AI-Enabled Framework for Augmenting Upper Limb Rehabilitation With a Social Robot

AI-Enabled Framework for Augmenting Upper Limb Rehabilitation With a Social Robot

1st Gloria Beraldo
Institute of Cognitive Sciences and Technologies
National Research Council of Italy (CNR-ISTC)
Rome, Italy
gloria.beraldo@cnr.it

2nd Albin Bajrami
DIISM Department
Polytechnic University of Marche
Ancona, Italy
a.bajrami@pm.univpm.it

3rd Alessandro Umbrico
Institute of Cognitive Sciences and Technologies
National Research Council of Italy (CNR-ISTC)
Rome, Italy
alessandro.umbrico@cnr.it

4th Gabriella Cortellessa
Institute of Cognitive Sciences and Technologies
National Research Council of Italy (CNR-ISTC)
gabriella.cortellessa@cnr.it

5th Matteo Claudio Palpacelli
DIISM Department
Polytechnic University of Marche
Ancona, Italy
m.palpacelli@univpm.it

***Abstract*—This paper proposes an innovative framework that is designed to strengthen human upper limb rehabilitation through the social robot TIAGo. Such a system is designed as a complementary aid used to encourage the patient during a training session by providing contextual feedback based on the comparison of his/her performance with respect to the expected motion. The ground truth is acquired in an initial stage via an accurate motion capture system, where the physiotherapist teaches the targeted motions to the robot. The result is an archive of rehabilitation exercises that the robot can deliver to the patient, the intent of which is to replicate them under the robot supervision. The approach followed in this work is mainly focused on re-targeting the human trajectories into the robot: first, by extracting the marks of the human arm joints with a skeleton tracking software based on the robot's camera images, then, by operating a scaling and an optimization of the robot arm motion employing the robot's kinematic model, with the aim of improving the naturalness of its movement.**

***Index Terms*—Rehabilitation, Robotic Arm Kinematics, Human-Robot Interaction**

I. Introduction

After critical events like strokes, heart attacks, or during progressive diseases such as Amyotrophic lateral sclerosis (ALS), an individual's musculoskeletal capacity may be significantly limited or worsen over time. Effective rehabilitation interventions are required not only to try to regain lost mobility, but also to slow the rate of functional decline.

This research received external funding from the project FOCAAL, FOg Computing in Ambient Assisted Living, funded by MIMIT, the Ministry of Enterprise and Made in Italy, under the Life Sciences Innovation Agreement DM 05/03/2018.

Robotics has been shown to provide significant benefits in this field, due to the use of advanced technologies such as robotic exoskeletons for walking and hand rehabilitation, virtual reality-assisted systems, and soft robotic devices that can assist patients with the complex challenges of rehabilitation [1]–[4]. Although such devices can support patients only on the physical side, socially-assistive robots (SARs) can provide assistance to end-users also through social interaction [5]. Thus, SARs are a promising solution to establish an empathetic interaction and motivate the patient during the rehabilitation process by making the experience more engaging [6]–[8]. The robot is exploited with a twofold purpose: (a) monitor the patient and analyze his/her motion; (b) actively guide the users through exercises aimed at improving daily function and increasing independence, adapting to individual needs. In such a context, traditionally, two figures are involved: the physiotherapist that chooses the rehabilitative exercise to be performed according to the expected healthcare goal; and the patient who is the primary user. The patient has to undergo a series of rehabilitation sessions that are typically performed in a gym environment shared by multiple people.

This paper proposes a novel framework for upper limb rehabilitation that relies on the social robot TIAGo, shown in Fig. 1. The robot aims at monitoring and providing contextual feedback that accurately guides patients in performing functional rehabilitation exercises, such as drinking from a glass of water or zipping up, when the physiotherapist is not there. The robot acquires the data related to the patient's movements and compares them with respect to expected performance using the kinesthetic motions taught by a physiotherapist

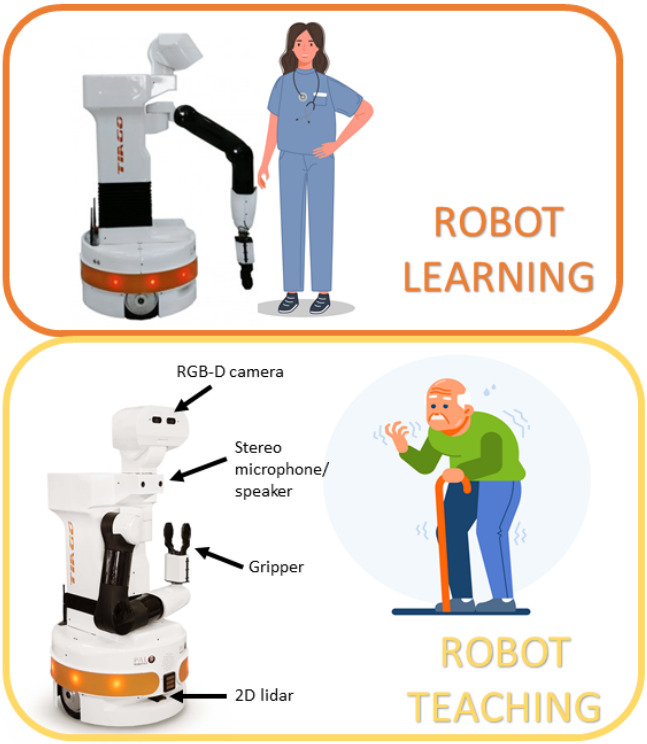

Fig. 1. In the upper part of the figure, the TIAGo robot first learns kinesthetic movements from a physiotherapist. Then, in the lower part of the figure, it can train each patient by providing contextual feedback thanks to its sensors by comparing the resulting motion with respect to the ground truth acquired in the first stage.

at the initial stage, as ground truth. One of the challenges behind the system is to ensure a robust re-targeting operation that aligns the physiotherapist's and the patient's motion into the robot arm. Previous studies using re-targeting based on artificial intelligence (AI) have, indeed, demonstrated that the movements reproduced by the robot were unstable and unreliable [9]. Differently, in [10], re-targeting is performed based on a motion capture system, Xsense, which is IMU based, where IMU stands for Inertial Measurement Unit. However, reconstructing the acceleration signal is a difficult task that produces errors in the trajectory. In the present work, to create an archive of reference rehabilitation exercises, the physiotherapist's hand trajectories together with the movements of his/her elbow and shoulder are extracted using the motion capture system called OptiTrack, a commercial product that is generally more accurate than IMU equipment. The OptiTrack software allows to extract the keypoints of the human arm joints by exploiting a skeleton tracking module based on high-resolution images, taken from fast cameras. A less accurate procedure can be performed by the physiotherapist in the absence of a high-quality equipment: targeted motions can be taught visually to the robot, by using its RGB cameras and extracting the human arm joint keypoints, retrospectively correcting the robot's movement with manual interventions. Another important development presented in this paper is a scaling operation and an optimization procedure for the robot's motion planning. These are proposed to make the robot's arm motion natural and human-like by employing the robot's kinematic model.

Finally, the system relies on multimodal AI services to evaluate movement precision and collect data for doctors, improving patient feedback, and enriching the clinical picture with detailed information. The system supports vocal command and voice communication to make the interaction more intuitive.

II. Framework Description

The proposed framework is illustrated in Fig. 2 and described here below.

A. Motion Acquisition with Optitrack

The Optitrack motion capture system is used to collect data on specific functional movements, such as drinking a glass of water. This enables detailed movements of the shoulder, elbow, wrist, and metatarsal to be captured, creating a customized movement model.

B. Simulation and Validation with MATLAB

The collected data are validated and adapted using the MATLAB environment to replicate human movements as closely as possible with the TIAGo robotic arm. MATLAB offers many tools to visually simulate the TIAGo robot performing desired tasks, in particular the Robotics System Toolbox. After defining the functional gestures to be replicated, an archive of reference rehabilitation movements is created. This will be used to plan the movements of TIAGo and verify, during the rehabilitation session, if the patient performs the corresponding movements correctly.

C. Integration with ROS and Docker

MATLAB interfaces with the Robot Operating System (ROS), the standard de facto in robotics, through Docker containers, which are minimal virtual machines that enable faster communication between different operating systems, such as Windows and Linux. Within Docker, ROS middleware and the skeleton tracking module for the pose analysis are used.

The choice to use MATLAB for offline motion planning and analysis is due to its versatility and efficiency as an engineering software, as well as enabling symbolic and numerical modeling of the kinematics of the TIAGo robot.

D. Real-Time Interaction and Feedback with TIAGo

The TIAGo robot's arm is used to guide patients through rehabilitation exercises. The system exploits the onboard TIAGo camera to analyze the movement execution and compare it with the optimal movement patterns defined in the Simulation Phase. Based on the patient's progress and comfort, adaptive feedback is provided, with suggested movements and exercise intensity adjusted in real-time. This approach prioritizes the patient in the rehabilitation process, maximizing the effectiveness of treatment and improving the user experience. The

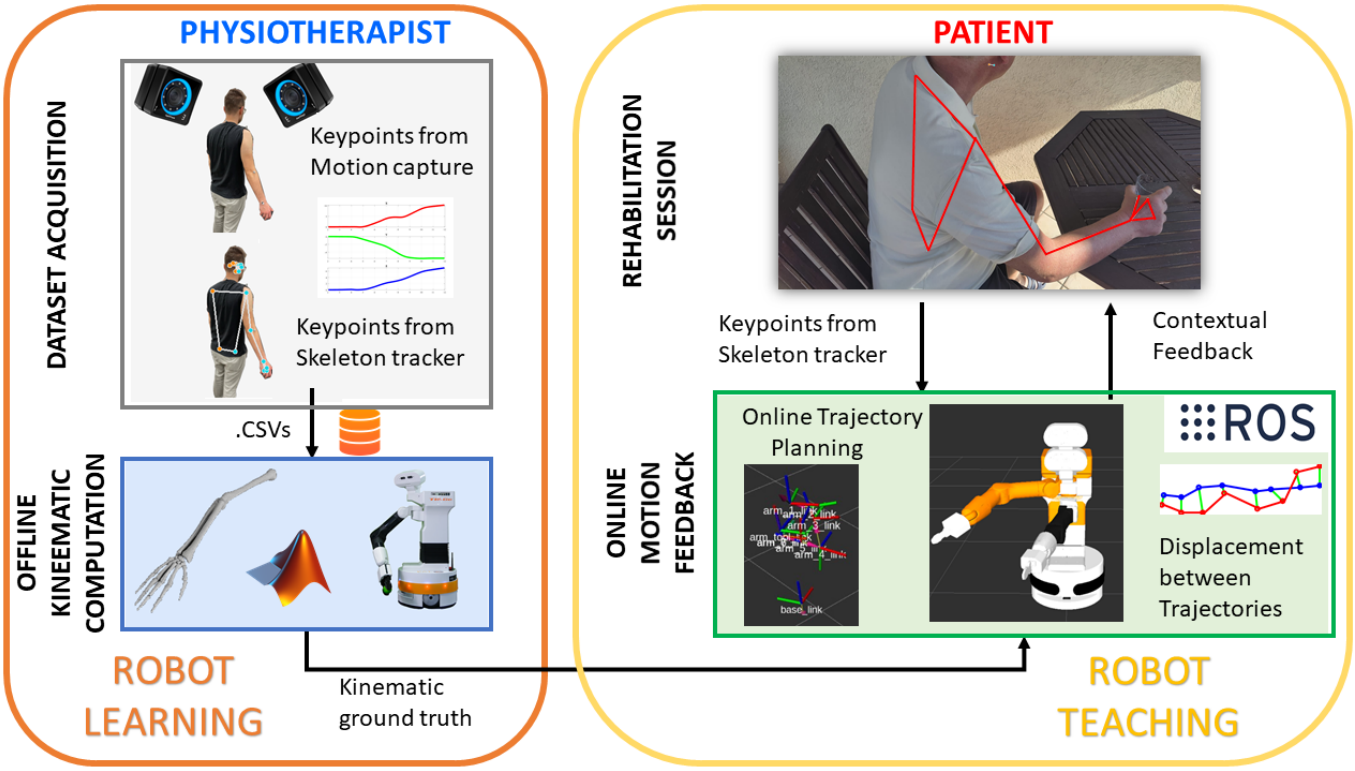

Fig. 2. Illustrative block diagram of the proposed framework for both robot learning from the physiotherapist and teaching to the patient.

system also collects valuable data on the patient's anatomical angles and progress, providing clinicians with detailed information on possible adjustments to the rehabilitation plan. By implementing a personalized feedback strategy managed through ROS and using Python, TIAGo can provide verbal or visual advice that adapts in real time to the patient's performance, encouraging and guiding them through each step of their rehabilitation.

III. Methods

A. Motion Capture System

The Optitrack system uses twelve high-speed precision cameras positioned around the room's perimeter to capture real-time motion using passive markers. These cameras are calibrated to ensure complete coverage of the area without blind spots, which is critical to accurately modeling movements.

The cameras' calibration involves calculating a camera matrix $\mathbf{P}$ that transforms 3D coordinates into 2D projections on the image plane, using intrinsic parameters (like focal length and principal point) and the camera's position and orientation. Triangulation from these 2D projections helps determine the 3D positions of the points. The process involves aligning these projections from at least two different camera views and adjusting the estimates to minimize discrepancies between the observed and expected positions.

This optimized alignment, known as bundle adjustment, refines the cameras' settings and the 3D point positions to enhance the accuracy of the motion model. For rigid bodies, maintaining consistent distances between markers allows precise pose estimation.

B. Human and Robotic Arm Kinematics

The reconstruction of the kinematics of a human arm is facilitated by tools provided by motion capture systems, which offer the positions of key points at the arm joints through 3D skeleton tracking, as well as by computer vision tools and machine learning systems. However, models of human arm kinematics are still useful for refining the acquired data and achieving higher stability in the results. Many alternatives can be found in the literature, involving different levels of detail, ranging from simple spherical-revolute-spherical (SRS) sequences of joints [11] to more advanced models with complex shoulder kinematics, for instance involving a rotation about the center of the sternoclavicular joint [12] or even more sophisticated kinematic architectures [13].

The kinematics of both the human arm and TIAGo robot's arm can be described using the same mathematical approach, based on 4×4 homogeneous transformation matrixes, called ${}^{i}_{j}\mathbf{T}$ in the following between rigid bodies $\{i\}$ and $\{j\}$, and kinematic constraints. The sequence of links and joints in a robotic (or human) arm, from the base (or shoulder) to the end-effector (or hand), defines the serial kinematic chain. Figure 3 illustrates the kinematics of TIAGo robot's arm, fully investigated in [14].

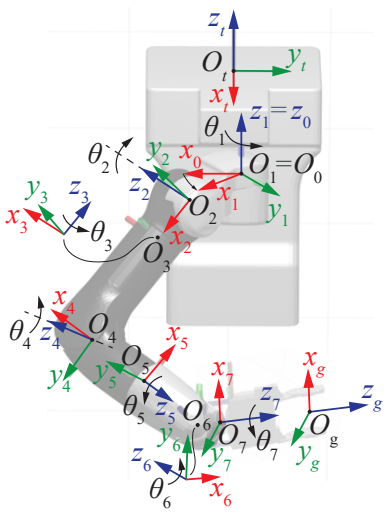

Fig. 3. Robotic arm kinematics of the TIAGo robot and its reference systems.

A common reference frame has been considered for both the robot and the human arm, with the origin placed at the root O_0 of the arm. For the human arm, this can be considered as the base of the neck. With reference to Fig. 3, it can be observed that the x_0 axis is directed in the opposite direction of the y_t axis, which is fixed to the robot's trunk, in order to ensure that the rotation of the first joint has only positive values. From O_0 to O_g, which represents the center of the hand or gripper, the direct kinematics of the robot arm can be written:

$$ {}^{0}_{g}\mathbf{T} = {}^{0}_{1}\mathbf{T}\left(\theta_1\right) \cdots {}^{6}_{7}\mathbf{T}\left(\theta_7\right) {}^{7}_{g}\mathbf{T} \tag{1} $$

where ${}^{0}_{g}\mathbf{T}$ represents the total transformation from the base to the center of the robot's gripper, and the matrix product on the right represents the sequence of rototranslations occurring between the adjacent reference systems positioned at the endpoints of each link. These links are connected to each other through revolute joints, whose z-axis rotations are given by the joint variables θ_i, with $i = 1, \ldots, 7$, gathered in the joint vector $\mathbf{q}$. Matrix ${}^{7}_{g}\mathbf{T}$ is only a constant rigid translation to reach point O_g. By looking at Fig. 3 it can be noted that the TIAGo robot was designed to replicate human arm kinematics as closely as possible, even approximating the mobility of a human shoulder.

C. Integrating OptiTrack Data with Robot Kinematics

OptiTrack data analysis identifies the reference trajectories of the key points of the human arm, in the form of position vectors:

$$ \mathbf{P}_{ref}(n) = \left[\mathbf{p}_s(n), \mathbf{p}_e(n), \mathbf{p}_w(n), \mathbf{p}_h(n)\right], \tag{2} $$

where $\mathbf{P}_{ref}(n)$ is a reference matrix, related to the time frame n of a time sequence, which gathers the absolute position vectors of the shoulder, elbow, wrist, and hand, called $\mathbf{p}_s$, $\mathbf{p}_e$, $\mathbf{p}_w$, and $\mathbf{p}_h$ respectively, with respect to a fixed frame taken as absolute reference. These trajectories have to be translated into corresponding actions for TIAGo robot's arm. In order to match the robot's movements to the patterns of the human arm, it is necessary to follow a procedure made of two steps:

1) *Scaling Operation*: after defining a common reference system on the shoulders of both the robot and the human arm, it is possible to resize the data available for the human arm by scaling them to the robot's size. The scale factor is given by $sc = l_r/l_h$, where l_r is the length of the robot arm in home configuration, represented by vector $(\mathbf{O}_g - \mathbf{O}_0)$ when the arm is fully extended, and l_h is obtained as the average length of the fully extended human arm, derived from motion capture system data. This step is performed only at the beginning of the procedure and provides the pose ${}^{0}_{g}\mathbf{T}$ of the robot's end-effector for the entire duration of the task.
2) *Robot Inverse Kinematics*: redundant kinematics robots allow infinite solutions for the inverse kinematics problem, where the term 'infinite' can also be understood as an exponential of infinite solutions, according to the null space of the Jacobian matrix. This means that an optimization strategy is required to solve the inverse kinematics. When the 6×7 analytical Jacobian matrix $\mathbf{J}_A(\mathbf{q})$ of the TIAGo robot's arm has full rank, infinite solutions can be found for a given pose of the end-effector. Higher-order infinities in the number of solutions can be achieved if more freedom is given to the end-effector, such as allowing certain rotations to remain unconstrained. For example, consider the task of grasping cylindrical objects. The redundancy in the robot's inverse kinematics can be used to mimic the behavior of the human arm. This allows the robot not only to follow a specific trajectory with its end-effector but also to adjust its arm configuration so that the plane of the robot's arm aligns as closely as possible with the plane of the human arm. Several approaches have been proposed in the literature to guide inverse kinematics in mimicking human arm behavior [15]–[17].

In order to perform an optimized inverse kinematics, a cost function can be defined:

$$ g(\mathbf{q}) = 1 - \left[\mathbf{n}^T \hat{\mathbf{z}}_4(\mathbf{q})\right]^2, \tag{3} $$

where $\mathbf{n} = (\mathbf{p}_e - \mathbf{p}_s) \times (\mathbf{p}_w - \mathbf{p}_e) / \| (\mathbf{p}_e - \mathbf{p}_s) \times (\mathbf{p}_w - \mathbf{p}_e) \|$ is the normal unit vector obtained from the motion capture system as the vector product of the human arm and forearm vectors, and $\hat{\mathbf{z}}_4$ is the joint axis between the arm and forearm of the TIAGo robot, depending on the joint configuration $\mathbf{q}$. Minimizing $g(\mathbf{q})$ means achieving a robot arm configuration that lies on a plane oriented as closely as possible to the plane of the human arm.

The inverse kinematics is solved using an extended Jacobian matrix approach, where the constraint of minimizing the cost function g is integrated into the rectangular Jacobian by means of the gradient $\nabla g = \partial g / \partial \mathbf{q}$, guiding the solution towards this objective [18]. A Damped Least Squares (DLS) algorithm is also implemented to provide robustness to the inverse solution

when singularities are encountered and the Jacobian matrix is ill-conditioned [19]. The analytical Jacobian matrix of the TIAGo robot changes accordingly, becoming a square 7×7 matrix:

$$\mathbf{J}_e = \begin{bmatrix} \mathbf{J}_A \\ \alpha \left(\nabla g^{-1}\right)^T \end{bmatrix}_{7\times 7}, \tag{4}$$

where the parameter α must be calibrated to assign more or less weight to the condition of planes parallelism.

An iterative algorithm based on the extended Jacobian matrix can be finally elaborated to solve the inverse kinematics problem:

$$\mathbf{q}_{i+1} = \mathbf{q}_i + \mathbf{J}_e^T \left(\mathbf{J}_e \mathbf{J}_e^T + \lambda^2 \mathbf{I}_7\right)^{-1} \Delta\mathbf{p}, \tag{5}$$

where $\mathbf{p}$ is the 6-dimensional position vector consisting of three Cartesian coordinates and three Euler angles, based on the chosen angle representation. $\Delta\mathbf{p} = \mathbf{p}_f - \mathbf{p}\left(\mathbf{q}_i\right)$ represents the error between the final pose and the pose evaluated at the i-th step, starting from an initial pose as the first attempt. A tolerance is imposed on the error $\Delta\mathbf{q} = \mathbf{q}_{i+1} - \mathbf{q}_i$ to determine when to exit the iteration. The parameter λ must be also calibrated according to the robot's behavior around singular configurations.

Further constraints can be used to prevent reaching the mechanical joint limits by introducing additional cost functions and weights that operate in the null space of the Jacobian matrix [20].

D. Vision System & Motion Assessment

To evaluate the patient's exercise performance, the robot exploits RGB-D camera located on its head. The acquired data is processed by a *skeleton tracker* that extracts the joints coordinates both in the pixels and world reference system. Such a module relies on *Mediapipe library* and has been previously analyzed with respect to the IMU data in our previous work [21]. The extracted poses are then compared with the ground truth acquired in the first stage. The alignment between the two trajectories is based on the Dynamic Time Warping algorithm (DTW), a popular and widespread algorithm used in several domains, such as speech recognition, human motion animation, human activity recognition, and time series classification [22]–[26].

The objective is to verify the correctness of the exercise performed, allowing TIAGo to give further feedback to the operator related to the shape of the trajectories associated with the patient's motion (e.g., trajectories similarity, Fréchet distance, smoothness, timing, angles between joints per exercise phase). Moreover, the robot trains the patient to perform the exercises correctly, optimizing the effectiveness of the treatment and motivating him/her via vocal interaction making the session more engaging.

IV. Preliminary experiments

Two preliminary tests were conducted: a preliminary evaluation of the ground truth creation and skeleton comparison with MediaPipe, and the evalutation of the robotic arm mimicry.

A. Creating and Comparing Ground Truth Trajectory: Pose Detection with Mediapipe vs. Optitrack

This preliminary experimental phase involved a detailed comparison between the ground truth and pose detection technologies, Mediapipe and Optitrack, to assess their effectiveness in accurately capturing human movements. The experiments took place in a controlled laboratory environment, capturing functional movements such as drinking from a glass (A#), pour water into a glass (B#), touching the nose (C#), and unzipping a zip (D#) (shown in Fig. IV-A), using only the right arm to ensure consistency across all tests. Twelve Optitrack cameras provided reduced occlusion and a RealSense D455 camera was used to capture motion data via ROS2 (latest version of ROS, Robot Operating System). Also, the reference frame of Optitrack and MediaPipe is shown in Fig. IV-A.

In the case of MediaPipe, the reference frame is standard and provided by the library. On the other hand, OptiTrack's reference system is arbitrary and in this case the upper left corner of the table has been chosen. In addition, the rototranslation matrix is obtained using the Singular Value Decomposition (SVD) by analyzing the points recorded between OptiTrack and MediaPipe. Note that SVD is applied to MediaPipe in relation to the OptiTrack reference system, so MediaPipe is rototranslated accordingly.

Each movement was repeated ten times to ensure sufficient data for analysis. This redundancy was intended to establish a solid ground truth for later comparisons. Motion data from the RealSense camera was processed via ROS2, capturing individual frames to facilitate detailed pose analysis.

For space constraints, herein, only the analysis of the first gesture sequence (A#) is shown, for the sake of conciseness.

A frame-by-frame analysis was conducted using MediaPipe to extract keypoints, which were then compared with those obtained independently via OptiTrack. This analysis was performed using a MATLAB code to calculate rotation matrices and translation vectors between the MediaPipe and OptiTrack datasets. This approach facilitated synchronisation and accurate comparison between the two tracking methods. The MATLAB code used for the comparison is based on the following algorithm steps:

1) Importing data from CSV files containing the corresponding detections;
2) Synchronization and FPS (frames per second) alignment, using the Dynamic Time Warping (DTW) algorithm to temporally align data acquired from different technologies, ensuring accurate comparison of motion sequences;
3) Calculation of the coordinates of the transformed keypoints using rotation matrices and translation vectors, leveraging SVD (Singular Value Decomposition).

This analysis has revealed a small discrepancy between the Mediapipe Prediction and Optritrack acquisition, as illustrated in Fig. 5. Notably, MediaPipe relies only on images from an RGB camera for the detections. Consequently, the predictions tend to align with the camera plane. The Optitrack points, marked with asterisks, have a dispersion in the 3D coordinate

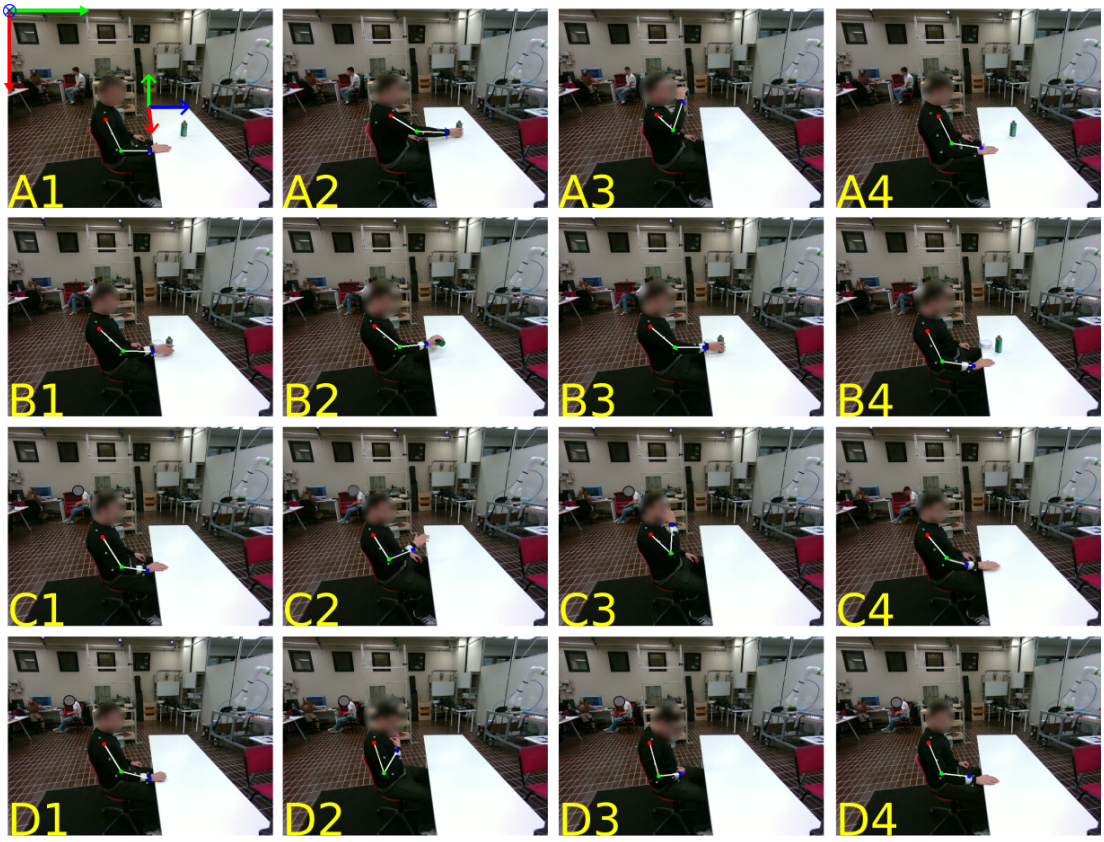

Fig. 4. Sequences of everyday actions with the right arm: (A) drinking, (B) pouring water, (C) touching the nose, (D) unzipping. The key points of the shoulder, elbow and wrist are highlighted in red, green and blue respectively, with white segments in between. The reference system on the top left is that of MediaPipe, while the one on the table is that of OptiTrack.

frame. This discrepancy is relatively small as shown in Fig. 6 and summarised in Table I. In particular, the shoulder detected by OptiTrack is on average about 20 mm off from that detected

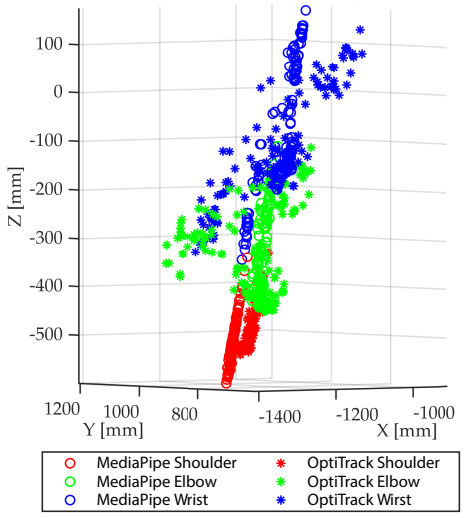

Fig. 5. 3D scatter plot comparing human body motion captured by MediaPipe (circles) and OptiTrack (stars) in shoulder, elbow, and wrist movements. For better readability, not all points are shown.

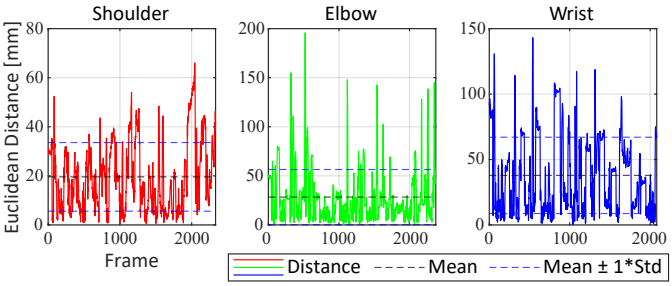

Fig. 6. Discrepancy in the number of frames between the shoulder, elbow and wrist, after applying the discrete time warping (DTW) technique, which modifies the length of the signal in order to achieve a superior degree of synchronization between the Optitrack and Mediapipe data.

by MediaPipe. However, the elbow and wrist show increasing errors as they accumulate the errors of the previous joints.

B. Robotic Arm Mimicry

The test on the mimicry of the robotic arm is conducted in simulation, focusing on kinematics inversion and the optimization and convergence of the algorithm. Once a time-based joint motion law is determined, the robot can easily follow these joint laws.

TABLE I
EUCLIDEAN DISTANCES AND STANDARD DEVIATIONS

Body Part	Eucliden Distance	
	Mean [mm]	Std [mm]
Shoulder	19.59	13.98
Elbow	28.38	28.10
Wrist	37.88	29.15

Data from the motion-capture vision system can be processed to obtain the homogeneous transformation ${}^{0}_{h}\mathbf{T}$ of the human hand over time while performing the task. Position is obtained directly from the coordinates of the markers placed on the hand or from the available skeleton tracking algorithms, as indicated in Equation (2). The rotation matrix, on the other hand, is reconstructed by defining the direction from the center of the wrist to the center of the hand as the z-axis of the terminal triad and assuming a y-axis that is always horizontal, to facilitate the gripper in eventually grasping objects, such as a bottle on top of a table.

The algorithm proposed in Sec. III-B is then applied to scale the data to the robot's size and find the optimized pose to closely match the human arm's posture. In Fig. 7, the motion of a human arm is simulated in MATLAB to demonstrate its behavior when moving from a gathered configuration to an extended one, where the grasping operation is intended to take place. The trajectory of point H represents the position of the human hand over time, while the blue vector $(\mathbf{H}-\mathbf{W})$ indicates the direction of the z-axis of the hand frame. The reference frame $O_n - x_n y_n z_n$ is placed at the base of the neck.

The scaled data is input into the inverse kinematics algorithm, along with the normal vector to the human arm recorded at the same time instants. Figure 8 shows the optimized motion, highlighting that while TIAGo's arm cannot perfectly match the imposed Cartesian constraints, the algorithm finds configurations with plane orientations closest to the human arm. Note the cyan unit vectors orthogonal to the robotic arm's plane and the black unit vectors orthogonal to the human arm's plane at corresponding times. The figure also reveals that the end-effector trajectory (in magenta) deviates from the reference trajectory (in black). This deviation occurs because the inverse kinematics algorithm balances the need to follow the trajectory, avoid singularities for continuous joint movement, and emulate the human arm's plane of motion. Table II presents the simulation results, showing the errors in Cartesian position, $\|\Delta\mathbf{p}\|$, and rotation, as the Euler angles difference, $\|\Delta\boldsymbol{\Theta}\|$, in solving the iterative algorithm: a higher weight is given to singularity handling than to plane similarity to ensure a stable solution. The high number of iterations required is not a limitation, as real-time requirements are not imposed at this stage.

Finally, some frames from the simulation are extracted and used in MATLAB Robotics Toolbox to demonstrate the TIAGo robot's behavior in performing the task, as shown in Fig. 9.

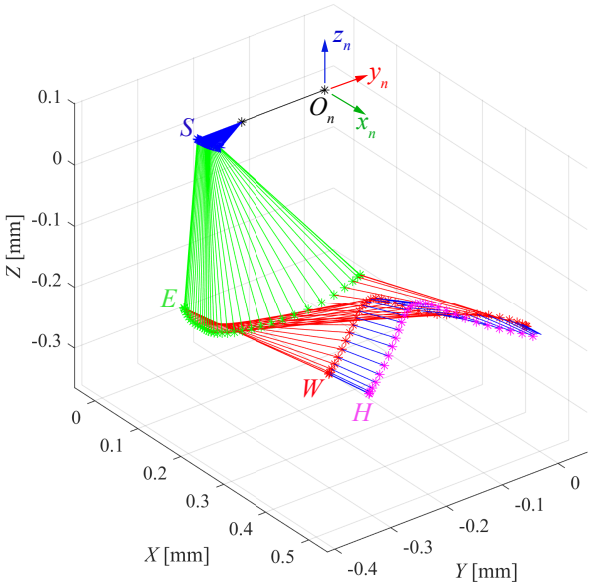

Fig. 7. Extension of the right human arm, showing the movement from a resting configuration to a grasping position.

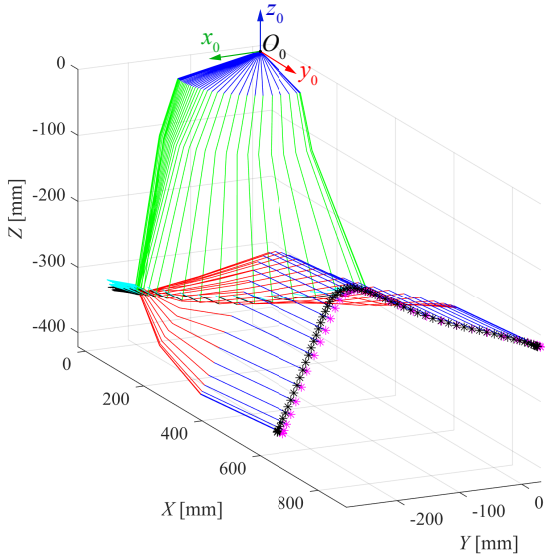

Fig. 8. Optimized robot poses based on the minimizing of function $g(\mathbf{q})$.

TABLE II
CARTESIAN AND PLANE INCLINATION ERRORS
$(\lambda = 1, \alpha = 0.03)$

Error	min	max
$\|\Delta\mathbf{p}\|$	4.4 mm	11.8 mm
$\|\Delta\boldsymbol{\Theta}\|$	0.01°	0.24°
γ	10.37°	20.38°

MEAN ITER $\approx$ 1500, MAX ITER $\approx$ 4000

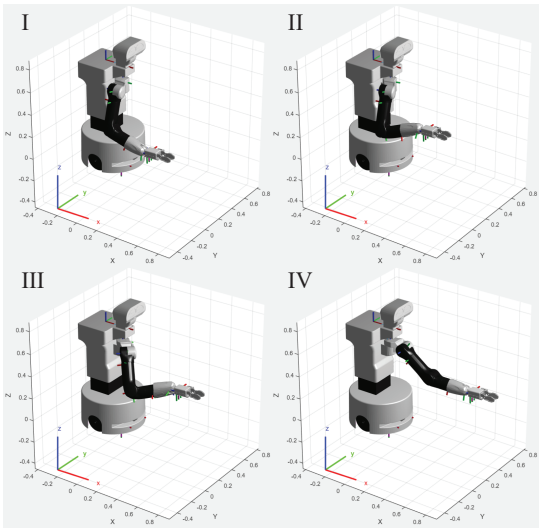

Fig. 9. Optimized motion performed by the TIAGo Robot.

V. Conclusion

This paper proposes an innovative AI-enabled framework for enhancing upper limb rehabilitation using the TIAGo social robot. The system operates with two users: (a) the physiotherapist, who teaches the robot the appropriate motions, and (b) the patient, who receives personalized feedback from the robot. The TIAGo robot is designed for use in shared environments like rehabilitation gyms, where the physiotherapist cannot directly monitor patients. During training, the robot guides patients through exercises and collects data via its integrated vision system. This data enriches the clinical understanding of patient progress and aids in biomechanical analysis, adapting the rehabilitation process accordingly.

This preliminary work highlights the system's main components, focusing on re-targeting operations and computing the robot's kinematics for planning optimized and natural trajectories. The results demonstrate the potential of the system and the ROS-based robotic platform, which offers a wide range of tools and openness. Future work will aim to close the interaction loop with patients, using their motor feedback and assessing treatment effectiveness through questionnaires.

References

[1] A. W. Heinemann, A. Jayaraman, C. K. Mummidisetty, J. Spraggins, D. Pinto, S. Charlifue, C. Tefertiller, H. B. Taylor, S.-H. Chang, A. Stampas *et al.*, "Experience of robotic exoskeleton use at four spinal cord injury model systems centers," *Journal of Neurologic Physical Therapy*, vol. 42, no. 4, pp. 256–267, 2018.

[2] A. Montagner, A. Frisoli, L. Borelli, C. Procopio, M. Bergamasco, M. C. Carboncini, and B. Rossi, "A pilot clinical study on robotic assisted rehabilitation in vr with an arm exoskeleton device," in *2007 Virtual Rehabilitation*. IEEE, 2007, pp. 57–64.

[3] M. Haghshenas-Jaryani, R. M. Patterson, N. Bugnariu, and M. B. Wijesundara, "A pilot study on the design and validation of a hybrid exoskeleton robotic device for hand rehabilitation," *Journal of Hand Therapy*, vol. 33, no. 2, pp. 198–208, 2020.

[4] D. Kinnett-Hopkins, C. K. Mummidisetty, L. Ehrlich-Jones, D. Crown, R. A. Bond, M. H. Applebaum, A. Jayaraman, C. Furbish, G. Forrest, E. Field-Fote *et al.*, "Users with spinal cord injury experience of robotic locomotor exoskeletons: a qualitative study of the benefits, limitations, and recommendations," *Journal of neuroengineering and rehabilitation*, vol. 17, pp. 1–10, 2020.

[5] A. Tapus, M. J. Mataric, and B. Scassellati, "Socially assistive robotics [grand challenges of robotics]," *IEEE robotics & automation magazine*, vol. 14, no. 1, pp. 35–42, 2007.

[6] J. Fasola and M. J. Mataric, "Using socially assistive human–robot interaction to motivate physical exercise for older adults," *Proceedings of the IEEE*, vol. 100, no. 8, pp. 2512–2526, 2012.

[7] B. Görer, A. A. Salah, and H. L. Akın, "A robotic fitness coach for the elderly," in *Ambient Intelligence: 4th International Joint Conference, AmI 2013, Dublin, Ireland, December 3-5, 2013. Proceedings 4*. Springer, 2013, pp. 124–139.

[8] ——, "An autonomous robotic exercise tutor for elderly people," *Autonomous Robots*, vol. 41, pp. 657–678, 2017.

[9] Y. Yan, E. V. Mascaro, and D. Lee, "Unsupervised human-to-robot motion retargeting via expressive latent space," *arXiv preprint arXiv:2309.05310*, 2023.

[10] M. Arduengo, A. Arduengo, A. Colomé, J. Lobo-Prat, and C. Torras, "Human to robot whole-body motion transfer," in *2020 IEEE-RAS 20th International Conference on Humanoid Robots (Humanoids)*. IEEE, 2021, pp. 299–305.

[11] G. Chiriatti, A. Bottiglione, and G. Palmieri, "Manipulability optimization of a rehabilitative collaborative robotic system," *Machines*, vol. 10, no. 6, p. 452, 2022.

[12] S. Williams, R. Schmidt, C. Disselhorst-Klug, and G. Rau, "An upper body model for the kinematical analysis of the joint chain of the human arm," *Journal of Biomechanics*, vol. 39, no. 13, pp. 2419–2429, 2006.

[13] N. Klopčar and J. Lenarčič, "Kinematic model for determination of human arm reachable workspace," *Meccanica*, vol. 40, pp. 203–219, 2005.

[14] A. Bajrami, M.-C. Palpacelli, L. Carbonari, and D. Costa, "Posture optimization of the tiago highly-redundant robot for grasping operation," *Robotics*, vol. 13, no. 4, p. 56, 2024.

[15] M. Alibeigi, S. Rabiee, and M. N. Ahmadabadi, "Inverse kinematics based human mimicking system using skeletal tracking technology," *Journal of Intelligent & Robotic Systems*, vol. 85, pp. 27–45, 2017.

[16] W. Liu, D. Chen, and J. Steil, "Analytical inverse kinematics solver for anthropomorphic 7-dof redundant manipulators with human-like configuration constraints," *Journal of Intelligent & Robotic Systems*, vol. 86, pp. 63–79, 2017.

[17] S. Li, K. Han, P. He, Z. Li, Y. Liu, and Y. Xiong, "Human-like redundancy resolution: An integrated inverse kinematics scheme for anthropomorphic manipulators with radial elbow offset," *Advanced Engineering Informatics*, vol. 54, p. 101812, 2022.

[18] J. Park, W. Chung, and Y. Youm, "Behaviors of extended jacobian method for kinematic resolutions of redundancy," in *Proceedings of the 1994 IEEE International Conference on Robotics and Automation*, vol. 1, 1994, pp. 89–95.

[19] D. Di Vito, C. Natale, and G. Antonelli, "A comparison of damped least squares algorithms for inverse kinematics of robot manipulators," *IFAC-PapersOnLine*, vol. 50, no. 1, pp. 6869–6874, 2017.

[20] A. Atawnih, D. Papageorgiou, and Z. Doulgeri, "Kinematic control of redundant robots with guaranteed joint limit avoidance," *Robotics and Autonomous Systems*, vol. 79, pp. 122–131, 2016.

[21] G. Beraldo, R. De Benedictis, A. Cesta, F. Fracasso, and G. Cortellessa, "Toward ai-enabled commercial telepresence robots to combine home care needs and affordability," *IEEE Robotics and Automation Letters*, 2023.

[22] L. Rabiner and B. Juang, "Fundamentals of speech recognition. ptr prentice hall; englewood cliffs, nj: 1993."

[23] E. Hsu, K. Pulli, and J. Popović, "Style translation for human motion," in *ACM SIGGRAPH 2005 Papers*, 2005, pp. 1082–1089.

[24] K. Kulkarni, G. Evangelidis, J. Cech, and R. Horaud, "Continuous action recognition based on sequence alignment," *International Journal of Computer Vision*, vol. 112, pp. 90–114, 2015.

[25] Y. Chen, E. Keogh, B. Hu, N. Begum, A. Bagnall, A. Mueen, and G. Batista, "The ucr time series classification archive," 2015.

[26] L. Ye and E. Keogh, "Time series shapelets: a new primitive for data mining," in *Proceedings of the 15th ACM SIGKDD international conference on Knowledge discovery and data mining*, 2009, pp. 947–956.