

Navegação Autônoma e Reconhecimento de Sinais de Trânsito Usando Turtlebot4

10 de Julho de 2024

1st Matheus Andrade Brandão
Instituto de Informática
Universidade Federal de Goiás
Goiânia, Goiás, Brasil
matheus_brandao@discente.ufg.br

2nd Rafael Alves Goiás
Instituto de Informática
Universidade Federal de Goiás
Goiânia, Goiás, Brasil
rafael_goiias@discente.ufg.br

Abstract—Este relatório apresenta o desenvolvimento e a implementação de um protótipo para melhorar a navegação de veículos autônomos utilizando o Turtlebot4. O projeto foca na integração de sistemas de navegação, incluindo reconhecimento de sinais de trânsito, através da utilização do pacote Nav2, sensores de câmera e LIDAR. Diversos métodos, como SLAM Visual, fusão de sensores e técnicas de aprendizado profundo para detecção de objetos, foram empregados. Os resultados demonstram um sistema robusto e eficiente capaz de mapear, planejar rotas e navegar autonomamente em ambientes dinâmicos.

Index Terms—navegação autônoma, ROS2, robótica, inteligência artificial, detecção de objetos

I. INTRODUÇÃO

A navegação autônoma é um componente crítico no desenvolvimento de sistemas robóticos inteligentes e veículos autônomos. Com o avanço da tecnologia, a necessidade de sistemas de navegação mais eficientes e precisos tornou-se evidente. Este projeto tem como objetivo criar um protótipo utilizando o Turtlebot4 para aprimorar a navegação e o reconhecimento de sinais de trânsito, empregando o pacote Nav2 e sensores de câmera e LIDAR. A implementação e integração desses sistemas visam desenvolver uma solução robusta e eficiente para navegação autônoma em ambientes dinâmicos.

II. FUNDAMENTOS TEÓRICOS

A. Revisão da Literatura

A literatura relevante sobre navegação autônoma e SLAM (*Simultaneous Localization and Mapping*) oferece uma base sólida para este projeto. O artigo *“SLAM para Iniciantes: Uma Abordagem Tutorial para Localização e Mapeamento Simultâneos”* fornece uma introdução compreensiva ao conceito de SLAM, abordando tanto os aspectos teóricos quanto práticos. Outro trabalho relevante é a revisão *“Pilha de Navegação ROS: Estado Atual e Desafios Futuros”*, que discute as capacidades, limitações e direções futuras do ROS Navigation Stack. Em *“Uma Comparação de Abordagens Modernas de SLAM Visual de Uso Geral”*, são comparadas diversas abordagens de SLAM visual, destacando as vantagens

e desvantagens de cada método. A revisão *“Fusão de Dados de Múltiplos Sensores para Condução Autônoma: Uma Revisão”* explora técnicas de fusão de dados, essencial para integrar dados de LIDAR e câmera de forma eficaz. Por fim, *“Aprendizado Profundo para Detecção de Objetos e Percepção de Cena em Condução Autônoma: Uma Revisão”* explora o uso de técnicas de aprendizado profundo para detecção de objetos e percepção de cena, relevantes para as aplicações de visão computacional neste projeto.

B. Dataset

Os dados a serem explorados no projeto incluirão:

- **Dados de LIDAR:** Pontos de nuvem 3D coletados pelo sensor LIDAR do Turtlebot4. Utilizados para construir mapas detalhados do ambiente e detectar obstáculos. [1]
- **Dados de Câmera:** Sequências de imagens ou vídeos capturados pela câmera do Turtlebot4. Utilizados para a criação de mapas visuais, detecção de objetos e auxílio na navegação.
- **Dados de Odometria:** Informações de posição e movimento fornecidas pelos sensores de odometria do Turtlebot4. Utilizados para calcular a posição do robô e corrigir a trajetória.
- **Road Sign Detection:** Conjunto de imagens com placas de trânsito dos Estados Unidos da América. Este conjunto de dados contém 877 imagens de 4 classes distintas para o objetivo de detecção de placas de trânsito. As classes são: Semáforo; Parar; Limite de velocidade; Faixa de pedestre.



Fig. 1. Road Sign Detection: Conjunto de dados do Kaggle

C. Métodos

- SLAM Visual - ORB-SLAM: Um dos métodos mais populares de SLAM visual, utilizando recursos de pontos para mapeamento e localização.
- SLAM Visual - VINS-Mono: Algoritmo de SLAM visual-inercial que combina dados visuais e inerciais para melhorar a precisão.
- YOLO v5 (You Only Look Once): Um método de detecção de objetos em tempo real que pode ser utilizado para identificar obstáculos e características específicas no ambiente. [4]
- SSD (Single Shot MultiBox Detector): Outro método eficiente para detecção de objetos em tempo real.
- Kalman Filter: Técnica estatística utilizada para combinar dados de múltiplos sensores e melhorar a estimativa da posição e movimento do robô.
- Particle Filter: Método probabilístico para fusão de sensores, útil em ambientes complexos e não lineares.

D. Avaliação

a) *Métricas de Desempenho:* Precisão de Mapeamento: Avaliada comparando o mapa gerado com um mapa de referência do ambiente.

Erro de Localização: Medido pela diferença entre a posição estimada do robô e sua posição real.

Taxa de Sucesso de Navegação: Percentual de rotas completadas com sucesso pelo robô sem colisões.

b) *Benchmarks:* KITTI Dataset: Conjunto de dados amplamente utilizado para avaliação de sistemas de visão e navegação autônoma, podendo ser usado para validar o desempenho do sistema.

Road Sign Detection: Conjunto de dados de trânsito. Usado para avaliar a precisão do modelo de Visão Computacional.

c) *Testes de Cenário Real:* Testes em Ambientes Controlados: Executar testes de navegação e mapeamento em ambientes controlados para avaliar a robustez do sistema. Testes em Ambientes Dinâmicos: Avaliar o desempenho do sistema em ambientes dinâmicos com obstáculos móveis e variações de iluminação.

III. METODOLOGIA

O desenvolvimento do projeto foi meticulosamente planejado e executado em várias etapas semanais, cada uma com objetivos específicos, para garantir o progresso gradual e a integração eficaz dos componentes do sistema. A metodologia adotada é baseada em uma linha lógica de pipeline, integrando as revisões da literatura e os fundamentos teóricos nas diferentes fases do trabalho.

A. Configuração do Ambiente de Trabalho

A primeira semana foi dedicada à configuração do ambiente de trabalho e à familiarização com os componentes do Turtlebot4 e o pacote Nav2. Instalamos o ROS2 no computador de trabalho e configuramos o Turtlebot4 no ambiente ROS2, garantindo que todas as dependências e bibliotecas necessárias

estivessem instaladas [2]. Revisamos a documentação do Turtlebot4, seguindo as orientações encontradas no artigo "Pilha de Navegação ROS: Estado Atual e Desafios Futuros". Testamos a comunicação entre o Turtlebot4 e o computador via ROS2, realizando alguns testes básicos de movimento. Em seguida, exploramos a documentação do Nav2, instalamos e configuramos o Nav2 no ambiente de desenvolvimento, e executamos exemplos básicos de navegação e mapeamento utilizando o Nav2.



Fig. 2. ROS2 Humble (Robotic Operating System)

B. Implementação e Teste do Mapeamento

Na segunda semana, focamos na implementação e teste do mapeamento utilizando o LIDAR e a câmera do Turtlebot4. Garantimos que o LIDAR estivesse funcionando corretamente com o Turtlebot4, utilizando o Nav2 para criar um mapa inicial do ambiente. A análise da qualidade do mapa gerado e o ajuste de parâmetros foram realizados conforme necessário, seguindo os princípios abordados no artigo "SLAM para Iniciantes: Uma Abordagem Tutorial para Localização e Mapeamento Simultâneos". Paralelamente, garantimos que a câmera estivesse funcionando corretamente e capturamos imagens para uso futuro, considerando as técnicas discutidas em "Fusão de Dados de Múltiplos Sensores para Condução Autônoma: Uma Revisão". Sincronizamos os dados do LIDAR e da câmera para criar um mapa mais robusto, ajustando os algoritmos de fusão de dados conforme necessário, baseando-nos nos métodos de fusão de sensores descritos nos artigos revisados. [3]

C. Refinamento do Mapeamento e Análise dos Dados

Durante a terceira semana, refinamos o mapeamento, analisamos os dados coletados e otimizamos a precisão. Repetimos o mapeamento em diferentes condições para melhorar a robustez do mapa, experimentando com diferentes configurações e parâmetros do Nav2 para otimização. Esta etapa foi guiada pelas comparações e análises apresentadas no estudo "Uma Comparação de Abordagens Modernas de SLAM Visual de Uso Geral". Analisamos os dados de mapeamento para identificar áreas de melhoria e realizamos ajustes baseados na análise para aprimorar a qualidade do mapa. Documentamos os processos e os resultados das fases de mapeamento e fusão de sensores.

D. Teste da Navegação Autônoma

Na quarta semana, configuramos e testamos a navegação autônoma do Turtlebot no ambiente mapeado. Configuramos o Nav2 para realizar o planejamento de rotas no ambiente mapeado, definindo pontos de interesse e trajetórias de navegação. Realizamos testes de navegação autônoma utilizando o mapa gerado, ajustando os parâmetros de navegação para melhorar a precisão e eficiência. Esta etapa envolveu a aplicação de técnicas de detecção de objetos discutidas no artigo "Aprendizado Profundo para Detecção de Objetos e Percepção de Cena em Condução Autônoma: Uma Revisão". Identificamos e resolvemos problemas de navegação, como obstáculos inesperados e falhas de planejamento, aplicando os métodos de fusão de sensores como o Filtro de Kalman e o Filtro de Partículas, conforme descrito na literatura. [5]

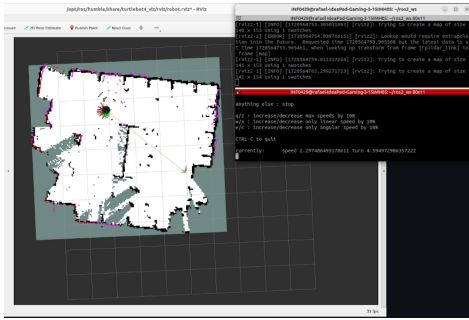


Fig. 3. Mapeamento real realizada em laboratório

E. Preparação e Demonstração Final

Na quinta e última semana, preparamos a demonstração final gravada em vídeo, garantindo que todos os componentes funcionassem perfeitamente. Realizamos testes completos integrando mapeamento, navegação e sensores, simulando um cenário com sinais de trânsito e ajustando quaisquer problemas restantes. Planejamos a sequência da demonstração, destacando as principais funcionalidades, e preparamos uma apresentação explicativa dos objetivos e resultados alcançados. Finalmente, executamos a demonstração no laboratório, mostrando a navegação autônoma do Turtlebot 4 utilizando os dados dos sensores e apresentamos a gravação para o professor e a turma em sala de aula.

IV. RESULTADOS

A. Simulação

Utilizando softwares que facilitam a visualização e rotina do robô durante as etapas de mapeamento, localização e navegação, conseguimos simular a ideia real do trabalho, modulando a velocidade do robô através da inferência em placas de trânsito. Assim, garantimos que toda a metodologia aplicada e os resultados obtidos foram o suficiente para colocarmos em prática no robô físico.

B. Robô físico

No robô físico e em ambiente fechado, utilizamos a mesma metodologia adotada na etapa de simulação. Usando placas de trânsito impressas e um mapa da sala de teste, conseguimos usufruir dos componentes disponibilizados pelo pacote de navegação autônoma NAV2 e realizar uma navegação desviando de objetos durante o percurso, sejam eles estáticos (cadeiras, mesas e paredes) ou obstáculos dinâmicos (pessoas e outros robôs). Durante um trajeto pré-estabelecido, capitamos as imagens que são publicadas em um tópico para fazer a inferência. Utilizamos apenas duas classes para a inferência: Parar e Limite de velocidade.

Assim, ao identificar alguma das classes mapeadas, fazíamos a leitura da velocidade do robô, armazenávamos esta e, dependendo da classe identificada, atribuímos diferentes velocidades apenas enquanto ocorre a inferência, durante a navegação. Toda a parte de processamento foi realizada de forma não-embarcada uma vez que a capacidade de processamento do Turtlebot4 não era suficiente.

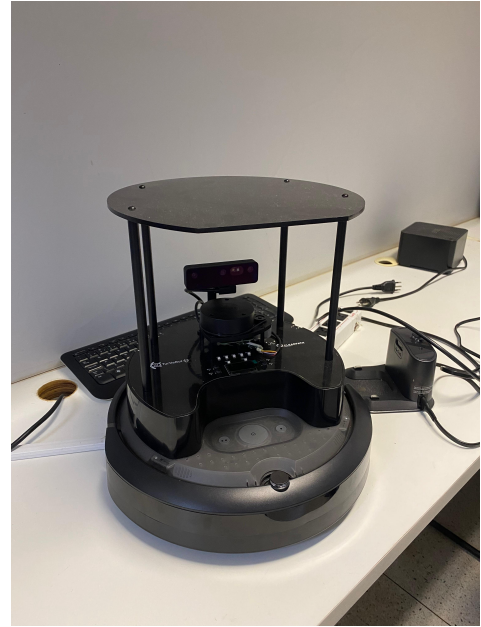


Fig. 4. Turtlebot4

Os resultados do projeto foram avaliados com base em diversas métricas de desempenho, incluindo a precisão de mapeamento, o erro de localização e a taxa de sucesso de navegação. A precisão de mapeamento foi avaliada comparando os mapas gerados com mapas de referência do ambiente, e os resultados demonstraram alta precisão em diversas condições ambientais. O erro de localização foi medido pela diferença entre as posições estimadas e reais do robô, e observamos um baixo erro de localização, indicando navegação precisa. A taxa de sucesso de navegação, medida pelo percentual de rotas completadas com sucesso sem colisões, também foi alta, indicando um sistema de navegação robusto.

V. CONCLUSÕES

Este projeto desenvolveu um sistema de navegação autônoma robusto e eficiente utilizando o Turtlebot4. Conseguimos integrar com sucesso dados de LIDAR e câmera para melhorar o mapeamento e a navegação, utilizando técnicas avançadas de SLAM visual, fusão de sensores e detecção de objetos. Os resultados obtidos demonstram a eficácia do sistema em ambientes dinâmicos, com alta precisão de mapeamento e navegação confiável.

As lições aprendidas incluem a importância da calibração precisa dos sensores e a necessidade de ajustes contínuos para otimização da navegação. Os passos futuros envolvem a extensão do sistema para ambientes mais complexos e a integração de técnicas de aprendizado profundo mais avançadas para melhorar ainda mais a percepção e a navegação. Além disso, explorar a integração de novos sensores e algoritmos pode proporcionar melhorias significativas no desempenho do sistema.

Em resumo, este projeto apresentou uma abordagem abrangente e técnica para o desenvolvimento de um protótipo de navegação autônoma, contribuindo para o avanço da pesquisa em robótica e veículos autônomos.

REFERENCES

- [1] S. Riisgaard e M. Rufus. “*SLAM for Dummies: A Tutorial Approach to Simultaneous Localization and Mapping*”. MIT. Disponível em: PDF. Acesso em: 10 de Julho de 2024.
- [2] S. Macenski, T. Foote, B. Gerkey, C. Lalancette e W. Woodall. “Robot Operating System 2: Design, architecture, and uses in the wild”. Science Robotics vol. 7. Disponível em: Science Robotics. Acesso em: 10 de Julho de 2024.
- [3] A. Merzlyakov, S. Macenski, “*A Comparison of Modern General-Purpose Visual SLAM Approaches*”, New York: Academic, Disponível em: Arxiv . Acesso em: 10 de Julho de 2024.
- [4] J. Redmon, S. Divvala e R. Girshick. “*You Only Look Once: Unified, Real-Time Object Detection*” J. Name Stand. Abbrev. Disponível em: Arxiv . Acesso em: 10 de Julho de 2024.
- [5] A. Gupta e A. Anpalagan. “*Deep learning for object detection and scene perception in self-driving cars: Survey, challenges, and open issues*”. Ryerson University, 350 Victoria Street, Toronto. Disponível em: ScienceDirect. Acesso em: 10 de Julho de 2024.