



Instituto Politécnico  
de Castelo Branco  
Escola Superior  
de Tecnologia

• **A P T I V** •

## **Sistema Robótico para Recolha de Objetos em Ambiente Industrial**

### **Aluno**

António Miguel Oliveira Alves, no 20210476

Finalista da Licenciatura em Engenharia e Gestão Industrial

### **Orientador**

Doutor Paulo Jorge Sequeira Gonçalves

Relatório de Projeto apresentado à Escola Superior de Tecnologia do Instituto Politécnico de Castelo Branco e à empresa APTIV para cumprimento dos requisitos necessários à obtenção do grau de Licenciado em Engenharia e Gestão Industrial

**Julho 2024**



## Composição do júri

Presidente do júri

Doutor Paulo Jorge Sequeira Gonçalves

Professor coordenador na Escola Superior de Tecnologia do Instituto Politécnico de Castelo Branco

Vogais

Doutor Paulo Jorge Sequeira Gonçalves

Professor Coordenador na Escola Superior de Tecnologia do Instituto Politécnico de Castelo Branco

Engenheiro Luís Leão

Licenciado em Engenharia e Gestão Industrial. Responsável de manutenção da Empresa APTIV,

Doutor Nuno Octávio Garcia Fernandes

Professor Coordenador na Escola Superior de Tecnologia do Instituto Politécnico de Castelo Branco

## **Agradecimentos**

O meu sincero agradecimento ao meu orientador, Doutor Paulo Gonçalves por todo o apoio, dedicação, paciência, amizade, assim como todos os ensinamentos e conselhos transmitidos.

Um agradecimento ao Mestre Rodrigo Bernardo pelo apoio fornecido neste trabalho.

Agradeço a oportunidade e colaboração do Eng.º Luís Leão da Aptiv.

Por fim, agradeço as condições de acolhimento do Laboratório de Robótica e Equipamentos Inteligentes da Escola Superior de Tecnologia do Instituto Politécnico de Castelo Branco.

## **Resumo**

Este relatório descreve o desenvolvimento inicial de um sistema robótico para a distribuição e recolha de objetos em ambiente industrial na empresa APTIV. O projeto foi motivado pela necessidade de otimizar a distribuição e recolha de caixas SUC na fábrica, substituindo processos manuais por um sistema automatizado utilizando o robô MiR100. Para esse robô, foi projetado um carro com um sistema robótico de três graus de liberdade para manipular as caixas SUC. O carro é acoplado e puxado de forma autónoma pelo MiR100.

A implementação do projeto incluiu a instalação e configuração de um sistema de deteção de caixas SUC utilizando uma câmara RGB-D, a integração com o ROS (Robot Operating System), e o desenvolvimento de software para controlo de motores e leitura de sensores, baseado em Arduino num computador industrial. Os testes de validação confirmaram a precisão da navegação e movimentação do MiR100 de forma autónoma, utilizando a API do MiR100. Este projeto demonstra a viabilidade da automação na logística interna, destacando-se pela adaptação de tecnologias existentes para solucionar problemas específicos da empresa.

## **Palavras-chave**

Robótica

Automação Industrial

Gestão de Armazéns

Distribuição de Materiais

## Abstract

This report describes the initial development of a robotic system for the distribution and collection of objects in an industrial environment at APTIV. The project was motivated by the need to optimize the distribution and collection of SUC boxes in the factory, replacing manual processes with an automated system using the MiR100 robot. For this robot, a cart with a three-degree-of-freedom robotic system was designed to handle the SUC boxes. This cart is autonomously coupled and pulled by the MiR100.

The project implementation included the installation and configuration of an SUC box detection system using an RGB-D camera, integration with ROS (Robot Operating System), and development of software for motor control and sensor reading, based on Arduino and an industrial computer. Validation tests confirmed the accuracy of the MiR100's autonomous navigation and movement, utilizing the MiR100 API. This project demonstrates the feasibility of automation in internal logistics, highlighting the adaptation of existing technologies to solve specific problems for the company.

## Keywords

Robotics

Industrial Automation

Warehouse Management

Material Distribution

## Índice geral

|   |    |
|---|----|
| 1. Introdução.....  | 1  |
| Contribuições.....  | 3  |
| Organização do relatório.....   | 4  |
| 2. Estado da Arte.....  | 5  |
| 2.1. Robótica na Logística Industrial .....   | 5  |
| 2.2. Tipos de Robôs para Logística Industrial .....                                     | 6  |
| 2.2.1. Robôs Autónomos de Navegação (AMRs) .....  | 6  |
| 2.2.2. Robôs de Picking e Packing.....  | 7  |
| 2.2.3. Robôs Antropomórfico.....  | 7  |
| 2.2.4. Sistemas de Armazém Automático (AS/RS) .....                                     | 7  |
| 2.2.5. Drones de Inventário .....   | 8  |
| 2.2.6. Robôs Colaborativos (Cobots).....  | 8  |
| 2.2.7. Veículos Guiados Automaticamente (AGVs).....                                     | 9  |
| 2.2.8. Robôs de Paletização .....   | 9  |
| 2.3. Robôs Existentes com Capacidades Avançadas de Recolha e Distribuição. <b>Erro!</b> |    |
| <b>Marcador não definido.</b>   |    |
| 2.3.1. Fetch Robotics Freight500 e Freight1500 com FetchArm.....                        | 10 |
| 2.3.2. Omron LD Series com braço robótico .....   | 11 |
| 2.3.3. MIR100 com MIRHook e MIR250 com braços robóticos.....                            | 11 |
| 2.3.4. KUKA KMR iiwa .....  | 12 |
| 2.3.5. Robotnik RB-KAIROS+ .....  | 12 |
| 2.3.6. Magazino SOTO .....  | 13 |
| 2.3.7. Locus Robotics LocusBots.....  | 14 |
| 2.4. Discussão sobre as Soluções Encontradas .....                                      | 15 |
| 2.4.1 Verificação de Capacidade de Recolher Caixas SUC.....                             | 15 |
| Omron LD Series com braço robótico.....   | 16 |
| MIR100 com MIRHook e MIR250 com braços robóticos.....                                   | 16 |
| KUKA KMR iiwa.....  | 16 |
| Robotnik RB-KAIROS+.....  | 16 |
| Magazino SOTO .....   | 17 |
| Locus Robotics LocusBots.....   | 17 |

|  |    |
|--|----|
| 3. Metodologia para Projeto do Sistema Robótico.....   | 18 |
| 3.1. Análise de requisitos.....  | 19 |
| 3.1.1. Carro de Transporte .....   | 19 |
| 3.1.2. Dispositivo de Manipulação.....   | 19 |
| 3.1.3. Navegação e Transporte do Carro.....  | 19 |
| 3.2. Desenvolvimento do Projeto Global.....  | 19 |
| 3.2.1. Carro de Transporte .....   | 20 |
| 3.2.2. Mecanismo para manipulação de caixas SUC .....  | 21 |
| 3.2.3. Navegação do MIR100 com o Carro acoplado .....  | 21 |
| 3.3. Especificações do Robô MIR100.....  | 22 |
| Especificações Técnicas:.....  | 23 |
| Segurança e Navegação: .....   | 23 |
| Controlo e Configuração:.....  | 23 |
| 3.4. Requisitos para o Mecanismo de Armazenamento no Carro .....   | 24 |
| 4. Desenvolvimento do Sistema .....  | 26 |
| Neste capítulo, abordarei o desenvolvimento do mecanismo manipulador e a estrutura do carro onde será instalado o mecanismo..... |    |
| 26   | 26 |
| 4.1. Estrutura Mecânica.....   | 26 |
| 4.1.1. Design para Protótipo .....   | 27 |
| 4.1.2 Seleção dos componentes.....   | 28 |
| 4.1.3 Características dos atuadores selecionados.....  | 30 |
| NEMA 17.....   | 30 |
| NEMA 32.....   | 30 |
| Caixas de Redução Sem Fim.....   | 31 |
| Princípios de Funcionamento: .....   | 31 |
| 4.2. Montagem.....   | 32 |
| 4.3.3 Cinemática do mecanismo de 3 graus de liberdade desenvolvido.....  | 33 |
| 4.4. O robô movel MIR100.....  | 37 |
| Definição e Instalação da estação de Carregamento:.....  | 37 |
| Implementação do Sistema de Captura do Carrinho.....   | 37 |
| Desenvolvimento e Realização de Missões:.....  | 38 |
| Implementação de um programa para deteção de tags .....  | 38 |
| 5. Programação e Validação do sistema robótico.....  | 40 |

|   |    |
|---|----|
| 5.1. Arquitetura do software desenvolvido .....   | 40 |
| 5.2. Desenvolvimento de programas .....   | 41 |
| Linguagens de Programação Utilizadas:.....  | 41 |
| Sistema Operativo:.....   | 41 |
| Preparação para melhorias futuras:.....   | 41 |
| Programa de Cinemática: .....   | 41 |
| Controlo de Motores de Passo:.....  | 41 |
| Integração com MIR100:.....   | 42 |
| Localização de Tags: .....  | 42 |
| Teste Global de Mecanismos:.....  | 42 |
| 5.3. Validação e testes do MIR100 .....   | 43 |
| 5.4. Validação e testes do mecanismo de aproximação a tag.....                                    | 46 |
| 5.5. Testes global de Controlo e Comunicação.....   | 48 |
| Diagrama do ROS .....   | 49 |
| Nó Principal do Programa: /mission_client .....   | 50 |
| Nó Responsável pela Aproximação às Tags: /tag_move_1 .....  | 52 |
| Os nós Responsáveis pelo controlo dos motores do mecanismo:<br>/driver_motores, /serial_node..... | 54 |
| Nós Responsáveis pelo Controlo do MIR100:/MIR100_missions,<br>/mission_state_publisher .....      | 56 |
| Nó Responsável pela Detecção das Tags: /apriltag.....   | 58 |
| Conclusão da Validação do Teste Global.....   | 59 |
| 6. Conclusão.....   | 60 |
| 7. Referências.....   | 60 |

## Índice de figuras

|  |                                     |
|--|-------------------------------------|
| Figura 1 - Fachada APTIV Castelo Branco .....                                | 2                                   |
| Figura 2-Robô Móvel Autônomo de (AMR) .....                                  | 6                                   |
| Figura 3-Robô de Picking e Packing.....                                      | 7                                   |
| Figura 4-Robô Antropomórfico.....  | 7                                   |
| Figura 5-Sistemas de armazém automáticos (AS/RS) .....                       | 7                                   |
| Figura 6-Drones de Inventário .....  | 8                                   |
| Figura 7-Robô Colaborativo (Cobot) .....                                     | 8                                   |
| Figura 8-Veículos Guiados Automáticos (AGVs) .....                           | 9                                   |
| Figura 9-Robôs de Paletização .....  | 9                                   |
| Figura 10-Fetch Robotics Freight500 e Freight1500 com FetchArm .....         | 10                                  |
| Figura 11-Omron LD Series com braço robótico.....                            | 11                                  |
| Figura 12-MIR100 com MIRHook e MIR250 com braços robóticos.....              | 11                                  |
| Figura 13-KUKA KMR iiwa .....  | 12                                  |
| Figura 14-Robotnik RB-KAIROS+.....   | 12                                  |
| Figura 15-Magazino SOTO.....   | 13                                  |
| Figura 16-Locus Robotics LocusBots.....                                      | 14                                  |
| Figura 17-Caixas SUC.....  | 15                                  |
| Figura 18 -Desenho do Carro de Transporte .....                              | 20                                  |
| Figura 19-MIR 100 .....  | 22                                  |
| Figura 20-Desenho do mecanismo telescópico.....                              | 24                                  |
| Figura 21-Desenho de mecanismo elevador .....                                | 24                                  |
| Figura 22-Desenho de mecanismo de movimento lateral .....                    | 25                                  |
| Figura 23-caxas tipo suc.....  | 25                                  |
| Figura 24-Desenho da estrutura do carrinho 2d[mm].....                       | 26                                  |
| Figura 25-Desenho inicial da estrutura e mecanismo .....                     | 27                                  |
| Figura 26 - NEMA 17 .....  | 30                                  |
| Figura 27 - NEMA 32.....   | 30                                  |
| Figura 28-Caixas de Redução Sem Fim .....                                    | 31                                  |
| Figura 29-Estrutura com MIR100 para testes.....                              | 32                                  |
| Figura 30-Montagem dos atuadores do mecanismo .....                          | 32                                  |
| Figura 31-Formula geral de Denavit-Hartenberg .....                          | 34                                  |
| Figura 32-Sistema de Captura do Carrinho ao MIR100 .....                     | 37                                  |
| Figura 33-Câmara Intel RealSense.....  | 38                                  |
| Figura 34-Logotipo VISP .....  | 39                                  |
| Figura 35-Arquitetura do software desenvolvido.....                          | 40                                  |
| Figura 36-Eschema de calibração do MIR 100 .....                             | 43                                  |
| Figura 37-Mecanismo de aproximação em y .....                                | 47                                  |
| Figura 38-diagrama ROS 1/3 .....   | <b>Erro! Marcador não definido.</b> |
| Figura 39-diagrama ROS 2/3 .....   | <b>Erro! Marcador não definido.</b> |
| Figura 40-diagrama ROS 3/3 .....   | 49                                  |
| Figura 41-diagrama ROS nó /mission_client .....                              | 51                                  |
| Figura 42- diagrama ROS no /tag_move_1.....                                  | 53                                  |
| Figura 43-diagrama ROS nós /driver_motores, /serial_node.....                | 55                                  |
| Figura 44-diagrama ROS nos :/MIR100_missions, /mission_state_publisher ..... | 57                                  |
| Figura 45-diagram ROS nó /apriltag .....                                     | 58                                  |

## **Lista de tabelas**

|   |    |
|---|----|
| Tabela 1-Medidas das caixas SUC [mm].....             | 15 |
| Tabela 2- Legenda dos referenciais da Cinemática..... | 33 |
| Tabela 3-Tabela de parâmetros de cinemática .....     | 34 |
| Tabela 4-Dados de Calibração de MIR 100 .....         | 44 |

## Lista de abreviaturas, siglas e acrónimos

**AGVs:** *Automated Guided Vehicles* (Veículos Guiados Automáticos)

**AMRs:** *Autonomous Mobile Robots* (Robôs Móveis Autónomos)

**API:** *Application Programming Interface* (Interface de Programação de Aplicações)

**APTIV:** *The company where the project was carried out* (Nome da empresa onde o projeto foi realizado)

**AS/RS:** *Automated Storage and Retrieval Systems*

**tag:** *A type of fiducial marker used in robotics and computer vision for object identification and localization.* (Um tipo de marcador fiduciário utilizado em robótica e visão computacional para identificação e localização de objetos.)

**C++:** *Programming language* (Linguagem de programação)

**CNC:** *Computer Numerical Control* (Controlo Numérico Computadorizado)

**Cobots:** *Collaborative Robots* (Robôs Colaborativos)

**GPS:** *Global Positioning System* (Sistema de Posicionamento Global)

**Li-ion:** *Lithium-Ion* (Íon de Lítio)

**MIR:** *Mobile Industrial Robots*

**PBS:** *Portable Batch System* (Sistema de Lotes Portáteis)

**ROS:** *Robot Operating System* (Sistema Operacional de Robôs)

**SUC:** *Standard Utility Container* (Um tipo de caixa padronizada utilizada para armazenar e transportar materiais em ambientes industriais.)

**ViSP:** *Visual Servoing Platform* (Plataforma de Servo Visual)

**XYZ:** *Three-dimensional coordinates used to define robot movements* (Coordenadas tridimensionais usadas para definir os movimentos dos robôs)

## 1. Introdução

Este relatório descreve o desenvolvimento inicial de um sistema robótico para a distribuição e recolha de objetos em ambiente industrial. O projeto aborda os desafios de automatizar a recolha e distribuição de caixas SUC numa zona fabril. Visa reduzir a carga de trabalho dos funcionários e poderá ser rentabilizado a longo prazo devido à possibilidade de integração com o armazém e o stock dos postos de trabalho reduzindo a complexidade da distribuição atual das caixas SUC.

A proposta centra-se no desenvolvimento de um sistema autónomo capaz de recolher e distribuir caixas do tipo SUC, utilizando um robô móvel MIR100 existente na empresa. O desenvolvimento envolveu a integração de diversos componentes mecânicos e eletrónicos, além da implementação de algoritmos de controlo e navegação dos vários subsistemas.

A motivação para este projeto surgiu da necessidade de melhorar o controlo na distribuição e recolha de caixas SUC pelos postos de trabalho. A implementação de um sistema robótico melhorará o controlo dos stocks da fábrica e reduzir o esforço manual, proporcionando um ambiente de trabalho mais organizado.

Para alcançar os objetivos propostos, foram realizadas as seguintes etapas:

- **Instalação e Configuração de Sensores:** Sensores de laser e câmaras foram integrados para medir distâncias e visualizar o ambiente.
- **Desenvolvimento de Mecanismos de Armazenamento:** Um mecanismo de distribuição e recolha com armazenamento de caixas SUC foi adaptado ao MIR100, permitindo a manipulação das caixas.
- **Implementação de Algoritmos de Controlo e Navegação:** Algoritmos específicos foram desenvolvidos para garantir a precisão nas tarefas de recolha e distribuição, utilizando a visão computacional e o sistema de controlo do MIR100.
- **Integração com o Sistema de Controlo:** O sistema foi configurado para realizar tarefas de forma autónoma, utilizando diversos algoritmos implementados no ROS, utilizando um computador industrial e um Arduino.

## 1.1. Apresentação da Empresa

A APTIV é uma empresa multinacional que se destaca no setor de tecnologia e soluções de mobilidade automóvel. Especializada na fabricação e no desenvolvimento de componentes eletrónicos e sistemas para automóveis, incluindo soluções para conectividade avançada e sistemas de segurança. A Aptiv está presente em Castelo Branco há mais de 34 anos. Atualmente, a empresa conta com mais de 1.300 funcionários.

Ao constatar que os funcionários da empresa tinham dificuldades em estabelecer rotas para a distribuição e recolha de caixas tipo SUC pelos postos de trabalho, foi proposta a criação de um robô que pudesse realizar esse serviço de forma mais controlada, permitindo um cálculo otimizado das rotas de distribuição e consumo das matérias-primas da empresa.

Para evitar um grande investimento inicial, foi necessário encontrar um mecanismo que permitisse testar essas hipóteses. O projeto teve como ponto de partida um robô MIR100 já existente na empresa, o que levou ao desenvolvimento de um mecanismo adaptado ao robô MIR 100 para a recolha e distribuição das matérias-primas que se encontram em caixas do modelo SUC numa zona fabril.



*Figura 1 - Fachada APTIV Castelo Branco*

## Contribuições.

O desenvolvimento deste sistema robótico contribui significativamente para a empresa, destacando-se três áreas principais:

**Carro de Transporte:** Um carro de transporte foi projetado para otimizar a movimentação de caixas dentro da zona fabril. Este carro, acoplado ao robô MIR100, permite ainda uma manipulação das matérias-primas, facilitando o transporte entre diferentes pontos de trabalho e reduzindo o tempo de deslocamento.

**Manipulador para Caixas SUC:** Foi criado um manipulador específico para manipular com caixas do tipo SUC. Este mecanismo garante a recolha e a distribuição precisas das caixas, adaptando-se às suas dimensões e pesos variados. Isso aumenta a operacionalidade e minimiza o risco de danos aos materiais, melhorando a qualidade e a confiabilidade dos processos logísticos.

**Navegação do MIR100 com o Carro:** Foi desenvolvida em python utilizando a API do MIR100 para controlar deste robô. Possibilitou uma navegação autónoma e precisa, otimizando as rotas de transporte e garantindo o funcionamento do sistema.

Essas contribuições visam ser um passo para a empresa, promover grandes melhorias na organização de stocks, na organização do ambiente de trabalho e logística, ao mesmo tempo em que reduzem custos operacionais e aprimoram a organização dos processos industriais.

## Organização do relatório

A **Introdução** apresenta o objetivo do trabalho, a motivação para o desenvolvimento do sistema robótico e um resumo das principais etapas realizadas. No capítulo do **Estado da Arte**, descreve-se a aplicação da robótica na logística industrial, destacando diferentes tipos de robôs e suas funções específicas, além de discutir soluções existentes e sua aplicabilidade.

A capítulo de **Metodologia para Projeto do Sistema Robótico** detalha a análise de requisitos e o desenvolvimento do projeto, incluindo o carro de transporte, manipulador de caixas SUC, e a navegação do MIR100, além da validação e montagem dos componentes. O capítulo dedicado ao **Desenvolvimento do Sistema** apresenta o design e montagem da estrutura mecânica, a configuração dos sistemas de controlo, e a integração com o MIR100.

No capítulo **Programação e Validação**, são apresentados os testes realizados para verificar a precisão do MIR100, a implementação do sistema de leitura de tags, e a validação dos componentes mecânicos e de controlo. Finalmente, a **Conclusão** resume as principais realizações do projeto, identifica limitações e desafios encontrados, e sugere direções para trabalhos futuros.

## 2. Estado da Arte

Neste capítulo, são exploradas as aplicações atuais da robótica na logística industrial, destacando como os avanços tecnológicos estão a transformar os processos logísticos. São discutidos os diferentes tipos de robôs utilizados na indústria.

São analisados os benefícios trazidos pela automação e robótica: maior precisão nas operações, redução de custos laborais, melhoria nas condições de trabalho, integração com sistemas avançados e práticas sustentáveis. São ainda avaliadas as soluções robóticas mais avançadas disponíveis no mercado e discutidas as suas capacidades de atender às necessidades específicas de manipulação e distribuição de materiais em ambientes industriais.

O capítulo conclui-se com uma discussão sobre as inovações tecnológicas e as adaptações necessárias para que esses robôs atendam plenamente às necessidades da indústria, destacando os desafios e as oportunidades no desenvolvimento de sistemas robóticos avançados.

### 2.1. Robótica na Logística Industrial

A robótica na logística industrial[1] tem-se tornado um elemento crucial para a modernização e otimização das cadeias de abastecimento. A aplicação de robôs em diversos processos logísticos, como transporte, armazenamento de materiais, processamento de pedidos e gestão de inventário, tem transformado a forma como as operações são conduzidas, trazendo uma série de benefícios significativos.

- **Eficiência e Precisão:** A automação de tarefas rotineiras e repetitivas através de robôs permite um aumento substancial na eficiência e precisão das operações logísticas. Robôs são capazes de realizar atividades com um nível de consistência que supera o desempenho humano, minimizando erros e aumentando a produtividade.
- **Redução de Custos Laborais:** Com a introdução de robôs, as empresas podem reduzir os custos associados ao trabalho humano, especialmente em tarefas que não requerem capacidades específicas e são fisicamente exigentes. Desta forma, é possível que a força de trabalho humana seja direcionada para funções mais estratégicas e de maior valor acrescentado para a empresa.
- **Melhoria nas Condições de Trabalho:** Robôs assumem tarefas perigosas e repetitivas, reduzindo a exposição dos trabalhadores a riscos físicos e ambientes insalubres. Este facto, não só melhora a segurança no local de trabalho, mas também contribui para um ambiente de trabalho mais saudável e satisfatório.
- **Integração com Sistemas Avançados:** A combinação de robôs móveis autónomos (AMRs) e sistemas automatizados (AS/RS) com softwares avançados de gestão de armazém permite uma coordenação precisa das operações logísticas. Essa integração possibilita uma resposta rápida às necessidades do mercado, garantindo que os produtos certos estejam no lugar certo e na hora certa.

- **Práticas Sustentáveis:** A automação também promove práticas sustentáveis ao otimizar o uso de recursos e reduzir desperdícios. Robôs são capazes de operar com maior eficiência energética e utilizar o espaço de armazenamento de forma mais eficaz, contribuindo para a sustentabilidade ambiental das operações logísticas.

- **Flexibilidade e Escalabilidade:** A capacidade de adaptar rapidamente as operações para atender a mudanças na procura é outro benefício importante da robótica na logística. Robôs podem ser reprogramados ou reconfigurados para executar novas tarefas ou aumentar a escala das operações conforme necessário, proporcionando uma grande flexibilidade às empresas.

Em resumo, a robótica na logística industrial não só melhora a eficiência e a precisão das operações, mas também reduz custos, melhora as condições de trabalho, promove a sustentabilidade e aumenta a flexibilidade e escalabilidade das operações. A adoção crescente de tecnologias robóticas está a moldar o futuro da logística, criando cadeias de abastecimento mais inteligentes, seguras e sustentáveis.

## 2.2. Tipos de Robôs para Logística Industrial

A robótica na logística industrial abrange uma variedade de robôs, cada um projetado para desempenhar funções específicas e otimizar diferentes aspetos das operações logísticas. Em seguida, são apresentados os principais tipos de robôs utilizados na logística industrial:

### 2.2.1. Robôs Autônomos de Navegação (AMRs)

Os Robôs Móveis Autônomos (AMRs) operam de forma independente utilizando tecnologias avançadas de navegação e sensores. São capazes de ajustar as suas rotas em tempo real para otimizar o fluxo de trabalho e evitar obstáculos. Estes robôs são ideais para o transporte de materiais dentro de armazéns e fábricas, garantindo a movimentação .



*Figura 2-Robô Móvel Autônomo de (AMR)*

### 2.2.2. Robôs de Picking e Packing

Estes robôs automatizam o processo de seleção e embalagem de produtos, funções essenciais em centros de distribuição e armazéns. Equipados com sistemas de visão e algoritmos de inteligência artificial, os robôs de picking e packing podem identificar e manipular produtos com alta precisão e velocidade, reduzindo significativamente o tempo de processamento das tarefas requeridas.



Figura 3-Robô de Picking e Packing

### 2.2.3. Robôs Antropomórfico

Utilizados principalmente para manipular objetos, os robôs antropomórficos possuem braços robóticos com diversas articulações, proporcionando alta precisão e flexibilidade. São capazes de executar uma ampla gama de tarefas, desde a movimentação de objetos pesados até operações de montagem.



Figura 4-Robô Antropomórfico

### 2.2.4. Sistemas de Armazém Automático (AS/RS)

Estes sistemas consistem em robôs e equipamentos automatizados que otimizam o espaço de armazenamento de objetos. Utilizando software avançado de gestão de armazéns, os AS/RS podem armazenar produtos de maneira muito eficiente, melhorando a utilização do espaço e reduzindo o tempo de acesso aos objetos armazenados.



Figura 5-Sistemas de armazém automáticos (AS/RS)

### 2.2.5. Drones de Inventário

Os Drones de inventário são utilizados para realizar inspeções e contagens de Stock em grandes armazéns. Equipados com câmaras e sensores, esses Drones podem navegar autonomamente pelos corredores dos armazéns, capturando dados de inventário de maneira rápida e precisa, reduzindo a necessidade de inspeções manuais.



*Figura 6-Drones de Inventário*

### 2.2.6. Robôs Colaborativos (Cobots)

Os robôs colaborativos, ou cobots, são projetados para trabalhar ao lado de humanos em ambientes colaborativos.

São seguros para operar próximo de trabalhadores humanos, e partilhar tarefas que exigem uma combinação de habilidades humanas e de máquinas. Os Cobots são frequentemente utilizados em linhas de montagem, onde a flexibilidade e a interação com trabalhadores humanos são essenciais.



*Figura 7-Robô Colaborativo (Cobot)*

### 2.2.7. Veículos Guiados Automaticamente (AGVs)

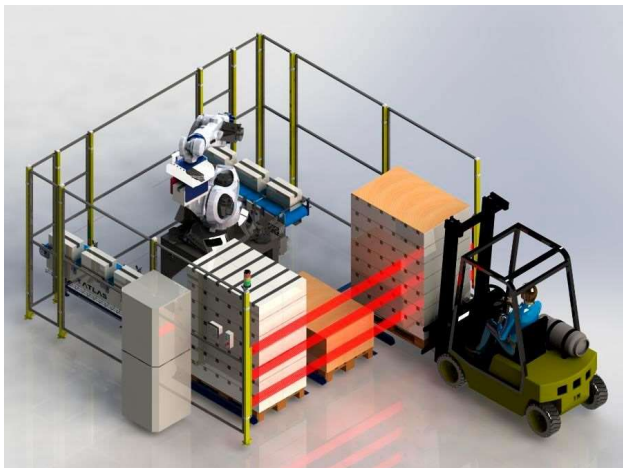
Os Veículos Guiados Automaticamente (AGVs) seguem trajetos pré-definidos para transportar cargas pesadas dentro de instalações industriais. Utilizando guias magnéticos, lasers ou outras tecnologias de orientação, os AGVs são eficazes na movimentação de materiais entre diferentes pontos de um armazém ou fábrica, oferecendo uma solução confiável e eficiente para o transporte interno.



*Figura 8-Veículos Guiados Automáticos (AGVs)*

### 2.2.8. Robôs de Paletização

Especializados na organização e para empilhar objetos em paletes, os robôs de paletização são essenciais para operações de expedição e armazenamento. São capazes de manipular caixas e outros objetos com precisão, obtendo-se paletes estáveis ao otimizar o espaço de armazenamento e transporte.



*Figura 9-Robôs de Paletização*

A diversidade de robôs disponíveis para a logística industrial permite que as empresas escolham as soluções mais adequadas para suas necessidades específicas, aumentando a eficiência, reduzindo custos e melhorando a segurança e a qualidade das operações logísticas.

## 2.3. Robôs Existentes com Capacidades Avançadas de Recolha e Distribuição

Para aplicações industriais que requerem robôs móveis com capacidades de manipulação, várias soluções inovadoras estão disponíveis no mercado. Esses robôs combinam a mobilidade de plataformas autônomas com a destreza de braços robóticos, oferecendo uma funcionalidade semelhante ao Magazino SOTO [2].

### 2.3.1. Fetch Robotics Freight500 e Freight1500 com FetchArm

- **Descrição:** A série Freight de robôs móveis autônomos (AMRs) da Fetch Robotics pode ser equipada com o FetchArm, um braço robótico que permite manipulação de materiais. O Freight500 pode transportar até 500 kg, enquanto o Freight1500 é capaz de transportar até 1500 kg.
- **Aplicação:** Automação do transporte interno e manipulação de materiais em fábricas e armazéns.
- **Fonte:** Fetch Robotics[3].



Figura 10-Fetch Robotics Freight500 e Freight1500 com FetchArm

### 2.3.2. Omron LD Series com braço robótico

- **Descrição:** A série LD de robôs móveis autônomos da Omron pode ser integrada com braços robóticos para manipulação de objetos. Eles são projetados para navegar autonomamente e realizar tarefas de transporte e manipulação.
- **Aplicação:** Operações de picking, transporte de peças e montagem em ambientes industriais.
- **Fonte:** Omron LD Series[4].



Figura 11-Omron LD Series com braço robótico

### 2.3.3. MIR100 com MIRHook e MIR250 com braços robóticos

- **Descrição:** A Mobile Industrial Robots (MIR) oferece AMRs como o MIR100 e o MIR250, que podem ser equipados com acessórios como o MIRHook para puxar carrinhos ou braços robóticos para manipulação de materiais.
- **Aplicação:** Transporte interno e manipulação de materiais em armazéns e centros de distribuição.
- **Fonte:** MIR Robots[5].



Figura 12-MIR100 com MIRHook e MIR250 com braços robóticos

### 2.3.4. KUKA KMR iiwa

- **Descrição:** O KMR iiwa combina a mobilidade de uma plataforma robótica autónoma com a flexibilidade e sensibilidade do braço robótico LBR iiwa. Esta combinação permite que o robô realize tarefas complexas de manipulação em ambientes dinâmicos.
- **Aplicação:** Montagem, pick-and-place, e operações de transporte em fábricas.
- **Fonte:** KUKA KMR iiwa[6].



Figura 13-KUKA KMR iiwa

### 2.3.5. Robotnik RB-KAIROS+

- **Descrição:** RB-KAIROS+ da Robotnik é um robô móvel autónomo que integra um braço robótico UR da Universal Robots. É projetado para manipulação e transporte de materiais em ambientes industriais.
- **Aplicação:** Tarefas de logística interna, montagem, e operações de pick-and-place.
- **Fonte:** Robotnik RB-KAIROS+[7].



Figura 14-Robotnik RB-KAIROS+

### 2.3.6. Magazino SOTO

- **Descrição:** Magazino oferece o SOTO, um robô móvel autónomo com capacidades de manipulação, projetado para operações de intralogística. É capaz de carregar, transportar e entregar caixas e peças em ambientes industriais.
- **Aplicação:** Linhas de produção e armazéns para otimizar a logística interna.
- **Fonte:** Magazino SOTO[2].



Figura 15-Magazino SOTO

### 2.3.7. Locus Robotics LocusBots

- **Descrição:** LocusBots são robôs móveis autônomos que podem ser equipados com vários acessórios, incluindo braços robóticos para picking e manipulação de materiais. Trabalham colaborativamente com operadores humanos.
- **Aplicação:** Centros de distribuição e armazéns para operações de coleta e transporte de pedidos.
- **Fonte:** Locus Robotics[8].

Estes robôs representam algumas das soluções mais avançadas e flexíveis disponíveis atualmente para a automação de tarefas de recolha e distribuição em ambientes industriais. Combinam mobilidade autônoma com a capacidade de manipulação precisa, permitindo uma maior eficiência e produtividade nas operações logísticas.



*Figura 16-Locus Robotics LocusBots*

## 2.4. Discussão sobre as Soluções Encontradas

A adaptação de robôs móveis para tarefas específicas de manipulação é um campo dinâmico e repleto de inovações tecnológicas. De seguida, é discutido como os robôs mencionados na secção anterior fornecem informações valiosas e direções práticas para o desenvolvimento de uma solução para a recolha e distribuição de caixas tipo SUC. É ainda verificado se estes robôs têm a capacidade de recolher e armazenar caixas SUC desde uma altura a partir do chão, de 80 cm até 2 m. Foi também verificada a capacidade máxima de armazenamento no stock interno do robô.

Dimensões das Caixas:

- Caixa A: 20 cm x 10 cm x 11 cm
- Caixa B: 32 cm x 15 cm x 13 cm
- Caixa C: 40 cm x 20 cm x 17 cm

### 2.4.1 Verificação de Capacidade de Recolher Caixas SUC

Nesta secção, é avaliado se os robôs mencionados são capazes de recolher e manipular as caixas SUC de acordo com suas dimensões e requisitos operacionais.

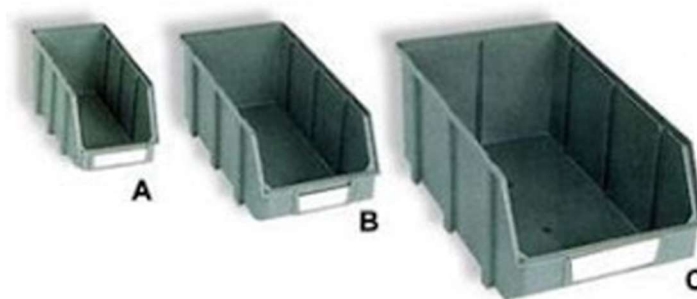


Figura 17-Caixas SUC

| [cm] | A  | B  | C  |
|------|----|----|----|
| c    | 20 | 32 | 40 |
| l    | 10 | 15 | 20 |
| h    | 11 | 13 | 17 |

Tabela 1-Medidas das caixas SUC [mm]

Fetch Robotics Freight500 e Freight1500 com FetchArm

- Capacidade de Recolher/Armazenar: O FetchArm pode ser configurado para alcançar alturas variáveis, adequando-se ao intervalo de 20 cm a 2 m. Tem precisão suficiente para manipular as caixas A, B e C.

- Capacidade Máxima de Armazenamento: Dependendo do tamanho das caixas, o Freight500 pode armazenar aproximadamente 20 caixas A, 10 caixas B ou 6 caixas C. O Freight1500 pode armazenar até 60 caixas A, 30 caixas B ou 18 caixas C.
- Fonte: Fetch Robotics[3].

### **Omron LD Series com braço robótico**

- Capacidade de Recolher/Armazenar: Os braços robóticos podem ser configurados para operar dentro do intervalo de 20 cm a 2 m. Podem manipular as caixas A, B e C com eficiência.
- Capacidade Máxima de Armazenamento: A série LD pode transportar cargas de até 130 kg, o que pode corresponder a aproximadamente 13 caixas A, 8 caixas B ou 5 caixas C.
- Fonte: Omron LD Series[7].

### **MIR100 com MIRHook e MIR250 com braços robóticos**

- Capacidade de Recolher/Armazenar: Os braços robóticos integrados ao MIR podem ser configurados para alcançar alturas variáveis, atendendo ao intervalo de 20 cm a 2 m. São capazes de manipular as caixas A, B e C.
- Capacidade Máxima de Armazenamento: O MIR100 pode transportar até 100 kg, equivalente a aproximadamente 10 caixas A, 6 caixas B ou 4 caixas C. O MIR250 pode carregar até 250 kg, armazenando aproximadamente 25 caixas A, 15 caixas B ou 10 caixas C.
- Fonte: MIR Robots[5].

### **KUKA KMR iiwa**

- Capacidade de Recolher/Armazenar: O braço robótico LBR iiwa pode ser programado para operar dentro do intervalo de altura especificado. Pode manipular com precisão as caixas A, B e C.
- Capacidade Máxima de Armazenamento: A capacidade de carga do KMR iiwa varia, mas geralmente pode manipular e transportar cargas significativas. Pode armazenar aproximadamente 20 caixas A, 12 caixas B ou 8 caixas C dependendo da configuração.
- Fonte: KUKA KMR iiwa[6].

### **Robotnik RB-KAIROS+**

- Capacidade de Recolher/Armazenar: O braço robótico UR pode ser ajustado para operar em alturas de 20 cm a 2 m. É adequado para manipular as caixas A, B e C.
- Capacidade Máxima de Armazenamento: A capacidade de carga do RB-KAIROS+ permite armazenar aproximadamente 20 caixas A, 10 caixas B ou 6 caixas C.
- Fonte: Robotnik RB-KAIROS+[7].

### **Magazino SOTO**

- Capacidade de Recolher/Armazenar: O SOTO é capaz de operar dentro do intervalo de altura de 20 cm a 2 m. Pode manipular com eficiência as caixas A, B e C.
- Capacidade Máxima de Armazenamento: Este robô pode manipular e transportar várias caixas simultaneamente, armazenando aproximadamente 15 caixas A, 8 caixas B ou 5 caixas C.
- Fonte: Magazino SOTO[2].

### **Locus Robotics LocusBots**

- Capacidade de Recolher/Armazenar: Equipados com braços robóticos, os LocusBots podem alcançar alturas variáveis conforme necessário. São adequados para manipular as caixas A, B e C.
- Capacidade Máxima de Armazenamento: A capacidade de carga depende do modelo específico, mas geralmente suporta operações de armazém padrão, armazenando aproximadamente 10 caixas A, 6 caixas B ou 4 caixas C.
- Fonte: Locus Robotics[8].

Apesar das capacidades avançadas desses robôs, nenhum deles consegue-se adaptar 100% às necessidades específicas de manipulação de caixas SUC sem o desenvolvimento de mecanismos personalizados.

Adaptar esses robôs para corresponder completamente a essas necessidades pode ser muito dispendioso, pois requer modificações específicas e ajustes personalizados que vão além das capacidades padrão oferecidas pelos fabricantes.

Portanto, embora esses robôs ofereçam uma base sólida e flexível para automação, a adaptação total às necessidades específicas de manipulação das caixas SUC exigiria investimentos significativos de desenvolvimento e customização da solução.

### 3. Metodologia para Projeto do Sistema Robótico

Neste capítulo, é apresentada a metodologia utilizada para desenvolver o sistema robótico, abordando a análise de requisitos, o desenvolvimento do projeto global, a validação dos componentes e a montagem dos componentes mecânicos.

Inicialmente, é realizada uma análise detalhada dos requisitos necessários para o sistema, focando nas necessidades específicas da empresa para a recolha e distribuição de caixas tipo SUC.

O projeto global foi desenvolvido em três etapas principais:

1. **Carro de Transporte:** Projeto e construção de um carro de transporte adaptado para movimentar caixas dentro da zona fabril.
2. **Manipulador para Caixas SUC:** Criação de um manipulador específico para recolher e distribuir caixas SUC, adaptando-se às suas dimensões e pesos variados.
3. **Navegação do MIR100 Transportando o Carro:** Implementação de algoritmos de navegação utilizando a API do MIR100, com interfase programada em Python, para garantir a movimentação autónoma e precisa do robô transportando o carro.

Foram realizados testes de validação para garantir a funcionalidade e a precisão de cada componente do sistema, incluindo os sensores, atuadores e o sistema de controlo.

A montagem dos componentes mecânicos foi realizada de acordo com o projeto, integrando todas as partes do sistema robótico de forma coesa e funcional.

Os requisitos para o mecanismo de armazenamento focaram a capacidade de recolha de caixas SUC de diferentes tamanhos, levando à necessidade de um mecanismo adaptável e preciso para manipular as caixas.

A adaptação do MIR100 foi crucial para o projeto, utilizando o robô existente como base. Foi desenvolvido um sistema de elevação para manipulação das caixas, um sistema de navegação para o transporte do carro, e ajustado o manipulador para a recolha e distribuição das caixas SUC.

### **3.1. Análise de requisitos**

Nos próximos pontos, ire abordar a análise de requisitos e o desenvolvimento de todos os mecanismos, bem como a estrutura do carro necessários para fazer tudo funcionar de forma integrada.

#### **3.1.1. Carro de Transporte**

O carro de transporte deve ser capaz de armazenar caixas tipo SUC de tamanhos variados em quantidades suficientes para garantir o bom funcionamento do sistema. Deve ser robusto, permitindo a movimentação segura e organizada das caixas dentro da zona fabril. O design do carro deve facilitar a inserção e remoção das caixas pelo manipulador robótico, além de assegurar a estabilidade da carga durante o transporte.

#### **3.1.2. Dispositivo de Manipulação**

O dispositivo de manipulação (manipulador) deve ser capaz de recolher as caixas SUC do posto de armazenamento, organizá-las dentro do carro de transporte e depositá-las nos postos de trabalho. Para isso, deve reconhecer os diferentes tamanhos das caixas e otimizar o espaço de armazenamento dentro do carro. O manipulador deve possuir sensores avançados para detetar as dimensões das caixas e um mecanismo adaptável para manipular cargas de diversos tamanhos e pesos com precisão e segurança.

#### **3.1.3. Navegação e Transporte do Carro**

Para a navegação e transporte do carro, foi utilizado o robô MIR100, já disponível nas instalações da empresa, reduzindo os custos do projeto. O MIR100 deve acoplar o carro de forma autónoma e precisa, além de se mover pelo chão de fábrica de forma segura. A navegação autónoma do MIR100 deve incluir a capacidade de evitar obstáculos, seguir rotas otimizadas e ajustar-se dinamicamente às mudanças no ambiente fabril. O sistema de captura do carro deve garantir um acoplamento seguro e fácil desacoplamento, permitindo operações contínuas e minimizando o tempo de inatividade.

### **3.2. Desenvolvimento do Projeto Global**

O projeto global foi realizado em três etapas principais, focando-se no carro de transporte, no manipulador para caixas SUC, e na navegação do MIR100 que transporta o carro.

### 3.2.1. Carro de Transporte

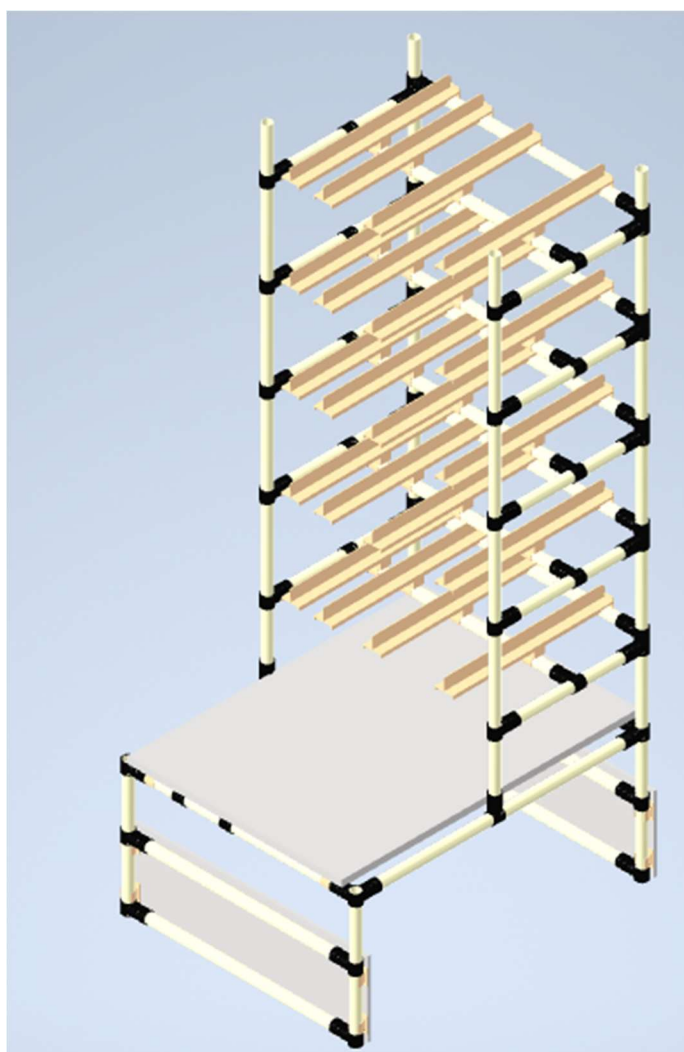
O carro de transporte foi desenvolvido a partir de uma estrutura **triforme** já amplamente utilizada na empresa, o que facilitou a sua montagem e implementação prática.

Essa estrutura é baseada na utilização de tubos metálica com pintura resistente, permitindo a construção de uma estrutura simples, resistente e de rápida montagem.

A união da estrutura é feita por peças metálicas que permitem a conexão dos tubos em diversas configurações, como em T, L, e ângulos variados.

Além disso, foram adicionadas quatro rodas para mobilidade e um sistema de acoplamento que permite ao robô MiR100 acoplar-se ao carro sem a necessidade de elevá-lo completamente, possibilitando que o robô movimente o carro até ao objetivo, i.e., um local no mapa do chão de fábrica conhecido pelo robô.

O carro de transporte é apresentado na imagem seguinte.



*Figura 18 -Desenho do Carro de Transporte*

### 3.2.2. Mecanismo para manipulação de caixas SUC

O Mecanismo para manipulação foi desenvolvido com três eixos de translação XYZ, similar a uma impressora 3D.

Para o eixo Z, é proposto um sistema com cabos de aço tracionados por um motor elétrico NEMA 32,[10] que possui a capacidade necessária para elevar toda a estrutura de movimentação.

O eixo Y, responsável pela recolha das caixas das prateleiras do carro e a sua colocação no posto de trabalho, movimento em profundidade. Utiliza um mecanismo baseado numa plataforma telescópica, capaz de se deslocar tanto para frente quanto para trás, aproximadamente 20 cm.

O eixo X, que proporciona movimento horizontal, utiliza um atuador linear que usa um motor NEMA 17[11] com polias e uma base móvel, garantindo precisão e estabilidade durante o movimento de translação, esquerda/direita e vice-vers.

Para assegurar a precisão e do Mecanismo de Manipulação, foi integrada uma câmara RealSense da Intel[12], que é utilizada para efetuar a leitura de tags nos postos de trabalho, localizando o ponto exato para colocar a caixa e corrigir eventuais erros de movimentação do MIR100[13].

Foram ainda instalados fins de curso em todos os três eixos para definir os zeros dos motores de forma precisa, evitando problemas de sincronismos dos três eixos.

### 3.2.3. Navegação do MIR100 com o Carro acoplado

A navegação autónoma do MIR100 foi um componente crucial do projeto.

A implementação da navegação foi realizada utilizando a API fornecida pela MIR, que permite a execução de missões previamente definidas na interface do próprio robô, facilitando assim o seu desenvolvimento.

Também foram criadas missões específicas para acoplar (dock) o robô ao carregador e ao carro de transporte, bem como desacoplar (undock) do mesmo em pontos pré-estabelecidos.

Adicionalmente, foi definida uma missão que permite, através de um painel de controlo no próprio carro, enviar o robô para um local específico utilizando coordenadas X, Y do mapa, e a rotação do MIR100. Este procedimento facilita correções e ajustes na navegação do robô, garantindo maior precisão no transporte de materiais.

Essas etapas foram fundamentais para desenvolver um sistema robótico integrado, adaptável às necessidades da empresa, garantindo a automação de processos de recolha e distribuição de caixas SUC.

### 3.3. Especificações do Robô MIR100

O robô MIR100, uma inovação da Mobile Industrial Robots, representa uma solução avançada em robótica móvel autónoma, destinada a otimizar o transporte de materiais em ambientes industriais e comerciais. Equipado com sensores de última geração e um software de navegação altamente sofisticado, este robô consegue desviar-se de obstáculos e pessoas, garantindo uma operação segura. Capaz de carregar até 100 kg, o MIR100 pode ser customizado com diversos módulos e acessórios, ampliando sua aplicabilidade em várias tarefas, incluindo o transporte de paletes e caixas, até equipamentos delicados.



*Figura 19-MIR 100*

A implementação do MIR100 nos sistemas de produção e logística resulta numa redução substancial dos custos operacionais e um incremento na eficiência, permitindo que os trabalhadores se dediquem a atividades mais complexas e de maior valor acrescentado. Com uma interface de utilizador extremamente intuitiva, o MIR100 pode ser operado facilmente via smartphone, tablet ou computador, simplificando-se a sua programação e monitorização sem a necessidade de formação avançada.

À medida que a procura por soluções de automação flexíveis e eficazes cresce, o MIR100 emerge como um componente crucial em diversos setores, revolucionando as operações e promovendo a transição para processos mais digitalizados e automatizados. Este robô não apenas melhora a produtividade, mas também redefine as práticas de trabalho convencionais, estabelecendo-se como uma ferramenta vital para o futuro da automação industrial.

O MIR100 é um robô móvel autónomo, especialmente projetado para o transporte de materiais em ambientes industriais, como fábricas, hospitais e armazéns, operando

de forma independente sem intervenção humana. Este robô destaca-se pela sua capacidade de navegar autonomamente, evitando obstáculos e pessoas, sendo equipado com uma interface web para a configuração de rotas e controlo.

**Especificações Técnicas:**

- Capacidade de Carga: Até 100 kg numa superfície de 600 x 800 mm.
- Dimensões: Comprimento de 850 mm, largura de 600 mm e altura de 350 mm.
- Autonomia e Desempenho: Opera até 10 horas ou percorre até 20 km numa única carga, com uma velocidade máxima de 1,5 m/s.
- Bateria: De tipo. Li-ion, com capacidade de 40 Ah e 24 V, possibilitando uma recarga rápida até 80% em apenas 3 horas.

**Segurança e Navegação:**

- Medidas de Segurança: Inclui paragem de emergência e sensores avançados para a deteção e evitação de obstáculos, garantindo operação segura em ambientes movimentados.
- Conformidade com Normas de Segurança: Cumpre as normas industriais, como Norma Portuguesa NP *EN 1525 2000* : segurança dos carros para movimentação de carga [13].
- Sistema de Navegação: Utiliza um sistema de planeamento global e local, ajustando-se automaticamente ao detetar obstáculos não mapeados.

**Controlo e Configuração:**

O controlo do MIR100 pode ser realizado através de uma interface web, permitindo que o robô seja configurado para executar rotas fixas, ser chamado ou realizar operações especiais. O robô MIR100 também suporta programação através do ROS. Esta funcionalidade permite a sua integração em rede com um computador, que controla a gestão de stocks.

### 3.4. Requisitos para o Mecanismo de Armazenamento no Carro

Os requisitos utilizados como base para o desenvolvimento do mecanismo de armazenamento focaram-se na capacidade de recolha de caixas do tipo SUC A, B e C, que possuem alturas, comprimentos e larguras distintos.

Este fato, levou à necessidade de cada prateleira acomodar um tipo específico de caixa e de empregar uma garra capaz de capturar caixas de qualquer tamanho.

Assim, foi desenvolvido um mecanismo telescópico com capacidade de movimento em duas direções: para frente e para trás. Este mecanismo foi projetado para ser controlado por um motor de passo, permitindo um posicionamento preciso da base.

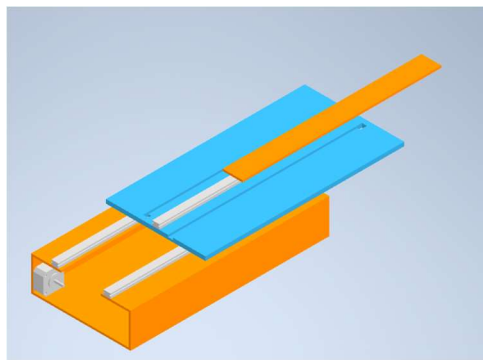


Figura 20-Desenho do mecanismo telescópico

O sistema funciona com uma lógica semelhante a um porta-paletes, posicionando-se em baixo da caixa e elevando-a utilizando um mecanismo de elevação com cabos de aço. Este é tracionado por um motor de passo de maior potência, com uma caixa de redução para auxiliar a movimentação e aumentar segurança, evitando quedas abruptas de toda a estrutura móvel.

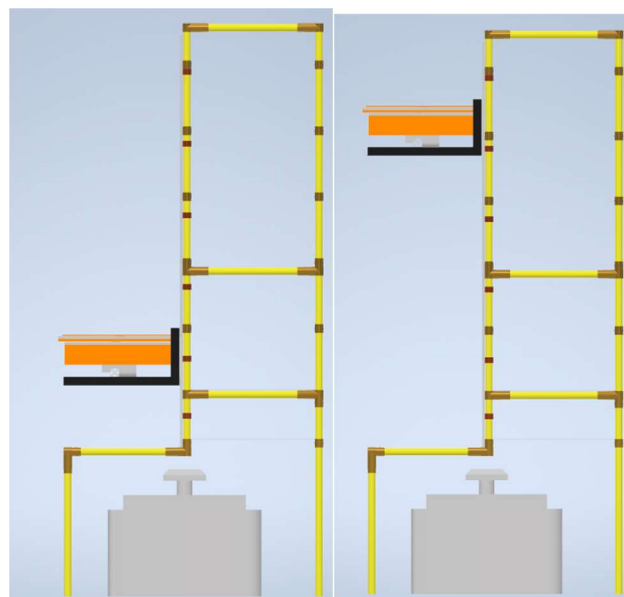


Figura 21-Desenho de mecanismo elevador

O mecanismo de movimento lateral emprega uma estrutura similar ao eixo X de uma impressora 3D, com movimento total ao longo da prateleira colocada no carro, que está acoplado ao robô MIR 100.

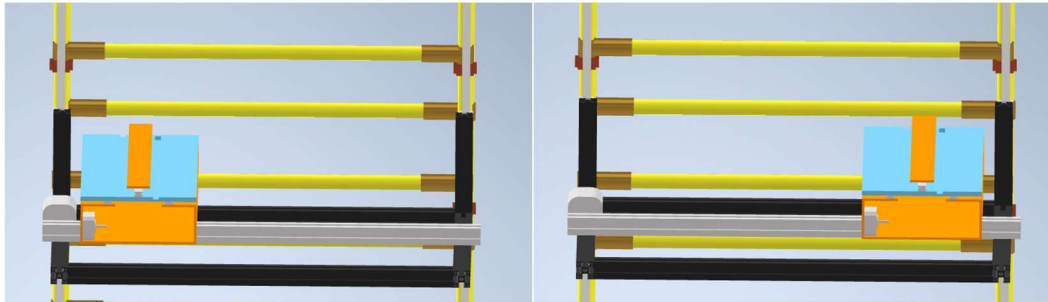


Figura 22-Desenho de mecanismo de movimento lateral

Com todos os mecanismos mencionados anteriormente, torna-se possível a criação de um sistema de armazenamento automático simplificado, com capacidade de recolha e distribuição para caixas tipo SUC nos três tamanhos referidos (tabela 1, figura 17).



Figura 23-caixas tipo suc

## 4. Desenvolvimento do Sistema

Neste capítulo, abordarei o desenvolvimento do mecanismo manipulador e a estrutura do carro onde será instalado o mecanismo.

### 4.1. Estrutura Mecânica

A estrutura do carro foi realizada utilizando o método triforme como é possível visualizar na imagem seguinte.

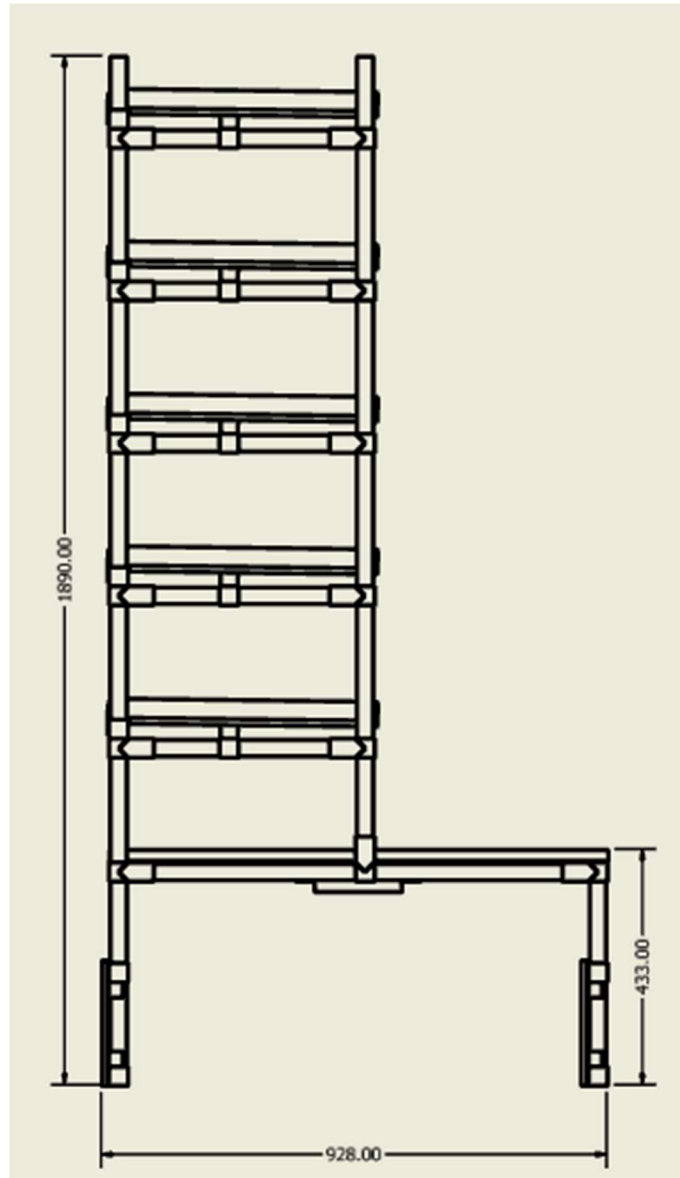
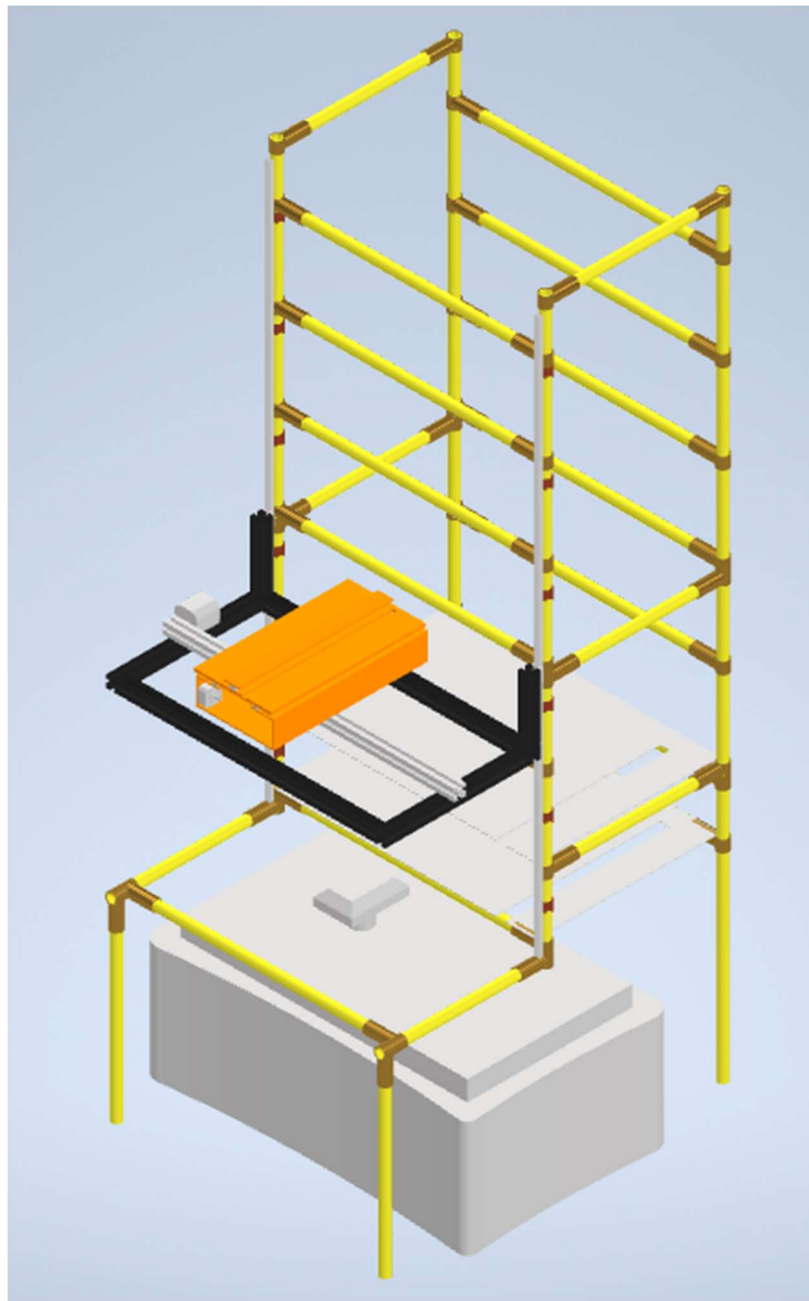


Figura 24-Desenho da estrutura do carrinho 2d[mm]

### 4.1.1. Design para Protótipo

O design do protótipo foi realizado no software Autodesk Inventor. Todo o mecanismo e a estrutura foram desenhados neste software para permitir uma análise visual preliminar antes da montagem. Este facto possibilitou uma análise detalhada de algumas partes do mecanismo, conforme representado na imagem seguinte.



*Figura 25-Desenho inicial da estrutura e mecanismo*

### 4.1.2 Seleção dos componentes

Todos os componentes foram selecionados com base em três parâmetros principais, sendo eles os seguintes:

- **Preço:** Foi considerada a relação custo-benefício para garantir uma escolha económica sem comprometer a qualidade.
- **Simplicidade de Montagem:** Os componentes foram escolhidos para facilitar a montagem, reduzindo a complexidade do processo.
- **Simplicidade de Utilização:** Foi tido em conta a facilidade de utilização, especialmente devido ao nível de conhecimento necessário para o controlo dos sistemas.

Com base nos parâmetros estabelecidos anteriormente, todos os mecanismos e seus componentes foram selecionados da seguinte forma:

#### 1. Estrutura Geral das Prateleiras:

Foi decidido utilizar o método das estruturas triformes, uma vez que a empresa possui relativa facilidade na aquisição dos materiais necessários.

#### 2. Mecanismo de Elevação:

Optou-se por um mecanismo baseado em cabos de aço, o que permite reduzir o seu tamanho. Inspirando-se nos princípios das janelas elétricas de carros, este sistema otimiza a segurança, evitando falhas devido à instabilidade do mecanismo e garantindo que, em caso de falta de energia, o mecanismo não caia de forma descontrolada. Este mecanismo é baseado num motor NEMA 32 com encoder, acoplado a uma caixa de redução para um controlo mais preciso da elevação.

#### 3. Mecanismo de Movimento Horizontal:

Foi escolhido um atuador linear controlado por um motor de passo NEMA 17, adequado para movimentos horizontais. Este atuador utiliza correias sincronizadas para converter a rotação do motor num movimento linear. Ideal para aplicações que não exigem elevação de carga, este sistema maximiza a velocidade e a capacidade de carga horizontal, sendo perfeito para o deslocamento das caixas nas prateleiras.

#### 4. Mecanismo para a Recolha das Caixas Tipo SUC:

Foi decidido desenvolver uma base telescópica elaborada sob medida para o projeto. Isso permite a recolha das caixas diretamente das prateleiras no carro acoplado ao robô MIR 100 e a colocação destas nas prateleiras dos postos de trabalho, sem a necessidade de um mecanismo complexo de rotação.

#### 5. Controlo dos Mecanismos:

Todo o controlo é realizado por um computador industrial de alto desempenho, com um Arduino como interface entre o primeiro e os atuadores e sensores. Este facto facilita a futura utilização de algoritmos complexos para localização das

prateleiras e das caixas, bem como para a gestão do inventário interno e externo do robô, otimizando o funcionamento geral. É utilizado um Arduino Mega, que, devido ao seu número considerável de portas de entrada e saída, facilita a integração de todos os sensores e atuadores analógicos e digitais necessários.

**6. Minimização de Erros de Posicionamento:**

Foram utilizadas duas câmaras RGBD para localizar tags nos postos de trabalho e nos locais nas prateleiras onde as caixas devem ser colocadas.

**7. Sensores de fim de curso:**

Sensores de fim de curso foram escolhidos para os movimentos horizontal e de recolha das caixas, enquanto o mecanismo vertical utiliza um sensor de distância.

**8. Nema 17:**

Este motor foi selecionado pelo seu tamanho reduzido e pela sua potência elevada, permitindo assim o acionamento do mecanismo da base telescópica bem como da guia linear, facilitando o controlo desses dois movimentos e aumentando a sua precisão o máximo possível, mantendo o baixo custo.

**9. Nema 32:**

Este motor foi escolhido para o mecanismo de elevação, em conjunto com uma taxa de redução para tornar possível a elevação de todos os outros mecanismos na lateral da margem, que se encontra no carro em cima do robô MIR100, mantendo assim a sua segurança, precisão e velocidade.

**10. Caixa de redução:**

Foi selecionada uma caixa de redução para ser colocada associada ao motor Nema 32, com o objetivo de maximizar a precisão do elevador e tornar mais seguro o mecanismo do elevador, evitando que, em caso de falta de bateria, o mecanismo caia de forma abrupta. A caixa de redução tem a capacidade de ser acionada no fuso, fazendo a engrenagem rodar e travar-se, funcionando assim como uma segurança mecânica para o mecanismo.

**11. Driver qrxq:**

Este driver foi selecionado para efetuar o controlo dos motores Nema 17 e 32, facilitando o comando pelo Arduino, bem como tornando mais fiáveis e precisos os motores de passo 17 e 32.

**12. Guias lineares:**

Foi selecionado um conjunto de guias lineares para facilitar o movimento vertical do elevador bem como o movimento horizontal da plataforma telescópica, simplificando a montagem da mesma.

### 4.1.3 Características dos atuadores selecionados

#### NEMA 17

- **Dimensões:** O NEMA 17 tem uma face de montagem de aproximadamente 43,18 mm x 43,18 mm. É um dos tamanhos mais comuns de motores de passo, conhecido por ser compacto e oferecer um bom equilíbrio entre binário e volume.
- **Aplicações:** Este motor é muito usado em impressoras 3D, pequenas máquinas CNC, robótica e outros dispositivos que requerem controlo preciso de movimento.
- **Características:** Os motores NEMA 17 podem variar em termos de binário, dependendo das especificações do enrolamento e da corrente elétrica. São ideais para aplicações onde o espaço é limitado, mas ainda é necessário um controlo preciso.



Figura 26 - NEMA 17

#### NEMA 32

- **Dimensões:** O NEMA 32 é consideravelmente maior, com uma face de montagem de aproximadamente 81,28 mm x 81,28 mm. Este tamanho é menos comum que o NEMA 17 e é utilizado em aplicações que exigem maior binário.
- **Aplicações:** Frequentemente encontrado em maquinaria industrial, equipamentos de automação de maior escala e aplicações onde altos níveis de binário são necessários para movimentar cargas pesadas.
- **Características:** O NEMA 32 oferece um torque significativamente maior que os modelos menores, como o NEMA 17, tornando-o adequado para tarefas mais robustas e exigentes.



Figura 27 - NEMA 32

## Caixas de Redução Sem Fim

Uma caixa redutora sem fim, também conhecida como redutor de rosca sem fim ou engrenagem sem fim, é um tipo de redutor de velocidade que utiliza uma configuração de parafuso sem fim e uma roda dentada (coroa) para modificar a velocidade e o binário de um motor. Este tipo de mecanismo é bastante seguro para aplicações que requerem reduções de velocidade em espaços limitados e onde é necessária uma saída perpendicular ao motor.

### Princípios de Funcionamento:

- O parafuso sem fim (que parece um parafuso ou uma espiral) encaixa-se nos dentes da roda dentada (coroa), transmitindo movimento com alta redução de velocidade.
- Quando o parafuso gira, move os dentes da coroa apenas um dente de cada vez. Isso proporciona uma redução de velocidade significativa num espaço compacto.

### Vantagens:

- Alto binário e baixa velocidade de saída: Ideal para aplicações que necessitam de muito binário e pouca velocidade.
- Compacto: Devido ao design vertical do parafuso sem fim e da coroa, este sistema é mais compacto do que outros tipos de redutores.
- Auto travamento: Em alguns designs, o parafuso sem fim pode impedir que a coroa gire quando o motor está desligado, funcionando como um freio ou bloqueio de segurança.

### Desvantagens:

- Eficiência: A eficiência deste tipo de redutor pode ser menor devido ao atrito constante entre o parafuso sem fim e a coroa.
- Desgaste: O contato contínuo entre o parafuso e a coroa pode resultar em desgaste mais rápido dos componentes, especialmente se não estiverem adequadamente lubrificados.

Os redutores sem fim são amplamente utilizados em diversas aplicações industriais e dispositivos, como elevadores, tapetes rolantes, portões automáticos, e em sistemas de automação onde são necessárias reduções de velocidade num espaço reduzido.



Figura 28-Caixas de Redução Sem Fim

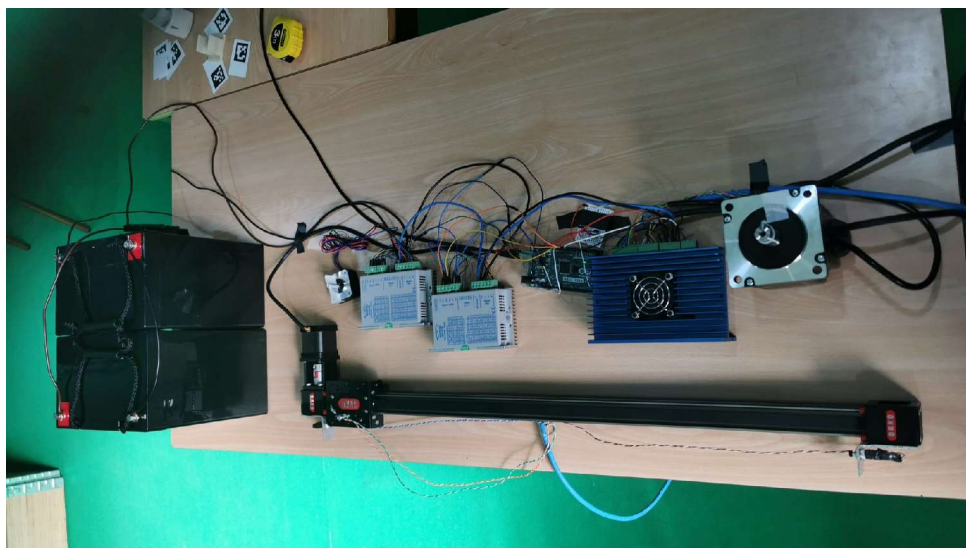
## 4.2. Montagem

A montagem da estrutura do carro foi realizada na empresa.



*Figura 29-Estrutura com MIR100 para testes*

A montagem de todo o mecanismo foi realizada no Laboratório de Robótica e Equipamentos Inteligentes da Escola Superior de Tecnologia do Instituto Politécnico de Castelo Branco com o fim de testar todos os componentes.



*Figura 30-Montagem dos atuadores do mecanismo*

### 4.3.3 Cinemática do mecanismo de 3 graus de liberdade desenvolvido

Para a análise cinemática do sistema robótico, foi aplicado o algoritmo de Denavit-Hartenberg[14] para determinar as matrizes de transformação entre as coordenadas do espaço 3D das prateleiras e as coordenadas de cada eixo do sistema robótico. Este método exige o estabelecimento dos referenciais seguindo uma sequência de passos específicos.

- **Referencial de base:**
  - Um referencial fixo ( $i=0$ ), onde o eixo  $Z_0$  fica alinhado com o eixo de movimento da junta 1.
  - Este é um sistema referencial direto.
- **Referencial  $0 < i < n$ :**
  1. Colocar o eixo  $Z_i$  sobre o eixo de movimento da junta  $i+1$ .
  2. Colocar a origem  $O_i$  na intersecção de  $Z_i$  com  $Z_{i-1}$ . Caso não se encontrem, posicionar a origem  $O_i$  na intersecção da normal comum a esses dois eixos com  $Z_i$ .
  3. Posicionar o eixo  $X_i$  perpendicularmente a  $Z_i$  e  $Z_{i-1}$ , de forma a interceptá-los.
  4. Colocar o eixo  $Y_i$  para completar um sistema referencial direto.
- **Referencial  $i = n$ :**
  1. Colocar o eixo  $Z_n$  na direção de aproximação da mão do robô aos objetos a manipular, apontando para fora do robô.
  2. Posicionar o eixo  $X_n$  perpendicular ao movimento de abertura e fechamento da mão.
  3. Colocar o eixo  $Y_n$  para completar um sistema referencial direto.

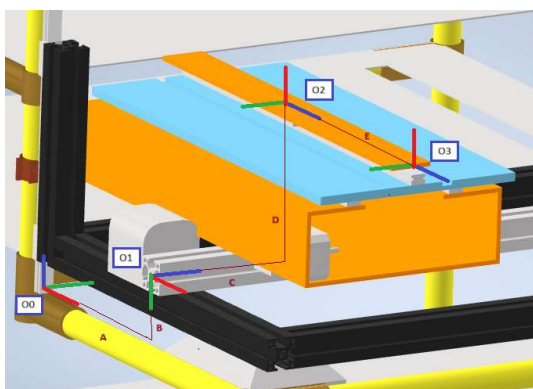


Figura 31-Referenciais da cinemática

| O0 | X0 | Y0 | Z0 | D[m]              | d[m]              |
|----|----|----|----|-------------------|-------------------|
| O1 | X1 | Y1 | Z1 | $A \approx 0,237$ | $C \approx 0,113$ |
| O2 | X2 | Y2 | Z2 | $B \approx 0,060$ | $E \approx 0,2$   |
| O3 | X3 | Y3 | Z3 | $D \approx 0,188$ |                   |

Tabela 2- Legenda dos referenciais da Cinemática

Seguindo o algoritmo de Denavit-Hartenberg,

- **Objetivo:** Determinar os parâmetros de cada ligação com base no algoritmo de Denavit-Hartenberg para calcular as matrizes de transformação correspondentes.
  - **$\theta_i$  (Theta):** Ângulo entre ligações, representando a rotação de  $Z_{i-1}$  em torno de  $Z_{i-1}$  para ficar coincidente/paralelo a  $X_i$ .
  - **$d_i$  (Distância):** Distância entre  $O_{i-1}$  e a intersecção de  $Z_{i-1}$  com  $X_i$ , representando a distância entre o plano  $O_{i-1}; X_{i-1}; Y_{i-1}$  e o eixo  $X_i$ .
  - **$a_i$  (Comprimento da ligação):** Distância entre a intersecção de  $Z_{i-1}$  com  $X_i$  e  $O_i$ , representando a distância do eixo  $Z_{i-1}$  ao plano  $O_i; Y_i; Z_i$ .
  - **$\alpha_i$  (Torção da ligação):** ângulo de rotação de  $Z_{i-1}Z_{i-1}$  em torno de  $X_i$  para ficar coincidente/paralelo a  $Z_i$ .

foi construída a tabela de parâmetros

|   | $\theta_i$       | $\alpha_i$       | $a_i$ | $d_i$         |
|---|------------------|------------------|-------|---------------|
| 1 | 0                | $-\frac{\pi}{2}$ | 0.237 | $d_1 + 0.060$ |
| 2 | $-\frac{\pi}{2}$ | $-\frac{\pi}{2}$ | 0.188 | $d_2 + 0.113$ |
| 3 | 0                | 0                | 0     | $d_3 + 0.20$  |

Tabela 3-Tabela de parâmetros de cinemática

Após construir a tabela de parâmetros a partir dos referenciais, foi aplicada a fórmula geral da matriz de Denavit-Hartenberg para calcular a relação entre os referenciais (0 e 1), (1 e 2), e (2 e 3).

$${}^{i-1}A_i = \begin{bmatrix} \cos(\theta_i) & -\sin(\theta_i) \cdot \cos(\alpha_i) & \sin(\theta_i) \cdot \sin(\alpha_i) & a_i \cdot \cos(\theta_i) \\ \sin(\theta_i) & \cos(\theta_i) \cdot \cos(\alpha_i) & -\cos(\theta_i) \cdot \sin(\alpha_i) & a_i \cdot \sin(\theta_i) \\ 0 & \sin(\alpha_i) & \cos(\alpha_i) & d_i \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Figura 31-Formula geral de Denavit-Hartenberg

Após a aplicação da matriz anterior foram obtidos os seguintes resultados para as matrizes de transformação entre os diversos eixos:

A01

$$\begin{bmatrix} 1 & 0 & 0 & 0.237 \\ 0 & 0 & 0 & 0 \\ 0 & -1 & 0 & d1 + 0.06 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

A12

$$\begin{bmatrix} 0 & 0 & 1 & 0 \\ -1 & 0 & 0 & -0.188 \\ 0 & -1 & 0 & d2 + 0.113 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

A23

$$\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & d3 + 0.2 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Ao multiplicar as três matrizes, obtém-se a matriz de relação entre os referenciais 0 e 3:

A03

$$\begin{bmatrix} 0 & 0 & 1 & d3 + 0.437 \\ 0 & -1 & 0 & d2 + 0.113 \\ 1 & 0 & 0 & d1 + 0.248 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Após obter esta matriz foram obtidas as equações da cinemática direta:

$$\begin{bmatrix} 0 & 0 & 1 & d3 + 0.437 \\ 0 & -1 & 0 & d2 + 0.113 \\ 1 & 0 & 0 & d1 + 0.248 \\ 0 & 0 & 0 & 1 \end{bmatrix} = \begin{cases} x = d1 + 0.248 \\ y = d2 + 0.113 \\ z = d3 + 0.437 \end{cases}$$

E de seguida as equações da cinemática inversa:

$$\begin{cases} x = d1 + 0.248 \\ y = d2 + 0.113 \\ z = d3 + 0.437 \end{cases} \Leftrightarrow \begin{cases} d1 = x - 0.248 \\ d2 = y - 0.113 \\ d3 = z - 0.437 \end{cases}$$

Posteriormente foram aplicadas as equações da cinemática inversa num programa em python como representado abaixo:

```
def cinematica_inversa_do_elemento_terminal( x, y, z):
```

```
    pm1 = 23 #passos/metro do motor 1
```

```
    pm2 = 23 # passos/metro do motor 2
```

```
    pm3 = 23 # passos/metro do motor 3
```

```
    np_1 = (x - 0.437) * pm1
```

```
    np_2 = (y - 0.113) * pm2
```

```
    np_3 = (z - 0.248) * pm3
```

```
    return [round(np_1), round(np_2), round(np_3)]
```

Este programa é meramente demonstrativo da aplicação da cinemática inversa para o controlo dos motores. Foi utilizado como um programa de testes nas etapas iniciais para verificar o funcionamento conjunto dos motores e validar a viabilidade de sua utilização.

## 4.4. O robô movel MIR100

### Definição e Instalação da estação de Carregamento:

Foi definida a localização da estação de carregamento autónomo do MIR100 utilizando a interface web do MIR100. A estação foi instalada e configurados os parâmetros necessários para garantir o carregamento.

### Implementação do Sistema de Captura do Carrinho

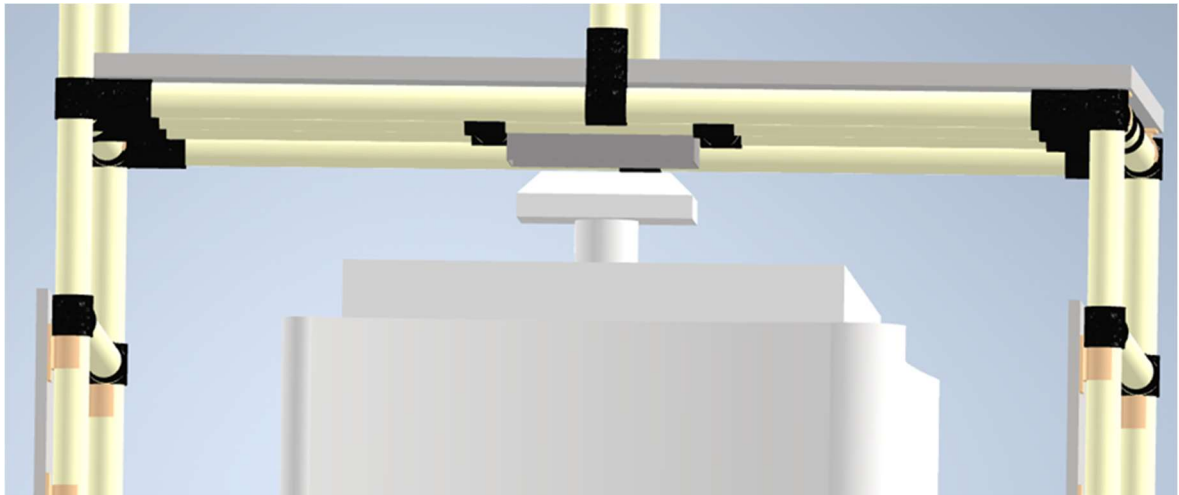


Figura 32-Sistema de Captura do Carrinho ao MIR100

#### 1. Utilização da Tecnologia Existente:

Foi utilizada a tecnologia existente no próprio robô MIR100 para implementar o sistema de captura do carrinho.

#### 2. Ajustes e Calibrações:

Foram realizados ajustes e calibrações para assegurar que o sistema de captura funcionasse de forma precisa e confiável.

#### 3. Definição do Protocolo de Captura:

A captura do carrinho foi definida utilizando o protocolo de docking do software do MIR. Definição das barras laterais do carro, especificando o comprimento e a distância entre elas.

#### 4. Localização e Captura do Carro:

O MIR100, com base nas distâncias definidas antes, consegue localizar o carro mesmo se ele estiver deslocado do local pré-definido, desde que esteja no mapa.

Este processo garante que o sistema de captura do carro funcione, mesmo em condições de posicionamento não ideal, proporcionando flexibilidade e precisão nas operações fabris.

## **Desenvolvimento e Realização de Missões:**

Foram criadas diversas missões diretamente no software do MIR100, ajustando os parâmetros conforme necessário. Para tal foi utilizada a API do MIR100 para permitir que as missões fossem acionadas por um programa externo.

### **Integração com Programa em Python:**

Foi desenvolvido um programa em Python capaz de enviar comandos ao servidor interno do MIR100, utilizando a API do robô. Foi simplificada a criação de protocolos e missões, permitindo uma integração mais flexível com o sistema de controlo.

### **Envio de Coordenadas e Rotação:**

Foi desenvolvido um protocolo específico para enviar as coordenadas X, Y e a rotação do robô, possibilitando o movimento do carro até pontos especificados no mapa. Foi assegurado que o robô pudesse interpretar e executar esses comandos.

### **Criação de Funções em Python para Facilitar envio de Missões:**

Foram desenvolvidas diversas funções em Python para facilitar a execução das missões, como a função "dock". A função "dock" direciona o robô até o ponto de carga, conduzindo-o através de uma aproximação precisa, seguida pela ancoragem autónoma e início do carregamento.

### **Implementação de um programa para deteção de tags**

Foi implementado um programa[15] baseado na biblioteca ViSP(Visual Servoing Platform)[16], integrado com o ROS utilizando a biblioteca ROS Bridge. O objetivo foi realizar a leitura da câmara e publicar outro a lista de tags que aparecem na cena, obtendo-se as coordenadas 3D relativas à câmara de cada tag.



*Figura 33-Câmara Intel RealSense*

Foi utilizada a biblioteca ViSP para deteção e análise de tags. Para tal foi necessário instalar os drivers da câmara Intel RealSense e realizar testes iniciais para verificar a funcionalidade da câmara. Esses drivers permitiram iniciar um conjunto de nós no ROS para interação com a câmara.

Foi implementado um programa com a biblioteca ViSP para processar imagens capturadas pela câmara RealSense que utiliza a biblioteca ViSP e a biblioteca ROS Bridge para a comunicação.



*Figura 34-Logotipo ViSP*

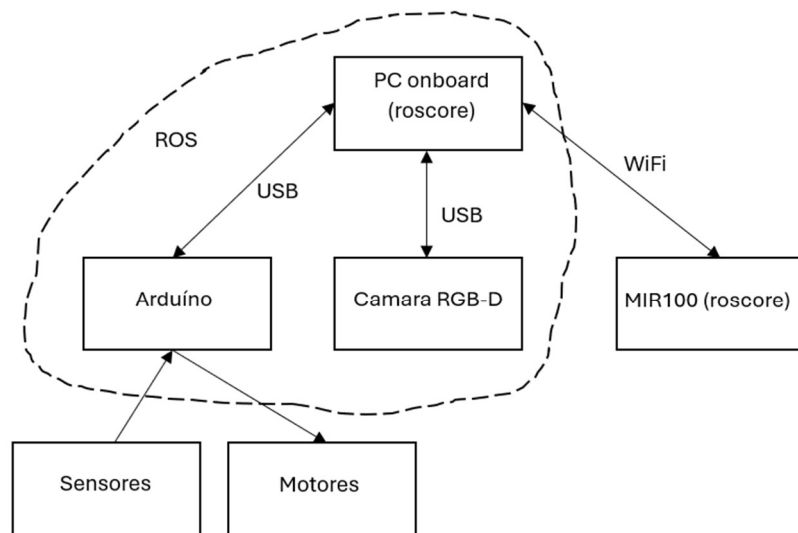
O programa implementado, publica as coordenadas das tags detetadas num tópico ROS, estruturando os dados de forma a garantir compatibilidade na comunicação com os diversos componentes do sistema. Este programa será utilizado posteriormente para corrigir o erro de movimentação do MIR100, bem como do mecanismo de armazenamento do carro.

## 5. Programação e Validação do sistema robótico

No presente capítulo, é apresentada a arquitetura de software de todos o sistema baseado em ROS, o funcionamento dos programas desenvolvidos e a integração total de todos os componentes. É apresentado em detalhe cada ponto, validando o funcionamento de cada um dos elementos do sistema.

### 5.1. Arquitetura do software desenvolvido

Os controladores foram seleccionados para poderem comunicar entre si utilizando o ROS como representado na imagem abaixo.



*Figura 35-Arquitectura do software desenvolvido*

O PC onboard (roscore) é o núcleo central do sistema do carro, onde o ROS está em execução.

O PC onboard comunica com a câmara RGB-D via USB 3.0/3.1 e ao Arduíno via USB utilizando o protocolo RS232 através do pacote de ROS, ROSserial.

O PC onboard comunica com o MIR100 via Wi-Fi utilizando REST API, através de requests em http[12].

O Arduíno controla diretamente os sensores de fim de curso e os motores, recebendo e enviando dados para o PC onboard.

A REST API é utilizada para interligar os ambientes ROS do PC onboard e do MIR100, evitando conflitos e simplificando o uso de um único carro e MIR100. Posteriormente, o sistema pode ser ajustado para ser controlado diretamente pelo ROS, permitindo a utilização de múltiplos processadores (roscores).

## 5.2. Desenvolvimento de programas

### Linguagens de Programação Utilizadas:

**C++:** Utilizada exclusivamente para a análise de imagens e localização de tags, otimizando a identificação dos objetos a serem manipulados pelo mecanismo. Esta linguagem foi escolhida pelo seu bom desempenho em operações de processamento de imagem em tempo real.

**Python:** Utilizada para a integração entre o Arduino e o computador de bordo do carro, envio de missões para o MIR100 e desenvolvimento de sistemas mais simples que não exigem programação de baixo nível como o C++.

A linguagem Python foi selecionada devido à sua facilidade de uso e extensa biblioteca de suporte para desenvolvimento rápido.

### Sistema Operativo:

O código foi executado num sistema operativo Ubuntu, utilizando o ROS.

Ubuntu foi escolhido pela sua estabilidade e ampla compatibilidade com ROS, que é essencial para o desenvolvimento de sistemas robóticos complexos.

### Preparação para melhorias futuras:

Todos os sistemas de controlo foram projetados com uma margem de flexibilidade para futuras melhorias e implementação de algoritmos. Essa abordagem garante que o sistema possa evoluir e adaptar-se a novos requisitos e avanços tecnológicos, mantendo-se funcional.

### Programa de Cinemática:

Foi desenvolvido um programa específico para a cinemática dos movimentos do mecanismo de manipulação, levando em consideração os cálculos da cinemática direta e inversa. Esses cálculos são fundamentais para garantir a precisão na manipulação de objetos pelo robô.

### Controlo de Motores de Passo:

Foi desenvolvido um programa para controlo dos motores de passo utilizando Arduino, que subscreve a um nó em robôs para efetuar os movimentos e recebe as coordenadas XYZ, movimentando os motores nesses três eixos conforme os cálculos da cinemática e o número de passos fornecido para atingir o objetivo.

Tem ainda a capacidade de publicar no ecossistema ROS quando se define o número de passos máximo e sempre que um objetivo é atingido.

Este controlo preciso é essencial para garantir que o manipulador executa as tarefas de movimentação de forma correta.

### **Integração com MIR100:**

Foi desenvolvido um programa para interligar o computador onboard com o MIR100 utilizando Python, que funciona via REST API e é capaz de adicionar missões ao MIR100 com base no nome da missão ou nas coordenadas do mapa, definidas como objetivo. Essa integração permite que o manipulador de caixas informe qual é o ponto necessário para recolher ou depositar as caixas de forma simplificada.

### **Localização de Tags:**

Foi implementado um programa em C++ que, com as imagens da câmara, é capaz de localizar tags e devolver as suas coordenadas no referencial da câmara.

Esta funcionalidade é crucial para a navegação e manipulação precisa de objetos no ambiente do robô, bem como na aproximação do mecanismo manipulador ao posto de trabalho.

### **Teste Global de Mecanismos:**

Foi desenvolvido um programa que unifica todos os anteriores para realizar um teste global de todos os mecanismos de forma simultânea.

Este programa é utilizado para validar a integração do sistema como um todo, assegurando que todos os componentes cumprem as suas funções.

### 5.3. Validação e testes do MIR100

Os testes de precisão do MIR100 são necessários para verificar a capacidade de se deslocar até um ponto do mapa estabelecido no seu próprio programa. Foram realizadas medições a partir da lateral do MIR100 até caixas colocadas numa posição fixa, seguindo o procedimento descrito abaixo:

1. Enviar o MIR100 para um ponto externo, próximo ao centro das caixas.
2. Após isso, o MIR100 retorna ao ponto inicial.

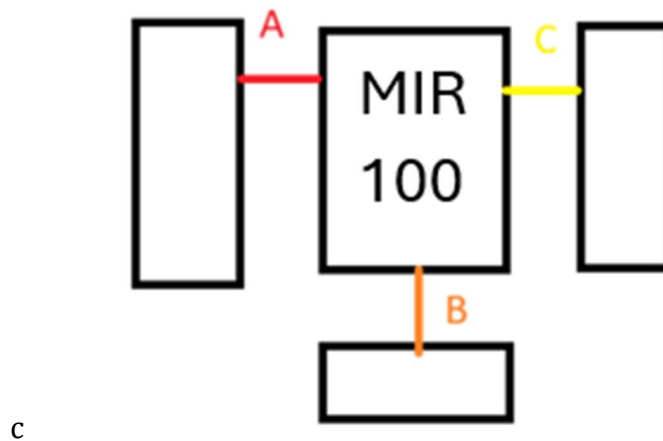


Figura 36-Esquema de calibração do MIR 100

**Medição A:** Distância lateral esquerda

**Medição B:** Distância frontal

**Medição C:** Distância lateral direita

Nesse ponto, procedeu-se à medição das distâncias com o objetivo de verificar se o robô mantinha a distância das paredes laterais e frontal em relação às caixas.

Este procedimento foi repetido 40 vezes, resultando nos seguintes valores representativos:

Os valores coletados para cada medição são os seguintes:

*Tabela 4-Dados de Calibração de MIR 100*

| A[cm] | B[cm] | C[cm] |
|-------|-------|-------|
| 24    | 14,5  | 20    |
| 26,5  | 15    | 17,5  |
| 23,5  | 12,5  | 20,5  |
| 26    | 13,5  | 18    |
| 25    | 16    | 19    |
| 23    | 16    | 21    |
| 24,5  | 16,5  | 19,5  |
| 24,5  | 15    | 19,5  |
| 23,5  | 13,5  | 20,5  |
| 22    | 12,5  | 22    |
| 27    | 14    | 17    |
| 25    | 18,5  | 19    |
| 26,5  | 16,5  | 17,5  |
| 24,5  | 15    | 19,5  |
| 30    | 12,5  | 14    |
| 23,5  | 13    | 20,5  |
| 29,5  | 15    | 14,5  |
| 28    | 15,5  | 16    |
| 29    | 15    | 15    |
| 26,5  | 15,5  | 17,5  |

| A[cm] | B[cm] | C[cm] |
|-------|-------|-------|
| 24,5  | 15    | 19,5  |
| 24,5  | 15    | 19,5  |
| 25    | 14    | 19    |
| 26    | 17    | 18    |
| 25,5  | 13,5  | 18,5  |
| 26,5  | 13    | 17,5  |
| 26    | 16,5  | 18    |
| 27    | 18    | 17    |
| 24    | 14,5  | 20    |
| 26    | 15    | 18    |
| 22    | 16    | 22    |
| 23,5  | 17,5  | 20,5  |
| 23    | 16,5  | 21    |
| 24    | 14,5  | 20    |
| 23    | 17,5  | 21    |
| 27    | 17    | 17    |
| 26    | 16,5  | 18    |
| 27    | 15    | 17    |
| 24,5  | 15,5  | 19,5  |
| 28    | 17,5  | 16    |

### Estatísticas dos Dados

- Medição A:
  - Média: 25.38 cm
  - Desvio padrão (erro):  $\pm 1.94$  cm
  - Mínimo: 22.00 cm
  - Máximo: 30.00 cm
- Medição B:
  - Média: 15.26 cm
  - Desvio padrão (erro):  $\pm 1.57$  cm
  - Mínimo: 12.50 cm
  - Máximo: 18.50 cm
- Medição C:
  - Média: 18.63 cm
  - Desvio padrão (erro):  $\pm 1.94$  cm
  - Mínimo: 14.00 cm
  - Máximo: 22.00 cm

Assim, os erros em cada medição são:

- O erro na Medição A é aproximadamente  $\pm 1.94$  cm.
- O erro na Medição B é aproximadamente  $\pm 1.57$  cm.
- O erro na Medição C é aproximadamente  $\pm 1.94$  cm.

As medições realizadas demonstram uma precisão promissora na movimentação do robô, com o erro na Medição A em  $\pm 1.94$  cm, na Medição B em  $\pm 1.57$  cm, e na Medição C em  $\pm 1.94$  cm. Esses resultados indicam que o sistema de navegação do MIR100 opera com um nível de precisão que está dentro de limites aceitáveis. Com ajustes adicionais, é possível melhorar ainda mais a consistência do sistema, para a otimização das operações do mecanismo de manipulação

## 5.4. Validação e testes do mecanismo de aproximação a tag

Para testar e validar o mecanismo de aproximação à tag, foi desenvolvido um programa em Python. Este programa ajusta iterativamente o posicionamento do mecanismo nos eixos Y e Z com base na localização da tag em relação à câmara, até que o erro de alinhamento seja até 1 mm.

### **Funcionamento do Programa:**

**Inicialização:** O programa começa por capturar a posição inicial da tag em relação à câmara.

**Cálculo de Erro:** A cada iteração, calcula-se o erro de alinhamento atual entre a posição da tag e a posição alvo da câmara.

**Ajuste de Posição:** Com base no erro calculado, o programa ajusta a posição do mecanismo nos eixos Y e Z para reduzir a discrepância.

**Iteração:** Este processo é repetido iterativamente, ajustando continuamente a posição até que o erro seja menor que o valor predefinido.

**Convergência:** O programa termina quando o erro de alinhamento é suficientemente pequeno, garantindo que a câmara esteja corretamente alinhada com a tag.

### **Resultados Obtidos:**

Após a execução do programa, foi constatado que o erro máximo alcançado pelo sistema foi de aproximadamente 1 mm. Este alto grau de precisão valida o mecanismo de aproximação para tarefas críticas.

### **Aplicações Práticas:**

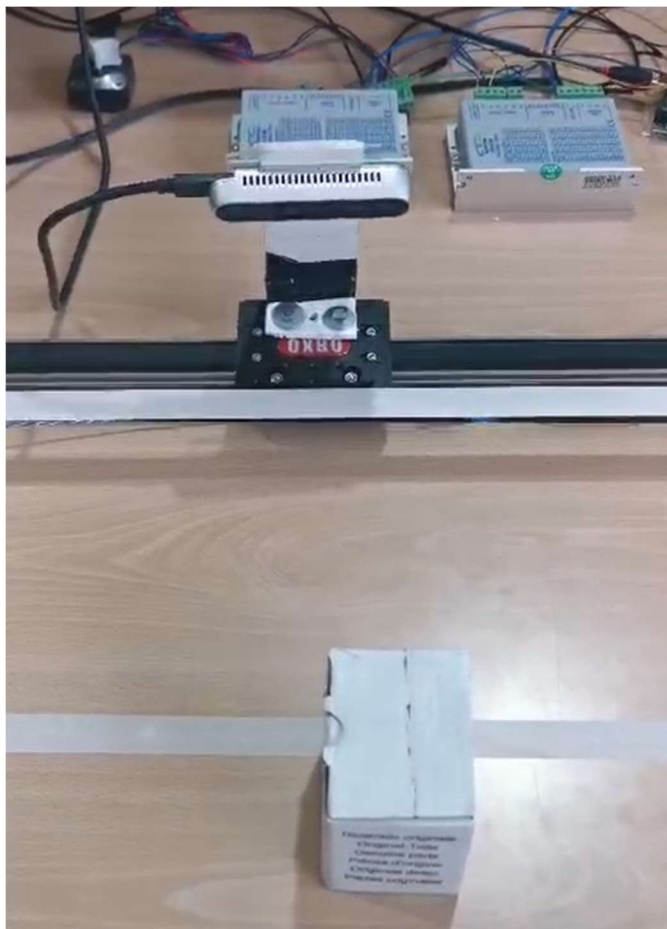
**Recolha de Caixas:** A precisão do alinhamento permite ao mecanismo aproximar-se do posto de trabalho de maneira confiável, facilitando a recolha de caixas.

**Reposição de Caixas:** A mesma precisão é aplicada na reposição de caixas, assegurando que o mecanismo possa operar com alta precisão, minimizando erros do sistema.

### **Conclusão:**

A implementação deste programa de ajuste iterativo de posição provou ser confiável para a aproximação precisa ao posto de trabalho.

O erro mínimo de aproximadamente 1 mm confirma a adequação do mecanismo para operações que exigem alta precisão, como a manipulação de caixas num ambiente automatizado.



*Figura 37-Mecanismo de aproximação em y*

## 5.5. Testes global de Controlo e Comunicação

Para efetuar o teste global de controlo e comunicação, foram desenvolvidos vários programas em Python que foram interligados utilizando o ROS. Esses programas baseiam-se na publicação e subscrição de mensagens para compartilhar informações entre si. O resultado é o diagrama seguinte (Figura 37, Figura 38, Figura 39), que abordarei mais pormenorizadamente mais a frente, explicando as suas funções.

Os programas em Python desempenham diversas funções, como controlo de movimento, processamento de dados de sensores e comunicação entre diferentes componentes do sistema. Utilizando o ROS, é possível criar uma arquitetura modular e escalável, onde cada nó ROS executa uma tarefa específica e comunica com outros nós através de tópicos. Esta abordagem facilita a eliminação de erros e o desenvolvimento incremental, permitindo adicionar ou modificar funcionalidades sem afetar o sistema global.

O diagrama ilustra como os diferentes nós ROS interagem entre si, incluindo a comunicação entre o PC onboard, a câmara RGB-D, o Arduino e o MIR100. Cada componente é responsável por uma parte crítica do sistema:

- O PC onboard atua como o controlador principal, executando os algoritmos de controlo e coordenação.
- A câmara RGB-D fornece dados visuais e de profundidade para navegação e deteção de obstáculos.
- O Arduino controla os sensores de fim de curso e os motores, garantindo a precisão do movimento.
- O MIR100, conectado via REST API, é responsável pela mobilidade do carro no ambiente de trabalho.

Em seguida são detalhados cada um desses componentes e as suas funções específicas no sistema, bem como os desafios enfrentados e as soluções implementadas durante o desenvolvimento.

## Diagrama do ROS

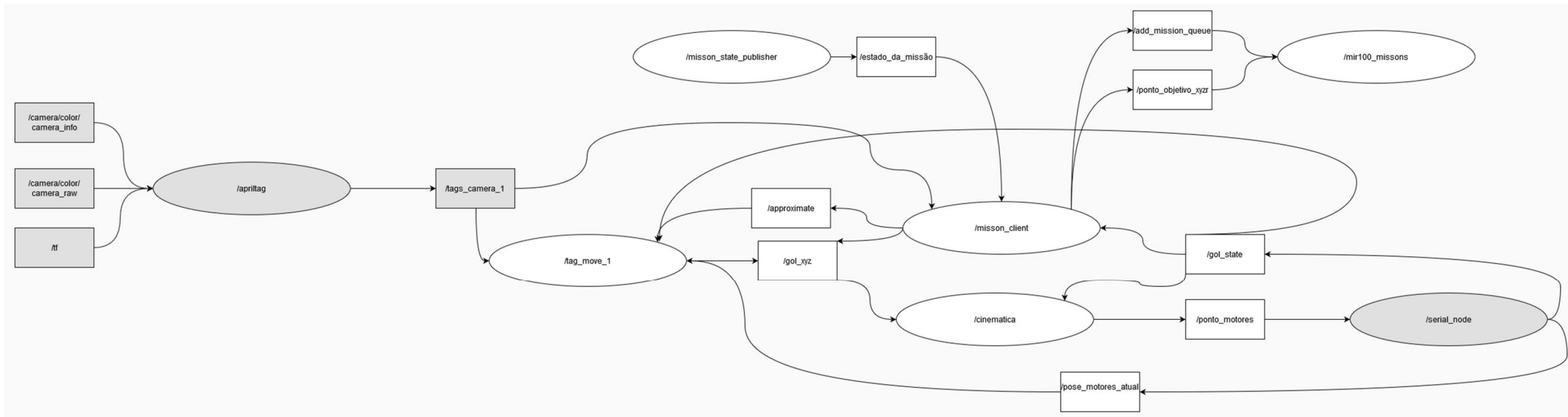


Figura 38-diagrama ROS

## **Nó Principal do Programa: /mission\_client**

A parte principal do programa é o nó /mission\_client, onde estão definidos os pontos objetivos para os quais o mecanismo deve ir recolher a caixa.

É neste nó que são tomadas as principais decisões do processo de movimentação do MIR100 e do mecanismo de manipulação de caixas.

O nó /mission\_client publica e subscreve as seguintes mensagens, conforme representado no diagrama.

### **Publicações:**

/approximate

/gol\_xyz

/add\_mission\_queue

/ponto\_objetivo\_xyzr

### **Subscrições:**

/estado\_da\_missao

/gol\_state

/tags\_camera\_1

### Objetivos das Mensagens:

`/estado_da_missao`: Informa o estado atual da última missão na lista de missões em espera do MIR100, permitindo ao sistema monitorizar o progresso e tomar decisões baseadas no status da missão.

`/gol_state`: Indica o ponto máximo de movimentação dos motores do mecanismo e notifica quando o mecanismo atinge o objetivo solicitado, permitindo ajustes finos e precisão no movimento.

`/tags_camera_1`: Fornece as coordenadas 3D das tags detetadas pela câmara, essenciais para a navegação e alinhamento preciso do mecanismo.

`/approximate`: Sinaliza ao mecanismo para iniciar a aproximação ao posto de trabalho, utilizando a informação da tag específica para guiar o movimento.

`/gol_xyz`: Envia as coordenadas de destino para o mecanismo, definindo exatamente onde ele se deve posicionar, em cada prateleira.

`/add_mission_queue`: Adiciona missões à fila do MIR100, utilizando palavras-chave para simplificar o processo de acoplamento e desacoplamento.

`/ponto_objetivo_xyzr`: Define o ponto de destino específico para o MIR100, utilizando coordenadas XYZ e rotação R para posicionamento preciso.

### Diagrama Representativo

O diagrama a seguir ilustra a interação entre os diferentes tópicos e como o nó `/mission_client` utiliza essas mensagens para coordenar o movimento do MIR100 e do mecanismo de manipulação de caixas.

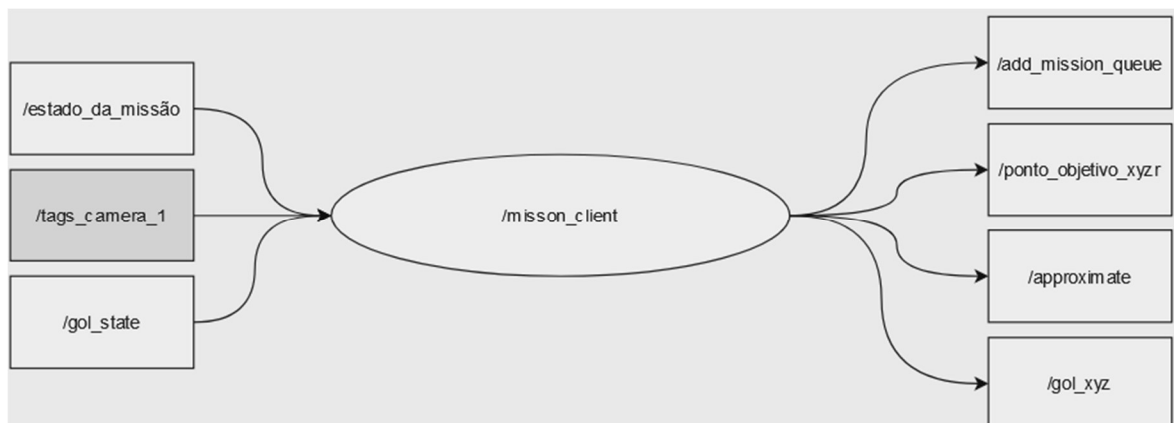


Figura 39-diagrama ROS nó `/mission_client`

## Nó Responsável pela Aproximação às Tags: /tag\_move\_1

A principal função do nó /tag\_move\_1 é efetuar a aproximação do mecanismo de manipulação das caixas ao posto de trabalho nos eixos Y e Z.

### **Publicações:**

/gol\_xyz

### **Subscrições:**

/goal\_state

/tags\_camera\_1

/pose\_motores\_atual

/approximate

### **Objetivos das Mensagens:**

/goal\_state: Indica o ponto máximo de movimentação dos motores do mecanismo e notifica quando o mecanismo atinge o objetivo solicitado, permitindo ajustes finos e precisão no movimento.

/tags\_camera\_1: Fornece as coordenadas das tags detetadas pela câmara, essenciais para a navegação e alinhamento preciso do mecanismo.

/pose\_motores\_atual: Fornece a posição atual dos motores nos eixos XYZ do mecanismo, permitindo monitorizar e ajustar a posição em tempo real.

/approximate: Sinaliza ao mecanismo para iniciar a aproximação ao posto de trabalho, utilizando a informação da tag específica para guiar o movimento.

/gol\_xyz: Envia as coordenadas de destino para o mecanismo, definindo exatamente onde ele se deve posicionar.

## Diagrama Representativo

O diagrama a seguir ilustra a interação entre os diferentes tópicos e como o nó `/tag_move_1` utiliza essas mensagens para coordenar o movimento do mecanismo de manipulação de caixas para efetuar a aproximação ao posto de trabalho. `/apriltag`

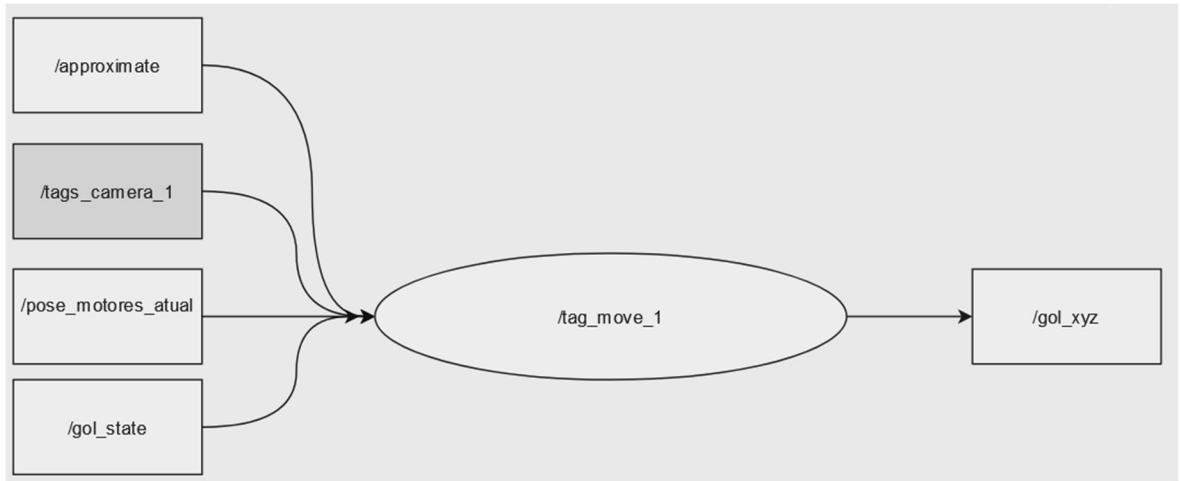


Figura 40- diagrama ROS no `/tag_move_1`

## Os nos Responsáveis pelo controlo dos motores do mecanismo: `/driver_motores`, `/serial_node`

A principal função do nó `/serial_node` é efectuar a comunicação entre o Arduino e o computador permitindo assim o Arduino subscrever e publicar mensagens em ROS de forma simples e direta enquanto o nó `/driver_motores` é utilizado para simplificar o controlo dos motores aplicando a cinemática inversa (coodenadas 3D) para não haver necessidade de inserir os movimentos em passos. O nó realiza a conversão de metros para passos e a conversão do referencial do robô para o referencial da garra.

### **Publicações:**

`/pose_motores_atual`.

`/gol_state`

### **Subscrições:**

`/gol_xyz`

`/gol_state`

### **Objetivos das Mensagens:**

`/pose_motores_atual`: Fornece a posição atual dos motores nos eixos XYZ do mecanismo.

`/gol_state`: Fornece o ponto máximo de movimentação dos motores do mecanismo e indica quando o mecanismo atinge o objetivo solicitado recebido pelo nó `/driver_motores`.

`/gol_xyz`: resebe as coordenadas para as quais o mecanismo deve ser movido.

`/gol_state`: Fornece o ponto máximo de movimentação dos motores do mecanismo e indica quando o mecanismo atinge o objetivo solicitado enviado a partir do nó `/serial_node`.

O diagrama a seguir ilustra a interação entre os diferentes tópicos e como os nós: /cinematica, /serial\_node utilizam essas mensagens para coordenar o movimento dos motores do mecanismo de manipulação de caixas.

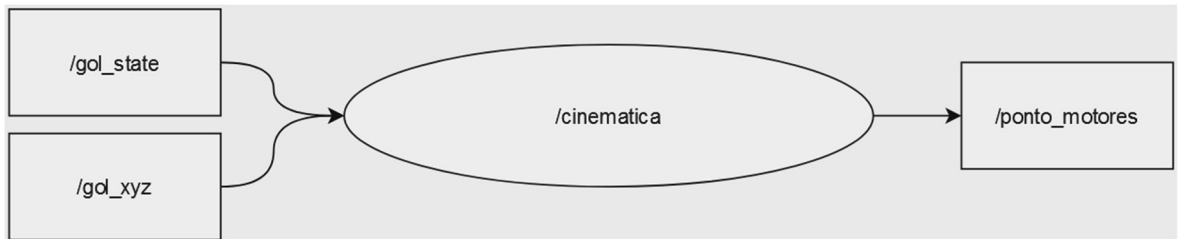


Figura 41-diagrama ROS nós /cinematica

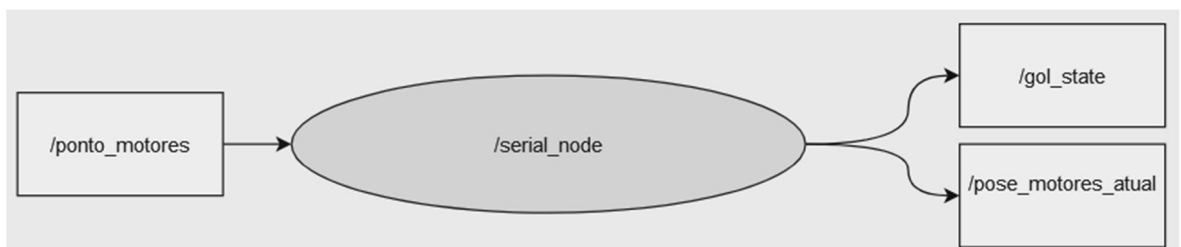


Figura 42-/serial\_node

## **Nós Responsáveis pelo Controlo do MIR100:/MIR100\_missions, /mission\_state\_publisher**

A principal função do nó /MIR100\_missions é adicionar missões à fila de espera do MIR100, enquanto a função do nó /mission\_state\_publisher é publicar o estado atual da última missão da fila de espera do MIR100.

### **Publicações:**

/estado\_da\_missão

### **Subscrições:**

/add\_mission\_queue

/ponto\_objetivo

### **Objetivos das Mensagens:**

/add\_mission\_queue: Adiciona missões à fila de espera do MIR100 através do nome da missão.

/ponto\_objetivo: Adiciona à fila de espera do MIR100 uma missão com o objetivo de movimentar o robô para um ponto no mapa utilizando as coordenadas XYR.

/estado\_da\_missão: Fornece o estado atual da última missão da fila do MIR100.

## Diagrama Representativo

O diagrama a seguir ilustra a interação entre os diferentes tópicos e como os nós `/MIR100_missions` e `/mission_state_publisher` são utilizados para coordenar as missões do MIR100.

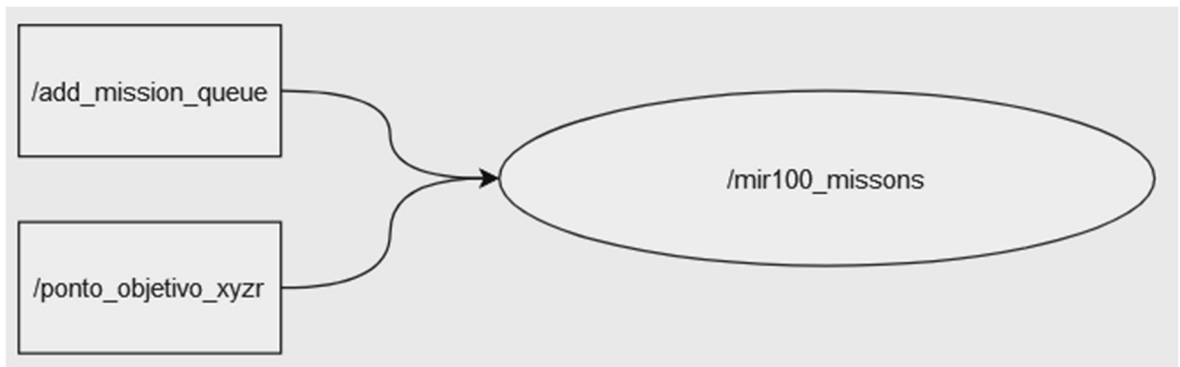


Figura 43-diagrama ROS nos `:/MIR100_missions`



Figura 44-`/mission_state_publisher`

## Nó Responsável pela Detecção das Tags: /apriltag

A principal função do nó /apriltag é publicar uma mensagem com a lista de todas as tags encontradas em frente da câmara, incluindo o ID correspondente a cada uma e as suas respetivas coordenadas.

### Publicações:

/tags\_camera\_1

### Subscrições:

/camera/color/camera\_info

/camera/color/image\_raw

/tf

### Objetivos das Mensagens:

/tags\_camera\_1: Fornece o dicionário das tags existentes na frente da câmara.

/camera/color/camera\_info: Contém as informações da matriz colorida da imagem da câmara.

/camera/color/image\_raw: Contém a matriz colorida da imagem da câmara.

/tf: Fornece a transformada da câmara.

### Diagrama Representativo

O diagrama a seguir ilustra a interação entre os diferentes tópicos e como o nó /apriltag é utilizado para detetar e publicar informações sobre as tags encontradas pela câmara.

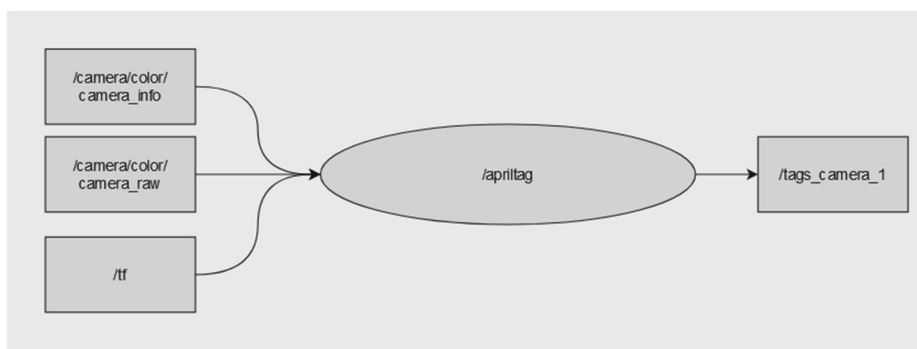


Figura 45-diagram ROS nó /apriltag

## **Conclusão da Validação do Teste Global**

Após a implementação da lógica explicada anteriormente, conclui-se que, utilizando este método, é possível efetuar o controlo de todos os mecanismos anteriormente dimensionados e idealizados de forma prática.

Apesar de algumas dificuldades encontradas na integração de 100% de todos os programas, essas questões podem sempre ser corrigidas e melhoradas para otimizar o funcionamento de todo o sistema.

Portanto, é possível concluir que é viável continuar com esta abordagem para o controlo dos mecanismos, garantindo a precisão necessária para o funcionamento adequado do sistema.

## 5. Conclusão

O objetivo geral deste projeto foi desenvolver um primeiro protótipo de um sistema robótico para a distribuição e recolha de caixas SUC em ambientes industriais. Os objetivos específicos incluíram o desenvolvimento de um Carro de Transporte, um Manipulador para Caixas SUC e a implementação de um sistema de navegação para o robô MIR100 para transporte do Carro.

Nas diferentes fases do projeto, foi possível desenvolver e montar a estrutura do carro de transporte, assim como os mecanismos preliminares para a movimentação das caixas. Testes iniciais nos atuadores e sensores, incluindo a câmara RealSense, foram realizados com sucesso. Desenvolveu-se um programa para a utilização da câmara na localização de tags e implementou-se o sistema de navegação do MiR100, permitindo a execução de missões automatizadas.

Apesar das dificuldades encontradas na integração total dos programas e alguns desafios mecânicos, como o tamanho do carro de transporte, os objetivos do projeto foram alcançados. Os testes realizados validam a viabilidade do sistema proposto, ao nível de hardware e de software. O desenvolvimento deste sistema robótico abordou as necessidades identificadas, permitindo a recolha e distribuição de caixas SUC na fábrica.

Para trabalhos futuros, recomenda-se o desenvolvimento de formas para tornar o mecanismo mais compacto, permitindo a sua operação em espaços mais restritos, além de melhorar o cálculo das rotas e interligar o robô com a gestão de stocks da fábrica. A integração completa com o sistema de stocks da fábrica permitirá uma operação integrada.

Em conclusão, o projeto foi bem-sucedido no seu estágio inicial, demonstrando a viabilidade da automação na logística interna da empresa para caixas SUC.

## 7. Referências

Lista de todos os documentos, artigos, e outros tipos de literatura consultada para o desenvolvimento do projeto.

- [1] Bernardo, R., Sousa, J. M., & Gonçalves, P. J. (2022). Survey on robotic systems for internal logistics. *Journal of manufacturing systems*, 65, 339-350.
- [2] <https://www.magazino.eu/production-logistics/?lang=en>(consultado a 8 de julho de 2024)
- [3] <https://www.zebra.com/us/en/products/autonomous-mobile-robots.html>(consultado a 8 de julho de 2024)
- [4] <https://industrial.omron.eu/en/products/ld-series>(consultado a 8 de julho de 2024)
- [5] <https://mobile-industrial-robots.com/>(consultado a 8 de julho de 2024)
- [6] <https://www.kuka.com/pt-pt/produtos-servi%C3%A7os/amr-robotica-movel-autonoma/rob%C3%B4s-m%C3%B3veis/kmr-iiwa>(consultado a 8 de julho de 2024)
- [7] <https://robotnik.eu/products/mobile-manipulators/rb-kairos/>(consultado a 8 de julho de 2024)
- [8] <https://locusrobotics.com/>(consultado a 8 de julho de 2024)
- [9] MiR shelf lift application Operating guide , 11/2019(consultado a 8 de julho de 2024)
- [10] <https://gemsmotor.com/nema-32-brushless-dc-motor> (consultado a 8 de julho de 2024)
- [11] [https://www.linengineering.com/products/stepper-motors/hybrid-stepper-motors/4418-series?gad\\_source=1&gclid=Cj0KCQjw-uK0BhC0ARiANQtgG03X6KdLBsEofZRMzBCT9IWGGv\\_UqHKX\\_doXX-ajV0nRhNuabEWkjcaAglCEALw\\_wcB](https://www.linengineering.com/products/stepper-motors/hybrid-stepper-motors/4418-series?gad_source=1&gclid=Cj0KCQjw-uK0BhC0ARiANQtgG03X6KdLBsEofZRMzBCT9IWGGv_UqHKX_doXX-ajV0nRhNuabEWkjcaAglCEALw_wcB) (consultado a 8 de julho de 2024)
- [12] <https://www.intel.com.br/content/www/br/pt/architecture-and-technology/realsense-overview.html> (consultado a 8 de julho de 2024)
- [13] MiR shelf lift application Operating guide , 11/2019 (consultado a 8 de julho de 2024)
- [14] [https://actamechatronica.eu/issues/2020/IV\\_2020\\_01\\_Prada\\_Murali\\_Mikova\\_Ligusova.pdf](https://actamechatronica.eu/issues/2020/IV_2020_01_Prada_Murali_Mikova_Ligusova.pdf) (consultado a 8 de julho de 2024)
- [15] Bernardo, R., Sousa, J. M., & Gonçalves, P. J. (2023). A novel framework to improve motion planning of robotic systems through semantic knowledge-based reasoning. *Computers & Industrial Engineering*, 182, 109345.

[16] E. Marchand, H. Uchiyama, and F. Spindler, "Pose estimation for augmented reality: a hands-on survey," *IEEE Transactions on Visualization and Computer Graphics*, vol. 22, no. 12, pp. 2633-2651, Dec. 2016.