Computing and Informatics, Vol. 45, 2026, 936–963, doi: 10.31577/cai_2026_4_936

DEVELOPMENT OF A TELEOPERATION SYSTEM USING MOTION CAPTURE METHODOLOGIES FOR BAXTER ROBOT

Jiaying Huang

Guangzhou Academy of Fine Arts, Guangzhou, 510000, China
e-mail: huangjiaying@gzarts.edu.cn

Youssef Essa, Chunxu Li*

Mechanical Engineering Department
Faculty of Science and Engineering
Swansea University, SA1 8EN, UK
e-mail: {801253, chunxu.li}@swansea.ac.uk

Abstract. This paper presents the development and implementation of a teleoperation system using motion capture methodologies for the Baxter robot. The system utilizes a Microsoft Kinect V2 camera to capture human motion data and translates it into robot joint angles in real time, enabling intuitive control of the robot's movements. A function approximation technique (FAT) based controller was implemented to enhance the system's stability and performance while compensating for unknown dynamics. The mathematical framework incorporates the Denavit-Hartenberg model for mapping human arm movements to the robot's 7-DOF structure, along with space vector calculations for joint angle determination. Experimental results demonstrate successful real-time motion replication across multiple joint configurations, with the robot effectively mimicking human arm movements. Data analysis of joint positions, velocities, and accelerations confirms the system's capability to maintain accuracy while managing inherent challenges such as network latency and motion jittering. The developed system proves efficient and cost-effective, utilizing readily available hardware components while achieving reliable performance. This work contributes to advancing human-robot interaction in applications where direct human presence may be hazardous or impractical, offering

* Corresponding author

potential benefits across industries including manufacturing, healthcare, and hazardous environment operations.

Keywords: Teleoperation, motion capture, function approximation technique (FAT), Baxter robot, human–robot interaction, neural networks

Mathematics Subject Classification 2010: 70E60, 68T40, 93C40

1 INTRODUCTION

Technological advancements in the field of robotics have enhanced and influenced the integration of automation in fields ranging from manufacturing to healthcare services. These advancements have created a variety of applications to use in order to achieve the desired outcomes with maximum possible efficiency. Among various robotics applications, teleoperation systems stand out for their ability to enable human operators to control robotic systems remotely, combining the precision of a robot with human adaptability. This teleoperation system allows users to execute complex tasks in dangerous situations without the risk of being harmed. For example, a review conducted on the use of teleoperation methods for heavy machinery in the field of construction stated that implementing teleoperation systems allows workers to perform tasks in dangerous or remote construction environments, which enhances safety and efficiency [1].

The effectiveness of these systems depends heavily on the accuracy and speed with which human movements are captured and translated into robotic movements. This dependence on the speed and accuracy of the data captured creates a limitation in real-life scenarios, as delay and background noise could be present and could complicate the process for the sensors used. These limitations create a bubble that surrounds these technologies where, for these technologies to be implemented, it would require expensive tools and experienced personnel, which would be difficult to implement in industries. It would be easy to implement if the robot would not require expensive equipment, while maintaining the ability to translate the human motion perfectly and obtain the ability to adapt to the surroundings. In other words, it would be best for robots to be able to recreate human motion in real time without expensive and complex equipment.

The motion capture methodologies can be narrowed down to two main systems: wearable sensors-based systems and vision-based systems. The wearable sensor-based systems can be described as sensors attached to the users' body to capture the precise skeletal movement; however, they require a complex setup, and they are labelled as disturbing which makes them less ideal for applications that require user comfort and simplicity [2]. The vision-based systems are described as a system which rely on cameras and depth sensors to capture the user's motion without requiring physical attachments or markers. These systems are simple and cost-effective, which makes them suitable to be used for teleoperation without complex and expensive setup. However, their dependability on the lighting available in the place of use affects the tracking accuracy in a dynamic environment [3].

In this paper, an experiment was conducted on the Baxter robot to test the response levels of the robot and to test if the robot will be able to successfully mimic the user's movements in real time. The paper will consist of a review, methodology, results, and the conclusion sections, where all the experiment's details, procedure, and results will be explained thoroughly for the reader.

2 RELATED WORKS

Teleoperation and motion capture can be classified as two separate technologies that, through enhancement and the evolution of modern technology, have gone through different forms. Teleoperation systems were developed to enable human operators to control robots in various environments, especially in environments where it is dangerous for the operator to achieve the desired task. The integration of motion capture methodologies with robotic systems enhances the field of teleoperation of robotics, offering more interactive and precise control mechanisms. This literature review examines the current state of teleoperation systems, focusing on motion capture-based approaches and their application to the Baxter robot platform.

2.1 Early Development of Teleoperation Systems

The concept of teleoperation, or remotely controlling machines, was created, or discovered, in the middle of the 20^{th} century when an engineer named Raymond Goertz developed the master–slave manipulators at Argonne National Laboratory in 1949 [4]. This mechanical system, designed to handle radioactive material, employed direct mechanical linkages between the operator's control device and the remote manipulator which helped to establish the foundation of the master–slave connection that continues to influence modern-day systems [5]. In 1954, Goertz developed the mechanical system into an electro-mechanical system, model E1, by introducing the servo control mechanisms and separating the master and slave components physically, thus allowing for a greater flexibility in operation [5].

Using prior advancements like Goertz's invention, the space race in the 1960s and 1970s have significantly accelerated the teleoperation development. NASA's investment in the teleoperation technology for space exploration led to numerous innovations, including supervisory control systems and predictive displays to compensate for the communication delays and to improve operational efficiency [6]. Bilateral control schemes also emerged, allowing force feedback to the operator, enhancing precision and task performance [6]. In the 1970s, the integration of haptic feedback systems was introduced, allowing operators to feel tactile sensations from remote environments. This breakthrough improved precision and control, paving the way for advanced applications in fields such as medical surgery, where surgeons could perform complicated and delicate procedures remotely, and underwater explorations, allowing for complex operations in deep-sea environments [7].

In the 1980s, the rise of computer-aided systems showed significant advancements in teleoperation technology. Digital control and graphical interfaces were introduced, allowing for real-time data processing and more intuitive user interaction, which improved the efficiency and ease of use [8]. This period also brought the development of shared control architectures, where automated functions worked alongside operator commands, particularly benefiting underwater operations like offshore oil exploration [9]. Additionally, underwater robotics progressed with the widespread use of remotely operated vehicles (ROVs). These ROVs became essential for marine research and industrial tasks, such as inspecting pipelines, enabling operators to work effectively in harsh underwater environments without needing to be physically present [10].

In the 1990s, advancements in virtual reality and haptic feedback technologies enhanced teleoperation systems, creating more immersive and user-friendly interfaces. This period also saw the introduction of Internet-based teleoperation protocols, enabling long-distance operation but presenting challenges such as time delays and network reliability [11]. During this time, teleoperation systems began incorporating semi-autonomous capabilities, allowing robots to perform routine tasks independently, with human input reserved for more complex operations. These innovations expanded the applications of teleoperation, especially in areas requiring precision and safety [12]. Additionally, improved communication technologies led to the rise of telepresence systems, which gave operators a sense of being present in remote environments. This proved especially useful in high-risk situations like disaster response and military operations, where minimizing human exposure was critical [13].

2.2 Motion Capture Systems

Optical motion capture systems have transformed how human movements are converted into robot commands. These systems use special cameras to track markers or key features on the operator's body with high accuracy. High-speed cameras, working at up to 480 Hz, capture the 3D positions of reflective or active markers, allowing real-time motion tracking with precision up to sub-millimetre levels [14, 15]. Advanced systems use algorithms to identify markers and reconstruct their paths, solving issues like marker overlap or noise. Modern marker-less systems now use deep learning to get skeletal data directly from RGB or RGB-D camera feeds, removing the need for physical markers while keeping a high level of accuracy. Advances in computer vision and AI have made these systems effective even in poor lighting or complex environments. Using multiple cameras, up to 24, provides better coverage and redundancy. Many systems now include real-time calibration to adjust for camera movement and environmental changes, maintaining accuracy during long use [15, 16].

Inertial measurement technology has become a powerful alternative to optical motion capture systems, offering notable advantages in mobility and operational versatility. Modern inertial measurement unit (IMU)-based systems utilize miniaturized micro-electromechanical systems (MEMS) sensors, combining accelerometers, gyroscopes, and magnetometers to deliver detailed motion data. These systems excel in tracking complex movements with high temporal resolution, often sampling at frequencies above 1 000 Hz [17, 18].

Recent advancements in sensor fusion algorithms have significantly enhanced the accuracy of IMU-based systems. Sophisticated Kalman filtering techniques, integrated with biomechanical models, enable precise estimation of joint angles and body segment positions. Additionally, the incorporation of magnetometers, although their performance can be affected by magnetic interference, provides an absolute reference for orientation, particularly in industrial environments [18, 19].

Hybrid systems that integrate IMU and optical technologies have shown considerable potential in teleoperation applications. These systems combine the high update rate and robustness of IMUs with the drift-free precision of optical systems. Through advanced integration algorithms, hybrid systems optimize the fusion of data from both sources, delivering motion tracking capabilities that surpass the limitations of standalone technologies [20].

2.3 Baxter Robot

The Baxter robot represents a milestone in the field of collaborative robotics, designed to enable safe and efficient interaction with humans in shared workspaces. Each of its dual arms is equipped with seven degrees of freedom, closely replicating the kinematics of human arms and facilitating the execution of complex manipulation tasks. The inclusion of series elastic actuators provides inherent mechanical compliance, which enhances safety while maintaining a positioning accuracy of ± 5 mm [21].

The control system of Baxter is built on the Robot Operating System (ROS), offering flexibility for integration with external devices and the development of custom software solutions. The platform includes integrated force sensing at each joint, enabling precise force control and reliable collision detection. Its end effectors are designed to be modular, allowing customization with various tools and sensors to adapt to specific task requirements [21].

Safety is a cornerstone of Baxter's design, with a robust control system incorporating multiple layers of safety features. These include velocity and force monitoring, joint-level impedance control, and emergency stop mechanisms. Additionally, advanced trajectory planning algorithms are employed to generate smooth and natural motion while strictly adhering to safety constraints. This combination of features makes Baxter a versatile and reliable platform for collaborative tasks [21]. The Baxter robot has proven to be a versatile platform utilized across a wide range of research domains, particularly excelling in teleoperation and collaborative tasks. In manufacturing, Baxter's compliant control capabilities have been employed in systems designed for complex assembly tasks. These systems enable precise part mating while ensuring safe interaction with human operators, making Baxter an effective tool for shared workspaces [22].

In healthcare, researchers have explored applications such as rehabilitation exercises and medical training simulations, leveraging Baxter's programmable impedance and inherent safety features. These attributes allow the robot to safely interact with patients or trainees, providing a controlled environment for learning and therapeutic exercises [23]. The educational sector has greatly benefited from Baxter's user-friendly programming interface and integrated safety features. Studies indicate that hands-on interaction with Baxter significantly improves students' understanding of robotics concepts, including kinematics, dynamics, and control theory [24]. The robot's intuitive design makes it an effective teaching tool for both beginners and advanced learners.

Additionally, Baxter's dual-arm configuration has enabled innovative research into coordinated manipulation tasks. Researchers have developed novel control strategies for bimanual operations, particularly in scenarios requiring complex object manipulation or tool use. This capability has expanded the robot's applicability to tasks demanding high levels of coordination and dexterity, further demonstrating its research potential [25].

2.4 Integration of Motion Capture with Baxter

The integration of motion capture data with Baxter's control system has introduced both challenges and opportunities in teleoperation research. Mapping human motion to robotic movements requires advanced algorithms that account for differences in kinematics, workspace constraints, and dynamic limitations. Researchers have explored various strategies to overcome these challenges, tailoring their approaches to specific applications.

Direct joint mapping strategies establish a one-to-one correspondence between human and robot joint angles. While this method provides intuitive control for operators, it must address discrepancies in joint configurations and ranges of motion between human and robotic arms. To overcome these issues, advanced algorithms employ optimization techniques to resolve redundancies, maintain natural motion, and respect joint limits while avoiding singularities [26].

Task-space control methods have proven especially effective for teleoperation. These techniques map human hand movements to robot end-effector positions using inverse kinematics to compute appropriate joint configurations. Recent advancements in task-space control include impedance modulation, enabling the robot to adapt its responses based on task requirements and environmental interactions, which enhances flexibility and precision [27].

Optimization-based mapping approaches consider multiple objectives simultaneously, such as task efficiency, energy conservation, and operator ergonomics. These methods utilize real-time optimization to generate robot trajectories that meet defined goals while ensuring smooth, natural movements. Incorporating machine learning techniques has further improved the prediction of operator intent, resulting in more fluid and responsive robot behaviour [28].

The implementation of effective real-time control systems in teleoperation requires addressing multiple factors that influence system performance. Modern teleoperation systems for Baxter utilize advanced state estimation algorithms to combine data from various sensors, ensuring accurate and robust motion tracking. These systems must process operator inputs with minimal latency to maintain stability and safety during operation [29].

Feedback mechanisms are integral to teleoperation performance, providing operators with critical information about the robot's state and task execution. Visual feedback systems have advanced from basic camera feeds to augmented reality (AR) overlays that display task-relevant data, such as force vectors, proximity warnings, and execution progress. Some systems have incorporated predictive displays that address communication delays by presenting anticipated robot positions based on operator inputs, enhancing responsiveness and precision [30]. Force feedback systems play a vital role in conveying haptic information about the robot's interaction with its environment. By leveraging Baxter's integrated force sensing capabilities, these systems provide operators with real-time feedback, improving task performance and situational awareness. Recent innovations have also explored vibrotactile feedback as an alternative or complementary method to traditional force feedback, offering a different modality for conveying interaction cues and enhancing operator experience [31, 32]. Compared with previous Kinect-based teleoperation systems [33], our approach integrates an FAT-based adaptive control strategy to improve motion stability under network-induced jitter and simplifies real-time joint mapping through a space-vector method without requiring additional IMU sensors, achieving a cost-effective yet accurate teleoperation framework.

3 PRELIMINARIES

The experimental setup consists of three main components: a motion capture system utilizing Microsoft Kinect V2, a control PC running the motion capture software, and a dedicated Linux-based workstation controlling the Baxter robot. These components are interconnected via a local router to facilitate data transmission between the motion capture and robot control systems. The overall setup can be found in Figure 1.

3.1 Kinect and SDK

The Microsoft Kinect 2$^{\text{nd}}$ generation (V2) camera was utilized in the teleoperation control system to capture the operator's movement in 3D. The reason for this choice is due to the Kinect's cheap price and output accuracy. In a study [34], the author concludes that the Kinect V2 camera offers enhanced depth sensing accuracy, higher resolution, improved tracking of additional skeletal nodes, better performance in

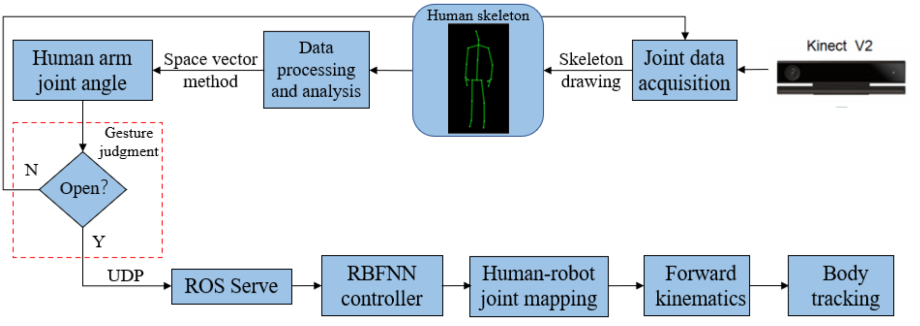


Figure 1. System architecture of the Kinect-based teleoperation framework: The Kinect V2 sensor captures skeletal data at 30 Hz, processed via the Kinect SDK and transmitted to the host PC; processed joint angles are relayed to the Baxter robot controller through the ROS communication framework (UDP protocol). Within the Baxter system, an FAT-based adaptive control loop operates at 100 Hz to ensure stable joint tracking and smooth motion replication. The diagram illustrates the real-time data flow, encompassing skeletal data acquisition, joint angle computation, network transmission, and robot control execution.

capturing dynamic motions, reduced stray motion errors, and increased precision in motion asymmetry analysis, making it significantly more suitable for detailed real-time applications than the previous versions. The Kinect V2 camera, as shown in Figure 2, consists of three components: RGB camera, Depth sensor, and the microphone array. As the microphone array is not included in the experiment, the main components needed is the camera and the depth sensor. The RGB camera in the Kinect sensor captures colour images of the surroundings, which helps provide visual context and supports the mapping of skeletal joint positions in alongside depth information data. The Kinect V2 depth sensor uses time-of-flight (ToF) technology, where an infrared emitter projects light into the environment, and an infrared camera captures the reflected light to calculate depth based on the time it takes for the light to return, enabling the creation of detailed 3D maps [34, 35].

The Kinect V2 sensor employs an infrared emitter and depth sensor to capture 3D data, which is processed through the Kinect SDK's advanced skeletal tracking algorithms. The SDK utilizes machine learning models to map raw depth data onto a predefined human skeletal structure, identifying and tracking 25 key joints in real time. It integrates flawlessly with Microsoft Visual Studio C++, allowing developers to access, customize, and utilize the skeletal data for applications such as teleoperation systems to accurately interpret and replicate human movements [34, 36, 37].

One limitation of using a single Kinect V2 sensor is its susceptibility to occlusion. During complex gestures, fast arm movements, or when the operator's body partially blocks the sensor's line of sight, the skeletal tracking may temporarily fail, causing a brief loss of joint data. In addition, the Kinect V2 has inherent field-of-view (FOV) constraints; since its effective depth sensing angle is approximately 70 degrees horizontally and 60 degrees vertically, the operator must remain within a limited workspace to ensure stable tracking. Moreover, the system performance is sensitive to environmental lighting conditions. Dynamic illumination changes, low-light environments, or reflective surfaces can introduce depth measurement errors and degrade tracking accuracy.

To address these limitations, future versions of the system could incorporate multi-sensor fusion techniques by combining data from multiple Kinect sensors or integrating additional vision-based or inertial measurement units (IMUs). Such enhancements would provide a wider tracking range, reduce occlusion effects, and improve robustness under varying environmental conditions.

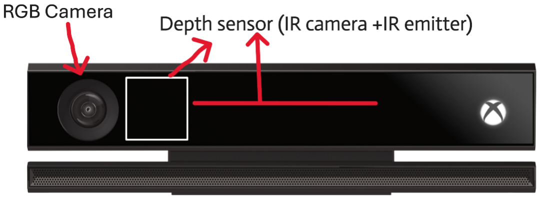


Figure 2. Kinect V2 camera

3.2 ROS/Rospy

In this paper, ROS (Robot Operating System) and its Python client library, Rospy, play a pivotal role in enabling the teleoperation of the Baxter robot. The Kinect V2 sensor captures 3D skeletal data of the human operator, which is processed using the Kinect SDK to generate joint angles. This data is transmitted to the robot through ROS's communication framework. Using Rospy, nodes are created to handle the flow of information: one node publishes the joint angle data in real time to a designated ROS topic, while another node running on the Baxter robot subscribes to this topic to receive and process the data. The subscriber node converts the joint angles into robot-specific commands, allowing the Baxter robot to accurately mimic the operator's movements. This integration of ROS and Rospy ensures flawless communication, modularity, and real-time synchronization between the human motion input and the robot's output, making it a robust solution for teleoperation [38, 39].

3.3 Baxter Robot

The Baxter robot, developed by Rethink Robotics, is a versatile dual-arm humanoid robot designed for research and industrial automation. Each of its two arms offers seven degrees of freedom, allowing for complex manipulation tasks. Equipped with cameras and force-sensing actuators, Baxter can adapt to changes in its environment, enhancing its ability to work safely alongside humans [40]. Its user-friendly interface

Figure 3. Baxter robot

enables programming through direct manipulation of its arms, eliminating the need for extensive coding. Baxter operates on the open-source Robot Operating System (ROS), facilitating integration with various software and hardware components [21]. As shown in Figure 3, the Baxter robot arm's seven degrees of freedom can be described as: shoulder joint: sj0, sj1, sj2; elbow joint: ej0, ej1; wrist joint: wj0, wj1, wj2. Each joint can be driven individually while being controlled in torque and in position mode to provide an ability for complex manipulation tasks as previously mentioned.

3.4 Network System

In the experimental setup, a router facilitates communication between two computers: one controlling the Baxter robot via a wired LAN connection, and the other connected to a Kinect camera through Wi-Fi. This configuration enables the transmission of data between the Kinect and the Baxter robot's control system. Implementing the User Datagram Protocol (UDP) for this communication ensures low-latency data transfer, which is crucial for real-time teleoperation tasks [41].

4 METHODOLOGY

The methodology of the teleoperation system developed in this research was structured upon Li et al.'s research, were their project aims, and structure aligns with the aims and structure of this project. Their work demonstrated the feasibility of using Kinect V2 sensor for real-time motion capture in human-robot teleoperation systems [33]. In the next sections of the methodology, the mathematical calculations and further explanation of the model used are viewed.

4.1 Human Arm Kinematics and Motion Mapping

The human arm can be structurally simplified as a rigid body system consisting of three primary joints: the shoulder, elbow, and wrist. This simplification of the human arm aids in translating the human arm motion into a mechanical motion which would develop the ability for the robot to understand and mimic the operator's motion in real time. In relation to the kinematic properties of a human arm, the arm operates with seven degrees of freedom (DOF): three degrees of freedom at the shoulder joint, two at the elbow joint, and two at the wrist joint. This would allow for a wide range of motion and flexibility [33, 42].

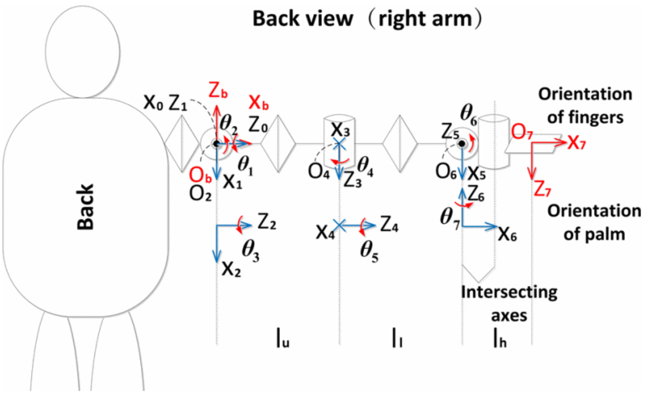


Figure 4. D-H model schematic [42]

The Denavit-Hartenberg (D-H) model, introduced by Denavit and Hartenberg in 1955 to simplify the mathematical representation of kinematic chains, simplifies the computation of kinematic relationships between joints and links of a robot by representing each joint and link using four parameters (d, β, a, α) [42, 43]. In this study, the D-H model serves as the basis for mapping human arm movements, captured using motion capture technology, to the 7-DOF robotic arm of the Baxter robot. This approach leverages the D-H model's ability to replicate human-like kinematic behaviour, ensuring precise and intuitive teleoperation. Figure 4 shows the D-H model of a human arm along with a table of the parameters of the D-H model [42].

The D-H model in Figure 4 shows the point O_b as the origin of the shoulder joint and the origin where the co-ordinate system to be established from for the arm. The Z_b axis of the origin O_b is in the upward direction; X_b axis is in the horizontal right direction; Y_b axis is in the horizontal facing direction in reference to the origin point. Using the same procedure, the co-ordinate system was established for the elbow and wrist joints with l_u, l_l, and l_h marking the length of the upper arm, forearm, and the hand, respectively [42]. D-H parameters are shown in Table 1.

$i(T_i^{i-1})$	θ_i	d_i	a_i	α_i
0	$-90°$	0	0	$-90°$
1	$\theta_1(90°)$	0	0	$90°$
2	$\theta_2(0°)$	0	0	$-90°$
3	$\theta_3(90°)$	l_u	0	$90°$
4	$\theta_4(0°)$	0	0	$-90°$
5	$\theta_5(-90°)$	l_l	0	$90°$
6	$\theta_6(90°)$	0	0	$-90°$
7	$\theta_7(0°)$	0	l_h	$180°$

Table 1. D-H model parameters [42]

To accurately describe and analyse the position of the human arm during motion, the space vector method was utilized to calculate joint angles. Avoiding complicated geometric calculations, this method simplifies the process by focusing on the spatial relationships between joints. Using the Cartesian coordinates of joint positions captured by the Kinect sensor, the distance between two points can be calculated. In two-dimensional space, the distance (d_1) is found using the formula:

$$d_1 = \sqrt{(x_b - x_a)^2 + (y_b - y_a)^2}. \tag{1}$$

For three-dimensional space, the distance (d_2) includes the Z-axis:

$$d_2 = \sqrt{(x_b - x_a)^2 + (y_b - y_a)^2 + (z_b - z_a)^2}. \tag{2}$$

These distances describe the length of the vectors connecting the joint points (x_a, y_a, z_a) and (x_b, y_b, z_b) in a Kinect co-ordinate system. To calculate the joint angles, the vector angle formula, Equation (3), was utilized where the vector angle can be calculated using the skeleton joint data obtained by Kinect [33].

$$\cos\left\langle \overrightarrow{a}, \overrightarrow{b} \right\rangle = \frac{\overrightarrow{a} \bullet \overrightarrow{b}}{|\overrightarrow{a}| \bullet \left|\overrightarrow{b}\right|}. \tag{3}$$

Using the vector angle formula, $\overrightarrow{a}$ and $\overrightarrow{b}$ represent the vectors formed by the joint coordinates, and the dot product gives the cosine of the angle between them. This method is efficient and accurate, as it reduces complex calculations to simple vector operations. The joint angles such as shoulder pitch and elbow roll are determined using this approach, allowing the Baxter robot to accurately replicate human movements [33].

4.2 Mathematical Framework for Motion Analysis

After placing the groundwork for the general calculation method, the Kinect camera identifies the skeletal system of a human arm and translate it into a three-dimensional cartesian co-ordinates of its 7-DOF [33]. The geometry model of the human arm is shown in Figure 5.

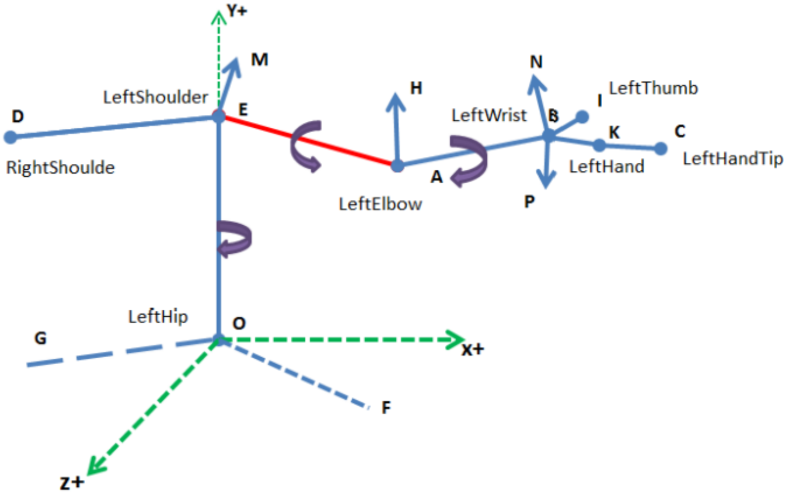


Figure 5. Geometry model of human arm [33]

Figure 5 identifies the shoulder pitch angle $\angle OEA$ and the elbow pitch angle $\angle EAB$. The co-ordinates for vectors $\overrightarrow{OE}$ and $\overrightarrow{EA}$ are extracted from the Kinect sensor readings, then Equations (1) and (2) are applied to calculate the magnitude of the vectors. After calculating the vector magnitude, the values of the vectors were utilized to calculate shoulder pitch angle by applying Equation (3) and using the inverse trigonometry. This process is repeated to determine the elbow pitch angle $\angle EAB$.

The shoulder yaw angle, which represents the angle between planes OEA and OED, is determined by calculating the normal vector between the two planes. The normal vector of a plane is calculated using the cross-product of vectors within the plane:

$$\overrightarrow{n} = \overrightarrow{u} \times \overrightarrow{v}. \tag{4}$$

The shoulder roll and elbow roll angles are calculated using plane normal vectors. For instance, the vector $\overrightarrow{EM}$ is derived from the cross-product of vectors $\overrightarrow{ED}$ and $\overrightarrow{EA}$, giving the normal vector for the plane defined by these vectors. Similarly, normal vectors such as $\overrightarrow{BP}$, $\overrightarrow{AH}$, and $\overrightarrow{BN}$ are computed for planes IKC, EAB, and ABI respectively. Translating vector $\overrightarrow{BN}$ along vector $\overrightarrow{AB}$ allows for the calculation of the angle between $\overrightarrow{AH}$ and $\overrightarrow{BN}$, yielding the elbow roll angle. The shoulder roll angle is similarly determined as the angle between vectors $\overrightarrow{AH}$ and $\overrightarrow{EM}$. The hand yaw angle is defined as the angle between the lower arm and the hand, which could be calculated by finding the angle between vector $\overrightarrow{AB}$ and the plane IKC.

The wrist yaw angle can be defined as the angle between vectors $\overrightarrow{X}_5$ and $\overrightarrow{Y}_7$. Vector $\overrightarrow{Y}_7$ is obtained through the cross-product of vectors $\overrightarrow{X}_7$ and $\overrightarrow{Z}_7$ and where $\overrightarrow{X}_7$ and $\overrightarrow{Z}_7$ were derived using vector normalization and cross-product:

$$\overrightarrow{X_7} = \frac{\overrightarrow{BK}}{\left|\overrightarrow{BK}\right|}, \quad \overrightarrow{Z_7} = \frac{\overrightarrow{BK} \times \overrightarrow{BI}}{\left|\overrightarrow{BK} \times \overrightarrow{BI}\right|}. \tag{5}$$

The vector $\overrightarrow{X_5}$ lies the plane BIC and is perpendicular to $\overrightarrow{AB}$. Therefore, the following equations are established:

$$\overrightarrow{X_5} = k_1 \bullet \overrightarrow{BI} + k_2 \bullet \overrightarrow{BK} \tag{6}$$

$$\left(k_1 \bullet \overrightarrow{BI} + k_2 \bullet \overrightarrow{BK}\right) \bullet \overrightarrow{AB} = 0 \tag{7}$$

$$\left|k_1 \bullet \overrightarrow{BI} + k_2 \bullet \overrightarrow{BK}\right| = 1 \tag{8}$$

After all the seven joints' angles are calculated, they are defined as q_{d1}, q_{d2}, q_{d3}, q_{d4}, q_{d5}, q_{d6}, q_{d7}, where they are identified as follows:

$$\begin{aligned}
&q_{d1} = \left\langle \overrightarrow{ED}, \overrightarrow{EA} \right\rangle, && q_{d2} = \left\langle \overrightarrow{EO}, \overrightarrow{EA} \right\rangle, \\
&q_{d3} = \left\langle \overrightarrow{EM}, \overrightarrow{AH} \right\rangle, && q_{d4} = \left\langle \overrightarrow{AE}, \overrightarrow{AB} \right\rangle, \\
&q_{d5} = \left\langle \overrightarrow{AH}, \overrightarrow{BN} \right\rangle, && q_{d6} = \frac{\pi}{2} - \left\langle \overrightarrow{BA}, \overrightarrow{BP} \right\rangle \\
&q_{d7} = \left\langle \overrightarrow{X_7}, \overrightarrow{Y_7} \right\rangle.
\end{aligned} \tag{9}$$

where each bracket is defined as the angle between each vector, for example, q_{d1} is the angle between vectors $\overrightarrow{ED}$ and $\overrightarrow{EA}$.

To translate the calculated human joint angles to the Baxter robot's coordinate system, the transformation T and offset matrices B are used to transfer the human joint angles to the Baxter's joint angles. The equation is expressed as follows:

$$\theta_{Baxter} = T \bullet \theta_{Kinect} + B. \tag{10}$$

4.3 Function Approximation Technique Based Control

The function approximation technique-based control (FAT) is a method that employs mathematical models to estimate the systems dynamics with unknown or complex behaviours [44]. A FAT-based controller is applied to achieve the subsequent control of the joint space trajectory. The output signal of the system is forced to trace the expected input signal. Hence, design of the controller should allow the system to be fast, accurate, and stable. The reference signal is represented by the angle vector $q_d \in \mathbb{R}^7$, which is provided by the Kinect, while the referenced angle vector $q \in \mathbb{R}^7$ returned by the robot is the actual angles. This means that the q_d is the target position for the robot, based on the human operator's movement, while the q is the current position of the robot. The dynamic behaviour of the manipulator can be mathematically described as:

$$M(q)\ddot{q} + C(q,\dot{q})\,\dot{q} + G(q) + \tau_{ext} = \tau, \tag{11}$$

where $M(q)$ is the inertia matrix, $C(q,\dot{q})$ is the Coriolis matrix, $G(q)$ is the gravity term, and τ_{ext} is the external torque applied on the joints. Then by deriving the manipulator's dynamic equation while incorporating s, v, and the robot's physical properties; the following equation is given:

$$M(q)\dot{s} + C(q,\dot{q})\,s + G(q) + M(q)\dot{v} + C(q,\dot{q})\,v = \tau - \tau_{ext}, \tag{12}$$

where s and v are the sliding surface and velocity term respectively, and they are defined in the following equations:

$$s = \dot{e}_q - \Lambda e_q, \tag{13}$$

$$v = \dot{q}_d - \Lambda e_q, \tag{14}$$

where $\dot{e}_q$is the rate of change of the error vector, Λ is the diagonal gain matrix ($\Lambda = diag(\lambda_1, \lambda_2, \ldots, \lambda_n)$), $\dot{q}_d$ is the rate of change of reference angles (q_d), Λe_q is the correction term to account for proportional control, and e_q is the error vector which is defined as:

$$e_q = q - q_d, \quad e_q \in \mathbb{R}^7, \tag{15}$$

where q is the actual joint angle vector of the robot, $q_d$is the reference joint angle vector of the motion capture system. After deriving Equation (12) to reach Equation (16), the joint torque (τ) in joint space can be calculated by:

$$\tau = \widehat{G} + \widehat{M}\dot{v} + \widehat{C}v + \tau_{ext} - Ks, \tag{16}$$

where $\widehat{G}$, $\widehat{M}$, and $\widehat{C}$ are the estimated gravity, inertia, and Coriolis matrices respectively, and K is the positive gain matrix. Then, the closed loop system can be written as:

$$M(q)\dot{s} + C(q,\dot{q})\,s + Ks = -\left(M - \widehat{M}\right)\dot{v} - \left(C - \widehat{C}\right)v - \left(G - \widehat{G}\right). \tag{17}$$

To approximate system dynamics, the function approximation technique is employed:

$$M(q) = W_M^T Z_M(q), \tag{18}$$

$$C\left(q, \dot{q}\right) = W_C^T Z_C\left(q, \dot{q}\right), \tag{19}$$

$$G(q) = W_G^T Z_G(q), \tag{20}$$

where the W_M, W_C, and W_G are the weight matrices; $Z_M(\theta), Z_C(\theta, \dot{\theta})$, and $Z_G(\theta)$ are the basis function matrices. The estimates of $M(q)$, $C\left(q, \dot{q}\right)$, and $G(q)$ can be written as:

$$\widehat{M}(q) = \widehat{W}_M^T Z_M(q), \tag{21}$$

$$\widehat{C}\left(q, \dot{q}\right) = \widehat{W}_C^T Z_C\left(q, \dot{q}\right), \tag{22}$$

$$\widehat{G}(q) = \widehat{W}_G^T Z_G(q). \tag{23}$$

Then by substituting the values in the close loop system, the closed loop system becomes:

$$M(q)\dot{s} + C\left(q, \dot{q}\right)s + Ks = -\widehat{W}_M^T Z_M \dot{v} - \widehat{W}_C^T Z_C v - \widehat{W}_G^T Z_G, \tag{24}$$

where:

$$\widetilde{W}_M^T = W_M^T - \widehat{W}_M^T, \tag{25}$$

$$\widetilde{W}_C^T = W_C^T - \widehat{W}_C^T, \tag{26}$$

$$\widetilde{W}_G^T = W_G^T - \widehat{W}_G^T. \tag{27}$$

By completing the design of the adaptive controller, now the next step is to verify the system's stability by applying the following Lyapunov function:

$$V = \frac{1}{2}s^T M s + \frac{1}{2}tr\left(\widetilde{W}_M^T Q_M \widetilde{W}_M + \widetilde{W}_C^T Q_C \widetilde{W}_C + \widetilde{W}_G^T Q_G \widetilde{W}_G\right), \tag{28}$$

where the Q_M, Q_C, and Q_G are the positive definite weight matrices. The derivative of the Lyapunov function $\dot{V}$ is derived to verify stability, the equation is as follows:

$$\begin{aligned}\dot{V} = &- s^T K s - \mathrm{tr}\left[\widehat{W}_M^T\left(Z_M \dot{v}_s^T + Q_M \dot{\widehat{W}}_M\right)\right] + \mathrm{tr}\left[\widehat{W}_C^T\left(Z_C \dot{v}_s^T + Q_C \dot{\widehat{W}}_C\right)\right] \\ &+ \mathrm{tr}\left[\widehat{W}_G^T\left(Z_G \dot{v}_s^T + Q_G \dot{\widehat{W}}_G\right)\right].\end{aligned} \tag{29}$$

By selecting the adaptive update laws for the weight matrices as:

$$\dot{\widehat{W}}_M = -Q_M^{-1} Z_M \dot{v}_s^T, \tag{30}$$

$$\dot{\widehat{W}}_C = -Q_C^{-1} Z_C v_s^T, \tag{31}$$

$$\dot{\widehat{W}}_G = -Q_G^{-1} Z_G s^T. \tag{32}$$

Substituting the above into Equation (29) gives:

$$\dot{V} = -s^T K s \tag{33}$$

Since $\dot{V}$ is negative definite, this guarantees the system's stability according to Lyapunov theory.

5 EXPERIMENTAL STUDIES

An experiment was conducted to evaluate a teleoperation system using motion capture with the Baxter robot. The setup included a user computer connected to a Kinect camera to capture skeletal joint data and a Linux/ROS computer running Baxter, which received and translated the Kinect data in real time to enable imitation of the user's movements. Both systems were networked via a router. During the test, the operator moved their arms while Baxter mimicked the motion, and images (Figure 6) were captured for visual validation. The system's performance, particularly its responsiveness and stability, was assessed using recorded motion data, with key plots shown in Figures 7 and 8.

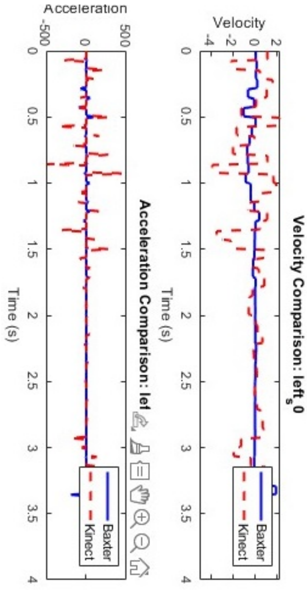
The experimental figures present data sets labeled by arm (left/right) and joint (shoulder – s, elbow – e, wrist – w), comparing real-time position, velocity, and acceleration between the Kinect and Baxter robot. Visual validation of the robot's imitation is supported by the position graphs, confirming accurate motion tracking. The FAT-based controller effectively reduced abrupt joint movements and spikes, indicating its robustness in adapting to dynamic conditions and mitigating inertia and disturbances. Some graphs reveal minor delays in the robot's response, evidenced by a temporal offset between the Kinect and Baxter curves. This lag is attributed to network discrepancies, as the user computer connected via Wi-Fi and the Baxter system via LAN. Though insignificant in real-time observation, these delays are evident in the recorded data. Figure 9 shows the compensation torque $\tau_c = \widehat{G} + \widehat{M}\dot{v} + \widehat{C}v$ generated by the FAT controller for the left arm, which as viewed shows the effect of delay and jittering that occurs during the robot's operation time in relevance to the robot's joint points. The elbow joint is the most effected as shown in the graph due to the presence of the elbow movement in each arm movement during the experiment unlike other points on the robot arm.

The experimental results demonstrate that the Baxter robot effectively imitates user movements in real time, with the robot's joint behavior closely mirroring that of

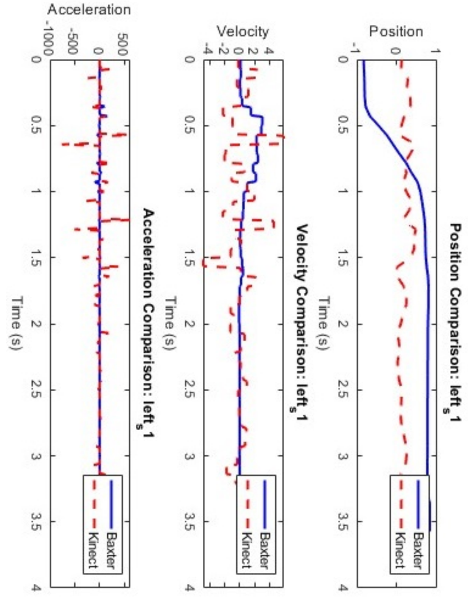
Figure 6. Posture comparison of human and robot

the skeletal system captured by the Kinect. Both visual evidence and graphed data confirm accurate mapping of human motion. While the system performs well, minor issues such as data transmission delays, jittering, and mismatched data sampling rates between the Kinect and the robot remain. These discrepancies are primarily due to network latency and differences in processing capabilities, with Wi-Fi-induced delays being a key contributor. A full LAN setup could mitigate this. The findings align with those of Li et al. [33], where actual and reference joint angles showed high correspondence, further validating the success of the teleoperation system based on motion capture.

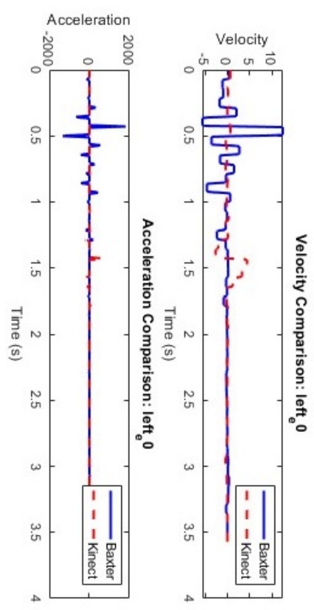
To complement the qualitative visual comparisons, quantitative performance indicators were estimated based on the recorded experimental data. For the left elbow joint (e1), the average joint angle tracking error was approximately 0.07 radians (4 degrees), while the maximum peak deviation reached about 0.15 radians. The system latency was visually estimated to be within 70–100 ms by analyzing the temporal offsets between Kinect reference curves and Baxter response curves. In ad-



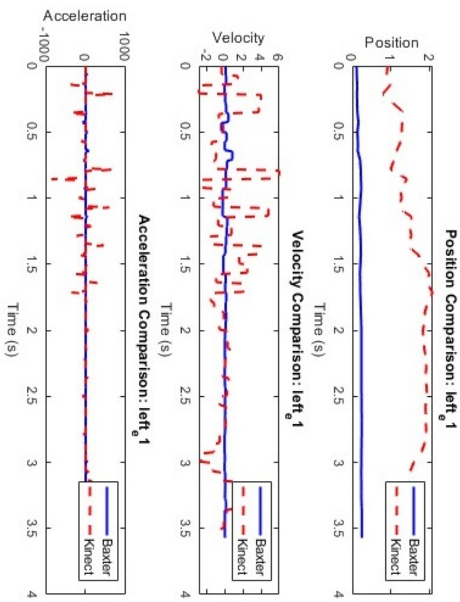
a) Left shoulder joint s0



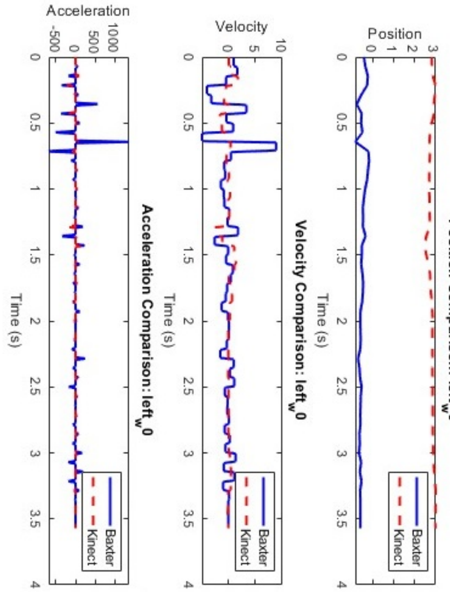
b) Left shoulder joint s1



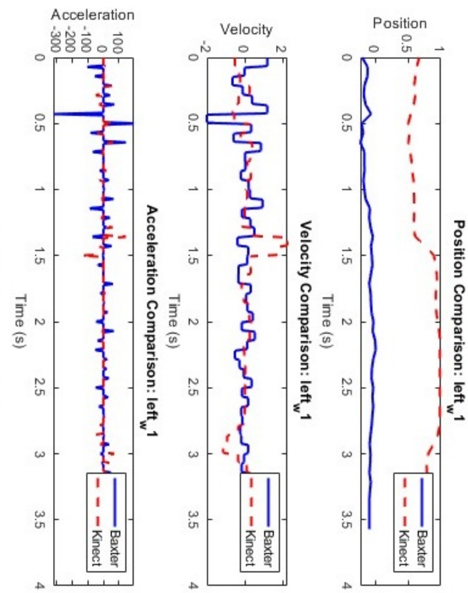
c) Left elbow joint e0



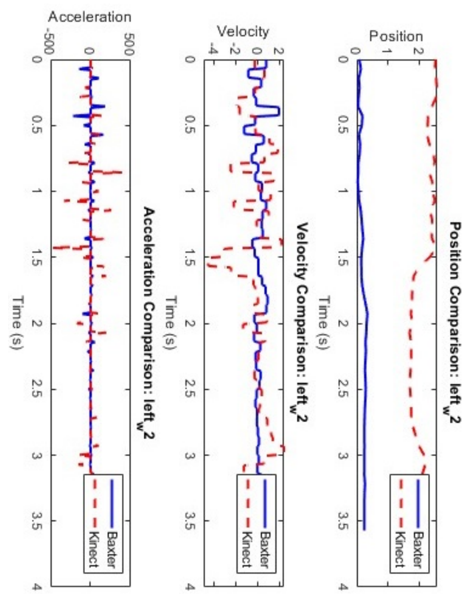
d) Left elbow joint e1



e) Left wrist joint w0

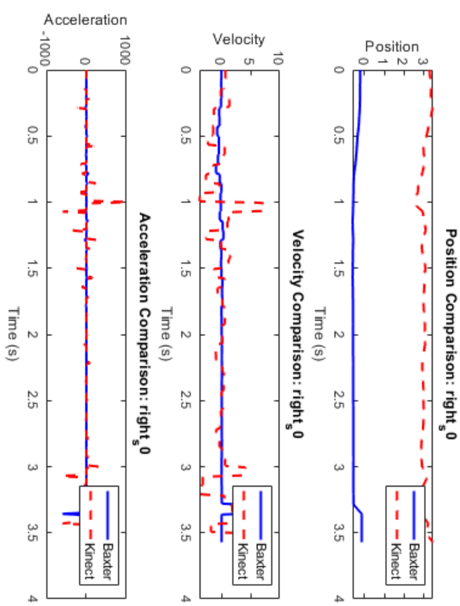


f) Left wrist joint w1

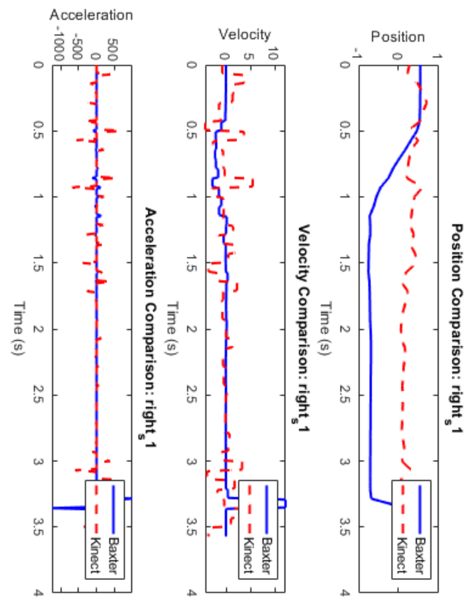


g) Left wrist joint w2

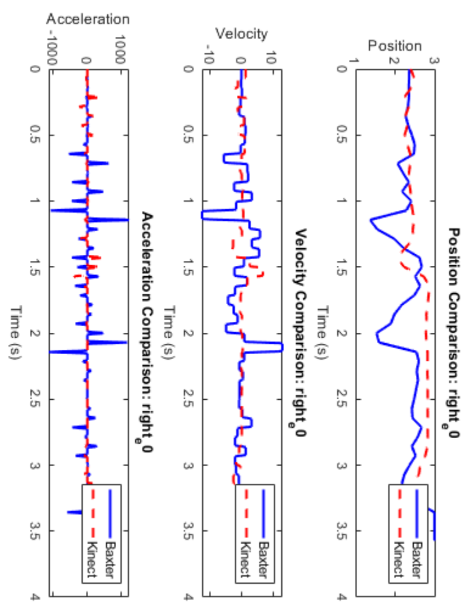
Figure 7. Left arm joint motion data



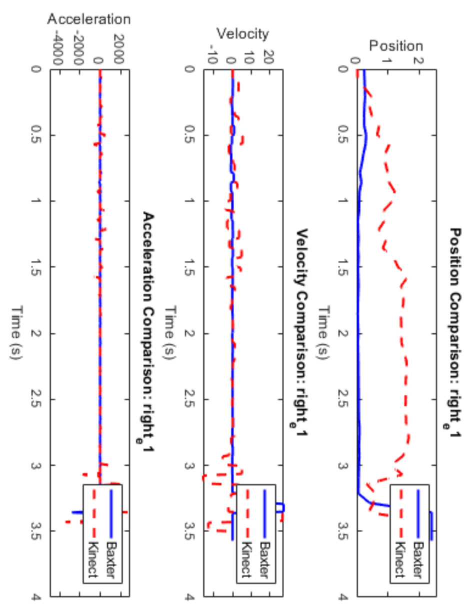
a) Right shoulder joint s0



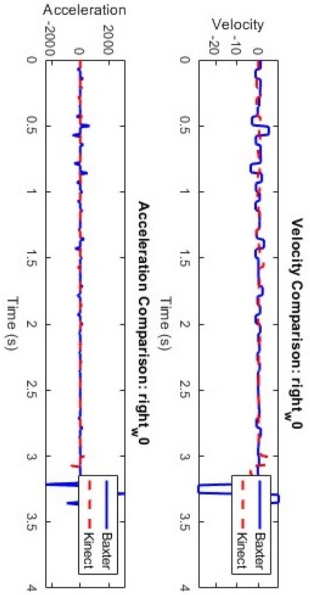
b) Right shoulder joint s1



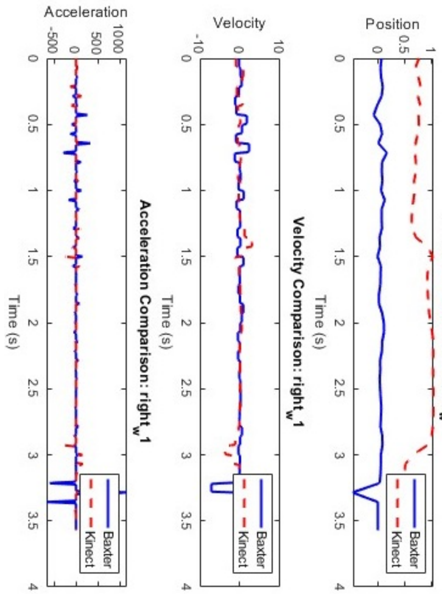
c) Right elbow joint e0



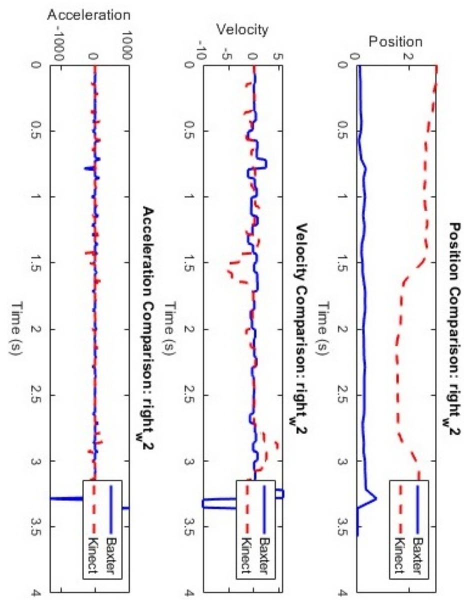
d) Right elbow joint e1



e) Right wrist joint w0



f) Right wrist joint w1



g) Right wrist joint w2

Figure 8. Right arm joint motion data

dition, the acceleration jitter observed at the wrist joints reached up to $\pm 1.5\,\mathrm{rad/s^2}$, which primarily resulted from network variability, data sampling differences, and signal noise.

These quantitative indicators demonstrate that the proposed teleoperation framework maintains acceptable tracking accuracy and responsiveness despite network-induced latency and jitter. They also confirm the effectiveness of the FAT-based adaptive control strategy in stabilizing joint trajectories and compensating for disturbances in real time.

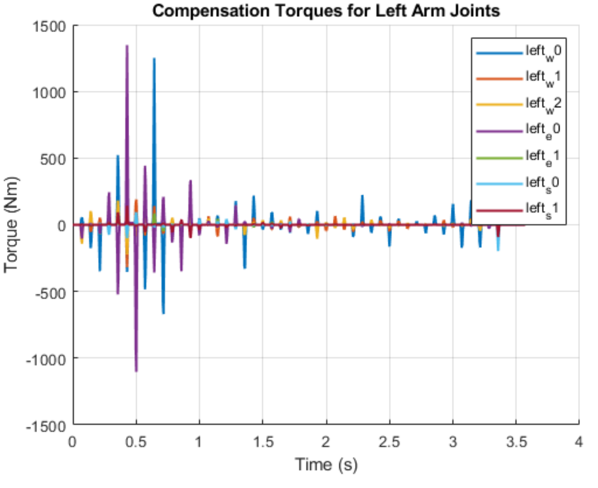


Figure 9. Compensation torque for left arm joints

6 CONCLUSION

This paper presented the successful development of a teleoperation system for the Baxter robot using motion capture methodologies. The system's effectiveness was confirmed through real-time visual validation and quantitative analysis of joint movements. Building on the mathematical model and FAT-based controller design, the system demonstrated the ability to map human motion to robotic response reliably using a Kinect V2 sensor. The setup's simplicity and adaptability suggest potential for wider applications, particularly in environments where direct human operation poses health risks.

The system has not yet been tested across a wide range of user body types or movement patterns. Future work will explore adaptive mapping strategies to accommodate variations in operator kinematics, enabling more generalized applicability across diverse users. We would also enhance motion capture accuracy by integrating depth sensors with inertial measurement units. Reducing network latency would improve real-time responsiveness, while machine learning could enable more nuanced motion replication. Expanding into fields requiring high precision – such as robotic surgery or hazardous material handling – would test the system's robustness. Adding tactile feedback and computer vision could further improve user control and environmental adaptability. Real-world testing, better user interfaces, and semi-autonomous capabilities would position the system for broader deployment across healthcare, manufacturing, and research.

Availability Statement

The datasets and coding scripts used and analyzed during the current study are available from the corresponding author on reasonable request.

REFERENCES

[1] You, K.—Zhou, C.—Ding, L.: Deep Learning Technology for Construction Machinery and Robotics. Automation in Construction, Vol. 150, 2023, Art. No. 104852, doi: 10.1016/j.autcon.2023.104852.

[2] Škulj, G.—Vrabič, R.—Podržaj, P.: A Wearable IMU System for Flexible Teleoperation of a Collaborative Industrial Robot. Sensors, Vol. 21, 2021, No. 17, Art. No. 5871, doi: 10.3390/s21175871.

[3] Mizera, C.—Delrieu, T.—Weistroffer, V.—Andriot, C.—Decatoire, A.—Gazeau, J.: Evaluation of Hand-Tracking Systems in Teleoperation and Virtual Dexterous Manipulation. IEEE Sensors Journal, Vol. 20, 2020, No. 3, pp. 1642–1655, doi: 10.1109/JSEN.2019.2947612.

[4] Goertz, R. C.: Master–Slave Manipulator. Technical Report. U.S. Department of Energy Office of Scientific and Technical Information, 1949, doi: 10.2172/1054625.

[5] Li, Z.—Xia, Y.—Su, C. Y.: Intelligent Networked Teleoperation Control. Springer, 2015, doi: 10.1007/3-540-44589-7.

[6] Sheridan, T. B.: Space Teleoperation Through Time Delay: Review and Prognosis. IEEE Transactions on Robotics and Automation, Vol. 9, 1993, No. 5, pp. 592–606, doi: 10.1109/70.258052.

[7] Brewster, S.—Murray-Smith, R.: Haptic Human-Computer Interaction. Springer, 2001, doi: 10.1007/3-540-44589-7.

[8] Schenker, P. S.—Bejczy, A. K.—Kim, W. S.: Advanced Teleoperation: Technology Innovations and Applications. Technology 2003: The Fourth National Technology Transfer Conference and Exposition, Vol. 2, 1994.

[9] Yoerger, D.—Slotine, J.: Supervisory Control Architecture for Underwater Teleoperation. 1987 IEEE International Conference on Robotics and Automation, Vol. 4, 1987, pp. 2068–2073, doi: 10.1109/ROBOT.1987.1087890.

[10] Lichiardopol, S.: A Survey on Teleoperation. Ph.D. Thesis. Technische Universiteit Eindhoven, 2007, `https://pure.tue.nl/ws/portalfiles/portal/4419568/656592.pdf`.

[11] Song, D.—Goldberg, K.—Chong, N. Y.: Networked Robots. In: Siciliano, B., Khatib, O. (Eds.): Springer Handbook of Robotics. Springer, Cham, 2016, pp. 1109–1134, doi: 10.1007/978-3-319-32552-1_44.

[12] Tang, H.—Cao, X.—Song, A.—Guo, Y.—Bao, J.: Human-Robot Collaborative Teleoperation System for Semi-Autonomous Reconnaissance Robot. 2009 IEEE International Conference on Mechatronics and Automation, 2009, pp. 1934–1939, doi: 10.1109/ICMA.2009.5246589.

[13] Antona, M.—Stephanidis, C.: Universal Access in Human-Computer Interaction: Access to Learning, Health and Well-Being (UAHCI 2015). Springer, 2015, doi: 10.1007/978-3-319-20684-4.

[14] PhaseSpace Motion Capture – Solutions. PhaseSpace, 2026, https://www.phasespace.com/solutions.

[15] Menache, A.: Understanding Motion Capture for Computer Animation. 2nd Edition. Morgan Kaufmann, 2011, doi: 10.1016/C2009-0-62989-5.

[16] Shotton, J.—Sharp, T.—Kipman, A.—Fitzgibbon, A.—Finocchio, M.—Blake, A.—Cook, M.—Moore, R.: Real-Time Human Pose Recognition in Parts from Single Depth Images. Communications of the ACM, Vol. 56, 2013, No. 1, pp. 116–124, doi: 10.1145/2398356.2398381.

[17] Sabatini, A. M.: Estimating Three-Dimensional Orientation of Human Body Parts by Inertial/Magnetic Sensing. Sensors, Vol. 11, 2011, No. 2, pp. 1489–1525, doi: 10.3390/s110201489.

[18] Seel, T.—Raisch, J.—Schauer, T.: IMU-Based Joint Angle Measurement for Gait Analysis. Sensors, Vol. 14, 2014, No. 4, pp. 6891–6909, doi: 10.3390/s140406891.

[19] Zihajehzadeh, S.—Park, E. J.: A Novel Biomechanical Model-Aided IMU/UWB Fusion for Magnetometer-Free Lower Body Motion Capture. IEEE Transactions on Systems, Man, and Cybernetics: Systems, Vol. 47, 2017, No. 6, pp. 927–938, doi: 10.1109/TSMC.2016.2521823.

[20] Roetenberg, D.—Luinge, H.—Slycke, P.: Xsens MVN: Full 6DOF Human Motion Tracking Using Miniature Inertial Sensors. Technical Report No. 3, Xsens Motion Technologies B.V., 2009, https://www.xsens.com/hubfs/Downloads/Whitepapers/MVN_whitepaper.pdf.

[21] Williams II, R. L.: Baxter Humanoid Robot Kinematics. 2017, https://www.ohio.edu/mechanical-faculty/williams/html/pdf/BaxterKinematics.pdf.

[22] Huang, Y.—Zhang, X.—Chen, X.—Ota, J.: Vision-Guided Peg-in-Hole Assembly by Baxter Robot. Advances in Mechanical Engineering, Vol. 9, 2017, No. 12, pp. 1–9, doi: 10.1177/1687814017748078.

[23] Fitter, N. T.—Mohan, M.—Kuchenbecker, K. J.—Johnson, M. J.: Exercising with Baxter: Preliminary Support for Assistive Social-Physical Human-Robot Interaction. Journal of NeuroEngineering and Rehabilitation, Vol. 17, 2020, No. 1, Art. No. 19, doi: 10.1186/s12984-020-0642-5.

[24] Ju, Z.—Yang, C.—Ma, H.: Kinematics Modeling and Experimental Verification of Baxter Robot. Proceedings of the 33rd Chinese Control Conference, 2014, pp. 8518–8523, doi: 10.1109/ChiCC.2014.6896430.

[25] Rastegarpanah, A.—Marturi, N.—Stolkin, R.: Autonomous Vision-Guided Bi-Manual Grasping and Manipulation. 2017 IEEE Workshop on Advanced Robotics and Its Social Impacts (ARSO), 2017, pp. 1–7, doi: 10.1109/ARSO.2017.8025192.

[26] Arduengo, M.—Arduengo, A.—Colomé, A.—Lobo-Prat, J.—Torras, C.: Human to Robot Whole-Body Motion Transfer. 2020 IEEE-RAS 20th International Conference on Humanoid Robots (Humanoids), 2021, pp. 299–305, doi: 10.1109/HUMANOIDS47582.2021.9555769.

[27] Michel, Y.—Abdelhalem, Y.—Cheng, G.: Passivity-Based Teleoperation with Variable Rotational Impedance Control. IEEE Robotics and Automation Letters, Vol. 9, 2024, No. 12, pp. 11658–11665, doi: 10.1109/LRA.2024.3490260.

[28] Chen, Y. H.—Yang, W. T.—Chen, B. H.—Lin, P. C.: Manipulator Trajectory Optimization Using Reinforcement Learning on a Reduced-Order Dynamic Model with Deep Neural Network Compensation. Machines, Vol. 11, 2023, No. 3, Art. No. 350, doi: 10.3390/machines11030350.

[29] Zhang, X.—Wang, Y.—Li, C.—Fahmy, A.—Sienz, J.: Innovative Multi-Dimensional Learning Algorithm and Experiment Design for Human-Robot Cooperation. Applied Mathematical Modelling, Vol. 127, 2024, pp. 730–751, doi: 10.1016/j.apm.2024.01.001.

[30] Richter, F.—Zhang, Y.—Zhi, Y.—Orosco, R. K.—Yip, M. C.: Augmented Reality Predictive Displays to Help Mitigate the Effects of Delayed Telesurgery. 2019 International Conference on Robotics and Automation (ICRA), 2019, pp. 444–450, doi: 10.1109/ICRA.2019.8794051.

[31] Ju, Z.—Yang, C.—Li, Z.—Cheng, L.—Ma, H.: Teleoperation of Humanoid Baxter Robot Using Haptic Feedback. 2014 IEEE International Conference on Multisensor Fusion and Information Integration for Intelligent Systems (MFI), 2014, pp. 1–6, doi: 10.1109/MFI.2014.6997721.

[32] Brygo, A.—Sarakoglou, I.—Garcia-Hernandez, N.—Tsagarakis, N.: Humanoid Robot Teleoperation with Vibrotactile Based Balancing Feedback. In: Auvray, M., Duriez, C. (Eds.): Haptics: Neuroscience, Devices, Modeling, and Applications (EuroHaptics 2014). Springer, Berlin, Heidelberg, Lecture Notes in Computer Science, Vol. 8619, 2014, pp. 266–275, doi: 10.1007/978-3-662-44196-1_33.

[33] Li, C.—Yang, C.—Wan, J.—Annamalai, A. S. K.—Cangelosi, A.: Teleoperation Control of Baxter Robot Using Kalman Filter-Based Sensor Fusion. Systems Science and Control Engineering, Vol. 5, 2017, No. 1, pp. 156–167, doi: 10.1080/21642583.2017.1300109.

[34] Lachat, E.—Macher, H.—Landes, T.—Grussenmeyer, P.: Assessment and Calibration of an RGB-D Camera (Kinect V2 Sensor) Towards a Potential Use for Close-Range 3D Modeling. Remote Sensing, Vol. 7, 2015, No. 10, pp. 13070–13097, doi: 10.3390/rs71013070.

[35] Corti, A.—Giancola, S.—Mainetti, G.—Sala, R.: A Metrological Characterization of the Kinect V2 Time-of-Flight Camera. Robotics and Autonomous Systems, Vol. 75, 2016, pp. 584–594, doi: 10.1016/j.robot.2015.11.016.

[36] Yadav, S. K.—Tiwari, K.—Pandey, H. M.—Akbar, S. A.: Skeleton-Based Human Activity Recognition Using ConvLSTM and Guided Feature Learning. Soft Com-

puting, Vol. 26, 2022, No. 2, pp. 877–890, doi: 10.1007/s00500-021-06238-7.

[37] Download Kinect for Windows SDK 2.0 from Official Microsoft Download Center. Microsoft, 2014, `https://www.microsoft.com/en-us/download/details.aspx?id=44561`.

[38] ROS Introduction. ROS Wiki, 2018, `https://wiki.ros.org/ROS/Introduction`.

[39] Rospy – ROS Wiki. 2014, `https://wiki.ros.org/rospy`.

[40] Hu, L.—Zhai, D. H.—Yu, D.—Xia, Y.: A Hybrid Framework Based on Bio-Signal and Built-in Force Sensor for Human–Robot Active Co-Carrying. IEEE Transactions on Automation Science and Engineering, Vol. 22, 2025, pp. 3553–3566, doi: 10.1109/TASE.2024.3395921.

[41] Avalos, J.—Ramos, O. E.: Real-Time Teleoperation with the Baxter Robot and the Kinect Sensor. 2017 IEEE 3rd Colombian Conference on Automatic Control (CCAC), 2017, pp. 1–4, doi: 10.1109/CCAC.2017.8320346.

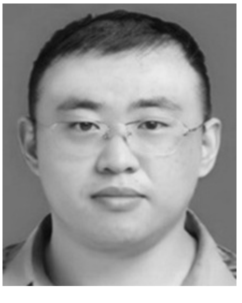
[42] Fang, C.—Ding, X.: A Set of Basic Movement Primitives for Anthropomorphic Arms. 2013 IEEE International Conference on Mechatronics and Automation, 2013, pp. 639–644, doi: 10.1109/ICMA.2013.6617991.

[43] Denavit, J.—Hartenberg, R. S.: A Kinematic Notation for Lower-Pair Mechanisms Based on Matrices. Journal of Applied Mechanics, Vol. 22, 1955, No. 2, pp. 215–221, doi: 10.1115/1.4011045.

[44] Li, C.—Yang, C.—Wan, J.—Annamalai, A.—Cangelosi, A.: Neural Learning and Kalman Filtering Enhanced Teaching by Demonstration for a Baxter Robot. 2017 23rd International Conference on Automation and Computing (ICAC), IEEE, 2017, pp. 1–6, doi: 10.23919/IConAC.2017.8081985.

Jiaying Huang is Lecturer at the Guangzhou Academy of Fine Arts and Council Member of the Guangdong Folk Artists Association in China. She was a visiting scholar at the University of Florence, Italy, in 2016. Her research interests include human–computer interaction, visual multimedia performance, and design innovation. She has published two EI-indexed papers and three CPCI-indexed papers, contributed to an academic book, and holds one invention patent. She has led research projects funded by the Guangdong Provincial Philosophy and Social Sciences Planning Program and the Department of Education of Guangdong Province, and has participated in several other provincial-level research projects. She has also received multiple design awards.

Youssef Essa received his Master of Engineering degree from the Swansea University. His research focuses on human–machine skill transfer, including teleoperation, robot teaching programming and skill generalization, and on enhancing humanoid skill expression and compliant control in robotic systems.



Chunxu Li received his Ph.D. degree from the Swansea University in 2019, and is currently Full Professor at the Faculty of Health Sciences, University of Health and Rehabilitation Science. He has authored more than 50 academic publications in leading journals such as AMM, JCDE, SR, IEEE Transactions on Circuits and Systems, IEEE Transactions on Industrial Informatics, IEEE/ASME Transactions on Mechatronics, and Information Sciences, accumulating over 1 200 citations. He has also delivered keynote speeches at IEEE international conferences (ICIICII 2022, ISRIMT 2021, HFR 2017), on remote human–machine collaboration, multimodal data fusion, and adaptive robotic control strategies. As PI/main co-I, he has secured a number of projects and grants.