applied sciences

MDPI

Article

A Hybrid Architecture for Safe Human–Robot Industrial Tasks

Gaetano Lettera [1], Daniele Costa [2] and Massimo Callegari [1,*]

[1] Dipartimento di Ingegneria Industriale e Scienze Matematiche (DIISM), Universitá Politecnica delle Marche, 60131 Ancona, Italy; g.lettera@univpm.it
[2] Dipartimento di Scienze Teoriche e Applicate (DiSTA), Universitá Telematica eCampus, 22060 Novedrate, Italy; daniele.costa@uniecampus.it
* Correspondence: m.callegari@univpm.it

Abstract: In the context of Industry 5.0, human–robot collaboration (HRC) is increasingly crucial for enabling safe and efficient operations in shared industrial workspaces. This study aims to implement a hybrid robotic architecture based on the Speed and Separation Monitoring (SSM) collaborative scenario defined in ISO/TS 15066. The system calculates the minimum protective separation distance between the robot and the operators and slows down or stops the robot according to the risk assessment computed in real time. Compared to existing solutions, the approach prevents collisions and maximizes workcell production by reducing the robot speed only when the calculated safety index indicates an imminent risk of collision. The proposed distributed software architecture utilizes the ROS2 framework, integrating three modules: (1) a fast and reliable human tracking module based on the OptiTrack system that considerably reduces latency times or false positives, (2) an intention estimation (IE) module, employing a linear Kalman filter (LKF) to predict the operator's next position and velocity, thus considering the current scenario and not the worst case, and (3) a robot control module that computes the protective separation distance and assesses the safety index by measuring the Euclidean distance between operators and the robot. This module dynamically adjusts robot speed to maintain safety while minimizing unnecessary slowdowns, ensuring the efficiency of collaborative tasks. Experimental results demonstrate that the proposed system effectively balances safety and speed, optimizing overall performance in human–robot collaborative industrial environments, with significant improvements in productivity and reduced risk of accidents.

Keywords: collaborative robots; intention estimation; industrial robotics; AI-based decision-making

check for updates

Academic Editors: Gómez-Escudero Gaizka, Amaia Calleja-Ochoa and Haizea González-Barrio

Received: 20 December 2024
Revised: 15 January 2025
Accepted: 20 January 2025
Published: 24 January 2025

Citation: Lettera, G.; Costa, D.; Callegari, M. A Hybrid Architecture for Safe Human–Robot Industrial Tasks. *Appl. Sci.* **2025**, *15*, 1158. https://doi.org/10.3390/app15031158

Copyright: © 2025 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 (https://creativecommons.org/licenses/by/4.0/).

1. Introduction

1.1. Context and Motivations

Collaborative robots (cobots) are poised to revolutionize manufacturing by working alongside humans, interacting during shared tasks that combine human expertise with the precision and repeatability of robotic automation. Industry 5.0 emphasizes the integration of cobots into industrial workflows to increase productivity, safety, and operator well-being.

The research work is carried out within the framework of the DYNAMICA project [1], which proposes novel robotized assembly paradigms for SMEs that aim to evaluate the impact of cobots on production and human energy expenditure, facilitate the teaching of new tasks, and define novel safety assessment strategies. This work focuses on the third objective and addresses the challenge of balancing human safety specifications with production requirements. While current state-of-the-art methods primarily focus on limited

aspects such as human detection or just robot speed scaling strategies (see Section 3), the proposed research introduces a comprehensive approach, which combines a robust and reactive human tracking module (see Section 4), an intention estimation algorithm (see Section 5), and a robot control module (see Section 6). The reactive perception setup consists of a few retro-reflective markers worn by each human operator who moves in the collaborative space, which are continuously identified by a reliable perception system. The speed scaling algorithm is based on real-time risk assessment and adapts the robot speed according to the actual behaviors of operators, so as to allow the robot speed to increase during low-risk situations while reducing it during more hazardous situations. To do that, the algorithm computes a minimum protective distance in real time, according to current safety regulations (see Section 2) and factoring in the prediction of the operator's intentions. The proposed approach ensures operator safety while optimizing production efficiency, aiming at improving industrial profitability in collaborative workcells. Furthermore, by gradually adjusting the robot's speed, the system also considers psychological factors, ensuring comfort for operators, whether working alone or with robots. Finally, robust experimental results demonstrate the applicability of the strategy, which improves the overall performance of the HRC task for industrial environments (see Section 7).

1.2. Paper Contributions

This paper addresses solutions to balance safety specifications and production requirements in industrial HRC environments. Although there are several studies on this topic, to the best of the authors' knowledge, few, if any, comprehensively tackle all the relevant challenges or are specifically designed to align with production requirements, with the goal of maximizing overall throughput. The main paper contributions are as follows:

- The development of a reliable human detection strategy based on a robust and fast perception system able to detect multiple workers in large workspaces (see Section 4);
- The definition of a simple and fast human-tracking pipeline that considers only the points of the human body that are truly exposed to risk, avoiding complex software architectures and data processing latency;
- The development of an intention estimation predictive technique through the real-time estimation of human velocity based on a linear Kalman filter (see Section 5);
- The development of a smooth robot control system that performs a real-time risk analysis and, therefore, evaluates the actual danger of the scenario to appropriately adjust the speed of the robot and avoid unnecessary slowdowns or stops (see Section 6).

In summary, the proposed strategy uses commercial devices, such as the OptiTrack vision system, which can be easily adopted in real industrial applications. The software architecture is based on existing methods integrated with new solutions to develop a complete collaborative application. Furthermore, the system is flexible and easily applicable in a lot of industrial contexts. The three modules presented work together to better predict the operator's intentions and prevent unnecessary slowdowns, adjusting the robot's speed only when an imminent risk is detected. The theoretical novelty lies in the interaction between these modules, which optimizes both human safety and production time.

2. Safety Standards

The safety standards represent the preliminary starting point of the proposed methodology. In fact, the main focus of the robotic safety regulations is operators' safety during industrial robotic tasks. The safety standards for these applications are laid out by the International Organization for Standardization (ISO) 10218-1 [2], 10218-2 [3], and by the ISO/TS 15066 [4]. The research work implements a *pre-collision* scenario called SSM (Clause 5.5.4). As expected by the definition of SSM described in ISO/TS 15066 Clause 5.5.4.1 [4], this

collaborative scenario allows the robotic system and the operator to move simultaneously in the collaborative workspace. Risk reduction is achieved by always maintaining at least the protective separation distance, S, between the robot and the operator. During operations, the robotic system never approaches the operator beyond S. When the separation distance between the robot and the operator, d, decreases to less than S, the robot stops. When the operator moves away, the robot automatically resumes the task according to the requirements of this clause while maintaining at least the protective separation distance. When the robot system reduces its speed, S decreases accordingly.

The regulation provides additional information and further guidelines to compute S (Clause 5.5.4.2.3):

$$S = v_H T_r + v_H T_s + v_R T_r + B + C + Z_d + Z_r, \quad (1)$$

where the following definitions apply:

- v_H represents the maximum speed of the operator and it is assumed as 2000 mm/s with the option to use 1600 mm/s when $S > 500$ mm;
- v_R is the maximum robot speed;
- T_r is the time required by the machine to respond to the operator's presence;
- T_s represents the response time of the machine, which brings the robot to a safe, controlled stop;
- B is the Euclidean distance traveled by the robot while braking;
- C is the intrusion distance safety margin, which represents an additional distance, based on the expected intrusion toward the critical zone prior to the actuation of the protective equipment;
- Z_d is the operator position uncertainty (i.e., the sensor uncertainty);
- Z_r is the robot position uncertainty.

Note that Clause 5.5.4.2.1 specifies that S must be computed for all human operators within the collaborative workspace. According to Clause 5.5.4.2.2, the maximum allowable speeds, v_H and v_R, and the minimum protective separation distances, S, can be constant or variable. For constant values, the maximum allowable speed must be determined through risk assessment during the entire course of the application. For variable values, the speeds must be continuously adjusted based on the relative human–robot speeds and distances.

3. Literature Review

In the literature, two types of collaborative scenarios are identified: *post-collision* and *pre-collision* scenarios [5]. In the first case, the collision could be dangerous if the robot limits are poorly defined or when the robot is equipped with sharp tools; moreover, any collision stops the task execution, thus affecting production time. Several techniques have been proposed to minimize the energy transmitted during the contact [6] by using robots endowed with sensors for assessing force exchange when the impact occurs, e.g., force or tactile sensors [7], or by using only proprioceptive measurements in the case of industrial manipulators with closed control architecture [8]. However, industrial safety standards prohibit the use of *post-collision* systems, as they involve physical impact between the robot and the human operator before the machinery comes to a complete stop.

In contrast, a *pre-collision* approach uses exteroceptive sensors to detect the presence of humans and prevent potential collisions before they occur. This paper presents a novel strategy, which implements an SSM *pre-collision* scenario. The proposed strategy complies with the most recent perspective of Industry 5.0, as described in [9], where HRC enables seamless collaboration between humans and robots, with a focus on enhancing worker safety and comfort. Indeed, this collaboration involves robots that are capable of understanding human actions, responding to their intentions, and adapting their behavior

accordingly. To address this, three modules have been developed and integrated, each one proposing innovative technologies with respect to the state of the art: the human tracking module, the intention estimation module, and the robot control module.

3.1. The Human Tracking Module

Human tracking task requires to robustly fuse several sensors with different physical properties. For example, refs. [10,11] propose a novel CNN that processes images obtained by an innovative multimodal vision system that merges depth and thermal data into a single image. However, in the field of human–machine safety, the response time of safety systems is a very important parameter. Unfortunately, the cited work is based on image processing, which is a very computationally intensive operation; thus, the human tracking response time is unsuitable for industrial collaborative applications. Similar sensor fusion approaches are proposed in [12,13] where a data association approach is developed for HRI, based on the covariance intersection, with the aim of increasing the robustness and reliability of HRI. In this paper, however, the system classifies a very limited number of human behaviors used to determine the movements of the robot; therefore, this approach cannot be applied in a real-world environment due to the large set of natural behaviors that human operators can attain during collaborative tasks. On the contrary, the human tracking module described in Section 4 defines a human tracking pipeline that uses the OptiTrack system, which is a gold standard in motion capture and tracking systems, to consider only the points of the human body that are truly exposed to risk (e.g., hands and head), avoiding both complex software architectures and the latency of data processing, and false positives, which are not negligible in the applications mentioned above.

3.2. The Intention Estimation Module

Most systems developed by researchers for human–robot distance monitoring and intention estimation are based on point cloud-based pipelines [14] or computer vision systems [15]. The use of the OptiTrack system enables reliable and accurate detection of retro-reflective markers, which are used to track the separation distance between each human operator and the robot and are convenient for estimating any human behavior in a very fast way. More in detail, approaches based on point clouds require extremely high processing times: just think that a human cluster is on average represented by a cloud of (minimum) 500 points, while the proposed strategy can also adopt just one marker. The intention estimation module described in Section 5 considers only the points of the human body that are truly exposed to risk, avoiding complex software architectures and data processing latency. In addition, while standard SSM implementations are conservative and retain the constant value of S (Equation (1)), which corresponds to the worst-case application, the proposed intention estimation module executes a real-time evaluation of human velocity, v_H, based on an LKF with the aim of optimizing the S computation according to the actual risky situation.

3.3. The Robot Control Module

The primary goal of the proposed robot control module is to reduce the robot's pre-programmed speed in response to a real and imminent collision risk with a human worker. Achieving this objective is particularly challenging because human behavior is highly variable and unpredictable, making it difficult to reliably foresee or react to potential interactions. In the literature, a common approach uses reactive motion planning that modifies the pre-programmed path to generate a new collision-free path [16,17]. Unfortunately, in industrial environments, it is often required not to modify the robot's pre-programmed path because it can involve the violation of some of the production constraints. On the

other hand, the research work [18] presents a general framework for computer-aided risk assessment that uses a temporal logic language. It discusses the SSM method as a measure for risk reduction, but the robot's speed is reduced based on predefined thresholds for the protective distance S. More in detail, the control logic uses a predefined warning zone and a danger zone around the robot base and continuously monitors the proximity of the operator: if the operator enters the warning zone, the speed of the robot becomes a prevention default value; if the operator gets too close, entering the dangerous area, the robot stops completely. The control module described in Section 6 develops a smooth robot control system that performs real-time risk analysis and then evaluates the actual danger of the scenario to adequately adjust the robot speed. The entire range from 0% to 100% of the robot speed override is considered, and not a limited number of predefined thresholds, so as to avoid unnecessary slowdowns or stops, taking into account both safety and production requirements.

4. Human Tracking Module

4.1. Experimental Setup Configuration

The experimental setup demonstrates the advantages of using the OptiTrack [19] vision system for the workspace monitoring, i.e., a reduction of occlusions and false positive detections. The use of a multi-camera setup, covering the entire workspace, ensures that the human operators are tracked from multiple angles, significantly reducing the chances of occlusion as the system can continue tracking through any visual obstruction by relying on alternative camera views. The proactive nature of the system comes from its ability to adjust tracking based on real-time data from multiple cameras, which helps avoid false positive detections caused by partial occlusions or errors in individual camera views. In cases of full occlusions, the system may experience temporary tracking loss, but once the occlusion is cleared (e.g., the human moves out of the blocked area), the system quickly re-establishes tracking. Figure 1 shows an image of the experimental setup, acquired by one of the cameras. The laboratory environment is approximately 80 sqm large and simulates a real industrial environment in which there may be multiple operators, multiple robots, desks, and work equipment. The whole space is monitored by 12 cameras that use infrared light to capture retro-reflective markers, also called *rigid bodies* (RB), as shown in the top left of the image. It is here assumed that the part of the human body that is most representative of the operator's motion is the hand; therefore, rigid bodies are attached to operators' gloves. The fundamental requirement for the OptiTrack system to be able to distinguish one rigid body from another is the uniqueness of the number of markers and the relative distances between them.

OptiTrack is considered the gold standard in motion capture and tracking systems due to its high accuracy (0.2 mm), high frame rate (from 20 Hz up to 1000 Hz), reliability, and versatility. Its advanced software, called Motive, enables easy integration with other systems, making it a trusted solution for complex applications of advanced robotics in collaborative robotic tasks for industrial, large spaces.

4.2. Networking Configuration

The diagram shown in Figure 2 represents how to network the necessary devices together. A computer capable of running Motive 3.1.0 software is the slave PC and it is connected to the 12 cameras through the OptiTrack switch to serve as the Motive PC. A second computer running Ubuntu 20.04 is the master PC on which ROS2 executes the software pipeline. It is connected to a router with an internet connection, separated from the OptiTrack switch. The router connects a second Network Interface Card for the Motive PC, the Ubuntu PC, and the UR5e robot controller through ethernet cables.

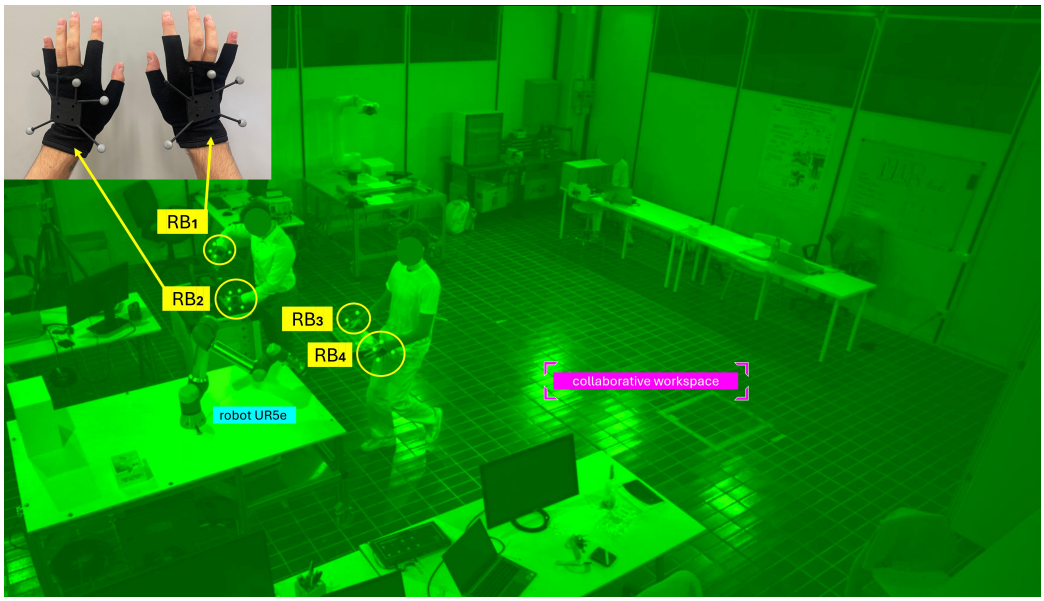

Figure 1. Experimental setup of the collaborative workspace.

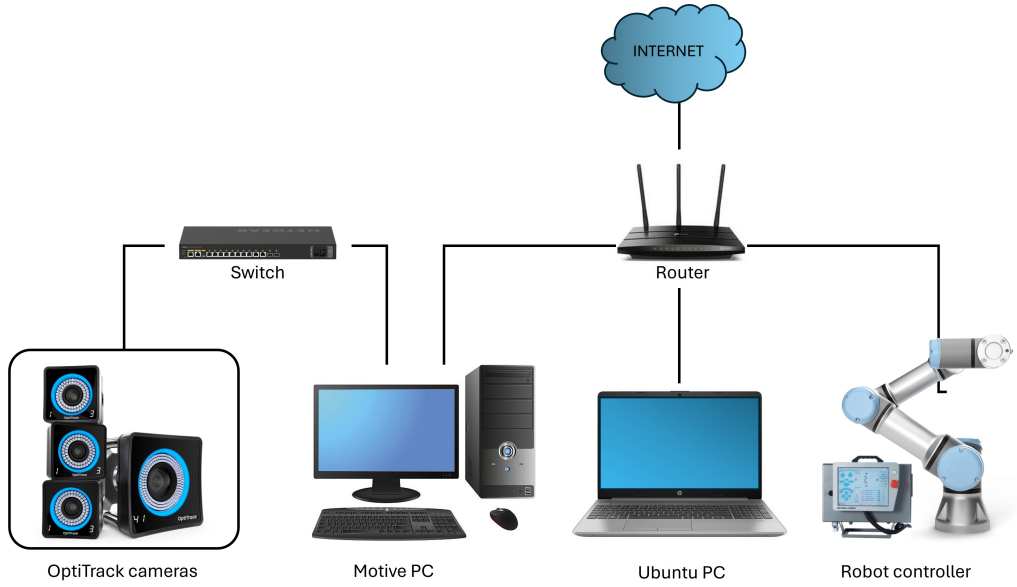

Figure 2. Network configuration of the proposed hybrid architecture.

Motive offers multiple options to stream tracking data onto external applications in real time. For the proposed architecture, tracking data were streamed in Live mode at 20 Hz through the NatNet SDK, which enables the construction of a custom client–server application to stream captured data. In the field of human–machine cooperation, the response time of safety systems is an important feature, so details are provided below. The 20 Hz rate refers to the frequency at which the OptiTrack system provides tracking data to the robot. It is important to note that the frequency control of the robot used for the experimental results (see Section 7) was 500 Hz. This means that the robot can adjust its movements and react to control commands at a faster rate than the tracking data are being updated. The combination of the robot control frequency rate and the vision data processing rate impacts the robot's ability to respond to speed adjustments and stop thresholds. Even though the OptiTrack system's frequency can be increased up to 1000 Hz, in the described experiments, the minimum frequency of 20 Hz (comparable to the frequency used by other vision systems, e.g., Intel RealSense) was used to highlight how smooth the system operates even with the lowest configurable performance settings.

4.3. System Calibration

The OptiTrack system needs two different calibration tools to ensure accurate motion tracking and system alignment: the CW-500 tool and the CS-400 tool. Figure 3 shows the calibration tool; each one is used for specific calibration purposes: the calibration with the CW-500 is performed for environment identification, while the calibration with the CS-400 is mandatory for reference system definition.

The CW-500 tool is a calibration tool that is manually moved throughout the working volume for wanding acquisition, as shown in Figure 3 left and center. This process allows the Motive 3.1.0 software to dynamically gather the poses of the three retro-reflectors attached to the tool from all the cameras. The captured data help the system to identify the working environment by defining the tracking boundaries and optimizing the intrinsic configurations of the cameras to ensure accurate coverage of the entire space. Reference [20] details the calibration procedure that has been followed.

Figure 3. The CW-500 tool (**left** and **center**) and the CS-400 tool (**right**) used for OptiTrack calibration.

The CS-400 tool is a Custom Calibration Square used to define the reference frame of the OptiTrack system, T_{opt}, against which each rigid body pose is accurately expressed. For the proposed application, the tool was positioned on the edge of the robot workbench, as shown in the Figure 3, on the right. This choice was made to calibrate the base frame of the robot, T_{base}, intuitively and accurately with respect to T_{opt}, which provides sensory vision data. In this way, the kinematic chain of the robot, contained in the URDF file that simulates the environment in ROS2, is accurately connected to the reference frame of the vision system, avoiding positioning errors of the rigid bodies. To do that, four markers were positioned on the robot base; then, a rigid body was created including these markers in order to obtain the transformation matrix T_{base}^{opt} of the centroid of the rigid body itself. Finally, the original OptiTrack drivers for ROS2 were edited in order to send to the Ubuntu PC the tracked rigid bodies expressed in the T_{base} reference system, i.e., $T_{RB_i}^{base}$.

4.4. Rigid Body Tracking

At this point, the last configuration step of the Motive 3.1.0 software is the definition of the rigid bodies. The number of rigid bodies to be used for the application depends on the number of gloves to be tracked. Assuming that a maximum of five human operators are allowed to access the monitored work environment, ten rigid bodies should be prepared, thus identified from the software for the first time and finally registered with an ID. The strong advantage of this step is that, during the collaborative task, it is not mandatory that all the registered rigid bodies are actually present in the scene: in case of absence, the system simply will not acquire their pose. This means that the rigid body array that is sent to the Ubuntu PC, and which contains the pose of each tracked rigid body, has a dynamic size depending on how many rigid bodies are present in the scene at that moment. In other words, when a new operator enters the collaborative environment with pre-registered gloves, these will be instantly added to the sent array; in the same way, they will disappear

if an operator leaves the monitored environment. This process is completely automatic and does not need any system reconfiguration.

Figure 4 shows the result of the human tracking module: the four rigid bodies tracked by the OptiTrack system through the Motive PC are sent to the Ubuntu PC, where they have been acquired, processed, and visualized into the simulated environment RViz (right) as green spheres.

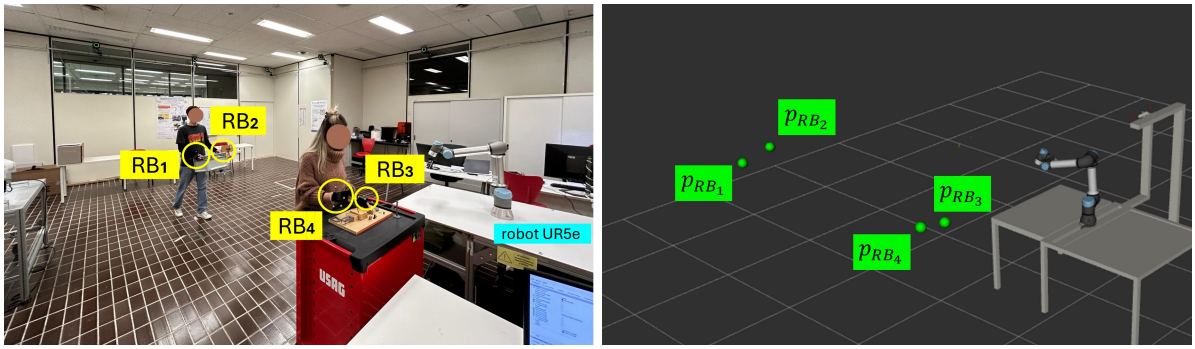

Figure 4. Four rigid bodies tracked by the OptiTrack system (**left**) and visualized in ROS2 (**right**).

5. Intention Estimation Module

5.1. Estimation of the Operator Velocities

Another key aspect of the HRC problem is intention estimation (IE), which involves predicting human movements. In this work, a linear Kalman filter (LKF) was implemented to estimate the operator's velocity by combining OptiTrack sensor acquisitions with a motion model. It predicts the future position and velocity of each rigid body present in the collaborative workspace and updates these predictions with new sensor readings, improving accuracy. This real-time estimation, applied to Equation (1), allows the robot to anticipate the operators' movements, adjust its actions, and avoid potential collisions, ensuring safe and efficient interaction, as explained in Section 6. The advantage of using an LKF is its ability to smooth trajectory data and continuously refine velocity estimates, which enhances the robot's responsiveness and safety in dynamic, industrial environments.

The implemented LKF solves the problem of estimating the state of a discrete-time process governed by the equations

$$x_{k+1} = \begin{bmatrix} I_3 & \Delta t I_3 \\ O_3 & I_3 \end{bmatrix} x_k + w_k, \tag{2}$$

$$y_k = \begin{bmatrix} I_3 & O_3 \end{bmatrix} x_k + n_k \tag{3}$$

where Δt is the sampling time, I_3 and O_3 are the identity and zero matrices of size 3×3, respectively; w and n are the process and measurement noises with covariance matrices W and N, respectively. Finally, x is the state vector of the system, i.e., the position and the velocity of the rigid body $x = \begin{bmatrix} p_{RB_i}^T & \dot{p}_{RB_i}^T \end{bmatrix}^T$, and the measured output y is a vector containing the coordinates of the point P_{RB_i} described in Section 4.4. The covariance matrix N is experimentally estimated, while the covariance matrix Q has been chosen as

$$Q = \begin{bmatrix} I_3 \Delta t^2 & O_3 \\ O_3 & Q_2 \end{bmatrix} \tag{4}$$

where Q_2 quantifies the uncertainty on the velocity dynamics (assumed constant) of the state equations. Note that the LKF equations used in this work are the standard ones and are therefore not included here for the sake of brevity.

Figure 5 on the left shows the estimated velocities of the four rigid bodies appearing in Figure 4: two belonging to an operator holding a toolbox and the other two belonging to another operator carrying out an assembly operation. The developed LKF software computes and displays the three components of the estimated velocity (rgb arrows) and the resulting speed (yellow arrow). Their length is scaled according to their own module: v_{RB_1} and v_{RB_2} represent a human walking and they are significantly higher than v_{RB_3} and v_{RB_4} representing the speeds used for the assembly task.

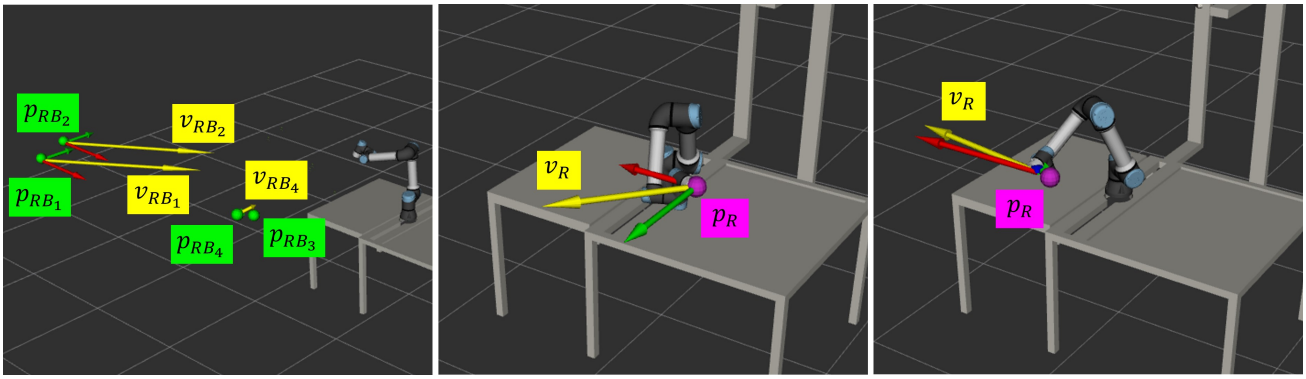

Figure 5. Human operator (**left**) and robot (**right**) velocities estimated through the LKF.

5.2. Estimation of the Robot Velocity

In order to obtain a practical solution to minimize production time, the proposed algorithm monitors the robot end-effector position, p_R, by using a common convention based on the knowledge of the robot joint angles q [rad]. In fact, p_R is computed by retrieving the transform between the robot base frame, T_{base}, and the end-effector frame, T_{ee}, which are updated in real time by the robot controller and drivers.

On the other hand, the equation implemented to compute the end effector velocity, v_R, is based on the change in the end-effector position over time. More in detail, the change in position is calculated between current and previous positions of p_R, while the time difference, dt, is calculated between the current and previous timestamps of the transform. The velocity components in the x, y, and z directions are computed by dividing the position differences by the time difference:

$$v_{R_x} = \frac{p_{R_x}^{curr} - p_{R_x}^{prev}}{dt}, \qquad v_{R_y} = \frac{p_{R_y}^{curr} - p_{R_y}^{prev}}{dt}, \qquad v_{R_z} = \frac{p_{R_z}^{curr} - p_{R_z}^{prev}}{dt} \tag{5}$$

Figure 5 (center and right) shows the robot end-effector positions and velocities during its motion. The yellow arrow represents the speed, calculated as the Euclidean norm (magnitude) of the velocity vector in 3D space. The velocity vector is represented by the three components v_{R_x}, v_{R_y}, and v_{R_z} in rgb convention.

5.3. Relative Speed Computation

The last step of the proposed IE module is to calculate the relative speeds between the humans and the robot: they represent how fast the humans and robot are moving towards each other (or away from each other), which is crucial for computing the protective distance S, according to ISO/TS 15066 [4].

Human-directed speed, $v_{RB_i}^d$, measures how fast the i-th rigid body is moving towards (or away from) the robot. It is calculated by projecting the rigid body velocity, v_{RB_i}, onto the direction from the RB_i to the robot, expressed by the unit vector $d_{RB_i \to R}$. Thus, $v_{RB_i}^d$ is the dot product of the rigid body velocity, v_{RB_i}, and the direction from the human to the robot, $d_{RB_i \to R}$: it measures how much the human operator velocity is aligned with the

direction towards the robot. A positive value indicates that the human is moving towards the robot, while a negative value indicates that they are moving away.

$$d_{RB_i \to R} = \frac{p_R - p_{RB_i}}{\|p_R - p_{RB_i}\|}, \qquad v^d_{RB_i} = v^d_{RB_i} \cdot d_{RB_i \to R} \tag{6}$$

On the other hand, the robot-directed speed, v^d_R, measures how fast the robot is moving towards (or away from) the human. It is calculated by projecting the robot velocity, v_R, onto the direction from the robot to the selected rigid bodies, RB_i. The equations are similar to the previous ones:

$$d_{R \to RB_i} = \frac{p_{RB_i} - p_R}{\|p_{RB_i} - p_R\|}, \qquad v^d_R = v^d_R \cdot d_{R \to RB_i} \tag{7}$$

These directed speeds are crucial for determining the protective distance, S_p, between each human and the robot, which depends on how fast both the human and the robot are closing the distance between each other. If $v^d_{RB_i}$ is large and positive, it means the human is moving quickly towards the robot, which might reduce the time until the robot needs to stop or react. Similarly, if v^d_R is large and positive, the robot is moving quickly towards the human, which might reduce the time to avoid the human.

6. Robot Control Module

6.1. Human–Robot Separation Distances

A crucial step for the implementation of the SSM collaborative task is the computation of the separation distance array between the robot and each human operator. As explained above, the developed pipeline identifies in real time both the robot end-effector point, p_R, and the tracked rigid body, p_{RB_i}. The separation distances between the robot and each i-th rigid body are computed by means of the Euclidean norm:

$$d_i = \|p_R - p_{RB_i}\| = \sqrt{(p_{R_x} - p_{RB_{xi}})^2 + (p_{R_y} - p_{RB_{yi}})^2 + (p_{R_z} - p_{RB_{zi}})^2} \tag{8}$$

6.2. Minimum Protective Separation Distance Computation

As suggested in ISO/TS 15066 [4], the minimum protective distance S is computed for each rigid body (human) by the Formula (1). Bringing the equation back into the application context described in this paper, in which there are i human operators, S_i is based on various factors such as the velocities of the robot and the i-th rigid body, the reaction time of the robot, the robot's stopping distance, and the uncertainties in positions. The protective distance ensures that the robot is kept at a safe distance from the human to prevent any collision. The pipeline computes S_i using several components:

- S_{RB_i}: contribution from the rigid body speed towards the robot;
- S_R: contribution from the robot's reaction time;
- S_S: contribution from the robot's stopping distance;
- Safety margins: intrusion distance (C) and uncertainties in robot and human positions (Z_r and Z_d).

The equation for the minimum protective distance for each rigid body (human) is

$$S_i = S_{RB_i} + S_R + S_S + C + Z_d + Z_r \tag{9}$$

where the following definitions apply:

- S_{RB_i}: the contribution from the human velocity is calculated as

$$S_{RB_i} = v^d_{RB_i} \cdot (T_s + T_r) \tag{10}$$

  where $v^d_{RB_i}$ is the directed speed of the i-th rigid body towards the robot, and T_s and T_r are the stopping and reaction times of the robot.
- S_R: the contribution from the robot velocity is calculated as

$$S_R = v^d_R \cdot T_r \tag{11}$$

  where v^d_R is the directed speed of the robot towards the human.
- S_S: the robot stopping distance is a constant value B.
- C: intrusion distance, a small buffer to avoid accidents.
- Z_r: robot position uncertainty.
- Z_d: rigid body position uncertainty.

6.3. Robot Speed Override Computation

The last step of the developed robot control module is to process the information about the separation distances, d_i, and the minimum separation distances, S_i, related to each i-th tracked rigid body according to the regulations. More in detail, these values have been compared to compute the robot speed override, or scaling factor, k, which satisfies the SSM principle. A key aspect of the strategy is also to ensure smooth and stable changes in robot speed.

The speed override is computed by checking the relationship between the maximum scaling factor (S_{max}) and the corresponding distance, d_i. The speed override is defined as follows:

$$k = \begin{cases} 1.0 & \text{if } d_i \geq \Delta S \times S_{max} \\ 0.0 & \text{if } d_i < S_{max} \text{ or if } d_i < d_{min} \\ \frac{d_i - S_{max}}{\Delta S \times S_{max} - S_{max}} & \text{otherwise} \end{cases} \tag{12}$$

where S_{max} is the maximum value in the S_i array, ΔS is a constant scaling factor that determines the range of distances for which the speed override is computed (it was chosen as 2) and d_{min} is a minimum distance threshold that has been introduced to stop the robot if the operator is extremely close to the robot itself (it was chosen as 0.30 m). The first condition represents a totally safe condition for the human operators; thus, the robot can move to its maximum speed while the second condition satisfies the regulation ISO/TS 15066 [4]. For intermediate distances, the proposed strategy uses a linear interpolation to gradually shape the scaling factor between 0.0 and 1.0 based on the distance.

7. Experimental Results

This section presents the experimental results to evaluate the practical significance of the proposed methodology. The experiments were conducted using the same setup for two different tasks. The setup, designed for an industrial environment, features a Universal Robot, the UR5e, a cobot engaged in a pick-and-place operation, specifically, a cyclic pre-programmed trajectory. This application uses a ROS2 industrial architecture, which is connected to the robot according to the networking configuration described in Section 4.2. The first task represents an inspection task in which a technician approaches the robot to carry out some visual assessments, while the second task represents a co-kitting task in which two operators work close to the robot.

The proposed algorithm, except the Motive 3.1.0 software, runs on a standard desktop PC running Ubuntu 20.04 LTS with an Intel Core i7-8700K CPU at 3.70 GHz, 12 GB of RAM

and an nVidia GeForce GT 740 GPU with 12 GB of memory. The Motive 3.1.0 software, runs on a separate PC running Windows 11 Enterprise with an Intel Core i9-13900KF CPU at 3.0 GHz, 128 GB of RAM and an nVidia RTX A4000 GPU.

The accompanying videos that show the full experiments are available at the following links: inspection task (https://www.dropbox.com/scl/fi/v0mkt2fn0z23lfjkagojm/inspection-task-MDPI-applsci-3412264.mp4?rlkey=t6r5webo1j6ythdkep6yrd1ef&dl=0, accessed on 21 January 2025) and co-kitting task (https://www.dropbox.com/scl/fi/2ba9w31dfmjy70q8vhnbe/cokitting-task-MDPI-applsci-3412264.mp4?rlkey=x2x3os2i9922epr8iqd4bjwz1&dl=0, accessed on 21 January 2025).

7.1. Inspection Task

7.1.1. Task Description

The scene simulates a robot executing a pre-planned path at a given nominal speed, and a human operator who enters the collaborative workspace to perform some inspection checks. The operator handles a notepad, close to the robot, at different distances and approaches it randomly. Figure 6 shows the main snapshots of the task execution evidencing the performances of the proposed algorithm. The full demo is provided in the accompanying video. More in detail, at the beginning, the operator is stationary and the robot moves at its maximum predefined speed (STEP 1). At $t = 8$ s the operator walks towards the robot station for about 7 s (STEP 2), then stops in front of the desk and takes some notes (STEP 3). After this phase, the operator gets close to the robot less than the minimum distance, d_{min}; thus, the robot stops immediately at $t = 21$ s (STEP 4). Finally, he steps away for a few seconds (STEP 5) before leaving the robot station at about $t = 35$ s (STEP 6).

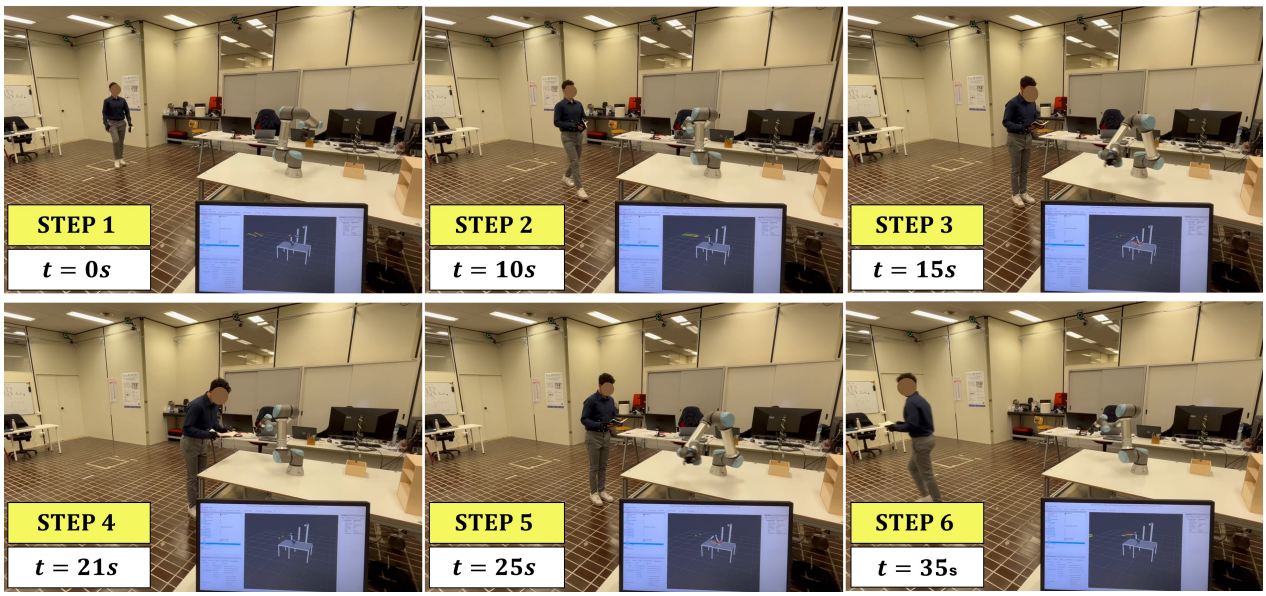

Figure 6. Snapshots of the inspection task experiment.

7.1.2. Task Results

Results are shown in Figure 7. The top graph compares the speeds v_{RB_1} (dark green line, operator's right hand), v_{RB_2} (light green line, operator's left hand), and v_R (yellow line, robot end effector), while the bottom graph represents the computed distances, which are processed to compute the robot speed scaling factor, k (black line). The minimum protective separation distances S_{RB_1} (dark blue line) and S_{RB_2} (light blue line) are compared with the respective distances d_{RB_1} (purple line) and d_{RB_2} (violet line) and the minimum allowed distance d_{min} (red line).

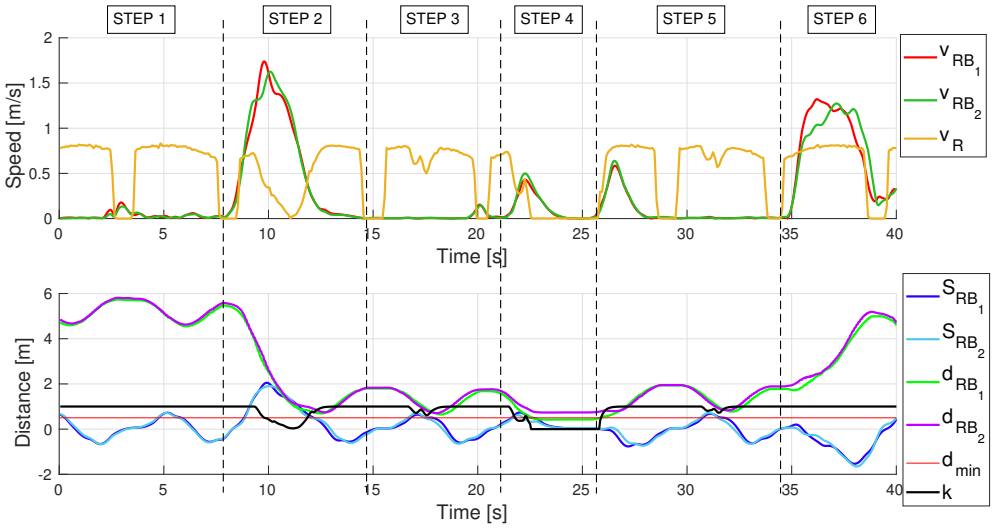

Figure 7. Results of the inspection task experiment.

More in detail, during STEP 1, the operator is stationary ($v_{RB_1} \approx v_{RB_2} \approx 0$ m/s), so S_{RB_i} only depends on v_R: the separation distances d_{RB_1} and d_{RB_2} are much greater than the minimum permitted, so the robot moves at its maximum speed ($k = 1$).

STEP 2 starts at about 8 s: the operator walks towards the robot station; thus, the speeds v_{RB_i} increase up to a peak of 1.8 m/s. During this motion, since the direction of motion of the operator is opposite to the direction of motion of the robot, S increases according to the contribution of v_{RB_i}. Also, the human–robot distances d_{RB_i} decrease, until they become very similar: the system supposes a possible risk for the operator, so it slows down the speed of the robot from approximately 9 s to 11.5 s, until the robot stops ($k = 0$) because $S_{RB_2} > d_{RB_2}$. When the operator stops in front of the robot, $v_{RB_i} \approx 0$ m/s and the robot resumes its speed gradually (k increases).

STEP 3 highlights that, even if the operator is almost stationary, the robot still slows down slightly when it passes in front of him to prevent possible risky situations ($t \approx 17$ s).

During STEP 4, the operator approaches the robot at a distance lower than the minimum allowed ($t \approx 21$ s, $d_{RB_2} < d_{min}$), so the robot immediately stops ($k = 0$). After that, the operator backs away for a few seconds (STEP 5) and the behavior is similar to STEP 3.

Finally, in STEP 6, at $t \approx 35$ s, the operator moves away from the robot station. Differently from STEP 2 in which they move at opposite directions that causes a decrease in k, in this case, S decreases even if v_{RB_i} increases because the human operator directions of motions are the same, so the v_{RB_i} contribution is negative in the S_{RB_i} computation: the collaborative scenario is strongly safe and $k = 1$.

7.2. Co-Kitting Task

7.2.1. Task Description

The scene simulates a robot executing a pre-planned path at a given nominal speed, and two human operators who enter the collaborative workspace to perform some co-kitting tasks. Figure 8 shows the main snapshots of the task execution, which demonstrate the performances of the applicability of the proposed strategy in multi-operator scenarios. The full demo is provided in the accompanying video. More in detail, in the beginning, the operators are not present in the collaborative workspace; thus, the robot moves at its maximum predefined speed (STEP 1). At $t = 6$ s the operator A, wearing gloves numbered RB_3 and RB_4, walks towards the robot station for about 6 s (STEP 2), then stops in front of the assembly desk and starts some activities (STEP 3). After this phase, the operator B, wearing gloves numbered RB_1 and RB_2, carries a toolbox to the work table and leaves the

scene (STEP 4). At this point, the operator A takes the toolbox and brings it to his assembly desk, getting close to the robot at less than the minimum distance, d_{min}; thus, the robot slows down and then stops at about $t = 29$ s (STEP 5). Finally, operator A leaves the scene, while operator B enters the workspace (STEP 7) to replace a drip tray on the robot table (STEP 8).

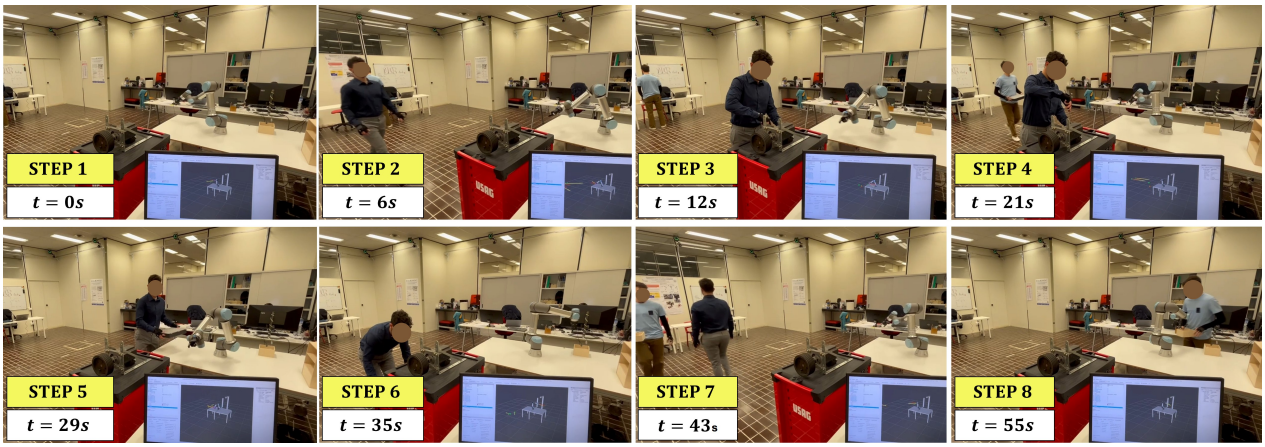

Figure 8. Snapshots of the co-kitting task experiment.

7.2.2. Task Results

Results are shown in Figure 9. To facilitate understanding of the graph, only the maximum values of the data related to the pairs of rigid bodies attached to each operator are shown, considering that the behavior of the two hands of each operator is almost similar during the task. The top graph compares the speeds of the operators $v_{max_{1,2}}$ (light green line, operator B), $v_{max_{3,4}}$ (blue line, operator A), and v_R (yellow line, robot end effector), while the bottom graph represents the computed distances that are processed to compute the robot speed scaling factor, k (black line). The minimum protective separation distances are $S_{max_{1,2}}$ for operator B (orange line) and $S_{max_{3,4}}$ for operator A (light blue line). They are compared with the respective distances, $d_{max_{1,2}}$ (purple line) and $d_{max_{3,4}}$ (violet line) and the minimum allowed distance d_{min} (red line).

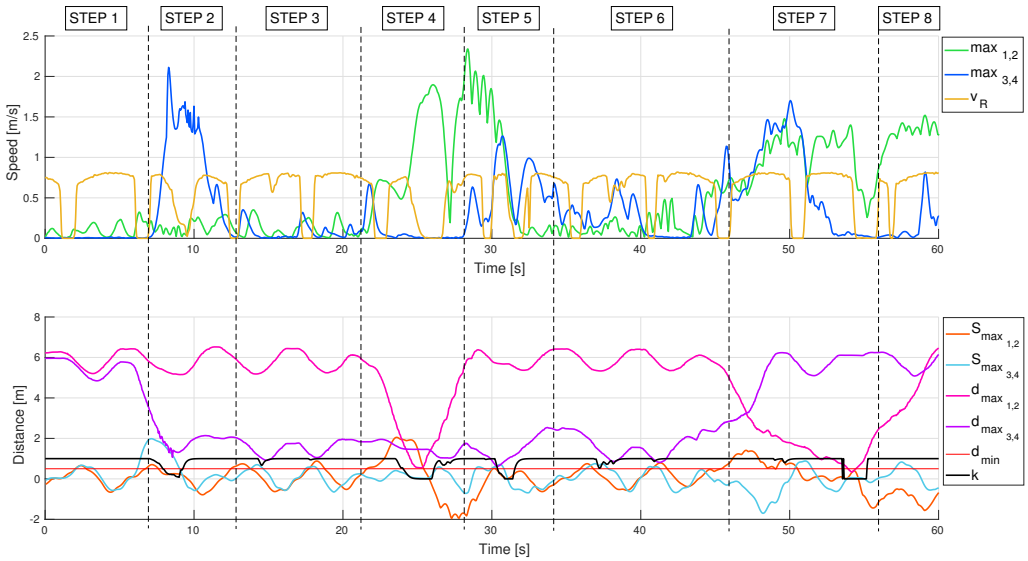

Figure 9. Results of the co-kitting task experiment.

More in detail, during STEP 1, the operators are not present in the collaborative workspace; thus, the robot moves at its maximum speed ($k = 1$).

STEP 2 starts at about 6 s: operator A walks towards the robot station; thus, the speed $v_{max_{3,4}}$ increases up to a peak of about 2.0 m/s. During this motion, $S_{max_{3,4}}$ increases and the distance $d_{max_{3,4}}$ decreases, until they become very close: the robot slows down and then it stops ($k = 0$) at about 9 s when $S_{max_{3,4}} > d_{max_{3,4}}$. When operator A decelerates until he arrives in front of his assembly desk, $v_{max_{3,4}} \approx 0$ m/s and k gradually increases.

As for the inspection task, STEP 3 highlights that, even if the operator is almost stationary, the robot still slows down slightly when it passes in front of him to prevent risks ($t \approx 12$ s).

During STEP 4, operator B carries a toolbox to the robot work table and leaves the scene: $S_{max_{1,2}}$ becomes more than $d_{max_{1,2}}$; thus, the robot stops. It resumes the motion only when the operator reverses the direction of motion and moves away, at about $t = 25$ s.

The robot also stops in STEP 5, when operator A takes the toolbox and its distance becomes less than d_{min}: the robot slows down and then it stops at about $t = 31$ s.

Finally, operator A leaves the scene at about $t = 46$ s while, at the same time, operator B enters the workspace (STEP 7). Note that, even if both $v_{max_{1,2}}$ and $v_{max_{3,4}}$ increase, $S_{max_{1,2}}$ increases because operator B is approaching the robot while $S_{max_{3,4}}$ decreases because operator A is moving away. During the last step of the experiment, STEP 8, operator B replaces a drip tray on the robot table and $d_{max_{3,4}} < d_{min}$, thus $k = 0$. Only when he leaves the scene does the robot resume its trajectory at its maximum speed.

8. Conclusions

The paper presents a hybrid ROS2-based architecture addressing the challenges encountered when integrating collaborative robots in industrial settings. Specifically, the research introduces a reliable human detection and tracking system using the OptiTrack perception system, alongside an intention estimation algorithm and a robot control module. The integration of these components enables the implementation of a robust SSM collaborative scenario which can be used in industrial robotic workcells. The developed speed scaling algorithm performs a real-time risk assessment, adjusting robot speed based on operators' behaviors. It increases robot speed in low-risk situations and reduces it in higher-risk scenarios by calculating the minimum protective separation distance, considering current safety regulations and by evaluating operator intention predictions. The great advantage of the proposed technology is that the approach ensures operator safety while optimizing production efficiency, aiming at improving the industrial profitability in collaborative workcells. The experimental results demonstrate improved performance in balancing safety and speed. However, the selected OptiTrack tracking system is highly robust in dealing with partial occlusions, but it cannot completely eliminate the challenge of full occlusions where the operator is entirely out of view (i.e., blocked from view by other objects) and this is a drawback of the proposed technology. The technology can be used for any type of collaborative activity, e.g., inspection [21] or assembly [22]. This work represents the starting point for the development of a ready-to-use package that can be easily adapted to different types of robots and vision systems, e.g., depth cameras such as Orbec or Intel RealSense can replace the OptiTrack system, to adapt the cost of the workcell to the application's needs. The proposed control strategy can be integrated with advanced real-time trajectory replanning techniques, adaptable to unstructured environments. In addition, the SSM safety index could be dynamically correlated to the ergonomics of the operator, e.g., analyzing his posture, integrating exoskeletons, and tracking the entire human skeleton to correlate the presumed risk to the most exposed part of the human body,

according to the actual regulations. Furthermore, object recognition and manipulation algorithms can be developed for real applications.

Author Contributions: Conceptualization, G.L.; Software, G.L.; Validation, D.C.; Writing—original draft, G.L.; Writing—review & editing, D.C.; Supervision, M.C.; Funding acquisition, M.C. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by PIANO NAZIONALE DI RIPRESA E RESILIENZA (PNRR); Missione 4 "Istruzione e Ricerca"—Componente C2; Investimento 1.1, "Fondo per il Programma Nazionale di Ricerca e Progetti di Rilevante Interesse Nazionale (PRIN)"—D.D. n. 104 del 2 febbraio 2022, Bando PRIN 2022—CUP I53D23001930006—Cod. prog: 2022XXH9JZ_002/ERC: PE8—Project title: "DYNamic Assessment and Mitigation of the Impact of Collaborative Applications (DYNAMICA)".

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The data presented in this study are available in the article.

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

Abbreviations

The following abbreviations are used in this manuscript:

CNN	Convolutional Neural Network
DYNAMICA	DYNamic Assessment and Mitigation of the Impact of Collaborative Applications
HRC	Human–Robot collaboration
HRI	Human–Robot Interaction
IE	intention estimation
ISO	International Organization for Standardization
LKF	Linear Kalman Filter
RB	Rigid Body
ROS	Robot Operating System
SMEs	Small and Medium Enterprises
SoA	State-of-the-Art
SSM	Speed and Separation Monitoring
TS	Technical Specification
URDF	Unified Robot Description Format

References

1. The DYNAMICA Project. 2024. Available online: https://www.cobotics.it/ (accessed on 21 January 2025).
2. *ISO 10218-1:2011*; Robots and Robotic Devices—Safety Requirements for Industrial Robots. Part 1: Robots. Technical Report; International Organization for Standardization: Geneva, Switzerland, 2011.
3. *ISO 10218-2:2011*; Robots and Robotic Devices—Safety Requirements for Industrial Robots. Part 2: Robot System and Integration. Technical Report; International Organization for Standardization: Geneva, Switzerland, 2011.
4. *ISO/TS 15066:2016*; Robots and Robotic Devices—Collaborative Robots. Technical Report; International Organization for Standardization: Geneva, Switzerland, 2016.
5. Heinzmann, J.; Zelinsky, A. Quantitative Safety Guarantees for Physical Human-Robot Interaction. *Int. J. Robot. Res.* **2003**, *22*, 479–504. [CrossRef]
6. Haddadin, S.; Albu-Schaffer, A.D.L.A.H.G. Collision Detection and Reaction: A Contribution to Safe Physical Human-Robot Interaction. In Proceedings of the 2008 IEEE/RSJ International Conference on Intelligent Robots and Systems, Nice, France, 22–26 September 2008. [CrossRef]
7. Cirillo, A.; Ficuciello, F.; Natale, C.; Pirozzi, S.; Villani, L. A Conformable Force/Tactile Skin for Physical Human–Robot Interaction. *IEEE Robot. Autom. Lett.* **2016**, *1*, 41–48. [CrossRef]
8. Geravand, M.; Flacco, F.; Luca, A.D. Human-robot physical interaction and collaboration using an industrial robot with a closed control architecture. In Proceedings of the 2013 IEEE International Conference on Robotics and Automation, Karlsruhe, Germany, 6–10 May 2013. [CrossRef]

9. Lou, S.; Hu, Z.; Zhang, Y.; Feng, Y.; Zhou, M.; Lv, C. Human-cyber-physical system for industry 5.0: A review from a human-centric perspective. *IEEE Trans. Autom. Sci. Eng.* **2024**, 1–18. [CrossRef]
10. Costanzo, M.; De Maria, G.; Lettera, G.; Natale, C.; Perrone, D. A Multimodal Perception System for Detection of Human Operators in Robotic Work Cells. In Proceedings of the 2019 IEEE International Conference on Systems, Man and Cybernetics (SMC), Bari, Italy, 6–9 October 2019; pp. 692–699. [CrossRef]
11. Costanzo, M.; De Maria, G.; Lettera, G.; Natale, C. A Multimodal Approach to Human Safety in Collaborative Robotic Workcells. *IEEE Trans. Autom. Sci. Eng.* **2022**, *19*, 1202–1216. [CrossRef]
12. Ong, K.S.; Hsu, Y.H.; Fu, L.C. Sensor fusion based human detection and tracking system for human-robot interaction. In Proceedings of the 2012 IEEE/RSJ International Conference on Intelligent Robots and Systems, Vilamoura-Algarve, Portugal, 7–12 October 2012.
13. Kaplan, H. *Practical Applications of Infrared Thermal Sensing and Imaging Equipment*; SPIE Publications: Bellingham, WA, USA, 2007.
14. Lettera, G.; Bajrami, A.; Costa, D.; Callegari, M. Dynamic Safety Evaluation and Risk Mitigation Strategies for Collaborative Kitting. In *New Trends in Mechanism and Machine Science*; Springer: Cham, Switzerland, 2024; pp. 286–293. [CrossRef]
15. Flacco, F.; Kroger, T.; Luca, A.D.; Khatib, O. A depth space approach to human-robot collision avoidance. In Proceedings of the 2012 IEEE International Conference on Robotics and Automation, Saint Paul, MN, USA, 14–18 May 2012. [CrossRef]
16. Zhang, P.; Jin, P.; Du, G.; Liu, X. Ensuring safety in human-robot coexisting environment based on two-level protection. *Ind. Robot. Int. J.* **2016**, *43*, 264–273. [CrossRef]
17. Lippi, M.; Marino, A. Safety in human-multi robot collaborative scenarios: A trajectory scaling approach. *IFAC-PapersOnLine* **2018**, *51*, 190–196. [CrossRef]
18. Vicentini, F.; Pedrocchi, N.; Beschi, M.; Giussani, M.; Iannacci, N.; Magnoni, P.; Pellegrinelli, S.; Roveda, L.; Villagrossi, E.; Askarpour, M.; et al. PIROS: Cooperative, Safe and Reconfigurable Robotic Companion for CNC Pallets Load/Unload Stations. In *Bringing Innovative Robotic Technologies from Research Labs to Industrial End-Users: The Experience of the European Robotics Challenges*; Caccavale, F., Ott, C., Winkler, B., Taylor, Z., Eds.; Springer International Publishing: Cham, Switzerland, 2020; pp. 57–96.
19. The OptiTrack Website. 2024. Available online: https://www.optitrack.com/ (accessed on 21 January 2025).
20. The OptiTrack Calibration Procedure. 2024. Available online: https://docs.optitrack.com/motive/calibration (accessed on 21 January 2025).
21. Ulrich, M.; Lux, G.; Jürgensen, L.; Reinhart, G. Automated and cycle time optimized path planning for robot-based inspection systems. *Procedia CIRP* **2016**, *44*, 377–382. [CrossRef]
22. Foumani, M.; Gunawan, I.; Smith-Miles, K.; Ibrahim, M.Y. Notes on feasibility and optimality conditions of small-scale multifunction robotic cell scheduling problems with pickup restrictions. *IEEE Trans. Industr. Inform.* **2015**, *11*, 821–829. [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.