POLITECNICO DI TORINO
Repository ISTITUZIONALE

A Practical and Effective Layout for a Safe Human-Robot Collaborative Assembly Task

Original
A Practical and Effective Layout for a Safe Human-Robot Collaborative Assembly Task / Scimmi, L.S., Melchiorre, M., Troise, M., Mauro, S., Pastorelli, S.P.. - In: APPLIED SCIENCES. - ISSN 2076-3417. - ELETTRONICO. - 11:4(2021), pp. 1-14. [10.3390/app11041763]

Availability:
This version is available at: 11583/2891179 since: 2021-04-13T08:48:15Z

Publisher:
MDPI

Published
DOI:10.3390/app11041763

Terms of use:

This article is made available under terms and conditions as specified in the corresponding bibliographic description in the repository

Publisher copyright

(Article begins on next page)

21 November 2026

applied sciences

MDPI

Article

A Practical and Effective Layout for a Safe Human-Robot Collaborative Assembly Task

Leonardo Sabatino Scimmi, Matteo Melchiorre *, Mario Troise, Stefano Mauro and Stefano Pastorelli

Department of Mechanical and Aerospace Engineering, Politecnico di Torino, 10129 Torino, Italy; leonardo.scimmi@polito.it (L.S.S.); mario.troise@studenti.polito.it (M.T.); stefano.mauro@polito.it (S.M.); stefano.pastorelli@polito.it (S.P.)

* Correspondence: matteo.melchiorre@polito.it; Tel.: +39-0110-9069-42

Abstract: This work describes a layout to carry out a demonstrative assembly task, during which a collaborative robot performs pick-and-place tasks to supply an operator the parts that he/she has to assemble. In this scenario, the robot and operator share the workspace and a real time collision avoidance algorithm is implemented to modify the planned trajectories of the robot avoiding any collision with the human worker. The movements of the operator are tracked by two Microsoft Kinect v2 sensors to overcome problems related with occlusions and poor perception of a single camera. The data obtained by the two Kinect sensors are combined and then given as input to the collision avoidance algorithm. The experimental results show the effectiveness of the collision avoidance algorithm and the significant gain in terms of task times that the highest level of human-robot collaboration can bring.

Keywords: collaborative robotics; collaborative task; collision avoidance; trajectory planning; vision system; motion tracking

check for updates

Citation: Scimmi, L.S.; Melchiorre, M.; Troise, M.; Mauro, S.; Pastorelli, S. A Practical and Effective Layout for a Safe Human-Robot Collaborative Assembly Task. *Appl. Sci.* **2021**, *11*, 1763. https://doi.org/10.3390/app11041763

Academic Editor: Carlo Canali
Received: 21 January 2021
Accepted: 10 February 2021
Published: 17 February 2021

Publisher's Note: MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Copyright: © 2021 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

In the last years, collaborative robotics is becoming a more and more relevant topic. Historically, to ensure safety, the workers have been kept physically separated from the robot systems [1]. Nowadays, the basic idea of collaborative robotics is to ensure safe robotic systems to allow the removal of the fences that divide the working area of the robots from that of human operators, so that robots and humans can work together to accomplish a task.

The introduction of a new class of robot designed to work side-by-side with humans, named "cobots", increases the chances for a successful collaboration [2,3]. The existing cobots are mostly anthropomorphic manipulators, which are equipped with passive and active safety features. The passive features are design choices that intrinsically reduce the risk of injury in the case of undesired contact, such as lightweight materials and rounded edges. The most advanced cobots mount also a set of sensors and dedicated software packages to actively handle unexpected contacts by stopping the robot if the joint torques or the contact forces exceed the defined thresholds. However, these features, that come with the cobot out of the box, may not guarantee safe collaborative robotics. In fact, the workspace sharing can introduce new hazards and risks for the workers that must be taken into account when designing the application [4,5].

The design of the collaborative robotics systems is principally related to the task. In fact, according to the International Federation of Robotics (IFR), one may require a different level of interaction depending on the human and robot activities [6]. The most common examples of collaborative robotics provide a shared workspace where the robot and employee work sequentially. This type of application is also identified as sequential collaboration. On the other hand, the most technically challenging application requires a

responsive collaboration, which means that the human and robot work at the same time in the shared workspace and they have to coordinate to accomplish the task safely [7].

Regardless of the level of collaboration, if the human and robot workspaces intersect, the industrial application must be designed according to the related standards [8]. In particular, two different approaches can be implemented to guarantee safety: The power and force limiting (PFL) and the speed and separation monitoring (SSM). The PFL combines an accurate risk assessment with risk reduction measures so that the contact between the operator and the cobot does not result in harm to the operator. Therefore, in addition to the passive safety design methods, one can actively control the energy of the moving parts of the robot by limiting forces, torques, velocities or momentum within a given range. In contrast, the SSM does not specify limitation to the robotic system as long as it maintains a protective separation distance from the operator. In practice, if the operator is within the robot workspace, the SSM translates into online robot controlling to generate alternative collision-free paths. In this case, the use of human tracking systems is needed for the minimum distance monitoring and collision avoidance algorithms shall be implemented to update in real-time the robot trajectory.

Concerning the tracking device, wearable systems and 3D vision systems are the best alternatives [9]. The first one would limit the human operator's movements with suits and sensors, while the use of cameras does not require wearing dedicated components. On the other hand, the data obtained from vision systems are sensitive to problems such as occlusions of the sensor and variable light conditions [10].

The signals coming from sensors are used to apply control strategies on the robot. Two basic approaches are possible: "Global" (or "long-range") and "local" (or "short-range") methods [11,12]. In global methods, a complete knowledge of the workspace is required. The geometry and position of the obstacles are known and are given as inputs to the path planner algorithms. The planner produces a path from the initial state to the target state that avoids the obstacles. On the other hand, local strategies use local spatial information and do not require a global knowledge of the workspace. The local control algorithms can modify online the trajectories of the robot to avoid moving obstacles, but they do not guarantee the reaching of the target. Local strategies are preferred when dealing with dynamic environments, since fast algorithms are needed to update the robot trajectory.

The various aspects that have been introduced indicate that collaborative robotics is a complex scenario. More precisely, the highest level of collaboration, which has been identified as "responsive", is a synthesis of the whole set composed of cobot, operator, tracking system, and control strategy for the safe accomplishment of a task.

This work presents a practical and effective layout for responsive collaboration that goes in the direction of the SSM approach and that can potentially increase productivity in semi-automatic assembly processes where the human and robot tasks can be parallelized. The layout and the control strategy described here combine the major outcomes of the previous works [13–21]. The layout is composed of an assembly table, a collaborative robot UR3 from Universal Robots mounted in front of the operator to facilitate the range of movements [22], a vision system made of two Microsoft Kinect v2 and three computers. A 3D visual servo control based on optimized skeleton tracking and repulsive velocities is implemented to drive the robot away from the operator. In particular, the proposed layout is designed to exploit the advantages of the highest level of collaboration in a specific assembly task. The idea is to employ a human worker in the added value operations of the task, i.e., assembly, while letting a robot collect the parts to be assembled and deliver them on the assembly table, where the human operates. Thanks to the collision avoidance algorithm, the operator can access the shared workspace anytime, without being influenced by the presence of the robot.

Since the paper focuses on the assembly task, the major results among the most recent or promising studies are reported. Due to the severity of the standards, the real-world application is mainly limited to sequential collaboration. For example, a collaborative cell for the assembly of an homokinetic joint is presented in [23]. This is a significant example of

how collaboration can improve the employee health in terms of musculoskeletal disorders, but the case is limited to a sequential task so that the robot stops if a camera detects the human hand in the proximity of the robot tool. Another case study focuses on the introduction of an UR3 robot in a pocket assembly cell, to mitigate the physical fatigue of the worker [24]. However, the operator has the role of a supervisor and a laser scanner triggers a safety stop if the human accesses the robot workspace.

Since safety stops may affect productivity, researchers are moving towards a better integration of the robot, the vision system, and the control strategies for a higher level of collaboration. In the following, some examples of the most recent control strategies are reported that have been applied to pick-and-place, which is relevant for the purpose of this work since it is intended as a subtask of the assembly process. The authors of [25] proposed a method which is based on a depth camera and dynamic safety zones. The strategy in [25] activates stopping trajectories only if the distance between the human and robot bounding volumes exceeds the threshold defined by the standards. An improved sensor fusion algorithm based on a multiple depth camera and real-time collision avoidance by means of an optimization problem are presented in [26]. The results in [26] show that the robot slightly deviates from the nominal path if the predicted occupancy of the operator gets close to the robot planned trajectory. A more precise robot trajectory generation which can deal with the human intrusion within the workspace is shown in [27]. However, in [27] the robot can only follow waypoints across a trapezoidal path.

Other researchers focus on assembly workspace layout or task optimization. The case study of the battery cell assembly proposed in [28], which is conducted comparing a laser scanner with radio wave sensors for motion tracking, shows that the relative position of the human and robot in the working area may affect the runaway motion of the robot. Concerning task optimization, in [29] a method which allows predicting the human activity pattern is implemented for the assembly of an integrated circuit. The results in [29], which are obtained by using a single depth camera for human tracking, prove that pattern prediction leads to better cycle times in sequential collaboration.

This work distinguishes from the state of the art since it describes a practical solution to effectively deal with human-robot responsive collaboration by using multiple vision sensors and a local control strategy to ensure the safety of the human worker. None of the works previously discussed present a hardware/software setup as the one described in this paper, which has been adopted to carry out a responsive collaborative task. Furthermore, the choice of using a UR3 robot, Kinect v2 cameras, and implementing a collision avoidance based on artificial potential fields was made in order to use devices and algorithms well known and robust, giving the readers the possibility to replicate the setup and develop collaborative robotics applications. Moreover, this work focuses on assembly tasks where the operator is designated for the added value activities.

The paper is organized as follows: The description of the task and the experimental layout are presented in Section 2. Section 3 describes the implemented control strategy. The experimental tests and the discussion of the results are reported in Section 4. Finally, Section 5 includes the conclusions and future works.

2. Design of Collaborative Layout and Application

The case study is an assembly cell which receives three parts to be assembled as input and produces an assembly made of the same parts. The parts consist of cubes of different sizes, while the assembly is obtained by sticking the cubes together, as shown in Figure 1. The entire task can be divided in two main operations: Collecting the components to be assembled and assembling them. Concerning the first operation, the components are supposed to be stored so that a robot can easily pick them up. The second operation is supposed to be too complex or too variable for a full automation solution, so that the human presence is required. For this reason, the operator should be able to take the parts to be assembled from a certain area which must be a portion of the robot workspace, since the robot has the role to provide the parts to the operator. With this aim, the workspace of the

robot is divided in two zones, as shown in Figure 2. The parts are stored in the "pick zone", where no human presence is expected. In the "collaborative zone", where the human and robot can potentially come to contact as their workspaces intersect, the operator takes the components to be assembled.

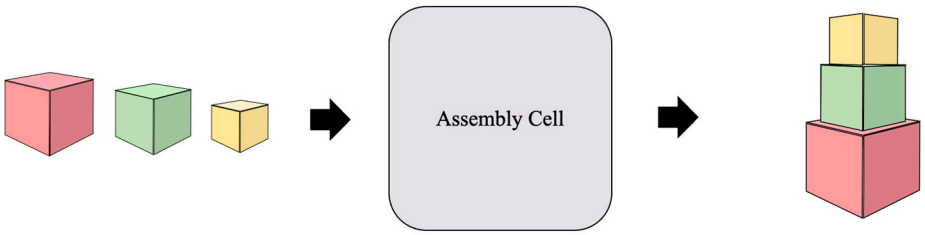


Figure 1. Assembly process.

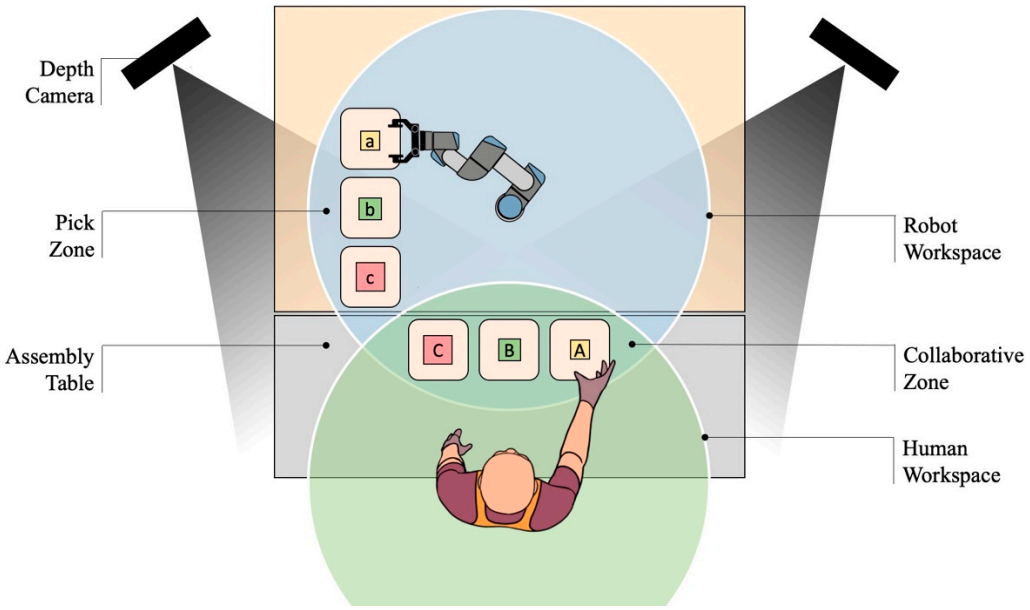


Figure 2. Collaborative assembly cell.

The human worker and robot perform seven operations in parallel to carry out the assembly task, as reported in Figure 3. The task, which is also identified as a cycle, starts when the operator takes the cube positioned in station A. As the human hand enters in the collaborative zone, the robot task begins by replacing the cube in A with a new one from station a. In parallel, the operator puts the glue on the first cube, takes the second cube by station B, and prepares both cubes to be glued. The next step for the robot is taking a cube from station B and placing it in station B, to replace the one taken by the operator. Meanwhile, the human picks up the last cube in station C and assembles the tower of cubes. Finally, the robot replaces the cube in station C and returns in the home position, while the worker checks the edges of the assembly to remove the exceeding glue before delivering. Concerning the human task, the phases in which he takes the components happen in the collaborative zone, while the other phases are carried out on the assembly table, outside of the robot workspace. On the other hand, the "pick" stages of the robot task identify the robot motion in the pick zone, i.e., outside the collaborative zone, while the "place" stages identify the robot motion within the collaborative zone. The execution of these operations involves a responsive collaboration. In fact, as the human and robot carry out the subtasks in parallel, the operator can access the collaborative zone at any time, even if the robot is placing a component.

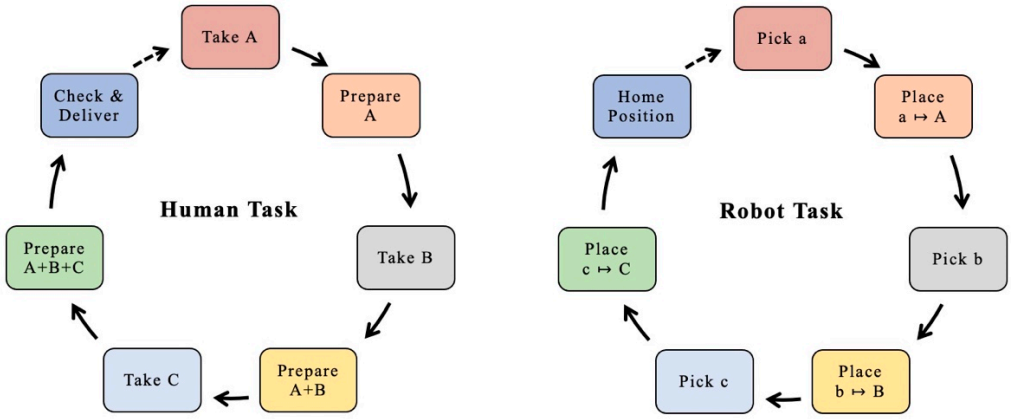


Figure 3. Task phases.

The laboratory setup of the collaborative assembly cell is visible in Figure 4. Three PCs, two Microsoft Kinect v2, a router, and a collaborative robot UR3 from Universal Robots equipped with a Robotiq gripper 2F-85 are used. The three PCs are connected via Ethernet network and communicate using the TCP/IP protocol. The implemented algorithms are written in MATLAB environment. The two Kinect sensors are displaced at 1.5 m from the human, so that their field of view covers the entire workbench from different angles. Each Kinect can feedback depth data up to 30 Hz [30]. Two PCs are connected to the Microsoft Kinect v2 cameras, to obtain the data acquired by each sensor. The third PC is connected to the controller of the UR3 robot. Further information related to the control strategy will be given in the next section.

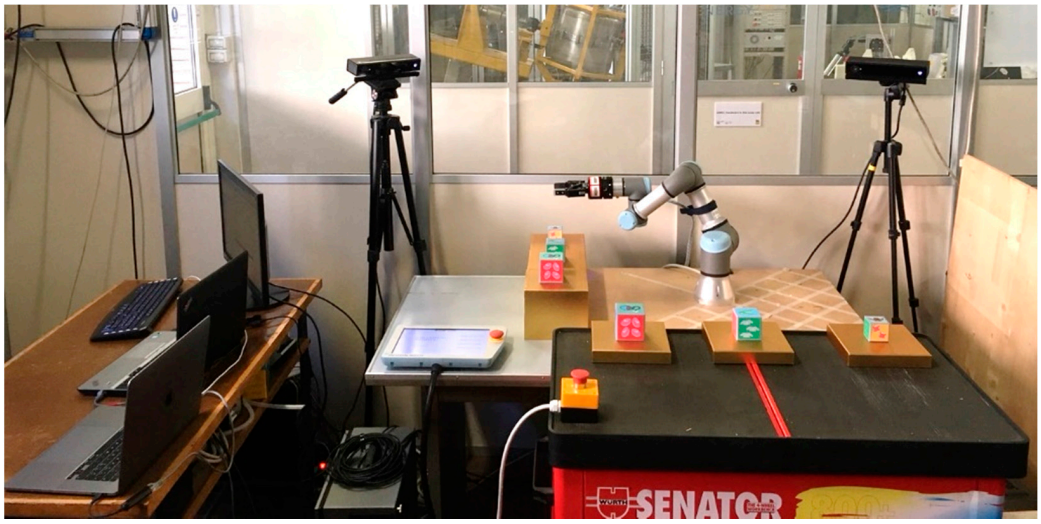


Figure 4. Experimental layout of the collaborative assembly cell.

3. Robot Control Strategy

The robot tool center point (TCP) motion is planned by combining planar trajectories that trigger and terminate at strategical nodes. When the robot moves above the stations of the components from the pick zone to the collaborative zone, the TCP planned trajectory is planar at $z = 0.3$ m, where z is measured from the robot mounting table. To grasp or release the cubes, the robot follows a vertical path. The strategical nodes are defined as the intersection points between the plane $z = 0.3$ m and the vertical paths. For example, in Figure 5 the main directions of the robot planned trajectory in the collaborative zone are shown, together with the x and y axis of the robot base. Regardless of the cube to be placed, the robot approaches the collaborative zone always from point p_C, then it

navigates towards the station to be refilled. Once the component has been replaced, the robot drives backwards through the same path, returns in p_c, and leaves the collaborative zone. However, due to the human presence, the planned trajectory may deviate from the rectilinear paths. In fact, if the operator takes a component and gets close to the robot, which is going to replace another one, the risk of a collision occurs. Then, the robot planned trajectory is modified through the collision avoidance algorithm, which is discussed below.

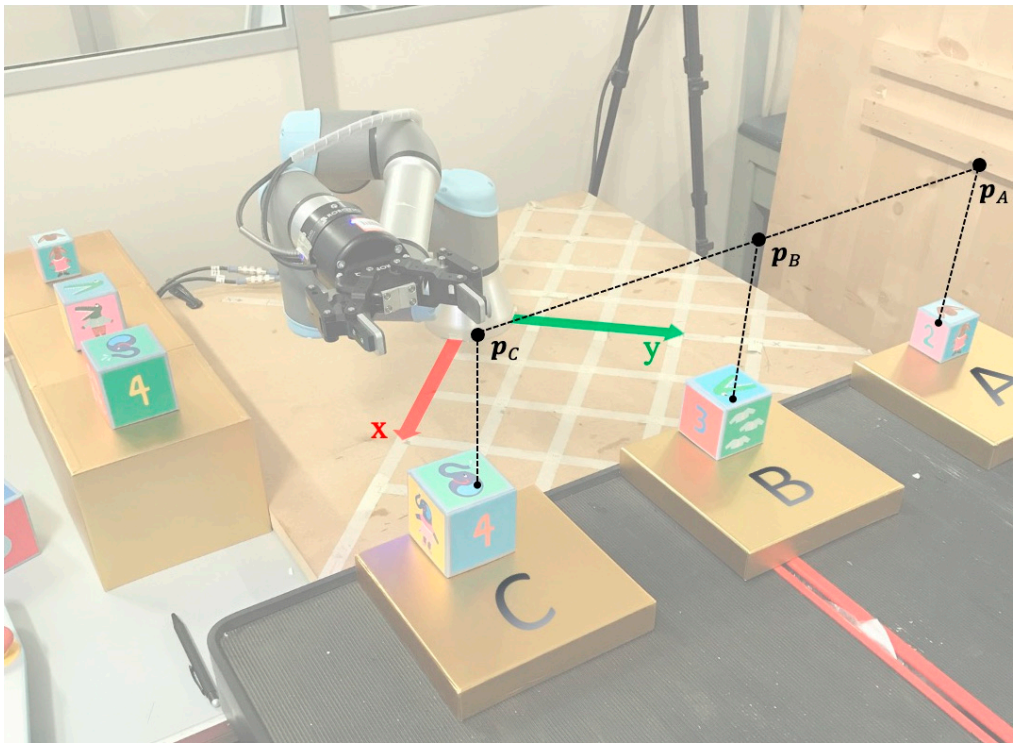


Figure 5. Planned trajectory in the collaborative zone.

Each Kinect gives as output the position of 25 points on the human body. These points will be indicated hereafter as skeleton [30,31]. The multiple sensors layout allows overcoming the problems related to occlusions of a single camera [10]. However, the use of multiple sensors shifts the attention into the data fusion problem. One of the common techniques developed to merge the raw data obtained from the sensors consists of using a predefined skeleton fitted model [32], while other techniques synthetize the skeletons generated by a couple of Kinects to better estimate the human joints position [33]. In the system proposed here, an optimization problem is solved to properly combine the two skeletons. Further information related to the skeletons merging process and to the spatial matching are reported in [19,33].

The two PCs connected to the Kinect sensors work in a master-slave architecture. The PC-Master is responsible for the data acquisition synchronization of the sensors, sending a trigger signal to the PC-Slave. The skeleton obtained by the PC-Slave is then sent to the PC-Master, that manages the combination of the skeletons acquired by the two Kinects. The optimized skeleton is calculated with a frequency of 30 Hz. The optimized skeleton is one of the inputs of the collision avoidance algorithm. The control algorithm runs in the third PC, hereafter called PC-UR. This PC receives the skeleton from the PC-Master and feedback data of the robot from the controller of the UR3. The feedback data related to the robot are described in [34].

The collision avoidance algorithm is based on the artificial potential fields approach [12,35–37] and permits avoiding collisions between the upper part of the human body and all the moving links of the robot. Repulsive velocities are associated to the operator and attractive velocities to the desired task. The repulsive and attractive velocities

are then added up to obtain a resultant velocity, which is the output of the collision avoidance algorithm. In this way, the implemented control algorithm can modify the planned trajectory to move the robot away from the human operator, but the task is preserved and accomplished when human-robot collisions are no longer possible. Both the repulsive velocities and the attractive ones are calculated in the operative space. To avoid undesired contacts between the worker and the moving links of the manipulator, nine points, divided in five clusters, have been assumed within the robot, as seen in Figure 6.

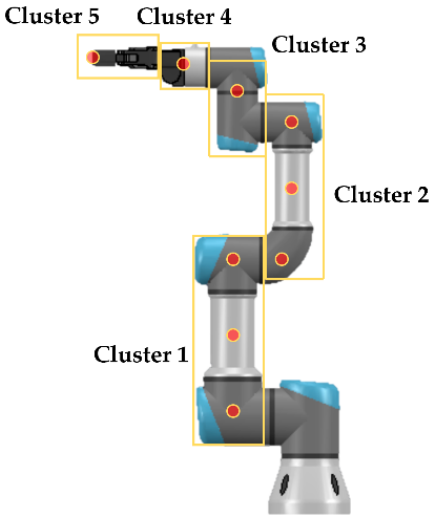


Figure 6. Points and clusters considered on the UR3 robot.

The positions of these reference points are calculated by direct kinematics, exploiting the feedback joint degrees of freedom provided by the UR controller. The distances between the human joints and the points on the UR3 robot are then evaluated using the *knnsearch* algorithm [38]. This algorithm, given as inputs of two sets of points X and Y (in this case the points are respectively the position vectors of the joints of the optimized skeleton and of the points of the clusters) finds the nearest neighbor in X for each query point in Y and gives as outputs the distances. Given the human-robot distances $\boldsymbol{d_i}$, the repulsive velocities $\boldsymbol{v_{rep,i}}$ are calculated for each i cluster using the following equations:

$$V_i = \frac{V_{MAX}}{\left(1 + e^{(|\boldsymbol{d_i}| \times \frac{2}{\rho} - 1)\alpha}\right)} \tag{1}$$

$$\boldsymbol{v_{rep,i}} = V_i \frac{\boldsymbol{d_i}}{|\boldsymbol{d_i}|} \tag{2}$$

where V_{MAX} is the maximum value of the magnitude of the repulsive velocity, ρ is the reference distance, and α is a shape factor. In this work, the magnitude V_i has been modelled as in Figure 7. This curve is obtained with V_{MAX} = 0.3 m/s, ρ = 0.4 m, and α = 5, which have been used during the experimental phase. Concerning the value of ρ (that determines the distance between the control points on the robot and the skeleton that triggers the repulsive effect), even if it may seem quite conservative, takes into account the real dimensions of the human body and robot. In fact, $\boldsymbol{d_i}$ are point-to-point distances, so the human body and robot dimensions have been included in the reference distance ρ. Moreover, ρ also considers the position uncertainties of the operator and robot resulting from measurement tolerances of the sensing devices according to what is required by the standards. Concerning V_{MAX} and α, their values were chosen considering the TCP velocity during the pick-and-place phases, to obtain evasive movements of the robot from the operator which were smooth and seamless.

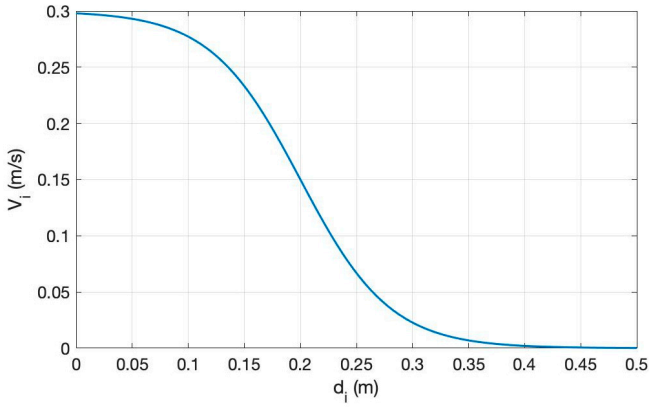


Figure 7. Magnitude of the repulsive velocity V_i versus the human-robot distance.

It is important to highlight that the values of the three parameters V_{MAX}, ρ, and α used in the experimental phase result in a safe collision avoidance in the sense that they can prevent the collision with the operator, even if the algorithm described in this section does not fully respect the equations related to the SSM presented in [8]. In the technical specification, the value of the minimum human-robot distance is not constant and depends on several terms. The velocity of the human operator is one of the terms that modify the value of the minimum distance. In this work, a constant reference distance ρ was adopted and the influence of the human velocity was taken into account choosing sufficiently precautionary values of the three parameters involved in the calculation of the repulsive velocities.

Concerning the repulsive velocities associated to the points of Clusters 1 and 2, the distances $\boldsymbol{d_i}$ are the minimum human-robot distances of each cluster. Instead, concerning the TCP of the robot, which is Cluster 5 in Figure 6, the repulsive velocity $\boldsymbol{v_{rep,TCP}}$ is added to the attractive velocities associated to the task to obtain the resultant velocity $\boldsymbol{v_{res,TCP}}$:

$$\boldsymbol{v_{res,TCP}} = \boldsymbol{v_{rep,TCP}} + \boldsymbol{v_{task}} + \boldsymbol{K} \times \boldsymbol{e} \tag{3}$$

where $\boldsymbol{v_{task}}$ is the velocity vector associated to the planned task, $\boldsymbol{K}$ is a diagonal matrix, and $\boldsymbol{e}$ is the error vector, which is the difference between the desired pose of the TCP and the actual one. The orientation error is calculated considering the axis-angle convention [39]. If the minimum human-robot distance is lower than a given threshold ρ_{min} even in presence of the repulsive velocities, the robot is stopped and it will move again when the minimum distance is greater than ρ_{min}.

The information useful to properly control the UR3 robot is the joint velocities vector $\dot{\boldsymbol{q}}$. Therefore, the velocities from the operative space have to be mapped to the joint space. The joint velocities vector $\dot{\boldsymbol{q}}$ is calculated as follows:

$$\dot{\boldsymbol{q}} = \boldsymbol{J^{-1}} \boldsymbol{v_{res,TCP}} + \sum_{\boldsymbol{i=1}}^{\boldsymbol{4}} \boldsymbol{J_{part,i}^{-1}} \boldsymbol{v_{rep,i}} \tag{4}$$

where $\boldsymbol{J}$ is the geometric Jacobian matrix associated to the TCP and $\boldsymbol{J_{part,i}}$ is the partial Jacobian of each i cluster. The joint velocities vector $\dot{\boldsymbol{q}}$ is calculated with a frequency of 62.5 Hz and then sent to the controller of the UR3 to properly move the robot, in order to accomplish the task avoiding collisions with the human worker. Custom functions developed by the authors permit the communication between the PC-UR and controller of the robot, as reported in [17]. A schematic representation of the control strategy is shown in Figure 8. The scheme is divided in two parts: In the upper part, the data flow related to the acquisition of the movements of the human operator is reported, while the calculation process of the vector $\dot{\boldsymbol{q}}$ is shown in the lower part of the scheme. Concerning the movements acquisition, the PC-Master and the PC-Slave firstly obtain the data related

to the skeletons acquired by the two Kinect sensors. Then, the PC-Slave sends the skeleton to the PC-Master, which generates the optimized skeleton. The position vectors of the joints of the optimized skeleton are finally sent to the third PC. Meantime, the PC-UR has calculated the position of the points of the five clusters by direct kinematics, given the feedback data from the controller of the robot. The position vectors of the clusters' points and of the skeleton joints are the inputs of the collision avoidance algorithm, that gives as output the joint velocities vector $\dot{q}$. The joint velocities vector is sent to the UR-Controller every 0.016 ms.

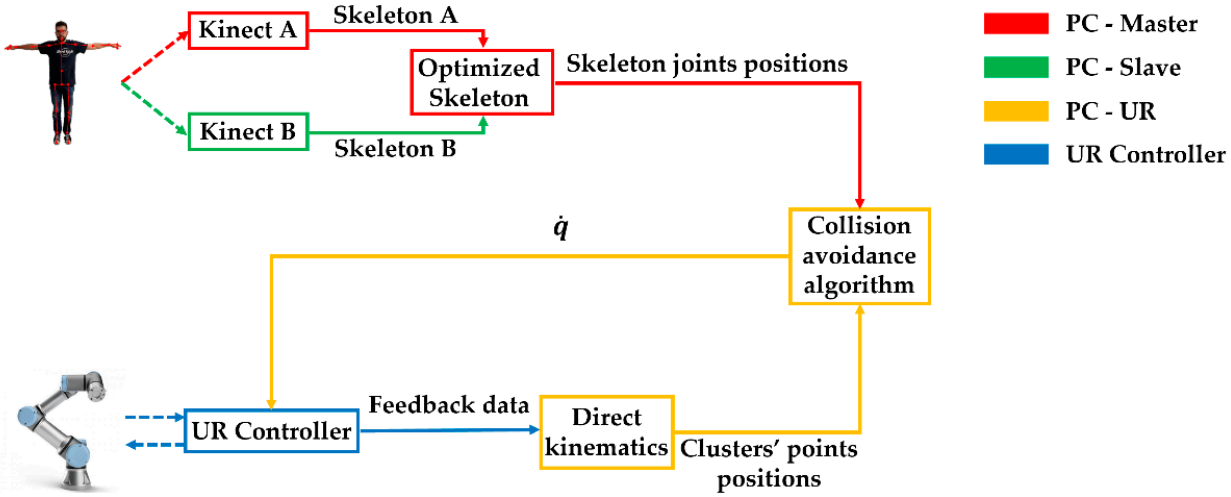


Figure 8. Scheme of the control strategy. Data and algorithms are reported in different colors to indicate which hardware device they are associated with.

4. Results and Discussion

The effectiveness of the proposed collaborative layout is evaluated in terms of safety and productivity. Concerning the safety, the experimental tests prove that the collision avoidance algorithm can quickly drive the robot through alternative trajectories in order to avoid the human. Figure 9 shows an example of collision avoidance that happens when the robot is placing a↦A and the operator is taking B (for a more detailed overview of the entire assembly cycle the reader is referred to the video in "Supplementary Materials"). For each frame, the MATLAB calculation environment is reported in the robot base reference frame. In the latter, the robot links are represented as blue lines, together with the nine control points. The optimized skeleton of the human upper body is colored in red and made of 15 significant joints. The robot planned trajectory is also shown as the rectilinear dashed path connecting p_C to p_A, while the solid line is the trace of the actual collision avoidance trajectory. The robot approaches the collaborative zone from point p_C and starts following the rectilinear planned path towards p_A, while the human is still applying the glue on the first cube (Frame 1). Then, the operator finishes preparing A (Frame 2) and takes B, while the robot is crossing the relative station, so that the robot reacts backwards to avoid the human hand (Frame 3). As the hand retracts, the robot returns on the planned path (Frames 4 and 5) and reaches p_A, while the human is preparing A+B (Frame 6).

The collision avoidance trajectory is analyzed in Figure 10. The robot planned trajectory is also shown with the dashed line, together with the strategical points. To give an idea of the repulsive effect, the repulsive velocity vectors acting on the TCP are reported on the actual path at significant samples. The repulsive effect increases nearby p_B, which is the point above station B where the human takes the cube. Here, the actual path bends backwards due to the presence of the human hand. The operator takes the cube within a time interval of 1 s, so the repulsive effect is localized only on a limited portion of the collision avoidance path.

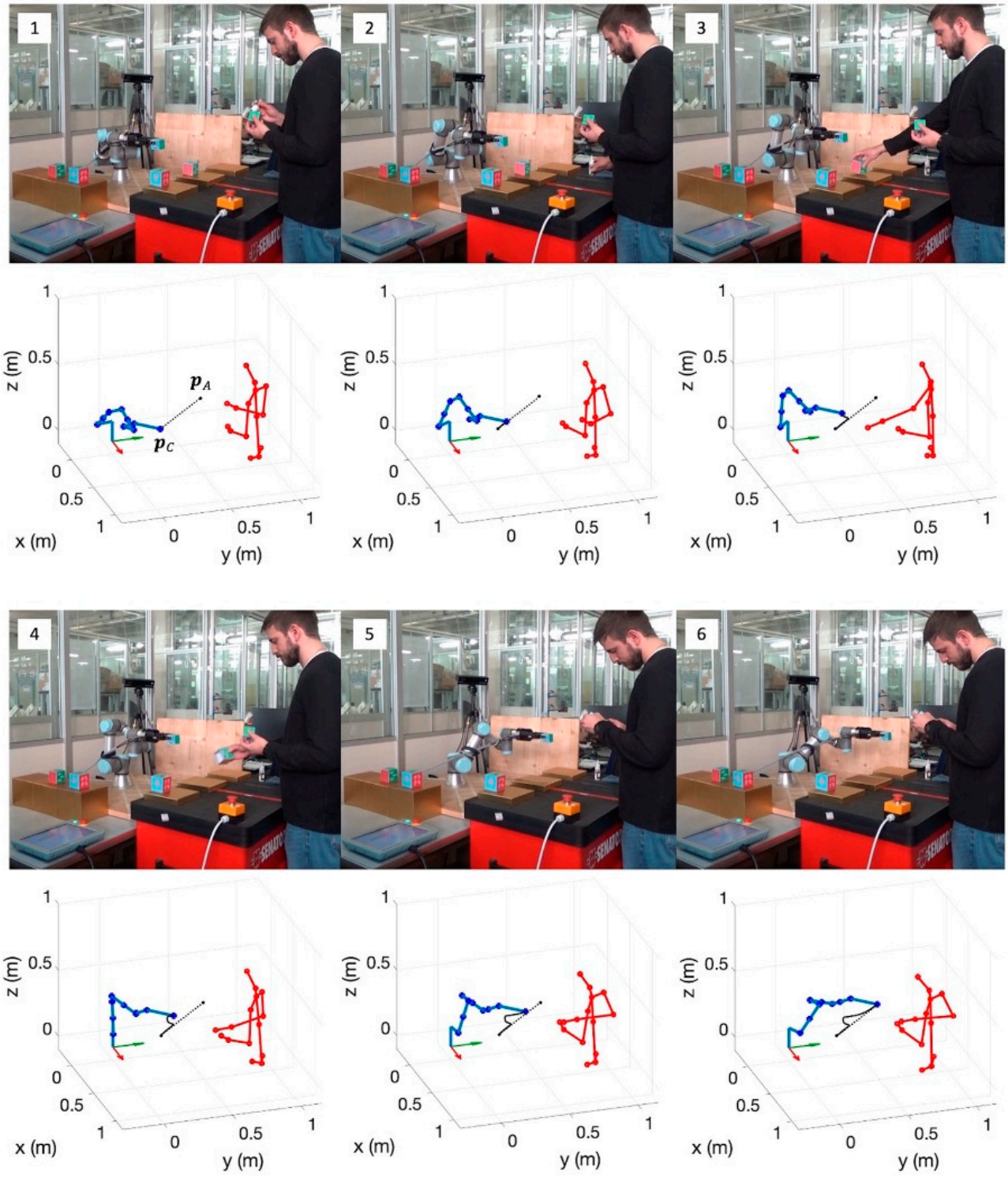


Figure 9. Frames of the collision avoidance during the "Place a↦A" stage.

The possibility to carry out the task with the highest level of collaboration brings significant advantages in terms of task times. At the top of Figure 11, the workflow diagram of the proposed assembly cell is shown. In this case, which represents a responsive collaboration, the human and the robot complete their task in the same time interval, resulting in a mean cycle time of 112 s. The human and robot tasks run in parallel and the robot collaborates while the human executes value added operations, i.e., actions that carry on the assembly process.

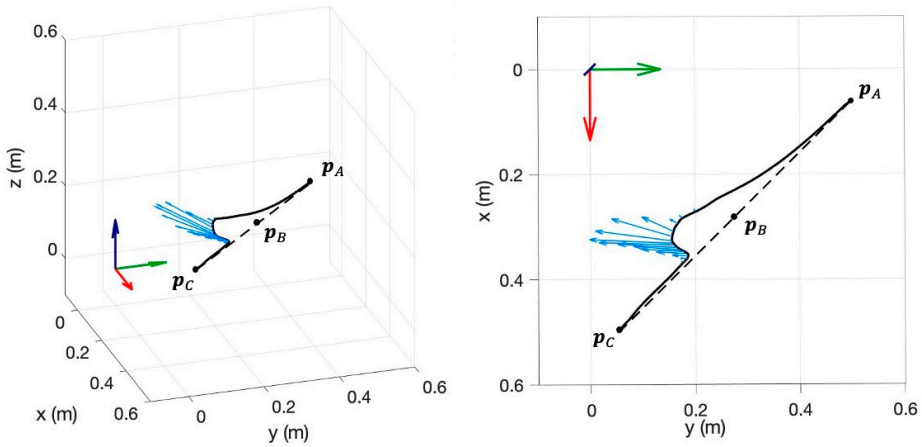
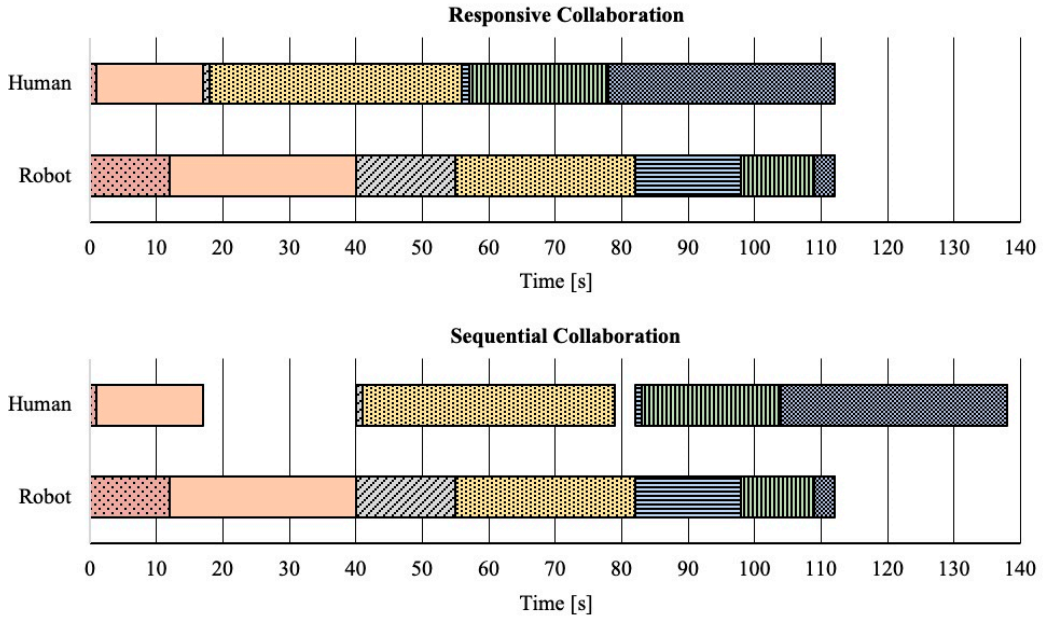


Figure 10. Collision avoidance trajectory during the "Place a↦A" stage.



	Human	Robot
	Take A	Pick a
	Prepare A	Place a↦A
	Take B	Pick b
	Prepare A+B	Place b↦B
	Take C	Pick c
	Prepare A+B+C	Place c↦C
	Check & Deliver	Home Position

Figure 11. Workflow diagram of the assembly cycle for responsive and sequential collaborations.

To stress the importance of this result, it is assumed that the same assembly cycle is carried out by considering the lowest level of collaboration. The latter identifies a sequential collaboration, where the human and robot workspaces intersect, but without the possibility for the human to access the collaborative zone if it is already occupied by the robot, and vice-versa. With this more restrictive safety constraint, the workflow diagram would be the one in the middle of Figure 11. In fact, when the operator finishes preparing A, he cannot enter in the collaborative zone until the robot accomplishes the "Place a↦A" phase. After

that, the robot leaves the collaborative zone and the operator can take B. This generates a first delay of approximately 23 s on the human's task. A second delay of approximately 3 s would be at the end of the "Prepare A+B" phase, since the robot has not finished yet the "Place a↦A" stage and is still in the collaborative zone, therefore, the human must wait before he takes C. The mean cycle time, in this case, is 138 s. Comparing the latter result with the previous case, the major result is that the highest level of collaboration optimizes productivity by saving up to the 18% of the cycle time.

5. Conclusions

This work investigates the feasibility and the potential benefits of a collaborative assembly cell characterized by the highest level of collaboration. A practical and effective collaborative layout based on multiple Kinect cameras has been presented. Then, a representative assembly task which can take advantage from collaborative robotics has been chosen. The robot and operator carry out their task in parallel, with the possibility to operate at the same time in the shared workspace. An UR3 collaborative robot has been used for the experimental test. The robot control strategy has been discussed to describe a collision avoidance algorithm based on repulsive velocities. The algorithm has a low computational cost and can drive the robotic arm away from the human body represented as a skeleton. The result section has shown the capabilities of the proposed algorithm by analyzing the robot alternative trajectory. The robot is able to avoid the human and return on the planned path to carry on the task. Moreover, the cycle time has been compared for the cases of responsive and sequential collaboration. The highest level of collaboration, which has been tested in this work, promises significant improvement in terms of task time, even if the actual possible benefit should be carefully evaluated for each specific application.

Software and hardware improvements of the experimental setup will be involved in future works. The control algorithms will be written in a low-level programming language to package the code for real-time application in the industrial field. The Microsoft Kinect v2 sensors will be substituted by Azure Kinect DK [40], which is the new RGBD camera by Microsoft, and the number of the PCs will be optimized. These software and hardware improvements will lead to a more efficient and cheaper experimental setup.

The authors are currently working on algorithms useful to predict the movements of the operator and evaluate his/her velocity using either vision sensors or wearable devices [20,21,41]. These algorithms will permit strictly implementing the equations written in [8] without considering that the velocity of the human operator is constant and equal to 1.6 m/s.

Furthermore, the application of the proposed layout in a real industrial scenario will be studied in the future.

Supplementary Materials: Video of the assembly cycle with responsive collaboration. The video is also available at: https://www.youtube.com/watch?v=5RV4LYVeAvw (accessed on 12 February 2021).

Author Contributions: Conceptualization, L.S.S., M.M., M.T., S.M., and S.P.; methodology, L.S.S., M.M., M.T., S.M., and S.P.; software, L.S.S., M.M., and M.T.; validation, L.S.S., M.M., and M.T.; formal analysis, L.S.S., M.M., and M.T.; investigation, L.S.S., M.M., and M.T.; resources, S.M. and S.P.; data curation, L.S.S., M.M., and M.T.; writing—original draft preparation, L.S.S., M.M., and M.T.; writing—review and editing, L.S.S., M.M., S.M., and S.P.; visualization, L.S.S., M.M., and M.T.; supervision, S.M. and S.P.; project administration, S.M. and S.P.; funding acquisition, S.M. and S.P. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The data presented in this study are available on request from the corresponding author.

Conflicts of Interest: The authors declare no conflict of interest. The funders had no role in the design of the study; in the collection, analyses, or interpretation of data; in the writing of the manuscript, or in the decision to publish the results.

References

1. Pearson, T.; Menassa, R.; Shi, J.; Jimmerson, G. Levels of human and robot collaboration for automotive manufacturing. *Proc. Workshop Perform. Metr.* **2012**, 95–100. [CrossRef]
2. Villani, V.; Pini, F.; Leali, F.; Secchi, C. Survey on human–robot collaboration in industrial settings: Safety, intuitive interfaces and applications. *Mechatronics* **2018**, *55*, 248–266. [CrossRef]
3. Matheson, E.; Minto, R.; Zampieri, E.G.G.; Faccio, M.; Rosati, G. Human–robot collaboration in manufacturing applications: A Review. *Robotics* **2019**, *8*, 100. [CrossRef]
4. Franklin, C.S.; Dominguez, E.G.; Fryman, J.D.; Lewandowski, M.L. Collaborative robotics: New era of human–robot cooperation in the workplace. *J. Saf. Res.* **2020**, *74*, 153–160. [CrossRef]
5. Vicentini, F. Collaborative Robotics: A Survey. *J. Mech. Des.* **2021**, *143*, 040802. [CrossRef]
6. IFR. *Demystifying Collaborative Industrial Robots*; IFR: Frankfurt, Germany, 2018.
7. Krüger, J.; Lien, T.K.; Verl, A. Cooperation of human and machines in assembly lines. *CIRP Annals* **2009**, *58*, 628–646. [CrossRef]
8. ISO/TS 15066. *Robots and Robotic Devices: Collaborative Robots*; International Organization for Standardization: Geneva, Switzerland, 2016.
9. Chen, C.; Roozbeh, J.; Nasser, K. A survey of depth and inertial sensor fusion for human action recognition. *Multimed. Tools Appl.* **2017**, *76*, 4405–4425. [CrossRef]
10. Zanuttigh, P.; Marin, G.; Dal Mutto, C.; Dominio, F.; Minto, L.; Cortelazzo, G.M. *Time-of-Flight and Structured Light Depth Cameras*; Springer International Publishing: Berlin/Heidelberg, Germany, 2016.
11. Tilove, R.B. Local obstacle avoidance for mobile robots based on the method of artificial potentials. In Proceedings of the Proceedings IEEE International Conference on Robotics and Automation, Cincinnati, OH, USA, 13–18 May 1990; Volume 1, pp. 566–571.
12. Khatib, O. Real-time obstacle avoidance for manipulators and mobile robots. *Int. J. Robot. Res.* **1986**, *5*, 90–98. [CrossRef]
13. Mauro, S.; Scimmi, L.S.; Pastorelli, S. Collision avoidance system for collaborative robotics. In *Advances in Service and Industrial Robotics*; Ferraresi, C., Quaglia, G., Eds.; Springer: Cham, Switzerland, 2017; Volume 49, pp. 344–352.
14. Mauro, S.; Pastorelli, S.; Scimmi, L.S. Collision avoidance algorithm for collaborative robotics. *Int. J. Autom. Technol.* **2017**, *11*, 481–489. [CrossRef]
15. Scimmi, L.S.; Melchiorre, M.; Mauro, S.; Pastorelli, S. Multiple collision avoidance between human limbs and robot links algorithm in collaborative tasks. In *ICINCO 2018, Proceedings of the 15th International Conference on Informatics in Control, Automation and Robotics*; Madani, K., Gusikhin, O., Eds.; SciTePress: Porto, Portugal, 2018; Volume 2, pp. 291–298.
16. Melchiorre, M.; Scimmi, L.S.; Mauro, S.; Pastorelli, S. Influence of human limb motion speed in a collaborative hand-over task. In *ICINCO 2018, Proceedings of the 15th International Conference on Informatics in Control, Automation and Robotics, Porto, Portugal, 29–31 July 2018*; Madani, K., Gusikhin, O., Eds.; SciTePress: Porto, Portugal, 2018; Volume 2, pp. 349–356.
17. Scimmi, L.S.; Melchiorre, M.; Mauro, S.; Pastorelli, S. Implementing a vision-based collision avoidance algorithm on a UR3 Robot. In Proceedings of the 2019 23rd International Conference on Mechatronics Technology (ICMT), Salerno, Italy, 23–26 October 2019; pp. 1–6.
18. Melchiorre, M.; Scimmi, L.S.; Pastorelli, S.; Mauro, S. Collision avoidance using point cloud data fusion from multiple depth Sensors: A Practical Approach. In Proceedings of the 2019 23rd International Conference on Mechatronics Technology (ICMT), Salerno, Italy, 23–26 October 2019; pp. 1–6.
19. Scimmi, L.S.; Melchiorre, M.; Mauro, S.; Pastorelli, S. Experimental real-time setup for vision driven hand-over with a collaborative robot. In Proceedings of the 2019 International Conference on Control, Automation and Diagnosis (ICCAD), Grenoble, France, 2–4 July 2019; pp. 1–5.
20. Melchiorre, M.; Scimmi, L.S.; Mauro, S.; Pastorelli, S.P. Vision-based control architecture for human–robot hand-over applications. *Asian J Control.* **2021**, *23*, 105–117. [CrossRef]
21. Palmieri, P.; Melchiorre, M.; Scimmi, L.S.; Pastorelli, S.; Mauro, S. Human arm motion tracking by Kinect sensor using Kalman filter for collaborative robotics. In *Advances in Italian Mechanism Science*; IFToMM ITALY 2020; Mechanisms and Machine Science; Niola, V., Gasparetto, A., Eds.; Springer: Cham, Switzerland, 2021; Volume 91.
22. Berg, J.; Gebauer, D.; Reinhart, G. Method for the evaluation of layout options for a human-robot collaboration. *Procedia CIRP* **2019**, *83*, 139–145. [CrossRef]
23. Cherubini, A.; Passama, R.; Crosnier, A.; Lasnier, A.; Fraisse, P. Collaborative manufacturing with physical human–robot interaction. *Robot. Comput. -Integr. Manuf.* **2016**, *40*, 1–13. [CrossRef]
24. Realyvasquez-Vargas, A.; Arredondo-Soto, K.C.; Garcia-Alcaraz, J.L.; Marquez-Lobato, B.Y.; Cruz-Garcia, J. Introduction and configuration of a collaborative robot in an assembly task as a means to decrease occupational risks and increase efficiency in a manufacturing company. *Robot. Comput. -Integr. Manuf.* **2019**, *57*, 315–328. [CrossRef]
25. Scalera, L.; Giusti, A.; Di Cosmo, V.; Riedl, M.; Vidoni, R.; Matt, D. Application of dynamically scaled safety zones based on the ISO/TS 15066:2016 for collaborative robotics. *Int. J. Mech. Control.* **2020**, *21*, 41–50.

26. Ragaglia, M.; Zanchettin, A.M.; Rocco, P. Trajectory generation algorithm for safe human-robot collaboration based on multiple depth sensor measurements. *Mechatronics* **2018**, *55*, 267–281. [CrossRef]
27. Zanchettin, A.M.; Rocco, P.; Chiappa, S.; Rossi, R. Towards an optimal avoidance strategy for collaborative robots. *Robot. Comput. -Integr. Manuf.* **2019**, *59*, 47–55. [CrossRef]
28. Kim, E.; Yamada, Y.; Okamoto, S.; Sennin, M.; Kito, H. Considerations of potential runaway motion and physical interaction for speed and separation monitoring. *Robot. Comput. -Integr. Manuf.* **2021**, *67*, 102034. [CrossRef]
29. Zanchettin, A.M.; Casalino, A.; Piroddi, L.; Rocco, P. Prediction of human activity patterns for human–robot collaborative assembly tasks. *IEEE Trans. Ind. Inform.* **2019**, *15*, 3934–3942. [CrossRef]
30. Kinect SDK. Available online: https://developer.microsoft.com/it-IT/windows/kinect (accessed on 12 February 2021).
31. Shotton, J.; Fitzgibbon, A.; Cook, M.; Sharp, T.; Finocchio, M.; Moore, R.; Kipman, A.; Blake, A. Real-time human pose recognition in Parts from Single Depth Images. *CVPR* **2011**, *2011*, 1297–1304.
32. Barros, J.M.D.; Garcia, F.; Sidibè, D. Real-time human pose estimation from body-scanned point clouds. In Proceedings of the International Conference on Computer Vision Theory and Applications, Berlin, Germany, 11–14 March 2015.
33. Yeung, K.Y.; Kwok, T.H.; Wang, C.L. Improved skeleton tracking by a duplex kinects: A practical approach for real-time applications. *ASME J. Comput. Inf. Sci. Eng.* **2013**, *13*, 041007. [CrossRef]
34. Universal Robots. Available online: https://www.universal-robots.com/articles/ur/remote-control-via-tcpip/ (accessed on 12 February 2021).
35. Murphy, R. *Introduction to AI Robotics*; The MIT Press: Cambridge, MA, USA, 2000.
36. Flacco, F.; Kröger, T.; De Luca, A.; 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; pp. 338–345.
37. Scoccia, C.; Palmieri, G.; Palpacelli, M.C.; Callegari, M. Real-time strategy for obstacle avoidance in redundant manipulators. In *The International Conference of IFToMM Italy*; Niola, V., Gasparetto, A., Eds.; Springer: Cham, Switzerland, 2020; Volume 91, pp. 278–285.
38. MathWorks. Available online: https://www.mathworks.com/help/stats/knnsearch.html (accessed on 12 February 2021).
39. Sciavicco, L.; Siciliano, B. *Modelling and Control of Robot Manipulators*; Springer Science & Business Media: London, UK, 2012.
40. Azure Kinect DK. Available online: https://docs.microsoft.com/en-us/azure/kinect-dk/ (accessed on 12 February 2021).
41. Digo, E.; Antonelli, M.; Cornagliotto, V.; Pastorelli, S.; Gastaldi, L. Collection and analysis of human upper limbs motion features for collaborative robotic applications. *Robotics* **2020**, *9*, 33. [CrossRef]