

# Autonomous Spacecraft Safety

Pedro H. F. Santana N950625,

Bárbara Nascimento G5934H6

Ciência da Computação - CC6A28

Universidade Paulista

07/11/2024

## ABSTRACT

This paper investigates and compares edge detection methods in real-time autonomous object avoidance systems, with a focus on space environments. Using image processing and artificial intelligence (AI), we analyze the Canny, Laplacian of Gaussian (LoG), and Sobel operator methods for obstacle detection. Each technique was evaluated for detection accuracy, processing time, and computational efficiency and tested in simulated scenarios within Webots. Results show that the Canny method offers high precision and robustness, the Laplacian of Gaussian excels in noisy environments, and the Sobel operator is faster but less effective in complex settings. We conclude that combining techniques or integrating additional sensors, such as LIDAR, may further enhance the resilience and performance of autonomous systems. This study provides a foundation for developing safer and more efficient real-time object detection and avoidance systems.

## 1. INTRODUÇÃO

Com o aumento da exploração espacial, a necessidade de sistemas autônomos avançados e confiáveis para garantir a segurança nunca foi tão urgente. Missões tripuladas e não tripuladas estão expostas a riscos causados por várias formas de detritos, micrometeoritos e outros obstáculos. Sistemas existentes, como o Debris Collision Avoidance Systems (DCAS), dependem principalmente de radares e rastreamento baseados na Terra; no entanto, eles têm limitações quanto à capacidade de resposta em tempo real, especialmente para detritos pequenos ou imprevisíveis.

Sistemas autônomos a bordo, equipados com processamento de imagem e inteligência artificial (IA), oferecem uma solução para esse desafio ao possibilitar a detecção e evasão de obstáculos em tempo real, sem a necessidade de intervenção terrestre. A velocidade dos detritos em órbita, frequentemente excedendo 27.000 km/h, exige que as decisões sejam tomadas em milissegundos. Ao integrar técnicas de processamento de imagem, como a detecção de bordas, as espaçonaves podem detectar obstáculos, avaliar probabilidades de colisão e fazer ajustes autônomos em sua trajetória.

Alternativas como o LiDAR (Light Detection and Ranging) podem fornecer dados confiáveis sobre objetos, utilizando lasers para emitir pulsos de luz e medir o tempo que a luz leva para retornar. Essa tecnologia também oferece dados tridimensionais de alta resolução; contudo, seu custo e eficiência energética fazem com que outros sistemas sejam mais viáveis, dependendo do cenário específico em que a espaçonave será colocada.

Este trabalho investiga três técnicas de detecção de bordas—Canny, Laplaciano do Gaussiano (LoG) e operadores de Sobel—para avaliar sua eficácia em sistemas autônomos de desvio de obstáculos em tempo real. Serão analisadas as vantagens e limitações de cada método, com ênfase na velocidade de processamento, precisão e eficiência computacional, com o objetivo de identificar um método ideal para a detecção e desvio autônomo de obstáculos.

## 2. TÉCNICAS DE PROCESSAMENTO DE IMAGEM E IA PARA DETECÇÃO DE OBSTÁCULOS

Incorporar processamento de imagem e aprendizado de máquina permite que espaçonaves identifiquem, rastreiem e evitem obstáculos de maneira eficaz. Abaixo estão os principais métodos avaliados neste artigo:

### 2.1 TÉCNICAS DE PROCESSAMENTO DE IMAGEM

A detecção de bordas permite que a espaçonave identifique os contornos de objetos, o que é essencial para diferenciar obstáculos do fundo espacial. Focamos nos seguintes métodos:

Detecção de Bordas de Canny: Conhecido por sua alta precisão, o método Canny é um processo em várias etapas que suaviza imagens, detecta gradientes de intensidade, aplica supressão de não-máximos e realiza a ligação de bordas. Este método é particularmente eficaz em

cenários com informações visuais complexas. No entanto, requer ajuste cuidadoso de limiares para equilibrar sensibilidade e redução de ruído.

Laplaciano do Gaussiano (LoG): O LoG aplica suavização Gaussiana para reduzir o ruído e, em seguida, detecta bordas calculando o Laplaciano (segunda derivada) da intensidade. Essa abordagem em duas etapas o torna robusto para detectar bordas mais finas em ambientes ruidosos, mas é mais exigente computacionalmente.

Operador Sobel: Este método baseado em gradiente detecta bordas calculando mudanças de intensidade nas direções x e y. É computacionalmente mais rápido, mas é mais sensível ao ruído, tornando-o adequado para ambientes onde a velocidade é priorizada sobre a precisão.

Deteção e Classificação de Objetos: Além da detecção de bordas, a classificação baseada em IA permite que a espaçonave diferencie entre detritos e outros objetos. Usando uma Rede Neural Convolutiva (CNN), objetos são classificados por forma e movimento, ajudando a identificar obstáculos relevantes. As CNNs são eficazes nessa tarefa, pois podem ser treinadas para reconhecer objetos em várias configurações de luz e espaço.

## 2.2 ABORDAGENS DE INTELIGÊNCIA ARTIFICIAL

A IA aprimora a tomada de decisões ao prever trajetórias e respostas ideais:

Redes Neurais Convolutivas (CNNs): Treinadas para reconhecer detritos espaciais, as CNNs podem classificar objetos detectados com base em suas características visuais, fornecendo informações sobre o tipo de objeto (por exemplo, detritos ou partículas não ameaçadoras) com alta precisão e velocidade. Quando combinadas com Redes Neurais Recorrentes (RNNs), que se especializam em previsões de séries temporais, o sistema pode prever a posição futura dos detritos com base em sua trajetória atual.

Aprendizado por Reforço (RL): Em um ambiente de RL, a espaçonave "aprende" estratégias ideais de evasão simulando milhares de cenários possíveis. Com o tempo, o sistema melhora na seleção da manobra mais eficiente em termos de energia e segurança. Esse método permite que a espaçonave continue aprimorando sua tomada de decisões mesmo em condições novas e imprevistas.

### 3. REDUNDÂNCIA E TRATAMENTO DE FALHAS

A redundância é crucial em qualquer sistema de espaçonave. Em caso de falha de propulsores ou sensores, a IA deve ajustar dinamicamente sua estratégia de controle. Uma solução possível é o uso de um sistema de controle tolerante a falhas, que ajusta os movimentos da espaçonave com base nos propulsores operacionais.

A IA pode ser treinada com simulações onde diferentes propulsores ou sensores são desativados. O objetivo é garantir que, mesmo com falhas parciais no sistema, a espaçonave ainda consiga calcular uma manobra segura. Por exemplo, algoritmos de aprendizado supervisionado podem ser treinados para reconhecer falhas de sensores com base em anomalias nos fluxos de dados.

### 4. PROCESSO DE TESTE

Para avaliar a eficácia das técnicas de processamento de imagem, foi desenvolvido um processo estruturado de teste:

**Coleta de Imagens de Amostra:** Imagens representativas de obstáculos foram coletadas para simular condições reais.

**Desenvolvimento de Algoritmos:** Cada técnica de processamento de imagem foi implementada com parâmetros ajustáveis para avaliar seus próprios indicadores de desempenho, armazenando-os em arquivos independentes para análise futura.

**Análise de Desempenho:** Com a nova saída de imagem e um log de desempenho, outro código foi desenvolvido para calcular o tempo médio de processamento de cada método, e os resultados foram plotados em um gráfico para comparação.

**Cálculo de Evasão de Objetos:** Um algoritmo mestre de evasão foi criado para calcular manobras com base nas bordas detectadas, com foco na detecção do centro dos objetos.

**Ambiente de Simulação:** Os algoritmos de processamento de imagem e evasão foram testados capturando dados em tempo real da câmera simulada do ambiente do Webots, para validar a eficiência do sistema.

## 5. COMPARAÇÃO DE ABORDAGENS

Os sistemas atuais, como os DCAS, dependem de radares e monitoramento terrestre para rastrear grandes detritos. No entanto, esses sistemas têm limitações de resposta, especialmente para detritos menores que não são rastreados ativamente.

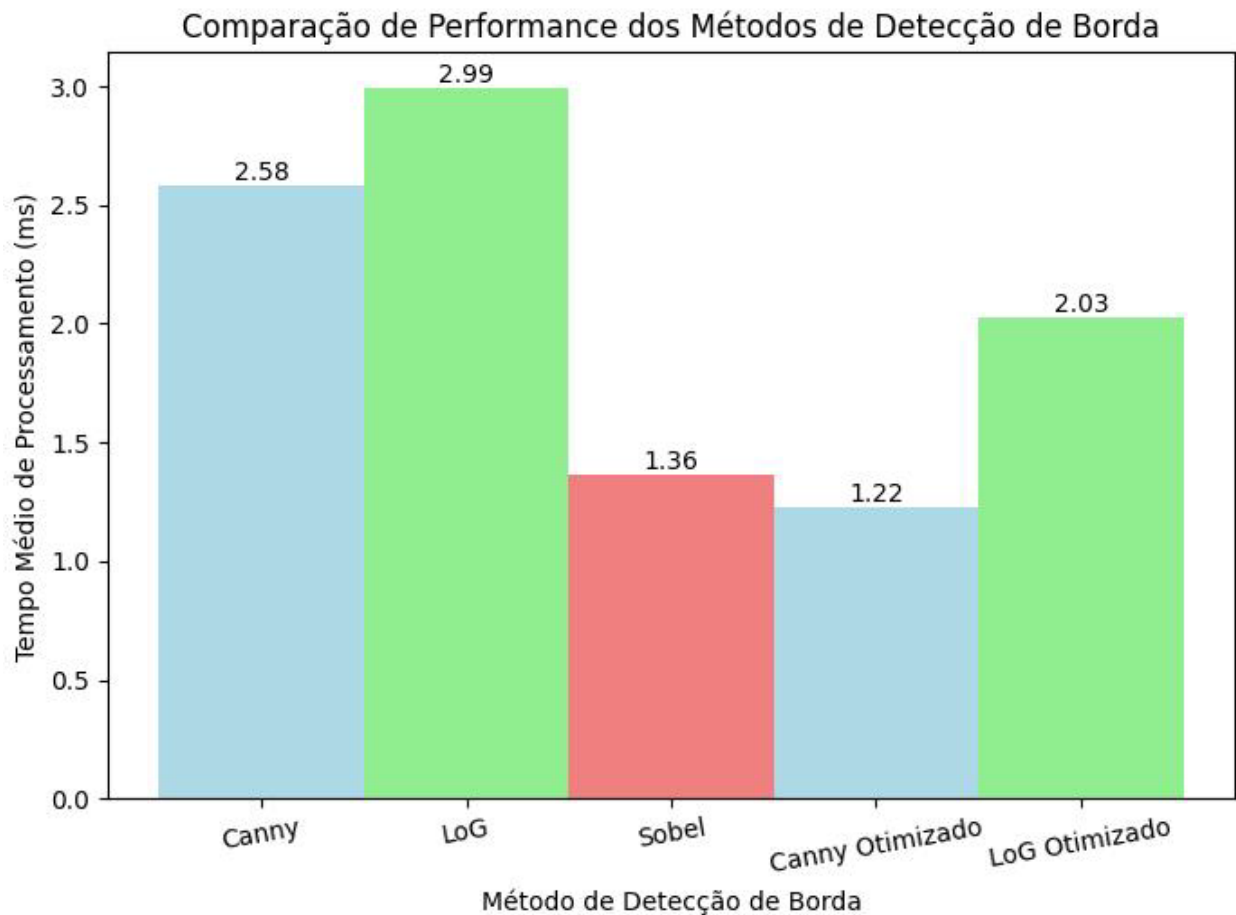
Método	Vantagens	Desvantagens
DCAS (Baseado em Radar)	Atualizações em tempo real para detritos grandes	Requer Intervenção Terrestre
IA + Processamento de Image	Pode operar de forma autônoma e rastrear todos os tamanhos	Computacionalmente intensiva a bordo
Baseado em LiDAR	Funciona em ambientes com baixa luminosidade	Caro e complexo

A utilização de IA e processamento de imagem a bordo resolve o problema de atrasos de comunicação e permite que a espaçonave opere autonomamente. Enquanto sistemas baseados em radar detectam objetos grandes, dependem de rastreamento externo, e sistemas baseados em IA podem lidar com a detecção de detritos pequenos em tempo real.

### 5.1 TÉCNICAS DE PROCESSAMENTO DE IMAGEM

Método	Tempo de Processamento	Qualidade de Detecção	Robustez	Uso de Recursos
Canny	2.6 ms	Alta	Alta	Baixo
Laplaciano de Gaussiano	3 ms	Muito Alta	Moderada	Moderado
Operador Sobel	1.4 ms	Alta, direcional	Moderada para baixa	Baixo

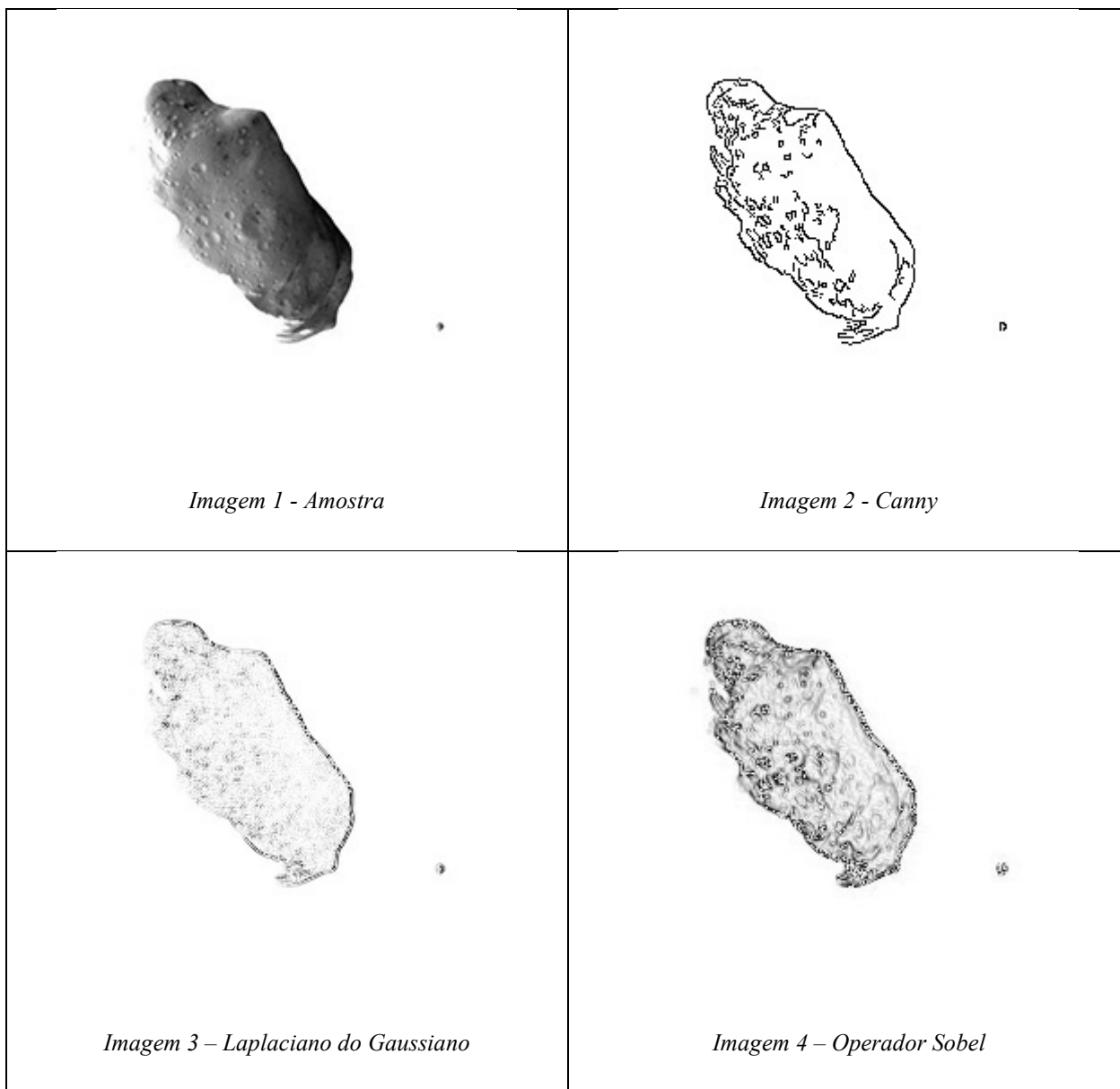
Cada Técnica apresenta vantagens e desvantagens. Diferenças em contraste, brilho e luz natural, assim como objetos distantes, podem alterar drasticamente os resultados. Ajustes menores podem otimizar tempo, eficiência energética, robustez e qualidade de cada técnica.



Além do método Sobel, tanto Canny quanto Laplaciano de Gaussiano apresentam vantagens significativas no tempo de processamento. Em imagens de amostra controladas de 256x256 pixels, a diferença de desempenho após otimizações torna-se praticamente irrelevante – de 1.2 ms a 2 ms. Portanto, a técnica utilizada será baseada na qualidade da saída e no desempenho futuro quando integrada com os algoritmos de classificação e simulação de IA.

## 5.2 QUALIDADE DE SAÍDA DA IMAGEM

Diferentes resultados podem ser observados, e o processamento subsequente pode variar para cada um. Todos os três métodos são eficazes em destacar os objetos, embora o Canny seja mais especializado na detecção de bordas, enquanto os outros mostram uma representação mais completa dos objetos.

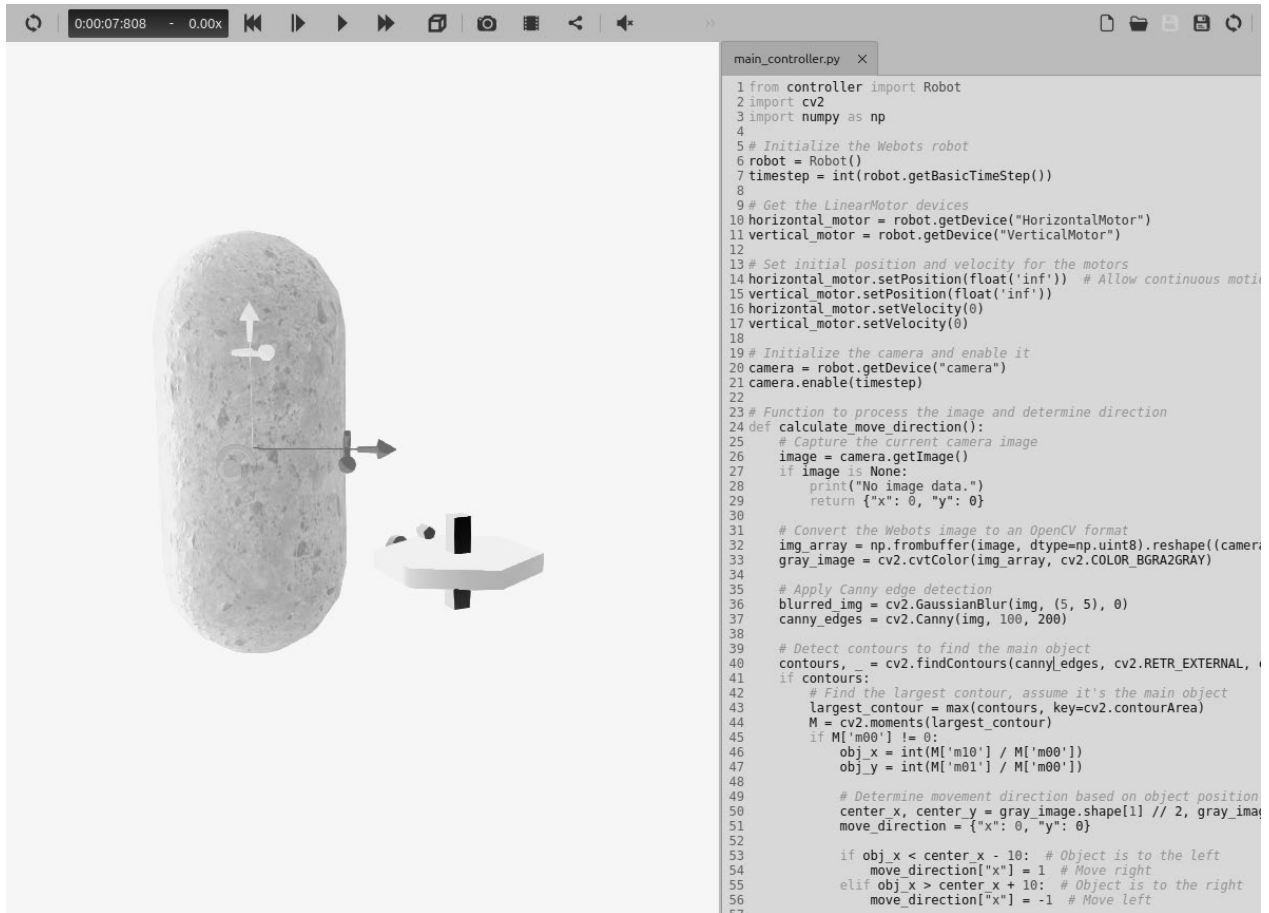


## 6. TESTES E SIMULAÇÃO

O criação do ambiente virtual com os softwares, bibliotecas e suas dependências foram instaladas na distribuição 24.04.1 do Ubuntu (Noble), utilizando um processador Ryzen 5 5600X – 6 núcleos/12 threads a 4.2GHz, com 32GB de memória RAM a 3000MT/s com latência CL 16, em um SSD NVMe R: 2100MB/s, W: 1700MB/s, e uma placa de vídeo Radeon RX 7600.

A versão do Python utilizada foi a 3.10.15, por motivos de compatibilidade com bibliotecas, juntamente com numpy, opencv, scikit, matplotlib, rocm, ros2-rolling, webots, entre outros.

Durante as simulações, cada método de detecção de bordas foi aplicado para transformar as imagens capturadas em representações das bordas dos objetos. O sistema de controle do robô, após processar a imagem, calculava a manobra necessária para evitar uma colisão iminente.



A escolha da manobra era baseada na posição e no contorno dos objetos detectados

A seguir, as principais etapas realizadas e os resultados obtidos:

Ambiente de Teste: O ambiente no Webots foi configurado com gravidade zero e sem fricção, simulando um cenário espacial realista.

Captura e Processamento de Imagens: A câmera capturava continuamente imagens dos objetos, que eram então processadas em tempo real pelos algoritmos Canny, LoG e Sobel.

Resultados de Detecção de Bordas:

- O método Canny mostrou-se superior na detecção precisa de bordas, especialmente em objetos mais distantes e com contornos bem definidos. Sua capacidade de eliminar ruídos durante a detecção permitiu uma maior precisão e confiabilidade nas manobras.



- O Laplaciano do Gaussiano apresentou resultados robustos em ambientes com ruído, embora seu tempo de processamento fosse ligeiramente mais alto.
- O Operador Sobel ofereceu a detecção mais rápida, mas com menor precisão em comparação aos outros métodos, sendo menos eficiente para contornos complexos ou pouco contrastantes.

Esses resultados demonstram que o método Canny, apesar de ser mais sensível ao ajuste de parâmetros, oferece uma combinação eficaz de precisão e tempo de processamento, tornando-se o mais adequado para o sistema autônomo de desvio de obstáculos.

## 7. CONCLUSÃO

Este trabalho investigou e comparou três técnicas de detecção de bordas — Canny, Laplaciano do Gaussiano (LoG) e Sobel — para aplicações em sistemas autônomos de desvio de objetos em tempo real no espaço. Após testes exaustivos realizados no Webots, o método de Canny destacou-se pela alta precisão e robustez em condições de simulação espacial. O Canny demonstrou uma capacidade superior de eliminação de ruídos e definição clara de contornos, essencial para o cálculo seguro de manobras evasivas.

Por outro lado, o LoG ofereceu desempenho sólido, enquanto o operador de Sobel sua maior velocidade de processamento não foi razão suficiente para justificar a menor precisão em cenários com bordas complexas. Com base nos resultados, concluímos que o método de Canny é a melhor opção para aplicações que exigem alta precisão e confiabilidade na detecção de obstáculos em tempo real.

Futuras pesquisas poderão explorar a combinação de métodos de detecção de bordas com dados de sensores adicionais, como LiDAR, para aumentar ainda mais a resiliência e o desempenho de sistemas autônomos no espaço. Além disso, o desenvolvimento de algoritmos de aprendizado por reforço pode proporcionar maior adaptabilidade em condições novas e imprevistas, tornando as manobras ainda mais otimizadas e eficientes.

## Referências

- [1] **ESA Space Debris Office**. Retirado de [https://www.esa.int/Enabling\\_Support/Operations/Ground\\_Systems\\_Engineering/ESA\\_Space\\_Debris\\_Office](https://www.esa.int/Enabling_Support/Operations/Ground_Systems_Engineering/ESA_Space_Debris_Office)
- [2] **Autonomous Safety Algorithms for Spacecraft**. Retirado de <https://phys.org/news/2024-08-algorithms-autonomous-spacecraft-safety.html>
- [3] **AI in Autonomous Systems**. Retirado de <https://www.science.org/doi/10.1126/scirobotics.adn4722>
- [4] **Real-time Software Simulation**. Retirado de <https://cyberbotics.com/doc/reference/introduction>
- [5] C. GONZALEZ, Rafael; EUGENE WOODS, Richard. **Digital Image Processing**, 3<sup>rd</sup> Ed. 2007
- [6] SUÁREZ SÁNCHEZ, Pablo. **Webots-Based Implementation and Simulation of Robotics Algorithms**. Ingeniería Eléctrica, Electrónica, de Computadores y Sistemas, Departamento de: 10/06/2024.

Todo o projeto pode ser encontrado em: <https://github.com/vladrosant/autonomous-spacecraft-safety/>