IEEE Access
Multidisciplinary : Rapid Review : Open Access Journal

Received 26 January 2025, accepted 15 April 2025, date of publication 23 April 2025, date of current version 5 May 2025.

Digital Object Identifier 10.1109/ACCESS.2025.3563824

RESEARCH ARTICLE

Embedded System for Interactive Pneumatic Hand Rehabilitation: Real-Time Gaming Interface With Cognitive Stimulation for Motor Recovery

NARGES GHOBADI [1], (Graduate Student Member, IEEE), WITOLD KINSNER [2], (Life Senior Member, IEEE), TONY SZTURM [3], AND NARIMAN SEPEHRI [1], (Senior Member, IEEE)

[1]Department of Mechanical Engineering, Price Faculty of Engineering, University of Manitoba, Winnipeg, MB R3T 2N2, Canada
[2]Department of Electrical and Computer Engineering, Price Faculty of Engineering, University of Manitoba, Winnipeg, MB R3T 2N2, Canada
[3]College of Rehabilitation Sciences, Rady Faculty of Health Sciences, University of Manitoba, Winnipeg, MB R3T 2N2, Canada

Corresponding author: Nariman Sepehri (Nariman.Sepehri@umanitoba.ca)

This work was supported by the Natural Sciences and Engineering Research Council of Canada (NSERC) under Grant RGPIN-2024-05468.

This work involved human subjects or animals in its research. Approval of all ethical and experimental procedures and protocols was granted by the University of Manitoba Human Research Ethics Board under Application No. HS25163 and H2021:335.

ABSTRACT Restoring fine motor skills in individuals with upper extremity sensory-motor post-stroke impairments necessitates repetitive, task-specific exercises to promote functional recovery. Given the substantial time commitment required for therapy, rehabilitation tools must be not only effective but also engaging by adopting a game-based approach to mitigate the monotony of prolonged repetitive exercises. This paper presents a user-friendly finger-thumb mechanism designed to support the index and middle fingers as well as the thumb, specifically for patients with hand injuries. The device establishes a wireless connection to a gaming platform enabling patients to engage with computer games in real-time through purposeful thumb and finger movements. This connectivity is established through two Raspberry Pi boards utilizing a server-client network. Additionally, the device incorporates assistive or resistive forces during gameplay to adjust the level of assistance or challenge based on the individual's motor control proficiency. With a minimal data transfer delay of 10 ms, and a 50 ms delay for game event updates, patients can seamlessly participate in the gaming experience and modify events in real-time through the newly developed wearable device. In the experiments, assistive mode achieved a 100% success rate, while resistive mode dropped to 37-45%. Movement errors varied across modes, with assistive mode showing the lowest errors, indicating more accurate and consistent performance.

INDEX TERMS Real-time interfacing, edge computing, embedded system, rehabilitation, pneumatic actuator.

I. INTRODUCTION

Stroke often results in significant impairments in sensorimotor control of the hand and impacts an individual's independence and quality of life [1]. Robotic systems are increasingly popular in post-stroke rehabilitation because they provide the necessary intensity, motivation, and customization [2], [3]. A successful rehabilitation program should employ a holistic approach that targets motor and cognitive skills to enhance certain patients' less engaging and more challenging aspects of rehabilitation sessions. This can be achieved by connecting rehabilitation equipment to computer games, which boosts the efficacy of rehabilitation and makes it more enjoyable for patients [4], [5]. Integrating computer games into rehabilitation, however, is complex, and involves synchronization and comparison of data from both the game and the device and transmitting commands for device interaction [6]. This synchronization results in a game

The associate editor coordinating the review of this manuscript and approving it for publication was Ganesh Naik.

delay that hinders users from interacting with it effectively. Researchers have suggested different methods, such as using artificial neural networks (ANNs) to compensate for latency in gaming, which leads to higher player performance and a better game experience [7]. This highlights the importance of minimizing game latency to ensure a satisfactory gaming experience. The spectrum of these speed limits is defined by a 10 ms threshold, corresponding to the average human muscle response time [8].

In this regard, developing real-time serious gaming systems, such as the one discussed by Tannous et al. [9], using Kinect sensors for musculoskeletal rehabilitation, demonstrates the robustness and user-friendliness of such systems in motivating patients during their rehabilitation program. Incorporating location-based and proximity-based features in computer-augmented games has shown potential in enhancing the rehabilitation experience further. The concept of "Implicit Player Input," where players use sensors attached to their bodies, emphasizes body control in gaming. Cantu et al. [10] explored a cane-based 3D interface for rehabilitation games to demonstrate the potential of such technologies. However, limitations exist in capturing complete finger and thumb states, as noted by Ghosh and Saha [11] in their work on interactive game-based motor rehabilitation using hybrid sensor architecture. Vogiatzaki et al. [12] addressed stroke rehabilitation by proposing a game-based training system that allows physicians to supervise rehabilitation at home. This method has been shown to increase rehabilitation speed and improve the quality of life for patients, demonstrating the versatility and effectiveness of game-based systems in remote settings. The REACH robot by Yap et al. [13] shows the application of game-based interfaces in pediatric rehabilitation settings. Garcia-Hernandez et al. [14] developed a touch-based game system for training and assessing unilateral and bilateral reaching movements. Athanasiou et al. [15] discussed the development and optimization of a platform using game-based interfaces to promote neurorehabilitation in individuals with spinal cord injuries. The study highlights the broader applicability of such technologies beyond stroke rehabilitation. Homola et al. [16] developed a low-cost toolkit interfacing a LEGO Technic arm exoskeleton with serious gaming.

Integrating game-based rehabilitation interfaces also offers enhanced engagement and leverages the therapeutic potential of virtual reality (VR) and mixed reality (MR). Postolache et al. [17] presented a VR game for upper limb rehabilitation using a Leap Motion controller, effectively comparing training results in virtual and real environments. Krukowski and Vogiatzaki [18] highlighted the use of game-based rehabilitation systems that utilize natural human-machine interfaces, such as augmented and virtual reality headsets, to improve physical capabilities through games and competitions. This approach is particularly effective due to its engaging natural human senses and interaction in familiar environments, thus enhancing rehabilitation. Paraense et al. [19] worked on a game prototype for rehabilitation, utilizing VR technology, including modes that allow users to use any arm to hit objects and another mode that provides the illusion of movement in the arm affected by the stroke.

The application of Motor-Imagery based Brain-Computer Interfaces (MI-BCI) in game-based interfaces is also gaining attention. Ianoşi-Andreeva-Dimitrova and Mândru [20] discussed how game-based interfaces can make training for MI-BCI more entertaining and potentially allow for earlier implementation of medical rehabilitation techniques. Camargo-Vargas et al. [21] described the implementation of a brain-computer interface (BCI) prototype device with a virtual environment to aid in upper limb rehabilitation. Their findings suggest that their prototype has the potential for real-time applications and can significantly motivate and support individuals in their rehabilitation process. Gaafer et al. [22] discussed using immersive virtual reality games with brain-computer interfaces in neurorehabilitation.

Rehabilitation of hand functions, especially following conditions such as stroke, is a critical challenge in healthcare, requiring innovative solutions that go beyond traditional passive exercises. The motivation behind this work is to create a system that not only helps restore motor function but also actively involves the patient's cognitive processes, thereby promoting neuroplasticity and improving long-term outcomes. Additionally, enhancing user comfort and freedom of movement during rehabilitation sessions is a key driver, as these factors play a significant role in user compliance and the overall effectiveness of therapy. This paper aims to bridge the gap between clinical effectiveness and user-centered design by offering a portable, real-time, and engaging rehabilitation solution. Accordingly, this paper introduces a new wearable device for finger-thumb rehabilitation that targets both motor and cognitive functions. The device connects to a computer game via a server-client network using Raspberry Pi development boards, enabling real-time interaction without requiring wired connections between the device and the screen, thereby enhancing user comfort and mobility. The paper presents the hardware and software components of the system. Clinical trials were conducted with three participants (one healthy and two post-stroke) to validate the system's performance in real-time data transfer, ensuring a smooth gaming experience and demonstrating the potential for effective rehabilitation.

The innovation of this study lies in several key aspects:

- Real-time cognitive stimulation for motor recovery:

This paper presents a novel embedded system for finger rehabilitation that integrates real-time gaming as an interface. Unlike passive rehabilitation devices, the proposed system is capable of engaging participants involving decision-making and active interaction, focusing not only on motor recovery but also on stimulating cognitive function. This dual emphasis on cognitive engagement and motor rehabilitation distinguishes the proposed device from conventional systems that primarily focus on passive movement [23], [24].

- Wireless, user-friendly setup for enhanced comfort and mobility:

The setup is designed with participant comfort in mind, enabling them to move freely without concerns about wire disconnections compared to existing fixed hand exoskeletons [25]. A dual-board design facilitates real-time communication between the mechanism and the main processor without requiring a wired connection between the device and the screen. This wireless feature enhances user mobility and reduces fatigue during rehabilitation sessions, making it more practical and engaging for extended use.

- Compact, active mechanism for effective joint rehabilitation

The mechanism is carefully designed to minimize bulkiness compared to existing devices [26], ensuring that it remains easy to wear while effectively assisting in the motion of larger finger joints, particularly the metacarpophalangeal (MCP) joint. Despite its compact form, the device is equipped with an actuator capable of providing various rehabilitation modes, including both assistive and resistive. While assistive devices have been used considering their effectiveness in rehabilitation, taking advantage of resistive mode can also challenge the participants during therapy and further improve the motor recovery.

Table 1 presents a comparison of the proposed device with selected similar devices by other researchers. The table highlights the advantages of the proposed mechanism in relation to existing designs and reflects the authors' underlying motivation.

TABLE 1. Comparison between different hand rehabilitation devices.

Device	Dimensions[1]	Engagement[2]	Modes[3]	Control[4]	Mobility[5]
Exo-Glove Poly [23]	C	M	A	O	M
Virtual Haptic Device [24]	B	MC	A	C	L
Hand Exoskeleton Robot [25]	C	M	A	C	L
Finger Exoskeleton [26]	M	M	A	O	M
Soft Glove [27]	C	M	A	O	M
Current work	C	MC	A, R	C	H

[1] Dimension: B=Bulky, M=Moderate, C=Compact
[2] Engagement: M=Motor, MC= Motor-Cognitive
[3] Mode: A=Assistive, R=Resistive
[4] Control: O=Open loop, C=Closed loop
[5] Mobility: L=Low, M=Moderate, H=High

II. SYSTEM HARDWARE

Figure 1a displays comprehensive experimental arrangement, while Fig. 1b details the individual components of the setup. A rotary incremental encoder (A) is employed to gauge the angle at each joint. The Raspberry Pi Pico microcontroller (B) handles sensor data collection, pre-processing, and data transmission to the primary processor through Wi-Fi. The Raspberry Pi 4B microcomputer (C) executes the game and activates the pneumatic valves that control the actuator's force. An LCD monitor (D) is integrated into the setup and offers a visual interface for user interaction with the game. The Arduino Due board (E) transforms the digital output from the Pi 4B into an analogue form. A control valve (F) is in place to manage the pressure within the actuator's chambers. The pneumatic actuator (G) provides either assistive or resistive forces, aiding the patient's interaction with the setup.

The mechanism comprises two links: one supports the index and middle fingers together, while the other supports the thumb. Its revolute joints accommodate the metacarpophalangeal (MCP) joints of the fingers and the thumb carpometacarpal (CMC) joint. This design facilitates the opening and closing movements of both the fingers and the thumb, ensuring they have the necessary range of motion. To achieve the proper movement range, we utilized a specific pneumatic actuator, the Festo Round cylinder DSNU-S-8-30-P-A-MQ, which has a stroke length of 30 mm. The actuator can produce up to 22.6 N of force during retraction phases and up to 30.2 N during extension. In this system, the pneumatic pressure is controlled to regulate the interior pressure of the actuator chambers. For each participant, the pressure is set to a specific value based on their individual requirements, and the setup ensures that the desired pressure is accurately maintained.

Each joint is equipped with an encoder to track the movements of both the thumb and fingers and determine their relative positioning. For this purpose, the US Digital E16 micro-optical kit encoder was chosen. These encoders are ideal for limited spaces where high-volume data acquisition is necessary. Each encoder operates on a 5V supply and consumes 18-26 mA. They have a maximum output frequency of 1.6 MHz and output fall and rise times of 80 ns. With an ability to provide 2000 counts per revolution and a resolution of 0.18 degree, the encoders offer high precision.

Control over this device is achieved through a dual microcontroller setup, consisting of a Raspberry Pi 4 Model B (4GB) and a Raspberry Pi Pico W. This combination enhances the overall dependability of the system. The Raspberry Pi4 is powered by a 64-bit Broadcom BCM2711, a Quad-core Cortex-A72 (ARM v8) processor, complemented by 4GB LPDDR4-3200 SDRAM, and operates on the Debian-based Raspberry Pi OS. The Raspberry Pi Pico employs an RP2040 chip, designed in-house by Raspberry Pi which features a dual-core Arm Cortex-M0+ processor with a clock speed of up to 133 MHz, along with 264KB of SRAM and 2MB of flash memory.

Data acquisition is primarily handled by Raspberry Pi Pico, which preprocesses the data before forwarding it to Raspberry Pi4. The actuator then uses this data to apply the necessary forces during a patient's rehabilitation activity. The actuator airflow is regulated by analog signals originating from an

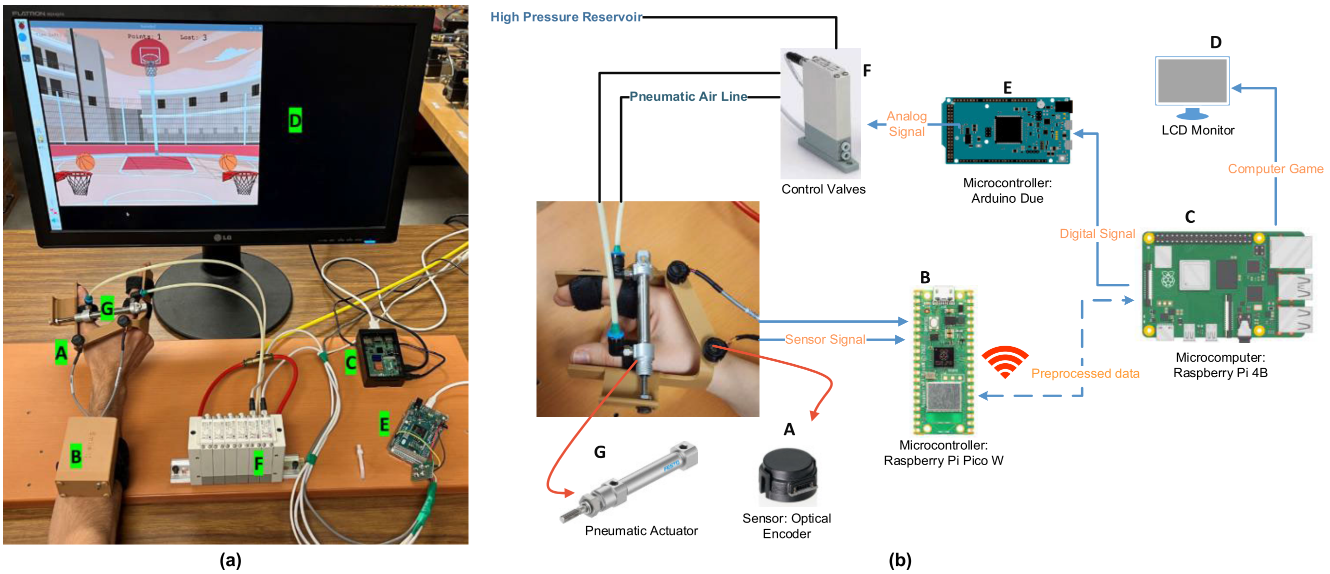


FIGURE 1. **(a) Experimental setup; (b) setup components [28].**

Arduino Due board, which operates on an Atmel SAM3 × 8E ARM Cortex-M3 CPU.

The system incorporates two independent boards, each fitted with Wi-Fi capabilities. The Raspberry Pi Pico, attached to the patient's forearm, forms a part of the wearable device. Due to the wireless technology on both boards, the patient retains mobility without risking wire damage. Had the design been limited to a single board, it would have restricted patient movement and their ability to maintain a distance from the screen, as the Raspberry Pi 4 requires direct connections to all interfaces, including the display monitor.

III. SYSTEM SOFTWARE

Programming is performed using Python, enabling easy access to various modules and libraries. Several steps are performed to achieve real-time interfacing, as illustrated in Fig. 2 and outlined in the following subsections.

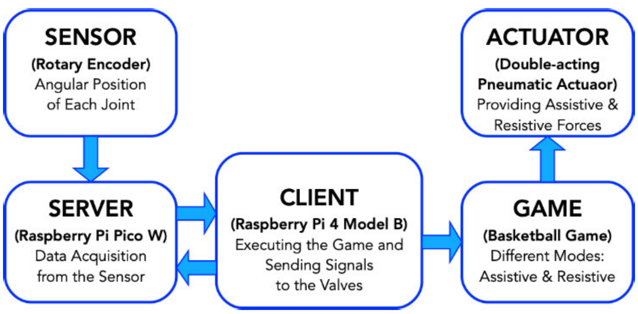


FIGURE 2. **Data handling from the sensor to the actuator.**

A. SENSOR AND ACTUATING FORCE

In the mechanism shown in Fig. 1, the angular position of each joint is quantified using an incremental rotary encoder. These encoders were chosen for their high precision capabilities, especially given the size constraints of the mechanism. However, it is important to note that these encoders are designed to track relative movement rather than pinpointing an absolute position. Consequently, starting the measurement process from a well-defined initial state, known as the 'reference state', is necessary each time the encoders are activated. This requirement is accomplished by an index signal. This signal generates a single pulse with each complete shaft rotation, identifying a specific, repeatable point in the cycle of the encoder's rotation. Ordinarily used for counting the number of shaft rotations, in our application, this index signal is critical for accurate identification of the reference position and precise monitoring of the movements of the mechanism during its opening and closing sequences. When either Index 1 or Index 2 is triggered, it sets the starting rotational angle of the corresponding link (first or second) to zero degrees. This is the main reference point from which all subsequent rotational measurements of the link are calculated.

In incremental encoders, orthogonal signal pulses are emitted during rotation to measure angular displacement and the direction of the rotation precisely. These pulses are necessary to ensure accurate measurements and highlight the importance of high-frequency signal reading from the encoders. One method to achieve high-frequency signal reading involves dedicating a processor core of the microcontroller for continuous signal reading from the sensors using multi-thread programming. This approach, however, may not be the most efficient in processing and could burden the microcontroller. The interrupt request (IRQ) technique is an alternate, more efficient method. In this method, any change in the encoders' signals triggers an interrupt signal to the processor, which then updates the counter to reflect each joint's angular position accurately. The IRQ is activated only upon detecting a change in the index signal, causing the processor to execute corresponding commands. Furthermore, the IRQ system accommodates both rotational directions, facilitating increments and decrements in the angle measurement. It is worth noting that the IRQ is conditional, as it is granted only when nothing more important is being processed. Therefore, to ensure the reliability and accuracy of our system, careful consideration and prioritization of interrupt handling have been implemented to ensure that the IRQ does not skip any signals related to measuring angular displacement accurately. Careful consideration involves thoughtfully analyzing which events or signals should trigger interrupts, such as changes in sensor signals in the present case. Prioritization of interrupt handling assigns priority levels to different interrupt sources to ensure critical interrupts, such as those related to measuring angular displacement, are processed promptly without skipping any signals.

The encoder connects to the Raspberry Pi Pico WH and is programmed using MicroPython - the primary language for this board. The system noise that can affect the encoder's output is categorized as follows:

(i) Electrical noise, stemming from external electrical interference with the encoder's signal, such as electromagnetic radiation or proximity to other electrical devices such as a WiFi router. While the encoder's digital output can tolerate a certain noise level, excessive electrical noise could corrupt sensor data. To mitigate this, sensor wires are shielded and securely soldered to the circuit board.

(ii) Mechanical noise, which could be caused by vibrations or issues like backlash in the shaft and encoder system. To improve accuracy and reliability, the mechanical design has been refined, and several iterations of 3D printing have been conducted.

A pneumatic actuator is integrated into the device, for gaming applications, providing assistive or resistive forces during gameplay. Valves control the actuator and receive analog signals from an Arduino Due board. Python controls Arduino's serial monitor window, ensuring the valves receive the correct voltage for each game event.

B. GAMING ENVIRONMENT

In this study, a computer game was developed utilizing the Pygame framework. This game features a scenario where two balls symmetrically emerge from the top of the screen, equally spaced from the central vertical line, and descend at a uniform velocity. The player's task is to manipulate the spacing between a pair of baskets placed on the floor, using their thumb and finger movements, aiming to let the balls fall through these baskets. The game's visual interface is shown in Fig. 3a. The correlation between the encoders' rotations and the gap between the baskets is depicted in Fig. 3b. During the game, the positions of the left ball and basket are captured at a frequency of 10 Hz. After the game, these data are utilized to graph the variation in distance between the balls and baskets over time, offering insights into the player's skill level and tracking patient rehabilitation progress.

The game offers three unique modes, each providing a distinct playing experience. In the standard mode, players rely solely on their thumb and finger movements without any external force assistance. In the assistive mode, the pneumatic actuator applies a constant force designed to help individuals align the baskets with the balls, thereby minimizing positional errors. The direction of this assistive force depends on the direction of the error—whether the ball is on the right or left side of the basket. The actuator adjusts to provide assistive force during either retraction or extension as needed, ensuring that the baskets are accurately positioned relative to the balls throughout the task. Conversely, the resistive mode increases the challenge by having the actuator apply force in the opposite direction, compelling players to work against it to guide the balls toward the baskets.

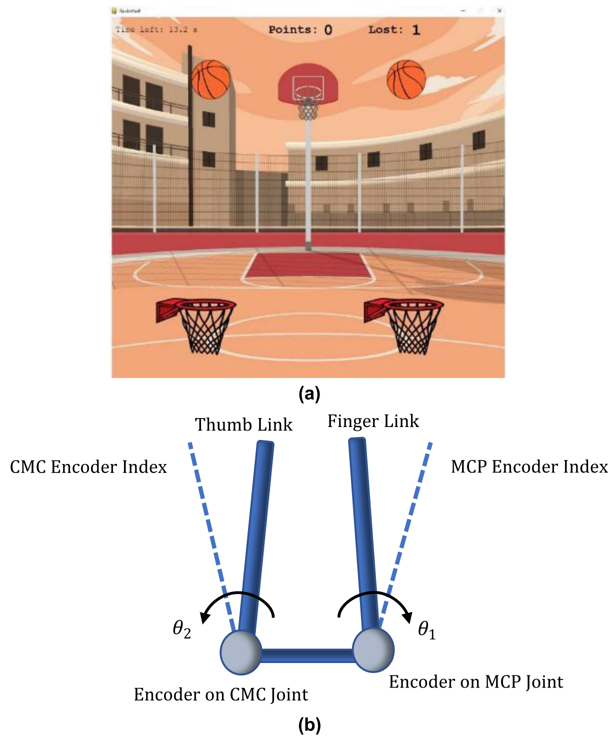


FIGURE 3. (a) Basketball game; (b) schematic of thumb-finger mechanism.

C. DATA TRANSFER

In the initial stages of data processing, the encoder's data undergoes a preprocessing step. This step involves aggregating the absolute rotation angles from both encoders, as shown in Fig. 3b, by the Raspberry Pi Pico. A wired or wireless connection between the two boards is essential to convey this preprocessed data to Raspberry Pi 4 and integrate it with the game dynamics. Given the device's design for easy and unrestricted use, particularly for patients during training sessions, a wireless connection has been chosen to eliminate the encumbrance of additional wires. To facilitate this,

TABLE 2. Information on participants with hand impairment.

Participant	Years post stroke	Affected side	Spasticity of fingers and wrists - Modified Ashworth Scale	Wolf Motor Function Test – Total Time (s)	Wolf Motor Function Test – Total Functional Ability Score
Participant 1	2	Right	0 *No spasticity (ataxia is present)	42.10	66
Participant 2	19	Left	1 *Higher spasticity (grade 2 over elbow flexors and shoulder adductors)	185.60	61

a server-client protocol has been implemented, with the Raspberry Pi Pico functioning as the server and the Raspberry Pi 4 as the client. They connect via a 2.4 GHz 802.11n wireless LAN, supported by both Raspberry Pi models. Following this, the transmitted data are integrated into the game, influencing the gameplay accordingly.

To manage the integration effectively, a distinct thread has been created for the game using multi-thread programming. This approach ensures that the game operates independently, avoiding interference with the ongoing data transmission and reducing system latency. On the Raspberry Pi 4B microcomputer, the gaming and data transfer tasks run simultaneously but on separate processor cores. The necessity for this separation arises from the distinct update frequencies required for the game screen and the data transfer. Running these processes in a single loop would compromise both frequencies, leading to increased system delay.

The frequencies for the various software components mentioned earlier are as follows. The encoder operates at 1.6 MHz, the IRQ at 33 MHz, the Wi-Fi connection at 2.4 GHz, and the Arduino's serial communication is set at a baud rate of 9600, which is equivalent to a data rate of 9.6 kbps (single-bit transmission).

IV. IMPLEMENTATION AND RESULTS

A. METHOD

The mechanism underwent testing with three participants. For the first participant, a left ischemic cerebrovascular incident resulted in impairment of control of the right dominant hand as well as numbness in the fingers and thumb. As for the second participant, the residual impairment following a right ischemic cerebrovascular insult resulted in limited control of the left hand and wrist.[1] The third participant was selected to have an uninjured hand to facilitate a comparison with the other two participants who had undergone a stroke. Table 2 provides clinical information for the first and second participants. The Ashworth test was used to grade spasticity of the hand [29], and upper extremity motor ability was measured by the Wolf Motor Function Test (WMFT) [30], [31]. Participants were asked to complete the 15 tasks of the WMFT as quickly as possible. The time taken to complete each task was recorded, and a time limit of 100 seconds was set for each task. The movement quality of each task was also graded using an ordinal scale of 0 to 5, where 0 indicates no performance, and 5 indicates normal movement. The final WMFT scores were the total time taken for the 15 tasks and the summed movement quality grades of the 15 tasks.

The participants selected for this study had varying levels of hand impairments, as summarized in Table 2. This variation was intentional to evaluate the effectiveness of the proposed mechanism across a range of injury severities, as well as to assess its performance in both assistive and resistive modes. By including participants with different impairment levels, we aimed to determine how well the mechanism can provide tailored rehabilitation support and accommodate various levels of hand functionality.

Four different modes were tested on participants during the evaluations to assess and analyze their performance thoroughly. The game provides the passive mode, the assistive mode, and the resistive mode. Several assistive robotic manipulandum and exoskeleton devices have been used by therapists in the rehabilitation process of children and adults with sensory-motor impairments of the upper extremity. These devices include various sensors to record segment motions (arm, forearm, and individual finger-thumb segments). The motion signals are used to interact with virtual avatars/objects, or to control a game sprite/paddle for play. However, these applications do not include actual object handling and manipulation. For successful recovery of precision, object manipulation tasks require practices that include sensory-motor processing of tactile and proprioceptive information.

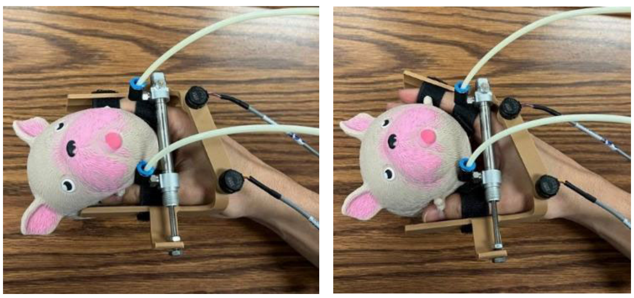
Four different modes were tested on participants during the evaluations to assess and analyze their performance thoroughly. The game provides the passive mode, the assistive mode, and the resistive mode. Several assistive robotic manipulandum and exoskeleton devices have been used by therapists in the rehabilitation process of children and adults with sensory-motor impairments of the upper extremity. These devices include various sensors to record segment motions (arm, forearm, and individual finger-thumb segments). The motion signals are used to interact with virtual avatars/objects, or to control a game sprite/paddle for play.

[1]Participants were recruited at the clinical rehabilitation research facility of the University of Manitoba. Participants had adequate vision to see images on a standard computer monitor. Exclusion criteria were: (a) excessive spasticity of the fingers and wrist (grade 2 and above on the Modified Ashworth Scale), (b) significant cognitive impairment (Montreal Cognitive Assessment scores less than 25), and (c) any other neurological disorder except a single stroke before testing. The University of Manitoba ethics board reviewed and approved the study, and all participants provided informed consent. University of Manitoba Human Research Ethics Board file number of approved study: HS25163 (H2021:335).

However, these applications do not include actual object handling and manipulation. For successful recovery of precision, object manipulation tasks require practices that include sensory-motor processing of tactile and proprioceptive information.

Somatosensory information is required to maintain stability, to detect and minimize slip of the moving object, and for eye-hand coordination [32]. For this reason, as shown in Fig. 4, we incorporated the handling of a compressible air-filled toy sphere, 5 cm in diameter, during gameplay. This involved flexion and extension movements of the finger and thumb, serving as the fourth mode of gaming. The toy is compressed by finger flexion or thumb flexion-adduction. During compression (Fig. 4a), resistance to the flexion movements would slightly increase. During movements involving finger extension or thumb extension-abduction, the compressed toy expands (Fig. 4b) and provides some movement assistance.

When the participant engages in the passive mode, the therapist can evaluate the participant's overall performance using the mechanism. This assessment helps determine the appropriate force for other gaming modes, such as assistive or resistive, to optimize rehabilitation.



(a) (b)

FIGURE 4. Handling of compressible air-filled toy sphere: (a) finger flexion or thumb flexion-adduction; (b) finger extension or thumb extension-abduction.

B. RESULTS

Four distinct operational modes of the device are explored: (i) passive mode, (ii) game with grasp, (iii) game with assistive force, and (iv) game with resistive force. Each game session lasts for 25 seconds. The ball's speed is consistent throughout the game, ensuring uniform motion durations. Exceptions occur for missed balls, which continue their trajectory until they reach the screen's boundary. These prolonged movements are omitted from the figures as the motion from the top edge of the baskets to the screen's boundary does not contribute to the analysis.

The mechanism's movement frequency correlates with the in-game ball speed, requiring players to adjust the mechanism accordingly by moving their fingers and thumb attached to it to catch the balls. In the tests' standard speed mode, the balls take about 4.2 seconds to reach the basket level. Considering the mechanism's movement from a completely closed to a fully open position (or the reverse) during each game event, the input signal's period, T, which is the time taken for one complete cycle of thumb-finger movement, is approximately 8.4 seconds.

The angular position of each link, denoted as θ, and its maximum rate of change are given in (1) and (2), respectively:

$$\theta = \frac{\mathrm{FS}}{2} \sin\left(\frac{2\pi}{T} t\right) \tag{1}$$

$$\max\left(\frac{\mathrm{d}\theta\,(t)}{\mathrm{d}t}\right) = \frac{\mathrm{FS}\pi}{T} \tag{2}$$

The amplitude of the signal is referred to as FS. The aperture time, T_a, is calculated based on (3) [8]. For effective interaction between the mechanism and the game, the quantization step, Q, is set to $Q = \frac{\mathrm{FS}}{10^6}$. Therefore, the sampling frequency, $f = \frac{1}{T_a}$, should be greater than the value indicated in (4) [28].

$$T_a \leq \frac{Q}{\max\left(\frac{\mathrm{d}\theta(t)}{\mathrm{d}t}\right)} = \frac{\frac{\mathrm{FS}}{10^6}}{\frac{\mathrm{FS}\pi}{T}} = 2.67 \times 10^{-6} \tag{3}$$

$$f \geq \frac{1}{2.67 \times 10^{-6}} = 0.374\mathrm{MHz} \tag{4}$$

Since data acquisition occurs at the sensor's maximum output frequency of 1.6 MHz, the sampling frequency meets the requirement of (4). As a result, the analog input signal is replicated with a minimal quantization error. It is important to note that the game's speed can be increased by up to four times. In such scenarios, recalculating these values yields a minimum sampling frequency of 1.496 MHz, which remains below the encoder's sampling frequency.

During the game and considering that each ball takes about 4.2 seconds to reach the basket level, each line in the following plots for different modes signifies the positional difference (PD) between the basket and the ball in the game as shown in (5), from the moment the ball appears on the screen to its disappearance (whether it lands in the baskets or is lost).

$$\mathrm{PD} = X_{\mathrm{Basket}} - X_{\mathrm{Ball}} \tag{5}$$

where X_{Ball} and X_{Basket} represent the distances of the balls and baskets from the game's central vertical axis, respectively. As shown in Fig. 5, the scoring range is defined to vary from 0 pixels to -100 pixels, as a ball within this positional difference will be successfully placed in the basket. The described system employs a closed-loop control approach over the positional difference, where real-time feedback from encoders is used to measure the bending angle of the finger. The logic continuously calculates the positional difference, applying assistive or resistive forces accordingly to assist or challenge the participants.

Figures 6 and 7 show that utilizing the compressible toy facilitated easier management of finger flexion and extension movements for participants with hand injuries. The compressible toy, introduced between the fingers and the thumb, provides participants with an added sense of grip and stabilization. During the handling of the compressible toy, there is

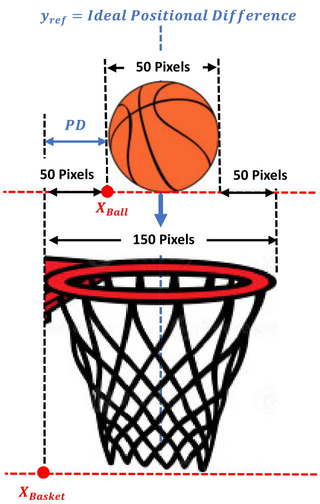


FIGURE 5. Ideal position of the ball and basket.

an increased sense of pressure and tension in the movements of the segments, thus increasing tactile and proprioceptive signals. This added information helps in movement control and the slight resistance helps to guide and stabilize the movements. Adding the compressible toy proved advantageous for the second participant who had limited control over thumb movements in various directions. Figure 8 shows the performance of Participant 3 with full control over their hand. As can be seen, grasping an object between the fingers did not exhibit a substantial improvement, compared to passive mode.

Referring to Table 3, the success rates in the passive mode for the first and second participants are 75% and 42%, respectively. In the grasping mode, these rates increase to 87% and 62%, respectively.

TABLE 3. Success rate of each gaming mode.

		Participant 1	Participant 2	Participant 3
Passive Mode	Success Balls	9	3	12
	Lost Balls	3	4	0
	Success Rate	75%	42%	100%
Assistive Mode	Success Balls	10	6	10
	Lost Balls	0	3	0
	Success Rate	100%	66%	100%
Resistive Mode	Success Balls	5	3	7
	Lost Balls	6	5	2
	Success Rate	45%	37%	77%
Passive Mode with Grasp	Success Balls	7	5	10
	Lost Balls	1	3	1
	Success Rate	87%	62%	90%

During the assistive and resistive simulation modes, the status of the actuator chambers changes based on the positional difference throughout the game. The Arduino Due's analog pins (DAC0 and DAC1) produce output voltages with a resolution of 8 bits, which implies that they can represent 256 different voltage levels between 0V and the reference voltage, typically 3.3V.

In the presented test, two control valves are utilized with voltage levels of 115 and 140. For the 115 voltage level, which is used for assistance or resistance during extension, the output voltage produced by the DAC pin can be calculated using (6), where V_{out} is the output voltage of the DAC pin, V_{ref} is the reference voltage of the Arduino Due and D is the digital 8-bit value. Applying this formula, the output voltage for the 115-voltage level is computed as 1.48 V. This voltage corresponds to a pressure of 9.48 psi in each chamber. When operating at this voltage, the actuator provides a force of approximately 3 N for extension.

$$V_{\text{out}} = \frac{V_{\text{ref}} D}{256} = \frac{3.3 \times 115}{256} = 1.48\text{V} \tag{6}$$

For the 140-voltage level, the same formula can be used to calculate the output voltage, resulting in 1.79 V. This higher voltage level provides assistance or resistance during retraction (closing). The higher voltage level of 140 corresponds to a pressure of 11.50 psi in each chamber, allowing the actuator to provide a force of approximately 3.3 N for retraction. This actuation level is sufficient to facilitate the targeted physical therapy intervention approved by the therapist.

Enabling the actuator force in the assistive mode leads to more accurate positioning of the basket in the desired spot compared to other modes, as depicted in Figs. 9a, 10a, and 11a. The positional differences are also lower than those observed in the two preceding modes, i.e., passive mode and passive mode with grasp. The reason for this can be credited to using the pneumatic actuator, which aids the player in minimizing the positional difference by utilizing feedback to maintain the assistive pressure proportional to the calculated positional difference.

Conversely, the resistive mode effectively provided valuable training for finger extensors. The resistive mode results, shown in Figs. 9b, 10b and 11b, demonstrate that the player requires more time to position the basket near the desired location, and the positional difference is notably higher. This is because the resistive force produced by the pneumatic actuator impedes the player's ability to manipulate the device freely, making it more challenging to overcome the applied force and move the baskets.

The information in Table 3 reveals that, under the assistive mode, the initial success rates for the first and second participants were 100% and 66%, respectively. Subsequently, when utilizing the resistive mode, these rates declined to 45% and 37% for the first and second participants, respectively. This trend is illustrated in Figs. 9, 10 and 11, where plots under the assistive mode exhibit convergence to the target value, in contrast to the plots for the resistive mode, which display a scattered pattern with a significant amount of positional difference.

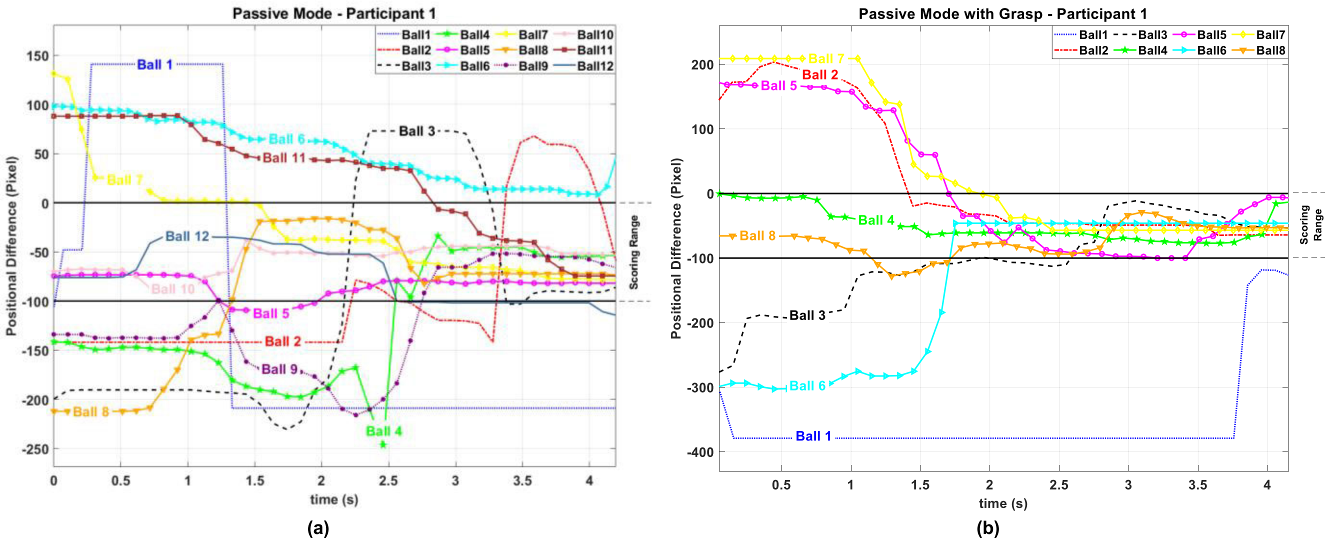


FIGURE 6. **Participant 1: Positional difference (PD) in (a) passive mode; (b) passive mode with grasp. One pixel is equivalent to 0.3 mm on the screen.**

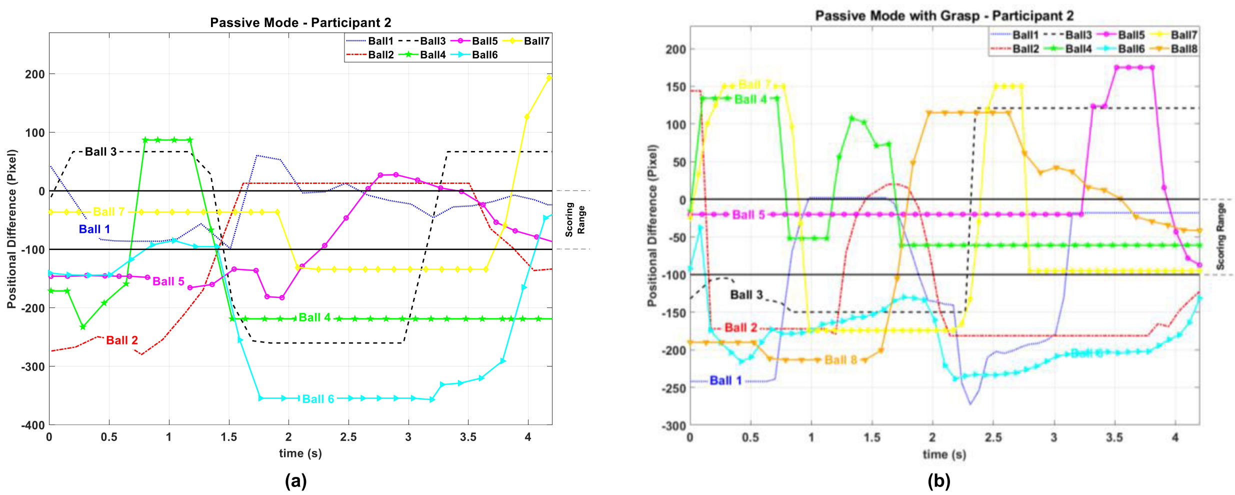


FIGURE 7. **Participant 2: Positional difference in (a) passive mode; (b) passive mode with grasp. One pixel is equivalent to 0.3 mm on the screen.**

Figures 9, 10, and 11, parts c and d, depict the actuator force during gameplay for each ball from its appearance on the screen until its disappearance. Based on the positional difference during the game, the actuator provides assistive or resistive forces. The actuator's force is directly dependent on the positional error of the ball. Both chambers are involved in generating assistive and resistive forces during both flexion and extension movements. Specifically, these figures indicate that the force alternates between two distinct values, reflecting a switch in the pressurized chamber based on the sign of the positional error. When the positional error transitions from negative to positive (or vice versa), the system responds by pressurizing the opposite chamber, resulting in a different force value. This behavior highlights the coordinated role of both chambers in dynamically adjusting the force to correct positional errors

One can compute the difference between the basket's movement trajectory and the ball's position, to assess how participants manipulate the device while attempting to position the ball accurately. To calculate the integral distance between the target ball and the basket during the game, the area under each plot is measured using the Integral Absolute Error (IAE) method [33]. For this purpose, the trapezoidal rule for integration is shown in Fig. 12 for one ball during the game and calculated using (7).

$$A = \int_{t_1}^{t_n} \mathrm{PD}(t)\,\mathrm{d}t \approx \sum_{i=1}^{n-1} \frac{(t_{i+1} - t_i)}{2} (|\mathrm{PD}(t_{i+1}) - y_{\mathrm{ref}}| + |\mathrm{PD}(t_i) - y_{\mathrm{ref}}|) \tag{7}$$

where A is the area under the curve shown in Fig. 12, n is the number of data points, t_i is the time step, $\mathrm{PD}(t_i)$ is the

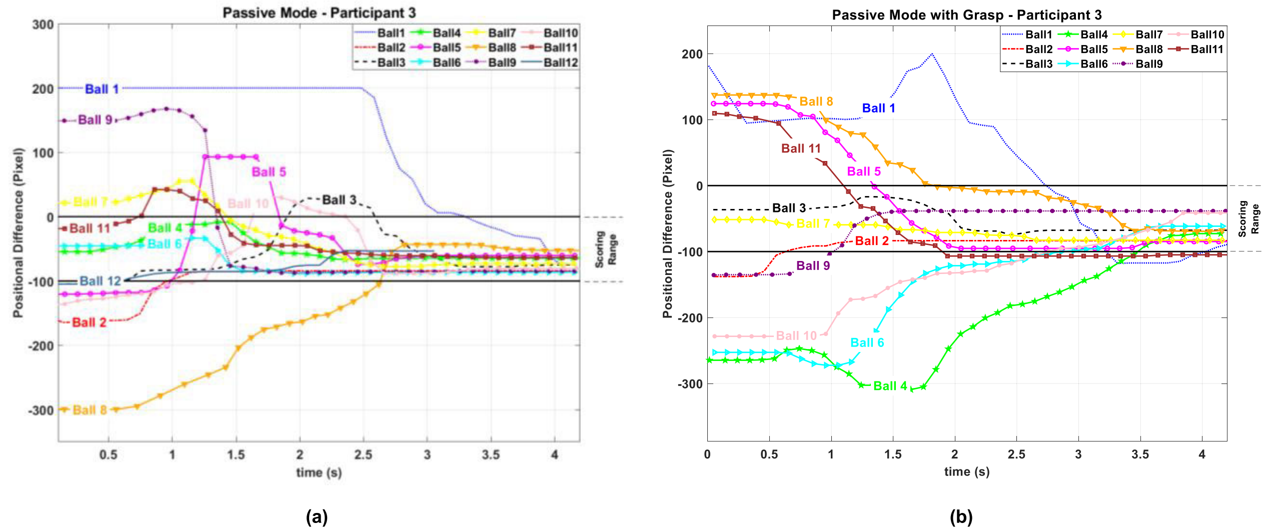


FIGURE 8. Participant 3: Positional difference in (a) passive mode; (b) passive mode with grasp. One pixel is equivalent to 0.3 mm on the screen.

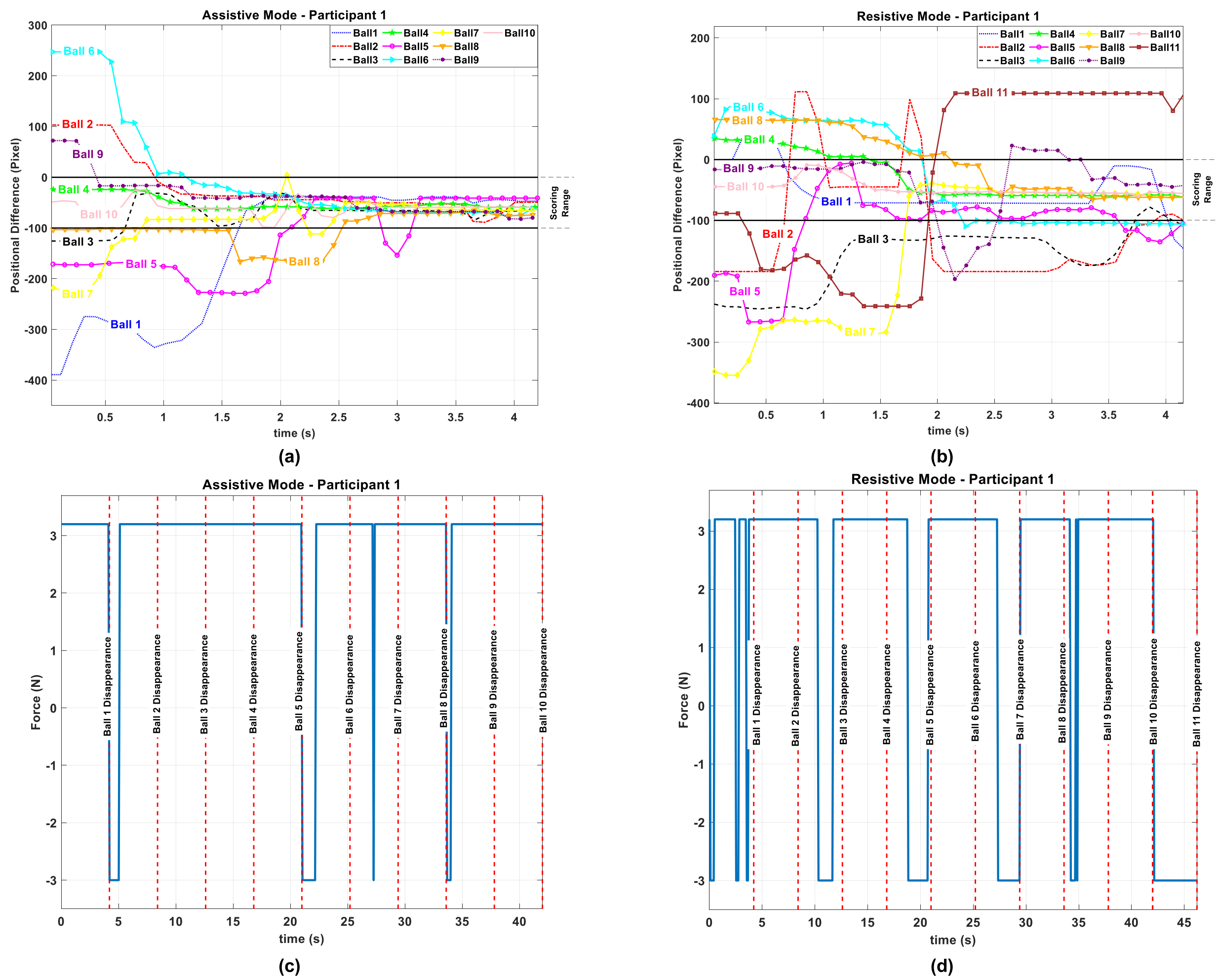


FIGURE 9. Participant 1: Positional difference in (a) assistive mode; (b) resistive mode; (c) actuator force in the assistive mode; (d) actuator force in the resistive mode. One pixel is equivalent to 0.3 mm on the screen.

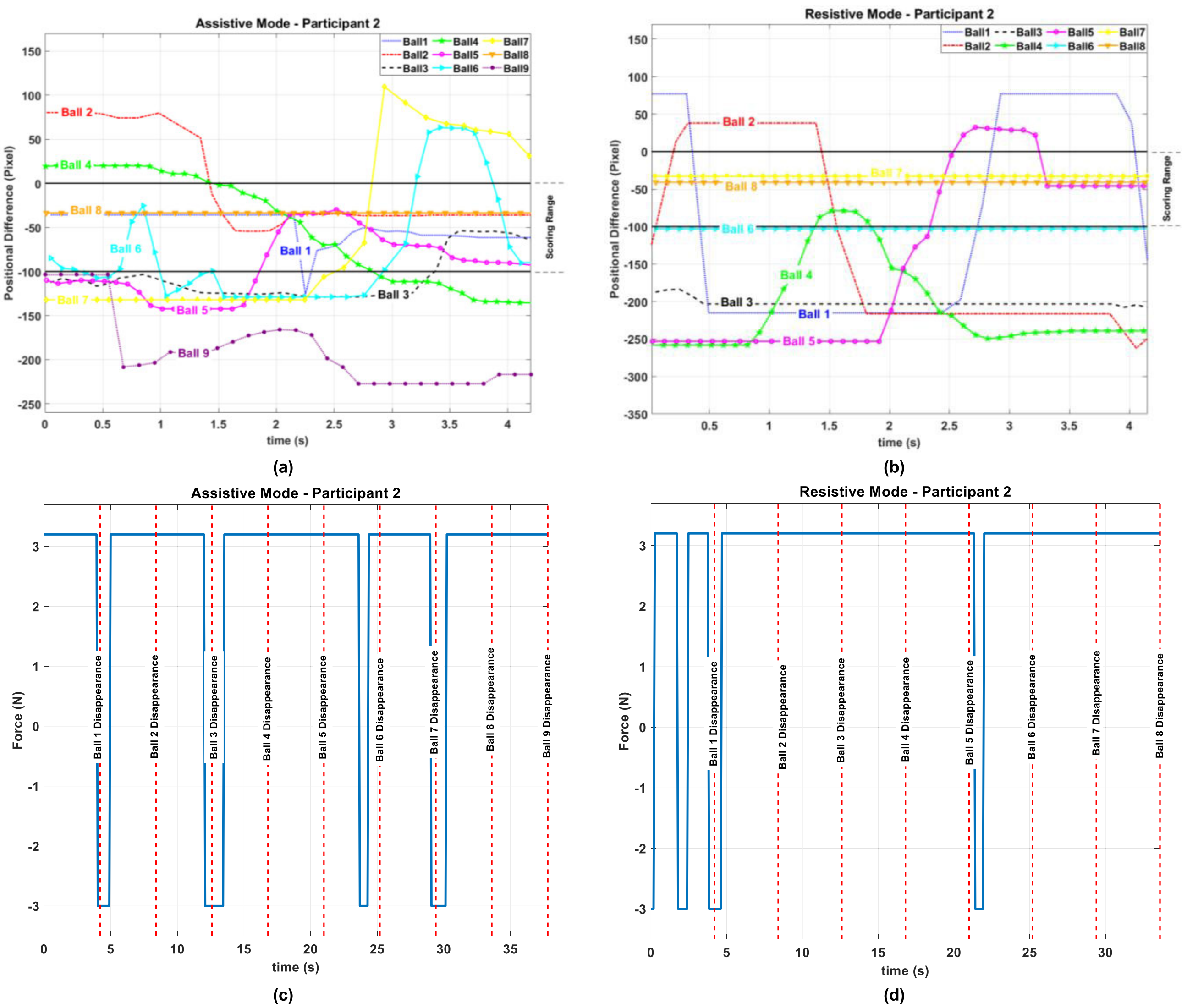


FIGURE 10. **Participant 2: Positional difference in (a) assistive mode; (b) resistive mode; (c) actuator force in the assistive mode; (d) actuator force in the resistive mode. One pixel is equivalent to 0.3 mm on the screen.**

positional difference corresponding to t_i. The IAE is computed relative to the ideal positional difference, y_{ref}, which corresponds to a positional difference of −50 pixels as shown in Fig. 5. During the game, balls randomly appear on the screen, and the initial position of each ball can significantly influence the degree of IAE in the basket's movement for those specific balls. Consequently, although the overall trajectory of some balls may seem desirable in the plots, the initial positional differences introduced by the random positions of the balls can lead to larger IAE for these plots. To address this issue, when calculating the IAE in Fig. 12 for balls appearing at random positions on the screen, we exclude the first second of each ball's presence, as this allows the player sufficient time to observe the ball and react to its position. Hence, the initial second is not factored into the IAE calculations, and the period from 1 second to approximately 4.2 seconds, during which the balls are detected, and the player adjusts the basket's position, is considered the appropriate range for IAE calculations.

With reference to Fig. 13, it is evident that the assistive mode yields the smallest IAE among all players when compared to other game modes. This reduction in IAE can be attributed to the actuator's assistance in guiding the player towards their intended position. In contrast, the largest IAE is observed in the resistive mode, where the resistive force hinders the player from effectively moving the basket. Notably, the use of a compressible toy for grasping results in a significant IAE reduction for players who have experienced a stroke and consequently struggled with hand control. For player 1, for example, the initial IAE for grasp mode is recorded as 223.51 Pixels*time. However, upon closer examination of the plot, it becomes evident that this IAE is primarily attributed to the first ball. Considering that players typically require some time to familiarize themselves with the game during the

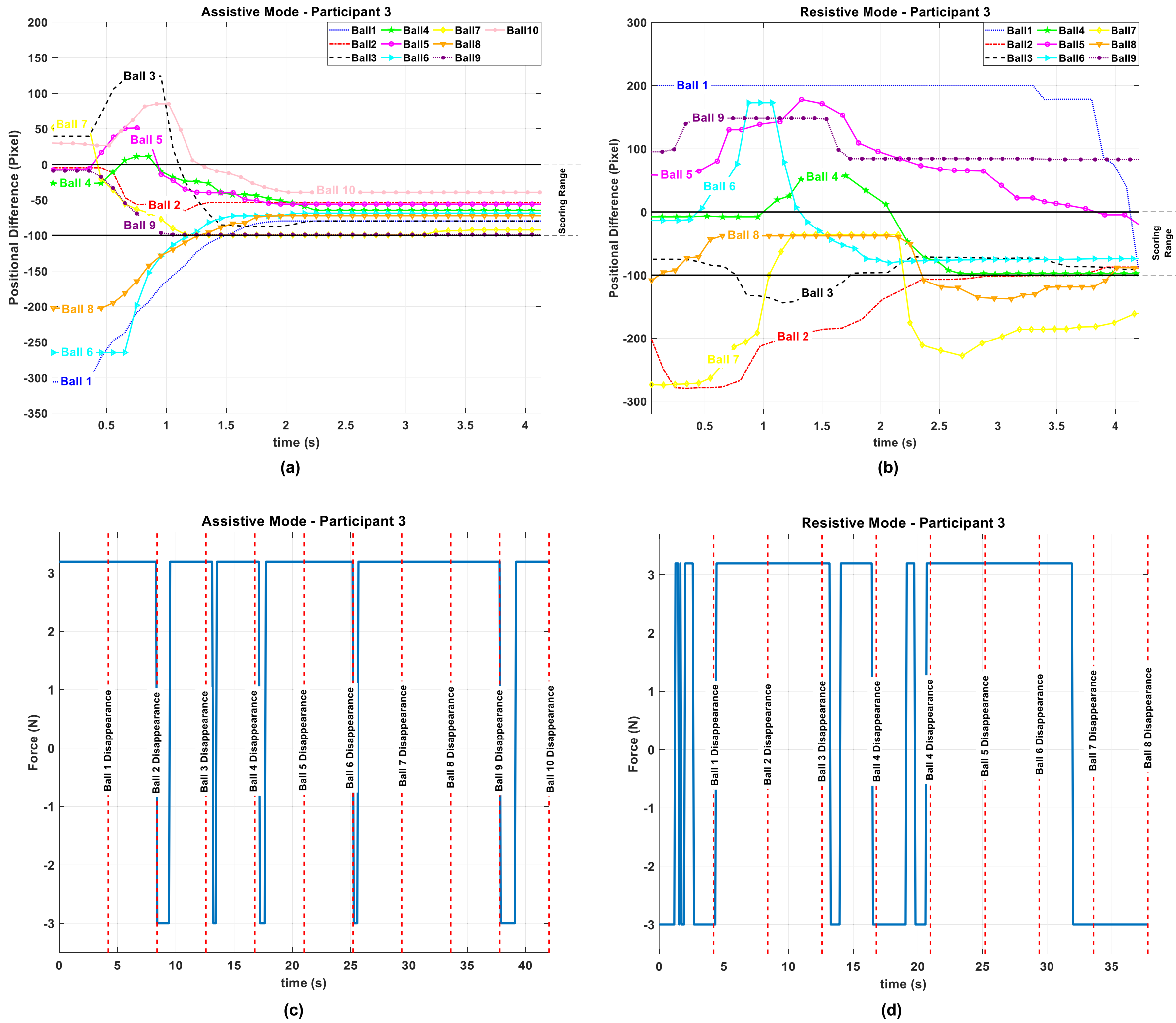


FIGURE 11. Participant 3: Positional difference in (a) assistive mode; (b) resistive mode; (c) actuator force in the assistive mode; (d) actuator force in the resistive mode. One pixel is equivalent to 0.3 mm on the screen.

first ball, excluding this ball from the calculations leads to a reduced IAE of 103.80 Pixels*time. As Participant 3 played with their healthy hand and had full control over it, grasping an object between the fingers did not improve the IAE when compared to the passive mode.

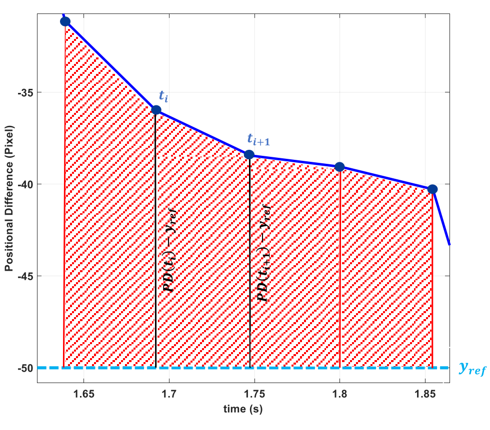
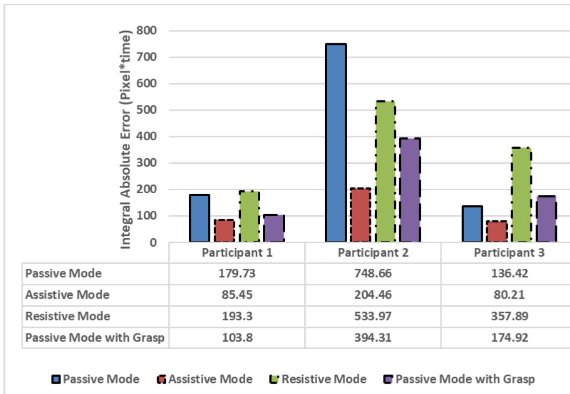


FIGURE 12. Trapezoidal rule for integration.



	Participant 1	Participant 2	Participant 3
Passive Mode	179.73	748.66	136.42
Assistive Mode	85.45	204.46	80.21
Resistive Mode	193.3	533.97	357.89
Passive Mode with Grasp	103.8	394.31	174.92

FIGURE 13. Average integral of absolute error. One pixel is equivalent to 0.3 mm on the screen.

Table 4 displays the error in movement for the four modes. This calculation involves considering the movement error of the lost balls, which is the disparity between the final position of the basket and the desired position where the ball is situated. The plots indicate a scoring range with an upper limit of 0 pixels and a lower limit of −100 pixels, designating successful ball placements within this range. For balls exceeding 0, the error is computed relative to the upper limit. The error is computed for those balls falling below −100, relative to the lower limit. Upon inspecting the plots, it is apparent that the first ball significantly influences the total error, particularly in the passive mode. This is attributed to participants familiarizing themselves with the game during the first ball, causing a notable impact on their readiness. Table 4 also provides total errors with the exclusion of the first ball.

TABLE 4. Movement error (pixels).

		Participant 1	Participant 2	Participant 3
Passive	Movement error	170	411	0
	Movement error excluding ball 1	61	411	0
Assistive	Movement error	0	182	0
	Movement error excluding ball 1	0	182	0
Resistive	Movement error	179	479	144
	Movement error excluding ball 1	129	386	144
Passive with Grasp	Movement error	26	174	4
	Movement error excluding ball 1	0	174	4

Insights into the system's delay characteristics are provided in Table 5. Data transfer between the two Raspberry Pi boards takes a swift 10 ms, while game screen updates have a slightly longer interval of 50 ms. Command signals are transmitted to the Arduino board with minimal latency, at just 0.1 ms. Users also have the option to utilize the device for other games, where it serves as a computer mouse, allowing seamless cursor control on the screen. This versatile functionality makes the device compatible with a wide range of commercial computer games that rely on mouse input. Notably, in this mode, the overall time delay adds up to 13 ms.

TABLE 5. Delay of the system.

	Delay (ms)	Boards	Connection
Data transfer	10	Raspberry Pi 4 – Raspberry Pi Pico	Wireless
Game events update	50	Raspberry Pi 4	-
Arduino serial communication	0.1	Raspberry Pi 4 – Arduino	USB
Moving mouse for other games	13	Raspberry Pi 4 – Raspberry Pi Pico	Wireless

Based on the results of the experiments and the observed system delay, the real-time interfacing of the setup with a game was successfully achieved. However, there are some items to consider for further enhancing the system setup and design:

- The rigid design of the device caused slight discomfort during experimental tests. Although it was designed to accommodate the movements of the MCP and CMC joints, the fixed positions of the other joints of the index finger and thumb led to minor discomfort, particularly for clients with more severe hand injuries. Enhancing the flexibility of the links could improve overall comfort.
- The current implementation focuses on the index finger and thumb; however, the proposed mechanism can be generalized for all five fingers. The primary consideration in extending it is the adaptation of the gaming interface logic to accommodate movements of additional fingers. Currently, the game logic includes specific events tailored for index and thumb movements, but with appropriate modifications, the system can be expanded to support a broader range of finger motions for comprehensive hand rehabilitation.
- The system operated with pre-defined settings that required manual adjustments for each patient, lacking the capability to dynamically adapt based on individual needs. It did not incorporate automated mechanisms for using patient-specific data—such as range of motion, resulting in a one-size-fits-all approach.

In future work, the experimental setup will be improved by the enhancements including using soft actuators and vision system for bending angle measurements to enhance user comfort and provide personalized interaction.

V. CONCLUSION

The main goal of this research was to develop an innovative tool for rehabilitation that uses computer games to provide engaging training and cognitive improvement. The system uses a wireless connection between the device (equipped with a Raspberry Pi Pico) and the computer (Raspberry Pi 4). The device can transmit data with a minimal delay of 10 ms, achieved through the integration of multithreading and IRQ capabilities in the software. This slight 10 ms delay allows participants to interact with the game in real-time. The game offers three different playing modes: passive, assistive, and resistive. Additionally, experimental tests were conducted by introducing another mode: passive with grasp. The developed system was tested with participants - two who had experienced strokes and one who was healthy. The overall performance of the device was evaluated based on their performance. The results showed the effectiveness of various gaming modes for rehabilitation purposes.

REFERENCES

[1] E. S. Lawrence, C. Coshall, R. Dundas, J. Stewart, A. G. Rudd, R. Howard, and C. D. A. Wolfe, "Estimates of the prevalence of acute stroke impairments and disability in a multiethnic population," *Stroke*, vol. 32, no. 6, pp. 1279–1284, Jun. 2001, doi: 10.1161/01.STR.32.6.1279.

[2] K. E. Laver, B. Lange, S. George, J. E. Deutsch, G. Saposnik, and M. Crotty, "Virtual reality for stroke rehabilitation," *Cochrane Database Systematic Rev.*, vol. 2018, no. 1, Nov. 2017, doi: 10.1002/14651858.cd008349.pub4.

[3] G. Burdea, N. Kim, K. Polistico, A. Kadaru, N. Grampurohit, J. Hundal, and S. Pollack, "Robotic table and serious games for integrative rehabilitation in the early poststroke phase: Two case reports," *JMIR Rehabil. Assistive Technol.*, vol. 9, no. 2, Apr. 2022, Art. no. e26990, doi: 10.2196/26990.

[4] M. Sierotowicz, N. Lotti, L. Nell, F. Missiroli, R. Alicea, X. Zhang, M. Xiloyannis, R. Rupp, E. Papp, J. Krzywinski, C. Castellini, and L. Masia, "EMG-driven machine learning control of a soft glove for grasping assistance and rehabilitation," *IEEE Robot. Autom. Lett.*, vol. 7, no. 2, pp. 1566–1573, Apr. 2022, doi: 10.1109/LRA.2021.3140055.

[5] G. B. Prange-Lasonder, B. Radder, A. I. R. Kottink, A. Melendez-Calderon, J. H. Buurke, and J. S. Rietman, "Applying a soft-robotic glove as assistive device and training tool with games to support hand function after stroke: Preliminary results on feasibility and potential clinical impact," in *Proc. Int. Conf. Rehabil. Robot. (ICORR)*, Jul. 2017, pp. 1401–1406, doi: 10.1109/ICORR.2017.8009444.

[6] S. Ghate, L. Yu, K. Du, C. T. Lim, and J. C. Yeo, "Sensorized fabric glove as game controller for rehabilitation," in *Proc. IEEE Sensors*, Oct. 2020, pp. 1–4, doi: 10.1109/SENSORS47125.2020.9278938.

[7] D. Halbhuber, N. Henze, and V. Schwind, "Increasing player performance and game experience in high latency systems," *Proc. ACM Hum.-Comput. Interact.*, vol. 5, pp. 1–20, Oct. 2021, doi: 10.1145/3474710.

[8] W. Kinsner, "Microcontroller, microprocessor, and microcomputer interfacing for real-time systems," Lecture Notes, Dept. Elect. Comput. Eng., Price Fac. Eng., Univ. Manitoba, Winnipeg, MB, USA, 2013.

[9] H. Tannous, T. T. Dao, D. Istrate, and M. H. B. Tho, "Serious game for functional rehabilitation," in *Proc. Int. Conf. Adv. Biomed. Eng. (ICABME)*, Sep. 2015, pp. 242–245, doi: 10.1109/ICABME.2015.7323297.

[10] M. Cantu, E. Espinoza, R. Guo, and J. Quarles, "Game cane: An assistive 3DUI for rehabilitation games," in *Proc. IEEE Symp. 3D User Interfaces (3DUI)*, Mar. 2014, pp. 43–46, doi: 10.1109/3DUI.2014.6798840.

[11] A. Ghosh and S. Saha, "Interactive game-based motor rehabilitation using hybrid sensor architecture," in *Handbook of Research on Emerging Trends and Applications of Machine Learning*. Pennsylvania, PA, USA: IGI Global, 2020, pp. 312–337, doi: 10.4018/978-1-5225-9643-1.ch015.

[12] E. Vogiatzaki, Y. Gravezas, and I. S. A. T. Solutions, "Rehabilitation system for stroke patients using mixed-reality and immersive user interfaces," in *Proc. Eur. Conf. Netw. Commun. (EuCNC)*, Bologna, Italy, 2014, pp. 23–26.

[13] G. Yap, C. Lau, K. Goonetilleke, and D. Levac, "REACH robot: Motion capture-based robotic interfaces to enhance engagement and adherence in pediatric rehabilitation," in *Proc. IEEE MIT Undergraduate Res. Technol. Conf. (URTC)*, Nov. 2017, pp. 1–3, doi: 10.1109/URTC.2017.8284173.

[14] N. Garcia-Hernandez, K. Huerta-Cervantes, I. Muñoz-Pepi, and V. Parra-Vega, "Personalized touch-based exergame system for unilateral and bilateral rehabilitation training," *Games for Health J.*, vol. 11, no. 3, pp. 157–167, Jun. 2022, doi: 10.1089/g4h.2021.0115.

[15] A. Athanasiou et al., "Neurorehabilitation through synergistic man-machine interfaces promoting dormant neuroplasticity in spinal cord injury: Protocol for a nonrandomized controlled trial," *JMIR Res. Protocols*, vol. 11, no. 9, Sep. 2022, Art. no. e41152, doi: 10.2196/41152.

[16] B. Homola, I. Sheldon, S. Ago, M. Mariani, and J. P. Hansen, "Prototyping exoskeleton interaction for game-based rehabilitation," in *Proc. Conf. Hum. Factors Comput. Syst. Extended Abstracts*, 2022, pp. 1–6, doi: 10.1145/3491101.3503566.

[17] O. Postolache, F. Lourenço, J. M. Dias Pereira, and P. Girão, "Serious game for physical rehabilitation: Measuring the effectiveness of virtual and real training environments," in *Proc. IEEE Int. Instrum. Meas. Technol. Conf. (I2MTC)*, May 2017, pp. 1–6, doi: 10.1109/I2MTC.2017.7969978.

[18] A. Krukowski and E. Vogiatzaki, "User interfaces and 3D environment scanning for game-based training in mixed-reality spaces," in *Proc. Int. Conf. ArtsIT, Interactivity Game Creation*. cham, Switzerland: Springer, 2017, pp. 227–237, doi: 10.1007/978-3-319-76908-0_22.

[19] H. Paraense, B. Marques, P. Amorim, P. Dias, and B. S. Santos, "Whac-A-mole: Exploring virtual reality (VR) for upper-limb post-stroke physical rehabilitation based on participatory design and serious games," in *Proc. IEEE Conf. Virtual Reality 3D User Interfaces Abstr. Workshops (VRW)*, Mar. 2022, pp. 716–717, doi: 10.1109/VRW55335.2022.00209.

[20] A. Ianoşi-Andreeva-Dimitrova and S.-D. Mândru, "Serious game for motor-imagery based brain–computer interface training," in *Proc. Int. Conf. e-Health Bioengineering (EHB)*, Nov. 2021, pp. 1–4, doi: 10.1109/EHB52898.2021.9657609.

[21] D. Camargo-Vargas, M. Callejas-Cuervo, and A. C. Alarcón-Aldana, "Brain–computer interface prototype to support upper limb rehabilitation processes in the human body," *Int. J. Inf. Technol.*, vol. 15, no. 7, pp. 3655–3667, Oct. 2023, doi: 10.1007/s41870-023-01400-w.

[22] M. N. Gaafer, Z. M. Ahmed, A. S. El-Husseiny, A. R. Youssef, and A. Al-Kabbany, "Immersive virtual reality games in neuromotor rehabilitation with brain–computer interfaces: A scoping review," *MedRxiv*, Jan. 2024, Art. no. 24300991, doi: 10.1101/2024.01.08.24300991.

[23] B. B. Kang, H. Choi, H. Lee, and K.-J. Cho, "Exo-glove poly II: A polymer-based soft wearable robot for the hand with a tendon-driven actuation system," *Soft Robot.*, vol. 6, no. 2, pp. 214–227, Apr. 2019, doi: 10.1089/soro.2018.0006.

[24] F. Wang, Z. Qian, Y. Lin, and W. Zhang, "Design and rapid construction of a cost-effective virtual haptic device," *IEEE/ASME Trans. Mechatronics*, vol. 26, no. 1, pp. 66–77, Feb. 2021, doi: 10.1109/TMECH.2020.3001205.

[25] J. Lai and A. Song, "Design and characterization of a novel compact hand exoskeleton robot for telerehabilitation and muscle spasticity assessment," *IEEE/ASME Trans. Mechatronics*, vol. 29, no. 4, pp. 2416–2427, Aug. 2024, doi: 10.1109/TMECH.2023.3336313.

[26] R. Liang, G. Xu, M. Li, B. He, and U. Khalique, "Fusing topology optimization and pseudo-rigid-body method for the development of a finger exoskeleton," *IEEE Robot. Autom. Lett.*, vol. 7, no. 2, pp. 1721–1728, Apr. 2022, doi: 10.1109/LRA.2021.3114418.

[27] J. Lai, A. Song, J. Wang, Y. Lu, T. Wu, H. Li, B. Xu, and X. Wei, "A novel soft glove utilizing honeycomb pneumatic actuators (HPAs) for assisting activities of daily living," *IEEE Trans. Neural Syst. Rehabil. Eng.*, vol. 31, pp. 3223–3233, 2023, doi: 10.1109/TNSRE.2023.3302612.

[28] N. Ghobadi, M. Khajoee, W. Kinsner, T. Szturm, and N. Sepehri, "Real-time interfacing of a pneumatically-actuated finger-thumb rehabilitation device," presented at the CIST, Aug. 2023, doi: 10.11159/cist23.145.

[29] A. D. Pandyan, C. I. Price, M. P. Barnes, and G. R. Johnson, "A biomechanical investigation into the validity of the modified Ashworth scale as a measure of elbow spasticity," *Clin. Rehabil.*, vol. 17, no. 3, pp. 290–294, May 2003, doi: 10.1191/0269215503cr610oa.

[30] S. L. Wolf, P. A. Catlin, M. Ellis, A. L. Archer, B. Morgan, and A. Piacentino, "Assessing wolf motor function test as outcome measure for research in patients after stroke," *Stroke*, vol. 32, no. 7, pp. 1635–1639, Jul. 2001, doi: 10.1161/01.str.32.7.1635.

[31] D. M. Morris, G. Uswatte, J. E. Crago, E. W. Cook, and E. Taub, "The reliability of the wolf motor function test for assessing upper extremity function after stroke," *Arch. Phys. Med. Rehabil.*, vol. 82, no. 6, pp. 750–755, Jun. 2001, doi: 10.1053/apmr.2001.23183.

[32] A. R. Sobinov and S. J. Bensmaia, "The neural mechanisms of manual dexterity," *Nature Rev. Neurosci.*, vol. 22, no. 12, pp. 741–757, Dec. 2021, doi: 10.1038/s41583-021-00528-7.

[33] M. K. Joyo, Y. Raza, S. F. Ahmed, M. M. Billah, K. Kadir, K. Naidu, A. Ali, and Z. Mohd Yusof, "Optimized proportional-integral-derivative controller for upper limb rehabilitation robot," *Electronics*, vol. 8, no. 8, p. 826, Jul. 2019, doi: 10.3390/electronics8080826.

NARGES GHOBADI (Graduate Student Member, IEEE) received the M.Sc. degree in mechanical engineering from Iran University of Science and Technology, Tehran, Iran. She is currently pursuing the Ph.D. degree with the Department of Mechanical Engineering, University of Manitoba, Winnipeg, MB, Canada. Her research interests include embedded systems, edge computing, robotics and robot control, human–robot interaction, and vision systems. She is an EIT Member with the Association of Engineers Geoscientists Manitoba.

WITOLD KINSNER (Life Senior Member, IEEE) is a Professor Emeritus with the Department of Electrical and Computer Engineering, University of Manitoba, Winnipeg, MB, Canada. He has been with the department, since 1978, and was a University Senator, from 2007 to 2020. He has been involved in research on robust real-time algorithms and software/hardware computing engines for signal and data processing in cognitive machines and systems, with an emphasis on multiscale and polyscale complexity metrics for dynamical systems. He has authored or co-authored over 860 publications, including journal and conference papers, patents, and books. He has supervised master's and doctoral theses of over 80 graduate students and over 200 undergraduate final-year capstone project students. He has taught 165 undergraduate and graduate courses, while developing new approaches to personalized experiential learning. Together with colleagues, he started the design and development of small satellites with the Faculty of Engineering under the Canadian Satellite Design Challenge (CSDC) and remained its Lead Advisor for four generations. He has dedicated much effort to continuing education and outreach focusing on indigenous students and gave on-demand certification courses to the amateur radio community. He was an Elected IEEE Canada President (2016–2017) and the Vice President of the IEEE Educational Activities (2018–2019).

TONY SZTURM is a Professor with the Department of Physical Therapy, College of Rehabilitation Sciences, University of Manitoba, Winnipeg, MB, Canada. His primary research area is the design, development, and validation of affordable computer-aided and game-based systems; and tools for rehabilitation and telerehabilitation. Application areas include upper extremity function, particularly manual dexterity, mobility limitations and cognitive decline, and vestibular disorders.

NARIMAN SEPEHRI (Senior Member, IEEE) received the M.Sc. and Ph.D. degrees from The University of British Columbia, Vancouver, BC, Canada. He is a Distinguished Professor with the Department of Mechanical Engineering, University of Manitoba, Winnipeg, MB, Canada. His research and development activities with the Fluid Power and Telerobotics Research Laboratory are primarily centered in all fluid power related aspects of systems, fault diagnosis, and manipulation and control.