robotics

MDPI

Article

A Collaborative Robotic System for Autonomous Object Handling with Natural User Interaction

Federico Neri *, Gaetano Lettera, Giacomo Palmieri and Massimo Callegari

DIISM—Department of Industrial Engineering and Mathematical Sciences, Polytechnic University of Marche, 60131 Ancona, Italy; g.lettera@staff.univpm.it (G.L.); g.palmieri@staff.univpm.it (G.P.); m.callegari@staff.univpm.it (M.C.)

* Correspondence: federico.neri@staff.univpm.it

Abstract

In Industry 5.0, the transition from fixed traditional automation to flexible human–robot collaboration (HRC) needs interfaces that are both intuitive and efficient. This paper introduces a novel, multimodal control system for autonomous object handling, specifically designed to enhance natural user interaction in dynamic work environments. The system integrates a 6-Degrees of Freedom (DoF) collaborative robot (UR5e) with a hand-eye RGB-D vision system to achieve robust autonomy. The core technical contribution lies in a vision pipeline utilizing deep learning for object detection and point cloud processing for accurate 6D pose estimation, enabling advanced tasks such as human-aware object handover directly onto the operator's hand. Crucially, an Automatic Speech Recognition (ASR) is incorporated, providing a Natural Language Understanding (NLU) layer that allows operators to issue real-time commands for task modification, error correction and object selection. Experimental results demonstrate that this multimodal approach offers a streamlined workflow aiming to improve operational flexibility compared to traditional HMIs, while enhancing the perceived naturalness of the collaborative task. The system establishes a framework for highly responsive and intuitive human–robot workspaces, advancing the state of the art in natural interaction for collaborative object manipulation.

Keywords: collaborative robotics; multimodal HRI; voice control interface; 6D manipulation; hand–eye vision; human-aware handover

1. Introduction

1.1. Background and Motivation

The transition from Industry 4.0 to Industry 5.0 [1] represents a paradigm shift in manufacturing and production systems, moving from highly closed and efficiency-driven automated production lines to open, flexible, human-centric workspaces where human intelligence and decision-making are integrated with robotic automation. While Industry 4.0 focused on the integration of cyber-physical systems, IoT and data-driven optimization, Industry 5.0 highlights human–robot collaboration (HRC), adaptability and the well-being of humans operators [2], aiming at creating environments where humans and robots work synergistically to achieve a shared goal [3,4]. Collaborative robots, also cobots, have become central to this transition. Unlike traditional industrial robots, which typically operate in isolated cells due to strict safety constraints, cobots are designed for safe, direct physical interaction with humans. This is achieved through force-limited actuators, collision detection mechanisms and adaptive control algorithms. These capabilities enable operation in

Check for updates

Academic Editor: Stefano Baraldo

Received: 24 January 2026
Revised: 19 February 2026
Accepted: 24 February 2026
Published: 27 February 2026

Copyright: © 2026 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license.

shared workspaces. However, current implementations often rely on pre-programmed task sequences which limit system flexibility, as they may not adapt effectively to dynamic environmental changes, variations in operator behavior or unexpected workflow disruptions. In this work, the notion of dynamic work environment refers to collaborative industrial scenarios in which task conditions evolve due to human presence and interaction, rather than to large-scale or fully unstructured environmental changes. Specifically, dynamism arises from variations in object poses, human motion within the shared workspace and on-the-fly task reconfiguration enabled through natural language commands. Within this scope, the proposed framework focuses on real-time perception and adaptive human–robot interaction to support flexible and intuitive collaboration.

To fully leverage the potential of HRC, it is possible to develop Natural User Interaction (NUI) mechanisms that allow operators to communicate with robots intuitively, using common modalities such as gestures or voice commands. NUI significantly improves operational efficiency by enabling real-time adaptation to task changes. In particular, it aims to reduce the cognitive load on the operator, who no longer needs to remember complex sequences or navigate complicated menus and it decreases physical effort by minimizing repetitive movements or manual corrections during interaction with the robot.

Moreover, in complex and unstructured work environments, where objects may vary in size, shape, or location, autonomous perception integrated with NUI becomes essential. It allows robots not only to detect and manipulate objects accurately but also to respond fluidly to human guidance. By combining intuitive control with autonomous capabilities, advanced HRC systems can achieve higher flexibility and safety, paving the way for truly human-centric industrial workspaces.

1.2. Related Work

Although significant progress has been made in human-aware object handover [5] and multimodal human--robot interaction [6], several important limitations remain in the state of the art. For instance, Costanzo et al. [7] propose a reactive control scheme using visual and haptic cues to achieve object handover in collaborative tasks, but rely on pre-defined behaviors and constrained hand configurations, which limits adaptability and naturalness in dynamic scenarios. Reinforcement-learning approaches for human-to-robot handover [8] improve adaptability, but they typically require extensive training and do not tightly integrate with real-time perception and natural interaction channels like speech.

On the interaction interface side, multimodal systems combining gesture and speech have shown significant potential for enhancing natural communication in HRC. For example, Salinas-Martínez et al. [9] demonstrated a gesture + Automatic Speech Recognition (ASR) platform for PCB manufacturing with high interaction accuracy. However, the proposed architecture remains largely modular and not deeply integrated with perception or manipulation pipelines. Similarly, Chen et al. [10] developed a real-time HRC framework capable of processing dynamic gestures and speech in parallel, yet it still lacks autonomous object-handling and human-aware grasp/handover capabilities. More broadly, recent multimodal frameworks are increasingly hierarchical and long-term, as shown in the work of Yu et al. [11], who propose a multimodal planning model based on vision and speech. However, these approaches tend to focus on high-level task planning and user modeling rather than on integrated perception-to-handover pipelines for grounded object manipulation. Complementing these efforts, recent studies in industrial ASR highlight how speech recognition can reduce reliance on additional sensors and enhance interaction naturalness in precision-critical environments [12]. Nevertheless, to the best of the authors knowledge, these ASR-enabled systems are typically not coupled with real-time manipulation modules, limiting their applicability to real collaborative tasks. Recent survey works highlight the

growing interest in vision–language model-based human–robot collaboration for smart manufacturing, emphasizing the potential of tightly coupled perceptual and linguistic representations to enable flexible, context-aware interaction and long-horizon task execution [13]. These approaches represent an important step toward more general and intuitive HRC systems. However, these approaches primarily focus on high-level reasoning and task planning, and typically do not explicitly address real-time perception-to-manipulation coupling or safety-critical physical interaction. In parallel, vision-driven manipulation strategies based on fuzzy visual servoing have been successfully applied to the handling of flexible or deformable objects, such as fabric sheets lying on a work table, demonstrating robustness against geometric uncertainty and object deformation through continuous visual feedback and adaptive control [14]. While highly effective within their application domain, these methods are generally tailored to specific manipulation tasks involving non-rigid materials and do not consider multimodal human–robot interaction, natural language-driven task adaptation, or human-aware grasp and handover scenarios.

In contrast, our work addresses these limitations by proposing a fully integrated architecture that closes the perception–interaction–manipulation loop seamlessly, unifying autonomous 6D object localization, object grasping, human-aware handover, and a real-time ASR-based interaction interface. This tight integration allows the robot to dynamically adapt to user commands, recover from execution errors and modify tasks on the fly, enabling safe and natural object handover in collaborative scenarios while reducing the rigidity and training overheads typical of existing approaches.

1.3. Scientific Contribution

This paper presents an integrated collaborative robotic framework designed to support flexible and natural HRC in Industry 5.0 scenarios. The main scientific contributions of this work can be summarized as follows:

- Integrated perception–interaction–manipulation architecture: a unified system architecture is proposed, combining autonomous 6D object localization, grasp execution, human-aware handover and natural language-based interaction within a single ROS 2 framework, while leveraging standard perception and middleware components where appropriate.
- Robust vision-based pipeline for autonomous grasping and handover: a vision pipeline is developed that bridges 2D deep learning-based object detection with 3D point cloud processing to estimate reliable 6D grasping poses in unstructured environments. The same perception framework is extended to enable human-aware object handover by accurately localizing the operator's hand online using an eye-in-hand RGB-D setup.
- Voice Control Interface for flexible task-level interaction: a fully offline Voice Control Interface is introduced, leveraging Automatic Speech Recognition and Large Language Model-based Natural Language Understanding to translate unstructured human speech into deterministic robot commands. This approach enables intuitive task selection, modification and error handling without requiring rigid command templates or manual reprogramming.
- Safe and responsive handover strategy based on multimodal sensing: a human-aware handover strategy is implemented by combining vision-based hand tracking with force-based release logic through the robot's real-time control interface. This multimodal approach ensures safe and natural object transfer while maintaining responsiveness and robustness during close physical interaction.

2. System Architecture

2.1. Hardware Setup

The experimental workstation, illustrated in Figure 1, is designed as a collaborative robotic cell capable of autonomous object manipulation and natural interaction with a human operator. The hardware architecture comprises the robotic manipulator, the vision system, the voice interface, and the processing unit.

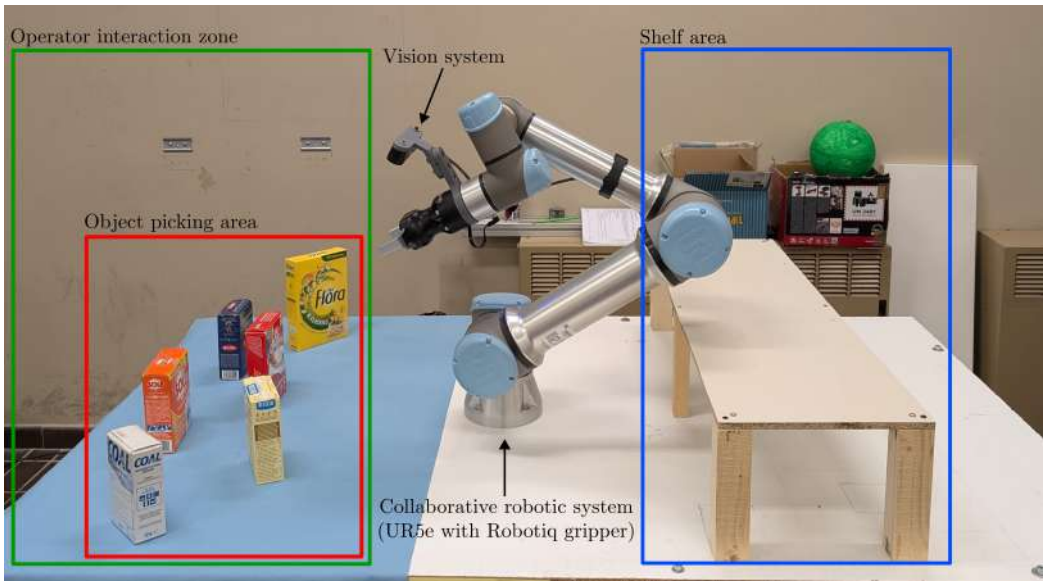


Figure 1. The experimental setup featuring the collaborative robot, the eye-in-hand vision system, and the workspace with the target objects.

The motion capabilities are provided by a Universal Robots UR5e, a 6-Degrees of Freedom (DoF) collaborative robot. This manipulator was selected for its safety certification and collision detection features, which allow it to operate in shared workspaces without physical barriers. Its kinematic configuration offers the necessary flexibility to reach various zones of the workbench, facilitating both object picking and handover tasks. The end-effector mounted on the robot flange is a Robotiq Hand-E adaptive electric parallel gripper. This device provides the necessary stroke and force control to handle the target objects, which consist of various commercial products simulating a supermarket inventory. These items are characterized by differing dimensions and branding, though they share a predominantly parallelepiped shape. The vision system consists of an Orbbec Gemini 2L RGB-D camera, which utilizes Active Stereo IR technology to generate dense depth maps. The sensor is mounted in an eye-in-hand configuration using a custom-designed bracket manufactured via Stereolithography (SLA). The bracket positions the camera at a fixed inclination of 35° relative to the wrist axis; this angle was optimized to ensure simultaneous visibility of both the manipulation targets and the operator's hand during collaborative tasks. This configuration enables the real-time acquisition of RGB streams and 3D point clouds.

To facilitate natural user interaction, the configuration includes a wireless lavalier microphone. This device captures the operator's voice commands, enabling hands-free communication and eliminating the need for physical input devices such as buttons, keyboards, or touchscreens. Finally, the synchronisation and control of all hardware components are managed by a central computer. This workstation establishes communication between devices, processing visual and audio data streams and executing algorithms for object recognition and motion planning.

2.2. Software Architecture and Communication

The software framework governing the robotic cell is built upon ROS 2, specifically the Humble Hawksbill distribution running on Ubuntu 22.04 LTS. The architecture adopts a modular design composed of independent nodes that communicate asynchronously to manage perception, decision-making and user interaction [15]. Within this architecture, established off-the-shelf software components are adopted where possible, while custom-developed modules are introduced to enable the tight integration required for HRC. In particular, the ROS 2 middleware, the YOLO backbone for 2D object detection, Google MediaPipe for hand detection and the ASR engine are employed as standard components, whereas the multimodal coordination logic, the 2D–3D fusion pipeline for 6D pose estimation, the LLM-based intent grounding and the human-aware handover strategy have been specifically developed by the authors.

The system logic is structured around three primary computational nodes, as illustrated in the block diagram in Figure 2:

1. Perception Node: this module processes raw sensory data to reconstruct the workspace state. It integrates a vision pipeline that combines YOLO for object detection and Google MediaPipe for hand detection. These 2D features are mapped into metric 3D space using ordered point clouds, enabling precise 6D pose estimation of both target objects and the operator's hand relative to the robot base.
2. Voice Interface Node: this module implements the Voice Control Interface (VCI). It leverages OpenAI Whisper (Large-v3) for robust Automatic Speech Recognition (ASR), capable of filtering industrial background noise. The transcribed text is processed by a local instance of the Mistral 7B Large Language Model (LLM), which performs semantic parsing to extract structured intents (e.g., action type, target selection) from unstructured natural language commands.
3. Control Node: acting as a central supervisor, this module functions as a finite state machine. It synchronises multimodal inputs, activating specific vision tasks based on the analysed vocal intention, and generates high-level motion plans for the manipulator.

To support a responsive and safe human–robot handover, the system interfaces with the robotic manipulator through the Real-Time Data Exchange (RTDE) protocol provided by Universal Robots. Unlike standard TCP/IP socket communication, which may suffer from non-deterministic latency, RTDE is designed to synchronize external applications with the robot's internal control loop over a standard Ethernet connection [16]. The communication is established by defining specific input and output "recipes"—lists of variables such as joint positions, velocities, forces, and digital I/O states—that are exchanged cyclically. This mechanism allows the Control Node to send motion commands and receive the robot's state feedback continuously. Furthermore, the RTDE interface provides direct access to the robot's safety registers, allowing the software to monitor protective stops or emergency states and halt execution if an anomaly is detected.

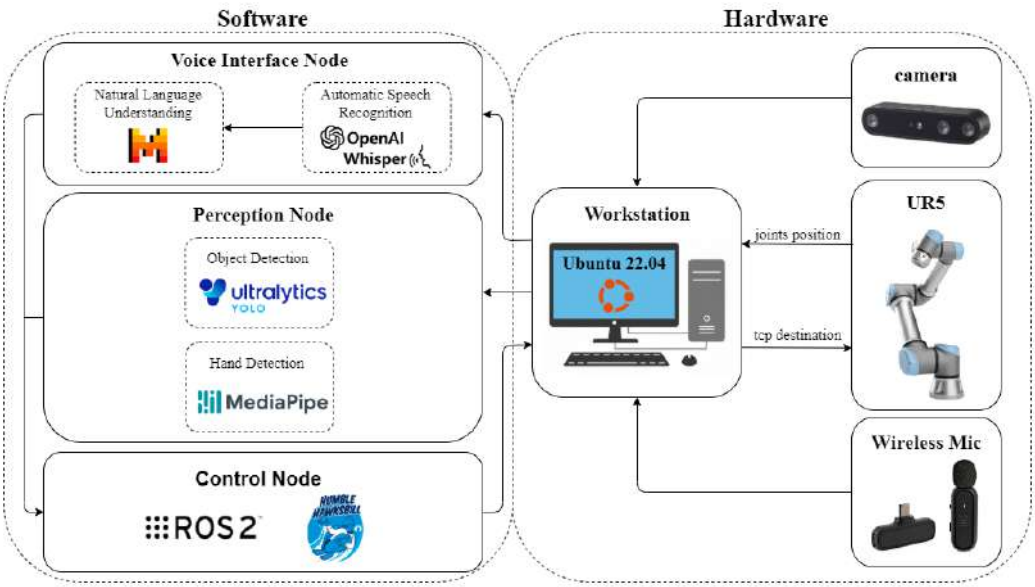


Figure 2. Block diagram of the ROS 2 software architecture, illustrating the integration of deep learning-based vision and LLM-based voice interaction with the RTDE robot control loop.

3. Object Recognition and Grasping Pipeline

3.1. Pose Estimation and Grasping Pipeline

The object recognition and pose estimation pipeline is designed to bridge the gap between 2D visual perception and 3D physical manipulation, enabling autonomous grasp execution in unstructured environments. The workflow follows a sequential process that starts from deep learning-based object localization and concludes with the computation of a full 6D grasping pose through point-cloud-based manipulation processing. Object detection is performed on the RGB stream using the YOLOv8 architecture, specifically the yolov8m.pt (medium) model. This architecture was selected as it offers the optimal trade-off between inference speed and detection accuracy, ensuring robust recognition for grasping tasks, while maintaining real-time performance within the ROS 2 Humble framework. For each detected supermarket item, the network outputs a class label and a 2D bounding box, as shown in Figure 3A.

To transition from the 2D image plane to the 3D Cartesian space, the system leverages the depth data synchronized with the RGB frame and the organized point cloud generated by the RGB-D sensor (see Figure 3B). A segmentation step is executed to delete the surrounding environment, such as shelves or support surfaces, using a plane segmentation filter. To ensure robust background removal, a distance threshold of 0.01 m was applied to identify and subtract the planar support surface. The global point cloud is then cropped using the bounding box provided by YOLOv8, retaining only the points within the target object to be grasped and significantly reducing point-cloud cardinality, as shown in Figure 3C. Subsequently, a Euclidean clustering technique is employed to isolate only the points on the front surface of the object (see Figure 3D). A clustering tolerance of 0.02 m was selected to group neighboring points, effectively filtering out sensor noise and isolated outliers. Once a clean point cloud cluster is obtained, the 6D pose of the object can be estimated as follows. This second segmentation step plays a crucial role in the accurate estimation of the 6D object pose: the system enables a reliable computation of the surface centroid using the MomentOfInertiaEstimation function provided by the Point Cloud Library (PCL). In particular, the centroid is obtained through the getOBB subroutine, which computes an Oriented Bounding Box (OBB) enclosing the input point cloud. When applied to the segmented front surface, the resulting OBB degenerates into a planar structure rather

than a volumetric one, since it represents only the visible surface of the object. Although a small thickness may be present due to imperfect segmentation, this effect is negligible for the purposes of pose estimation. Once the centroid of the front surface is computed, a translation is applied along the direction normal to the surface, corresponding to the object local Z-axis, as shown in Figure 3E. The centroid is shifted by a distance equal to half of the estimated object thickness (a priori known value derived from object specifications), resulting in an accurate approximation of the object's geometric center (see Figure 3F).

This simplified geometric approach was selected to prioritize system responsiveness, which is essential for the proposed NUI framework. While the method relies on a semi-rigid object model, the system robustness is ensured by the adaptive capabilities of the Robotiq Hand-E gripper. By utilizing internal force sensing and object detection, the gripper terminates the closing action upon encountering resistance, thereby compensating for potential discrepancies between the estimated and actual object thickness during the physical interaction.

Finally, the estimated pose, initially expressed in the camera optical frame, is transformed into the robot base frame through the calibrated homogeneous transformation $\mathbf{T}_{camera}^{base}$ obtained from a standard hand–eye calibration procedure. This transformation ensures that the computed grasping pose is directly compatible with the UR5e controller and can be seamlessly used for motion planning and grasp execution.

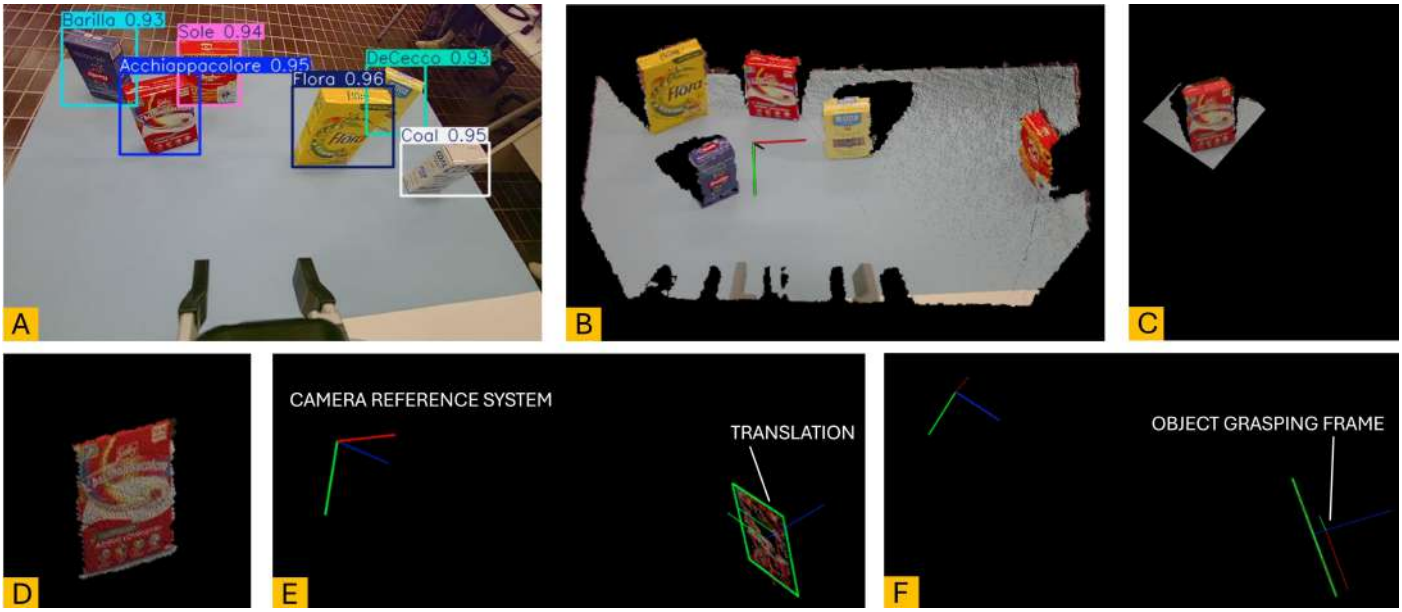


Figure 3. Grasping pose estimation pipeline from RGB-D data: YOLOv8-based object detection (**A**), point-cloud cropping (**B**) and clustering (**C**), surface centroid estimation via OBB (**D**) and centroid translation along the surface normal (**E**) to obtain the final 6D grasping pose (**F**). The red, green, and blue lines represent the X, Y, and Z axes of the coordinate frames, respectively.

3.2. Handover Planning

The handover phase constitutes the interface between autonomous perception and physical human–robot interaction. The process begins with the localization of the operator's hand. The vision pipeline employs Google MediaPipe Hands to infer a skeletal model consisting of 21 landmarks from the RGB stream. To ensure tracking stability, the Metacarpophalangeal (MCP) joint of the middle finger (Landmark 9) was selected as the reference target, as illustrated in Figure 4a. Unlike fingertips, which exhibit high positional variance during grasping gestures, the MCP joint provides a stable centroid for the hand's palm, offering an optimal target for object placement.

Since the extracted landmarks $\mathbf{u} = (u, v)$ are defined in the 2D image plane, a deprojection step is performed by fusing the visual data with the aligned depth map. By querying the organized point cloud (Figure 4b) at the indices of the target landmark, the system retrieves the depth value z_{camera} and reconstructs the 3D coordinates $\mathbf{P}_{camera} = [x_c, y_c, z_c]^T$ using the camera intrinsic parameters. Subsequently, the target coordinate is transformed into the robot base frame, $\mathbf{P}_{base}$, to enable motion planning:

$$\mathbf{P}_{base} = \mathbf{T}_{base}^{TCP}(q) \cdot \mathbf{T}_{TCP}^{camera} \cdot \mathbf{P}_{camera} \tag{1}$$

where $\mathbf{T}_{base}^{TCP}(q)$ denotes the pose of the Tool Center Point (TCP) derived from the instantaneous forward kinematics and the joint state q, and $\mathbf{T}_{TCP}^{camera}$ represents the extrinsic calibration matrix of the eye-in-hand system described in the hardware setup.

(**a**) 2D skeletal tracking (**b**) 3D point cloud

Figure 4. Visualization of the hand localization pipeline. (**a**) RGB view showing the MediaPipe skeletal tracking; the arrow highlights the Middle Finger MCP (Landmark 9) used as the stable handover target. (**b**) The corresponding 3D point cloud of the scene used to extract the metric depth of the operator's hand.

To facilitate a natural interaction, the motion planner avoids direct linear interpolation to the moving target. Instead, a "pre-place" approach strategy is adopted. The system computes a safety waypoint $\mathbf{P}_{safe} = \mathbf{P}_{base} + [0, 0, z_{offset}]^T$, with a vertical offset $z_{offset} = 0.2\,\text{m}$ above the operator's hand. This fixed offset was selected as a conservative approach heuristic to ensure sufficient clearance during the approach phase within the considered collaborative scenario. While this choice proved effective for the proposed supermarket inventory task, future work will investigate more geometry-independent handover strategies to extend the applicability of the method. The robot traverses to this waypoint using joint-space interpolation to minimize execution time, before descending to the final handover coordinate. Crucially, the TCP orientation is constrained to align the Z-axis with the gravity vector. This vertical configuration maximizes the object's affordance, allowing the operator to grasp it intuitively from the sides.

The final release mechanism is governed by an impedance-based logic to ensure safety. Instead of relying on visual triggers, which can be prone to occlusion during close contact, the system utilizes haptic feedback via the RTDE interface. Upon reaching the safety waypoint, the robot descends vertically towards the handover coordinate at a reduced speed of 0.03 m/s. During this phase, the robot enters a "wait-and-sense" state, monitoring the vertical force component F_z acting on the end-effector. The gripper release is triggered only when the operator actively retrieves the object, detected when the external force exceeds a predefined threshold:

$$|F_z| > F_{th} \implies \text{Open Gripper} \tag{2}$$

The release logic relies on the UR5e's integrated 6-axis Force/Torque sensor, monitored at 500 Hz via the RTDE interface. This reading represents the estimated external force acting on the TCP, as the robot's internal controller actively compensates for gravity and payload dynamics based on the current kinematic configuration. Consequently, the measurement is effectively configuration-independent during the static holding phase. Regarding sensor performance, the UR5e technical specifications state a force precision of 3.5 N and an accuracy of 4.0 N. Therefore, the threshold $F_{th} = 5.0$ N was empirically established to strictly exceed the sensor's worst-case tolerance range. This value creates a robust safety margin against signal noise and measurement inaccuracies while ensuring the operator can trigger the release with minimal physical effort.

4. Voice Control Interface

The Voice Control Interface (VCI) enables hands-free management of the robotic cell [17]. To ensure reliability in industrial environments, the system implements a fully offline, two-stage deep learning pipeline processed on the local edge workstation.

4.1. VCI Design and Components

The pipeline bridges the gap between unstructured audio and deterministic robotic actions through two primary computational blocks:

1. Automatic Speech Recognition (ASR) [18,19]: Audio is captured via a wireless microphone and processed by OpenAI Whisper (Large-v3 model). Operating locally, it provides a textual transcription of the speech, filtering out industrial ambient noise.
2. Natural Language Understanding (NLU) [20,21]: The transcription is processed by Mistral 7B via Ollama. This component acts as the semantic reasoning engine, utilizing attention mechanisms to perform intent recognition and slot filling. It extracts the target Object ID and the intended destination directly from the semantic structure of the sentence.

4.2. Integration into Robot Control

The integration into the ROS 2 control loop relies on a domain-specific prompt engineering strategy. The LLM is conditioned to translate stochastic human speech into structured control messages containing specific Object IDs matching the computer vision classes.

The prompt supports operational flexibility by distinguishing between two manipulation primitives:

- Handover Mode: Delivering the object to the operator (default).
- Shelf Mode: Placing the object in storage.

The prompt structure is illustrated in Figure 5.

This configuration enables unconstrained phrasing: the operator is not required to use a rigid command template. As shown in Table 1, conversational fillers are filtered out, and the system maps the intent to the correct ID and destination.

Table 1. Semantic parsing results. The system handles flexible sentence structures while correctly mapping intents to the structured JSON required by the control node.

Intent Type	User Utterance (Input)	Structured JSON (Output)
Handover	"Hand me Object 1, please."	{"action": "place_operator", "parameters": {"label": "Object 1"}}
Shelf	"Put Object 2 on the shelf."	{"action": "place_shelf", "parameters": {"label": "Object 2"}}

Table 1. *Cont.*

Intent Type	User Utterance (Input)	Structured JSON (Output)
Phrasing	"I really need Object 3."	{"action": "place_operator", "parameters": {"label": "Object 3"}}
Safety	"Stop the robot."	{"action": "stop", "parameters": {}}

Prompt

```
You are a command parser for a UR5e robot in a supermarket.
Convert the following command into a STRICT JSON object.
DO NOT output anything outside of JSON.
DO NOT add comments or explanations.
Actions:
• pick, move, home, stop
• place_shelf (if destination is 'shelf')
• place_operator (if destination is 'operator' or 'hand')
Rules:
• Extract label (the object name mentioned).
• Extract brand if present.
• If destination contains 'shelf', use place_shelf.
• If destination contains 'operator' or 'hand', use place_operator.
• If ambiguous → action: "unknown".
Output format:
{
   "action": "...",
   "parameters": {
     "label": "...",
     "brand": "...",
     "destination": "shelf|operator|<other>"
   }
}
Examples:
User: 'Give me Object 1.' →
Output: { "action": "place_operator", "parameters": { "label": "Object
1", "brand": null, "destination": "operator" } }
User: 'Put Object 2 on the shelf.' →
Output: { "action": "place_shelf", "parameters": { "label": "Object 2",
"brand": null, "destination": "shelf" } }
User: 'Stop immediately.' →
Output: { "action": "stop", "parameters": {} }
User: [INPUT_TRANSCRIPT] → Output:
```

Figure 5. Structure of the system prompt used for LLM parsing.

The communication flow is managed via ROS 2. The mistral node publishes the JSON command to /parsed_command. The orchestration node decodes this message: if the action is place_operator, the robot retrieves the target object using YOLO coordinates, moves to the detected hand position, and releases the object upon detecting a pull force ($F_z > 5\,\mathrm{N}$).

4.3. LLM Model Selection and Latency Analysis

To determine the optimal NLU engine for the robotic cell, a comparative analysis was conducted using a high-performance edge workstation. The objective was to identify a model that balances response latency with semantic reasoning capabilities.

Three quantized models were evaluated on the standard command defined in the previous section:

1. Qwen 2.5 (1.5B): Lightweight architecture designed for speed.
2. Llama 3.2 (3B): Mid-sized optimized model.
3. Mistral 7B Instruct (v0.2): Larger parameter count, offering higher reasoning capabilities.

The inference times reported in Table 2 represent the average warm-start performance, excluding the initial model loading time.

Table 2. Comparison of LLM inference performance.

Model	Params	Time (Avg) [s]	Selection Analysis
Qwen 2.5	1.5B	0.19	Valid alternative for low-power hardware.
Llama 3.2	3B	0.19	Valid alternative with good speed-to-size ratio.
Mistral 7B	7B	0.37	Selected model. Optimal balance of robustness and speed.

The experimental data demonstrate that hardware acceleration brings all tested models well within the real-time domain. While Qwen 2.5 and Llama 3.2 offer the fastest response times (0.19 s), the system adopts Mistral 7B. All models achieved real-time performance ($<$1 s). Mistral 7B was selected as the optimal choice, providing the best trade-off between execution speed and model capacity for robust language understanding. The latency difference of approximately 180 ms is imperceptible to the human operator during natural conversation. However, utilizing a 7B-parameter architecture provides a significant advantage in terms of semantic understanding and robustness against ambiguous commands compared to smaller models, making it the most reliable engine for the proposed collaborative architecture.

5. Experimental Results

5.1. Experimental Setup

The experimental validation was conducted in a laboratory environment simulating a collaborative industrial workstation. The workspace consists of a table where the objects are initially placed and a storage shelf positioned behind the base of the robotic manipulator.

The manipulation targets were selected from a set of commercial products representative of standard supermarket inventory. To ensure robust detection, the vision system utilizes a YOLO model specifically fine-tuned on these items. To maintain neutrality, the objects are classified by an alphanumeric identifier (Object 1 to 6). The test set, illustrated in Figure 6, includes:

- Object 1: Pasta box (Dark blue packaging);
- Object 2: Pasta box (Yellow/Blue packaging);
- Object 3: Rice box (Bright yellow packaging);
- Object 4: Detergent box (Orange/White packaging);
- Object 5: Laundry additive (Red packaging);
- Object 6: Sodium bicarbonate (White/Blue packaging).

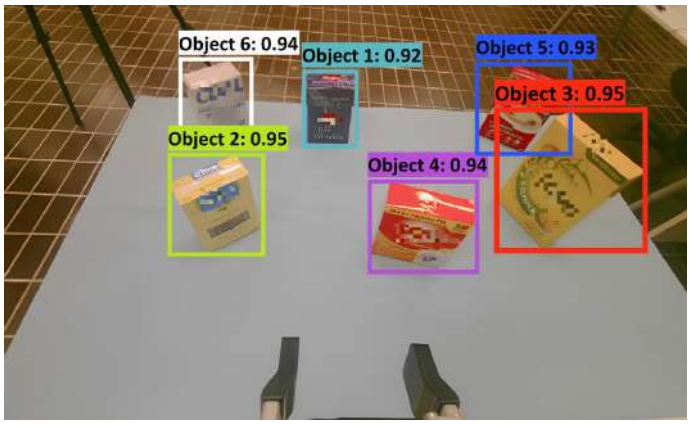


Figure 6. Commercial objects used for experimental validation, labeled according to the object IDs (Object 1–Object 6) adopted in the study.

5.2. YOLO-Based Object Detection Results

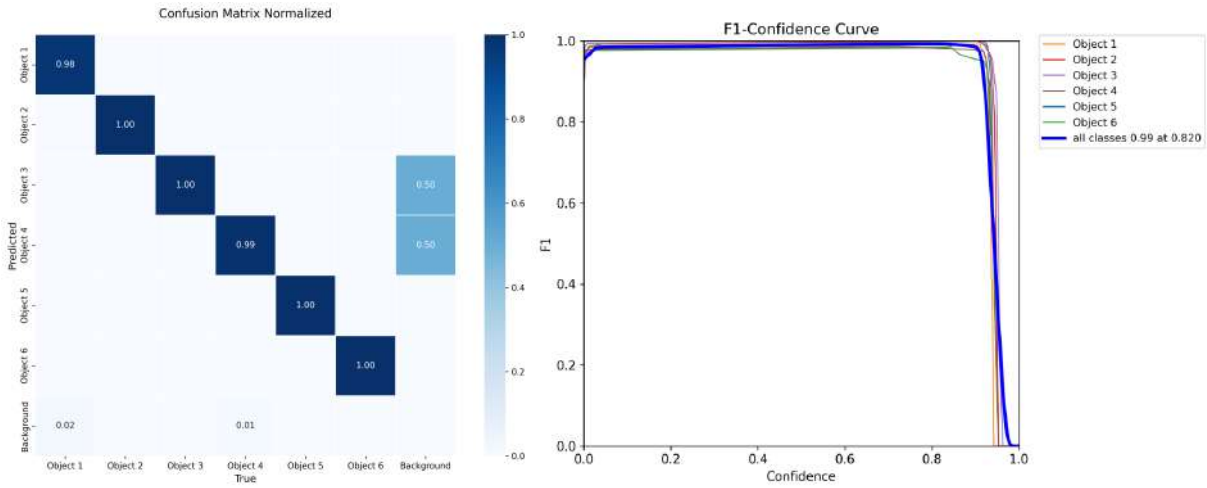
To evaluate the reliability of the perception module, the YOLOv8 neural network was trained and tested on a custom dataset constructed specifically for the supermarket inventory task. The dataset comprises 500 RGB images captured using the Orbbec Gemini 2L camera in the eye-in-hand configuration. To ensure robustness against environmental variability, images were collected with the robot moving through different approach angles and varying lighting conditions. The dataset was partitioned into a training set (70%, 350 images) and a validation set (30%, 150 images).

The primary evaluation objective was the model's ability to correctly detect and classify the six target items (Object 1–6) while reliably distinguishing them from the background. Performance was quantitatively assessed using standard object detection metrics, including mean Average Precision (mAP). To ensure reproducibility and robustness against overfitting on the task-specific dataset, the training process was conducted using an input image resolution of 640×640 pixels and the default YOLOv8 data augmentation pipeline, including Mosaic and Mixup strategies. The optimization was performed using Stochastic Gradient Descent (SGD) with an initial learning rate of 10^{-2}, momentum of 0.937, and a batch size of 16 over 100 epochs. Evaluation on the validation set demonstrated strong generalization performance, achieving an mAP@0.5 of 0.992 and an mAP@0.5:0.95 of 0.88, in agreement with the classification results observed in the confusion matrix.

Figure 7a presents the normalized confusion matrix obtained after training. The model demonstrates excellent classification performance, achieving true positive rates between 0.98 and 1.00 across all classes. Classification errors are negligible, with only minor confusion observed between two packages due to similar color palettes in low-light scenarios. This high precision is critical for the subsequent pose estimation step, as an incorrect class label would retrieve the wrong geometric parameters for the point cloud processing.

To further analyze the trade-off between precision and recall, the F1–confidence curve was generated (Figure 7b). The F1 score, which represents the harmonic mean of precision and recall, peaks at 0.99 with a confidence threshold of 0.82.

Based on these results, an operational confidence threshold of 0.90 was selected for the application. Although this threshold is slightly higher than the optimal F1 peak, it prioritizes precision to eliminate false positives, ensuring that the robot attempts a grasp only when the certainty is maximal.



(**a**) Normalized confusion matrix (**b**) F1–confidence curve

Figure 7. Training metrics overview.

5.3. System Execution Test

To validate the integration of the multimodal architecture, a complete collaborative cycle was executed and recorded. Figure 8 illustrates the key phases of the interaction, demonstrating the system's ability to translate high-level natural language instructions into physical actions.

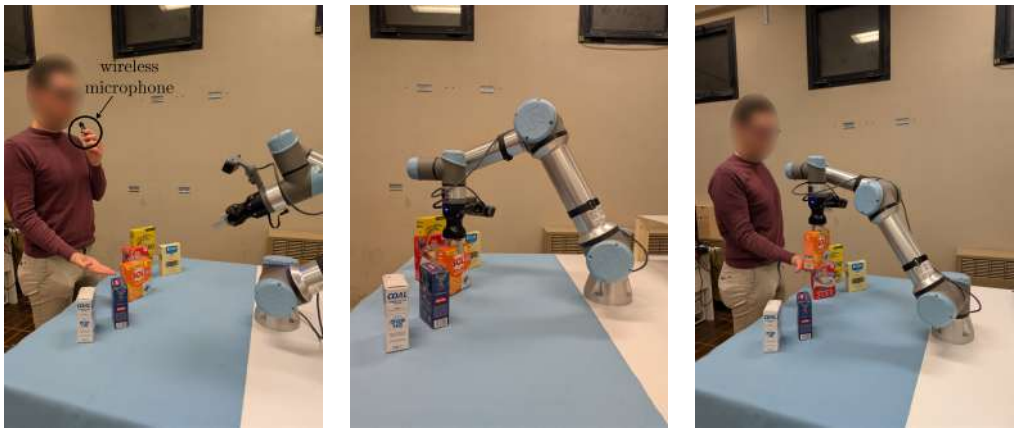
The sequence proceeds as follows:

1. Natural Language Input (Figure 8a): the operator issues a voice command using unstructured phrasing (e.g., "I need Object 4 for the washing"). The VCI processes the audio, extracts the semantic intent ("Object 4") and the destination ("operator"), and forwards the task to the robot controller.
2. Autonomous Picking (Figure 8b): once the command is received, the vision system identifies the target object and the operator's hand. The robot performs the approach and grasping movement based on the calculated 6D position.
3. Human-Aware Handover (Figure 8c): the robot follows the trajectory towards the operator's hand and releases the object only when the force detects contact, ensuring a smooth and safe transfer.

This qualitative validation confirms the system's capability to handle the entire process, from voice activity detection to physical release, while maintaining a natural interaction flow. It is important to highlight that, while the presented experiments focus on validating the technical feasibility and real-time integration of the proposed multimodal architecture, the evaluation of human-centered metrics such as cognitive load and usability was planned as part of a broader user study.

5.4. System Responsiveness and Technical Validation

To substantiate the architectural efficiency and the "real-time" processing capabilities of the individual modules, we conducted a quantitative performance analysis over 30 continuous execution cycles. To evaluate the system across different use cases, the trials were designed to cover a wide range of operational conditions rather than repeating a single task. The test set included:



(**a**) Voice command (**b**) Autonomous picking (**c**) Handover

Figure 8. Sequential frames of the collaborative task: (**a**) the operator issues a natural voice command via the wireless microphone; (**b**) the robot autonomously identifies and grasps the requested object; (**c**) the object is safely delivered directly to the operator's hand.

- Object variability: the target was randomly selected among the six items (Objects 1–6) for each run, ensuring the vision system's robustness across different textures and shapes.
- Spatial uncertainty: objects were placed at random positions and orientations within the camera's Field of View, testing the 6D pose estimation pipeline.
- Linguistic flexibility: the operators used varying phrasings, validating the LLM's intent classification capabilities.

Table 3 details the computational time for the key pipeline stages. The data confirms that the perception module operates with high responsiveness:

- Vision processing: the complete 6D pose estimation (Detection + Point Cloud Processing) requires on average 66 ms (≈15 Hz), supporting a fluid workflow.
- Safety control: the force monitoring and release logic operate in strict hard real-time at 500 Hz (2 ms) via the RTDE interface, ensuring safety during the physical handover.

While the Voice Interface introduces a natural latency due to the ASR and LLM inference (≈3.6 s), the robotic subsystem reacts immediately once the intent is parsed.

The intermediate outputs of the perception pipeline were analyzed to verify alignment with the training metrics shown in Figure 7b. During the experiments, the object detection confidence averaged 0.94 ± 0.03 across all successful detections. This value is consistent with the optimal F1-score threshold identified during training (>0.90), confirming that the model generalizes well to the real-world setup without significant performance degradation.

The system's technical validity was further assessed through a success rate analysis. Out of 30 trials involving random object placements and voice commands:

- Physical handover success: 100% (30/30). The force-based release triggered correctly in all instances without false positives.
- End-to-End task success: 93% (28/30). Two failures occurred, due to background noise affecting the ASR.

These results demonstrate that the proposed architecture effectively integrates high-level reasoning with robust low-level control.

Table 3. Computational processing times.

Subsystem	Component	Avg. Time [ms]
Voice Interface	ASR (Whisper)	3200 ± 450
	Intent Parsing (LLM)	370 ± 50
Vision Pipeline	Obj. Detection	22 ± 4
	6D Pose Estimation	44 ± 8
Control Loop	RTDE	2 (fixed)

6. Conclusions

This work presented a comprehensive framework for Industry 5.0 applications, integrating autonomous 6D perception with a natural language voice interface to support human–robot collaboration. By coupling deep learning-based object detection (YOLOv8) with point cloud processing, the system achieves precise grasping and human-aware handover in unstructured environments. The incorporation of a Large Language Model (Mistral 7B) into the control loop represents a significant step forward in interaction flexibility, allowing operators to interact with the robotic cell using unstructured natural speech rather than rigid command templates.

Experimental results confirmed the technical reliability of the proposed vision pipeline, demonstrating high classification accuracy and stable pose estimation, which are essential for safe physical interaction. The system effectively bridges the gap between high-level semantic intents and low-level motion control, providing a flexible architecture that limits the need for complex reprogramming when handling task variations or unexpected interaction events. In this work, environmental dynamism is mainly related to human-in-the-loop interaction, including human motion, object reconfiguration and task updates driven by natural language commands, while large-scale environmental changes are left to future extensions.

Another limitation of the current approach is the reliance on pre-defined object thickness values for centroid estimation. While this assumption holds for standardized rigid packaging in retail inventories, it may introduce positional offsets when handling unknown or deformable objects. However, the sensitivity to such errors is mitigated by the mechanical compliance and force feedback of the adaptive gripper, which compensates for axial misalignments during the grasping phase. Future work will address this by investigating data-driven mesh reconstruction and tactile feedback strategies to handle non-rigid objects without a priori geometric knowledge.

Future work will focus on expanding the system's adaptability in dynamic shared workspaces. One of the main objectives is the integration of real-time obstacle avoidance strategies [22] to further enhance safety during close human–robot collaboration. Moreover, while this study validated the technical feasibility and real-time performance of the proposed architecture, a comprehensive user study is planned to evaluate human-centered aspects of the interaction. Systematic evaluations will be conducted using standardized questionnaires, such as the NASA Task Load Index (NASA-TLX) and the System Usability Scale (SUS), to quantitatively assess operator workload and perceived interaction quality, and to compare the proposed multimodal interface with traditional industrial controllers.

Author Contributions: Conceptualization, F.N. and G.L.; Methodology, F.N., G.L. and G.P.; Software, F.N. and G.L.; Validation, F.N.; Investigation, F.N. and G.L.; Writing—original draft preparation, F.N. and G.L.; Writing—review and editing, G.P.; Visualization, G.P. and M.C.; Supervision, G.P. and M.C. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Data Availability Statement: Data are contained within the article.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Mourtzis, D. Challenges and opportunities of the transition from Industry 4.0 to Industry 5.0. In *Manufacturing from Industry 4.0 to Industry 5.0*; Elsevier: Amsterdam, The Netherlands, 2024; pp. 97–131.
2. Antonaci, F.G.; Olivetti, E.C.; Marcolin, F.; Castiblanco Jimenez, I.A.; Eynard, B.; Vezzetti, E.; Moos, S. Workplace well-being in industry 5.0: A worker-centered systematic review. *Sensors* **2024**, *24*, 5473. [CrossRef] [PubMed]
3. Dhanda, M.; Rogers, B.A.; Hall, S.; Dekoninck, E.; Dhokia, V. Reviewing human-robot collaboration in manufacturing: Opportunities and challenges in the context of industry 5.0. *Robot. Comput.-Integr. Manuf.* **2025**, *93*, 102937. [CrossRef]
4. Forlini, M.; Neri, F.; Carbonari, L.; Callegari, M.; Palmieri, G. Enhanced Human-Robot Collaboration through AI Tools and Collision Avoidance Control. In *Proceedings of the 2024 20th IEEE/ASME International Conference on Mechatronic and Embedded Systems and Applications (MESA), Genova, Italy, 2–4 September 2024*; IEEE: Piscataway, NJ, USA, 2024; pp. 1–7.
5. Costanzo, M.; De Maria, G.; Natale, C. Handover control for human-robot and robot-robot collaboration. *Front. Robot. AI* **2021**, *8*, 672995. [CrossRef] [PubMed]
6. Duarte, L.; Polito, M.; Gastaldi, L.; Neto, P.; Pastorelli, S. Demonstration of real-time event camera to collaborative robot communication. In *Proceedings of the The International Conference of IFToMM ITALY, Turin, Italy, 11–14 September 2024*; Springer: Berlin/Heidelberg, Germany, 2024; pp. 351–358.
7. Costanzo, M.; Natale, C.; Selvaggio, M. Visual and Haptic Cues for Human-Robot Handover*. In *Proceedings of the 2023 32nd IEEE International Conference on Robot and Human Interactive Communication (RO-MAN), Busan, Republic of Korea, 28–31 August 2023*; IEEE: Piscataway, NJ, USA, 2023; pp. 2677–2682. [CrossRef]
8. Kim, M.; Yang, S.; Kim, B.; Kim, J.; Kim, D. Human-to-Robot Handover Based on Reinforcement Learning. *Sensors* **2024**, *24*, 6275. [CrossRef] [PubMed]
9. Salinas-Martínez, Á.G.; Cunillé-Rodríguez, J.; Aquino-López, E.; García-Moreno, A.I. Multimodal Human-Robot Interaction Using Gestures and Speech: A Case Study for Printed Circuit Board Manufacturing. *J. Manuf. Mater. Process.* **2024**, *8*, 274. [CrossRef]
10. Chen, H.; Leu, M.C.; Yin, Z. Real-Time Multi-Modal Human-Robot Collaboration Using Gestures and Speech. *J. Manuf. Sci. Eng.* **2022**, *144*, 101007. [CrossRef]
11. Yu, P.; Abuduweili, A.; Liu, R.; Liu, C. Robustifying Long-term Human-Robot Collaboration through a Multimodal and Hierarchical Framework. *arXiv* **2024**, arxiv:2411.15711. [CrossRef]
12. Kheddar, H.; Himeur, Y.; Al-Maadeed, S.; Amira, A.; Bensaali, F. Deep Transfer Learning for Automatic Speech Recognition: Towards Better Generalization. *Knowl.-Based Syst.* **2023**, *277*, 110851. [CrossRef]
13. Fan, J.; Yin, Y.; Wang, T.; Dong, W.; Zheng, P.; Wang, L. Vision-language model-based human-robot collaboration for smart manufacturing: A state-of-the-art survey. *Front. Eng. Manag.* **2025**, *12*, 177–200. [CrossRef]
14. Zacharia, P.; Aspragathos, N.; Mariolis, I.; Dermatas, E. A robotic system based on fuzzy visual servoing for handling flexible sheets lying on a table. *Ind. Rob.* **2009**, *36*, 489–496. [CrossRef]
15. Macenski, S.; Soragna, A.; Carroll, M.; Ge, Z. Impact of ros 2 node composition in robotic systems. *IEEE Robot. Autom. Lett.* **2023**, *8*, 3996–4003. [CrossRef]
16. Slepicka, M.; Helou, J.; Borrmann, A. Real-time data exchange (RTDE) robot control integration for Fabrication Information Modeling. In *Proceedings of the ISARC, Proceedings of the International Symposium on Automation and Robotics in Construction, Montreal, QC, Canada, 11–15 August 2023*; IAARC Publications: Chennai, India, 2023; Volume 40, pp. 102–109.
17. Norda, M.; Engel, C.; Rennies, J.; Appell, J.E.; Lange, S.C.; Hahn, A. Evaluating the efficiency of voice control as human machine interface in production. *IEEE Trans. Autom. Sci. Eng.* **2023**, *21*, 4817–4828. [CrossRef]
18. Al-Fraihat, D.; Sharrab, Y.; Alzyoud, F.; Qahmash, A.; Tarawneh, M.; Maaita, A. Speech recognition utilizing deep learning: A systematic review of the latest developments. *Hum.-Centric Comput. Inf. Sci.* **2024**, *14*, 1–33.
19. Pallavoor, A.; Jalan, A.; Ballapur, S.C.; Kiran, S.; Anantharaman, P.; Shylaja, S. Gesture and Speech Recognition for Real-Time Multi-modal Human–Robot Interaction Using Deep Learning Based Approach. In *Proceedings of the International Conference on Computing and Machine Learning, Mt. Pleasant, MI, USA, 2–5 February 2024*; Springer: Berlin/Heidelberg, Germany, 2024; pp. 251–266. [CrossRef]
20. Khurana, D.; Koli, A.; Khatter, K.; Singh, S. Natural language processing: State of the art, current trends and challenges. *Multimed. Tools Appl.* **2023**, *82*, 3713–3744. [CrossRef] [PubMed]

21. Thakkar, H.; Manimaran, A. Comprehensive examination of instruction-based language models: A comparative analysis of mistral-7b and llama-2-7b. In *Proceedings of the 2023 International Conference on Emerging Research in Computational Science (ICERCS), Prague, Czech Republic, 3–5 July 2023*; IEEE: Piscataway, NJ, USA, 2023; pp. 1–6.
22. Neri, F.; Forlini, M.; Scoccia, C.; Palmieri, G.; Callegari, M. Experimental evaluation of collision avoidance techniques for collaborative robots. *Appl. Sci.* **2023**, *13*, 2944. [CrossRef]

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.