar-los e verificar como está o funcionamento do coração.

Os sinais adquiridos pelo equipamento podem identificar vários tipos de problemas, como por exemplo a arritmia cardíaca, onde há uma alteração do batimento cardíaco do paciente, podendo ser irregular em um certo tempo [Neto 2012]. Estes traços são analisados por médicos ou máquinas, porém para isso necessita-se algumas vezes uma grande quantidade de dados para identificar as anomalias.

Uma maneira de guardar todos os sinais seria utilizar equipamentos novos que podem preservar várias horas de captura, porém esses aparatos são de alto custo deixando assim inviável o seu uso em alguns locais. É preferível utilizar uma maneira mais econômica e ao mesmo tempo eficaz para extrair esses traços do aparelho. Geralmente os eletrocardiogramas utilizam um monitor para uma fácil visualização em tempo real dos sinais. Utilizar técnicas de visão computacional juntamente com uma câmera é um modo de obter e guardar os sinais produzidos pelo o monitor, já que é desconfortável uma pessoa ficar olhando para o dispositivo para registrar os dados obtidos durante os exames.

O objetivo principal deste trabalho é criar um algoritmo robusto e altamente preciso para analisar os vídeos gerados pelo eletrocardiograma, assim extrair os sinais cardíacos do paciente e guardá-los em forma de sinal. Esses processos consistem em três etapas (Figura 1): (i) aplicação de um pré processamento em cada quadro, logo depois (ii) segmentar o sinal cardíaco utilizando a técnica de subtração de quadros, e por fim (iii) tem-se a utilização de um algoritmo de rastreamento para encontrar todos os pontos que compõem o sinal, para que no final possa-se construir uma réplica do sinal adquirido.

2. Revisão bibliográfica

A subtração de quadros tem como principal objetivo detectar objetos em movimento, capturados por uma câmera estática. Ao decorrer dos anos, vários algoritmos foram criados com a intenção de implementar esta técnica, cada um com o seu objetivo específico. Em [Piccardi 2004], foram analisados sete métodos, com o intuito de demonstrar todos os benefícios e limitações de cada algoritmo. Durante a análise dos métodos, utilizou-se como comparação, a sua velocidade, o seu uso de memória e a sua acurácia. Um dos

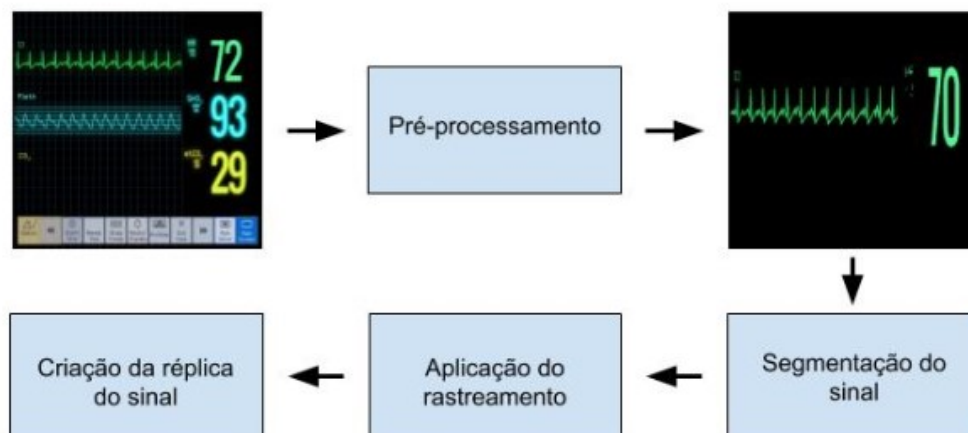


Figura 1.

métodos mais simples analisados é o *Running Gaussian Average*, proposto por Wren em [Wren et al. 1997], que tem como objetivo analisar cada plano de fundo independentemente em cada localização de pixel, para isso o modelo utilizou como ajuda a distribuição normal Gaussiana. Contudo este algoritmo demonstrou ser rápido e eficiente, entretanto, demonstrou ter uma baixa acurácia nos seus resultados. Outro método analisado foi o *Mixture of Gaussians*, proposto por Stauffer e Grimson em [Stauffer and Grimson 2007], que tem como objetivo analisar quantas vezes um objeto apareceu em determinado pixel, nesse técnica supõe-se que os valores de intensidade de cada pixel no vídeo possam ser modelados usando um modelo de mistura Gaussiano, definindo assim qual pixel está representando um objeto. Contudo este método demonstrou ter uma acurácia muito alta, porém a sua eficiência e velocidade não foram considerados ótimas, comparadas com o primeiro método abordado acima, todos os resultados estão no quadro da Figura 2.

Atualmente os problemas encontrados nesta área estão mais complexos e os vídeos encontrados também dependem de uma técnica mais aprimorada para encontrar uma solução mais precisa. Como por exemplo em [Qiu et al. 2018], foi criado um algoritmo de segmentação em vídeo utilizando redes convolucionais, o algoritmo possui o nome de *deep spatio-temporal fully convolutional networks* e utiliza duas redes totalmente convolucionais em sua solução, uma focada na parte 2D do vídeo, que é específica em explorar a diferença de cores dos *pixels* na cena e já a outra rede é focada na parte 3D, que possibilita uma exploração dos níveis dos *voxels* durante o decorrer do vídeo. Para testar o método criado, foi utilizado duas base de dados, conhecidas como *A2D* e *CamVid*, por fim os testes demonstraram que os resultados encontrados são melhores do que os encontrados no estado da arte.

Outro algoritmo que consegue inovar nos dias atuais se encontra em [Sun and Lu 2017], nesse método é levado em consideração a forma e as cores do objeto pesquisado, para isso é usado um algoritmo para encontrar as bordas do objeto, para que depois junto com as cores encontradas se possa segmentar o objeto em cena. Os resultados encontrados foram bastante satisfatórios, considerando que o método criado apresentou um melhor resultado do que o Rotobrush, este que pertence a empresa Adobe

com o objetivo de segmentar objetos em imagens.

Logo após o uso de um método para a realização da segmentação do objeto em vídeo, necessita-se de um rastreador para seguir e guardar os dados necessários para que a resolução do problema seja encontrada, em [Chakraborty and Meher 2013], se tem como objetivo criar um algoritmo que utiliza conceitos de subtração de quadros e rastreamento de objetos para recuperar a trajetória de um bola de basquete, para isso foi-se aplicado uma segmentação em vídeo para encontrar a bola em cena e logo depois o rastreador foi feito utilizando a forma da bola como base, para que depois nos quadros seguintes possa-se seguir a bola e guarda a sua trajetória traçada. Outros artigo como [Draschkowitz et al. 2015] e [Yu et al. 2006], também utilizam dessas técnicas para resolverem seus problemas, onde se tinha que rastrear e logo depois prever o percurso de uma bolinha de Tênis de mesa, já o outro algoritmo fez um rastreador para perseguir uma bola de futebol. Entretanto um dos principais problemas encontrados nessas soluções, são que os objetos rastreados se movem muito rapidamente durante a cena, prejudicando assim o resultado do rastreador.

| Method | Spd | Mem | Acc |
|---------------------------------------|----------|----------|-------|
| Running Gaussian average [1,2] | I | I | L/M |
| Temporal median filter [3,4] | n_s | n_s | L/M |
| Mixture of Gaussians [5] | m | m | F |
| Kernel density estimation (KDE) [7] | n | n | F |
| Sequential KD approximation [11] | $m + I$ | m | M/H |
| Cooccurrence of image variations [12] | $8n/N^2$ | nK/N^2 | M |
| Eigenbackgrounds [13] | M | n | M |

Figura 2. Quadro retirado do artigo [Piccardi 2004], demonstrando os resultados obtidos nas comparações dos métodos testados.

3. Metodologia

Na Figura 3 pode ser observado o fluxo que a metodologia seguirá, primeiramente será feito uma aquisição de todos os dados necessários para a criação do algoritmo, e servirão de testes no futuro, logo depois será feita uma segmentação do sinal cardíaco do paciente, utilizando técnicas de subtração de quadros oferecidas pela biblioteca *OPENCV*. Depois deste processo, será criado um rastreador para seguir e guarda os pontos gerados pelo sinal cardíaco segmentado e por fim será a etapa de reconstrução do sinal, utilizando todos os pontos coletados pelo o rastreador.

3.1. Aquisição dos dados

A amostragem dos dados foi adquiridas no site do Youtube. Os materiais adquiridos possuem grandes semelhanças, os sinais estão sempre localizados na parte superior do monitor e as cores do sinal cardíaco estão sempre no tom de verde, e portanto o trabalho será totalmente desenvolvido baseado em eletrocardiogramas específicos, não levando em

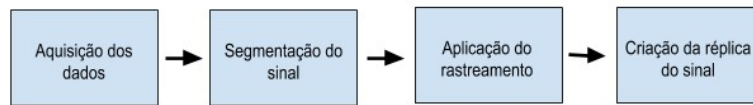


Figura 3. Fluxo da metodologia.

consideração toda a gama de equipamentos do mercado. O número de vídeos encontrados é baixo, levando em consideração de que os dados pessoais de um paciente não são disponibilizados de graça legalmente, a maioria das matérias são pagas e possuem restrições de uso. Portanto ao todo foram utilizados dois vídeos para a criação desse algoritmo.

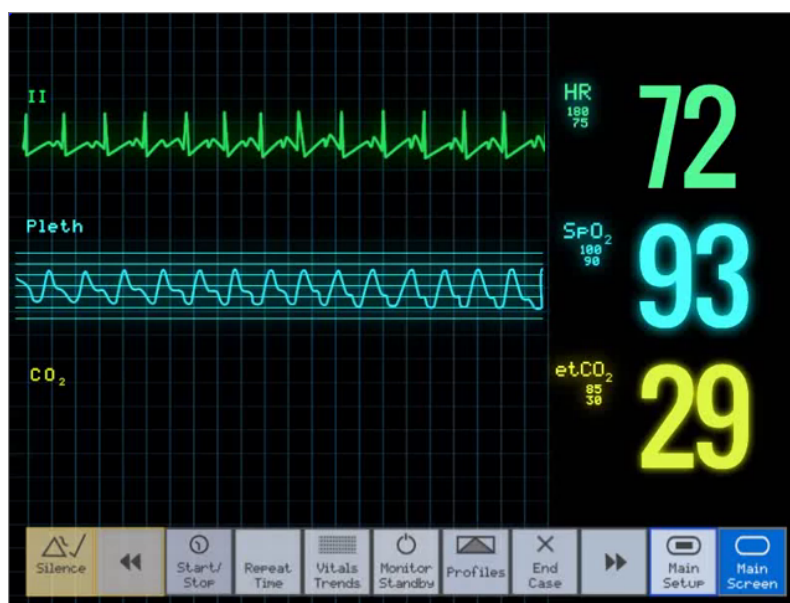


Figura 4. Figura de um quadro do vídeo capturado, encontrado no link https://www.youtube.com/watch?v=QwWlWMV_RyI.

3.2. Segmentação do sinal

Durante a aquisição dos dados, foi observado que há vários pré-processamentos que precisam ser feitos, para que os resultados fiquem melhores no final.

Logo depois de adquirir os dados, precisa-se tratá-los frame a frame, para depois possa-se utilizar a subtração de quadros para rastrear o traço cardíaco. O objetivo desta parte é segmentar somente o sinal, que se encontra na cor esverdeada e na parte superior do monitor. No começo do algoritmo, é perguntado ao usuário o local que o sinal está, e logo depois é feita uma segmentação por cor, usando limites para a cor verde.

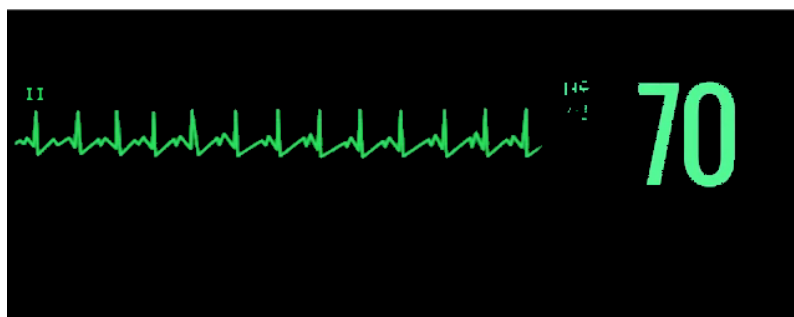


Figura 5. Figura do monitor com apenas a cor verde destacada.

3.3. Aplicação do rastreamento

Foi escolhido esses parâmetros pois, necessita-se encontrar somente o novos sinais com precisão, traços antigos não são considerados pois podem atrapalhar nos resultados. Logo após isso, é feito um novo quadro onde possui somente o sinal da esquerda sendo construído, e o sinal da direita sendo destruído, com isso podia-se começar a utilizar o rastreamento dos pontos, para logo depois reconstruir o traço.



Figura 6. Figura do sinal cardíaco logo após a aplicação da subtração de quadros.

3.4. Criação da replica do sinal

Logo depois de adquirir o novo vídeo gerado pela subtração de quadros, necessita-se criar um *track* para seguir e guardar o local do sinal mais a esquerda, gerado no monitor. Para isso, se utilizou o método para encontrar contornos nos objetos em cada quadro, ao executar o método, ele retorna todos os contornos na imagem. Cada limite obtido tem-se sua área, nisso como deve-se pegar somente um ponto, o seu tamanho não será muito grande, por isso no algoritmo só é considerado os contornos com uma área pequena. Com isso cada novo quadro gera pontos que podem ser sinais novos ou antigos, para considerar somente os novos, é preciso considerar somente os traços a esquerda.

Durante a aquisição dos pontos, há um momento que o sinal pode ser recommçado, logo depois utilizando o mesmo local é criado outro traço, com isso a cada recommço o algoritmo gera uma nova imagem, para não sobrescrever os sinais já adquiridos. Logo depois de todo esses processos, é feito uma ordenação de todos os pontos, para que na hora de criar linhas entre dois deles não haja falhas na criação do sinal adquirido, e logo

após é criado e guardado várias imagens para demonstrar todo o sinal adquirido no vídeo base.

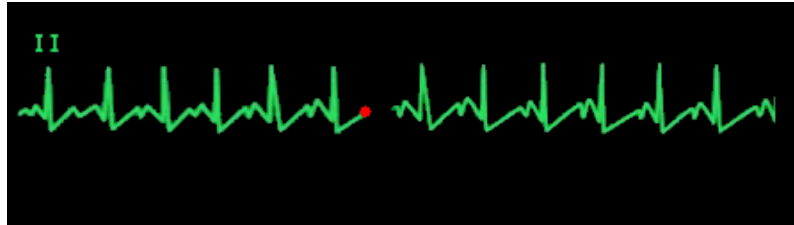


Figura 7. Figura do monitor sendo aplicado o algoritmo do rastreamento.

4. Resultados

Logo depois da aplicação do algoritmo nos vídeos, se obtém os resultados através de várias imagens que mostram o sinal recriado. Para verificar a veracidade dos resultados, se optou analisar da seguinte maneira: Primeiramente é pego a quantidade de pixels verdes do sinal original, e depois o sinal gerado em vermelho é colocado sobre o sinal original para verificar qual é a porcentagem de acerto do algoritmo. Isso será feito para os três algoritmos criados, GMM, MOG e MOG2.



Figura 8. Resultado do vídeo um, primeiro sinal criado.



Figura 9. Resultado do vídeo um, segundo sinal criado.

Referências

- Chakraborty, B. and Meher, S. (2013). A real-time trajectory-based ball detection-and-tracking framework for basketball video. *Journal of Optics*, 42:156–170.
- Davey, P. and Sharman, D. (2018). The electrocardiogram. *Medicine*, 46(8):443 – 452.

-
- Draschkowitz, L., Draschkowitz, C., and Hlavacs, H. (2015). Using video analysis and machine learning for predicting shot success in table tennis. *EAI Endorsed Transactions on Creative Technologies*, 2(5).
- Neto, O. S. (2012). Arritmia cardíaca: fibrilação atrial. *Revista da Faculdade de Ciências Médicas de Sorocaba*, 14(3):87–94.
- Piccardi, M. (2004). Background subtraction techniques: a review. In *2004 IEEE International Conference on Systems, Man and Cybernetics (IEEE Cat. No.04CH37583)*, volume 4, pages 3099–3104 vol.4.
- Qiu, Z., Yao, T., and Mei, T. (2018). Learning deep spatio-temporal dependence for semantic video segmentation. *IEEE Transactions on Multimedia*, 20(4):939–949.
- Stauffer, C. and Grimson, W. (2007). Adaptive background mixture models for real-time tracking. *Proceedings of IEEE Conf. Computer Vision Patt. Recog*, vol. 2, 2.
- Sun, C. and Lu, H. (2017). Interactive video segmentation via local appearance model. *IEEE Transactions on Circuits and Systems for Video Technology*, 27(7):1491–1501.
- Wren, C. R., Azarbayejani, A., Darrell, T., and Pentland, A. P. (1997). Pfnder: real-time tracking of the human body. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 19(7):780–785.
- Yu, X., Leong, H. W., Xu, C., and Tian, Q. (2006). Trajectory-based ball detection and tracking in broadcast soccer video. *IEEE Transactions on Multimedia*, 8(6):1164–1178.