Universidade de Brasília – UnB
Faculdade UnB Gama – FGA
Engenharia Eletrônica

Análise Cinemática Automática usando OpenPose e Dynamic Time Warping com Aplicações no Remo

Autores: Victor Oliveira Corrieri de Macedo
Joyce da Costa Santos
Orientador: Roberto de Souza Baptista

Brasília, DF
2019

Victor Oliveira Corrieri de Macedo
Joyce da Costa Santos

Análise Cinemática Automática usando OpenPose e Dynamic Time Warping com Aplicações no Remo

Monografia submetida ao curso de graduação em Engenharia Eletrônica da Universidade de Brasília, como requisito parcial para obtenção do Título de Bacharel em Engenharia Eletrônica.

Universidade de Brasília – UnB

Faculdade UnB Gama – FGA

Orientador: Roberto de Souza Baptista

Brasília, DF
2019

Resumo

Este trabalho propõe um sistema de baixo custo para analisar automaticamente parâmetros cinemáticos no remo, a partir da captura e processamento de vídeo, usando uma única câmera RGB e sem a necessidade de marcadores no corpo do indivíduo. As coordenadas das articulações são estimadas a cada *frame* usando a API da OpenPose em conjunto com um filtro *offline* para contornar as possíveis perdas de *frames* e oscilações na trajetória. Os ângulos das articulações são obtidos por meio das coordenadas em *pixels* das articulações estimadas. Suas trajetórias são, então, avaliadas utilizando uma técnica computacional chamada *Dynamic Time Warping*, a qual realiza uma comparação entre duas séries temporais, uma denominada referência e a outra, alvo. A série de referência consiste em um padrão de remada a ser seguido e é usada como base para avaliar a série alvo. No teste do sistema compara-se cada remada em um treino de cinco minutos de um remador iniciante com uma remada de referência, executada por um remador profissional. Além disso, avalia-se um treino também de cinco minutos do mesmo remador profissional para conferir a consistência em sua própria remada. Por fim, todas as métricas cinemáticas extraídas são exibidas em uma interface para monitorar o movimento do remador e fornecer um *feedback*. A abordagem proposta permite a análise automática de sessões de treinamento gravadas com câmera simples, e pode ser útil para auxiliar na melhoria de movimento de remadores, principalmente, iniciantes.

Palavras-chaves: Remo, OpenPose, Análise Automática, Filtro de Kalman, Dynamic Time Warping.

Abstract

This work proposes a low cost system to automatically analyze kinematic parameters in rowing, using video capture and processing, with a single RGB camera and without the need for markers on the individual's body. The coordinates of the joints are estimated in each frame using the OpenPose API together with an offline filter to overcome frame loss and oscillations in the trajectories. The joint angles are obtained by means of the pixel coordinates from the estimated joints. Their trajectories are then evaluated using a computational technique named Dynamic Time Warping, which performs a comparison between two time series, one denominated reference and the other, target. The reference series consists of a rowing pattern to be followed and it is used as basis to evaluate the target series. The system test compares each stroke in a five-minute workout by a novice rower with a reference stroke, executed by a professional rower. In addition, a five-minute workout by the same professional rower is evaluated for consistency in his own stroke. Finally, all extracted kinematic metrics are displayed in an interface to monitor rower movement and provide feedback. The proposed approach allows automatic analysis of simple camera recorded training sessions, and could be useful to assist in improving the movement of rowers, especially unexperienced.

Key-words: Rowing, OpenPose, Motion Analysis, Kalman Filter, Dynamic Time Warping.

Lista de ilustrações

Lista de abreviaturas e siglas

RGB	Red Green and Blue
API	Aplication Programming Interface
MOCAP	Motion Capture
3D	Tridimensional
DTW	Dynamic Time Warping
MMM	Master Motor Map DoF
DoF	Degrees of Freedom
RGB-D	Red Green and Blue - Depth
ANN	Artificial Neural Network
DNN	Deep Neural Network
CNN	Convolutional Neural Network
RNN	Recurrent Neural Network
YOLO	You Only Look Once
2D	Bidimensional
PAF	Part Affinity Fields
COCO	Common Objects in Context
VGG	Visual Geometry Group
CNN	Convolutional Neural Network
MPII	Max Planck Institute for Informatic
NMS	Non-maximum Suppression
rpm	remadas por minuto

Sumário

1 Introdução

1.1 Considerações iniciais

O remo é uma atividade física que requer resistência e movimento coordenado do corpo para mover o barco com eficiência. Esse movimento pode ser dividido em duas fases: propulsão e recuperação [5]. Como avaliado em [6], uma diferença observada nos remadores de elite é a consistência, pois conseguem manter uma técnica semelhante independente da cadência (velocidade de execução do movimento medida, normalmente, em remadas por minuto). As fases das remadas devem ser realizadas de forma sequenciada para uma correta execução e melhor desempenho [7].

Com a inserção do remo ergômetro em academias, houve um aumento na prática desse esporte. No entanto, não necessariamente com a técnica adequada. O remo quando realizado com a técnica incorreta pode levar a lesões, mais usualmente, nas costas [8]. Nessas academias, a técnica, geralmente, é apresentada pelo treinador e depois é feita uma supervisão do exercício, de forma que a falha técnica é identificada visualmente por parte do treinador e corrigida verbalmente. Em treinamentos para competição, é comum utilizar análise de vídeo para detectar falhas de execução.

Dentre os parâmetros de interesse na biomecânica do remo estão: a amplitude de remada, a cadência, a força aplicada no puxador, a força aplicada no pé e os ângulos do quadril e do joelho, durantes as etapas do movimento [5].

Ainda que a remada não seja um movimento estritamente simétrico, a maior parte da informação está contida no plano sagital e pode ser analisada em apenas um dos lados do corpo [9]. Essa é uma simplificação recorrente em pesquisas [10], [6].

No remo, existem muitos projetos de análise cinemática automática que utilizam sensores "vestíveis" (como unidades inerciais [11], [12], [13]) ou instrumentação no remo ergômetro [14], [15]. Apesar desses sistemas obterem resultados precisos, os dispositivos utilizados podem causar incômodo ao remador em razão da longa duração dos treinos e competições. Além disso, podem ocorrer deslocamentos na posição dos sensores, pelo deslize com a pele.

Uma análise automática de movimento em vídeo requer duas etapas: captura de movimento e análise de dados. A captura de movimento inclui não somente a captura em si, mas também a extração da posição e dos ângulos das articulações.

O padrão ouro, para efeito de comparação, são os sistemas com base em marcadores MOCAP (captura de movimento) como o Vicon ou Qualisys. Existem várias aplicações

usando esses sistemas para análise de remo [8], [16], [10], [17], [18]. No entanto, são sistemas de custo elevado e requerem procedimentos de calibração. Além disso, existem três problemas principais no uso de marcadores:

- A fixação é dificultada em áreas de muito movimento, como no joelho ou no cotovelo, pois, além da movimentação da pele, o suor e a umidade também podem descolar os marcadores durante o exercício.
- A própria movimentação da pele durante o exercício desloca o marcador do ponto real de articulação, introduzindo erros.
- O uso dos marcadores atrapalha o desempenho em competições e, portanto, só podem ser usados em treinos.

Com os avanços da inteligência artificial e aprendizagem de máquina, muitas técnicas de processamento de vídeo, sem utilizar marcadores, para a extração de parâmetros de forma automática, foram propostas, como a OpenPose e outros [2], [19]. Os métodos sem marcadores têm como principal vantagem a praticidade para realizar a medição, que requer apenas a configuração das câmeras.

Técnicas sem marcadores são, geralmente, aplicadas para reconhecimento de movimentos, mas raramente, para análises biomecânicas devido à baixa precisão e *frames* perdidos. Isso se dá, em muitos casos, quando o algoritmo não leva em consideração as dependências temporais entre os pontos e realiza estimação em cada *frame* de forma independente.

Uma forma de mitigar esse problema é realizar um pós processamento nesses pontos estimados Essa abordagem pode ser modelada como um problema de *tracking*, onde a detecção de cada ponto pode ser conciliada com algoritmos de rastreamento para completar os dados perdidos e, eventualmente, melhorar a medição.

Outra maneira de melhorar a acurácia em sistemas sem marcadores é acrescentar informação *a priori*. Entende-se por informação *a priori*, qualquer informação específica para o tipo de aplicação, seja informações sobre o tipo de movimento, seja as características físicas da pessoa avaliada. Esse tipo de dado pode ser usado de forma complementar para melhorar o desempenho do método. Sua desvantagem está em tornar o sistema limitado para a aplicação em que foi projetado, perdendo a robustez para atuar em diferentes cenários.

Além da divisão dos sistemas de captura de movimento entre com ou sem marcadores, existe também a divisão entre câmera única ou múltiplas câmeras. No geral, se o objetivo é a reconstrução tridimensional do movimento, a maioria dos métodos envolvem múltiplas câmeras, tanto para sistemas com marcadores, como o Vicon, quanto para

sistemas sem marcadores [20] usando o Kinect ou usando câmeras RGB simples, como a OpenPose. Apesar da OpenPose permitir a estimação com uma câmera, a funcionalidade de reconstrução 3D exige mais de duas, atualmente.

Ainda assim, alguns sistemas baseados em um único Kinect [21], [22] ou até sistemas sem restrições de câmera [23], [24] se propõem a realizar a reconstrução tridimensional do movimento por apenas uma imagem e demonstram conseguir resultados próximos ao estado da arte.

Em posse dos dados cinemáticos do movimento, uma análise pode ser realizada para fornecer *feedbacks* do exercício para o usuário. Esses *feedbacks* são recursos, raramente, explorados para remadores iniciantes. Um treinador pode analisar visualmente o desempenho de um remador e dar o devido suporte técnico. Entretanto, algoritmos capazes de detectar padrões de movimentos do corpo podem ser usados como ferramenta de auxílio para o treinador, facilitando o acompanhamento do exercício e melhorando o desempenho do movimento.

Neste contexto, uma forma de realizar a avaliação do movimento no remo seria utilizando algoritmos de comparação entre séries temporais como o *Dynamic Time Warping* (DTW). Esse algoritmo permite fazer um mapeamento entre pontos em diferentes momentos de cada série temporal, de acordo com suas características. Dessa forma, seria possível encontrar e comparar fases características em um ciclo de remada entre duas técnicas diferentes.

1.2 Objetivos

Tendo em vista a importância da análise biomecânica no esporte e prezando por configurações de baixo custo e alta praticidade, este trabalho propõe um sistema de estimação de pose, sem marcadores e com uma única câmera RGB de baixa resolução, para obter os ângulos das articulações a fim de avaliar os ciclos de remada utilizando o algoritmo *Dynamic Time Warping*. Esse sistema deve rastrear as posições dos pontos de articulação durante o movimento e extrair tanto as angulações de interesse como a cadência, além de comparar cada ciclo de remada com uma referência, ou seja, um padrão de remada a ser seguido. Por fim, o sistema deve fornecer ao remador um *feedback* do seu desempenho, usando como métrica as diferenças em angulação com relação à referência e análises gráficas de consistência com o decorrer dos ciclos.

Considerando a simetria mencionada, a análise do movimento será simplificada para o plano sagital e, portanto, a reconstrução tridimensional não será necessária. Além disso, serão avaliadas seis articulações: tornozelo, joelho, quadril, ombro, cotovelo e pulso.

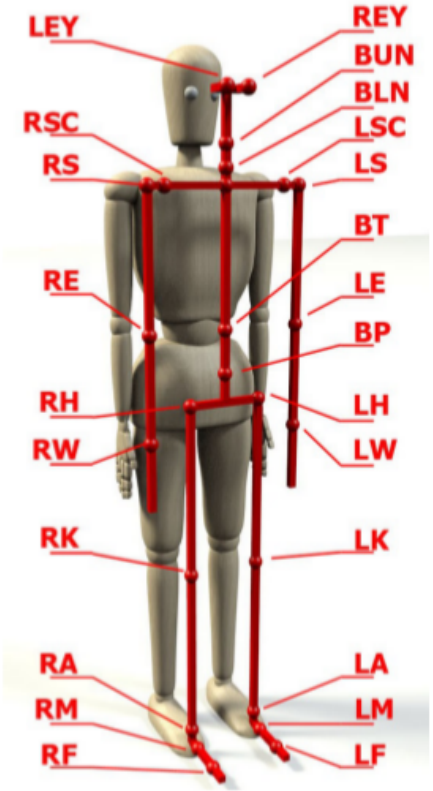
A avaliação da técnica deve ser realizada automaticamente para qualquer momento

do ciclo de remada desejado, o qual deve ser definido, manualmente, pelo usuário ou treinador a partir da remada de referência. Neste trabalho, serão avaliados cinco momentos. A descrição desses momentos será realizada nas próximas seções.

2 Fundamentos Teóricos e Estado da Arte

2.1 Estimação de Pose Humana

Em visão computacional, o corpo humano pode ser considerado como um objeto articulado que consiste de parte móveis rígidas, conectadas através das articulações [25]. No projeto Master Motor Map (MMM) desenvolvido por Terlemez et.al [1], um modelo em corpo rígido do corpo humano foi realizado, detalhando os graus de liberdade e limites de rotação para cada articulação do corpo humano, como ilustrado na Figura (1).



Joint	DoF	X-Limits	Z-Limits	Y-Limits
LF/RF	1+1	[-30°,45°]	-	-
LM/RM	1+1	-	[-30°,45°]	-
LA/RA	3+3	[-40°,30°]	[-30°, 30°]	[-20°, 20°]
LK/RK	1+1	[-130°,0°]	-	-
LH	3	[-50°,95°]	[-45°,45°]	[-20°,65°]
RH	3	[-50°,95°]	[-45°,45°]	[-65°,20°]
LW	2	[-30°,20°]	[-70°,50°]	-
RW	2	[-30°,20°]	[-50°,70°]	-
LE/RE	2+2	[0°,160°]	[-90°,90°]	-
LS	3	[-70°,190°]	[-70°,60°]	[0°,160°]
RS	3	[-70°,190°]	[-60°,70°]	[-160°,0]
LSC/RSC	2+2	-	[-20°,20°]	[-20°,20°]
LEY/REY	2+2	[-60°,60°]	-	[-60°,60°]
BUN	3	[-20°,30°]	[-20°,20°]	[-15°,15°]
BLN	3	[-45°,15°]	[-15°,15°]	[-20°,20°]
BT	3	[-35°,27°]	[-36°,36°]	[-20°,20°]
BP	3	[-50°,35°]	[-45°,45°]	[-20°,20°]

Figura 1 – Modelo cinemático do corpo humano com os limites angulares em cada eixo (*X-Limits*, *Z-Limits* e *Y-Limits*) por articulação, assim como o número de graus de liberdade (*DoF*), desenvolvido no projeto *Master Motor Map (MMM)*. Adaptado de [1].

Neste contexto, a estimação de pose por ser definida como a identificação da posição bidimensional ou tridimensional desses pontos de articulação no corpo, de forma a realizar suas interconexões e formar um *sticker*, ou boneco, com a pose estimada.

Como mencionado anteriormente, no caso da estimação de pose por imagens ou

vídeos, essa identificação pode ser realizada com ou sem marcadores e com uma única ou múltiplas câmeras. Além disso, as câmeras podem fornecer uma imagem em três dimensões (RGB) ou em quatro (RGB-D), onde as três primeiras correspondem aos canais de cor vermelho, verde e azul, e a quarta dimensão representa a profundidade.

Existem diversas abordagens para a estimação de pose sem marcadores. Métodos com múltiplas câmeras, como o *visual hull* [26] eram, tradicionalmente, muito utilizados. Entretanto, com a popularização das redes neurais e sua aplicação em estimação de pose [27], a tendência de pesquisa se voltou para essa área.

Uma rede neural é um sistema bioinspirado e projetado para a solução de problemas de classificação e regressão, por meio da identificação de padrões em dados rotulados.

Uma rede neural artificial (ANN) implementa uma função não-linear na entrada, que tem dimensão fixa para o número de dados n para qual foi projetada. Essa função é formada por um conjunto de nós ou neurônios, onde cada um implementa uma combinação linear de suas entradas $\mathbf{x}$ com pesos $\mathbf{W}$, somadas de um fator de deslocamento ou *bias* $\mathbf{b}$. Essa combinação linear é, então, aplicada em uma função de ativação f', a qual introduz a não linearidade do sistema. Assim, a saída $\mathbf{y}$ obtida pode ser representada pela seguinte equação:

$$\mathbf{y} = f' \left(\sum_{i=0}^{k} (\mathbf{W_i x} + \mathbf{b_i} \right) \tag{2.1}$$

onde k é o número de nós da rede e f' é a função de ativação do nó. É possível construir um número arbitrário de nós, sendo que, teoricamente, uma maior quantidade permite a identificação de padrões mais complexos, mas ao mesmo tempo requer um maior conjunto de dados para ser treinada.

As funções de ativação aplicada nos nós podem ser pensadas como os limites de ativação de um neurônio, onde a despolarização da célula só ocorre a partir de um limiar de ativação. Para simular esse efeito é comum utilizar funções de ativação do tipo ReLu, na qual a saída é igual a entrada apenas para entradas maiores que zero, no resto é nula.

Para problemas de classificação é comum utilizar uma função de ativação diferente no nó de saída, como a *softmax*. Nessa função a saída é limitada entre 0 e 1, o que pode ser associado a uma probabilidade para a classificação. Em problemas de regressão, não é necessário utilizar uma função de ativação na saída.

Além disso, uma ANN pode ter múltiplas camadas internas ou "ocultas", passando a ser consideradas redes neurais profundas (DNN) e aumentando a quantidade de parâmetros da rede. Existem também as redes neurais convolucionais (CNN), onde além dos parâmetros da combinação linear, acrescenta-se parâmetros de filtros convolucionais que são aplicados nas primeiras camadas da rede. As CNNs são, hoje, o estado da arte em classificação de imagens.

Dessa forma, as rede neurais são funções não lineares que dependem de um conjunto de parâmetros. Esses parâmetros são determinados através de uma etapa de treinamento que utiliza, normalmente, o algoritmo de *backpropagation*.

O algoritmo de *backpropagation* consiste em realizar a etapa de *feedforward* para cada dado do *dataset*. O *feedforward* realiza o cálculo da saída $\mathbf{y}$ pela Eq. (2.1) em cada nó, com os parâmetros iniciados aleatoriamente. Em seguida, o valor resultante é comparado com o valor esperado ou valor rotulado $\mathbf{y_e}$, por meio de alguma função de erro E ou função custo, como o erro absoluto:

$$E = |\mathbf{y} - \mathbf{y_e}| \tag{2.2}$$

Com isso, realiza-se um processo de otimização pelo cálculo do gradiente da função custo, de forma a encontrar um vetor na direção que minimiza essa função.

Assim, para cada dado rotulado, é aplicado um processo de *feedforward* seguido da atualização dos parâmetros $\mathbf{P}$ pela seguinte equação:

$$P' = P - l_r \nabla E \tag{2.3}$$

onde $\mathbf{P'}$ são os pesos atualizados, l_r é a taxa de aprendizado e ∇E é o gradiente da função custo. A taxa de aprendizado indica o quanto se deseja atualizar os parâmetros em função de cada dado ou conjunto de dados de entrada, sendo um dos hiper-parâmetros de redes neurais, o que significa que não é um parâmetro treinado e deve ser especificado inicialmente. Esse valor influencia na sensibilidade da rede para cada dado de entrada e está relacionado ao problema de *overfitting*, onde a rede pode ter aprendido padrões muito específicos para o conjunto de dados de treino e não se generaliza para o problema.

Após o processo de treinamento, a rede neural treinada pode ser utilizada para situações de teste na etapa de inferência, aplicando apenas o *feedforward* para gerar a estimativa de classificação ou regressão sem continuação do aprendizado.

O potencial das redes neurais está no fato de satisfazer o teorema da aproximação universal [28]. Esse teorema, de maneira simplificada, estabelece que para uma arquitetura de tamanho limitado, é possível aproximar qualquer função. Isso é importante, pois conclui que uma rede neural poderia, idealmente, resolver qualquer problema de classificação ou regressão dada uma arquitetura adequada e um conjunto determinado de parâmetros, que podem ser obtidos, ou ao menos aproximados, através do processo de treinamento.

Entretanto, como ainda não há uma forma direta de especificar a arquitetura ideal, além dos valores dos hiper-parâmetros para cada aplicação, o processo de escolha da rede é em, grande parte, empírico. Além disso, a etapa de treinamento pode ser muito custosa computacionalmente e necessitar de uma grande quantidade de dados, o que nem sempre é viável.

Uma forma de reduzir a quantidade de dados e tempo de processamento necessários durante a etapa de treinamento é utilizar o conceito de transferência de aprendizado. A motivação desse método é a observação de que as camadas iniciais da rede tendem a identificar padrões mais simples, os quais vão sendo associados nas camadas subsequentes para compor padrões mais complexos e abstratos [29].

Então, por exemplo, se o problema se trata de uma classificação de rostos em imagens, é provável que as primeiras camadas sejam sensíveis à padrões simples, como linhas retas ou forma circulares, enquanto as camadas mais internas identificam padrões mais complexos como olhos ou bocas.

Assim, a ideia da transferência de aprendizado é que, caso haja uma rede neural já treinada para um problema semelhante, é possível aproveitar os pesos treinados das camadas iniciais dessa rede e re-treinar apenas as camadas finais para se adequar ao seu problema. Esse método é muito usado em classificação de imagens, em que redes otimizadas para classificação de múltiplos objetos, como a YOLO [30], podem ser usadas como base para extrair características ou *features* enfatizando padrões genéricos que podem ser introduzidos como entrada para outra rede neural mais específica para o problema.

Dentre as principais arquiteturas de redes neural para a estimação de pose, atualmente se destacam: Openpose [2], VNect [23], DensePose [19], PoseFlow [31], PoseTrack [32]. Outro trabalho relacionado foi desenvolvido em [24], e trata-se de um sistema auto supervisionado que realiza a rotulação dos dados 2D a partir de dados 3D.

Em geral, essas arquiteturas possuem milhões de parâmetros e foram treinadas em longos *datasets*, o que exige muito tempo de processamento e *hardware* otimizado. Por esse motivo, é comum em pesquisas, utilizar arquiteturas pré-treinadas e de código aberto como a OpenPose ou a DensePose. Para esse propósito, a OpenPose se destaca por possuir uma documentação detalhada e funcionalidades com suporte para Python e C++, além de apresentar resultados no estado da arte em estimação de pose.

OpenPose

A OpenPose foi desenvolvida com base no trabalho de Cao et. al [2], o qual introduziu o conceito de campos de afinidade entres partes (PAFs). Essa rede foi treinada com longos *datasets*, como o COCO Keypoints [33] com 25 pontos de articulação (Fig. 2).

O maior diferencial da OpenPose está na sua capacidade de fazer estimação de múltiplas pessoas de maneira conjunta, sem afetar significativamente o tempo de processamento. Isso ocorre pois, ao invés de realizar a estimação individualmente para cada pessoa identificada, o método realiza a maior parte dos procedimentos em paralelo.

A arquitetura dessa rede está apresentada na Figura (3). Como é possível observar, a primeira etapa da rede consiste na extração de *features* da imagem crua, usando a rede

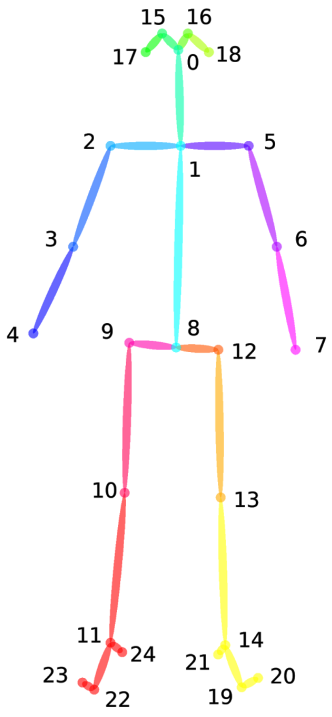


Figura 2 – Modelo de corpo humano *Body-25* com base no *dataset* COCO para 25 pontos articulação.

VGG-19 [34], por transferência de aprendizado. A imagem de entrada nessa rede possui dimensão $h \times w$ sendo, normalmente, uma versão reescalada da imagem original. Em seguida, as *features* extraídas passam por dois estágios que são replicados paralelamente para cada articulação a ser identificada.

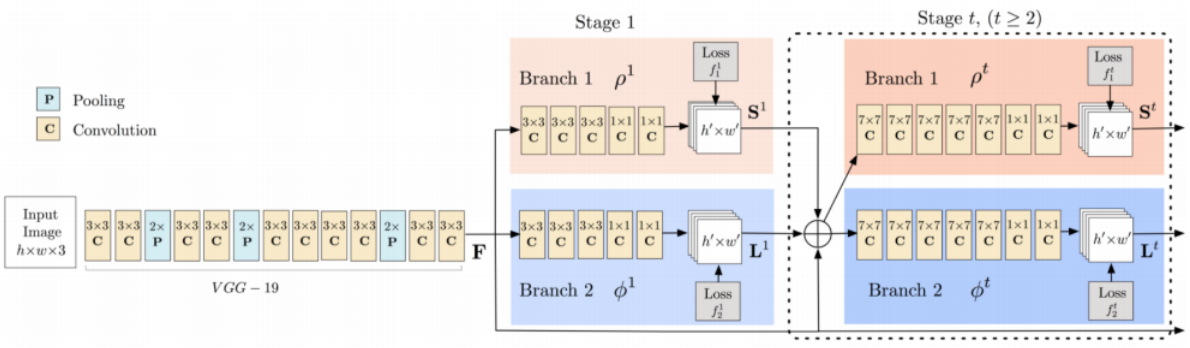


Figura 3 – Arquitetura da CNN de múltiplos estágios OpenPose [2].

Os dois estágios da CNN utilizam as *features* de entrada para gerar três tipos mapas de probabilidade ou *heatmaps*:

- **Por articulação**: *heatmaps* de cada uma das 25 articulações, representando a probabilidade estimada em cada sub-região na imagem, considerando que a imagem foi subamostrada para dimensão 46×74.
- **Do fundo**: *heatmap* que mostra a probabilidade do *pixel* representar um ponto do ambiente em vez da pessoa, ou seja, fornece um mapa de probabilidade de regiões que não são articulações.
- **Por PAF**: campos vetoriais da afinidade entre duas articulações, formados pela associação em pares de dois mapas de intensidade, os quais representam a direção nos eixos x e y para qual o campo aponta.

De forma a definir os pontos de articulação a partir dos mapas de probabilidade, utiliza-se o algoritmo *non-maximum suppression* (NMS), como definido em [2]. Esse algoritmo encontra os máximos locais no mapa, os quais são armazenados como prováveis pontos de articulação. As etapas desse procedimento estão exemplificadas na Fig. (4), para o mapa de probabilidade do quadril direito, e seguem o seguinte algoritmo:

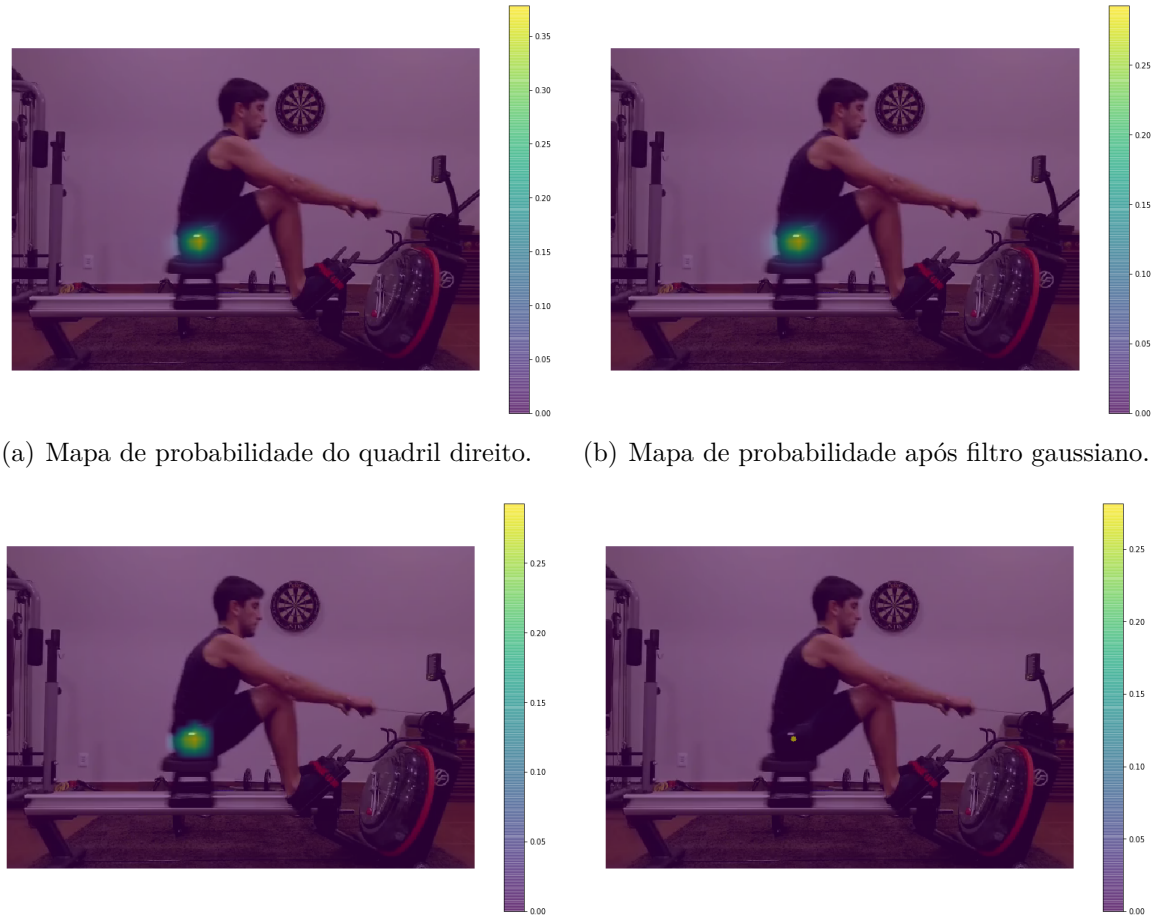
1. A partir do *heatmap* da articulação (Fig. 4(a)), aplicar um filtro gaussiano com janela 3×3 para borrar a imagem, tirando uma média (Fig. 4(b)).
2. Limitar a imagem resultante por um *threshold* ajustável e definido como $0, 1$, zerando todos os valores menores (Fig. 4(c)).
3. Encontrar os máximos locais dos agrupamentos resultantes. Note que na Fig. (4(d)), o círculo amarelo representa apenas um *pixel*, sendo aumentado apenas para facilitar sua visualização.

Após a obtenção do mapa da Fig. (4(d)), é necessário fazer a sobreamostragem da imagem de volta para o tamanho original, sem os canais de cores. Nessa matriz, os *pixels* diferentes de zero são dados como pontos válidos com seus valores indicando a confiabilidade da medida.

Por fim, cada ponto detectado é armazenado em um vetor e descrito por quatro componentes: coordenada em x, coordenada em y, confiabilidade e número da articulação.

Caso o problema fosse simplificado para detecção de apenas uma pessoa no *frame*, esses pontos poderiam ser ligados diretamente, assumindo que não foi detectado mais de um ponto por articulação. Assim, não seria necessário utilizar os *heatmaps* de PAF. Entretanto, para evitar falsos positivos e não restringir o método, um segundo procedimento é aplicado para realizar a conexão entre os pontos.

Uma abordagem simplificada seria definir a associação entre os pontos a partir da distância entre os candidatos, porém esse método pode gerar muitos erros, principalmente



(a) Mapa de probabilidade do quadril direito.

(b) Mapa de probabilidade após filtro gaussiano.

(c) Mapa de probabilidade limitado por um *threshold*.

(d) Pontos de máximo local encontrados no mapa.

Figura 4 – Exemplo do algoritmo NMS para extração do ponto de articulação do quadril direito.

para pessoas próximas. Uma estratégia mais eficiente é usar os mapas PAF gerados na rede.

O algoritmo de associação dos pontos por meio do PAF tem como entrada os pontos encontrados e os mapas PAF armazenados, seguindo a lógica exemplificada na Fig. (5) (para o caso da conexão entre o quadril e o joelho direito):

1. Escolher uma combinação permitida entre duas articulações j_1 e j_2.
2. Traçar um vetor $\vec{d_j}$ entre os pontos j_1 e j_2 e normalizá-lo com a seguinte equação:

$$\hat{d}_j = \frac{\vec{d_{j_2}} - \vec{d_{j_1}}}{\left|\left|\vec{d_{j_2}} - \vec{d_{j_1}}\right|\right|_2} \tag{2.4}$$

3. Definir um número n de pontos para serem interpolados entre j_1 e j_2.
4. Encontrar o valor do vetor de campo $\vec{L_j}$ (Fig. 5(a)), formado pela associação entre duas PAFs nos pontos interpolados u e realizar o produto interno com $\vec{d_j}$ (Fig. 5(b)),

de acordo com a seguinte equação:

$$E_{ji} = \vec{L_j}(u(i)) \cdot \hat{d_j} \tag{2.5}$$

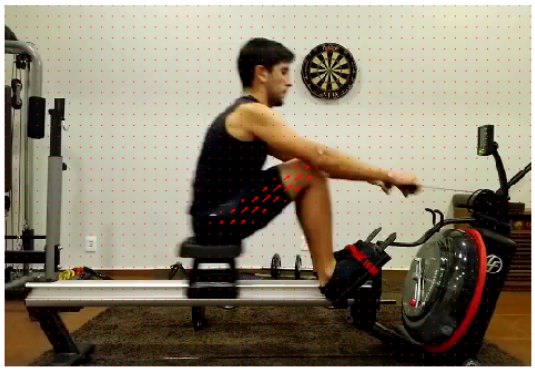
para cada i variando de 0 até o número de pontos. Nesse caso, E_{ji} se aproxima de 1, conforme o vetor de campo no ponto e o vetor da direção entre as articulações se alinham.

5. Tirar a percentagem de pontos i, para os quais E_{ji} é maior que um *threshold* p_{th}. Esse *threshold* é definido empiricamente e determina a sensibilidade do sistema.

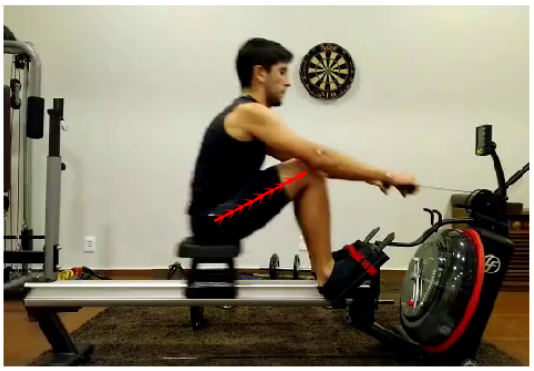
6. Se essa percentagem for maior que outro *threshold* c_{th}, a associação entre os pontos é válida.

7. Repetir os passos para todas as associações entre dois pontos possíveis e, caso exista mais de uma, escolher a que apresentar a maior percentagem de pontos maiores que c_{th}.

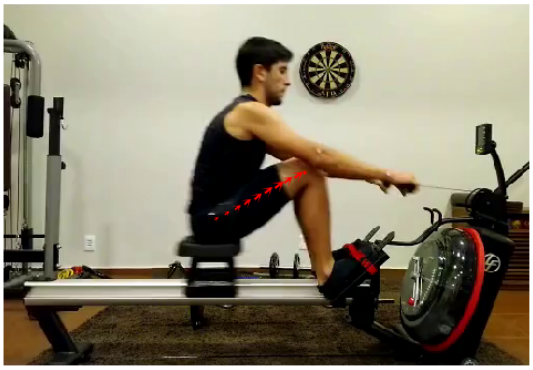
8. Repetir os passos para todos os pares de pontos possíveis.



(a) Campo vetorial da PAF.



(b) Vetores normalizados na direção entre os pontos.



(c) Campo vetorial da PAF nos pontos interpolados.

Figura 5 – Exemplo do algoritmo de associação por PAF para o quadril e joelho direito.

Finalmente, uma nova matriz é formada, onde para cada pessoa P identificada em cada *frame*, são definidas as coordenadas x, y de cada ponto de articulação J. Este resultado está demonstrado na Fig. (6) para a mesma imagem de teste.

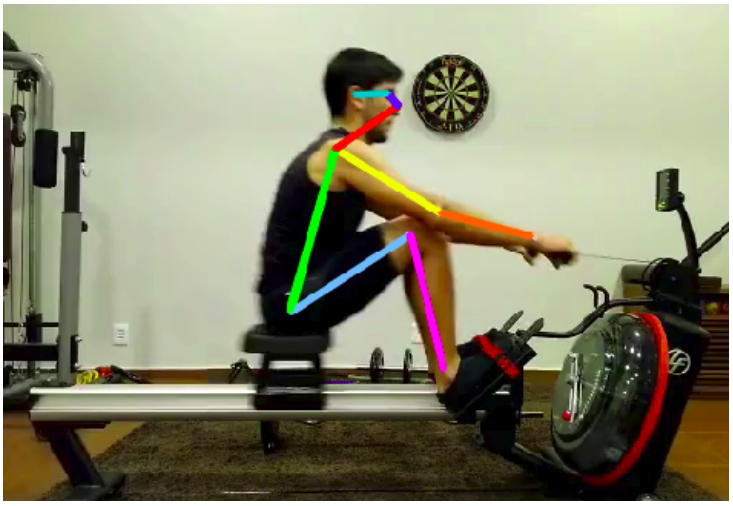


Figura 6 – Resultado do algoritmo de estimação de pose usando a rede neural da OpenPose para 18 pontos de articulação do modelo COCO.

2.2 Filtro de Kalman

O Filtro de Kalman é uma técnica muito utilizada em engenharia de controle, com aplicações em diversas áreas [35]. Uma dessas aplicações é no rastreamento da trajetória de partículas ou de objetos no espaço [36].

Em sua formulação mais básica, o algoritmo estima as variáveis de um processo modelado como um sistema linear em espaço de estados com as seguintes relações [37]:

$$\begin{aligned} \mathbf{x_k} &= \mathbf{F_{k-1}} + \mathbf{G_{k-1}u_{k-1}} + \mathbf{w_{k-1}} \\ \mathbf{y_k} &= \mathbf{H_k x_k} + \mathbf{v_k} \end{aligned} \tag{2.6}$$

onde $\mathbf{x}$ é o vetor de estados, $\mathbf{y}$ é o vetor de saída, $\mathbf{u}$ é o vetor de entrada, $\mathbf{w}$ e $\mathbf{v}$ são os ruídos referentes ao processo e a medição, respectivamente. $\mathbf{F}$ é a matriz de estados do sistema, $\mathbf{G}$ a matriz de entrada e $\mathbf{H}$, a matriz de observação.

Após modelado o sistema, pode-se aplicar o algoritmo de Kalman. Esse algoritmo funciona com base em um predição seguida de uma correção. Essa correção é realizada comparando a distribuição estatística das medições com a distribuição das estimativas e produzindo uma saída corrigida com distribuição proporcional entre as duas, de acordo com os parâmetros especificados.

A primeira etapa do algoritmo de Kalman é a predição do vetor de estado a partir da equação de dinâmica dos estados, definida como:

$$\mathbf{\hat{x}_{k|k-1}} = \mathbf{F_{k-1}\hat{x}_{k-1}} + \mathbf{G_{k-1}u_{k-1}} \tag{2.7}$$

onde $\hat{\mathbf{x}}_{\mathbf{k|k-1}}$ é o vetor de estado estimado a partir do estado anterior e $\hat{\mathbf{x}}_{\mathbf{k-1}}$ é vetor de estado estimado anteriormente. O próximo passo é a predição da matriz de covariância do erro da estimativa no instante k, dado o instante anterior $k-1$, seguindo a equação:

$$\hat{\mathbf{P}}_{\mathbf{k|k-1}} = \mathbf{F}_{\mathbf{k-1}}\mathbf{P}_{\mathbf{k-1}}\mathbf{F}_{\mathbf{k-1}}^{\mathbf{T}} + \mathbf{Q}_{\mathbf{k-1}} \tag{2.8}$$

Em sequência, obtém-se a matriz de ganho de Kalman $\mathbf{K_k}$ como:

$$\mathbf{K_k} = \mathbf{P}_{\mathbf{k|k-1}}\mathbf{H}_{\mathbf{k}}^{\mathbf{T}}(\mathbf{H_k}\mathbf{P}_{\mathbf{k|k-1}}\mathbf{H}_{\mathbf{k}}^{\mathbf{T}} + \mathbf{R_k})^{-1} \tag{2.9}$$

sendo $\mathbf{R_k}$ a covariância do ruído da medida. O vetor de estado é, então, atualizado pela equação:

$$\hat{\mathbf{x}}_{\mathbf{k}} = \hat{\mathbf{x}}_{\mathbf{k|k-1}} + \mathbf{K_k}(\mathbf{z_k} - \mathbf{H_k}\hat{\mathbf{x}}_{\mathbf{k|k-1}}) \tag{2.10}$$

onde $\mathbf{z_k}$ é a medida da saída. Por fim, a covariância do erro de estado é atualizada:

$$\hat{\mathbf{P}}_{\mathbf{k}} = (\mathbf{I} - \mathbf{K_k}\mathbf{H_k})\hat{\mathbf{P}}_{\mathbf{k|k-1}} \tag{2.11}$$

com $\mathbf{I}$ representando a matriz identidade.

A vantagem do filtro de Kalman em relação a um filtro comum, usando a transformada de Fourier, está no fato de utilizar uma modelagem do sistema para corrigir a medição. Isso permite a construção de um filtro específico para a aplicação, além de lidar com dados perdidos, ou falsos negativos.

2.3 Dynamic Time Warping

O *Dynamic Time Warping* (DTW) busca interpretar distorções entre dois sinais de comportamento semelhante, onde um é a referência e o outro o alvo da comparação, de forma a mapear os pontos de maior semelhança. Para isso o sinal alvo é comprimido ou, em outros casos, estendido, buscando a maior similaridade entre as sequências. Isso dá ao algoritmo uma robustez quanto a frequência, duração ou distorção dos sinais.

Uma técnica comum utiliza o algoritmo computacional que tem como princípio encontrar um alinhamento ideal entre duas séries temporais independentes. Embora duas sequências possam ter o mesmo tamanho, as formas podem não se alinhar no eixo x. O algoritmo percorre o eixo y, enquanto distorce o eixo x. Isto resulta em um mesmo ponto sendo conectado a mais de um ponto de outra série [3].

Contudo, em alguns casos, o algoritmo pode não encontrar a correta correspondência e isso pode ocorrer devido a um pico ou vale, mais alto ou mais baixo que a referência [3].

A Figura (7) mostra a diferença na representação quando o algoritmo não corresponde ao ponto corretamente. Se essas sequências fossem idênticas, idealmente, a representação não seria distorcida, pois cada ponto estaria interligado verticalmente em ângulos de 90° [38].

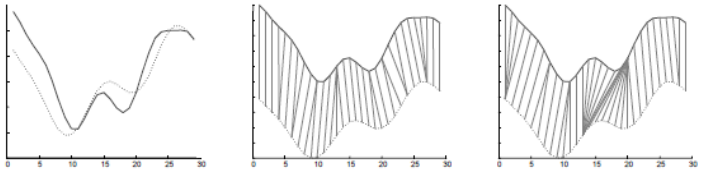


Figura 7 – Representação de um alinhamento ideal e não-ideal de duas séries temporais. Adaptado de [3].

O algoritmo

Para encontrar o alinhamento ideal entre duas séries, suponha duas sequências temporais, A e B, de tamanho n e m, respectivamente:

$$\begin{aligned} A &= a_1, a_2, ..., a_i, ..., a_n \\ B &= b_1, b_2, ..., b_j, ..., b_m \end{aligned} \tag{2.12}$$

Suponha que a sequência A esteja representada no eixo x e a sequência B no eixo y. A Figura (8) mostra duas séries temporais dispostas em uma matriz. Essa matriz $n \times m$ começa a ser construída a partir da distância euclidiana entre os vetores.

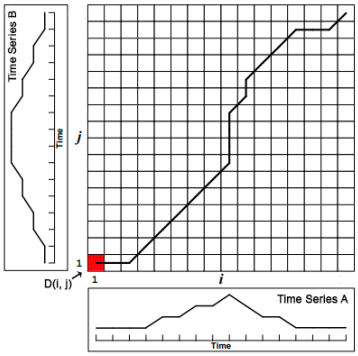


Figura 8 – Matriz representando duas séries temporais e o custo cumulativo $D(i, j)$. Adaptado de [4].

Essa distância é dada pelo elemento $d(i, j)$ e definida por:

$$d(i, j) = (a_i - b_j)^2 \tag{2.13}$$

Em paralelo, o algoritmo faz uma comparação entre três valores adjacentes (linha e coluna anterior; linha anterior; coluna anterior) ao valor atual e identifica o menor valor entre eles. Como resultado, o elemento $D(i, j)$ (Eq. 2.14) armazena a soma da distância euclidiana com o menor valor. Esse algoritmo é descrito, matematicamente, da seguinte forma:

$$D(i, j) = d(i, j) + min[D(i-1, j-1), D(i-1, j), D(i, j-1)] \tag{2.14}$$

Cada elemento (i, j) da matriz corresponde ao alinhamento entre os pontos (a_i, b_j). Um caminho W representa o conjunto desses elementos que configura o mapeamento entre as séries. Esse caminho, comumente chamado de *Warping path*, é descrito por:

$$W = w_1, w_2, ..., w_K \tag{2.15}$$

Onde w_K é o *K-ésimo* elemento da diagonal alinhada da matriz. Segundo [3], para encontrar esse caminho, adotam-se as seguintes condições:

- **Fronteira**: para garantir que toda a matriz tenha sido percorrida, inicia-se o caminho do primeiro elemento (a_i, b_j), canto inferior esquerdo da matriz, e finaliza no canto superior direito (a_n, b_m).
- **Monotonicidade**: para garantir que o caminho mapeado seja crescente, ou seja, que o caminho não volta para si mesmo, as direções são restringidas a $(i-1, j-1), (i-1, j), (i, j-1)$. Além disso, percorrendo-se a diagonal, encontra-se sempre uma correspondência de um ponto do vetor A com um ponto de B.
- **Continuidade**: garante-se que nenhum elemento da matriz foi ignorado, uma vez que o caminho percorrido não realiza saltos em decorrência do incremento unitário dos índices, percorrendo linhas e/ou colunas.

Existem diversos caminhos que satisfazem essas condições, porém existe um único caminho que minimiza a distorção. Esse caminho pode ser encontrado através da distância cumulativa $\gamma(i, j)$ definida como a distância $d(i, j)$ e o valor mínimo dos elementos adjacentes ao elemento atual. Portanto, a Eq. (2.16) descreve as menores distâncias resultantes entre as séries, servindo como parâmetro de comparação para equivalência entre todos os pontos das duas séries temporais.

$$\gamma(i, j) = d(i, j) + min\gamma(i-1, j-1), \gamma(i-1, j), \gamma(i, j-1) \tag{2.16}$$

2.4 Trabalhos Semelhantes

A primeira etapa do trabalho consiste na realização de um pós-processamento nos dados adquiridos pela OpenPose, acrescentando informações temporais e específicas do movimento para realizar, além da detecção, o rastreamento dos pontos. Essa ideia é conhecida também como *Pose tracking*.

Já a segunda etapa, se propõe a usar o algoritmo DTW para detectar e segmentar momentos de interesse em um ciclo de remada, a partir dos sinais de ângulos das articulações.

Um algoritmo de *Pose tracking* foi desenvolvido recentemente pelos mesmos autores da OpenPose [39]. A solução implementada foi alterar a própria arquitetura da rede, usando redes neurais recorrentes (RNN) em vez de acrescentar um pós-processamento. Segundo os autores, essa rede também será disponibilizada em breve na API da OpenPose, porém, até o momento, ainda não pode ser testada.

A utilização desse método poderia suprimir a necessidade de parte da filtragem proposta neste trabalho. Entretanto, no contexto do remo, um filtro especificamente ajustado ainda pode produzir resultados melhores que os obtidos em [39].

Com relação aos trabalhos de estimação de pose com aplicações no remo, em [40] foi realizada a estimação de pose usando um método de segmentação do ambiente. Os resultados desse método foram comparados com o algoritmo de PAF descrito em [41], obtendo resultados semelhantes.

Apesar da segmentação automática de movimento com base em dados cinemáticos já ter sido proposta [42], não foram encontradas aplicações específicas para o cenário do remo.

3 Materiais e métodos

Este capítulo tem como objetivo descrever e simplificar a compreensão do sistema de análise cinemática do movimento proposto, abordando cada bloco funcional dentro dos dois subsistemas divididos: extração cinemática e avaliação de movimento. O fluxo de informação entre os blocos foi enfatizado, com o intuito de modularizar o sistema e permitir futuras alterações em bloco individuais, sem o comprometimento geral do sistema.

3.1 Extração cinemática

A extração cinemática consiste em duas etapas: medição e análise, como ilustrado no diagrama da Figura (9). Nas próximas subseções cada bloco será detalhado.

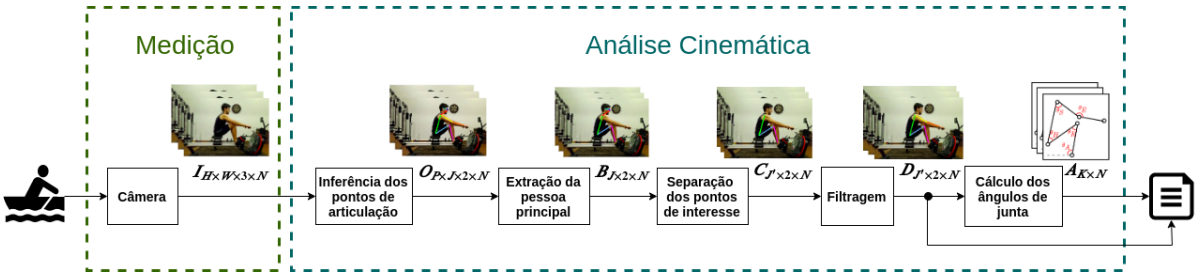


Figura 9 – Diagrama de blocos da primeira etapa de extração cinemática.

Câmera

A etapa de medição especifica o tipo de grandeza a ser medida, assim como o instrumento de medição (sensor). O vídeo fornecido pela câmera pode ser visto como uma matriz $I_{H\times W\times 3\times N}$, onde H e W definem a altura e largura do *frame*, respectivamente, e N define a quantidade de *frames*.

A escolha da câmera influencia nos resultados obtidos, pois as características de foco e resolução afetam a qualidade dos resultados. Além disso, a taxa de amostragem a ser utilizada também é um fator relevante, considerando que uma maior quantidade de informação facilita a etapa de pós processamento.

Contudo, como o objetivo do trabalho é desenvolver um sistema de baixo custo e pouca configuração, foi utilizada uma *webcam* simples modelo Logitech C920 com uma taxa de amostragem de 30 *frames* por segundo (fps).

Inferência dos pontos de articulação

O segundo passo consiste no processamento dos dados vindos do sensor para realizar uma estimação da pose do indivíduo filmado. A entrada desse bloco é a sequência de imagens $I_{H\times W\times 3\times N}$ e a saída é uma matriz $O_{P\times J\times 2\times N}$, onde P é o número de pessoas encontradas, J é o número de articulações consideradas, o número 2 se refere às duas coordenadas (i e j) em *pixel* do ponto de articulação na imagem e N é o número de *frames* avaliados.

Dessa forma, utilizou-se a API em Python da OpenPose para receber o vídeo $I_{H\times W\times 3\times N}$ e retornar a estimação dos pontos de articulação $O_{P\times J\times 2\times N}$ para cada *frame*. A API fornece diferentes modelos de articulações como o COCO, MPII e BODY-25, os quais foram obtidos pelo treinamento da CNN com diferentes *datasets*. Para o experimento foi utilizado o modelo BODY-25, que dispõe de 25 pontos de articulação. Essa escolha se deu por se tratar do modelo padrão da API e por produzir os melhores resultados nos experimentos, o que foi avaliado de forma subjetiva pela análise do vídeo processado.

Extração da pessoa principal

Os primeiros vídeos de teste foram realizados em um clube de Remo e um dos desafios encontrados foi separar a pessoa principal no vídeo, enquanto haviam outras pessoas no campo de visão da câmera, ou quando o método produzia falsos positivos indicando outras pessoas.

Esse problema ocorreu pois, apesar do algoritmo de detecção separar os pontos de cada pessoa por imagem, não é garantido que de um *frame* para o outro a pessoa identificada ocupará a mesma posição no vetor resultante. Portanto, não há uma forma direta de distingui-las.

Entretanto, esse problema pode ser resolvido aplicando um método de extração da pessoa principal, que consiste em calcular a mínima área retangular ocupada pelas articulações estimadas de cada pessoa. A partir dos pontos detectados de cada pessoa, sua área A_P pode ser calculada por meio da seguinte equação:

$$A_P = (x_{max} - x_{min}) \times (y_{max} - y_{min}) \tag{3.1}$$

Onde x_{max} representa o máximo valor no eixo x, dentre as coordenadas dos pontos de articulação encontrados, e x_{min} representa o valor mínimo, seguindo a mesma lógica para o y. Uma vez calculada a área de ocupação de cada pessoa, o vetor de pontos é reorganizado e a pessoa com maior área é separada para análise, produzindo uma matriz no formato $B_{J\times 2\times N}$, a qual contém informação cinemática de apenas uma pessoa, em geral, da pessoa no plano central do vídeo.

Separação dos pontos de interesse

Considerando o movimento do remo, foram definidos seis pontos de articulação para serem utilizados para a análise: ombro, cotovelo, pulso, quadril, joelho e tornozelo. Além disso, o movimento foi reduzido ao plano sagital e, devido a sua simetria, foi analisado apenas um dos lados do corpo. Esses pontos estão demonstrados na Fig. (10) que representa um movimento no plano sagital para o lado direito.

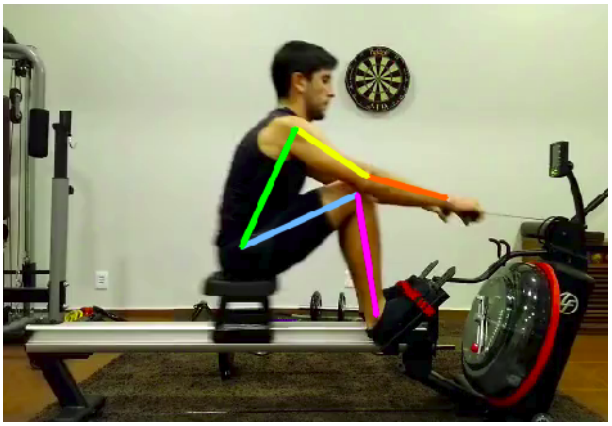


Figura 10 – Pontos de articulação no plano sagital direito.

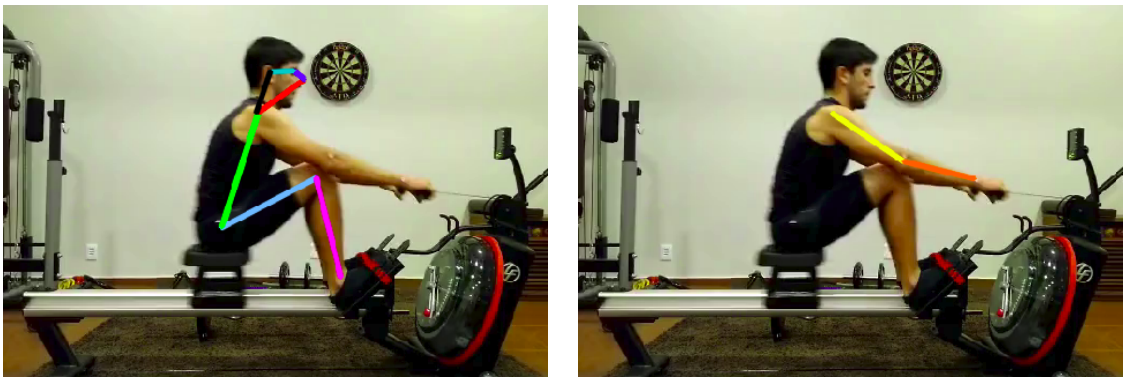
Essa etapa consistiu na remoção dos pontos não desejados, gerando uma matriz no formato $C_{J' \times 2 \times N}$, para um J' igual a 6, de acordo com o número de articulações definidas.

Filtragem

Antes de realizar a filtragem, o sistema deve lidar com os falsos negativos da rede, ou seja, os pontos de articulação não-identificados. A partir dos testes, foram identificadas duas causas principais para esse problema:

- A primeira ocorre quando o mapa de probabilidade gera valores abaixo do *threshold* definido, na região da articulação. Isso pode ocorrer por oclusão ou por limitações da rede em condições específicas.
- A segunda ocorre quando o sistema de detecção encontra o ponto, mas o caracteriza como parte de outra pessoa, como na Figura (11).

De forma a resolver esse problema, foram testadas, inicialmente, duas abordagens. A primeira foi aplicar um filtro de Kalman para estimar os pontos perdidos. A vantagem desse método está na possibilidade de ajustes dos parâmetros para o problema em questão e na possibilidade de processamento *online* a medida que os dados são adquiridos. O segundo método mais simples é a interpolação dos pontos perdidos, o que é possível considerando um processamento *offline* com os dados de todos os *frames* já disponíveis.



(a) Primeira pessoa identificada. (b) Segunda pessoa identificada.

Figura 11 – Exemplo de problema na conexão entre os pontos pelo sistema de detecção usando a OpenPose. Na imagem, a mesma pessoa foi dividida em duas.

Pelos testes foi observado que a perda do ponto de articulação, normalmente, não ocorria por mais de 10 *frames* e, portanto, sua posição não sofria um deslocamento suficiente para causar a inversão do movimento, o que prejudicaria o segundo método.

Durante os experimentos, o método de interpolação cúbica produziu os melhores resultados e, por esse motivo, foi o método escolhido para compor o sistema. A interpolação foi aplicada separadamente nos eixos x e y das coordenadas das articulações.

Em seguida foi realizada a etapa de filtragem para eliminar as oscilações de alta frequência na trajetória das articulações. As causas dessas oscilações foram mapeadas no seguintes fatores:

- Borramento da imagem devido à perda de foco da câmera com o movimento, levando o sistema a considerar uma maior região de probabilidade para a articulação e, consequentemente, produzindo alterações ou oscilações não desejadas na coordenada estimada de um *frame* para o outro. Esse problema poderia ser minimizado usando câmeras mais robustas a movimentos no vídeo.

- A segunda ocorre devido à própria funcionalidade de auto-foco da câmera, a qual produz um efeito de aproximação e afastamento da imagem das bordas para o centro, conforme o foco se ajusta para o movimento. Esse efeito é mais visível nas bordas do vídeo e produz variações nas coordenadas em *pixel* dos pontos de articulação.

- Outro fator é a forma que os resultados da rede são fornecidos. Como os *heatmaps* de saída apresentam uma dimensão menor do que a imagem de entrada, sua classificação funciona apenas para uma região da imagem e não para um *pixel*. Considerando a proporção entre as matrizes de entrada e saída, tem-se que cada ponto classificado é representativo de uma sub-região na imagem. Por esse motivo, a sobreamostra-

gem do ponto identificado tende a variar dentro da região classificada, provocando também as oscilações.

- O último fator, provavelmente o mais relevante, é a característica estocástica das CNNs, levando em consideração que o sistema foi treinado em um conjunto de dados enviesado que não produz, necessariamente, classificações generalizadas. Esse efeito é visível quando observa-se a estimação da rede para pessoas estáticas, onde a coordenada inferida de uma mesma articulação varia mesmo em *frames* com condições muito semelhantes.

Analisando o sinal gerado pela trajetória dos pontos no espaço em função dos *frames*, esses ruídos ou oscilações podem ser vistos como um acréscimo de altas frequências no sinal. Portanto, para remover esse ruído, ou para mitigar seu efeito, é comum realizar uma etapa de filtragem.

Considerando a biomecânica do movimento do remo, percebe-se que, se executado corretamente, trata-se de um movimento periódico. Para a trajetória horizontal, em específico, esse movimento também é aproximadamente senoidal com frequência definida pela cadência, ou seja, remadas em função do tempo.

Como a cadência do remo é baixa, se comparada com a frequência das oscilações, uma abordagem possível para o problema seria aplicar um filtro passa-baixas com frequência de corte definida pela cadência máxima esperada para o movimento. Isso seria suficiente para suavizar a trajetória, mas não necessariamente para corrigi-la.

Entretanto, a aplicação de um filtro de Kalman é uma estratégia mais robusta. Em seu modelo mais simples, o filtro de Kalman é um passa-baixas ideal, porém, sua vantagem está na possibilidade de ajustar as características do filtro em função da modelagem do problema em questão.

Para este trabalho, o filtro foi implementado de forma genérica considerando a trajetória de uma partícula pontual em função do tempo e em um espaço bidimensional. O vetor de estados $\mathbf{x}$ é do formato:

$$\mathbf{x} = [x, \dot{x}, y, \dot{y}] \tag{3.2}$$

onde x e y representam as coordenadas espaciais do pontos, correspondentes ao valor em *pixels* na imagem, e $\dot{x}$ e $\dot{y}$ são as componentes de velocidade em cada eixo. Já o vetor de medidas $\mathbf{z}$ tem o formato:

$$\mathbf{z} = [x, y] \tag{3.3}$$

Com isso, a matriz de medidas $\mathbf{H}$ deve ter os seguintes componentes:

$$H = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \tag{3.4}$$

De forma a simplificar a dinâmica do sistema, o movimento foi considerado retilíneo uniforme com as seguintes equações de transição:

$$\begin{aligned} x[k] &= x[k-1] + dt \cdot \dot{x}[k-1] \\ \dot{x}[k] &= \dot{x}[k-1] \\ y[k] &= y[k-1] + dt \cdot \dot{y}[k-1] \\ \dot{y}[k] &= \dot{y}[k-1] \end{aligned} \tag{3.5}$$

onde k é o índice dos vetores de posição, representando o número do *frame* e dt é o intervalo de tempo em segundos de um *frame* para o outro. Nesse caso, como a frequência de amostragem do vídeo foi de 30 fps, o intervalo de tempo dt é de $0,033$ segundos, aproximadamente.

Assim, a matriz de transições $\mathbf{F}$ modelada tem o formato:

$$F = \begin{bmatrix} 1 & dt & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & dt \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{3.6}$$

Com relação às estimativas iniciais $\mathbf{x_0}$, foi adotado:

$$\mathbf{x_0} = [x_0, 0, y_0, y] \tag{3.7}$$

sendo x_0 e y_0 as coordenadas do primeiro *frame* capturado.

O resultado dessa etapa foi uma matriz do formato $D_{J' \times 2 \times N}$ com mesma dimensão que a matriz de entrada, porém sem pontos não identificados e com as coordenadas filtradas.

Cálculo dos ângulos das articulações

Após a filtragem e processamento das coordenadas de articulação, calcula-se o ângulos das articulações. Os cinco ângulos de interesse estão ilustrados na Fig. (12) e serão usados na etapa de avaliação de movimento.

A extração dos ângulos foi realizada adotando-se as distâncias em *pixels* entre os pontos. Assumindo dois pontos A e B conectados em um ponto central O, o ângulo α entre eles pode ser encontrado por:

$$\alpha = \tan\left[\frac{O_y - B_y}{O_x - B_x}\right] - \tan\left[\frac{O_y - A_y}{O_x - A_x}\right] \tag{3.8}$$

Considerando que, no geral, o ângulo encontrado está limitado entre 0 e π, é retirado o complemento do ângulo caso ele exceda π. Entretanto, no caso do ângulo do

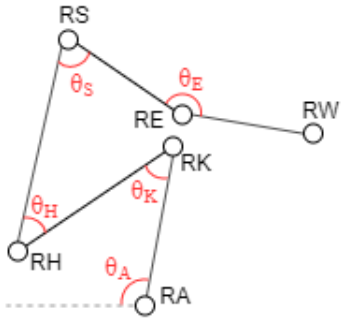


Figura 12 – Pose do remo e ângulos das articulações no plano sagital. As iniciais R representam o lado separado para o movimento, no caso o direito. As letras A, K, H, S, E e W representam *Tornozelo*, *Joelho*, *Quadril*, *Ombro*, *Cotovelo* e *Pulso*, respectivamente. O símbolo θ serve como uma variável de ângulo.

ombro, ocorre uma inversão do ângulo quando a linha do braço passa a linha do tronco. Nesse caso, o ângulo passa a ser negativo. Por isso, para o ângulo do ombro, realiza-se um teste antes de sua limitação, para saber em quais quadrantes (superiores ou inferiores) o ângulo calculado está, pois esse quadrante é alterado no momento em que ocorre a inversão. Essa alteração depende da ordem de atribuição das articulações para as variáveis A e B, assim como do lado da pose do movimento (direito ou esquerdo).

Um exemplo do resultado obtido para o ângulo de joelho está apresentado na Figura (13). Nesse caso, o ângulo estimado foi de $76,44$ graus.

Figura 13 – Ângulo de joelho obtido para o vídeo de teste.

Assim, o cálculo dos ângulos é realizado para cada *frame* N de entrada pela matriz $D_{J' \times 2 \times N}$, resultando em uma matriz $A_{K \times N}$, com K tendo dimensão 5, de acordo com o número definido de articulações.

3.2 Avaliação de Movimento

A avaliação do movimento é implementada por meio da comparação entre dois movimentos:

- O primeiro é o movimento de referência, no qual realiza-se a técnica desejada, ou seja, o padrão de remada que deseja-se seguir durante o treino, sendo representada pela trajetória dos cinco ângulos de interesse durante um ciclo de remada.
- O segundo é o movimento do alvo, ou seja, o movimento a ser avaliado, consistindo em toda a série de remadas realizadas, onde cada ciclo de remada será comparado com a referência.

Dessa forma, o sistema tem como entradas os dados dos ângulos das articulações, coletados na etapa de extração cinemática, para cada um dos movimentos definidos, como representado no diagrama da Figura (14). E assim como a saída do sistema de extração cinemática, esses ângulos são representados por matrizes $A_{rK\times N_r}$ e $A_{aK\times N_a}$, com os subíndices r e a, representando o movimento de referência e do alvo, respectivamente, e servindo para mostrar que, diferentemente da quantidade de ângulos K, a duração dos sinais N_r e N_a não precisa ser igual. Nas próximas subseções cada bloco desta etapa será detalhado.

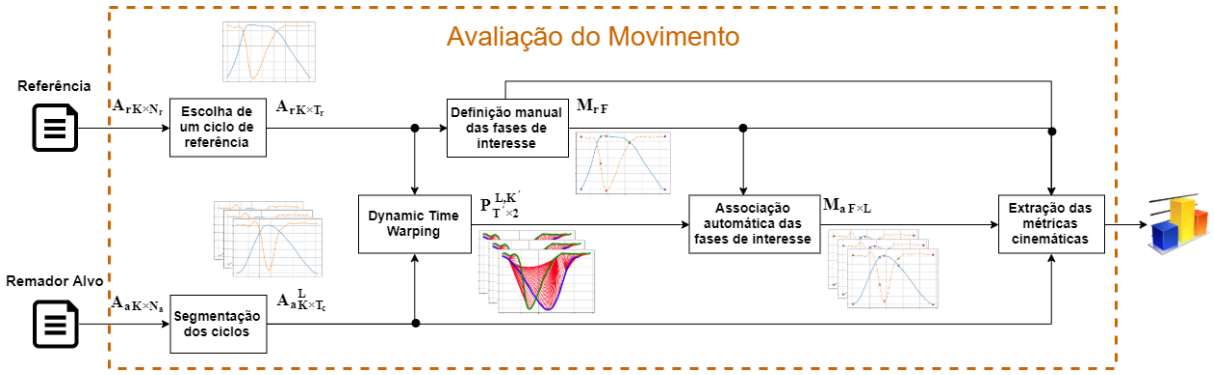


Figura 14 – Diagrama de blocos da segunda etapa de avaliação de movimento.

Segmentação dos ciclos

Considerando que a avaliação do movimento é realizada separadamente para cada ciclo de remada e que consiste em uma avaliação automática, é necessário segmentar cada ciclo a partir da série temporal dos ângulos das articulações, coletada na etapa anterior.

Uma forma simples de realizar essa segmentação seria usar a posição do banco, tendo em vista que o comportamento de sua trajetória horizontal apresenta a forma de

onda aproximadamente senoidal, o que, em geral, independe da técnica ou da pessoa que realiza o movimento. Além disso, o início e fim do ciclo de remada são bem definidos pelos picos do sinal, facilitando a segmentação.

Contudo, para não depender de um monitoramento paralelo da posição do banco, pode-se utilizar o sinal de angulação do joelho. Escolhe-se esse ângulo em específico por ser o ângulo mais relacionado com a posição do banco. Isso porque a fixação do tornozelo com o equipamento e do quadril com o banco fazem com que qualquer alteração nesse ângulo acarrete em uma alteração proporcional à posição do banco.

Assim, para o ângulo do joelho, é possível definir os momentos de início e fim do ciclo de remada como os vales ou mínimos de cada período do sinal, sendo o fim de um ciclo equivalente ao início do próximo. Dessa forma, para determinar e segmentar os ciclos, basta aplicar um algoritmo de identificação de picos e vales.

Tendo em vista que o sinal de angulação do joelho é bem comportado, no que diz respeito a estar limitado pela angulações máximas e mínimas de sua articulação (entre 30 e 180 graus, aproximadamente), é possível utilizar algoritmos de detecção de picos mais simples com base em limites ou *thresholds*.

A partir da análise do comportamento dos ângulos de joelho coletados, definiu-se um *threshold* empírico em 60 graus. Assim, para cada intervalo do sinal em que o ângulo é menor que 60 graus, define-se o valor mínimo desse intervalo como um ponto de transição entre os ciclos de remada. Em seguida, todos os intervalos entre os pontos identificados são segmentados para compor um ciclo. Um exemplo do resultado desse algoritmo está apresentado na Figura (15).

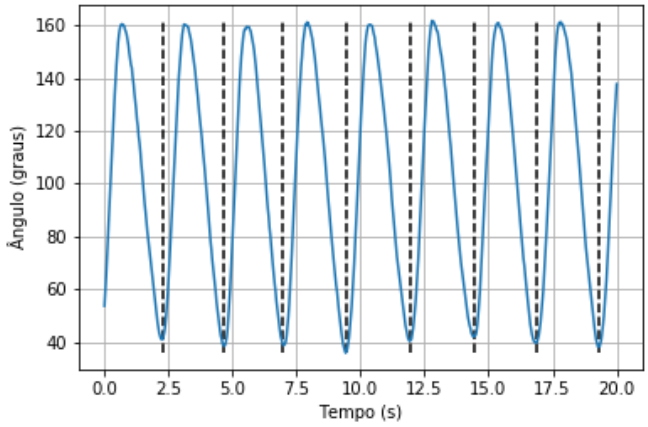


Figura 15 – Segmentação dos ciclos de remada a partir do ângulo de joelho (sinal de cor azul) usando um algoritmo por *threshold*. As linhas pontilhadas representam os limites de cada ciclo.

Um dos problemas desse algoritmo está na detecção do primeiro e último ciclo, já

que o início e fim do vídeo podem não condizer com o início e fim do ciclo de remada. Por esse motivo, optou-se por descartar ambos os ciclos.

Em uma visão geral, entra neste bloco o sinal de ângulo do alvo coletado $A_{aK\times N_a}$, onde será separado apenas o componente referente ao ângulo do joelho. Esse componente é introduzido no algoritmo de detecção de picos, resultando nas posições encontradas para cada transição de ciclo. Em seguida, segmentam-se as janelas do sinal de angulação entre essas posições, resultando em um conjunto de sinais $A^{L}_{aK\times T_c}$, onde L é o número de ciclos encontrados, K é o mesmo número de ângulos de interesse e T_c é o número de amostras de cada ciclo individual.

Escolha de um ciclo de referência

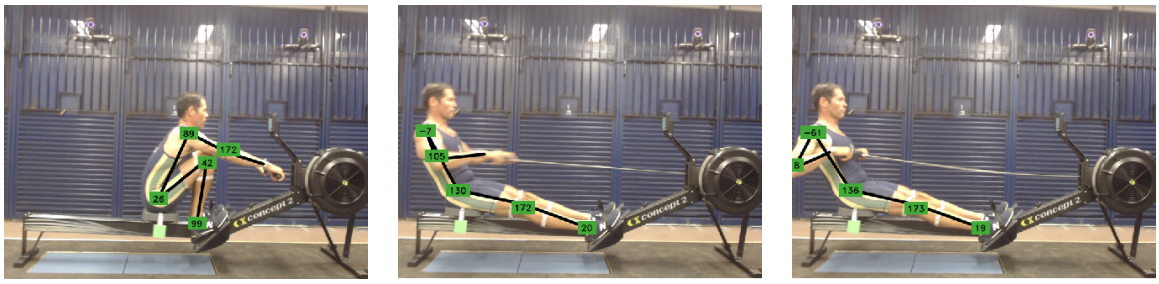
Este bloco é executado em paralelo com o bloco anterior e, somente se faz necessário se o movimento de referência contiver mais de um ciclo. Nesse caso, deve-se selecionar manualmente o ciclo de interesse para que o sinal de angulação $A_{rK\times N_r}$ se torne um sinal $A_{rK\times T_r}$ com apenas um ciclo de remada de T_r amostras.

Definição manual das fases de interesse

Nessa etapa, define-se as fases de interesse a serem avaliadas no movimento do remo. Neste caso, foram selecionadas cinco fases representativas do movimento:

- Catch (Fig. 16(a)): momento inicial da remada com a manopla mais próxima ao remo ergômetro.
- Leg Drive (Fig. 16(b)): momento de extensão completa da perna durante a fase de propulsão ou Drive.
- Arm Drive (Fig. 16(c)): momento de flexão completa do braço durante a fase de propulsão ou Drive.
- Arm Recovery (Fig. 16(d)): momento de extensão completa do braço durante a fase de recuperação ou Recovery.
- Leg Recovery (Fig. 16(e)): momento final da remada caracterizado pela flexão completa da perna durante a fase de recuperação ou Recovery e coincidindo com o momento de Catch do próximo ciclo de remada.

Então, essas fases de interesse foram definidas manualmente na remada de referência a fim de servir como base para encontrar os mesmos momentos nos ciclos segmentados do alvo.



(a) Catch. (b) Leg Drive. (c) Arm Drive.

(d) Arm Recovery. (e) Leg Recovery.

Figura 16 – As cinco fases selecionadas para análise de movimento do remo com a seguinte ordem temporal: Catch, Leg Drive, Arm Drive, Arm Recovery e Leg Recovery.

É importante notar que, apesar da informação de entrada neste bloco estar representada pela matriz de ângulo da referência $A_{rK\times T_r}$, a informação que foi realmente utilizada como base para marcar os momentos foi o vídeo, o qual foi omitido para simplificar o diagrama. Entretanto, de acordo com a definição das fases, também seria possível determinar esses momentos pelos gráficos de ângulo.

Assim, foi implementada uma interface simples para alterar e selecionar cada *frame* manualmente, com o intuito de auxiliar na definição dos momentos. Como resultado dessa interface, tem-se um vetor dos momentos de referência M_{rF}, onde F são os cinco momentos definidos e os valores representam o número do *frame* em que ocorreu aquele momento.

Dynamic Time Warping

Para comparar as sequências, o algoritmo DTW identifica a similaridade entre os sinais de referência e o alvo, podendo haver variação de tempo ou velocidade na execução da remada [3]. A similaridade pode ser detectada mesmo se a velocidade dos movimentos não forem as mesmas. Nesta etapa serão descritos os passos que foram seguidos para encontrar o alinhamento ideal e a correspondência entre dois ciclos de remada.

Como ilustrado no diagrama de blocos da Fig. (14), esse bloco tem como entrada o ciclo de referência $A_{rK\times T_r}$ e os ciclos segmentados do alvo $A^{L}_{aK\times T_c}$. Entretanto, nem todos os ângulos são utilizados nessa etapa.

Considerando que as fases de Catch e Leg Recovery já foram identificadas pelo

bloco de separação dos ciclos (em decorrência de comporem o início e o fim de um ciclo de remada) para o alvo e manualmente para a referência, esse bloco tem como objetivo encontrar apenas as fases de Leg Drive, Arm Drive e Arm Recovery dentro dos ciclos segmentados do alvo.

De acordo com a definição das fases na seção anterior, é possível observar que o Leg Drive é caracterizado apenas pela extensão do joelho e, portanto, depende apenas desse ângulo. Já o Arm Drive e o Arm Recovery dependem somente do ângulo do cotovelo, já que são caracterizados pela flexão e extensão do cotovelo, respectivamente. Por esse motivo, dentro dos sinais de entrada são separados apenas os componentes dos ângulos do cotovelo e do joelho para a análise do algoritmo, a qual será realizada individualmente.

Inicialmente, as distâncias euclidianas são encontradas para o joelho e cotovelo por meio da Eq. (2.13), onde as entradas são os conjuntos de ângulos, em graus, da referência e do alvo. As distâncias para cada articulação são dispostas em uma matriz, da qual será utilizada, posteriormente, para encontrar o *Warping path*.

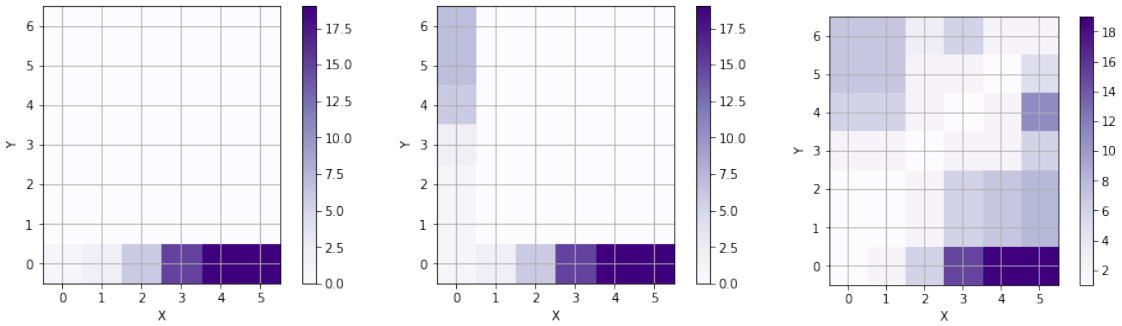
A partir dessa matriz de distâncias euclidianas, se inicia o somatório das distâncias cumulativas, ou o custo cumulativo, partindo do primeiro elemento do canto inferior esquerdo da matriz. Aplicando-se a Eq. (2.15), comparam-se os valores dos elementos adjacentes com o elemento atual para encontrar o menor valor entre eles.

A matriz é percorrida horizontal ou verticalmente, ou ainda pela diagonal, dependendo do resultado da comparação dos elementos adjacentes. Dessa forma, a matriz é percorrida até alcançar o último elemento, o elemento superior mais à direita. Este caminho é armazenado ao *path* que representa a correspondência dos índices de cada série, tanto da referência (eixo x) como do alvo (eixo y).

Na Figura (17) está representado o custo cumulativo em um *heatmap* retirado de um exemplo com vetores aleatórios, não correspondendo a nenhuma articulação, ou seja, nenhum valor angular, apenas para tornar melhor a visualização de um exemplo de alinhamento. A representação de uma articulação pode ficar prejudicada em decorrência da quantidade de *frames*, mas ainda sendo possível ver regiões de menores e maiores distâncias, assim como no exemplo ilustrado.

Nessa figura é possível verificar, em um *heatmap*, os passos que o algoritmo realiza para encontrar a diagonal aproximada. Primeiramente, percorre-se as colunas e as linhas da matrize, acumulando-se as distâncias. A representação da diagonal (Fig. 17(c)) demonstra que para as distâncias, os valores mais escuros se encontram mais distantes do ideal e os valores mais claros, mais aproximados. Portanto, a diagonal aproximada mostra através da matriz, o quão próximo, numericamente, uma série está da outra.

Posteriormente, um procedimento chamado *backtracking* é realizado, partindo do último elemento da matriz até o primeiro $(0, 0)$. Esse caminho, realizado de trás para



(a) (b) (c)

Figura 17 – *Warping path* para um conjunto de vetores aleatórios. No eixo x, o que seria o conjunto da referência, e no eixo y, o que seria o conjunto do alvo. Preenchimento matricial do custo cumulativo para (a) coluna, (b) linha e (c) distâncias resultantes.

frente, tem como parâmetro o custo cumulativo, ou seja, as distâncias acumuladas de todos os pontos.

Em uma representação utilizando DTW, um mesmo ponto de uma única série pode ser interconectado a vários outros de outra série comparativa [3]. Essas associações são pares de índices (i, j) das quais carregam as menores distâncias entre as sequências de cada articulação. Esses pares são os mencionados anteriormente como *path* e representam as correspondências entre pontos.

O resultado é uma representação "entortada", devido à correspondência baseada nas menores distâncias, e têm como coordenadas os ângulos das sequências (eixo y) e o tempo (eixo x), o que será mostrado no capítulo referente aos resultados para as articulações do joelho e do cotovelo.

Em suma, esse sistema recebe o ciclo de referência $A_{rK\times T_r}$ e os ciclos segmentados do alvo $A_{a}{}^{L}_{K\times T_c}$ e retorna L vezes K' vetores de *path*, para L sendo o número de ciclos e K' os dois componentes de ângulo avaliados (joelho e cotovelo). Dessa forma, o dado composto tem formato $P^{L,K'}_{T'\times 2}$, onde T' representa o número de associações ponto a ponto no *path* e a dimensão 2 está relacionada aos dois componentes associados.

Associação automática das fases de interesse

A associação automática das fases de interesse pode ser realizada, encontrando o *frame* do sinal segmentado equivalente para cada *frame* do sinal de referência em M_{rF}. Com as fases do Catch e Leg Recovery já estabelecidas, basta definir as três fases intermediárias. Para o Leg Drive, associa-se o *frame* na referência pela matriz do *path* para o ângulo do joelho. Já para o Arm Drive e Arm Recovery realiza-se o mesmo procedimento com a matriz *path* do ângulo do cotovelo.

Dessa forma, obtém-se um novo sinal de momentos $M_{aF\times L}$ com as cinco fases F identificadas para cada ciclo L.

Extração das métricas cinemáticas

O bloco de extração das métricas cinemáticas une todas as informações coletadas para calcular as métricas de desempenho do movimento para cada ciclo e as apresenta por meio da interface desenvolvida. Essas métricas são:

- Cadência (V): a velocidade de execução do movimento medida em remadas por minuto *rpm* (em inglês *strokes per minute*). Para calcular essa métrica basta determinar o período ou tempo de duração T_D (em segundos) do ciclo, de acordo com a seguinte relação:

$$V = \frac{60}{T_D} = \frac{60}{fps \cdot T_c} \tag{3.9}$$

  onde *fps* é a taxa de amostragem do sinal e T_c é a quantidade de *frames* no ciclo.

- Porcentagem de Drive (Drive %): quantos por cento da duração do ciclo fizeram parte do Drive, sendo Drive o intervalo entre as fases do Catch e do Arm Drive. É calculada dividindo o tempo entre esses momentos pelo T_D e multiplicando por 100.

- Porcentagem de Recovery (Recovery %): quantos por cento da duração do ciclo fizeram parte do Recovery, sendo Recovery o intervalo entre as fases do Arm Recovery e do Leg Recovery. Pode ser calculado com 100 menos o Drive %.

- Erro de articulação (E_K): diferença entre cada ângulo estimado para o alvo com seu respectivo ângulo na remada de referência.

Além da avaliação dessas métricas para cada ciclo segmentado, também serão realizadas as análises de consistência para todo o vídeo alvo, dentre elas:

- Consistência da Cadência (V): avalia-se o gráfico da cadência estimada em função dos ciclos segmentados do vídeo.

- Consistência do Ângulo/Fase (V): avalia-se o gráfico de uma determinada angulação articular para uma determinada fase em função dos ciclos segmentados do vídeo.

Protocolo do experimento

Com o intuito de testar o sistema, assim como demonstrar seu potencial de aplicação, foram realizados três vídeos de remada:

- Vídeo de referência: vídeo de cinco remadas técnicas, sem restrição de cadência, executadas por um remador profissional.
- Vídeo alvo 1: vídeo de cinco minutos de um remador iniciante para uma cadência requisitada de $24s/m$.
- Vídeo alvo 2: vídeo de cinco minutos do mesmo remador profissional para uma cadência requisitada de $24s/m$.

Durante a aquisição do vídeo de referência, o remador profissional foi instruído a realizar cinco remadas consecutivas nas quais ele julgaria ter realizado a técnica adequada. Dentre as cinco, uma delas foi separada para servir como referência. Como não foram percebidas diferenças significativas nas trajetórias das cinco remadas, foi utilizada a última remada.

Para a aquisição do primeiro vídeo alvo, o remador iniciante foi instruído a buscar uma cadência de $24s/m$ com a técnica que desejasse, apenas buscando manter consistência no movimento. O remador profissional foi instruído da mesma forma para a aquisição do segundo vídeo alvo, porém, com uma técnica que poderia ser usada em uma competição.

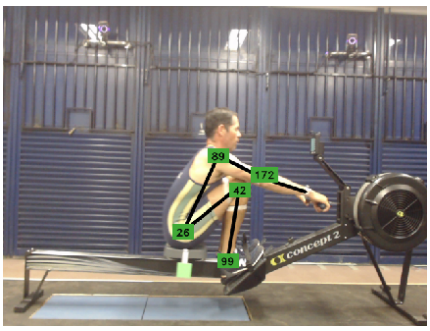
A etapa de extração cinemática e avaliação do movimento foram realizadas, separadamente, para cada vídeo alvo e usando o mesmo vídeo de referência. Contudo, nas análises de consistência os gráficos de ambos os remadores alvos foram comparados.

Uma observação importante nos vídeos coletados é que os remadores estão utilizando marcadores no mesmos pontos de articulação. Entretanto, esses marcadores foram usados apenas para aproveitar os dados da coleta em outra pesquisa com o sistema de marcadores Vicon e, portanto, não foram considerados no algoritmo de estimação de pose.

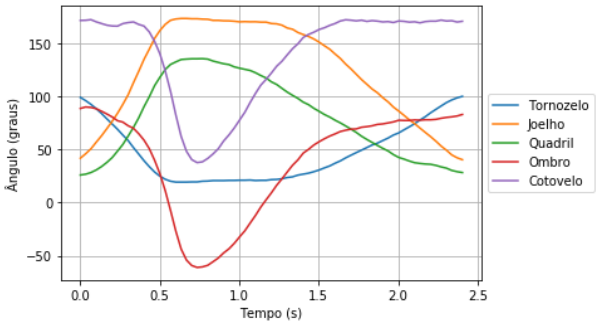
4 Resultados e Discussões

Este capítulo aborda os resultados obtidos para o experimento, enfatizando as saídas dos sistemas de extração cinemática e avaliação de movimento, assim como de alguns blocos internos. Além disso, são realizadas as discussões e considerações acerca dos resultados obtidos.

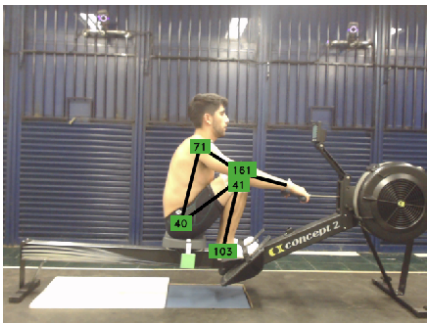
Estimação e segmentação dos ângulos de articulações



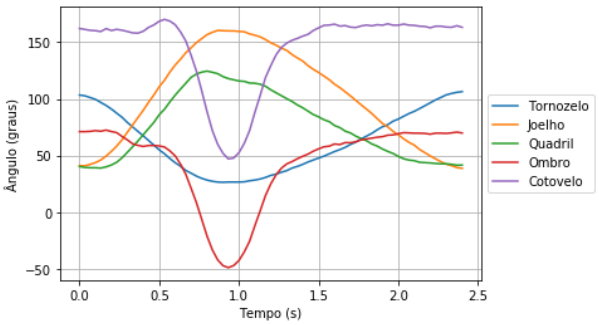
(a) Imagem dos ângulos das articulações para o remador de referência na fase do Catch.



(b) Gráfico dos ângulos de articulação para o ciclo de referência na referência.



(c) Ângulos de articulação para o remador iniciante na fase do Catch.



(d) Gráfico dos ângulos de articulação para um ciclo segmentado do movimento alvo do remador iniciante.

Figura 18 – Comparação dos sinais de ângulo de articulação do remador iniciante com a remada de referência.

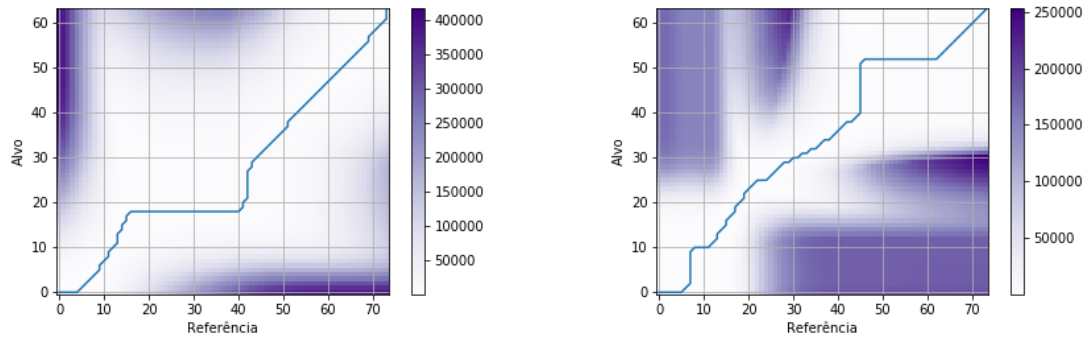
Os três vídeos coletados (remador iniciante, profissional e de referência) foram processados na etapa de extração cinemática e armazenados em arquivos contendo o vetor de posições e ângulos das articulações para cada *frame*. Em seguida, os sinais dos remadores alvos foram segmentados e verificados visualmente, para garantir que não houve nenhum falso positivo ou falso negativo.

Um exemplo dos dados obtidos está demonstrado na Fig. (18), onde pode-se observar os ângulos e posições de articulações estimadas para o vídeo de referência e para um ciclo do vídeo do remador iniciante.

Comparando-se os sinais de ângulos, é possível observar que as formas de onda entre os movimentos são semelhantes, principalmente em amplitude. Contudo, algumas características importantes podem ser avaliadas para diferenciar as técnicas, como por exemplo, as relações temporais entre o ângulo de joelho e o ângulo de cotovelo. No caso do remador profissional, o ângulo de joelho se movimenta minimamente durante o período de flexão do cotovelo, indicando uma melhor separação das fases de Leg Drive e Arm Drive, o que não é observado nos ângulos do remador iniciante, o qual começa a flexão do cotovelo antes da extensão completa do joelho.

Algoritmo DTW

A partir do conjunto de dados da referência e de um alvo, obtém-se o caminho em um *heatmap*, seguindo os passos do algoritmo. Para as articulações escolhidas, obtém-se o resultado da Fig. (19), que representa as matrizes mostrando os caminhos correspondentes ao joelho, na Fig. (19(a)) e ao cotovelo na Figura (19(b)). Ao lado dos *heatmaps* se encontram as escalas das distâncias. E os caminhos desenhados na diagonal representam as correspondências entre a referência e o alvo. As regiões mais escuras no *heatmap* indicam pontos nas matrizes onde as distâncias são maiores, por conseguinte, as regiões mais claras são referentes às pequenas distâncias.



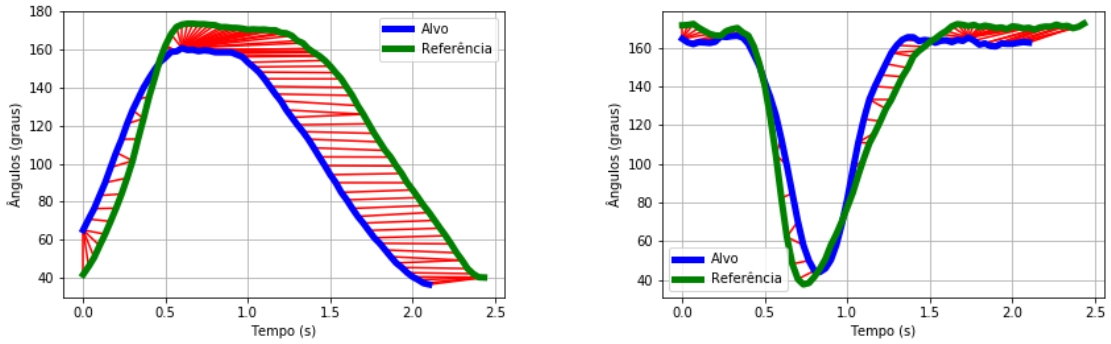
(a) (b)

Figura 19 – *Warping path* para as articulações do joelho e cotovelo. Menores distâncias e caminho para (a) joelho e (b) cotovelo.

Para ilustrar a correspondência com base nas distâncias euclidianas e, consequentemente, no custo cumulativo encontrado, é armazenado aos *paths* as menores distâncias para o joelho e o cotovelo.

Na Figura (20) estão representados os resultados dos sinais comparados no tempo de um ciclo de remada entre a referência e o alvo. No gráfico, as linhas vermelhas representam o mapeamento ponto a ponto dos *frames* entre os sinais. Nota-se que um mesmo ponto conecta-se a vários outros pontos, revelando o ajuste do algoritmo para a distorção causada no tempo.

Nota-se que na Fig. (20(a)) a partir de 0,5 segundos, os pontos associados estão agrupados verificando que um ponto do alvo se associou a outro subgrupo de pontos da referência. Essa representação em forma de "leque", mostra como o remador alvo estendeu o ângulo do joelho em menos tempo que o esperado, além de obter um ângulo menor, quando realizada a fase de Leg Drive (máxima extensão do joelho). Caso o joelho fosse estendido por mais tempo, a representação seria outra, os pontos seriam conectados em linhas retas, ou seja, diferentes pontos do alvo seriam conectados a diferentes pontos da referência. Os pontos estariam mais espaçados, assim como acontece após 1 segundo.



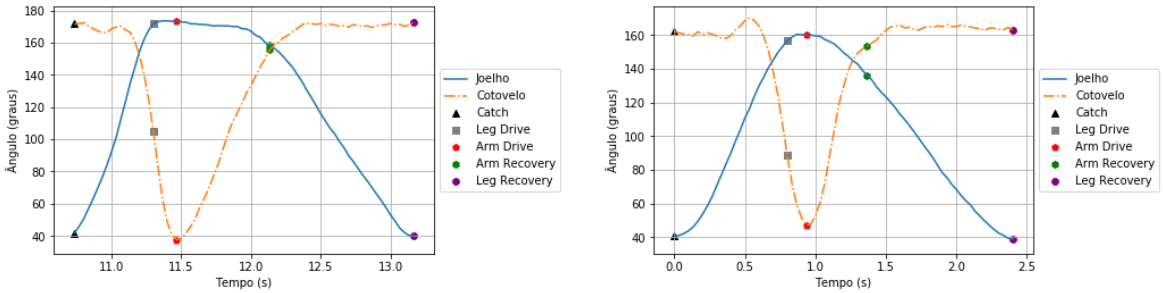
(a) (b)

Figura 20 – Correspondências entre alvo e referência para as articulações do joelho e cotovelo utilizando DTW. Em vermelho, estão os *paths* referentes às correspondências entre os pontos. Respostas obtidas para o ângulo do (a) joelho e (b) cotovelo.

Detecção automática das fases do movimento

Uma vez que a extração cinemática foi realizada, assim como a segmentação e aplicação do algoritmo DTW nos ciclos, foi possível realizar a associação do momentos da remada para o sinal alvo (Fig. 21). Para isso, o movimento de referência foi rotulado manualmente pelos *frames* do vídeo, o que está representado no gráfico da Figura (21(a)). Ao lado, o gráfico da Fig. (21(b)) apresentam-se os resultados da rotulação automática para um ciclo do remador iniciante. Novamente, os resultados de cada ciclo foram avaliados visualmente para conferir se os momentos estimados estavam de acordo com o esperado.

Observando os dados obtidos, nota-se que os momentos estimados não condizem estritamente com o que foi definido anteriormente. Por exemplo, na Fig. (21(b)) o momento do Leg Drive não foi exatamente no momento de extensão máxima do joelho. Em decorrência disso, uma pergunta pertinente é se não seria mais eficiente utilizar um algoritmo de detecção de picos para encontrar esses momentos, já que de acordo com suas definições eles sempre são caracterizados pelas condições máximas ou mínimas de algum ângulo durante o ciclo.



(a) Gráfico dos ângulos de articulação para o ciclo de referência com os momentos rotulados manualmente.

(b) Gráfico dos ângulos de articulação para um ciclo segmentado do movimento alvo do remador iniciante com os momentos rotulados automaticamente.

Figura 21 – Comparação dos momentos estimados em um ciclo de um remador iniciante com o ciclo rotulado da remada de referência.

O problema do algoritmo de detecção de picos para esta aplicação, está na maneira em que a avaliação de movimento normalmente é realizada. Em geral, a avaliação de movimento no remo é realizada de forma mais flexível, onde em vez de considerar os momentos de flexão ou extensão máxima de algum ângulo, consideram-se momentos anteriores em que a taxa de variação do ângulo está próxima de zero.

Isso é importante, pois movimentos biológicos não são precisos e, portanto, ainda que a intenção fosse manter um mesmo ângulo, (como no período entre o Leg Drive e Arm Recovery) no geral, sempre haverão pequenas oscilações do movimento (como observado no ângulo do joelho da remada de referência), principalmente em se tratando de movimentos dinâmicos como a remada (Fig. 21(a)). Além disso, o próprio erro de medição do sistema poderia gerar essas variações.

Por esse motivo, algoritmos como os de detecção de pico não serviriam para realizar essa análise mais flexível. Ainda, realizar um processamento mais robusto como o DTW permite a avaliação de outros momentos do movimento, basta mudar os momentos escolhidos no bloco de definição manual das fases de interesse. E esses momentos não precisam ser definidos pela flexão ou extensão máxima de algum ângulo.

Interface de análise para cada ciclo

Todos os dados obtidos até esta etapa foram integrados de forma a compor a interface de análise para o usuário. Nesta interface o usuário pode acessar cada *frame* processado do vídeo ou pesquisar por um ciclo/fase específica do exercício. A interface foi desenvolvida em inglês visando a escalabilidade de um possível produto futuro.

Na Figura (22) está demonstrado um exemplo do *layout* da interface para o vídeo processado do remador iniciante. Nesse caso, o momento selecionado para a análise foi a fase do Arm Drive do primeiro ciclo, condizente com o *frame* 96 do vídeo. Além das métricas de erro nos ângulos das articulações, porcentagem de *Drive*/*Recovery* e a cadência (*Pace*), também são apresentadas informações adicionais como o tempo em que o momento ocorreu no vídeo (*Video Time*), o tempo com referência ao início do ciclo (*Cycle Time*) e o período ou duração do ciclo de remada em segundos. Para o momento exemplificado na imagem, pode-se concluir que o ângulo de maior diferença para a referência, ou seja, de maior erro, foi o ângulo do quadril, o qual estava 18 graus mais estendido do que o desejado.

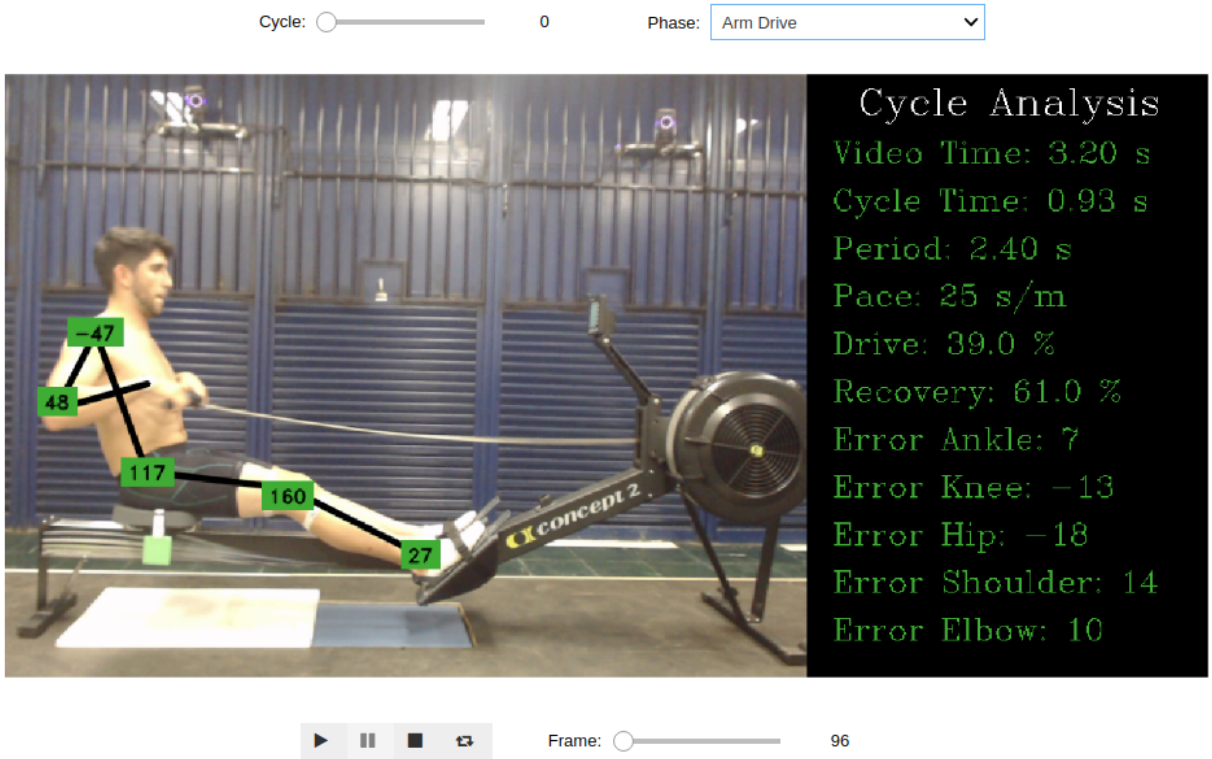


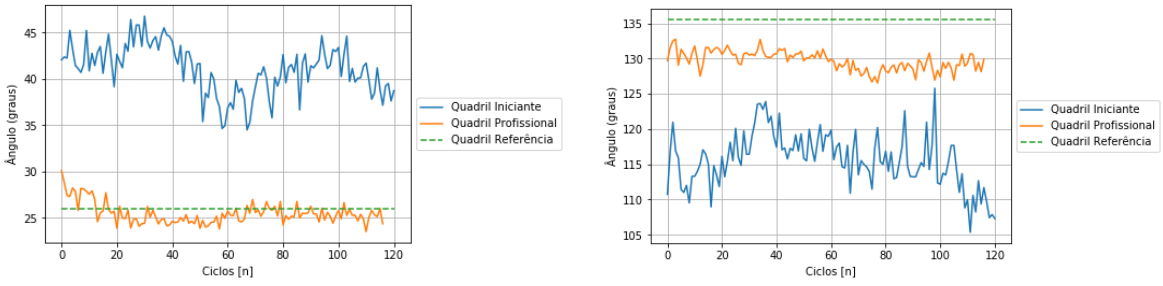
Figura 22 – Interface para a avaliação do movimento demonstrando as métricas cinemáticas calculadas automaticamente. As barras de menu superiores permitem selecionar tanto o número do ciclo de remada a ser analisado quanto a fase, enquanto as barras inferiores permitem uma seleção do *frame* específico com funcionalidades de reprodução de vídeo.

Análise de consistência entre os remadores alvo

Em contraste com a análise de um momento específico, apresentada na interface anterior, os gráficos de análise de consistência avaliam todos os ciclos do vídeo para uma fase e angulação específica. Essa análise pode ser usada em conjunto com a análise anterior, de forma que os ciclos destoantes no gráfico de consistência podem ser procurados na interface para observar as características gerais daquele ciclo.

Na Figura (23) estão representados dois gráficos de análise de consistência. O primeiro (Fig. 23(a)) representa o ângulo do quadril durante a fase do Catch. A diminuição do ângulo do quadril na fase de Catch aumenta a distância horizontal de alcance do braço e, portanto, aumenta a amplitude de movimento na propulsão. Como é possível observar, o remador profissional realizou, durante o exercício de cinco minutos, uma maior flexão do quadril no Catch e, consequentemente, uma maior amplitude quando comparado com o remador iniciante. Além disso, o remador profissional estava mais próximo do ângulo de referência e com um menor desvio padrão, ou seja, maior consistência, como esperado.

Já o segundo gráfico de consistência (Fig. 23(b)) indica o ângulo do quadril durante a fase do Arm Drive, sendo o final do movimento de propulsão. Ao contrário do caso anterior, o aumento do ângulo do quadril na fase de Arm Drive aumenta a distância horizontal de finalização do braço, o que também aumenta a amplitude de movimento. Novamente, o remador profissional realizou um movimento de maior amplitude, mais próximo do ângulo de referência e com um menor desvio padrão. Enquanto o remador iniciante realizou um movimento com ângulos menores porém, com maior diferença entre os picos e vales, tornando o movimento menos consistente e mais distante da referência.



(a) Comparação do ângulo do quadril na fase do Catch.

(b) Comparação do ângulo do quadril na fase do Arm Drive.

Figura 23 – Comparação do ângulo do quadril entre os movimentos do remador iniciante, profissional e a referência em função dos ciclos e para uma fase definida.

Nos casos apresentados para o ângulo de quadril, as diferenças entre o movimento do remador profissional e iniciante foram claras. Entretanto, isso não é observado em todos os casos. Para o gráfico de consistência do ângulo de joelho durante a fase de Arm Drive, fase onde se espera uma extensão máxima do joelho, (Fig. 24) ambos os remadores se

distanciaram da referência no decorrer dos ciclos. Observa-se uma diminuição da extensão completa do joelho após os primeiros ciclos para o remador profissional e após os últimos para o remador iniciante. A extensão desta articulação também influencia positivamente na amplitude do movimento e, apesar de ainda ser maior para o remador profissional, seu movimento não foi tão consistente, sofrendo variações superiores à 10 graus e não encontradas nos casos anteriores.

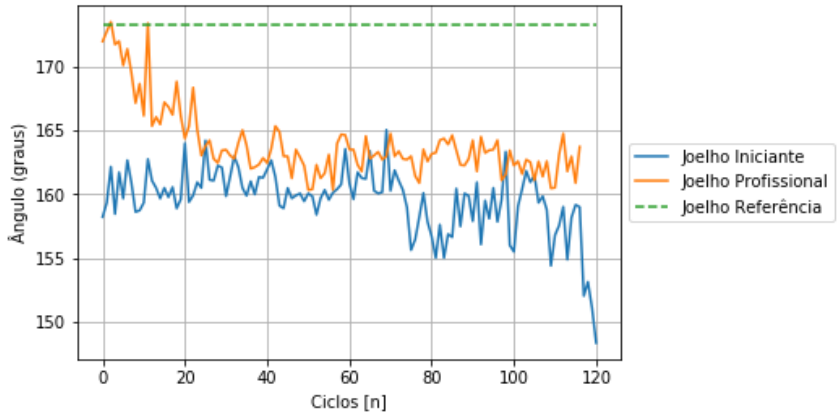


Figura 24 – Comparação do ângulo do joelho entre os movimentos do remador iniciante, profissional e a referência em função dos ciclos para fase do Arm Drive.

Por fim, é possível analisar no gráfico da Fig. (25) a consistência da cadência durante os ciclos para cada remador. Neste caso, a referência representa a cadência requisitada (24 *rpm*) ao invés da cadência do movimento de referência, que foi executado sem restrição de velocidade. No caso do remador profissional, após os primeiros 15 ciclos, aproximadamente, a cadência do movimento se manteve em média em 24 *rpm*, passando no máximo para 25. Já a cadência do remador iniciante oscilou entre 22 e 27 rpm e não melhorou após os primeiros ciclos.

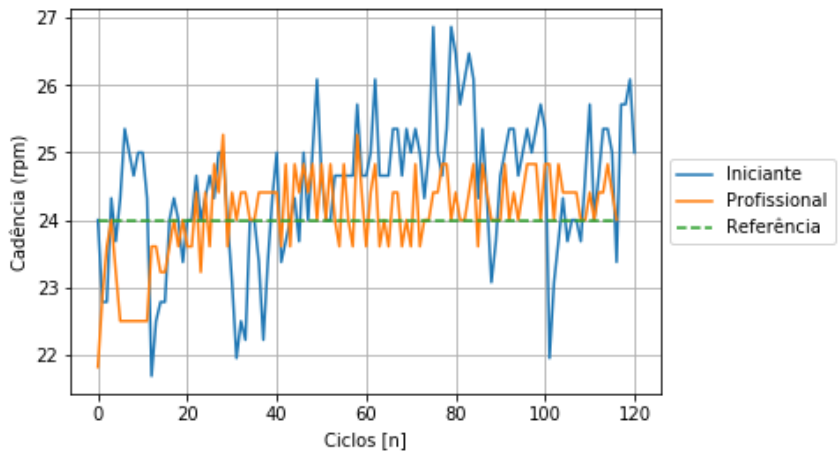


Figura 25 – Comparação da cadência entre o remador iniciante e profissional de acordo com a cadência requisitada (24 *rpm*).

Considerando o aspecto biomecânico do movimento, pessoas com diferentes relações antropométricas e distribuições de forças musculares podem não ter a mesma técnica como ideal. Portanto, é importante notar que a ideia de referência introduzida neste trabalho não está associada necessariamente à técnica ideal de remada, mas apenas a uma técnica a ser seguida a qual deve ser definida por um treinador.

Nesse caso, a referência de movimento de uma pessoa foi usada para avaliar outra, porém, outra forma possível seria usar a mesma pessoa, ainda que iniciante, para gerar seu movimento de referência. Essa situação pode ser pensada para um iniciante que não consegue realizar de forma consistente o movimento, mas algumas vezes realiza uma técnica boa. Assim, separando estas ocorrências e usando-as como referência, seria possível ter um monitoramento do ganho de consistência, de acordo com a progressão dos treinos.

5 Conclusão

Neste trabalho, foi implementado um sistema para análise biomecânica automática do movimento do remo, pela extração cinemática dos dados do vídeo. Utilizou-se o algoritmo DTW em conjunto com uma detecção de picos por *threshold* para segmentar e avaliar cada ciclo de remada e suas fases distintas. O sistema usa um vídeo de referência por um remador profissional como base para analisar um vídeo de outro remador.

O método proposto é capaz de produzir dados cinemáticos e compará-los com uma remada padrão especificada, enfatizando os desvios na execução. Isso foi testado comparando uma remada de referência do remador profissional com cinco minutos de exercício no remo ergômetro por um remador iniciante e também pelo mesmo remador profissional, onde a técnica do remador profissional se mostrou mais consistente além de estar mais próxima do padrão.

Além disso, foi demonstrado que a utilização do algoritmo DTW permitiu uma definição mais flexível dos momentos, o que não seria possível com algoritmos mais simples como o de detecção de picos.

Como proposta de avaliação biomecânica de baixo custo e tempo de configuração, foi concluído que um sistema com apenas uma câmera e sem restrição de qualidade pode gerar informações úteis para a avaliação e monitoramento de treinos de remo. Assim, o sistema pode ser usado como ferramenta de treino, considerando que, tradicionalmente, a correção de postura e *feedback* são providos, exclusivamente, pelo técnico, o que se torna inviável para múltiplas pessoas sendo treinadas por um mesmo treinador, como é comumente feito com remadores iniciantes.

6 Trabalhos Futuros

Uma restrição do trabalho presente é estar limitado a uma análise *offline*, devido às interpolações e ao algoritmo DTW. Trabalho futuros irão incluir a investigação de técnicas de estimação em tempo real.

Ainda, é importante notar que a validação da extração cinemática, ou seja, da extração dos ângulos das articulações, não foi realizada. Para isso, serão conduzidos experimentos com o sistema Vicon como padrão ouro. Assim a captura será realizada simultaneamente e de forma sincronizada entre o sistema proposto e o Vicon, comparando os ângulos obtidos entre eles.

Outra consideração relevante é que a associação dos momentos de interesse também não foi validada. Uma forma de fazer isso seria rotular manualmente os momentos desejados em cada ciclo de um treino, de acordo com os momentos definidos no ciclo de referência, e, em seguida compará-los com os momentos obtidos pelo sistema.

Referências

1 TERLEMEZ, et al. Master Motor Map (MMM) - Framework and toolkit for capturing, representing, and reproducing human motion on humanoid robots. *IEEE-RAS International Conference on Humanoid Robots*, v. 2015-Febru, n. Mmm, p. 894–901, 2015. ISSN 21640580. Citado 2 vezes nas páginas 5 e 15.

2 CAO, Z. et al. Openpose: realtime multi-person 2d pose estimation using part affinity fields. *arXiv preprint arXiv:1812.08008*, 2018. Citado 5 vezes nas páginas 5, 12, 18, 19 e 20.

3 KEOGH, E. J.; PAZZANI, M. J. Derivative Dynamic Time Warping. p. 1–11, 2001. Citado 6 vezes nas páginas 5, 24, 25, 26, 38 e 40.

4 SALVADOR, S.; CHAN, P. FastDTW: Toward Accurate Dynamic Time Warping in Linear Time and Space. *Intelligent Data Analysis*, v. 11, p. 561–580, 2007. Citado 2 vezes nas páginas 5 e 25.

5 FLETCHER, G. et al. Determining key biomechanical performance parameters in novice female rowers using the rosenberg and pose techniques during a 1 km ergometer time trial. *International Journal of Performance Analysis in Sport*, v. 15, n. 2, p. 723–748, 2015. ISSN 14748185. Citado na página 11.

6 ČERNE, T. et al. Differences between elite, junior and non-rowers in kinematic and kinetic parameters during ergometer rowing. *Human Movement Science*, v. 32, n. 4, p. 691–707, 2013. ISSN 01679457. Citado na página 11.

7 ISHIKO, T. Biomechanics of Rowing. p. 249–252, 2015. Citado na página 11.

8 SFORZA, C. A Three-Dimensional Study of Body Motion During Ergometer Rowing. *The Open Sports Medicine Journal*, v. 6, n. 1, p. 22–28, 2012. ISSN 18743870. Citado 2 vezes nas páginas 11 e 12.

9 HASE BRIAN J. ANDREWS, A. B. Z. S. E. H. K. Biomechanics of Rowing. *JSME International Journal Series C Mechanical Systems, Machine Elements and Manufacturing*, v. 45, n. 4, p. 1082–1092, 2002. Citado na página 11.

10 ČERNE, T.; KAMNIK, R.; MUNIH, M. The measurement setup for real-time biomechanical analysis of rowing on an ergometer. *Measurement*, v. 44, n. 10, p. 1819–1827, 12 2011. ISSN 02632241. Citado 2 vezes nas páginas 11 e 12.

11 GRAVENHORST, F. et al. Strap and row: Rowing technique analysis based on inertial measurement units implemented in mobile phones. *IEEE ISSNIP 2014 - 2014 IEEE 9th International Conference on Intelligent Sensors, Sensor Networks and Information Processing, Conference Proceedings*, n. April, p. 21–24, 2014. Citado na página 11.

12 TESSENDORF, B. et al. An IMU-based sensor network to continuously monitor rowing technique on the water. *Proceedings of the 2011 7th International Conference on Intelligent Sensors, Sensor Networks and Information Processing, ISSNIP 2011*, p. 253–258, 2011. Citado na página 11.

13 BOSCH, S. et al. Analysis of Indoor Rowing Motion using Wearable Inertial Sensors. In: *Proceedings of the 10th EAI International Conference on Body Area Networks*. [S.l.]: ICST, 2015. p. 233–239. ISBN 978-1-63190-084-6. Citado na página 11.

14 GRAVENHORST, F. et al. Sonicseat: A seat position tracker based on ultrasonic sound measurements for rowing technique analysis. *BODYNETS 2012 - 7th International Conference on Body Area Networks*, 2012. Citado na página 11.

15 FRANKE, T.; PIERINGER, C.; LUKOWICZ, P. How should a wearable rowing trainer look like? A user study. *Proceedings - International Symposium on Wearable Computers, ISWC*, p. 15–18, 2011. ISSN 15504816. Citado na página 11.

16 BINGUL, B. M. et al. Two-dimensional kinematic analysis of catch and finish positions during a 2000m rowing ergometer time trial. *South African Journal for Research in Sport, Physical Education and Recreation*, v. 36, n. 3, p. 1–10, 2014. ISSN 03799069. Citado na página 12.

17 SKUBLEWSKA-PASZKOWSKA, M. et al. Motion Capture As a Modern Technology for Analysing Ergometer Rowing. *Advances in Science and Technology Research Journal*, v. 10, n. 29, p. 132–140, 2016. ISSN 2080-4075. Citado na página 12.

18 FOTHERGILL, S.; HARLE, R.; HOLDEN, S. Modeling the model athlete: Automatic coaching of rowing technique. *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, v. 5342 LNCS, p. 372–381, 2008. ISSN 03029743. Citado na página 12.

19 GÜLER, R. A.; NEVEROVA, N.; KOKKINOS, I. DensePose: Dense Human Pose Estimation in the Wild. *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, p. 7297–7306, 2018. ISSN 10636919. Citado 2 vezes nas páginas 12 e 18.

20 ASTERIADIS, S. et al. Estimating human motion from multiple Kinect sensors. In: *ACM International Conference Proceeding Series*. [S.l.: s.n.], 2013. ISBN 9781450320238. Citado na página 13.

21 PAPADOPOULOS, G. T.; AXENOPOULOS, A.; DARAS, P. Real-time skeleton-tracking-based human action recognition using kinect data. In: *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*. [S.l.: s.n.], 2014. v. 8325 LNCS, n. PART 1, p. 473–483. ISBN 9783319041131. Citado na página 13.

22 SHUM, H. P. et al. Real-time posture reconstruction for Microsoft Kinect. *IEEE Transactions on Cybernetics*, v. 43, n. 5, p. 1357–1369, 10 2013. ISSN 21682267. Citado na página 13.

23 MEHTA, D. et al. VNect: Real-time 3D human pose estimation with a single RGB camera. In: *ACM Transactions on Graphics*. [S.l.]: Association for Computing Machinery, 2017. v. 36, n. 4. Citado 2 vezes nas páginas 13 e 18.

24 WANG, K. et al. 3D Human Pose Machines with Self-supervised Learning. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, p. 1–1, 2019. ISSN 0162-8828. Citado 2 vezes nas páginas 13 e 18.

25 GONG, W. et al. Human pose estimation from monocular images: A comprehensive survey. *Sensors (Switzerland)*, v. 16, n. 12, p. 1–39, 2016. ISSN 14248220. Citado na página 15.

26 CORAZZA, S. et al. A markerless motion capture system to study musculoskeletal biomechanics: Visual hull and simulated annealing approach. *Annals of Biomedical Engineering*, v. 34, n. 6, p. 1019–1029, 2006. ISSN 00906964. Citado na página 16.

27 TOSHEV, A.; SZEGEDY, C. DeepPose: Human Pose Estimation via Deep Neural Networks. p. 1653–1660, 2014. Citado na página 16.

28 HORNIK, K.; STINCHCOMBE, M.; WHITE, H. Multilayer feedforward networks are universal approximators. *Neural Networks*, v. 2, n. 5, p. 359–366, 1989. ISSN 08936080. Citado na página 17.

29 PAN, S. J.; YANG, Q. A survey on transfer learning. *IEEE Transactions on Knowledge and Data Engineering*, IEEE, v. 22, n. 10, p. 1345–1359, 2010. ISSN 10414347. Citado na página 18.

30 REDMON, J. S. D. R. G. A. F. (YOLO) You Only Look Once. *Cvpr*, 2016. ISSN 01689002. Citado na página 18.

31 XIU, Y. et al. Pose flow: Efficient online pose tracking. *British Machine Vision Conference 2018, BMVC 2018*, p. 1–12, 2019. Citado na página 18.

32 ANDRILUKA, M. et al. PoseTrack: A Benchmark for Human Pose Estimation and Tracking. *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, p. 5167–5176, 2018. ISSN 10636919. Citado na página 18.

33 LIN, T.-Y. et al. Microsoft COCO: Common Objects in Context. 5 2014. Citado na página 18.

34 SIMONYAN, K.; ZISSERMAN, A. Very deep convolutional networks for large-scale image recognition. *3rd International Conference on Learning Representations, ICLR 2015 - Conference Track Proceedings*, p. 1–14, 2015. Citado na página 19.

35 FALLIS, A. *Optimal State Estimation - Kalman, H infinity, and Nonlinear Approaches*. [S.l.: s.n.], 2013. v. 53. 1689–1699 p. ISSN 1098-6596. ISBN 9788578110796. Citado na página 23.

36 WENG, S. K.; KUO, C. M.; TU, S. K. Video object tracking using adaptive Kalman filter. *Journal of Visual Communication and Image Representation*, v. 17, n. 6, p. 1190–1208, 2006. ISSN 10473203. Citado na página 23.

37 RHUDY, M. B.; SALGUERO, R. A.; HOLAPPA, K. A Kalman Filtering Tutorial for Undergraduate Students. *International Journal of Computer Science & Engineering Survey*, v. 08, n. 01, p. 01–18, 2017. ISSN 09763252. Citado na página 23.

38 MUDA, L.; BEGAM, M.; ELAMVAZUTHI, I. Voice Recognition Algorithms using Mel Frequency Cepstral Coefficient (MFCC) and Dynamic Time Warping (DTW) Techniques. *JOURNAL OF COMPUTING*, v. 2, n. 3, 2010. Citado na página 24.

39 RAAJ, Y. et al. Efficient Online Multi-Person 2D Pose Tracking with Recurrent Spatio-Temporal Affinity Fields. 11 2018. Citado na página 27.

40 SZŰCS, G.; TAMÁS, B. Body Part Extraction and Pose Estimation Method in Rowing Videos. *Journal of Computing and Information Technology*, v. 26, n. 1, p. 29–43, 2018. ISSN 13301136. Citado na página 27.

41 CAO, Z. et al. Realtime multi-person 2D pose estimation using part affinity fields. *Proceedings - 30th IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017*, v. 2017-Janua, n. Xxx, p. 1302–1310, 2017. ISSN 10636919. Citado na página 27.

42 BAPTISTA, R. D. S.; BÓ, A. P.; HAYASHIBE, M. Automatic Human Movement Assessment with Switching Linear Dynamic System: Motion Segmentation and Motor Performance. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, v. 25, n. 6, p. 628–640, 2017. ISSN 15344320. Citado na página 27.