

Sistema de Detecção de Gestos Para Controle de Drones

João P. S. S. Santos¹, Benicio B. Cruz², Maria C. A. de Andrade³, Fábio L. S. Prudente¹, Stephanie K. A. Sousa¹, Phillippe C. Santos¹

¹Instituto Federal de Educação, Ciência e Tecnologia de Sergipe (IFS)
Sergipe - Brazil

²Instituto Tecnológico de Aeronáutica (ITA)
São José dos Campos - SP - Brazil

³Faculdade de Ciências Aplicadas - Universidade Estadual de Campinas
Limeira - SP

{joao.santos032}@academico.ifs.edu.br

{benicio.cruz.101559}@ga.ita.br

{aragao.carol02}@gmail.com

{stephanie.sousa, phillipe.santos, fabio.prudente}@ifs.edu.br

Abstract. This article explores the development and implementation of a drone control system using gesture detection, highlighting the use of computer vision techniques. The proposed system employs the use of tools for capturing and interpreting manual gestures, which are associated with action commands for the drone. The solution aims to facilitate human interaction with drones, offering intuitive control and without the need for traditional physical controls. Experiments carried out demonstrated the functionality of the system, which responds in real time to gesture commands. The study also discusses challenges such as calibration for gesture detection and the influence of environmental variables, such as lighting and visual background. The work concludes that gesture detection for drone control is a viable approach with the capacity for expansion in future applications.

Resumo. Este artigo explora o desenvolvimento e implementação de um sistema de controle de drones utilizando detecção de gestos, destacando o uso de técnicas de visão computacional. O sistema proposto emprega uso de ferramentas de captura e interpretação de gestos manuais, que são associados a comandos de ação para o drone. A solução visa facilitar a interação humana com drones, oferecendo um controle intuitivo e sem a necessidade de controles físicos tradicionais. Experimentos realizados demonstraram a funcionalidade do sistema, que responde em tempo real aos comandos gestuais. O estudo discute também desafios como a calibração para a detecção dos gestos e a influência de variáveis ambientais, como iluminação e fundo visual. O trabalho conclui que a detecção de gestos para controle de drones é uma abordagem viável e com capacidade de expansão em futuras aplicações.

1. Introdução

Nos últimos anos, o interesse em Unmanned Aerial Vehicle (UAV) aumentou, tais como na área de entrega, transporte, resgate e vigilância. Naturalmente, a interação entre humanos e UAVs tornou-se frequente, e com o desenvolvimento de tecnologias de Inteligência Artificial (IA), a abordagem Interface Humano Computador (IHC) está sendo ativamente desenvolvida como uma alternativa para controlador tradicional baseado em joystick [Tezza et al. 2019]. Assim, ao contrário do controlador baseado em joystick, que é limitado para profissionais altamente treinados com interfaces complexas, o sistema IHC permite que usuários não qualificados controlem facilmente, com o objetivo de projetar um ambiente natural e intuitivo centrado no usuário. Com isso, o campo da pesquisa Interface Humano Computador está progredindo no sentido de projetar soluções mais inovadoras e interfaces naturais e podem ser amplamente classificados em quatro categorias: sensores vestíveis, controles remotos mais fáceis de usar, reconhecimento de fala e reconhecimento de gestos manuais [Chen et al. 2019].

Especificamente, o uso de gestos humanos para controlar dispositivos sem a necessidade de controles físicos tradicionais, oferece novas possibilidades de interação entre humanos e máquinas. Esse tipo de controle baseado em gestos é uma alternativa para controlar os drones, tornando-os mais versáteis em ambientes onde o controle remoto tradicional pode ser limitado ou impraticável.

Nesse sentido, esse trabalho propõe um controle multimodal através de gestos manuais das mãos, permitindo que novos usuários interajam e controlem drones em um primeiro contato. Visando simplicidade nos comandos, foram implementados gestos pré- definidos no sistema, através de um treinamento prévio com o usuário. O drone poderá ser controlado através da sua própria câmera ou remotamente via webcam de um dispositivo externo, sendo possível a visualização em tempo real de ambas as câmeras. Assim, para a detecção de gestos foram utilizadas técnicas de processamento de imagem, que permite interpretar movimentos humanos e traduzi-los em comandos executáveis pelo drone. Essa tecnologia possibilita uma interação mais natural e direta, o que pode ser útil em diversas situações, como filmagens aéreas, entregas e monitoramento. Utilizando as bibliotecas como *OpenCV*¹ e *MediaPipe*², foi possível capturar e interpretar gestos em tempo real. Neste contexto, o sistema foi testado com sucesso em um drone simples.

O trabalho está organizado da seguinte maneira. Posteriormente, na seção 2 é descrito trabalhos relacionados com a linha de pesquisa deste artigo. Na Seção 3 é descrita a configuração experimental. Após isso, na Seção 4 é descrita a arquitetura do sistema e os resultados obtidos. Por fim, na Seção 5 conclui-se o artigo.

2. Trabalhos Relacionados

Diversas abordagens para o reconhecimento de gestos manuais têm sido exploradas na literatura. No trabalho descrito em [Singha et al. 2013] é apresentado um método de reconhecimento de gestos manuais baseado na transformada de Karhunen-Loeve. Além disso, em um estudo preliminar, foi feito em [Silva et al. 2013] o uso do sensor Leap Motion como um controlador para novos instrumentos musicais digitais, demonstrando o potencial de sensores de profundidade para interação baseada em gestos. Já em [Kainz

¹ <https://opencv.org/>

² <https://chuoling.github.io/mediapipe/>

et al. 2014] foi feita uma abordagem para rastreamento de mãos e reconhecimento de gestos utilizando câmeras de detecção de profundidade e monitoramento EMG. No contexto específico do controle de drones, pesquisas recentes têm explorado o uso de aprendizado profundo para o reconhecimento dinâmico de gestos manuais.

[Liu et al. 2021] realizou um trabalho sobre detecção humana em tempo real e reconhecimento de gestos para resgate com drones embarcados. Eles utilizaram o modelo yolo3-tiny para detecção humana. Assim, uma vez detectada uma pessoa, o sistema entrava na fase de reconhecimento de gestos corporais e manuais, com o objetivo de estabelecer uma comunicação breve e eficaz, evitando as limitações da comunicação por voz. Um conjunto de dados de dez gestos de resgate corporal foi criado, e o sistema alcançou alta precisão nos dados de teste para gestos corporais (99,80%) e gestos manuais (94,71%) usando métodos de aprendizado profundo. A implementação prática foi realizada em um drone com GPU Jetson Xavier, com testes em laboratório simulando um ambiente de campo.

[Yoo et al. 2022] investigou uma abordagem de interação humano-VANT baseada em estimativa de movimento e reconhecimento de gestos manuais em tempo real. Eles categorizaram os métodos de reconhecimento de gestos manuais em SGR (Sensor-Based Gesture Recognition) e VGR (Vision-Based Gesture Recognition). Para o reconhecimento de gestos estáticos, o sistema proposto utilizou o modelo yolov4³, alcançando algumas métricas, como a Mean Average Precision (mAP) de 98,3% e um Intersection over Union (IoU) médio de 91,7% em um conjunto de dados construído para o estudo. Os experimentos foram realizados em um ambiente de simulação utilizando Gazebo com PX4, demonstrando a usabilidade do sistema para não especialistas no controle de UAVs.

Como diferencial do trabalho apresentado, destaca-se a abordagem prática e acessível adotada para o controle de drones via detecção de gestos. Ao contrário de métodos que dependem de sensores especializados ou de hardware robusto, o sistema utiliza apenas a câmera do drone ou de um dispositivo externo, o que reduz significativamente os custos e amplia o acesso a usuários sem experiência prévia. Além disso, a implementação foi cuidadosamente otimizada para operar em plataformas com recursos computacionais limitados, como o drone DJI Tello, garantindo um desempenho eficaz mesmo em condições adversas. Outro ponto relevante é a disponibilização do código-fonte em um repositório aberto, o que não só assegura a reproduzibilidade dos resultados, mas também facilita futuras melhorias e adaptações, consolidando a proposta como uma solução viável para a interação humano-drone.

3. Abordagem Proposta

O presente trabalho propõe um sistema de controle de drones fundamentado na detecção de gestos manuais, objetivando oferecer uma interface intuitiva e de baixo custo para a interação entre o usuário e o dispositivo aéreo. Diferentemente dos métodos tradicionais, que exigem o uso de joysticks ou sensores especializados, nossa abordagem utiliza apenas a câmera do drone ou uma webcam externa para captar os gestos, viabilizando um primeiro contato simplificado mesmo para usuários não especializados. Assim, o sistema foi desenvolvido para proporcionar um controle multimodal, onde os gestos pré-definidos são convertidos em comandos de ação para o

³ <https://docs.ultralytics.com/pt/models/yolov4/>

drone. Essa proposta visa não apenas reduzir custos e complexidade, mas também ampliar o acesso à tecnologia em plataformas com limitações de processamento, como o drone DJI Tello, demonstrando a viabilidade prática e a capacidade de expansão para futuras aplicações.

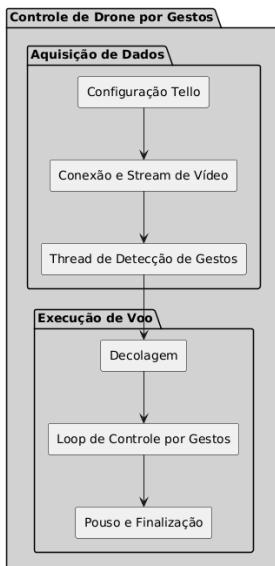


Figura 1. Abordagem Proposta

4. Arquitetura do Sistema

O drone escolhido para o projeto foi o DJI Tello, como mostrado na Figura 1, um quadricóptero de pequeno porte que pode ser pilotado manualmente ou atuar como um robô aéreo autônomo, graças a sua câmera e seu sistema de posicionamento visual. Desenvolvido pela startup chinesa Ryze Tech e utilizando um controlador de voo fabricado pela DJI, o Tello é equipado com uma câmera que facilita sua operação em ambientes internos. Compacto, seguro e extremamente leve, o Tello pesa apenas 87 gramas, incluindo bateria e hélices. Além disso, suas dimensões são 98 x 92,5 x 41 milímetros, podendo alcançar uma velocidade máxima de 28,8 km/h e voar por até 13 minutos sob condições ideais.

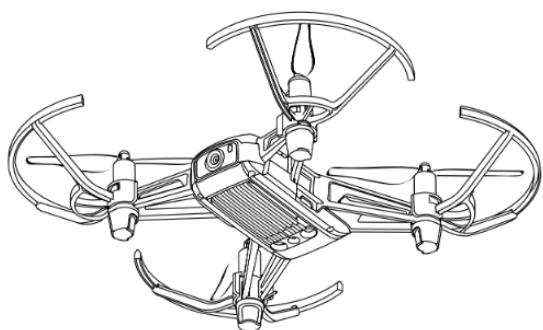


Figura 2. Drone Tello. Retirado de [Tello user manual v1.4.]⁴

⁴ <https://www.ryzerobotics.com/tello>

O Tello conta com sensores barométricos, infravermelhos e uma câmera, que juntos compõem o Sistema de Posicionamento de Visão (VPS). Esse sistema é essencial para manter a estabilidade do drone, operando com maior eficiência em altitudes entre 0,3 e 10 metros, sendo mais preciso entre 0,3 e 6 metros. Além disso, o drone possui uma Unidade de Medição Inercial (IMU), que inclui acelerômetro e giroscópio, responsável por medir a aceleração e orientar o drone. O processamento das informações é feito por um processador Intel de 14 núcleos.

A câmera frontal do Tello, como mostrada na Figura 2, possui 5 megapixels, podendo capturar imagens com resolução de 2592x1936 pixels e gravar vídeos em HD (720p). Localizada na parte dianteira do drone, essa câmera desempenha um papel fundamental no desenvolvimento de aplicações autônomas, permitindo a captura de imagens para sistemas de visão computacional e possibilitando o monitoramento remoto da aeronave.

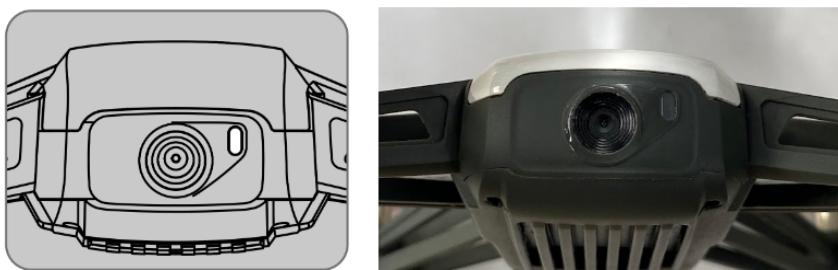


Figura 3. Câmera do Tello.

Na parte de visão computacional foi utilizado a linguagem de programação Python e as bibliotecas *OpenCV* para o processamento de imagens e vídeos, *Mediapipe* para detecção e rastreamento das mãos e *djitellopy*⁵, que é uma biblioteca específica para o controle do drone. O sistema foi programado para apenas funcionar com gestos feitos pela mão esquerda, por conta do Tello ter um poder de processamento inferior para processar as duas mãos ao mesmo tempo. Foram programados diversos gestos, os apresentados na Figura 3, que representam movimentos básicos do drone (Cima, Baixo, Esquerda, Direita e Pousar).



Figura 4. Modelo dos gestos.

⁵ <https://djitellopy.readthedocs.io/en/latest/tello/>

O fluxo do código mostrado no algoritmo da Figura 4 consiste principalmente na leitura desses gestos pré-programados. Inicialmente, a captura contínua dos frames, oriundos da câmera do drone ou de uma câmera de um dispositivo externo, é realizada, e cada frame passa por um processamento com *OpenCV*, onde operações de pré-processamento, como conversão de cores e suavização, são aplicadas para facilitar a detecção dos contornos e estruturas da mão. Em seguida, utilizando os recursos do *Mediapipe*, o sistema detecta os pontos de referência da mão esquerda, permitindo identificar padrões de posição que correspondem aos gestos pré-definidos, mesmo com variações de iluminação e ângulos.

Assim que o gesto é reconhecido, os padrões identificados são mapeados para comandos específicos, e estes são imediatamente enviados para o drone através da biblioteca *djitellopy*, que estabelece a interface entre o código Python e o hardware do drone Tello. Após o drone completar o ciclo de movimento pré-estabelecido para o comando executado, o sistema espera pelo próximo comando. Para garantir a segurança e economia de bateria, foi implementado um mecanismo de controle de tempo: se nenhum novo gesto for detectado em 4 segundos, o drone cancela a execução do programa e pousa automaticamente. Essa condição previne que o drone permaneça no ar com os motores ligados consumindo bateria sem necessidade. Além disso, o sistema monitora continuamente a execução dos comandos e o feedback do drone, permitindo ajustes em tempo real que garantem uma operação fluida e precisa. Dessa forma, a integração entre a captura de vídeo, a detecção dos gestos e o envio dos comandos resulta num fluxo de controle dinâmico, onde cada movimento do drone é sincronizado com os gestos detectados, proporcionando uma experiência segura e eficiente no controle do drone. Por fim, o código-fonte está disponível em um repositório no Github⁶.

⁶ https://github.com/J3ipy/drone_controlado_gestos

Algoritmo 1: Controle do Tello com Gestos de Mão

Entrada: Drone Tello com *stream* de câmera ligado
Saída: Movimentos do drone com base nos gestos da mão detectados

```
1  início
2      tello = conectar();
        // Conectar ao drone e configurar tello.streamon();
3      frame == None frame = capturar(tello);
4      gesto = 'Desconhecido';
5      start_thread(detecção_de_mão(tello));
        // Iniciar detecção de mão em thread
6      se fly == True então
7          tello.takeoff();
8          tello.set_speed(10);
9          tello.move_up(80);
10     fim
11    true se gesto == 'Pousar' então
12    | break;
13    fim
14    senão
15    | se gesto == 'Direita' então
16    | | hV = -15;
17    | fim
18    | se gesto == 'Esquerda' então
19    | | hV = 15;
20    | fim
21    | se gesto == 'Cima' então
22    | | vV = 20;
23    | fim
24    | se gesto == 'Baixo' então
25    | | vV = -20;
26    | fim
27    | tello.send_rc_control(hV, dV, vV, rV);
28    | fim
29    esperar(0.2);
        // Delay para leitura do próximo gesto
30   se fly == False então
31   | tello.land();
32   fim
33   tello.streamoff();
34   'Bateria:', tello.get_battery();
35 fim
```

Figura 5. Algoritmo do Sistema.

4.1 Resultados

Foram realizados diversos experimentos controlando o drone através da sua própria câmera e da webcam de algum dispositivo externo, sendo o mesmo controlado remotamente dessa maneira, o vídeo dos dois testes está disponível através do link: <https://youtu.be/BZeZuVFEO6M?si=6la4tmEkyFZHuddQ>.

Durante os experimentos, foram coletadas amostras do percurso e das velocidades do drone ao longo dos eixos X, Y e Z, permitindo uma análise do comportamento do dispositivo. Nas Figuras 5 a 9 está representado os comandos feitos durante o percurso. Esse experimento foi conduzido utilizando um gesto de cada vez, que foram captados pela própria câmera do drone, onde, por exemplo, o primeiro gesto de "cima" acionou a trajetória, após isso os gestos específicos de "direita" e "esquerda" direcionaram os movimentos laterais, e por fim, o gesto de "pousar" foi interpretado como comando para iniciar o pouso. Tais gestos resultaram em respostas adequadas do drone, evidenciadas na Figura 10 e 11, onde traz uma visualização da posição do drone durante o percurso e as variações de velocidade registradas em cada eixo durante o percurso.



Figura 6. Gesto “Cima”



Figura 7. Gesto “Esquerda”



Figura 8. Gesto “Direita”



Figura 9. Gesto “Baixo”

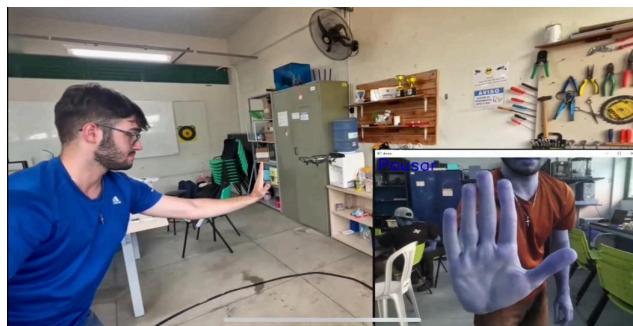


Figura 10. Gesto “Pousar”

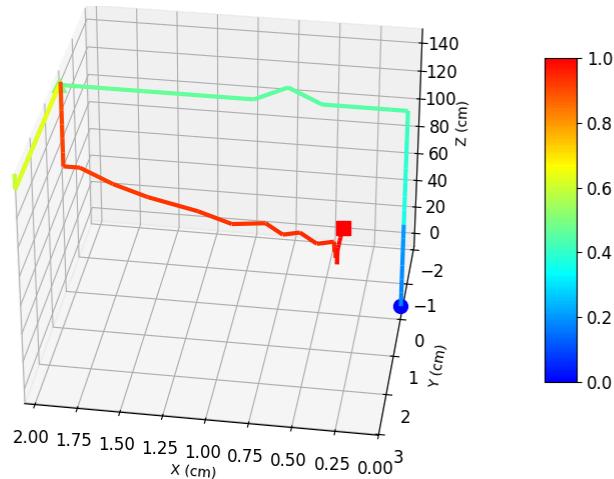


Figura 11. Gráfico 3D da posição do drone (câmera do drone).

Legenda: — Início do percurso, — Meio do percurso, — Fim do percurso.

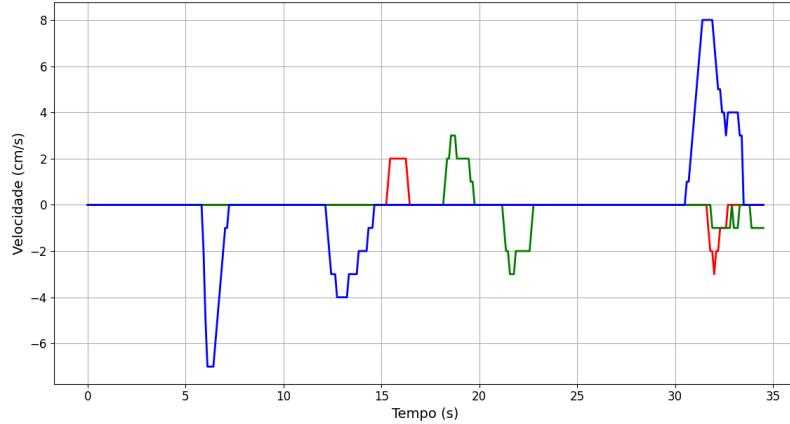


Figura 12. Gráfico da velocidade do drone durante o percurso.

Legenda: — Eixo X — Eixo Y — Eixo Z

Por sua vez, nas Figuras 12 e 13, o drone realizou um percurso diferente, controlado por meio da webcam de um dispositivo externo. Nesse caso, os gestos foram interpretados pelo sistema de visão do computador, na Figura 14 evidencia o controle através da webcam e a visualização da câmera simultaneamente, possibilitando uma comunicação remota mais refinada, onde comandos semelhantes ao cenário anterior foram utilizados para dirigir o movimento e o pouso. Todos os experimentos foram realizados em um ambiente fechado e controlado, assegurando a reproduzibilidade dos testes e a segurança durante as manobras.

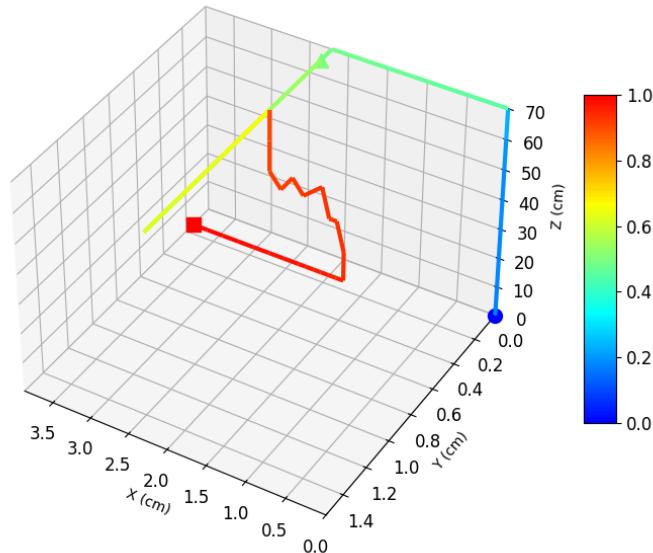


Figura 13. Gráfico 3D da posição do drone (webcam).

Legenda: — Início do percurso, — Meio do percurso, — Fim do percurso.

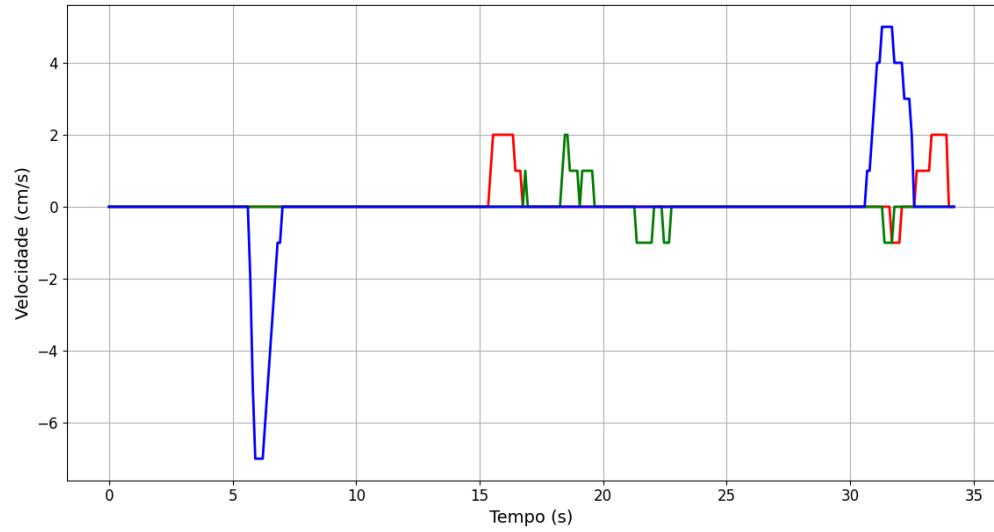


Figura 14. Gráfico da velocidade do drone durante o percurso.
Legenda: — Eixo X — Eixo Y — Eixo Z



Figura 15. Visualização do sistema de controle por meio de uma webcam externa

Os resultados dos experimentos até o momento cumpriram com os objetivos esperados, no entanto, alguns fatores do ambiente em que o drone opera podem interferir no processo de reconhecimento de gestos. Um exemplo é a alta incidência de luz solar, que pode comprometer a qualidade das imagens captadas pela câmera do Tello, dificultando o reconhecimento preciso dos gestos. Isso acontece, pois a câmera do drone pode não lidar bem com o excesso de brilho ou reflexos, o que impacta diretamente na capacidade do sistema de identificar movimentos com precisão. Além disso, locais com ventos fortes representam um desafio adicional. O drone, ao ser desestabilizado durante o voo por rajadas de vento, também pode perder a capacidade de manter a câmera focada adequadamente nos gestos, o que afeta a detecção correta. Portanto, ambientes controlados, com pouca interferência luminosa e baixa incidência de vento, são mais adequados para garantir a estabilidade e precisão do sistema.

Em relação ao reconhecimento de gestos, os testes revelaram que o uso de apenas uma mão para o comando é mais eficiente. Quando implementado o reconhecimento de gestos com ambas as mãos, o sistema apresentou lentidão e baixa

precisão. Isso ocorre por conta do processamento simultâneo de dois conjuntos de gestos que requer maior capacidade de processamento do drone, o que sobrecarrega o seu processador e diminui o desempenho geral do sistema. O Tello não possui um hardware de processamento robusto o suficiente para lidar com múltiplas entradas de imagem de forma eficiente, o que explica essa lentidão e imprecisão ao tentar reconhecer gestos de ambas as mãos ao mesmo tempo. Por essa razão, a opção por utilizar apenas uma mão tem se mostrado mais eficaz, evitando problemas de desempenho e garantindo que os comandos sejam executados de maneira mais fluida.

5. Conclusão

Em síntese, a motivação central deste trabalho foi desenvolver uma ferramenta de Interface Humano-Computador (IHC) centrada no usuário final, proporcionando segurança e conforto para um primeiro contato com drones. Os experimentos realizados demonstraram que o sistema é capaz de controlar o drone de forma adequada, evidenciando seu potencial prático e a eficácia da abordagem adotada.

Como uma melhoria futura, está sendo considerada a implementação de um modelo de redes neurais artificiais para otimizar o reconhecimento de gestos. Com essa implementação, a necessidade de processamento contínuo seria reduzida, já que o drone passaria a identificar padrões previamente aprendidos pela rede neural, diminuindo o consumo de recursos computacionais e, ao mesmo tempo, aumentando a precisão do reconhecimento. Essa abordagem pode melhorar significativamente o desempenho do sistema, especialmente em ambientes onde os fatores externos possam interferir na captura de imagens. Trabalhos futuros também incluem a ampliação do conjunto de gestos reconhecidos, a adaptação do sistema para lidar com duas mãos simultaneamente e uso de técnicas de visão computacional para diminuir os ruídos visuais captados. Testes adicionais em cenários indoor variados contribuirão para validar a robustez da solução e ampliar seu potencial de aplicação em contextos educacionais, de demonstração tecnológica ou mesmo em atividades de inspeção e monitoramento controladas por gestos.

Referências

- Chen, B.; Hua, C.; Li, D.; He, Y.; Han, J. Intelligent Human–UAV interaction system with joint cross-validation over Action–Gesture recognition and scene understanding. *Appl. Sci.* 2019, 9, 3277.
- Kainz, O.; Jakab, F. Approach to Hand Tracking and Gesture Recognition Based on Depth-Sensing Cameras and EMG Monitoring. *Acta Informatica Pragensia*, v. 3, n. 1, p. 104–112, 2014.
- Khaksar, S. et al. Design and Evaluation of an Alternative Control for a Quad-Rotor Drone Using Hand-Gesture Recognition. *Sensors*, v. 23, n. 12, p. 5462, 2023.
- Liu, C.; Szirányi, T. Real-Time Human Detection and Gesture Recognition for On-Board UAV Rescue. *Sensors*, v. 21, n. 6, p. 2180, 2021.
- Silva, E. S. et al. A preliminary evaluation of the leap motion sensor as controller of new digital musical instruments. In: Proceedings of a conference, 2013.
- Singha, J.; Das, K. Hand gesture recognition based on Karhunen–Loeve transform. arXiv preprint arXiv:1306.2599, 2013.
- Tezza, D.; Andujar, M. The state-of-the-art of human–drone interaction: A survey. *IEEE Access* 2019, 9, 167438–167454.
- Yoo, M. et al. Motion Estimation and Hand Gesture Recognition-Based Human–UAV Interaction Approach in Real Time. *Sensors*, v. 22, n. 7, p. 2513, 2022.