SAPIENZA
UNIVERSITÀ DI ROMA

A thesis submitted in fulfillment of the requirements for the degree of

Ph.D. in Engineering in Computer Science

Dipartimento di Ingegneria informatica automatica e gestionale "Antonio Ruberti" (DIAG)

Multimodal Communication for Enhancing Human-Robot Interaction: Virtual Simulations to Real Robots

Advisor:
Prof. Daniele Nardi

Co-Advisor:
Prof. Francesco Leotta

Author:
Sandeep Reddy Sabbella
1826732

Academic Year(s) 2023 - 2024 (XXXVII Cycle)
La borsa di dottorato è stata cofinanziata con risorse del Programma Operativo Nazionale Ricerca e Innovazione 2014-2020, risorse FSE REACT-EU
Azione IV.4 "Dottorati e contratti di ricerca su tematiche dell'innovazione" e Azione IV.5 "Dottorati su tematiche Green"

Doctoral Thesis

Multimodal Communication for Enhancing Human-Robot Interaction: Virtual Simulations to Real Robots

Department of Computer, Control and Management Engineering
"Antonio Ruberti" (DIAG)

A thesis submitted in fulfillment of the requirements for the degree of

Ph.D. in Engineering in Computer Science
(XXXVII cycle)

Author
Sandeep Reddy Sabbella
ID number 1826732

Advisor
Prof. Daniele Nardi

Co-Advisor
Prof. Francesco Leotta

Academic Year 2023/2024

La borsa di dottorato è stata cofinanziata con risorse del
Programma Operativo Nazionale Ricerca e Innovazione 2014-2020, risorse FSE REACT-EU
Azione IV.4 “Dottorati e contratti di ricerca su tematiche dell’innovazione"
e Azione IV.5 “Dottorati su tematiche Green"

Thesis defended on May 19, 2025
in front of a Board of Examiners composed by:

Prof. Chiara Ghidini (Libera Università di Bolzano) (chairman)
Prof. Federica Mandreoli (Università degli Studi di Modena e Reggio Emilia)
Prof. Ivan Serina (Università degli Studi di Brescia)

PhD thesis reviewed by:

Prof. Leon Bodenhagen (University of Southern Denmark)
Prof. Alessandra Rossi (University of Naples Federico II)

Multimodal Communication for Enhancing Human-Robot Interaction: Virtual Simulations to Real Robots
PhD thesis. Sapienza University of Rome

This thesis has been typeset by LaTeX and the Sapthesis class.

Author's email: sabbella@diag.uniroma1.it

Abstract

Human-robot interaction (HRI) is a rapidly evolving domain focused on the interaction between humans and robots, exploring robotic systems' design, functionality, and social implications in various environments. Virtual Reality (VR) has emerged as a valuable tool for evaluating HRI solutions before real-world deployment, ensuring safety and scalability. The main objective of this thesis is to present a multimodal interaction framework integrating speech and gesture recognition to enhance collaboration between humans and robots in precision agriculture, particularly in table-grape vineyards under the CANOPIES project, as well as in broader contexts of indoor and outdoor logistics.

In collaborative robotics, where human-robot collaboration (HRC) is essential, multimodal communication between humans and robots is crucial during each interaction. To address this challenge, building on a categorization of the information content and speech act classification in the context of HRI in shared environments, Speech and Gesture recognition pipelines were designed and integrated into HRI architecture for cobots. Leveraging virtual reality (VR) as a testbed, the work generates synthetic datasets to train robust gesture and speech recognition models, overcoming the scarcity of real-world data in agricultural contexts. The framework is empirically validated through VR-based user studies and field experiments, demonstrating improved communication reliability in noisy vineyard environments and reduced task completion times. Notably, the system emphasizes modularity, allowing interchangeable components (e.g., pose estimators and speech classifiers) to adapt to dynamic tasks. Key contributions include (i) a standardized gesture taxonomy tailored to agricultural workflows, (ii) open-source datasets produced from both real and synthetic sources, (iii) a synthetic data generation pipeline for pose estimation, and (iv) a multimodal communication architecture augmented by large language models (LLMs) for contextual reasoning using limited computational capacity in agricultural logistics. By bridging virtual simulations and real-world deployment, this research advances human-robot collaboration in precision agriculture, offering interactive solutions for harvesting, pruning, and logistics tasks. The findings underscore the potential of multimodal HRI and immersive technologies to address collaboration between human expertise and robots and enhance safety and efficiency across both indoor and outdoor collaborative environments.

Keywords: Human-Robot Interaction (HRI), Human-Robot Collaboration (HRC), Virtual Reality (VR), Synthetic Data Generation, Multimodal Communication, Gesture Recognition, User Evaluation, Large Language Models (LLMs), Precision Agriculture, Collaborative Robotics.

Abstract

L'interazione uomo-robot (HRI) è un ambito in rapida evoluzione che si concentra sull'interazione tra esseri umani e robot, esplorando la progettazione, la funzionalità e le implicazioni sociali dei sistemi robotici in diversi contesti. La realtà virtuale (VR) si è affermata come uno strumento prezioso per valutare le soluzioni HRI prima della loro implementazione nel mondo reale, garantendo sicurezza e scalabilità. L'obiettivo principale di questa tesi è presentare un framework di interazione multimodale che integri il riconoscimento vocale e gestuale per migliorare la collaborazione tra esseri umani e robot nell'agricoltura di precisione, in particolare nei vigneti di uva da tavola nell'ambito del progetto CANOPIES, così come in contesti più ampi di logistica interna ed esterna.

Nella robotica collaborativa, in cui la collaborazione uomo-robot (HRC) è essenziale, la comunicazione multimodale tra esseri umani e robot riveste un ruolo cruciale in ogni interazione. Per affrontare questa sfida, basandosi su una categorizzazione del contenuto informativo e sulla classificazione degli atti linguistici nel contesto dell'HRI in ambienti condivisi, sono state progettate pipeline di riconoscimento vocale e gestuale, successivamente integrate nell'architettura HRI per i cobot. L'utilizzo della realtà virtuale (VR) come banco di prova consente di generare set di dati sintetici per l'addestramento di modelli robusti di riconoscimento di gesti e parlato, superando così la scarsità di dati reali nei contesti agricoli. Il framework è stato validato empiricamente attraverso studi utente basati sulla VR ed esperimenti sul campo, dimostrando una maggiore affidabilità della comunicazione in ambienti rumorosi come i vigneti e una riduzione dei tempi di completamento delle attività. In particolare, il sistema enfatizza la modularità, permettendo ai componenti intercambiabili (ad esempio, stimatori di pose e classificatori del parlato) di adattarsi a compiti dinamici. I principali contributi della ricerca includono: (i) una tassonomia standardizzata dei gesti adattata ai flussi di lavoro agricoli, (ii) set di dati open-source generati da fonti sia reali che sintetiche, (iii) una pipeline per la generazione di dati sintetici finalizzata alla stima delle pose e (iv) un'architettura di comunicazione multimodale potenziata da modelli linguistici di grandi dimensioni (LLM) per il ragionamento contestuale, con un utilizzo limitato delle risorse computazionali nella logistica agricola. Collegando simulazioni virtuali e implementazioni nel mondo reale, questa ricerca promuove la collaborazione uomo-robot nell'agricoltura di precisione, offrendo soluzioni interattive per attività di raccolta, potatura e logistica. I risultati evidenziano il potenziale dell'HRI multimodale e delle tecnologie immersive nel favorire la collaborazione tra esseri umani e robot, migliorando al contempo sicurezza ed efficienza in ambienti collaborativi sia interni che esterni.

Keywords: Interazione uomo-robot (HRI), Collaborazione uomo-robot (HRC), Realtà virtuale (VR), Generazione di dati sintetici, Comunicazione multimodale, Riconoscimento dei gesti, Valutazione utente, Modelli linguistici di grandi dimensioni (LLM), Agricoltura di precisione, Robotica collaborativa.

Acknowledgments

I would like to express my sincere appreciation to my supervisor, ***Professor Daniele Nardi****, for his exceptional leadership and invaluable guidance throughout my research. His support allowed me to contribute to various aspects of Human-Robot Interaction within the scope of the important European research project, CANOPIES. I also extend my deepest thanks to my co-supervisor,* ***Professor Francesco Leotta****, for his continuous encouragement and support throughout this challenging yet rewarding journey. My gratitude also goes to* ***Professor Thomas A. Ciarfuglia*** *for his insightful suggestions and thoughtful discussions, which have not only enhanced my research but also helped shape my development as a person.*

I am profoundly grateful to ***my family*** *for their unwavering support, the countless long conversations over the phone, and for always standing by me throughout this demanding yet thrilling journey. Though miles apart, you are always in my heart. I am incredibly fortunate to have my friend* ***Abdallah Kobresli*** *by my side, who has been more than a brother, providing steadfast support through my highest highs and lowest lows. What an extraordinary journey we've shared so far!*

I deeply appreciate the collaboration of my colleagues, particularly ***Sara Kaszuba****, whose contributions to the speech communication modality were indispensable. Her support made this experience not only professionally enriching but personally rewarding. A special acknowledgement goes to* ***Alexia T. Salomons*** *for her expertise and assistance in conducting a user study for multimodal communication evaluations as part of her Master's Thesis. I thoroughly enjoyed our time together and the conversations we shared. My thanks also go to* ***Julien Caposiena*** *and* ***Ziba Khani*** *for their contributions to the speech pipeline and the design of light and sound signals, respectively, during their time at Sapienza. I am grateful to* ***Vincenzo Suriani****, a friend and colleague, for his unwavering support with any ad-hoc requests. I would also like to thank all the members and partners of the* ***CANOPIES consortium*** *for their vital roles in enabling my research.*

I wish to express my special thanks to ***Felix Gorbatsevich****, CEO of PaleBlue, for facilitating my research exchange period in Stavanger, Norway. The time spent with the team at PaleBlue, exchanging ideas over tea and coffee, was invaluable. Similarly, I am thankful to* ***Vasily Morzhakov****, CEO of RemBrain, for allowing me to continue my research on Human-Robot Interaction in the context of Logistic Delivery robots. I greatly appreciate the ideas and discussions we shared, as well as the unforgettable trip to visit the Fjords. My sincere thanks also go to* ***Dmitry Sannikov*** *for his insightful suggestions and for sharing hiking trips and other adventures during this time.*

I am deeply thankful to the participants of the experiments for generously contributing their time and sharing valuable insights during the user studies. Your

involvement has been crucial to the success of my research and to the development of Human-Robot Interaction systems.

This acknowledgment would be incomplete without mentioning my dear friends and brothers, ***Ionuts M. Motoi*** *and* ***Leonardo Saraceni****. The time I spent with both of you during this period was truly unforgettable. Thank you for the incredible memories we created together.*

I would also like to extend my gratitude to all the remarkable individuals I met at various summer schools, conferences, and workshops, with whom I shared unforgettable experiences, knowledge, and common goals. I am thankful to all my friends and everyone who contributed, both in significant and seemingly small ways, to shaping who I am today and to the success of this journey and beyond!

Finally, I wish to acknowledge all the PhD students and researchers who are actively advancing Human-Robot Interaction in diverse fields, ensuring safer, more robust, and effective communication.

With love and gratitude, I dedicate this PhD thesis to my family.
Sandeep Reddy Sabbella, Rome, February 2025

Contents

List of Figures

List of Tables

Chapter 1

Introduction

Human-robot interaction (HRI) is an emerging field focused on the interaction between humans and robots, exploring robotic systems' design, functionality, and social implications in various environments. As robots increasingly integrate into everyday life, from healthcare care [157] to education and customer service [261], the study of HRI has gained notable significance due to its potential to improve efficiency, safety, and user experience. HRI is a multidisciplinary domain, incorporating perspectives from human-computer interaction, artificial intelligence, robotics, natural language understanding, design, and psychology. As artificial intelligence advances, research in HRI addresses the physical safety of interactions and the social appropriateness of robot behaviour, often shaped by cultural criteria. A key prerequisite for effective Human-Robot Interaction (HRI) is establishing a shared understanding of the operational environment and the objects within it by humans and robots. The embodiment of a robot places physical constraints on how it can sense and act in the world, but it also represents an opportunity for interaction with people. The physical makeup of the robot elicits people to respond in a way similar to that in which they interact with other people. The robots' human-likeness enables humans to use their existing experience of human-human interaction in human-robot interaction [271]. These experiences can help construct an interaction but can also lead to frustration if the robot cannot meet user expectations [28].

Robots are classified based on design, functionality, and applications. *Industrial robots* perform repetitive tasks in manufacturing using predefined scripts. *Service robots* assist humans in dynamic environments like hospitality and healthcare, and can be either static or mobile. *Autonomous robots* use AI to adapt and perform tasks independently, while humanoid robots mimic human behaviors and expand into sectors like healthcare. *Collaborative robots* (cobots) work alongside humans, equipped with safety features to operate near them. Recent research in *collaborative HRI* explores how humans and robots work together on shared tasks [95]. HRI applications now span industrial, medical, agricultural, and space sectors. Studies show that human-robot collaboration reduces task time and increases production [137], improving overall system efficiency [91]. Effective collaboration relies on verbal (speech) and non-verbal cues (gestures, visual, and tactile feedback), with multimodal communication enhancing efficiency.

Collaborative environments come in different configurations, such as one human-one robot (diadic), two humans-one robot or one human-two robots (triadic), and multiple humans - multiple robots(teams or groups) [100, 7]. Irrespective of the configuration, Cobots have to share space with humans. Human safety is given utmost priority in designing HRI architectures and solutions for cobot problem space. As human safety is at play, there is increased interest in using virtual reality (VR) to create simulated settings that closely mirror the real world [276]. Such VR environments aid in evaluating algorithms and approaches, while also serving as a platform for training personnel. Despite advancements, developing and validating HRI solutions still lacks rapid and efficient methods, often relying on complex physical setups where humans interact with robots [68], resulting in costly and time-consuming trial-and-error cycles. Under these circumstances, VR offers a safe, cost-effective platform [151] for iterative testing and observation of cooperative tasks in a virtual space. Since effective human–android interaction involves detailed scrutiny of communication modes, sensor positioning, human-robot distance, and the specific application domain, VR proves advantageous for verifying HRI methodologies [123], assessing potentially risky actions [167, 117], and training users in specialised skills [163, 198, 258]. It further facilitates running multiple experiments within condensed time frames.

1.1 What is Multimodality in HRI?

Humans perceive the world multimodally, integrating visual, auditory, tactile, olfactory, and gustatory inputs. Even though all these senses are from different sense organs, they provide coordinated information and enable humans to make informed decisions. *Multimodality* refers to the integration of multiple communication channels—such as speech, gestures, visual cues, and body language—to convey information and facilitate interaction. In Human-Robot Interaction (HRI), multimodality is a cornerstone for effective communication, particularly in dynamic, real-world environments where a single communication mode may not capture the complexity of human intentions or context [257]. While humans naturally combine verbal speech, facial expressions, gestures, and other non-verbal cues to convey meaning, robots traditionally rely on a single modality, such as speech or visual cues, which can be limiting, especially in noisy or visually complex environments [118].

In outdoor or noisy settings, like agriculture, speech recognition may be impaired, making it essential for robots to rely on other modalities, such as gestures or visual cues, to understand human communication [232]. Similarly, in more structured indoor environments, multimodal communication allows robots to better interpret user intentions by integrating speech, body language, and sensory cues like touch or proximity. This enables robots to adapt their responses according to context, improving collaboration and task efficiency. For example, in an agricultural context, when verbal communication is hindered by background noise, a robot can use gestures or visual cues for basic communication, while leveraging speech for more specific or complex instructions [118]. The goal of multimodal interaction in HRI is to create systems that are not only more robust and flexible but also more intuitive for

human users. By combining multiple interaction modes, robots can overcome the limitations inherent in relying on a single modality and interact more naturally and effectively with humans, making the interaction richer and adaptive. This thesis emphasises the integration of both speech and gesture recognition systems, forming a multimodal framework aimed at enhancing the efficiency, safety, and adaptability of collaborative tasks in both indoor and outdoor environments.

The remainder of this chapter is organized as follows: Section 1.2 discusses HRI in agricultural robots. Section 1.3 presents the primary objectives and goals of the thesis, while Section 1.4 outlines the research challenges addressed, the contributions made to overcome these challenges, and a list of publications resulting from this work. Finally, Section 1.5 provides an overview of the overall structure of the thesis.

1.2 AgRobotics

Agriculture has employed specialised machinery and automated systems for decades to meet the growing demand for food production. Typically, one individual operates each machine. As global food needs escalate and skilled labour becomes increasingly scarce, production is shifting toward alternative solutions. Robotic systems in agriculture, or agrobotics, encompass a spectrum of tasks—including seeding, harvesting, weeding, and crop monitoring- to enhance efficiency, reduce labour costs, and optimise resources. By harnessing sensors, computer vision, and machine learning, these robots can identify plant health, guide targeted pesticide or nutrient application, and carefully harvest high-value crops such as strawberries and tomatoes [21, 280]. For instance, harvesting robots for high-value crops like tomatoes or strawberries use imaging sensors to identify ripe produce and employ gentle grippers to avoid damage [21]. Autonomous ground vehicles can navigate between rows of plants, mapping weeds for targeted herbicide applications, whereas drones conduct aerial surveys to assess plant vigour and detect disease hotspots. They should operate on both scenarios; they must navigate unpredictable terrains and varying crop canopies [229]-and controlled environments like greenhouses or vertical farms, which enable more predictable conditions but demand delicate end-effectors and sophisticated handling of perishable produce [223]. These technological advances address labour shortages and boost consistency in farm operations, reflecting a trend toward greater automation in modern agriculture [30].

Despite their potential, agricultural robots face significant challenges in reliability, adaptability, and cost-effectiveness, especially when handling soft, non-uniform crops under diverse environmental conditions. As a result, this field increasingly focuses on incorporating human-centred design principles, leading to the rise of cobots that emphasise safe, intuitive, and flexible interaction with human operators. This human-robot collaboration enhances scalability in demanding agricultural contexts and helps bridge the gap between full automation and human expertise. Ultimately, overcoming these technical, economic, and design hurdles will be pivotal for maximising the benefits of agrobotics across both outdoor fields and controlled-environment facilities.

Harvesting table grapes is a delicate task that demands careful handling to preserve the quality of the fruit. Vineyards featuring expansive canopies and trellis systems create a distinctive setting requiring precise harvesting methods, offering opportunities and challenges for implementing intelligent farming solutions. Tools ranging from robotic harvesters and pruning devices to autonomous sensing and monitoring systems can elevate precision agriculture by complementing human labour. Indeed, when seamlessly integrated into existing farming practices, robotic innovations can help reduce labour costs, minimise environmental impact, and consistently deliver premium-quality table grapes to the market.

1.2.1 CANOPIES: Collaborative Human-Robot Interaction in Precision Agriculture

The CANOPIES project[1] is a Horizon 2020-funded European research initiative aimed at transforming human-robot collaboration in the domain of precision agriculture, particularly for permanent crops such as table grapes. It envisions seamless cooperation between human workers and autonomous robotic systems to perform labour-intensive tasks, including harvesting and pruning, within complex and dynamic outdoor vineyard environments. To realise this vision, the project focuses on the development of advanced methodologies in Human-Robot Interaction (HRI), Human-Robot Collaboration (HRC), and Multi-Robot Coordination (MRC), enabling safe, efficient, and intuitive joint task execution.

The research emphasises safe human-robot coexistence in shared workspaces—whether involving physical contact or not—by empowering robots with capabilities to anticipate human motion, particularly of the torso and arms. Additionally, the project promotes natural and efficient communication between humans and robots, fostering mutual awareness of intentions, which is critical for smooth and coordinated task performance. CANOPIES also integrates human-like bimanual manipulation to enhance robot dexterity and enable intuitive collaboration. Programming of robotic behaviours is achieved through Learning by Demonstration (LbD), allowing non-expert users to train robots via demonstration. Furthermore, adaptive learning mechanisms are implemented using human-in-the-loop approaches to manage unexpected events. At a broader systems level, the project explores robust coordination strategies among multiple robots working in tandem on shared tasks. Collectively, these efforts aim to establish a new paradigm in collaborative robotics that can support sustainable, safe, and productive agricultural practices.

Building on these technological advancements, this thesis investigates effective HRI strategies in outdoor collaborative settings through empirical experimentation. Evaluations are conducted in the challenging context of table-grape vineyards, where close coordination between human workers and robots is essential. In parallel, the research extends to logistics and delivery scenarios, analysing the performance of ground robots in both indoor and outdoor environments. The overarching objective is to develop functional and context-aware HRC methodologies for use in agriculture

[1] `https://canopies.inf.uniroma3.it`

and related sectors by leveraging state-of-the-art robotics, artificial intelligence, and interaction design principles.

Within the CANOPIES framework, two primary categories of ground robots are deployed (see Section 9.4):

- **Farming Robots (FR)** are designed to collaborate directly with human operators in agronomic tasks such as grape harvesting and vine pruning.
- **Logistic Robots (LR)** support operations by managing the transport and replacement of boxes containing harvested or pruned material, and by facilitating delivery-related tasks in both indoor and outdoor contexts.

The farming robot is equipped with two anthropomorphic arms, allowing it to mimic human-like bimanual actions during pruning and harvesting. It includes a dedicated platform with removable boxes for collecting produce. In contrast, the logistics robot is engineered with multiple modular containers and a box-exchange mechanism that enables efficient swapping between full and empty boxes during operation. These design features support close human-robot coordination in two critical vineyard tasks: pruning, which ensures long-term vineyard health, and harvesting, which requires precise collection of ripe grape clusters. The humanoid design of the farming robot enhances task fluidity and safety in shared spaces. To fully harness the potential of HRI and HRC in such environments, the establishment of robust, real-time communication protocols is essential. These systems must ensure operational safety, protect crop integrity, and support effective task execution. Consequently, continuous innovations in both hardware and software are vital to overcoming existing communication and coordination challenges and driving the future of intelligent, user-centric agricultural robotics.

1.3 Objectives of the Thesis

The primary objective of this thesis is to develop a robust, efficient, and reliable interaction framework that elevates non-verbal cues and gestures to a central role in Human-Robot Interaction (HRI) within collaborative settings, while simultaneously recognising speech as a key communication channel. The proposed system effectively tackles the challenges posed by noisy or visually demanding environments by emphasising non-verbal elements—such as hand gestures, body poses, and other expressive cues. At the same time, spoken commands remain essential for conveying detailed instructions and facilitating natural collaboration, particularly when precision and clarity are paramount. Ultimately, speech and gestures work together within a multimodal pipeline, serving as complementary and supplementary forms of communication that enhance the overall efficiency and reliability of HRI.

A core insight derived from analysing frequently exchanged information in outdoor scenarios is that no single modality can address every communicative challenge. Consequently, this thesis targets the interplay among gestures, non-verbal signals, and verbal utterances to strengthen the robot's comprehension of human input, while

also enhancing safety by minimising misinterpretations that can arise from relying solely on speech, particularly in noisy or visually obstructed conditions. Although much of this research focuses on agricultural fields as primary outdoor environments, these multimodal interaction strategies equally apply to indoor settings such as warehouses, hospitals, or service sectors—where cluttered layouts, varied lighting, or acoustic constraints may pose additional challenges. By integrating both speech and embodied signals, robots can better navigate shared spaces, interpret user intentions, and seamlessly coordinate tasks with human teammates. Moreover, employing this consolidated approach in controlled indoor simulations allows for comprehensive testing of hardware and software components, ensuring the interaction framework remains robust and adaptable across a broad spectrum of operational contexts.

To refine and validate these interaction strategies, Virtual Reality (VR) simulations serve as a critical test bed for preliminary experimentation. VR enables the creation of synthetic data given the input modality ranging from Audio, 2D, character and 3D model data, allowing virtually unlimited data to train and test the algorithms for controlled and safe user studies. These synthetic datasets also support pose estimation and gesture recognition research, alleviating data scarcity issues that frequently hinder real-world data collection [250]. By combining both speech and non-verbal cues in a multimodal pipeline and rigorously evaluating this pipeline in VR environments, the thesis demonstrates how an HRI system can reduce ambiguity and miscommunication, leading to more fluid and resilient Human-Robot Collaboration (HRC).

Overall, this thesis presents a cohesive multimodal framework in which gestures, body language, and vocal commands work together to streamline communication between humans and robots. Through this integrative approach, the proposed system aims to advance HRI in tasks ranging from vineyard maintenance to logistics, ultimately paving the way for safer, more efficient, and context-adaptive collaboration in diverse indoor and outdoor settings.

1.4 Research Challenges in HRI and Contributions

Human-Robot Interaction (HRI) poses unique challenges depending on whether the environment is controlled (indoor) or dynamic (outdoor). Indoor environments, such as manufacturing and warehouse settings, offer stable conditions, including reliable lighting and connectivity. However, even in these controlled settings, HRI systems must be designed to adapt to rapid changes in workflows, roles, and tasks. In contrast, outdoor environments—such as agriculture or search and rescue operations—introduce further complexities, including fluctuating weather, uneven terrain, ambient noise, and limited connectivity. These environmental variables complicate robot operation and disrupt essential communication channels, such as speech recognition and visual perception, both of which are crucial for real-time decision-making and task coordination.

Table 1.1. Mindmap: Research Challenges to Contributions

Research Challenge	**Contributions Addressing the Challenge**
RC1: Categorisation and Classification of Communication in Outdoor HRI	C1: Innovative categorisation of content information C2: Classification of delivery modalities for non-verbal cues C3: Spoken utterance understanding system for complex communication C4: Speech act recognition using intonation cues
RC2: Data Scarcity for Outdoor Multimodal HRI	C5: English and Italian transcribed speech dataset C10: Gestural dataset at multiple distances with synthetic augmentation C13: Hybrid dataset combining synthetic and real-world data
RC3: Simulation-Based Evaluation and VR Integration	C7: VR-based user study comparing IVE and NIVE C8: Virtual simulation environment for multimodal data synthesis C11: 3D object generation pipeline for cost-efficient simulations C12: Pose estimation benchmark using synthetic VR data
RC4: Robust Gesture and Speech Integration for Collaboration	C6: Speech-based pipeline using content information and frame semantics C9: Definition of feasible gestures for outdoor HRI C14: Gesture recognition pipeline supporting virtual, real, and hybrid datasets C15: Gesture recognition for small logistic robots in indoor/outdoor use C16: VR-based study on multimodal communication feasibility C17: Modular architecture for spoken and gestural input integration
RC5: Context-Aware Multimodal HRI with LLMs	C18: Integration of Large Language Models for context-aware interaction C19: Reformulation of HRI framework using State Machines for task handling
RC6: Deployment on Resource-Constrained Edge Devices	C20: Multimodal HRI deployment on Jetson Orin for real-time interaction

Fluctuating lighting conditions in outdoor environments, for instance, pose significant challenges for visual systems, while ambient noise from machinery or natural elements like wind or foliage further impedes speech recognition. Furthermore, the lack of reliable connectivity in outdoor settings restricts real-time data exchange between robots and human operators, which is essential for synchronising actions in collaborative tasks. These challenges are compounded by data scarcity, particularly in agricultural settings, where collecting diverse and representative datasets for training machine learning models is labour-intensive and costly. Additionally, outdoor robots often operate with limited computational resources, which restricts their ability to process complex tasks under time-sensitive conditions.

Addressing the challenges of Human-Robot Interaction (HRI) requires innovative solutions that are adaptable to both indoor and outdoor environments. This thesis identifies key challenges specific to these domains and provides solutions aimed at enhancing the effectiveness and reliability of HRI. This section outlines the research challenges this thesis aims to address and discusses their impact on HRI systems. Proactively identifying and tackling these hurdles makes refining hardware and software components possible, facilitating more natural, efficient, and robust Multimodal Human-Robot Interaction (MHRI). Although the primary focus lies in precision agriculture, the resulting strategies are adaptable to various context-sensitive applications, including diverse indoor and outdoor robotic collaborations. The Table 1.1 summarised the following challenges and contributions.

- **RC1: Categorisation and Classification of Communication in Outdoor HRI**:
  A fundamental challenge in collaborative outdoor HRI is the lack of systematic frameworks to categorise content information and associated delivery modalities. The absence of structured taxonomies for interpreting spoken language and non-verbal cues (e.g., gestures, prosody) hinders effective communication between humans and robots in dynamic agricultural settings.

- **RC2: Data Scarcity for Outdoor Multimodal HRI**:
  Addressing the scarcity of outdoor collaborative HRI datasets is essential for evaluating algorithm performance and system efficacy. The chances of speech misrecognition and spoken utterances varying with different accents are high, and the lack of speech commands in the context of agriculture makes it more challenging to have a proper interaction. Pose data and gesture recognition algorithms were not applied to the agricultural context. Defining and generalising the gestures and collecting data across variable environmental conditions and user behaviours remains a critical bottleneck.

- **RC3: Simulation-Based Evaluation and VR Integration**:
  Conducting field experiments in real agricultural environments poses logistical and safety constraints. Therefore, selecting the appropriate virtual reality experience—Immersive (IVE) or Non-Immersive (NIVE)—for simulating outdoor HRI scenarios becomes essential. The challenge lies in ensuring that virtual simulations provide realistic, transferable insights to real-world deployments.

- **RC4: Robust Gesture and Speech Integration for Collaboration**:
  Seamless collaboration in HRI requires effective multimodal integration of verbal and non-verbal communication. Challenges include designing robust pipelines that process spoken language, gesture input, and contextual awareness simultaneously, particularly in noisy, variable, and unpredictable outdoor environments such as vineyards.

- **RC5: Context-Aware Multimodal HRI with LLMs**:
  Integrating Large Language Models (LLMs) into HRI frameworks introduces challenges related to latency, contextual understanding, and computational complexity. Outdoor collaborative robots must manage multiple modalities in real-time while maintaining high responsiveness, requiring LLMs to be modular, efficient, and aware of both physical and linguistic context.

- **RC6: Deployment on Resource-Constrained Edge Devices**:
  Deploying advanced HRI systems in field conditions necessitates real-time performance on resource-limited edge devices. Challenges include low-latency processing, minimal power consumption, and reliable operation without dependence on cloud connectivity. Each module in the pipeline must be optimised for hardware constraints typical of mobile agricultural robots.

In response to the key challenges outlined in Section 1.4, the following presents the research contributions made to address those challenges. Each contribution targets specific aspects of multimodal Human-Robot Interaction (HRI), advancing both theoretical understanding and practical implementations. The preliminary identification of communication categories (Speech Acts), which was crucial for the methodology (refer to subsection 3.1.1), the analysis of vocal utterance data, and the development of the Spoken Human-Robot Interaction pipeline without LLMs (refer to Chapter 6) was a significant effort and key collaboration from my colleague Sara Kaszuba. The user study on Immersive and Non-Immersive Virtual Experiences (refer to section 4.3, which was also a collaborative effort, saw my contribution in system development.

C1. Developed an innovative categorisation system for content information in the context of Human-Robot Interaction (HRI) within collaborative outdoor environments.

C2. Established an original classification of delivery modalities linked to content information, tailored for HRI involving non-verbal cues in outdoor teamwork scenarios.

C3. Designed an accurate content information recognition system to enhance the robot's understanding of spoken utterances from human teammates, enabling it to handle complex statements involving multiple communication acts.

C4. Developed an enhanced speech act recognition system for collaborative robotics that utilises intonation analysis to reduce ambiguity and improve the robot's ability to differentiate between user requests and informational utterances by analysing pitch variations in vocal expressions.

C5. Created a transcribed textual speech dataset in English and Italian from recorded vocal utterances to facilitate HRI research in collaborative outdoor scenarios, specifically focusing on table-grape vineyards.

C6. Established a speech-based pipeline for collaborative robotics that leverages content information identification and frame semantics.

C7. Conducted a user study to determine the appropriate type of VR experience—distinguishing between Immersive Virtual Experience (IVE) and Non-Immersive Virtual Experience (NIVE)—for validating approaches and experimenting with novel HRI solutions.

C8. Presented a virtual simulation environment to synthesise and extract several types of data.

C9. Defined an innovative set of feasible gestures for interacting with ground robots in outdoor collaborative environments.

C10. Acquired a gestural dataset from field participants performing gestures at four different distances and developed a synthetic data pipeline using virtual avatars in simulated environments to enhance gestural interaction in collaborative scenarios reducing Sim-to-Real gap.

C11. Extended an image-to-3D model pipeline to generate 3D objects from text and image descriptions for use in virtual simulations, significantly reducing the cost of creating virtual environments.

C12. Established pose estimation benchmark on synthetic data using the virtual simulator for validating pose datasets.

C13. Integrated synthetic and real data to create a hybrid dataset, and presented the results on the performance of algorithms trained with this data in real-world scenarios.

C14. A novel and adaptable gesture recognition pipeline capable of training and evaluating virtual, real, or combined datasets, incorporating diverse pose estimation and Convolutional Neural Network (CNN) algorithms.

C15. Extended gesture recognition algorithms to detect hand-based gestures in small logistic robots for both indoor and outdoor scenarios.

C16. Assessed the feasibility of multimodal communication in collaborative contexts by conducting VR-based user studies.

C17. Developed a multimodal modular architecture that effectively recognises and processes both spoken and gestural information from human teammates, facilitating the integration or substitution of specific components and enhancing the overall capabilities of HRI systems in outdoor collaborative environments.

C18. Extended the defined multimodal architecture to incorporate Large Language Models (LLMs) to enhance existing accuracy and ensure modularity for introducing context-awareness in indoor and outdoor collaborative robots (cobots).

C19. Reformulated the HRI framework to incorporate State Machines, enhancing task management and decision-making in complex scenarios.

C20. Deployed the multimodal HRI solution on edge devices (e.g., Jetson Orin) for logistic robots, enabling real-time processing with low-latency task execution in practical applications.

1.4.1 List of Publications

A significant portion of the research contributions presented in this thesis have been showcased and published in international conferences, workshops, and a journal.

P1. **Sandeep Reddy Sabbella**, Sara Kaszuba, Francesco Leotta, Pascal Serrarens and Daniele Nardi. 2023. **Evaluating Gesture Recognition in Virtual Reality, Workshop Your Study Design**, HRI' 23.

P2. **Sandeep Reddy Sabbella**, Sara Kaszuba, Francesco Leotta, and Daniele Nardi. **Virtual Reality Applications for Enhancing Human-Robot Interaction: A Gesture Recognition Perspective**. In ACM International Conference on Intelligent Virtual Agents (IVA' 23), Sept 19–22, 2023, Würzburg, Germany.ACM, New York, NY, USA, 4 pages, `https://doi.org/10.1145/3570945.3607333`.

P3. **Sandeep Reddy Sabbella**, Sara Kaszuba, Francesco Leotta, and Daniele Nardi. **Gesture Recognition for Human-Robot Interaction through Virtual Characters**. 15th International Conference on Social Robotics (ICSR' 23), Doha, Qatar, Dec 3-7, 2023.

P4. **Sandeep Reddy Sabbella**, Pascal Serrarens, Francesco Leotta and Daniele Nardi, **Generating and Evaluating Synthetic Data in Virtual Reality Simulation Environments for Pose Estimation**, 2024 33rd IEEE International Conference on Robot and Human Interactive Communication (ROMAN), Pasadena, CA, USA, 2024, pp. 2319-2326, `https://doi.org/10.1109/RO-MAN60168.2024.10731179`.

P5. **Sandeep Reddy Sabbella**, Alexia T. Salomons, Francesco Leotta, Daniele Nardi. **Assessing Multimodal Communication in Human-Robot Interaction: A User Study** 16th International Conference on Social Robotics (ICSR '24), Odense, Denmark, Oct 23-26, 2024.

P6. Sara Kaszuba, **Sandeep Reddy Sabbella**, Francesco Leotta, Pascal Serrarens, and Daniele Nardi. 2023. **Testing Human-Robot Interaction in Virtual Reality: Experience from a Study on Speech Act Classification**, In Proceedings of International Workshop on Virtual, Augmented, and Mixed-Reality for Human-Robot Interactions, Stockholm, Sweden (VAM-HRI' 23). ACM, New York, NY, USA, 8 pages.

P7. Sara Kaszuba, **Sandeep Reddy Sabbella**, Francesco Leotta, and Daniele Nardi. **Speech Act Classification in Collaborative Robotics**. 32nd IEEE International Conference on Robot and Human Interactive Communication, IEEE RO-MAN 2023, Busan, South Korea.

P8. Sara Kaszuba, Julien Caposiena, **Sandeep Reddy Sabbella**, Francesco Leotta, and Daniele Nardi. **Empowering Collaboration: A Pipeline for Human-Robot Spoken Interaction in Collaborative Scenarios**. 15th International Conference on Social Robotics (ICSR' 23), Doha, Qatar, Dec 3-7, 2023.

1.5 Structure of the Thesis

This thesis is structured as follows, with each Chapter briefly summarised to highlight its key contributions.

Chapter 2: Literature Review
This Chapter presents a comprehensive review of the foundational concepts, datasets, and methodologies relevant to Human-Robot Interaction (HRI). It begins by exploring the theoretical underpinnings of speech acts as a communication framework in HRI. The discussion then shifts to pose estimation techniques, covering 2D and 3D approaches, along with single-person and multi-person estimation, highlighting key datasets and benchmarks essential for robotic perception. The role of Virtual Reality (VR) and immersive technologies is examined, particularly in their application to HRI research and synthetic data generation for training machine learning models. Additionally, the Chapter delves into core HRI principles, focusing on collaborative speech, gesture-based interaction, and multimodal communication strategies that enhance human-robot collaboration. The Chapter concludes by synthesising key insights from the literature, providing a foundation for the subsequent methodological approach.

Chapter 3: Methodology
This Chapter presents a systematic approach to evaluating and integrating multimodal interaction techniques in designing the collaborative framework for human-robot interaction in a table-grape vineyard scenario. It begins by identifying and categorising speech acts, establishing the foundation for effective communication between humans and robots. The methodology then details the theoretical principles behind speech and gesture recognition systems as key communication channels for outdoor environments. Additionally, it explores the role of virtual reality as an experimental platform for controlled testing and discusses synthetic data generation techniques to address data scarcity. Finally, the Chapter outlines the integration of speech, gestures, and other multimodal cues within the HRI framework, enhanced by large language models (LLMs) for adaptive reasoning in dynamic environments.

Chapter 4: Virtual Reality Applications in HRI
This Chapter explores the integration of Virtual Reality (VR) as a key tool in HRI research, particularly for evaluating collaborative robotic systems in a controlled environment. It begins by detailing the essential system setup, including hardware and software components, as well as the multimodal interaction techniques (speech, gestures, sound, and light signals) used to develop realistic HRI scenarios. The role of VR as a testbed is then examined, highlighting its capability to simulate complex

real-world conditions for risk-free prototyping and data collection. The Chapter further presents a user study comparing immersive and non-immersive configurations, designed to evaluate predeployment interactions between humans and robots. Speech act recognition experiments are incorporated to validate collected vocal utterances and assess the efficacy of different interaction modalities. Additionally, insights from internal deliverables authored by PaleBlue are included to describe the simulated vineyard scenario, enriching the fidelity of the VR model. The Chapter concludes with an overview of experimental findings, discussing both the effectiveness and limitations of VR-based approaches in advancing HRI research and real-world robot deployment.

Chapter 5: Synthetic Data Generation and Evaluation
This Chapter underscores the role of synthetic data generation in advancing HRI research, particularly within the table-grape vineyard application domain. Given the logistical and ethical challenges of real-world data collection, a VR simulator—developed specifically for this project—facilitates the acquisition of spoken and gestural data in controlled environments. Additionally, virtual avatars generate large-scale pose estimation datasets, enabling detailed analyses of simulated human motion. The Chapter further explores how text/image-to-3D modelling accelerates prototyping by transforming standard figures into structured digital objects, enhancing virtual simulation fidelity. Finally, a validation framework is introduced to compare synthetic datasets against real-world benchmarks, ensuring the reliability, scalability, and generalizability of results for HRI applications.

Chapter 6: Spoken Human-Robot Interaction
This Chapter presents a speech-based framework for Human-Robot Interaction, focusing on verbal communication as a key modality in outdoor collaborative environments. It begins by exploring speech act classification principles and evaluating existing Speech-To-Text (STT) and Natural Language Understanding (NLU) technologies to inform the development of a robust recognition pipeline. A user study is introduced to assess various Text-To-Speech (TTS) tools and libraries, analysing their effectiveness for generating robot responses in the table-grape vineyard scenario. The Chapter then outlines the architecture of a speech-driven interaction system, addressing challenges in interpreting human vocal expressions and refining classification strategies. The speech recognition and classification pipeline, developed as part of my colleague Sara Kaszuba's work, serves as a stepping stone for further advancements using LLMs, forming the foundation for subsequent improvements in human-robot communication. Implementation details and empirical findings underscore the pipeline's efficacy in outdoor collaborative environments, highlighting how each module contributes to a cohesive architecture that enhances human-robot communication.

Chapter 7: Gestural Human-Robot Interaction
This Chapter explores the design and evaluation of a gesture-based interaction framework for Human-Robot Interaction (HRI), focusing on both full-body and hand gestures in collaborative environments. The discussion begins by establishing a taxonomy of context-aware gestures tailored to application domains such as viticulture-based workflows (pruning and harvesting) and indoor logistics. It

then outlines data acquisition protocols leveraging virtual avatars in simulated environments to train robust recognition models. A comparative analysis of machine learning and deep learning approaches is conducted, centring on three pose estimation techniques — MediaPipe, MoveNet, and VitPose — combined with convolutional neural networks, trained across real, virtual, and hybrid datasets. Emphasis is placed on the VR-based experimental design, which provides scalable data generation and controlled testing conditions. Finally, the Chapter presents an innovative gesture pipeline that integrates multiple recognition algorithms to enhance accuracy and resolve ambiguities. These findings establish the groundwork for a cohesive multimodal framework that combines gestures and speech, which will be further explored in the next Chapter.

Chapter 8: Enhancing Collaboration with Multimodal Interaction
This Chapter underscores the importance of integrating complementary and redundant communication channels to create a robust and efficient multimodal interaction pipeline for human-robot collaboration. Building upon the previously developed speech pipeline, it introduces the fusion of vocal and gestural inputs to enhance communication in outdoor settings. The Chapter details the proposed architecture, covering how speech and gestures are acquired, processed, and merged using multimodal fusion techniques. A demonstrative simulation showcases the system's effectiveness, and user studies analyse key usability factors—including efficiency, memorability, and trust—based on both participant feedback and technical performance measures. These findings provide a strong foundation for future human-robot interaction studies across diverse application domains. Finally, while this architecture is initially designed for future integration with Large Language Models (LLMs), their implementation will be explored in later extensions to enhance adaptability for both indoor and outdoor robotic platforms.

Chapter 9: Conclusions and Future Directions
In summary, this work advances human-robot interaction by integrating speech and gestural inputs to enhance collaboration in both indoor and outdoor scenarios. Despite these contributions, challenges persist in adapting to varying environments and refining seamless communication. Future efforts will focus on improving the robustness of recognition algorithms, exploring more large language models for richer context understanding, and developing adaptive, user-centred designs that further streamline interactive processes in diverse operational settings.

Chapter 2

Literature Review

2.1 Preambles for HRI: Speech Acts

Speech Act Theory offers a comprehensive approach for interpreting human speech in the development of conversational agents, thereby enhancing the robot's ability to understand user intentions and respond effectively [219, 245]. This theory serves as a foundation in the study of communication, language use, and interaction within Human-Robot Interaction (HRI), stemming from the seminal works of philosophers John Langshaw Austin and John Searle. Austin introduced the concept in 1962, which Searle further developed, defining speech acts as actions performed through utterances [18, 219]. These acts extend beyond mere information exchange, encompassing various nuances such as politeness, urgency, and emotion, which significantly influence the effectiveness of communication.

Speech acts are categorized into three primary types:

1. **Locutionary Acts:** These focus on the literal meaning of words and sentences, such as naming objects or describing situations.

2. **Illocutionary Acts:** These represent the speaker's intent behind an utterance, encompassing commands, requests, and assertions.

3. **Perlocutionary Acts:** These pertain to the effects an utterance has on the listener, which can range from inducing beliefs to triggering actions.

Austin's initial framework and Searle's subsequent expansions [18, 219] have significantly influenced Natural Language Processing (NLP), particularly in analyzing and interpreting human interactions. Their categorization into assertives, directives, commissives, expressives, and declarations has paved the way for protocols in agent communication, from early models like KQML to the current FIPA standards [86, 183].

Understanding speech acts is crucial in HRI, especially in contexts where humans and robots share physical spaces, such as collaborative robotics. The precise interpretation of speech acts, particularly from human to robot, is vital for ensuring safety and operational continuity. Misinterpretations or delays in response to speech acts can have varying consequences, underscoring the importance of promptly classifying and responding to these communicative acts to prioritize safety-critical alerts. This

nuanced understanding and classification facilitate the deployment of specialized mechanisms for comprehending and reacting to different categories of speech acts, enhancing both human safety and interaction efficiency. A comprehensive discussion on Speech Acts Theory detailing its significance to the HRI architecture designed is discussed in Chapter 3.

2.2 Pose Estimation Datasets and Benchmarks

Pose estimation has become a pivotal task in computer vision, involving the identification of the configuration of body parts in a given image or video [141]. Recent advancements in this field have revolutionized applications such as human-computer interaction, robotics, and augmented reality. This section reviews the state-of-the-art (SoTA) methods in pose estimation, covering both 2D and 3D estimations, single-person and multi-person tracking, significant datasets, and available benchmarks, with a focus on the findings from [233]. Pose estimation methods are critical for recognizing human gestures in HRI [44]. Benchmarks such as MPII and COCO datasets have facilitated advancements in this area [147]. This topic will resurface in Chapters 5 and 7

2D Pose Estimation

2D pose estimation involves detecting keypoints such as joints or landmarks of a human body in two-dimensional images. The key challenge lies in accurately identifying these keypoints despite occlusions, varying body poses, and complex backgrounds. In recent years, deep learning techniques have dominated 2D pose estimation. Notable SoTA models include the stacked hourglass network [172], which uses a multi-resolution approach, and OpenPose [44], which achieves real-time performance. The key component of these models is the use of convolutional neural networks (CNNs) for feature extraction and heatmap regression for predicting joint locations. Despite significant progress, challenges such as occlusion, cluttered scenes, and varying body poses remain. Methods like part affinity fields (PAFs) [44] and the use of attention mechanisms [54] have been proposed to address these issues.

3D Pose Estimation

3D pose estimation involves determining the 3D coordinates of human body joints from either monocular images or video sequences. Recent SoTA methods in 3D pose estimation rely heavily on deep learning models, such as the work presented by [192] and [124], which estimate 3D poses from a single image by combining 2D keypoint detection with depth prediction. More recently, models like VoxelPose [267] and Monocular 3D Human Pose Estimation via Graph Convolutional Networks [144] have further improved accuracy by leveraging graph-based representations and volumetric data. The main challenge in 3D pose estimation is the inherent ambiguity in depth, especially from a monocular image. To mitigate this, multi-view systems and multi-frame approaches have been proposed, such as in [156].

Single-Person Pose Estimation

Single-person pose estimation refers to detecting the pose of a single human in an image or video sequence. For single-person pose estimation, the majority of SoTA methods use convolutional neural networks. The pioneering work of [172] introduced the stacked hourglass network, which provides high accuracy for single-person pose detection. More recently, methods like HRNet [255] have significantly improved the accuracy of keypoint detection through multi-resolution representations. Challenges in single-person pose estimation often arise from complex poses, occlusions, and background interference. To address these, current models focus on using context information and more sophisticated network architectures, such as HRNet [255], which maintains high-resolution representations throughout the network.

Multi-Person Pose Estimation

Multi-person pose estimation aims to detect and track the poses of multiple individuals in a single image or video sequence. Multi-person pose estimation builds upon the same techniques used in single-person pose estimation but introduces methods for resolving the association of keypoints to different individuals. OpenPose [44] is one of the earliest models to solve this problem, using part affinity fields (PAFs) to link body parts to the correct person. More recent methods, such as Mask R-CNN [108] and PoseNet [243], have improved multi-person tracking by introducing instance segmentation and spatial-temporal consistency. The primary challenge in multi-person pose estimation lies in the efficient handling of occlusions and overlapping body parts. Solutions include tracking by detection, temporal consistency modeling, and leveraging the geometry of human poses [146].

Significant Datasets and Benchmarks

Several large-scale datasets have been introduced to benchmark the performance of pose estimation algorithms and benchmarks have been introduced to evaluate pose estimation models.

Dataset	Description	Type
COCO [147]	Large-scale dataset for 2D pose estimation	2D, Multi-person
MPII [14]	Human pose dataset for 2D estimation	2D, Single-person
Human3.6M [116]	Dataset for 3D pose estimation	3D, Single-person
PoseTrack [13]	Multi-person pose tracking	2D, Multi-person

Table 2.1. Significant Datasets for Pose Estimation

Benchmark	Description	Type
COCO Keypoints [147]	2D pose estimation	2D, Multi-person
Human3.6M [116]	3D pose estimation	3D, Single-person
PoseTrack Challenge [13]	Multi-person pose tracking	2D, Multi-person

Table 2.2. Benchmarks for Pose Estimation

Pose estimation has seen significant advancements in recent years, with deep learning methods such as CNN architectures and Generative Adversarial Networks (GANS), providing impressive results in both 2D and 3d scenarios. The introduction of large-scale datasets and evaluation benchmarks has accelerated progress in this field. However, challenges such as occlusion, varying body poses, and multi-person tracking remain important areas for future research. Pose estimation is one of the crucial component for gesture recognition and its importance extends to effective non-verbal human-robot interaction. Algorithms used as sub-components in the HRI architecture are detailed in Chapter 3 and 7.

2.3 Virtual Reality (VR) and Immersive Technologies

Collaborative robotics in precision agriculture is a recently explored area, and researchers have only begun studying HRI in such scenarios. To this aim, the growing interest in immersive and non-immersive VR in evaluating HRI studies in other disciplines led to adopt VR in both its forms to conduct experiments in collaborative robotics for agriculture. However, to the best of our knowledge, there are no available works investigating Immersive Virtual Experience (IVE) and Non-Immersive Virtual Experience (NIVE) in such a scenario.

The evaluation of immersive VR with respect to non-immersive applications, in terms of learning effects on middle school students with ASD, is presented by Carreon et al. in [50]. In such work, the authors, through a user study, showed that the students achieved notable learning improvements in both experiences, screen-based VR and head-mounted display VR. Interestingly, a significant increase in learning gain, user enjoyment and concentration emerged from the immersive experience described in [160], where Mahmoud et al. conducted an experimental analysis by comparing the impact of immersive and non-immersive systems in learning. The effectiveness of immersive VR in teaching medical students practical skills and clinical interventions is demonstrated by Omlor et al. in [184]. In this article, the authors compare students' learning experiences through VR-viewer and non-immersive screens. Similarly, the potential of IVE in teaching crystal lattices was presented in [252]. Vergara-Rodríguez et al. highlight how the level of immersion could influence relevant design aspects, in particular usability, ease of use, motivation and interactivity.

At the same time, Renganayagalu et al. compared immersive and non-immersive experiences in maritime education to train and improve seafarers' skills in [202]. In such a study, higher motivation and preference have been demonstrated towards immersive training simulator engines than non-immersive ones. Stronger memory performance associated with users participating first in the immersive experience emerged from [251], where Ventura et al. conduct a user study with a few people performing a task in IVE, then in NIVE; while, another group began with the task in the non-immersive scenario, moving then to the immersive experience.

2.3.1 Synthetic Data Generation for Robotics

The generation of synthetic data addresses challenges in acquiring large-scale annotated datasets for training machine learning models in robotics [242]. Techniques

such as domain randomization and simulation-to-real transfer have improved model generalization [196] and can be learned in detail in Chapter 5.

As the synthetic data and the simulations are use-case and application-specific, there is no single state-of-the-art system for benchmarking or evaluating the performance of the simulation environments. However, for the virtual character, the benchmark lies in how photorealistic and natural compared to actual humans.

Synthetic data can be of help in pose estimation tasks. State-of-the-art techniques for pose estimation and gesture recognition [214] are evaluated according to different metrics. Ikeda et al. [114] highlights how synthetic data can be leveraged to train 6D pose estimation networks. They address the challenge of domain gaps between synthetic and real data by proposing a novel sim-to-real instance-level style transfer technique, improving the realism of synthetic data and achieving better pose estimation performance. Belke et al. [31] present the generation of synthetic data (6D pose estimation) based on the production requirements, followed by an evaluation of the algorithms to assess the generalization performance from generic benchmark datasets to custom industrial datasets. A. G. Florea et al. [87] explore the benefits and challenges of artificially generated datasets on one 3D pose estimation model and the ML model transfer learning process.

Engemann et al. [79] proposed AutoSynPose, an automated system for generating synthetic datasets to estimate 6D object pose (position and orientation). This method addresses the challenge of time-consuming manual data collection and labeling. AutoSynPose leverages the Unreal Engine 4 (UE4) game engine to create diverse virtual environments with varying lighting and backgrounds. The generated data includes RGB, depth, and class segmentation images suitable for training machine-learning models for 6D object pose estimation tasks. Juraev et al. [121] highlight the limitations of real-world data for training fall detection systems. They demonstrate how synthetic data augmentation can improve the performance of human pose estimation models in real-world scenarios like elderly fall detection. Clever et al. [56] describe a physics-based method that simulates human bodies at rest in a bed with a pressure sensing mat and present PressurePose, a synthetic dataset with 206K pressure images with 3D human poses and shapes. This article presented PressureNet, a deep learning model that estimates human pose and shape, given a pressure image and gender.

Robots trained in simulated environments can be programmed to respond safely and effectively to human gestures and actions. Buxbaum et al. [43] present the use of a simulation environment for HRI in manufacturing industry settings. Mohsen et al. [161] developed an effective method of creating a virtual environment in Unity for performing simulations on industrial robots, mobile robots, and autonomous vehicles (AGV-s) from the safety perspective for humans. Krupke et al. [135] introduced an open-source software toolkit for combining the Robot Operating System (ROS) and Unity3D to versatile robotic applications involving virtual environments. Lier et al. [145] presented a ROS and MORSE [76] based simulation environment for seamless integration with the robot's ecosystem, e.g., NAOqi and ROS and basic human-robot-interaction capabilities that can foster behavior modeling and functional regression testing using PEPPER robot. Carbone et al. [47] presented a simulation in the Unity game engine that builds fields of sugar beets with weeds in the agricultural domain. Images were generated to create datasets that are ready to train

CNNs for semantic segmentation. Shamshiri et al. [222] reviewed several professional simulators and custom-built virtual environments that have been used for agricultural robotic applications. A simulation case study was demonstrated to highlight some of the powerful functionalities of the Virtual Robot Experimentation Platform. Summarizing the works, researchers have been working with domain-specific synthetic data and simulations for a while, yet a major challenge remains – the sim-to-real gap. Overcoming this gap is essential for successfully deploying robots in real-world HRI tasks. While large datasets are crucial for training high-performing machine learning models, the quality of the data and domain randomization significantly impact the model's ability to generalize to real-world scenarios [242].

Several strategies can be employed to mitigate the sim-to-real gap. One such approach involves incorporating real-world elements directly into the simulation. This can involve importing specific objects as 3D assets, replicating entire scenes, or including realistic poses or actions. Cutting-edge machine-learning models like VIBE [132] can even be used to import real-world 3D body movements into simulations. Essentially, the goal is to create "digital twins" of the real world within the simulation to accurately represent key aspects in synthetic datasets.

The entertainment industry has been at the forefront of developing techniques for creating visually stunning graphics with photorealism as the primary metric, and these advancements are now being applied to generate more realistic synthetic data for machine learning tasks. Advanced techniques like Generative Adversarial Networks (GANs) [113], Low-Rank Adaptation (LoRa) [111], and Stable Diffusion techniques [204] are being actively integrated into large language models (LLMs) and gaming engines to produce photorealistic synthetic data in various formats, including audio, images, and videos.

2.4 Foundations of Human-Robot Interaction

Human-Robot Interaction (HRI) is an interdisciplinary field encompassing robotics, artificial intelligence, psychology, and human-computer interaction [95, 65]. The development of effective HRI systems requires understanding human behavior, communication patterns, and interaction modalities [36]. This extensive review covers speech modality in Chapter 6, gesture modality in Chapter 7 and multimodal communication in Chapter 8.

2.4.1 Collaborative Speech for HRI

Spoken communication has been identified as the most appropriate, intuitive and natural means for exchanging information in such scenarios. To this aim, the majority of the developed robotic systems, for both indoor and outdoor environments, employ speech as the primary interaction channel. Speech-based interaction is crucial for natural HRI, requiring advancements in speech recognition, synthesis, and understanding [195]. Context-aware dialogue systems improve robot responsiveness and engagement [273].

Briggs et al. tackle the problem of enabling taskable robots in [38], by proposing pragmatic and dialogue-based mechanisms to interpret indirect speech acts frequently

employed by humans in socially conventionalized contexts. Through empirical evidence, the study showcases the efficacy of these mechanisms in deducing intended meanings from indirect speech acts, encompassing a range of request forms encountered during the experiment. Hosseini-Asl et al. introduce SimpleTOD in [110], a unified approach for task-oriented dialogue that surpasses previous benchmarks on the MultiWOZ dataset [41]. The authors demonstrate that by utilizing a single language model trained on all sub-tasks, SimpleTOD achieves enhanced performance in dialogue state tracking, action decisions, and response generation. Hanna and Richards analyse verbal communication from speech act perspective in Human-Agent Collaboration (HAC) in their work [106]. In such study, authors describe the structure of the agent's speech acts, the intention behind them, their impact on the person's mental state, and the effect of human perception of such categories on collaborative performance. In [185], Onnasch and Roesler present a novel taxonomy that encompasses multiple facets of HRI research, facilitating comparability and generalizability of findings through predefined categories and structured comparisons. Their taxonomy offers a valuable framework for analyzing and evaluating various aspects of HRI, enhancing the understanding and advancement of the field.

2.4.2 Gesture Interaction for HRI

Gestures complement verbal communication, making HRI more intuitive [128]. Gesture recognition models leverage computer vision and deep learning to improve real-time interaction [169]. The state-of-the-art in gesture recognition for HRI involves a variety of techniques that can be broadly categorized into the following classification:

- **Computer Vision-based techniques:** These techniques use computer vision algorithms to process images or videos of a person performing a gesture and extract features that can be used to recognize the gesture. This can include techniques such as skin color detection, motion tracking, and feature extraction. These techniques are widely used in gesture recognition systems, as they can provide accurate and real-time gesture recognition.

- **Machine Learning-based techniques:** These techniques use machine learning algorithms to train models that can recognize gestures from image or video data. This can include techniques such as decision trees, Support Vector Machines (SVMs), and deep learning-based techniques such as Convolutional Neural Networks (CNNs) and Recurrent Neural Networks (RNNs). These techniques have become increasingly popular in recent years, as they have been shown to be effective at recognizing gestures in real-time.

- **Wearable-based techniques:** These techniques use wearable devices, such as smartwatches or smart glasses, to recognize gestures. These devices can use sensors such as accelerometers or gyroscopes to detect and track the gestures. This approach has the advantage of being non-intrusive, and it can be used to recognize gestures in real-world scenarios, such as in a manufacturing plant, where users may not be able to use a traditional camera-based system.

- **Hybrid techniques:** These techniques combine multiple input sensor modalities, such as cameras, microphones, depth cameras, etc. to improve the accuracy and robustness of gesture recognition systems. For instance, using a combination of audio and video data can help to improve the recognition of gestures performed in noisy environments, or using a combination of color and depth cameras can help to improve the recognition of gestures performed in different lighting conditions.
- **Virtual Reality-based techniques:** These techniques use VR simulations to generate datasets of gestures, which can be used to train and evaluate gesture recognition systems.

Several challenges need to be addressed in gesture recognition systems for HRI [48]. Some of the main challenges include the following:

- **Variations in gesture performance:** People can perform gestures in different ways, making it difficult for the system to recognize them consistently. This can be due to factors such as age, gender, cultural background, or physical abilities.
- **Occlusions and self-occlusions:** Gesture recognition systems can be affected by occlusions, which occur when part of the body performing the gesture is obscured by another object or person. This can make detecting and tracking the gesture difficult for the system.
- **Background noise and lighting:** The performance of gesture recognition systems can be affected by lighting changes or the presence of background noise. This can make it difficult for the system to recognize gestures accurately in different environments.
- **Real-time processing:** Gesture recognition systems need to process and respond to gestures in real-time, which can be challenging due to the high computational demands of the system.
- **Privacy and security concerns:** Gesture recognition systems can raise privacy and security concerns, as they involve collecting and processing of personal data. Ensuring that data is collected, stored, and used in a secure and compliant manner is essential.
- **Scalability and Generalizability:** Some systems are trained on a limited set of gestures and people, making it difficult to generalize to new gestures or individuals, limiting the system's scalability.
- **Benchmarking and Standardization:** There is currently a lack of benchmarking and standardization in gesture recognition systems, making it difficult to compare the performance of different systems and evaluate the progress of the field.

Addressing these challenges will require a combination of advances in computer vision, machine learning, and robotics, as well as a better understanding of the

human factors involved in HRI. Most of the troubles addressed above can be solved with the use of simulated virtual human avatars and will be detailed in the upcoming section.

The recent advent of deep learning-based techniques such as Convolutional Neural Networks (CNNs) and Recurrent Neural Networks (RNNs) have been shown to be effective at recognizing gestures in real-time. Some of such approaches that have been proposed for sign recognition include 2D CNN Models [237, 274], 3D CNN Models[102, 228, 281], GANs[236, 234], Spatio-temporal networks and RNN Models[213, 140]. These systems have been reported to have high accuracy in gesture recognition tasks. Still, the accuracy may change depending on the data's amount and quality, and the model's specific implementation. Recently, transformers and their efficiency in speech and vision domains caught the attention of researchers. A comprehensive review of temporal modeling in action recognition was performed in the article by Elham et al.(2022) [221].

In recent years, researchers have made significant progress in developing new techniques and algorithms to recognize, generate, and animate hand and body gestures in virtual environments. With the advances in virtual avatars, several reviews [119, 279] were being conducted on Human-Avatar Interaction (HAI) in virtual environments, highlighting the importance of gesture-based communication and rehabilitation. They discussed various approaches for animating avatars, including motion capture, keyframe animation, and physics-based animation. This article [143] presents the generation of human-like avatars from images, which involves learning a generative model of the 3D avatar from a large dataset of motion-captured videos. In the end, the authors demonstrated their method's effectiveness in generating realistic and diverse avatars for different poses and actions.

2.4.3 Multimodality in Human-Robot Interaction

Combining multiple modalities (speech, gestures, vision) enhances HRI robustness. Multimodal human-robot interaction research centers on integrating speech and gesture, as evidenced by studies such as Cutugno et al. [64] (2013), Deng et al. [272] (2018), Rossi et al. [227] (2013), and Stiefelhagen et al. [231] (2007). Nine of ten studies report speech and gesture as primary channels, with additional modalities such as head orientation, hand sign, emotion recognition, and tactile inputs appearing less frequently. The studies were identified through a systematic search of major academic databases, including IEEE Xplore, ACM Digital Library, and ScienceDirect, using keywords such as "multimodal human-robot interaction," "speech and gesture in HRI," and "communication modalities in robotics." Inclusion criteria encompassed studies that were published in peer-reviewed journals or conference proceedings between 2013 and 2023. Integrated and modular architectures are common; for example, Rossi et al. [227] (2013) describe a modular system that uses a late fusion strategy with support vector machines, while Stiefelhagen et al. [231] (2007) employ a joint particle filter framework for early fusion of audio and visual data. The state-of-the-art in human-robot interaction combines speech and gesture recognition through multiple fusion architectures, supported by studies showing high accuracy rates in both controlled and industrial environments. Multimodal fusion techniques improve context-awareness and adaptability [26]. A comprehensive analysis of these

architectures is presented in Table 2.3.

2.4.4 Key Findings

- **Multimodal Interaction:** Most studies focus on the integration of speech and gesture. The use of additional modalities enhances interaction quality and efficiency.
- **Fusion Techniques:** Different studies apply different fusion techniques. For example, late fusion is used in some studies for integrating modalities at the decision level.
- ***Performance Metrics:*** High performance metrics are reported, such as accuracy, precision, and recall rates, indicating the effectiveness of the fusion strategies.

Validation methods vary and include controlled experiments, real industrial case studies, and simulated Wizard of Oz approaches. Quantitative performance metrics are reported in several studies, demonstrating the robustness of the methodologies employed. The findings support a methodological approach that builds on speech, gesture, and multimodal communication interactions through a variety of fusion techniques—semantic, late, and early—tailored to distinct application scenarios.

Table 2.3. Comprehensive Analysis of Included Multimodal HRI Studies

Study	Study Focus	Interaction Modalities	Architecture Type	Validation Method	Key Components	Integration Method	Scalability
Cutugno et al. [64], 2013	Multimodal communication	Speech, Gesture	Modular architecture	Case study	Speech and gesture modules, Fusion module	Semantic extraction	Not mentioned
Deng et al. [272], 2018	Human-robot interaction	Speech, Gesture	Robot control system	Experimental study	Speech SDK, Gesture algorithms	Fusion of speech and gesture	Not mentioned
Luo et al. [155], 2015	Information fusion	Hand sign, Emotion recognition	Integrated system	Experimental proof	Hand sign recognizer, Emotion recognizer	Combinatorial approach	Capable of tracking multiple people
Maurtua et al. [112], 2017	Industrial collaboration	Speech, Gesture	Semantic approach	Industrial case studies	Speech and gesture recognition, Semantic technologies	Fusion of interaction mechanisms	Implemented in real industrial cases
Mead and Matarić, [206] 2015	Proxemics in communication	Speech, Gesture	Data-driven models	No mention found	Proxemics and multimodal communication models	Unified framework	Proposed extensions for dynamic adaptation
Rossi et al. [227], 2013	Robust communication	Speech, Gesture	Modular architecture	Experimental study	Gesture recognition (Kinect), Speech recognition (ASR, SLU)	Late fusion classification-based approach	Designed to be extensible
Salem et al. [158], 2011	Robot behavior effects	Speech, Gesture	Not specified	Two experimental studies	Speech and gesture production modules	Not mentioned	Not mentioned
Stiefelhagen et al. [230], 2004	Natural interaction	Speech, Gesture, Head orientation	Integrated system	Experimental study	Speech recognition, Dialogue processing, Visual perception	Integration on mobile robot platform	Demonstrated in kitchen scenario
Stiefelhagen et al. [231], 2007	Humanoid robots	Speech, Gesture, Head orientation	Integrated system	Experimental study	Speech recognition, Gesture recognition, Dialogue management	Joint particle filter framework, Constraint-based fusion	Implemented on humanoid robot platform
Strazdas et al. [71], 2020	Natural interaction	Speech, Gesture, Head pose, Gaze, Posture, Touch	Simulated system	Wizard of Oz study	Path planning, Gesture interpretation	Wizard-controlled integration	Not applicable (simulated system)

Chapter 3

Methodology

This chapter presents a systematic approach to evaluate and integrate multimodal interaction techniques in designing the collaborative framework for human-robot interaction. The study establishes the cornerstone of effective communication between humans and robots by identifying and categorising distinct communicative acts—commonly known as speech acts. The methodology outlines the theoretical principles that underpin the development of speech and gesture recognition systems as communication channels for outdoor settings like table-grape vineyards. It incorporates the use of virtual reality as an experimental platform and the development of synthetic data generation techniques to address the issue of data scarcity. The research prioritises analytical and algorithmic rigour to ensure that subsequent experimental validations are robust and replicable.

The chapter is structured as follows: Section 3.1 presents the theoretical background underlying speech act classification, outlining the frame semantics and pragmatic principles used to interpret human utterances. Section 3.1.1 further details the various speech act categories relevant to this study. Section section 3.2 reviews the different communication modalities—such as voice, gestures, sound, and visual signals—and explains their respective contributions to effective human-robot communication. Section 3.3 illustrates using virtual reality as a controlled testbed for simulating real-world interactions, while Section 3.4 outlines the synthetic data generation strategies that underpin the training of robust recognition models. Finally, Section 3.5 describes the integrated HRI architecture that merges speech, gestures, and additional multimodal cues, augmented by large language models, to adaptive reasoning in dynamic operational environments.

3.1 Theory of Speech Act Classification

The design and implementation of the collaborative framework for Human-Robot Interaction have been examined from an agent communication perspective. The identification of different message types, also known as speech acts, and the definition of their meaning is of utmost importance to ensure a proper understanding and efficient interaction between humans and robots. This analysis is essential to comprehend what communication modalities are most commonly associated with certain speech acts, and determine the interaction channels that are more suitable to

be employed in outdoor collaborative environments, such as in the testbed table-grape vineyard.

3.1.1 Speech Act Categories

The term speech act is used to identify a set of classes that differ according to the informative content exchanged in HRI. The communication act categories that emerged from our preliminary study conducted on VR solutions for HRI are the following: "Information", "Command", "Alert", "Request", "Instruction", and "Greeting", [126] shown in Figure 3.1. For this purpose, dedicated sections are created to discuss each proposed discourse act, aiming to clarify their distinction and usage in the context of table-grape vineyards. A detailed explanation of the identified sub-classes of each speech act is also provided. However, only a subset of the presented discourse categories is considered in this thesis, which includes "Information", "Command", "Request", and their combination. Special emphasis is placed on human-to-robot communication, which explores ways to enhance the robot's comprehension of the informative content exchanged during the interaction.

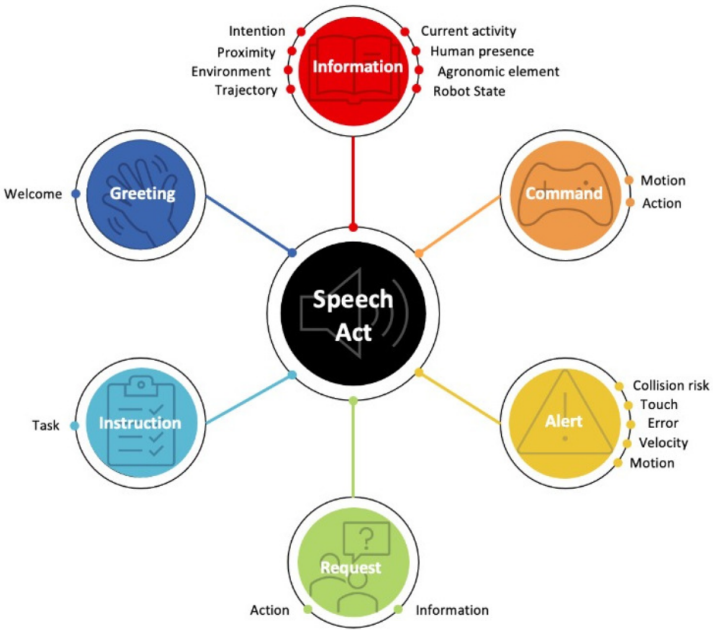


Figure 3.1. Proposed speech act categories.

Source: [126]

Information

Information is one of the most frequently encountered speech acts in human-robot interaction (HRI) experiments conducted in virtual reality (VR). Such exchanges enable both humans and robots to inform their respective collaborative partners about specific events and circumstances in the field, including intentions [16, 164], current tasks, measured distances from the target [164] or from a human collaborator

[226], object properties [259], and trajectory details [17]. Human presence, which refers to awareness of a person within the environment—features are considered as another relevant element in various studies [175], as do positional relationships with an object [82, 162] or teammate [176], current status [226], and performance outcomes [168, 226]. In these settings, bidirectional communication (from androids to humans and vice versa) is typically preferred.

Intention

Intention refers to announcing planned activities or movements before they happen. Robots and humans can better coordinate tasks, avoid collisions, and strengthen trust by sharing upcoming actions. This is especially useful in environments like vineyards, where farmers and robots navigate cramped spaces and delicate crops.

Key Points:

- Anticipating movements improves safety and efficiency.
- Understanding a partner's intentions reduces the chance of accidental interference or damage.
- Enhances overall trust and collaboration between people and robots.

Example:
Information_Intention (human, robot, moving left): A human tells a robot they will move to the left, ensuring the robot takes this into account when planning its trajectory.

Current Task

The current task aspect clarifies each agent's immediate assignment. Human-robot teams can determine when help is needed and reassign or reprioritise tasks as necessary.

Key Points:

- Communicating real-time responsibilities ensures both parties stay informed.
- Reassigning tasks dynamically improves overall workflow and reduces downtime.
- Encourages immediate assistance where it is most required.

Example:
Information_CurrentTask (robot, human, leaves removal): The robot informs the human that it is engaged in removing leaves, prompting the human to decide whether to assist or proceed with another task.

Proximity

Proximity deals with safe and efficient spacing among humans, robots, and the surrounding environment. Human-robot distance is a fundamental concept that ensures safety and trust while performing tasks in the collaborative table-grape vineyard scenario. To this aim, humans and robots must be informed about the distance from each other and the object's proximity. For instance, a notification

about the depth from the cutting device (from the robot's side) and the grape cluster could be advantageous. Interpersonal distance is key in defining human-robot distance when executing a shared task since it is fundamental for robots to avoid colliding into humans' private space, which could lead to a consequent reduction in trust and safety [191]. Moreover, the position of the logistic robot on the field is essential, as it should facilitate the box exchange mechanism between the farming and logistic machine when the boxes carried by the farming robot are almost full.

Key Points:

- Establishes safe human-robot distances for tasks like harvesting and pruning.
- Considers linear and angular distances to facilitate accurate cutting or picking.
- Adapts robot positioning to a user's preference (e.g., right-handed or left-handed approach).

Examples:

- *Information_Proximity (human, robot, moving close to the grape bunches)*: A human notifies the robot that they are approaching a cluster, ensuring it maintains an appropriate distance.
- *Information_Proximity (human, robot, moving on human's right side)*: Alerts the robot of the user's right-handed orientation, enabling safe and convenient collaboration.

Agronomic Element

This category captures details about grape clusters, branches, and other vineyard-related features (e.g., size, type, and colour). Robots assist by using image evaluation, while humans provide complementary information from experience or direct observation.

Key Points:

- Sharing agronomic data (e.g., identifying hidden clusters) improves harvest and pruning decisions.
- Helps align robot cutting poses to avoid damaging the crops.
- Supports managing box capacity for collected grapes or trimmed branches.

Examples:

- *Information_AgronomicElement (robot, human, specific grape cluster)*: The robot identifies a particular cluster and informs the human, aiding precise harvesting.
- *Information_AgronomicElement (human, robot, pizzutello grapes)*: The human specifies the grape variety, guiding the robot's detection process.

Human Presence

Human presence highlights situations where new individuals enter the workspace or when colleagues require assistance. The system maintains safety and supports collaborative tasks by quickly notifying robots and humans.

Key Points:

- Ensures awareness of additional people in the shared environment.
- Coordinates group work with multiple robots and human workers.
- Minimises risks by anticipating actions and presence in confined vineyard rows.

Examples:

- *Information_HumanPresence (human, robot, X needing help)*: The human indicates that another individual in the vicinity needs assistance from the robot.
- *Information_HumanPresence (human, robot, human approaching from the right)*: Signals the robot to be aware of a new person nearby.

Environment

Environmental factors, such as weather conditions or obstacles (e.g., leaves or stones), can significantly impact vineyard tasks. Continuous updates enable robots and humans to adapt their strategies, especially when faced with sudden meteorological changes.

Key Points:

- Humidity, temperature, and time of day influence ripeness decisions.
- Obstacle detection (e.g., stones or leaf piles) ensures safe and uninterrupted work.
- Timely notifications (e.g., stopping before a forecasted storm) protect workers and crops.

Examples:

- *Information_Environment (robot, human, time of ripeness for a specific grape)*: The robot estimates when the grapes are ready for harvest based on environmental data.
- *Information_Environment (human, robot, presence of stones on the row)*: The human alerts the robot about hazards in its path.

Robot State

The state category includes the operational status of robots and humans, indicating battery levels, hardware malfunctions, or even speech, gesture and object recognition confidence. Prompt awareness of these details guides strategic task decisions and fosters faster troubleshooting.

Key Points:

- Combines activity-related status (e.g., full box) with internal conditions (e.g., hardware failures).
- Allows humans to track battery charge or hardware integrity, ensuring timely maintenance.
- Communicating speech recognition accuracy helps humans refine their instructions.

Examples:

- *Information_State (robot, human, battery completely recharged)*: The robot notifies the user that its battery is at full capacity and can resume tasks.
- *Information_State (robot, human, listening state)*: Conveys that the robot is actively waiting for further commands.
- *Information_State (robot, human, certain word not correctly understood)*: Alerts the human to rephrase or repeat instructions for better clarity.

Command

A unidirectional speech act category, expressing the commands given by the human to the robot, is represented by the Command class, in which two sub-classes have been identified: motion, where the android has to reach a new position [162, 164], change its velocity, or stop its operation [57, 164]; and action, that consists of performing specific activities, such as following a person [115, 176], picking up [57] or positioning an object [127]. All messages that must be immediately executed by the android belong to this group of information.

Action/Activity

Commands in this category instruct the robot to carry out specific tasks, such as following a person, harvesting a grape cluster, or pruning branches. By issuing precise directives, humans can guide the robot effectively in situations where simply describing the destination or target is insufficient.

Key Points:

- Enables humans to directly request or guide a desired robot action.
- Includes *follow* commands for leading robots to a goal location.
- Facilitates targeted tasks like pruning or harvesting, where the human's expertise complements the robot's capabilities.

Example:
Command_ActionActivity (human, robot, pruning a specific branch)
A human instructs the farming robot to prune a designated branch in the vineyard.

Motion

While movement is fundamentally an action, it warrants a dedicated category due to its range of actions (e.g., reaching targets and adjusting speed). Motion commands encompass speeding up, slowing down, stopping, or shifting to a new position to enhance safety and efficiency during vineyard operations, presuming that there are no obstacles or collisions in its path.

Key Points:

- Covers position adjustments (e.g., moving forward, sideways) and velocity changes.
- Allows rapid reaction to obstacles or potential field damage (e.g., sudden stop).
- Facilitates navigation between corridors and fine-tuning robot approach for pruning or harvesting.

Example:
Command_Motion (human, robot, moving forward)
A human instructs the robot to move forward, helping it reposition for better access to clusters or branches.

Alert

Notification of a dangerous situation is a fundamental aspect that needs to be analysed when developing human-robot communication in CANOPIES. For this reason, the bidirectional Alert speech act category is provided. Collision risk (higher probability of human-robot collision [175]), touch (the collision is verified, so the person and the android are in contact [88]), error (about task execution), velocity (reduction or increase of robot's speed [166]), and motion (change in machine's motion trajectory [166]) belong to this classification.

Collision Risk

Alerts about a potential collision are essential for ensuring human safety and preventing damage to the vineyard. These notifications can come from either humans or robots, forming a bidirectional communication channel that increases trust and system reliability. Early warnings help avoid collisions with grape clusters, branches, or vineyard structures such as pergolas.

Key Points:

- Aims to prevent human-robot or robot-environment collisions.
- Fosters a safer, more robust architecture via bidirectional alerts.
- Minimises damage to crops and infrastructure.

Example:
Alert_CollisionRisk (human, robot, probable collision with the grape cluster)
A human warns the robot that it risks colliding with a grape cluster.

Touch

Touch alerts notify humans or robots that physical contact with agronomic elements (e.g., grape clusters or branches) has occurred. Quick awareness of these unplanned collisions allows the human to intervene promptly, potentially stopping the robot's operation to prevent further damage.

Key Points:

- Specifically addresses physical collisions already taking place.
- Prompts immediate intervention by the human if necessary.
- Particularly relevant for robots performing delicate tasks in cramped vineyard conditions.

Example:
Alert_Touch (robot, human, impacting with the grape cluster)
The robot signals that it has made contact with the grape cluster, allowing the human to take corrective action.

Error

This category covers errors encountered during task execution or interpretation (e.g., misheard commands, incorrectly performed actions). By quickly reporting these issues, the system prevents further complications, conserves time, and maintains operational efficiency. Errors can be divided into:

- *Execution errors*: The robot understood the instruction correctly but performed it incorrectly.
- *Interpretation errors*: The robot misunderstood the command, gesture, or signal, leading to an undesired outcome.

Key Points:

- Immediate alerts allow humans to intervene and correct or clarify the situation.
- Reduces uncertainty in communication and ensures safe collaboration.
- Critical to maintaining trust and efficiency in dynamic field tasks.

Examples:

- *Alert_Error (robot, human, wrongly understood a word)*: The robot informs the human about a misunderstanding.
- *Alert_Error (robot, human, throwing away a good grape cluster)*: The robot executed a task incorrectly due to an error in processing.

Velocity

Velocity alerts inform the human of speed changes—critical for ensuring safety and maintaining awareness of the robot's movements. Knowing if the robot accelerates or decelerates helps workers plan their actions and respond promptly if the robot's speed shift is inappropriate.

Key Points:

- Communicates speed increases or reductions during tasks.
- Ensures human awareness of robot dynamics.
- Enhances safety by avoiding unexpected robotic movements.

Example:
Alert_Velocity (robot, human, increasing velocity to reach the end of the row)
The robot notifies the human that it will move faster, helping the worker anticipate and coordinate.

Motion

Motion alerts warn of significant trajectory changes. Notifying humans about upcoming robot movements (e.g., a sudden rotation) enables them to prepare or intervene if needed. Similarly, humans can alert the robot of unsafe moves, prompting it to replan its route.

Key Points:

- Provides advance notice of major shifts in the robot's orientation or path.

- Strengthens trust by ensuring predictable motion in tight vineyard conditions.
- Allows the human to halt or redirect the robot if the planned motion poses risks.

Example:
Alert_Motion (robot, human, farming robot's body rotation of 180 degrees on the right)
The robot reports its intention to rotate, giving the human time to adjust or intervene.

Request

A *Request* is a unidirectional form of communication in which a human asks the robot to undertake a particular action[259, 171] or provide specific information [162, 168]. Unlike a command, however, the robot is permitted to refuse or delay fulfilling a request. This distinction is crucial: a command obliges the robot to comply immediately, whereas a request gives the robot discretionary power to accept, postpone, or reject.

Action/Activity

In this category, humans can request that the robot perform certain operations or tasks, such as following the user to a target location (instead of merely being commanded to do so). Typical applications include:

- **Following a person:** Minimises uncertainty by letting the human guide the robot to a precise destination.
- **Harvesting or pruning**: Asking the robot to target specific grapes or branches, thereby speeding up vineyard operations and dividing workload.
- **Clearing debris**: Removing fallen leaves or broken branches for a safer, more navigable corridor.

These tasks overlap with those in the *command* category; however, as requests, the robot may opt to defer or refuse depending on circumstances like task prioritisation or environmental constraints.

Examples:

- *Request_ActionActivity (human, robot, humans follow to reach the other row)*: The human asks the robot to trail behind them to a different row without providing strict instructions.
- *Request_ActionActivity (human, robot, selection of all the branches to remove)*: The human requests the robot to identify problematic branches, allowing the robot to decide *when* to initiate the task.

Information

This subcategory pertains to instances when humans seek additional details from the robot. Although this subclass bears resemblance to the *Information* speech act discussed earlier in the section 3.1.1, it now specifically describes the human's

request for knowledge rather than a spontaneously provided update. The focus of these requests may include:

- **Environmental conditions**: Weather updates or obstacles in the vineyard.
- **Robot state and intention**: Battery status, ongoing tasks, or planned movements.
- **Human-robot proximity**: Relative distance or orientation for safety and convenience.
- **Agronomic elements**: The location, quality, or specific properties of grape clusters and branches.

Example:
Request_Information (human, robot, time):
The human inquires about the current time, illustrating how the robot may supply purely informative responses upon request.

Instruction

An *Instruction* is a unidirectional speech act where the robot offers guidance [70] or advice to the human. Instruction information can be delivered visually, verbally, or through multiple combined channels. Conversely, humans may also provide step-by-step directions to the robot on completing certain tasks or navigating to specific vineyard locations. These directions may involve:

- Performing specific hand motions [166, 175] or tasks.
- Locating and accessing hidden grape clusters.
- Choosing an optimal path to a vineyard destination.

Key Points:

- Enables the robot to coach humans, reducing uncertainties in complex agronomic tasks.
- Merges various modalities (visual, spoken) for comprehensive support.
- Allows humans to supply robots with precise directives (e.g., how to reach a particular cluster or corridor).

Greeting

Greeting emerges in social robotics as a way for machines to establish rapport with human users. Such personalised and friendly interactions can bolster the human-robot relationship [217]. A welcoming robot can heighten trust and smooth the path for effective teamwork by tailoring greetings to the time of day (e.g., morning, evening) or the worker's context (e.g., starting or ending a shift).

Key Points:

- Builds a sense of familiarity and trust with human partners.
- Fosters a positive work atmosphere, improving collaboration and satisfaction.
- Demonstrates social awareness, which can be critical in team-based tasks.

Examples of Combined Interactions

Following are some illustrative scenarios where multiple speech acts are combined to support effective human-robot interaction (HRI). Each example specifies the speech act categories in parentheses.

Example	Interaction Types	Dialog Acts
1	Request + Information	• *Information_CurrentTask* (robot, human, current task) • *Request_Information* (robot, human, finishing harvesting the clusters of the row) • *Information_CurrentTask* (human, robot, harvesting not completed) • *Request_ActionActivity* (human, robot, harvesting grape clusters of the row)
2	Request + Instruction	• *Request_Information* (robot, human, reaching the hidden cluster) • *Instruction* (human, robot, cutting the leaves surrounding the cluster)
3	Command + Alert	• *Command_ActionActivity* (human, robot, harvesting the grapes at the end of the row) • *Alert_Motion* (robot, human, problem with the motor)
4	Instruction + Alert	• *Instruction* (human, robot, teaching the specific grape type to harvest) • *Alert_Error* (robot, human, camera sensor not working)
5	Instruction + Information	• *Instruction* (human, robot, how to put the harvested grape cluster in the box) • *Information* (robot, human, learning the movement)
6	Request + Instruction + Greeting	• *Greeting* (robot, human, greeting) • *Request_ActionActivity* (robot, human, teaching how to reach the field) • *Greeting* (human, robot, greeting) • *Instruction* (human, robot, exiting the Cooperativa Agricola Corsira, turn right and then move forward)

Table 3.1. Examples of Combined Interactions

3.2 Communication Modalities

Starting from the selected speech act categories and their classification based on the informative content, different communication modalities that could be adopted in HRI are presented here. The selection of the main interaction channels is reached from the preliminary study conducted on existing works concerning VR tools for HRI experiments, as shown in Table 3.2. For this reason, an in-depth examination of voice, gesture, sound, visual signals, and other devices is provided as the main delivery modalities. However, smartphones, tablets and augmented reality technologies, such as smart glasses or head-up displays, could be adopted in CANOPIES in order to guarantee a safer and stronger way to exchange information.

Each of these communication modalities will be briefly described in a dedicated subsection, focusing on the gesture modality more while discussing their strengths and weaknesses. However, an important aspect to be examined is the introduction of multi-modality (combination of multiple interaction channels) to ensure more stable, efficient, and natural communication between humans and robots, which will also be analysed in a dedicated section.

	Information	Command	Alert	Request	Instruction	Greeting
Voice	✓	✓	✓	✓	✓	✓
Gesture	✓	✓			✓	✓
Sound	✓	✓	✓		✓	
Visual Signals	✓	✓	✓		✓	

Table 3.2. The black ✓ represent the outcome of our study, while the green ✓ identify the additional modality combinations that could be investigated within the context of outdoor collaborative environments, such as table grape-vineyards.

3.2.1 Voice/Spoken language

The most natural way of communication in a Human-Human interaction is through voice, so it is natural to adopt this channel also in vineyards for the exchange of information between humans and robots. Indeed, from the analysis of HRI studies in VR, spoken language emerged as the most used medium for information. However, the improvement of the performance of such communication modules is another relevant aspect to achieve with CANOPIES. This can be achieved by clarifying ambiguous situations, giving suggestions to correctly complete a task [217], and communicating the outcome [168] of an action. Considering that one of the primary goals is to develop a robust and efficient communication system, a bidirectional vocal interaction is required so that both the robot (either farming or logistics) and the human can vocally exchange information. Generally, when designing a Human-Robot Communication, different aspects should be taken into consideration, in particular, shared terms that are common to be found in a conversation concerning vineyard activities, such as grape clusters, bunches, branches, harvesting, and pruning. Indeed, voice is obviously the modality adopted to provide requests. In some situations, a teammate could require either a detailed description or information [168, 171] concerning the environment or a specific element of the vineyard. Frequently, requests for performing an action [115, 259], selecting an object or information about other teammates' tasks [162, 176] could be provided (mainly by a human worker). In summary, the idea is to develop a robot that is able to express concepts in a human-like manner so that people feel comfortable and safe. For this reason, we also would like to present a friendly and polite robot that greets a person when starting/finishing the working day and thanks the human worker for their support, trying to prioritise people's requests. However, in outdoor scenarios such as vineyards, voice could not always be the best modality to adopt due to the presence of noise, high uncertainty, and misunderstandings. For these reasons, a solution could be the combination of multiple interaction channels to overcome problems related to using a single modality. To this aim, to strengthen communication, usually spoken language and gestures are combined in systems requiring information exchange between people and robots.

A detailed exploration of spoken Human-Robot Interaction (HRI) is provided in Chapter 6, delving into various technologies for capturing, recognizing, and synthesizing vocal utterances. It presents an overview and in-depth analysis of Speech-To-Text and Natural Language Understanding modules in Section 3.5.1 and Chapter 6, which culminates in the creation of an advanced speech act classification system based on textual sentences. This system, incorporated into the refined speech-based pipeline, effectively identifies distinct speech categories such as "Command," "Information," "Request," and "Complex" statements, the latter being a combination of the three aforementioned categories. A speech act recognizer is developed to minimize the robot's uncertainty and incorrect recognition of the information exchanged during interactions with human workers. This innovative recognizer utilizes spectrum images, which relate to the intonation patterns of vocal expressions, to categorize the content being communicated. Further research is undertaken to identify the most suitable tool within the Text-To-Speech module for generating the robot's verbal feedback. As a result, a comprehensive architecture for Human-Robot Collaboration (HRC) is proposed, designed to facilitate effective teamwork in outdoor settings, specifically within the context of table-grape vineyards. This architecture is meticulously detailed, highlighting its applicability and effectiveness in real-world scenarios.

3.2.2 Gestures

In a Human-Human conversation, gestures are often employed to clarify a concept, to explain better how to reach a certain place and to provide commands [94]. Gestures have been explored as a communication channel both in human-computer (HCI) and human-robot (HRI) interaction. In these domains, usually, gestures unlock a new communication channel to send intentional commands to a machine[48]. Indeed, considering the preliminary study conducted on HRI experiments in VR, it emerged that gestures are the preferred communication modality for the command speech act category. Hence, when the interaction between humans and robots is required, as in CANOPIES, such a channel could be exploited by a person in order to immediately stop the machine's activity by showing his/her opened hand to the mobile vehicle [57, 164], but also to indicate the next position to reach by the robot.

Pointing gestures could be employed in different tasks, not only to show a precise position to reach but also to specify an element [115] of the vineyard (a certain grape cluster, branch) or to indicate the person to follow [176]. However, the introduction of gestures in CANOPIES requires the robot to first detect the position of both a human's arm and hand and, secondly, to understand their meaning. Conversely, the android should also be able to communicate with the person through gestures, by translating a non-verbal behaviour into an understandable action. Unfortunately, gestures alone would not be sufficient to develop a robust Human-Robot Interaction system, because they could cause misunderstandings and uncertainty. Without a vocal explanation, the meaning of some actions would be difficult to capture; as for example, "pointing" is not a self-explanatory gesture thus, if a person points a branch without providing any command or information to the android, the robot would not know the action to associate to such interaction (cut, move, discard the branch). For this reason, in order to disambiguate gestures, a combination with

spoken language will be adopted, preferring a multi-modal approach rather than a single communication channel.

This thesis explores gestural HRI in Chapter 7, by investigating the functionalities of three pose estimation models: MediaPipe, MoveNet, and VitPose, integrated with CNN networks, which are trained and evaluated on various gestural data combinations (solely real, solely virtual, and a fusion of real and virtual). Simulated data are collected by creating digital humans in the virtual environment of the table-grape vineyards, while actual data are gathered directly in the real field. Additionally, a novel gesture recognition pipeline is proposed, capable of employing two distinct recognition algorithms to effectively disambiguate gestures.

3.2.3 Sound

From an analysis of contemporary HRI research in virtual reality, sound signals emerge as the principal interaction channel for the alert speech act category. In vineyard applications, these auditory cues may be necessary on the robot side to warn human teammates of potential dangers, such as when immediate or time-sensitive responses are required, critical events mandate redundant notifications, or an operator's awareness and feedback must be ensured. Various audible signals can be considered by examining parameters such as the following:

- **Distance**: The performance of audible signalling devices is influenced by the distance to be covered. As human workers move farther from the sound source, loudness decreases. This principle is important for determining the coverage area in the table-grape vineyard.

- **Ambient Background Noise**: The effective range of a sounder depends on surrounding noise levels. Maintaining a signal output at least 5 dB(A) above background noise allows for clear communication, which is particularly relevant in noisy vineyard environments.

- **Pattern and Frequency of a Tone**: Sound frequency and pattern strongly affect the success of audible alerts. Lower frequencies travel farther and penetrate structures more effectively. Varying frequency and temporal patterns improve signal distinctiveness, which is vital in high-noise areas.

- **Categories of Audible Signalling Devices**: Two key categories include electronic sirens (e.g. multi-tone alarms, buzzers) and electromechanical horns (e.g. mini horns, signal horns). Electronic sirens are suited for emergency signals, whereas electromechanical horns are better for industrial contexts with mechanical signal tones.

Depending on the distance between humans and robots, different sound intensities may be explored [32]. When people and androids are in close proximity, a higher-frequency signal might be emitted to alert the user of unsafe conditions. However, additional parameters such as pulse pattern speed, frequency, and intensity determine the attention-capturing features of these sound signals. These factors correlate with the priority level of an alert, indicating notifications, warnings, or error messages in

the vineyard context. In particular, tonal signals should comply with recommended design parameters (e.g. intensity, frequency range, and use of specific frequency bands), while speed and fundamental frequency influence both perceived urgency and reaction times. Louder and faster signals tend to shorten reaction times and heighten perceived urgency, although a balance is required to avoid irritation. Further design considerations include unpleasantness (without causing discomfort), composition of sound bursts, synthetic or natural tones, duration, and prevention of interference with previously learned responses. An extended audio signal may also prove helpful when contact occurs between a robot and a human [88], or if the robot's speed is altered or its motion changes (e.g. moving in reverse). In industrial environments, these auditory cues complement visual feedback by conveying information regarding errors, idle conditions, ongoing tasks, or task completion, enhancing overall operator awareness.

3.2.4 Visual Signals

Light signals constitute an emerging visual feedback modality for conveying information and alert speech act categories within the table-grape vineyard context. However, the pergola system is susceptible to light intensity, necessitating careful consideration of signal devices, including beacons and towers. Strobe lights are particularly noteworthy for generating high attention levels, with the choice between LED and Xenon strobe lights influenced by factors such as operational temperature, voltage requirements, and instant lighting capabilities. Intense sunlight may interfere with robot-emitted lights, potentially undermining the effectiveness of visual cues in bright conditions. Nevertheless, integrating light signals can improve human awareness of the robot's activities [88], thereby fostering trust in collaborative scenarios.

LED-based indicators, placed horizontally at the robot's back, front, and sides, prove cost-effective, reliable, salient, and adaptable for expressing diverse states through modulated parameters (e.g. colour, position, duration, and frequency). Signal categories such as "Notification," "Warning," and "Error" help distinguish urgency and importance, while patterns like Steady, Blink, Beacon, and Wipe may be adjusted in speed and repetition. Motion parameters, including approach and speed, also influence signal perception, guided by three dependent measures: urgency, error, and difficulty to ignore. Furthermore, combining audio and visual cues could be examined to enhance communication [127, 226]. In determining variables (e.g. position, pattern, pattern frequency), design choices may draw on standard vehicular lighting practices, ensuring familiar and intuitive cues for human operators.

In this thesis, light signals are integrated into the multimodal system to enhance robot-to-human communication and awareness in simulation. Due to hardware constraints, Sound emission and Light signals were not implemented in the real world by the time of this Thesis. The robot utilises these signals within the virtual table-grape vineyard environment to provide visual feedback during specific task commands. In doing so, it considers the positioning and interpretation of illumination indicators, similar to those employed on vehicles such as cars.

3.3 VR as a Testbed for Human-Robot Interaction

Virtual simulators have gained prominence as they allow for rapid prototyping, extensive algorithm evaluation, and the generation of synthetic datasets crucial for training machine learning models, making them indispensable in advancing robotics technology[53]. A realistic, interactive Virtual Reality (VR) simulator of the table-grape 3D vineyard model was designed with grapevines, support structures, humans, and other robots (see Figure 3.2) in the context of the CANOPIES project for experimental activities, including data acquisition, evaluation of proposed solutions, and user studies. The robot can observe and interact with this environment using the same ROS interface as in the real world. To make this possible, a digital twin of the robot used in the field was created with all its sensors like cameras, LIDAR, and the humanoid upper body with positional, velocity, and force/torque feedback. The physical interaction with the environment was implemented using a dynamic physics model to achieve realistic feedback. In this simulation, we have two types of robots: the farming robot, which has a humanoid upper body for harvesting and pruning tasks, and a logistic robot, which has a storage capability for empty boxes or boxes of grapes. The details of the Simulation Environment and how it incorporates HRI testing is detailed in Chapter 4.

The simulation scene can be configured with various parameters, such as the number of vines, grape bunches, robots, type of digital humans, the position of humans in the field, the positions of the grape bunches, and the type and number of the robots in the simulated field. Additionally, the lighting conditions in the environment can be adjusted to enable variation in the perception, including the sun's position in the sky, light colour, and intensity. These conditions can be generated using a seed value, making it possible to run multiple tests in the same conditions.

The human avatars can be controlled by pre-recorded animations or live-action through body capture devices. All the human and robot movements were subject to dynamic physics and thus will collide and interact with the environment with forces. The robot can observe these movements and environmental changes using its simulated RGB(-D) and LIDAR cameras or force feedback.

The appearance and body proportions can be determined for all human movements by choosing specific avatars. Finally, one or more human figures can be placed anywhere in the field, which enables testing with occluded body parts or simultaneous input from multiple humans. All these fused individual elements enable an extensive range of conditions for training and testing the robots' capabilities, which are impossible in real-life possibilities.

In addition to the aforementioned features, the simulation supports user interaction and control through speech, gestures, and multi-modal commands (a combination of gestures and speech). The virtual characters within the simulation can also execute gestures to direct the robot's navigation and other functionalities. All commands are communicated to the simulator using ROS messages, topics, and services.

Figure 3.2. Simulation environment with a virtual character, farming robot and the logistic robot in the simulated grape field

3.4 Synthetic Data Generation Strategies

Synthetic data is crucial in enabling robust training and evaluation for machine learning models, especially in contexts such as gesture recognition, environment perception, and advanced human-robot interaction. Real-world data collection often poses logistical, ethical, or scalability challenges; hence, simulation-based data generation has emerged as an efficient and flexible alternative. By carefully designing virtual characters, environments, and interactions, it becomes possible to capture diverse human motions and environmental states without the high costs and limitations typically associated with on-site data recording.

A key aspect of this process is *character creation*. Tools such as Autodesk Character Generator [19], Unreal Engine's MetaHuman Creator [80], and Unity's Character Creator [246] were evaluated for their ability to produce a broad spectrum of demographic and visual variations. Autodesk Character Generator was ultimately selected for its adaptability in defining gender, age, ethnicity, and physical attributes, ensuring that the generated characters reflected a wide range of real-world conditions. Each model was verified for compatibility with Unity, including rig configurations and file format requirements, thereby easing the workflow from character creation to deployment within the simulation.

Animation extraction further refined realism by converting recorded human movements into high-fidelity motion data. The DeepMotion [66] platform, for instance,

extracts animations from RGB-D footage or motion capture sessions. These animations were then mapped onto virtual models and adjusted for joint accuracy, collision handling, and visual consistency. By following strict guidelines on camera placement, occlusion avoidance, and environmental lighting, the resulting animations closely resembled actual human activity. Through the motion capture pipeline, time and effort were saved, and the final animations reached quality benchmarks necessary for credible synthetic datasets.

To enhance simulation fidelity, the same Intel RealSense D435i sensors used in actual robotics hardware were integrated into the virtual environment, replicating real-world perception constraints and output. Incorporating these sensors allows for a seamless transition between simulated and real data, as the captured depth and RGB information closely align with field conditions. Virtual modifications such as lighting variation and configurable sound and visual alerts further enrich the dataset, providing extensive coverage of potential use cases and edge scenarios. In summary, these generation strategies forge a strong connection between simulation and reality, affording both scalability and versatility in data creation. The forthcoming Chapter 5 on "Synthetic Data" will delve more deeply into each aspect of this pipeline, detailing the methodological intricacies and key insights gleaned from implementing and refining these approaches.

3.5 Multimodal Communication in HRI

A fundamental requirement for ensuring effective HRI is a shared understanding of the operational environment and the objects within it by both humans and robots. Collaboration in a shared space involves continuous interaction across multiple tasks using different communication modalities. Verbal and non-verbal communication cues serve as primary means of interaction with robots. While verbal communication (*speech*) takes centre stage, non-verbal cues (*gestures, tactile feedback, and visual perception*) play an equally pivotal role in engagement and environmental awareness. The integration of these multimodal interactions enhances overall system efficiency. This section introduces the core concepts of speech and gesture communication modalities, followed by the proposed general architecture for multimodal interaction and fusion. Additionally, the role of large language models (LLMs) in enhancing contextual reasoning and improving multimodal communication in HRI is outlined.

3.5.1 Verbal Interaction

The speech processing pipeline plays a crucial role in verbal interaction and follows a structured flow as shown in Figure 3.3, beginning with speech-to-text (STT) conversion, followed by natural language understanding (NLU), semantic role labelling (SRL), frame argument extraction, and speech act classification. Each stage plays a critical role in ensuring accurate interpretation of human utterances, particularly in dynamic, real-world environments such as precision agriculture.

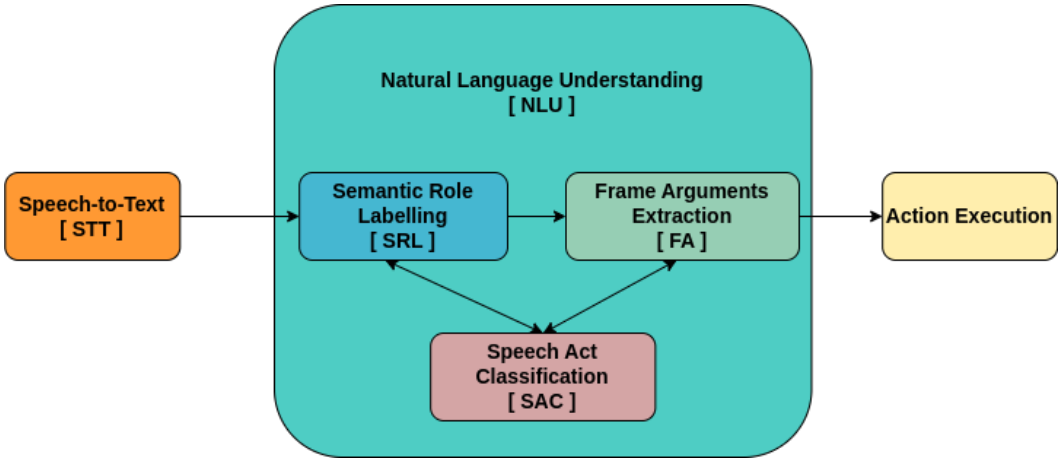


Figure 3.3. Speech processing pipeline structured flow

Speech-to-Text Conversion

The pipeline starts with speech-to-text (STT) processing, which transcribes spoken commands into text. The accuracy of this conversion is crucial, as any transcription errors can propagate through the pipeline, affecting subsequent natural language processing tasks.

Natural Language Understanding (NLU)

Following transcription, the NLU module processes the acquired text to extract both speech acts and frame semantics. This step ensures that the linguistic structure of the input is correctly mapped to actionable meaning. The integration of semantic role labelling (SRL) techniques facilitates the identification of frames—linguistic structures that describe real-world situations—based on the lexical resource FrameNet [25]. Frames encapsulate events, relations, and objects, while their associated frame elements (FEs) describe the roles of participants within those events [84, 85]. A multilingual SRL approach was employed to generate a dataset of frame arguments, leveraging a subset of sentences acquired through Jotform. Approximately 1,000 labelled sentences in Italian and English were manually validated, incorporating additional linguistic information such as lemmas, part-of-speech (POS) tags, and dependency structures extracted using the Stanza library. This process ensured high-quality annotations for training SRL models.

Semantic Role Labelling and Frame Argument Extraction

FrameNet[1] [25] remains the most extensively utilised resource for researchers investigating frames and frame elements. It is commonly employed in the training of Semantic Role Labelling (SRL) systems, which autonomously identify frames and their corresponding arguments within a given sentence [165]. Semantic Role Labelling (SRL) is a fundamental component of the pipeline, responsible for automatically recognising frames and their associated arguments within a sentence

[1] `https://framenet.icsi.berkeley.edu/`

[165]. By analysing sentence structure, SRL determines key aspects of an utterance, answering questions such as: *"Who did what to whom, when, where, and how?"* [92]. Frame elements denote the participants or objects pertinent to a particular frame, delineating the roles that various entities assume within that frame. Typical frame elements encompass "Agent", "Patient", "Instrument", "Location", and "Time", among others. Frame arguments, in contrast, refer to linguistic expressions or lexical items within a sentence that trigger the activation of a specific frame. These may manifest as words, phrases, or even syntactic structures that indicate the presence of a frame and its associated frame elements.

A crucial distinction exists between core and peripheral frame arguments. Core frame arguments are indispensable for evoking a frame and typically include essential components such as the "Agent" and "Patient", which are fundamental to constructing a coherent sentence within the specified frame. Conversely, peripheral frame arguments serve as supplementary elements that, while not strictly necessary for the activation of a frame, enrich the interpretation by providing additional contextual details. These may include attributes, modifiers, and adverbial elements.

The practical application of these principles can be illustrated through an example drawn from an activity in the context of a table-grape vineyard. Consider the following statement: *"The robot harvested all the ripe grape clusters in its row"*. Here, the frame corresponds to *harvesting*, with the following frame elements identified:

- **Agent**: the entity performing the harvesting action.
- **Patient**: the entity being cut.
- **Location**: the spatial setting where the event transpires.

The core frame arguments in this instance are:

- **Agent**: "robot"
- **Patient**: "grape clusters"
- **Location**: "its row"

The SRL model, trained using XLM-RoBERTa [59], enables multilingualism, supporting applications in diverse linguistic contexts. For example, given the Italian command "Posizionati a destra per raccogliere l'uva" (translated as "Position yourself to the right for harvesting the grapes"), SRL identifies "Posizionati" (position) and "raccogliere" (harvest) as the primary verbs. The corresponding frame arguments include "Location" and "Purpose" for the first verb, and "Location" and "Theme" for the second verb, capturing the essential components required for robotic execution. Recent advancements in Large Language Models (LLMs) have facilitated the development of highly accurate SRL methodologies leveraging deep neural networks [59, 60, 4].

Speech Act Classification

Once frame arguments are extracted, speech act classification determines the intent behind the utterance. Speech act understanding is a subfield of natural language

processing (NLP) concerned with analysing and interpreting the communicative function of a statement. Speech acts can be categorised into directives, statements, queries, and other functional types relevant to human-robot interaction. The system employs a fine-tuned classification model, adapted to domain-specific requirements. Traditional SRL methods often rely on generic frame definitions that may not align well with specialised domains such as precision agriculture. To address this, the proposed approach utilises separate models tailored to specific interaction categories, enhancing inference reliability in critical tasks where safety and precision are paramount.

Frame-Based Representation and Action Mapping

Following speech act classification, the extracted information is structured into a frame-based representation that enables robots to interpret and execute commands. Each utterance is decomposed into a structured format, comprising a frame type, associated arguments, and contextual attributes. An example representation of a navigation command is shown below:

```
{
    "frame": "Go",
    "arguments": [{"direction": "Forward", "distance": "10",
                   "unit": "km"}],
    "full_sentence": "Move 10 km forward",
    "language": "En",
    "speech_act": "Directive"
}
```

This structured representation provides a machine-readable format that informs robotic decision-making. The extracted arguments, such as *direction* and *distance*, are mapped to motion primitives, enabling effective task execution in real-world scenarios.

The speech processing pipeline integrates multiple NLP components to achieve robust and context-aware interpretation of spoken commands. Speech-to-text conversion ensures accurate transcription, while NLU incorporates semantic role labelling and speech act classification to extract meaningful intent. The adoption of a structured frame-based representation facilitates seamless mapping of linguistic input to executable robotic actions. By fine-tuning models for domain-specific applications, the framework enhances reliability, particularly in complex, dynamic environments such as agricultural robotics. A comprehensive analysis of the entire collaborative speech pipeline, including its implementation, evaluation, and empirical results, is provided in Chapter 6. Additionally, the integration of speech-based interaction within the broader multimodal HRI framework, encompassing gestures and other communication modalities, is detailed in Chapter 8.

3.5.2 Gestural Interaction

Gesture recognition is another essential modality in human-robot interaction (HRI), enabling intuitive and contactless communication. This non-verbal interaction employs human pose estimation techniques to detect and interpret gestures, allowing robots to respond to human commands in real-world scenarios. The methodology follows a structured pipeline that includes pose estimation, body modelling, and gesture classification.

Human Pose Estimation

Human pose estimation is the process of predicting the spatial configuration of body parts and joints from images or videos. It plays a crucial role in understanding human actions and is widely applied in activity recognition, video analysis, and augmented reality [45]. Pose estimation methods can be broadly categorised into 2D and 3D approaches:

- **2D Pose Estimation:** This method estimates the two-dimensional locations of key body joints using visual data. Traditional 2D pose estimation approaches relied on hand-crafted feature extraction techniques, while modern deep learning-based models such as OpenPose, AlphaPose, and HRNet [264] provide more robust and scalable solutions.

- **3D Pose Estimation:** Unlike 2D approaches, 3D pose estimation predicts joint positions in three-dimensional space, offering richer structural information about human movement [193]. This method is particularly beneficial for robotics applications requiring depth-aware gesture recognition. Advanced techniques also recover 3D human mesh representations from monocular images or videos, enabling a more comprehensive analysis of body motion [283].

Human Body Modelling

To extract meaningful features from visual data, human pose estimation employs various body modelling techniques. These models represent the body as a structured entity, facilitating accurate gesture interpretation. The three primary models used in pose estimation are Figure 3.4:

- Kinematic Model: Also known as the skeleton-based model, this approach represents the human body as a set of joints connected by limbs, capturing relative orientations and relationships between body parts [233]. It is widely used for both 2D and 3D pose estimation but does not account for body texture or shape.

- Planar Model: This contour-based model represents body shape using geometric primitives such as rectangles and silhouettes. Active Shape Models (ASM) [83] are commonly used to approximate human body contours and extract shape-based features for gesture recognition [61].

- Volumetric Model: Used primarily for 3D pose estimation, volumetric models construct statistical representations of human body shapes and movements. Methods such as GHUM and GHUML(ite) employ deep learning pipelines trained on large-scale full-body scans to generate high-fidelity 3D human models [265].

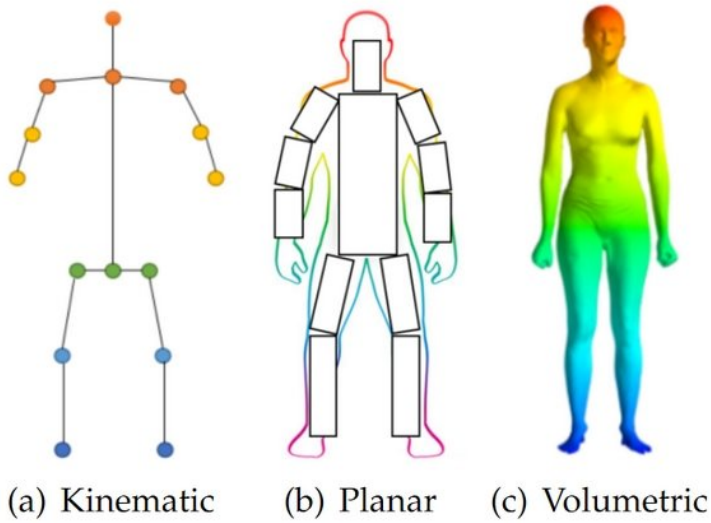


Figure 3.4. Human Pose Modeling: The three types of models for human body modelling

Source: [284]

Challenges

Accurate pose estimation remains a challenging task due to variations in human appearance, clothing, occlusions, and environmental conditions. Factors such as lighting, background complexity, and extreme body postures significantly impact model performance. Additionally, small and barely visible joints, such as fingers, are particularly difficult to track, requiring high-resolution image processing and robust feature extraction techniques [55].

Pose estimation serves as a foundation for gesture recognition by mapping human body movements to predefined commands. This thesis employs both full-body and hand pose estimation techniques to enable gesture-based robot control. Architectures such as DensePose and PoseNet [101, 189] are commonly used for activity and gesture recognition.

In the proposed system, specific body and hand gestures are mapped to robot commands, allowing seamless human-robot collaboration. The gesture definitions, underlying algorithms, and classification techniques are discussed in greater detail in Chapter 7, where empirical evaluations and implementation details are provided.

3.5.3 Multimodal Interaction

Multimodal interaction in human-robot interaction (HRI) involves the integration of multiple communication channels—such as visual, auditory, and tactile modalities—to enhance system effectiveness and efficiency. This approach facilitates richer information exchange, enabling robots to process diverse inputs and respond in a human-like manner. Historically, HRI relied on unimodal communication, but increasing robot integration in industrial settings has driven demand for multimodal systems.

Key advantages include robustness in noisy or visually challenging environments, combining speech with gestures mitigates ambiguities inherent to isolated modalities. Gestures can point in the direction the robot has to move in case the speech command does not explicitly say any direction. Likewise, if more modalities resonate the same information, it can strengthen the accuracy of communication with the robot. Two primary modality combinations are identified:

- **Complementary**: Distinct modalities convey unique information to form a complete message (e.g., pairing the verbal command "Robot, move there" with a pointing gesture).
- **Redundant**: Overlapping information across modalities reduces misclassification risks (e.g., reinforcing a verbal command with a confirmatory hand gesture).

Fusion methods for integrating modalities, as outlined in [96, 139], include:

- **Signal-level fusion**: Synchronised homogeneous signals.
- **Feature-level fusion**: Synchronised inputs with shared information.
- **Decision-level fusion**: Heterogeneous modalities with differing temporal scales.
- **Hybrid fusion**: Integration across multiple processing stages.

For table-grape vineyard HRI, where speech and gestures dominate, *decision-level fusion* is prioritised. This method combines outputs from independent modality-specific recognisers, offering modularity, simplicity, and adaptability to diverse operational scenarios. While *hybrid fusion* could enhance robustness, decision-level integration aligns with scalability and ease of deployment objectives, ensuring compatibility with future recognition modules.

General Architecture for HRI

The preliminary multimodal HRI pipeline, designed to enhance collaboration between humans and robots in the challenging outdoor environment of table-grape vineyards, is shown in Figure 3.5. The proposed architecture illustrates both human-to-robot and robot-to-human interactions. This structure will be refined as

improvements are made to each communication modality detailed in the Chapters 6 and 7, with the lessons learned contributing to the development of an effective multimodal communication pipeline for HRI. The same architecture is adopted to use LLMs and reciprocate information and commands between humans and the robot.

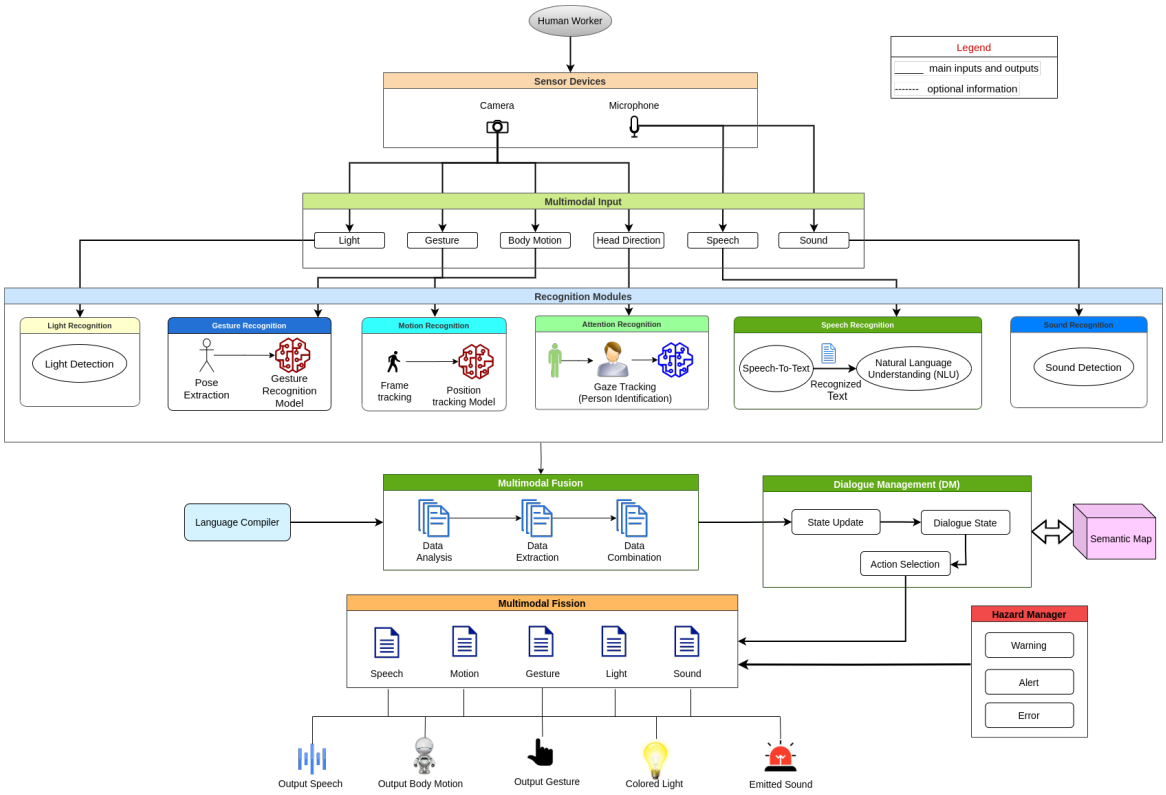


Figure 3.5. Preliminary architecture for multimodal communication in HRI.

LLMs in Multimodal HRI

Large Language Models (LLMs) enhance multimodality in Human-Robot Interaction (HRI) by integrating heterogeneous sensory inputs (e.g., speech, vision, gestures) into a unified decision-making framework. They can generalise the tasks based on example prompts tailored to custom actions to span their functionality over similar tasks. Their information exchange can be extended to all the sensory information, provided they are connected in real time. Their capabilities are discussed in Chapter 8, these include:

- **Cross-Modal Understanding**: LLMs like *PaLM-E* [74] process multimodal inputs (text, images, sensor data) by embedding them into a shared latent space, enabling robots to interpret commands like "harvest the ripe grape bunch" while referencing camera feeds.
- **Contextual Continuity**: LLMs maintain dialogue history and environmental context, as demonstrated in [152], where vision-language models align visual

inputs with textual prompts for coherent robotic task execution.

- **Intent Disambiguation**: By parsing implicit meaning from multimodal signals, LLMs resolve ambiguities (e.g., distinguishing "turn left" with a pointing gesture). This aligns with [225], where LLMs disambiguate user intent using real-time sensor data.
- **Real-Time Adaptation**: LLMs dynamically adjust robot behaviour based on sensor feedback. For instance, [74] shows how LLMs enable robots to replan trajectories upon detecting obstacles.

Synchronous Task Execution Using State Machines

State machines ensure deterministic task execution in dynamic HRI environments, addressing concurrency and interruptions:

- **State Representation**: Tasks are modelled as discrete states (e.g., *idle*, *moving*, *grasping*), a method validated in [77] for modular robot behavior management.
- **Transition Logic**: Libraries like *PyTransitions*[2] enable event-driven transitions (e.g., `start`, `pause`), ensuring atomic operations. This mirrors frameworks in [131], where states prevent resource conflicts.
- **Concurrency Control**:
  - *Mutual Exclusion*: Only one task accesses shared resources (e.g., gripper, wheels), critical for safety in collaborative robots [240].
  - *Interruptible States*: High-priority tasks (e.g., emergency stops) override active states, as in [39].
  - *Atomic Sub-States*: Granular task division enables rollback, as seen in industrial robot programming [142].

Integration of LLMs and State Machines

The synergy between LLMs and state machines enables context-aware HRI:

- **LLM-Driven Transitions**: LLMs generate state transition commands (e.g., "Terminate navigation") by interpreting natural language, as in [152].
- **Feedback Loops**: Sensor data (e.g., "object misplaced") triggers LLM-guided state adjustments, similar to [225].
- **Error Recovery**: LLMs propose corrective actions (e.g., "Retry grasping"), modifying state machine parameters dynamically [74].

[2] `https://github.com/pytransitions/transitions`

Chapter 4

Virtual Reality for Human-Robot Interaction

Virtual simulators have become essential tools in the fields of robotics and human-robot interaction (HRI), providing a safe and cost-effective platform for testing and data gathering. These sophisticated digital environments enable researchers to simulate complex robotic behaviours and interactions with humans without the risks associated with physical experiments. They have gained prominence as they allow for rapid prototyping, extensive algorithm evaluation, and the generation of synthetic datasets crucial for training machine learning models, making them indispensable in advancing robotics technology [53]. The notable evolution of virtual simulators has transformed the landscape of robotics research, particularly since the advent of advanced computational power and AI integration. As simulation techniques improve, researchers can create highly realistic scenarios that closely mimic real-world environments, facilitating more effective training and interaction models. This shift enhances the development cycle of robotic systems and promotes safer testing practices, as potentially hazardous experiments can be conducted in a controlled virtual setting [103, 9]. This chapter illustrates how the virtual real simulation has been created and used as a tool for HRI. How was this system used to understand the HRI dynamics of predeployment in real robots? A user study was needed to understand immersive vs non-immersive experiences and ensure they can operate effectively alongside human users in diverse settings. Explanations of the hardware requirements, software features, and how the interaction between humans and robots is developed within the virtual environment, including verbal communication, physical gestures, light indications and sound, are described in this section. However, to provide details about the simulated table-grape vineyard, some information from internal deliverables authored by PaleBlue has been reviewed and included in this section.

4.1 Virtual Reality Simulation Architecture

A Virtual Reality (VR) simulator of the table-grape vineyard was developed in the context of the CANOPIES project for experimental activities, including data

acquisition, evaluation of proposed solutions, and user studies. PaleBlue[1], one of the project's partners, was in charge of creating the digital twin of the actual vineyard in Agrimessina[2] premises, a table-grape producer in southern Italy (also a partner of the project), where the final developed approaches would experiment on real robots. Considering the difficulty in reaching Agrimessina to conduct preliminary analyses and acquire the necessary data for speech and gestures, a table-grape vineyard in Aprilia, in the Lazio region (central Italy), has been selected as a worksite for the project.

However, given the logistical constraints of physically visiting the field multiple times for data acquisition and approach evaluation, VR is leveraged in both immersive and non-immersive forms to conduct HRI experiments on collaborative activities, like grape harvesting and branch pruning irrespective of the seasonal changes and table-grape harvesting cycle. Any data that is required to replicate is collected in person during their respective seasons / yield cycles. To this aim, robots and humans are able to communicate in the simulation environment similarly to how they do in the real world: talking, listening, pushing, grabbing, and looking. Hence, the input cues simulate the regular sensors: ears, hands, and eyes for humans, while joint states, RGB(-D) cameras, and microphones for robots. Essentially, immersive VR allows the user to "be part of the environment" by projecting them into an entirely digitally generated world. In such a scenario, the person can manipulate virtual objects and interact in the simulated environment as in a Human-Human Interaction. On the contrary, with non-immersive VR, the user can see the virtual scene through a screen without losing the perception of the real world. In the last situation, generally, the person embodies a controllable simulated agent through the keyboard or a joystick.

Ensuring prompt response, particularly when the robot moves in the scenario, or its end-effector interacts with objects, such as grape bunches, are essential aspects of the digital world. All interactions are locally computed to achieve this, with only the outcomes synchronised across the network. Figure 4.1 distinguishes two fundamental actor types involved in the VR farming environment: Robot Operating System (ROS) actors tasked with controlling virtual robots and human agents.

- The Farming robot (robot actor) - implements the functionality for navigation, harvesting and pruning of grape clusters.
- The Logistic robot (robot actor) - implements functionality for the transport of boxes with and without grapes.
- Farmers (human actors) - real people who control 3D avatars.
- Spectators (human actors) - real people with some sort of interaction, e.g., voice interaction using a microphone and speaker, without avatars.

Each robot or human user is considered an individual client and operates a distinct client application to interact with the simulation and ensure smooth motion and communication. These interactions from robotic clients are translated into

[1] `https://pale.blue`
[2] `https://www.agrimessina.it/it`

different ROS actions, while those from human users are converted into input and output for VR hardware. Each client will interact with its matching actor, such as Robotic clients through ROS, VR clients through VR headsets, Desktop clients through microphones / speakers and traditional display/keyboard/mouse interfaces and finally, Mobile clients through microphones/speakers, touchscreens and potentially any available built-in sensor.

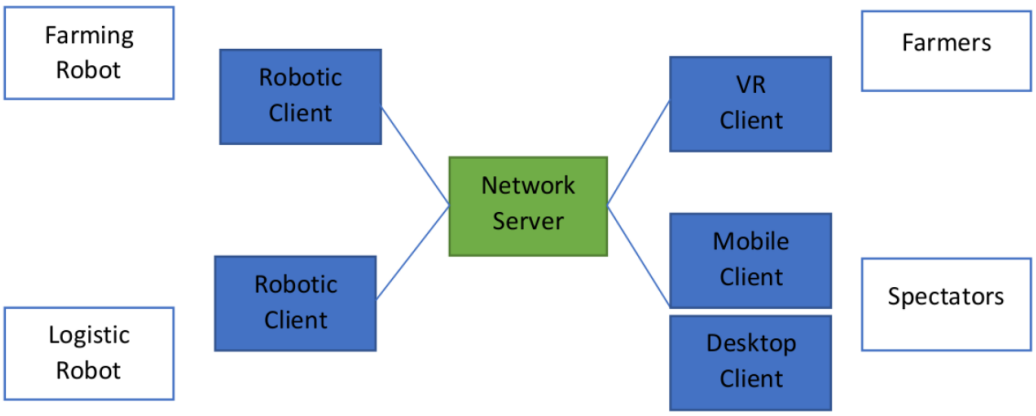


Figure 4.1. VR system actors.

Source: PaleBlue simulation environment developed for CANOPIES project.

4.1.1 Simulation Engine Setup

Development of this solution is mainly done using a 3D simulation environment, but for the robotic client, a separate part for the ROS interfacing is developed.

Unity 3D Engine

The simulation is realised using the Unity 3D[3] engine. This choice is because key components required for the simulation are available off-the-shelf and, thus, do not need to be developed separately. The Unity engine is a widely used 3D engine in many gaming and professional projects to simulate 3D interactive environments. It can simulate high-fidelity and high-performance 3D environments. The basic functionality of the engine can be extended with many packages that provide improvements and specialised support for specific needs.

Multi-user Environment

An extension package for Unity which supports multi-user training environment for VR. Users can use VR headsets in a shared environment to execute tasks together. They will be represented with full-body avatars and can use voice-interaction for communication. This environment supports dynamic interaction between the users which makes it possible to manipulate objects in a realistic way. All movements

[3] `https://unity.com`

of the users and relevant objects are synchronised across the network such that all users will experience the same situation at the same time. This extension has been successfully used in various applications. The simulated model representing the robot in the Figure 4.2 is a prototype under development, it is used purely as representation and does not reflect the final stage of development of the robot.

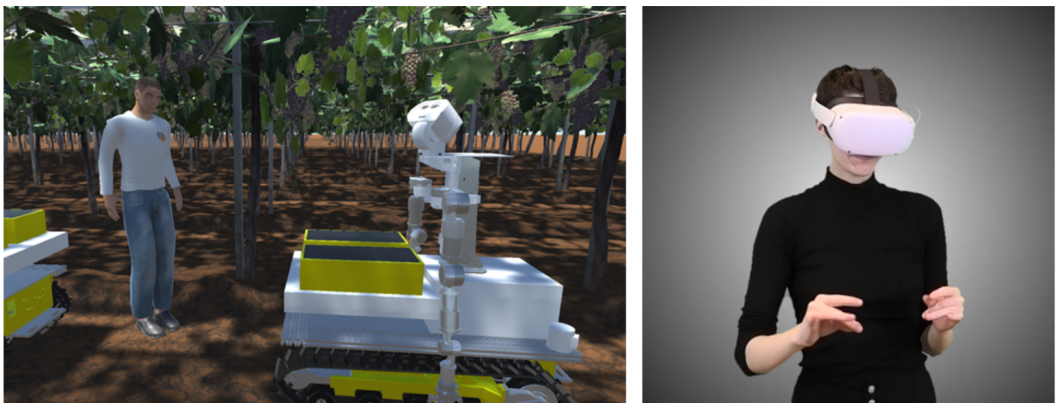


Figure 4.2. Multi-user Environment representation with simulator and VR headset.

Source: PaleBlue simulation environment developed for CANOPIES project.

Unity-ROS Bridge

The Unity-ROS bridge is a ROS package which implements the interface between Unity and ROS. Unity will communicate with this bridge using a TCP connection, so it is possible to run this bridge on a computer different from the one running the Unity simulation. It is also possible to run the bridge on Ubuntu while the Unity simulation is running on Windows. The unity simulation can communicate via an ROS interface using the ROS package. This allows us to send and receive ROS messages via a TCP connection to a specific ROS package which handles the interfacing between Unity and ROS. ROS topics, actions, states and parameters can be realised in this way which enables us to implement simulated components like sensors which communicate with other ROS components in the same way as the original hardware devices do. The bridge will do the actual implementation of the ROS topics, actions, states and parameters and forwards these interfaces to Unity. It is also possible to implement functionality in this bridge itself for cases when it is more fitting to do it there instead of in the Unity simulation.

4.1.2 Hardware Configuration

The system needs to be a distributed system where participants in the same simulation environment can be located in different physical locations. For this reason, we have separate hardware for each client and a central server to connect these together. The hardware for robotic clients is divided into two components, both fundamental for the overall system functionality. Starting from the left, the blue computers in Figure 4.3 execute ROS software, while the black ones handle Unity 3D

environment simulations. The VR setup for human clients encompasses VR hardware (on the right), including headsets and controllers, and optionally, a PC to run the 3D software. For optimal performance, the specifications that are considered for the various clients are presented below: For the hardware we distinguish 3 different setups:

- ROS client: Minimum 8-core with GTX1650 or equivalent
- VR client: Minimum 8-core with RTX3070 or equivalent for PC-based VR
- Desktop client: Minimum 4-core with GTX1650 or equivalent

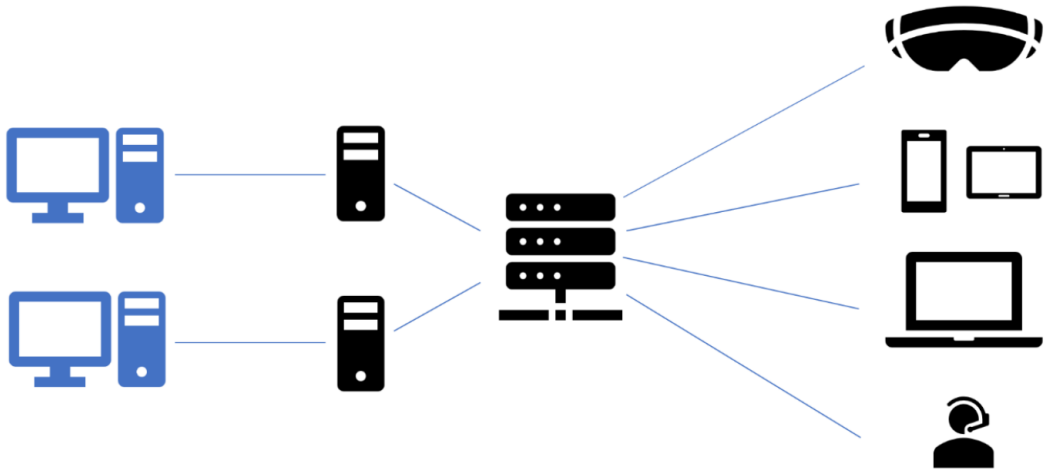


Figure 4.3. Distributed Simulation hardware architecture.

Source: PaleBlue simulation environment developed for CANOPIES project.

In order to facilitate rapid testing of the developed HRI approaches in ROS before conducting VR user studies, PaleBlue ensured compatibility of the virtual environment with Linux in a non-immersive setup. In this case, the software solution comprised a ROS package and a Unity simulation, designed to run on Ubuntu 20.04 LTS for the robot client. On the other hand, for the VR client platform, after a joint exploration of commercially available VR headsets that could suit our goals, the Oculus Quest 2 in Figure 4.4, was selected. This standalone headset offers both an optional cable and cable-free experience for expanded versatility. With a sharp display, powerful processor, and accurate motion tracking, users can engage in an immersive experience with intuitive controllers. Although native VR software support is lacking on Linux, using the Quest on its own or with a PC may require a Windows computer. Spectator clients can join the simulated environment on desktops (e.g., laptops) with Ubuntu 20.04 LTS or Windows, or on mobile devices (tablets or smartphones) with iOS or Android.

4.1.3 Virtual Sensor Integration

The digital twins of both farming and logistics robots are equipped with various sensors, positioned similarly to those on the real robot. This setup replicates the

Figure 4.4. Oculus Quest 2 VR Device used in simulation for immersive experiences.

Source: Meta Oculus Quest 2 Product Page.

robot's perspective, simulating their actual field of view. For the robotic interface, several sensors need to be simulated. Each of these sensors replicates the actual values to the simulation environment in real time and produces data on a ROS interface. Due to the simulation's computational complexity, the number and type of sensors that can be included in the scene depend on the hardware specifications of the computer running the simulation. However, it is expected that multiple sensors can be used simultaneously in this setup. The configurations of the sensors can be modified by providing a diverse URDF setup at the start of the simulation. RealSense cameras on the end-effectors could be re-positioned to achieve optimal views for tasks like harvesting and pruning. LIDAR and RGB-D cameras are essential to validate human detection, perceive the grape bunches in the simulation, recognise gestural commands, and ensure safety features like collision avoidance in the digital world as presented in Figure 4.5.

1. 2 Ouster OS1 LIDARs on the mobile base of the robot.
2. 2 Septentrio GPS-RTK receivers on the mobile base.
3. 1 SBG Ellipse IMU on the mobile base.
4. 2 RealSense D435i RGB-D cameras, each mounted on an end-effector of the robotic arm.
5. 1 RealSense D435i RGB-D camera inside the robot's head.

RGB Camera

A colour or RGB camera is a standard component in the Unity 3D engine. A free-hand RGB Camera was placed in the simulation, and configurable based on the position coordinates to follow the human avatars and robots and to record the environment with rostopic type *Image_raw*. It can be attached at any position in

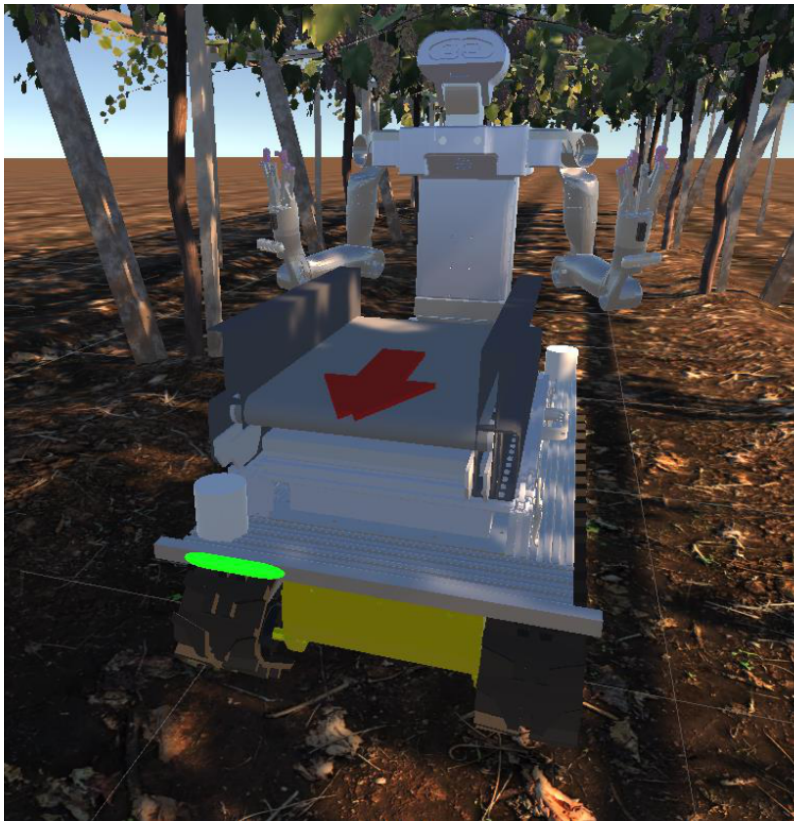


Figure 4.5. Farming robot with multiple sensors.

Source: PaleBlue simulation environment developed for CANOPIES project.

the scene, and its resolution, field-of-view, colour space, and related settings can be configured as needed. Typically, the image captured by the camera is rendered on a screen, but it is possible to capture this data and process ROS messages based on the images captured by individual cameras. The sensor properties were calibrated subjectively such that the output of the camera looks as it would be expected in given situations.

Depth Camera

The simulated depth cameras actually consist of two sensors: the depth sensor and the colour sensor. This matches the configuration of the real-world depth cameras in which those sensors are separate as well. The depth sensor itself is not a stereo camera setup like in the image above, but it is a single sensor placed in the middle between the stereo cameras. For the depth sensor, the image is actually generated from the depth map, colour map images and point-clouds of the unity camera. This depth map is used internally by the unity camera to compute the occlusion of objects in the camera. This is used to generate an image with the requested resolution in which every point contains information about the distance to the closest object at the given camera pixel. The depth values coming from the simulated depth camera are very accurate as they are directly derived from the scene, as can be seen

in Figure 4.6. This is the same as when using Gazebo. In real-world data, we have random and systematic errors that are not simulated in this way.

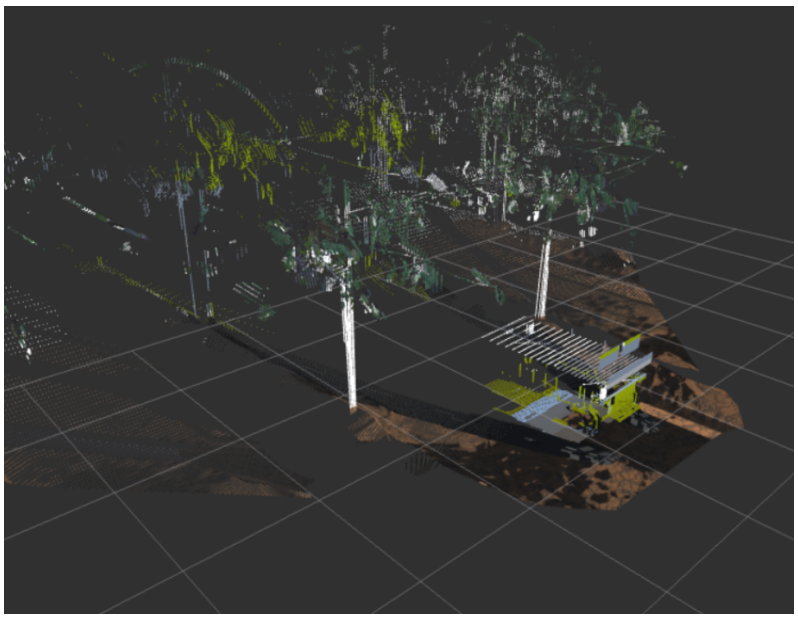


Figure 4.6. Simulated Depth camera Point cloud.

Source: PaleBlue simulation environment developed for CANOPIES project

LIDAR

A lidar sensor is simulated using a rotating camera sensor with a sample size equal to the real hardware lidar. For example, the Ouster OS1-64 will have a vertical resolution of 64 pixels and a horizontal 360 degrees resolution of 1024 pixels with a vertical field-of-view of 45 degrees. With a rotation frequency of 10Hz, the lidar simulation needs to compute 64*1024*10 = 655,360 points per second. Like with the depth camera, the default data produced by the simulated lidars is error-free. These sensors were replicated using two approaches:

- **3D ray casting using graphics depth maps on the GPU:** Depth maps are basically used to determine which parts of the meshes are occluded by other meshes and should, therefore, not be drawn. These were generated using GPU to render in the 3D scene. This solution is very fast as it is executed by the GPU, and data is already generated and made available in the GPU.

- **Point Clouds** The ROS point cloud is generated from the depth maps received from the GPU. The intensity value is based on the grayscale value of the texture of the mesh point of the point found by ray-cast. As this mesh point is not influenced by lighting or shading it should have a similar value as the real-world intensity value.

GPS/IMU Sensor

For the moving base, IMU and GPS sensors are used to get information about the position and orientation. The chosen sensors are the SBG Ellipse-E IME and a Septentrio GPS-RTK system. The SBG sensor uses GPS-RTK data to get more accurate location information. On the ROS side, the SBG sensor outputs the combined data, so no direct interface to the Septentrio GPS sensor is needed. In the simulation, only one sensor was placed to produce the same simulated data in the same way as the SBG sensor does. The values of the GPS/IMU sensors can be derived directly from the movements of the mobile base in the simulated environment.

4.2 Virtual Simulation Environment and Interaction

The virtual grape harvesting and branch pruning activities are performed in a 3D environment that closely replicates the real testing site in Agrimessina. The simulation environment in which the virtual representations of the robots and users interacting through avatars work together and consists of the following components to replicate the worksite vineyard in Aprilia, in the Lazio region (central Italy).

Ground and Boundaries

The terrain in the simulated environment is a deformable polygon mesh modelled to provide a non-flat surface that resembles the actual ground and enables the slipping of the mobile base tracks if they cannot move. These requirements can be fulfilled with the use of a Terrain component from Unity Engine. The interaction of the tracks of the mobile base with the ground is modelled by a Physic Material which defines the static and dynamic friction of the ground. Loose objects like rubble and rocks can be configured to the ground to mimic the real conditions of the environment further. Other variables like small plants, dried leaves, foliage and grass can be added at places around the plot, or random seed parameters generate them at random locations on the ground by configuring in the simulator settings.

Support structure

The support structure is a permanent structure called *trellis* to support the growth of the vines. It consists of vertical and diagonal concrete poles to guide the vine trunks and support for the following items as shown in the Figure 4.7: Iron wires to support the vine's branches, the irrigation system and the cover structure; Tunnel-shaped plastic cover supported by the arches; and an Irrigation system. All these components were modelled as a static structure. This means that it is a permanent, unmovable structure. It stops the movements of any object colliding with this structure without movement. This includes collisions with the mobile base, the dual arm, humans, grape vines and/or grape clusters.

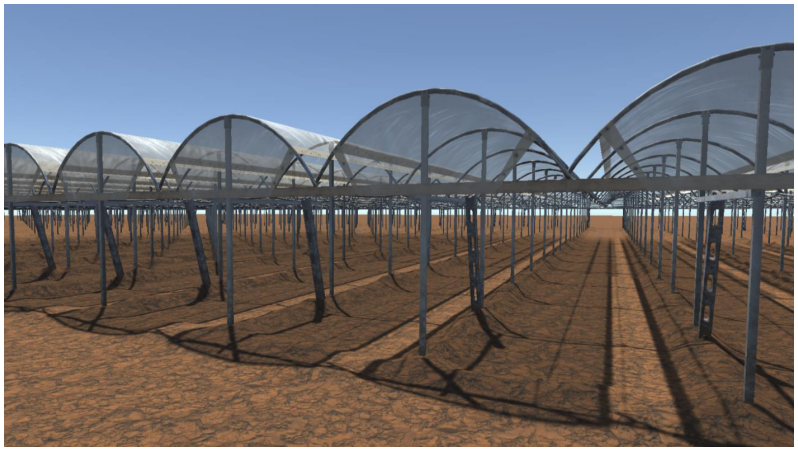


Figure 4.7. Virtual Simulation Environment trellis.

Source: PaleBlue simulation environment developed for CANOPIES project

Vegetation

The vegetation consists of the grapevines and the attached grape clusters. The specific requirements set for the clusters depend on the setting for harvesting grapes or pruning grape vines. Each plant will have the following additional characteristics for the harvesting period: 1. 2-3 grape bunches per cane; 2. 2-3 canes per branch; 3. 4-6 branches per plant. This results in 16-54 grape clusters per plant and an average of 32 clusters per plant.

The grape clusters will be a single model, meaning no movement between the grapes within a cluster will be possible. The peduncle connecting the grape cluster to the cane can be cut to remove the grape cluster from the cane. The simulated farming environment has two stages of ripeness: one with the correct colour for ripeness and the other with a (partially) green colour, indicating that the cluster should not be harvested yet. These grape clusters are modelled after taking samples from the worksite.

Figure 4.8. Types of grape clusters in the simulation.

Source: PaleBlue simulation environment developed for CANOPIES project

Boxes

The position of the boxes (with their contents) is synchronised when they are moving in their local space. When a box has a fixed place on a robot and this robot moves,

Figure 4.9. Vineyard at worksite after defoliating.

Source: PaleBlue simulation environment developed for CANOPIES project

the box position is not synchronised as it is not moving relative to the robot. These boxes come with an ID when spawned given the position so they can be removed from the environment using a ROStopic.

Environmental conditions

Environmental conditions such as light, dry weather and exposure can be parametrised to specific settings to resemble day, night and specific sunlight directions. This can be achieved by adjusting the colour temperature, directional light intensities, local light intensities, and exposure value.

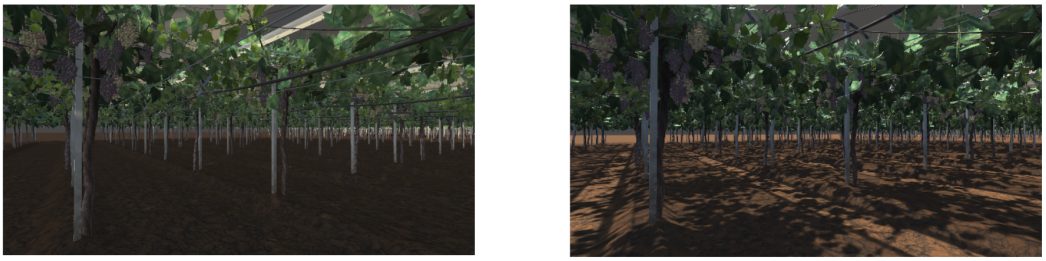


Figure 4.10. Simulated vineyard at different environment conditions resembling morning and noon.

Source: PaleBlue simulation environment developed for CANOPIES project

Robot Clients

The robot clients replicate a real robot as a digital twin. The Canopies consortium partner PAL Robotics provides the URDF file for both robots, which were

automatically constructed from the URDF (Universal Robot Description Format) description of the dual arm to ensure that it is the same as the real thing. This model can be easily updated according to the prototype release and simulate the robot's movements, physics interactions and the sensors attached to the robot. The communication with the simulated robot will use the same ROS topics and messages as the real robot such that it is possible to use the same robot control software with both the real and simulated robots. There will be two types of mobile base. A farming robot with a dual arm executes pruning and harvesting activities, and a logistic robot carries boxes for grapes and exchanges the boxes with the farming robot. The mobile base can move around using the differential drive implemented with two simulated tracks. This enables it to move forward and backwards, make turns and rotate 360 degrees on the spot. The tracks will interact with the objects on the ground which will result in a bumping movement when driving across uneven terrain. When climbing small elevations in the terrain, the mobile base will pitch and roll to follow these elevations.

Figure 4.11. Farming and Logistic robots with a human in virtual simulation.

Source: PaleBlue simulation environment developed for CANOPIES project

4.2.1 Human Avatars

The environment supports the introduction of multiple types of human avatars in a scene. All of these human avatar types can be combined within the same environment when necessary as shown in Figure 4.12. Static and animated humans may be added using the dynamic scene configuration, which allows the system to specify each human's position, orientation, and appearance. The key distinction between static and animated humans lies in the animator controller included with animated humans, enabling them to play various pre-recorded animations, as detailed below.

1. **Static Humans**: These figures remain stationary and can serve as obstacles that robots need to detect. An unlimited number of static humans may be

introduced into the scene, either by using the dynamic scene configuration process or by making ROS service calls.

2. **Animated Humans**: These are capable of movement based on pre-recorded animations, which can be started or stopped through ROS commands. Each animated human in the environment can run a unique animation selected from a built-in library. Additional animations can be created, removed, or modified within the Unity3D development environment using a separate build configuration.

3. **Controlled Humans**: These characters are driven in real-time by a user operating a virtual reality device. The device tracks the user's head and hands, and maps these movements onto the corresponding characters in the simulation. The resulting poses are derived by applying inverse kinematics to the upper body, combined with procedural animations for the legs.

Figure 4.12. Multiple Human Avatars in simulation environment.

Source: PaleBlue simulation environment developed for CANOPIES project

4.2.2 Virtual Human-Robot Interaction

This section describes the various methods of interaction between multiple humans and between robots in the simulation environment.

Physical Interaction

Humans can interact physically with robots through actions such as grabbing or pushing, resulting in forces exerted on the robot's joints. These forces are reported in ROS via the *Joint States* topic or through feedback from the *Joint Trajectory Controller* state, mirroring the interface used by the real robot. If this force feedback

is processed appropriately, the robot can behave in a compliant manner; if ignored, it attempts to maintain its pose and resists external forces.

An impedance controller was placed to enhance compliance. This approach adjusts the robot's stiffness and damping gains, potentially leveraging the Tiago impedance controller deployed in the Canopies robotic prototype. Dedicated topics publish each simulated Series Elastic Element (SEE) at each joint. The impedance controller was directly implemented and integrated into the simulation. When collisions occur between the robot and a human, the resultant joint forces can be applied to correct the robot's motion. Without such corrections, the robot effectively "fights" against human movements, generally prevailing due to its mechanical strength.

Human Hand Interaction The simulation applies dynamic physics only to the avatar's hands, allowing them to exchange forces with other objects. Other body parts, such as the legs and head, remain kinematic and thus cannot be moved by external forces. Consequently, collisions with the robot's mobile base do not displace the avatar's legs or head; instead, the mobile base halts, regardless of its strength. Hand forces are computed from the offset between the avatar's hand in the virtual space and the user's real controller. Under normal conditions, this offset is zero, ensuring that the virtual and real hands coincide. If the virtual hand collides with an object, the robot and the avatar's hand are prevented from penetrating it, whereas the physical controller can continue to move as shown in Figure 4.13. The difference in positions generates a proportional force, limited to 100 N for every metre of separation.

Figure 4.13. Human hand and the controller position without collision vs collision. The yellow arrow indicates the force applied to the robot.

Source: PaleBlue simulation environment developed for CANOPIES project

Users also experience a subjective form of force feedback through two main mechanisms. First, the object's visual response to applied forces (influenced by its mass and friction) offers cues about its weight and inertia. Second, larger movements are required to manipulate heavier objects, causing an internal sense of effort or

muscular strain. Although the user's muscles do not directly feel a counterforce, the combined visual feedback and increased physical motion contribute to an overall perception of force feedback.

Voice and Sound Interaction

Participants can interact orally within the table-grape simulation environment, mirroring real-world conversational dynamics. In immersive settings, headsets equipped with integrated microphones and speakers facilitate voice communication, whereas in non-immersive experiences, Bluetooth devices can serve the same purpose. Alternatively, the system supports virtual avatars equipped with simulated microphones and speakers on the robots, thereby enabling flexible arrangements for speech-based user studies.

Furthermore, to investigate methods of enhancing participants' awareness of the robot's actions, PaleBlue integrated functionality within the simulation to load and play custom sounds, thereby improving the auditory feedback provided to users.

Light Indications

To deliver notifications, alarms, and warnings, the robot can combine auditory signals with indicator lights that activate in specific patterns to denote various conditions. Humans and other robots readily perceive these visual cues, thereby enhancing the clarity of the robot's alerts.

Gestural Interaction

Human avatars employ dynamic physics for multiple hand poses, with button inputs dictating specific gestures (see Figure 4.14). Introducing gestural interaction in the virtual table-grape vineyard environment is crucial for accurately representing finger positions during gesture execution. Because hand recognition systems may be sensitive to slight misalignments, a predefined library of hand and finger poses was provided to PaleBlue to facilitate the study of gesture-based human-robot interaction (HRI) in the simulator. Each custom-designed hand pose is linked to a controller button, while the desired avatar gesture can be modified through ROS topics and services. Furthermore, full-body gestures were integrated into avatars, which were further discussed in Chapter 5 and 7.

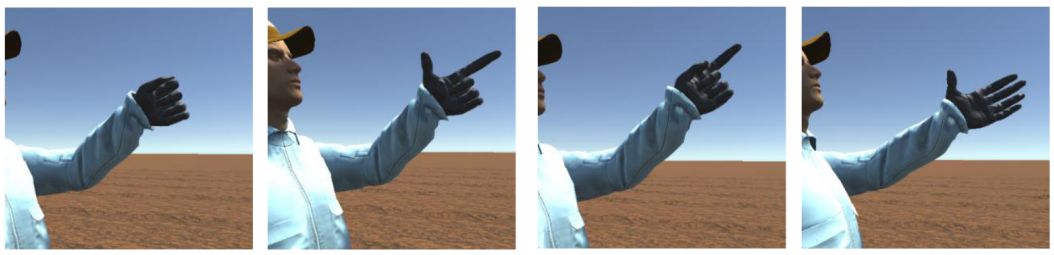


Figure 4.14. Various hand poses with controller input.

Source: PaleBlue simulation environment developed for CANOPIES project

VR Applications in HRI

Virtual Reality (VR) plays an important role in human-robot interaction (HRI), providing a controlled yet highly flexible environment for *simulation-driven user studies, bidirectional learning, and synthetic data generation.* These environments enable the replication of real-world scenarios while capturing a wide range of human behaviours, making them invaluable for developing and refining interactive robotic systems.

User Studies and Interaction Modelling Extensive user studies similar to the section section 4.3 can be conducted in VR environments to examine human-robot interactions, incorporating participants with diverse body types and demographic backgrounds. Systematic data collection has revealed a consistent improvement in user scores across multiple interactions, indicating that participants gradually *adapt to the robot's interaction style* while developing a deeper understanding of its intentions. By incorporating *closed-loop feedback mechanisms*, interaction models are fine-tuned based on user responses, fostering a more *intuitive and adaptive collaboration* in shared spaces.

Bidirectional Learning for Adaptive HRI For seamless human-robot collaboration, bidirectional learning is essential. Robots must not only execute commands but also *perceive, reason, and respond* to human intentions. Through *gesture recognition, pose estimation, and multimodal inputs*, robots can adapt their behavioural strategies over time, leading to a more natural and efficient interaction process.

Synthetic Data Generation in VR Simulations VR simulation environments offer a particularly promising avenue for *producing synthetic data* to train and evaluate AI-driven robotic systems. These environments can be *customised to replicate real-world conditions*, facilitating data collection on *pose estimation* and *gesture recognition* under controlled yet varied conditions. By simulating *diverse gestures, postures, and environmental factors* (e.g., occlusions, challenging lighting, or dynamic obstacles), developers can iteratively refine algorithms before deploying them in real-world applications.

Beyond *pose and gesture data*, VR simulation can incorporate *vocal utterances*, replicating *natural multimodal human-robot interactions* involving *speech, auditory cues, and voice commands.* The ability to generate combined datasets that integrate *visual, skeletal, and auditory* features significantly enhances the training of *robust recognition and dialogue systems*, allowing robots to respond to more nuanced human inputs.

Rapid Prototyping using Image-to-3D Modelling An additional advantage of *synthetic environments* is their ability to facilitate *rapid prototyping* through recently developing and thriving *image-to-3D modelling techniques.* By leveraging *automated 3D reconstruction*, 2D images can be converted into *high-fidelity 3D environments*, enabling researchers to create and modify virtual scenes that closely resemble real-world settings. This technique significantly reduces reliance on physical prototypes, streamlining the *iterative design cycle for robotic applications.*

Robotic Simulators for HRC and Data Collection Robotic simulators serve as critical tools for *testing algorithms, evaluating human-robot collaboration (HRC), and generating synthetic datasets.* The simulation platforms like *HumanTHOR* [210, 254], for instance, facilitates the benchmarking of HRC tasks by assessing *success rates and execution times.* These simulators also enable *zero-shot Sim2Real transfer*, allowing models trained in simulation to perform comparably in real-world scenarios.

Advantages of VR-Based Simulators The *cost-efficiency* of VR testing minimises physical prototyping expenses while reducing the risk of costly design flaws. Additionally, VR simulations allow for *safe exploration of high-risk tasks*, protecting both humans and robotic systems. The *iterative nature of virtual environments* accelerates the development process by enabling rapid modifications to robotic designs without the constraints of physical testing. Furthermore, VR provides *immersive training environments*, equipping human operators with the skills necessary for *effective robot interaction.*

Challenges Despite its advantages, VR-based simulation faces *several challenges. Limited asset libraries* constrain adaptability, as new robotic tasks often require the manual integration of additional 3D models. While simulators may not be cost-effective for short-term use, they prove to be cost-efficient over the long term—provided they remain relevant and are constantly developed and maintained. The *Sim2Real Gap* though solved in the case of CANOPIES project, remains a key issue in general HRI applications, where the transition from simulated learning to real-world deployment is hindered by *visual inconsistencies, simplified physics, and rendering limitations.* Moreover, current *XR frameworks* often lack *cross-platform extensibility*, making it difficult to adapt simulations across multiple VR headsets.

4.3 Immersive and Non-immersive User Studies

The application of VR in all its forms of experience (either immersive or non-immersive) is highly employed in different fields of study, such as medicine, education, industry, aerospace, architecture and history. Essentially, immersive VR allows the user to “be part of the environment" by projecting him/her into an entirely digitally generated world. It allows the person to manipulate virtual objects and interact in the simulated environment as it happens in a Human-Human Interaction. On the contrary, with non-immersive VR, the user sees the virtual scene through a screen without losing the perception of the real world. Both immersive and non-immersive cases have been considered to collect data, and conduct initial experiments on people’s perception and the robot’s understanding. In particular, in other application scenarios, it was demonstrated that acquiring data and evaluating preliminary solutions or complete approaches in VR could provide multiple benefits in terms of experimental settings, costs, and safety [149]. Nevertheless, very few studies cover the use of VR in collaborative robotics for precision agriculture scenarios. However, none of the available works describe VR as a reliable solution for both data acquisition and system evaluation. The developed simulation environment

could be a reliable solution as it was made as a digital twin to the table-grape worksite. Thus investigating the adoption of this novel technology to carry out HRI-related experimental activities and collaboration between human workers and robots, such as in grape harvesting and pruning is essential to enhance the robot's comprehension of the human (and vice-versa) while performing collaborative tasks in CANOPIES application scenario, the interaction has been considered from an agent communication perspective, by analysing the message type (speech act) exchanged between people and robots.

Two user studies have been conducted in VR, differing in the type of experience: immersive and non-immersive. In both experiments, the 81 participants involved vocally interacted in Italian with the agronomic robot in the table-grape vineyard simulated environment of the CANOPIES project, by providing utterances with diverse tones and content details: Information, Command, Request, and a combination of them. Verbal feedback of the robot's comprehension (as a classification of the sentence provided in input by the person) was delivered to the user, along with feedback in terms of the robot's motion in the virtual world (in response to the command execution). These user studies conducted in an Immersive Virtual Environment (IVE) and Non-Immersive Virtual Environment (NIVE) were compared to discuss participants' feelings and identify the most suitable type of experience that could be adopted for the upcoming experiments in the CANOPIES project. In both studies performed in a virtual reality environment created in the project context, the developed speech act classification system was only used as a tool to evaluate and compare the two experiences. To this aim, the research questions addressed in this section are the followings:

- Q1: Is the IVE experience preferable to conduct user studies in the table-grape scenario compared to NIVE?
- Q2: How the experience in NIVE could influence users' feelings of the immersive world compared to people participating only in IVE?

4.3.1 Study Design

A series of user studies were conducted to compare immersive and non-immersive virtual reality (VR) experiences for human–robot interaction (HRI) within the CANOPIES project. The researchers primarily recruited students from the host institution, based on the assumption that young adults would provide detailed and meaningful insights into the adoption of such emerging technology. The studies were carried out over one month in a dedicated laboratory, where student engagement was notably high.

Before commencing the experiment, each participant was asked to sign a consent form to allow voice recording during interactions with the virtual robot. Subsequently, a concise overview of the CANOPIES project and the objectives of the experimental activities was provided. This explanation placed particular emphasis on the agronomic robot's capabilities, since participants would interact with it in the virtual environment, and on various table-grape types (including “white pizzutello", “black pizzutello", and “black magic") to enrich the interactive experience. Although most participants had an engineering background, which might have limited their

use of domain-specific terminology for vineyard-related tasks, their behaviour was nonetheless observed throughout the study as the tasks changed. At the end of the session, each participant completed a questionnaire comprising 23 statements, rated on a 5-point Likert Scale from 1 (*Strongly Disagree*) to 5 (*Absolutely Agree*). The questionnaire was based on the user study evaluation for immersive virtual environments (IVEs) presented in [238], with certain statements reviewed and adapted to allow a balanced comparison between non-immersive virtual experiences (NIVEs) and IVEs in line with the experimental goals. Vocal utterance data acquired during this study was analysed and discussed in depth in section 6.1.

Non-Immersive Experience

During a two-week period, 40 participants (32 male, 8 female) were involved in evaluations using the Non-Immersive Virtual Environment (NIVE). Despite the one-hour duration of the experience, a sufficient number of users participated. Approximately 75% of these participants had prior experience with VR applications, and 17.5% reported negative feelings towards VR usage in the past. Concerning domain expertise, 27.5% had previous familiarity with vineyard activities, whereas 40% had worked with robots in the NIVE. In addition, 52.5% had interacted with real robots on multiple occasions.

To run the CANOPIES non-immersive simulator (developed by PaleBlue, Norway) and the accompanying speech system, an Alienware Aurora Ryzen Edition R14 computer was employed. This machine featured 64 GB of RAM and an AMD Ryzen 9 5950 X 16 core processor running Ubuntu 20.04, providing sufficient computational power for the Unity Linux executable of the simulator. In order to minimise background noise and record spoken commands, participants used a headset equipped with a toggle to enable or disable the microphone. Within the NIVE, each user controlled a virtual avatar—representing their own position—via keyboard inputs. They could adjust the horizontal camera view using either the keyboard or mouse, enabling them to observe the robot's autonomous movements. These movements were determined by the system's interpretation of spoken commands, cross-referenced with relevant terms in its knowledge base.

Immersive Experience

A separate set of studies focusing on the Immersive Virtual Environment (IVE) was conducted over the course of one week. Similar to the NIVE experiment, the sample primarily consisted of Master's and PhD students in Engineering and Computer Science, resulting in a group of 41 participants. Of these, 25 had previously taken part in the NIVE, while 16 were new to the research. As before, there was a higher representation of male participants (33) compared with female participants (8). Approximately 61% had utilised immersive VR applications in the past, with 19.5% indicating negative experiences. Furthermore, 43.9% of participants were knowledgeable about vineyard activities, 48.8% had worked with real robots, and 17.1% had experience with androids in immersive environments.

Two high-performance machines were employed for the IVE studies: the aforementioned Alienware Aurora Ryzen R14 (running Ubuntu 20.04) for the speech

system, and an Alienware x17 R2 notebook (running Windows 11 Home) equipped with 32 GB of RAM, a 12th generation i9 CPU, and an Nvidia RTX 3080 Ti GPU. The Alienware x17 was connected to an Oculus Quest 2 headset via an Oculus Link cable, allowing real-time observation of each participant's viewpoint during the experiment. To limit potential discomfort, the maximum exposure in the IVE was set to 30 minutes. A handheld Jabra microphone served as the communication device, enabling participants to press a button to begin and end speech recordings.

In contrast to the NIVE, participants in the IVE were free to walk within a $2\,\text{m} \times 2\,\text{m}$ space in the laboratory. This area facilitated physical movement in the immersive vineyard, allowing users to monitor the robot's activities and responses as they navigated the environment. The robot's motion in the IVE was not directly controlled by the ROS-based speech system, as a robust connection between ROS and Unity was not yet established. Instead, the robot's actions—based on the user's spoken commands—were manually replicated using the Oculus Quest 2 controllers, ensuring fidelity to the intended interaction despite the temporary limitations in system integration.

4.3.2 Results and Observations

The outcome of the questionnaires from user studies in the NIVE and the IVE are presented and discussed in this Section. Precisely, the results of the first 23 questions are reviewed based on their measurement categories: Immersion, Presence, Engagement, Flow, Emotion, Skill, Judgement, Experience Consequence, and Technology Adoption.

Users' impressions are compared by identifying two macro-groups: people involved in the NIVE (40 subjects) and participants that took part in the IVE (40 subjects). Moreover, users experiencing the immersive scenario can be distinguished into two sub-groups: people participating only in the fully-immersive environment (16 subjects) and a set of users that first experienced the NIVE, then the IVE (25 subjects). Such a choice was driven by our interest in investigating how the experience in the non-immersive scenario could influence users' feelings of the immersive world compared to people participating only in the immersive environment. Hence, the three groups analysed in the measurements are summarised below:

- people participating only in the first experiment experiencing the NIVE
- people participating only in the second experiment experiencing the IVE
- people participating in the second experiment in the IVE, but they were first involved in the NIVE study

All the questions with the corresponding average score, emerging respectively from each of the aforementioned groups of participants, are available in Table 4.1, with the highest value for each statement emphasised in bold. At the same time, a graphical representation is provided in Figure 4.15.

However, an examination of the 81 returned questionnaires revealed that, in terms of the ***Engagement*** category, most participants regarded the virtual world's visual elements as highly beneficial for interacting with the robot, thereby enhancing

Statement	Measurement Category	NIVE Experience	IVE Experience	IVE + NIVE Experience
1. The visual aspects of the virtual environment helped me in the interaction.	Engagement	4.18	4.25	**4.32**
2. I felt involved in the virtual environment experience.	Engagement	4.05	**4.69**	4.32
3. I could actively survey what was happening in the virtual environment.	Presence	4.48	3.94	**4.56**
4. I was able to examine objects closely.	Presence	3.68	4.13	**4.60**
5. I could examine objects from multiple viewpoints.	Presence	4.25	4.31	**4.40**
6. I felt proficient in moving and interacting with the robot.	Presence	**4.10**	3.88	3.96
7. I could concentrate on the task rather than on the devices.	Presence	**4.28**	4.00	**4.28**
8. I was so involved in the virtual environment that I was unaware of things happening around me in the real world.	Immersion	2.73	3.31	**3.40**
9. I was so involved in the virtual environment that I thought I was in the scene.	Immersion	2.50	3.06	**3.68**
10. I was so involved in the virtual environment that I lost track of time.	Immersion	2.83	3.19	**3.48**
11. I knew what to say and/or do in each scenario.	Flow	3.50	3.38	**3.92**
12. I was not worried about people's judgment during the interaction.	Flow	**4.50**	4.25	4.48
13. Personally, I would say the virtual environment is a valid solution to test the robot's comprehension.	Judgement	4.48	**4.56**	4.36
14. Personally, I would say the experience in the virtual environment is exciting.	Judgement	3.65	**4.19**	3.84
15. I felt confident using the keyboard/Oculus Quest 2 and interacting through the headset/microphone (depending on the type of experience).	Skill	**4.28**	4.00	3.92
16. If I use the same virtual environment again, the interaction would be faster and more spontaneous.	Technology Adoption	**4.13**	4.06	4.04
17. Using devices (headset, keyboard/Oculus Quest 2, microphone) to interact in the simulated environment is simple and practical (depending on the type of experience).	Technology Adoption	**4.33**	4.19	4.00
18. I would like to interact more often with virtual systems similar to this experience.	Technology Adoption	3.65	**3.94**	3.92
19. I suffered from fatigue, headache, eyestrain, vertigo or nausea during my interaction with the virtual environment.	Experience Consequence	1.05	**1.75**	1.60
20. I enjoyed the experience so much that I felt energised at the end of the experience.	Emotion	3.40	**3.56**	3.40
21. During the interaction in the virtual environment, I felt anxious.	Emotion	1.48	**1.81**	1.44
22. I enjoyed dealing with interaction devices.	Emotion	3.95	**4.50**	4.20
23. I felt natural interacting vocally in the virtual environment.	Emotion	**4.35**	3.56	3.92

Table 4.1. English translation of the proposed user experience questionnaire with average scores from the three groups of participants.

their overall involvement. On average, responses in the Engagement category received a score of 4 across all groups. The lowest mean (4.11/5) arose from the group participating solely in the non-immersive virtual experience (NIVE), whereas the highest mean (4.47/5) occurred among individuals who took part only in the immersive experience (IVE).

When assessing the ***Presence*** category, several factors were considered, including participants' ability to manage the robot actively, inspect objects in the environment

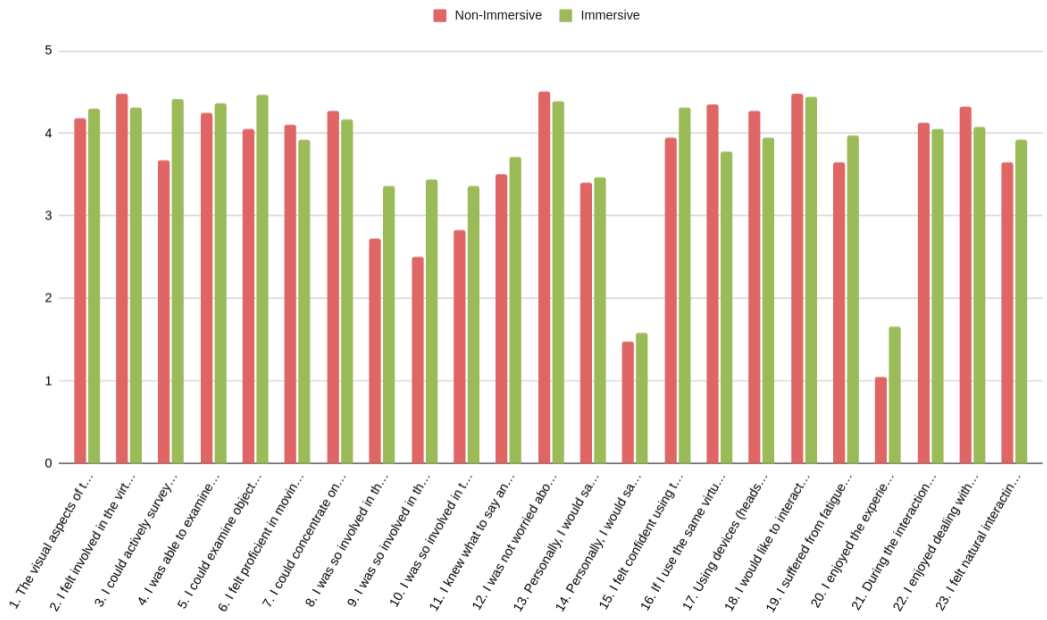


Figure 4.15. Graphic representation of the questionnaire responses. Non-immersive average values are presented in red, while immersive ones are in green.

(e.g. grapes, leaves, and branches) from diverse distances and viewpoints, move freely, interact with the android, and concentrate on the task rather than the hardware. A comparison of responses from the three groups indicates that those who first experienced the non-immersive scenario and then progressed to the immersive setting demonstrated the strongest sense of presence in the IVE (4.36/5). By contrast, participants involved exclusively in the immersive study reported a lower sense of presence (4.05/5) than those who only engaged in the NIVE (4.16/5).

For the ***Immersion*** measure, which examines participants' sense of time distortion and awareness of the real environment, findings suggest that immersive scenarios delivered a more pronounced sense of immersion compared to non-immersive ones (2.68/5). Notably, the highest average value for immersion (3.52/5) was reported by those who had previously taken part in the NIVE.

In evaluating ***Flow***, an assessment was made of whether participants felt comfortable with the interaction, including what to say and how to act without concern for external scrutiny when engaging with the virtual robot. Survey data reveal that the highest average score (4.2/5) was achieved by the group who participated in the IVE following a prior experience in the NIVE. Conversely, the lowest flow score (3.81/5) was observed among individuals who only took part in the immersive setting.

The ***Judgement*** category focuses on feedback regarding the suitability of the virtual environment for evaluating the robot's understanding of user inputs, as well as its overall level of enjoyment. In this category, participants involved exclusively in the IVE returned the highest average (4.38/5), while the NIVE resulted in the lowest mean (4.06/5).

Skill was measured to gauge participants' confidence using the relevant inter-

action devices: the keyboard and headset for the NIVE, and the Oculus Quest 2 with a Jabra microphone for the IVE. The questionnaires indicate that those who took part solely in the NIVE felt more comfortable with the equipment (4.28/5), whereas participants who engaged in the IVE after the NIVE provided a lower score (3.92/5).

Technology Adoption assesses the extent to which individuals are inclined towards technology and the degree of interest they show in using it. Indicators included participants' beliefs regarding the potential to improve speed and communication through continued use, the practicality of the interaction devices, and their willingness to engage frequently in either NIVEs or IVEs. The highest ratings for technology adoption (4.06/5) came from those solely experiencing the IVE, followed by the group that participated in the NIVE (4.03/5).

In the ***Consequence*** category, the questionnaire evaluated whether participants experienced any adverse reactions (e.g. fatigue, headaches, eye strain, vertigo, or nausea) while interacting in the environment. As anticipated, the IVE participants reported the highest values. Notably, the group who participated only in the immersive scenario yielded an average of 1.75/5, followed by those who had also taken part in the NIVE (1.6/5). By contrast, the non-immersive cohort presented the lowest mean (1.05/5).

Finally, the ***Emotion*** category aimed to capture both positive and negative emotions experienced in each scenario. Positive indicators involved ratings of energy post-experiment, preferences for the interaction devices, and the perceived naturalness of verbal communication, whereas negative aspects centred on participants' anxiety levels. Positive emotion scores appeared relatively consistent among the three groups, ranging from 3.84/5 (those who participated in the IVE after the NIVE) to 3.9/5 (the NIVE-only cohort). Nevertheless, the lowest level of anxiety (1.44/5) emerged in individuals who had experienced the IVE following the NIVE, while the highest was recorded in those who had taken part only in the IVE (1.81/5).

To test the statistical significance of the results presented in Figure 4.15, a two-tailed unpaired t-test was applied using $p < 0.05$ as a threshold. Notably, a statistically significant difference in mean was identified for questions 3, 5, 8, 9, 16, and 20, and borderline values ($p \approx 0.055$) were observed for questions 10 and 15. Differences in mean for questions 17 ($p \approx 0.06$), 22 ($p \approx 0.07$), and 23 ($p \approx 0.13$) were not statistically significant.

4.3.3 Discussion and Implications

Identifying the most suitable virtual reality (VR) experience for a user study can be challenging. An analysis of questionnaire data indicates that six out of nine measurement categories—*Engagement*, *Presence*, *Immersion*, *Flow*, *Judgment*, and *Technology Adoption*—received their highest ratings from participants in the immersive virtual experience (IVE). Moreover, most users who had previously participated in a non-immersive virtual experience (NIVE) preferred the immersive approach. Nonetheless, extended use of the Oculus Quest 2 headset and the limited space for user motion emerged as concerns in an open-ended question about negative aspects of the experience.

Based on these observations, forthcoming experiments in the CANOPIES project

predominantly employed immersive user studies to gather data and assess human-robot interaction (HRI) methods that do not require individuals to stay in the virtual environment for longer than thirty minutes or engage in extensive physical movement, thereby reducing the risk of discomfort. Consequently, immersive approaches will be avoided when a person's mobility is necessary for evaluating specific solutions but sufficient physical space is unavailable. Although teleportation can provide movement in confined areas, researchers have found that this unnatural type of locomotion may heighten motion sickness [253]. Therefore, non-immersive settings remain a valuable alternative for more extensive HRI investigations and for evaluating robotic performance across different regions of the virtual environment.

Chapter 5

Synthetic Data Generation and Evaluation

Understanding and interpreting human poses and gestures is indispensable for creating natural interactions between humans and machines across collaborative domains, such as smart manufacturing and precision agriculture. However, obtaining real-world datasets can be expensive, time-consuming, and entangled with ethical and logistical challenges. Synthetic data has gained prominence as an alternative to address these obstacles. It allows researchers to generate extensive and varied samples that accurately reflect human motions, gestures, and speech without incurring the complexities of real-world data collection. In the context of human-robot interaction (HRI), advanced simulation platforms enable the assessment of system architectures for pose estimation, gesture recognition, and spoken dialogue under realistic conditions. They also integrate environmental complexities and accurately represent motions, interactions, and digital objects [63], helping developers refine algorithms, accelerate deployment, and reduce costs.

This chapter details how synthetic datasets generated within such simulations can support *pose estimation*, *gesture recognition*, and *vocal utterances*. It also examines how *text / image-to-3D modelling* expedites prototyping by converting standard figures into detailed objects that can be used in virtual environments. Combining these techniques facilitates the creation of robust ground-truth data for early-stage development, enabling faster iteration and safer, more cost-effective evaluation of emerging HRI solutions.

5.1 Data Generation Strategy

Synthetic data was necessary for training machine learning models, especially for tasks in computer vision and natural language processing. It allows the creation of realistic and diverse datasets for training using virtual characters, enabling them to exhibit lifelike behaviours, gestures, and expressions. However, one persistent challenge is ensuring diversity within the generated data. The following sections explain how synthetic data was generated for training machine learning algorithms using the virtual simulation.

5.1.1 Character Creation

Character creation presents several challenges while designing characters with variability as the process is time-consuming, has limited variability, has consistency issues, repetitive tasks, iteration challenges, skill dependency, and scalability issues. It demands meticulous attention to detail, making the achievement of consistent results across multiple variations difficult. To overcome these challenges, alternative approaches such as procedural generation or automation tools may offer solutions to streamline the character design process and achieve more significant variability across character sets. While Machine learning and Generative AI tools offer numerous models for digital humans, integrating them into existing software platforms poses several challenges due to configuration mismatches. Conversely, software solutions like Autodesk Character Generator[19], Unreal Engine's MetaHuman Creator[80], and Unity's Character Creator[246] empower users to sculpt, texture, and animate characters with unparalleled detail, variability, and realism.

Autodesk Character Generator was chosen for creating characters as it offers required diversity across various demographics, including gender, age, ethnicity, skin colour, eyes, hair, type, body size, clothing, and height. With this tool, 30 unique characters were generated, comprising both male and female subjects with several variations. Each character was meticulously customised utilising the tool's extensive library of body types, outfits, hairstyles, and physical attributes. The character design process was aimed to ensure a balanced representation of various demographics to capture the diversity of actual humans accurately.

While the character design option offers variability on several factors, the Generate character tool also offers configuration to suit the needs of users based on what software tool they are going use. Figure 5.1 illustrates how a user can tailor a character's features to their specific requirements. This customisation encompasses everything from the character's detail level (polygon density) to their facial expression and even the file format in which the character is saved. All the generated characters were then tested for compatibility with the rigs and the file format to fit the software Unity Hub.

5.1.2 Animation Extraction

Motion capturing through videos using RGB-D cameras allows to recreate the motion into animation using animation extraction tools. When conducting human motion capture with the camera, focusing on camera placement, character positioning, and ground motions is crucial. Positioning the camera 2-4 meters (6-15 feet) away from the subject and keeping it stationary and perpendicular ensures optimal capture conditions. Additionally, capturing the full body, half body, or face of the human subject without any occlusions or objects obstructing the view is essential for accurate tracking. When recording ground motions, angling the camera at a 3/4 angle helps prevent key joint occlusions. These precautions can ensure smooth capture of the motion for better animation and its extraction. Defined gesture motion for each gesture was captured into rosbags at several distance settings using an Intel RealSense D435i Camera, which has robust depth sensing capabilities and an inertial measurement unit (IMU). These rosbags were processed into video clips per gesture

Figure 5.1. A sample of the Characters generated compatible to unity after the designing

animation.

We used *Deep Motion*[66] tool for Animation extraction as it introduces a novel and easy approach to streamline the process through the application of advanced AI algorithms. Unlike traditional manual extraction methods, Deep Motion offers an automated solution, aiming to reduce time and effort while ensuring the extraction of high-quality animations. By analysing the video footage, motion capture data or existing animations, Deep Motion employs precise algorithms to identify and extract relevant motion data efficiently using real-time 3D body tracking technologies. To ensure the highest quality animations, it is essential to adhere to specific video capture guidelines discussed above while recording the gesture motion data. While capturing gesture motion from the face and hands requires intricate details for facial tracking and hand tracking algorithms to produce good animation, but doesn't need facial data.

To enhance the realism and quality of animations, DeepMotion's motion capture system includes features like the Physics Filter and Motion Smoothing. The Physics Filter reinforces joint limits, addresses self-collisions, and reduces clipping to improve animation realism. On the other hand, Motion Smoothing utilises advanced AI filters

Figure 5.2. Virtual character with Grape field background in simulation

to remove jitter and enhance animation smoothness. Lighting and contrast play crucial roles in motion tracking, with neutral lighting conditions and high contrast between the subject and background recommended for optimal results. Additionally, considerations such as clothing choice, foot locking, and video length restrictions contribute to achieving high-quality motion capture results. Final animations were downloaded with initial T-pose and I-pose options in *.fbx* format rigged to an empty mannequin-like character to be fitted with the virtual human characters we have created.

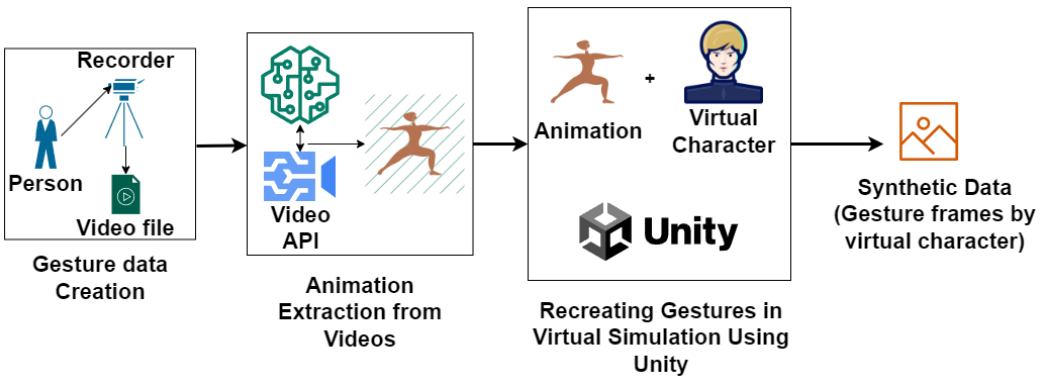


Figure 5.3. Synthetic data extraction pipeline from simulation

5.1.3 Character Merge to Environment

In the process of merging character models and animations in Unity [247], several key steps are typically followed. Firstly, the character model is imported into the Unity project, ensuring it is correctly rigged as humanoid to enable animation. Unity supports various rigging methods, including humanoid and generic rigging. Gesture Animations were then imported and subsequently added to the project. An Animator Controller is created to manage the animations, with each animation state represented within the controller. Transitions between states are defined to ensure smooth movement between different animations. Scripting can be employed for more intricate control over animations, allowing for dynamic responses to characters based on triggers. Once set up, the Animator Controller is attached to the character model, integrating animations with the character's behaviour. These animations can be called using ROS commands whenever the character needs to perform the intended gesture in front of the robot. When multiple characters are simulated, the animator control can map one or many gestures to one or more characters, which can be triggered by ROS commands that identify each character with a unique identifier.

Throughout this process, attention to detail was crucial for seamlessly integrating character models and animations within Unity. There was no adequate metric for this evaluation, so the animation's effectiveness was measured through human observation. Testing was essential to ensure that animations function as intended, which helped identify any issues or inconsistencies, allowing adjustments for refinement and optimisation of animations and their transitions. Under-performing animations were discarded and re-recorded from different angles, which can capture the movement optimally to generate the proper animation to map back into the simulation. These steps created an immersive experience where characters move fluidly and respond dynamically to the simulation environment, enhancing overall engagement.

5.1.4 Real-Time Data from the field

The real-time data was captured using Intel RealSense Depth camera D435i with IMU and stereo vision in the table-grapes field in Aprilia (Lazio), a worksite for the project. The gesture data was recorded after consent to use and publish it for research purposes. To this aim, 8 persons from different age groups and genders participated in recording the defined gestures, as shown in Figure 5.4. These gestures were recorded at the resolution 1280x720 at 30 FPS with depth RGB-D information. The gestures were performed and captured at 4 different distance settings, starting at 1.5 meters from the camera and increasing the interval by considering multiples of 1.5, to store such information in RGB-D rosbags. All 21 gestures were recorded into a single ROS-Bag file for each distance per person. All the files were then post-processed to extract videos and frames of the performed signs based on the type of the gesture(Static/Dynamic). Then, Mediapipe's pose estimation algorithm was applied to verify if all the key points of the poses were being detected. For convenience in experimentation, 12 (11 static gestures + standing pose) were primarily considered, with all belonging to the initiation and navigation of the robot in different directions. For the purpose of classification, an extra class of Unknown Pose was added to categorise a gesture out of these 12 classes. Finally, a balanced dataset of 25,192

Figure 5.4. Real-time gestures captured in the table-grape field with different people at various distances.

frames for these 13 classes was obtained after data augmentation. The methodology for training and evaluating the gesture recognition algorithm using hybrid data (combining real-world and synthetic datasets) is detailed in section 7.1.3, including its implementation and performance analysis.

5.2 Synthetic Data Creation

5.2.1 Gesture Synthetic Data

The Unity simulation system was utilised to capture the nuanced gestures enacted by a cohort of 30 virtual characters, employing multiple cameras within the Unity tool. This process involved meticulous attention to varying distances and angles, ensuring comprehensive coverage. Iterative data collection sessions were conducted to diversify the dataset, ranging from individual characters executing gestures to simultaneous performances by multiple characters within a single frame as shown in Figure 5.5. The methodological approach demonstrated significant potential for producing the large-scale datasets required in the development of advanced gesture recognition systems. Notably, the virtual data faithfully mirrored real-world scenarios, enhancing the applicability and relevance of our findings.

With a focus on realism and diversity, our efforts yielded a corpus of 26,000 frames of meticulously balanced virtual data representing a set of 13 distinct gesture classes. These data were derived from a spectrum of animations characterised by varying brightness, illumination changes, introduced noise, varying distances, varying camera angles, and several other configuration possibilities listed in Table 5.1. Such deliberate diversification ensured the dataset's robustness, scalability, and generalizability for training deep learning algorithms. Consequently, our research lays a good foundation for the advancement of gesture recognition technologies by ensuring that through simulated data, there will not be an extensive need for vast and varied real-world data collection, underscoring the importance of comprehensive and diverse data produced using simulation can facilitate the development of computational models in this domain.

Figure 5.5. Virtual Human Avatars performing Gestures in the Virtual Simulation.

Unity Item	Configurable parameters
Light source	Position, Orientation, Color, Intensity, No . of Sources
Camera	Position, Orientation, Optical Characteristics, No. of Cameras
Character	Position, Orientation, Skin, Color, Texture, Hair
Background	Plain Canopy, Field Canopy with grapes and leaves for harvesting, Field Canopy with branches and twigs for Pruning
Simulation	Rows and Columns of canopy, Leaf density, Bunch density, Time of the day, Light intensity, Ridges / No ridges for terrain, Number and type of Robots, No. of characters

Table 5.1. Unity Editor and Simulation configurable items

5.2.2 Additional Synthetic Data

In addition to the gesture data obtained from the simulated environment, the simulated robot(digital twin of the actual robot) was equipped with a suite of sensors. This indicates the capacity of the simulation environment to produce several sensory and visual data for navigation and perception tasks. The Inertial Measurement Unit (IMU) can navigate and interact with its surroundings with heightened awareness and notifies the robot with precise orientation, acceleration, and angular velocity information. By integrating IMU data into its control algorithms, the simulated robot can maintain stable locomotion, adapt to changes in terrain, and execute manoeuvres with certain agility. Similarly, in simulation, LiDAR sensors enable the robot to create point clouds and detailed maps, accurately detect obstacles, and plan collision-free paths, enhancing its navigational capabilities in complex virtual environments. These sensors match the calibration of the real world in the simulation, resulting in readily available data that can be used to test the robot's capabilities in the simulation so the algorithms can be translated without much effort to the real robot.

The robot is equipped with RGB-D cameras on either of the manipulator ends to acquire perception data of the grape bunches and branches for harvesting and pruning actions. All the RGB-D cameras used in both simulations and the real world are Intel RealSense D435i to maintain the same configuration. The data acquired from the head camera that captures both colour (RGB) and depth information further enriches the simulated robot's abilities to perceive the human behaviour and gestures performed in front of the robot. This data helps the robot to recognise objects, estimate their positions, and understand the spatial layout of its surroundings with enhanced depth perception. The RGB data coupled with point clouds helps estimate the bunch's position, and the segmentation of the bunches in simulation acts as ground truth for detection. Simulated odometry, based on wheel encoders or visual odometry techniques, provides the robot with self-motion estimates, aiding in localisation and mapping tasks within the virtual environment. Additionally, the ability of the robot to produce lights and sounds in simulation adds an interactive dimension, allowing it to convey information, alerts, and warnings and engage with virtual entities or human users. By integrating these sensor modalities and interactive features, the simulated robot gains a comprehensive understanding of its surroundings and configuration to change the simulated environment to a custom degree, letting us extract as much necessary data as possible and bridging the gap between the simulation and the real world.

5.3 Enhancing Synthetic Data with 3D Model Generation

In the realm of synthetic data generation, the integration of advanced AI models has revolutionised the creation of high-quality 3D assets from textual and visual inputs, significantly enhancing the efficiency of virtual simulator prototyping and data augmentation. A notable contribution in this field is Microsoft's *TRELLIS* model, which supports the generation of structured 3D latent representations from text and image inputs [263]. The system enables the conversion of data into multiple formats (e.g., *.fbx, .obj, .stl, and .ply*), including Radiance Fields, 3D Gaussians, and meshes, making it highly adaptable to different simulation environments.

5.3.1 Text/Image to 3D Model Implementation

The process begins with a *text-to-image generation* step, where *Large Language Models (LLMs)* and *text-to-image diffusion models* are used to create structured image representations based on textual prompts carefully describing the object features and the view-angles for maximum feature visibility. These generated images are aligned at multiple angles to provide a comprehensive visual dataset, facilitating accurate 3D reconstruction.

To ensure high-quality model generation, *background removal techniques* are applied to isolate the primary subject from the generated or real-world images. This preprocessing step eliminates extraneous elements, preventing interference during 3D reconstruction. The refined images are then passed into the *TRELLIS* model, which converts them into structured 3D latent representations [203]. This modified pipeline

enables the generation of realistic 3D assets from *both synthetic text-generated images and direct image inputs* from local storage. Samples of the reconstructed grape bunches and grape leaf are shown in Figure 5.6 and Figure 5.7.

Figure 5.6. 3D Grape bunch generated using the image-3D model

Figure 5.7. 3D Grape leaf generated using the image-3D model

5.3.2 Data Diversity and Noise Inclusion

To improve the robustness of AI models, it is imperative to introduce diverse 3D objects within synthetic training environments. Including multiple, non-repetitive objects enhances the model's generalisation capabilities by exposing it to a broader range of scenarios. Additionally, controlled noise injection into the datasets further strengthens model adaptability, particularly in applications requiring resilience to real-world environmental variability.

5.3.3 Challenges in Multi-Object 3D Reconstruction

Despite significant advancements, reconstructing multiple objects from a single image remains a complex challenge, particularly in cases of overlapping objects. When objects obscure each other, critical visual features become indistinguishable, affecting the accuracy of 3D reconstruction. Overcoming these limitations requires advanced segmentation techniques capable of disentangling occluded elements and preserving the structural integrity of each object. The scope of this research is beyond the aim

of this thesis, and the research will continue to focus on expanding the scope of 3D model generation to encompass the entire *scene reconstruction*. This involves *sequential semantic understanding* and *object-level segmentation*, ensuring that each non-repetitive element in a scene is accurately modelled. Such advancements will pave the way for the development of *fully reconstructed virtual environments*, providing enhanced realism for synthetic data generation and AI training.

5.4 Performance Evaluation

This section presents the system setup, the metrics employed and the results obtained during the evaluation.

System Setup

ROS Noetic was employed on an Ubuntu 20.04 LTS machine to operate the robot. Additionally, virtual character development and evaluation were performed on an Alienware x17 R2 equipped with 32 GB of RAM, a 12th-generation Intel i9 processor, and an NVIDIA RTX 3080 Ti GPU, running both Windows 11 Home and Ubuntu 20.04 LTS. The simulation was made for Linux as an executable with the possibility of changing the simulation environment with configurable settings. For the synthetic gestural data frames, the Unity hub (V 3.7) and editor applications (V 2021.3 - forward compatible) can run in either Linux or Windows systems. To enable any further changes to the characters or animations, a separate git repository was automated to produce a final executable with the changes incorporated as a new version of the simulator to be used for the experiments.

Evaluation Metrics

Evaluation metrics are essential to measure how performant the trained models are on the given data. Once human poses have been predicted, the predicted 2D joint locations are compared with the ground truth annotations in the dataset. Evaluation metrics are then calculated based on the difference between the predicted and ground truth joint locations. A model is considered to have high performance when it meets certain thresholds. These thresholds vary from metric to metric. A list of comprehensive metrics used in pose estimation was listed in [214]'s evaluation metrics section.

Several metrics were used to evaluate the performance of human pose estimation (HPE) models. *Precision*, *Recall*, and *Object Keypoint Similarity* (OKS) [170] are the most popular metrics for evaluating 2D multi-pose models [284, 170], while *Percentage of Correct Parts* (PCP), *Percentage of Correct Key points* (PCK), and *Percentage of Detected Joints* (PDJ) evaluate single-pose models [173, 72]. *Mean Per Joint Position Error* (MPJPE) is generally used for 3D pose estimation. However, the Unity editor calculates the *Z Coordinate* of the joint based on the *Euclidean distance* from the centre of the camera in the scene. For this reason, *MPJPE* based on 2D joint coordinates was calculated. In this case, the evaluation is based on the hypothesis that Unity editor-provided joint coordinates are considered ground truth. Mediapipe's Pose (Trained on BlazePose 33 3D landmarks) [29] and YOLOv8s-Pose

Gesture Actions	No Occlusions					With Partial Occlusions (min 2 joints)				
	PCK @0.2	***PDJ @0.2***	***PCK @0.5***	***PDJ @0.5***	***MPJPE***	***PCK @0.2***	***PDJ @0.2***	***PCK @0.5***	***PDJ @0.5***	***MPJPE***
T-Pose	86.6	86.6	100	100	9.71	85.1	85.1	95	96	11.3
Attention	86.6	93.3	100	100	7.55	83.3	89.2	97	97	9.12
Move_Back	80	86.6	100	100	8.41	76	82.1	96	96	10.28
Move_Front	80	80	100	100	7.04	74	71	95	96	12.4
Move_Left	86.6	86.6	100	100	7.21	82	84	96	96	8.67
Move_Right	86.6	86.6	100	100	7.26	82	84	96	96	8.92
Pause	93.3	100	100	99	7.78	91.2	100	92	92	8.23
Resume	53	53	100	98	13.71	51	51	92	92	14.53
Standing	86.6	86.6	100	100	7.61	85.1	85.1	100	100	8.1
Start	86.6	86.6	100	100	9.64	84	84	98	98	10.64
Stop	78.3	79.2	95	96	11.54	73.2	73.4	91	91	13.89
Terminate	63	66.6	94	94	12.27	60	63.3	88	89	13.72

Table 5.2. Evaluation of 2D key points on the joint coordinates of the virtual characters at threshold 0.2. Unity-generated joint coordinates (Ground Truth) vs MediaPipe-generated joint coordinates (Predictions)

(Trained on OpenPose 17 2D landmarks) [120],[45] predictions on the virtual human's joints with the simulation background were evaluated against the synthetic data to validate if the models trained on real data can detect the joint coordinates properly. As the evaluation is based on the joint coordinates alone, *PCP*, *PCK*, *MPJPE*, and *OKS* metrics were chosen to evaluate the effectiveness of detecting joint coordinates of virtual characters with and without occlusions by pose estimation algorithms. *PCK, PDJ* works better for the higher values with a range of 0-100%. *MPJPE* gives better results with lower values, and *OKS = 1* means a perfect prediction matching all point switch ground truth. All these metrics were evaluated at *threshold = 0.2 and 0.5*, where the threshold was the Euclidean distance between the left and right hip joints.

Results

Several frames of the virtual human were collected with 1280x960 resolution at four distance settings from the camera. These distances were maintained constant throughout the evaluation of 30 characters. Different gesture actions were considered for the evaluation. Though initially 21 gesture actions were defined, we evaluated the static poses resulting in 12 gesture actions and character's T-Pose discarding the 13th UNKNOWN class. These gestures were performed without any occlusions and with partial occlusions to the joints by objects or other virtual characters in the simulation. The following results in Table 5.2 and Table 5.3 suggest that the pose estimation algorithms can accurately detect the coordinates in virtual characters and vice-versa. It was observed that a greater distance from the camera, combined with multiple occlusions, led to diminished accuracy in joint coordinate detection. The OKS results were always in the range of 0.01 to 0.09. These results support the claim that synthetic data can indeed be a valuable tool for gesture recognition through pose estimation and can reduce the effort of creating and labelling data.

Gesture Actions	No Occlusions					With Partial Occlusions (min 2 joints)				
	PCK @0.2	***PDJ @0.2***	***PCK @0.5***	***PDJ @0.5***	***MPJPE***	***PCK @0.2***	***PDJ @0.2***	***PCK @0.5***	***PDJ @0.5***	***MPJPE***
T-Pose	86.6	86.6	100	100	10.03	82.1	82.1	95	96	11.94
Attention	86.6	90.6	100	100	9.05	84.4	87.4	93	93	10.12
Move_Back	86.6	86.6	100	100	8.93	80.1	80.1	93	93	10.79
Move_Front	73.3	80	100	100	8.99	70	73.3	93	96	10.27
Move_Left	73.3	73.3	100	100	8.82	70	70	93	93	10.68
Move_Right	73.3	73.3	100	100	8.87	70	70	93	93	10.58
Pause	86.6	86.6	100	100	9.51	83.3	83.3	91.1	91.1	11.23
Resume	56.6	56.6	93	93	12.04	51.1	51.1	89	89	13.93
Standing	80	86.6	100	100	9.49	82.3	82.3	98	98	12.78
Start	80	80	100	100	7.28	78	78.3	98	98	9.64
Stop	73.3	80	93	96	12.54	71.2	70.3	89	89	14.51
Terminate	66.6	80	100	100	12.76	63.3	63.3	88.3	89	14.79

Table 5.3. Evaluation of 2D key points on the joint coordinates of the virtual characters at threshold 0.2. Unity-generated joint coordinates (Ground Truth) vs YOLOv8s-Pose generated joint coordinates (Predictions)

5.5 Discussion

A synthetic dataset generation strategy based on the employment of a VR environment was introduced. This strategy has been applied to the challenging scenario of precision agriculture and of the CANOPIES project. The quality of the generated dataset has been evaluated by checking the quality of pose estimation, which is a fundamental aspect of many tasks in human-robot collaboration, including gesture recognition.

Other existing datasets, such as SURREAL [250] and Human 3.6M [116] datasets, offer high visual realism and annotation precision. Yet, this VR-based synthetic data generation stands out in terms of cost-effectiveness, ease of customisation, and the ability to create targeted scenarios for specific evaluations. This flexibility is beneficial for testing the robustness of pose estimation algorithms under controlled conditions like occlusions and varying distances, making it a valuable tool for benchmarking pose estimation and gesture recognition research.

The proposed approach represents a solution that can be generalised to any environment where acquiring real datasets can be challenging due to the nature of the environment and the effort needed. Dataset acquisition is indeed a challenging task in many learning tasks where knowledge is not general (as in the case of LLMs), being instead specific to the application scenario.

The proposed approach indeed has its own limitations. Developing a virtual environment that accurately mimics the real one can be challenging. Consequently, algorithm evaluations should not rely solely on synthetic data but should also incorporate real data.

Similarly, while this chapter advocates the use of synthetic data generation methods for training, incorporating a subset of real-world data with synthetic data helped mitigate the sim-to-real gap and improve model robustness and generalisation. This "hybrid data" strategy, which combines the strengths of synthetic and real data to enhance model performance, is discussed in the next chapter.

Chapter 6

Spoken Human-Robot Interaction

Verbal communication plays a pivotal role in information exchange, whether between humans or between humans and robots. Owing to recent advances in Natural Language Processing (NLP), speech has emerged as a vital medium for interacting with robotic systems. Indeed, numerous platforms employ voice as the principal input and output modality. This chapter offers a comprehensive exploration of the factors that prompted the design of our speech recognition pipeline, tailored to enhance interaction in outdoor collaborative settings. It presents preliminary investigations and assessments of existing Speech-To-Text and Natural Language Understanding technologies. Furthermore, a user study examining various Text-To-Speech tools and libraries is introduced, shedding light on participants' impressions and offering an in-depth review of their suitability in table-grape vineyard environments. The chapter then transitions to the architecture of a speech act classification system, discussing both the obstacles encountered and potential strategies for refining the robot's interpretation of human vocal expressions. Finally, it provides insights into the developed speech-driven pipeline, encompassing preliminary experiments and application scenarios.

6.1 Vocal Utterance Dataset

The increased demand for developing agricultural systems and applications inspired many researchers to collect and distribute diverse datasets within the research community. However, regarding the vineyard scenario, most of the released data consists of crop and/or fruit images required to train computer vision algorithms and perform perception tasks, such as fruit detection, identification and segmentation, disease recognition, and colour identification. To the best of our knowledge, there is no spoken or textual dataset available for conducting studies and analyses on HRI in similar scenarios. The scarcity of data, also present in other applications for outdoor collaborative environments that require continuous interaction between humans and robots, led us to gather and share the transcription of spoken utterances within the research community. Users interested in the experiment were asked to sign a consent document to record their voices. Considering the speech acts discussed in

the previous section as a basis, the data was collected separately with three different scenarios, one for each category under investigation:

- Information (22 questions)
- Command (20 questions)
- Request (20 questions)

The customised questionnaires were created through the Jotform[1] platform to reach, with our forms, as many participants as possible. Jotform was chosen for its practicality in utterance recording, facilitating speech acquisition from the user perspective and, consequently, speeding up the entire data collection process. In all three forms, we described the general context and some technical terms that users could exploit in generating the vocal responses. Additionally, to avoid uncertainty or misunderstanding in the description of the scenario, a representative image followed each question. Such pictures (some captured from the simulation environment, others in the real field, and a few taken from the web) illustrate tasks, activities, actions, and vineyard elements such as grapes, leaves, branches, and the pergola system.

Participants

Given the difficulties in recruiting a sufficient number of vineyard operators for the data collection, we relied upon Bachelor's, Master's, and PhD students in Engineering in Computer Science at our university. Hence, at the end of the acquisition process, we reached around 40 participants with each form (most of the users provided vocal utterances for all three forms). However, we noticed that people who participated in the data acquisition process were mainly males between 18 and 30 years old. Regarding the level of experience with robots and expertise in the vineyard, we detected quite balanced robotic skills among the participants, with half of them having good proficiency or interacting several times with a robot. Nevertheless, looking at the expertise in the vineyards, we noticed that most people had little knowledge about the activities conducted in such an environment. However, such results are reasonable since we mostly spread the forms with Bachelor, Master, and PhD students in Engineering in Computer Science at our university. From this point of view, unfortunately, recruiting a sufficient number of vineyard operators was impossible. In addition, we imagine such kind of system to be more of interest to the next generation of digital farmers, of which young students may represent a reasonable proxy.

Utterances

The data acquisition process of the Information, Command, and Request speech act categories lasted around two months. At the end of this period, we collected around 1,800 vocal utterances with their corresponding textual transcriptions. All the details concerning the data analysis are provided in Table 6.1, such as the number

[1] https://eu.jotform.com/

of sentences before and after removing duplicated utterances and the final number of statements obtained after rearranging them according to the category of the content information they represent.

From Table 6.2, it is clearly noticeable that the categories involved in the sentence adjustments are *Command* and *Request*. Such an outcome is reasonable, as the difference between the two aforementioned classes could be very subtle from the user's perspective. We listed under the 'Command' speech act, all the sentences provided by the human that the robot must execute in the short term, such as "Turn left", "Harvest the ripe grape bunches", or "Remove all the dry branches". At the same time, utterances asked by the person that do not require an immediate interruption of the robot's activity have been included in the 'Request' class. For instance, sentences belonging to this category are "Can you tell me your battery level?", "Could you help me in harvesting these grapes?", "How many damaged grapes have you identified?".

	Sentences		
Speech Act	with duplicates	without duplicates	after category adjustments
Information	900	804	804
Command	728	517	665
Request	719	534	342

Table 6.1. Analysis of the acquired data

	Request sentences		
		False	True
Command sentences	False	-	11
	True	200	-

Table 6.2. Utterances belonging to the incorrect category.

6.2 Speech Pipeline Tools: Evaluation and Selection

This section introduces analyses and investigations of various modules related to speech, specifically focusing on the challenges in the robot's acquisition of human utterances (Speech-To-Text) and the recognition of content information (Natural Language Understanding). Moreover, a user study was conducted to gather human perceptions of how the robot conveys information through various speech synthesis libraries and tools (Text-To-Speech). Such an aspect is fundamental for enhancing trust and establishing a stronger relationship between humans and robots.

6.2.1 Speech-To-Text

In developing the Natural Language Processing (NLP) pipeline for the CANOPIES project, a preliminary survey of existing speech recognition libraries and systems was undertaken. Given the environmental constraints of table-grape vineyards, identifying the most appropriate Speech-To-Text solution was crucial for ensuring

reliable and robust communication between humans and robots. Consequently, both online and offline systems were evaluated using English and Italian utterances, including DeepSpeech [107] (offline), Vosk [51] (offline), Sphinx (offline), and the Google Web Speech API (online) [277]. This review considered transcription accuracy, internet connectivity requirements, and model availability across multiple languages.

DeepSpeech [107], an open-source Speech-To-Text library developed by Mozilla, uses state-of-the-art machine learning techniques and offers training options over custom datasets to achieve better results. Nonetheless, it was set aside due to uncertainties regarding its ongoing support and its recognition performance being less impressive than anticipated. In Figure 6.1, the correctly transcribed utterances are illustrated. The vocal information given in input does not include subordinates. However, it is noticeable that even with short and straightforward statements, the library provided the right transcriptions in 16 out of 30 cases (approximately 0.53%).

Vocal Utterance	Recognized Sentence	Corresponding?
Yes	Recognized: yes	✓
No	Recognized: no	✓
Yes	Recognized: yes	✓
Help me	Recognized: help me	✓
Can you help me in this?	Recognized: can you help me in this	✓
Right	Recognized: right	✓
Left	Recognized: last	x
Left	Recognized: left	✓
What time is it?	Recognized: they	x
What time is it?	Recognized: time is it	✓
What's the weather for today?	Recognized: what's the weather for to day	✓
Remove these branches	Recognized: remove these branches	✓
Remove these bunches	Recognized: remove these banks	x
Remove these bunches	Recognized: remove these bunches	✓
Can you show me pizzutello grape?	Recognized: you saw me in withered	x
Can you show me the grape?	Recognized: can you show me the great	x
Can you show me the grape?	Recognized: can you show me these great	x
Goodbye	Recognized: good by	x
Goodbye	Recognized: by	x
See you later	Recognized: later	x
See you later	Recognized: you later	x
See you later	Recognized: see you later	✓
Go forward	Recognized: go forward	✓
Go back	Recognized: go back	✓
Go right	Recognized: go right	✓
Go left	Recognized: we left	x
Let's start working	Recognized: let's start working	✓
Put the grape bunches in the box	Recognized: the great bear in the box	x
Put this grape in the box	Recognized: to this great in the box	x
-	Recognized:	-
Grape	Recognized: great	x

Figure 6.1. DeepSpeech recognition results. The green ✓ represents an utterance correctly understood, while the red X identifies a wrongly transcribed sentence.

Furthermore, most spoken sentences are repeated twice (sometimes even 3 times) before the system recognises the utterances properly. Subsequent tests focused on Sphinx, an offline Python-based speech recogniser, and the Google Web Speech API. While Sphinx permits relatively swift adaptation to new languages without an internet connection, its recognition accuracy did not meet expectations. By contrast, the Google Web Speech API delivered high transcription accuracy with multiple language options but relied on constant connectivity [277].

Vosk[2] was finally examined as an offline open-source toolkit with support for numerous languages and dialects. It provides various model sizes according to the target platform's requirements and desired accuracy. One of the advantages of Vosk is its scalability and precision in providing utterance textual representation, as demonstrated in Figure 6.2. In the presented examples, all the vocal sentences are different from each other, with 8 out of 11 adequately written by the system. It is noticeable that in the remaining 2 cases, one word is misunderstood, and in the last statement, which is the longest among all, only two words are wrongly interpreted by the model. Vosk demonstrated solid results even with smaller models, which proved sufficient for voice-based tasks in diverse scenarios. From these assessments, Vosk and the Google Web Speech API emerged as the two most promising candidates for use in table-grape vineyards. However, the lack of a dependable internet connection in the field led to the selection of Vosk as the Speech-To-Text module for the CANOPIES project.

Vocal Utterance	Recognized Sentence	Corresponding?
-	{ "text" : "" }	✓
Robot prendi l'uva	{ "text" : "rabat prendi l'uva" }	X
Robot	{ "text" : "robot" }	✓
Robot prendi l'uva e mettila nella scatola	{ "text" : "robot prendi l'uva e metti la nella scatola" }	X
Potresti darmi una mano	{ "text" : "potresti darmi una mano" }	✓
Taglia questo ramo	{ "text" : "taglia questo ramo" }	✓
Vai alla tua destra	{ "text" : "vai alla tua destra" }	✓
Spostati alla tua sinistra	{ "text" : "spostati alla tua sinistra" }	✓
Seguimi	{ "text" : "seguimi" }	✓
Vieni con me	{ "text" : "vieni con me" }	✓
Che tempo fa oggi	{ "text" : "che campo fa oggi" }	X
Quanti gradi fanno oggi	{ "text" : "quanti gradi fanno oggi" }	✓

Figure 6.2. Vosk recognition results. The green ✓ represents an utterance correctly understood, while the red **X** identifies a wrongly transcribed sentence.

6.2.2 Natural Language Understanding

Hermit-NLU, a hierarchical multi-task system with a multi-layer representation of dialogue acts, frames and frame elements presented in [4], is a baseline for developing a robust Natural Language Understanding (NLU) module. Since the network is trained on the NLU-Benchmark dataset and structured with scenario, action, and entity information, their corresponding association with speech acts, frames, and frame elements is required. Unfortunately, such a dataset is unsuitable for our purposes as it includes multiple home assistant task domains and conversational

[2] `https://github.com/alphacep/vosk-api`

exchanges among individuals. Therefore, the dataset presented in Section 4.2.1 was explicitly collected for the table-grape vineyard collaborative environment and used to analyse frame semantic identification and argument extraction. At the time of conducting such an investigation on multilingual frame semantic parsing, there was no stable and reliable system capable of understanding specific verbs or actions conveyed in a text written in a language different from English. Since the data acquired for our application scenario was in Italian, the initial solution was to translate the acquired utterances into English. Hence, the preliminary NLU speech pipeline composed of 5 modules, as in Figure 6.3, is described.

- **Acquisition**
  Evaluations are performed on the vocal utterances collected with Jotform questionnaires. People interacted with the system using diverse equipment during the data acquisition process. Most participants used mobile phones, computers, and tablets, in combination with external devices such as microphones, headsets and headphones, to reduce environmental noise while recording the spoken sentences.

- **Speech-To-Text**
  The choice of the Speech-To-Text recogniser was derived from the initial analysis conducted on existing online and offline state-of-the-art systems and libraries. Since Vosk has demonstrated to provide the most accurate utterance transcriptions, with a limited percentage of uncertainty and misrecognised words, it was employed in this module.

- **Punctuation Restoration**
  The punctuation restoration applied to the transcribed text is one fundamental step introduced in the proposed NLU pipeline that is necessary for proper sentence identification. The addition of this intermediate module is required in order to obtain a more accurate sentence translation. Indeed, punctuation restoration generally helps to separate utterances with diversified content information, clarify a concept and, in some cases, identify a particular sentence belonging to the category. For instance, there is a higher probability of associating a statement to the "Request" rather than to the "Command" or "Information" class if it ends with a question mark (?). Therefore, the model in [99] is experimented. However, its limitation in the supported punctuation (only 5 types) motivated us to look for a more accurate system. Hence, the Bert restore punctuation model by Hugging Face[3], is chosen for this purpose, being able to manage more English language punctuation than the previously tested network, such as: exclamation marks (!), question marks (?), dots (.), commas (,), hyphen (-), colon (:), semicolon (;), apostrophe (').

- **Translation**
  The lack of an open-source, robust, multilingual system for frame semantic parsing inspired us to consider the idea of sentence translation. Therefore, a translation module is included in the presented NLU preliminary pipeline.

[3] `https://huggingface.co/felflare/bert-restore-punctuation`

However, to identify the most suitable translation system, two different Transformer models from Hugging Face are evaluated: the multilingual (mBart) for multilingual machine translation in [269], which manages sentence transcription in any of the 50 covered languages, and the Helsinki-NLP/opus-mt [241] model available for Italian-to-English translation. One of our goals is to design a user-friendly system that ensures collaboration and enhances communication between humans and robots, so multilinguality has a key role. For these reasons, Helsinki-NLP/opus-mt models are tested and employed to support sentence translations from any language to English.

- **Frame Semantic Parsing**
  The frame semantic parser represents the last stage of the proposed NLU pipeline. The network works only with English sentences, being trained on FrameNet. Since it is not feasible to pass the original utterance (in any other language differing from English) directly to this system, it is necessary to perform sentence translation before accessing this module. The outcome of the Frame Semantic Transformer, described in [52], is the identification of the trigger, frame, and frame arguments associated with a specific sentence provided in the input. The network, developed on top of the T5 Transformer model [199], can accurately identify frame and frame arguments in a certain utterance.

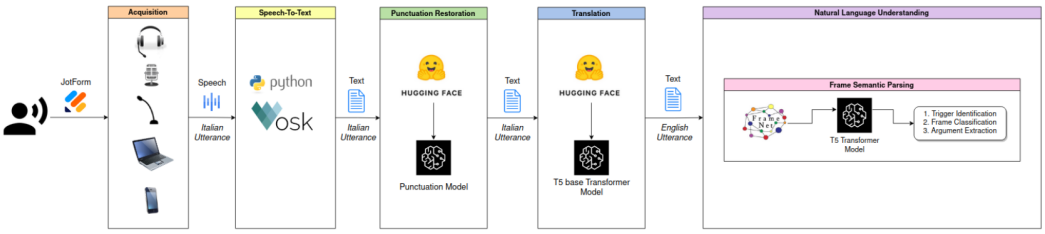


Figure 6.3. NLU speech pipeline for frame semantic parsing.

The outcome is obtained by passing the collected Italian sentences through all the modules of the developed pipeline. Hence, such a system can extract information about frame semantics and associated frame arguments from most of the analysed utterances. Additionally, verb identification through Part-Of-Speech (POS) tagging libraries [186] and their association with some predefined frames is investigated. Verb recognition is fundamental to proceed in understanding the roles in a sentence through the employment of Semantic Role Labeling (SRL) systems, such as the SRL model using contextualised word embeddings, Graph Convolutional Network (GCN), Biafne Attention mechanism (BiLSTM) and syntactic features [22], Unify-SRL [59] or InVeRo-XL [60]. All these applications leverage Bert or XLM-RoBERTa language models to perform multilingual SRL.

6.2.3 Text-To-Speech

Text-to-Speech (TTS) technology has undergone significant advancements, transforming how machines communicate with users across various domains, including adaptive technologies, virtual assistants, and robotics. TTS plays a fundamental role in facilitating natural speech synthesis from written text. In CANOPIES, where machines collaborate with people in dynamic table-grape vineyard environments, selecting the most appropriate TTS library is crucial. Challenges such as environmental noise, weak internet connections, and the need for efficient HRC define the significance of this choice. Addressing and mitigating environmental noise in CANOPIES requires reliable Robot-To-Human communication, complemented by headsets to enhance user comprehension. Moreover, handling weak internet connections in this outdoor domain necessitates a TTS library capable of robust performance under limited bandwidth situations. Furthermore, ensuring efficient HRC entails carefully considering the robot's answer time, emphasising the importance of selecting a library that minimises latency and delivers human-like fast responses. In order to determine the most suitable TTS application for CANOPIES and similar scenarios where HRC is essential, a comprehensive analysis of common TTS libraries and tools, both online and offline, is conducted. The investigation involved technical assessments and user studies to identify the optimal solution to employ in environments characterised by dynamic challenges and where seamless communication between humans and robots is paramount.

Library Selection and Criteria Definition

TTS library selection focuses on two primary features: the number of supported languages and the internet connection requirements. Given the objective of developing a multilingual system, the support of diverse languages is a crucial aspect. Moreover, addressing the challenge of an unstable internet connection in our application scenario required an online and offline analysis of common TTS libraries and frameworks. Five tools are identified to conduct the investigation, with the first one working online and the others offline:

1. gTTS with Google Translate API (online)
2. PyTTSx with eSpeak (offline)
3. Mbrola (offline)
4. SVOX PicoTTS (offline)
5. Bark with pre-trained AI models (offline)

A set of criteria is defined to evaluate the selected TTS applications and identify the most suitable one for the table-grape vineyard scenario. This evaluation involves categorising criteria into two groups: technical and human. The technical class encompasses parameters essential for the system's technical analysis, including processing time, Mean Opinion Score (MOS), RAM, and CPU usage. In contrast, human criteria, as outlined in [11], are necessary to comprehend how users perceive

the robot's voice, focusing on factors like intelligibility, expressiveness, artificiality, and suitability. An explanation of each criterion is provided below:

- **Technical criteria:**
  - **Execution Time:** Time required by the system to generate a vocal utterance from a textual sentence.
  - **Performance:** RAM and CPU usage percentage.
  - **Mean Opinion Score (MOS):** Measure to evaluate the overall subjective quality of audio.
- **Human criteria:**
  - **Intelligibility:** How easily and accurately listeners can understand the generated speech
  - **Expressiveness:** The ability of the system to convey emotions or nuances in speech.
  - **Artificiality:** How natural or human-like the synthesised speech sounds.
  - **Suitability:** The generated speech fits the intended context or purpose.

The analysis considers four utterances varying in the informative content and sentence length. These are evaluated in both Italian and English, which represent the two main languages that have to be supported by our system. Three brief statements cover the "Information", "Command", and "Request" categories, while the longer subordinate encompasses multiple classes.

Technical Analysis

As part of the technical assessment, the time taken by each TTS system to generate spoken output from a textual sentence is compared. This measurement is conducted using Python's time module [4]. The box plots in Figure 6.4 clearly illustrate the variation in average processing times among the evaluated systems. Mbrola and PicoTTS prove to be the fastest, requiring 0.04 and 0.05 seconds, respectively, whereas Bark is the slowest at approximately 35 seconds. Table 6.3 summarises the results for the selected TTS modules, ranked according to their average computation times (in seconds).

Python cProfile [5] was employed to examine RAM and CPU usage, enabling the identification of the TTS library with the least computational overhead. As illustrated in Figure 6.5, each application—except Bark—demonstrates nearly negligible RAM consumption (close to zero) and maintains CPU utilisation below 10% for the short durations during which it is active.

[4] `https://docs.python.org/3/library/time.html`
[5] `https://docs.python.org/3/library/profile.html`

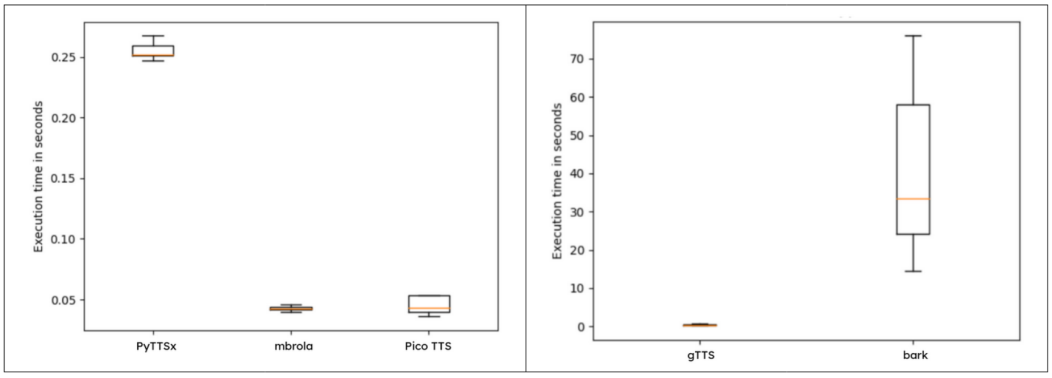


Figure 6.4. TTS libraries execution time lower than 0.3 seconds (PyTTSx, Mbrola and PicoTTS) are provided on the left, while the ones requiring more than 0.3 seconds are presented on the right (gTTS and Bark).

Rank	Library	Language	Language Average	Total Average
1	Mbrola	Italian	0.0425	0.0423
		English	0.0422	
2	Pico TTS	Italian	0.0507	0.05
		English	0.0494	
3	PyTTSx	Italian	0.2642	0.259
		English	0.2538	
4	gTTS	Italian	0.3419	0.3427
		English	0.3435	
5	Bark	Italian	38.1321	34.9835
		English	31.8349	

Table 6.3. TTS libraries ranked by average computation time (in seconds) on both Italian and English statements.

The pre-trained model in [154] was employed to compute the Mean Opinion Score (MOS) following the approach outlined in [260]. In this procedure, eight test sentences (four in English and four in Italian) are generated by each of the selected TTS libraries, and the resulting WAV files are then input to MOSNet. This technique automatically evaluates newly produced vocal outputs without human feedback. This offers valuable insights for researchers and engineers seeking to determine which tool or library generates the most natural and high-quality speech. The distribution of MOS values, depicted in Figure 6.6, peaks between 3.5 and 4. Typically, MOS scores range from 1 to 5, with 1 signifying the lowest perceived quality and 5 the highest. Higher MOS ratings are generally linked with natural or human-like utterances, whereas lower values indicate a high degree of artificial-sounding outputs.

User Study

A Jotform questionnaire was administered to gather insights into how individuals perceive various synthesised robot voices. A total of 36 participants (24 men and 12 women), predominantly young adults from Sapienza University, took part by

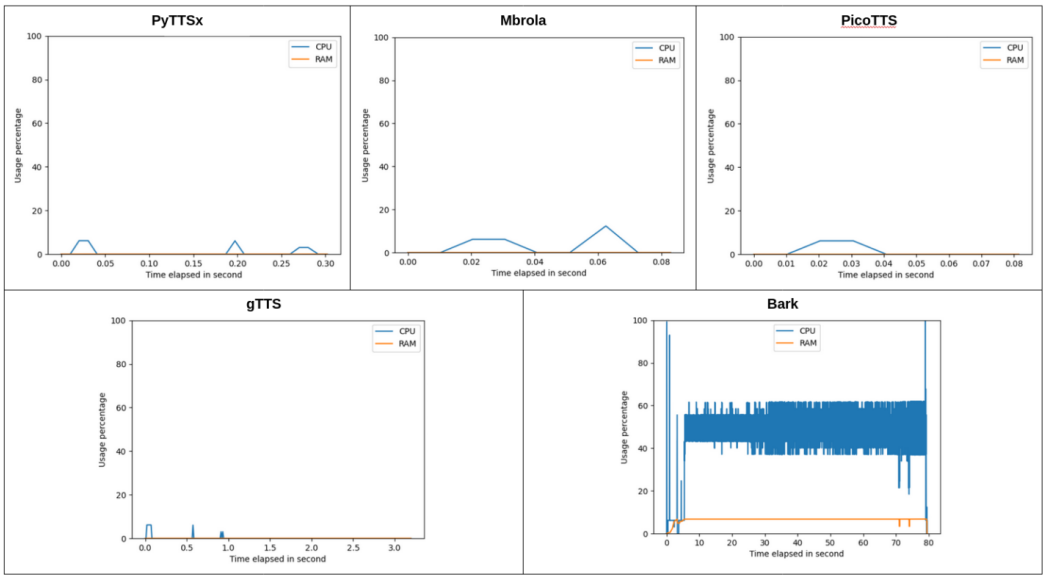


Figure 6.5. Graphical representation of CPU and RAM consumption over time for PyTTSx, Mbrola,PicoTTS (on top) and gTTS, Bark (at the bottom).

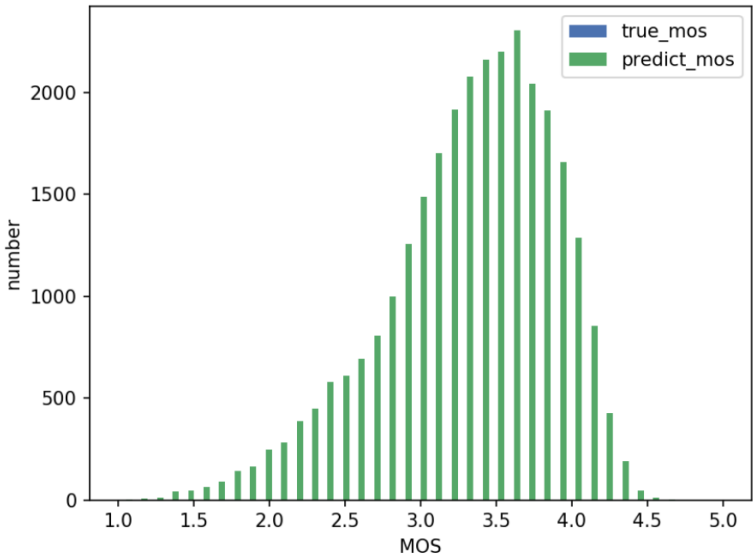


Figure 6.6. MOS distribution associated with vocal utterances generated through PyTTSx, Mbrola, PicoTTS, gTTS and Bark.

evaluating both Italian and English vocal samples generated by the chosen TTS libraries. The study examined several criteria, adapted from [11] and tailored to the research context:

- **Intelligibility:** "Can you clearly understand what the robot says?"
- **Expressiveness:** "How is this robot's voice perceived, from monotonous to expressive?"

- **Artificiality:** "Does this voice sound like a robotic voice?"
- **Suitability:** "Is this voice appropriate for any of the robots presented below?"

A 5-point Likert scale was used for the first three criteria. For intelligibility and artificiality, 1 indicated "Not at all", and 5 indicated "Yes, absolutely", while expressiveness ranged from 1 ("Very monotonous") to 5 ("Very expressive"). To assess suitability, images of farming and logistics robots used in the CANOPIES project [205] were presented, and participants were asked to assign each synthesised recording to one, both, or neither of the robots based on their appearance. Figure 6.7 illustrates each participant's evaluations for the Italian and English samples, yielding 72 responses per TTS system. Regarding intelligibility, the number of "Agree" and "Strongly Agree" responses indicated that gTTS (68) and Bark (63) most effectively conveyed the intended message, followed by PicoTTS (41). A similar trend was observed in expressiveness, with gTTS (42) and Bark (34) emerging as the most expressive tools. In contrast, Mbrola (65) and PyTTSx (64) ranked highest in terms of artificiality, suggesting that these libraries produced the least human-like sounds. For suitability, Bark (55) and gTTS (54) were deemed most appropriate for the robots, although PicoTTS also received favourable feedback for use in the project scenario.

Figure 6.7. A graphical representation of users' perception of intelligibility, expressiveness, artificiality and suitability across the selected libraries.

PicoTTS demonstrated the most significant potential for table-grape vineyard applications based on technical outcomes and participants' impressions. Its resource efficiency, reflected in low CPU/RAM consumption and swift text-to-speech conversion, proves especially valuable in the CANOPIES setting, where internet

connectivity is unreliable. Although gTTS excelled in several areas, it was excluded due to its dependence on a stable internet connection. Despite Bark's popularity, its high computational requirements in terms of CPU/RAM usage and lengthy processing times rendered it unsuitable for this particular application scenario.

6.3 Speech Act Classification

This section examines the system's implementation and reports on the performance of the speech act classification models designed to facilitate the robot's comprehension of human utterances. The training dataset comprises 1,800 Italian sentences collected through Jotform, as outlined in Section 4.2.1, while the evaluation relies on 3,364 utterances obtained from user studies conducted in NIVE and IVE, detailed in Section 4.3.1. The results indicate potential misclassification issues that may arise, including misunderstandings of individual words, extended pauses during speech (prompting the system to segment a single sentence into multiple parts), and an absence of intonation analysis. These challenges must be carefully addressed to develop an effective speech-based collaborative pipeline.

6.3.1 System Implementation

The necessity of a modular and robust system that could exploit the speech act classification to simplify the robot's comprehension of the specific information content led us to develop a preliminary ROS framework for spoken interaction between humans and robots. Several existing works that identify speech as the primary communication modality, based on old ROS versions, were used as baselines in developing our system [90, 78, 138]. In particular, the preliminary framework was implemented on ROS Noetic in Ubuntu. The proposed pipeline used for both immersive and non-immersive experiments is presented in Figure 6.8.

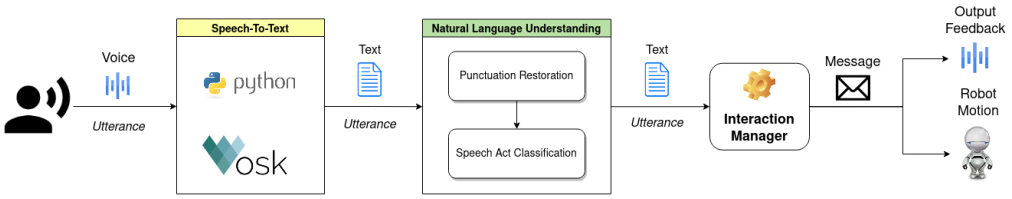


Figure 6.8. Developed ROS speech pipeline.

The first module is the *Speech-to-Text*, which is responsible for acquiring human spoken utterances and converting them into their textual representation through Vosk [12], an offline open-source speech recognition toolkit. Vosk was chosen for its higher accuracy in utterance transcriptions, also with the smallest models, and for supporting different languages. Since our final goal is to develop a robust and reliable system with whom people are expected to interact spontaneously, there is a higher probability that allowing communication in different languages (for foreign human workers) could improve communication with the robot. Subsequently, the textual transcription of the utterance is processed by the *Natural Language Understanding*

module. Such a component introduces punctuation in the sentences using the Bert restore punctuation model by Hugging Face to simplify the process of speech act classification. For instance, a question mark at the end of a statement has a higher probability of associating the sentence with the 'Request' class rather than the 'Command' or 'Information' categories. The speech act classification model was trained on the dataset acquired through Jotform by adopting the AutoGluon library [81] for Natural Language Processing (NLP) tasks. Such an automated machine learning library was chosen since it could select the best training parameters by optimising them for the final goal (in our case, a classification problem). However, another relevant advantage of this library is associated with the existence of a specific predictor for supporting multimodality, which could help us improve our final system.

Finally, based on the sentence prediction class obtained through AutoGluon, the *Interaction Manager* performs some checks to verify if any of the words of the sentence appears in the system's knowledge and generates a message that is sent to the speakers, so to provide vocal feedback of the robot's speech act understanding ("Information", "Command", "Request" or a combination of different categories in case of subordinate sentences) by exploiting PicoTTS[6] for the *Text-To-Speech* module. Therefore, when the android recognises an utterance belonging to the 'Command' class, a motion message is sent to the virtual robot's motors, attaining the corresponding motion in the simulated environment (both immersive and non-immersive). Nevertheless, even if a different category is identified, the robot provides feedback to the user by moving its head and looking around to give a feeling of comprehension.

6.3.2 Classification Results

The evaluation results on the robot's speech act understanding of the utterances collected in immersive and non-immersive experiences are presented and discussed here. By the end of the NIVE and IVE experiments, we acquired 5,180 vocal recordings and their corresponding textual transcription. However, after removing duplicated sentences (mostly belonging to the Command category), the number of statements we achieved for the final evaluation was 3,364. As anticipated in Section 6.3.1, the speech act classification network could assign one of the following classes to the input utterance: *'Information'*, *'Command'*, *'Request'* or a combination of these categories in case of a complex subordinate. The outcome of our evaluation is represented in Figure 6.9 and available at this link.

From this analysis, it has been noticed that the sentence misclassification depends on one (or multiple) issues:

1. Misunderstanding of one or more words in the sentence due to ambient noise, words not pronounced clearly or similarities with other terms

2. Prolonged pause between words within the same sentence, causing the system to separate it into different statements, to each of which one (or more) categories will be assigned

[6] `https://github.com/gooofy/py-picotts`

3. Lack of intonation analysis

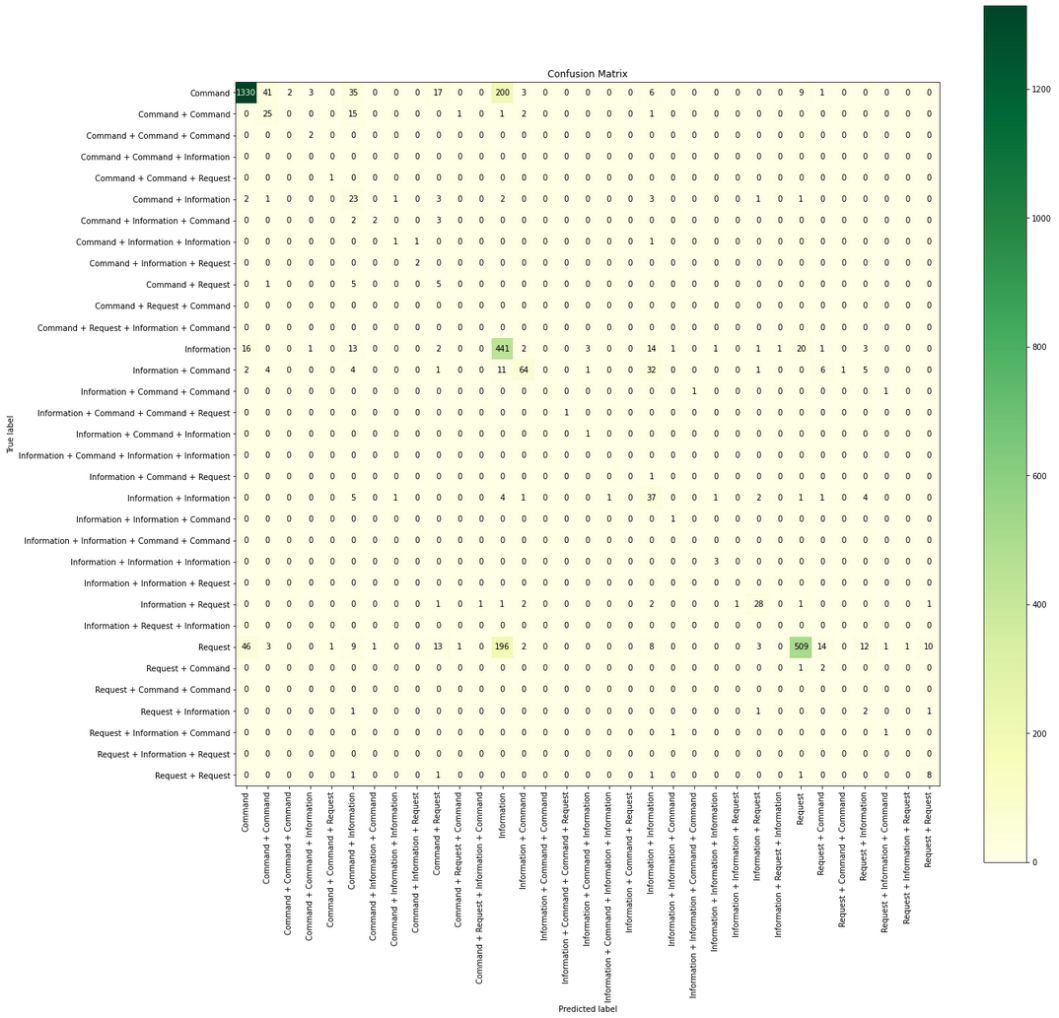


Figure 6.9. Confusion matrix of all the 3364 sentences in the test set.

Since our goal is to primarily evaluate the developed speech act classification module in an ideal situation, in which comprehension failures are not detected and propagated among the pipeline by the speech recognition module, all the acquired sentences were double-checked and the ones belonging to the first (346 statements) and second issues (63 statements) listed above were fixed, by listening to the recordings collected during the IVE and NIVE user studies. Hence, we ran the speech act classification module on the corrected sentences to identify the issues presented previously from these classification outcomes.

In both cases, the performances of the category classification network increased when working with precise transcribed sentences. In particular, from the 63 statements previously misunderstood, in the recent analysis: 46 were correctly classified,

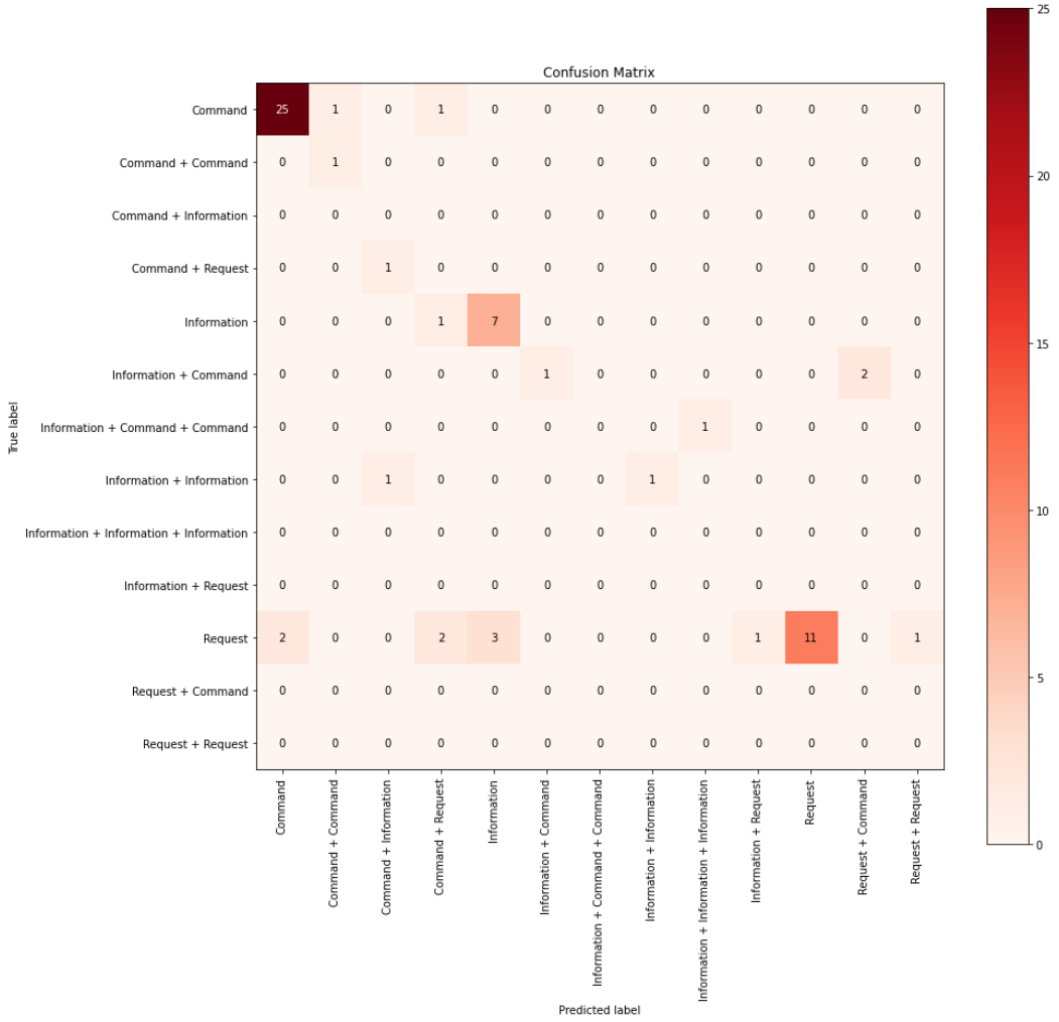


Figure 6.10. Confusion matrix of the corrected sentences belonging to 63 long pauses.

Metric	Set 1	Set 2	Set 3
Sentences	3,364	346	63
Revised	✗	✓	✓
Correctly classified	2,746	186	46
Misrecognised	346	-	-
Prolonged pauses	63	-	-
Intonation necessity	209	84	10

Table 6.4. Analysis of the acquired data

10 were wrongly recognised due to intonation issues, while an incorrect category was assigned to the remaining 7, as shown in Figure 6.10. Similar results were also encountered with the 346 statements that previously contained one or more misinterpreted words. Indeed, after sentence correction, 186 were associated with the suitable class, 84 fell under the intonation issue, and 76 were not located adequately by the speech act classification system, as presented in Figure 6.11. The illustrated

results are summarised in Table 6.4.

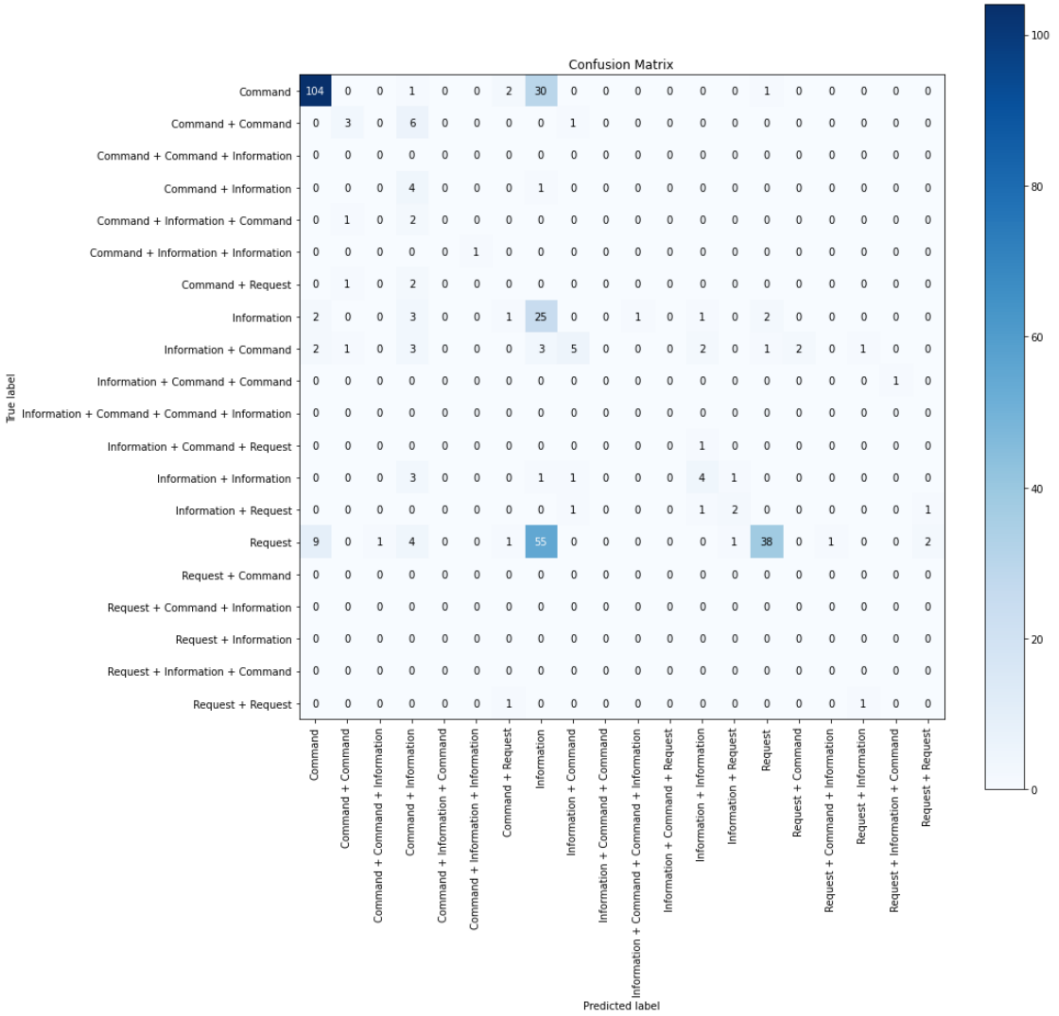


Figure 6.11. Confusion matrix of the corrected sentences belonging to the 346 misrecognised.

6.4 Collaborative Speech for HRI

At the beginning of this chapter, it was established that speech serves as the most intuitive and natural means of communication for collaborative robots functioning in indoor and outdoor settings. Although several applications address spoken Human-Robot Collaboration (HRC), no single system unifies multilingual capabilities, multi-user interaction, offline functionality, robust data logging, safety mechanisms, and modularity. Within the CANOPIES project, the solution must seamlessly integrate with perception, motion, and manipulation modules on a physical robot platform.

In light of the constraints observed in existing systems, this section introduces a novel speech-based communication pipeline—entirely grounded in speech acts and

frame semantics—to improve collaboration between humans and robots in outdoor environments. Each module of this architecture is explained in detail, with particular emphasis on the aforementioned features. Furthermore, an analysis of the average CPU time required by each module is provided to illustrate the efficiency of the implemented system.

Consequently, a more streamlined pipeline was developed without relying on Large Language Models (LLMs) for generating spoken outputs. Nonetheless, LLMs were incorporated in subsequent development phases, as detailed in Chapter 8. Overall, this approach yielded a comprehensive and flexible system that can be deployed effectively in a wide range of real-world collaborative scenarios.

6.4.1 System Architecture and Implementation

The developed spoken pipeline for empowering collaboration between humans and robots in outdoor environments, specifically in table-grape vineyards, is presented in Figure 6.12. The system has been implemented with a special emphasis on the following features, which are especially important in the context of the CANOPIES project:

- **multilingual**: the user has the chance to engage with the robot using any of the following supported languages: English, French, German, Italian, or Spanish. The android will communicate using the same language spoken by the human.
- **multi-user**: the robot can manage requests from multiple users by recalling who made each request.
- **offline**: no module within the speech pipeline necessitates an internet connection, making the system fully operational in an offline capacity and viable even within environments featuring limited internet connectivity.
- **robust**: data logging is pivotal in robotics, enabling transparency and informed decisions. Hence, the implemented logging strategy is straightforward yet effective. Every module input is logged, safeguarding against communication disruptions between nodes by ensuring swift troubleshooting and seamless operations. Each log is saved in a dedicated JSON file, to facilitate efficient retrieval and analysis.
- **safe**: the system preserves human safety by notifying the user about criticality issues when unexpected situations are encountered by the robot.
- **modular**: each module of the pipeline has its own dedicated ROS package to allow future changes and improvements.

The system was developed based on the proposed modular architecture in section 6.4.1, which consists of 7 modules, each with a dedicated section explaining the motivation behind the implementation choices and providing further details. The

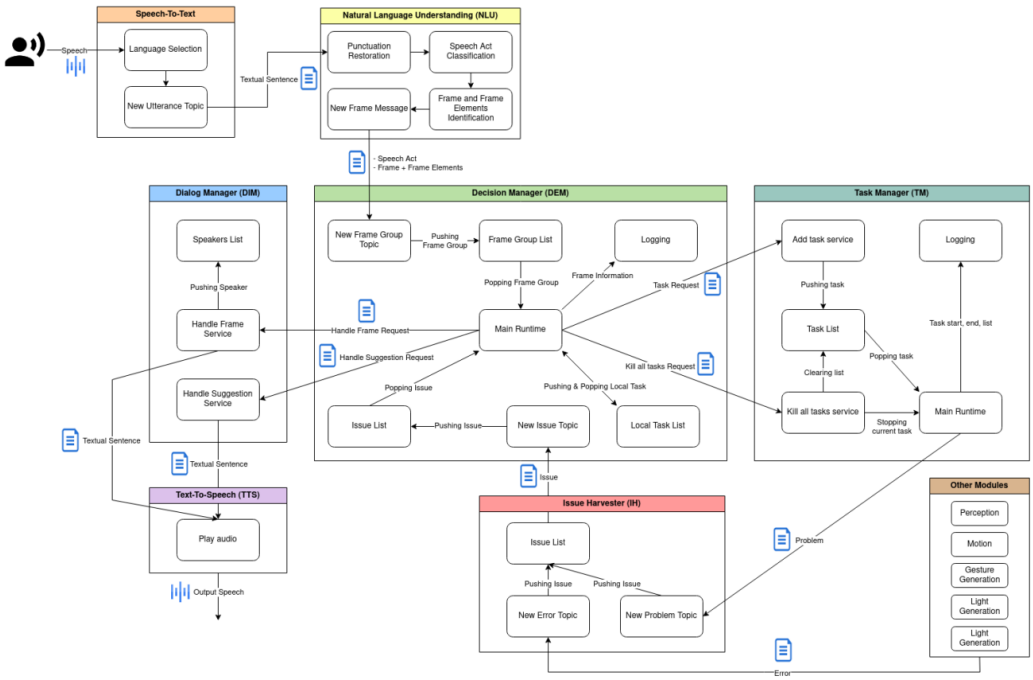


Figure 6.12. The proposed speech-based architecture.

communication between modules occurs through the use of ROS topics. The proposed pipeline is closely aligned with frame semantics, drawing upon a predefined set of English frames encompassing general-purpose and domain-specific predicates. Since FrameNet and other existing repositories do not offer frame elements that are intuitively applicable and easily extendable to precision agriculture scenarios, a custom collection of predetermined elements—such as "Theme," "Location," "Attribute," "Possessor," and "Time"—was defined to minimise user effort in interpreting the robot's understanding. Although the framework is developed with table-grape vineyards in mind, it is deliberately structured to enable adaptation to various application contexts. In practice, modifications to the Dialog Manager and Task Manager allow the pipeline to be tailored to individual project requirements.

Speech-To-Text Module

The Speech-to-Text (STT) module captures vocal inputs from users and transcribes them into textual commands. This module ensures that spoken instructions are processed efficiently, even in dynamic and noisy environments. The identification of the most suitable tool for the Speech-To-Text (STT) module is of fundamental importance to ensure stable and robust communication between human workers and robots in CANOPIES. Obtaining accurate transcriptions from speech with a low percentage of uncertainty and a minimal number of words erroneously recognised is already challenging in indoor environments. It becomes even more difficult in outdoor scenarios. When developing a voice-based system, external factors like an unreliable internet connection and disruptive background noise can compromise the

quality of interaction. To this aim, Vosk [7], an offline open-source speech recognition tool supporting multiple languages, has been employed in our pipeline. It has been demonstrated to be particularly precise in vocal utterance transcriptions also with smaller models.
Prior to starting the interaction, to load the appropriate Vosk model, the user is requested to indicate the preferred language for communication with the robot among the following ones: English, French, German, Italian, or Spanish.

NLU Module

The Natural Language Understanding (NLU) module is responsible for analysing the textual transcription received from the STT component. It identifies key elements such as intent, speech acts, and frame semantics and ensures that commands are correctly classified and contextualised before being processed by higher-level decision-making components. Therefore, the process of punctuation restoration is of significant importance to split long and complex sentences into shorter (atomic) ones, allowing for the precise assignment of a speech act, representing the informative content exchanged during an interaction with each of them. For instance, in Human-to-Human Interaction, it is common for people to use a combination of sentences belonging to different speech act classes while communicating, such as a "Command", followed by an "Information" and concluding with a "Request", as in: *Turn left, there is a ripe grape on your left, do you need help in harvesting it?* is illustrated in the Figure 6.13. Before entering the process of frame elements prediction, utterances in languages other than English must undergo a translation phase. To this aim, we considered the Helsinki-NLP/opus-mt [241] translation models, which can transform a statement from any of the four other languages supported by the system to English. However, to obtain the frame elements prediction of a sentence, information about words, lemmas, positional tags, dependency heads and frames must be provided as shown in Figure 6.14. Therefore, Stanza library [186] has been employed to collect the first four required data. The frames, instead, have been defined by us through a lemma-frame association, representing the robot's knowledge and allowing the system to precisely classify novel utterances. The combination of a frame with the associated speech act class is called a *frame group*.

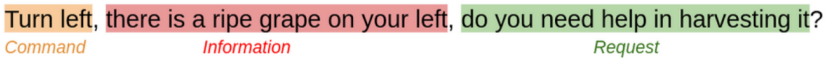


Figure 6.13. Speech act identification by dividing the long sentence based on punctuation.

Decision Manager

At the heart of the developed speech pipeline, the Decision Manager (DEM) emerges as a pivotal orchestrator, adeptly managing issues and tasks in real-time. This module cleverly navigates through a network of features and concepts, ensuring prompt issue resolution and task execution. It verifies input consistency, resolves

[7] `https://github.com/alphacep/vosk-api`

```
{"sentence_id": 3,
      "words": ["Posizionati", "a", "destra", "per", "raccogliere", "l'", "uva", "."],
      "predictions":
          {"0":
              {"predicate": "PLACE",
               "roles": {"1,3": "Location", "3,7": "Purpose"}},
           "4":
              {"predicate": "HARVEST",
               "roles": {"1,3": "Location", "5,7": "Theme"}}},
      "speech_act": {"0,7": "Command"},
      "lemmas": ["posizionare", "a", "destra", "per", "raccogliere", "la", "uva", "."],
      "pos_tags": ["V", "E", "S", "E", "V", "RD", "S", "FS"],
      "dep_head": [0, 3, 1, 5, 1, 7, 5, 1]
}
```


Figure 6.14. A training sample of the SRL system

ambiguities, and determines the appropriate response based on available multimodal cues. It acts as the central control unit, ensuring that interactions are coherent and task execution follows user intentions. Moreover, it demonstrates flexibility when encountering new frame groups provided by the NLU module. For the sake of brevity, from now on, the term *frame* is intended as a combination of the predicate and associated frame elements.

In normal operative conditions, for each received frame group, the DEM module sends a *"handle frame"* request to the Dialog Manager. Depending on the presence of an associated executable task, the Task Manager may be engaged for task execution. Additionally, task execution is concluded by reproducing an auditory response to indicate its completion. Fundamental features of the frame definition are: remarkably fast response generation and error handling. For instance, if a user queries the system regarding the color of grapes and provides an unrelated term, such as "airplane", the DIM immediately provides an issue message, clarifying that "airplane" is not a valid color for grapes.

Dialogue Manager

The **Dialog Manager (DIM)** orchestrates seamless human-robot interactions by managing conversation flow. Upon receiving a frame, the DIM logs it in the speaker's conversation history and validates it.

- **Invalid Frames:** The DIM promptly generates an *"error"* message to guide the conversation.

```
{
    "CHECK": {
        "request": {
            "args": [
                {
                    "name": "object",
                    "values": {
                        "grape": [true, null, false],
                        "temperature": [false, false, false],
                        "ground": [false, false, false]
                    }
                },
                {
                    "name": "quality",
                    "values": [
                        "bad",
                        "good",
                        "ripe"
                    ]
                },
                {
                    "name": "color",
                    "values": [
                        "purple",
                        "pink",
                        "green"
                    ]
                }
            ],
            "handle": "handle_check"
        }
    }
}
```

Figure 6.15. Single frame definition example.

- **Valid Frames:** The DIM checks if the frame is linked to an external task:
  - If **yes**, it retrieves the associated task and generates a *"start task"* message.
  - If **no**, and the frame pertains to a local operation (e.g., adding a speaker), a *"result"* message is produced.

For first-time interactions, the system stores the user's **name, surname, and spoken language** and assigns a unique identifier(UUID), while archiving the full conversation to ensure continuity in future interactions. If no active conversation exists, the module initiates a *"conversation introduction"* message to establish context.

To enhance inclusivity, the DIM detects the speaker's **language preference** and translates responses accordingly.

A key improvement in the **CANOPIES project** is the introduction of **frame definitions**, stored in a structured **JSON file** as shown in Figure 6.15. These definitions categorise actions into:

- **Targets** (e.g., *grape* or *temperature*)—the main subject of the action.
- **Arguments** (e.g., *quality* or *color*)—additional details refining the request.

Furthermore, the system validates argument values to prevent misinterpretations. For instance, if a user inquires about **grape quality**, the system recognises valid responses like *bad, good,* or *ripe*. If an invalid term, such as *airplane*, is provided, the DIM instantly issues an *error message*, maintaining dialogue integrity.

This structured approach ensures **fast response generation, efficient error handling**, and **precise user intent recognition**, significantly enhancing Human-Robot Interaction.

Task Manager

The Task Manager (TM) is characterised by strategic capabilities and essential traits that ensure smooth task completion and management. It oversees the execution of assigned commands, prioritising and queuing actions based on the received communication input. It interacts with the physical robot, ensuring that the system responds effectively to user instructions. The TM iterates over each task, starts the assignment, and executes it. Once a task is accomplished, the module verifies whether an outcome is returned (e.g., the quality of the grapes) and notifies the DEM about its completion. In order to associate each task with the service in charge of execution (e.g., the quality checking module in CANOPIES), the TM is configured with a list of available services and their association with predicates and frame elements.

Issue Harvester

The Issue Harvester (IH) is responsible for gathering, transforming and managing issues within the developed speech system. When a new problem arises, the IH module turns it into an actionable issue that can be easily managed. The IH acts similarly when encountering new errors, by converting them into issues through a defined transformation process. It ensures that the robot can notify the user when assistance is required, enhancing system reliability and safety. These issues fall into one of the following categories: "error" and "problem" refer to Table 6.5, which are then forwarded to the DEM for appropriate management. This process is then repeated iteratively, underscoring the module's commitment to addressing issues in a systematic and efficient manner.

Text-To-Speech Module

The Text-To-Speech (TTS) emerges as a dynamic component for seamless communication. Once a response is formulated, the TTS module generates verbal feedback, ensuring bidirectional communication between the human and the robot. PicoTTS[8] library, an offline open-source speech synthesiser supporting multiple languages, has been employed in our pipeline. Carefully designed, this module encompasses a variety of advanced elements that are arranged together to transform written inputs into engaging spoken outputs. A key aspect is its ability to work with different languages. Indeed, before generating the spoken output, the TTS verifies if the requested language is compatible. Such comprehensive analysis ensures that the

[8] `https://pypi.org/project/py-picotts/`

Severity Level	Type of Issue	Description
1	Minor Error	Such issues are associated with minor software errors that do not interfere with task execution. The error is only logged, and human safety is preserved.
2	Minor Problem	These issues involve minor problems in task execution. The issue is logged, and the user is vocally notified by the robot about the problem with the current task. Human safety is preserved.
3	Major Problem	These issues relate to significant problems in task execution. The issue is logged, and the robot vocally notified the user about the problem encountered. The android requests human support and prioritises the person's suggestions for overcoming the obstacle. Human safety is preserved.
4	Major Error	These issues involve critical software errors or hardware malfunctions where human safety is not ensured. The system stops the execution of the ongoing task, deletes all queued tasks, blocks the DEM runtime from accepting new tasks, and vocally notifies the user of the major error.

Table 6.5. Issue categorisation and explanation

verbal interaction feels authentic to the user, in line with the project's aim of fostering meaningful engagement. The TTS dexterously handles different types of commands using a decision framework.

When receiving a "say" command, the module works on the provided text to create audio recording for playback. This process is supported by a well-organised file management system that safeguards generated audios while efficiently clearing space by removing older recordings. In response to a "play" command, the TTS quickly confirms the presence of the specified audio file, allowing smooth playback. In case the audio file is missing, the module's error handling promptly intervenes. Hence, this proactive issue management strategy enables developers to swiftly identify and fix problems, promoting an efficient development process. After concluding its internal checks, the TTS module plays the audio, delivering the synthesised speech to the user.

6.5 Empirical Results and Discussion

In evaluating our HRI pipeline, we conducted preliminary experiments by engaging Italian, English and French native speakers. Each user was requested to vocally introduce three sentences in their native language, formulating each utterance ten times. Participants wore a headset to communicate, in order to minimise background noise and closely replicate the ideal way in which the interaction between humans and robots would occur in practical scenarios. Furthermore, French and Italian participants were also encouraged to interact in English. Communicating in a non-native language played a crucial role in determining the error rate linked to the STT

module, as well as the count of inaccurately recognised speech acts and misclassified frame elements in the NLU component.

The time between the end of the user's utterance pronunciation and the start of the system's response was recorded and analysed. The outcomes of our investigation are summarised in Table 6.6, which presents specific information such as the input sentence, interaction language, speech act, frame semantics, and both the worst and the best response times. Table 6.7 provides details about the average time required by each module. On average, we observed that the STT module accurately transcribed vocal utterances 8 times out of 10 when users interacted in English, even if it was not their native language. Regarding NLU, this module was expected to utilise more time to process the sentences since first, the prediction of the speech act was made, then the association of the frame elements, which are both computationally expensive. On the other hand, TTS was not expected to employ more than 1 second to reproduce the utterance vocally, but it was noticed that in the final value, the time required to generate, save, open and play the audio was also taken into account. The system appropriately generated error responses when faced with sentences containing information beyond its knowledge. For instance, frame semantics were not generated for the sentence "Hello, can you cook?". In this case, the system did not recognise the frame "cook", resulting in a response notifying the user: "I do not know how to cook".

The pipeline's issue management was showcased in a French context, where the robot was queried with the prompt, "Can you check the grape?". In such a scenario, the robot could not assess the fruit's condition due to a *major problem*. The issue stemmed from the ripe grape being previously harvested by a human worker. Consequently, the robot informed about the missing cluster and inquired whether the person could address the issue, as illustrated in Figure 6.16. In case of confirmation, the speech pipeline prioritises tasks subsequently communicated by the user.

Precisely, once the STT module transcribed the vocal utterance into its corresponding textual transcription, the NLU component analysed the sentence by identifying the "Request" speech act along with the frame "CHECK" and element "grape". Such information was passed to the DEM, in order to verify if the data received from the NLU was valid and forwarded them to the DIM, while generating the "check_grape" task to be handled by the TM. However, the task could not be executed due to the disappeared grape, and the TM sent a message to the IH in order to manage the encountered "major problem" to whom criticality 3 was assigned. Such issue was then communicated back to the DEM, so to inform DIM to provide a "Suggestion" to the person. Hence, this interaction between the modules is fundamental in our pipeline when an issue is encountered by the robot, so the user can assist in resolving it. In such cases, the robot prioritises the user's subsequent commands by placing them at the beginning of both the "Frame List" and "Task List" ensuring the completion of the current task before moving on to the next one.

Sentence	Language	Speech Act	Frame Semantics	Worst time	Best time
Follow me	English	Command	COME-AFTER_ FOLLOW-IN-TIME (me)	2.0109 s	1.6553 s
The ground is wet	English	Information	BE_EXIST (ground)	4.2523 s	3.8717 s
Can you help me?	English	Request	HELP (me)	3.1524 s	2.7651 s
Allez à droite *(Go to the right)*	French	Command	GO (right)	8.7831 s	7.1507 s
Le raisin est malade *(The grape is sick)*	French	Information	BE_EXIST (grape, sick)	11.559 s	9.2889 s
Pouvez-vous vérifier les raisins? *(Can you check the grape?)*	French	Request	CHECK (grape)	9.6143 s	7.3952 s
Muoviti avanti lentamente *(Move forward slowly)*	Italian	Command	GO (forward, slowly)	9.6244 s	7.7365 s
Le foglie sono secche *(The leaves are dry)*	Italian	Information	BE_EXIST (leaves, dry)	10.6146 s	8.0871 s
Riesci a esaminare il terreno? *(Can you examine the ground?)*	Italian	Request	CHECK (ground)	10.499 s	7.3775 s

Table 6.6. Evaluation on the HRI speech pipeline.

Sentence	STT	NLU	DEM	DIM	TTS
Follow me	0.0007 s	1.1117 s	0.0049 s	0.0097 s	0.7301 s
The ground is wet	0.0016 s	1.1613 s	0.0021 s	0.0115 s	2.8913 s
Can you help me?	0.0013 s	1.1303 s	0.002 s	0.0228 s	1.7923 s
Allez à droite \textit{(Go to the right)}	0.0003 s	6.4374 s	0.0065 s	0.0067 s	1.4604 s
Le raisin est malade \textit{(The grape is sick)}	0.0023 s	7.7119 s	0.002 s	0.0116 s	2.7797 s
Pouvez-vous vérifier les raisins? \textit{(Can you check the grape?)}	0.0008 s	6.7171 s	0.0063 s	0.0067 s	1.5798 s
Muoviti avanti lentamente \textit{(Move forward slowly)}	0.0016 s	6.8703 s	0.0047 s	0.0072 s	1.5592 s
Le foglie sono secche \textit{(The leaves are dry)}	0.0018 s	6.4003 s	0.002 s	0.0097 s	2.6335 s
Riesci a esaminare il terreno? \textit{(Can you examine the ground?)}	0.0011 s	6.4513 s	0.0047 s	0.0076 s	1.6583 s

Table 6.7. Average time required by each module of the HRI pipeline.

```
HUMAN REQUEST: ['Can you check the grape?']
ROBOT ANSWER: ['I am checking grape']
ROBOT PROBLEM: ['The grape I was trying to check disappeared']
ROBOT SUGGESTION: ['Are you able to fix the problem?', 'Please respond only by yes or no']
(1 minute passed)
ROBOT ANSWER: ['I did not receive your response, so I am ending the current task']
```

Figure 6.16. System's issue management.

Additionally, based on these findings, it also emerged that handling multiple languages might result in translation delays influenced by both input sentence and response lengths. However, in the best cases, the interaction time was below 10

seconds [224] on CPU, showcasing the real-time capabilities of our system and the potential for significant processing time reduction in the NLU module when the pipeline is run on a GPU. Additionally, participants expressed particular appreciation for the system's response speed, transcription accuracy, and comprehension correctness, underlining their interest in future interactions with the real robot.

Chapter 7

Gestural Human-Robot Interaction

Non-verbal communication cues play a pivotal role in human-robot interaction (HRI), enabling intuitive collaboration in environments where speech alone is insufficient due to noise, distance, or cultural barriers. Non-verbal communication, such as gestures and facial expressions, can convey a lot of information about a person's intentions, emotions, and attitudes. Since non-verbal interaction conveys much information about a person's intentions, emotions, and attitudes and brings continuity or completes the communication [249], enabling robots to understand and respond to human signals is crucial to bridge the interaction gap. As discussed in Chapter 1, in agricultural settings like vineyards, where fluctuating illumination and machinery noise disrupt traditional communication channels, gestures provide a robust alternative for conveying commands, directing attention, and coordinating tasks. This chapter advances gestural HRI by establishing a taxonomy of context-aware gestures tailored to viticulture-based workflows for CANOPIES project, such as pruning and harvesting, and designing a scalable gesture recognition pipeline validated across virtual and real-world domains. Building on synthetic data generation techniques (Chapter 5), gesture datasets curated using virtual avatars in simulated environments, train pose estimation models to interpret both full-body and hand-centric gestures, and evaluate their performance on indoor/outdoor logistics robots and outdoor farming robots. Through rigorous testing in virtual reality and field experiments, these systems demonstrated that gesture-based interaction improves safety, efficiency, and adaptability in shared workspaces, thereby establishing the groundwork for integrating gestures and speech into a cohesive multimodal framework to be discussed in Chapter chapter 8.

7.1 Gesture Taxonomies and Data Acquisition

There were several definitions of gestures based on the applications and their usage context, yet most still need to be standardised. Regarding industrial applications, gesture definitions change according to the type of industry and context used. General movement and context-specific gestures were defined to generalise and standardise the signs in the industry. These were defined after following some

gestures for communicative purposes [235] in HRI and from the gesture language between human-human and human-robot and in the workspace for the CANOPIES project. The total of 21 defined gestures consists of 10 dynamic and 11 static signs, where 9 gestures can be performed using two hands, while the rest are with one hand. Some gesture definitions were specific to the CANOPIES project [*CUT, CHANGE BOX, RECEIVE, TAKE*], while the rest are generic to ground robots. Human workers in the table-grape field will be working along with robots in performing agricultural activities, which require at least one hand in most cases to instruct the robot, without interrupting their activities. The 9 gestures that need two hands can have any other tools in the hands of the human in most scenarios. Table 7.1 explains the gestures defined and how the gestures are performed, and the pictorial explanations for each gesture are shown in Figure 7.1.

7.1.1 Full-Body and Hand-Centric Gestures

Functional human-robot gestural interaction has been explored since the late 1990s when the recognition of six different arm gestures was used to control a wheeled robot. Gesture-based interaction has been traditionally paired with speech-based interaction to enrich the communication spectrum, and it can substitute speech entirely in noisy environments or be used to substitute such tools as teach pendants when human operators can not get their hands free. Since the first attempts, gesture recognition has been used in different applications to interact with robots. User-defined hand and finger gestures have been coupled together with face identification to allow people with disabilities to control an intelligent wheelchair [136].

In general terms, providing a universal definition of gesture remains a challenge, as existing literature lacks a single, overarching classification. However, in the specific context of human-robot interaction (HRI), it is possible to define gestures within a functional framework [194]. The concept of functional gestures is widely accepted in the literature, yet often remains implicit, as most works developing gesture-based *Human-Machine Interaction* (HMI) techniques and frameworks do not explicitly define the term. From an analysis of state-of-the-art literature, functional gestures in HMI scenarios can be described as trajectories of body motion or static poses, intentionally performed to convey meaningful information or interact with the surrounding environment. In this context, we will refer to functional gestures simply as gestures.

By identifying human-performed gestures to facilitate control or communication with devices, gesture recognition enables intuitive and contactless interaction. According to [48], *Gestures are body actions that humans intentionally perform to affect the behaviour of an intelligent system.* The concept of intention is fundamental to this process, as gestures inherently express meaning and purpose. Recognised gestures may include waving, pointing at an object or device, opening a hand, clapping, touching both hands or sign language movements. These air gestures provide a natural and seamless mode of interaction, offering an intuitive alternative to conventional input methods in human-robot collaboration. Thus, gesture recognition

Gesture Command	Gesture Description	Type	Body Parts Involved
Start	Start the gesture module	Dynamic	Two Hands
Stop	Stop the current command (can be used on multiple commands)	Static	One Hand
Pause	Pause the current action until resume command is given	Static	Two Hands
Resume	Resume the action from the Pause command	Static Dynamic	Two Hands
Attention	Get the attention of the Robot (This can be used to take commands from another person in the field of view of Robot)	Dynamic	One Hand
Take an item (Object Detection in Human's hand)	To give the human a tool or some object or grape bunch (Place object in the Hand of the human)	Static	One Hand
Receive an item (Hand and position Tracking)	To give the Robot a tool or some object or grape bunch (object in the Hand of the human)	Static	One Hand
Point an item?	Point an object or area in the field of view of Robot (Object detection enabled and finger pointing direction will be tracked)	Dynamic	One Hand
Move forward	Ask the robot to move forward. (Default distance will be set based on the available space for the robot to move.)	Dynamic	One Hand
Move backward	Ask the robot to move backward. (Default distance will be set based on the available space for the robot to move.)	Dynamic	One Hand
Move Left	Ask the robot to move Left. (Default distance will be set based on the available space for the robot to move.)	Static	One Hand
Move Right	Ask the robot to move Right. (Default distance will be set based on the available space for the robot to move.)	Static	One Hand
Turn Left	Ask the robot to Turn left. (Default check will be set based on the available space for the robot to move.)	Dynamic	Two Hands
Turn Right	Ask the robot to Turn Right. (Default check will be set based on the available space for the robot to move.)	Dynamic	Two Hands
Turn Around/Back	Ask the robot to Turn back (180). (Default check will be set based on the available space for the robot to move.)	Dynamic	One Hand
Slow down	Ask the robot to slow down (moving or any task)	Dynamic	One Hand
Stop executing Everything	The robot will stop executing any work it is doing. This includes the movement. (Robot will continue to be in pause if the previous command is paused before this gesture)	Static	Two Hands
Cut the bunch; Start Pruning; Start Harvesting	Start cutting the grape bunch(es). It can be combined with a pointing gesture to cut a specific bunch. To begin the activity of pruning. (The same gesture of Harvesting can be used as the activities take place in different timelines)	Static Dynamic	One Hand Two Hands
Change the basket	To change the baskets in the truck when the basket is full. Detection algorithm will estimate if the bunch is full.	Dynamic	Two Hands
Follow me	Ask the robot to follow the human.	Dynamic	Two Hands

Table 7.1. Full-Body Gesture definitions. Applicable for General UGV's and CANOPIES.

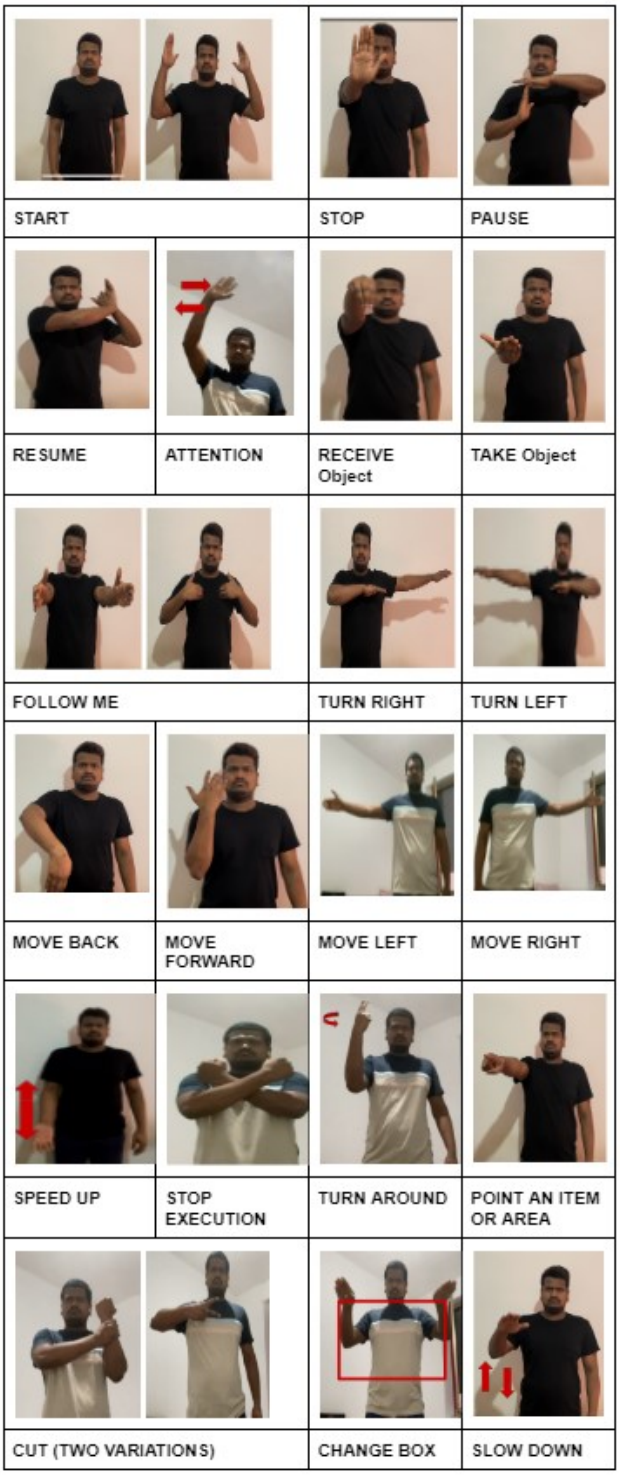


Figure 7.1. Full-Body Gesture definitions based on the workspace ethic and human communication. Applicable for general UGV's and CANOPIES.

serves as a crucial bridge between human intent and robotic response, enhancing both usability and interaction efficiency.

Most of the literature and the implementations point towards 1. Full-body gestures, 2. Hand gestures. Full-body gestures are being used in systems through pose detection and other techniques based on depth; the same applies to Hand gestures.

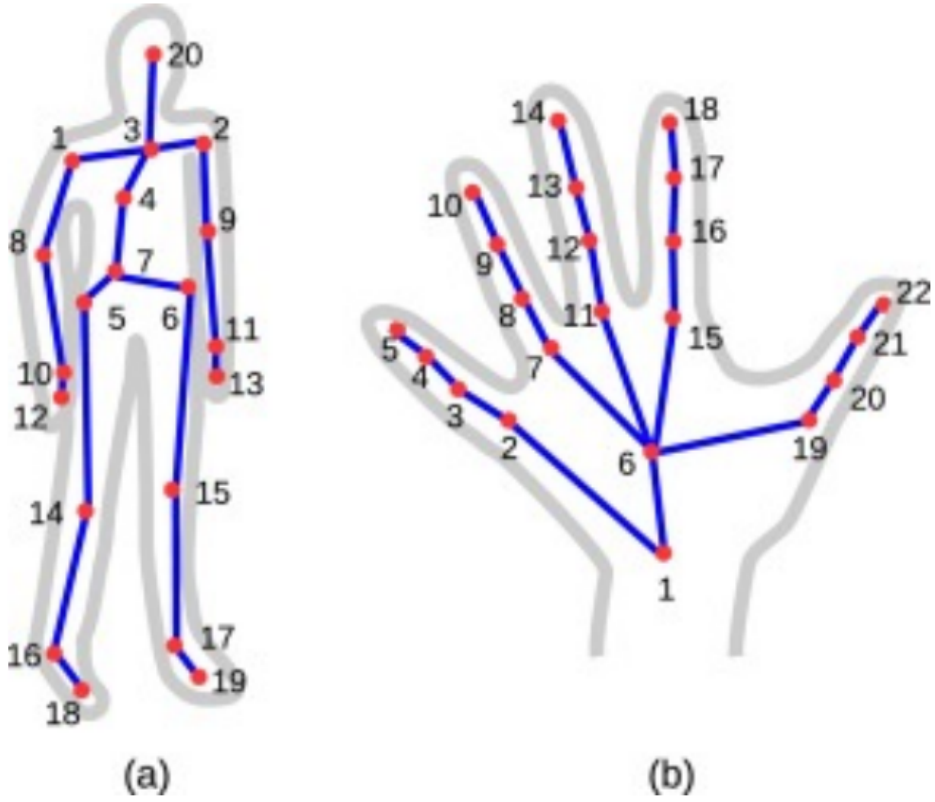


Figure 7.2. General representation of keypoint mapping of Full-body (a) and Hand (b) for pose estimation and gesture recognition.

7.1.2 Pose Estimation using MediaPipe

MediaPipe offers a sophisticated framework for human pose estimation, enabling researchers and developers to detect and track human body landmarks in both two-dimensional and three-dimensional space. This technology represents a significant advancement in computer vision applications, providing accessible tools for implementing complex pose detection systems across various platforms. The framework implements BlazePose, a lightweight neural network pipeline capable of predicting 3D landmarks of the human body in real-time, including detailed hand representations, from a single monocular image [29]. This technology runs efficiently on most modern mobile phones and browsers, democratising access to sophisticated pose estimation capabilities.

Full-Body Pose Estimation

The BlazePose framework employed by MediaPipe represents a significant advancement in on-device pose detection technology. It utilises a machine learning solution designed for high-fidelity body pose tracking, inferring the 3D coordinates of 33 joint points on the whole body in real time from each frame of RGB video [148]. The system implements a sophisticated pipeline that processes input images through several stages to ensure consistent performance across different input sources and environmental conditions.

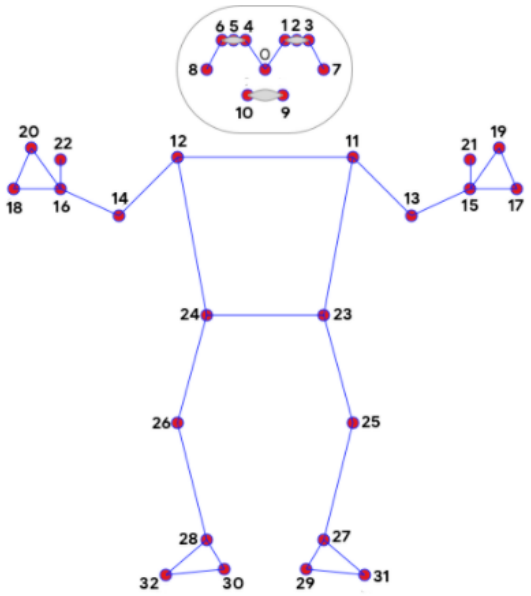
The Z-coordinate in the framework represents depth, with the hip midpoint serving as the origin. Smaller values indicate proximity to the camera, creating a consistent spatial reference system [148]. This dimensional representation enables applications to interpret not just the position of body parts within the frame but also their spatial relationships in depth, providing critical information for applications requiring spatial awareness of human positioning.

Recent evaluations demonstrate MediaPipe's performance metrics, with improvements showing average accuracy increasing from 71.76% to 93.52% through optimisation techniques [148]. However, researchers note that pose estimation accuracy is highly dependent on camera viewing angle and the specific exercises performed, with accuracy decreasing under less favourable conditions [69]. These findings highlight the importance of controlled implementation environments for applications requiring high precision.

3D Keypoint Representation and GHUM Model The three-dimensional keypoint system in MediaPipe transforms flat image detections into spatially meaningful representations through the integration of the Google Human Model (GHUM), a parametric 3D human model. This statistical human shape and pose model captures correlations between body parts to ensure consistent pose estimations [266]. The GHUM model supports both moderate-resolution (10,168 vertices) and low-resolution (3,194 vertices) representations, providing flexibility for different processing capabilities [266].

Rather than solving inverse kinematics for given 2D skeletal models, the framework employs a heuristic optimisation method that directly adjusts camera-relative body angles and intra-body joint angles to match 2D projected humanoid models to 2D skeletal models [130]. This approach addresses the depth ambiguity problem by incorporating loss functions that consider the deviation of centre of mass from supporting feet and appropriate penalty functions for natural joint angle rotations [130].

The GHUM lifter neural network takes concatenated 3D body and hand landmarks as input and outputs mesh parameters, employing an MLP-Mixer architecture that processes 75 tokens (33 body landmarks as shown in Figure 7.3 and 42 hand landmarks) [97]. This architecture transforms the landmarks to produce GHUM



0. nose
1. left_eye_inner
2. left_eye
3. left_eye_outer
4. right_eye_inner
5. right_eye
6. right_eye_outer
7. left_ear
8. right_ear
9. mouth_left
10. mouth_right
11. left_shoulder
12. right_shoulder
13. left_elbow
14. right_elbow
15. left_wrist
16. right_wrist
17. left_pinky
18. right_pinky
19. left_index
20. right_index
21. left_thumb
22. right_thumb
23. left_hip
24. right_hip
25. left_knee
26. right_knee
27. left_ankle
28. right_ankle
29. left_heel
30. right_heel
31. left_foot_index
32. right_foot_index

Figure 7.3. MediaPipe Pose estimation landmarks

Source: [29]

state parameters for pose and shape, enabling high-fidelity reconstruction of human forms from detected landmarks. Comparative evaluations show the BlazePose GHUM Holistic approach achieves a Mean Per Joint Positional Error with Procrustes Alignment (MPJPE-PA) of 78mm and MPJPE of 121mm, outperforming other methods while operating an order of magnitude faster [97].

Hand Pose Estimation

MediaPipe's hand pose estimation technology represents a specialised subset of its human tracking capabilities, focusing on the detailed articulations of human hands. The system employs a dual-model approach comprising a palm detection model and a hand landmark model working in concert to provide comprehensive hand tracking. The landmark detector identifies 21 specific hand keypoints, creating a detailed skeletal representation of hand positioning that captures the complex articulations of fingers and palm [29].

The spatial transformer approach crops high-resolution hand regions from input images, enabling detailed analysis without requiring the entire pipeline to process full-resolution images. This optimisation allows for efficient processing while maintaining the fidelity necessary for tracking fine hand movements. The system further implements intelligent tracking to minimise computational demands, using bounding boxes defined by hand landmarks from previous frames to localise hands in subsequent frames [97].

Integration with Full-Body Tracking The BlazePose GHUM Holistic approach addresses limitations of standard on-device body landmark prediction by including

hands and fingers, enabling unified motion capture for the entire body. The integration of hand landmarks (21 per hand) with the 33 body landmarks creates a comprehensive skeletal representation that captures both gross motor movements and fine manipulations [29].

This holistic approach provides a unified representation for applications ranging from avatar control to gesture detection and motion analysis. The system outputs handedness classification to distinguish between left and right hands, enabling applications to implement hand-specific interactions and analyse bimanual activities. This classification proves particularly valuable for applications requiring detailed understanding of which hand performs specific actions [97].

MediaPipe's pose estimation framework provides researchers and developers with sophisticated tools for human pose tracking in three-dimensional space. The integration of BlazePose with the GHUM model creates a powerful system capable of detailed full-body and hand pose estimation from single monocular images. Despite current limitations in certain environmental conditions and with non-standard body types, the technology demonstrates impressive accuracy and efficiency across diverse implementation scenarios.

7.1.3 Hybrid Dataset

The hybrid dataset represents a synergistic integration of virtual (synthetic) data (VD) and real-world data (RD) at different percentage combinations, as detailed in sections 5.2.1 and 5.1.4, respectively. This combined dataset leverages the complementary strengths of both data sources to enhance the robustness and generalizability of gesture recognition models.

The synthetic component, generated using the Unity simulation framework (Section 5.2.1), comprises 26,000 frames of gesture data from 30 virtual characters. These data incorporate diverse variations in illumination, camera angles, noise, and distance configurations, ensuring scalability and controlled environmental diversity. Conversely, the real-world component (Section 5.1.4) consists of 25,192 frames captured in a table-grape vineyard using an Intel RealSense Depth camera. It includes gestures performed by 8 individuals across age groups and genders, recorded at multiple distances (1.5–6 meters) under authentic field conditions. Both datasets are aligned to 13 classes: 12 static gestures for robot navigation/initiation, one standing pose, and an *"Unknown"* Pose class for outlier detection.

By merging these datasets, the hybrid dataset achieves a total of 51,192 frames, balancing the realism of physical-world dynamics with the scalability and variability of synthetic data. This combination mitigates the inherent limitations of each individual dataset: synthetic data, while versatile, may lack nuanced environmental interactions, whereas real data, though authentic, is resource-intensive to collect. The hybrid approach ensures broader coverage of scenarios, including variations in lighting, occlusions, human demographics, and background complexity, while

maintaining class consistency. This integration not only amplifies training data volume but also reduces overfitting risks by exposing models to both idealized and real-world conditions. Subsequent experiments (Section 7.2.1) demonstrate that models trained on the hybrid dataset exhibit improved generalization, particularly in challenging field environments, compared to those trained on purely synthetic or real data alone.

7.2 Gesture Recognition Pipeline

The necessity of a modular and robust system that could exploit gestures in HRI to enhance the robot's comprehension of specific information motivated us to develop an innovative application for recognising gestural interaction between humans and robots. The proposed architecture is implemented on ROS Noetic, on a computer running Ubuntu 20.04 LTS. The system and its evaluation are deployed on an Alienware x17 R2 with 32 GB of RAM, a 12th-generation i9 CPU, and a Nvidia RTX 3080 Ti GPU notebook. The pipeline, presented in Figure 6.8, is created for training and evaluating the gesture recognition model on:

1. Exclusively gestural data generated from human avatars.
2. A combination of virtual and real information.

The developed architecture consists of two main pathways. The first one, depicted by a continuous black line, serves as a reference or ground truth during experiments with alternative models. It also functions as the primary direction for assessing the data, extracting pose landmarks, and determining the gesture class for Real-Time Data (RD), Virtual Data (Synthetic Data) (VD), or a combination of data sources. The second pathway, represented by the blue line, involves gesture evaluation using either a CNN or a Transformer model. This pathway acts as a supplementary algorithm to improve the final gesture classification. Hence, the pipeline design is highly flexible, allowing for the use of either RD, VD, or a fusion of real and simulated information. Additionally, users can employ one or both recognition algorithms in the process. Indeed, depending on the configuration, either of the pathways can behave as a complete framework. This adaptability enables the incorporation of one or more detection models into the pipeline to suit specific needs.

7.2.1 Implementations and Empirical Results

Experiment Design

The experimentation was conducted in three distinct phases to answer the research question, **"How do the models trained on Hybrid data perform in real-world?"**. The first phase generates more virtually simulated data by creating additional virtual human characters as discussed in the section 5.1.1. These characters were mapped to the extracted gesture animations. For the preliminary investigation of this phase, the simulation environment recorded the gestures performed by these characters through multiple cameras using the Unity tool from various distances and angles. These characters are mapped to 7 gestures of the extracted gesture

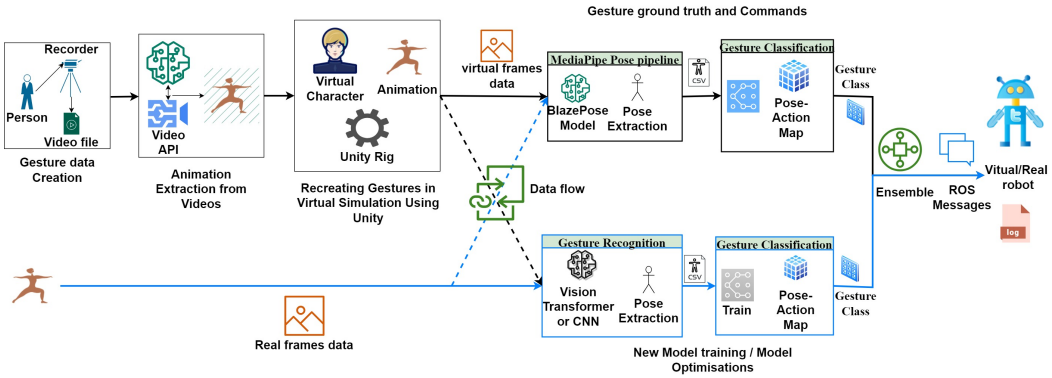


Figure 7.4. Gesture Pipeline for generation and evaluation on virtual simulated data. Blue and black lines indicate the flow of data into two models, which can act as individual recognition for a unified pipeline based on data and model configuration.

animations, out of which 6 are static, and 1 is a dynamic sign. The selected motions belong to the initiation and navigation of the robot in different directions. Moreover, to be able to recognise also the signs out of the chosen classes, an extra category of *Unknown* Pose is introduced for classification purposes.

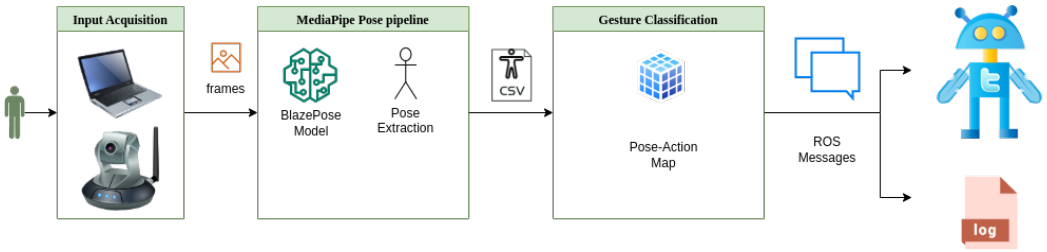


Figure 7.5. Initial gesture recognition pipeline using Mediapipe

The Second phase of the experimentation uses Mediapipe's pose estimation [29] to evaluate the generated data as discussed in section 5.2.1. The recorded frames were evaluated to check if the key points were being recognised and at what threshold and taken as a base score. In the case of virtual simulated data, the identified keypoints were evaluated using Unity-generated joint coordinates as the ground truth. A Convolutional Neural Network (CNN) of 8 convolutional layers was trained on the key points obtained by running the pose estimation algorithm and utilised to classify the avatar position into one of the predefined gesture classes. These results were depicted in Figure 7.6. This laid the foundation for the initial gesture recognition pipeline using Mediapipe.

In the third phase of the experimentation, more suitable deep-learning approaches are adopted to train diverse systems on the acquired data. These experiments and evaluations are based on the hypothesis that a model trained on the hybrid data can be used to identify the gestures both in simulation and the real world. A suitable deep-learning approach was adopted to train a network on such data. The

network was trained only on real data, Virtual-simulated data, and hybrid data. This dataset was explained in the section 7.1.3. Mediapipe's model and the trained CNN models were compared against each other using variations of training data. After evaluations, the possible data combinations (virtual-real) were evaluated to improve the network's performance. Hence, the network with the best performances will be deployed to a virtual robot in the simulation, being able to see the virtual human avatar performing gestures so that the virtual robot will communicate or execute the corresponding actions based on the gesture commands given by the virtual character. The users in the simulation will have control of choosing gestures via a keyboard (input device) in a *non-immersive* experience and with the hand controllers in an *immersive* experience based on the device. All these experiments will further be catalogued for system enhancements, and the model's performance evaluation will be conducted on real-world data by deploying such an approach into the real robot. The same experiment was repeated with various combinations of RD and VD with simple multiples of 10, i.e., for example, 30% RD + 70% VD from their respective frames contribute to the combined dataset for training and evaluating the network. Two 2D pose estimation deep-learning approaches were identified to train on such data.

1. MoveNet(single pose)(Thunder) [239] uses MobileNetV2 as its backbone, followed by Centernet with a depth multiplier of 1.75.

2. ViTPose [268] with YOLO V5 as its backbone for person detection, followed by a transformer block to estimate the key points.

These networks were pre-trained on the MS COCO Human Keypoint Detection dataset and then fine-tuned on the defined gestures. The same CNN was adopted with modifications to the input layer to incorporate the changes in the keypoint data structure, but the rest of the layers remained the same to have consistency across these 3 algorithms. These pose algorithm-based CNNs were trained on real data, virtual-simulated data, and several combinations of data with train-val-test proportions of 70-15-15% on the data. The fixed test data was set to validate real, virtual, and combination data. Mediapipe's model and the trained models were compared using variations of training data. A probability weight-based voting ensemble between the two best models determines the gesture class to communicate to the robot for task execution. This two-model approach was employed as the final recognition pipeline as shown in Figure 7.4for real-world deployment, as false positives were reduced.

Emperical Results on Synthetic Data

The evaluation results on gesture recognition through pose estimation using the virtual simulated data are presented and discussed in this Section. By the end of the two phases of experiments presented previously, 3,380 frames of gestures were acquired from all 30 characters for 8 (7+1) gestures. However, after post-processing the frames for inconsistencies and left with 3,096 frames for these classes. All these frames were passed through Mediapipe's pose estimation algorithm for estimating the key-points. As mentioned in section 7.2, the pose classification network (CNN)

could assign one of the 7+1 gesture classes to the input frame. To this aim, the outcome of this evaluation is presented in Figure 7.7. From this analysis, it was noticed that the gesture misclassification depends on one (or multiple) issues:

1. Pose occlusion with any other item or person in the simulation.

2. Camera angle for the character as the parts of characters occlude some of the keypoints in certain angles.

3. Lack of brightness and animation inconsistencies.

Figure 7.6. Pose detected on Virtual Human Avatars performing animated gestures in the table-grape field simulation.

The overall performance accuracy in all the classes was recorded to be 94%, whereas the individual classes' precision ranged between 92 - 98%. The recorded f1-score was 0.94. Since the primary goal is to evaluate the developed gesture recognition module and classify the gestures in an ideal situation in which gesture failures are not detected and propagated among the pipeline by the gesture recognition module, all the acquired frames were double-checked and the ones that are inconsistent with the animation were removed Hence, the gesture classification network was run on the corrected frames to identify the issues presented previously from these classification outcomes. In future, additional pose estimation networks will be evaluated on this data to draw comparisons between the networks and evaluate their performances in hybrid data settings. The outcome of the evaluation is presented in Figure 7.7 and Table 7.2

TARGET / OUTPUT	Start	Stop	Move_left	Move_right	Turn_left	Turn_right	Pause	Unknown
Start	387 12.50%	3 0.10%	2 0.06%	0 0.00%	3 0.10%	1 0.03%	1 0.03%	24 0.78%
Stop	5 0.16%	381 12.31%	6 0.19%	2 0.06%	1 0.03%	1 0.03%	0 0.00%	12 0.39%
Move_left	2 0.06%	1 0.03%	336 10.85%	1 0.03%	1 0.03%	2 0.06%	1 0.03%	10 0.32%
Move_right	1 0.03%	3 0.10%	5 0.16%	352 11.37%	2 0.06%	0 0.00%	0 0.00%	7 0.23%
Turn_left	2 0.06%	1 0.03%	1 0.03%	2 0.06%	348 11.24%	2 0.06%	1 0.03%	5 0.16%
Turn_right	6 0.19%	2 0.06%	1 0.03%	1 0.03%	1 0.03%	365 11.79%	0 0.00%	15 0.48%
Pause	2 0.06%	1 0.03%	1 0.03%	1 0.03%	1 0.03%	0 0.00%	371 11.98%	3 0.10%
Unknown	22 0.71%	1 0.03%	2 0.06%	1 0.03%	2 0.06%	1 0.03%	2 0.06%	379 12.24%

Figure 7.7. Confusion Matrix for 7 Gestures and an Unknown gesture class.

Class Name	**Precision**	**1-Precision**	**Recall**	**1-Recall**	**f1-score**
Start	0.92	0.08	0.91	0.09	0.91
Stop	0.93	0.07	0.97	0.03	0.95
Move_left	0.95	0.05	0.95	0.05	0.95
Move_right	0.95	0.05	0.98	0.02	0.96
Turn_left	0.96	0.04	0.97	0.03	0.97
Turn_right	0.93	0.07	0.98	0.02	0.96
Pause	0.98	0.02	0.99	0.01	0.98
Unknown	0.92	0.08	0.83	0.17	0.88
Accuracy	0.94				
Misclassification Rate	0.06				
Macro-F1	0.94				
Weighted-F1	0.94				

Table 7.2. Training performance statistics for 7 gestures and the Unknown gesture class.

Emperical Results on Hybrid Data

After evaluations, Table 7.3 shows the possible data combinations (virtual-real) and the respective network's performance. All these experiments were catalogued for system enhancements and the model's performance. Evaluations were conducted on both real-world and virtual data by employing these models using the camera on the robot.

Train Data		Test Data		Mediapipe +CNN		MoveNet + CNN		VitPose + CNN	
Real	Virtual	Real	Virtual	Accuracy	F1-Score	Accuracy	F1-Score	Accuracy	F1-Score
100	-	✓	X	0.85	0.85	0.74	0.74	0.80	0.80
100	-	X	✓	0.81	0.81	0.71	0.70	0.79	0.79
-	100	✓	X	0.95	0.95	0.76	0.76	0.85	0.85
-	100	X	✓	0.97	0.97	0.78	0.78	0.85	0.85
80	20	✓	✓	0.83	0.82	0.68	0.67	0.71	0.70
70	30	✓	✓	0.85	0.85	0.69	0.69	0.72	0.74
60	40	✓	✓	0.89	0.89	0.71	0.70	0.74	0.74
50	50	✓	✓	0.91	0.91	0.72	0.71	0.78	0.78
40	60	✓	✓	0.91	0.92	0.73	0.72	0.79	0.79
30	70	✓	✓	0.93	0.93	0.74	0.73	0.81	0.81
20	80	✓	✓	**0.94**	**0.93**	**0.76**	**0.76**	**0.83**	**0.83**
100	100	✓	✓	**0.97**	**0.96**	**0.78**	**0.77**	**0.86**	**0.86**

Table 7.3. Results of evaluations conducted on various data combinations and various Pose estimation + CNN algorithms. Accuracy and F1-Score readings are evaluated on test data samples. Train Data represents the percentage combinations of real and virtual data used to train the models. ✓ represents data used to test and X marks not used.

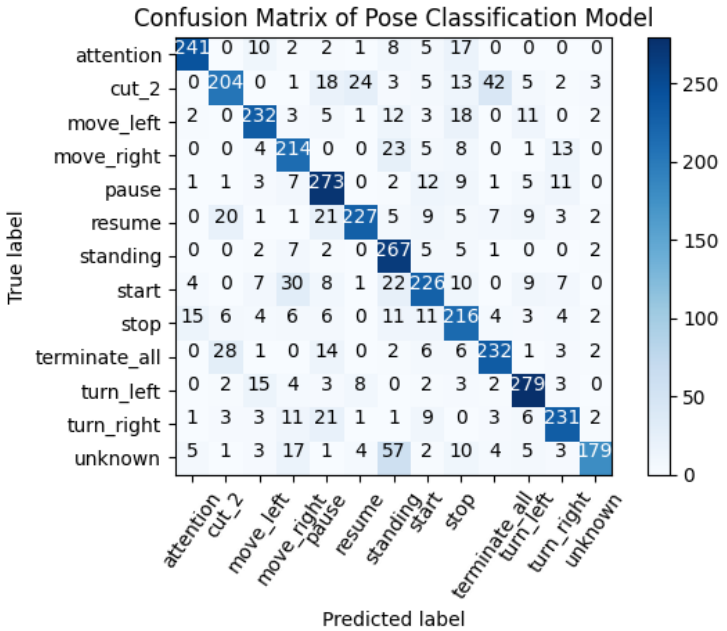


Figure 7.8. Confusion matrix for 13 Gesture classes using 20% RD and 80% VD.

By the end of the experimentation, various data combinations were evaluated and it was observed that the models trained on full RD and VD datasets performed better due to the availability of more training data. However, As the goal is to use the

combination data, the 20-80 ratio of RD-VD models performed close to the full data combination. It was also observed that Mediapipe + CNN performed superior to others as the model evaluates 33(3D) landmarks where, whereas the MoveNet model evaluates 17(2D) landmarks and the ViTPose model 25(2D) landmarks. Based on observations, more landmarks result in more features to learn, leading to the model's better performance. The outcome of this evaluation is presented in Figure 7.8.

7.2.2 Performance in Indoor and Outdoor Settings

The gesture recognition pipeline was deployed across two distinct robotic platforms—the Farming Robot (FR) and the Logistics Robot (LR)—achieving a combined accuracy of 94% under diverse operational conditions, including fluctuating lighting, variable target distances, and heterogeneous hardware configurations in the person's hand. Failure in recognition resulted from several factors, such as participants' clothing being blended with the background and baggy clothing hindering joint recognition. Tall participants under the canopy faced difficulties with the '*stop*' command, often misclassified as '*move backwards*' due to hand position similarities. Camera placement also impacted gesture recognition accuracy. Cultural factors may also play a role. For example, Italian participants' natural inclination to use more gestures compared to other cultures may impact the performance of gesture recognition systems, increasing the number of false positives. The Farming Robot employed gesture commands to navigate in the field during harvesting, subjected to outdoor environmental variances. In real-world collaborative scenarios—where users might be carrying tools or objects, reaching around corners, or moving through crowded spaces—arms and torsos can quickly become obscured. However, full-body gestures are unaffected by the tools in the hand as the recognition algorithm does not take hand key points into consideration.

In indoor environments, the robot employs a MediaPipe-based hand gesture recognition system to facilitate intuitive user commands such as opening and closing the lid, following a person, stopping, and displaying the battery level (Figure 7.9). These commands were chosen for their practical relevance in day-to-day use and distinguishable hand gestures. The robot's (**Segway E1**) physical constraints were considered specifically as its fixed height of 1.15 meters and its downward-facing camera placement made it impractical to rely on full-body gestures since body movements or poses would often fall outside the camera's field of view or become distorted due to the steep viewing angle. Instead, focusing on hand gestures within a close interaction range of 0.5–1 meter ensured the robot could reliably capture and process the necessary landmarks, even in tight or cluttered indoor spaces.

To achieve high recognition accuracy under these constrained conditions, the standard MediaPipe hand landmark detection framework was optimised for close-range interaction. This involved tuning parameters such as detection confidence thresholds, region-of-interest scaling, and tracking stability settings, all of which improved precision when a user's hand is very close to the camera. Extensive testing in controlled indoor lighting conditions demonstrated a 98% recognition accuracy, underscoring the robustness of the approach for everyday deployment. From a

Hand Gesture	Command
👌	Close Lid
👍	Open Lid
✌️ / Inverted	Display Battery percent
🖐️	Stop
🖐️ inverted	Follow-Me

Figure 7.9. Hand-gestures and their respective commands for logistic robot

computational standpoint, the entire gesture recognition pipeline was deployed on an NVIDIA Jetson Orin edge processor-enabled Segway E1 delivery robot. This processor meets the computational expense of running the robot within a small power footprint, which is critical for a mobile, battery-powered robot that needs to maintain real-time performance. Even with all the overhead required for vision-based detection and the subsequent gesture classification, the system maintained a solid 25 frames per second (FPS) throughput. This rate was found to be sufficiently responsive for seamless user interaction while staying within the robot's computational and energy constraints.

Finally, by including hand gestures, the system remains adaptable to different robot types and scenarios based on their hardware configurations and is robust to occlusions and partial hand visibility. In the case of robots with small stature, the hands of the human often stay within view of the downward-facing camera, and the optimised hand landmark detection can handle brief periods of partial occlusion. As a result, the robot can consistently recognise user commands without needing large, exaggerated motions, making the overall interface more natural, efficient, and user-friendly in indoor settings.

Chapter 8

Enhancing Collaboration with Multimodal Interaction

Multimodal interaction is a key advancement in improving communication between humans and robots. Traditionally, HRI has been limited to unimodal exchanges, predominantly relying on either speech or gestures. However, integrating multiple modalities—such as speech, gestures, and visual cues—enhances the robustness, adaptability, and naturalness of interactions. This chapter explores how multimodality strengthens collaboration by improving the accuracy of command interpretation, reducing ambiguity, and increasing the efficiency of task execution in real-world scenarios.

Humans naturally employ multimodal communication, often combining speech with gestures, facial expressions, and other non-verbal cues to enhance clarity. In HRI, multimodal integration allows robots to overcome challenges posed by environmental conditions, such as background noise or poor visibility, by leveraging redundant or complementary information. Redundant interaction occurs when the same information is conveyed through multiple channels, reinforcing the message and increasing recognition accuracy. For instance, a user saying, "Move there" while pointing towards a location provides redundant cues that allow the robot to determine the intended destination with greater confidence. Complementary interaction, on the other hand, occurs when different modalities contribute distinct pieces of information that together form a complete message. For example, a user might say, "harvest the grape bunch" while pointing at a specific fruit, where speech provides the action, and the gesture provides the target object.

The architecture presented in this chapter is designed to support multimodal interaction by systematically integrating speech and gesture inputs into a multimodal fusion-based decision-making pipeline. This architecture consists of multiple core modules that enable real-time processing, information fusion, and action execution. Notably, the architecture is designed with modularity in mind, allowing for the future integration of Large Language Models (LLMs) to extend its capabilities. While LLMs are introduced later in the chapter, the foundational modules remain the same, ensuring that multimodal interaction is handled consistently and efficiently at

the core.

8.1 Multimodal Interaction in HRI: Integrating Speech and Gestures

The Multimodal Human-Robot Interaction (MHRI) architecture builds upon the Collaborative Speech Pipeline Architecture presented in subsection 6.4.1, extending it with gesture-based interaction and multimodal fusion. The architecture consists of the following additional modules and is shown in Figure 8.2.

8.1.1 Gesture Acquisition and Recognition Module

The *Gesture Acquisition* module collects visual input from the camera feed to identify user gestures and transforms it into a ROS message. This message is published to the dedicated ROC topic ready to be subscribed by the pipeline. The *Gesture Recognition* module discussed in section 7.2 subscribes to this topic and classifies the detected movements into predefined categories, allowing the system to interpret non-verbal commands accurately. The designated topic to which the *Gesture Recognition* module is subscribed becomes the channel for this publication, ensuring a seamless flow of visual data within the system.

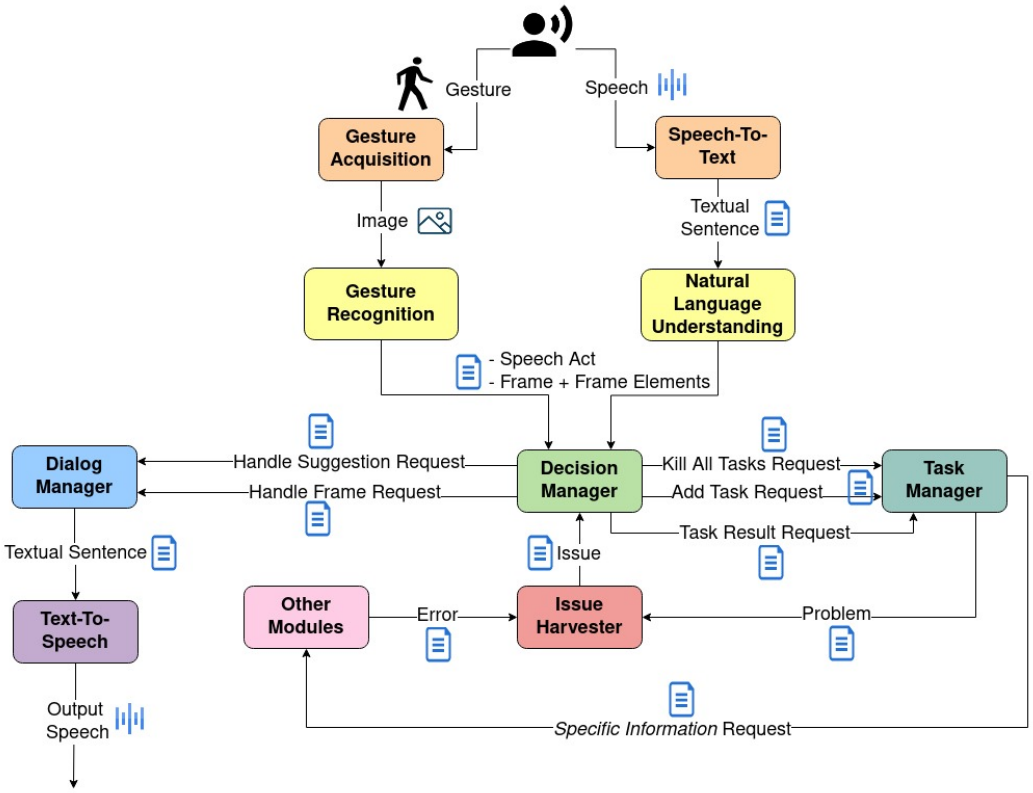


Figure 8.1. Multimodal Human-To-Robot Interaction pipeline handling Speech and Gestures

8.1.2 Multimodal Fusion Module

This module integrates speech and gesture data to determine whether they convey complementary or redundant information. It plays a crucial role in ensuring the robot accurately interprets the user's intent by cross-referencing multiple input sources. The schematic information flow between these communication modules is shown in Figure 8.1.

For the remaining seven modules—Speech-to-Text, Natural Language Understanding, Decision Manager, Dialogue Manager, Task Manager, Issue Harvester, and Text-to-Speech — refer to subsection 6.4.1, where they have been discussed in detail. Below, we illustrate examples of how these modules function in the multimodal interaction pipeline:

- **Speech-to-Text & Gesture Fusion:** When a user says "Move left" while pointing in a direction, the Speech-to-Text module transcribes the speech, and the Gesture Recognition module interprets the pointing gesture. The Decision Manager validates the consistency between the two inputs before instructing the robot.
- **Dialogue Manager Handling Ambiguity:** If a user gives an ambiguous command like "Go there," the Dialogue Manager engages in a clarification dialogue by asking the user to specify the destination.
- **Task Execution through Multimodal Input:** If a worker in an agricultural setting says, "Pick that grape" while pointing, the Task Manager prioritises executing the harvesting action only after confirming the gesture-defined object with the vision system.
- **Issue Harvester Detecting Errors:** If the robot misclassifies a command due to background noise, the Issue Harvester logs the error and prompts the user for clarification to ensure safe operation.

Task Manager (TM)

TM module (discussed in section 6.4.1) oversees task management, including the addition of new assignments to the task list and notifies the Decision Manager upon their completion. Tasks are appended to the list with a designated name and an ID for identification in case a termination request is received. When an issue is encountered during the task execution, this module kills the current assignment after receiving termination instructions from the Decision Manager. This module was upgraded to a state machine that handles other modules on the described architecture for seamless information exchange and control of the hallucinations of LLMs. Command tasks available for execution by the robot encompass:

- stop: halts the current task, signalled with a red illumination of the entire LED strip at the robot's rear

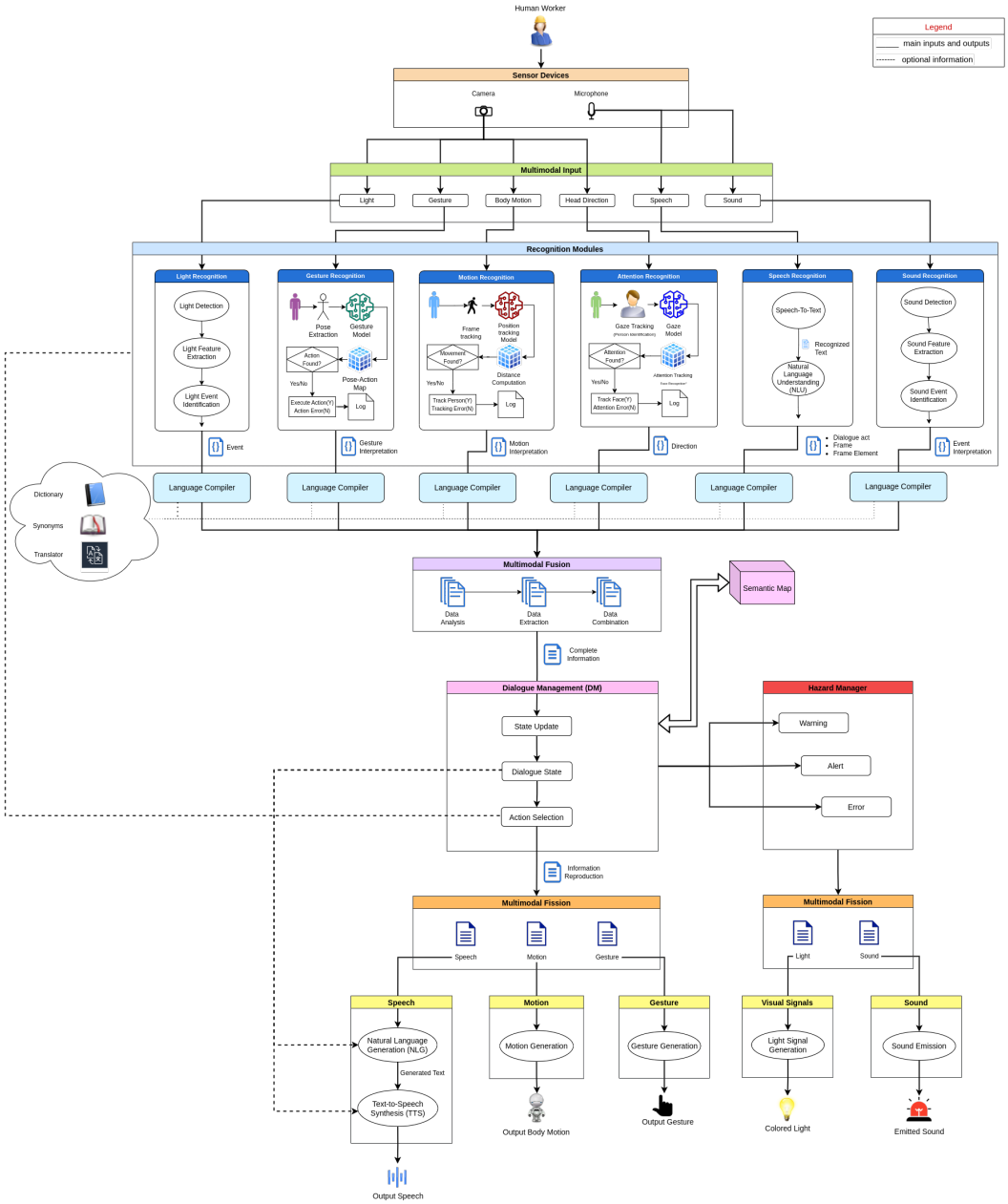


Figure 8.2. General multimodal architecture.

- go_forward: prompts the robot to move forward with intermittent green light signals until a "stop" command is received
- go_backward: initiates backward movement, signalled with intermittent red light and a high-frequency "beep" until stopped
- turn_right: rotates the robot 90° right, emitting orange light signals from the right corners
- turn_left: turns the robot 90° left, illuminating orange LEDs at the front and back left corners
- turn_180: prompts a 180° rotation
- check: enables the robot to perform verifications of specific elements in the vineyard based on frame argument values
- harvest: lifts the robot to gather ripe grapes
- prune: involves cutting dry or broken leaves or branches

Light signals and **sounds** enhance user awareness in both virtual simulation and real scenarios. Indeed, actions can be executed either by the digital or the actual robot by adjusting the topics where the messages are sent. Due to hardware constraints, lights and sound emission hardware were not installed on the farming and logistic robots in the CANOPIES project. However, these modules were tested in the simulation environment using the designed multimodal communication fusion architecture. Communication with the "Visual Perception" module, responsible for assessing grape attributes, such as *"brix level", "colour", "size", and "position of the grape bunch"*, is facilitated through a function that actively parses this data in a JSON file. Accessing grape-related details within a specific frame is more efficient, empowering the robot to provide particular information and users to provide requests or commands related to grape ripeness, cluster count and harvest plans.

8.1.3 Hybrid Decision-Level Fusion for Multimodality

Before integrating LLMs, multimodality was achieved by leveraging deterministic rule-based processing in combination with sensor fusion techniques. The system relied on structured mappings between speech and gesture inputs, using predefined grammar-based recognition models to interpret commands. While effective, this approach had limitations in handling ambiguous or complex instructions, as it required predefined scenarios and lacked contextual adaptability.

By employing a hybrid decision-level fusion approach, the architecture was able to combine complementary and redundant inputs effectively. Complementary multimodal fusion allowed gestures to fill in missing details when speech was insufficient, while redundant fusion improved recognition confidence by cross-verifying multiple input sources. This ensured reliable interaction in challenging environments, such as agricultural settings where auditory and visual conditions can fluctuate.

The architecture was designed with the flexibility to extend over several models and algorithms in mind, ensuring that the same core modules responsible for interaction handling would remain functional even with the introduction of LLMs. Adding LLMs is an enhancement rather than a replacement, allowing the system to extend its capabilities without disrupting the fundamental processing pipeline. A user study was conducted using the simplified version of this system to assess multimodal interaction in HRI and check what interaction comes on top. The user study conducted with this systemic approach is discussed in the next section.

8.2 Assessing Multimodal Communication in HRI

A user study was conducted in a virtual simulation environment to evaluate the effectiveness of multimodal interaction in HRI. The primary objective was to compare the usability, efficiency, and user preferences of speech-only, gesture-only, and multimodal interaction modalities. Previous studies have established the robustness of multimodal communication, but limited research has explored user experience factors such as trust, perceived competence, and perceived discomfort. The study aimed to address these gaps by analysing how different communication modalities impact user satisfaction and performance in a controlled yet realistic setting.

8.2.1 Study Design and Experimental Setup

This user study was conducted to evaluate the effectiveness of multimodal interaction in HRI. The study simulated real-world conditions with varying background noise levels and illumination changes, mimicking outdoor environments where such interactions are most applicable. The same system (refer to section 5.4) and environment setup were utilised for all input modalities. A Jabra SPEAK 510 USB Bluetooth audio device was used for the speech input, which can toggle the microphone on/off to reduce noise when the user is not speaking. Intel's RealSense D435i Camera served as the vision input for the gestures. Both devices were employed for multimodal interaction.

The primary motivation for conducting a multimodal interaction study stems from the challenges encountered in outdoor environments. As discussed in the introduction, these settings are frequently noisy, posing difficulties for a single modality to provide all the necessary information for effective robot performance. This study investigates the impact of multimodal input, precisely the combination of speech and gestures, on user experience. Each modality has been individually assessed to comprehensively evaluate this impact, resulting in three distinct input settings: speech, gestures, and multimodal interactions. During our experiments, if speech input proves inadequate, the robot prompts the user for additional details using either speech or gestures. If speech fails to provide the required information, gestures take precedence. This approach ensures the robot can accurately interpret commands and execute tasks effectively, capitalising on the strengths of both modalities to tackle the complexities of outdoor settings.

Input modality	Age	Male	Female	Other	Prefer not to say
Speech-only	25.60	20	10	0	0
Gestures-only	26.43	25	5	0	0
Multi-modal	30.43	21	7	1	1

Table 8.1. Experimental groups

The gestures designed for the CANOPIES project (refer to Figure 7.1) were carefully curated to ensure they do not overlap with any natural spoken gestures, making it challenging for participants to learn how to interact with the robot using these gesture-based modalities. If a participant were first to experience the gesture-only modality and then the multimodal setting (or vice versa), they would have an advantage in the second experiment, which could undermine the study's validity. A between-subjects design was employed to prevent bias, ensuring participants engaged with only one modality during the study. This design decision aims to maintain the integrity and validity of the research findings.

Participant Demographics

To conduct the experiments, we recruited 90 participants from the Dipartimento di Ingegneria Informatica Automatica e Gestionale Antonio Ruberti (DIAG) at Sapienza University and the Istituto Tecnico Agrario Emilio Sereni in Rome. All data collected were anonymous, with no personal identifiers. Participants had no speech impairments or physical disabilities, enabling them to participate in any of the three modalities without constraints.

The participants were evenly divided into three groups, with 30 individuals per group. Among the 90 participants, 66 identified as male, 22 as female, 1 as 'other', and 1 preferred not to disclose their gender. Educational backgrounds varied: 11 participants were in high school, 6 had completed high school, 14 were enrolled in a bachelor's program, 32 were pursuing a master's degree, and 27 were either pursuing or had completed a doctorate. The average age of the participants was 27.49 years. Table 8.1 presents the mean age and gender distribution of each group, while Figure 8.3 illustrates the education level distribution per group.

Regarding prior experiences, 52 participants had interacted with virtual reality (VR) before. On a 7-point Likert scale (ranging from 'never' to 'daily'), participants rated their experiences with robots in VR at an average of 3.18, indicating infrequent interaction with simulated robots. Similarly, their experience with real robots averaged 3.15 on the same scale. Lastly, when asked about their experience in vineyards, the average score was 3.08, suggesting limited familiarity with vineyard environments.

Virtual Reality

The proliferation of virtual reality (VR) technologies has opened up new avenues for studying human-robot interaction (HRI) in various contexts [23]. The efficacy

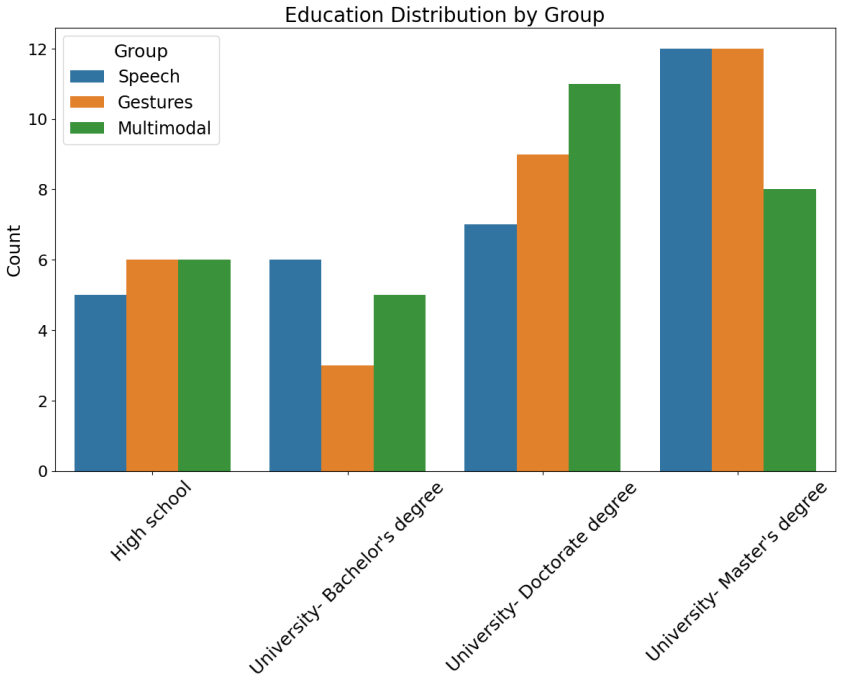


Figure 8.3. Distribution of participants' level of education per group.

of utilising VR for HRI research in agriculture has been demonstrated within the CANOPIES project. Moreover, the logistical challenges associated with transporting both the robot and participants to a vineyard environment have been addressed by conducting experiments within a simulated environment. In this simulated setting, participants' real actions, such as speech or gestures, directly influence the behaviour of the simulated robot, allowing for a more controlled and efficient study of HRI interactions.

Tasks

All participants, regardless of group, were tasked with instructing the robot to complete specific actions. During the task, participants in all groups stood within 2 meters of the simulation display. The tasks were designed to be identical across all groups and involved navigating the robot to three boxes placed in the field. The boxes were positioned to the right, in front of, and to the left of the robot's starting point (see Figure 8.4). Participants were required to navigate the robot to each box, and upon approaching each one, the robot would automatically collect it. Once all boxes were collected, participants had to direct the robot to harvest a grape bunch. Instructions were to be conveyed in the participant's assigned modality.

The instructions included six commands: *move forward, move backwards, turn left, turn right, stop, harvest.* Since the participants were Italian, the vocal commands for the speech and multimodal groups were given in Italian. However, the system could be set to other languages if necessary. The specific Italian commands used were: *'vai avanti'* (move forward), *'vai indietro'* (move backward), *'gira a sinistra'*

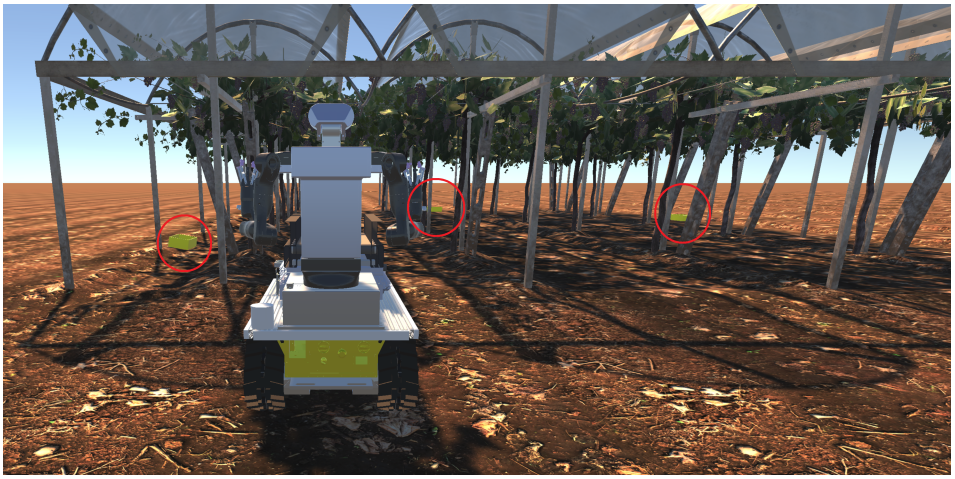


Figure 8.4. Participant's initial view of the simulation, each red circle depicts the placement of a box and was not part of the simulation.

(turn left), *'gira a destra'* (turn right), *'fermati'* (stop), and *'raccogli'* (harvest). The associated gestures for the gestures and multimodal groups are shown in Figure 8.5.

8.2.2 Measures and Instruments

A comprehensive questionnaire was administered to assess the participants' experiences after interacting with the robot to gauge key concepts: usability, trust, perceived competence, and discomfort. The complete questionnaire is presented in Table 8.6. Since the participants were native Italian speakers with varying levels of English proficiency, the questionnaire was translated into Italian and reviewed by individuals proficient in both languages to ensure clarity and accuracy, thus mitigating potential language-related ambiguities.

Usability

The Alternative Usability Scale (AUS) was employed to evaluate usability, which was extensively validated and researched [8]. This questionnaire includes a single item for each usability component, rated on a 7-point Likert scale ranging from 'strongly disagree' to 'strongly agree'. This allows for the analysis of learnability, memorability, efficiency, satisfaction, and error recovery, which are the five components of usability defined by Nielsen's Five-Factor Model [174].

Additionally, the perceived effort expended during the experiment was also assessed. This included mental and physical effort, measured using the Subjective Mental Effort Questionnaire (SMEQ)[215] and an adapted version of the SMEQ, respectively. The SMEQ asks users to rate the effort expended on the interaction on a scale from 0 to 150, as depicted in Figure 8.6. Objective usability measures were also employed to assess the participants' performance. Task duration was used as a critical metric, measuring users' time to complete the tasks [89]. This timing was used to evaluate relative user efficiency, comparing participants' task completion times with expert users.

Figure 8.5. Gesture commands

Perceived Competence and Perceived Discomfort

Both competence and discomfort are social perceptions of robots that can be measured using the Robotic Social Attributes Scale (RoSAS), a standardised 18-item questionnaire designed to assess perceived competence, warmth, and discomfort [49]. Given the CANOPIES robot's utilitarian nature, the warmth subscale was excluded from the questionnaire [153]. A detailed breakdown of the items evaluated in the RoSAS subscale can be found in Table 8.2. Each item was evaluated by participants on a 7-point Likert scale to determine the extent to which each item accurately describes the robot.

Trust

In the experiment, trust was one of several measures assessed, necessitating a concise questionnaire to minimise user burden. This led to the selection of a 3-item trust questionnaire [190]. Additionally, since one of the items overlapped with the Robotic Social Attributes Scale (RoSAS) reliability measure, only two items were included:

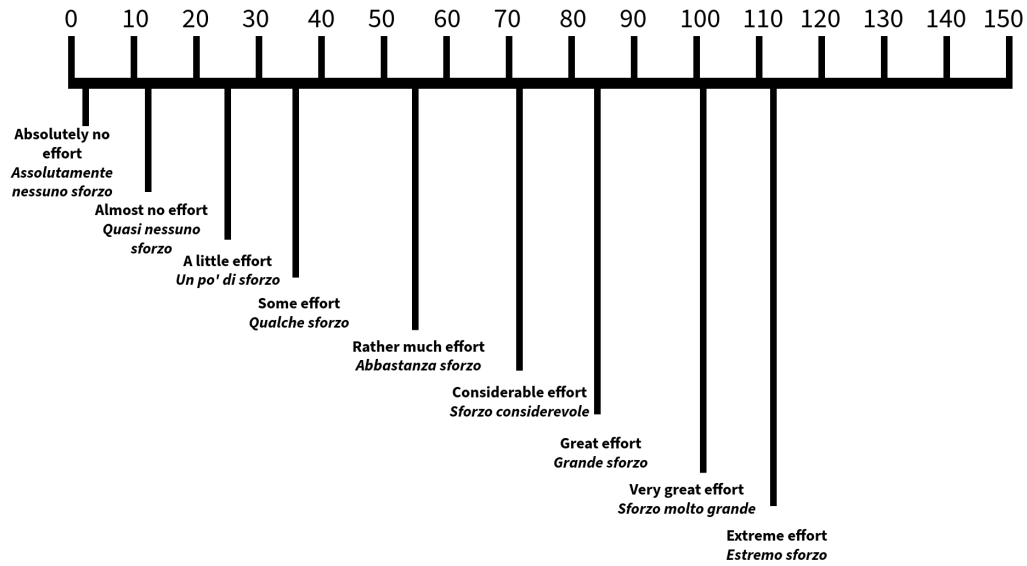


Figure 8.6. Subjective effort scale

Competence items	Discomfort items
Reliable	Scary
Competent	Dangerous
Knowledgeable	Aggressive
Interactive	Strange
Responsive	Awful
Capable	Awkward

Table 8.2. RoSAS items

'The robot is trustworthy' and 'I can trust the information presented by this robot'. Both items were evaluated on a 7-point Likert scale, ranging from 'strongly disagree' to 'strongly agree'.

8.2.3 Study Results

To facilitate statistical analysis of mental effort and physical effort, the participants' inputs had to be transformed. The increments of 10 were categorized from 0 to 14 in ascending order (e.g.,0-10 was turned into 0, 11-20 was turned into 1, and so on). The error recovery item from the usability scale was reversed and then averaged with the other four usability items (memorability, learnability, efficiency, and satisfaction). In order to assess the internal validity of usability as a single measure, Cronbach's alpha was computed [33]. This yielded a value of 0.55, which is below the established 0.70 needed to deem the validity acceptable, and thus usability is not further analysed as a single measure, but rather as separate subcomponents [33]. The two items of trust were averaged into a single trust measure, this measure yielded a Cronbach's alpha of 0.76, and thus trust can be analysed as a single measure. Additionally, the six items of perceived competence and the six items of perceived discomfort were

Table 8.3. Average scores (scaled from 0 to 1)

Measure	Speech	Gestures	Multimodal
* Relative user efficiency	0.21	0.46	0.50
Efficiency	0.62	0.70	0.59
Satisfaction	0.78	0.77	0.77
Memorability	1.00	0.84	0.69
Learnability	0.92	0.91	0.80
* Error recovery	0.50	0.39	0.47
* Mental effort	0.20	0.20	0.30
* Physical effort	0.00	0.25	0.26
Trust	0.72	0.68	0.63
Perceived competence	0.67	0.72	0.66
* Perceived discomfort	0.15	0.09	0.17

* Measures marked with an asterisk indicate negative aspects, where lower scores are better.

averaged into two single measures, these items yielded a Cronbach's alpha of 0.86 and 0.71, respectively, meaning that both items can be used as single measures [49].

To ensure the quality of the data, the resulting variables were checked for outliers within each group using the interquartile range (IQR) method [209]. Thereafter, the data was scaled on a range of [0,1] for each variable, and outliers were imputed with the mean for each variable in their respective groups [177]. After the transformations were completed, the mean score of each variable was calculated for the three groups, which can be seen in Table 8.3.

In order to determine the most suitable test for comparing means, it is essential to assess the normality of the distributions. The Shapiro-Wilk test, recognised as one of the most robust formal normality tests, particularly for small sample sizes, was employed for this purpose [200, 10]. Using a significance level of $\alpha = 0.05$, the following variables within each group demonstrated normality:

- **Speech:** relative user efficiency, memorability, physical effort, and perceived competence.
- **Gestures:** relative user efficiency, efficiency, trust, and perceived competence.
- **Multimodal:** relative user efficiency, error recovery, trust, and perceived competence.

Among these, *relative user efficiency* and *perceived competence* were the only variables that met the normality assumption across all three groups. Given that not all variables exhibited a normal distribution, a non-parametric method was employed to compare differences between group means. The Kruskal-Wallis test was

Table 8.4. Average scores of pairwise comparisons with significant differences

Measure	Speech	Gestures	Multimodal
* Relative user efficiency	0.21	0.46	
* Relative user efficiency	0.21		0.50
Memorability	1.00	0.84	
Memorability	1.00		0.69
Learnability	0.92		0.80
* Mental effort	0.20		0.30
* Physical effort	0.00	0.25	
* Physical effort	0.00		0.26

* Measures marked with an asterisk indicate negative aspects, where lower scores are better.

employed to assess the difference between groups [93]. Table 8.4 provides an overview of pairwise comparisons with statistically significant differences across group means, with detailed explanations of each variable presented subsequently.

Relative User Efficiency

The analysis revealed significant differences in relative user efficiency, with a $p-value < .001$, indicating statistical significance at $\alpha = .05$. However, the Kruskal-Wallis test alone indicates only that there is a significant difference among at least two of the three different groups. To delve deeper into these discrepancies, Dunn's test was employed as a post hoc analysis [75]. Subsequent analysis using Dunn's test unveiled significant differences between the gestures and speech groups, as well as between the multimodal and speech groups, both yielding *p-values* $< .001$. Conversely, no significant difference was observed between the gestures and multimodal groups.

To evaluate the practical significance of these discrepancies, Cohen's d and Cohen's *U3* were computed. These differences can be seen in Table 8.5. The table can be interpreted as follows: the differences between the 1^{st} group and the 2^{nd} group yield a difference, the effect size of this difference is expressed in Cohen's d, in which a value greater than $|0.8|$ indicates a large effect size, a value greater than $|0.5|$ indicates a medium effect size, and a value larger than $|0.2|$ indicates a small effect size [58]. Furthermore, Cohen's *U3* indicates the ratio of participants from the first group that scores higher than the mean of the second group [58]. For example, the comparison between multimodal and speech modalities resulted in a Cohen's d of 1.69, indicating a large effect size. Additionally, a Cohen's *U3* value of 0.96 was obtained, signifying that 96% of the multimodal group will be positioned above the mean of the speech group [58].

Memorability

The differences in memorability between modalities were found to be statistically significant, with a $p - value < .001$, which is below the established $\alpha = .05$. The

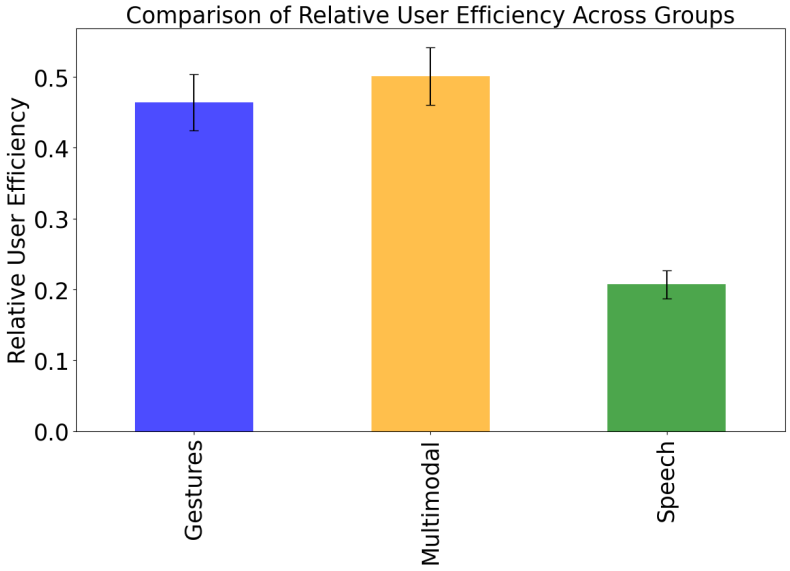


Figure 8.7. Comparison of relative user efficiency

Table 8.5. Cohen's d and Cohen's $U3$ of significant differences

Measure	1^{st} Group	2^{nd} Group	Cohen's d	Cohen's $U3$
* Rel. user efficiency	Gestures	Speech	1.49	0.93
* Rel. user efficiency	Multimodal	Speech	1.69	0.96
Memorability	Speech	Gestures	1.05	0.85
Memorability	Speech	Multimodal	1.30	0.90
Learnability	Speech	Multimodal	0.66	0.75
* Mental effort	Multimodal	Speech	0.55	0.71
* Physical effort	Gestures	Speech	1.64	0.95
* Physical effort	Multimodal	Speech	1.32	0.91

* Measures marked with an asterisk indicate negative aspects, where lower scores are desirable.

post hoc analysis yielded a $p - value = .01$ for the differences between speech and gestures, and a $p - value < .001$ for the differences between multimodal and speech.

Learnability

The Kruskal-Wallis test for the differences between means in learnability yielded a $p - value = .04$, which is below the established alpha and thus meets the criteria for statistical significance. In a post hoc analysis, a $p - value = .04$ was obtained for the differences between multimodal and speech.

Mental Effort

The differences between mental effort expended per modality yielded a $p-value = .01$. A post hoc analysis showed that these differences were only significant between

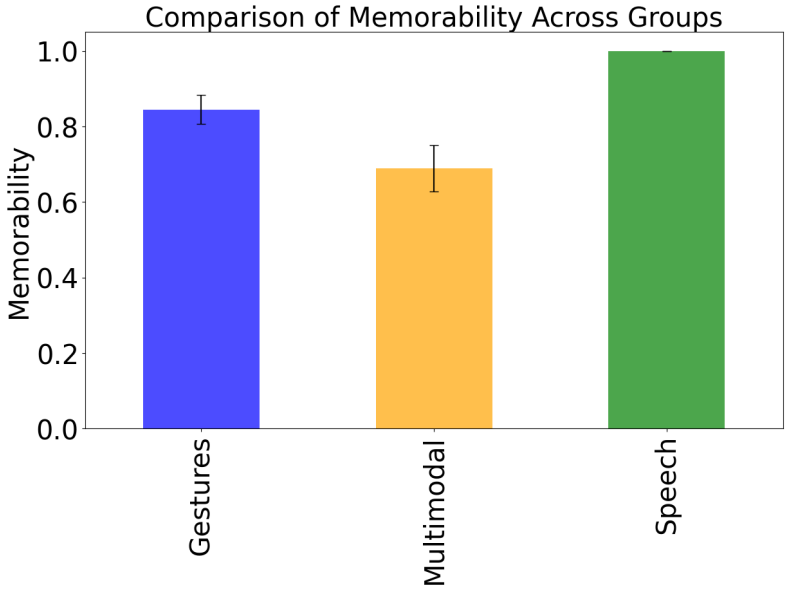


Figure 8.8. Comparison of memorability

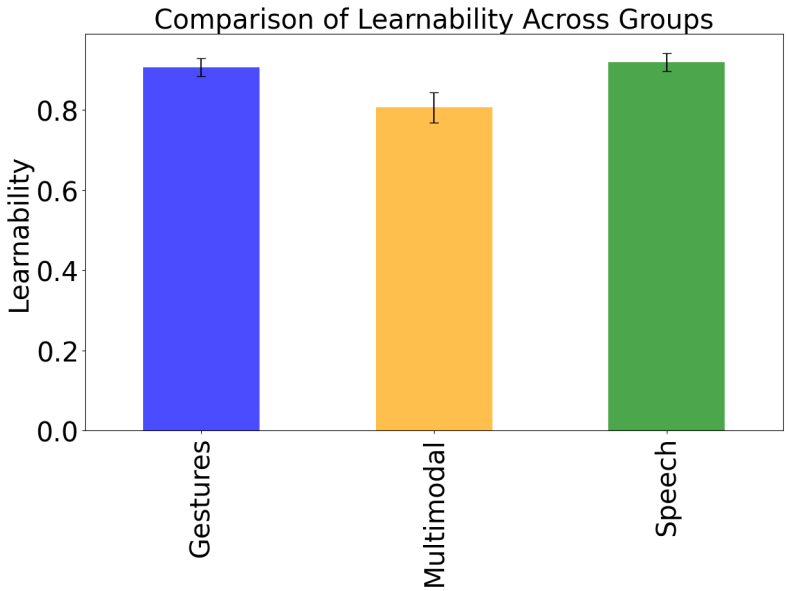


Figure 8.9. Comparison of learnability

multimodal and speech, with a $p - value = .01$.

Physical Effort

The differences in physical effort expanded on the task yielded a $p - value < .001$ and were thus statistically significant. After applying Dunn's test, the differences were only found to be significant between multimodal and speech ($p - value < .001$) and between gestures and speech ($p - value < .001$).

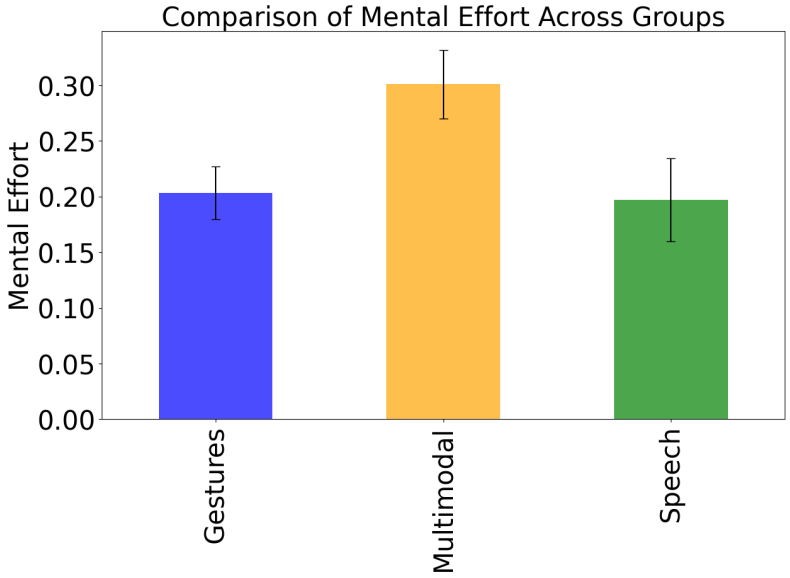


Figure 8.10. Comparison of mental effort expended on task

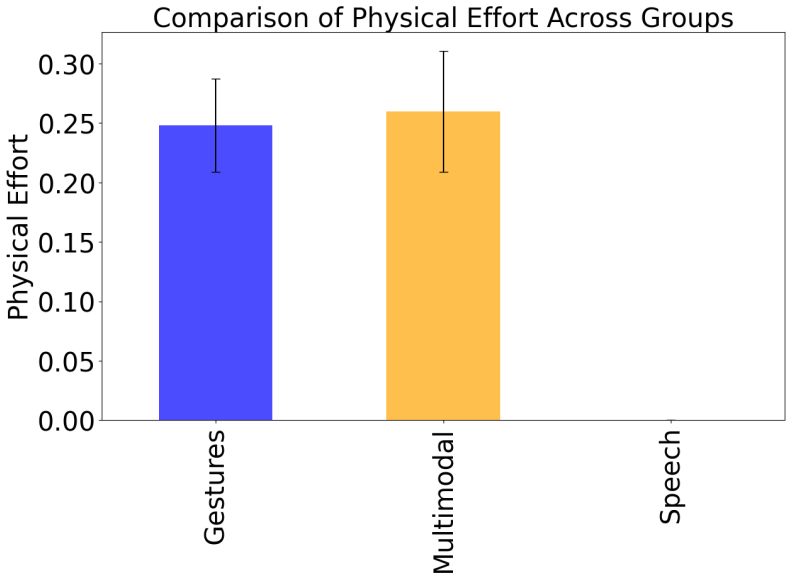


Figure 8.11. Comparison of physical effort expended on task

8.2.4 Discussion

Usability: This study evaluated the usability of three interaction modalities—speech, gestures, and multimodal—using the Five-Factor Model of Usability and the Alternative System Usability Scale [174, 8]. Although overall usability could not be analysed as a single measure, subcomponent analysis revealed important insights.

Memorability and Learnability: Speech interactions scored significantly higher in memorability compared to gestures and multimodal interaction and in learnability

compared to multimodal interactions. These findings align with the nexus between the two concepts, as memorability is a key component of learnability [62].

Relative User Efficiency: In line with speech's high scores in memorability and learnability, it exhibited lower relative user efficiency compared to gestures and multimodal interactions. This indicates that participants in the speech group performed more closely to expert users [89].

Mental and Physical Effort: Speech required significantly less mental effort than multimodal interactions and significantly less physical effort than both gestures and multimodal interactions, consistent with the initial expectations [6].

Trust and Perceived Social Attributes

Trust: Contrary to expectations, trust ratings did not favour speech, not differing significantly across any group. This discrepancy may be due to the use of a simulated robot, as VR provides a safer interaction environment that could affect perceived trust [150, 129]. Furthermore, as all participants managed to fulfill the task, it can be argued that the robot presented high perceived performance, one of the most influential factors of trust [104]. This can have contributed to the equality of the scores in trust.

Perceived Competence and Discomfort: These measures showed no significant differences across modalities. Previous research suggested higher perceived competence and lower perceived discomfort correlating with increased trust [105, 278], but these expectations were not met in this study.

8.2.5 Implications for LLM Integration

After the questionnaire, participants were asked which modality they would prefer if they had a choice. In the gestures group, 17 participants preferred having both speech and gestures, six were satisfied with gestures only, four preferred speech only, and three had other suggestions. In the multimodal group, 22 participants were satisfied with both modalities, two preferred gestures only, four preferred speech only, one preferred unimodality, and one suggested alternative controls. In the speech group, 13 participants desired multimodality, 13 were satisfied with speech only, one preferred gestures only, and three suggested a controller. Due to the between-subjects design, these claims do not offer grounds for a fair comparison but offer a basis for future studies.

Overall, the results obtained for gesture-only and multimodal modalities were similar; however, both showed significant differences with respect to speech, scoring lower in several usability subcomponents. Speech is the most used modality in interacting with smart/handheld devices such as Amazon Alexa, Apple Siri, Google Assistant, and recently ChatGPT voice input, etc [208]. Due to this, people may be biased in using speech as their interaction driver, which could have led to the results where it emerged as the primary modality in the case of subcomponents of

Table 8.6. Questionnaire

English	Italian	Measure
1. This system is efficient to use	1. Questo sistema è efficiente da usare	Efficiency
2. The steps needed to operate this system are easy to remember	2. I passi necessari per utilizzare questo sistema sono facili da ricordare	Memorability
3. This system is pleasant to use	3. Questo sistema è piacevole da usare	Satisfaction
4. This system was easy to learn for me	4. Questo sistema è stato facile da imparare per me	Learnability
5. When I make errors in this system, I cannot easily recover from them	5. Quando faccio errori in questo sistema, non posso facilmente correggerli	Error recovery
6. Based on the following scale (Figure 8.6), what was the mental effort required to perform the tasks?	6. Sulla base della seguente scala (Figure 8.6), qual era lo sforzo mentale richiesto per eseguire i compiti?	Mental effort
7. Based on the following scale (Figure 8.6), what was the physical effort required to perform the tasks?	7. Sulla base della seguente scala (Figure 8.6), qual era lo sforzo fisico richiesto per eseguire i compiti?	Physical effort
8. The robot is trustworthy	8. Il robot è affidabile	Trust
9. I can trust the information presented by this robot	9. Posso fidarmi delle informazioni presentate da questo robot	Trust
10. How well does the word '*Reliable*' describe the robot you just interacted with?	10. Quanto bene la parola *'Affidabile'* descrive il robot con cui hai appena interagito?	Competence
11. How well does the word '*Competent*' describe the robot you just interacted with?	11. Quanto bene la parola *'Competente'* descrive il robot con cui hai appena interagito?	Competence
12. How well does the word '*Knowledgeable*' describe the robot you just interacted with?	12. Quanto bene la parola *'Informato'* descrive il robot con cui hai appena interagito?	Competence
13. How well does the word '*Interactive*' describe the robot you just interacted with?	13. Quanto bene la parola *'Interattivo'* descrive il robot con cui hai appena interagito?	Competence
14. How well does the word '*Responsive*' describe the robot you just interacted with?	14. Quanto bene la parola *'Reattivo'* descrive il robot con cui hai appena interagito?	Competence
15. How well does the word '*Capable*' describe the robot you just interacted with?	15. Quanto bene la parola *'Capace'* descrive il robot con cui hai appena interagito?	Competence
16. How well does the word '*Scary*' describe the robot you just interacted with?	16. Quanto bene la parola *'Spaventoso'* descrive il robot con cui hai appena interagito?	Discomfort
17. How well does the word '*Dangerous*' describe the robot you just interacted with?	17. Quanto bene la parola *'Pericoloso'* descrive il robot con cui hai appena interagito?	Discomfort
18. How well does the word '*Aggressive*' describe the robot you just interacted with?	18. Quanto bene la parola *'Aggressivo'* descrive il robot con cui hai appena interagito?	Discomfort
19. How well does the word '*Strange*' describe the robot you just interacted with?	19. Quanto bene la parola *'Strano'* descrive il robot con cui hai appena interagito?	Discomfort
20. How well does the word '*Awful*' describe the robot you just interacted with?	20. Quanto bene la parola *'Terribile'* descrive il robot con cui hai appena interagito?	Discomfort
21. How well does the word '*Awkward*' describe the robot you just interacted with?	21. Quanto bene la parola *'Imbarazzante'* descrive il robot con cui hai appena interagito?	Discomfort

usability. While the study demonstrated the effectiveness of multimodal interaction, it also highlighted the limitations of rule-based decision-making. Current processing relies on predefined mappings between speech and gestures, limiting flexibility in ambiguous scenarios. To overcome this, LLM integration is proposed to enhance contextual understanding and resolve ambiguity. LLMs can infer intent from partial or noisy inputs, dynamically adjusting interactions based on conversation history and environmental cues.

8.3 LLMs to Extend the HRI capabilities

Large Language Models (LLMs) have revolutionised the field of artificial intelligence by providing sophisticated natural language understanding and reasoning capabilities. Their ability to process complex instructions, extract contextual meaning, and generate coherent responses makes them invaluable for Human-Robot Interaction (HRI). Their flexibility in handling diverse inputs allows them to support a range of robot applications, including navigation, manipulation, and decision-making.

In multimodal HRI (MHRI), LLMs facilitate seamless integration of multiple sensory inputs—such as speech, gestures, and environmental data—into a unified decision-making framework. This capability enhances robots' ability to understand commands, infer intent, and dynamically adapt to contextual variations. Additionally, LLMs enable:

- Context-Aware Interactions: By maintaining a history of user interactions, LLMs allow robots to understand past commands and refine responses based on situational context, thus improving user engagement and communication.
- Intent Disambiguation: Multimodal signals, such as voice and gestures, may be ambiguous in isolation. LLMs use probabilistic reasoning to infer intent from incomplete or conflicting inputs, reducing errors in command execution.
- Dynamic Task Adaptation: Unlike traditional rule-based systems, LLMs allow robots to modify tasks in real-time based on sensor feedback, ensuring greater flexibility in unpredictable environments where static programming would be insufficient.
- Improved User Experience: LLMs enable robots to engage in more natural conversations, providing human-like interaction patterns that increase user trust and satisfaction, allowing for deeper and more complex interactions.

Given these advantages, LLMs play a crucial role in enhancing robotic decision-making, allowing for more robust multimodal communication. The following sections focus on how LLaMA, a cutting-edge LLM, has been integrated into our HRI framework to optimise performance and adaptability across various robotic applications.

8.3.1 LLaMA as the LLM of Choice

LLaMA (Large Language Model Meta AI) [244] has been selected as the preferred LLM for integration within the multimodal HRI system. LLaMA provides an optimal

balance between computational efficiency and advanced language understanding, making it well-suited for real-time robotic applications. One of its primary advantages is its adaptability for edge computing, which is essential for robotics applications that require low-latency and high-performance processing. To accommodate real-world constraints, the **3B and 8B parameter models** of LLaMA are deployed on **NVIDIA Jetson Orin** [178], a high-performance edge AI computing platform optimised for power-efficient deep learning inference.

Comparison of LLaMA Versions

Experiments were conducted with multiple versions of LLaMA (3.0, 3.1, and 3.2) across different robot platforms. The models tested ranged from **3B to 11B parameters**, with both text-only and multimodal variants. Specifically:

- LLaMA 3.0 and 3.1 (8B, Text-Only) were used in indoor and outdoor logistics robots, offering strong language understanding and inference capabilities.
- LLaMA 3.2 (3B, Text-Only and 11B Vision-Enabled) were evaluated for various autonomous delivery applications, where multimodal processing was necessary.
- Performance Observation: LLaMA 3.1 (8B) outperformed 3.0 (8B) and 3.2 (3B) in logistics and delivery tasks due to its improved token handling and efficiency, ensuring better command interpretation and adaptability.
- Token Limitations: Despite the improvements in LLaMA 3.1, token length constraints still affected complex command sequences. Higher-parameter models, such as LLaMA 11B, showed promise but required extensive optimisation for edge deployment.

8.3.2 Multimodal Architecture with LLMs

The introduction of LLMs does not replace the existing multimodal architecture but enhances it by replacing certain libraries and modules with advancements. The Task Manager within the state machine architecture now acts as the primary execution unit, interpreting commands received via LLM processing and multimodal fusion.

Key enhancements include:

- **Natural Language Command Execution**: LLMs parse complex user instructions, breaking them down into task primitives handled by the state machine.
- **Error Recovery and Adaptation**: When unexpected issues arise, the LLM suggests corrective actions, modifying state transitions dynamically.
- **Personalised Interaction**: The system learns user preferences over multiple interactions, leading to a more intuitive HRI experience.

KEY FEATURES	LLAMA 3	LLAMA 3.1	LLAMA 3.2
Parameter Sizes	8B, 70B	8B, 70B, 405B	1B, 3B (text only) 11B, 90B (vision enabled)
Token Limit	Up to 2,048 tokens	Up to 128K tokens	Varies by model, optimized for real-time
Deployment	Cloud or on-premise	Cloud or specialized hardware	Cloud, edge, or mobile environments
Use Cases	Chatbots, content generation	Decision support, complex query resolution	AR/VR, mobile apps, interactive services
Text Generation	✓	✓	✓
Advanced Reasoning	✗	✓	✓
Extended Context	✗	✓	✓
Multimodal Reasoning	✗	✗	✓
Mobile Optimization	✗	✗	✓
Real-Time Interactions	✗	✗	✓

Figure 8.12. Comparing Critical Aspects of Llama Models across versions

Source: [159]

Integration of LLaMA with State Machine

A state machine is a computational model used to control execution flow in a structured and deterministic manner. It operates by transitioning between discrete states based on predefined rules, making it particularly suitable for handling robotic decision-making tasks. In the context of HRI, state machines provide a structured way to manage complex interactions by ensuring predictable and controlled responses to multimodal inputs. The state machine used in this architecture is implemented using the **PyTransitions** Python library [197]. PyTransitions enables event-driven state transitions and provides a lightweight but powerful framework for managing task execution. This allows for easy integration with multimodal inputs and efficient coordination of robotic actions.

Role of the State Machine in Task Handling

To enhance task execution and contextual understanding in HRI, we integrate LLMs within a state machine-based framework. This approach ensures structured, deterministic task execution while leveraging the adaptive reasoning capabilities of LLMs. The custom state machine is responsible for handling tasks and logging all conversations with the LLM, allowing traceability and refinement of interaction strategies over time. It also manages dialogues, task delegation, decision managing, and issue logging. Unlike traditional architectures where task management follows pre-defined rules, the integration of LLMs with a state machine allows for:

- **Dynamic Task Adjustments**: The system refines robot behaviour based on real-time feedback from multimodal sensors.

- **Logging and Analysis**: All user-robot interactions are logged, allowing iterative improvement and adaptation of commands.
- **Interrupt Handling**: High-priority interruptions (e.g., emergency stops) can override current tasks and trigger corrective actions.

8.3.3 Prompting for Multimodal HRI Integration

Integrating LLaMA into the multimodal HRI architecture to bridge the gap between human instructions and robotic actions requires careful prompt design to ensure the model's responses align with robotic execution tasks. These structured prompts allowed for seamless interaction with the multimodal state-machine framework, ensuring coherence between user commands and robotic execution, even in dynamically changing environments. LLM was provided with structured prompts that defined speech-act frames, enabling it to understand and process multimodal inputs such as speech and gestures. The state machine categorises these inputs into complementary (where speech and gestures provide partial but complete information) and redundant (where both modalities provide the same information for confirmation). By incorporating LLM into the decision-making loop, we improved the system's ability to process requests efficiently while reducing content hallucinations and errors. Example prompts were engineered to LLaMA (LLM) by realising two key components and the example conversation is provided:

```
Action Commands: "Move forward three meters and stop."

Clarification Requests: "What do you mean by ‘adjust position’?"

Contextual Queries: "Should I pick the closest accessible object
                        based on my last position?"

Multi-Modal Coordination: "Check for obstacles while moving
                            forward and alert me."
```

NLU functionality

The LLM was primed with structured prompts to interpret natural language commands into MHRI architecture-understandable intents. STT module integrated to MHRI pipeline transcribes speech to sentences and SRL module parses these sentences into actionable Frame Elements and Frame Arguments. Prompt examples on how to parse, what to parse from the sentences must be injected to State Machine that is handling NLU functions. For instance:

- **Example Prompt 1**:

  "Move the robot arm 30 degrees clockwise and grip the object"
  → Parsed as:
  {"action": "move_arm", "angle": 30, "direction": "clockwise", "grip": true}

- **Example Prompt 2**:

"Navigate to Room B and avoid obstacles"
→ Translated to navigation waypoints with obstacle-avoidance protocols.

Robot Hardware Realization

LLM outputs need to be linked to hardware APIs for physical execution. The prompt should explain robot hardware and the capabilities of the robot to not deviate from command maps to task executable actions. Examples prompts that were injected into LLaMA are given below.

- **Opening lid Example**:

  "Open lid when the gesture was received"
  → Triggers lid open code: publish("/Gesture", {"type": open_lid, "log": true})

- **Safety Constraint Prompt**:

  "If wheel speed exceeds 5Nm, stop immediately"
  → Embedded as: if (torque > 5) { emergency_stop(); }

Workflow Example

1. **User Input**: "Turn 90 degrees, scan the room, and return to base."
2. **LLM + NLU Parsing**:

```
Extract {action: str, angle: int, task: str, return_command: bool}
Output: {"action": "turn", "angle": 90, "task": "scan",
                "return_command": true}
```

3. **Hardware Execution**:
   - Invoke path-planning API for `turn(90)`
   - Activate LiDAR via `scan()`
   - Trigger `return_to_base()` command

Logistic delivery Robot's Prompt

The following prompt was introduced to realise the robot's hardware capacities, available functions and executable actions.

```
You are a logistic robot consisting of a wheeled platform, a storage
space attached to the wheeled platform, a lid on the storage space,
and 4 cameras. You are capable of performing the following operations:

"move_to_marker",
"open_lid",
"close_lid",
```

```
"start_path_remember",
"finish_path_remember",
"execute_path_trajectory",
"follow_me",
"stop",
"display_battery".

The command "move_to_marker" is to control AGV (wheeled platform).
The following destination points (markers) are available:
MEMORY_MARKERS.

The command "execute_arm_trajectory" starts a blind trajectory to
perform by the arm. The following trajectories (trajectory) are
available: MEMORY_ARM.

- You will maintain a simple and brief dialogue with the user.
- If the user asks you to perform an action, respond briefly
and output the corresponding JSON action.
- If no action is required, respond with a simple, brief sentence
and output {"action": null}.

Here are a few examples to follow:

Example 1:
{"role":"User","text":"could you come near me"}
{"role":"Robot","text":"Sure!","action"
    {"op":"move_to_marker","marker":"person","type":"agv"}}
{"role":"User","text":"Thanks!"}
{"role":"Robot":"text":"No worries","action":null}

Example 2:
{"role":"User","text":"open the lid"}
{"role":"Robot","text":"Of course","action":
    {"op":"open_lid","type":"lid"}}

Example 3:
{"role":"User","text":"close the lid"}
{"role":"Robot","text":"Of course","action":
    {"op":"close_lid","type":"lid"}}

Example 4:
{"role":"User","text":"Let me show you the path to  go"}
{"role":"Robot","text":"Ok, how do I name this action?","action":null}
{"role":"User","text":"path_1"}
{"role":"Robot","text":"thanks, I'll remember this","action":
    {"op":"start_path_remember","name":"path_1","type":"sege"}}
{"role":"User","text":"this is the whole path, let's remember that"}
```

```
{"role":"Robot","text":"got it!","action":
    {"op":"finish_path_remember","type":"sege"}}

Example 5:
{"role":"User","text":"Go on the path"}
{"role":"Robot","text":"Ok!","action":
    {"op":"execute_path_trajectory","name":"path_1","type":"sege"}}
{"role":"User","text":"Move along the path"}
{"role":"Robot","text":"Ok!","action":
    {"op":"execute_path_trajectory","name":"path_1","type":"sege"}}

"Let me show how to grab this" = "I'll show the trajectory for the arm"
    = "Remember how to grasp this object" and so on.

No more examples. Avoid hallucinating.

Now, respond to the following input with the appropriate JSON:
```

8.3.4 Deployment of LLM on Jetson Orin

The integration of LLaMA into Jetson Orin involves several optimisations to ensure efficient performance and adaptability in robotic tasks:

Model Optimization:

- Quantization for Reduced Computation: The LLaMA models are quantised to lower precision (e.g., INT8) using NVIDIA TensorRT to optimise inference without significant accuracy degradation [285]. This ensures fast response times even under computational constraints.
- Efficient Memory Management: The **3B model** is deployed for lower-latency tasks, while the **8B model** is used for more complex contextual reasoning, ensuring computational efficiency across different robotic applications [180].

Inference Acceleration:

- Parallel Processing: By leveraging the **multi-core GPU architecture** of NVIDIA Jetson Orin, the system can process multimodal inputs concurrently, reducing response times [179] and enabling real-time command execution. The model is optimized using NVIDIA's TensorRT inference framework to achieve up to 4x speed improvements in latency-sensitive applications [182].

Power-Efficient Execution:

- On-Device Processing: Deploying LLaMA directly on Jetson Orin eliminates cloud dependency, ensuring low-latency responses, data security, and robust autonomous operations even in network-limited environments.

By integrating LLMs with a state-machine framework into MHRI pipeline ensured robust, adaptable, and scalable multimodal interactions. This approach has been successfully applied to CANOPIES agricultural robots and particularly in logistics

and autonomous delivery robot Segway Outdoor E1 delivery robots, demonstrating its versatility across different domains. The ability to run LLaMA efficiently on the edge has also opened possibilities for deployment in robotic arms, warehouse automation, and autonomous mobile robots (AMRs). Future advancements will further refine the synergy between deterministic state control and adaptive language-based reasoning for even more intuitive and efficient HRI.

Performance Evaluation

The system was tested under various conditions, including indoor and outdoor environments, varying network latencies, and multi-agent coordination scenarios. Key performance indicators included:

Processing latency across different LLaMA versions: The inference speed of various LLaMA versions was measured to determine their suitability for real-time applications. Tests revealed that LLaMA 3.1 (8 B) with TensorRT optimization achieved up to 3.2x faster inference times compared to its non-optimized counterpart. In contrast, higher-parameter models such as LLaMA 11 B exhibited significant delays, making them less feasible for edge deployment.

Response accuracy in multi-modal command execution: The model's ability to correctly interpret and execute multimodal commands was evaluated through a series of real-world test scenarios. LLaMA 3.1 (8B) achieved an 89% success rate in accurately processing and executing commands, whereas LLaMA 3.0 (8B) had a 78% success rate, indicating significant improvements in contextual reasoning with the newer version. For example, in a logistics setting, robots were given the instruction.

```
"Pick up the blue box, move forward two meters, and place it
on the shelf."
```

Energy consumption for edge inference tasks: Power efficiency is crucial for robots operating on battery constraints. The evaluation measured energy draw across multiple deployments. LLaMA 3.1 (8B) consumed 12W per inference cycle, while LLaMA 3.2 (3B) demonstrated improved efficiency at 8W per cycle. These results highlight the trade-off between model size and energy efficiency, reinforcing the need for optimized deployments on embedded systems like Jetson Orin.

8.3.5 Open Challenges

Despite the advantages of LLMs in MHRI, a few challenges remain:

- **Edge Deployment Scaling**: Higher-parameter models still require significant optimisations to run efficiently on Jetson Orin. While quantization and pruning techniques mitigate this, achieving near-cloud performance on edge devices remains a key challenge. For instance, **running an 8B LLaMA model on Jetson Orin with TensorRT optimizations** still requires balancing precision and latency. Deploying a full-fledged model without optimization may cause inference lag, making real-time applications like **autonomous navigation in warehouses** impractical.

- **Token Management**: Managing token limits for long-duration interactions remains a challenge for handling complex tasks. The token constraints impact how much context the model can retain over extended conversations, potentially leading to inconsistencies in responses. For example, a delivery robot assisting with multi-step instructions such as:

  ```
  "Go to the storage unit, pick up the package labelled 'Fragile',
  take it to the customer, and confirm delivery."
  ```

  may forget earlier steps due to token limitations, resulting in *missed or incomplete actions*. This necessitates task segmentation and effective state memory management.

- **Human-Robot Trust**: The adaptability of LLaMA needs further refinement to ensure predictable and transparent decision-making in collaborative environments. Unexpected outputs from LLMs, such as hallucinated information or inconsistent task execution, can erode trust in human-robot interactions. For example, *if an industrial robot misinterprets "slowly adjust" as "increase speed," the consequences could be dangerous*. Ensuring *explainability in decision-making* and incorporating verification checkpoints can enhance trust.

- **Temporal Reasoning**: Handling latency constraints is crucial, particularly when executing time-sensitive commands. For example, an instruction such as:

  ```
  "Wait 5 seconds after moving before gripping"
  ```

  requires the state machine to integrate explicit delays while maintaining execution flow without blocking other processes. In **assembly-line robotics**, improper delay handling may result in components being processed too soon or too late, leading to *misalignments or defective products*.

- **Error Recovery**: Designing fallback protocols for unexpected edge cases is essential. For instance:

  ```
  "If slip detected, retry grip with 20% more force"
  ```

  requires the system to detect failure conditions dynamically and adapt execution strategies accordingly. Consider a *robotic arm in a logistics facility gripping different package sizes*; if a slip is detected while lifting a fragile item, the system should automatically adjust force instead of dropping or crushing the package.

- **Ethical Guardrails**: Ensuring compliance with safety standards through context-aware filtering and responsible AI implementation is crucial. Safety-first prompts include:

  ```
  "Do not exceed speed limits"
  ```

  and similar pre-programmed constraints that prevent unsafe behaviours in robots deployed in real-world environments. For example, *an autonomous vehicle robot should never override a speed cap in a pedestrian zone, even if*

prompted to do so. Ethical AI design must account for regulatory compliance and safe decision-making.

- **Multimodal Fusion Optimization**: The integration of speech, vision, and gestures through LLMs can create conflicts where multiple modalities provide overlapping or contradictory information. Refining multimodal fusion algorithms is necessary to improve consistency in responses. For instance, *if a user says "pick the bottle" while pointing to two bottles*, the model must intelligently disambiguate the reference. Implementing **cross-modal verification** can mitigate such conflicts.

- **Computational Overhead**: Despite optimizations, running high-parameter models like LLaMA 8 B in real-time requires significant computational resources. Reducing latency while maintaining model accuracy remains a core challenge in embedded AI systems. For example, *an agricultural robot detecting fruit ripeness may require processing of several image frames per second while responding to verbal commands.* Managing computational load without sacrificing responsiveness is an ongoing optimization concern.

Chapter 9

Conclusions and Future Directions

This dissertation has explored the intersection of **Human-Robot Interaction (HRI)**, **gesture-based communication**, and the integration of **Large Language Models (LLMs)**. Through empirical studies and computational approaches, the research has advanced the understanding of how robots can effectively interpret and generate gestures and language in human-centric environments. The findings contribute to the broader field of HRI by providing novel insights into multimodal communication, real-time gesture recognition, and context-aware robot responses.

The research has demonstrated that leveraging **LLMs** enhances the adaptability of robots in interpreting ambiguous human gestures, improving interaction fluidity. The integration of machine learning techniques for gesture recognition has enabled robots to respond in a more naturalistic and context-aware manner. Moreover, this work has addressed challenges related to real-time processing, multimodal fusion, and the alignment of gesture-based intent with natural language understanding, ensuring more effective human-robot collaboration. Additionally, the research has provided a structured analysis of how contextual reasoning, sensory data, and human feedback loops can be effectively utilized to improve interaction efficiency.

9.1 Summary of Contributions

- **Multimodal Communication Architecture**: Developed a decision-fusion-based architecture that integrates **speech**, **gesture**, and **visual cues**, allowing robots to effectively process and respond to user commands in real-time, even in complex, multi-human, multi-robot collaborative environments.
- **Enhanced Gesture Recognition**: Real-time gesture recognition systems were improved, tested, and deployed in both indoor and outdoor environments, enhancing the robot's ability to detect and respond to gestures in dynamic settings.
- **LLM Integration**: Integrated **Large Language Models (LLMs)** into the HRI framework to provide deeper context-awareness and improve the understanding of complex, ambiguous user commands in collaborative settings.

- **State Machine-Based Task Management**: Incorporated **State Machines** for efficient task management and decision-making, ensuring that the robot could handle multiple requests and tasks simultaneously in dynamic environments.

- **Real-Time Deployment on Edge Devices**: Deployed the multimodal HRI solution on **Jetson Orin** edge devices, enabling real-time, low-latency task execution in logistics robots, ensuring the practical applicability of the developed system.

- **VR Testing and Simulation**: Validated and tested the framework in **Virtual Reality (VR)** simulations, providing a safe and controlled environment for refining and assessing the system before real-world deployment in collaborative settings.

9.2 Limitations and Open Challenges

- **Real-time Processing Bottlenecks**: While gesture recognition and LLM integration have been optimized, real-time interaction remains computationally intensive, limiting deployment in resource-constrained environments. Future optimizations are needed to enhance processing efficiency without compromising accuracy.

- **Contextual Understanding Constraints**: Although LLMs improve response generation, their ability to fully interpret nuanced human intent remains imperfect, particularly in complex or ambiguous interactions. This limitation suggests a need for more sophisticated intent recognition mechanisms.

- **Hardware and Sensory Limitations**: The accuracy of gesture recognition is dependent on sensor quality and environmental conditions, such as lighting and occlusions, which can affect robustness. Future research should explore hybrid sensing approaches to mitigate these limitations.

- **Ethical and Trust Considerations**: Ensuring that robots do not reinforce biases in interpreting gestures and maintaining transparency in decision-making processes remains an open research challenge. More extensive regulatory and ethical guidelines must be established to address these concerns effectively.

9.3 Future Research Directions

While this work has achieved significant progress, there are still several areas to explore for further refinement and broader application to push on the boundaries of existing Multimodal Human-Robot Interaction:

- **Expanding Cross-Domain Applications**: The current system has been primarily tested in **agriculture** and **logistics** environments. Future research could focus on adapting the system for other domains such as **healthcare**,

elder care, and **service robotics**. These fields present unique challenges, including human-robot emotional interaction and autonomy in dynamic settings, which could benefit from the advancements in multimodal communication.

- **Human-Robot Collaboration in Dynamic Multi-Robot Environments**: Although this work has been tested in multi-human, multi-robot settings, further research can investigate more complex environments where robots must collaborate on a larger scale. This includes developing better **coordination algorithms** to manage task allocation, conflict resolution, and optimal resource usage across multiple robots in large-scale collaborative workspaces.

- **Autonomous Learning and Adaptation**: While LLMs and task management via State Machines have significantly enhanced system performance, future research could incorporate **autonomous learning** mechanisms. This would allow robots to continually learn from interactions and refine their behaviour and task management strategies based on feedback and evolving environments.

- **Real-Time Context-Awareness in Highly Dynamic Environments**: Despite the deployment on edge devices like **Jetson Orin**, challenges related to real-time processing in highly dynamic environments remain. Future work could explore how to further **optimise edge computing**, making the system more scalable, energy-efficient, and capable of handling more complex real-time processing tasks, especially in areas with limited connectivity.

- **Improving Robustness in Adverse Conditions**: The system has been tested in a variety of real-world settings, but future studies should focus on improving the robustness of the multimodal system in extreme conditions, such as **low-light environments** or **high-noise areas**. This includes enhancing the **sensor fusion algorithms** to maintain high accuracy in these challenging conditions.

- **Expanding Multimodal Interaction Capabilities**: Further work can explore more advanced **multimodal fusion techniques** that incorporate additional sensory data, such as **tactile feedback** or **context-aware vision systems**, enabling even more dynamic interaction between robots and humans in collaborative spaces.

- **User-Centered Design for Personalisation**: The multimodal framework could be further personalised to individual users through **human-centered design principles**. This includes adapting the robot's behaviour and communication style based on user preferences, learning from past interactions, and improving the efficiency of collaboration by adjusting to user-specific communication patterns.

- **Long-Term User Studies**: Long-term user studies in real-world environments will provide deeper insights into how robots and humans can work together more effectively over time. These studies can help identify areas for improving the **user interface**, **robot autonomy**, and overall **collaboration quality**.

9.4 Final Thoughts

The advancements presented in this thesis push the boundaries of multimodal Human-Robot Interaction, particularly in collaborative and dynamic environments. By integrating decision-fusion-based architectures and LLMs, this research enhances the ability of robots to understand and respond to complex human commands, paving the way for more natural and efficient collaborations. Through continued development and exploration, these systems hold immense potential to transform industries ranging from agriculture and logistics to healthcare and home assistance, ultimately enhancing the quality and efficiency of human-robot collaboration in various domains.

Robotic Platforms and Hardware Configuration Details

In the *CANOPIES* project, two primary robotic prototypes have been developed: a *farming robot* and a *logistic robot*. Both platforms share a common mobile base and certain sensing elements, yet they differ in their end-effectors and task-specific mechanisms. These two robots were designed by the combined effort of DTI [1], RomaTre [2], UPC [3], Sapienza [4] who were part of CANOPIES consortium [5].

In addition, a separate delivery-oriented platform (the *Segway Robotics E1* designed by Segway Robotics [6]) from RemBrain [7] company has been employed for research in HRI to address outdoor and indoor logistics. This section outlines the key specifications, mechanical considerations, and methodological rationale for selecting these hardware configurations, as well as the process of replicating the robots as digital twins for simulation and testing.

.0.1 CANOPIES Farming and Logistic Robots

The *farming robot* combines an industrial-grade mobile base with a dual-arm upper-body assembly and agronomic end-effectors. The prototype is designed to undertake tasks such as grape harvesting and pruning in challenging vineyard environments. The *logistic robot* shares the same underlying mobile platform but includes a box-exchange mechanism (BEM) to support the collaborative transport of harvested produce. A summary of the key hardware components and constraints follows:

- **Mobile Base:** Both prototypes use the Alitrak DCT-350P platform, noted for its four 12 V batteries in series (48 V nominal) and independent traction system [134]. The base is outfitted with:
    - *Sensor suite for navigation*, including an RTK-GNSS receiver (Septentrio AsteRx-m3 Pro+ for the rover and AsteRx-U for the base station), a high-grade IMU (SBG Ellipse-E), and two 3D LiDARs (Ouster OS1–64) mounted diagonally for full 360° coverage.

[1] https://www.dti.dk/
[2] https://www.uniroma3.it/
[3] https://www.upc.edu/en
[4] https://www.uniroma1.it/en/
[5] https://canopies.inf.uniroma3.it/consortium
[6] https://robotics.segway.com
[7] https://rembrain.ai/

- *Power management* featuring DC–DC converters and on-board distribution blocks. This arrangement provides stable 24 V outputs for sensors and computing units.
- *Networking*, realised through industrial-grade Gigabit Ethernet switches and WiFi routers to handle large data streams (e.g. LiDAR scans, camera images) and multi-robot connectivity.
- *Safety equipment*, including on-board and wireless emergency stops, redundant communication protocols, and a heartbeat mechanism between the base and the wireless controller.

- **Farming Robot Configuration:** A dual-arm torso, adapted from PAL Robotics' *TIAGo++*, is mounted on the rear section of the mobile base. Each arm provides 7 degrees of freedom, while the torso adds a prismatic lift and a rotational joint to extend the workspace. The integrated end-effectors (based on Robotiq grippers) are re-designed with modular fingers for pruning and harvesting tasks. On-board batteries and power boards are placed beneath the torso, rendering the upper body *self-contained* and ensuring sufficient reach to grape canopies and supply boxes [134].

- **Logistic Robot Configuration:** The box-exchange mechanism (BEM) occupies the central area of the base, enabling the robot to transport grape-filled crates away from the farming robot. Two primary conceptual designs have been examined: a conveyor-belt-based BEM with mechanical alignment features, and an articulated conveyor design allowing limited rotational degrees of freedom. These prototypes target minimal operator intervention, robust ingress protection (IP54 or higher), and precise alignment tolerances (horizontal errors of ± 100 mm and yaw errors of up to $\pm 20°$) [134].

.0.2 Digital Twins and Simulation Methodology

Both the farming and logistic robots have been replicated in simulation as digital twins to accelerate algorithmic development (explained in Chapter 4). The digital twin models incorporate:

- *Accurate geometry and inertial parameters,* ensuring that the arm kinematics and chassis dynamics mirror the real platforms as closely as possible.
- *Sensor streams,* including virtual LiDAR, GNSS signals, and IMU readings, thereby facilitating high-fidelity experiments in navigation and collaborative manipulation.
- *Software-in-the-loop (SIL) and hardware-in-the-loop (HIL) frameworks*, in which ROS-based controllers and planners run against simulated worlds to verify performance prior to field deployment.

This approach permits early detection of design flaws, more efficient prototyping of multi-robot interaction (especially for the BEM), and expedited integration of new software modules [134].

Figure .1. CANOPIES Farming Robot [134]

.0.3 Design Constraints and Rationale

The aforementioned choices reflect methodological constraints and academic goals:

1. **Modular Design Philosophy:** Each robot shares a standardised mobile base, creating a consistent foundational platform for sensor fusion, control, and communication modules.

2. **Agricultural Task Requirements:** The farming robot specifically addresses dual-arm manipulation of crops in uneven terrain. Reachability analyses, power autonomy, and safety overrides constitute primary design metrics.

3. **Multi-Robot Collaboration:** Both prototypes contribute to cooperative harvesting logistics, necessitating consistent mechanical interfaces and alignment tolerances in the BEM design.

Figure .2. CANOPIES Farming Robot (Left) and Logistic Robot (Right) in Box-Exchange Mechanism Configuration [134]

4. **Adaptive Simulation Environment:** Virtual models and real-time sensor emulation expedite iterative improvements, risk mitigation, and validation of integrated hardware-software pipelines.

.0.4 Summary of Key Performance Indicators

The *CANOPIES* consortium monitors hardware and system effectiveness through a set of Key Performance Indicators (KPIs). These range from navigation accuracy (GNSS interference, LiDAR coverage) to payload capacity and battery autonomy for each subsystem. Safety considerations remain integral, manifested in robust emergency-stop strategies and redundancy. As development progresses, KPI targets were refined iteratively during field tests.

.0.5 Logistic Delivery Robot (Segway E1)

The Segway system, designed for both tele-operation and autonomous driving, integrates a scalable solution with an optimized sensor suite and advanced computing platform. Equipped with multi-sensor fusion and cutting-edge perception technologies, including high-resolution cameras with high dynamic range (HDR) and mmWave radar, the system ensures exceptional performance even in extreme weather conditions. It is tailored for operations in real-world urban environments, offering a

user experience as smooth as riding on a Segway.

In addition to this, the Segway Robotics E1 platform has been employed in separate logistics trials as a complementary solution. This battery-powered, outdoor delivery robot showcases autonomous navigation capabilities and a durable chassis capable of operating both indoors and outdoors. Although the E1 differs from the CANOPIES prototypes in terms of chassis geometry, payload capacity, and top speed, it serves as a useful benchmark for evaluating sensor configurations, robustness to environmental factors, and user interface modules across diverse settings. The E1's key specifications include a net weight of approximately 62 kg, a maximum payload of 20 kg, and overall dimensions of $890 \times 615 \times 1148$ mm, along with integrated suspension and obstacle-crossing capabilities. These features underscore the E1's suitability for open-field environments, warehouses, and semi-structured terrains.

The Segway system, like the E1, is compatible with the Nvidia Jetson platform and supports an open architecture for enhanced flexibility, enabling collaboration within the robotics community. Furthermore, it is ROS (Robot Operating System) compatible, ensuring adaptability across a wide range of applications.

Figure .3. Indoor and Outdoor Logistic Delivery Robot

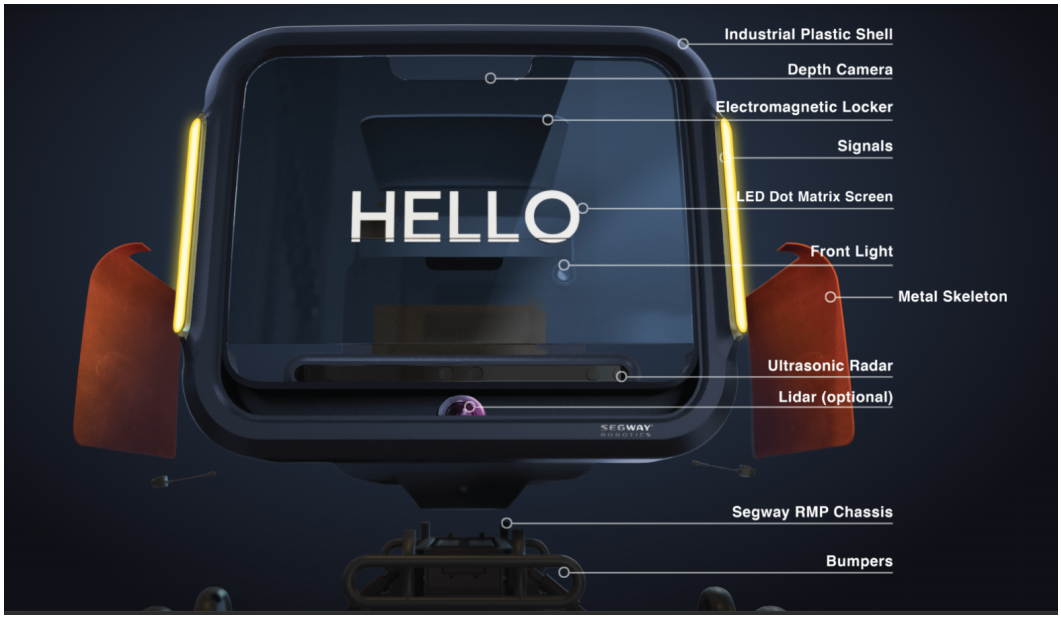


Figure .4. Segway E1 delivery robot sensors

Bibliography

[1] CORDIS | European Commission — cordis.europa.eu. https://cordis.europa.eu/project/id/101016906, [Accessed 30-Sep-2022]

[2] Gestures final commands designed for general ugv's and canopies. https://docs.google.com/document/d/14sFGAz8Crs7IOmlqpt4JgIjap34I_lDEtv0X3ufroDE/edit?usp=sharing, [Accessed 30-Sep-2022]

[3] European Union's Horizon 2020 research and innovation programme under grant agreement No 101016906: CANOPIES: A collaborative paradigm for human workers and multi-robot teams in precision agriculture systems. https://canopies.inf.uniroma3.it/ (March 2021), CANOPIES: A Collaborative Paradigm for Human Workers and Multi-Robot Teams in Precision Agriculture Systems.

[4] A. Vanzo, E.B., Lemon, O.: Hierarchical multi-task natural language understanding for cross-domain conversational ai: Hermit nlu. (2019)

[5] Abdi, H., Williams, L.J.: Tukey's honestly significant difference (hsd) test. Encyclopedia of Research Design **3**, 1–5 (2010)

[6] Abich, J., Barber, D.J.: The impact of human-robot multimodal communication on mental workload, usability preference and expectations of robot behavior. Journal of Multimodal User Interfaces **11**, 211–225 (2017). https://doi.org/10.1007/s12193-016-0237-4, http://dx.doi.org/10.1007/s12193-016-0237-4

[7] Abrams, A.M.H., von der Pütten, A.M.R.: I–c–e framework: Concepts for group dynamics research in human-robot interaction. International Journal of Social Robotics **12**, 1213 – 1229 (2019), https://api.semanticscholar.org/CorpusID:216410788

[8] Adhikari, P.: Construction and validation of alternative usability scale. Tribhuvan University Journal **38**(2), 83–92 (Dec 2023). https://doi.org/10.3126/tuj.v38i2.60770, http://dx.doi.org/10.3126/tuj.v38i2.60770

[9] Afzal, A., Katz, D.S., Goues, C.L., Timperley, C.S.: A study on the challenges of using robotics simulators for testing (2020), https://arxiv.org/abs/2004.07368

[10] Ahad, N.A., Yin, T.S., Othman, A.R., Yaacob, C.R.: Sensitivity of normality tests to non-normal data. Sains Malaysiana **40**, 637–641 (2011)

[11] Alonso Martin, F., Malfaz, M., Castro-González, , Castillo, J.C., Salichs, M.: Four-features evaluation of text to speech systems for three social robots. Electronics **9**(2) (2020). https://doi.org/10.3390/electronics9020267, `https://www.mdpi.com/2079-9292/9/2/267`

[12] Alpha Cephei Inc.: Vosk Speech Recognition Toolkit. `https://github.com/alphacep/vosk-api` (2021), accessed: February 17, 2023

[13] Andriluka, M., Iqbal, U., Insafutdinov, E., Pishchulin, L., Milan, A., Gall, J., Schiele, B.: Posetrack: A benchmark for human pose estimation and tracking. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR). pp. 6568–6577 (2018). https://doi.org/10.1109/CVPR.2018.00686

[14] Andriluka, M., Pishchulin, L., Gehler, P., Schiele, B.: 2d human pose estimation: New benchmark and model. Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR) pp. 1588–1595 (2014). https://doi.org/10.1109/CVPR.2014.206

[15] Anvari, T., Park, K.: 3d human body pose estimation in virtual reality: A survey. In: 2022 13th International Conference on Information and Communication Technology Convergence (ICTC). pp. 624–628 (2022)

[16] Arntz, A., Eimler, S.C.: Experiencing ai in vr: A qualitative study on designing a human-machine collaboration scenario. In: Stephanidis, C., Antona, M., Ntoa, S. (eds.) HCI International 2020 – Late Breaking Posters. pp. 299–307. Springer International Publishing, Cham (2020)

[17] Aschenbrenner, D., van Tol, D., Rusak, Z., Werker, C.: Using virtual reality for scenario-based responsible research and innovation approach for human robot co-production. In: 2020 IEEE International Conference on Artificial Intelligence and Virtual Reality (AIVR). pp. 146–150 (2020)

[18] Austin, J.L.: How to Do Things with Words. Oxford University Press, Oxford (1962)

[19] Autodesk Inc.: Autodesk® character generator. `https://charactergenerator.autodesk.com/` (March 2024), autodesk® Character Generator offers artists a web-based laboratory to create fully-rigged 3D characters for animation packages and game engines.

[20] Babel, F., Kraus, J., Miller, L., Kraus, M., Wagner, N., Minker, W., Baumann, M.: Small talk with a robot? the impact of dialog content, talk initiative and gaze behavior of a social robot on trust, acceptance and proximity. International Journal of Social Robotics **13**, 1485–1498 (2021). https://doi.org/10.1007/s12369-020-00730-0

[21] Bac, C.W., Henten, E.J., Hemming, J., Edan, Y.: Harvesting robots for high-value crops: State-of-the-art review and challenges ahead. J. Field Robot. **31**(6), 888–911 (Nov 2014). https://doi.org/10.1002/rob.21525, https://doi.org/10.1002/rob.21525

[22] Bacciu, A.: Semantic role labeling using gcn, bert and biaffine attention layer. https://github.com/andreabac3/NLP-Semantic-Role-Labeling. (2022)

[23] Bagheri, R.: Virtual reality: The real life consequences. UC Davis Business Law Journal 17 pp. 101–120 (2016)

[24] Baker, A., Phillips, E., Ullman, D., Keebler, J.R.: Toward an understanding of trust repair in human-robot interaction: Current research and future directions. ACM Trans. Interact. Intell. Syst. **8**(4) (2018). https://doi.org/10.1145/3181671

[25] Baker, C.F., Fillmore, C.J., Lowe, J.B.: The Berkeley FrameNet project. In: 36th Annual Meeting of the Association for Computational Linguistics and 17th International Conference on Computational Linguistics, Volume 1. pp. 86–90". Association for Computational Linguistics, Montreal, Quebec, Canada (Aug 1998). https://doi.org/10.3115/980845.980860, https://aclanthology.org/P98-1013/

[26] Baltrušaitis, T., Ahuja, P., Morency, L.P.: Multimodal machine learning: A survey and taxonomy. IEEE Transactions on Pattern Analysis and Machine Intelligence **41**(2), 423–443 (2018)

[27] Baraka, K., Alves-Oliveira, P., Ribeiro, T.: An extended framework for characterizing social robots pp. 21–64 (2019). https://doi.org/10.1007/978-3-030-42307-0_2

[28] Bartneck, C., Belpaeme, T., Eyssel, F., Kanda, T., Keijsers, M., Šabanović, S.: What is human–robot interaction? In: Human-Robot Interaction, pp. 6–17. Cambridge University Press (feb 2020). https://doi.org/10.1017/9781108676649.002, https://doi.org/10.1017%2F9781108676649.002

[29] Bazarevsky, V., Grishchenko, I., Raveendran, K., Zhu, T., Zhang, F., Grundmann, M.: Blazepose: On-device real-time body pose tracking (2020). https://doi.org/10.48550/ARXIV.2006.10204, https://arxiv.org/abs/2006.10204

[30] Bechar, A., Vigneault, C.: Agricultural robots for field operations: Concepts and components. Biosystems Engineering **149**, 94–111 (2016). https://doi.org/https://doi.org/10.1016/j.biosystemseng.2016.06.014, https://www.sciencedirect.com/science/article/pii/S1537511015301914

[31] Belke, M., Blanke, P., Storms, S., Herfs, W.: Object pose estimation in industrial environments using a synthetic data generation pipeline. In: 2022 Sixth IEEE International Conference on Robotic Computing (IRC). pp. 435–438 (2022). https://doi.org/10.1109/IRC55401.2022.00084

[32] Bellettiere, J., Hughes, S., Liles, S., Boman-Davis, M., Klepeis, N., Blumberg, E., Mills, J., Berardi, V., Obayashi, S., Allen, T., Hovell, M.: Developing and selecting auditory warnings for a real-time behavioral intervention. American Journal of Public Health Research **2**, 232–238 (1 2014)

[33] Bland, J.M., Altman, D.: Statistics notes: Cronbach's alpha. BMJ **314** (1997). https://doi.org/10.1136/bmj.314.7080.572

[34] Bommasani, R., et al.: On the opportunities and risks of foundation models. arXiv preprint arXiv:2108.07258 (2021)

[35] Bowman, D.A., Kruijff, E., LaViola Jr, J.J., Poupyrev, I.: Virtual environments: concepts and technologies. CRC Press (2004)

[36] Breazeal, C.: Toward sociable robots. Robotics and autonomous systems **42**(3-4), 167–175 (2003)

[37] Breazeal, C.: Social interactions in hri: The robot view. IEEE Transactions on Systems, Man, and Cybernetics (2004)

[38] Briggs, G., Williams, T., Scheutz, M.: Enabling robots to understand indirect speech acts in task-based interactions. J. Hum.-Robot Interact. **6**(1), 64–94 (may 2017)

[39] Brooks, R.: A robust layered control system for a mobile robot. IEEE Journal on Robotics and Automation (1986)

[40] Brown, T., Mann, B., Ryder, N., et al.: Language models are few-shot learners. Advances in Neural Information Processing Systems (NeurIPS) **33**, 1877–1901 (2020)

[41] Budzianowski, P., Wen, T.H., Tseng, B.H., Casanueva, I., Ultes, S., Ramadan, O., Gašić, M.: MultiWOZ - a large-scale multi-domain Wizard-of-Oz dataset for task-oriented dialogue modelling. In: Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing. pp. 5016–5026. Association for Computational Linguistics, Brussels, Belgium (Oct-Nov 2018), `https://aclanthology.org/D18-1547`

[42] Burger, B., Ferrane, I., Lerasle, F., Infantes, G.: Two-handed gesture recognition and fusion with speech to command a robot. Auton. Robots **32**, 129–147 (2012)

[43] Buxbaum, H., Kleutges, M., Sen, S.: Full-scope simulation of human-robot interaction in manufacturing systems. In: 2018 Winter Simulation Conference (WSC). pp. 3299–3307. IEEE (2018)

[44] Cao, Z., Simon, T., Wei, S., Sheikh, Y.: Realtime multi-person 2d pose estimation using part affinity fields. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR). pp. 1302–1310 (2017). https://doi.org/10.1109/CVPR.2017.139

[45] Cao, Z., Hidalgo, G., Simon, T., Wei, S.E., Sheikh, Y.: Openpose: Realtime multi-person 2d pose estimation using part affinity fields. IEEE Transactions on Pattern Analysis and Machine Intelligence **43**(1), 172–186 (2018). https://doi.org/10.1109/TPAMI.2019.2929257

[46] Carbone, C., Potena, C., Nardi, D.: Augmentation of sunflower-weed segmentation classification with unity generated imagery including near infrared sensor data. In: Simulation and Modeling Methodologies, Technologies and Applications: 10th International Conference, SIMULTECH 2020 Lieusaint-Paris, France, July 8-10, 2020 Revised Selected Papers 10. pp. 42–63. Springer (2022)

[47] Carbone, C., Potena, C., Nardi, D., et al.: Simulation of near infrared sensor in unity for plant-weed segmentation classification. In: 10th International Conference on Simulation and Modeling Methodologies, Technologies and Applications, SIMULTECH 2020. vol. 1, pp. 81–90. SciTePress (2020)

[48] Carfi, A., Mastrogiovanni, F.: Gesture-based human-machine interaction: Taxonomy, problem definition, and analysis. IEEE Transactions on Cybernetics pp. 1–17 (2021). https://doi.org/10.1109/tcyb.2021.3129119, `https://doi.org/10.1109%2Ftcyb.2021.3129119`

[49] Carpinella, C., Wyman, A., Perez, M., Stroessner, S.: The robotic social attributes scale (rosas): Development and validation. pp. 254–262 (03 2017). https://doi.org/10.1145/2909824.3020208

[50] Carreon, A., Smith, S.J., Frey, B., Rowland, A., Mosher, M.: Comparing immersive vr and non-immersive vr on social skill acquisition for students in middle school with asd. Journal of Research on Technology in Education pp. 1–14 (2023)

[51] Cephei, A.: Vosk speech recognition toolkit. `https://github.com/alphacep/vosk-api` (2025)

[52] Chanin, D.: Framenet parsing with transformers. `https://chanind.github.io/ai/2022/05/24/framenet-transformers.html` (2022), "

[53] Chen, H., Xu, Y., Ren, Y., Ye, Y., Li, X., Ding, N., Cong, P., Wang, Z., Liu, B., Chen, Y., Dou, Z., Leng, X., Li, M., Ma, Y., Tu, C.: Symbiosim: Human-in-the-loop simulation platform for bidirectional continuing learning in human-robot interaction (2025)

[54] Chen, Y., Xu, Y., Liu, Y., Huang, Y., Xu, Z., Chen, D.Y.: Attention-based multi-person pose estimation in videos. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR). pp. 2343–2352 (2018). https://doi.org/10.1109/CVPR.2018.00248

[55] Chen, Y., Liu, Z., Liu, Y., Zhang, Z.: 3d hand pose estimation with 2d marginal heatmaps. Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) pp. 5079–5088 (2017). https://doi.org/10.1109/CVPR.2017.539

[56] Clever, H.M., Erickson, Z., Kapusta, A., Turk, G., Liu, K., Kemp, C.C.: Bodies at rest: 3d human pose and shape estimation from a pressure image using synthetic data. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) (June 2020)

[57] Cockburn, J., Solomon, Y.: Multi-modal human robot interaction in a simulation environment (2013)

[58] Cohen, J.: Statistical Power Analysis for the Behavioral Sciences. Lawrence Erlbaum Associates, 2 edn. (1988)

[59] Conia, S., Bacciu, A., Navigli, R.: Unifying cross-lingual semantic role labeling with heterogeneous linguistic resources. In: Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies. pp. 338–351. Association for Computational Linguistics, Online (Jun 2021). https://doi.org/10.18653/v1/2021.naacl-main.31, https://aclanthology.org/2021.naacl-main.31/

[60] Conia, S., Orlando, R., Brignone, F., Cecconi, F., Navigli, R.: InVeRo-XL: Making cross-lingual Semantic Role Labeling accessible with intelligible verbs and roles. In: Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing: System Demonstrations. pp. 319–328. Association for Computational Linguistics, Online and Punta Cana, Dominican Republic (Nov 2021). https://doi.org/10.18653/v1/2021.emnlp-demo.36, https://aclanthology.org/2021.emnlp-demo.36/

[61] Cootes, T.F., Taylor, C.J., Cooper, D.H., Graham, J.: Active shape models—their training and application. Computer Vision and Image Understanding **61**(1), 38–59 (1995). https://doi.org/10.1006/cviu.1995.1004

[62] Coyle, C.L., Peterson, M.: Learnability testing of a complex software application. In: Marcus, A. (ed.) Design, User Experience, and Usability: Novel User Experiences - 5th International Conference, DUXU 2016, Held as Part of HCI International 2016, Toronto, Canada, July 17-22, 2016, Proceedings, Part II. Lecture Notes in Computer Science, vol. 9747, pp. 560–568. Springer (2016). https://doi.org/10.1007/978-3-319-40355-7_53, https://doi.org/10.1007/978-3-319-40355-7_53

[63] Cucari, G., Leotta, F., Mecella, M., Vassos, S.: Collecting human habit datasets for smart spaces through gamification and crowdsourcing. In: Games and Learning Alliance: 4th International Conference, GALA 2015, Rome, Italy, December 9-11, 2015, Revised Selected Papers 4. pp. 208–217. Springer (2016)

[64] Cutugno, F., Finzi, A., Fiore, M., Leone, E., Rossi, S.: Interacting with robots via speech and gestures, an integrated architecture. In: Interspeech 2013. pp. 3727–3731. ISCA (aug 25 2013). https://doi.org/10.21437/interspeech.2013-587, http://dx.doi.org/10.21437/Interspeech.2013-587

[65] Dautenhahn, K.: Socially intelligent robots: dimensions of human–robot interaction. Philosophical Transactions of the Royal Society B **362**(1480), 679–704 (2007)

[66] DeepMotion: Bringing digital humans to life with ai. https://www.deepmotion.com/ (Dec 2022), bringing Digital Humans to Life With AI.

[67] Dianatfar, M., Latokartano, J., Lanz, M.: Review on existing vr/ar solutions in human-robot collaboration. Procedia CIRP **97**, 407–411 (2021), https://api.semanticscholar.org/CorpusID:234197256

[68] Dianatfar, M., Latokartano, J., Lanz, M.: Review on existing VR/AR solutions in human–robot collaboration. Procedia CIRP **97**, 407–411 (2021). https://doi.org/10.1016/j.procir.2020.05.259, https://doi.org/10.1016%2Fj.procir.2020.05.259

[69] Dill, S., Rösch, A., Rohr, M., Güney, G., Witte, L.D., Schwartz, E., Antink, C.H.: Accuracy evaluation of 3d pose estimation with mediapipe pose for physical exercises. Current Directions in Biomedical Engineering **9**(1), 563–566 (2023). https://doi.org/doi:10.1515/cdbme-2023-1141, https://doi.org/10.1515/cdbme-2023-1141

[70] Dimitrokalli, A., Vosniakos, G.C., Nathanael, D., Matsas, E.: On the assessment of human-robot collaboration in mechanical product assembly by use of virtual reality. Procedia Manufacturing **51**, 627–634 (2020)

[71] Dominykas, S., Jan, H., Anna-Maria, F., A., A.H.: Robots and Wizards: An Investigation Into Natural Human–Robot Interaction. IEEE Access (2020). https://doi.org/10.1109/ACCESS.2020.3037724

[72] Dong, X., Yu, J., Zhang, J.: Joint usage of global and local attentions in hourglass network for human pose estimation. Neurocomputing **472**, 95–102 (2022)

[73] Dou, X., Yan, L., Wu, K., Niu, J.: Effects of voice and lighting color on the social perception of home healthcare robots. Applied Sciences **12**(23) (2022). https://doi.org/10.3390/app122312191

[74] Driess, D., et al.: Palm-e: An embodied multimodal language model (2023), https://arxiv.org/abs/2303.03378

[75] Dunn, O.: Multiple comparisons using rank sums. Technometrics **6**, 241–252 (1964)

[76] Echeverria, G., Lassabe, N., Degroote, A., Lemaignan, S.: Modular open robots simulation engine: Morse. In: 2011 ieee international conference on robotics and automation. pp. 46–51. IEEE (2011)

[77] Egerstedt, M., et al.: Behavior-Based Robotics. Springer (2004)

[78] Elmzaghi, M.: Voice-control-ros. https://github.com/moeelm/Voice-Control-ROS (2018), accessed: 2022-02-18

[79] Engemann, H., Du, S., Kallweit, S., Ning, C., Anwar, S.: Autosynpose: Automatic generation of synthetic datasets for 6d object pose estimation. In: Machine Learning and Artificial Intelligence, pp. 89–97. IOS Press (2020)

[80] Epic Games, Inc.: Metahuman: High-fidelity digital humans made easy. https://www.unrealengine.com/en-US/metahuman (March 2024), metaHuman: High-fidelity digital humans made easy. Unreal Engine. Retrieved March 10, 2024

[81] Erickson, N., Mueller, J., Shirkov, A., Zhang, H., Larroy, P., Li, M., Smola, A.: Autogluon-tabular: Robust and accurate automl for structured data (2020), https://arxiv.org/abs/2003.06505

[82] Etzi, R., Huang, S., Scurati, G.W., Lyu, S., Ferrise, F., Gallace, A., Gaggioli, A., Chirico, A., Carulli, M., Bordegoni, M.: Using virtual reality to test human-robot interaction during a collaborative task. In: International Design Engineering Technical Conferences and Computers and Information in Engineering Conference. ASME Digital Collection (2019)

[83] Fard, A.P., Abdollahi, H., Mahoor, M.: Asmnet: A lightweight deep neural network for face alignment and pose estimation. In: Proceedings of the IEEE/CVF Conference on computer vision and pattern recognition. pp. 1521–1530 (2021)

[84] Fillmore, C.: Frames and the semantics of understanding. Quaderni di semantica **6**, 222–254 (1985)

[85] Fillmore, C.J., Baker, C.F.: Frame semantics for text understanding. In: Proceedings of WordNet and Other Lexical Resources Workshop, NAACL. vol. 6 (2001)

[86] Finin, T., Fritzson, R., McKay, D., McEntire, R.: Kqml as an agent communication language. In: Proceedings of the Third International Conference on Information and Knowledge Management. p. 456–463. CIKM '94, Association for Computing Machinery, New York, NY, USA (1994). https://doi.org/10.1145/191246.191322, https://doi.org/10.1145/191246.191322

[87] Florea, A.G., Stoican, F., Buiu, C., Oară, C.: 3d pose estimation of custom objects using synthetic datasets. In: 2022 26th International Conference on System Theory, Control and Computing (ICSTCC). pp. 649–655 (2022). https://doi.org/10.1109/ICSTCC55426.2022.9931890

[88] Fratczak, P., Goh, Y.M., Kinnell, P., Soltoggio, A., Justham, L.: Understanding human behaviour in industrial human-robot interaction by means of virtual reality. In: Proceedings of the Halfway to the Future Symposium 2019. pp. 1–7 (2019)

[89] Freiberg, M., Baumeister, J.: A survey on usability evaluation techniques and an analysis of their actual application. University of Würzburg Institute of Computer Science Research Report Series **450** (2008)

[90] Furuta, Y.: Ros speech recognition. https://github.com/jsk-ros-pkg/jsk_3rdparty/tree/master/ros_speech_recognition (2021), accessed: 2022-02-18

[91] Galin, R., Meshcheryakov, R.: Review on human–robot interaction during collaboration in a shared workspace. In: Lecture Notes in Computer Science, pp. 63–74. Lecture notes in computer science, Springer International Publishing, Cham (2019)

[92] Gildea, D., Jurafsky, D.: Automatic labeling of semantic roles. Computational Linguistics **28**(3), 245–288 (2002), https://aclanthology.org/J02-3001

[93] Gleason, J.: Comparative power of the anova, randomization anova and kruskal-wallis test. Wayne State University Dissertations **658** (2013)

[94] Goldin-Meadow, S., Alibali, M.W.: Gestureapos;s role in speaking, learning, and creating language. Annual Review of Psychology **64**(Volume 64, 2013), 257–283 (2013). https://doi.org/https://doi.org/10.1146/annurev-psych-113011-143802, https://www.annualreviews.org/content/journals/10.1146/annurev-psych-113011-143802

[95] Goodrich, M.A., Schultz, A.C.: Human-robot interaction: A survey. Foundations and Trends® in Human-Computer Interaction **1**(3), 203–275 (2007). https://doi.org/10.1561/1100000005, https://doi.org/10.1561%2F1100000005

[96] Grifoni, P.: Multimodal Human Computer Interaction and Pervasive Services. Publisher Name (2009)

[97] Grishchenko, I., Bazarevsky, V., Zanfir, A., Bazavan, E.G., Zanfir, M., Yee, R., Raveendran, K., Zhdanovich, M., Grundmann, M., Sminchisescu, C.: Blazepose ghum holistic: Real-time 3d human landmarks and pose estimation. arXiv preprint arXiv:2206.11678 (2022)

[98] Groom, V., Nass, C.: Can robots be teammates?: Benchmarks in human–robot teams. Interaction Studies **8**, 483–500 (2007). https://doi.org/10.1075/is.8.3.10gro

[99] Guhr, O., Schumann, A.K., Bahrmann, F., Böhme, H.J.: Fullstop: Multilingual deep models for punctuation prediction. In: Swiss Text Analytics Conference (2021), https://api.semanticscholar.org/CorpusID:238232903

[100] Gvirsman, O., Koren, Y., Norman, T., Gordon, G.: Patricc: A platform for triadic interaction with changeable characters. In: Proceedings of the 2020 ACM/IEEE International Conference on Human-Robot Interaction. p. 399–407. HRI '20, Association for Computing Machinery, New York, NY, USA (2020). https://doi.org/10.1145/3319502.3374792, https://doi.org/10.1145/3319502.3374792

[101] Güler, R.A., Neverova, N., Kokkinos, I.: Densepose: Dense human pose estimation in the wild. Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) pp. 7297–7306 (2018). https://doi.org/10.1109/CVPR.2018.00762

[102] Hakim, N.L., Shih, T.K., Kasthuri Arachchi, S.P., Aditya, W., Chen, Y.C., Lin, C.Y.: Dynamic hand gesture recognition using 3dcnn and lstm with fsm context-aware model. Sensors **19**(24) (2019). https://doi.org/10.3390/s19245429, https://www.mdpi.com/1424-8220/19/24/5429

[103] Han, Z., Phan, A., Castro, A., Sandoval Garza, F., Williams, T.: Towards an understanding of physical vs virtual robot appendage design. In: International Workshop on Virtual, Augmented, and Mixed Reality for Human-Robot Interaction (2022)

[104] Hancock, P.A., Billings, D.R., Schaefer, K.E.: Can you trust your robot? Ergonomics in Design **19**, 24–29 (2011). https://doi.org/10.1177/1064804611415045

[105] Hancock, P.A., Billings, D.R., Schaefer, K.E., Chen, J.Y., de Visser, E.J., Parasuraman, R.: A meta-analysis of factors affecting trust in human-robot interaction. Human Factors **53**, 517–527 (2011). https://doi.org/10.1177/0018720811417254

[106] Hanna, N., Richards, D.: Speech act theory as an evaluation tool for human-agent communication. Algorithms **12**, 79 (2019)

[107] Hannun, A., Case, C., Casper, J., Catanzaro, B., Diamos, G., Elsen, E., Prenger, R., Satheesh, S., Sengupta, S., Coates, A., Ng, A.Y.: Deep speech: Scaling up end-to-end speech recognition. arXiv e-prints (2014)

[108] He, K., Gkioxari, G., Dollár, P., Girshick, R.B.: Mask r-cnn. In: Proceedings of the IEEE International Conference on Computer Vision (ICCV). pp. 2961–2969 (2017). https://doi.org/10.1109/ICCV.2017.319

[109] Holzapfel, H., Nickel, K., Stiefelhagen, R.: Implementation and evaluation of a constraint-based multimodal fusion system for speech and 3d pointing gestures. Proc of ICMI. ACM pp. 175–182 (2004)

[110] Hosseini-Asl, E., McCann, B., Wu, C.S., Yavuz, S., Socher, R.: A simple language model for task-oriented dialogue (2022)

[111] Hu, E.J., Shen, Y., Wallis, P., Allen-Zhu, Z., Li, Y., Wang, S., Wang, L., Chen, W.: Lora: Low-rank adaptation of large language models (2021)

[112] I., M., Izaskun, F., A., T., Johan, K., L., S., A., I., B., S.: Natural multimodal communication for human–robot collaboration (2017). https://doi.org/10.1177/1729881417716043, https://journals.sagepub.com/doi/pdf/10.1177/1729881417716043

[113] Iglesias, G., Talavera, E., Díaz-Álvarez, A.: A survey on gans for computer vision: Recent research, analysis and taxonomy. Computer Science Review **48**, 100553 (2023)

[114] Ikeda, T., Tanishige, S., Amma, A., Sudano, M., Audren, H., Nishiwaki, K.: Sim2real instance-level style transfer for 6d pose estimation (2022)

[115] Inamura, T., Mizuchi, Y.: Sigverse: A cloud-based vr platform for research on multimodal human-robot interaction. Frontiers in Robotics and AI **8** (2021)

[116] Ionescu, C., Papava, D., Olaru, V., Sminchisescu, C.: Human3.6m: Large scale datasets and predictive methods for 3d human sensing in natural environments. IEEE Transactions on Pattern Analysis and Machine Intelligence **36**(7), 1325–1339 (2014). https://doi.org/10.1109/TPAMI.2013.248

[117] Isleyen, E., Duzgun, S., Nelson, P.: Virtual reality for hazard assessment and risk mitigation in tunneling, pp. 4857–4863. CRC Press (07 2020). https://doi.org/10.1201/9781003031666-26

[118] Jaimes, A., Sebe, N.: Multimodal human–computer interaction: A survey. Computer Vision and Image Understanding **108**(1), 116–134 (2007). https://doi.org/https://doi.org/10.1016/j.cviu.2006.10.019, `https://www.sciencedirect.com/science/article/pii/S1077314206002335`, special Issue on Vision for Human-Computer Interaction

[119] Jiang, L., Yu, X., Wang, L.: A brief analysis of gesture recognition in vr. SID Symposium Digest of Technical Papers **51**(S1), 190–195 (2020). https://doi.org/https://doi.org/10.1002/sdtp.13787, `https://sid.onlinelibrary.wiley.com/doi/abs/10.1002/sdtp.13787`

[120] Jocher, G., Chaurasia, A., Qiu, J.: Ultralytics YOLO (Jan 2023), `https://github.com/ultralytics/ultralytics`

[121] Juraev, S., Ghimire, A., Alikhanov, J., Kakani, V., Kim, H.: Exploring human pose estimation and the usage of synthetic data for elderly fall detection in real-world surveillance. IEEE Access **PP**, 1–1 (01 2022). https://doi.org/10.1109/ACCESS.2022.3203174

[122] Kamińska, D., Zwoliński, G., Laska-Leśniewicz, A.: Usability testing of virtual reality applications - the pilot study. Sensors **22**(4) (2022), `https://www.mdpi.com/1424-8220/22/4/1342`

[123] Kamińska, D., Zwoliński, G., Laska-Leśniewicz, A.: Usability testing of virtual reality applications—the pilot study. Sensors **22**(4) (2022). https://doi.org/10.3390/s22041342, `https://www.mdpi.com/1424-8220/22/4/1342`

[124] Kanazawa, A., Black, M.J., Jacobs, D.W., Malik, J.: End-to-end recovery of human shape and pose (2018), `https://arxiv.org/abs/1712.06584`

[125] Kaszuba, S., Sabbella, S., Leotta, F., Serrarens, P., Nardi, D.: Testing human-robot interaction in virtual reality: Experience from a study on speech act classification. 6th International Workshop on Virtual, Augmented and Mixed-Reality for Human-Robot Interactions VAM-HRI 2023 (2023). https://doi.org/10.48550/arXiv.2401.04534

[126] Kaszuba, S., Leotta, F., Nardi, D.: A preliminary study on virtual reality tools in human-robot interaction. In: Lecture Notes in Computer Science, pp. 81–90. Lecture notes in computer science, Springer International Publishing, Cham (2021), https://link.springer.com/chapter/10.1007/978-3-030-87595-4_7

[127] Kaufeld, M., Nickel, P.: Level of robot autonomy and information aids in human-robot interaction affect human mental workload – an investigation in virtual reality. pp. 278–291 (2019)

[128] Kendon, A.: Gesture: Visible action as utterance. Cambridge University Press (2004)

[129] Khastgir, S., Birrell, S., Dhadyalla, G., Jennings, P.: Calibrating trust through knowledge: Introducing the concept of informed safety for automation in vehicles. Transportation Research Part C: Emerging Technologies **96**, 290–303 (2018). https://doi.org/10.1016/j.trc.2018.07.001.

[130] Kim, J.W., Choi, J.Y., Ha, E.J., Choi, J.H.: Human pose estimation using mediapipe pose and optimization method based on a humanoid model. Applied Sciences **13**(4) (2023). https://doi.org/10.3390/app13042700, https://www.mdpi.com/2076-3417/13/4/2700

[131] Kober, J., Bagnell, J.A., Peters, J.: Reinforcement learning in robotics: A survey. The International Journal of Robotics Research **32**(11), 1238–1274 (2013)

[132] Kocabas, M., Athanasiou, N., Black, M.J.: Vibe: Video inference for human body pose and shape estimation. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 5253–5263 (2020)

[133] Kok, B.C., Soh, H.: Trust in robots: Challenges and opportunities. Current Robotics Reports **1**, 297–309 (2020). https://doi.org/10.1007/s43154-020-00029-y

[134] Kounalakis, T., Jeppesen, K.C., Bæch, J., Hallam, B., Gasparri, A., Sanfeliu, A., Nardi, D., Ciarfuglia, T.A., Pagès, J., Bozdemir, G., Santamaria, M.: Canopies document d2.2: Specifications and kpis for the two canopies robot prototypes (rel. 01). Tech. Rep. 101016906, European Commission H2020 Project CANOPIES (2021)

[135] Krupke, D., Starke, S., Einig, L., Zhang, J., Steinicke, F.: Prototyping of immersive hri scenarios. In: Human-Centric Robotics: Proceedings of CLAWAR 2017: 20th International Conference on Climbing and Walking Robots and the Support Technologies for Mobile Machines. pp. 537–544. World Scientific (2018)

[136] Kuno, Y., Murashina, T., Shimada, N., Shirai, Y.: Intelligent wheelchair remotely controlled by interactive gestures. In: Proceedings 15th International Conference on Pattern Recognition. ICPR-2000. IEEE Comput.

Soc (2000). https://doi.org/10.1109/icpr.2000.903007, https://doi.org/10.1109%2Ficpr.2000.903007

[137] Kurth, J., Wagner, M.: New planning methods for systems with human-robot collaboration. ZWF Zeitschrift für wirtschaftlichen Fabrikbetrieb **115**(10), 698–702 (oct 2020). https://doi.org/10.3139/104.112419, https://doi.org/10.3139%2F104.112419

[138] lab, P.S.: ros-tutorial-voice. https://github.com/SMARTlab-Purdue/ros-tutorial-voice (2018), accessed: 2022-02-18

[139] Lackey, S., Barber, D., Reinerman, L., Badler, N., Hudson, I.: Defining next-generation multi-modal communication in human robot interaction. Proceedings of the Human Factors and Ergonomics Society Annual Meeting **55**, 461–464 (09 2011). https://doi.org/10.1177/1071181311551095

[140] Lai, K., Yanushkevich, S.: Cnn+rnn depth and skeleton based dynamic hand gesture recognition. pp. 3451–3456 (08 2018). https://doi.org/10.1109/ICPR.2018.8545718

[141] Lan, G., Wu, Y., Hu, F., Hao, Q.: Vision-based human pose estimation via deep learning: A survey. IEEE Transactions on Human-Machine Systems **53**(1), 253–268 (Feb 2023). https://doi.org/10.1109/thms.2022.3219242, http://dx.doi.org/10.1109/THMS.2022.3219242

[142] Latombe, J.C.: Robot Motion Planning. Springer (1991)

[143] Li, Z., Chen, L., Liu, C., Zhang, F., Li, Z., Gao, Y., Ha, Y., Xu, C., Quan, S., Xu, Y.: Animated 3d human avatars from a single image with gan-based texture inference. Computers & Graphics **95**, 81–91 (2021). https://doi.org/https://doi.org/10.1016/j.cag.2021.01.002, https://www.sciencedirect.com/science/article/pii/S0097849321000029

[144] Liang, J., Xu, L.Z., Lam, K.H., K., K.P.R.: Monocular 3d human pose estimation via graph convolutional networks. In: Proceedings of the IEEE International Conference on Computer Vision (ICCV). pp. 945–954 (2020). https://doi.org/10.1109/ICCV.2020.00018

[145] Lier, F., Wachsmuth, S.: Towards an open simulation environment for the pepper robot. In: Companion of the 2018 ACM/IEEE International Conference on Human-Robot Interaction. pp. 175–176 (2018)

[146] Lin, H., Tian, J., Lu, H.: Human pose estimation via recurrent spatial-temporal modeling. arXiv preprint arXiv:1806.09210 (2018), https://arxiv.org/abs/1806.09210

[147] Lin, T.Y., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D., Dollár, P., Zitnick, C.L.: Microsoft coco: Common objects in context. European Conference on Computer Vision (ECCV) pp. 740–755 (2014)

[148] Lin, Y., Jiao, X., Zhao, L.: Detection of 3d human posture based on improved mediapipe. Journal of Computer and Communications **11**(2), 102–121 (2023)

[149] Liu, D., Bhagat, K.K., Gao, Y., Chang, T.W., Huang, R.: The potentials and trends of virtual reality in education: A bibliometric analysis on top research studies in the last two decades. Virtual, augmented, and mixed realities in education pp. 105–130 (2017)

[150] Liu, D., Bhagat, K.K., Gao, Y., Chang, T.W., Huang, R.: The potentials and trends of virtual reality in education: A bibliometric analysis on top research studies in the last two decades. Virtual, augmented and mixed realities in education pp. 105–130 (2017)

[151] Liu, D., Bhagat, K.K., Gao, Y., Chang, T., Huang, R.: The potentials and trends of virtual reality in education (2017), `https://api.semanticscholar.org/CorpusID:158170306`

[152] Liu, H., Li, C., Wu, Q., Lee, Y.J.: Visual instruction tuning (2023), `https://arxiv.org/abs/2304.08485`

[153] Liu, X.S., Yi, X.S., Wan, L.C.: Friendly or competent? the effects of perception of robot appearance and service context on usage intention. Annals of Tourism Research **92**, 103324 (2022). https://doi.org/10.1016/j.annals.2021.103324

[154] Lo, C.C., Fu, S.W., Huang, W.C., Wang, X., Yamagishi, J., Tsao, Y., Wang, H.M.: Mosnet: Deep learning-based objective assessment for voice conversion. In: Interspeech 2019. interspeech$_2$019, $ISCA(sep2019).https$: $//doi.org/10.21437/interspeech.2019 - 2003$

[155] Luo, R.C., Wu, Y.C., Lin, P.H.: Multimodal information fusion for human-robot interaction. In: 2015 IEEE 10th Jubilee International Symposium on Applied Computational Intelligence and Informatics. pp. 535–540. IEEE (5 2015). https://doi.org/10.1109/saci.2015.7208262, `http://dx.doi.org/10.1109/SACI.2015.7208262`

[156] Ma, Y., Lin, Z., Han, Z., Liu, H.: Learning to estimate 3d human pose with part graphs. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR). pp. 734–743 (2019). https://doi.org/10.1109/CVPR.2019.00089

[157] Macalupu, V., Miller, E., Martin, L., Caldwell, G.: Human–robot interactions and experiences of staff and service robots in aged care. Scientific Reports **15** (01 2025). https://doi.org/10.1038/s41598-025-86255-w

[158] Maha, S., K., R., S., K., F., J.: A friendly gesture: Investigating the effect of multimodal robot behavior in human-robot interaction. IEEE International Symposium on Robot and Human Interactive Communication (2011). https://doi.org/10.1109/ROMAN.2011.6005285

[159] Mahmood, H.: Llama model debate. Available at: `https://datasciencedojo.com/blog/llama-model-debate/`, `https://datasciencedojo.com/blog/llama-model-debate/`

[160] Mahmoud, K., Harris, I., Yassin, H., Hurkxkens, T.J., Matar, O.K., Bhatia, N., Kalkanis, I.: Does immersive vr increase learning gain when compared to a non-immersive vr learning experience? In: Learning and Collaboration Technologies. Human and Technology Ecosystems: 7th International Conference, LCT 2020, Held as Part of the 22nd HCI International Conference, HCII 2020, Copenhagen, Denmark, July 19–24, 2020, Proceedings, Part II 22. pp. 480–498. Springer (2020)

[161] Malayjerdi, M., Kuts, V., Sell, R., Otto, T., Baykara, B.C.: Virtual simulations environment development for autonomous vehicles interaction. In: ASME International Mechanical Engineering Congress and Exposition. vol. 84492, p. V02BT02A009. American Society of Mechanical Engineers (2020)

[162] Malik, A.A., Masood, T., Bilberg, A.: Virtual reality in manufacturing: Immersive and collaborative artificial-reality in design of human-robot workspace. International Journal of Computer Integrated Manufacturing **33**(1), 22–37 (2020)

[163] Mao, R.Q., Lan, L., Kay, J., Lohre, R., Ayeni, O.R., Goel, D.P., de Sa, D.: Immersive virtual reality for surgical training: A systematic review. The Journal of surgical research **268**, 40–58 (2021), `https://api.semanticscholar.org/CorpusID:236158203`

[164] Mara, M., Meyer, K., Heiml, M., Pichler, H., Haring, R., Krenn, B., Gross, S., Reiterer, B., Layer-Wagner, T.: Cobot studio vr: A virtual reality game environment for transdisciplinary research on interpretability and trust in human-robot collaboration (2021)

[165] Màrquez, L., Carreras, X., Litkowski, K.C., Stevenson, S.: Semantic role labeling: an introduction to the special issue (2008)

[166] Matsas, E., Vosniakos, G.C., Batras, D.: Prototyping proactive and adaptive techniques for human-robot collaboration in manufacturing using virtual reality. Robotics and Computer-Integrated Manufacturing **50** (2017)

[167] McIntosh, V.: Dialing up the danger: Virtual reality for the simulation of risk. In: Frontiers in Virtual Reality (2022), `https://api.semanticscholar.org/CorpusID:251475476`

[168] Milliez, G., Ferreira, E., Fiore, M., Alami, R., Lefèvre, F.: Simulating human-robot interactions for dialogue strategy learning. In: International Conference on Simulation, Modeling, and Programming for Autonomous Robots. pp. 62–73. Springer (2014)

[169] Molchanov, P., Gupta, S., Kim, K., Kautz, J.: Hand gesture recognition with 3d convolutional neural networks. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR). pp. 4903–4911. IEEE (2016)

[170] Munea, T.L., Yang, C., Huang, C., Elhassan, M.A., Zhen, Q.: Simplecut: A simple and strong 2d model for multi-person pose estimation. Computer Vision and Image Understanding **222**, 103509 (2022)

[171] Murnane, M., Higgins, P., Saraf, M., Ferraro, F., Matuszek, C., Engel, D.: A simulator for human-robot interaction in virtual reality. In: 2021 IEEE Conference on Virtual Reality and 3D User Interfaces Abstracts and Workshops. pp. 470–471 (2021)

[172] Newell, A., Yang, K., Deng, J.: Stacked hourglass networks for human pose estimation. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR). pp. 483–490 (2016). https://doi.org/10.1109/CVPR.2016.60

[173] Nguyen, H.C., Nguyen, T.H., Nowak, J., Byrski, A., Siwocha, A., Le, V.H.: Combined yolov5 and hrnet for high accuracy 2d keypoint and human pose estimation. Journal of Artificial Intelligence and Soft Computing Research **12**(4), 281–298 (2022)

[174] Nielsen, J.: Designing Web Usability: The Practice of Simplicity. New Riders Publishing, USA (1999)

[175] Nikolakis, N., Maratos, V., Makris, S.: A cyber physical system (cps) approach for safe human-robot collaboration in a shared workplace. Robotics and Computer-Integrated Manufacturing **56**, 233–243 (2019)

[176] Novitzky, M., Semmens, R., Franck, N.H., Chewar, C.M., Korpela, C.: Virtual reality for immersive human machine teaming with vehicles. In: International Conference on Human-Computer Interaction. pp. 575–590. Springer (2020)

[177] Nugroho, H., Utama, N.P., Surendro, K.: Normalization and outlier removal in class center-based firefly algorithm for missing value imputation. Journal of Big Data **8** (2021). https://doi.org/10.1186/s40537-021-00518-7

[178] NVIDIA Corporation: NVIDIA Jetson Orin: Next-Generation Edge AI Platform (2022), `https://developer.nvidia.com/embedded/jetson-orin`

[179] NVIDIA Corporation: Jetson orin architecture overview (2023), `https://developer.nvidia.com/jetson-orin-architecture`, accessed: 2024-02-26

[180] NVIDIA Developer Blog: Deploying Accelerated LLaMA 3.2 from the Edge to the Cloud (2024), `https://developer.nvidia.com/blog/deploying-accelerated-llama-3-2-from-the-edge-to-the-cloud/`

[181] NVIDIA Jetson AI Lab: Deploying LLaMA on Jetson Orin (2024), `https://www.jetson-ai-lab.com/tutorial_llamaspeak.html`

[182] NVIDIA NGC: LLaMA-3.1-8b-Instruct PB Deployment (2024), `https://catalog.ngc.nvidia.com/orgs/nim/teams/meta/containers/llama-3.1-8b-instruct-pb24h2/security`

[183] O'Brien, P.D., Nicol, R.C.: Fipa — towards a standard for software agents. BT Technology Journal **16**(3), 51–59 (Jul 1998). https://doi.org/10.1023/A:1009621729979, `https://doi.org/10.1023/A:1009621729979`

[184] Omlor, A.J., Schwärzel, L.S., Bewarder, M., Casper, M., Damm, E., Danziger, G., Mahfoud, F., Rentz, K., Sester, U., Bals, R., Lepper, P.M.: Comparison of immersive

and non-immersive virtual reality videos as substitute for in-hospital teaching during coronavirus lockdown: a survey with graduate medical students in germany. Medical Education Online **27**(1), 2101417 (2022), https://doi.org/10.1080/10872981.2022.2101417

[185] Onnasch, L., Roesler, E.: A taxonomy to structure and analyze human–robot interaction. International Journal of Social Robotics **13**(4), 833–849 (Jul 2021)

[186] P. Qi, Y. Zhang, Y.Z.J.B., D, C.: Stanza: A python natural language processing toolkit for many human languages. pp. 101–108. Association for Computational Linguistics (2020). https://doi.org/10.18653/v1/2020.acl-demos.14, https://aclanthology.org/2020.acl-demos.14/

[187] Pan, M.K., Croft, E.A., Niemeyer, G.: Evaluating social perception of human-to-robot handovers using the robot social attributes scale (rosas). In: Proceedings of the 2018 ACM/IEEE International Conference on Human-Robot Interaction. p. 443–451. HRI '18, Association for Computing Machinery, New York, NY, USA (2018). https://doi.org/10.1145/3171221.3171257, https://doi.org/10.1145/3171221.3171257

[188] Pan, X., Hamilton, A.F.d.C.: Virtual reality for social robotics. Science Robotics **2**(7) (2017)

[189] Papandreou, G., Zhu, T., Chen, L.C., Gidaris, S., Tompson, J., Murphy, K.: Personlab: Person pose estimation and instance segmentation with a bottom-up, part-based, geometric embedding model. Proceedings of the European Conference on Computer Vision (ECCV) pp. 269–286 (2018). https://doi.org/10.1007/978-3-030-01225-0_16

[190] Park, S.: Multifaceted trust in tourism service robots. Annals of Tourism Research **81** (2020). https://doi.org/10.1016/j.annals.2020.102888.

[191] Patompak, P., Jeong, S., Nilkhamhang, I., Chong, N.Y.: Learning proxemics for personalized human-robot social interaction. International Journal of Social Robotics **12** (1 2020)

[192] Pavlakos, G., Zhou, X., Murphy, K.P., Rodriguez, J.M.G.: Harvard human pose estimation from a single rgb image. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR). pp. 1635–1644 (2017). https://doi.org/10.1109/CVPR.2017.00177

[193] Pavllo, D., Feichtenhofer, C., Grangier, D., Auli, M.: 3d human pose estimation in video with temporal convolutions and semi-supervised training. Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) pp. 7753–7762 (2019). https://doi.org/10.1109/CVPR.2019.00795

[194] Pavlovic, V., Sharma, R., Huang, T.: Visual interpretation of hand gestures for human-computer interaction: a review. IEEE Transactions on Pattern Analysis and Machine Intelligence **19**(7), 677–695 (jul 1997). https://doi.org/10.1109/34.598226, https://doi.org/10.1109%2F34.598226

[195] Pieraccini, R.: The Voice in the Machine: Building Computers That Understand Speech. MIT Press (2012)

[196] Prakash, A., Chitta, K., Geiger, A.: Structured domain randomization: Bridging the reality gap by context-aware synthetic data. arXiv preprint arXiv:1904.08890 (2019)

[197] PyTransitions: A Lightweight, Object-Oriented State Machine Implementation in Python (2023), `https://github.com/pytransitions/transitions`

[198] Radhakrishnan, U., Koumaditis, K., Chinello, F.: A systematic review of immersive virtual reality for industrial skills training. Behaviour & Information Technology **40**, 1310 – 1339 (2021), `https://api.semanticscholar.org/CorpusID:238861127`

[199] Raffel, C., Shazeer, N., Roberts, A., Lee, K., Narang, S., Matena, M., Zhou, Y., Li, W., Liu, P.J.: Exploring the limits of transfer learning with a unified text-to-text transformer. Journal of Machine Learning Research **21**(140), 1–67 (2020), `http://jmlr.org/papers/v21/20-074.html`

[200] Razali, N.M., Wah, Y.B.: Power comparisons of shapiro-wilk, kolmogorov-smirnov, lilliefors and anderson-darling tests. Journal of Statistical Modeling & Analytics **2**, 21–33 (2011)

[201] Reed, S., Zolna, K., Parisotto, E., Colmenarejo, S.G., Novikov, A., Barth-Maron, G., Giménez, M., Sulsky, Y., Kay, J., Springenberg, J.T., Eccles, T., Bruce, J., Razavi, A., Edwards, A.D., Heess, N.M.O., Chen, Y., Hadsell, R., Vinyals, O., Bordbar, M., de Freitas, N.: A generalist agent. ArXiv **abs/2205.06175** (2022), `https://api.semanticscholar.org/CorpusID:248722148`

[202] Renganayagalu, S.K., Mallam, S., Nazir, S., Ernstsen, J., Hogström, P.H.: Impact of simulation fidelity on student self-efficacy and perceived skill development in maritime training. TransNav, the International Journal on Marine Navigation and Safety of Sea Transportation (2019)

[203] Research, M.A.: Trellis github repository (2023), available at `https://github.com/microsoft/TRELLIS/tree/main`

[204] Rombach, R., Blattmann, A., Lorenz, D., Esser, P., Ommer, B.: High-resolution image synthesis with latent diffusion models (2022)

[205] Romportl, J.: Speech synthesis and uncanny valley. In: Sojka, P., Horák, A., Kopeček, I., Pala, K. (eds.) Text, Speech and Dialogue. pp. 595–602. Springer International Publishing, Cham (2014)

[206] Ross, M., Matarić, M.: Toward Robot Adaptation of Human Speech and Gesture Parameters in a Unified Framework of Proxemics and Multimodal Communication (2015)

[207] Rossi, S., Leone, E., Fiore, M., Finzi, A., Cutugno, F.: An extensible architecture for robust multimodal human-robot communication. 2013 IEEE/RSJ International Conference on Intelligent Robots and Systems, Tokyo, Japan pp. 2208–2213 (2013). https://doi.org/10.1109/IROS.2013.6696665

[208] Rothkrantz, L., Beelen, M.: An agent based travel assistant for the dutch railway network pp. 1–8 (2023). https://doi.org/10.1109/InfoTech58664.2023.10266886

[209] Rousseeuw, P.J., Hubert, M.: Robust statistics for outlier detection. WIREs Data Mining Knowl Discov **1**, 73–79 (2011). https://doi.org/10.1002/widm.2

[210] Rudenko, A., Kucner, T.P., Swaminathan, C.S., Chadalavada, R.T., Arras, K.O., Lilienthal, A.J.: ThÖr: Human-robot navigation data collection and accurate motion trajectories dataset. IEEE Robotics and Automation Letters **5**(2), 676–682 (Apr 2020). https://doi.org/10.1109/lra.2020.2965416, `http://dx.doi.org/10.1109/LRA.2020.2965416`

[211] Ríos-Hernández, M., Jacinto-Villegas, J.M., Portillo-Rodríguez, O., Vilchis-González, A.H.: User-centered design and evaluation of an upper limb rehabilitation system with a virtual environment. Applied Sciences **11**(20) (2021), `https://www.mdpi.com/2076-3417/11/20/9500`

[212] Salanitri, D., Hare, C., Borsci, S., Lawson, G., Sharples, S., Waterfield, B.: Relationship between trust and usability in virtual environments: An ongoing study. In: Kurosu, M. (ed.) Human-Computer Interaction: Design and Evaluation. pp. 49–59. Springer International Publishing, Cham (2015)

[213] Samaan, G., Wadie, A., Attia, A., Asaad, A., Kamel, A., Slim, S., Abdallah, M., Cho, Y.I.: Mediapipe's landmarks with rnn for dynamic sign language recognition. Electronics **11**, 3228 (10 2022). https://doi.org/10.3390/electronics11193228

[214] Samkari, E., Arif, M., Alghamdi, M., Al Ghamdi, M.A.: Human pose estimation using deep learning: A systematic literature review. Machine Learning and Knowledge Extraction **5**(4), 1612–1659 (2023)

[215] Sauro, J., Dumas, J.S.: Comparison of three one-question, post-task usability questionnaires. In: Proceedings of the SIGCHI Conference on Human Factors in Computing Systems. pp. 1599–1608. Boston, MA, USA (2009)

[216] Schaefer, K.E.: Measuring trust in human robot interactions: Development of the "trust perception scale-hri" (2016). https://doi.org/10.1007/978-1-4899-7668-0_10

[217] Schmitz, N., Hirth, J., Berns, K.: A simulation framework for human-robot interaction. In: 2010 Third International Conference on Advances in Computer-Human Interactions. pp. 79–84. IEEE (2010)

[218] Schneider, S., Kummert, F.: Comparing robot and human guided personalization: Adaptive exercise robots are perceived as more competent and trustworthy. International Journal of Social Robotics **13**, 169–185 (2021). https://doi.org/10.1007/s12369-020-00629-w

[219] Searle, J.R.: Speech acts: An essay in the philosophy of language. Cambridge University Press (1969)

[220] Sekine, K., Stam, G., Yoshioka, K., Tellier, M., Capirci, O.: Cross-lingusitic views of gesture usage. Vigo International Journal of Applied Linguistics VIAL **12**, 91–105 (2015)

[221] Shabaninia, E., Nezamabadi-pour, H., Shafizadegan, F.: Transformers in action recognition: A review on temporal modeling (2023). https://doi.org/10.48550/ARXIV.2302.01921, https://arxiv.org/abs/2302.01921

[222] Shamshiri, R., Hameed, I., Pitonakova, L., Weltzien, C., Balasundram, S., Yule, I., Grift, T., Chowdhary, G.: Simulation software and virtual environments for acceleration of agricultural robotics: Features highlights and performance comparison. International Journal of Agricultural and Biological Engineering **11**, 15–31 (01 2018). https://doi.org/10.25165/ijabe.v11i4.4032

[223] Shamshiri, R., Kalantari, F., Ting, K., Thorp, K., Hameed, I., Weltzien, C., Ahmad, D., Shad, Z.: Advances in greenhouse automation and controlled environment agriculture: A transition to plant factories and urban agriculture. International Journal of Agricultural and Biological Engineering **11**(1), 1–22 (2018). https://doi.org/10.25165/j.ijabe.20181101.3210, publisher Copyright: © 2018, Chinese Society of Agricultural Engineering. All rights reserved.

[224] Shiwa, T., Kanda, T., Imai, M., Ishiguro, H., Hagita, N.: How quickly should a communication robot respond? delaying strategies and habituation effects. I. J. Social Robotics **1**, 141–155 (04 2009)

[225] Shridhar, M., et al.: Rt-2: Vision-language-action models transfer web knowledge to robot control (2023), https://arxiv.org/abs/2307.15818

[226] Shu, B., Sziebig, G., Pieters, R.: Architecture for safe human-robot collaboration: Multi-modal communication in virtual reality for efficient task execution. pp. 2297–2302 (2019)

[227] Silvia, R., Enrico, L., M., F., Alberto, F., F., C.: An extensible architecture for robust multimodal human-robot communication. 2013 IEEE/RSJ International Conference on Intelligent Robots and Systems (2013). https://doi.org/10.1109/IROS.2013.6696665

[228] Singh, D.K.: 3d-cnn based dynamic gesture recognition for indian sign language modeling. Procedia Computer Science **189**, 76–83 (2021). https://doi.org/https://doi.org/10.1016/j.procs.2021.05.071, https://www.sciencedirect.com/science/article/pii/S1877050921011650, aI in Computational Linguistics

[229] Slaughter, D., Giles, D., Downey, D.: Autonomous robotic weed control systems: A review. Computers and Electronics in Agriculture **61**(1), 63–78 (2008). https://doi.org/https://doi.org/10.1016/j.compag.2007.05.008, https://www.sciencedirect.com/science/article/pii/S0168169907001688, emerging Technologies For Real-time and Integrated Agriculture Decisions

[230] Stiefelhagen, R., Fügen, C., Gieselmann, P., Holzapfel, H., Nickel, K., Waibel, A.: Natural human-robot interaction using speech, head pose and gestures. IEEE/RJS International Conference on Intelligent Robots and Systems (2004). https://doi.org/10.1109/IROS.2004.1389771

[231] Stiefelhagen, R., Ekenel, H.K., Fugen, C., Gieselmann, P., Holzapfel, H., Kraft, F., Nickel, K., Voit, M., Waibel, A.: Enabling multimodal human–robot interaction for the karlsruhe humanoid robot. IEEE Transactions on Robotics **23**(5), 840–851 (2007). https://doi.org/10.1109/TRO.2007.907484

[232] Su, H., Qi, W., Chen, J., Yang, C., Sandoval, J., Laribi, M.A.: Recent advancements in multimodal human–robot interaction. Frontiers in Neurorobotics **Volume 17 - 2023** (2023). https://doi.org/10.3389/fnbot.2023.1084000, https://www.frontiersin.org/journals/neurorobotics/articles/10.3389/fnbot.2023.1084000

[233] Sun, X., Xiao, B., Wei, F., Liang, S., Wei, Y.: Deep high-resolution representation learning for human pose estimation. Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) pp. 5693–5703 (2019). https://doi.org/10.1109/CVPR.2019.00584

[234] Suzuki, N., Watanabe, Y., Nakazawa, A.: Gan-based style transformation to improve gesture-recognition accuracy. Proc. ACM Interact. Mob. Wearable Ubiquitous Technol. **4**(4) (dec 2020). https://doi.org/10.1145/3432199, https://doi.org/10.1145/3432199

[235] Tan, J.C.A., Chan, W.P., Robinson, N.L., Croft, E.A., Kulic, D.: A proposed set of communicative gestures for human robot interaction and an rgb image-based gesture recognizer implemented in ros (2021). https://doi.org/10.48550/ARXIV.2109.09908, https://arxiv.org/abs/2109.09908

[236] Tang, H., Wang, W., Xu, D., Yan, Y., Sebe, N.: Gesturegan for hand gesture-to-gesture translation in the wild (2018). https://doi.org/10.48550/ARXIV.1808.04859, https://arxiv.org/abs/1808.04859

[237] Tasmere, D., Ahmed, B., Das, S.: Real time hand gesture recognition in depth image using cnn. International Journal of Computer Applications **174**, 28–32 (01 2021). https://doi.org/10.5120/ijca2021921040

[238] Tcha-Tokey, K., Christmann, O., Loup-Escande, E., Richir, S.: Proposition and validation of a questionnaire to measure the user experience in immersive virtual environments. Int. J. Virtual Real. **16**, 33–48 (2016)

[239] TensorFlow, Google: Movenet model card. https://tfhub.dev/google/movenet/singlepose/thunder/4, https://storage.googleapis.com/movenet/MoveNet.SinglePose%20Model%20Card.pdf

[240] Thrun, S., et al.: Probabilistic Robotics. MIT Press (2005)

[241] Tiedemann, J., Thottinga, S.: OPUS-MT – building open translation services for the world. In: Proceedings of the 22nd Annual Conference of the European Association for Machine Translation. pp. 479–480. European Association for Machine Translation, Lisboa, Portugal (Nov 2020), https://aclanthology.org/2020.eamt-1.61/

[242] Tobin, J., Fong, R., Ray, A., Schneider, J., Zaremba, W., Abbeel, P.: Domain randomization for transferring deep neural networks from simulation to reality. In:

2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS). pp. 23–30. IEEE (2017)

[243] Tompson, J., Jain, A., LeCun, Y., Bregler, C.: Joint training of a convolutional network and a graphical model for human pose estimation (2014), https://arxiv.org/abs/1406.2984

[244] Touvron, H., et al.: Llama: Open and efficient foundation language models. arXiv **2302.13971** (2023)

[245] Traum, D.R.: Speech acts for dialogue agents. In: Foundations of conversational informatics. pp. 103–140 (2003)

[246] Unity Technologies: Characters. https://unity.com/solutions/characters (March 2024), create rich, complex, and believable characters for any size or style of production.

[247] Unity Technologies: Unity engine: Unity's real-time 3d development engine. https://unity.com/products/unity-engine (March 2024), unity's real-time 3D development engine lets artists, designers, and developers collaborate to create amazing immersive and interactive experiences.

[248] Unsöld, M.: Measuring learnability in human-computer interaction. Masters thesis, Ulm University (2018)

[249] Urakami, J., Seaborn, K.: Nonverbal cues in human–robot interaction: A communication studies perspective. ACM Transactions on Human-Robot Interaction **12**(2), 1–21 (Mar 2023). https://doi.org/10.1145/3570169, http://dx.doi.org/10.1145/3570169

[250] Varol, G., Romero, J., Martin, X., Mahmood, N., Black, M.J., Laptev, I., Schmid, C.: Learning from synthetic humans. CoRR **abs/1701.01370** (2017), http://arxiv.org/abs/1701.01370

[251] Ventura, S., Brivio, E., Riva, G., Baños, R.M.: Immersive versus non-immersive experience: Exploring the feasibility of memory assessment through 360° technology. Frontiers in Psychology **10** (2019), https://www.frontiersin.org/articles/10.3389/fpsyg.2019.02509

[252] Vergara-Rodríguez, D., Gómez-Asenjo, A., Fernández-Arias, P., Gómez-Vallecillo, A.I., Lamas-Álvarez, V.E., de Santos de La Iglesia, C.: Immersive vs. non-immersive virtual reality learning environments. In: 2021 XI International Conference on Virtual Campus (JICV). pp. 1–3 (2021)

[253] Vlahovic, S., Suznjevic, M., Skorin-Kapov, L.: A survey of challenges and methods for quality of experience assessment of interactive vr applications. Journal on Multimodal User Interfaces **16**, 257 – 291 (2022)

[254] Wang, C., Du, B., Xu, J., Li, P., Guo, D., Liu, H.: Demonstrating humanthor: A simulation platform and benchmark for human-robot collaboration in a shared workspace (2024), https://arxiv.org/abs/2406.06498

[255] Wang, J., Xie, Y., Zhang, J., Yu, W., Lu, Z., Zhang, T.: Deep human pose estimation with hrnet. IEEE Transactions on Pattern Analysis and Machine Intelligence **42**(8), 2017–2028 (2020). https://doi.org/10.1109/TPAMI.2020.2970925

[256] Wang, L.: Developing scenarios and research methodologies for evaluating human-robot interaction in social contexts (2022), `https://hdl.handle.net/11573/1631217`

[257] Wang, T., Zheng, P., Li, S., Wang, L.: Multimodal human–robot interaction for human-centric smart manufacturing: A survey. Advanced Intelligent Systems **6**(3), 2300359 (2024). https://doi.org/10.1002/aisy.202300359, `https://doi.org/10.1002/aisy.202300359`

[258] Wheeler, S.G., Engelbrecht, H., Hoermann, S.: Human factors research in immersive virtual reality firefighter training: A systematic review. In: Frontiers in Virtual Reality (2021), `https://api.semanticscholar.org/CorpusID:238586492`

[259] Wijnen, L., Bremner, P., Lemaignan, S., Giuliani, M.: Performing human-robot interaction user studies in virtual reality. In: 29th IEEE International Conference on Robot and Human Interactive Communication (RO-MAN). pp. 794–794 (2020)

[260] Williams, J., Rownicka, J., Oplustil, P., King, S.: Comparison of speech representations for automatic quality estimation in multi-speaker text-to-speech synthesis. In: The Speaker and Language Recognition Workshop (Odyssey 2020). pp. 222–229 (2020). https://doi.org/10.21437/Odyssey.2020-32

[261] Wirtz, J., Patterson, P.G., Kunz, W.H., Gruber, T., Lu, V.N., Paluch, S., Martins, A.: Brave new world: service robots in the frontline. Journal of Service Management **29**(5), 907–931 (2018). https://doi.org/10.1108/JOSM-04-2018-0119, `https://doi.org/10.1108/JOSM-04-2018-0119`

[262] Wu, C.J., Raghavendra, R., Gupta, U., Acun, B., Ardalani, N., Maeng, K., Chang, G., Aga, F., Huang, J., Bai, C., Gschwind, M., Gupta, A., Ott, M., Melnikov, A., Candido, S., Brooks, D., Chauhan, G., Lee, B., Lee, H.H., Akyildiz, B., Balandat, M., Spisak, J., Jain, R., Rabbat, M., Hazelwood, K.: Sustainable ai: Environmental implications, challenges and opportunities **4**, 795–813 (2022)

[263] Xiang, J., Lv, Z., Xu, S., Deng, Y., Wang, R., Zhang, B., Chen, D., Tong, X., Yang, J.: Structured 3d latents for scalable and versatile 3d generation (2024), `https://arxiv.org/abs/2412.01506`

[264] Xiao, B., Wu, H., Wei, Y.: Simple baselines for human pose estimation and tracking. In: Proceedings of the European Conference on Computer Vision (ECCV). pp. 466–481 (2018)

[265] Xu, H., Trulls, E., Billard, A., Fua, P.: 3d human pose, shape and motion estimation using multi-view image fusion. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). pp. 5084–5094 (2020). https://doi.org/10.1109/CVPR42600.2020.00512

[266] Xu, H., Bazavan, E.G., Zanfir, A., Freeman, W.T., Sukthankar, R., Sminchisescu, C.: Ghum & ghuml: Generative 3d human shape and articulated pose models. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 6184–6193 (2020)

[267] Xu, X., Chen, W., Li, Y., Dai, D., Song, S.: Voxelpose: A volumetric representation for human pose estimation. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR). pp. 3432–3441 (2020). https://doi.org/10.1109/CVPR42600.2020.00351

[268] Xu, Y., Zhang, J., Zhang, Q., Tao, D.: Vitpose: Simple vision transformer baselines for human pose estimation. arXiv e-prints (2022)

[269] Y. Tang, C. Tran, X.L.P.J.C.N.G.V.C.J.G., Fan, A.: Multilingual translation with extensible multilingual pretraining and finetuning. arXiv e-prints (2020)

[270] Yan, Y., Jia, Y.: A review on human comfort factors, measurements and improvements in human–robot collaboration. Sensors **22**(19) (2022). https://doi.org/10.3390/s22197431

[271] Ye, T., Minato, T., Sakai, K., Sumioka, H., Hamilton, A., Ishiguro, H.: Human-like interactions prompt people to take a robot's perspective. Frontiers in Psychology **14**, 1190620 (2023). https://doi.org/10.3389/fpsyg.2023.1190620, `https://doi.org/10.3389/fpsyg.2023.1190620`

[272] Yongda, D., Fang, L., Huang, X.: Research on multimodal human-robot interaction based on speech and gesture. Computers & Electrical Engineering **72**, 443–454 (11 2018). https://doi.org/10.1016/j.compeleceng.2018.09.014, `http://dx.doi.org/10.1016/j.compeleceng.2018.09.014`

[273] Young, S., Gasic, M., Thomson, B., Williams, J.D.: Pomdp-based statistical spoken dialog systems: A review. Proceedings of the IEEE **101**(5), 1160–1179 (2013)

[274] Yu, J., Qin, M., Zhou, S.: Dynamic gesture recognition based on 2d convolutional neural network and feature fusion. Scientific Reports **12**, 4345 (03 2022). https://doi.org/10.1038/s41598-022-08133-z

[275] Yu, X., Jiang, L., Wang, L.: Virtual reality gesture recognition based on depth information. SID Symposium Digest of Technical Papers **51**, 196–200 (2020)

[276] Zacharaki, A., Kostavelis, I., Gasteratos, A., Dokas, I.M.: Safety bounds in human robot interaction: A survey. Safety Science **127**, 104667 (2020), `https://api.semanticscholar.org/CorpusID:216294346`

[277] Zhang, A.: Speechrecognition: Library for performing speech recognition, with support for several engines and apis, online and offline. `https://pypi.org/project/SpeechRecognition/` (2022)

[278] Zhang, B.J., Peterson, K., Sanchez, C.A., Fitter, N.T.: Exploring consequential robot sound: Should we make robots quiet and kawaii-et? In: 2021 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS). pp. 3056–3062 (2021). https://doi.org/10.1109/IROS51168.2021.9636365

[279] Zhang, J., Jiang, Z., Yang, D., Xu, H., Shi, Y., Song, G., Xu, Z., Wang, X., Feng, J.: Avatargen: A 3d generative model for animatable human avatars. In: Karlinsky, L., Michaeli, T., Nishino, K. (eds.) Computer Vision – ECCV 2022 Workshops. pp. 668–685. Springer Nature Switzerland (2023)

[280] Zhang, Q.: Precision Agriculture Technology for Crop Farming. CRC Press (2015), https://doi.org/10.1201/b19336

[281] Zhang, W., Wang, J.: Dynamic hand gesture recognition based on 3d convolutional neural network models. In: 2019 IEEE 16th International Conference on Networking, Sensing and Control (ICNSC). pp. 224–229 (2019). https://doi.org/10.1109/ICNSC.2019.8743159

[282] Zhang, Y., Liu, H., Kang, S.C., Al-Hussein, M.: Virtual reality applications for the built environment: Research trends and opportunities. Automation in Construction **118**, 103311 (oct 2020). https://doi.org/10.1016/j.autcon.2020.103311, https://doi.org/10.1016%2Fj.autcon.2020.103311

[283] Zheng, C., Wu, M., Wang, W., Wang, L.: 3d human pose estimation with spatial and temporal transformers. Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV) pp. 5664–5673 (2021). https://doi.org/10.1109/ICCV48922.2021.00562

[284] Zheng, C., Wu, W., Chen, C., Yang, T., Zhu, S., Shen, J., Kehtarnavaz, N., Shah, M.: Deep learning-based human pose estimation: A survey. ACM Comput. Surv. **56**(1) (aug 2023). https://doi.org/10.1145/3603618, https://doi.org/10.1145/3603618

[285] Zmora, N.: Achieving FP32 accuracy for INT8 inference using quantization aware training with NVIDIA TensorRT. NVIDIA Blog (Jul 2021), accessed: 2025-27-2