

CONTROLE DE JOGOS POR GESTOS COM VISÃO COMPUTACIONAL - IMPLEMENTAÇÃO USANDO *MEDIAPIPE* E *OPENCV*

**Eduardo Augusto Goetz, João Pedro Oliveira Torres, Julio Samuel Rodrigues,
Vinícios Matos Pellin e Larissa Daiane Caneppele Guder**

Faculdade Três de Maio (SETREM), Três de Maio, RS, Brasil.

RESUMO

Este trabalho apresenta o desenvolvimento de uma interface gestual baseada em visão computacional para controle do jogo *Geometry Dash*, utilizando apenas uma webcam e as bibliotecas *MediaPipe* e *OpenCV*. A proposta visa proporcionar uma alternativa acessível e portátil ao uso de periféricos físicos, por meio do reconhecimento de gestos como mão aberta, punho fechado e movimento de pinça. A metodologia adotada incluiu o rastreamento de mãos em tempo real, identificação de padrões gestuais e integração com o jogo via simulação de eventos de teclado. Os testes demonstraram uma taxa de reconhecimento de aproximadamente 95,7% em ambientes controlados, com baixa latência e resposta fluida durante a jogabilidade. Os resultados indicam o potencial da solução para aplicações em jogos, educação e acessibilidade, mesmo sem sensores dedicados.

Palavras-chave: *MediaPipe*. *OpenCV*. Reconhecimento. Jogos.

1 INTRODUÇÃO

O crescimento das interfaces naturais têm impulsionado novas formas de interação em jogos digitais, superando a dependência de teclado, mouse e controles físicos. Com os avanços em visão computacional e inteligência artificial, tecnologias como *MediaPipe* e *OpenCV* permitem reconhecer gestos e movimentos corporais em tempo real, possibilitando sua tradução em comandos computacionais. Este trabalho propõe a implementação de um sistema de controle baseado em gestos capturados por câmera, explorando sua viabilidade para promover acessibilidade, inovação e novas experiências de usuário.

2 FUNDAMENTAÇÃO TEÓRICA

A visão computacional, subárea da Inteligência Artificial, é fundamental para o reconhecimento automático de gestos e posturas a partir de imagens ou vídeos. Ferramentas como o *MediaPipe* e o *OpenCV* têm se consolidado nesse cenário. O *MediaPipe*, desenvolvido pelo Google, é uma biblioteca *open-source* que cria *pipelines* modulares usando aprendizado de máquina, operando eficientemente em diversas plataformas. Sua arquitetura baseada em grafos permite paralelismo e

reutilização, viabilizando aplicações em jogos, educação e reabilitação física (Lugaresi et al., 2019). Já o *OpenCV* é uma biblioteca consolidada para processamento de imagens e vídeos, permitindo operações como segmentação, detecção de contornos e visualização em tempo real (Bradski, 2000).

O *MediaPipe Hands* é um módulo consolidado que realiza rastreamento de mãos em tempo real, detectando 21 pontos-chave por mão a partir de imagens RGB, sem sensores adicionais (MediaPipe, 2024). Utilizando uma rede SSD MobileNetV2 (Single Shot MultiBox Detector) para detectar palmas e uma segunda etapa para regressão dos landmarks, o módulo é amplamente usado no controle de jogos digitais devido à sua precisão e baixa latência.

O *MediaPipe Holistic* expande essa capacidade ao integrar pose corporal, face e mãos em um único pipeline, utilizando modelos como GHUM (Generative Human Model) para manter a consistência estrutural do corpo humano (Bazarevsky et al., 2022).

3 METODOLOGIA

Este trabalho foi desenvolvido com o objetivo de criar uma interface gestual baseada em visão computacional para controle de jogos digitais, utilizando apenas uma webcam convencional. A metodologia adotada compreende quatro etapas principais: (i) arquitetura da aplicação, (ii) integração com o ambiente de jogo, e (iii) validação funcional em cenários interativos.

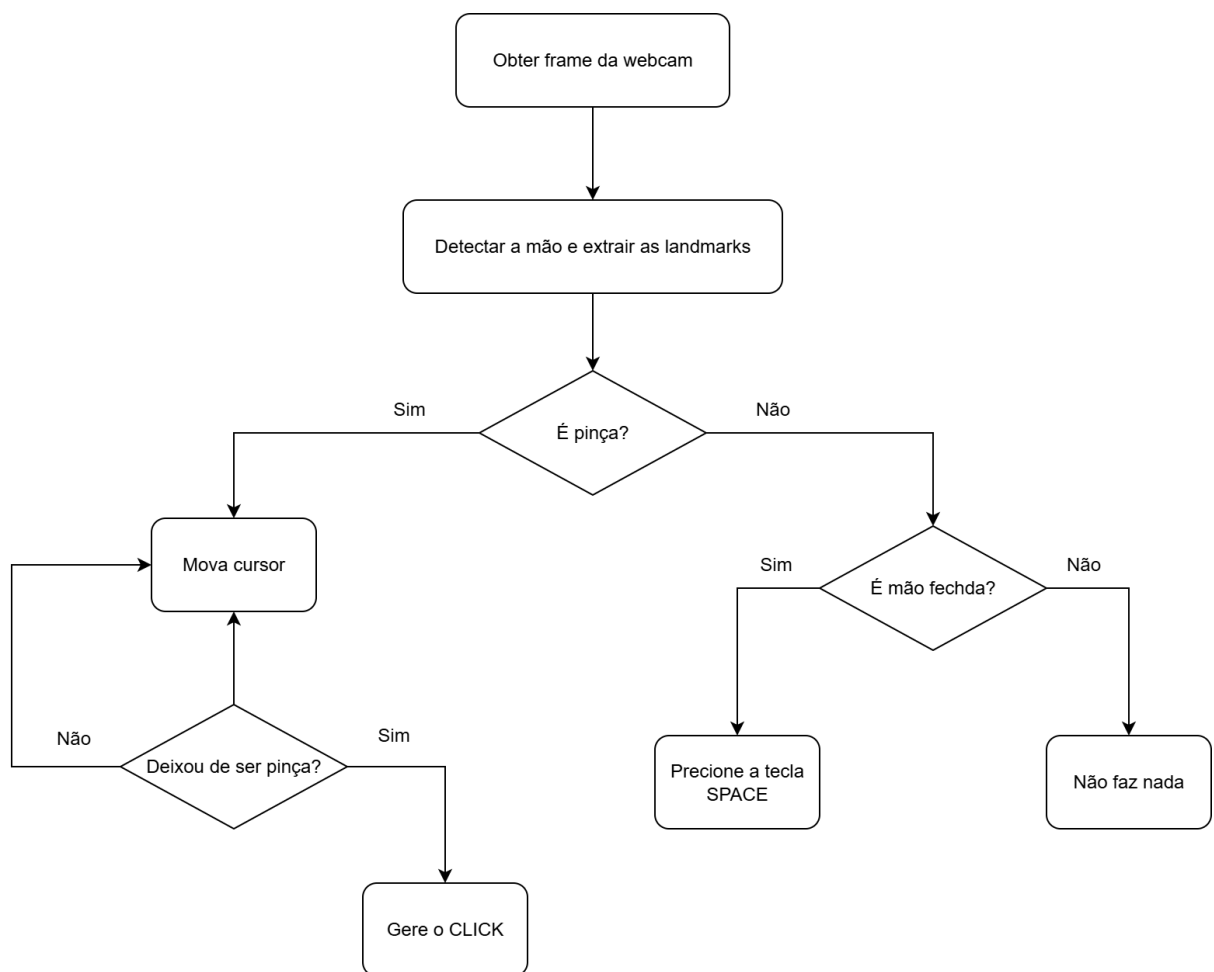
A arquitetura do sistema engloba captura de vídeo, processamento em tempo real e comunicação dos gestos com o jogo, além da escolha das bibliotecas *MediaPipe* e *OpenCV*. Já no desenvolvimento, a detecção gestual foi implementada a partir dos módulos *MediaPipe Hands* e *Holistic*, que extraíram landmarks das mãos, face e corpo. A partir desses pontos de referência, regras lógicas e relações espaciais entre os dedos permitiram identificar padrões de gestos, enquanto a *OpenCV* foi usada para exibir os resultados com sobreposição gráfica.

A integração com o *Geometry Dash*¹ ocorreu por meio da simulação de eventos de teclado, como o gesto de mão fechada acionando a tecla espaço, responsável pelo salto no jogo, sem necessidade de alterar o código original. Por

¹ Disponível em: <https://geometrydashlitepc.io/#google_vignette>

fim, a validação considerou diferentes cenários de iluminação, latência, precisão e usabilidade. Os testes confirmaram a responsividade e aplicabilidade do sistema, inclusive com usuários leigos, destacando sua viabilidade como alternativa acessível para controle em jogos digitais. Na Figura 1 é apresentado o diagrama de atividade do funcionamento do código.

Figura 1 - Diagrama de Atividade da Aplicação



4 APRESENTAÇÃO, ANÁLISE E DISCUSSÃO DOS RESULTADOS

Os resultados obtidos foram avaliados em três aspectos principais: detecção de gestos, resposta em tempo real e interação com o jogo.

A interface gestual desenvolvida mostrou-se eficaz ao reconhecer três estados principais: mão aberta (*Idle*), pinça entre polegar e indicador (*Aiming*) e punho fechado (*Jump*). Esses gestos foram implementados a partir da extração dos

21 pontos de referência da mão utilizando os módulos MediaPipe Hands e Holistic, permitindo substituir interações convencionais com periféricos, como mouse e teclado, apenas por movimentos manuais. Dessa forma, a proposta ampliou as possibilidades de acessibilidade e tornou a experiência mais intuitiva para o usuário. Esses estados são ilustrados nas Figuras 2, 3, 4:

Figura 2 – Estados: Idle

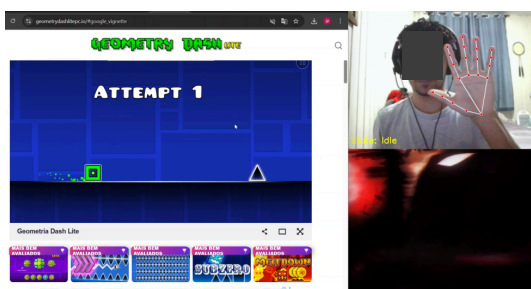


Figura 3 – Estados: Aiming

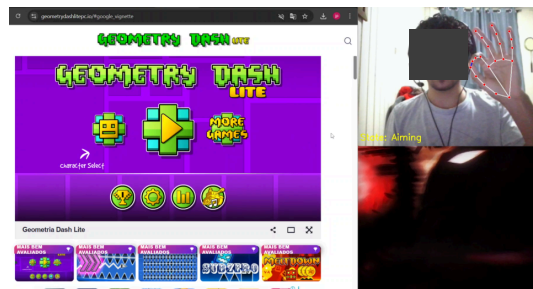
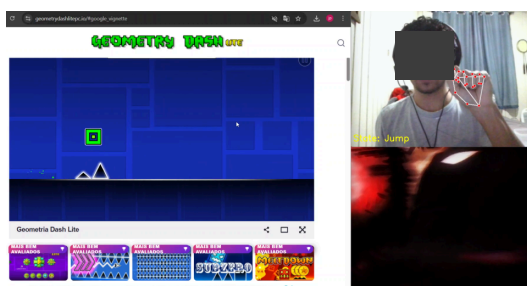


Figura 4 – Estados: Jump



A integração com o jogo ocorreu por meio da simulação de entrada de teclado `pyautogui.keyDown("space")` e `keyUp("space")` acionada pelo gesto de punho fechado. Como mostra a Figura 3, o personagem responde corretamente ao gesto de salto durante o *gameplay*, permitindo ao usuário jogar sem utilizar periféricos físicos. Além do pulo, o gesto de pinça se mostrou funcional para simular interações com menus, demonstrando a versatilidade da abordagem em interfaces mais amplas dentro e fora do jogo.

O modelo pré-treinado utilizado possui precisão média de 95,7% na detecção de palmas. Com resultados estáveis mesmo diante de oclusões parciais ou movimentos rápidos. Apesar de uma leve perda de desempenho em ambientes com pouca iluminação, a robustez geral da solução foi mantida, evidenciando sua viabilidade prática para uso em jogos digitais.

5 CONCLUSÃO

O trabalho desenvolveu e validou uma interface gestual baseada em visão computacional, utilizando *MediaPipe* e *OpenCV* com webcam, para controlar o jogo *Geometry Dash*. O sistema reconheceu com precisão e baixa latência os gestos de mão aberta (*Idle*), pinça (*Aiming*) e punho fechado (*Jump*), este último simulando a tecla espaço e permitindo uma jogabilidade fluida e intuitiva. Visto que o jogo é web, qualquer jogo que use esses movimentos em teoria funciona, contudo vale ressaltar que devido ao delay da aplicação e processo de *output* do modelo a certas limitações do quanto a jogabilidade é viável a depender da necessidade de reflexos. A solução apesar mostra potencial para aplicações em jogos, educação e reabilitação física, embora apresente limitações relacionadas à iluminação e ao posicionamento da câmera.

Como trabalhos futuros, propõe-se a expansão do conjunto de gestos reconhecidos, e a integração com outros gêneros de jogos e plataformas interativas.

6 REFERÊNCIAS

- GOOGLE. **MediaPipe Hands: High-fidelity hand and finger tracking solution.** Google AI Edge Documentation, 2023. Disponível em: <<https://mediapipe.readthedocs.io/en/latest/solutions/hands.html#layout-forward-target-https-developers-google-com-mediapipe-solutions-vision-hand-landmarker-title-hands-parent-mediapipe-legacy-solutions-nav-order-4>>. Acessado em 10 de Set. 2025.
- Zhang, F., Bazarevsky, V., Vakunov, A., Tkachenka, A., Sung, G., Chang, C.-L., et al. **MediaPipe Hands: On-device Real-time Hand Tracking.** 2020. Disponível em: <<https://arxiv.org/pdf/2006.10214>>. Acessado em 10 de Set. 2025.
- Lugaresi, C., Tang, J., Nash, H., McClanahan, C., Uboweja, E., Hays, M., et al. **MediaPipe: A Framework for Building Perception Pipelines.** 2019. Disponível em: <<https://arxiv.org/pdf/1906.08172>>. Acessado em 10 de Set. 2025.
- Szeliski, R. **Computer vision: algorithms and applications.** Springer, Cham. 2022.
- Bradski, G. **A Biblioteca OpenCV.** Revista de Ferramentas de Software do Dr. Dobb, 120; 122-125. 2000