

Article

# Innovative Collaborative Method for Interaction between a Human Operator and Robotic Manipulator Using Pointing Gestures

Marek Čorňák \* , Michal Tölgyessy  and Peter Hubinský 

Faculty of Electrical Engineering and Information Technology, Institute of Robotics and Cybernetics, Slovak University of Technology in Bratislava, Ilkovičova 3, 812 19 Bratislava, Slovakia; michal.tolgyessy@stuba.sk (M.T.); peter.hubinsky@stuba.sk (P.H.)

\* Correspondence: marek.cornak@stuba.sk

**Abstract:** The concept of “Industry 4.0” relies heavily on the utilization of collaborative robotic applications. As a result, the need for an effective, natural, and ergonomic interface arises, as more workers will be required to work with robots. Designing and implementing natural forms of human–robot interaction (HRI) is key to ensuring efficient and productive collaboration between humans and robots. This paper presents a gestural framework for controlling a collaborative robotic manipulator using pointing gestures. The core principle lies in the ability of the user to send the robot’s end effector to the location towards, which he points to by his hand. The main idea is derived from the concept of so-called “linear HRI”. The framework utilizes a collaborative robotic arm UR5e and the state-of-the-art human body tracking sensor Leap Motion. The user is not required to wear any equipment. The paper describes the overview of the framework’s core method and provides the necessary mathematical background. An experimental evaluation of the method is provided, and the main influencing factors are identified. A unique robotic collaborative workspace called Complex Collaborative HRI Workplace (COCOHRIP) was designed around the gestural framework to evaluate the method and provide the basis for the future development of HRI applications.

**Keywords:** HRI (human–robot interaction); Leap Motion; UR5; hand gesture recognition; pointing gestures; collaborative robotics



**Citation:** Čorňák, M.; Tölgyessy, M.; Hubinský, P. Innovative Collaborative Method for Interaction between a Human Operator and Robotic Manipulator Using Pointing Gestures. *Appl. Sci.* **2022**, *12*, 258. <https://doi.org/10.3390/app12010258>

Academic Editors: Giuseppe Carbone and Med Amine Laribi

Received: 10 November 2021

Accepted: 20 December 2021

Published: 28 December 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

Collaborative robotic applications are, nowadays, well-established and commonly used concepts of technology. The main goal of such applications is to combine the strengths of both humans and robotic systems to achieve maximum effectiveness in completing a specified task while minimizing the risks imposed on the human worker. In the manufacturing process, these applications are crucial in enabling the concept referred to as “Industry 4.0”. Industry 4.0 focuses on developing so-called cyber-physical systems or CPS for short, aiming to create highly flexible production systems capable of fast and easy changes, addressing the need for individualized mass production of the current markets [1].

Collaborative applications are becoming increasingly popular on the factory floors due to their various benefits, such as lower deployment costs, compact sizes, and easier repurposing, than standard robotic systems [2]. As a result, more factory workers will be required to work in close contact with robotic systems. This, however, introduces new arduous challenges to overcome, primarily how to secure the safety of a human worker, while ensuring high work effectiveness. The former problem focuses mainly on minimizing potential risks and avoiding accidents from collaborative work between the human worker and a robot, not considering the work efficiency. The latter aims to find the methods capable of maximizing the overall productivity of such collaboration. Albeit the safety of the human worker is an absolute priority, also being the most currently researched topic [3], the economic benchmarks are vital in determining further deployment of these

applications. Thus, it is necessary to pay more attention to the interaction between the human and robotic systems, also referred to as HRI, researching the impact and influence of the whole human–robot relationship, developing new approaches capable of meeting today’s production demands.

In today’s conventional robotic application, usually involving robust industrial manipulators, the robot works in a cell that must be wholly separated from the workers’ environment by a physical barrier [4]. The human only comes in contact with the robot during maintenance and programming, carried out by highly qualified personnel. All of this is not only expensive, but also time-consuming. Alternatively, in collaborative applications, factory workers may often get into contact with parts of the robot and are even encouraged to influence the robot’s actions based on the current circumstances to secure maximum work flexibility. This could pose a severe problem, where technically unskilled workers may suffer great difficulties communicating with the robotic system or conducting some form of control. A solution to this problem lies in developing such HRI methods, which would enable natural and ergonomic human–robot communication without the need for prior technical knowledge of the robotic systems. Hence, the workers could solely focus on the task at hand and not waste time figuring out the details of the application’s interface.

For humans, the most natural way of communication is through speech, body, or facial gestures. Implementing these types of interfaces into collaborative applications could yield many benefits, which lead to increased productivity and overall worker contentment [5].

This paper presents a gesture-based collaborative framework for controlling a robotic manipulator with flexible potential deployment in the manufacturing process, academic field, or even a commercial sector. The underlying basis of the proposed framework is based on the previous work of Tölgyessy et al. [6], which introduced the concept of “linear HRI,” initially designed for mobile robotic applications. This paper further develops this concept, widening its usability from mobile robotic applications to robotic arm manipulation while utilizing a new state-of-the-art visual-based sensory system. The main goal of the proposed framework is to provide systematic foundations for gesture-based robotic control, which would support a wide variety of potential use-cases, could be applied universally regardless of the specific software or robotic platform while providing a basis on which other more complex applications could be built.

## 2. Related Work

Gestural HRI is a widely researched topic across the whole spectrum of robotics. Allowing the robot to detect and recognize human movements and act accordingly is a powerful ability enabling the robot and the human worker to combine their strengths and achieve various challenging tasks. In mobile robotics, gestures can be used to direct and influence the robot’s movement. Cho and Chung [7] used a mobile robotic platform, Pioneer 3-DX, and a Kinect sensor to recognize a human body and follow the operator’s movement. Tölgyessy et al. [6] proposed a method for controlling a mobile robotic platform, iRobot Create, equipped with a Kinect sensor via pointing gestures. Chen et al. [8] used a Leap Motion Controller to control the movement of the mobile robotic platform with a robotic arm. Both the chassis and the robotic arm could be controlled via hand gestures. Gesture-based control was also used in projects RECO [9] and TeMoto [10], which focused on intuitive multimodal control of mobile robotic platforms. Besides mobile robotic platforms, force-compliant robotic manipulators are ideal for implementing various forms of gestural interaction, potentially deployable in multiple scenarios, such as object handling, assembly, assistance, etc. A significant part of the research is focused on teleoperation, in which the robot mirrors the human operator’s movements. Hernoux et al. [11] used a Leap Motion sensor and a collaborative manipulator UR10 to reproduce the movement of the operator’s hands. G. Du et al. [12] proposed a similar approach using Leap Motion to control a dual-arm robot with both hands. They used the interval Kalman filter and improved particle filter methods to improve hand tracking. Kruse et al. [13] proposed a

gesture control dual-arm telerobotic system, in which the operator controlled the position of an object held by the robotic system. Microsoft Kinect was used to track the human body. Many other researchers used Microsoft Kinect or a Leap Motion controller to control the robotic manipulator [14–20]. Tang and Webb [5] studied the feasibility of gesture control to replace conventional means of direct control through teach-pendant. Zhang et al. [21] proposed a gesture control for the delta architecture robot. Cipolla and Hollinghurst [22] investigated a gestural interface for teleoperating a robotic manipulator based on pointing gestures. Using stereo cameras, collineations, and an active contour model, they were able to pinpoint a precise location in the 40 cm workspace of the robot at which the user would point at with his finger. Object picking is another promising application area. A human hand can pick different objects of various sizes and shapes. Therefore, several works were conducted to mimic the human hand, to grasp objects [23–25]. An interesting concept was proposed by Razjigaev et al. [26], the authors proposed a gestural control for a concentric tube robot. The work focused on the potential use of such an interface in noninvasive surgical procedures. A gestural interface may be a promising concept in the control of UAV drones; some works [27,28] show potential for future development. Social robotics is another vast research area where gestural interfaces could yield many benefits. Natural interfaces could be especially advantageous in interaction with humanoid robots. Yu et al. [29] proposed a gestural control of the NAO humanoid robot. Cheng et al. [30] used this robot and a Kinect sensor to facilitate a gestural interaction between humans and the humanoid robot.

In Table 1, there is a comparison of key works related to the interaction method designed by us. The vast majority focus mainly on teleoperation of the robotic manipulator used, while the user has visual feedback of the resulting robot motion. Only three approaches present some form of direct interaction of the operator with the manipulator’s workspace and two of these allow the user to point and select objects present in the workspace. The major novelty and contribution of our design is that the operator can select objects on the planar surface; furthermore, she or he can point to any spot of the workspace and subsequently navigate the end effector to the desired destination.

**Table 1.** Comparison of key related works.

Lit.	Robotic System	Sensor	Visual Tracking	Weareables	Tele-Operation	Workspace Interaction	Pointing to Objects
[11]	UR10	Leap Motion	Yes	-	Yes	-	-
[17]	UR10 simulation	Leap Motion	Yes	-	Yes	-	-
[16]	EPSON SCARA LS3-401x	Leap Motion	Yes	-	Yes	-	-
[21]	Custom Delta architecture	Leap Motion	Yes	-	Yes	-	-
[5]	UR5	Leap Motion	Yes	-	Yes	-	-
[12]	GOOGOL GRB3016	Leap Motion	Yes	-	Yes	-	-
[31]	UR5	Leap Motion	Yes	-	Yes	-	-
[20]	ABB IRB120	Kinect V2	Yes	-	Yes	-	-
[32]	Baxter	MYO Armband	-	Yes	Yes	Yes	-
[33]	UR5	Microsoft HoloLens	Yes	Yes	Yes	Yes	Yes
[19]	Custom 7DOF robot	Kinect V2	Yes	-	Yes	-	-
[18]	UR5	Kinect V2	Yes	-	Yes	-	-
[13]	Motoman SDA 10	Kinect V1	Yes	-	Yes	-	-
[22]	Scorbot ER-7	Stereo CCD cameras	Yes	-	Yes	Yes	Yes

### 3. Our Approach

In a conventional manufacturing process, a worker usually performs repetitive manual and often unergonomic tasks. Workers do most of these tasks mainly using their hands, such as assembly, machine tending, object handling, material processing, etc. Manual work, however, has many limitations which consequently influence the whole process efficiency. Integrating a collaborative robotic solution could significantly improve efficiency while, at the same time, it can alleviate a human worker’s physical and mental workload.

Our proposed concept for the gestural HRI framework for the collaborative robotic arm aims to create such interaction, where the human worker could interact with the robot as naturally and conveniently as possible. Secondly, besides ergonomics, the framework focuses on flexibility, providing the user with functionality that could be used in numerous application scenarios. Lastly, the framework lays basic foundations for further application development, pushing the flexibility even further.

The fundamental principles of the proposed framework are based on the so-called “linear HRI” concept introduced by Tölgyessy et al. [6], which formulates three simple laws for HRI that state:

1. Every pair of two joints of a human sensed by a robot form a line.
2. Every line defined by the first law intersects with the robot’s environment in one or two places.
3. Every intersection defined by the second law is a potential navigation target or a potential object of reference for the robot.

The proposed framework was developed under the laws of linear HRI, where the core functionality lies in the ability to send the end effector of the robotic arm to a specific location on a horizontal/vertical plane using the operator’s hand pointing gestures.

Humans naturally use pointing gestures to direct someone’s attention to the precise location of something in space. Combined with speech, it can be a powerful communication tool. Pointing gestures have many great use-cases among people to signify the importance of a particular object (“That’s the pen I was looking for” (pointing to the specific one)), express intention (“I’m going there!” (pointing to the place)), or specify the exact location of something (“Can you hand me that screwdriver please?” (pointing to the specific location in space)). Pointing gestures are incredibly efficient when the traditional speech is insufficient or impossible due to circumstances, such as loud and noisy environments.

Let us imagine a collaborative application scenario where the worker performs a delicate assembly task that requires specific knowledge. A collaborative robot would fulfill the role of an intelligent assistant/co-worker. The worker could point to the necessary tools and parts out of his reach, which the robot would then bring to him; thus, he could specifically focus on the task while keeping his workspace clutter-free. Due to the natural character of the whole interaction, the worker could control the robot’s behavior conveniently without any prior technical skills. Such application would be accessible and easy to use across the entire worker spectrum, with various technical backgrounds. Our gestural framework aims to enable the described interaction in real-world conditions using the principles of linear HRI and state-of-the-art hardware. However, first, the following main challenges need to be addressed:

- Choosing suitable technology and a method of detecting the position and orientation of human joints.
- Calculating the exact position of the intersection between the formed line and the environment surface.
- Selecting the pair of joints to form and calculate the line following the first two principles of linear HRI.

Solving these challenges is crucial for ensuring efficient, reliable, and natural interaction. Additionally, we aimed to make the framework universal, not dependent on the specific robotic platform. For that reason, we chose the Robot Operating System or ROS as our software environment. ROS supports a wide variety of robotic platforms and is considered standard for programming robots in general. For the robot’s control and trajectory planning, the ROS package MoveIt was used.

In summary, the primary objectives of our approach lie in natural interaction, application flexibility, and scalability, while following the underlying concept of so-called linear HRI.

#### 4. The Human Body Tracking and Joint Recognition

Precise recognition and localization of joints are vital in gestural interaction. Several technologies and approaches exist, providing the user with tracking capabilities. According to Zhou and Hu [34], human motion tracking technologies can be divided into non-visual, visual, and robot-aided tracking.

In non-visual tracking, the human motion and joints are mapped via various sensors placed onto the human body; typically, MEMS IMU or IMMU devices are used. These sensors are usually a part of a suit or so-called “data glove”, which must be worn by the user [35–39]. Although these systems are precise and relatively cheap, they suffer from many shortcomings, such as the need for the user to wear the sensory suit, which may be uncomfortable and often calibrated for the specific user. The sensors on the suit may require extensive cabling, which can limit the user’s range of motion. The sensors themselves may suffer from several issues, such as null bias error, scale factor error, noise, or interference. Due to this, the general application focus shifted towards the use of visual-based human body tracking.

This approach uses optical sensors, such as RGB, IR, or depth cameras to extract the user’s spatial position. Complex image processing algorithms are then applied to pinpoint the precise position and orientation of human joints. These sensors do not require any equipment that the user needs to wear, thus not limiting him in any way. They can be easily calibrated to the specific user, which greatly improves their flexibility. Today’s camera technologies and image processing algorithms make them fast and accurate and relatively cheap. These attributes made these sensors popular and widely used in various commercial applications, mainly in HCI and video-gaming industry. However, visual-based sensors still have multiple drawbacks, such as the sensitivity to the lighting conditions and worsened tracking capabilities when the human body is occluded. The most widely recognizable sensor for visual body tracking is the Microsoft Kinect, released in 2010, initially intended for video gaming. However, due to its capabilities, the sensor became widely used in other applications. The Kinect’s principle of body tracking relies on capturing the depth data from the depth sensors and then applying their image processing methods to produce a so-called “skeletal stream” representing the human figure. The first iteration of the sensors used an IR projection pattern to acquire the depth data. The latter versions used the so-called “time-of-flight” technology or TOF. According to Shotton et al. [40], body part classification (BPC) and offset joint regression (OJR) algorithms, specially developed for Microsoft Kinect, are used to determine the parts of the human body. Following the success of Microsoft Kinect, other similar sensors were released, such as Intel RealSense or ASUS Xtion.

Another widely popular vision-based motion tracking sensor is the Leap Motion controller. This compact device was specially designed to track the human hands and arms at very high precision and accuracy. Alongside gaming and VR applications, the controller’s design was initially meant to replace conventional peripheral devices, such as a mouse and keyboard, and provide a more sophisticated and natural HCI. The Leap Motion controller uses two IR cameras, capturing emitted light from three LEDs with a wavelength of 850 nm. The depth data are acquired by applying unique algorithms on the raw data, consisting of infrared brightness values and the calibration data [41]. According to the manufacturer, the accuracy of position estimation is around  $\pm 0.01$  mm. However, several studies show [42,43] that this is highly dependent on the conditions.

##### *Sensor Choice for the Proposed Concept*

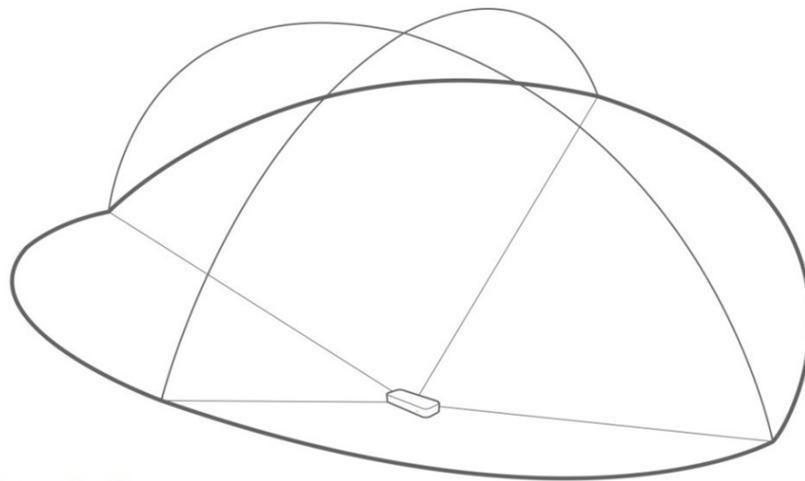
Due to the advantages of vision-based body tracking, we decided to use this technology in our concept. We believe that the proper HRI should not rely on “human-dependent devices” as data gloves or body-mounted sensors, but on the perceptive ability of the robot itself, as this, in our opinion, most accurately represents natural and ergonomic interaction.

As for the specific sensor, the Leap Motion controller was picked as the most suitable option. The main reason is the sensor’s application focus. The core of the proposed concept

centers around the hand gesture interaction between the robot and the human worker, as most manufacturing tasks are done by hand. Furthermore, the whole gestural framework is built on pointing gestures, performed by the arrangement of individual fingers. Leap Motion provides accurate and precise human hand tracking, explicitly focusing on the position and orientation of fingers. Other commercially available sensors on the market are not yet capable of such precision and generally focus on tracking the whole human body. The controller depicted in Figure 1 can track hands within a 3D interactive zone that extends up to 60 cm (24") or more, extending from the device in a  $140 \times 120^\circ$  typical field of view, which is illustrated in Figure 2. The controller produces an output in the form of gray-scale images captured by the IR cameras and the skeletal representation of the human hand. The data are shown in Figure 3. Leap Motion's software can differentiate between 27 distinct hand elements, such as bones and joints, and track them even when other hand parts or objects obscure them. The positions of joints are represented in the controller's coordinate system depicted in Figure 4. The connection with the PC is facilitated via USB 2.0 or USB 3.0 connection. The manufacturer also provides a software development kit or SDK for the Leap Motion controller, allowing the developers to create custom applications.



**Figure 1.** The Leap Motion controller (LMC).



**Figure 2.** Approximate workspace of the LMC.

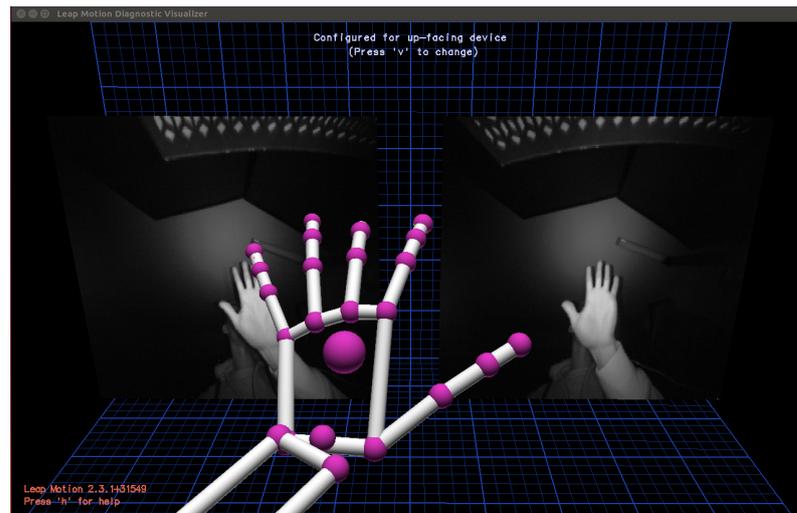


Figure 3. Visualization of detected hand joints.

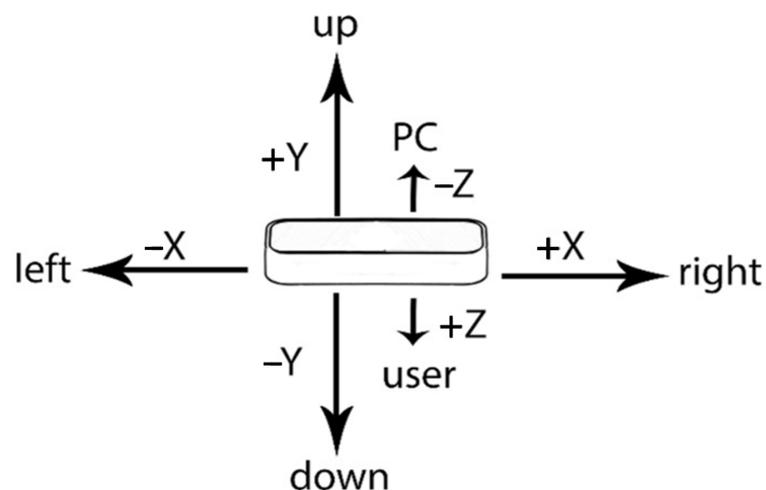
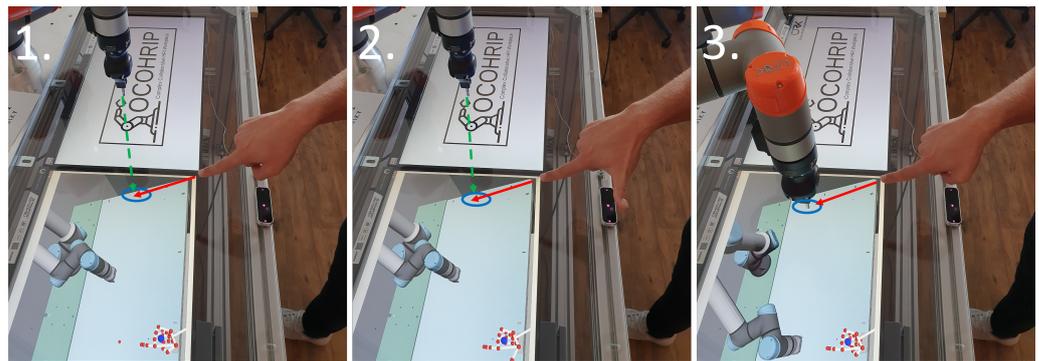


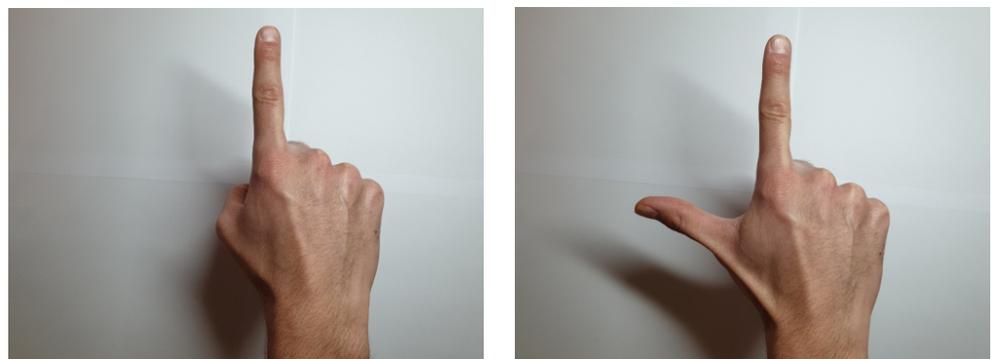
Figure 4. Coordinate system of the LMC.

## 5. Method Design

The core functionality of the proposed framework follows specific consecutive steps. The operator first performs a pointing gesture, pointing to the particular location in the (planar) workspace of the robot. Then, the robot computes the specified location defined by the planar surface and the half-line formed by the operator's joints. When the operator is satisfied with the pointed place, he performs a command gesture, triggering the signal to move the robot's end effector to the desired location. The whole process is illustrated in Figure 5. The half-line is defined by the direction of the index finger, as it is the most common way to represent a pointing gesture. The following command gesture was specifically designed to be performed naturally, fluently connecting with the previous gesture while not significantly influencing the pointing precision. The command gesture is achieved by extending the thumb, forming a so-called "pistol" gesture. Both gestures are depicted in Figure 6. Custom ROS packages and nodes were created for gesture recognition and intersection computation. The ROS-enabled machine connected to the robotic arm manages these nodes, ensuring proper communication between the individual parts of the whole application. The method's architecture in ROS framework is depicted in Figure 7. The flowchart of the method's process is in Figure 8. The most prominent geometrical features for the mathematical description of the concept are illustrated in Figure 9. The global coordinate system  $G$  defines the planar ground surface  $\pi$ .



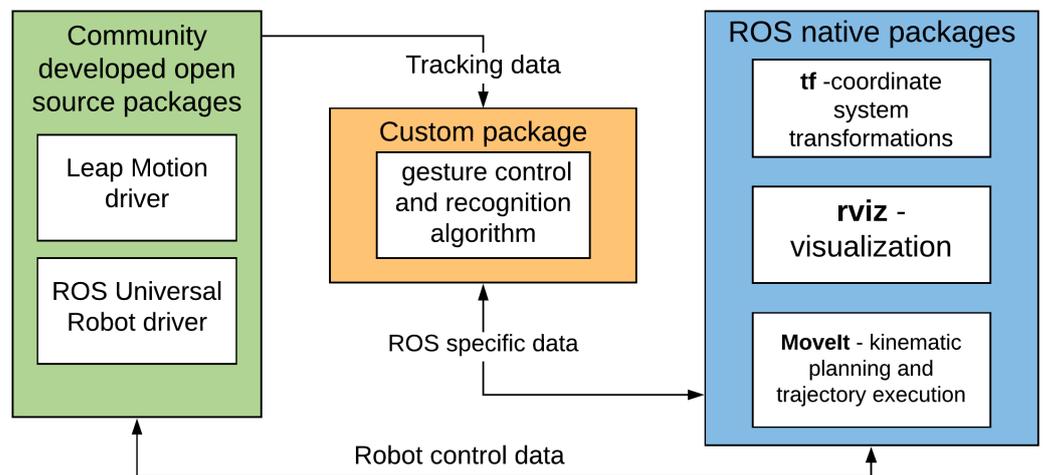
**Figure 5.** Process of controlling the robot via pointing gestures. 1—The operator points to a location. 2—The operator performs “pistol” gesture. 3—The robot’s TCP moves to the appointed location.



(a) Pointing gesture

(b) Pistol gesture

**Figure 6.** Two basic gestures used to interact with the system.



**Figure 7.** The package architecture of the method in ROS.

The Leap Motion sensor determines the position of the joints of the human hand in its coordinate system  $L$ ; furthermore, the robotic manipulator operates in its coordinate system  $R$ . The unification of those coordinate systems is vital to obtain points  $A$  and  $B$  coordinates defining the half-line  $p$ , which ultimately determines the coordinates of intersection  $I$ .

All of the calculations are done in the global coordinate system  $G$ . Hence, the following homogeneous transformations between  $R$  and  $G$ ,  $L$ , and  $G$  are defined:

$$\begin{pmatrix} X_G \\ Y_G \\ Z_G \\ 1 \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 & d_x \\ 0 & \cos \alpha & -\sin \alpha & d_y \\ 0 & \sin \alpha & \cos \alpha & d_z \\ 0 & 0 & 0 & 1 \end{pmatrix} \cdot \begin{pmatrix} X_L \\ Y_L \\ Z_L \\ 1 \end{pmatrix} \tag{1}$$

$$\begin{pmatrix} X_G \\ Y_G \\ Z_G \\ 1 \end{pmatrix} = \begin{pmatrix} 0 & \cos \alpha & -\sin \alpha & d_x \\ 0 & \sin \alpha & \cos \alpha & d_y \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \cdot \begin{pmatrix} X_R \\ Y_R \\ Z_R \\ 1 \end{pmatrix}$$

Analytical geometry and vector algebra are used to calculate the desired coordinates. The data acquired from the Leap Motion sensor are transformed based on Equation (1). The points and vectors used in the calculation are illustrated in Figure 10.

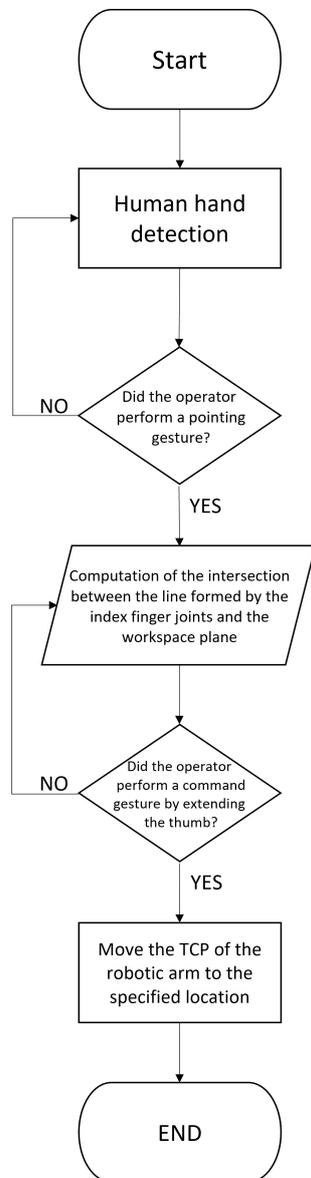


Figure 8. State machine flowchart of the method.

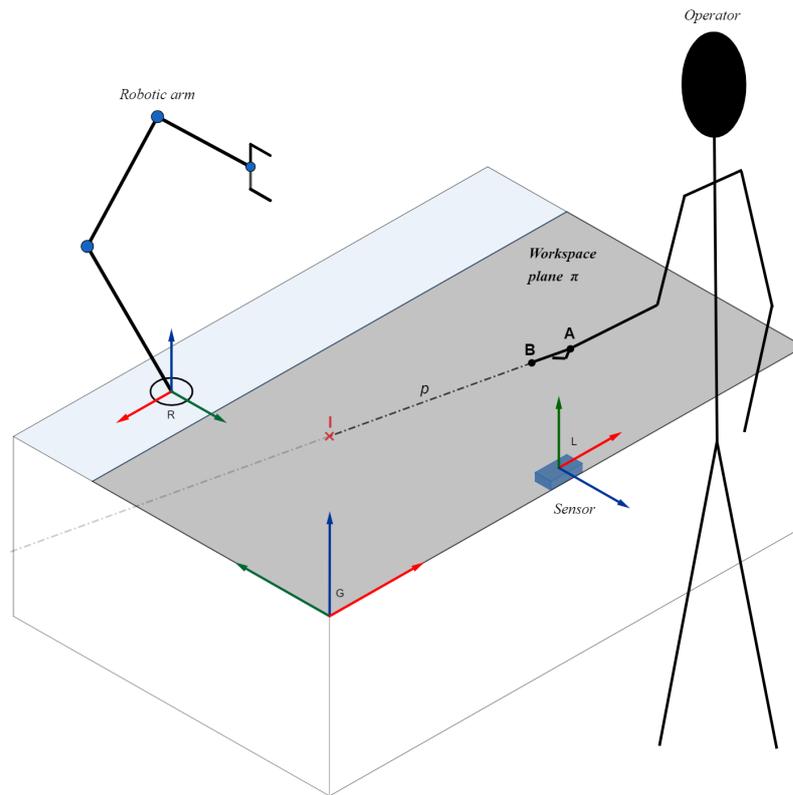


Figure 9. The abstract overview of the method.

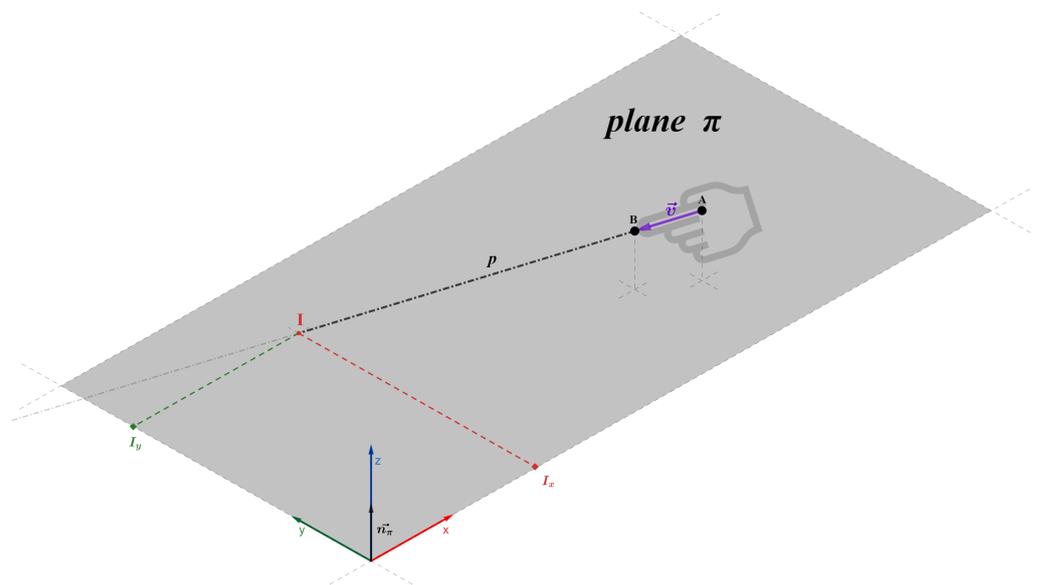


Figure 10. Extraction of fundamental geometrical features.

First, the direction of the vector  $\vec{v}$  defined by points A and B needs to be found. Vector  $\vec{v}$  is calculated by subtracting coordinates of point A from point B (positions of index finger joints).

$$\vec{v} = B - A \tag{2}$$

The direction of vector  $\vec{v}$  is calculated by dividing the coordinates of vector  $\vec{v}$  with its magnitude.

$$\hat{v} = \frac{\vec{v}}{\|\vec{v}\|} \tag{3}$$

Now, it is possible to define the half-line  $p$  by parametric equations.

$$\begin{aligned} x &= x_0 + \hat{v}_1 \times t \\ y &= y_0 + \hat{v}_2 \times t \\ z &= z_0 + \hat{v}_3 \times t \end{aligned} \tag{4}$$

where  $t$  is the parameter of  $p$ . The plane  $\pi$  is defined by the  $x$  and  $y$  axes with an arbitrary normal vector  $\vec{n}$ . Equation (5) can mathematically describe plane  $\pi$ .

$$z = 0 \tag{5}$$

By substituting Equation (5) into Equation (4), the parameter  $t$  can be calculated as follows.

$$t = -\frac{z_0}{\hat{v}_3} \tag{6}$$

Now, using substitution, the  $X$  and  $Y$  coordinates of the intersection  $I(X, Y)$  can be calculated by the following equations.

$$\begin{aligned} I_X &= A_X + \hat{v}_1 \times \left(-\frac{z_0}{\hat{v}_3}\right) \\ I_Y &= A_Y + \hat{v}_2 \times \left(-\frac{z_0}{\hat{v}_3}\right) \end{aligned} \tag{7}$$

Respectively, it is possible to calculate the intersection on the plane  $\omega$  perpendicular to  $\pi$  mathematically defined by the Equation (7).

$$y = d, \tag{8}$$

where  $d$  is the distance of the plane on  $y$  axis. The scenario is illustrated in Figure 11.

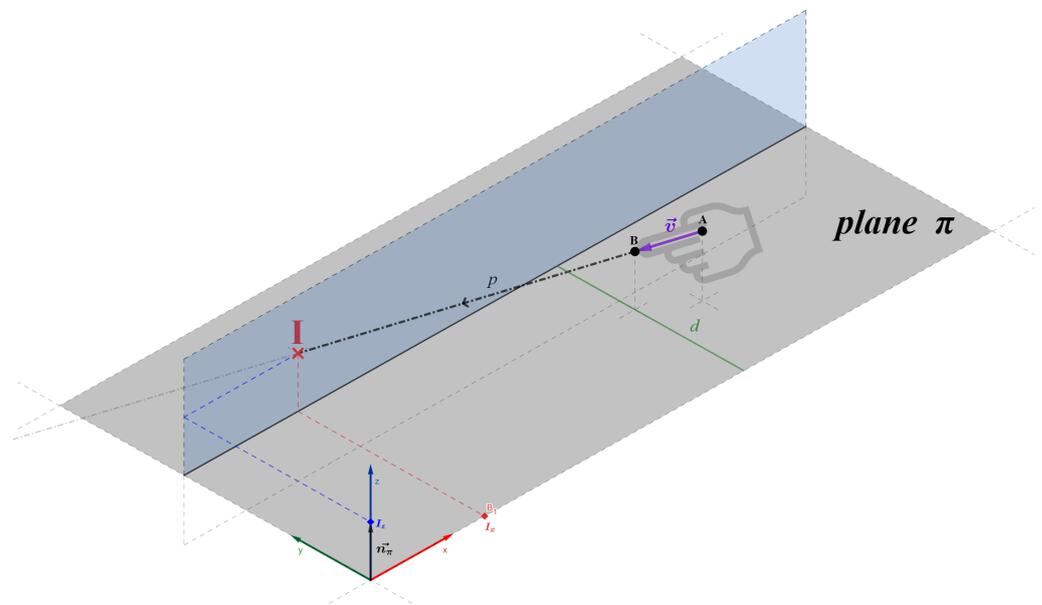


Figure 11. Abstraction of pointing to the perpendicular plane.

### 6. Method Implementation

The main application focus of the proposed concept is to create a natural HRI workplace where humans and robots can work together efficiently. For this reason, a specialized robotic workplace was built around the core concept’s functionality, supporting the ergonomics of the whole interaction between human and the robot, trying to maximize the

efficiency and convenience for the worker. Furthermore, it also acts as a modular foundation for implementing, testing, and evaluating other HRI concepts.

The whole workplace, called COCOHRIP, an abbreviation for Complex Collaborative HRI WorkPlace, is depicted in Figure 12. The COCOHRIP consists of three main parts, the sensors, the visual feedback, and the robotic manipulator. The sensory part contains the various sensors that gather data about the human worker and the environment. The visual feedback part consists of two LED monitors, providing the user with visual feedback from the sensor. The robotic part comprises the force-compliant robotic manipulator. All these parts are connected through the ROS machine, which manages all the communication and logic of the application. Transformation matrices between coordinate systems were:

$$T_{LG} = \begin{pmatrix} 1 & 0 & 0 & 670 \\ 0 & \cos(0.707) & -\sin(0.707) & 15 \\ 0 & \sin(0.707) & \cos(0.707) & 11 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

$$T_{RG} = \begin{pmatrix} 0 & \cos(3.14) & -\sin(3.14) & 1050 \\ 0 & \sin(3.14) & \cos(3.14) & 693.3 \\ 0 & 0 & 1 & 103 \\ 0 & 0 & 0 & 1 \end{pmatrix} \quad (9)$$

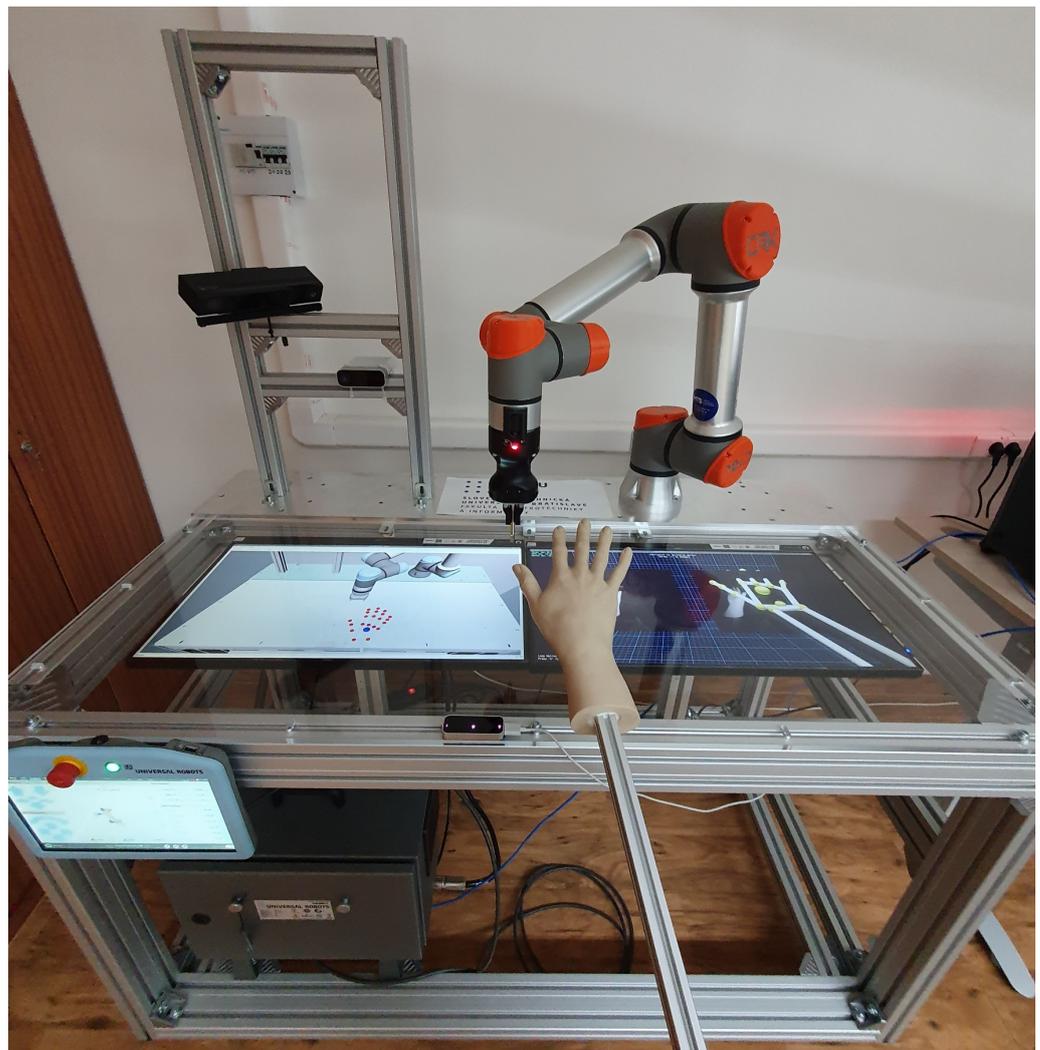


Figure 12. The COCOHRIP workplace.

## 7. Experimental Evaluation

In this section, we present two experimental scenarios to evaluate the basic concept principles, which quantify the overall usability of the concept and serve as a baseline for further development. The scenarios of each of the two experiments are derived from the proposed gesture pointing method, in which the operator points to a specific location on the workspace plane and performs a command gesture, sending the robot to the desired location. Each of the experiment scenarios has two variants. In the first scenario, the operator points to the designated horizontally placed markers; in the second scenario, the operator points at vertically placed markers. In the first variant, the operator has no visual feedback about where she or he is pointing. His task is to rely solely on his best guess. In the second variant, the user receives visual feedback from the interactive monitors showing him the exact place he is pointing. The illustration of the experimental setup and the precise positions of the markers in the global coordinate system is depicted in Figures 13 and 14. The position values are in millimeters. The process of obtaining one measurement is as follows:

1. The operator's hand is calibrated for optimal gesture recognition.
2. The operator points to the first point, and when he is satisfied with his estimation, he performs a command gesture.
3. When the operator performs a command gesture, the spatial data are logged, and the robot's TCP moves to the pointed location.
4. The operator repeats these previous two steps, iterating through each target point in order, as illustrated in Figures 13 and 14.

In total, the operator points ten times to each of the target markers for the current experiment. During the pointing, the only requirement on the user is to keep the hand approximately 200–250 mm above the Leap Motion sensor, as this is believed to be the optimal distance according to [42] and the empirical data acquired during the method pre-testing. The user is encouraged to perform the pointing gestures as naturally as possible, allowing him to position his body and arm as he sees fit. Seven male participants of different height, physical constitution, and height executed the experiment scenarios.

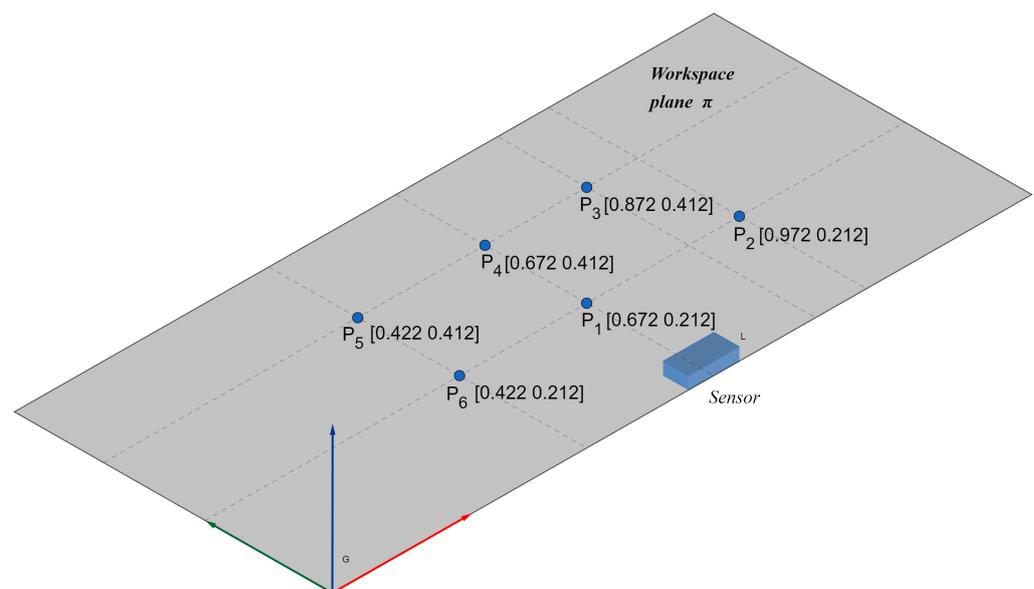


Figure 13. Scenario of experiment no. 1.

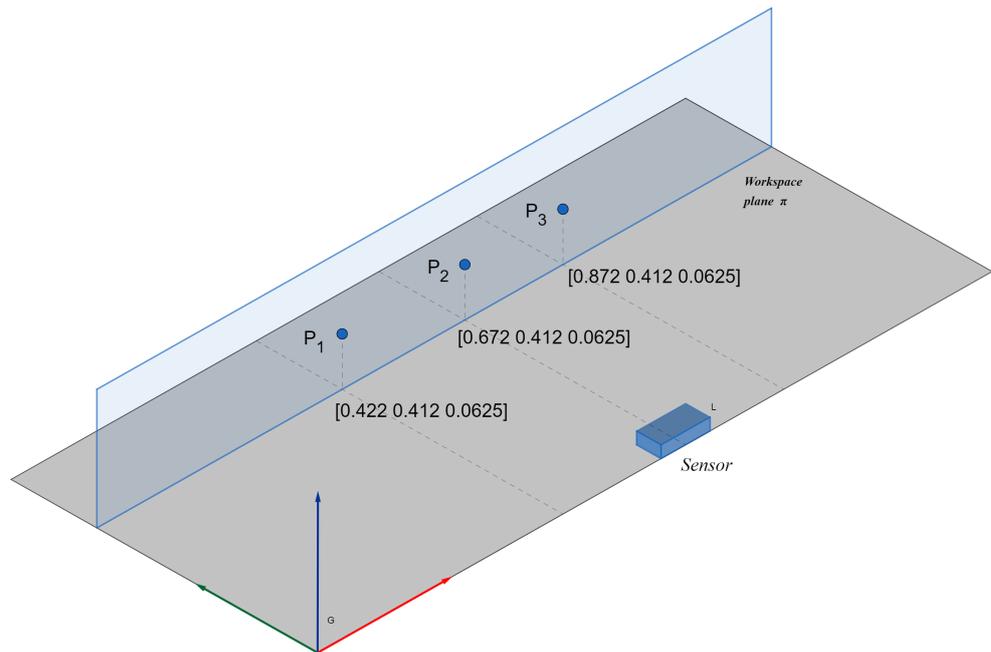


Figure 14. Scenario of experiment no. 2.

### 8. Results

The measured coordinates for each target marker relative to the desired position are depicted in Figures 15 and 16. The blue markers represent the first variant, where the user had no visual feedback. The red markers represent the second variant of each experiment. The euclidean distance between each measured point and the desired position was computed to quantify the precision of pointing for each experiment. Figures 17 and 18 show the boxplots of the euclidean distance data for each experiment scenario. The mean deviation in position and the overall deviation are shown in Tables 2 and 3.

Additionally, the dispersion of measured points in each axis was calculated to evaluate the repeat-ability of pointing. The results are shown in Tables 4 and 5. All the data are represented in millimeters.

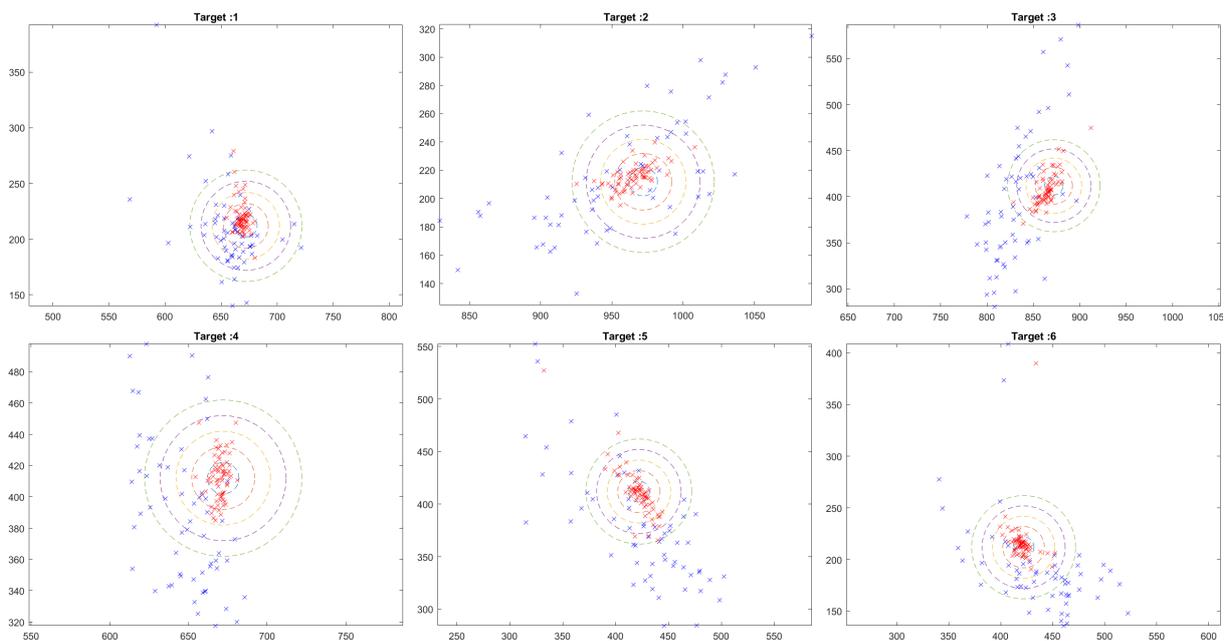


Figure 15. The acquired position data for experiment no. 1.

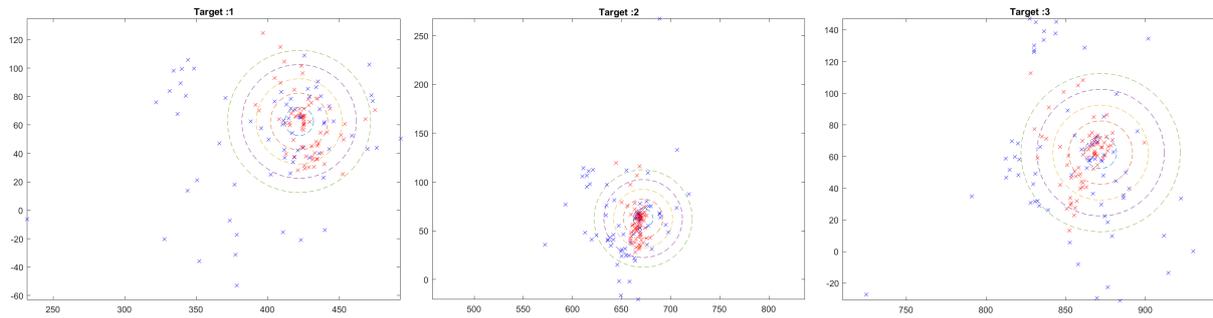


Figure 16. The acquired position data for experiment no. 2.

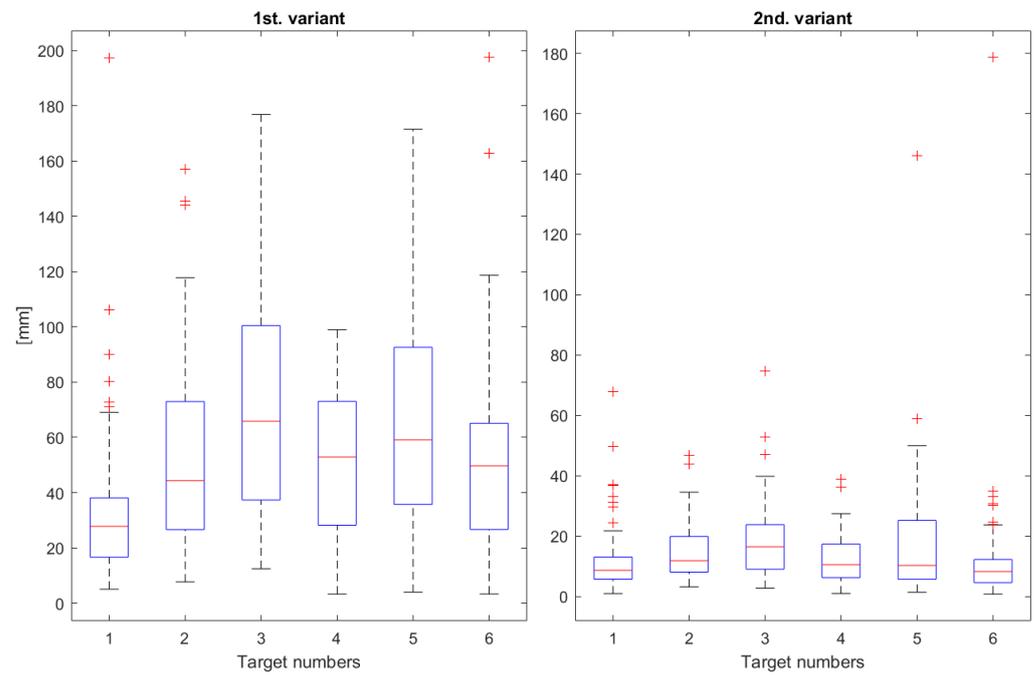


Figure 17. Boxplots of euclidean distance data for experiment no. 1.

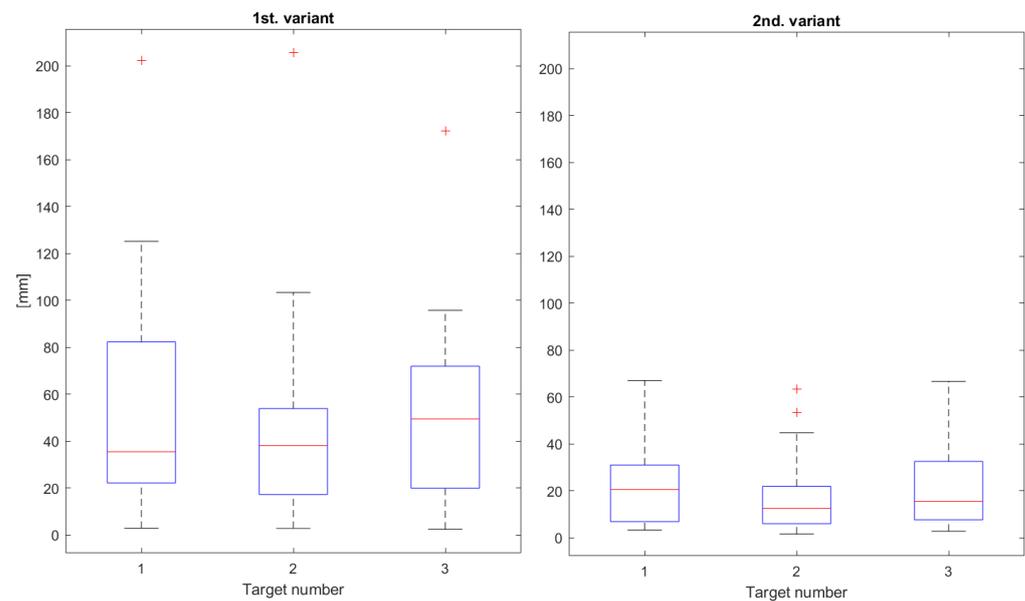


Figure 18. Boxplots of euclidean distance data for experiment no. 2.

**Table 2.** The mean of euclidean distances between the desired position and measured data for each horizontally placed target marker.

Target Marker No.	1st. Variant (mm)	2nd. Variant (mm)
1	33.42	12.16
2	54.18	14.81
3	70.71	18.68
4	50.88	12.60
5	63.99	17.64
6	51.45	12.54
overall	54.11	14.74

**Table 3.** The mean of euclidean distances between the desired position and measured data for each vertically placed target marker.

Target Marker No.	1st. Variant (mm)	2nd. Variant (mm)
1	49.77	21.17
2	40.56	15.62
3	47.25	20.21
overall	45.86	19.0

**Table 4.** The dispersion of measured points for each target in the first experimental scenario.

	1st. Variant (mm)		2nd. Variant (mm)	
	x Axis	y Axis	x Axis	y Axis
1	22.42	35.32	4.94	14.07
2	50.14	36.93	13.28	8.35
3	26.92	65.54	11.24	17.64
4	20.08	44.55	4.91	13.99
5	44.31	52.0	15.46	22.34
6	38.71	43.99	9.06	22.82
overall	33.77	46.39	9.81	16.53

**Table 5.** The dispersion of measured points for each target in the second experiment scenario.

	1st. (mm)		2nd. Variant (mm)	
	x Axis	y Axis	x Axis	y Axis
1	46.10	37.44	14.47	20.79
2	27.74	40.20	5.61	17.93
3	32.88	43.96	12.66	19.40
overall	35.57	40.53	10.91	19.37

## 9. Discussion

The data show that the error of pointing to the specific location, relying solely on the user's estimation, is approximately 50 mm. The user's consistency of pointing is around 30–50 mm, depending on the user's distance to the target. The results also show significant improvement in both accuracy and consistency when visual feedback is provided. These results outline the potential application use regarding the possible size of manipulated objects as mentioned previously in a real-life scenario. They also highlight the importance of some feedback that should be implemented to potential real use-case applications. However, the data are not yet sufficient to derive general conclusions. Further, more thorough analysis is needed. During the experiments, several factors were identified that could significantly influence the accuracy of pointing. The first factor is the posture and position of the human body and the distortion caused by the user's point of view. The second factor is the tracking capabilities and precision of the LMC sensor. During experiments, numerous glitches of

LMC controller were reported. The LMC occasionally misinterpreted the position of joints and fingers, which led to inaccuracies in the measurements. Another factor influencing the results is the joints selected to represent the half-line intersecting with the working plane. The last identified factor is the influence of the command pistol gesture on pointing accuracy, as extending the thumb may change the intended pointing location. Further evaluation of the method, considering all of the mentioned factors, must be done in the future to determine the full potential of the proposed gestural framework. In all, we believe that the experiments and the data served their purpose. The main factors influencing the accuracy were identified, and the obtained data can be used as the base for further experiments and evaluation of the method.

## 10. Conclusions

In this work, we proposed and successfully implemented a framework for gestural HRI using pointing gestures. The core concept of the method was based on the previous work by Tölgyessy et al. [6], particularly on the idea of linear HRI. In the work, they proposed a similar concept of controlling a mobile robotic platform via pointing gestures. This article further improves the work by extending the overall potential of this innovative concept from mobile platforms to other fields of robotics. Additionally, it promotes the universal nature of linear HRI as the concept uses a different sensory system and a different type of robotic system. The work contains a general overview as well as the mathematical background for the core method. A concept of the collaborative robotic workplace, also known as COCOHRIP, was built around the proposed framework. Practical experiments in a laboratory environment were carried out to prove the described method. The obtained data provided the basis for further development of the whole gestural framework.

The main contribution of this work is a novel approach to the control of the collaborative robotic arm using pointing gestures. In the majority of the cited work, the researchers focus on teleoperation of the robotic arm either through movement mirroring or by binding the control of the robot to a specific gesture. The operator is not directly interacting with the (shared) workspace. The robot is controlled directly by the user, and its movements are entirely subordinate to the operator's movements. In our approach, the operator interacts with the robot through the shared workspace, thus not teleoperating the robot directly. The robot is in the role of an assistant rather than a subordinate. Consequently, this dynamic could facilitate more effective collaboration between humans and the robotic systems. Cippolla and Hollinghurst [22] presented a gesture-based interface method that is most comparable to our approach. They similarly use pointing gestures to direct the robot's movement. In their work, hand tracking is achieved using stereo monochromatic CCD cameras. However, their method suffers shortcomings such as the need for high contrast background and specific light conditions. Additionally, the method's workspace is much smaller compared to our method. Our proposed concept has far more robust gesture recognition and tracking performance, additionally being able to recognize multiple gestures. Other cited works in which the user interacts with a workspace require some sort of wearable equipment. We believe the future of HRI lies in the utilization of multiple interconnected visual-based sensors with the ultimate purpose to rid humans of any additional restrictive equipment, allowing them to interact with robots as naturally and ergonomically as possible, without the need for superior technical skills. Furthermore, the core mathematical method in the proposed framework is universally applicable, not relying on the specific hardware or software.

The practical utilization of the proposed framework was outlined in the form of a collaborative robotic assistant that could potentially be used in the manufacturing process, academic field, or research and development. This paper further develops this concept, widening its usability from mobile robotic applications to robotic arm manipulation while utilizing a new state-of-the-art visual-based sensory system. Our framework, along with the built COCOHRIP workplace, could serve as a basis for more complex applications, such as naturally controlled object picking and manipulation, robot-assisted assembly, or material

processing. Our future research will focus on improving the system's accuracy based on the identified factors, and designing and implementing more complex algorithms and methods to further increase the usability and flexibility of the proposed gestural framework.

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

**Funding:** This work was funded by APVV-17-0214.

**Data Availability Statement:** All data presented in this paper are available at: <https://drive.google.com/drive/folders/1V4cUwvD7barr6BBPcmY9KFzPz3vc1Ee8?usp=sharing>.

**Conflicts of Interest:** The authors declare no conflict of interest.

## References

1. Tang, N. *Securing the Future of German Manufacturing Industry Recommendations for Implementing the Strategic Initiative INDUSTRIE 4.0 Final Report of the Industrie 4.0 Working Group*; Acatech: Munich, Germany, 2013.
2. 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](#)]
3. Gualtieri, L.; Rauch, E.; Vidoni, R. Emerging research fields in safety and ergonomics in industrial collaborative robotics: A systematic literature review. *Robot. Comput.-Integr. Manuf.* **2021**, *67*, 101998. [[CrossRef](#)]
4. Thoben, K.; Wiesner, S.; Wuest, T. "Industrie 4.0" and Smart Manufacturing—A Review of Research Issues and Application Examples. *Int. J. Autom. Technol.* **2017**, *11*, 4–16. [[CrossRef](#)]
5. Tang, G.; Webb, P. The Design and Evaluation of an Ergonomic Contactless Gesture Control System for Industrial Robots. *J. Robot.* **2018**, *2018*, 9791286. [[CrossRef](#)]
6. Tölgyessy, M.; Dekan, M.; Duchoň, F.; Rodina, J.; Hubinský, P.; Chovanec, L. Foundations of Visual Linear Human–Robot Interaction via Pointing Gesture Navigation. *Int. J. Soc. Robot.* **2017**, *9*, 509–523. [[CrossRef](#)]
7. Cho, H.; Chung, W. Preliminary research on robust leg-tracking indoor mobile robots by combining the Kinect and the laser range finder information. In Proceedings of the 12th International Conference on Ubiquitous Robots and Ambient Intelligence (URAI), Goyangi, Korea, 28–30 October 2015; [[CrossRef](#)]
8. Chen, M.; Liu, C.; Du, G. A human—Robot interface for mobile manipulator. *Intell. Serv. Robot.* **2018**, *11*, 269–278. [[CrossRef](#)]
9. Burger, B.; Ferrané, I.; Lerasle, F.; Infantes, G. Two-handed gesture recognition and fusion with speech to command a robot. *Auton. Robot.* **2012**, *32*, 129–147. [[CrossRef](#)]
10. Valner, R.; Kruusamäe, K.; Pryor, M. TeMoto: Intuitive Multi-Range Telerobotic System with Natural Gestural and Verbal Instruction Interface. *Robotics* **2018**, *7*, 9. [[CrossRef](#)]
11. Hernoux, F.; Béarée, R.; Gibaru, O. Investigation of dynamic 3D hand motion reproduction by a robot using a Leap Motion. In Proceedings of the 2015 Virtual Reality International Conference. Association for Computing Machinery, VRIC '15, Laval, France, 8–10 April 2015; pp. 1–10. [[CrossRef](#)]
12. Du, G.; Zhang, P.; Liu, X. Markerless Human–Manipulator Interface Using Leap Motion With Interval Kalman Filter and Improved Particle Filter. *IEEE Trans. Ind. Inform.* **2016**, *12*, 694–704. [[CrossRef](#)]
13. Kruse, D.; Wen, J.T.; Radke, R.J. A Sensor-Based Dual-Arm Tele-Robotic System. *IEEE Trans. Autom. Sci. Eng.* **2015**, *12*, 4–18. [[CrossRef](#)]
14. Bassily, D.; Georgoulas, C.; Guettler, J.; Linner, T.; Bock, T. Intuitive and Adaptive Robotic Arm Manipulation using the Leap Motion Controller. In Proceedings of the ISR/Robotik 2014: 41st International Symposium on Robotics, Munich, Germany, 2–3 June 2014.
15. Pititeeraphab, Y.; Choitkunnan, P.; Thongpance, N.; Kullathum, K.; Pintavirooj, C. Robot-arm control system using LEAP motion controller. In Proceedings of the 2016 International Conference on Biomedical Engineering (BME-HUST), Hanoi, Vietnam, 5–6 October 2016; pp. 109–112. [[CrossRef](#)]
16. Chen, C.; Chen, L.; Zhou, X.; Yan, W. Controlling a robot using leap motion. In Proceedings of the 2017 2nd International Conference on Robotics and Automation Engineering (ICRAE), Shanghai, China, 29–31 December 2017; pp. 48–51. [[CrossRef](#)]
17. Chen, S.; Ma, H.; Yang, C.; Fu, M. Hand Gesture Based Robot Control System Using Leap Motion. In Proceedings of the Intelligent Robotics and Applications, Portsmouth, UK, 24–27 August 2015; Lecture Notes in Computer Science; Liu, H., Kubota, N., Zhu, X., Dillmann, R., Zhou, D., Eds.; Springer International Publishing: Berlin/Heidelberg, Germany, 2015; pp. 581–591. [[CrossRef](#)]
18. Nogueira, R.; Reis, J.; Pinto, R.; Gonçalves, G. Self-adaptive Cobots in Cyber-Physical Production Systems. In Proceedings of the 2019 24th IEEE International Conference on Emerging Technologies and Factory Automation (ETFA), Zaragoza, Spain, 10–13 September 2019; pp. 521–528. [[CrossRef](#)]

19. Cueva, C.W.F.; Torres, S.H.M.; Kern, M.J. 7 DOF industrial robot controlled by hand gestures using microsoft kinect v2. In Proceedings of the 2017 IEEE 3rd Colombian Conference on Automatic Control (CCAC), Cartagena, Colombia, 18–20 October 2017; pp. 1–6. [CrossRef]
20. Kaczmarek, W.; Panasiuk, J.; Borys, S.; Banach, P. Industrial Robot Control by Means of Gestures and Voice Commands in Off-Line and On-Line Mode. *Sensors* **2020**, *20*, 6358. [CrossRef]
21. Zhang, X.; Zhang, R.; Chen, L.; Zhang, X. Natural Gesture Control of a Delta Robot Using Leap Motion. *J. Phys.* **2019**, *1187*, 032042. [CrossRef]
22. Cipolla, R.; Hollinghurst, N. A Human–Robot Interface using Pointing with Uncalibrated Stereo Vision. In *Computer Vision for Human-Machine Interaction*; Pentland, A., Cipolla, R., Eds.; Cambridge University Press: Cambridge, UK, 1998; pp. 97–110. [CrossRef]
23. Gunawardane, P.; Medagedara, N.T.; Madusanka, B.; Wijesinghe, S. The development of a Gesture Controlled Soft Robot gripping mechanism. In Proceedings of the 2016 IEEE International Conference on Information and Automation for Sustainability (ICIAfS), Galle, Sri Lanka, 16–19 December 2016; pp. 1–6. [CrossRef]
24. Devine, S.; Rafferty, K.; Ferguson, S. Real time robotic arm control using hand gestures with multiple end effectors. In Proceedings of the 2016 UKACC 11th International Conference on Control (CONTROL), Belfast, UK, 31 August–2 September 2016; pp. 1–5. [CrossRef]
25. Li, C.; Fahmy, A.; Sienz, J. Development of a Neural Network-Based Control System for the DLR-HIT II Robot Hand Using Leap Motion. *IEEE Access* **2019**, *7*, 136914–136923. [CrossRef]
26. Razjigaev, A.; Crawford, R.; Roberts, J.; Wu, L. Teleoperation of a concentric tube robot through hand gesture visual tracking. In Proceedings of the 2017 IEEE International Conference on Robotics and Biomimetics (ROBIO), Macau, Macao, 5–8 December 2017; pp. 1175–1180. [CrossRef]
27. Sarkar, A.; Patel, K.A.; Ganesh Ram, R.; Capoor, G.K. Gesture control of drone using a motion controller. In Proceedings of the 2016 International Conference on Industrial Informatics and Computer Systems (CIICS), Sharjah, United Arab Emirates, 13–15 March 2016; pp. 1–5. [CrossRef]
28. Hu, B.; Wang, J. Deep Learning Based Hand Gesture Recognition and UAV Flight Controls. In Proceedings of the 2018 24th International Conference on Automation and Computing (ICAC), Newcastle Upon Tyne, UK, 6–7 September 2018; pp. 1–6. [CrossRef]
29. Yu, N.; Xu, C.; Wang, K.; Yang, Z.; Liu, J. Gesture-based telemanipulation of a humanoid robot for home service tasks. In Proceedings of the 2015 IEEE International Conference on Cyber Technology in Automation, Control, and Intelligent Systems (CYBER), Shenyang, China, 8–12 June 2015; pp. 1923–1927. [CrossRef]
30. Cheng, L.; Sun, Q.; Su, H.; Cong, Y.; Zhao, S. Design and implementation of human–robot interactive demonstration system based on Kinect. In Proceedings of the 2012 24th Chinese Control and Decision Conference (CCDC), Taiyuan, China, 23–25 May 2012; pp. 971–975, ISSN 1948-9447. [CrossRef]
31. Liu, Y.; Zhang, Y. Toward Welding Robot With Human Knowledge: A Remotely-Controlled Approach. *IEEE Trans. Autom. Sci. Eng.* **2015**, *12*, 769–774. [CrossRef]
32. Yang, C.; Zeng, C.; Liang, P.; Li, Z.; Li, R.; Su, C.Y. Interface Design of a Physical Human–Robot Interaction System for Human Impedance Adaptive Skill Transfer. *IEEE Trans. Autom. Sci. Eng.* **2018**, *15*, 329–340. [CrossRef]
33. Krupke, D.; Steinicke, F.; Lubos, P.; Jonetzko, Y.; Görner, M.; Zhang, J. Comparison of Multimodal Heading and Pointing Gestures for Co-Located Mixed Reality Human–Robot Interaction. In Proceedings of the 2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Madrid, Spain, 1–5 October 2018; pp. 1–9. [CrossRef]
34. Zhou, H.; Hu, H. Human motion tracking for rehabilitation—A survey. *Biomed. Signal Process. Control* **2008**, *3*, 1–18. [CrossRef]
35. Vardhan, D.; Prasad, P.P. Hand Gesture Recognition Application for Physically Disabled People. *Int. J. Sci. Res.* **2014**, *3*, 765–769.
36. Parvini, F.; McLeod, D.; Shahabi, C.; Navai, B.; Zali, B.; Ghandeharizadeh, S. An Approach to Glove-Based Gesture Recognition. In Proceedings of the Human–Computer Interaction. Novel Interaction Methods and Techniques, San Diego, CA, USA, 19–24 July 2009; Lecture Notes in Computer Science; Jacko, J.A., Ed.; Springer: Berlin/Heidelberg, Germany, 2009; pp. 236–245. [CrossRef]
37. Allevard, T.; Benoit, E.; Foulloy, L. Fuzzy Glove For Gesture Recognition. In Proceedings of the 17th IMEKO World Congress, Dubrovnik, Croatia, 22–28 June 2003; pp. 2026–2031.
38. Ghunawat, M.R. Multi-point Gesture Recognition Using LED Gloves For Interactive HCI. *Int. J. Comput. Sci. Inf. Technol.* **2014**, *5*, 6768–6773.
39. Ganzeboom, M. How Hand Gestures Are Recognized Using a Dataglove. 2009. Available online: [https://www.researchgate.net/publication/228702251\\_How\\_hand\\_gestures\\_are\\_recognized\\_using\\_a\\_dataglove](https://www.researchgate.net/publication/228702251_How_hand_gestures_are_recognized_using_a_dataglove) (accessed on 19 December 2021).
40. Shotton, J.; Girshick, R.; Fitzgibbon, A.; Sharp, T.; Cook, M.; Finocchio, M.; Moore, R.; Kohli, P.; Criminisi, A.; Kipman, A.; et al. Efficient Human Pose Estimation from Single Depth Images. *IEEE Trans. Pattern Anal. Mach. Intell.* **2013**, *35*, 2821–2840. [CrossRef]
41. Guzsvinecz, T.; Szucs, V.; Sik-Lanyi, C. Suitability of the Kinect Sensor and Leap Motion Controller—A Literature Review. *Sensors* **2019**, *19*, 1072. [CrossRef]
42. Vysocký, A.; Grushko, S.; Oščádal, P.; Kot, T.; Babjak, J.; Jánoš, R.; Sukop, M.; Bobovský, Z. Analysis of Precision and Stability of Hand Tracking with Leap Motion Sensor. *Sensors* **2020**, *20*, 4088. [CrossRef]
43. Guna, J.; Jakus, G.; Pogačnik, M.; Tomažič, S.; Sodnik, J. An Analysis of the Precision and Reliability of the Leap Motion Sensor and Its Suitability for Static and Dynamic Tracking. *Sensors* **2014**, *14*, 3702–3720. [CrossRef]