

# Estimação de Pose para a Fisioterapia

Gabriel Luís de Araújo Freitas (15/0008872)

Faculdade do Gama - FGA  
Universidade de Brasília - UnB  
Brasília, Brasil  
Email: gabrieluis10@gmail.com

Victor Oliveira Corrieri de Macedo (14/0164961)

Faculdade do Gama - FGA  
Universidade de Brasília - UnB  
Brasília, Brasil  
Email: victormacedo10@yahoo.com.br

**Resumo**—Clínicas de fisioterapia e reabilitação têm dificuldades de custo e praticidade em acompanhamentos contínuos e tendem a estender o tratamento para fora do consultório. A realização de exercícios em casa limita a supervisão e correção dos exercícios, prejudicando a recuperação do paciente. Este trabalho trata da proposta de um aparelho de auxílio em exercícios de fisioterapia, por meio de um algoritmo de estimação de pose embarcado em um *system on chip* (SoC). O algoritmo, baseado em redes neurais convolucionais, usa um vídeo de referência para dar um *feedback* visual em tempo real ao usuário. Assim, permitindo a execução de um exercício mais próximo ao proposto, além do acompanhamento à distância pelo fisioterapeuta.

## I. INTRODUÇÃO

Em vista de uma tendência de automatização e personalização, sistemas assistivos vêm ganhando bastante espaço nas áreas da saúde. A geração de informações e registros temporais facilitam e embasam tomadas de decisão e diagnósticos médicos. Tal abordagem traz para áreas, como a fisioterapia, uma ferramenta para melhor acompanhamento do paciente, especialmente para tratamentos em casa.

Recentes avanços tecnológicos em visão computacional possibilitaram o monitoramento de atividades físicas com baixo custo e alto desempenho. A área de captura de movimento teve um crescente interesse de estudo, sobretudo devido a popularização de produtos como o Microsoft Kinect. Trabalhos que obtiveram resultados no estado da arte em estimação de pose disponibilizaram seus algoritmos para uso público [1], impulsionando as mais diversas aplicações [2]. Além disso, grandes bases de dados foram criadas permitindo testes e otimizações de novos algoritmos [3].

Diversos trabalhos foram desenvolvidos no intuito de monitorar e auxiliar o tratamento fisioterápico doméstico [4]. Porém a maioria faz uso de marcadores ou sensores acoplados ao paciente, o que dificulta a praticidade e aceitação do dispositivo. Outros artigos propuseram o uso do Microsoft Kinect para atacar o problema [5], contudo, apresentam um maior custo de hardware e se restringem ao grupo de usuários que já possuem a tecnologia. Neste sentido, a implementação de um sistema embarcado de baixo custo com uma câmera simples acoplada, mostra-se como abordagem viável para o problema.

## II. JUSTIFICATIVA E OBJETIVOS

Muitos dos tratamentos de fisioterapia necessitam de que o paciente repita em casa os exercícios realizados em consul-

tório. Porém, sem o acompanhamento adequado o tratamento pode perder efetividade e se estender por mais tempo. Logo, ferramentas que ajudem a realização do movimento e que possibilitem seu monitoramento são úteis no contexto da fisioterapia.

Dada a oportunidade de desenvolvimento de tecnologias que auxiliem a execução de exercícios fisioterápicos, este projeto visa apresentar uma solução, utilizando como plataforma a *Raspberry Pi 3* e algoritmos de estimação de pose, para uso em contexto doméstico. O produto, programável pelo fisioterapeuta, fornece ao paciente um *feedback* visual para a reprodução dos movimentos corretos de maneira mais adequada.

Juntamente com o auxílio na execução dos exercícios, são produzidos dados quantitativos referentes à qualidade dos movimentos que, fornecidos ao fisioterapeuta, são utilizados para o aprimoramento do tratamento. O equipamento, portanto, é uma ferramenta para facilitar a realização dos movimentos e possibilitar um acompanhamento mais efetivo dos pacientes.

## III. FUNCIONAMENTO PROPOSTO

O sistema funciona a partir de um vídeo de referência com o movimento certo realizado pelo fisioterapeuta. Uma vez filmado, o vídeo é previamente processado para extrair os pontos de articulação e determinar suas conexões, de forma a criar uma representação em *stickman*. O paciente então posiciona o dispositivo com a câmera apontada para o plano de maior informação do exercício, conforme determinado pelo fisioterapeuta, e seu vídeo é apresentado em um monitor de sua escolha. Para adequar o *stickman* ao paciente, uma etapa de calibração será necessária. Nesta etapa, as proporções do usuário são extraídas pelo mesmo algoritmo de estimação de pose e aplica-se uma redimensionalização do *stickman* salvo. Durante o exercício, o *stickman* é sobreposto no vídeo e o paciente tem como objetivo imitar seu movimento.

Deseja-se que os exercícios a serem feitos possam ser armazenados no sistema pelo fisioterapeuta. O paciente ao ligar o equipamento em uma televisão poderá escolher, por meio de um controle *bluetooth*, qual exercício realizar, terá acesso também às mensagens enviadas pelo fisioterapeuta e a um menu de calibração. Uma vez escolhido o exercício, ele será iniciado após um segundo comando dado pelo controle, para que o paciente posicione-se corretamente. Por fim, poderão

ser feitas pausas no movimento e o usuário poderá retornar ao menu de seleção.

#### IV. ARQUITETURA IMPLEMENTADA

Nesta seção, os módulos gerais implementados para o projeto foram definidos e seus funcionamentos foram detalhados.

##### A. Extração do movimento do vídeo

O algoritmo de detecção de pose implementado é baseado em uma rede neural convolutiva (CNN), treinada em um amplo banco de imagens para captura de movimentos [3]. A arquitetura da rede foi desenvolvida em [1], sendo constituída por duas partes, detalhadas na figura 1:

- (i) **Classificação de regiões:** Estima mapas de probabilidade que indicam as regiões de maior probabilidade da localização de cada ponto de articulação. Utiliza-se então uma política *greedy* para extrair um único ponto para cada mapa. Essa parte da rede é composta de uma CNN com suas 10 camadas iniciais adquiridas da arquitetura VGGNet, por transferência de aprendizado, e múltiplos estágios ramificados um para cada parte do corpo.
- (ii) **Afinidade entre partes:** Após a identificação de pontos individuais, a próxima etapa necessária é definir a conexão entre eles. Para isso, são gerados mapas de confiança e afinidade com o intuito de estimar regiões de probabilidade para a ligação entre pontos adjacentes. Novamente, uma inferência *greedy* é aplicada, definindo conexões e classificando cada região. A saída produzida pela rede é um vetor de 18 componentes, compondo as articulações principais e os pontos faciais, porém, apenas 13 componentes foram utilizados para o projeto, como ilustrado na figura 2



Figura 2. Pontos de articulação resultantes da estimação de pose.

A etapa de aquisição dos pontos no vídeo de referência foi realizada externamente em uma CPU, devido as limitações de poder computacional da placa utilizada. Para isso, o algoritmo de estimação de pose, descrito na seção anterior, foi executado *frame a frame* no vídeo e os pontos obtidos foram salvos em um arquivo texto, que foi armazenado na *Raspberry*.

##### B. Calibração do stickman

As proporções corporais variam de acordo com o indivíduo, este fato deve ser considerado no desenvolvimento do projeto. Para que o paciente possa seguir o movimento desejado de maneira mais natural é importante que as dimensões do *stickman* sejam coerentes com as suas próprias. Assim o *stickman* que executa o movimento de referência deve ser redimensionado para que as distâncias entre as articulações estejam próximas das do paciente. O módulo chamado de calibração é responsável por esta tarefa.

A etapa de calibração tem como entrada dois conjuntos de pontos, denominados “professor” e “aluno”. Cada um deles é a saída do algoritmo de detecção de pose para uma imagem em que a pessoa que executa o movimento está em uma posição padrão que favorece a visualização das articulações a serem vistas pela câmera durante a execução do exercício. O conjunto “professor” refere-se à pessoa que executou o movimento gravado, que será imitado pelo paciente, correspondente ao conjunto “aluno”.

Em cada um dos *frames* do vídeo deve-se manter o ângulo formado entre a horizontal e a reta que une dois pontos de articulação do conjunto “professor”, mas a distância entre os mesmos dois pontos deve ser a observada nos pontos correspondentes no conjunto “aluno”. Este procedimento é feito percorrendo todos os pares conectados da imagem de forma que cada ponto é movimentado uma única vez. As novas coordenadas do ponto  $[x_m \ y_m]^T$  movimentado a cada iteração é dado por:

$$[x_m \ y_m]^T = [x_p \ y_p]^T + d \cdot [\cos(\theta) \ \sin(\theta)]^T + \mu \quad (1)$$

Em que  $d$  é a distância vista no conjunto “aluno”,  $\theta$  o ângulo visto no conjunto “professor”,  $[x_p \ y_p]^T$  são as coordenadas do ponto que se mantém fixo na iteração e  $\mu$  é um vetor com duas posições, atualizado a cada nova iteração, que é utilizado para fazer correções relativas ao movimento global do *stickman*.

A figura 3(a) apresenta o resultado desse algoritmo em que o *stickman* em amarelo é o resultado do algoritmo propriamente dito, o *stickman* em preto é a entrada do algoritmo como “aluno” e os pontos vistos na figura 3(b) também são dados como entradas, mas para o conjunto “professor”.

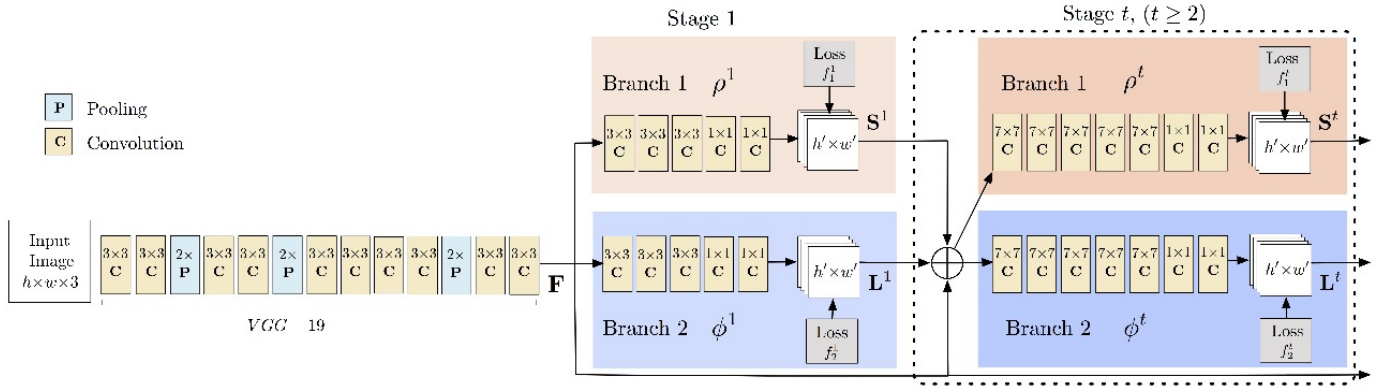


Figura 1. Arquitetura CNN para detecção de pose. Fonte: [1].



(a) Saída do algoritmo e pontos do "aluno" (b) Pontos do "professor"

Figura 3. Imagens de teste para o algoritmo de calibração

### C. Calibração do vídeo

Uma vez determinado o processo de reescalamiento para um único frame, pode-se se estender o método para um vídeo completo. Para isso, a mesma ideia de divisão dos *stickman* entre professor e aluno foi aproveitada. Porém, um único frame do aluno deve ser associado à cada frame do professor. Nesse caso, uma aplicação direta do método pode gerar problemas se o movimento sair de um plano de execução. Isso porque algumas distância do *stickman* do professor podem estar reduzidas, em relação ao frame de referência do professor, e o algoritmo tentaria manter as mesmas distâncias do frame de referência do aluno, o que não provocaria essa redução proporcional necessária.

Para resolver esse problema, o método descrito na subseção anterior foi alterado da seguinte forma:

- (i) Em uma primeira etapa são extraídas as distância dos pontos estimados para um único frame de referência para o professor e outro para o aluno.
- (ii) Depois, as distâncias do aluno foram divididas pelas distâncias do professor, um a um, para armazenar um fator de proporção entre eles.

- (iii) Por último, o algoritmo descrito na subseção anterior foi executado para cada frame do vídeo do professor. Porém, o que seria usado como a distância a ser mantida do aluno, vira a distância do professor naquele frame multiplicada pelo fator de proporção obtido para essa respectiva distância.

Esse algoritmo adaptado foi implementado e funcionou da forma prevista, proporcionando o reescalamiento do vídeo do professor com referência ao aluno que irá realiza-lo, como desejado.

### D. Controle bluetooth

O funcionamento do controle é de grande importância, pois ele é a maneira com que o paciente consegue interagir com o sistema. É o por meio dele que será possível a navegação entre as funcionalidades, tais como o menu de calibração e a escolha do exercício a ser realizado.

O controle é implementado em um microcontrolador e se comunica via *bluetooth* com a *Raspberry*. Existem 7 diferentes botões, denominados: (i) cima, (ii) baixo, (iii) esquerda, (iv) direita, (v) menos, (vi) mais e (vii) confirma. Cada um deles, ao ser pressionado, envia de maneira serial um valor inteiro (entre 1 e 7) ao sistema.

O *software* embarcado, recebe o sinal do controle em um processo. Em seguida, salva o valor lido em um arquivo de texto, que poderá ser acessado pelos demais processos em execução. Ao escrever no arquivo de texto, é enviado um *signal* que interromperá a execução dos processos que devem ser afetados, para que o arquivo de texto seja lido e tomada a ação desejada. Ao concluir a tarefa solicitada pelo botão, o processo que foi afetado escreve o valor "0" no arquivo, indicando que realizou a atividade e uma nova pode ser solicitada.

## V. OBJETIVOS FUTUROS

### A. Interface

Apesar de todas as funcionalidades desejadas já terem sido implementadas, ainda é necessário realizar sua integração. Isto é necessário para que se desenvolva um produto amigável ao usuário. Desta forma, é preciso desenvolver uma interface

intuitiva que possibilite a navegação entre as funções do sistema.

Ao ligar o aparelho em uma televisão, o paciente terá duas opções disponíveis: (i) modo de calibração, (ii) menu de escolha de exercícios. Ao entrar em (i), a câmera será ligada, por meio do controle será possível tirar uma fotografia que será utilizada para, por meio do algoritmo de estimação de pose, para determinar as proporções corporais do usuário, haverá ainda uma opção para ajuste manual dos pontos identificados.

Caso seja escolhido o menu de exercícios, aparecerão na tela todos os exercícios indicados pelo fisioterapeuta. Quando escolhido um deles, a câmera será ligada e o *stickman* realizando o exercício aparecerá na tela. Com o controle *bluetooth* será possível posicioná-lo da melhor maneira e iniciar ou pausar a execução. Caso seja pausado, haverá duas opções, ou o paciente retorna ao menu inicial ou continua o exercício atual.

### B. Estrutura

A parte de estrutura do produto pode ser dividida em três partes: (i) estrutura para a Raspberry Pi; (ii) estrutura para a câmera; (iii) estrutura para o controle *bluetooth*. Essas estruturas serão efetivamente produzidas em uma impressora 3D para o próximo ponto de controle. Para auxiliar esse desenvolvimento, os desenhos técnicos CAD já foram projetados, por meio do *tinkercad*, para a estrutura da placa e da câmera. Esses CADs estão apresentados na figura 4, com o intuito de ilustrar a ideia final pensada para a estrutura.

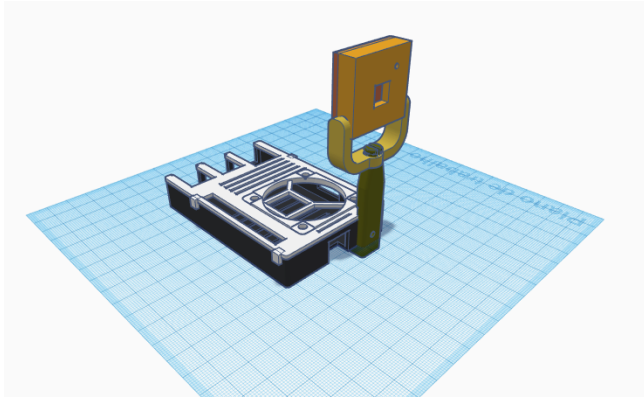


Figura 4. Estrutura projetada para impressão 3D.

## VI. CONCLUSÃO

Um equipamento como o descrito neste trabalho possibilitaria à fisioterapeutas um acompanhamento mais efetivo ao tratamento de seus pacientes, por possuir a sua disposição uma série de dados sobre como os movimentos foram executados. O paciente terá maior segurança ao fazer os exercícios em casa sem o acompanhamento e eventualmente conseguirá corrigir seus movimentos. Desta forma, os tratamentos podem apresentar os resultados esperados mais rapidamente. Além disso, o baixo custo torna o produto acessível e atrativo frente às outras alternativas ao problema.

## REFERÊNCIAS

- [1] Zhe Cao, Tomas Simon, Shih-En Wei, and Yaser Sheikh, "Realtime multi-person 2d pose estimation using part affinity fields," in *CVPR*, 2017.
- [2] Caroline Chan, Shiry Ginosar, Tinghui Zhou, and Alexei A Efros, "Everybody dance now," *arXiv preprint arXiv:1808.07371*, 2018.
- [3] Hanbyul Joo, Tomas Simon, Xulong Li, Hao Liu, Lei Tan, Lin Gui, Sean Banerjee, Timothy Scott Godisart, Bart Nabbe, Iain Matthews, Takeo Kanade, Shohei Nobuhara, and Yaser Sheikh, "Panoptic studio: A massively multiview system for social interaction capture," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2017.
- [4] Antonio Bo, Mitsuhiro Hayashibe, and Philippe Poignet, "Joint angle estimation in rehabilitation with inertial sensors and its integration with kinect," in *EMBC: Engineering in Medicine and Biology Conference*, 2011, pp. 3479–3483.
- [5] Stepan Obdrzalek, Gregorij Kurillo, Ferda Ofli, Ruzena Bajcsy, Edmund Seto, Holly Jimison, Michael Pavel, et al., "Accuracy and robustness of kinect pose estimation in the context of coaching of elderly population," in *Engineering in medicine and biology society (EMBC), 2012 annual international conference of the IEEE*. IEEE, 2012, vol. 28, pp. 1188–1193.