

UNIVERSIDADE FEDERAL DO ESPÍRITO SANTO

CENTRO TECNOLÓGICO

PROGRAMA DE PÓS-GRADUAÇÃO EM ENGENHARIA ELÉTRICA

IGOR BATISTA VIEIRA

DISSERTAÇÃO

---

**Virtualização de Ambientes em Tempo Real  
para Interação Multimodal: Teleoperação com  
uso de Realidade Mista e Feedback Háptico**

---

VITÓRIA, BRASIL  
OUTUBRO 2025

UNIVERSIDADE FEDERAL DO ESPÍRITO SANTO

DISSERTAÇÃO

---

**Virtualização de Ambientes em Tempo  
Real para Interação Multimodal:  
Teleoperação com uso de Realidade  
Mista e Feedback Háptico**

---

*Autor:*

Igor Batista Vieira

*Orientador:*

Prof. Dr. Anselmo Frizera Neto

*Coorientador:*

Prof. Dr. Ricardo Carminati de  
Mello

*Dissertação submetida em cumprimento aos requisitos para  
a obtenção do título de Mestre em Engenharia Elétrica*

*no*

Programa de Pós-Graduação em Engenharia Elétrica

Vitória, Brasil

Outubro 2025

Ficha catalográfica disponibilizada pelo Sistema Integrado de Bibliotecas - SIBI/UFES e elaborada pelo autor

---

V657v Vieira, Igor Batista, 1993-  
Virtualização de ambientes em tempo real para interação multimodal : teleoperação com uso de realidade mista e feedback háptico / Igor Batista Vieira. - 2025.  
111 p. : il.

Orientador: Anselmo Frizera Neto.  
Coorientador: Ricardo Carminati de Mello.  
Dissertação (Mestrado em Engenharia Elétrica) -  
Universidade Federal do Espírito Santo, Centro Tecnológico.

1. Realidade virtual. 2. Interfaces de usuário multimodal (Sistemas de computação). 3. Tato. 4. Robótica. I. Frizera Neto, Anselmo. II. Mello, Ricardo Carminati de. III. Universidade Federal do Espírito Santo. Centro Tecnológico. IV. Título.

CDU: 621.3

---

IGOR BATISTA VIEIRA

# **Virtualização de Ambientes em Tempo Real para Interação Multimodal: Teleoperação com uso de Realidade Mista e Feedback Háptico**

*Dissertação apresentada ao Programa de Pós-Graduação em Engenharia Elétrica da  
Universidade Federal do Espírito Santo, como requisito parcial para a obtenção do título de  
Mestre em Engenharia Elétrica.*

Aprovado em Outubro 06, 2025



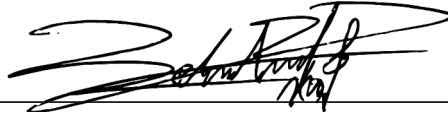
---

Prof. Dr. Anselmo Frizera Neto – Orientador  
Universidade Federal do Espírito Santo



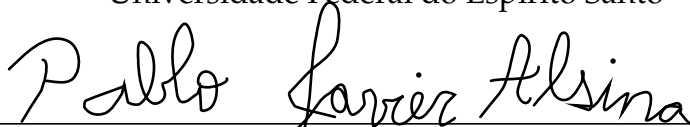
---

Prof. Dr. Ricardo Carminati de Mello – Coorientador  
Universidade Federal do Espírito Santo



---

Prof. Dr. Camilo Arturo Rodríguez Díaz  
Universidade Federal do Espírito Santo



---

Prof. Dr. Pablo Javier Alsina  
Universidade Federal do Rio Grande do Norte

# *Abstract*

Conventional teleoperation interfaces, based on two-dimensional monitors and non-immersive controllers, present significant limitations for human-robot interaction. The absence of depth perception, restricted field of view, and high cognitive load hinder the operator's ability to build an accurate mental model of the remote environment, reducing the effectiveness of robot control. In this context, the integration of immersive technologies and haptic devices emerges as an alternative to enhance the user's sense of presence, overcome perceptual barriers, and make teleoperation more natural and efficient.

To address these challenges, this dissertation proposes the development of a multimodal teleoperation system, composed of a mobile robotic platform equipped with perception sensors, a simultaneous localization and mapping (SLAM) module, and an immersive interface based on Virtual Reality integrated with a haptic device. The architecture was designed to operate in a distributed manner, with processing shared between the robot and the operator station, enabling the construction of a low-latency digital twin. Two experimental studies were conducted: the first validated the accuracy of the visual mapping system compared to classical approaches, while the second evaluated the haptic interface in user teleoperation tasks.

The results obtained confirmed the hypothesis that the combination between Virtual Reality and haptic feedback provides a telepresence experience superior to traditional solutions. The system demonstrated robustness in environment mapping, low response time in data transmission, and an increased sense of immersion reported by the users. Specifically, the user study demonstrated that the immersive interface was able to reduce the average number of collisions from 3.00 to less than 0.3 and decrease the perceived workload (NASA-TLX) by more than 50%. These findings highlight the potential of the proposed approach as a relevant contribution to the advancement of robotic teleoperation, with possible applications in remote inspection, hazardous environments, and human-robot collaboration systems.

**Keywords:** Teleoperation; Multimodal Interface; Virtual Reality; Haptic Feedback; Mobile Robotics.

## Resumo

As interfaces convencionais de teleoperação, baseadas em monitores bidimensionais e controles não imersivos, apresentam limitações significativas para a interação humano-robô. A ausência de percepção de profundidade, o campo de visão restrito e a elevada carga cognitiva do operador dificultam a construção de um modelo mental adequado do ambiente remoto e comprometem a eficácia das ações executadas. Nesse cenário, a integração de tecnologias imersivas e dispositivos hápticos surge como alternativa para ampliar o senso de presença do usuário, reduzir barreiras perceptivas e tornar a teleoperação mais natural e eficiente.

Com esse objetivo, esta dissertação propôs o desenvolvimento de um sistema de teleoperação multimodal, composto por uma plataforma robótica móvel equipada com sensores de percepção, um módulo de mapeamento e localização simultâneos (SLAM) e uma interface imersiva baseada em Realidade Virtual integrada a um dispositivo háptico. A arquitetura foi projetada para operar de forma distribuída, com processamento compartilhado entre o robô e a estação do operador, possibilitando a construção de um gêmeo digital atualizado em baixa latência. Dois estudos experimentais foram conduzidos: o primeiro validou a precisão do sistema de mapeamento visual em comparação com abordagens clássicas, enquanto o segundo avaliou a interface háptica em tarefas de teleoperação com usuários.

Os resultados obtidos confirmaram a hipótese de que a combinação entre Realidade Virtual e *feedback* háptico proporciona uma experiência de telepresença superior às soluções tradicionais. O sistema demonstrou robustez no mapeamento do ambiente, baixo tempo de resposta na transmissão de dados e maior sensação de imersão por parte dos usuários. A avaliação com usuários demonstrou que a interface imersiva foi capaz de reduzir o número médio de colisões de 3,00 para menos de 0,3 e diminuir a carga de trabalho percebida em mais de 50%. Esses achados evidenciam o potencial da abordagem proposta como uma contribuição relevante para o avanço da teleoperação robótica, indicando ainda oportunidades de aplicação em cenários de inspeção remota, ambientes de risco e sistemas de colaboração humano-robô.

**Palavras-chave:** Teleoperação; Interface Multimodal; Realidade Virtual; Feedback Háptico; Robótica Móvel.

## *Agradecimentos*

Gostaria de agradecer a todos que me ajudaram nessa jornada. Sem a presença e suporte de cada um, nada disso seria possível.

Primeiramente, agradeço a Deus e a Nossa Senhora, que sempre me ajudaram a carregar as cruzes que apareciam pelo caminho, nunca me deixando faltar o que fosse necessário para seguir em frente. Agradeço à toda minha família, principalmente a minha mãe, Benedita, e aos meus irmãos, Rhenzo e Beatriz, que me apoiaram em cada decisão e nunca me deixaram desanimar nos momentos difíceis, tornando cada um um pouco mais leve. E também ao meu pai que está hoje no céu intercedendo por mim. Agradeço à minha namorada Marcelli, por enfrentar os desafios ao meu lado e por me ajudar a superar cada um deles, incentivando-me a nunca desanimar e a sempre olhar o copo meio cheio. Assim como a sua família, que me acolheu e hoje em dia é a minha segunda família e segunda casa.

Também gostaria de agradecer a todos os amigos do laboratório *Human Centered Systems* (HCS). Em especial, gostaria de agradecer nominalmente a algumas pessoas. Ao Matheus Penido, com quem cultivo uma amizade desde a graduação e que sempre se fez presente no dia a dia. A Fabiana Machado e ao Marcio Loureiro, pelo suporte técnico e pela companhia. Também ao Sergio Carneiro, que entrou no meio do desenvolvimento do projeto e teve contribuições significativas. Foi muito bom compartilhar as alegrias e desafios do dia a dia com vocês.

Agradeço à todo o corpo docente do Programa de Pós-Graduação em Engenharia Elétrica (PPGEE) da UFES, e especialmente ao meu orientador, professor Anselmo Frizera. Desde a graduação, ele tem sido para mim uma referência como pesquisador e pessoa, ajudando-me a crescer em ambos os aspectos e não medindo esforços para prover todo o suporte necessário. E também ao meu coorientador Ricardo Carminati, que também esteve presente em cada passo desenvolvido, contribuindo diariamente com uma imensidão de ideias. A ambos, minha eterna gratidão.

Agradeço à Fundação de Amparo à Pesquisa e Inovação do Espírito Santo (FAPES) por financiar minha bolsa de estudos para participar deste mestrado, e também aos projetos da FAPES (2021-V4J3L, 2022-C5K3H, 2021-07KJ2, 2025-6VCJ4, 2023-XDHHW), da FAPESP (20/05182-3, 18/23097-3) e do CNPq (308155/2023-8, 403753/2021-0) pelo suporte financeiro fornecido ao longo deste estudo. O apoio desses projetos foi fundamental para a realização e conclusão deste trabalho.

# Sumário

<b>Abstract</b>	<b>i</b>
<b>Resumo</b>	<b>ii</b>
<b>Agradecimentos</b>	<b>iii</b>
<b>Lista de Figuras</b>	<b>vii</b>
<b>Lista de Tabelas</b>	<b>x</b>
<b>Lista de Abreviações</b>	<b>xi</b>
<b>Dedicatória</b>	<b>xii</b>
<b>1 Introdução</b>	<b>1</b>
1.1 Motivação . . . . .	1
1.2 Objetivos da Pesquisa . . . . .	3
1.3 Justificativa . . . . .	3
1.4 Contribuições e Publicações . . . . .	5
1.4.1 Publicações . . . . .	6
1.5 Organização da Dissertação . . . . .	7
<b>2 Fundamentos da Teleoperação Robótica Imersiva</b>	<b>8</b>
2.1 Teleoperação e suas categorias . . . . .	8
2.2 Barreiras em Interfaces de Teleoperação Convencionais . . . . .	11
2.2.1 Déficit na Percepção de Profundidade e Consciência Espacial	11
2.2.2 O Efeito Buraco de Fechadura e a Consciência Situacional . .	12
2.2.3 A Carga Cognitiva e o Problema do Mapeamento . . . . .	13
2.2.4 A Latência na Teleoperação . . . . .	14
2.3 Realidade Mista para Interação Humano-Robô . . . . .	15

2.3.1	Realidade Estendida como Interface Visual . . . . .	15
2.3.2	Virtualização do Ambiente Remoto . . . . .	16
2.3.3	O Gêmeo Digital como Representação Central . . . . .	17
2.3.4	Feedback Háptico e Controle Bilateral . . . . .	19
2.3.5	Interfaces de Teleoperação Imersiva e Multimodal . . . . .	20
<b>3</b>	<b>Interface Háptica Assistiva para Teleoperação Robótica</b>	<b>22</b>
3.1	Arquitetura Geral do Sistema . . . . .	22
3.2	Plataforma Robótica e Processamento dos Dados . . . . .	23
3.2.1	<i>Hardware</i> Embarcado e Dispositivos de Percepção . . . . .	24
3.2.2	Processamento e Otimização dos Dados para Transmissão . . . . .	25
3.3	Módulo de Mapeamento e Virtualização (MMV) . . . . .	28
3.3.1	Implementação do SLAM Visual com RTAB-Map . . . . .	28
3.3.2	Correção de Estimativa de Posição . . . . .	28
3.3.3	Otimização da Nuvem de Pontos para Visualização . . . . .	29
3.4	Interface de Teleoperação Imersiva . . . . .	30
3.4.1	Comunicação e Gerenciamento de Dados . . . . .	31
3.4.2	Gêmeo Digital da Plataforma . . . . .	31
3.4.3	Composição da Cena Virtual e Interface do Usuário . . . . .	32
3.5	Controle e Assistência Háptica . . . . .	35
3.5.1	Mapeamento de Controle e Navegação . . . . .	35
3.5.2	Estratégia de Assistência Háptica Baseada em Proxêmica . . . . .	36
<b>4</b>	<b>Validação Experimental: Comparativo entre Métodos de Mapeamento para Navegação Autônoma</b>	<b>39</b>
4.1	Metodologia Experimental . . . . .	40
4.1.1	Configuração do Sistema . . . . .	41
4.1.2	Protocolo Experimental . . . . .	44
4.2	Resultados e Discussão . . . . .	46
4.3	Conclusões . . . . .	48
<b>5</b>	<b>Avaliação da Interface de Teleoperação com Usuários</b>	<b>50</b>
5.1	Metodologia Experimental . . . . .	51
5.1.1	Delineamento Experimental . . . . .	51
5.1.2	Métricas de Avaliação . . . . .	51
5.1.3	Participantes . . . . .	53

5.1.4	Materiais e Ambiente Experimental	54
5.1.5	Tarefa Experimental	57
5.1.6	Protocolo Experimental	57
5.2	Análise Estatística	59
5.3	Resultados	61
5.3.1	Colisões	63
5.3.2	Tempo de Tarefa	64
5.3.3	Carga de Trabalho Subjetiva (NASA-TLX)	64
5.3.3.1	Demanda Mental (NASA_DM)	66
5.3.3.2	Demanda Física (NASA_DF)	67
5.3.3.3	Demanda Temporal (NASA_DT)	68
5.3.3.4	Percepção de Desempenho (NASA_DE)	68
5.3.3.5	Nível de Esforço (NASA_ES)	68
5.3.3.6	Nível de Frustração (NASA_FR)	69
5.3.4	Percepção de Presença (Questionário PQ)	69
5.3.4.1	Envolvimento (PQ_Envolvimento)	70
5.3.4.2	Adaptação (PQ_Adaptacao)	71
5.3.4.3	Fidelidade Sensorial (PQ_Fidelidade)	71
5.3.4.4	Qualidade da Interface (PQ_Qualidade)	72
5.4	Análise de Correlação e Componentes Principais	72
5.4.1	Análise de Correlação entre Variáveis	72
5.4.2	Análise de Componentes Principais (PCA)	74
5.4.3	Correlação entre Componentes Principais e Desempenho	75
5.5	Análise e Discussão dos Resultados	76
5.6	Conclusões	79
<b>6</b>	<b>Conclusões e Trabalhos Futuros</b>	<b>82</b>
	<b>Bibliografia</b>	<b>85</b>

## Lista de Figuras

2.1	Visão geral de um sistema telerrobótico. Adaptado de Siciliano e Khatib [2]. . . . .	9
2.2	Vistas egocêntrica e exocêntrica em um sistema de manipulação robótico. Adaptado de Jangir et al. [54]. . . . .	10
2.3	Cockpit para teleoperação remota de drones. O operador gerencia dados de multiplas fontes enquanto opera o sistema. Adaptado de Rea e Seo [3]. . . . .	11
2.4	Combinações de FOV de um conjunto de câmeras em um veículo. Adaptado de Cieřlik et al. [56]. . . . .	12
2.5	Interface exibindo multiplas vistas da área de trabalho do operador. Adaptado de Su et al. [10]. . . . .	13
2.6	O Contínuo Realidade-Virtualidade. Adaptado de Milgram e Kishino [68]. . . . .	15
2.7	Gêmeo digital do robô, que espelha os movimentos do robô físico, e nuvem de pontos do ambiente reconstruída via SLAM. Adaptado de Stedman et al. [12]. . . . .	17
2.8	Arquitetura de um DT com realidade mista que integra manipulação e diferentes formas de <i>feedback</i> háptico, comunicando bilateralmente. Adaptado de Fan et al. [79]. . . . .	18
2.9	Exemplo de uma estação de teleoperação utilizando um dispositivo háptico (Novint Falcon) como interface de controle e um monitor 2D para <i>feedback</i> visual. Adaptado de Gliesche et al. [28]. . . . .	19
2.10	DT sendo utilizado como âncora para o sensor LiDAR e estimativa de localização. Adaptado de Solanes et al. [44]. . . . .	20
3.1	Diagrama da arquitetura geral do sistema de teleoperação multimodal, mostrando o fluxo de dados entre o Lado Remoto e o Lado Local.	23

3.2	Plataforma robótica móvel do tipo Ackermann utilizada, mostrando o posicionamento dos sensores e dispositivos. . . . .	24
3.3	Fluxo final dos dados e pacotes do ROS utilizados no processamento. . . . .	25
3.4	Comparativo visual da nuvem de pontos. Em (a), a nuvem de pontos original, densa e com ruídos. Em (b), a mesma nuvem após a aplicação do filtro Voxel Grid. . . . .	26
3.5	A plataforma robótica utilizada e o seu gêmeo digital correspondente no ambiente Unity. . . . .	32
3.6	Estrutura do pacote de dados para um único <i>voxel</i> . A numeração indica o índice de cada <i>byte (offset)</i> no <i>array</i> de dados. . . . .	33
3.7	Dados exibidos na interface. Em (a), informações da câmera de profundidade, sensor LiDAR e janelas de visualização. Em (b), o mapa 3D virtualizado do ambiente. . . . .	33
3.8	(a) Câmeras seguindo o robô, capturando a (b) perspectiva exocêntrica traseira e (c) perspectiva exocêntrica superior. . . . .	34
3.9	Interface de teleoperação em Realidade Mista com três janelas de visualização. . . . .	35
3.10	Dispositivo háptico Novint Falcon, com 3 graus de liberdade para <i>feedback</i> de força e medição de posição, empregado na interação com o sistema robótico. . . . .	36
3.11	Representação visual das zonas de proxêmica elípticas e seus raios de detecção para a estratégia de desvio de obstáculos. Inspirado em [41]. . . . .	37
4.1	Visão geral da implementação do <i>stack</i> de navegação. Figura baseada no <i>stack</i> de navegação do ROS adaptado de Gill [108] para o sistema deste estudo. . . . .	42
4.2	Obstáculos de diferentes alturas posicionados à frente do robô. . . . .	42
4.3	(a) Nuvem de pontos para o <i>setup</i> experimental realizado e (b) mapa de ocupação projetado pela nuvem de pontos. (c) OctoMap gerado para o <i>setup</i> experimental e (d) mapa de ocupação gerado com base no OctoMap. . . . .	43
4.4	<i>Setup</i> experimental com os obstáculos detectados por sensores distintos. . . . .	44
4.5	Região explorada e segmentos de interesse do mapa. . . . .	45
4.6	Configuração dos obstáculos para realização de VSLAM e desvio de obstáculos. . . . .	46

4.7	(a) Mapa 3D do ambiente em nuvem de pontos. (b) OctoMap da região mapeada. (c) Mapa de ocupação 2D gerado por VSLAM. . . . .	46
4.8	Execução da tarefa de navegação autônoma. (a) Definição do ponto de destino (seta verde) no mapa de custo. (b) Posição final do robô após atingir o objetivo com sucesso. . . . .	48
5.1	Interface não imersiva exibindo a vista egocêntrica capturada pela câmera D435i. . . . .	54
5.2	Trajeto a ser seguido na teleoperação e posicionamento dos obstáculos. . . . .	57
5.3	(a) Posição inicial do robô e posição em relação ao teleoperador e (b) posicionamento dos obstáculos no ambiente. . . . .	58
5.4	Protocolo experimental utilizado para a execução de tarefa de teleoperação. As etapas com linhas tracejadas foram feitas com auxílio do ambiente virtual. . . . .	58
5.5	<i>Box plot</i> e número das colisões durante a teleoperação. . . . .	64
5.6	<i>Box plot</i> e violino do tempo para cumprir a tarefa. . . . .	65
5.7	<i>Box plot</i> e violino da média da carga de trabalho percebida utilizando o NASA-TLX. . . . .	65
5.8	Comparação das médias das seis subescalas do NASA-TLX entre as quatro condições experimentais. . . . .	66
5.9	<i>Box plot</i> e violino da média do senso de presença obtido pelo questionário PQ. . . . .	70
5.10	Comparação das médias das seis subescalas do senso de presença entre as quatro condições experimentais. . . . .	70

## Lista de Tabelas

3.1	Largura de Banda dos Tópicos de Sensores Transmitidos Antes e Depois do Processamento. . . . .	27
3.2	Parâmetros de Configuração do RTAB-Map. . . . .	29
4.1	Comparação dos tamanhos dos segmentos de mapa. . . . .	47
5.1	Valores ótimos de Lambda ( $\lambda$ ) da transformação de <i>Box-Cox</i> aplicados a cada variável dependente transformada. . . . .	60
5.2	Condições que não apresentavam distribuição normal (teste de Shapiro-Wilk, $p < 0,05$ ) e que passaram a apresentar após a transformação de <i>Box-Cox</i> . . . . .	61
5.3	Resultados da Análise de Variância Não Paramétrica (ART ANOVA). . . . .	61
5.4	Resultados da Análise de Variância Paramétrica (ANOVA de duas vias pós <i>Box-Cox</i> ). . . . .	62
5.5	Resultados detalhados das comparações <i>post-hoc</i> (ajuste de Tukey) para as variáveis com interação significativa. . . . .	62
5.6	Resultados detalhados das comparações <i>post-hoc</i> (ajuste de Holm) para as variáveis paramétricas com interação significativa. . . . .	63
5.7	Correlações de Pearson significativas ( $p < 0,05$ ) entre as variáveis do estudo. . . . .	73
5.8	Cargas dos componentes principais para as subescalas do NASA-TLX. Os dois componentes explicam 78,14% da variância total. . . . .	75
5.9	Cargas dos componentes principais para as subescalas do Questionário de Presença (PQ). O primeiro componente explica 83,98% da variância. . . . .	75
5.10	Correlações de Pearson significativas ( $p < 0,05$ ) entre os componentes principais (PCA) e as métricas de desempenho. . . . .	76

## Lista de Abreviações

<b>2D</b>	Bidimensional
<b>3D</b>	Tridimensional
<b>ANOVA</b>	Análise de Variância ( <i>Analysis of Variance</i> )
<b>AR</b>	Realidade Aumentada ( <i>Augmented Reality</i> )
<b>DoF</b>	Graus de Liberdade ( <i>Degrees of Freedom</i> )
<b>DP</b>	Desvio Padrão
<b>DT</b>	Gêmeo Digital ( <i>Digital Twin</i> )
<b>FOV</b>	Campo de Visão ( <i>Field of View</i> )
<b>IHR</b>	Interação Humano-Robô
<b>LiDAR</b>	Deteção de Luz e Distância ( <i>Light Detection and Ranging</i> )
<b>PCA</b>	Análise de Componentes Principais ( <i>Principal Component Analysis</i> )
<b>RGB-D</b>	Vermelho, Verde, Azul + Profundidade ( <i>Red, Green, Blue + Depth</i> )
<b>RM</b>	Realidade Mista
<b>ROS</b>	Sistema Operacional para Robôs ( <i>Robot Operating System</i> )
<b>SA</b>	Consciência Situacional ( <i>Situational Awareness</i> )
<b>SLAM</b>	Mapeamento e Localização Simultâneos ( <i>Simultaneous Localization and Mapping</i> )
<b>VSLAM</b>	SLAM Visual ( <i>Visual SLAM</i> )
<b>VR</b>	Realidade Virtual ( <i>Virtual Reality</i> )
<b>XR</b>	Realidade Estendida ( <i>Extended Reality</i> )

*“For those who come after..” - Expedition 33*

# Capítulo 1

## Introdução

### 1.1 Motivação

A teleoperação robótica é um dos paradigmas base da robótica, projetado para estender a presença e a capacidade humana a locais remotos ou perigosos [1, 2]. A eficácia desta interação, no entanto, é historicamente limitada não pela capacidade mecânica dos robôs, mas pelas barreiras sensoriais e cognitivas impostas pela interface que conecta o operador à máquina [3]. Para que a colaboração humano-robô seja bem-sucedida, é necessário que a interface permita ao sistema inferir a intenção humana e traduzi-la em ações eficazes, um desafio que se torna particularmente agudo quando o operador está fisicamente dissociado do robô [4].

O sucesso de um sistema de teleoperação depende, portanto, da capacidade da interface de induzir um estado de telepresença no operador, referenciada como a sensação psicológica de “estar lá” (do inglês “being there”) no ambiente remoto [5-7]. Este estado é alcançado quando o *feedback* sensorial fornecido é suficientemente rico e intuitivo para permitir que o operador forme um modelo mental preciso do espaço de trabalho e comande as ações do robô de forma natural [8].

No entanto, a própria interface, especialmente em operações remotas, pode criar uma dissociação entre o mundo tridimensional (3D) onde o robô está presente e a percepção do operador. Esta dissociação compromete a telepresença e, consequentemente, degrada o desempenho da tarefa, tornando o estudo e a melhoria das interfaces um ponto central da pesquisa na área [3, 9].

Essa desconexão manifesta-se principalmente nas interfaces de teleoperação remota, onde o operador não partilha o mesmo espaço físico que o robô. Nestes cenários,

a percepção do ambiente depende inteiramente de tecnologias de mediação que, tradicionalmente, se baseiam em *feeds* de câmeras 2D projetados em monitores planos [10, 11]. Esta abordagem fundamentalmente limita a capacidade do operador de construir um modelo mental preciso do espaço remoto, constituindo a primeira grande barreira para uma interação imersiva e eficaz [12].

Para realizar a teleoperação remota, o operador controla o sistema com dispositivos como *joysticks*, teclados ou *mouses*, recebendo simultaneamente *feedback* visual de monitores 2D [13]. Esta interface 2D não reflete a forma natural como os humanos interagem com o mundo 3D, tornando o controle do robô pouco intuitivo e exigindo grande esforço do operador para realizar até mesmo tarefas simples [14, 15]. Fatores como a falta de graus de liberdade no controle e, principalmente, a ausência de percepção de profundidade, reduzem drasticamente o desempenho da teleoperação [16, 17].

A falta de informações de profundidade da visão 2D causa ambiguidade sobre as posições espaciais dos objetos. Os operadores possuem dificuldades em estimar distâncias com precisão através da visão monoscópica, tornando a teleoperação dependente da habilidade e proficiência do operador [10, 18]. Embora arranjos com múltiplas câmeras possam compensar parcialmente essa limitação, eles inevitavelmente aumentam a carga de trabalho do operador à medida que a complexidade do sistema cresce, o restringindo a uma perspectiva egocêntrica [19, 20].

Visando superar as limitações de campo de visão, orientação espacial, percepção de profundidade e a inconveniência da troca de atenção em sistemas de múltiplas câmeras, a integração imersiva através da Realidade Mista (RM) surge não apenas como solução, mas como uma mudança de paradigma na perspectiva do operador, usualmente limitada à vista egocêntrica [21, 22]. A RM pode apresentar ao operador um ambiente operacional 3D intuitivo, aliviando a degradação da percepção espacial causada pelas interfaces 2D e permitir a visualização de múltiplas vistas simultaneamente [10].

Ao implementar a RM, o usuário pode experimentar uma sensação de presença física no ambiente remoto, coexistindo com a plataforma robótica enquanto a guia e monitora através de uma perspectiva exocêntrica, aumentando a consciência situacional e facilitando o planejamento de tarefas complexas [23, 24]. Para enriquecer ainda mais essa imersão, o *feedback* háptico (tátil) pode ser integrado, permitindo

que o operador não apenas veja, mas também "sinta" o ambiente virtual [25]. A adição de *feedback* tátil tem demonstrado melhorar o desempenho geral em tarefas que exigem precisão, impactando positivamente métricas como a duração da tarefa e a qualidade do resultado final [26].

Contudo, a integração de tecnologias imersivas não é uma solução trivial de implementar. A seleção de uma interface de usuário adequada é complexa e deve ser avaliada caso a caso [27]. Embora dispositivos hápticos possam reduzir o tempo de execução de certas tarefas, interfaces imersivas podem demandar mais atenção e recursos cognitivos do operador. Portanto, é necessário investigar como a combinação destas modalidades impacta a experiência do usuário e a eficácia da teleoperação [28, 29].

## 1.2 Objetivos da Pesquisa

Esta dissertação de mestrado tem como objetivo principal desenvolver e avaliar um sistema de teleoperação multimodal que integre Realidade Mista e *feedback* háptico, estudando seu impacto no senso de presença, na carga de trabalho do operador e na eficiência da execução de tarefas.

1. Desenvolver uma arquitetura de teleoperação que integre a reconstrução de ambientes 3D, a visualização em Realidade Mista e o *feedback* háptico.
2. Avaliar o impacto da interface de Realidade Mista, em comparação com interfaces de teleoperação 2D tradicionais, na percepção de presença do operador.
3. Analisar os efeitos do *feedback* háptico na carga de trabalho cognitivo e na eficiência da execução de tarefas de navegação.
4. Validar experimentalmente, por meio de estudo com usuários, a eficácia do sistema multimodal integrado na melhoria da performance do operador em tarefas de teleoperação.

## 1.3 Justificativa

Sistemas de teleoperação remota constituem a base para operações em ambientes complexos ou perigosos, sendo tradicionalmente controlados por meio de interfaces não imersivas [30]. Ferramentas como *joysticks*, teclados e monitores 2D são o

padrão da indústria, permitindo que um operador comande um robô à distância. No entanto, o controle do robô pode não ser intuitivo e natural para o usuário [9, 31].

Essa abordagem convencional impõe uma alta carga cognitiva. Ao observar a área de trabalho em apenas uma perspectiva, não é possível perceber a profundidade de forma direta, dificultando o julgamento entre a distância do robô e dos objetos [32], o que pode levar a erros de julgamento espacial e perda de desempenho [13]. Além disso, o uso de joysticks e teclados pode ser prejudicial em determinadas tarefas, principalmente as que exijam graus de liberdade, demandando mais esforço até mesmo em tarefas mais simples [15].

Para mitigar essas falhas, RM e o *feedback* háptico surgem como tecnologias capazes de enriquecer a Interação Humano-Robô (IHR) [33, 34]. A RM compensa problemas de visualização, permitindo criar um ambiente 3D imersivo que sobrepõe reconstruções do ambiente remoto e dados telemétricos diretamente no campo de visão do operador [24, 32, 35]. Dessa forma, a visualização intuitiva gerada pela RM corrige a falta de percepção de profundidade e aumenta a consciência situacional [9, 10].

Apesar de a RM resolver limitações visuais, sua atuação se restringe ao domínio óptico, não engajando outros sentidos, podendo prejudicar tarefas que demandem precisão [33]. A compreensão de sistemas de IHR por parte dos usuários é aprimorada quando estes conseguem visualizar os dados que os robôs capturam por meio de seus sensores, interpretar os estados de seus algoritmos, bem como as rotinas que estão executando e como eles respondem no mundo real [36].

Visando complementar esse *feedback* multissensorial, o *feedback* háptico é utilizado em tarefas que exigem atenção e precisão [18, 37]. Este fornece um senso de toque ao usuário ao controlar um robô de forma remota, podendo aumentar a performance ao desempenhar determinadas tarefas, assim como cirurgias robóticas [38]. A adição do *feedback* tátil pode elevar a sensação de imersão e, conseqüentemente, aumentar o senso de presença do operador [6].

Essa abordagem, que combina o *feedback* visual imersivo da RM com o *feedback* tátil da háptica, é o que define uma interface multimodal. Estas interfaces são projetadas para integrar dados de múltiplos dispositivos de entrada e sensores, não só para aumentar a imersão e o desempenho na tarefa, mas também para permitir que os

robôs processem e interpretem de forma mais eficaz as ações e intenções humanas [17, 39]. Uma das grandes vantagens das interfaces multimodais é a capacidade de envolver vários sentidos, permitindo incorporar elementos que não são possíveis apenas no mundo físico. Isso torna o sistema mais acessível e possibilita a exibição de dados complexos para o usuário sem necessariamente aumentar a sua carga de trabalho [40, 41].

Embora trabalhos recentes explorem essa multimodalidade para avaliar como os *feedbacks* visuais e hápticos contribuem para o senso de presença em um ambiente virtual, a integração de múltiplas fontes de dados ainda apresenta desafios [42]. Interfaces recentes integram o mapa do ambiente, mas aumentam a fadiga durante a operação [43], ou utilizam somente o LiDAR e a odometria do robô, não podendo visualizar o ambiente no qual o robô está inserido [44]. Além disso, todas essas técnicas tendem a criar uma quantidade aumentada de informação que, se não for bem gerenciada, pode ser irrelevante, aumentar a carga cognitiva e exigir maior largura de banda, o que pode impactar negativamente a experiência do teleoperador [40].

Neste contexto, esta dissertação de mestrado propõe o desenvolvimento e a avaliação de um sistema de teleoperação multimodal para a navegação de uma plataforma robótica com rodas, integrando Realidade Mista e feedback háptico. Nestes cenários de uso, serão estudados e analisados os efeitos relacionados ao senso de presença, à carga de trabalho do operador e à execução da tarefa, para investigar os resultados desta integração multimodal e seu impacto no desempenho da teleoperação.

## 1.4 Contribuições e Publicações

A principal contribuição desta dissertação de mestrado é o desenvolvimento e a avaliação de um sistema de teleoperação multimodal que integra uma interface imersiva de Realidade Mista (RM) com *feedback* háptico. Por meio da análise da execução de tarefas, o trabalho fornece *insights* sobre como essa integração multissensorial impacta o senso de presença, a carga de trabalho e o desempenho do operador. A implementação e validação do sistema estabelecem uma base para investigar os efeitos da combinação de estímulos visual e tátil na otimização da interação humano-robô em ambientes remotos. As seguintes publicações foram obtidas como resultado direto desta dissertação de mestrado:

### 1.4.1 Publicações

Os resultados consolidados desta dissertação formam a base de um artigo em preparação para submissão a um periódico da área. Adicionalmente, a pesquisa permitiu a publicação direta do seguinte trabalho em anais de congresso:

**(Congresso Internacional)** CARNEIRO, S. ; VIEIRA, I. ; MACHADO, F. ; FRIZERA NETO, A. ; MELLO, R. An Assistive Haptic Interface for Robotic Teleoperation in Mixed Reality Environments. **Brazilian Symposium on Robotics (SBR)**, 2025.

**(Congresso Nacional)** VIEIRA, I. ; BEZERRA, M. ; LOUREIRO, M. ; MACHADO, F. ; MELLO, R. ; FRIZERA, A. Integração de Câmeras Estéreo para Navegação Autônoma e SLAM Visual em um Robô Móvel. **XXV Congresso Brasileiro de Automática (CBA)**.

Outros trabalhos também foram publicados como consequência da interação com outros pesquisadores durante o desenvolvimento desta dissertação:

**(Congresso Internacional)** GODOY, M. ; MACHADO, F. ; VIEIRA, I. ; MELLO, R. ; FRIZERA, A. Evaluating QoE in Cloud-Based Virtual Reality Systems: A Pilot Study. **2024 Symposium on Internet of Things (SIoT)**.

**(Congresso Internacional)** LOUREIRO, M. ; OLIVEIRA, W. ; MACHADO, F. ; VIEIRA, I. ; MELLO, R. ; FRIZERA, A. Evaluating Differences in Smart Walker-Assisted Locomotion in Augmented and Virtual Reality Environment. **19th IEEE International Conference on Rehabilitation Robotics (ICORR)**.

**(Congresso Nacional)** LOUREIRO, M. ; MACHADO, F. ; BEZERRA, M. ; VIEIRA, I. ; MELLO, R. ; FRIZERA, A. Integrating a Smart Walker and a 3D Human Motion Tracking System for Mixed Reality Immersion Enhancement: A Pilot Study. **29th Congresso Brasileiro de Engenharia Biomédica (CBEB)**.

**(Congresso Nacional)** BEZERRA, M. ; MACHADO, F. ; LOUREIRO, M. ; BATISTA, I. ; MELLO, R. ; FRIZERA, A. ; VIEIRA, I.. Desenvolvimento de Sistemas Robóticos Reproduzíveis: Proposta de Arquitetura e Validação em Andadores Robóticos. **XXV Congresso Brasileiro de Automática (CBA)**.

## 1.5 Organização da Dissertação

Este documento está dividido em seis capítulos, introduzidos a seguir:

O Capítulo 1 apresenta a motivação, os objetivos, a justificativa da pesquisa e as contribuições e publicações resultantes deste trabalho.

O Capítulo 2 resume a base teórica utilizada para esta pesquisa. Nele, são explicados os conceitos de teleoperação e presença remota, interfaces de visualização e controle e mapeamento e localização para ambientes virtuais.

O Capítulo 3 apresenta a arquitetura proposta para sistemas de teleoperação multimodais, detalhando a plataforma robótica utilizada, a arquitetura geral do sistema, o módulo de localização e mapeamento utilizado e a interface desenvolvida.

O Capítulo 4 descreve o primeiro estudo experimental, focado na validação do subsistema de mapeamento e navegação, no qual a abordagem de VSLAM proposta é comparada com um método clássico.

O Capítulo 5 detalha o segundo estudo experimental, que consiste na avaliação da interface de teleoperação multimodal com usuários, analisando os impactos das diferentes configurações no desempenho, na carga de trabalho e no senso de presença dos operadores.

O Capítulo 6 detalha as conclusões gerais do trabalho e as contribuições desta dissertação. Por fim, são sugeridos trabalhos futuros para a continuação e o aprimoramento desta pesquisa.

## Capítulo 2

# Fundamentos da Teleoperação Robótica Imersiva

### 2.1 Teleoperação e suas categorias

A telerobótica, ou robótica à distância, representa um dos campos mais antigos da robótica e uma de suas principais manifestações. O termo, que herda o prefixo grego “tele” (distante), passou a ser usado para indicar a existência de uma barreira entre o utilizador e o ambiente, a qual é superada através do controle remoto de um robô. A telerobótica refere-se a sistemas nos quais um operador humano está no ciclo de controle (*in-the-loop*), de um robô [2].

Neste paradigma, as decisões de alto nível, o planejamento e as capacidades cognitivas são da responsabilidade do ser humano, enquanto o robô se encarrega da sua implementação mecânica no ambiente. O objetivo primário de um sistema telerobótico é, portanto, permitir que um operador transponha uma barreira, seja ela de distância [45], de escala (como na micromanipulação [46]) ou de perigo (como em ambientes radioativos ou submarinos [47]), que o impede de interagir diretamente com o ambiente de interesse.

A arquitetura desses sistemas é dividida em dois locais distintos: o lado operador (*local site*) e o lado remoto (*remote site*) [48]. O lado local engloba o operador humano e todos os elementos da interface do usuário, como *joysticks*, monitores e teclados. O lado remoto contém o robô, seus sensores e atuadores, e o ambiente com o qual ele irá interagir. A estrutura geral de um sistema telerobótico, que ilustra a comunicação entre os sítios local e remoto, é apresentada na Figura 2.1.

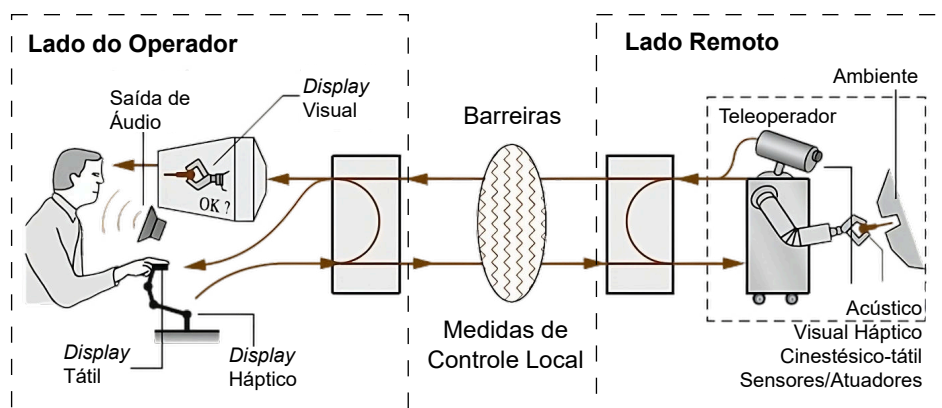


Figura 2.1: Visão geral de um sistema telerrobótico. Adaptado de Siciliano e Khatib [2].

A literatura classifica os sistemas de teleoperação de acordo com diferentes critérios. Uma das categorizações baseia-se na localização relativa entre o operador e o robô. Neste contexto, a teleoperação pode ser dividida em colocada e separada (ou remota) [9].

Na teleoperação colocada, o operador e o robô partilham o mesmo espaço físico. Isto concede ao operador uma visão 3D natural e clara do espaço de trabalho do robô, tornando desnecessário um sistema de *feedback* visual complexo. Neste cenário, tecnologias como a Realidade Aumentada (RA) podem ser utilizadas para sobrepor informações virtuais úteis (como trajetórias planejadas ou dados de sensores) diretamente sobre o ambiente real, enriquecendo a percepção do operador sem o remover do contexto físico [49–51].

A teleoperação remota é aplicada em cenários onde o usuário e o sistema robótico estão distantes, ou quando precisam ser mantidos separados, impedindo que o operador veja os movimentos do robô diretamente [52]. Nesses sistemas, o *feedback* visual e tátil é essencial para mostrar ao operador como o robô interage à distância, permitindo que ele faça ajustes precisos e em tempo real [22]. O grande desafio, contudo, é que os operadores geralmente percebem o ambiente apenas por meio de imagens de câmeras 2D [22, 35].

Outra forma de categorizar a teleoperação é através da perspectiva do operador durante a manipulação, sendo ela egocêntrica ou exocêntrica [53]. A teleoperação egocêntrica ocorre quando o operador observa o processo de manipulação a partir da

mesma perspectiva do robô, percebendo-se como se fosse um com a máquina. Esta abordagem visa transferir as capacidades motoras e a proficiência do humano para o sistema robótico, eliminando a necessidade de o operador mapear mentalmente os seus comandos para as ações do robô. Em contraste, a teleoperação exocêntrica envolve a observação do robô e do seu ambiente a partir de uma perspectiva de terceira pessoa, oferecendo uma visão mais ampla e contextualizada da operação, conforme exemplificado na Figura 2.2.

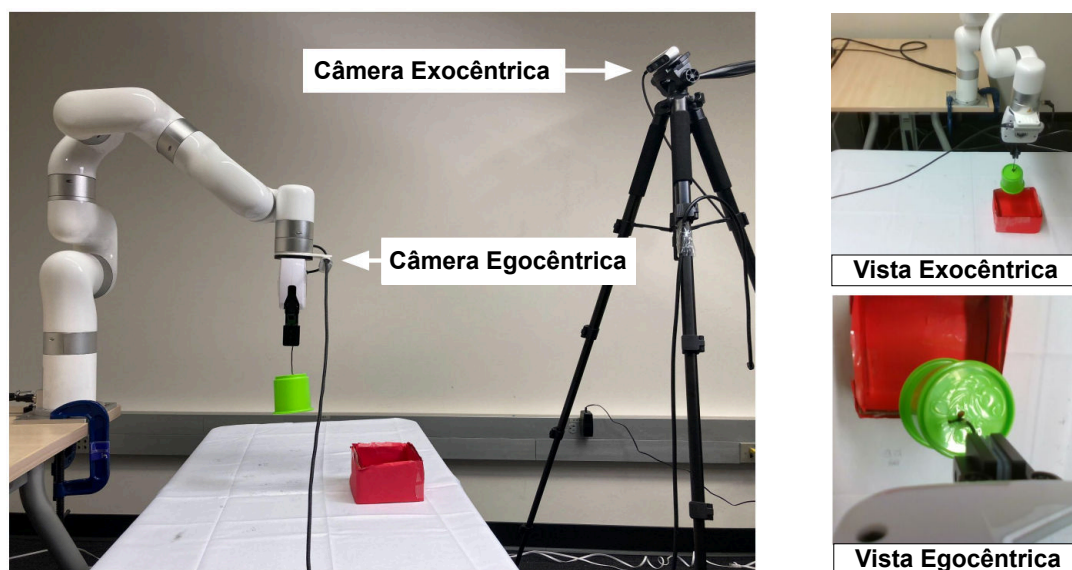


Figura 2.2: Vistas egocêntrica e exocêntrica em um sistema de manipulação robótico. Adaptado de Jangir et al. [54].

Em relação ao controle dos sistemas em telerobótica, podem ser organizadas de acordo com o nível de autonomia concedido ao robô, sendo controle direto, compartilhado e supervisor [52].

No controle direto o robô não possui qualquer inteligência ou autonomia local, tendo todos os seus movimentos são diretamente comandados pelo operador. Já o controle supervisor é o extremo oposto, o operador envia comandos de alto nível e o robô utiliza a sua autonomia local e os seus sensores para planejar e executar a tarefa. Esta arquitetura é particularmente útil em cenários com grande atraso de comunicação, como em operações espaciais. Entre os dois extremos, encontra-se o controle compartilhado, onde a ação do robô é uma combinação de comandos diretos do operador e alguma forma de assistência autônoma, que podem restringir o movimento do robô para evitar zonas perigosas ou guiá-lo ao longo de uma trajetória desejada, combinando a destreza humana com a precisão robótica [52]. *feedback*

## 2.2 Barreiras em Interfaces de Teleoperação Convencionais

A eficácia de um sistema de teleoperação depende da capacidade do operador de perceber o ambiente remoto e executar ações precisas dentro dele. As interfaces tradicionais, construídas em torno de monitores 2D e dispositivos de entrada como *joysticks* e teclados, criam uma série de barreiras que comprometem essa capacidade [3]. Estas barreiras não são questões isoladas, mas sim uma cascata de défices interligados, onde falhas perceptuais desencadeiam uma degradação da consciência situacional, que por sua vez impõe uma carga cognitiva elevada ao operador [3, 10]. A Figura 2.3 ilustra uma estação de controle para um drone, evidenciando a complexidade e a quantidade de informações visuais que o operador precisa processar simultaneamente em múltiplos monitores 2D.



Figura 2.3: Cockpit para teleoperação remota de drones. O operador gerencia dados de múltiplas fontes enquanto opera o sistema. Adaptado de Rea e Seo [3].

### 2.2.1 Déficit na Percepção de Profundidade e Consciência Espacial

A barreira mais fundamental das interfaces 2D é a sua incapacidade de transmitir com precisão a terceira dimensão. A representação de um ambiente 3D complexo numa tela 2D plana colapsa a informação de profundidade, forçando o operador a interagir com um mundo fundamentalmente distorcido [10]. O déficit central reside na ausência de referências estereoscópicas. Um *feed* de câmera monocular elimina a disparidade binocular, a referência que o cérebro humano utiliza para a percepção de

profundidade [40]. Sem ela, o operador é incapaz de julgar intuitiva e precisamente as distâncias e as relações espaciais entre os objetos [55], o que impede diretamente o controle preciso do robô e a manipulação de objetos [10].

Esta falta de percepção de profundidade contribui diretamente para uma fraca consciência espacial [10]. Para compensar, o operador é forçado a recorrer a soluções cognitivamente exigentes, como tentar inferir a profundidade a partir de pistas monoculares (e.g., oclusão de objetos, tamanho relativo), que são frequentemente ambíguas e insuficientes, especialmente em ambientes desconhecidos e não estruturados [22, 40].

### 2.2.2 O Efeito Buraco de Fechadura e a Consciência Situacional

Complementando o problema da profundidade, as interfaces 2D impõem uma segunda limitação perceptual severa: um Campo de Visão (*Field of View* - FOV) restrito. As câmeras montadas no robô fornecem uma janela limitada para o mundo remoto, um fenômeno conhecido como “efeito buraco de fechadura” (*keyhole effect*) ou “visão de canudo de refrigerante” (*soda straw vision*) [56]. Esta visão em túnel, exemplificada na Figura 2.4, elimina a consciência periférica, que é essencial para a navegação e interação seguras [57].

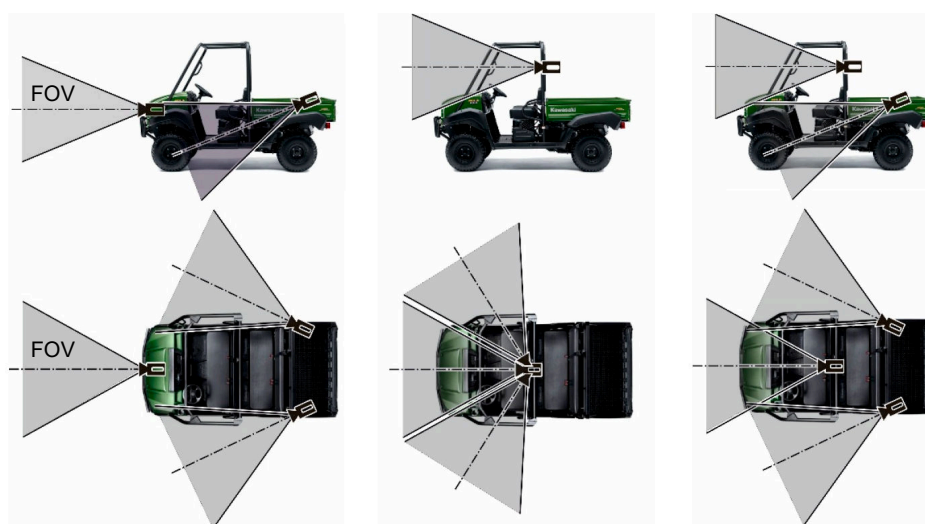


Figura 2.4: Combinações de FOV de um conjunto de câmeras em um veículo. Adaptado de Cieřlik et al. [56].

Esta limitação do FOV é um fator principal na degradação da Consciência Situacional (*Situational Awareness - SA*) — a capacidade do operador de perceber os elementos no ambiente, compreender o seu significado e projetar o seu estado no futuro próximo [58]. Com uma SA baixa, o operador pode não estar ciente de obstáculos fora do campo de visão imediato, tendo dificuldade em orientar o robô e antecipar as consequências das suas ações [40]. A tentativa de mitigar este problema com múltiplos *feeds* de vídeo 2D, como visto na interface da Figura 2.5, muitas vezes agrava a carga cognitiva, pois a carga de integrar mentalmente essas múltiplas visualizações desarticuladas recai inteiramente sobre o operador [10].

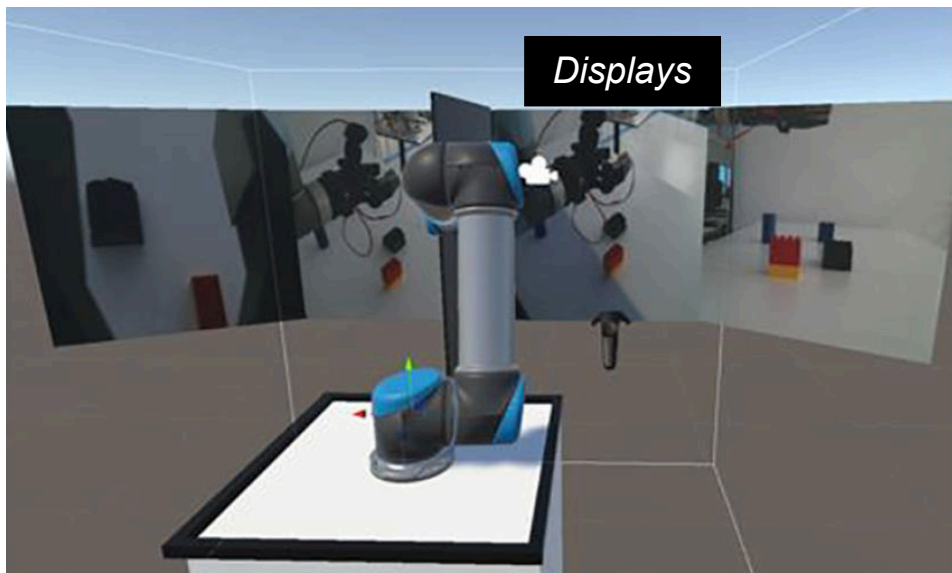


Figura 2.5: Interface exibindo múltiplas vistas da área de trabalho do operador. Adaptado de Su et al. [10].

### 2.2.3 A Carga Cognitiva e o Problema do Mapeamento

A junção de uma percepção deficiente e esquemas de controle não intuitivos impõe um fardo cognitivo significativo sobre o operador. Em vez de se concentrar na tarefa, o operador gasta uma quantidade desproporcional de recursos mentais para interpretar a interface e traduzir as suas intenções em ações do robô [3]. O aumento da carga cognitiva não afeta apenas o desempenho, mas também a segurança, pois pode levar a uma priorização de tarefas que negligencia a proximidade segura com o robô, especialmente em cenários de alta pressão [59].

Estudos comparativos demonstram que a mudança para interfaces imersivas pode reduzir a carga de trabalho, indicando uma mudança fundamental na forma como a tarefa é processada cognitivamente [31, 60, 61]. Por exemplo, no estudo de Triantafyllidis *et al.* [17], a adoção de uma interface multimodal que combinava *feedback* visual, auditivo e háptico resultou numa redução de aproximadamente 49% na carga de trabalho percebida pelos operadores, medida pela escala NASA-TLX, em comparação com uma interface puramente visual.

Além das barreiras perceptuais, o desafio da interação é agravado pelos próprios dispositivos de entrada. *Joysticks* e teclados apresentam um problema fundamental de mapeamento de controle devido à incompatibilidade de Graus de Liberdade (*Degree of Freedom* - DoF) com os manipuladores robóticos. Um *joystick* de 2 DoF controlando um braço de 6 ou 7 DoF força o operador a realizar “trocas de modo” cognitivamente exigentes e demoradas. Em vez de um movimento fluido, o controle torna-se uma série de operações desarticuladas, lentas e não intuitivas [33].

O objetivo final de um sistema de teleoperação avançado é alcançar a telepresença, um estado psicológico no qual o operador sente que está fisicamente presente no ambiente remoto [44]. Um sistema que induz este estado é considerado transparente, pois a interface torna-se “invisível”. As interfaces convencionais, devido a esta cascata de barreiras, falham em fornecer a riqueza sensorial necessária para alcançar uma transparência significativa, justificando a exploração de tecnologias imersivas [62].

#### 2.2.4 A Latência na Teleoperação

Além das barreiras da interface, a latência, sendo o tempo de atraso na comunicação entre o operador e o robô, é um desafio fundamental na teleoperação remota. Mesmo atrasos considerados baixos, a partir de 300 ms, já causam uma deterioração significativa no desempenho da tarefa [63]. Atrasos maiores quebram a sensação de presença e forçam o operador a adotar uma estratégia lenta de mover e esperar (*move-and-wait*), comprometendo a eficiência e a segurança [21, 64].

Este problema se agrava dependendo do meio de comunicação. Em redes locais, a latência é mínima, na ordem de 1 a 6 ms [65]. No entanto, em sistemas que operam via Internet utilizando redes celulares, os atrasos medidos em cenários reais são maiores e inconstantes, com valores médios para comandos de controle de 110 ms (4G)

e 217 ms (3G) [66]. Em casos extremos, como na teleoperação planetária (um robô na Lua controlado da Terra), a latência de aproximadamente 3 segundos inviabiliza o controle em tempo real e exige que o robô possua um alto grau de autonomia para executar tarefas a partir de comandos de alto nível [67].

## 2.3 Realidade Mista para Interação Humano-Robô

Para superar as barreiras perceptuais e cognitivas impostas pelas interfaces tradicionais e alcançar a telepresença, é necessário adotar tecnologias que permitam uma interação multimodal e imersiva. A Realidade Estendida (*Extended Reality - XR*), a virtualização de ambientes e o feedback háptico surgem como os pilares tecnológicos para a construção de sistemas de teleoperação, tornando a interação mais intuitiva, eficiente e acessível [9, 17].

### 2.3.1 Realidade Estendida como Interface Visual

A Realidade Estendida (*Extended Reality - XR*) é um termo abrangente que engloba a fusão de ambientes físicos e virtuais, bem como a interação humano-computador que ocorre nesse novo espaço [9]. O ecossistema da XR é frequentemente descrito pelo Contínuo Realidade-Virtualidade, um espectro definido por Milgram e Kishino [68], que vai desde o ambiente puramente real até o ambiente puramente virtual, como ilustrado na Figura 2.6.



Figura 2.6: O Contínuo Realidade-Virtualidade. Adaptado de Milgram e Kishino [68].

Dentro deste contínuo, a Realidade Virtual (*Virtual Reality - VR*) e a Realidade Aumentada (*Augmented Reality - AR*) são as tecnologias que mais se destacam. Em teleoperação, a VR transporta o operador para o ponto de vista do robô, recebendo uma visão estereoscópica que resolve diretamente o problema da percepção de profundidade e mitiga o efeito buraco de fechadura [69, 70]. Estudos demonstram consistentemente que interfaces imersivas melhoram a eficiência e a consciência situacional, sendo preferidas pelos operadores em comparação com as interfaces convencionais

[17, 30, 31, 71]. A AR, por sua vez, sobrepõe informações virtuais ao mundo real, sendo utilizada para projetar “gêmeos digitais” (*digital twins*) ou guias virtuais que auxiliam o operador em tarefas de manipulação precisa [31, 36, 72].

### 2.3.2 Virtualização do Ambiente Remoto

A viabilidade de uma interface de XR de alta fidelidade depende da capacidade de criar uma representação virtual precisa do ambiente remoto em tempo real. Esta tarefa é realizada por algoritmos de Mapeamento e Localização Simultâneos (*Simultaneous Localization and Mapping - SLAM*), que permitem construir um mapa de um ambiente desconhecido e, ao mesmo tempo, localizar o robô dentro dele [62]. Abordagens como o RTAB-Map, que utilizam dados de câmeras RGB-D, são capazes de gerar nuvens de pontos 3D que podem ser importadas para motores de jogo como o Unity, servindo como base para a cena de VR [58, 73].

A nuvem de pontos é a representação mais direta do ambiente capturada pelos sensores 3D, consistindo num vasto conjunto de vértices posicionados num sistema de coordenadas [74]. No entanto, a sua utilização em tempo real para teleoperação imersiva apresenta um *trade-off* fundamental entre a fidelidade da representação e o desempenho do sistema [12]. Nuvens de pontos densas oferecem um grande detalhe visual, mas são computacionalmente dispendiosas e exigem uma elevada largura de banda para transmissão, o que pode introduzir latência [60].

Para mitigar este problema, são frequentemente aplicadas técnicas de otimização, como a filtragem por voxel (*voxel filtering*), que reduz a densidade da nuvem de pontos agrupando vértices próximos, embora isso possa levar a uma perda de informação geométrica [12]. Abordagens mais recentes exploram até mesmo o uso de representações neurais, como o Gaussian Splatting, para criar visualizações fotorrealistas a partir dos dados de imagem, buscando otimizar ainda mais essa relação entre qualidade visual e desempenho [30, 75].

Esta virtualização, seja por nuvem de pontos direta ou por técnicas mais avançadas, permite que o operador interaja com uma representação tridimensional completa e intuitiva do espaço de trabalho [30]. Dentro deste ambiente virtual, um Gêmeo Digital (*Digital Twin - DT*), uma réplica virtual do robô atualizada em tempo real, pode ser exibido, como mostra a Figura 2.7. Esta abordagem permite que o operador se mova livremente pelo ambiente virtual para inspecionar a cena de diferentes

ângulos, contornar oclusões e obter uma compreensão espacial completa que seria impossível com câmeras fixas [30, 72].

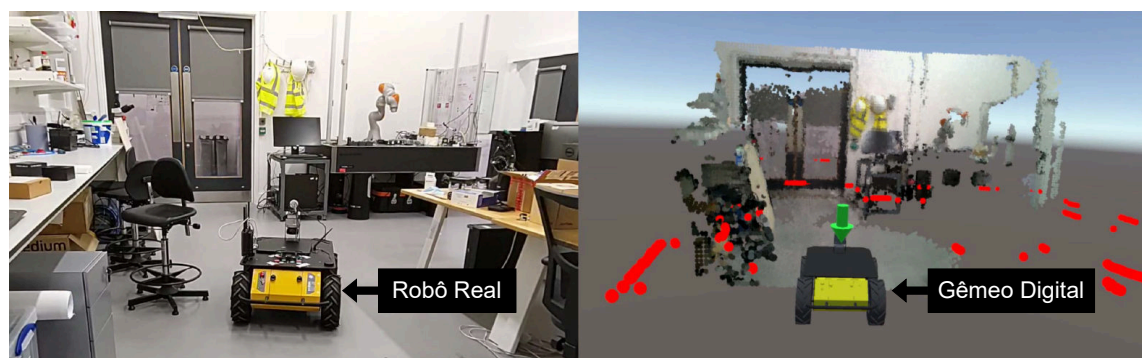


Figura 2.7: Gêmeo digital do robô, que espelha os movimentos do robô físico, e nuvem de pontos do ambiente reconstruída via SLAM. Adaptado de Stedman et al. [12].

### 2.3.3 O Gêmeo Digital como Representação Central

O uso de DT é uma das tecnologias centrais para a integração de sistemas ciberfísicos (*Cyber-Physical System - CPS*) [76]. Por definição, um DT é uma cópia virtual de um objeto ou sistema físico, que espelha as suas propriedades e o seu ambiente [77]. A Figura 2.8 exibe uma arquitetura onde é feita a integração de um *feedback* háptico a um robô por meio do gêmeo digital. Um DT sincroniza-se constantemente com o seu sistema físico por meio de uma transferência contínua de dados, fazendo com que o operador sinta como se estivesse a operar o sistema físico diretamente a partir do ambiente virtual [78].

A integração de tecnologias de RM com a tecnologia de DT facilita uma fusão entre os mundos virtual e real, proporcionando aos operadores um meio intuitivo de observar e controlar sistemas robóticos [80]. O DT pode ser personalizado livremente para um ângulo de visão apropriado com base na preferência do usuário, melhorando a ergonomia do sistema [79].

O elemento que define e diferencia um DT de um modelo digital é a presença de um fluxo de dados automatizado e bidirecional entre a entidade física e a sua contraparte virtual [81]. Essa transferência de dados sincronizada é considerada o principal facilitador do conceito de DT, permitindo que a cópia virtual interaja com o

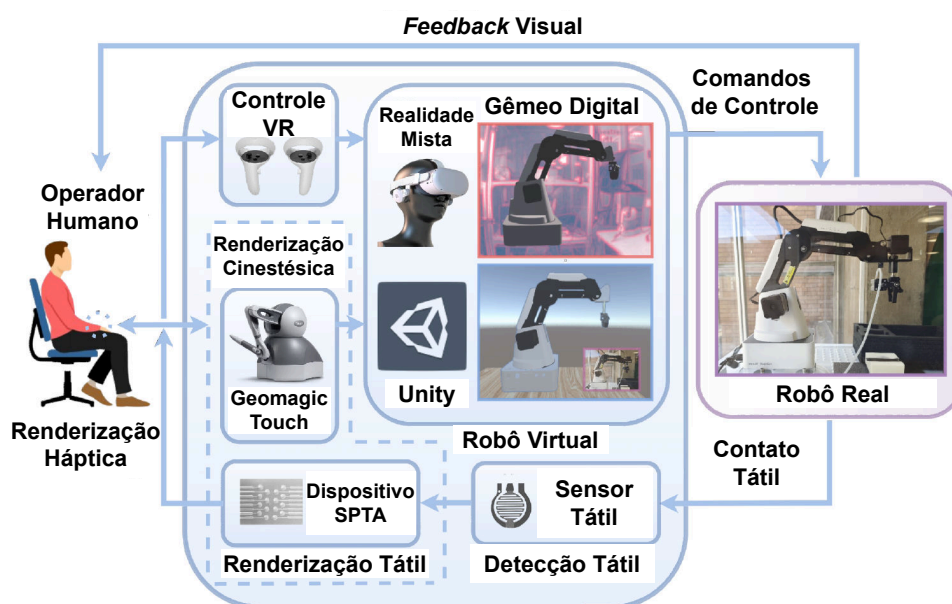


Figura 2.8: Arquitetura de um DT com realidade mista que integra manipulação e diferentes formas de *feedback* háptico, comunicando bilateralmente. Adaptado de Fan et al. [79].

sistema físico [82]. É essa comunicação contínua que efetivamente permite ao operador não apenas monitorar o estado do sistema físico, mas também controlá-lo remotamente, viabilizando a teleoperação [83].

Nesse contexto, o DT atua como uma ponte entre o mundo físico e o cibernético, permitindo que o operador interaja com as funções do robô físico (o *physical twin*) por meio da interface de usuário do DT [84]. A plataforma do DT gerencia essa comunicação, utilizando protocolos usualmente abertos para estabelecer um método eficaz para a teleoperação de robôs a grandes distâncias [83, 85].

A adoção de um DT, em vez da simples transmissão de um fluxo de vídeo, é, portanto, uma decisão de projeto funcional. Enquanto um vídeo oferece apenas uma perspectiva passiva, o DT atua como uma réplica interativa, permitindo que o operador comande o robô através da manipulação direta do modelo virtual [86, 87]. Adicionalmente, esta representação 3D do ambiente é um pré-requisito para funcionalidades avançadas, como a renderização de *feedback* háptico e a adoção de múltiplas perspectivas que aumentam a consciência situacional [88].

### 2.3.4 Feedback Háptico e Controle Bilateral

Enquanto a XR aborda as limitações visuais, o *feedback* háptico complementa a imersão ao engajar o sentido do tato, permitindo ao operador “sentir” o ambiente remoto. O *feedback* é tipicamente dividido em duas modalidades principais: o *feedback* cinestésico, que se refere à percepção de forças e movimentos nos músculos e articulações, e o *feedback* cutâneo (ou tátil), que envolve estímulos na pele, como vibração e textura [89]. A Figura 2.9 ilustra um sistema de teleoperação que integra um dispositivo háptico para fornecer este *feedback* ao operador.



Figura 2.9: Exemplo de uma estação de teleoperação utilizando um dispositivo háptico (Novint Falcon) como interface de controle e um monitor 2D para *feedback* visual. Adaptado de Gliesche et al. [28].

Para a teleoperação de manipuladores, o *feedback* cinestésico é viabilizado pelo controle bilateral. Num sistema de controle bilateral, a comunicação entre o dispositivo controlado pelo operador e o robô ambiente remoto ocorre em ambas as direções: o operador envia comandos de movimento e o robô envia informações de força de volta para a estação de controle [2]. Existem duas arquiteturas principais para implementar este controle: Posição-Posição, que é mais estável, e Posição-Força, que é mais transparente, mas naturalmente menos estável em contato com ambientes rígidos, como paredes, oferecendo resistência ao toque [6].

O *feedback* háptico pode melhorar o desempenho da tarefa e o esforço de controle [26, 33, 38]. Mesmo dispositivos hápticos mais simples, quando comparados com interfaces de teclado e mouse, podem levar a tempos de execução de tarefas significativamente menores, embora nem sempre resultem em maiores taxas de sucesso, evidenciando um complexo balanço de *trade-offs* que justifica a investigação [28, 90].

A integração de XR, virtualização de ambientes e controle bilateral representam o estado da arte das interfaces de teleoperação. Esta abordagem multimodal visa criar uma experiência de telepresença convincente, onde o operador pode ver em 3D, mover-se de forma natural e sentir as interações, superando as limitações fundamentais das interfaces convencionais e tornando a teleoperação mais segura, eficiente e intuitiva [17, 44].

### 2.3.5 Interfaces de Teleoperação Imersiva e Multimodal

A reconstrução tridimensional do ambiente remoto via algoritmos de SLAM, apresentada ao operador em interfaces VR, estabeleceu-se como uma abordagem robusta para superar as limitações de percepção das telas 2D [9, 40]. A eficácia desta técnica foi demonstrada tanto para a manipulação móvel, com redução do tempo de operação [62], quanto para a navegação, com diminuição no número de colisões [58].

De forma complementar, o *feedback* háptico é explorado para engajar o sentido do tato e aprimorar a precisão em tarefas de teleoperação. Investigações como a de Li *et al.* [33] validam que a adição de *feedback* de força em interfaces XR melhora significativamente a taxa de sucesso e a experiência do usuário, embora sua aplicação seja predominantemente focada em tarefas de manipulação.

A fronteira atual avança para a integração de múltiplas fontes de dados, onde a cena virtual é enriquecida com a visualização de sensores como LiDAR e a oferta de múltiplas perspectivas [40]. Nesses sistemas, DT torna-se a âncora da interação, como ilustra a Figura 2.10, espelhando o estado do robô físico e servindo de base para o comando do operador [44].

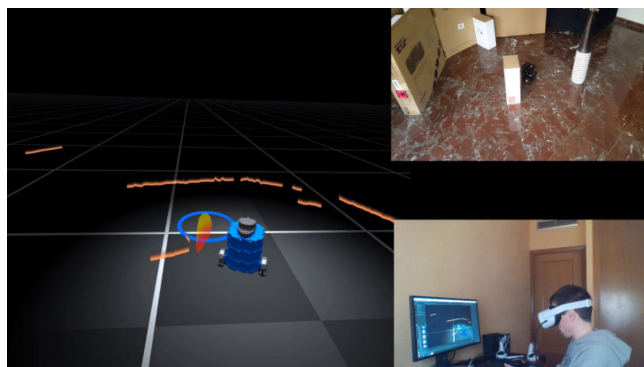


Figura 2.10: DT sendo utilizado como âncora para o sensor LiDAR e estimativa de localização. Adaptado de Solanes et al. [44].

Contudo, os sistemas focados em navegação imersiva raramente integram *feedback* háptico [44, 58, 91], enquanto os que exploram a háptica concentram-se na manipulação [33, 92–94]. Portanto, torna-se necessário desenvolver uma interface que integre e avalie sistematicamente a sinergia dessas modalidades para a locomoção remota. Desta forma, essa dissertação propõe e valida uma arquitetura multimodal que combina um ambiente VR gerado por VSLAM com um controle háptico assistivo para a navegação de um robô móvel, analisando os *trade-offs* de cada modalidade por meio de um estudo com usuários.

## Capítulo 3

# Interface Háptica Assistiva para Teleoperação Robótica

Para lidar com as barreiras de percepção e controle inerentes às interfaces de teleoperação convencionais, foi desenvolvida a arquitetura de um sistema multimodal que integra Realidade Mista e *feedback* háptico. Este capítulo é dedicado à apresentação dos detalhes técnicos desta solução. A discussão inicia-se com uma visão geral da arquitetura do sistema, para então aprofundar em cada um de seus quatro subsistemas principais: a plataforma robótica, o módulo de virtualização do ambiente, a interface imersiva do operador e o controle com assistência háptica.

### 3.1 Arquitetura Geral do Sistema

Os materiais utilizados neste trabalho consistem no sistema de teleoperação multimodal desenvolvido, projetado para oferecer navegação remota assistida e intuitiva. O sistema integra uma plataforma robótica móvel, um conjunto de sensores para percepção, incluindo um sensor LiDAR 2D, um conjunto de câmeras, um *headset* de realidade mista para a visualização imersiva e um dispositivo háptico para o controle e *feedback* de força. A arquitetura de processamento é distribuída: o robô transmite os dados brutos dos sensores, enquanto o processamento de SLAM e a renderização da interface ocorrem no lado do operador, em computadores distintos.

Para viabilizar essa arquitetura distribuída, a solução proposta se baseia na integração de duas plataformas de software principais, o Robot Operating System (ROS 2) e o motor de jogo Unity, conectados por uma rede local. A Figura 3.1 apresenta um

diagrama de blocos da arquitetura geral, ilustrando o fluxo de informações entre o lado remoto (a plataforma robótica) e o lado local (as estações de mapeamento e interação do operador).

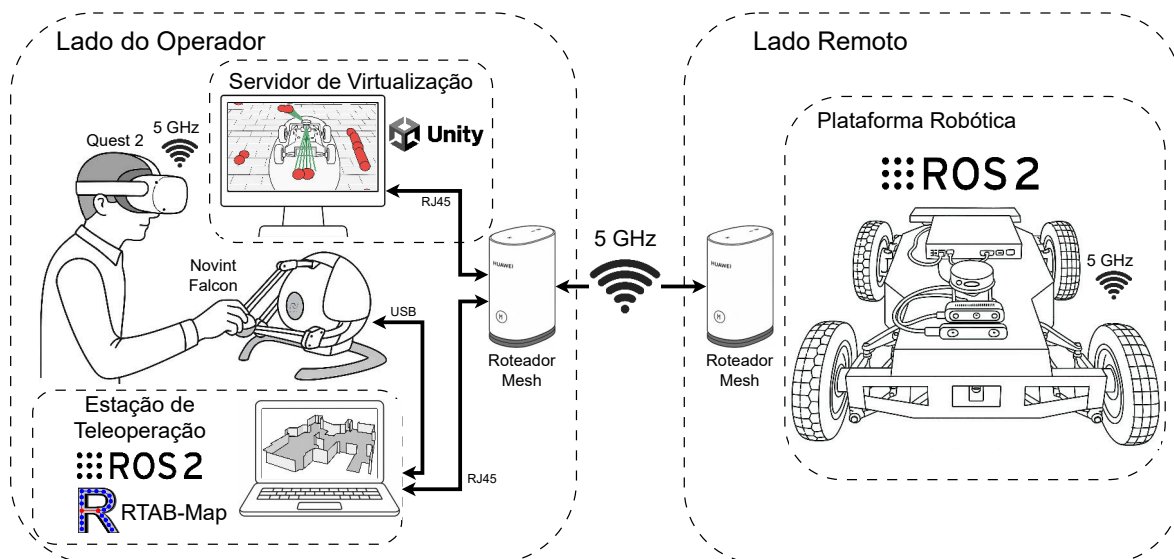


Figura 3.1: Diagrama da arquitetura geral do sistema de teleoperação multimodal, mostrando o fluxo de dados entre o Lado Remoto e o Lado Local.

O fluxo de informações opera em um ciclo contínuo: os sensores no robô capturam dados do ambiente, que são processados a bordo antes de serem transmitidos; a Estação de Teleoperação no lado local recebe esses dados e constrói a representação virtual do mundo, além de capturar a posição do manipulador do dispositivo háptico; o Servidor de Virtualização renderiza essa representação para o operador e utiliza a lógica de assistência para gerar os comandos de velocidade; e os comandos são enviados de volta ao robô, que os executa no ambiente físico. As seções a seguir detalharão cada um desses componentes.

## 3.2 Plataforma Robótica e Processamento dos Dados

O lado remoto é composto pela plataforma robótica física e pelos nós de software embarcados responsáveis por controlar a *hardware* e preparar os dados sensoriais para a transmissão através da rede. A seguir, são detalhados os componentes de hardware e as estratégias de otimização de dados implementadas.

### 3.2.1 Hardware Embarcado e Dispositivos de Percepção

A plataforma robótica móvel utilizada nos experimentos é um robô com modelo cinemático Ackermann (WHEELTEC, China), cujo processamento embarcado é gerenciado por um mini PC OptiPlex 3090 Micro (Dell, EUA), categorizado como *Next Unit of Computing* (NUC). Este computador é responsável pelo controle de baixo nível dos motores e pela publicação dos dados de todos os sensores na rede ROS 2. A plataforma e o posicionamento de seus sensores estão apresentados na Figura 3.2.

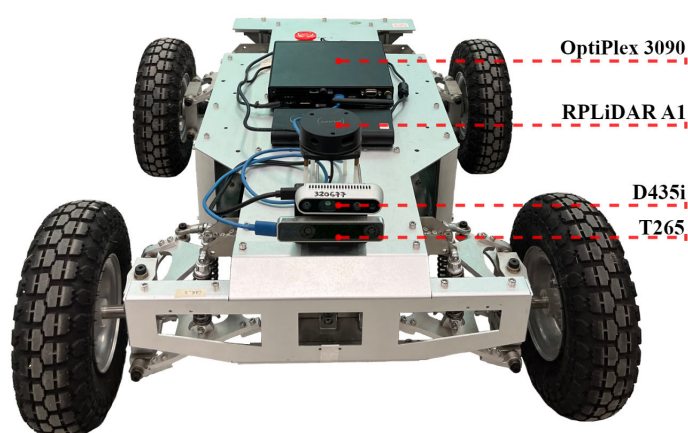


Figura 3.2: Plataforma robótica móvel do tipo Ackermann utilizada, mostrando o posicionamento dos sensores e dispositivos.

Para a percepção do ambiente, o robô foi equipado com um conjunto de câmeras e um sensor LiDAR (RPLiDAR A1, Slamtec, China). A câmera de rastreamento T265 (Intel RealSense T265, Intel, EUA) é utilizada como a fonte de odometria do sistema, fornecendo uma estimativa de pose através de odometria visual inercial. É importante destacar que, para o restante da arquitetura, a T265 não funciona como uma câmera de vídeo, mas sim como um sensor de localização, cuja única saída utilizada é a sua estimativa de posição.

A câmera de profundidade D435i (Intel RealSense D435i, Intel, EUA) é responsável por capturar as informações tridimensionais do ambiente, fornecendo informações de imagem e de profundidade (RGB-D). Adicionalmente, um sensor LiDAR 2D (RPLiDAR A1, Slamtec, China) foi integrado para fornecer medições de distância em um plano de 360 graus ao redor do robô.

### 3.2.2 Processamento e Otimização dos Dados para Transmissão

A transmissão contínua dos dados brutos dos sensores, principalmente da nuvem de pontos e das imagens da câmera RGB-D, gera um volume de informação que demanda uma alta largura de banda da rede sem fio (9,24 MB/s). Essa sobrecarga resulta em alta latência e perda de pacotes, comprometendo a comunicação em tempo real e a fluidez da experiência imersiva. Para mitigar esses problemas, é necessário realizar um pré-processamento nos dados antes de sua transmissão.

Essa abordagem permite filtrar informações irrelevantes, reduzir a densidade dos dados e otimizar sua estrutura, garantindo a compatibilidade e a eficiência no *pipeline* de comunicação. O fluxo final dos dados para serem enviados para o ambiente virtual, assim como os pacotes do ROS utilizados, estão representados na Figura 3.3.

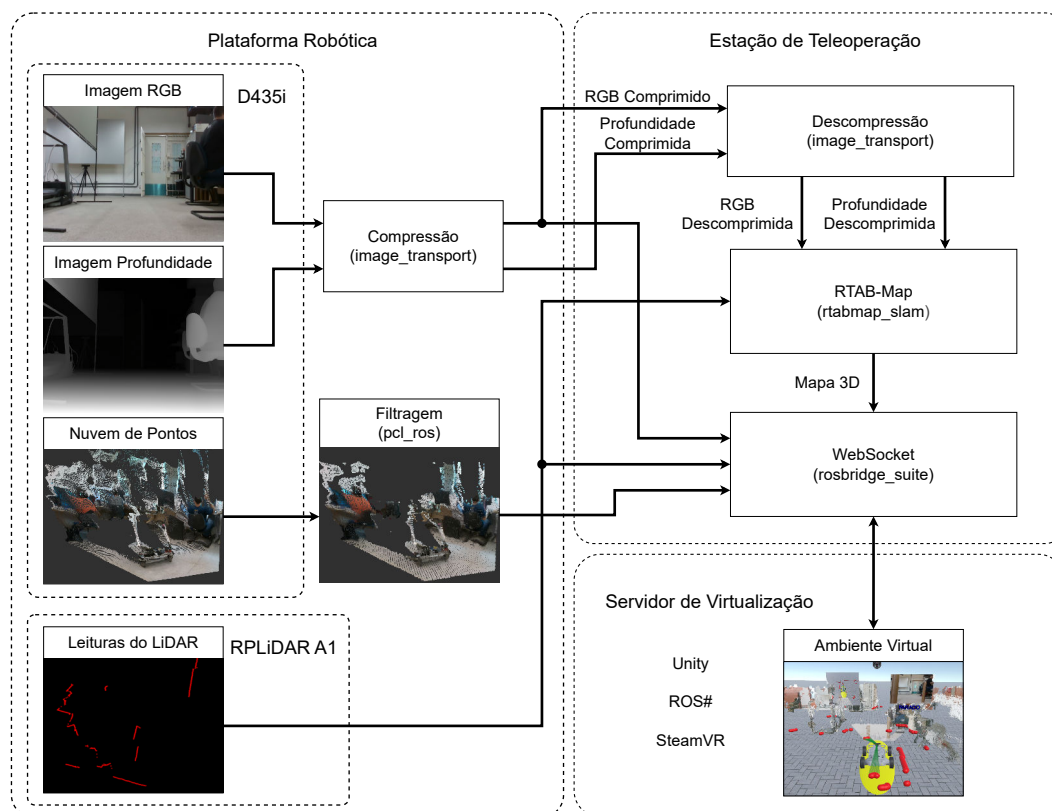


Figura 3.3: Fluxo final dos dados e pacotes do ROS utilizados no processamento.

O primeiro ponto a ser abordado é a Qualidade de Serviço (*Quality of Service - QoS*) dos dados a serem transmitidos. A política de *Reliability* foi definida como *Best Effort*, tentando entregar os dados, mas descartando-os caso haja alguma instabilidade na

rede. Em relação às amostras enviadas, o *History* foi definido como *Keep Last*, com uma profundidade de fila de 1, armazenando somente o último valor. Por fim, a *Durability* dos dados foi definida como *Volatile*, ignorando dados antigos para que nenhuma tentativa seja feita para garantir que os dados sejam entregues. Este conjunto de configurações tem o objetivo de retirar filas dos dados a serem enviados, mantendo o último dado adquirido na fila de processamento.

A imagem RGB capturada pela câmera D435i, sem qualquer processamento, possui uma densidade de dados elevada (13,83 MB/s). Para resolver essa limitação, foi adotado o pacote do ROS *image\_transport*, que comprime os dados da câmera RGB antes de serem enviados (1,12 MB/s), assim a imagem de profundidade, reduzindo a largura de banda utilizada de 9,24 MB/s para 1,1 MB/s. Os dados provenientes da nuvem de pontos também possuem alta densidade (62,77 MB/s). Para viabilizar seu envio e processamento, foi aplicada uma filtragem utilizando o pacote *pcl\_ros*, que realiza a amostragem da nuvem de pontos, aumentando o tamanho dos *voxels* de 0,01 m para 0,04 m, e um filtro de alcance de 0,04 m a 4 m. O resultado visual da filtragem da nuvem de pontos está apresentado na Figura 3.4.

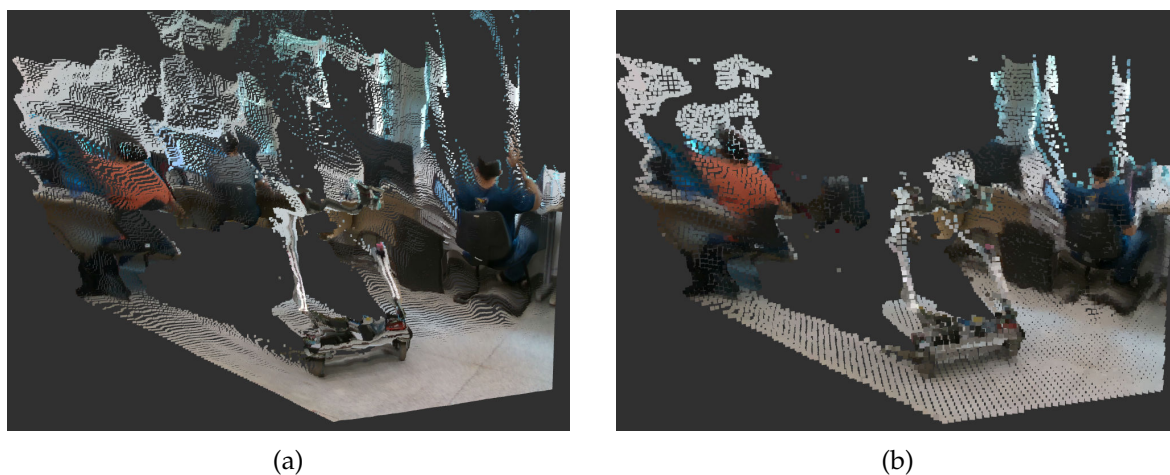


Figura 3.4: Comparativo visual da nuvem de pontos. Em (a), a nuvem de pontos original, densa e com ruídos. Em (b), a mesma nuvem após a aplicação do filtro Voxel Grid.

Por fim, para garantir a estabilidade numérica dos algoritmos de assistência e visualização, as leituras do sensor LiDAR que retornavam valores infinitos, indicando ausência de obstáculos dentro do alcance do sensor, foram saturadas para um valor fixo de 12 metros. Este valor corresponde ao alcance máximo de operação do RPLiDAR A1, tratando efetivamente direções sem retorno como espaço livre até o limite de percepção do *hardware*.

A transmissão simultânea de todos os dados dos sensores, após o processamento, demanda uma largura de banda de aproximadamente 4 MB/s. Este valor, no entanto, não é constante, variando principalmente em função da densidade da nuvem de pontos. A banda consumida diminui quando há menos objetos no campo de visão da câmera, pois a nuvem de pontos se torna mais esparsa. A Tabela 3.1 detalha a contribuição de cada sensor para a largura de banda total.

Tabela 3.1: Largura de Banda dos Tópicos de Sensores Transmitidos Antes e Depois do Processamento.

Formato do Dado	Parâmetros	Largura de Banda (MB/s)	
		Pré-processamento	Pós-processamento
Imagem RGB (D435i)	Resolução: 640x480 Taxa: 15 Hz Codificação: bgr8	13,83	1,12
Imagem de Profundidade (D435i)	Resolução: 640x480 Taxa: 15 Hz Codificação: 16UC1	9,24	1,1
Nuvem de Pontos (D435i)	Taxa: 10 Hz Filtro Voxel: 0,04 m	62,77	1,73
Scan do LiDAR (RPLiDAR A1)	Taxa: 8 Hz	-	0,04

A latência da rede foi avaliada em um cenário crítico, com o robô posicionado a 40 metros da estação do operador, no limite físico da área experimental, utilizando uma rede WiFi 802.11ac de 5 GHz. Ao longo de um minuto de operação, foram registradas 60 amostras de latência, resultando numa média de 5,14 ms com um desvio padrão de 2,85 ms. A literatura apresenta que a latência em ambientes virtuais começa a ser percebida como baixa pelo usuário na faixa de 10-20 ms [21], enquanto em tarefas de teleoperação, atrasos de até 100-250 ms podem não apresentar degradação significativa de desempenho quando comparados a cenários sem latência [95]. Portanto, os valores obtidos no sistema proposto estão bem abaixo dos limiares de percepção e de impacto no desempenho, indicando que a arquitetura de comunicação é robusta o suficiente para garantir uma experiência de usuário fluida e responsiva.

### 3.3 Módulo de Mapeamento e Virtualização (MMV)

A criação de uma experiência imersiva de alta fidelidade depende da capacidade de virtualizar o ambiente remoto de forma precisa e em tempo real. Esta tarefa é realizada pelo Módulo de Mapeamento e Virtualização, cujo núcleo é o algoritmo de Mapeamento e Localização Simultâneos (SLAM). Este módulo, executado na Estação de Teleoperação, é responsável por processar os dados brutos dos sensores do robô para construir um mapa tridimensional do ambiente e, ao mesmo tempo, estimar a pose do robô dentro desse mapa [62].

#### 3.3.1 Implementação do SLAM Visual com RTAB-Map

Neste trabalho, foi implementado um sistema de SLAM Visual (VSLAM) baseado na biblioteca RTAB-Map (*Real-Time Appearance-Based Mapping*), integrada ao ROS 2. A escolha pelo RTAB-Map deve-se à sua robustez, flexibilidade e capacidade de gerar mapas 3D densos a partir de dados de câmeras RGB-D [96]. O nó do RTAB-Map foi configurado para operar em modo de mapeamento ou localização, utilizando a odometria da câmera T265 como estimativa de posição e os dados da câmera D435i para a construção do mapa e detecção de fechamento de laço.

Para otimizar o desempenho e a precisão do mapeamento para a aplicação de teleoperação, uma série de parâmetros foram ajustados no arquivo de inicialização. A Tabela 3.2 detalha os principais parâmetros utilizados e os valores escolhidos, refletindo as necessidades específicas do sistema proposto.

#### 3.3.2 Correção de Estimativa de Posição

A odometria fornecida pela câmera T265, utilizada para o movimento contínuo do gêmeo digital, é suscetível a desvios cumulativos (*drift*). Embora o RTAB-Map forneça correções precisas para este erro, sua baixa frequência de publicação impede uma aplicação direta sem causar instabilidade visual e saltos abruptos na interface imersiva.

Para resolver essa questão, foi implementada uma estratégia de atualização de referência em baixa frequência. Foi desenvolvido um nó de fusão que subscreve ao fluxo de odometria de alta frequência da T265 e também às correções do RTAB-Map.

Tabela 3.2: Parâmetros de Configuração do RTAB-Map.

Parâmetro	Valor Utilizado
Kp/MaxDepth	3,0
Icp/RangeMax	3,0
Grid/RangeMax	3,0
RGBD/NeighborLinkRefining	true
RGBD/ProximityBySpace	true
RGBD/ProximityByTime	false
RGBD/ProximityPathMaxNeighbors	10
Reg/Strategy	1 (ICP)
Vis/MinInliers	12
RGBD/OptimizeFromGraphEnd	false
RGBD/OptimizeMaxError	4,0
Reg/Force3DoF	true
Mem/STMSize	30
RGBD/LocalRadius	5,0
Icp/CorrespondenceRatio	0,2
Icp/PointToPlane	false
Icp/MaxCorrespondenceDistance	0,15
Grid/Sensor	1 (Depth Image)

No entanto, em vez de utilizar cada nova correção assim que ela chega, o nó as armazena e apenas atualiza sua referência de correção interna a uma taxa controlada de 0.1 Hz (ou seja, a cada 10 segundos).

Durante os intervalos de 10 segundos, o nó aplica continuamente a mesma transformação de correção armazenada a todas as mensagens de odometria recebidas da câmera. Ao final de cada intervalo, a referência é atualizada para a correção mais recente fornecida pelo RTAB-Map. Essa abordagem garante que o movimento do gêmeo digital seja suave durante a maior parte do tempo, trocando múltiplos saltos pequenos e frequentes por um único reajuste discreto e menos frequente. O resultado desejado é uma melhora significativa na estabilidade da visualização, tornando a experiência de teleoperação mais confortável e contínua para o operador.

### 3.3.3 Otimização da Nuvem de Pontos para Visualização

A nuvem de pontos gerada pelo RTAB-Map pode ser excessivamente densa e ruidosa para ser renderizada de forma eficiente em tempo real pelo Unity, especialmente utilizando um *shader* de geometria simples. Para tornar a visualização viável sem sobrecarregar o motor gráfico, foi criado um *pipeline* de otimização.

Após o RTAB-Map gerar a nuvem de pontos dos obstáculos, um nó em C++ chamado realiza um processo de filtragem em duas etapas, utilizando a biblioteca PCL (*Point Cloud Library*). Primeiro, um filtro *Statistical Outlier Removal* é aplicado para remover pontos ruidosos e isolados, que são artefatos comuns dos sensores de profundidade, configurado com um número de vizinhos para análise de 50 e um limiar de desvio padrão de 1.0. Em seguida, um filtro *Moving Least Squares* é utilizado para suavizar a superfície da nuvem, com um raio de busca de 0,03 metros e uma ordem polinomial de 2, criando uma representação mais coesa e visualmente agradável, o que melhora a percepção do ambiente pelo operador.

A renderização de uma nuvem de pontos com dezenas de milhares de vértices num único frame pode causar picos de processamento significativos, resultando em quedas na taxa de quadros e quebrando a imersão do operador. A justificativa para a abordagem a seguir reside na simplicidade do shader de renderização utilizado, que não é otimizado para grandes volumes de geometria. Para contornar essa limitação sem recorrer a técnicas mais complexas, como *compute shaders*, foi implementada uma estratégia de segmentação.

Um nó em C++ recebe a nuvem de pontos já filtrada e, em vez de publicá-la de uma só vez, a divide em blocos menores (*chunks*) de no máximo 8000 pontos. Ele então publica esses blocos sequencialmente a uma taxa controlada. A aplicação em Unity, por sua vez, foi programada para receber e renderizar esses blocos de forma incremental. Esta estratégia distribui a carga de processamento ao longo de vários frames, visando uma experiência visual fluida e estável. É importante notar que, embora essa abordagem evite travamentos, ela introduz um efeito de renderização incremental, no qual o mapa do ambiente é construído progressivamente na visão do operador.

### 3.4 Interface de Teleoperação Imersiva

O Servidor de Virtualização é o componente central da experiência do operador, onde os dados do ambiente remoto são transformados em uma cena imersiva e interativa. Esta tarefa é gerenciada por uma aplicação desenvolvida no motor de jogo Unity, escolhido pela sua capacidade de renderização 3D em tempo real, pelo seu amplo suporte a dispositivos de RV e pela sua flexibilidade de integração com sistemas externos como o ROS.

### 3.4.1 Comunicação e Gerenciamento de Dados

A comunicação entre o ecossistema ROS 2 e a aplicação Unity é o núcleo da arquitetura. Para isso, foi utilizado o pacote *rosbridge\_suite*, que estabelece uma ponte de comunicação via *WebSocket*, permitindo que a aplicação Unity, desenvolvida em C#, subscreva e publique em tópicos no ROS 2. O ROS# oferece funcionalidades para a aquisição de dados como imagens de câmera e leituras do LiDAR. No entanto, foi necessário desenvolver rotinas específicas para a decodificação dos dados da nuvem de pontos, utilizando os *bytes* correspondentes para determinar as posições dos *voxels* 3D e os *bytes* destinados à codificação RGB das leituras da câmera de profundidade. Todos os pontos do array são processados e exibidos de forma tridimensional no Unity, atualizando a visualização a cada nova leitura recebida.

Para garantir a sincronização e a fluidez da experiência do usuário, a aplicação em Unity foi projetada para renderizar apenas os dados mais recentes. A cada atualização de frame, quaisquer valores residuais recebidos entre o início e o fim do processamento são descartados. Isso assegura que não haja atraso perceptível entre o ambiente virtual e o ambiente real, mantendo uma sincronização eficaz.

### 3.4.2 Gêmeo Digital da Plataforma

Conforme detalhado na Sessão 2.3.3, o Gêmeo Digital (DT) é o componente da arquitetura responsável por espelhar o estado e o comportamento da plataforma robótica física no ambiente virtual. Na presente arquitetura, ele atua como o elemento central da Interface de Teleoperação Imersiva, servindo como a principal referência visual e interativa para o operador. É o DT que recebe os dados de odometria para se localizar no mapa, que visualiza os movimentos comandados pelo usuário e que serve de âncora para a exibição de outros dados contextuais, como as leituras do LiDAR.

Para a construção do DT deste trabalho, foi utilizado o modelo 3D da plataforma robótica, fornecido pelo fabricante. A estrutura modular do modelo facilitou a manipulação individual de suas partes, como as rodas, permitindo que o seu comportamento no mundo virtual reflita a condição do modelo físico. Para garantir a fidelidade da representação, os principais sensores também foram posicionados no gêmeo digital de acordo com as suas respectivas localizações no robô real, com pequenas discrepâncias sendo corrigidas manualmente. O resultado é um modelo virtual fiel e funcional, como ilustrado na Figura 3.5.

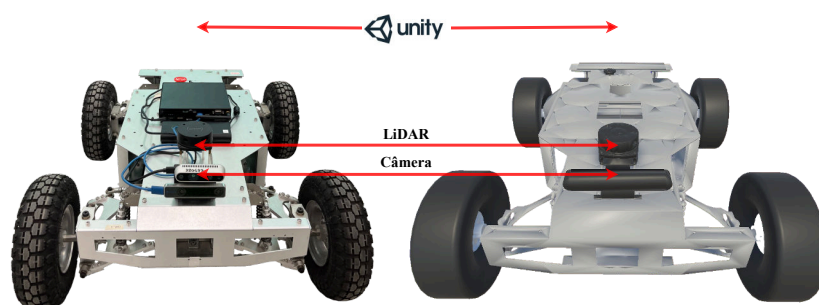


Figura 3.5: A plataforma robótica utilizada e o seu gêmeo digital correspondente no ambiente Unity.

### 3.4.3 Composição da Cena Virtual e Interface do Usuário

A cena virtual apresentada ao operador é composta pela integração de múltiplos dados dos sensores em tempo real. A estimativa de posição do gêmeo digital é atualizada continuamente com base nos dados de odometria gerada pelas leituras da câmera T265 corrigidas pelo fechamento de laço do RTAB-Map. Esta abordagem, validada em nosso trabalho prévio [73], permite que o gêmeo digital represente a pose do robô no ambiente sem comprometer a escala real.

Os dados provenientes da nuvem de pontos da câmera em tempo real e do mapa possuem codificações diferentes, sendo necessário decodificar as informações de cada fonte separadamente. A informação no formato PointCloud2 embarca o comprimento total do *array*, a quantidade de *bytes* utilizados para armazenar a informação de um *voxel* e o *array* com a informação de todos os *voxels*. Desta forma, é necessário decodificar as informações recebidas no Unity, extraindo a informação de cada voxel, conforme exibido na Figura 3.6, utilizando o *offset* de cada informação e o comprimento total dos dados.

Para a renderização do ambiente, a aplicação em Unity utiliza os blocos da nuvem de pontos filtrada, conforme descrito na Seção 3.3.3. A cada novo bloco recebido, a aplicação cria um novo objeto *mesh* na cena. Este é populado com os vértices e as cores do bloco de pontos recebido e, uma vez criado, permanece na cena de forma residual. Desta forma, o mapa 3D completo é construído de forma incremental, como uma composição de múltiplas camadas de *mesh* que são adicionadas à medida que o robô explora o ambiente. Esta estratégia de renderização incremental contorna as limitações de processamento do Unity com grandes nuvens de pontos, distribuindo a carga de trabalho ao longo do tempo e visando uma experiência visual fluida.

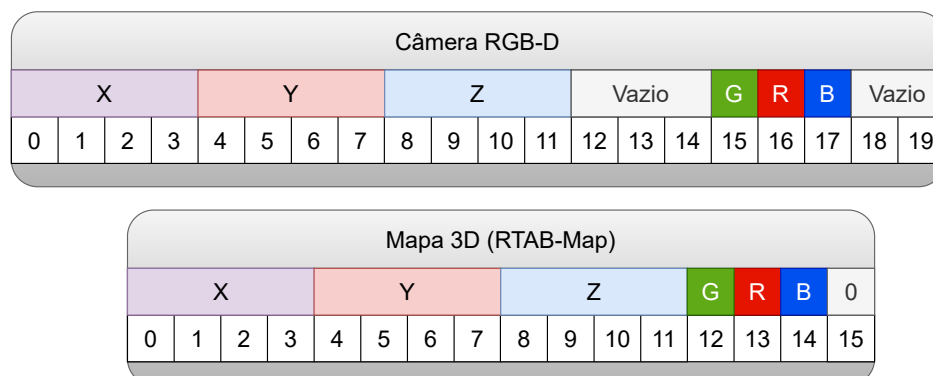


Figura 3.6: Estrutura do pacote de dados para um único *voxel*. A numeração indica o índice de cada *byte* (*offset*) no *array* de dados.

A interface do operador é composta pela visualização de dados em tempo real dos sensores, o mapa do ambiente e diferentes perspectivas do ambiente, conforme exibida na Figura 3.7.

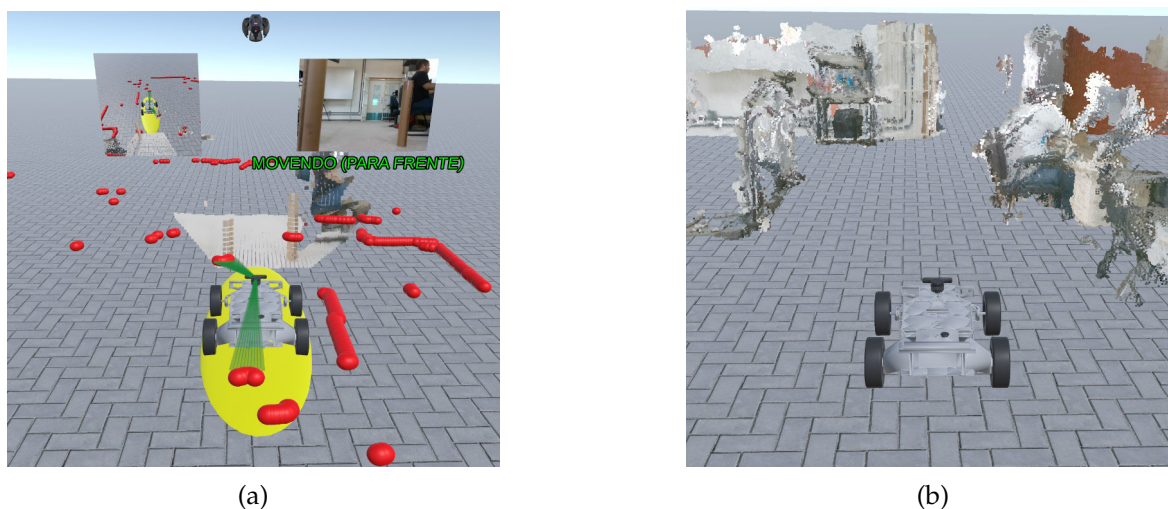


Figura 3.7: Dados exibidos na interface. Em (a), informações da câmera de profundidade, sensor LiDAR e janelas de visualização. Em (b), o mapa 3D virtualizado do ambiente.

A janela principal oferece uma visão exocêntrica da cena, perspectiva priorizada por fornecer uma consciência situacional superior, permitindo ao operador compreender a relação espacial do robô com o ambiente ao seu redor, o que é essencial para o planejamento de trajetórias seguras. A janela superior direita exibe a vista egocêntrica do robô, permitindo que ele tenha detalhes de objetos posicionados imediatamente à frente. Já a Figura 3.7b exibe o mapa 3D virtualizado do ambiente, que é renderizado de forma sobreposta à visão dos sensores em tempo real, permitindo

que o operador tenha tanto o contexto global do mapa quanto os detalhes imediatos do ambiente.

A terceira vista, posicionada na parte superior esquerda da Figura 3.7a, exibe a perspectiva de dois pontos representados por um conjunto de câmeras que seguem o robô durante a exploração (Figura 3.8a). A vista capturada pela primeira câmera resulta na perspectiva exocêntrica traseira, exibindo a visão de objetos e o ambiente atrás do robô (Figura 3.8b). Já a segunda câmera exibe a perspectiva exocêntrica superior do robô, visando melhorar a consciência espacial do operador (Figura 3.8c).

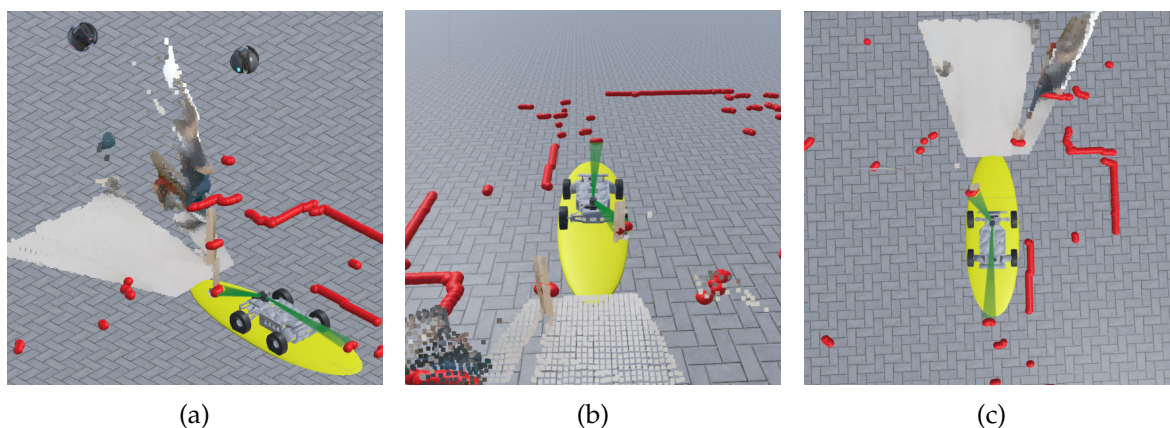


Figura 3.8: (a) Câmeras seguindo o robô, capturando a (b) perspectiva exocêntrica traseira e (c) perspectiva exocêntrica superior.

Por fim, todas as perspectivas são integradas em uma única interface, exibida na Figura 3.9. Desta forma, o operador tem disponível para visualização diferentes perspectivas do robô, incluindo perspectivas exocêntricas e egocêntrica, a interação do robô com o ambiente, e o estado atual do comando de movimento enviado ao robô. A interface é virtualizada e exibida em um *headset* de realidade mista (Oculus Quest 2, Meta, EUA) que está conectada ao sistema pela rede Wi-Fi 5 GHz e é gerenciado pelo *software* SteamVR.

Embora a apresentação de múltiplas fontes de informação exija atenção do operador, o objetivo é fornecer dados complementares que, juntos, maximizam a consciência situacional e reduzem a incerteza na tomada de decisão. A carga de trabalho resultante desta abordagem foi quantitativamente avaliada no estudo com usuários (Capítulo 5).

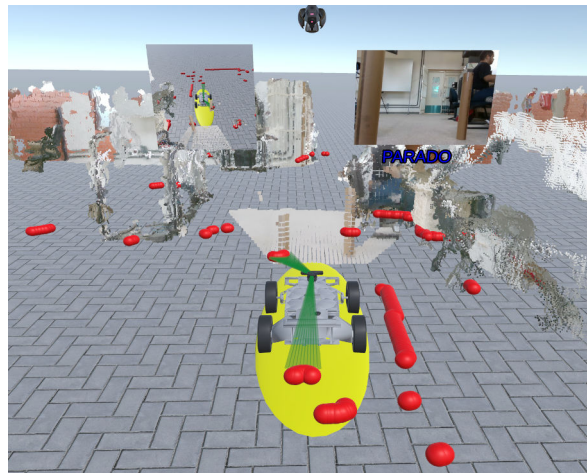


Figura 3.9: Interface de teleoperação em Realidade Mista com três janelas de visualização.

## 3.5 Controle e Assistência Háptica

O subsistema de Controle e Assistência Háptica é o ponto final e interativo do sistema, traduzindo a intenção de movimento do operador em comandos para o robô e, simultaneamente, modulando essa interação com um *feedback* de força. Esta modulação de comportamento é alcançada através de uma estratégia de assistência baseada em proximidade virtual, que cria uma zona de segurança em torno do robô e comunica suas restrições espaciais diretamente ao sentido do tato do operador.

### 3.5.1 Mapeamento de Controle e Navegação

O subsistema utiliza o dispositivo Novint Falcon (Figura 3.10) como a principal interface entre o operador e o robô. Este dispositivo háptico cumpre uma dupla função: serve como um controlador de movimento e, simultaneamente, como um atuador que aplica o *feedback* de força assistivo no operador.

Para o controle de navegação, a posição tridimensional do efector do dispositivo é mapeada para comandos de velocidade bidimensionais (mensagens do tipo *Twist*). O deslocamento para a frente e para trás no eixo Z do Falcon controla a velocidade linear do robô, enquanto o deslocamento para os lados no eixo X controla a sua velocidade angular. Estes comandos são publicados pelo Servidor de Virtualização e recebidos pelo NUC no robô, que os traduz em comandos de baixo nível para os motores.



Figura 3.10: Dispositivo háptico Novint Falcon, com 3 graus de liberdade para *feedback* de força e medição de posição, empregado na interação com o sistema robótico.

### 3.5.2 Estratégia de Assistência Háptica Baseada em Proxêmica

A estratégia de interação define uma zona de detecção elíptica ao redor do gêmeo digital do robô, conforme ilustrado na Figura 3.11. Esta zona é dividida em uma seção de detecção frontal (orientada para  $\pi$  radianos) e uma traseira (orientada para 0 radianos). A posição angular de um obstáculo, representada pela variável  $\beta$ , determina em qual seção ele se encontra. Para a seção frontal,  $\beta$  varia de  $\frac{\pi}{2}$  a  $\frac{3\pi}{2}$  radianos ( $90^\circ$  a  $270^\circ$ ), enquanto para a seção traseira, varia de  $-\frac{\pi}{2}$  a  $\frac{\pi}{2}$  radianos ( $-90^\circ$  a  $90^\circ$ ).

Quando um obstáculo é detectado, o sistema avalia sua posição em relação à zona elíptica. A variável  $l$  corresponde à distância do obstáculo medida pelo LiDAR, e a variável  $r$  indica a distância do centro da elipse até sua borda na direção do ângulo  $\beta$ . A partir disso, calcula-se a profundidade de avanço do obstáculo na zona, representada por  $\tilde{d} = r - l$ . Se  $\tilde{d} > 0$ , o obstáculo está dentro da zona de segurança, e valores maiores de  $\tilde{d}$  indicam uma maior proximidade ou um avanço maior, desencadeando o cálculo da força háptica.

A força de assistência ( $F_h$ ) é calculada e renderizada no dispositivo Novint Falcon para guiar o operador. As componentes da força nos eixos lateral (x) e longitudinal (z),  $F_{hx}$  e  $F_{hz}$  respectivamente, são determinadas pelas seguintes equações:

$$F_{hx} = \frac{1}{2} |\sin(\theta_d)| \tanh(K_1 \tilde{d}) + \text{sign}[\sin(\theta_d)] \cdot K_2 \quad (3.1)$$

$$F_{hz} = \frac{1}{2} |\cos(\theta_d)| \tanh(K_3 \tilde{d}) + K_4 \quad (3.2)$$

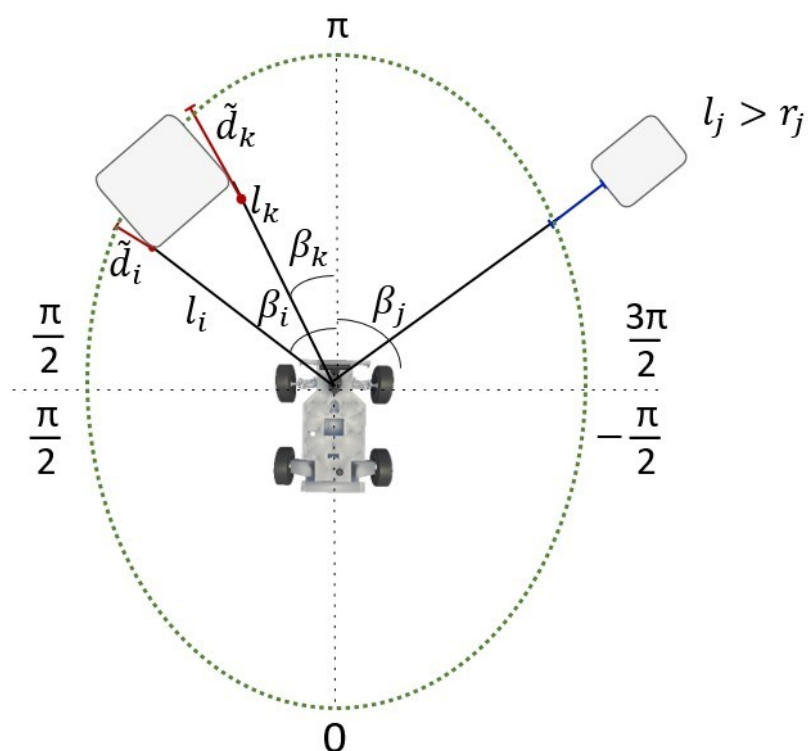


Figura 3.11: Representação visual das zonas de proximidade elípticas e seus raios de detecção para a estratégia de desvio de obstáculos. Inspirado em [41].

Nestas equações,  $\theta_d$  é o ângulo de desvio desejado, calculado ajustando-se o ângulo médio dos obstáculos detectados em  $\pm\frac{\pi}{2}$  radianos [41]. Esta abordagem não apenas empurra o operador para longe do obstáculo, mas o guia ao longo de uma trajetória tangencial a ele. As constantes de ganho  $K_1$  e  $K_3$  ajustam a inclinação da função de tangente hiperbólica ( $\tanh$ ), definindo a sensibilidade da resposta háptica à profundidade de avanço  $\tilde{d}$ . O ganho  $K_2$  ajusta a magnitude da força lateral baseada na função sinal ( $\text{sign}[\cdot]$ ), enquanto a constante  $K_4$  adiciona um componente de força longitudinal constante. Juntos, esses termos estabelecem os limites e o comportamento base da força de assistência. A combinação desses termos cria uma força que aumenta suavemente à medida que o operador se aproxima do obstáculo, proporcionando um *feedback* progressivo e informativo.

Estas equações são implementadas em um script C# no ambiente Unity. A aplicação comunica-se com o dispositivo háptico e atualiza continuamente a magnitude e a direção da força para refletir a proximidade do robô aos obstáculos. Essa modulação do controle permite que o sistema adapte o comportamento dinâmico da

interação, aumentando a resistência para desencorajar o movimento em direção a um obstáculo detectado ou guiando suavemente a mão do usuário para uma direção alternativa. Essa capacidade de modulação visa proporcionar uma experiência de navegação segura e assistida, ao mesmo tempo que permite ao operador manter um nível de controle compartilhado sobre a plataforma.

Neste capítulo, foram detalhadas a arquitetura e a implementação do sistema de teleoperação multimodal proposto. Apresentou-se a integração entre os componentes de *hardware* e *software*, o módulo de virtualização do ambiente, a interface imersiva e a estratégia de controle com assistência háptica, que juntos formam uma solução completa para a interação humano-robô remota. Com a plataforma de desenvolvimento estabelecida, o próximo passo consiste em sua validação experimental. O capítulo seguinte, portanto, se dedicará a avaliar o desempenho do subsistema de mapeamento, um componente fundamental para a viabilidade de toda a solução.

## Capítulo 4

# Validação Experimental: Comparativo entre Métodos de Mapeamento para Navegação Autônoma

Este primeiro estudo experimental tem como foco a validação dos componentes de percepção e locomoção da plataforma robótica. Antes de integrar a interface de teleoperação háptica, é essencial validar que o sistema de VSLAM é capaz de gerar mapas suficientemente precisos e que estes podem ser utilizados para uma navegação segura. Para realizar essa validação, este estudo utiliza a navegação autônoma como um método de avaliação. A premissa é que, se o mapa gerado for de qualidade suficiente para ser utilizado com sucesso pelo *stack* de navegação do ROS, sua confiabilidade para a teleoperação assistida por um operador humano estará validada.

A navegação é uma tarefa essencial para robôs móveis, pois eles precisam se mover pelo ambiente para executar suas atividades [97]. O SLAM (*Simultaneous Localization and Mapping*) é fundamental para essa finalidade, permitindo que o robô se localize em um ambiente desconhecido enquanto constrói seu mapa de forma incremental [98]. Para muitas aplicações, o SLAM é simplificado para duas dimensões (2D) com algoritmos como o GMapping, que constroem mapas de alta precisão [99, 100]. No entanto, essa abordagem 2D não detecta obstáculos localizados abaixo ou acima do plano de medição, o que pode levar a colisões e comprometer a navegação [101].

O SLAM Visual (VSLAM) surge como uma opção para extrair mais informações do ambiente, possibilitando a integração de informações visuais na navegação e abrangendo o plano 3D [102, 103]. Dentre as soluções existentes, algoritmos como

o ORB-SLAM2 destacam-se e podem ser integrados ao *stack* de navegação do ROS (*Robot Operating System*) para a execução de tarefas de navegação autônoma [104, 105].

Para a implementação de um sistema VSLAM, o uso de câmeras na estimativa de posição pode apresentar resultados superiores às técnicas mais convencionais, como a odometria a partir das rodas [106]. Além disso, erros de escala têm um impacto direto na realização do SLAM, tornando as abordagens baseadas em câmeras RGB-D mais eficazes do que aquelas que utilizam apenas câmeras RGB [96]. A abordagem com o uso do RTAB-Map foi, portanto, escolhida neste trabalho por sua capacidade de realizar SLAM em 2D e 3D e por sua flexibilidade na integração de múltiplos sensores [107].

Neste contexto, este estudo avalia a viabilidade da abordagem de VSLAM utilizando o RTAB-Map como base de percepção para a plataforma robótica. Para isso, foi conduzida uma análise quantitativa comparando os mapas 3D gerados pela solução visual com os mapas 2D de um método clássico baseado em LiDAR (GMapping). O objetivo é validar se o VSLAM é capaz de produzir mapas com precisão métrica suficiente para serem utilizados em uma tarefa de navegação autônoma com desvio de obstáculos, justificando assim a sua adoção como o principal módulo de percepção para os estudos de teleoperação com usuários.

Adicionalmente, a escolha por reconstruir o ambiente via VSLAM, em vez da transmissão direta de uma imagem estereoscópica, é uma decisão fundamental para a arquitetura de teleoperação proposta. Enquanto um vídeo estereoscópico poderia oferecer maior fotorrealismo, a representação 3D do ambiente é o que viabiliza funcionalidades centrais do sistema, como a perspectiva exocêntrica com livre movimentação do observador na interface de Realidade Virtual e o *feedback* háptico, superando as limitações de interfaces baseadas apenas em vídeo.

## 4.1 Metodologia Experimental

A metodologia para a validação da abordagem de VSLAM foi estruturada para avaliar tanto a qualidade do mapa gerado quanto a sua aplicabilidade em uma tarefa de navegação. Para isso, o protocolo experimental foi dividido em duas fases: um experimento de mapeamento comparativo e um experimento de navegação autônoma com desvio de obstáculos.

Para este primeiro estudo de validação, optou-se por uma configuração simplificada para isolar a *performance* da odometria visual. O fechamento de laço do RTAB-Map foi intencionalmente desativado, delegando a tarefa de localização e correção de *drift* exclusivamente à câmera de *tracking* T265 e seu fechamento de laço utilizando a odometria visual inercial interna. O objetivo desta configuração foi avaliar a precisão da odometria pura em um cenário de mapeamento, antes de introduzir as complexidades da fusão com o fechamento de laço do RTAB-Map, que será ativado no experimento com usuários apresentado no capítulo seguinte.

### 4.1.1 Configuração do Sistema

Para a integração dos dados das câmeras e a geração dos mapas 2D e 3D, utilizou-se a biblioteca RTAB-Map, integrada ao *middleware* ROS. A biblioteca RTAB-Map realiza uma correção na estimativa da posição da odometria de entrada do sistema por meio do fechamento de laço. Quando uma característica proeminente do ambiente é detectada, a odometria de entrada é ajustada para que a estimativa da posição do robô coincida com a característica detectada no ambiente.

Visando comparar a abordagem visual com o GMapping, um dos algoritmos de SLAM 2D mais utilizados, gerou-se um mapa 2D com o conjunto de câmeras e outro com o algoritmo GMapping, que utiliza o LiDAR e a odometria das rodas. Em seguida, compararam-se os tamanhos de determinados segmentos dos mapas obtidos para verificar a preservação da escala do ambiente mapeado.

Por fim, o mapa derivado das câmeras é integrado ao *stack* de navegação do ROS, juntamente com informações visuais sobre obstáculos na altura do robô, obtidas em conjunto com o sensor LiDAR. A navegação autônoma é executada em um ambiente não estruturado, enquanto o mapa é atualizado de forma incremental.

A Figura 4.1 detalha a implementação do *stack* de navegação do ROS, que coordena a movimentação autônoma do robô. O processo é iniciado quando um objetivo de navegação (tópico *move\_base\_simple/goal*) é enviado ao nó principal, o *move\_base*. A partir daí, o planejador global (*global\_planner*) utiliza o mapa estático principal, fornecido pelo RTAB-Map, e a posição atual do robô para calcular a rota ideal de longa distância até o alvo.

Essa rota global é então enviada ao planejador local (*local\_planner*), que é responsável por gerar os comandos de velocidade em tempo real (*cmd\_vel*) para o controlador

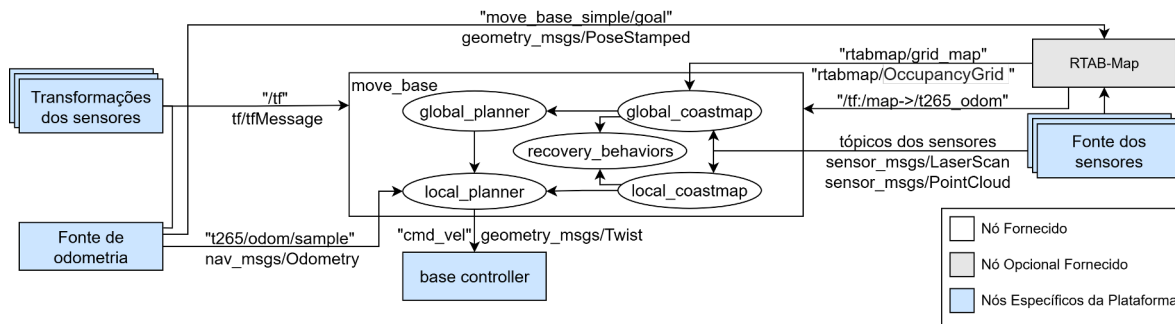


Figura 4.1: Visão geral da implementação do *stack* de navegação. Figura baseada no *stack* de navegação do ROS adaptado de Gill [108] para o sistema deste estudo.

da base (*base\_controller*). Para fazer isso, o *local\_planner* consulta um mapa de custo local (*local\_costmap*), que é atualizado dinamicamente com os dados dos sensores de percepção (LiDAR e nuvem de pontos), permitindo que o robô desvie de obstáculos não mapeados ou que surgem em seu caminho. Todo o sistema depende da árvore de transformações (*tf*) para manter a consistência espacial entre o mapa, a odometria e a posição dos sensores.

A Figura 4.2 ilustra um cenário de testes montado para avaliar o desempenho das câmeras e a qualidade dos mapas gerados. Esse ambiente de testes permite uma verificação detalhada dos mapas produzidos e a validação da integração do *stack* de navegação.



Figura 4.2: Obstáculos de diferentes alturas posicionados à frente do robô.

A Figura 4.3a exibe a nuvem de pontos gerada pela câmera RGB-D D435i. Esses pontos contêm informações tanto de cor quanto de profundidade.

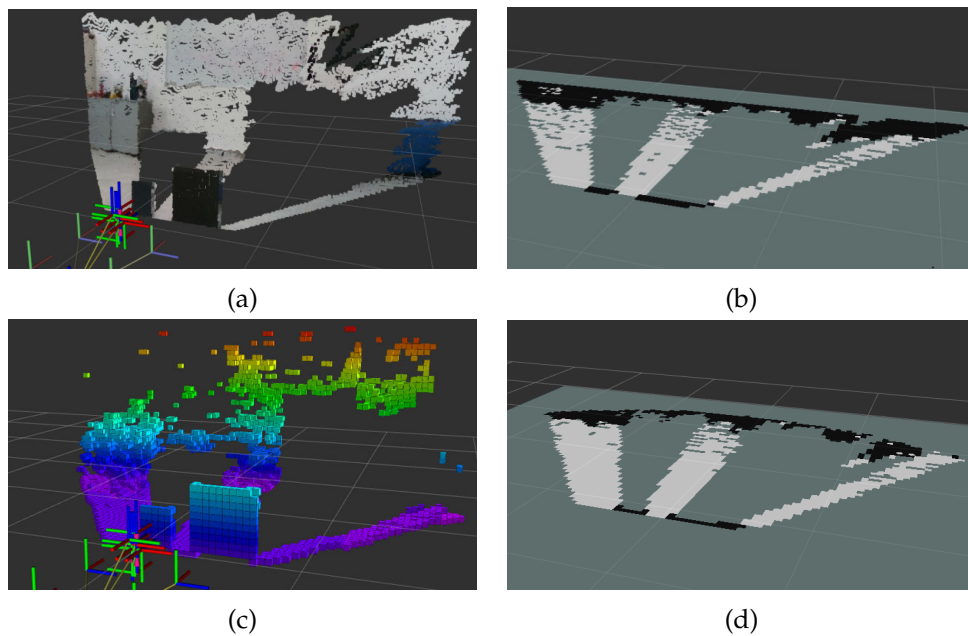


Figura 4.3: (a) Nuvem de pontos para o *setup* experimental realizado e (b) mapa de ocupação projetado pela nuvem de pontos. (c) OctoMap gerado para o *setup* experimental e (d) mapa de ocupação gerado com base no OctoMap.

Para o mapeamento do ambiente, foi empregado o RTAB-Map. Com o RTAB-Map, é possível gerar dois tipos de mapas de ocupação. O primeiro é um mapa de ocupação binário, onde cada ponto no espaço é identificado como ocupado ou livre com base nas leituras do sensor de profundidade da câmera RGB-D D435i. Esse mapa é projetado de forma projetiva, e um exemplo pode ser visualizado na Figura 4.3b.

O segundo mapa é gerado usando o OctoMap, onde cada célula armazena informações sobre a probabilidade de ocupação do respectivo *voxel* no espaço [109], utilizando um *voxel* de 5 cm. Esses dados de ocupação probabilística em três dimensões são usados para gerar o mapa de ocupação em duas dimensões. O OctoMap para o *setup* montado pode ser observado na Figura 4.3c.

O mapa de ocupação gerado pelo OctoMap (Figura 4.3d) é integrado ao *stack* de navegação do robô. Embora ambas as estratégias gerassem mapas 2D, os testes preliminares indicaram que o mapa derivado do OctoMap fornecia uma representação mais fiel e com menor ruído do ambiente em comparação com a projeção direta da nuvem de pontos. Esse mapa é utilizado como referência fixa do ambiente, enquanto a odometria visual gerada pela T265 é utilizada para estimar a posição do robô.

Os dados da nuvem de pontos são integrados ao *stack* de navegação por meio do pacote *depthimage-to-laserscan*. Foi incluído no *setup* experimental um obstáculo fora do campo de visão da câmera RGB-D D435i, com o objetivo de avaliar a conversão das informações da câmera RGB-D D435i para o formato de mensagem do tipo *sensor\_msgs/LaserScan*, a fim de ser integrado ao *stack* de navegação. Neste cenário, o primeiro obstáculo à frente e à esquerda possui uma altura de 30 cm, abaixo da altura do sensor LiDAR, posicionado a 35 cm do chão, sendo detectável apenas pela câmera RGB-D D435i (Figura 4.4).

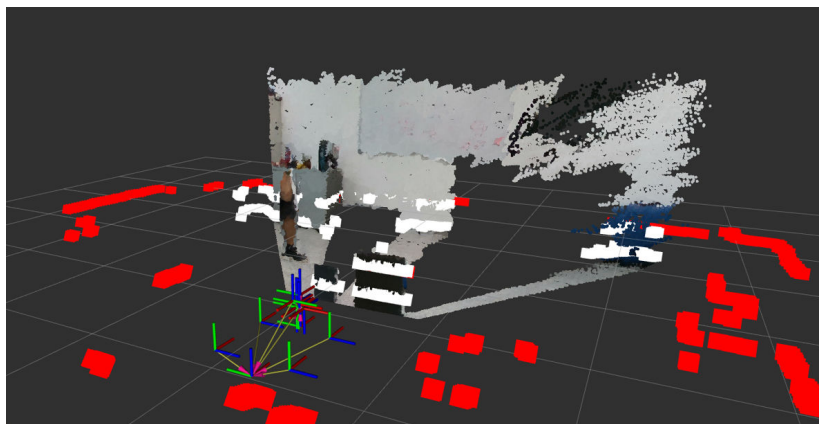


Figura 4.4: *Setup* experimental com os obstáculos detectados por sensores distintos.

As leituras dos obstáculos identificados pela câmera RGB-D D435i estão representadas em branco, enquanto as leituras do sensor LiDAR estão em vermelho. Os obstáculos detectados pela câmera RGB-D D435i e pelo sensor LiDAR contribuem para o mapa de custo final. Este mapa é continuamente atualizado e incorpora as informações dos obstáculos identificados pela câmera RGB-D D435i, os quais são marcados no mapa como regiões ocupadas no mapa de ocupação.

### 4.1.2 Protocolo Experimental

Para validar o sistema, foram realizados dois experimentos em um ambiente fechado não estruturado. O primeiro experimento tem por objetivo mapear um ambiente amplo para testar a robustez do sistema em relação à localização e a precisão do mapa tridimensional gerado, e o segundo tem por objetivo verificar a capacidade de navegação do sistema em questão, com a presença de obstáculos de diferentes alturas e posições.

No primeiro experimento, realizou-se o mapeamento de uma região ampla, contendo paredes de diferentes comprimentos. As áreas de interesse para o mapeamento, bem como os pontos utilizados pelo robô para a execução dessa tarefa, estão destacados na Figura 4.5.

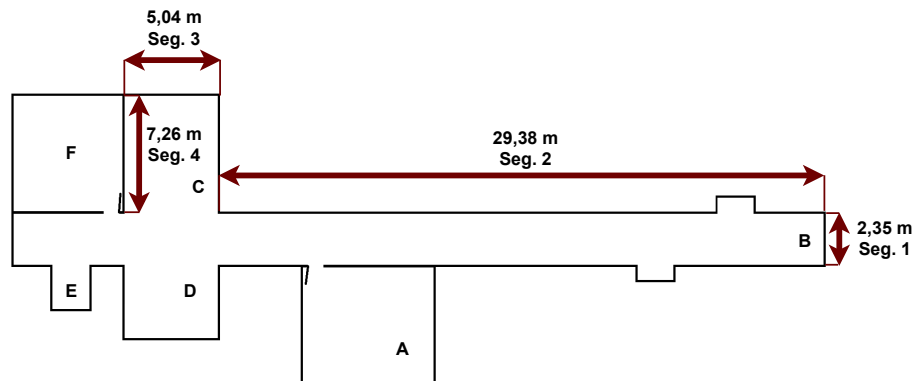


Figura 4.5: Região explorada e segmentos de interesse do mapa.

Para realizar o mapeamento, o robô partirá do ponto A e seguirá para os pontos B, C, D, E e F, sequencialmente, retornando ao ponto A após concluir o mapeamento. O robô executará uma rotina para realizar o VSLAM, gerando o mapa 3D e o mapa de ocupação correspondente utilizando o algoritmo RTAB-Map. A mesma rotina é executada com o algoritmo GMapping, gerando o mapa de ocupação do ambiente.

A partir do mapa gerado pelo algoritmo RTAB-Map e outro produzido pelo algoritmo GMapping, é realizada uma análise comparativa da escala de segmentos específicos do mapa, utilizando as dimensões reais como referência, conforme ilustrado na Figura 4.5, e calculado o erro em porcentagem. A medição dos segmentos nos mapas gerados foi realizada manualmente utilizando a ferramenta de medição do *software* RViz2, comparando os valores obtidos com as medidas reais do ambiente.

No segundo experimento, os obstáculos foram dispostos em locais estratégicos e com diferentes alturas, conforme mostrado na Figura 4.6. O posicionamento dos obstáculos tinha como objetivo formar um caminho estreito e único para se chegar a um ponto predeterminado. O robô recebeu um comando para navegar autonomamente em direção ao ponto desobstruído do ambiente, adaptando sua trajetória para contornar obstáculos de diferentes alturas.



ambiente previamente gerado é armazenado em um banco de dados. Desta forma, as aplicações podem variar desde o mapeamento de um ambiente desconhecido até a execução de rotinas a partir de um banco de dados previamente armazenado.

Por meio deste mapa, foi possível gerar o OctoMap do ambiente, conforme ilustrado na Figura 4.7b. É relevante destacar que, à medida que o robô se distanciava do ponto inicial, o sistema de odometria visual conseguiu corrigir com eficácia a estimativa de sua posição no ambiente, como evidenciado pela forma coesa da nuvem de pontos.

A partir do OctoMap, também é gerado um mapa de ocupação 2D (Figura 4.7c), o qual pode ser armazenado e inicializado em execuções futuras, atualizando-se com base nas leituras atuais. Essa capacidade permite que o planejador global mantenha um desempenho mais confiável, adaptando-se a ambientes não estruturados e atualizando dinamicamente o mapa conforme a movimentação dos obstáculos.

A partir dos mapas gerados pelo algoritmo RTAB-Map e por um mapa gerado pelo algoritmo GMapping, realizou-se uma análise comparativa da escala de alguns segmentos-chave do mapa (Tabela 4.1).

Tabela 4.1: Comparação dos tamanhos dos segmentos de mapa.

Seg.	Referência	GMapping		RTAB-Map	
	Real (m)	Medido (m)	Erro (%)	Medido (m)	Erro (%)
1	2,35	2,34	0,25	2,27	3,56
2	29,38	29,29	0,32	28,35	3,63
3	5,04	4,99	1,04	4,90	2,83
4	7,26	7,24	0,26	7,23	0,38

O GMapping apresenta erros menores na escala do ambiente, exibindo uma variação média de aproximadamente 0,73%, enquanto o RTAB-Map apresenta uma variação média de cerca de 3,34%. Essa disparidade pode ser atribuída à dificuldade do SLAM interno, realizado pela câmera de *tracking* T265, em estimar com precisão o deslocamento devido à ausência de características marcantes no ambiente. Em contrapartida, no Segmento 4, o erro da abordagem VSLAM (0,38%) foi comparável ao do GMapping (0,26%).

Tal efeito pode ser explicado ao observar que este segmento corresponde a uma área específica da região mapeada que possui características marcantes, como a presença de mesas e cadeiras. Neste segmento, o resultado da abordagem visual se aproxima

do obtido com o GMapping, que utiliza a odometria das rodas como base para suas estimativas, o que contribui para sua maior precisão.

Por ser um sistema que utiliza referências puramente visuais, o desempenho do sistema proposto está diretamente ligado à existência de características distintas no ambiente, assim como às condições de visibilidade. Ainda assim, o erro obtido mostra que a ausência de características marcantes não compromete o desempenho do sistema.

Para o teste de navegação, foi estabelecido um objetivo em um ponto desobstruído do ambiente, exigindo que o robô contornasse os obstáculos posicionados (Figura 4.6). A Figura 4.8 ilustra a execução bem-sucedida desta tarefa. Na Figura 4.8a, a pose inicial do robô pode ser vista no canto superior esquerdo do mapa (i), enquanto a seta verde representa o comando de posição (*goal*) enviado pela interface do RViz. Já a Figura 4.8b mostra a pose final do robô (ii), que navegou de forma autônoma e atingiu o ponto de destino com sucesso, demonstrando a viabilidade da integração entre o mapa gerado e os dados dos sensores em tempo real para uma tarefa de navegação.

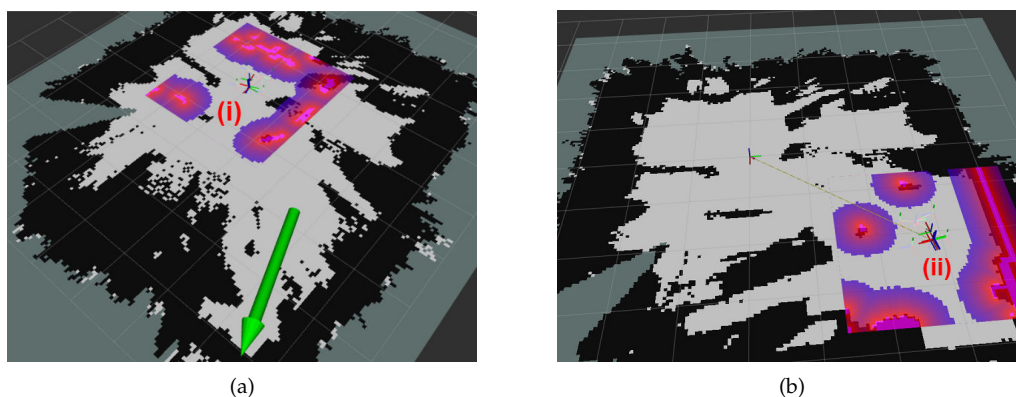


Figura 4.8: Execução da tarefa de navegação autônoma. (a) Definição do ponto de destino (seta verde) no mapa de custo. (b) Posição final do robô após atingir o objetivo com sucesso.

### 4.3 Conclusões

A integração de dados da câmera RGB-D D435i ao *stack* de navegação, utilizando a câmera de *tracking* T265 como fonte de odometria para o VSLAM, permitiu a incorporação de informações tridimensionais do ambiente. Isso foi alcançado pela

conversão dos dados de profundidade para o formato *laserscan*, o que facilita a integração com os algoritmos de navegação.

As informações provenientes das câmeras proporcionam um mapa de ocupação mais fiel à região efetivamente explorável pelo robô, uma vez que detectam obstáculos fora do alcance do LiDAR, o que viabiliza uma navegação autônoma mais confiável. Isso significa que a combinação de dados da câmera RGB-D D435i e do LiDAR pode fornecer uma representação mais completa do ambiente, permitindo que o robô tome decisões mais precisas e seguras durante a navegação.

Além disso, a escala da região mapeada não foi comprometida pela ausência de características marcantes no ambiente, obtendo resultados comparáveis aos do algoritmo GMapping quando o ambiente possui características visuais que auxiliam o processo de VSLAM. Embora o erro métrico da projeção 2D do VSLAM (3,34%) tenha sido superior ao do GMapping (0,73%), este resultado é considerado plenamente satisfatório para o objetivo final deste trabalho. Para uma tarefa de teleoperação assistida, onde o operador humano está sempre no controle, uma pequena imprecisão no mapa global é facilmente compensável.

A navegação teve um desempenho satisfatório com obstáculos estáticos, mas a configuração adotada neste estudo apresentou duas limitações principais. A primeira limitação, observada durante os testes de desenvolvimento do sistema, foi a dificuldade da configuração atual em lidar com obstáculos dinâmicos. Embora o experimento tenha sido conduzido apenas com obstáculos estáticos, notou-se que a persistência de obstáculos já removidos do ambiente no mapa de custos exigiria um ajuste fino dos parâmetros de navegação para garantir maior responsividade. A segunda foi a consequência de desativar o fechamento de laço do RTAB-Map: sem essa correção global, o sistema não era capaz de reconciliar *voxels* correspondentes ao visitar uma área, podendo levar a inconsistências no mapa.

As melhorias propostas para lidar com ambas limitações foram implementadas na configuração do sistema utilizada na fase seguinte deste trabalho, para avaliar a interface com usuários. Para resolver o problema da correspondência de *voxels* e aumentar a precisão global, o fechamento de laço será gerenciado pelo algoritmo RTAB-Map, desabilitando a função análoga da câmera T265. Adicionalmente, o algoritmo de mapeamento será realimentado com as leituras do LiDAR, conforme a arquitetura proposta, visando um mapa ainda mais robusto e fiel ao ambiente.

## Capítulo 5

# Avaliação da Interface de Teleoperação com Usuários

Com a validação do subsistema de mapeamento e navegação estabelecida no Capítulo 4, o foco desta segunda fase experimental se volta para a avaliação da interface de teleoperação multimodal com usuários finais. Este estudo foi projetado para investigar de forma sistemática o impacto das diferentes modalidades de visualização e controle na experiência e no desempenho do operador. O objetivo é responder aos objetivos de pesquisa centrais deste trabalho, quantificando os efeitos da imersão visual e do *feedback* háptico na teleoperação de um robô móvel em uma tarefa de navegação.

A literatura demonstra consistentemente que interfaces imersivas podem melhorar a eficiência e a consciência situacional em tarefas de teleoperação [30, 71]. Da mesma forma, o *feedback* háptico tem se mostrado eficaz em reduzir a carga de trabalho e melhorar o desempenho em tarefas que exigem precisão [26, 33]. No entanto, a combinação destas modalidades, especialmente no contexto da navegação de robôs móveis com assistência háptica, introduz um balanço complexo de *trade-off*. Interfaces imersivas podem demandar mais recursos cognitivos, e a eficácia do *feedback* háptico pode variar dependendo da tarefa e da experiência do usuário [28, 29].

Neste contexto, este estudo busca investigar os efeitos combinados e isolados dos ambientes de visualização não imersivo e imersivo, referenciados como 2D e RV, respectivamente, e das interfaces de controle (teclado vs. háptico) em uma tarefa de navegação teleoperada. O objetivo é verificar as seguintes hipóteses: (H1) O ambiente imersivo levará a um maior senso de presença do que o ambiente não imersivo; (H2) O controle háptico resultará em uma menor carga de trabalho percebida em

comparação com o controle por teclado; e (H3) A combinação do controle háptico e do ambiente imersivo (HP-RV) levará a um melhor desempenho geral, medido pelo tempo de tarefa e pelo número de colisões. A análise busca fornecer evidências quantitativas e qualitativas sobre a eficácia da abordagem multimodal proposta.

## 5.1 Metodologia Experimental

Para avaliar a interface proposta, foi conduzido um estudo com usuários em um ambiente de laboratório controlado. A metodologia foi projetada para comparar sistematicamente a interface imersiva e háptica com abordagens de teleoperação mais tradicionais.

### 5.1.1 Delineamento Experimental

O experimento seguiu um delineamento fatorial 2x2 com medidas repetidas (*within-subjects*). Os dois fatores intra-sujeitos foram a interface de controle, com dois níveis (teclado vs. háptico), e o ambiente de visualização, com dois níveis (ambiente não imersivo vs. ambiente imersivo).

Cada participante foi exposto a todas as quatro condições experimentais: (1) teclado + ambiente não imersivo (KB-2D); (2) háptico + ambiente não imersivo (HP-2D); (3) teclado + ambiente imersivo (KB-RV); e (4) háptico + ambiente imersivo (HP-RV). Para mitigar os efeitos de aprendizagem e de ordem, a sequência de apresentação das quatro condições foi contrabalanceada entre os participantes utilizando um delineamento em Quadrado Latino.

### 5.1.2 Métricas de Avaliação

As variáveis dependentes, ou métricas de avaliação, foram divididas em duas categorias: medidas objetivas de desempenho e segurança, e medidas subjetivas de percepção do participante. Na primeira categoria, as seguintes métricas objetivas foram registradas para avaliar a *performance* e a segurança dos participantes durante a tarefa de teleoperação.

A primeira delas foi o tempo de tarefa, medido como o intervalo total, em segundos, desde o início até a conclusão bem-sucedida de cada cenário experimental. Esta métrica serve como o principal indicador de eficiência operacional, sendo que um

tempo menor é interpretado como um desempenho superior, refletindo a facilidade e a rapidez com que o participante conseguiu manipular o sistema para completar o objetivo. A segunda métrica objetiva foi o número de colisões, que corresponde à contagem total de vezes que o efetuidor final do robô teleoperado fez contato não intencional com as paredes do ambiente real ou obstáculos. Esta variável é o principal indicador de segurança e precisão do controle, de modo que um número menor de colisões sugere um maior nível de controle espacial e uma operação mais segura do sistema.

Para obter uma compreensão mais profunda da experiência dos participantes com cada sistema, foram também coletadas as seguintes medidas de autorrelato. A primeira foi a carga de trabalho subjetiva, na qual, após cada condição experimental, os participantes avaliaram a carga de trabalho percebida utilizando o questionário NASA Task Load Index (NASA-TLX), em sua versão validada em português [110]. Esta ferramenta multidimensional captura seis subescalas (Demanda Mental, Física, Temporal, Esforço, Performance e Frustração), que são combinadas para gerar uma pontuação global de 0 a 100. A ferramenta é especialmente útil para avaliar as demandas cognitivas de tarefas que envolvem interação homem-máquina, como a teleoperação robótica, oferecendo *insights* sobre a carga de trabalho percebida pelos participantes e a complexidade da tarefa [111].

As seis subescalas que compõem o NASA-TLX foram avaliadas individualmente para fornecer uma análise detalhada da carga de trabalho. A Demanda Mental (DM) refere-se ao nível de atividade mental e perceptiva exigida (como pensar, decidir, calcular). A Demanda Física (DF) avalia a quantidade de atividade física necessária (como empurrar, puxar, controlar). A Demanda Temporal (DT) mede a pressão temporal sentida pelo participante em relação à taxa ou ritmo em que as tarefas ocorreram. O Nível de Esforço (ES) quantifica o quão arduamente o participante sentiu que precisou trabalhar (mental e fisicamente) para atingir seu nível de desempenho. A Percepção de Desempenho (DE) mede o quão bem-sucedido o participante se considerou na realização da tarefa. Por fim, o Nível de Frustração (FR) avalia o quão inseguro, desmotivado, irritado ou estressado o participante se sentiu durante a tarefa.

Adicionalmente, o senso de presença do usuário foi avaliado utilizando o *Presence Questionnaire* (PQ), um questionário que mede o senso de presença em ambientes

virtuais, validado em português [112]. Após cada condição, os participantes responderam a itens em uma escala Likert, que foram então convertidos para uma pontuação global de 0 a 100. O PQ tem como objetivo avaliar subescalas de Envolvimento, Fidelidade Sensorial, Imersão e Qualidade da Interface, e esta métrica é necessária para entender como cada combinação de ambiente de visualização e controle impactou a imersão e o conforto do participante, fatores que influenciam diretamente na aceitação do sistema.

### 5.1.3 Participantes

O tamanho da amostra para este estudo foi determinado por meio de uma análise de poder estatístico *a priori*, realizada com o *software* G\*Power (versão 3.1.9.7). O objetivo foi calcular o número mínimo de participantes necessário para detectar, com segurança, os efeitos de interesse. A análise foi configurada para uma ANOVA de medidas repetidas com fatores intra-sujeitos, refletindo o delineamento experimental 2x2 com um total de quatro medições por participante. Foi adotado um tamanho de efeito médio ( $f = 0,3$ ), conforme as diretrizes de Cohen, um poder estatístico desejado de 80% ( $1-\beta = 0,80$ ), e um nível de significância ( $\alpha$ ) de 0,05. Adicionalmente, assumiu-se uma correlação moderada de 0,5 entre as medidas repetidas, um valor conservador e comum na ausência de dados piloto, e a correção de esfericidade (epsilon) foi definida como 1, pois esta premissa é sempre perfeitamente satisfeita em delineamentos 2x2.

A análise indicou a necessidade de uma amostra mínima de 17 participantes. Com o intuito de garantir um poder estatístico ainda maior e prover uma margem de segurança contra eventuais perdas de dados, optou-se por recrutar um total de 20 participantes. Assim, participaram do estudo 20 voluntários (15 homens, 5 mulheres) com idade entre 19 e 40 anos ( $M = 24,4$ ;  $DP = 4,99$ ). Os critérios de inclusão buscaram uma amostra da população geral, recrutando voluntários que não possuíam experiência prévia ou treinamento especializado em sistemas de teleoperação robótica e interfaces hápticas para controle. A familiaridade cotidiana com dispositivos de entrada, como o teclado, foi considerada uma característica base da amostra, e seus efeitos foram normalizados por meio de uma sessão de familiarização antes de cada tarefa experimental.

### 5.1.4 Materiais e Ambiente Experimental

Todos os experimentos foram conduzidos utilizando a plataforma robótica móvel e a arquitetura de sistema integralmente detalhadas no Capítulo 3. O cenário para a interação foi um ambiente de laboratório, que se estendia por um percurso não linear de até 40 metros até o limite experimental. Neste espaço, um conjunto de obstáculos estáticos, incluindo mesas, cadeiras e cilindros de diferentes diâmetros. O arranjo desses objetos foi projetado para criar um percurso de navegação simples, onde a tarefa mais complexa seria o contorno dos cilindros, desafiando a percepção situacional.

Para investigar sistematicamente o impacto de cada modalidade sensorial, o estudo foi dividido em quatro condições experimentais distintas, cada uma isolando ou combinando as interfaces de visualização e controle.

A primeira condição, teclado e ambiente não imersivo (KB-2D), foi estabelecida como a linha de base do estudo, representando uma interface de teleoperação convencional. Nesta configuração, o participante visualizava o ambiente a partir de uma interface que exibia a visão egocêntrica da câmera do robô, como mostra na Figura 5.1. A interface utiliza a informação da câmera RGB descomprimida, publicada pelo ROS pelo pacote *web\_video\_server* que é acessada via HTTP.

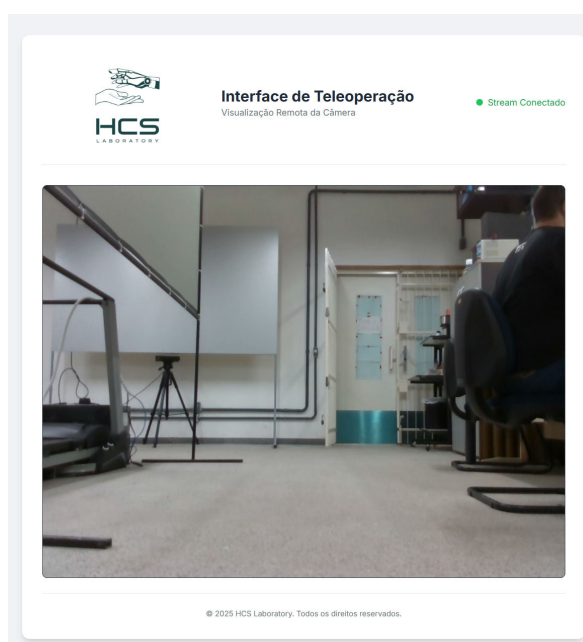


Figura 5.1: Interface não imersiva exibindo a vista egocêntrica capturada pela câmera D435i.

O controle da plataforma era realizado exclusivamente por meio das setas do teclado, onde as teclas direcionais para cima e para baixo comandavam a velocidade linear e as teclas para esquerda e direita controlavam a velocidade angular. Esta condição serviu como um ponto de referência para quantificar o desempenho e a carga de trabalho em um cenário que reflete as limitações clássicas de percepção de profundidade e controle não imersivo.

Para garantir uma comparação mais justa com o controle proporcional do dispositivo háptico e evitar movimentos abruptos, foi desenvolvido um nó em ROS 2 que converte os comandos discretos do operador em um fluxo contínuo e suave de velocidades para o robô. A interface captura eventos de múltiplas teclas pressionadas simultaneamente, permitindo movimentos combinados, como avançar enquanto gira. A principal característica deste controlador, no entanto, é a implementação de um filtro de suavização para o sinal de comando. Em vez de aplicar diretamente a velocidade alvo  $(v_d, \omega_d)$  correspondente às teclas pressionadas, o sistema interpola linearmente a velocidade atual do robô  $(v, \omega)$  em direção à velocidade alvo. Este método impõe uma taxa de variação máxima para as velocidades, resultando em uma aceleração e desaceleração controladas que previnem movimentos bruscos, buscando uma teleoperação intuitiva e responsiva.

A lógica de suavização pode ser descrita formalmente pelas seguintes equações de atualização em tempo discreto. A cada passo de tempo  $k$ , a velocidade para o próximo passo,  $k + 1$ , é calculada com base na velocidade atual e na velocidade desejada. Para o controle por teclado, a velocidade desejada é um valor máximo pré-configurado, acionado ao pressionar das teclas, que foi parametrizado para ser o mesmo valor máximo alcançável com o dispositivo háptico, garantindo a consistência entre as condições de controle.

A atualização da velocidade linear  $v$  é dada por:

$$v(k + 1) = v(k) + \text{sign}(v_d - v(k)) \cdot \min(|v_d - v(k)|, \alpha_v) \quad (5.1)$$

E a atualização da velocidade angular  $\omega$  é dada por:

$$\omega(k + 1) = \omega(k) + \text{sign}(\omega_d - \omega(k)) \cdot \min(|\omega_d - \omega(k)|, \alpha_\omega) \quad (5.2)$$

Nessas equações, os termos  $v(k)$  e  $\omega(k)$  representam as velocidades linear e angular atuais enviadas ao robô no instante de tempo discreto  $k$ , enquanto  $v_d$  e  $\omega_d$  são as respectivas velocidades desejadas (ou alvo), definidas pelas teclas pressionadas pelo operador. As constantes  $\alpha_v$  e  $\alpha_\omega$  determinam o incremento máximo de velocidade permitido a cada passo, funcionando efetivamente como um limitador de aceleração linear e angular. Por fim, a função  $\text{sign}(\cdot)$  é utilizada para definir a direção da mudança (aceleração ou desaceleração), e a função  $\text{min}(\cdot)$  garante que o incremento aplicado não ultrapasse a magnitude do erro, evitando que o valor alvo seja excedido.

Em seguida, a condição háptico e ambiente não imersivo (HP-2D) foi projetada para isolar o efeito da adição do *feedback* de força. A interface visual permanecia idêntica à da linha de base, com o operador observando a mesma interface não imersiva. Contudo, o teclado era substituído pelo dispositivo háptico Novint Falcon. Com ele, o participante controlava o robô de forma mais contínua e intuitiva, e recebia o *feedback* de força assistivo. Essa assistência, baseada na estratégia de proxêmica, manifestava-se como uma resistência no controle sempre que o robô se aproximava de um obstáculo, permitindo ao operador interagir com o ambiente remoto. O objetivo desta condição era avaliar o ganho de desempenho e a redução da carga cognitiva atribuíveis unicamente à modalidade háptica.

De forma análoga, a condição teclado e ambiente imersivo (KB-RV) foi criada para isolar o impacto da imersão visual. Nesta etapa, o participante utilizava o *headset* de realidade mista Oculus Quest 2, que o transportava para o ambiente virtual tridimensional, oferecendo uma percepção de profundidade e escala realistas. Apesar deste salto qualitativo na imersão visual, o método de controle foi mantido o mesmo da linha de base, sendo as setas do teclado. O propósito desta configuração era medir o quanto a visualização estereoscópica, por si só, poderia melhorar a consciência situacional e a eficiência da navegação, sem a influência de um dispositivo de controle avançado.

Finalmente, a condição háptico e ambiente imersivo (HP-RV) representou a implementação completa do sistema multimodal. Ela integra a visualização imersiva proporcionada pelo Oculus Quest 2 com o controle e *feedback* de força assistivo do Novint Falcon. Nesta configuração, o participante era capaz de não apenas ver o ambiente em 3D com percepção de profundidade, mas também de sentir as interações com os obstáculos por meio do controle háptico. Esta condição final foi projetada

para testar a hipótese de que a combinação de estímulos visual e tátil resulta na melhor *performance* de teleoperação, no maior senso de presença e na menor carga de trabalho para o operador.

### 5.1.5 Tarefa Experimental

A tarefa de teleoperação foi projetada para simular uma missão de navegação em um ambiente com obstáculos. O trajeto a ser seguido e o posicionamento dos obstáculos está representado na Figura 5.2. O participante era instruído a teleoperar o robô a partir de um ponto inicial A, navegar até o ponto B, onde deveria contornar o obstáculo por qualquer lado. A seguir, deveria ir para o ponto C, desviando dos obstáculos no caminho e contornar o obstáculo por qualquer lado. Por fim, deveria voltar até a posição inicial A, finalizando a tarefa ao tocar no quadrado azul. O *layout* dos obstáculos e dos *waypoints* foi mantido o mesmo para todas as condições e todos os participantes para garantir a comparabilidade dos resultados.

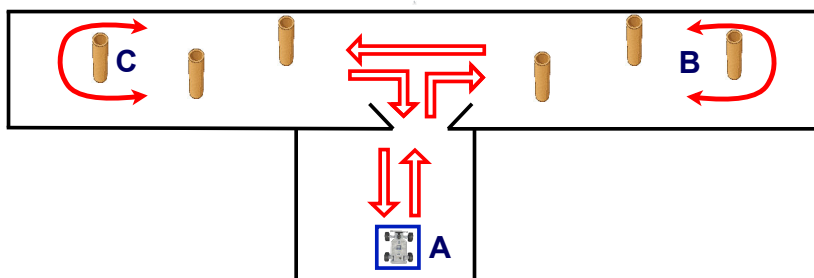


Figura 5.2: Trajeto a ser seguido na teleoperação e posicionamento dos obstáculos.

O posicionamento real do robô em relação ao teleoperador, bem como o posicionamento real dos obstáculos dos experimentos, podem ser observados na Figura 5.3.

### 5.1.6 Protocolo Experimental

O protocolo experimental foi desenhado para minimizar a fadiga dos participantes, conforme pode ser observado na Figura 5.4. Cada uma das quatro condições foi avaliada em sessões individuais com duração de aproximadamente 30 minutos, realizadas em dias distintos para cada voluntário, com um intervalo de pelo menos um dia entre elas. Ao chegar para a primeira sessão, o participante era recebido, recebia instruções detalhadas sobre os objetivos do estudo e os procedimentos que seriam seguidos.

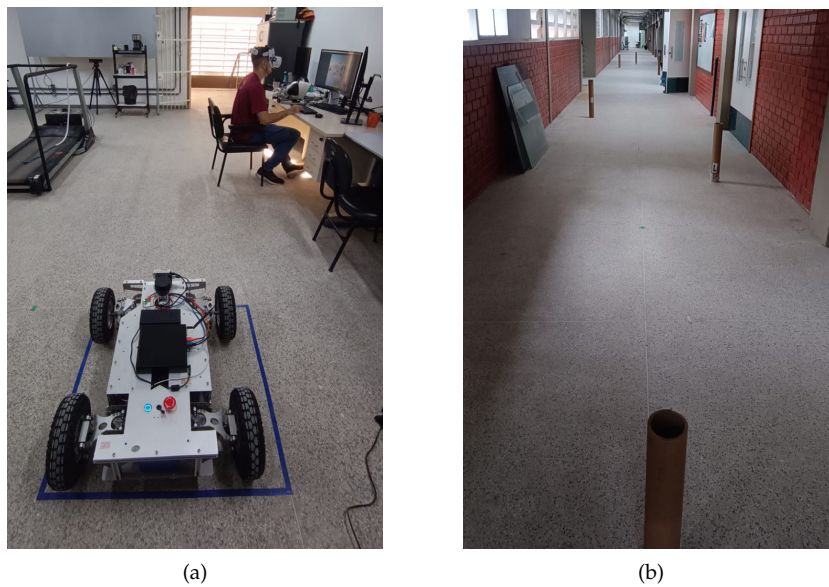


Figura 5.3: (a) Posição inicial do robô e posição em relação ao teleoperador e (b) posicionamento dos obstáculos no ambiente.

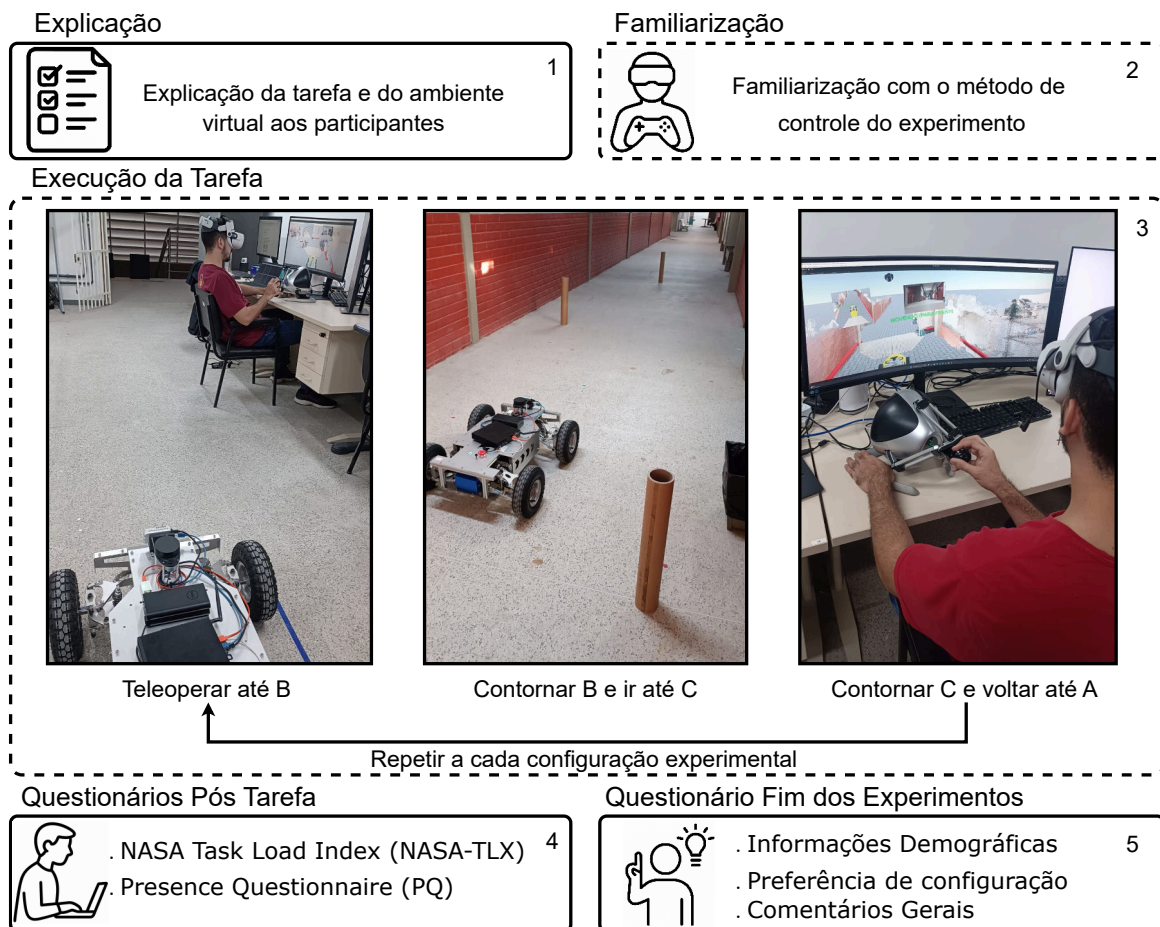


Figura 5.4: Protocolo experimental utilizado para a execução de tarefa de teleoperação. As etapas com linhas tracejadas foram feitas com auxílio do ambiente virtual.

O procedimento para cada uma das quatro condições era padronizado e composto por três fases sequenciais. A primeira fase era a de familiarização, na qual o participante passava por uma sessão de treinamento de cinco minutos. Durante este período, ele era encorajado a teleoperar o robô livremente, permitindo que se acostumasse com as particularidades do método de controle daquela condição específica.

Após o período de adaptação, iniciava-se a fase de execução da tarefa. O participante era posicionado na área experimental principal e instruído a navegar pelo percurso de obstáculos o mais rápido e seguro possível. Durante a execução, eram registradas as métricas objetivas de desempenho: o tempo total para completar a tarefa e o número de colisões com os obstáculos e o ambiente. Imediatamente após a conclusão do percurso, seguia-se para a fase de Coleta de Dados Subjetivos. O participante retirava o *headset* (quando aplicável) e respondia aos questionários NASA-TLX e *Presence Questionnaire* (PQ), cujas respostas eram referentes exclusivamente à experiência na condição que acabara de ser testada.

Este ciclo de familiarização, execução e avaliação era repetido integralmente para as quatro condições, com a ordem de apresentação sendo contrabalanceada entre os participantes por meio de um delineamento em Quadrado Latino para mitigar efeitos de aprendizagem. Ao término da quarta e última sessão, o participante respondia a um questionário final. Neste instrumento, ele indicava qual havia sido sua configuração preferida, fornecia uma justificativa para sua escolha e tinha um espaço para comentários qualitativos abertos sobre a experiência geral. Por fim, eram coletadas as informações demográficas para a caracterização do perfil da amostra.

## 5.2 Análise Estatística

Para avaliar o impacto das variáveis independentes, método de entrada (háptico, teclado) e ambiente de visualização (2D, RV) nas métricas de desempenho e percepção subjetiva, foi conduzida uma Análise de Variância (ANOVA) 2x2 de duas vias com medidas repetidas, adotada devido ao desenho experimental e aos efeitos de interesse a serem observados.

A adequação dos dados para a análise paramétrica foi avaliada através do teste de normalidade de *Shapiro-Wilk* ( $p > 0,05$ ) e do teste de homogeneidade de variâncias

de *Levene* ( $p > 0,05$ ). A verificação inicial revelou que a maioria das variáveis dependentes violava a premissa de normalidade. Um processo investigativo foi realizado, explorando múltiplas estratégias de pré-processamento, incluindo a remoção de *outliers* e diversas técnicas de transformação de dados (logarítmica e *Box-Cox*). Conseqüentemente, a transformação de *Box-Cox* foi selecionada como a abordagem principal para a análise, pois representou a tentativa eficaz de aproximar os dados a uma distribuição normal. Para garantir a consistência, um único valor de  $\lambda$  ótimo foi calculado para cada variável e aplicado uniformemente a todos os seus subgrupos, conforme detalhado na Tabela 5.1.<sup>1</sup>

Tabela 5.1: Valores ótimos de Lambda ( $\lambda$ ) da transformação de *Box-Cox* aplicados a cada variável dependente transformada.

Variável Dependente	Valor de Lambda ( $\lambda$ )
Colisoes	-0,8709
NASA_DE	0,3519
NASA_DF	0,1264
NASA_DM	0,5350
NASA_DT	0,3012
NASA_ES	0,4198
NASA_FR	0,1944
NASA_Media	0,5484
PQ_Adaptacao	4,2311
PQ_Envolvimento	4,2863
PQ_Fidelidade	2,2004
PQ_Media	3,8101
PQ_Qualidade	3,2087
Tempo	-2,4455

Após a transformação, as premissas foram reavaliadas. Como demonstrado na Tabela 5.2, o procedimento foi bem-sucedido em normalizar diversas condições. A premissa de esfericidade foi avaliada pelo teste de Mauchly, com a correção de Greenhouse-Geisser sendo aplicada nos casos de violação. Para as interações significativas, foram realizados testes *post-hoc* de comparações par a par com a correção de *Holm*. Para as variáveis (Colisoes, NASA\_DE, NASA\_FR) cujas premissas não foram totalmente satisfeitas, foi empregada uma ANOVA não paramétrica (ARTool) para garantir a robustez da análise.

<sup>1</sup>As variáveis com prefixo NASA\_ referem-se às subescalas do questionário NASA-TLX (Carga de Trabalho) e as com prefixo PQ\_ às subescalas do Questionário de Presença (PQ), ambas definidas na Sessão 5.1.2.

Tabela 5.2: Condições que não apresentavam distribuição normal (teste de Shapiro-Wilk,  $p < 0,05$ ) e que passaram a apresentar após a transformação de *Box-Cox*.

Variável	Condição	p-valor (Antes)	p-valor (Depois)
NASA_DE	HP_RV	0,0020	0,2188
NASA_DF	HP_2D	0,0036	0,3246
NASA_DM	KB_RV	0,0101	0,1852
NASA_DT	HP_2D	0,0060	0,2485
NASA_DT	KB_2D	0,0011	0,4406
NASA_DT	KB_RV	0,0106	0,1399
NASA_ES	HP_RV	0,0238	0,0693
NASA_ES	KB_RV	0,0349	0,2647
NASA_FR	HP_2D	0,0210	0,1102
PQ_Adaptacao	KB_2D	0,0219	0,1230
PQ_Envolvimento	HP_RV	0,0303	0,2796
PQ_Qualidade	KB_2D	0,0303	0,2237
Tempo	HP_2D	0,0007	0,3370
Tempo	HP_RV	0,0296	0,5118

### 5.3 Resultados

Nesta seção, são apresentados os resultados da análise estatística. A análise foca em identificar os efeitos principais de cada fator (método de entrada e ambiente de visualização), bem como o efeito de interação entre eles. As métricas abordadas são: Tempo, Colisões, Carga de Trabalho Subjetiva (NASA-TLX) e Percepção de Presença (PQ). A Tabela 5.3 apresenta o resultado encontrado das ANOVAS para as variáveis não paramétricas, enquanto a Tabela 5.4 apresenta as ANOVAS encontradas para as variáveis paramétricas. Já as Tabelas 5.5 e 5.6 detalham o *post-hoc* para as variáveis não paramétricas e paramétricas, respectivamente.

Tabela 5.3: Resultados da Análise de Variância Não Paramétrica (ART ANOVA).

Métrica	Efeito do Controle	Efeito de Visualização	Efeito de Interação
Colisões	$F(1; 57) = 48,42; \mathbf{p} < 0,001$	$F(1; 57) = 134,83; \mathbf{p} < 0,001$	$F(1; 57) = 31,70; \mathbf{p} < 0,001$
NASA_DE	$F(1; 57) = 0,36; p = 0,549$	$F(1; 57) = 26,92; \mathbf{p} < 0,001$	$F(1; 57) = 6,72; \mathbf{p} = 0,012$
NASA_FR	$F(1; 57) = 0,10; p = 0,755$	$F(1; 57) = 15,97; \mathbf{p} < 0,001$	$F(1; 57) = 0,12; p = 0,728$

Tabela 5.4: Resultados da Análise de Variância Paramétrica (ANOVA de duas vias pós *Box-Cox*).

Métrica	Efeito do Controle			Efeito de Visualização			Efeito de Interação		
	$F(1;19)$	$p$	$\eta_g^2$	$F(1;19)$	$p$	$\eta_g^2$	$F(1;19)$	$p$	$\eta_g^2$
<i>Métrica de Desempenho</i>									
Tempo	77,40	<0,001	0,467	2,74	0,115	0,027	0,99	0,332	0,008
<i>Métricas de Carga de Trabalho (NASA-TLX)</i>									
NASA Média	5,15	<b>0,035</b>	0,037	26,00	<0,001	0,158	7,40	<b>0,014</b>	0,034
NASA_DF	7,30	<b>0,014</b>	0,066	0,82	0,375	0,004	5,56	<b>0,029</b>	0,035
NASA_DM	3,92	0,062	0,023	10,53	<b>0,004</b>	0,060	4,69	<b>0,043</b>	0,024
NASA_DT	0,36	0,556	0,003	0,80	0,383	0,006	1,82	0,193	0,008
NASA_ES	2,16	0,158	0,020	10,07	<b>0,005</b>	0,076	1,46	0,242	0,009
<i>Métricas de Percepção de Presença (PQ)</i>									
PQ Média	1,60	0,221	0,010	29,03	<0,001	0,346	3,01	0,099	0,009
PQ_Adaptacao	10,24	<b>0,005</b>	0,070	36,46	<0,001	0,322	0,39	0,541	0,001
PQ_Envolvimento	0,99	0,333	0,005	22,72	<0,001	0,288	0,90	0,353	0,002
PQ_Fidelidade	2,02	0,171	0,017	34,45	<0,001	0,342	5,12	<b>0,036</b>	0,026
PQ_Qualidade	1,82	0,194	0,020	20,06	<0,001	0,269	4,11	0,057	0,017

Tabela 5.5: Resultados detalhados das comparações *post-hoc* (ajuste de Tukey) para as variáveis com interação significativa.

Métrica	Comparação de Contraste	t-valor	p-valor
Colisões	HP-2D vs. HP-RV	4,60	<b>0,0001</b>
	HP-2D vs. KB-2D	-5,46	<0,0001
	HP-2D vs. KB-RV	3,44	<b>0,006</b>
	HP-RV vs. KB-2D	-10,05	<0,0001
	HP-RV vs. KB-RV	-1,15	0,659
	KB-2D vs. KB-RV	8,90	<0,0001
NASA_DE (Desempenho)	HP-2D vs. HP-RV	1,93	0,228
	HP-2D vs. KB-2D	-1,34	0,540
	HP-2D vs. KB-RV	4,30	<b>0,0004</b>
	HP-RV vs. KB-2D	-3,27	<b>0,010</b>
	HP-RV vs. KB-RV	2,38	0,094
	KB-2D vs. KB-RV	5,65	<0,0001

Tabela 5.6: Resultados detalhados das comparações *post-hoc* (ajuste de Holm) para as variáveis paramétricas com interação significativa.

Métrica	Comparação de Contraste	t-valor	p-valor
NASA Média	HP-2D vs. HP-RV	2,29	<b>0,034</b>
	KB-2D vs. KB-RV	5,62	<b>&lt;0,001</b>
NASA_DF (Física)	HP-2D vs. HP-RV	2,06	0,107
	KB-2D vs. KB-RV	-1,51	0,148
NASA_DM (Mental)	HP-2D vs. HP-RV	0,97	0,346
	KB-2D vs. KB-RV	3,59	<b>0,004</b>
PQ_Fidelidade	HP-2D vs. HP-RV	-3,74	<b>0,001</b>
	KB-2D vs. KB-RV	-6,55	<b>&lt;0,001</b>

### 5.3.1 Colisões

Em relação ao número de colisões, a condição KB-2D apresentou a maior média de erros ( $M = 3,00$ ;  $DP = 1,52$ ), enquanto a condição HP-RV apresentou a menor ( $M = 0,10$ ;  $DP = 0,31$ ). A análise estatística revelou um efeito de interação significativo entre o método de entrada e o ambiente de visualização ( $F(1; 57) = 31,70$ ;  $p < 0,001$ ). Testes *post-hoc* detalharam esta interação, revelando que a mudança do ambiente não imersivo para o ambiente imersivo reduziu significativamente o número de colisões para ambos os métodos de controle.

No entanto, essa redução foi mais acentuada para o controle por teclado, que viu uma diminuição de 2,7 colisões em média, em comparação com a redução de 1,0 colisão para o controle háptico. Adicionalmente, foram encontrados efeitos principais significativos para o método de entrada ( $F(1; 57) = 48,42$ ;  $p < 0,001$ ) e para o ambiente de visualização ( $F(1; 57) = 134,83$ ;  $p < 0,001$ ), indicando que, de forma geral, o controle háptico foi superior ao teclado e o ambiente imersivo foi superior ao ambiente não imersivo na prevenção de colisões. A distribuição de colisões no experimento pode ser observada na Figura 5.5.

A análise *post-hoc* (com ajuste de Tukey) permite uma interpretação mais profunda deste efeito de interação. Ao comparar os métodos de controle dentro de cada ambiente de visualização, observa-se que no ambiente não imersivo, o controle háptico foi significativamente superior ao teclado, resultando em menos colisões ( $p < 0,0001$ ). No entanto, este cenário muda drasticamente na realidade virtual.

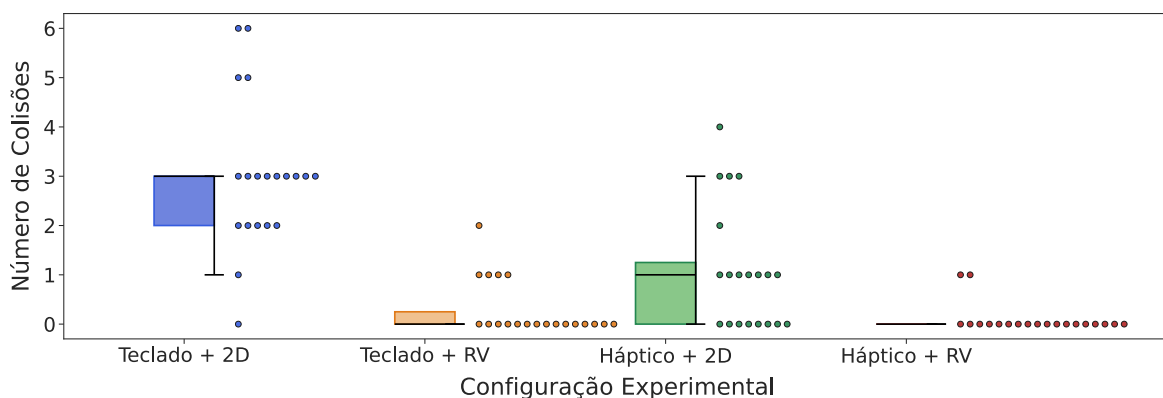


Figura 5.5: Box plot e número das colisões durante a teleoperação.

Dentro do ambiente imersivo, a vantagem do controle háptico desaparece, e não há mais uma diferença estatisticamente significativa no número de colisões entre o controle háptico e o teclado ( $p = 0,659$ ). Este resultado sugere que a maior consciência situacional proporcionada pelo ambiente imersivo foi tão eficaz em mitigar os erros de navegação que conseguiu nivelar a *performance* entre os dois métodos de controle, compensando as dificuldades inerentes ao controle por teclado.

### 5.3.2 Tempo de Tarefa

A ANOVA para o tempo de tarefa não revelou um efeito de interação significativo entre o controle e método de visualização ( $F(1; 19) = 0,99; p = 0,332$ ). Adicionalmente, o efeito principal do ambiente de visualização também não foi significativo ( $F(1; 19) = 2,74; p = 0,115$ ), indicando que não houve uma diferença clara no tempo de conclusão entre o uso do ambiente não imersivo e do ambiente imersivo.

No entanto, a análise revelou um efeito principal altamente significativo para o método de entrada ( $F(1; 19) = 77,40; p < 0,001$ ). Este resultado indica que a escolha do dispositivo de controle teve um impacto substancial no tempo que os participantes levaram para completar a tarefa, obtendo menores tempos de tarefa com o uso do teclado, conforme pode ser observado na Figura 5.6.

### 5.3.3 Carga de Trabalho Subjetiva (NASA-TLX)

A carga de trabalho geral, medida pela média do NASA-TLX e exibida na Figura 5.7, apresentou seu valor mais alto na condição HP-2D ( $M = 35,67; DP = 20,01$ ) e o mais baixo na condição KB-RV ( $M = 15,04; DP = 10,02$ ). A ANOVA revelou

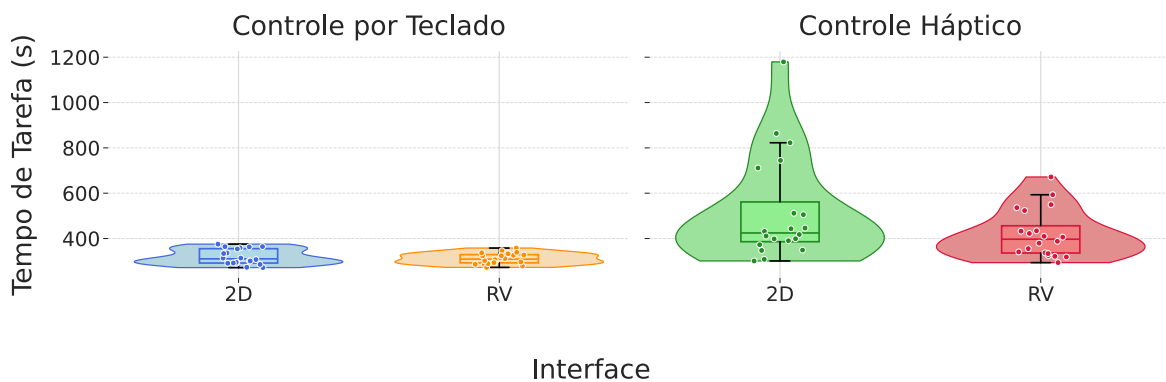


Figura 5.6: Box plot e violino do tempo para cumprir a tarefa.

um efeito de interação significativo entre o método de entrada e o ambiente de visualização ( $F(1; 19) = 7,40; p = 0,014$ ), indicando que o impacto do ambiente de visualização na carga de trabalho foi diferente para cada método de controle. Especificamente, embora a mudança do ambiente não imersivo para o ambiente imersivo tenha reduzido a carga de trabalho para ambos os controles, essa redução foi substancialmente mais acentuada para o controle por teclado do que para o controle háptico. Adicionalmente, foram encontrados efeitos principais significativos para o método de entrada ( $F(1; 19) = 5,15; p = 0,035$ ) e para o ambiente de visualização ( $F(1; 19) = 26,00; p < 0,001$ ).

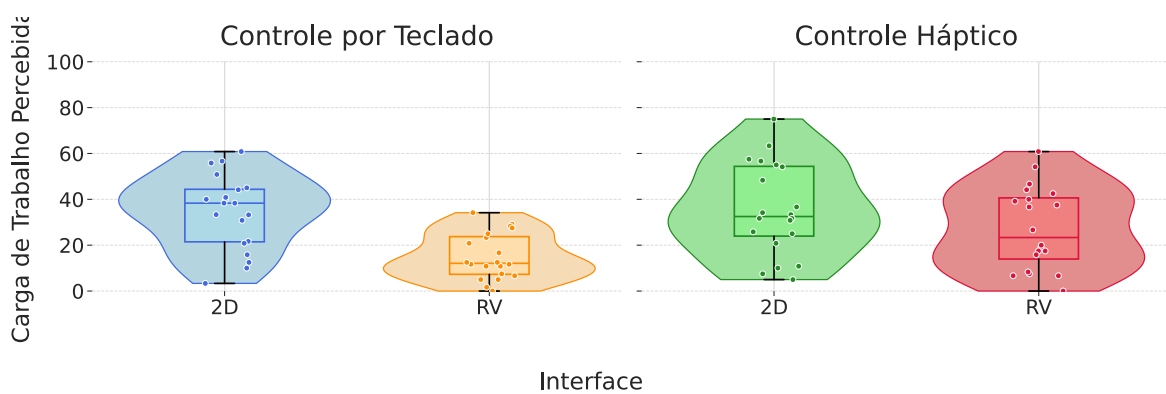


Figura 5.7: Box plot e violino da média da carga de trabalho percebida utilizando o NASA-TLX.

A análise *post-hoc* com correção de Holm permitiu detalhar a natureza da interação significativa encontrada. As comparações revelaram que a mudança do ambiente

não imersivo para o ambiente imersivo (RV) resultou em uma redução estatisticamente significativa na carga de trabalho percebida para ambos os métodos de controle. No entanto, a magnitude desse benefício foi diferente: para o controle háptico, a redução foi modesta ( $p = 0,034$ ), enquanto para o controle por teclado, a redução foi drasticamente maior e mais significativa ( $p < 0,001$ ). Este resultado comprova que o benefício do ambiente imersivo na diminuição da carga de trabalho foi desproporcionalmente maior para o controle por teclado.

Para aprofundar a compreensão sobre a origem da carga de trabalho, as seis subescalas do questionário NASA-TLX, definidas na Seção 5.1.2, foram analisadas individualmente. Esta análise detalhada permite identificar quais dimensões específicas mais contribuíram para os efeitos observados na média geral. Um resumo comparativo das pontuações para todas as subescalas é apresentado na Figura 5.8. A seguir, os resultados estatísticos para cada subescala são discutidos em detalhe.

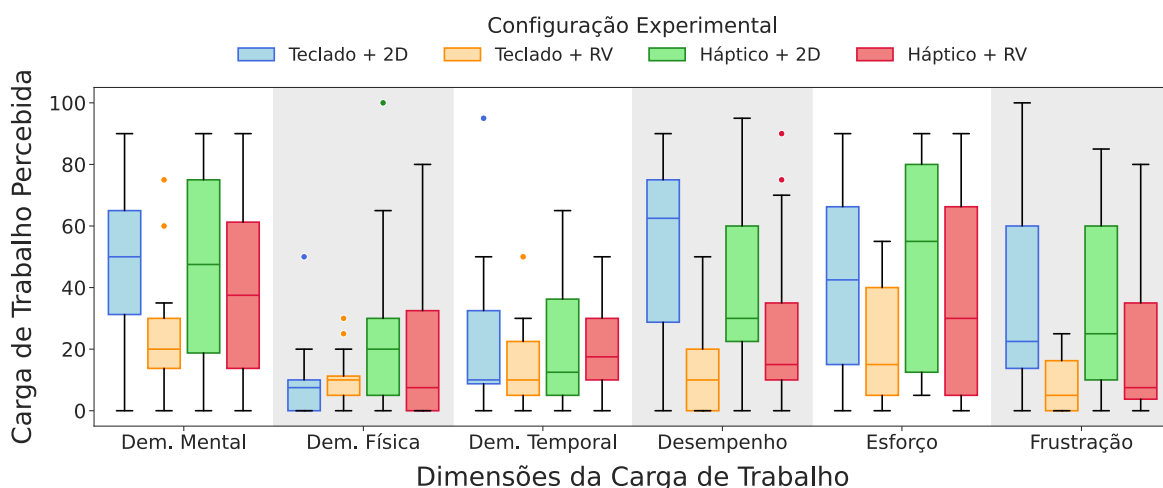


Figura 5.8: Comparação das médias das seis subescalas do NASA-TLX entre as quatro condições experimentais.

### 5.3.3.1 Demanda Mental (NASA\_DM)

A demanda mental percebida, medida pela subescala DM do NASA-TLX, apresentou níveis similares e altos nas condições com o ambiente não imersivo ( $M$  entre 46,0 e 46,25), mas diminuiu acentuadamente no ambiente imersivo, com a condição KB-RV ( $M = 22,75$ ;  $DP = 18,67$ ) registrando o valor mais baixo. A análise de variância confirmou esta observação com um efeito de interação significativo entre o método de entrada e o ambiente de visualização ( $F(1; 19) = 4,69$ ;  $p = 0,043$ ).

Comparações *post-hoc* revelaram que, para o controle por teclado, a mudança do ambiente não imersivo para o ambiente imersivo resultou em uma redução estatisticamente significativa na demanda mental ( $p = 0,004$ ). Em contrapartida, para o controle háptico, a leve redução observada entre o ambiente não imersivo ( $M = 46,00$ ) e o ambiente imersivo ( $M = 40,25$ ) não foi estatisticamente significativa ( $p = 0,346$ ). Isso sugere que o benefício do ambiente imersivo na redução da carga mental foi exclusivo à condição de controle por teclado.

### 5.3.3.2 Demanda Física (NASA\_DF)

A demanda física percebida pelos participantes foi notavelmente mais alta ao utilizar o controle háptico ( $M = 22,63$ ;  $DP = 25,01$ ) em comparação com o teclado ( $M = 9,13$ ;  $DP = 9,88$ ). A ANOVA confirmou esta observação com um efeito principal significativo para o método de entrada ( $F(1; 19) = 7,30$ ;  $p = 0,014$ ). Adicionalmente, foi encontrado um efeito de interação significativo entre o método de entrada e o ambiente de visualização ( $F(1; 19) = 5,56$ ;  $p = 0,029$ ).

A interação sugere que o impacto do ambiente de visualização na demanda física foi diferente para cada controle. Para o controle háptico, a demanda física foi percebida como menor no ambiente imersivo ( $M = 19,50$ ) do que no ambiente não imersivo ( $M = 25,75$ ). Em contraste, para o controle por teclado, a tendência se inverteu sutilmente, com a demanda física sendo ligeiramente maior no ambiente imersivo ( $M = 9,75$ ) do que no ambiente não imersivo ( $M = 8,50$ ). Não foi encontrado um efeito principal significativo para o ambiente de visualização ( $F(1; 19) = 0,82$ ;  $p = 0,375$ ).

As comparações específicas de *post-hoc* não atingiram o limiar de significância estatística. A diferença entre o ambiente não imersivo e o ambiente imersivo para o controle háptico ( $p = 0,107$ ) e para o controle por teclado ( $p = 0,148$ ) não foi estatisticamente significativa após a correção para múltiplas comparações. Este resultado sugere que, embora o padrão geral dos dados indique uma interação real (conforme detectado pela ANOVA), o efeito é de magnitude modesta e não é impulsionado por uma única diferença forte entre um par de condições, mas sim pela mudança sutil na tendência entre os dois métodos de controle.

### 5.3.3.3 Demanda Temporal (NASA\_DT)

A demanda temporal percebida pelos participantes, que avalia a pressão do tempo sentida durante a tarefa, apresentou valores médios relativamente consistentes entre as quatro condições experimentais, variando entre  $M = 16,00$  ( $DP = 15,36$ ) na condição KB-RV e  $M = 22,75$  ( $DP = 23,81$ ) na condição KB-2D. A ANOVA não revelou um efeito de interação significativo entre o método de entrada e o ambiente de visualização ( $F(1; 19) = 1,82; p = 0,193$ ). Adicionalmente, não foram encontrados efeitos principais significativos nem para o método de entrada ( $F(1; 19) = 0,36; p = 0,556$ ), nem para o ambiente de visualização ( $F(1; 19) = 0,80; p = 0,383$ ). Estes resultados sugerem que nenhuma das interfaces testadas alterou significativamente a percepção de pressão temporal dos operadores durante a tarefa de navegação.

### 5.3.3.4 Percepção de Desempenho (NASA\_DE)

A percepção de desempenho autoavaliada (NASA\_DE), onde valores menores indicam melhor desempenho, foi mais impactada pela combinação das interfaces, com a condição KB-2D sendo avaliada como a de pior desempenho ( $M = 51,00; DP = 30,11$ ) e a KB-RV como a melhor ( $M = 12,25; DP = 14,64$ ). A análise de variância não paramétrica (ART ANOVA) revelou um efeito de interação significativo entre o método de entrada e o ambiente de visualização ( $F(1; 57) = 6,72; p = 0,012$ ).

As comparações *post-hoc* confirmaram a natureza desta interação. A melhora na percepção de desempenho ao mudar do ambiente não imersivo para o ambiente imersivo foi estatisticamente significativa para o controle por teclado ( $p < 0,001$ ), mas não foi significativa para o controle háptico ( $p = 0,228$ ). Este resultado demonstra que o benefício da imersão da RV na melhora da percepção de desempenho foi impulsionado primariamente pela sua combinação com o controle por teclado, que se tornou a condição mais bem avaliada.

### 5.3.3.5 Nível de Esforço (NASA\_ES)

A análise do nível de esforço percebido não indicou um efeito de interação significativo entre o método de entrada e o ambiente de visualização ( $F(1; 19) = 1,46; p = 0,242$ ). Também não foi encontrado um efeito principal significativo para o método de entrada ( $F(1; 19) = 2,16; p = 0,158$ ). No entanto, foi observado um efeito principal significativo para o ambiente de visualização ( $F(1; 19) = 10,07; p = 0,005$ ).

Independentemente do método de controle, os participantes aparentam se esforçar significativamente menos ao usar o ambiente imersivo ( $M = 29,13$ ) em comparação com o ambiente não imersivo ( $M = 45,38$ ).

### 5.3.3.6 Nível de Frustração (NASA\_FR)

A análise do nível de frustração dos participantes não indicou um efeito de interação significativo entre o método de entrada e o ambiente de visualização ( $F(1; 57) = 0,12; p = 0,728$ ). Também não foi encontrado um efeito principal significativo para o método de entrada ( $F(1; 57) = 0,10; p = 0,755$ ). No entanto, foi observado um efeito principal significativo para o ambiente de visualização ( $F(1; 57) = 15,97; p < 0,001$ ). Independentemente do método de controle utilizado, os participantes relataram um nível de frustração significativamente menor ao usar o ambiente imersivo ( $M = 14,88$ ) em comparação com o ambiente não imersivo ( $M = 35,50$ ).

### 5.3.4 Percepção de Presença (Questionário PQ)

O senso de presença, exibido na Figura 5.9, apresentou seu valor mais alto na condição KB-RV ( $M = 0,84; DP = 0,10$ ) e seu valor mais baixo foi observado em ambas as condições com o ambiente não imersivo (HP-2D:  $M = 0,61; DP = 0,17$ ; KB-2D:  $M = 0,61; DP = 0,21$ ). A ANOVA revelou um efeito principal significativo para o ambiente de visualização ( $F(1; 19) = 29,03; p < 0,001$ ). Em contrapartida, não foi encontrado um efeito principal significativo para o método de entrada ( $F(1; 19) = 1,60; p = 0,221$ ), nem um efeito de interação entre o método de entrada e o ambiente de visualização ( $F(1; 19) = 3,01; p = 0,099$ ). Este resultado indica que a mudança do ambiente não imersivo para o ambiente imersivo aumentou significativamente o senso de presença, independentemente do método de controle utilizado.

Para aprofundar a compreensão sobre o senso de presença, as quatro subescalas do questionário PQ, Envolvimento, Fidelidade Sensorial, Adaptação e Qualidade, foram analisadas individualmente. Esta análise detalhada permite identificar quais dimensões específicas mais contribuíram para os efeitos observados na média geral do senso de presença. Um resumo comparativo das pontuações para todas as subescalas nas quatro condições experimentais é apresentado na Figura 5.10. A seguir, os resultados estatísticos para cada subescala são discutidos em detalhe.

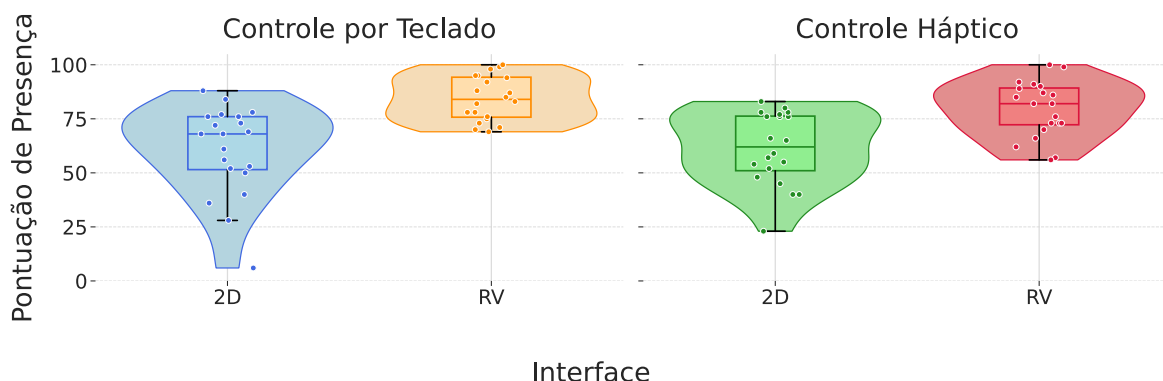


Figura 5.9: Box plot e violino da média do senso de presença obtido pelo questionário PQ.

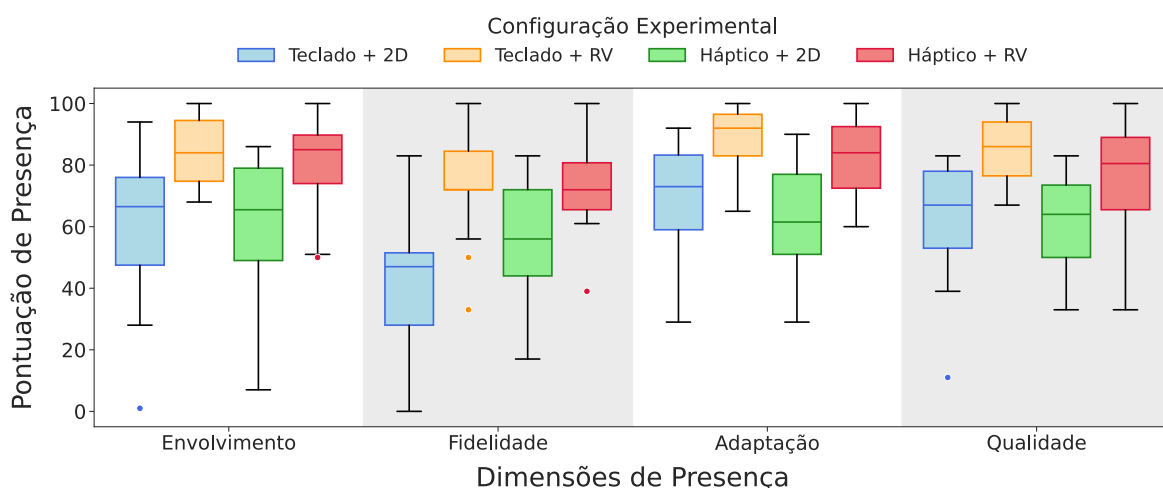


Figura 5.10: Comparação das médias das seis subescalas do senso de presença entre as quatro condições experimentais.

### 5.3.4.1 Envolvimento (PQ\_Envolvimento)

Os resultados para o envolvimento indicaram diferenças significativas em função do ambiente de visualização, mas não em relação ao método de entrada. Na condição háptica, a média aumentou de  $M = 0,61$  ( $DP = 0,21$ ) no ambiente não imersivo para  $M = 0,81$  ( $DP = 0,15$ ) no ambiente imersivo. De forma semelhante, com o controle por teclado, os valores cresceram de  $M = 0,62$  ( $DP = 0,23$ ) no ambiente não imersivo para  $M = 0,84$  ( $DP = 0,11$ ) no ambiente imersivo.

A ANOVA confirmou um efeito principal significativo para o ambiente de visualização ( $F(1; 19) = 22,72; p < 0,001; \eta_g^2 = 0,288$ ), mas não para o método de entrada ( $F(1; 19) = 0,99; p = 0,333$ ). A interação entre os fatores também não foi significativa ( $F(1; 19) = 0,90; p = 0,353$ ). Esses achados sugerem que o ambiente imersivo

promoveu maior envolvimento dos participantes em comparação com o ambiente não imersivo, independentemente do método de controle utilizado.

#### 5.3.4.2 Adaptação (PQ\_Adaptacao)

A adaptação percebida ao ambiente virtual apresentou valores mais altos nas condições com ambiente imersivo em comparação com o ambiente não imersivo. Na condição háptica, a pontuação média aumentou de  $M = 0,62$  ( $DP = 0,17$ ) no ambiente não imersivo para  $M = 0,82$  ( $DP = 0,13$ ) no ambiente imersivo. Um padrão semelhante foi observado no controle por teclado, em que os valores cresceram de  $M = 0,69$  ( $DP = 0,21$ ) no ambiente não imersivo para  $M = 0,90$  ( $DP = 0,09$ ) no ambiente imersivo.

A análise de variância confirmou esses resultados, revelando efeitos significativos tanto para o método de entrada ( $F(1; 19) = 10,24; p = 0,005$ ) quanto para o ambiente de visualização ( $F(1; 19) = 36,46; p < 0,001$ ). Entretanto, a interação entre os fatores não foi significativa ( $F(1; 19) = 0,39; p = 0,541$ ). Este resultado indica que o ambiente imersivo contribuiu para uma maior adaptação ao ambiente em comparação com o ambiente não imersivo, de forma consistente para ambos os métodos de controle.

#### 5.3.4.3 Fidelidade Sensorial (PQ\_Fidelidade)

A percepção de fidelidade do ambiente virtual foi consistentemente mais alta nas condições com ambiente imersivo em comparação com o ambiente não imersivo. Na condição háptica, a pontuação média aumentou de  $M = 0,55$  ( $DP = 0,20$ ) no ambiente não imersivo para  $M = 0,74$  ( $DP = 0,15$ ) no ambiente imersivo. Da mesma forma, para o controle por teclado, os valores cresceram de  $M = 0,43$  ( $DP = 0,21$ ) no ambiente não imersivo para  $M = 0,75$  ( $DP = 0,17$ ) no ambiente imersivo. A análise de variância indicou um efeito significativo para o ambiente de visualização ( $F(1; 19) = 34,45; p < 0,001$ ) e uma interação significativa entre o método de entrada e o ambiente de visualização ( $F(1; 19) = 5,12; p = 0,036$ ). O efeito principal do método de entrada, entretanto, não foi significativo ( $F(1; 19) = 2,02; p = 0,171$ ).

As comparações *post-hoc* revelaram que a mudança do ambiente não imersivo para o ambiente imersivo resultou em aumentos estatisticamente significativos na fidelidade percebida tanto para o controle háptico ( $p = 0,001$ ) quanto para o controle por teclado ( $p < 0,001$ ). A interação detectada pela ANOVA sugere que este aumento

na fidelidade, embora significativo em ambos os casos, foi de maior magnitude para o controle por teclado.

#### 5.3.4.4 Qualidade da Interface (PQ\_Qualidade)

A percepção de qualidade da interface, medida pela subescala do PQ, seguiu um padrão similar ao do envolvimento. Na condição háptica, a média aumentou de  $M = 0,61$  ( $DP = 0,21$ ) no ambiente não imersivo para  $M = 0,81$  ( $DP = 0,15$ ) no ambiente imersivo. De forma semelhante, com o controle por teclado, os valores cresceram de  $M = 0,62$  ( $DP = 0,23$ ) no ambiente não imersivo para  $M = 0,84$  ( $DP = 0,11$ ) no ambiente imersivo.

A ANOVA revelou um efeito principal significativo para o ambiente de visualização ( $F(1; 19) = 20,06; p < 0,001; \eta_g^2 = 0,269$ ), mas não para o método de entrada ( $F(1; 19) = 1,82; p = 0,194$ ). O efeito de interação entre os fatores aproximou-se do limiar de significância, mas não o atingiu ( $F(1; 19) = 4,11; p = 0,057$ ), indicando uma tendência marginal. Dado que a interação não foi estatisticamente significativa, o resultado principal sugere que o ambiente imersivo proporcionou uma maior percepção de qualidade em comparação com o ambiente não imersivo, de forma largamente consistente entre os dois métodos de controle.

## 5.4 Análise de Correlação e Componentes Principais

Para aprofundar a compreensão das relações entre as diversas métricas coletadas, foram conduzidas análises de correlação de Pearson e de Componentes Principais (*Principal Component Analysis* - PCA). O objetivo foi identificar padrões de associação entre as métricas de desempenho, carga de trabalho e presença, e reduzir a dimensionalidade dos dados subjetivos para extrair os fatores que melhor descrevem a experiência do usuário. As correlações significativas encontradas estão apresentadas na Tabela 5.7.

### 5.4.1 Análise de Correlação entre Variáveis

A análise de correlação revelou diversas associações significativas ( $p < 0,05$ ). Primeiramente, observa-se uma forte consistência interna dentro dos questionários: as subescalas do PQ estão fortemente correlacionadas entre si (como *PQ\_Envolvimento*

Tabela 5.7: Correlações de Pearson significativas ( $p < 0,05$ ) entre as variáveis do estudo.

Variável 1	Variável 2	r-valor	p-valor
PQ_Envolvimento	PQ_Media	0,987	<0,001
PQ_Adaptacao	PQ_Media	0,908	<0,001
PQ_Media	PQ_Qualidade	0,898	<0,001
NASA_DM	NASA_Media	0,886	<0,001
NASA_DM	NASA_ES	0,875	<0,001
NASA_ES	NASA_Media	0,870	<0,001
PQ_Adaptacao	PQ_Envolvimento	0,864	<0,001
PQ_Envolvimento	PQ_Qualidade	0,863	<0,001
PQ_Envolvimento	PQ_Fidelidade	0,858	<0,001
PQ_Fidelidade	PQ_Media	0,829	<0,001
NASA_FR	NASA_Media	0,787	<0,001
PQ_Adaptacao	PQ_Qualidade	0,771	<0,001
PQ_Adaptacao	PQ_Fidelidade	0,675	0,001
PQ_Fidelidade	PQ_Qualidade	0,675	0,001
NASA_DE	NASA_FR	0,650	0,002
NASA_DE	PQ_Adaptacao	-0,642	0,002
NASA_FR	PQ_Adaptacao	-0,641	0,002
NASA_DE	NASA_Media	0,628	0,003
NASA_DE	NASA_DM	0,627	0,003
NASA_DF	NASA_DT	0,621	0,003
NASA_DT	NASA_Media	0,593	0,006
NASA_Media	PQ_Adaptacao	-0,580	0,007
NASA_ES	PQ_Media	-0,567	0,009
NASA_DM	PQ_Adaptacao	-0,557	0,011
NASA_DE	PQ_Media	-0,553	0,011
NASA_ES	PQ_Envolvimento	-0,542	0,013
NASA_DF	NASA_Media	0,540	0,014
NASA_DM	PQ_Media	-0,535	0,015
NASA_DM	NASA_FR	0,525	0,017
Colisoes	Tempo	0,523	0,018
NASA_DM	PQ_Envolvimento	-0,520	0,019
NASA_DM	Tempo	0,520	0,019
NASA_DE	PQ_Envolvimento	-0,514	0,020
NASA_DE	PQ_Qualidade	-0,506	0,023
NASA_Media	PQ_Media	-0,496	0,026
NASA_ES	PQ_Adaptacao	-0,495	0,026
NASA_ES	PQ_Fidelidade	-0,481	0,032
NASA_ES	NASA_FR	0,480	0,032
NASA_ES	PQ_Qualidade	-0,479	0,033
NASA_DE	Tempo	0,479	0,033
NASA_DM	PQ_Fidelidade	-0,472	0,036
NASA_Media	PQ_Envolvimento	-0,465	0,039
NASA_FR	PQ_Media	-0,457	0,043
NASA_DT	NASA_FR	0,452	0,045
Colisoes	NASA_ES	0,445	0,049
Colisoes	NASA_DM	0,445	0,049
NASA_Media	PQ_Fidelidade	-0,445	0,050

e  $PQ\_Media$ ,  $r = 0,99$ ), assim como as subescalas do NASA-TLX (como  $NASA\_DM$  e  $NASA\_Media$ ,  $r = 0,89$ ).

Mais importante, foram encontradas correlações significativas entre os diferentes domínios. Notavelmente, métricas de presença mais altas foram associadas a uma menor carga de trabalho percebida. Por exemplo, a Adaptação ( $PQ\_Adaptacao$ ) teve uma forte correlação negativa com a Frustração ( $NASA\_FR$ ) ( $r = -0,64$ ) e uma forte associação com uma melhor Percepção de Desempenho ( $NASA\_DE$ ) (indicada por uma correlação negativa,  $r = -0,64$ ). Similarmente, as métricas de desempenho objetivo correlacionaram-se com a carga de trabalho: um maior número de Colisões foi associado a um maior Nível de Esforço ( $NASA\_ES$ ) ( $r = 0,45$ ) e a uma maior Demanda Mental ( $NASA\_DM$ ) ( $r = 0,45$ ). O Tempo de tarefa também apresentou correlação positiva com a Demanda Mental ( $r = 0,52$ ). Estes resultados sugerem que interfaces que promovem um maior senso de presença podem, conseqüentemente, reduzir a carga de trabalho percebida, e que um pior desempenho está diretamente ligado a uma maior carga cognitiva.

#### 5.4.2 Análise de Componentes Principais (PCA)

Para reduzir a dimensionalidade das subescalas dos questionários e identificar os principais fatores que descrevem a experiência do usuário, foi realizada uma Análise de Componentes Principais (PCA) separadamente para os dados do NASA-TLX e do PQ. Para a interpretação dos componentes, foram consideradas como relevantes as variáveis que apresentaram cargas com valor absoluto superior a 0,40. Este limiar indica que a variável original compartilha mais de 15% de sua variância com o componente principal, assegurando uma associação forte e significativa [113].

Os resultados obtidos para o NASA-TLX estão apresentados na Tabela 5.8. Os dois primeiros componentes principais explicaram 78,14% da variância total. O primeiro componente (53,39% da variância) foi interpretado como um fator de Carga Mental e Esforço, devido às altas cargas positivas das subescalas de Demanda Mental (0,49), Esforço (0,48) e Frustração (0,45). O segundo componente (24,76% da variância) representou uma dimensão de Desempenho sob Demanda Físico-Temporal. Este componente é caracterizado por uma oposição entre a alta demanda física (0,61) e temporal (0,48) de um lado, e uma *melhor* percepção de desempenho do outro (indicada pela carga fortemente negativa de -0,59 na escala invertida  $NASA\_DE$ ).

Tabela 5.8: Cargas dos componentes principais para as subescalas do NASA-TLX. Os dois componentes explicam 78,14% da variância total.

<b>Subescala NASA-TLX</b>	<b>Componente 1</b> <i>Carga Mental</i>	<b>Componente 2</b> <i>Demanda Físico-Temporal</i>
NASA_DE	0,342	-0,590
NASA_DF	0,314	0,605
NASA_DM	0,486	-0,209
NASA_DT	0,342	0,483
NASA_ES	0,480	-0,023
NASA_FR	0,449	-0,091
<i>% da Variância Explicada</i>	53,39%	24,76%

Os resultados obtidos do PCA para o PQ estão apresentados na Tabela 5.9. O primeiro componente principal sozinho explicou 83,98% da variância, indicando que as quatro subescalas medem um conceito unificado e robusto. Devido às altas cargas positivas de todas as subescalas, este componente foi interpretado como um fator de Presença Geral.

Tabela 5.9: Cargas dos componentes principais para as subescalas do Questionário de Presença (PQ). O primeiro componente explica 83,98% da variância.

<b>Subescala PQ</b>	<b>Componente 1</b> <i>(Presença Geral)</i>	<b>Componente 2</b>
PQ_Adaptacao	0,494	-0,420
PQ_Envolvimento	0,535	0,064
PQ_Fidelidade	0,477	0,800
PQ_Qualidade	0,493	-0,423
<i>% da Variância Explicada</i>	83,98%	8,94%

### 5.4.3 Correlação entre Componentes Principais e Desempenho

Finalmente, para entender como os fatores encontrados no PCA se relacionam com o desempenho real na tarefa, foi feita uma análise de correlação final. Os resultados significativos desta análise estão apresentados na Tabela 5.10.

Um dos achados notáveis foi a correlação negativa e significativa entre o fator de Presença Geral e o de Carga Mental ( $r = -0,485; p = 0,030$ ). Este resultado indica que um maior sentimento de presença esteve associado a uma menor percepção de

Tabela 5.10: Correlações de Pearson significativas ( $p < 0,05$ ) entre os componentes principais (PCA) e as métricas de desempenho.

Variável 1	Variável 2	r-valor	p-valor
Colisões	Tempo	0,523	<b>0,018</b>
Carga Mental	Presença Geral	-0,485	<b>0,030</b>
Demanda Físico-Temporal	Presença Geral	0,482	<b>0,032</b>
Tempo	Demanda Físico-Temporal	-0,456	<b>0,044</b>

carga de trabalho mental. As implicações deste achado serão exploradas em maior detalhe na seção de Discussão.

A análise também revelou outras associações importantes. Foi observada uma correlação positiva entre o Tempo de tarefa e o número de Colisões ( $r = 0,523; p = 0,018$ ), mostrando que as tarefas que levaram mais tempo também tiveram mais erros. Além disso, o Tempo de tarefa teve uma correlação negativa com o fator de Desempenho sob Demanda Físico-Temporal ( $r = -0,456; p = 0,044$ ). Isso indica que, quanto maior o tempo para completar a tarefa, pior foi a percepção do participante sobre seu próprio desempenho.

Por fim, o fator de Presença Geral (PQ) apresentou uma correlação positiva com o de Desempenho sob Demanda Físico-Temporal ( $r = 0,482; p = 0,032$ ). Isso sugere que um maior senso de presença esteve associado a uma maior percepção de demanda física e temporal, mas sem que isso prejudicasse a sensação de um bom desempenho.

## 5.5 Análise e Discussão dos Resultados

A análise dos experimentos realizados revelou diferenças significativas na percepção de controle, imersão e desempenho entre os modos de visualização e formas de controle avaliadas. Os resultados quantitativos, detalhados nas seções anteriores, foram enriquecidos e contextualizados pelo *feedback* qualitativo fornecido pelos 20 participantes. A seguir, é apresentada uma análise integrada que combina os dados estatísticos com as percepções subjetivas dos usuários.

A análise das preferências declaradas pelos participantes reforçou as tendências observadas nos dados quantitativos. A combinação teclado e ambiente imersivo (KB-RV) foi a opção mais votada, escolhida por 65% dos usuários, seguida pela combinação háptico e ambiente imersivo (HP-RV), com 30%. A combinação teclado e ambiente não imersivo (KB-2D) foi pouco preferida (5%), e a combinação háptico e ambiente não imersivo (HP-2D) não recebeu votos.

Estes resultados evidenciam que o ambiente imersivo foi um fator determinante para a aceitação e a sensação de imersão. No entanto, o *feedback* qualitativo revela um segundo fator crucial: a familiaridade. Muitos participantes atribuíram a preferência pelo teclado à sua experiência prévia, como relatado por um usuário: “*Acredito que por estar acostumada com as setas do teclado, foi mais fácil controlar o robô por elas.*” Outro participante mencionou a semelhança com controles de *videogames*, afirmando que o teclado foi “*mais fácil de utilizar, com baixa curva de aprendizado*” devido à “*similaridade com controles de vídeo games que estou familiarizada*”. Isso sugere que a facilidade de uso percebida foi fortemente influenciada pela transferência de habilidades de outras áreas, um fator que se sobrepôs, para a maioria, aos benefícios sensoriais de interfaces menos familiares.

Adicionalmente, é relevante discutir a dimensão do esforço físico. Embora a diferença no efeito principal de controle não tenha sido estatisticamente significativa, os dados descritivos revelaram uma tendência de maior esforço percebido com o controle háptico. Essa diferença pode ser atribuída à natureza biomecânica de cada dispositivo. O controle por teclado se baseia em movimentos distais, primariamente dos dedos e do pulso, que são de baixa exigência energética e com os quais os usuários têm vasta familiaridade. Em contraste, a manipulação do controle háptico exige o movimento do braço e do ombro, mantendo-os em uma posição sem suporte completo.

Esta forma de interação, na qual o usuário permanece com o membro suspenso em interações com o ambiente virtual, é comumente associada ao fenômeno conhecido como ‘*gorilla arm effect*’, que descreve a fadiga muscular resultante de se manter o braço suspenso para interagir com uma interface [114]. Portanto, a maior pontuação de esforço (NASA\_ES) relatada com o controle háptico é consistente com os princípios ergonômicos da interação humano-computador e ajuda a explicar por que o teclado, apesar de menos imersivo, foi percebido por muitos como mais confortável.

A superioridade do ambiente imersivo sobre o ambiente não imersivo, consistentemente observada nas análises estatísticas de presença e carga de trabalho, foi ecoada de forma enfática nos relatos dos participantes. O ambiente imersivo foi descrito como um componente que proporcionou uma percepção espacial muito mais ampla, permitindo uma melhor noção do robô no ambiente. Um participante descreveu a diferença de forma contundente: *“Quem pensou em utilizar, em algum momento da vida, forma de visualização não imersiva está completamente desconectado da realidade.”*

Essa percepção aprimorada se traduziu em maior confiança e segurança. Conforme um relato, *“o ambiente virtual dá uma noção muito melhor do espaço que a visualização não imersiva”*. A capacidade de ter múltiplas perspectivas, incluindo a visão *top-down*, foi frequentemente citada como um diferencial chave que ajudou a *“evitar obstáculos e a planejar trajetórias mais seguras”*. Este consenso entre os dados quantitativos e qualitativos corrobora a hipótese de que a visualização no ambiente imersivo permite um entendimento mais completo e funcional do ambiente de teleoperação.

O controle háptico apresentou um quadro mais complexo, caracterizado por um claro *trade-off* entre o potencial de imersão e a dificuldade de adaptação inicial. Por um lado, quando combinado com o ambiente imersivo, foi reconhecido como a configuração que conferiu a *“maior sensação de controle e imersão”*. Um participante destacou que *“o fato de ter o controle háptico o legal é que dá para entender melhor quando bate em alguma coisa, devido ao feedback”*. Essa percepção de maior segurança foi corroborada por outro usuário, que afirmou: *“por mais que a adaptação tenha sido difícil no começo, a segurança que senti ao utilizá-lo foi compensatória.”*

Por outro lado, a dificuldade de uso foi um tema recorrente. Vários participantes relataram problemas de precisão, como um que sentiu que *“o controle háptico não me dá a sensação de precisão no movimento, senti dificuldade para fazer curvas principalmente”*. Outro descreveu o controle como *“muito contraintuitivo”*. Essa dificuldade inicial, ou alta curva de aprendizado, explica por que, apesar de seus benefícios imersivos, o controle háptico não foi a escolha preferida da maioria. A experiência sugere que, embora o teclado seja mais intuitivo a curto prazo, o controle háptico tem o potencial de oferecer uma interação superior, desde que acompanhado de um período de familiarização adequado.

Finalmente, os participantes reconheceram a importância da integração de múltiplos sistemas de auxílio à navegação. Foi percebido que *“os sistemas de auxílio à*

*navegação são essenciais para reduzir a insegurança do usuário, especialmente quando combinados com o controle háptico e a RV*". A presença dos sensores e das diferentes visões no ambiente imersivo foi citada como um fator que *"facilitou a tomada de decisão e a percepção de segurança no ambiente"*.

No entanto, alguns problemas técnicos foram relatados, como a perda temporária de controle devido a questões de sincronização na rede, que *"influenciou na qualidade da visualização negativamente"* e afetou momentaneamente a sensação de domínio. Embora esses incidentes não tenham impedido a conclusão das tarefas, eles ressaltam a importância crítica da estabilidade da comunicação para garantir a confiabilidade da experiência de teleoperação.

De forma geral, a análise combinada dos dados quantitativos e qualitativos indica que, apesar das diferenças subjetivas entre teclado e controle háptico, a utilização do ambiente imersivo trouxe benefícios consistentes na navegação e percepção do ambiente. O teclado se destacou pela facilidade de uso, enquanto a combinação háptico e ambiente imersivo (HP-RV) forneceu a maior imersão e confiança, validando parcialmente as hipóteses iniciais e destacando os pontos fortes e limitações de cada configuração.

## 5.6 Conclusões

Os resultados obtidos na avaliação da interface de teleoperação com usuários evidenciam a importância da realidade virtual como elemento central para a percepção de imersão e controle no processo de navegação robótica. A superioridade do ambiente imersivo em relação ao ambiente não imersivo, confirmada estatisticamente, indica que a amplitude de visão e a sensação de presença no ambiente são determinantes para a tomada de decisão segura, o que se alinha a estudos anteriores sobre interfaces imersivas aplicadas à teleoperação. Dessa forma, o ambiente imersivo mostrou-se não apenas um recurso estético, mas um componente funcional essencial para a eficiência da tarefa.

Essa conexão entre presença e eficiência é diretamente sustentada pela análise quantitativa, que revelou uma forte correlação negativa entre o senso de presença e a carga de trabalho mental. A interface do ambiente imersivo, ao entregar uma representação espacial intuitiva, alivia o operador do esforço cognitivo necessário para interpretar e reconstruir um ambiente 3D a partir de uma tela plana. Essa liberação

de recursos mentais justifica a melhoria observada na tomada de decisão e a redução de erros, como as colisões.

Outro aspecto central da análise é a comparação entre teclado e controle háptico, que revela um claro *trade-off* entre a intuitividade e o potencial imersivo. Embora o teclado tenha se mostrado mais rápido e tenha sido o preferido pela maioria, a análise qualitativa sugere que essa preferência está fortemente ligada à familiaridade dos usuários com paradigmas de controle de jogos e computadores. Em contraste, a combinação háptico e ambiente imersivo (HP-RV), apesar de exigir uma maior curva de aprendizado, foi a que demonstrou maior capacidade de proporcionar imersão e sensação de presença.

A dificuldade relatada com o controle háptico sugere que o desafio transcende a mera familiaridade do usuário, apontando para uma questão fundamental de mapeamento de controle. O dispositivo utilizado, com 3 graus de liberdade, não possui uma correspondência cinemática direta com o modelo de um robô Ackermann, ao contrário de um volante, cujo controle rotacional espelha a ação de esterçar as rodas. Essa falta de isomorfismo pode aumentar a carga cognitiva e a curva de aprendizado. Portanto, este resultado abre espaço para trabalhos futuros que investiguem interfaces de controle com maior correspondência ao modelo do robô.

A análise também revelou a relevância dos sistemas de auxílio à navegação. Os dados qualitativos indicaram que, quando integrados ao ambiente imersivo e ao controle háptico, esses recursos reduziram a insegurança dos participantes e facilitaram a execução de manobras complexas. Isso sugere que o desempenho ideal não depende apenas da interface de controle, mas da sinergia entre controle, visualização e sistemas auxiliares, confirmando a necessidade de soluções multimodais na teleoperação.

Finalmente, os problemas ocasionais de rede, relatados durante o uso do controle háptico, apontam para a dependência da infraestrutura de comunicação no sucesso da experiência. Mesmo que os impactos tenham sido momentâneos, eles revelam uma vulnerabilidade crítica em aplicações práticas. Essa constatação reforça que, para além das escolhas de interface, a estabilidade do sistema de comunicação é um fator-chave para garantir confiabilidade e segurança.

De forma geral, a discussão dos resultados mostra que não há uma única solução

superior em todos os aspectos, mas sim diferentes vantagens dependendo do critério considerado: facilidade e rapidez de adaptação favorecem o teclado, enquanto imersão e confiança são reforçadas pela combinação háptico e ambiente imersivo (HP-RV). A síntese desses achados indica que o desenvolvimento de interfaces futuras pode se beneficiar de uma abordagem híbrida, que combine a familiaridade do teclado com os recursos sensoriais avançados da realidade virtual e do *feedback* háptico, conciliando aprendizado rápido e imersão plena.

## Capítulo 6

# Conclusões e Trabalhos Futuros

Esta dissertação propôs o desenvolvimento de um sistema de teleoperação multimodal com uma plataforma robótica móvel, um módulo de mapeamento e localização simultâneos (SLAM) e uma interface com ambiente virtual imersivo (RV) e dispositivo háptico. A arquitetura foi projetada para operar de forma distribuída, com processamento compartilhado entre o robô e o operador, viabilizando um gêmeo digital de baixa latência. Dois estudos experimentais foram conduzidos: o primeiro validou a precisão do mapeamento visual contra abordagens clássicas, e o segundo avaliou a interface com 20 usuários.

A validação com usuários confirmou que o ambiente virtual imersivo (RV) foi o fator mais impactante na experiência do operador, aumentando significativamente o senso de presença e reduzindo a carga de trabalho percebida e o número de colisões em comparação com o ambiente não imersivo (2D). A comparação entre os métodos de controle, no entanto, revelou um *trade-off* complexo entre familiaridade, desempenho e imersão. A análise estatística encontrou efeitos de interação significativos, demonstrando que o benefício proporcionado pelo ambiente virtual imersivo foi desproporcionalmente maior para o controle por teclado.

Embora preferido pela maioria por sua intuitividade e rapidez, o teclado só atingiu um nível de segurança comparável ao controle háptico quando utilizado no ambiente virtual imersivo, que atuou como um “equalizador” que mitigou as dificuldades do controle mais simples. Em contraste, o controle háptico, apesar de associado à maior imersão e confiança, apresentou uma acentuada curva de aprendizado. Este resultado pode ser atribuído não apenas à falta de familiaridade, mas a uma baixa correspondência cinemática entre o dispositivo e a tarefa de navegação, diferentemente de interfaces como um volante de carro.

Estes resultados reforçam a ideia de que a teleoperação se beneficia de uma combinação sinérgica de estímulos, onde a imersão visual proporcionada pelo ambiente virtual imersivo e o *feedback* tátil não apenas aprimoram a eficiência, mas promovem uma interação mais intuitiva. A pesquisa estabelece, assim, uma base sólida que aponta para a criação de interfaces de alta fidelidade como um passo essencial para o avanço da colaboração humano-robô.

Apesar dos resultados promissores, o sistema desenvolvido opera de forma idêntica para diferentes usuários, sem mecanismos de adaptação. A interação, embora enriquecida, ainda se dá em um ciclo aberto, no qual o sistema fornece informações, mas carece da capacidade de perceber e reagir ao estado interno do operador. Essa limitação evidencia o desafio contemporâneo da Interação Humano-Robô: transitar de sistemas funcionais para sistemas colaborativos e adaptativos.

Nesse sentido, os desdobramentos futuros desta pesquisa convergem para a evolução do sistema em direção a um arcabouço adaptativo. O emprego do sistema como ferramenta para coletar e modelar fatores humanos em tempo real poderá abrir caminho para interfaces que se modulam dinamicamente segundo as necessidades do operador. Essa perspectiva pode ser enriquecida pela exploração de tecnologias emergentes de visualização, como o uso de *Gaussian Splatting* para reconstruções fotorrealistas, e pela ampliação das capacidades do *feedback* háptico para transmitir informações mais complexas sobre o ambiente.

Adicionalmente, sugere-se a investigação de interfaces de controle alternativas que ofereçam um mapeamento mais isomórfico com a cinemática do robô, visando reduzir a carga cognitiva e aumentar a intuitividade da navegação. Além disso, a incorporação de comandos de mais alto nível, que aliem controle compartilhado e supervisão humana, pode otimizar a colaboração entre autonomia robótica e intervenção humana. Torna-se necessária a condução de um estudo sistemático sobre o impacto de latência, *jitter* e perda de pacotes na performance do operador, visando garantir a robustez da aplicação em cenários de teleoperação remota via internet.

Assim, a transição de um sistema multimodal estático para uma plataforma adaptativa e centrada no ser humano delineia a continuação natural desta pesquisa, consolidando a visão de que a próxima geração de sistemas robóticos colaborativos deverá ser não apenas funcional, mas também responsiva, capaz de ajustar-se às particularidades dos seus usuários.

Uma limitação da configuração atual é a discrepância sensorial que ocorre quando o operador está fisicamente parado, mas a interface visual simula o deslocamento do robô. Esse conflito entre a visão e a sensação corporal de movimento pode gerar desconforto. Outra direção para trabalhos futuros seria, portanto, investigar a integração de sistemas que forneçam ao operador um *feedback* físico de movimento, como plataformas móveis que simulem a aceleração do veículo. Tal adição teria o potencial de aumentar o conforto e elevar ainda mais o nível de imersão da experiência de teleoperação.

Por fim, o presente estudo quantificou o número total de colisões como uma métrica geral de segurança. Uma análise mais granular, distinguindo entre colisões frontais e de ré, representaria um valioso desdobramento. Tal detalhamento poderia oferecer *insights* mais profundos sobre os tipos de falhas de percepção ou controle associados a cada interface. Por exemplo, colisões frontais podem indicar falhas no julgamento de distância, enquanto colisões de ré podem apontar para uma baixa consciência periférica ou do espaço traseiro do robô. A inclusão desta métrica em estudos futuros permitiria uma avaliação mais completa dos fatores humanos na teleoperação.

## Bibliografia

- [1] G. LeMasurier et al., “Comparing a 2D Keyboard and Mouse Interface to Virtual Reality for Human-in-the-Loop Robot Planning for Mobile Manipulation”, em *2024 33rd IEEE International Conference on Robot and Human Interactive Communication (ROMAN)*, IEEE, 2024, pp. 2197–2203.
- [2] B. Siciliano e O. Khatib, *Springer Handbook of Robotics*, 2nd. Springer Publishing Company, Incorporated, 2016.
- [3] D. J. Rea e S. H. Seo, “Still not solved: A call for renewed focus on user-centered teleoperation interfaces”, *Frontiers in Robotics and AI*, v. 9, p. 704 225, 2022.
- [4] M. K. van Den Broek e T. B. Moeslund, “What is proactive human-robot interaction?-a review of a progressive field and its definitions”, *ACM Transactions on Human-Robot Interaction*, v. 13, n. 4, pp. 1–30, 2024.
- [5] T. B. Sheridan, *Telerobotics, automation, and human supervisory control*. MIT press, 1992.
- [6] M. Hejrati, P. Mustalahti e J. Mattila, *Robust Immersive Bilateral Teleoperation of Beyond-Human-Scale Systems with Enhanced Transparency and Sense of Embodiment*, arXiv:2505.14486, 2025.
- [7] L. Bareišytė et al., “Questionnaires for evaluating virtual reality: A systematic scoping review”, *Computers in Human Behavior Reports*, v. 16, p. 100 505, 2024.
- [8] C. E. Lathan e M. Tracey, “The effects of operator spatial perception and sensory feedback on human-robot teleoperation performance”, *Presence*, v. 11, n. 4, pp. 368–377, 2002.
- [9] Y.-P. Su, X.-Q. Chen, C. Zhou, L. H. Pearson, C. G. Pretty e J. G. Chase, “Integrating virtual, mixed, and augmented reality into remote robotic applications: A brief review of extended reality-enhanced robotic systems for intuitive telemanipulation and telemanufacturing tasks in hazardous conditions”, *Applied Sciences*, v. 13, n. 22, p. 12 129, 2023.

- [10] Y. Su, X. Chen, T. Zhou, C. Pretty e G. Chase, "Mixed reality-integrated 3D/2D vision mapping for intuitive teleoperation of mobile manipulator", *Robotics and Computer-Integrated Manufacturing*, v. 77, p. 102-332, 2022.
- [11] H. Liu e L. Wang, "Remote human-robot collaboration: A cyber-physical system application for hazard manufacturing environment", *Journal of manufacturing systems*, v. 54, pp. 24-34, 2020.
- [12] H. Stedman, B. B. Kocer, M. Kovac e V. M. Pawar, "VRTAB-map: A configurable immersive teleoperation framework with online 3D reconstruction", em *2022 IEEE International Symposium on Mixed and Augmented Reality Adjunct (ISMAR-Adjunct)*, IEEE, 2022, pp. 104-110.
- [13] Y.-P. Su, X.-Q. Chen, T. Zhou, C. Pretty e G. Chase, "Mixed-reality-enhanced human-robot interaction with an imitation-based mapping approach for intuitive teleoperation of a robotic arm-hand system", *Applied Sciences*, v. 12, n. 9, p. 4740, 2022.
- [14] D. Whitney, E. Rosen, D. Ullman, E. Phillips e S. Tellex, "Ros reality: A virtual reality framework using consumer-grade hardware for ros-enabled robots", em *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, IEEE, 2018, pp. 1-9.
- [15] J. Nakanishi, S. Itadera, T. Aoyama e Y. Hasegawa, "Towards the development of an intuitive teleoperation system for human support robot using a VR device", *Advanced Robotics*, v. 34, n. 19, pp. 1239-1253, 2020.
- [16] G. Li, F. Caponetto, E. Del Bianco, V. Katsageorgiou, I. Sarakoglou e N. G. Tsagarakis, "Incomplete orientation mapping for teleoperation with one DoF master-slave asymmetry", *IEEE Robotics and Automation Letters*, v. 5, n. 4, pp. 5167-5174, 2020.
- [17] E. Triantafyllidis, C. Mcgreavy, J. Gu e Z. Li, "Study of multimodal interfaces and the improvements on teleoperation", *IEEE Access*, v. 8, pp. 78 213-78 227, 2020.
- [18] M. Zolotas, M. Wonsick, P. Long e T. Padir, "Motion polytopes in virtual reality for shared control in remote manipulation applications", *Frontiers in Robotics and AI*, v. 8, p. 730 433, 2021.
- [19] M. Kamezaki, M. Miyata e S. Sugano, "Video presentation based on multiple-flying camera to provide continuous and complementary images for teleoperation", *Automation in Construction*, v. 159, p. 105 285, 2024.

- [20] A. E. Abdelaal, A. Avinash, M. Kalia, G. D. Hager e S. E. Salcudean, "A multi-camera, multi-view system for training and skill assessment for robot-assisted surgery", *International journal of computer assisted radiology and surgery*, v. 15, n. 8, pp. 1369–1377, 2020.
- [21] J. Y. Chen, E. C. Haas e M. J. Barnes, "Human performance issues and user interface design for teleoperated robots", *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, v. 37, n. 6, pp. 1231–1245, 2007.
- [22] R. Chen, X. Zhu, Z. Chen, Y. Tian, L. Liang e X. Wang, "A Mixed-Reality-Based Unknown Space Navigation Method of a Flexible Manipulator", *Sensors*, v. 23, n. 8, p. 3840, 2023.
- [23] J. DelPreto et al., "Helping robots learn: a human-robot master-apprentice model using demonstrations via virtual reality teleoperation", em *2020 IEEE International Conference on Robotics and Automation (ICRA)*, IEEE, 2020, pp. 10 226–10 233.
- [24] M. E. Walker, M. Gramopadhye, B. Ikeda, J. Burns e D. Szafir, "The cyber-physical control room: A mixed reality interface for mobile robot teleoperation and human-robot teaming", em *Proceedings of the 2024 ACM/IEEE International Conference on Human-Robot Interaction*, 2024, pp. 762–771.
- [25] A. Cannavò, "Interfaces for human-centered production and use of computer graphics assets", 2020.
- [26] V. Girbes-Juan, V. Schettino, Y. Demiris e J. Tornero, "Haptic and visual feedback assistance for dual-arm robot teleoperation in surface conditioning tasks", *IEEE Transactions on Haptics*, v. 14, n. 1, pp. 44–56, 2020.
- [27] F. P. Audonnet, "Robotic teleoperation: a cross-cultural study of user experience and performance", tese de dout., University of Glasgow, 2025.
- [28] P. Gliesche, T. Krick, M. Pflingsthor, S. Drolshagen, C. Kowalski e A. Hein, "Kinesthetic device vs. keyboard/mouse: a comparison in home care telemanipulation", *Frontiers in Robotics and AI*, v. 7, p. 561 015, 2020.
- [29] J. Allspaw, L. Heinold e H. A. Yanco, "Design of virtual reality for humanoid robots with inspiration from video games", em *International conference on human-computer interaction*, Springer, 2019, pp. 3–18.
- [30] T. Boehringer, J. Embley-Riches, K. Hammoud, V. Modugno e D. Kanoulas, "Immersive Teleoperation Framework for Locomanipulation Tasks", *arXiv preprint arXiv:2504.15229*, 2025.

- [31] A. Smith e M. Kennedy III, "An Augmented Reality Interface for Teleoperating Robot Manipulators", *arXiv preprint arXiv:2409.18394*, 2024.
- [32] Y. Wu, B. Zhao e Q. Li, "The Teleoperation of Robot Arms by Interacting with an Object's Digital Twin in a Mixed Reality Environment", *Applied Sciences*, v. 15, n. 7, p. 3549, 2025.
- [33] X. Li, X. Jiang, P. Dahlinger, G. Neumann e R. Lioutikov, "Beyond Visuals: Investigating Force Feedback in Extended Reality for Robot Data Collection", *arXiv preprint arXiv:2503.20714*, 2025.
- [34] Y.-P. Su, X.-Q. Chen, T. Zhou, C. Pretty e G. Chase, "Mixed Reality-Enhanced Intuitive Teleoperation with Hybrid Virtual Fixtures for Intelligent Robotic Welding", *Applied Sciences*, v. 11, n. 23, p. 11 280, 2021.
- [35] C.-Y. Kuo, C.-C. Huang, C.-H. Tsai, Y.-S. Shi e S. Smith, "Development of an immersive SLAM-based VR system for teleoperation of a mobile manipulator in an unknown environment", *Computers in Industry*, v. 132, p. 103 502, 2021.
- [36] R. Suzuki, A. Karim, T. Xia, H. Hedayati e N. Marquardt, "Augmented Reality and Robotics: A Survey and Taxonomy for AR-enhanced Human-Robot Interaction and Robotic Interfaces", em *Proceedings of the 2022 CHI Conference on Human Factors in Computing Systems*, 2022, p. 553.
- [37] S. B. Kamtam, Q. Lu, F. Bouali, O. C. L. Haas e S. Birrell, "Network Latency in Teleoperation of Connected and Autonomous Vehicles: A Review of Trends, Challenges, and Mitigation Strategies", *Sensors*, v. 24, n. 12, p. 3957, 2024.
- [38] M. Selim, D. Dresscher e M. Abayazid, "A comprehensive review of haptic feedback in minimally invasive robotic liver surgery: Advancements and challenges", *International Journal of Medical Robotics and Computer Assisted Surgery*, 2023.
- [39] C. Zheng, K. Wang, S. Gao, Y. Yu, Z. Wang e Y. Tang, "Design of multi-modal feedback channel of human-robot cognitive interface for teleoperation in manufacturing", *Journal of Intelligent Manufacturing*, 2024.
- [40] M. Moniruzzaman, A. Rassau, D. Chai e S. M. S. Islam, "Teleoperation methods and enhancement techniques for mobile robots: A comprehensive survey", *Robotics and Autonomous Systems*, v. 150, p. 103 973, 2022.

- [41] F. Machado, M. Loureiro, M. Bezerra, C. Zimerer, R. Mello e A. Frizera, "Virtual Obstacle Avoidance Strategy: Navigating through a Complex Environment While Interacting with Virtual and Physical Elements", *Sensors*, v. 24, n. 19, p. 6212, 2024.
- [42] J. K. Gibbs, M. Gillies e X. Pan, "A comparison of the effects of haptic and visual feedback on presence in virtual reality", *International Journal of Human-Computer Studies*, v. 157, p. 102717, 2022.
- [43] H. Stedman, B. B. Kocer, N. van Zalk, M. Kovac e V. M. Pawar, "Evaluating Immersive Teleoperation Interfaces: Coordinating Robot Radiation Monitoring Tasks in Nuclear Facilities", em *2023 IEEE International Conference on Robotics and Automation (ICRA)*, IEEE, 2023, pp. 11972–11978.
- [44] J. E. Solanes, A. Muñoz, L. Gracia e J. Tornero, "Virtual Reality-Based Interface for Advanced Assisted Mobile Robot Teleoperation", *Applied Sciences*, v. 12, n. 12, p. 6071, 2022.
- [45] M. Pan et al., "Collision risk assessment and automatic obstacle avoidance strategy for teleoperation robots", *Computers & Industrial Engineering*, v. 169, p. 108275, 2022.
- [46] K. Feng, Q. Xu e L. M. Tam, "Design and Development of a Teleoperated Robotic Microinjection System With Haptic Feedback", *IEEE Robotics and Automation Letters*, v. 6, n. 3, pp. 6092–6099, 2021.
- [47] A. Abdullah, D. Blow, R. Chen, T. Uthai, E. J. Du e M. J. Islam, *Human-Machine Interfaces for Subsea Telerobotics: From Soda-straw to Natural Language Interactions*, arXiv:2412.01753, 2025.
- [48] C. Del Gaizo, "Ready Robot One: teleoperation and manipulation", Università degli studi di Genova, rel. técn., 2023.
- [49] H. Hedayati, M. Walker e D. Szafir, "Improving collocated robot teleoperation with augmented reality", em *Proceedings of the 2018 ACM/IEEE International Conference on Human-Robot Interaction*, 2018, pp. 78–86.
- [50] A. d. I. Rosa-Garcia, Á. I. S. Marrufo, D. Luviano-Cruz, A. Rodriguez-Ramirez e F. Garcia-Luna, "Bridging Remote Operations and Augmented Reality: An Analysis of Current Trends", *IEEE Access*, v. 13, pp. 36502–36526, 2025.
- [51] M. E. Walker, H. Hedayati e D. Szafir, "Robot Teleoperation with Augmented Reality Virtual Surrogates", em *2019 14th ACM/IEEE International Conference on Human-Robot Interaction (HRI)*, 2019, pp. 202–210.

- [52] Y. Su, "Development and Evaluation of Mixed Reality-Enhanced Robotic Systems for Intuitive Tele-manipulation and Tele-manufacturing Tasks in Hazardous Conditions", Ph.D. Thesis, University of Canterbury, 2022.
- [53] F. Ferland, F. Pomerleau, C. T. L. Dinh e F. Michaud, "Egocentric and Exocentric Teleoperation Interface Using Real-Time, 3D Video Projection", em *2009 4th ACM/IEEE International Conference on Human-Robot Interaction (HRI)*, 2009, pp. 37–44.
- [54] R. Jangir, N. Hansen, S. Ghosal, M. Jain e X. Wang, "Look Closer: Bridging Egocentric and Third-Person Views With Transformers for Robotic Manipulation", *IEEE Robotics and Automation Letters*, v. 7, n. 2, pp. 3046–3053, 2022.
- [55] J. Chen, M. Glover, C. Yang, C. Li, Z. Li e A. Cangelosi, "Development of an immersive interface for robot teleoperation", em *Conference Towards Autonomous Robotic Systems*, Springer, 2017, pp. 1–15.
- [56] K. Cieřlik, P. Krogul, T. Muszyński, M. Przybysz, A. Rubiec e R. K. Typiak, "Influence of Camera Placement on UGV Teleoperation Efficiency in Complex Terrain", *Applied Sciences*, v. 14, n. 18, p. 8297, 2024.
- [57] E. J. Fabris, V. A. Sangalli, L. P. Soares e M. S. Pinho, "Immersive telepresence on the operation of unmanned vehicles", *International Journal of Advanced Robotic Systems*, v. 18, n. 1, p. 1729881420978544, 2021.
- [58] H. Stedman, B. B. Kocer, N. Van Zalk, M. Kovac e V. M. Pawar, "Evaluating immersive teleoperation interfaces: Coordinating robot radiation monitoring tasks in nuclear facilities", em *Proceedings-IEEE International Conference on Robotics and Automation*, IEEE, vol. 2023, 2023, pp. 11972–11978.
- [59] R. Bretin, E. Cross e M. Khamis, "Safety at Stake: How Individuals Task Prioritization Influences Human-Drone Proxemics", *ACM Transactions on Human-Robot Interaction*, 2025.
- [60] T. Zhou, Q. Zhu e J. Du, "Intuitive robot teleoperation for civil engineering operations with virtual reality and deep learning scene reconstruction", *Advanced Engineering Informatics*, v. 46, p. 101170, 2020.
- [61] E. Triantafyllidis, C. Mcgreavy, J. Gu e Z. Li, "Study of multimodal interfaces and the improvements on teleoperation", *IEEE Access*, v. 8, pp. 78213–78227, 2020.

- [62] C.-Y. Kuo, C.-C. Huang, C.-H. Tsai, Y.-S. Shi e S. Smith, "Development of an immersive SLAM-based VR system for teleoperation of a mobile manipulator in an unknown environment", *Computers in Industry*, v. 132, p. 103 502, 2021.
- [63] S. Neumeier, P. Wintersberger, A.-K. Frison, A. Becher, C. Facchi e A. Riener, "Teleoperation: The holy grail to solve problems of automated driving? Sure, but latency matters", em *Proceedings of the 11th International Conference on Automotive User Interfaces and Interactive Vehicular Applications*, 2019, pp. 186–197.
- [64] C. Blackett, A. Fernandes, E. Teigen e T. Thoresen, "Effects of signal latency on human performance in Teleoperations", em *International Conference on Human Interaction and Emerging Technologies*, Springer, 2021, pp. 386–393.
- [65] D. G. Black, D. Andjelic e S. E. Salcudean, "Evaluation of communication and human response latency for (human) teleoperation", *IEEE Transactions on Medical Robotics and Bionics*, v. 6, n. 1, pp. 53–63, 2024.
- [66] S. B. Kamtam, Q. Lu, F. Bouali, O. C. Haas e S. Birrell, "Network latency in teleoperation of connected and autonomous vehicles: A review of trends, challenges, and mitigation strategies", *Sensors (Basel, Switzerland)*, v. 24, n. 12, p. 3957, 2024.
- [67] R. Luz, J. L. Silva e R. Ventura, "Enhanced teleoperation interfaces for multi-second latency conditions: System design and evaluation", *IEEE Access*, v. 11, pp. 10 935–10 953, 2023.
- [68] P. Milgram e F. Kishino, "A taxonomy of mixed reality visual displays", *IEICE TRANSACTIONS on Information and Systems*, v. 77, n. 12, pp. 1321–1329, 1994.
- [69] M. Wonsick e T. Padir, "A systematic review of virtual reality interfaces for controlling and interacting with robots", *Applied Sciences*, v. 10, n. 24, p. 9051, 2020.
- [70] M. Maciaś et al., "Measuring performance in robotic teleoperation tasks with virtual reality headgear", em *Conference on Automation*, Springer, 2019, pp. 408–417.
- [71] A. Martín-Barrio, J. J. Roldán, S. Terrile, J. Del Cerro e A. Barrientos, "Application of immersive technologies and natural language to hyper-redundant robot teleoperation", *Virtual Reality*, v. 24, n. 3, pp. 541–555, 2020.

- [72] T. Kaarlela, H. Arnarson, T. Pitkääho, B. Shu, B. Solvang e S. Pieskä, “Common educational teleoperation platform for robotics utilizing digital twins”, *Machines*, v. 10, n. 7, p. 577, 2022.
- [73] I. Vieira, M. Bezerra, M. Loureiro, F. Machado, R. Mello e A. Frizzera, “Integração de Câmeras Estéreo para Navegação Autônoma e SLAM Visual em um Robô Móvel”, em *Anais do XXV Congresso Brasileiro de Automática (CBA)*, Barra da Tijuca, AL, Brasil, 2024.
- [74] X.-F. Han, J. S. Jin, M.-J. Wang, W. Jiang, L. Gao e L. Xiao, “A review of algorithms for filtering the 3D point cloud”, *Signal Processing: Image Communication*, v. 57, pp. 103–112, 2017.
- [75] V. Patil e M. Hutter, “Radiance fields for robotic teleoperation”, em *2024 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, IEEE, 2024, pp. 13 861–13 868.
- [76] J. Lee, B. Bagheri e H.-A. Kao, “A cyber-physical systems architecture for industry 4.0-based manufacturing systems”, *Manufacturing letters*, v. 3, pp. 18–23, 2015.
- [77] E. Negri, L. Fumagalli e M. Macchi, “A review of the roles of digital twin in CPS-based production systems”, *Procedia manufacturing*, v. 11, pp. 939–948, 2017.
- [78] P. Zheng e A. S. Sivabalan, “A generic tri-model-based approach for product-level digital twin development in a smart manufacturing environment”, *Robotics and Computer-Integrated Manufacturing*, v. 64, p. 101 958, 2020.
- [79] W. Fan et al., “Digital Twin-Driven Mixed Reality Framework for Immersive Teleoperation With Haptic Rendering”, *IEEE Robotics and Automation Letters*, v. 8, n. 12, pp. 8494–8501, 2023.
- [80] Y. Cai, Y. Wang e M. Burnett, “Using augmented reality to build digital twin for reconfigurable additive manufacturing system”, *Journal of Manufacturing Systems*, v. 56, pp. 598–604, 2020.
- [81] W. Kritzinger, M. Karner, G. Traar, J. Henjes e W. Sihn, “Digital Twin in manufacturing: A categorical literature review and classification”, *IFAC-PapersOnLine*, v. 51, n. 11, pp. 1016–1022, 2018.
- [82] C. Cimino, E. Negri e L. Fumagalli, “Review of digital twin applications in manufacturing”, *Computers in Industry*, v. 113, p. 103 130, 2019.

- [83] A. Protic, Z. Jin, R. Marian, K. Abd, D. Campbell e J. Chahl, "Implementation of a Bi-Directional Digital Twin for Industry 4 Labs in Academia: A Solution Based on OPC UA", em *2020 IEEE International Conference on Industrial Engineering and Engineering Management (IEEM)*, IEEE, 2020, pp. 979–983.
- [84] Q. Qi e F. Tao, "Digital Twin and Big Data Towards Smart Manufacturing and Industry 4.0: 360 Degree Comparison", *IEEE Access*, v. 6, pp. 3585–3593, 2018.
- [85] H. Laaki, Y. Miche e K. Tammi, "Prototyping a Digital Twin for Real Time Remote Control Over Mobile Networks: Application of Remote Surgery", *IEEE Access*, v. 7, pp. 20 325–20 336, 2019.
- [86] F. P. Audonnet, J. Grizou, A. Hamilton e G. Aragon-Camarasa, "TELESIM: A Modular and Plug-and-Play Framework for Robotic Arm Teleoperation using a Digital Twin", em *2024 IEEE International Conference on Robotics and Automation (ICRA)*, IEEE, 2024, pp. 17 770–17 777.
- [87] F. Zhao, W. Deng e D. T. Pham, "A Robotic Teleoperation System with Integrated Augmented Reality and Digital Twin Technologies for Disassembling End-of-Life Batteries", *Batteries*, v. 10, n. 11, p. 382, 2024.
- [88] W. Fan et al., "Digital Twin-Driven Mixed Reality Framework for Immersive Teleoperation With Haptic Rendering", *IEEE Robotics and Automation Letters*, v. 8, n. 12, pp. 8494–8501, 2023.
- [89] C. Pacchierotti e D. Prattichizzo, "Cutaneous/Tactile Haptic Feedback in Robotic Teleoperation: Motivation, Survey, and Perspectives", *IEEE Transactions on Robotics*, v. 40, pp. 978–998, 2024.
- [90] J. T. Dennerlein, D. B. Martin e C. Hasser, "Force-feedback improves performance for steering and combined steering-targeting tasks", em *Proceedings of the SIGCHI conference on Human factors in computing systems*, 2000, pp. 423–429.
- [91] K. Li, R. Bacher, S. Schmidt, W. Leemans e F. Steinicke, "Reality fusion: Robust real-time immersive mobile robot teleoperation with volumetric visual data fusion", em *2024 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, IEEE, 2024, pp. 8982–8989.
- [92] C. González, J. E. Solanes, A. Munoz, L. Gracia, V. Girbés-Juan e J. Tornero, "Advanced teleoperation and control system for industrial robots based on augmented virtuality and haptic feedback", *Journal of Manufacturing Systems*, v. 59, pp. 283–298, 2021.

- [93] T. Kamijo, C. C. Beltran-Hernandez e M. Hamaya, "Learning variable compliance control from a few demonstrations for bimanual robot with haptic feedback teleoperation system", em *2024 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, IEEE, 2024, pp. 12 663–12 670.
- [94] K. Wan, C. Li, F.-S. Lo e P. Zheng, "A virtual reality-based immersive teleoperation system for remote human-robot collaborative manufacturing", *Manufacturing Letters*, v. 41, pp. 43–50, 2024.
- [95] P. Farajiparvar, H. Ying e A. Pandya, "A brief survey of telerobotic time delay mitigation", *Frontiers in Robotics and AI*, v. 7, p. 578 805, 2020.
- [96] M. Labbé e F. Michaud, "RTAB-Map as an open-source lidar and visual simultaneous localization and mapping library for large-scale and long-term online operation", *Journal of field robotics*, v. 36, n. 2, pp. 416–446, 2019.
- [97] Y. D. Yasuda, L. E. G. Martins e F. A. Cappabianco, "Autonomous visual navigation for mobile robots: A systematic literature review", *ACM Computing Surveys (CSUR)*, v. 53, n. 1, pp. 1–34, 2020.
- [98] Y. Yang, D. Tang, D. Wang, W. Song, J. Wang e M. Fu, "Multi-camera visual SLAM for off-road navigation", *Robotics and Autonomous Systems*, v. 128, p. 103 505, 2020.
- [99] A. M. Abbad, I. Haouala, A. Raisov e R. Benkredda, "Low cost mobile navigation using 2d-slam in complex environments", *Google Scholar*, 2023.
- [100] C. Tian, H. Liu, Z. Liu, H. Li e Y. Wang, "Research on multi-sensor fusion SLAM algorithm based on improved gmapping", *IEEE Access*, v. 11, pp. 13 690–13 703, 2023.
- [101] S. Sandfuchs, M. P. Heimbach, J. Weber e M. Schmidt, "Conversion of depth images into planar laserscans considering obstacle height for collision free 2D robot navigation", em *2021 European Conference on Mobile Robots (ECMR)*, IEEE, 2021, pp. 1–6.
- [102] A. Tourani, H. Bavle, J. L. Sanchez-Lopez e H. Voos, "Visual slam: What are the current trends and what to expect?", *Sensors*, v. 22, n. 23, p. 9297, 2022.
- [103] C. Debeunne e D. Vivet, "A review of visual-LiDAR fusion based simultaneous localization and mapping", *Sensors*, v. 20, n. 7, p. 2068, 2020.
- [104] R. Mur-Artal e J. D. Tardós, "Orb-slam2: An open-source slam system for monocular, stereo, and rgb-d cameras", *IEEE transactions on robotics*, v. 33, n. 5, pp. 1255–1262, 2017.

- [105] T. Zhang, P. Wang, F. Zha, W. Guo e M. Li, “RGBD Navigation: A 2D navigation framework for visual SLAM with pose compensation”, em *2023 IEEE International Conference on Real-time Computing and Robotics (RCAR)*, IEEE, 2023, pp. 644–649.
- [106] D. Scaramuzza e F. Fraundorfer, “Visual odometry [tutorial]”, *IEEE robotics & automation magazine*, v. 18, n. 4, pp. 80–92, 2011.
- [107] K. De Silva, B. Cooray, J. Chinthaka, P. Kumara e S. Sooriyaarachchi, “Comparative analysis of octomap and rtabmap for multi-robot disaster site mapping”, em *2018 18th International Conference on Advances in ICT for Emerging Regions (ICTer)*, IEEE, 2018, pp. 433–438.
- [108] J. Gill, *Setup and Configuration of the Navigation Stack on a Robot*, 2018. endereço: <http://wiki.ros.org/navigation/Tutorials/RobotSetup/>.
- [109] A. Hornung, K. M. Wurm, M. Bennewitz, C. Stachniss e W. Burgard, “OctoMap: An efficient probabilistic 3D mapping framework based on octrees”, *Autonomous robots*, v. 34, n. 3, pp. 189–206, 2013.
- [110] C. L. Ciofi-Silva et al., “Workload assessment: cross-cultural adaptation, content validity and instrument reliability”, *Revista brasileira de enfermagem*, v. 76, n. 3, e20220556, 2023.
- [111] J. J. G. Cardenas, X. Hei e A. Tapus, “Exploring cognitive load dynamics in human-machine interaction for teleoperation: A user-centric perspective on remote operation system design”, em *2024 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, IEEE, 2024, pp. 12 204–12 211.
- [112] G. R. Silva, J. C. Donat, M. M. Rigoli, F. R. de Oliveira e C. H. Kristensen, “A questionnaire for measuring presence in virtual environments: factor analysis of the presence questionnaire and adaptation into Brazilian Portuguese”, *Virtual Reality*, v. 20, n. 4, pp. 237–242, 2016.
- [113] J. Stevens et al., *Applied multivariate statistics for the social sciences*. Lawrence erlbaum associates Mahwah, NJ, 2002, vol. 4.
- [114] E. G. Palmeira, A. Campos, Í. A. Moraes, A. G. de Siqueira e M. G. Ferreira, “Quantifying the ‘Gorilla Arm’ Effect in a Virtual Reality Text Entry Task via Ray-Casting: A Preliminary Single-Subject Study”, em *Proceedings of the 25th Symposium on Virtual and Augmented Reality*, 2023, pp. 274–278.