

# Visualizador de marcha humana com tecnologia WebGL

Alexandre Pereira de Faria  
Fabio Alencar Schneider

## Resumo

A marcha humana tem sido objeto de estudo em diversas áreas como medicina, fisioterapia, educação física, engenharia mecânica, robótica e computação gráfica. Neste trabalho apresenta-se uma proposta de construção de um aplicativo para web que deve permitir a visualização de dados da captura de movimentos da marcha humana. Num primeiro momento foi desenvolvido um protótipo que permite a visualização de dados de membros inferiores em marcha tendo em vista a necessidade de ser executado em navegadores da web, ou seja, sem a necessidade de instalação. Uma análise de aplicativos similares para a visualização da marcha disponibilizados gratuitamente na web permitiu a identificação de tecnologias utilizadas e recursos de interação com usuário. O protótipo foi implementado por meio das linguagens de criação de páginas como a linguagem de marcação HTML e de folha de estilos CSS, recursos interativos com JavaScript e visualização otimizada dos dados com a API WebGL.

**Palavras-chave:** Marcha humana, visualização científica, WebGL.

## **Introdução**

O estudo da marcha humana tem sido objeto de estudo em diversas áreas como medicina, fisioterapia, educação física, engenharia mecânica, robótica e computação gráfica. Na biomecânica, que integra pesquisadores daquelas áreas, a simulação e análise de dados do movimento por meio de aplicativos computacionais têm auxiliado profissionais de várias especialidades contribuindo no diagnóstico de patologias e prescrição terapêutica (TRILHA JR., 2006).

Conforme Barbosa (2011), Giovanni Borelli (1608 – 1679) foi o pioneiro estudo biomecânico da marcha. No século XIX, Marey e Muybridge realizaram o estudo cinemático e cinético do movimento utilizando fotografias. No século XX os dispositivos eletrônicos e computacionais tornaram o registro do movimento mais rápido e acurado fornecendo um conjunto de técnicas e instrumentos que ajudaram a fundar a metodologia clínica conhecido como análise da marcha. Na análise da marcha os movimentos dos membros envolvidos na marcha são descritos quantitativa e qualitativamente por meio de parâmetros cinemáticos, como a posição no espaço, e dinâmicos, como a distribuição de cargas.

As tecnologias de captura de movimentos baseada em imagens de vídeo têm sido popularizadas a partir da sua utilização na indústria do entretenimento como o cinema e os videogames. Enquanto dispositivos como o Kinect e o Wii levaram a tecnologia para dentro das casas dos usuários finais, outros sistemas, mais acurados, como o Vicon são utilizados por estúdios e centros de pesquisa. Bancos de dados de movimentos como, por exemplo, o Graphics Lab Motion Capture Database da Carnegie Mellon University, são disponibilizados gratuitamente na web em formatos como o .bvh, da Biovision ou o .c3d da Gait and Biomechanical Community. Embora estes dados possam ser acessados livremente, a visualização dos mesmos depende de aplicativos especializados para cada formato ou de softwares de modelagem 3D que permitam importar e ligar estes dados a um esqueleto ou armadura.

Neste trabalho é apresentada uma proposta de construção de um aplicativo para web que deve permitir a visualização de dados da captura de movimentos da marcha humana. Num primeiro momento foi desenvolvido um protótipo que permite a visualização de dados de membros inferiores em marcha com parametrização em coordenadas cartesianas. Os dados foram obtidos por meio de experimento de captura da marcha

sobre esteira com utilização de marcadores não reflexivos sobre as articulações dos membros inferiores conforme Barbosa (2011) e Brond (2009) e as imagens obtidas foram processados por meio do aplicativo Skillspector da VideoCoach (LOPES *et al*, 2015).

## **Tecnologias e Similares**

Para a elaboração do protótipo foram realizadas pesquisas sobre aplicativos de visualização da marcha similares disponíveis na web. A análise de similares permitiu a identificação de tecnologias utilizadas e recursos disponíveis. O protótipo do aplicativo de visualização será desenvolvido tendo em vista a necessidade de ser executado em navegadores da web, ou seja, sem a necessidade de instalação. Para tanto foram estudadas linguagens de criação de páginas como a linguagem de marcação HTML e de folha de estilos CSS, estruturação dos dados e recursos interativos com JavaScript e visualização otimizada dos dados com a API WebGL. O aplicativo deve prover ao usuário a manipulação do modelo 3D por meio translação, rotação e escala.

Conforme a página do desenvolvedor Khronos Group, a WebGL é uma API (Interface de Programação de Aplicativos), *free-royalties* e multi-plataforma. Baseada na OpenGL ES 2.0, a WebGL permite a exibição de conteúdo interativo 3D por meio de sua integração com os recursos das linguagens de programação JavaScript e o elemento *canvas* do HTML. Os gráficos definidos no ambiente WebGL dos elementos *canvas* podem ser visualizados diretamente nos navegadores mais comuns como o Chrome (Google), Firefox (Mozilla) e Safari (Apple).

A pesquisa e análise de similares foi focada em aplicativos web e permitiu a identificação de tecnologias e recursos que poderiam ser empregadas para a elaboração do visualizador objeto deste estudo. Para este trabalho foram selecionados três aplicativos disponíveis gratuitamente na web: o BMLWalker e WebGLWalker da BioMotionLab e o Kineman da NeoSim R&D.

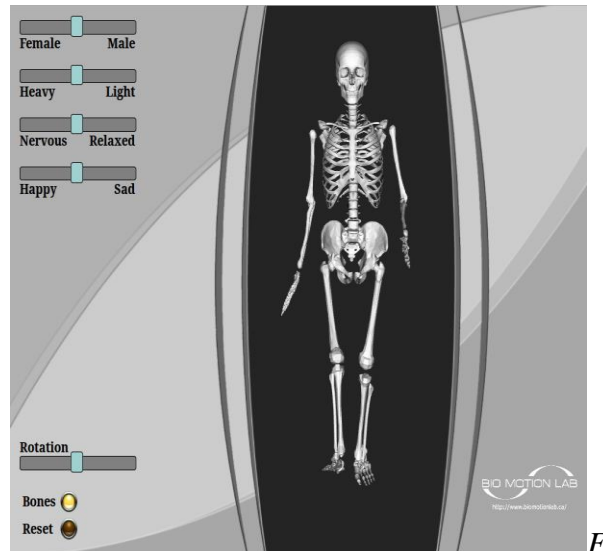
O BMLWalker (Figura 1) é um visualizador da marcha construído em Flash e permite a visualização de padrões da marcha humana a partir de um banco de dados de captura de movimentos de 20 homens e 20 mulheres. Os movimentos da marcha sobre esteira ou chão foram capturados por meio sistema ótico Vicon e classificados segundo características físicas como gênero (homem – mulher) e peso (gordo – magro) e

características comportamentais associados a estados de espírito (estressado – relaxado ou feliz – triste). Após a normalização dos dados, os desenvolvedores implementaram procedimentos de interpolação permitindo ao usuário criar combinações de marcha por meio da manipulação daquelas características. Outros recursos disponíveis por meio de botões permitem iniciar e pausar a animação, voltar a configuração inicial e alternar a visualização por meio de pontos nas posições dos marcadores de capturas ou linhas entre estes pontos. A manipulação da área de visualização pode ser feita diretamente por meio do ponteiro do mouse permitindo a rotação do modelo que retorna suavemente à posição frontal inicial.



*Figura 1. BMLWalker*

O segundo visualizador da Biomotionlab é o WebGLWalker (Figura 2), o qual, como o nome diz, foi implementado por meio da API WebGL e recursos da biblioteca JQuery. Nesta versão os desenvolvedores incluíram o recurso de visualização por meio de ossos (*bones*). Os recursos de interpolação entre as características dos sujeitos, botão de reset e alternância entre a visualização de pontos e ossos foram mantidos. Apenas o recurso de manipulação da área de visualização por meio da rotação do modelo foi alterado para um controle deslizante.



*Figura 2. WebGL Walker*

Embora não seja um visualizador da marcha, o terceiro aplicativo foi implementado por meio da WebGL e, por este motivo, consta nesta análise. O Kineman (KineMan Movable 3D Human Skeleton), versão básica, (Figura 3) é um visualizador dinâmico de poses de um modelo de esqueleto humano. Definido pelos desenvolvedores como uma ferramenta acurada de manipulação do modelo respeitando a hierarquia entre os ossos. O aplicativo permite rotar o modelo tanto por meio da área de visualização quanto por meio de controles deslizantes. As operações de translação e escala, no entanto, estão habilitadas somente por meio dos controles deslizantes. Do mesmo modo, após a seleção dos elementos a serem manipulados, o usuário pode dar a entrada de dados tanto por meio da área de visualização quanto por meio de controles deslizantes adicionais que são criados para cada osso ou articulação.

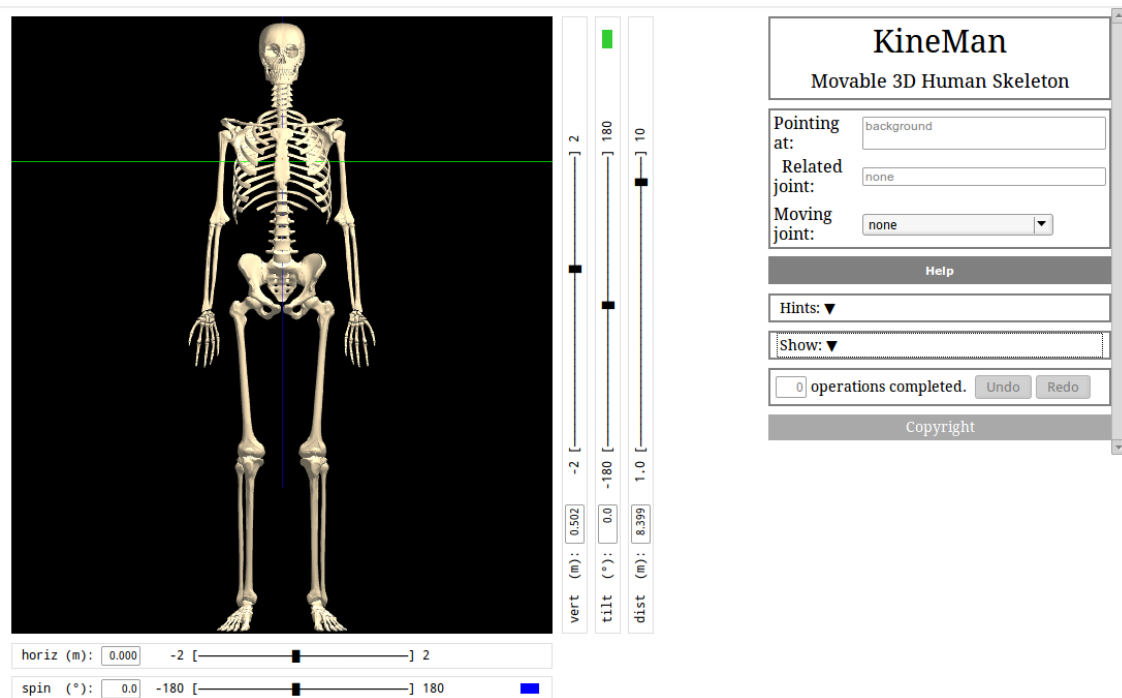


Figura 3. Kineman

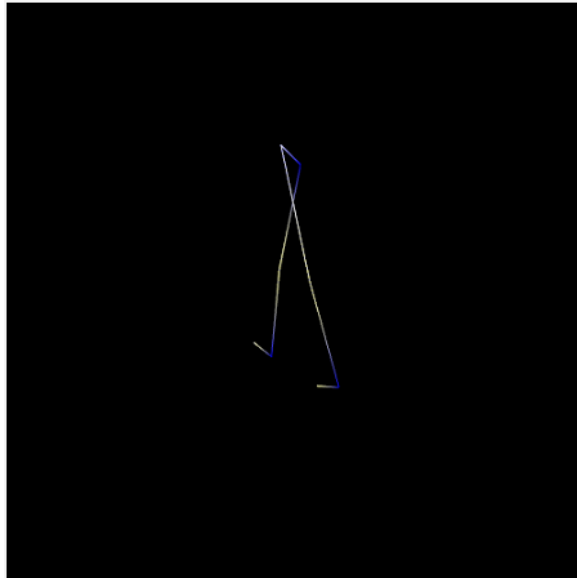
## Desenvolvimento e Resultado

A implementação do código foi realizada em quatro partes:

1. Definição da estrutura da página por meio das linguagens HTML e CSS.
2. Definição do ambiente WebGL por meio do elemento *canvas* HTML.
3. Definição da estrutura dos dados da marcha.
4. Definição do código com recursos de animação e de interação com o usuário.

O protótipo elaborado (Figura 4) foi simplificado para apresentar apenas a estrutura de arame (*wireframe*) do sistema biomecânico dos membros inferiores. Este sistema foi desenhado a partir de pontos correspondentes em cada perna as articulações do dedão do pé (hálux), a articulação do tornozelo (maléolo lateral), a articulação do joelho (epicôndilo lateral) e a articulação do quadril (espinha ilíaca). Estes pontos já haviam sido utilizados para a construção de outro visualizador por meio do aplicativo LabView (SCHNEIDER e FARIA, 2015) (Figura 5). Para este trabalho o código foi testado em dois navegadores de internet, o Firefox e o Chrome, e em dois sistemas operacionais diferentes, Windows 7 e Ubuntu 12.

### Simulador de Marcha Humana Normal com WebGL



Rotação em torno do eixo X : 0°  360°

Translação ao longo do eixo X : -  +

Figura 4: Protótipo com WebGL

A renderização dos pontos e interpolação linear dos pontos por meio da função `GL_LINE` da biblioteca WebGL permitiram a visualização da estrutura de arame. Os recursos de interação foram implementados com sucesso, porém ficaram limitados as transformações de translação em relação ao eixo horizontal e rotação do sistema em relação ao eixo vertical.

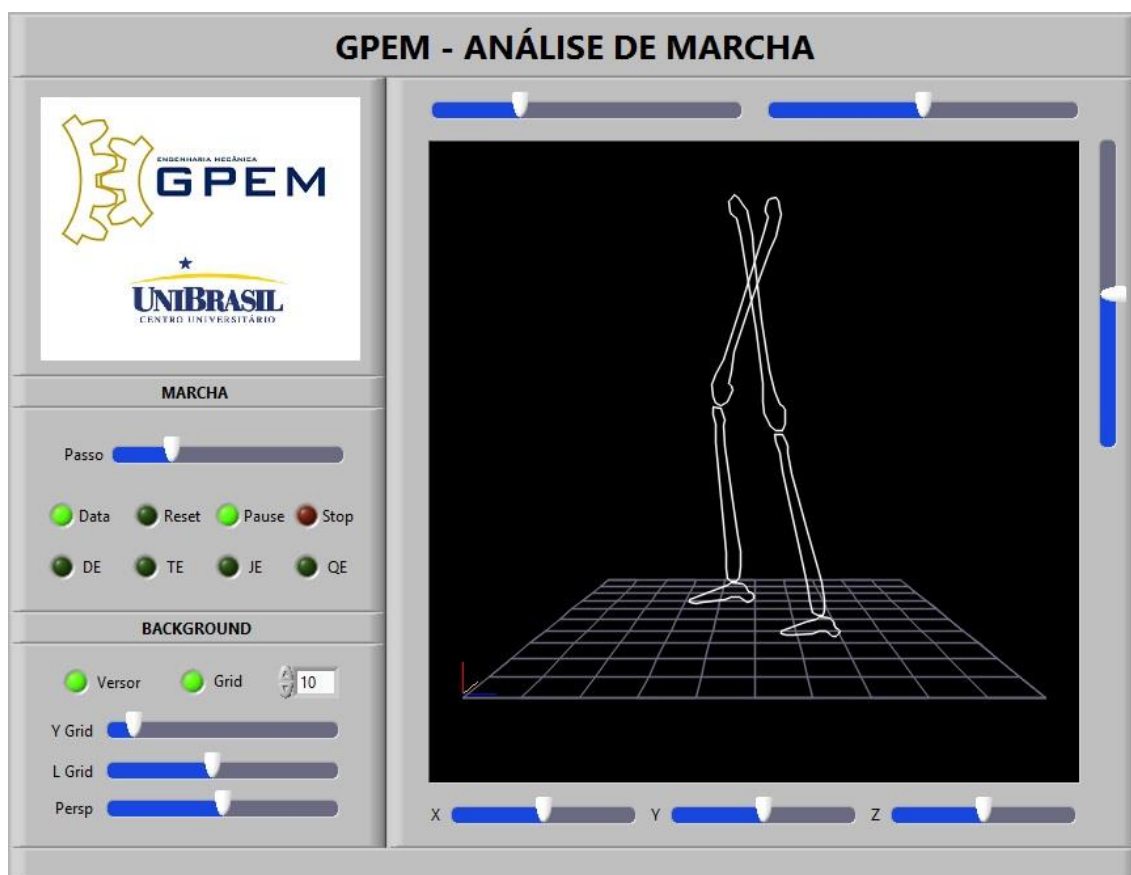


Figura 5: Protótipo desenvolvido em LabView

A atualização da animação por meio das funções `SetTimeout` e `setInterval` revelaram-se problemáticas apresentando variações em relação a velocidade da animação conforme o navegador utilizado e o sistema operacional. Para resolver este problema um segundo código foi implementado com substituição das funções acima pela alternativa `requestAnimationFrame`, porém o mesmo problema foi detectado.

Os problemas em relação as funções de animação têm sido reportados em diversos sites como o [Stackoverflow.com](https://stackoverflow.com), entretanto, no caso do visualizador em pauta acredita-se que o problema principal pode estar ocorrendo devido a atribuição de novos valores as variáveis de posição do modelo a cada nova iteração. Esta atualização das posições do modelo pode estar congestionando a memória e causando os problemas na visualização. Uma possível solução para estes problemas pode ser investigada a partir do uso da biblioteca `Three.js` ([threejs.org](https://threejs.org)), que é baseada na `WebGL`, o que demandaria, além da reelaboração do código, uma nova estrutura dos dados cinemáticos da marcha.



## Considerações Finais

Visualizadores do movimento humano tem sido disponibilizados por vários desenvolvedores. Contudo estes visualizadores, além de precisarem serem instalados nos computadores, carregam tipos específicos de arquivos de *motion capture*. Neste trabalho foi apresentado a proposta de um visualizador iterativo da marcha humana para aplicativos web. Foram estudados aplicativos similares disponíveis na web e as tecnologias utilizadas em sua elaboração. Uma mesma proposta foi elaborada pelos autores em outro trabalho utilizando recursos da programação visual dentro do ambiente de desenvolvimento do LabView. O visualizador objeto deste trabalho foi implementado por meio das linguagens de programação HTML, CSS, JavaScript com recursos da API WebGL. O resultado final foi considerado satisfatório, embora problemas em relação as funções de animação tivessem sido detectadas. O desenvolvimento futuro desta pesquisa deve ser pautado na investigação de bibliotecas livres, como a Three.js, que ofereçam a solução aos problemas encontrados. Além disto espera-se que a visualização de um modelo 3D do sistema biomecânico com outros recursos de iteração possa estar disponível nas novas versões. Tanto o visualizador elaborado em LabView quanto este em WebGL fazem parte do Projeto de Modelagem e Simulação do Desgaste de Próteses de Joelho do Grupo de Pesquisa em Engenharia Mecânica do UniBrasil.

## Referências bibliográficas

BARBOSA, C. P. F. M.. **Modelação biomecânica do corpo humano: aplicação na análise da marcha**. Dissertação de Mestrado em Engenharia Biomédica apresentada à Universidade do Minho. Braga, 2011.

BiomotionLab. **WebGLWalker**. Disponível em

<[https://www.biomotionlab.ca/Demos/webgl\\_walker/](https://www.biomotionlab.ca/Demos/webgl_walker/)> Acesso em: 23 de agosto de 2016

\_\_\_\_\_. **BMLWalker**. Disponível em <<https://www.biomotionlab.ca/>>. Acesso em: 23 de agosto de 2016

BROND, Jan. **Biomechanics made simple..... SkillSpector**. Video4Coach. 2009.

Disponível em <  
<http://video4coach.com/imagesfile/SkillSpector%20Documentation%20English.pdf>>.  
Acesso em: 23 de agosto de 2016

Khronos Group Inc. **History of OpenGL**. 2012. Disponível em:  
<<http://www.opengl.org/wiki/History>>. Acesso em: 23 de agosto de 2016.

Kineman. **KineMan Movable 3D Human Skeleton**. Disponível em  
<[www.KineMan.com](http://www.KineMan.com)> Acesso em: 23 de agosto de 2016

LOPES, A. H. F.; SCHNEIDER, F. A.; FARIA, A. P.; SOUZA, R. A.. Modelagem de marcha e simulação de desgaste em prótese de joelho. EVINCI 2015: Evento de Iniciação Científica do Centro Universitário Autônomo do Brasil, Disponível em:  
<<http://portaldeperiodicos.unibrasil.com.br/index.php/anaisevinci/article/view/990>>. Acesso em 23 de agosto de 2016.

SCHNEIDER, F. A. e FARIA, A. P.. Software de simulação de marcha em LabView. EVINCI 2015: Evento de Iniciação Científica do Centro Universitário Autônomo do Brasil, Disponível em:  
<<http://portaldeperiodicos.unibrasil.com.br/index.php/anaisevinci/article/view/995>>. Acesso em 23 de agosto de 2016.

TRILHA JR, M.. **Construção e validação qualitativa de um modelo de elementos finitos para a simulação mecânica do joelho humano**. Dissertação de Mestrado em Engenharia Mecânica apresentada à Universidade Federal de Santa Catarina Florianópolis, 2006.

Three.js. Disponível em <<https://threejs.org/>> Acesso em: 23 de agosto de 2016