FACULDADE DE ENGENHARIA DA UNIVERSIDADE DO PORTO

Serious games for hand and wrist rehabilitation

Jacinta Dias Ferreira

U. PORTO

FEUP FACULDADE DE ENGENHARIA UNIVERSIDADE DO PORTO

Master in Bioengineering

Supervisor: Prof. Joaquim Gabriel Magalhães Mendes (FEUP)

Co-Supervisor: Cláudia Daniela Costa Rocha (INESC TEC)

November 11, 2022

Serious games for hand and wrist rehabilitation

Jacinta Dias Ferreira

Master in Bioengineering

November 11, 2022

Resumo

O uso da mão e do pulso é crucial para a vida diária, e há muitas doenças e ferimentos que podem afetar o seu funcionamento. Devido a isto, manter o empenho dos pacientes na reabilitação da mão e pulso é importante para o processo de recuperação de muitas pessoas. Uma maneira eficaz de promover o empenho é o uso de jogos sérios, jogos cujo principal propósito não é o entretenimento. Esta estratégia tem recentemente sido aplicada em muitas áreas, incluindo reabilitação.

O objetivo deste trabalho foi o desenvolvimento de um sistema de reabilitação da mão e do pulso baseado em jogos sérios. O sistema é composto de duas partes. Uma luva de dados usa unidades de medição inercial para detetar flexão/extenção do pulso e pronação/supinação do antebraço, e sensores de flexibilidade para detetar flexão/extensão dos dedos. Uma aplicação para computador inclui dois jogos de reabilitação que podem ser adaptados ao nível de capacidade de movimento do paciente, dá feedback ao utilizador sobre o seu desempenho no jogo e amplitude de movimento, e permite-lhe avaliar o seu progresso. A informação obtida pela luva de dados é enviada para a aplicação através de Wi-fi.

Uma série de testes foi realizada no sistema para o avaliar. A capacidade da luva de dados de detetar a posição do pulso e dos dedos foi analisada, os jogos foram jogados várias vezes, ajustando os parámetros e assumindo diferentes níveis de incapacidade, e a função de exposição do progresso foi testada. O sistema desenvolvido permite aos pacientes treinar com independência, manter-se motivados durante a sua reabilitação, e perceber melhor a sua evolução.

Abstract

Hand and wrist use is crucial for daily life, and there are many illnesses and injuries that can affect their function. Because of this, maintaining patient engagement in hand and wrist rehabilitation is very important to the recovery process and well being of many people. An effective way of promoting engagement is the use of serious games, games whose main goal in not entertainment or enjoyment. This strategy has lately been applied to many fields, including rehabilitation.

The aim of this work was to develop a hand and wrist rehabilitation system based on serious games. The system is comprised of two parts. A data glove uses inertial measurement units to detect wrist flexion/extension and forearm pronation/supination, and flex sensors to detect finger flexion/extension. A computer application has two rehabilitation games which can be adapted to the patient's level of motion ability, gives the user feedback on their game performance and range of motion, and allows them to review their progress. The information obtained by the data glove is sent to the app through Wi-fi.

A series of tests were performed on the system to evaluate it. The data glove's ability to detect wrist and fingers position was analysed, the games were played several times, adjusting the game parameters, and assuming different levels of disability, and the progress function was tested. The developed system allows patients to train independently, stay motivated during their rehabilitation, and be more aware of their evolution.

Acknowledgments

I would like to thank my supervisors, Professor Joaquim Gabriel Magalhães Mendes and Cláudia Daniela Costa Rocha, for the availability and the guidance during the development of this work.

To my lab colleagues and to all the people who helped me carry out this work. I could not have made it without you.

To my family and friends, and particularly to my parents, for the support through my academic journey and through my life. Thank you for always being with me.

Jacinta Ferreira

Contents

List of Figures

List of Tables

Abbreviations and Symbols

ADL	Activity of Daily Living
CDC	Centers for Disease Control and Prevention
IMU	Inertial Measurement Unit
CP	Cerbral Palsy
PD	Parkinson's Disease
TIA	Transient Ischaemic Attack
SCI	Spinal Chord Injury
CTS	Carpal Tunnel Syndrome
ROM	Range of Motion
EMG	Eletromyography
ADC	Analog-to-Digital Converter
IoT	Internet of Things
I2C	Inter-Integrated Circuit
AHRS	Attitude and Heading Reference System
DCM	Direction Cosine Matrix
PI	Proportional Integral

Chapter 1

Introduction

Hand and wrist mobility is essential for people's daily lives. A great number of people in the world lose hand and wrist function every year, due to different conditions or injuries. Recovery can be complicated by the fact that rehabilitation is a time consuming and costly process, and, for many people, inaccessible. There is, therefore, a growing need for an automatizing of hand and wrist rehabilitation.

This chapter is divided in four sections: section 1.1 describes the context of this dissertation; section 1.2 explains the motivation behind its realization; section 1.3 defines its principal objectives; and section 1.4 delineates the structure of the remaining document.

1.1 Context

Hands are one of the most important tools we use to interact with the world. They perform activities that need a great deal of strength, such as carrying heavy objects, as well as the most delicate tasks the human body is capable of, such as writing. The use of a hand requires the sometimes very precise movement of the wrist and fingers.

The wrist contains one joint, which allows for flexion/extension and abduction/adduction movements. The fingers have four joints, also capable mostly of flexion/extension and abduction/adduction, while the thumb's three joints permit its rotation as well. Wrist position is also affected by the pronation/supination of the forearm [1]. The fine motor control of all these joints is essential for the complex hand movements that are frequently required in everyday life. This constant and often physically demanding movement exposes the limb to stress, and to possible injuries.

Hand and wrist disability can have several causes, such as injuries, strokes, neurological or musculoskeletal diseases, or geriatric complications. Physiotherapy is a rehabilitation treatment that can be used to help these patients regain mobility. It is often based on repetitive movements, which promote neuroplasticity, accelerating the recovery [17]. However, this repetitive nature of the treatment can easily decrease the motivation of the patients. Additionally, they will often be required to perform exercises at home, without the guidance of a therapist, which makes it less likely they will perform the tasks, or perform them correctly, and can cause both the patient and the therapist to be unsure of the result of the exercises.

To counter these problems, serious games are sometimes used. These are games whose main purpose is not entertainment [18]. The integration of rehabilitation tasks in an interactive, entertaining platform encourages patients to persist in their efforts and can help them obtain a better understanding of their performance and evolution. These platforms also often include various functionalities which offer the patient more independence and a more active participation on their own recovery process.

1.2 Motivation

The loss of hand and wrist mobility prevents many people from being able to perform activities of daily living (ADLs), greatly reducing their quality of life. According to the Centers for Disease Control and Prevention (CDC), there are 4,7 million adults in the US who find it very difficult or impossible to handle small objects [19].

Physical therapy is a solution, but it brings problems of its own. Patients will often become disinterested, feel they are not sufficiently involved in their rehabilitation, and that it is not being helpful. According to a 2022 survey, 47% of patients would like to understand their recovery better, and 24% of patients that gave up on therapy did so because they believed they were not improving [20]. The same survey found that 72% of physical therapy patients didn't perform their home exercises, which delays their recovery. Rehabilitation systems based on serious games are a good solution to increasing engagement. However, they can have some disadvantages. They are often very expensive, and many of them are bulky, heavy, and hard to move, constraining the rehabilitation process.

1.3 Objectives

The aim of this dissertation is the development and implementation of an interactive system for hand and wrist rehabilitation. It is composed of a data glove, which captures the user's hand and wrist movement, and a computer application. The glove holds flex sensors, which measure finger positions, and two inertial measurement units (IMUs), which measure wrist position. The application replicates the movement, giving the user real time visual feedback of their active range of motion, and includes two serious games based on rehabilitation exercises. The games instruct the player to open/close the hand, extend different numbers or combinations of fingers, and move the wrist. The application gives the user feedback on their performance and allows them to review their progress.

To accomplish this goal the following steps were defined:

- Creation of a data glove, with sensors which capture wrist flexion, extension, pronation and supination, and finger flexion and extension.

- Development of an application with a virtual model of a hand and two rehabilitation games
- Integration of the glove and application to form a finished system

The main contribution of this dissertation is the creation of a low cost, user friendly system, which enables patients to easily perform hand and wrist rehabilitation exercises without the direct supervision of a therapist, helps them maintain interest, and allows them to monitor their progress.

1.4 Dissertation Structure

This document is organized in six chapters. Chapter 1 introduces the work, giving the context and motivation, and detailing the objectives. Chapter 2 explains concepts related to hand and wrist rehabilitation, including hand and wrist anatomy, conditions that affect these body parts, and techniques used to recover their use. Chapter 3 describes the already existing serious games based systems for hand and wrist rehabilitation, and discusses their advantages and disadvantages. Chapter 4 describes the developed rehabilitation system: the data glove and the application. Chapter 5 describes the prototype validation process and discusses its results. Finally, chapter 6 presents the main conclusions of this work, and makes suggestions for the future development of the system.

Chapter 2

Hand and wrist

This chapter presents an overview of the anatomy of the hand and wrist, and the conditions that can lead to a loss of their function. Techniques and activities used for the recovery of this function are also described.

2.1 Hand and wrist anatomy

The wrist is the area between the forearm and the hand. It is composed of eight bones, the carpal bones, in two rows (Figure 2.1). In the proximal row are the scaphoid, the lunate, the triquetrum, and the pisiform. In the distal row are the hamate, the capitate, the trapezoid, and the trapezium. The eight bones, together, form an anteriorly concave surface. The carpal ligament goes from the trapezium to the hamate, forming, in the anterior surface of the wrist, the carpal tunnel, through which blood vessels, nerves and tendons pass [1].

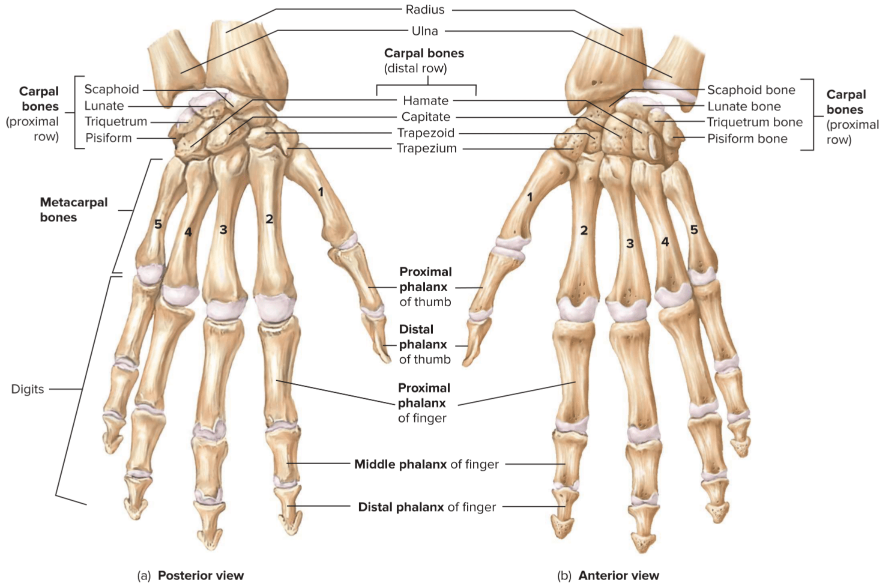


Figure 2.1: Bones of the hand and wrist [1]

The wrist joint, also known as radiocarpal joint, connects the forearm to the hand. It is classified as a synovial condyloid joint. A condyloid, or ellipsoid, joint has articular surfaces shaped as an ellipsoid. It allows for flexion, extension, abduction, and adduction, as shown in Figure 2.2. The joint is formed by the articulation of the distal end of the radius, and the scaphoid, lunate, and triquetrum bones. The distal part of the joint (carpal bones) is convex, and the proximal part (radius and articular disk) is concave.

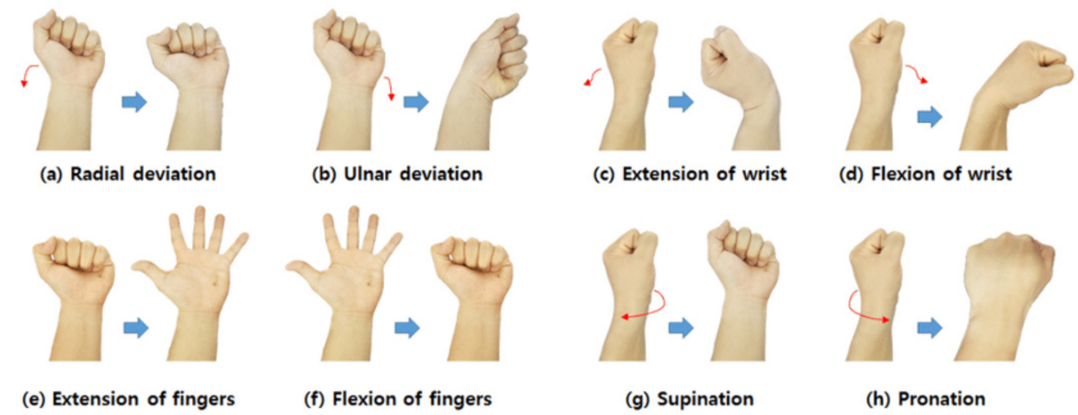
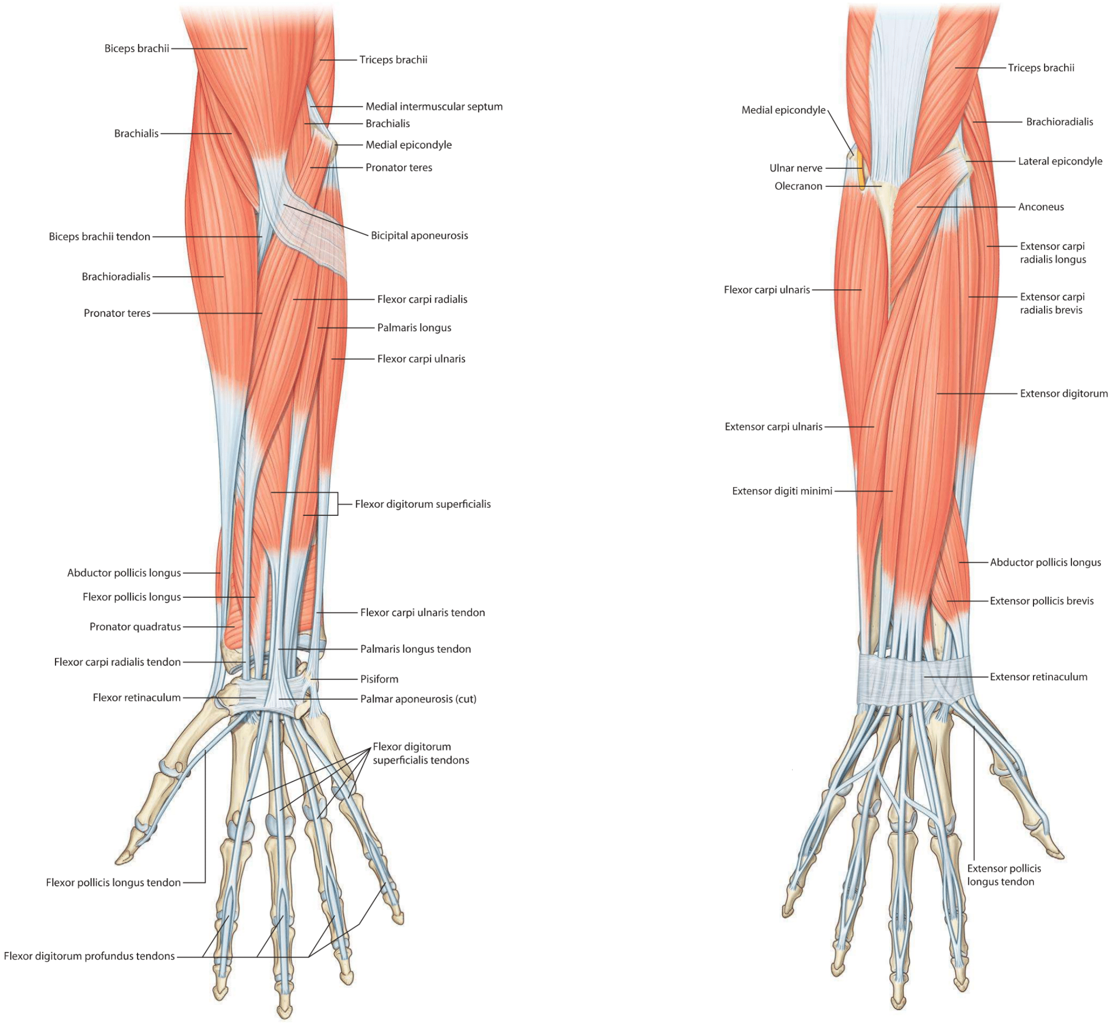


Figure 2.2: Hand and wrist movements [2]

The joint contains four ligaments. The palmar radiocarpal ligament is on the anterior side of the wrist and connects the radius to both rows of carpal bones. Its purpose is to guarantee stability and to ensure the forearm and hand move together during supination. The dorsal radiocarpal ligament, placed in a similar way on the posterior side of the wrist, ensures that the forearm and hand move together during pronation. The ulnar and radial collateral ligaments go from the ulna and radius to the carpal bones, and prevent excessive ulnar and radial deviation of the hand. The joint is surrounded by a joint capsule, which connects to the bones, and contains synovial fluid in its interior, the joint cavity. The joint obtains blood from the ulnar and radial arteries, which form the dorsal and palmar carpal arches in the hand. It is innervated by the median, radial, and ulnar nerves [21].

Wrist movements are controlled by several muscles (Figure 2.3). Wrist flexion is mostly caused by the flexor carpi radialis, and the flexor carpi ulnaris. Wrist extension is acomplished by the extensor carpi radialis longus, the extensor carpi radialis brevis, and the extensor carpi ulnaris. Wrist abduction (radial deviation) is controlled by the abductor pollicis longus, the flexor carpi radialis, the extensor carpi radialis longus, and the extensor carpi radialis brevis, while wrist adduction (ulnar deviation) happens through the action of the flexor carpi ulnaris and the extensor carpi ulnaris [21].

The hand is composed of the metacarpal bones and the digits. The metacarpal bones are five and are attached to the carpal bones. They form a concave surface, the palm of the hand. The digits consist of one thumb and four fingers, which are composed of phalanges. The fingers have three phalanges (proximal, middle, and distal), while the thumb has only two (proximal and distal) [1]. There are four joints in the fingers: the distal interphalangeal joint and the proximal



(a) Anterior view (b) Posterior view

Figure 2.3: Muscles of the hand, wrist, and forearm [3]

interphalangeal joint (which flex and extend the finger), the metacarpophalangeal joint (responsible for finger flexion, extension, abduction and adduction) and the carpometacarpal joint. As for the thumb, it has three joints: the interphalangeal joint, the metacarpophalangeal joint (which causes thumb flexion and extension), and the carpometacarpal joint (flexion, extension, adduction, abduction, opposition, and reposition of the thumb).

The interphalangeal joints are synovial hinge joints. They are surrounded by a joint capsule and stabilized by the two collateral ligaments and the palmar ligament [22]. The metacarpophalangeal joints are synovial condyloid joints. They also contain a capsule, collateral ligaments, and a palmar ligament. The joints of the four fingers are connected by the deep transverse metacarpal ligaments [23]. Finger carpometacarpal joints are synovial plane joints. They are capable of very little movement, especially the more lateral ones. The four joints share a joint capsule and are reinforced by the dorsal and palmar carpometacarpal ligaments, and the interosseous ligaments [24].

Table 2.1: Some disorders that cause hand impairment [17]

Disease	Prevalence (US)	Type of impairment	Impairment prevalence
Arthritis	78 million (by 2040)	Grasping	3 million (2009)
Cerebral palsy	1 in 323 children (2008)	Arm-hand dysfunction	50%
Parkinson’s Disease	500k (2010)	Tremor, rigidity	Not reported
Spinal Cord Injury	282k (2016)	Tetraplegia	58.3%
Stroke	795k (incidence, 2016)	Upper extremity hemiplegia	50%

The thumb carpometacarpal joint, or trapeziometacarpal joint, is a synovial saddle joint. It is strengthened by the radial carpometacarpal ligament and the anterior and posterior oblique ligaments. Unlike the finger equivalents, it is very mobile, allowing for flexion, extension, abduction, adduction, and rotation [25].

The fingers are flexed by the flexor digitorum superficialis and the flexor digitorum profundus and extended by the extensor digitorum. The index finger is also extended by the extensor indicis. The little finger has two additional muscles, the extensor digiti minimi, and the flexor digiti minimi brevis. The flexion of the thumb is performed by the flexor pollicis brevis and the flexor pollicis longus, and the extension by the extensor pollicis longus and the extensor pollicis brevis [1].

2.2 Hand and wrist disability

Loss of hand function can happen for various medical reasons. Chu et al. [17] has listed its most frequent causes, which can be seen in table 2.1:

- The word "arthritis" is used to describe various conditions that cause swelling or inflammation of the joints. Common symptoms are joint pain and stiffness. Osteoarthritis is the most common type. It is caused by the breakdown of cartilage, and later bone, usually in the hands, hips, or knees. Risk factors include repetitive stress on the joint, age, obesity, and joint injury, as well as genetic factors. It is treated by physical therapy, medication, risk factor management, and, if necessary, surgery [26].

- Rheumatoid arthritis is an autoimmune disease in which the body’s immune system attacks healthy tissue. In this case, the synovial membrane of joints, mostly in the hands, wrists, and knees, becomes swollen. The inflammation causes further damage to the joint tissue. The disease can also affect other areas of the body, and causes fever, weakness, and weight loss. It is found more often in women, older people, and people who smoke, are overweight, or have certain genes. The treatment involves medication, surgery, and physical therapy [27].

  There are over 100 other types of arthritis, such as gout, fibromyalgia, and lupus [26].

- Cerebral palsy (CP) is a group of disorders that affect movement, balance, and posture. It is the most common motor disability in children. There are several types of CP, depending on the kind of movement they hinder. Spastic CP is the most common, and is characterized by high muscle stiffness. Dyskinetic CP causes uncontrollable movements of the limbs and sometimes face. Ataxic CP hampers balance and coordination. Some patients have more than on type of CP. The conditions are treated with medication, surgery, and physical, occupational, and speech therapy [28].

- Parkinson's disease (PD) is a neurodegenerative disorder which causes uncontrollable tremors, slowness of movement, limb stiffness, and balance and coordination problems. Other symptoms are depression, anxiety, constipation, sleep disorders, and cognitive impairment. The symptoms tend to worsen over time. The condition is believed to be caused by a combination of genetic and environmental factors [29]. It is possible to alleviate some symptoms with medication, surgery, and treatments such as deep brain stimulation, physical, occupational, and speech therapy, and diet changes [30].

- Stroke is a condition in wich an artery that irrigates the brain is blocked by a blood clot (ischaemic stroke) or ruptures (hemorrhagic stroke), leaving part of the brain without access to oxygen. The brain's blood supply can also be obstructed by a temporary clot, causing a small stroke, known as transient ischaemic attack (TIA). The death of brain cells causes facial droop, arm weakness, and speech difficulty, among other symptoms. Ischaemic strokes are treated with clot dissolving medicine, as well as other drugs and procedures. Hemorrhagic strokes are mostly treated with surgery. Long term recovery may involve physical, speech, and occupational therapy, depending on the problems each patient has [31].

- Spinal cord injury (SCI) happens when there is damage to the nerves that send signals between the brain and the rest of the body, usually due to physical trauma. This loss of communication can be total or partial, and affects the body below the injury site. This includes loss of or change in sensation, pain, weakness and loss of movement. Depending on the way the damage happened and its severity, surgery may be necessary, along with various kinds of medication, breathing assistance, and treatment for other consequences of the injury. Physical rehabilitation can be used to help recover limb function [32].

There are as well some common conditions that affect hand movement specifically:

- Stenosing tenosynovitis (trigger finger/thumb) is a condition in which a tendon or tendon tunnel thickens, causing the tendon to become irritated. When the finger is flexed, the tendon catches in the tunnel mouth, causing pain, stiffness, and tenderness. The finger often clicks when moved, or gets stuck in a bent position. Treatment options include using a splint, avoiding activities that intensify the pain, medication, and surgically expanding the tendon tunnel [33]. De Quervain's syndrome, or De Quervain tenosynovitis, affects the tendons of the abductor pollicis longus and extensor pollicis muscles, causing the same symptoms in the thumb and wrist [34].

- In carpal tunnel syndrome (CTS), the median nerve is compressed in the carpal tunnel. Affected people feel tingling, numbness, weakness and pain in the hand. The symptoms often worsen over time. In cubital tunnel syndrome, the ulnar nerve is compressed as it passes through the cubital tunnel, in the elbow. The symptoms affect especially the ring and little fingers. The conditions can be treated with medication, the use of a splint, or surgery to reduce the pressure on the nerve [33].

- Dupuytren's disease, also known as Dupuytren's contracture, is a condition in which nodules appear in the ligaments of the hands, and sometimes form a cord that prevents a finger from being fully extended. Surgery may be used to remove part or all of the cord, returning some mobility to the finger [33].

2.3 Hand and wrist rehabilitation

Physical therapy can be used to recover hand function, as well as reducing pain, swelling, or other symptoms. Joint mobility is measured by the range of motion (ROM). Passive ROM is the total movement a joint can make under an external force. Active ROM is measured when the movement is caused only by the individual's muscles. To regain hand movement and strength, a therapist will chose rehabilitation exercises, depending on the needs of the patient. The exercises can be performed in three modes: active (the patient must make the movement entirely on their own), passive (the patient does not need to make effort to make the movement), or active assisted (the patient must make some effort, but is assisted by an outside force) [35].

The ROM of the wrist can be expanded by wrist flexion/extension, wrist ulnar/radial deviation, wrist circumduction, and forearm pronation/supination. To improve hand movement, some exercises include flexing/extending the fingers, joint blocking (part of the finger is held still and the finger is flexed and extended at the joints above it), tendon gliding (making different types of fists, then extending the fingers), and touching the thumb to each of the fingers [36] [37].

There are several techniques that can be used in hand rehabilitation [38]:

- Action observation therapy requires the patient to watch a healthy individual performing the tasks, and repeating them. The demonstration is often shown in video format.
- In mirror therapy, the patient performs the exercises with the affected limb and the healthy limb at the same time. A mirror is positioned so that the patient sees a reflection of the healthy limb in place of the impaired one, giving the illusion of normal movement.
- Bilateral training involves executing rehabilitation exercises with both limbs simultaneously, to encourage the preservation of limb coordination, necessary for bilateral tasks.
- In resistance training, the patient must execute a movement against the resistance of an external force.

The application of robotic devices to physiotherapy is very useful since they can provide movement assistance, helping the patient perform the exercises, or even moving the affected limb with no effort from the patient. By replacing the therapists in this way, it frees them to other activities and reduces their tiredness [38].

Chapter 3

State of the art

This chapter presents a study of the systems for hand and wrist rehabilitation based on serious games found available in the market. All of them target the rehabilitation of the wrist and/or fingers, but some also allow for elbow and shoulder movements. Serious games can be defined as games whose principal goal is not entertainment or enjoyment. They are often applied to several fields, such as health, education, or communication [18]. Serious games help increase the motivation of the targeted audience for the game's intended purpose. This makes their application to rehabilitation very useful.

3.1 Hand and wrist rehabilitation systems

There are several hand and wrist rehabilitation systems based on serious games in the market. Many of these can be found in literature surveys, such as [39], which reviews the gamification of physical rehabilitation for stroke patients, and [17], which describes soft robotics devices for hand rehabilitation.

The Neofect Smart Glove (Figure 3.1), previously known as RAPAEL Smart Glove is a wearable device for the rehabilitation of the wrist and fingers. The glove incorporates an IMU sensor, which measures the 3-dimensional orientation of the hand, and 5 bending sensors, which measure the movement of the fingers [40].

The system also includes video training games which promote the movement of the wrist and fingers. Many of these games represent ADLs, such as turning pages, filing a glass, chopping vegetables, squeezing oranges, or cleaning a table. They are classified as to the movements they target: forearm pronation/supination, wrist flexion/extension, wrist radial/ulnar deviation, finger flexion/extension, and complex movement [41]. The system's Learning Schedule Algorithm automatically adjusts the game's difficulty level, according to the patient's information and performance. The patient's performance is evaluated by several variables, such as passive and active ROM, and analysis of the movements, depending on the game. The patient can see their result and

Figure 3.1: Neofect Smart Glove [4]

their improvement in the performance report [42]. Users of the Neofect Smart Glove have demonstrated statistically significant improvements in hand function when compared to those receiving only standard occupational therapy [41].

PABLO® Upper Extremity (Figure 3.2(a)) is a device for rehabilitation of the arm. The system has three elements: the Handle, the Multiball, and the Multiboard. The Handle uses force sensors to measure grip and finger pressure, and detects shoulder abduction/flexion, elbow flexion/extension, forearm pronation/supination, wrist flexion/extension, and wrist radial/ulnar deviation. The Multiball is used for training of forearm pronation/supination and wrist flexion/extension. The Multiboard is used for bilateral rehabilitation of the arm. These elements can be used to measure hand strength and the active ROM, as well as to play eight interactive therapy games, which include catching falling objects, guiding a ball through a labyrinth, and setting a table. It is possible to select the level of difficulty of the game, as well as the motion or grip to be trained. The patient's history, results, and progress are recorded in a report [43]. The use of the product in combination with conventional neurorehabilitation has proven to help increasing functional hand recovery in multiple sclerosis patients [44].

Tyromotion has also developed AMADEO (Figure 3.2(b)), an end-effector robotic device for hand rehabilitation with 5 degrees of freedom [47]. It has assistive, passive, and active modes, and

(a) Pablo Upper Extremity [45]

(b) AMADEO [46]

Figure 3.2: Rehabilitation systems by Tyromotion

includes electromyography (EMG) sensors, allowing for the rehabilitation of people with different levels of strength and mobility. It also incorporates the same rehabilitation games as Pablo [48].

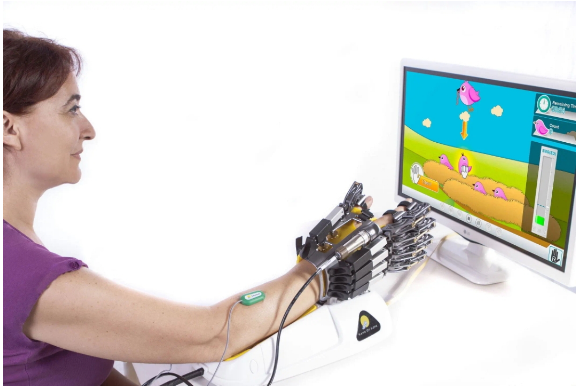


Figure 3.3: Hand of Hope [5]

The Hand of Hope (Figure 3.3) is a robotic device for neuromuscular rehabilitation of the hand and forearm. Two surface EMG sensors are attached to the extensor and flexor muscles of the forearm, and detect voluntary EMG signals, which indicate the intention to move. These signals are processed and sent to the hand brace, which supports the movement of the hand. This positive biofeedback encourages motor relearning [49]. There are several training modes, with adjustable difficulty levels. The brace can cause repetitive movements, without effort from the patient, or it can assist the patient in completing the movement if the EMG signal reaches or maintains a threshold. The movements trained are opening the hand, grasping, or both, as well as extension of the elbow and shoulder, which is tracked by the forearm support [50]. The system's interactive games are controlled by different hand and arm movements, depending on the level of difficulty selected, and include opening flowers, placing balls in baskets and using a claw machine. The settings and training information of each patient are stored in a database, and graphical reports can be generated [49].

Gloreha has developed several devices for upper limb rehabilitation, three of which are based on serious games. Crescendo (Figure 3.4(a)) offers motor rehabilitation of the hand, wrist, and arm, as well as neurocognitive treatment. It includes passive and assisted modes, and action-observation therapy, in which the movement is first showed on the screen and must then be performed by the patient. The games were developed in collaboration with a team of neuropsychologists to promote motor control and coordination, attention, memory, and problem solving. The system makes available reports of patients' performance and wrist ROM [51].

Sinfonia (Figure 3.4(b)) is a similar system. It has passive, assisted, and active modes, action-observation therapy, and bimanual mirror training (in which the movements of the healthy limb are reproduced on the injured limb). The second glove can also be worn by the therapist, who is then able to control the patient's glove. The exercises, in virtual games and with real objects, train closing and opening of the hand, flexion and extension of individual fingers, and tridigital pinch [52]. The software records and writes reports on the patient's active and passive ROM, movement speed, coordination, and the performance evolution by exercise and by session [54].

(a) Crescendo [51]

(b) Sinfonia [52]

(c) Aria [53]

Figure 3.4: Rehabilitation systems by Gloreha

Aria (Figure 3.4(c)) integrates neurocognitive and motor rehabilitation. Unlike the previously mentioned systems, it uses a vision sensor, rather than a glove, to monitor the patient's upper limb movement. The movements detected are wrist pronation/supination, wrist flexion/extension, wrist radial/ulnar deviation, finger flexion/extension, and arm movements in vertical and horizontal planes. In the interactive games, the patient makes a character execute various tasks, solves quizzes, or plays card games [55].

HandTutor (Figure 3.5) is a physical and occupational therapy system. It consists of a glove, with sensors that detect the flexion and extension of the wrist and fingers. The exercises can be customized to the training of wrist and/or finger(s), as well as to the patients abilities and ROM [56]. The games include guiding a ball through a track, fishing, and some know video games, such as helix jump and bubble shooter. After the game is over, the patient receives feedback on their performance. The software also keeps records of the treatment plan, session logs, and performance of each patient [57]. The system can be combined with functional electrical stimulation, allowing for assisted movement [58].

The Supervised Care and Rehabilitation Involving Personal Tele-robotics (SCRIPT, Figure 3.6) Project developed a prototype for a rehabilitation device for at home hand and wrist rehabilitation after stroke. An orthosis supports movement by offsetting the involuntary flexion torques caused by stroke, which hinder hand movement. The arm position is determined by IMUs, and the flexion of wrist and fingers is measured by a potentiometer and bending sensors, respectively. These sensors detect forearm pronation/supination, wrist flexion/extension, finger flexion/extension, and

Figure 3.5: Handtutor [6]

antero-posterior and lateral displacement of the hand.

SCRIPT has two interfaces: one for the healthcare professional and one for the patient. The healthcare professional can see an overview of the current condition of all the patients, exchange messages with them, analyze their progress, and assign them exercises. The patient can see their progress, exchange messages, and chose between the exercises assigned by the therapist. The exercises consist of games: moving a ball through a labyrinth, moving a character while avoiding obstacles, and open/close a sea shell to catch fish. The system can be calibrated to adjust to the patient's ROM and movement speed, and these values are updated considering the patient's evolution [59].

Figure 3.6: Supervised Care and Rehabilitation Involving Personal Tele-robotics (SCRIPT) [7]

The ReoGo (Figure 3.7) is a robotic system for upper limb therapy, composed of a motorized robotic arm and a screen which shows rehabilitation exercises. These exercises encourage movements in two and three dimensions, that can be made with different kinds of support (forearm, wrist, or hand), and in various levels, from entirely active to entirely passive [60]. The repetitive training, with audio and visual biofeedback, has the purpose of reducing muscle spacity and improving strength, ROM and accuracy of movement. To adapt the training to different patients, the existing exercises can be adjusted, and new ones can be designed [8].

Figure 3.7: ReoGo [8]

MusicGlove (Figure 3.8) is a finger rehabilitation device based on music-related computer games. It is composed of a glove, with six electrical leads, five on the fingertips and one on the side of the index finger. This glove is used to play a game inspired by Guitar Hero, where the notes are played by touching the thumb fingertip to each of the other five electrical leads. The performance evaluation is based on the number of correct notes hit, as well as how close they were to the correct time. The game contains various songs, with different difficulty levels, for different levels of hand impairment [61] [62].

Figure 3.8: MusicGlove [9]

The Syrebo hand rehabilitation system (Figure 3.9) is a soft robotic exoskeleton for single finger rehabilitation. It allows different types of training: active, passive, or assisted, mirror training, and resistance training, as well as interactive games. It is also possible to select which finger or fingers to train, and the intensity and speed of the exercise. During the training, the patient sees a tridimensional model of their hand on a screen and can interact with some real objects [63].

Figure 3.9: Syrebo [10]

GripAble (Figure 3.10) is a game-based hand rehabilitation system, composed of a handheld device which connects to an app on a tablet. The device uses motion and force sensors to detect wrist pronation/supination, wrist flexion/extension, wrist radial/ulnar deviation, and grip. It tracks the patient's activity and provides games designed to tackle different therapeutic goals, with real time feedback on the performance. It is available in two platforms: GripAble Home, for at-home patient training, and GripAble Pro, for therapists. The therapist can use their app to assign each patient a personalized therapy program and see their activity reports. The games can be calibrated to the patient strength and range of motion [64].

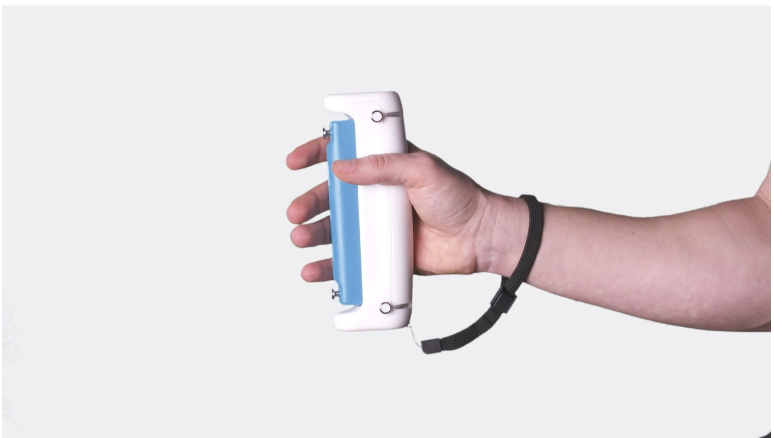


Figure 3.10: GripAble [11]

AbleX (Figure 3.11) is an upper limb rehabilitation system for patients with injuries caused by stroke, brain trauma, cerebral palsy, multiple sclerosis, or dementia. It includes two controllers, which engage different parts of the arm and/or hand: the armskate (a computer mouse, and a tray which supports the lower arm and constrains wrist movement) and the handlebar [65]. There are also eight therapy games designed to stimulate motor and cognitive skills. The system is meant to be used by the patient at home, while giving the therapist real time information on their progress [12].

Figure 3.11: AbleX [12]

Bimeo (Figure 3.12) is a system for arm and hand rehabilitation. It offers over 20 therapy exercises in the form of video games and ADLs, as well as assessment of various metrics of the patient's condition. The several devices can be used in different combinations for different types of therapy: with or without support, unilateral, bilateral, or of specific joints. Therapy settings can also be easily adapted to the patient. IMUs placed in the arm and a force sensor in a handheld spherical controller [66] measure, besides arm movements, wrist pronation/supination, wrist ulnar/radial deviation, and wrist flexion/extension [67].

Figure 3.12: Bimeo [13]

The Motus Hand (Figure 3.13) is a rehabilitation robot designed to be used by the patients at home, independently. It trains wrist flexion and extension, with the purpose of improving flexibility, grip strength, motor skills and coordination in stroke survivors [14]. A pneumatic artificial muscle is used to assist the movement. In the gamified exercises, the patient must hold the wrist in a determined position, flex it, or extend it, unsupported, with assistance, or against resistance [68]. An artificial intelligence algorithm adapts the levels of assistance and resistance to the patient's current situation, based on their progress. After the exercise, the patient can receive a report on their strength and range of motion.

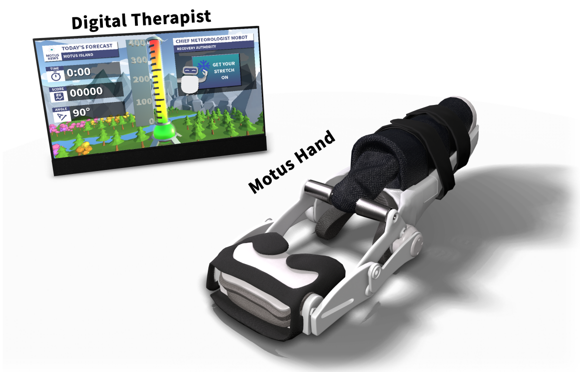


Figure 3.13: Motus Hand [14]

The SaeboReJoyce (Figure 3.14) is a worksation for upper extremity rehabilitatiom. It provides gamified exercises representing ADLs, which improve speed, endurance, coordination, range of motion, strength, and gross and fine motor skills, and can be customized to each patient. The Arm Hand Function Test assesses the patient's situation, tracks their improvement, and creates progress reports. The controller allows for the training of various movements, including wrist pronation/supination, wrist flexion/extension, and several specific grip and pinch patterns [69].

Figure 3.14: SaeboReJoyce [15]

3.2 Comparison of the features of the rehabilitation systems

Table 3.1 shows a summary of the hand and wrist rehabilitation systems described in section 3.1. The systems can be classified in several ways. Some of them are: the hardware used for interface with the system, the part(s) of the body they train, whether they provide movement assistance, if they allow for passive and active-assisted training, the sensors they use, if they are meant to be used by a healthcare professional or autonomously by the patient, if they can be adapted to patients with different types and levels of disability, if they give feedback on the patient's progress, and their cost.

Of the seventeen systems found, fourteen rehabilitate the fingers, eleven the wrist, and eight both. For finger rehabilitation, most of the systems train their flexion/extension, but a few also do abduction/adduction, opposition/reposition, grip force, and finger pressure. These are measured by bending sensors, force sensors, and EMG sensors. For wrist rehabilitation, the movements trained are flexion/extension, pronation/supination, and radial/ulnar deviation. In general, a wrist rehabilitation system will train for all three, but the latter two are sometimes excluded. These movements are measured, mostly, by IMUs. The sensors used to track the patient's movements are placed mostly in orthoses, as well as gloves, and other handheld devices. Others can be quite hard to move and have a complex setup, making their use considerably more complicated. About half of the systems can assist the user in performing the movements, which is especially useful for patients with very limited strength and range of motion. Six of the systems were developed specifically to be used in a clinical setting, one is intended for at home use and two have separate units meant for patient and therapist use. The rest can be used at home or in a clinic. All of them can be adapted to the patient's level of disability, and all but one give the user progress reports. The prices vary from $300 to $35,000, although most are on the higher end of the range.

There are some limitations to these systems. Some of the hardware is bulky and heavy, limiting its use. They can also be hard to use and become uncomfortable to the user. They are, in general, quite expensive, and, for most of them, the price is considerably above what the average patient is able to afford.

Table 3.1: Hand and wrist rehabilitation systems

System	Hardware	Target member(s)	Movement assistance	Sensors	Portability	Adaptability	Progress feedback	Price
Neofect Smart Glove [40] [41] [42]	glove	fingers and wrist	no	IMU and bending sensors	home or clinic	yes	yes	$15,000 or $99/mo
PABLO® Upper Extremity [43] [44]	handle, ball, board	fingers, wrist, and arm	no	force sensors, IMUs	home or clinic	yes	yes	$10,000
AMADEO [47] [48]	orthosis	fingers	yes	EMG	clinic	yes	yes	$35,000
Hand of Hope [49] [50]	orthosis	fingers and arm	yes	EMG	clinic	yes	yes	$20,000
Crescendo [51]	orthosis	fingers and wrist	yes	-	clinic	yes	yes	-
Sinfonia [52] [54]	orthosis	fingers	yes	-	clinic	yes	yes	-
Aria [55]	arm support	fingers, wrist, and arm	no	motion sensor	clinic	yes	yes	-
HandTutor [56] [57] [58]	glove	fingers and wrist	no	optical position and speed sensors	home or clinic	yes	yes	$20,000
SCRIPT [59]	orthosis	fingers, wrist, and arm	yes	IMUs, potentiometer, bending sensors	home + clinic	yes	yes	-
ReoGo [60] [8]	orthosis	fingers and arm	yes	-	clinic	yes	yes	$10,000+
MusicGlove [61] [62]	glove	fingers	no	electrical leads	home or clinic	yes	yes	$359
Syrebo [63]	orthosis	fingers	yes	-	home or clinic	yes	no	$350-550
GripAble [64]	handheld device	fingers and wrist	no	motion sensors, force sensors	home + clinic	yes	yes	$250 - $499
AbleX [65] [12]	armskate, handlebar	wrist	no	computer mouse, CyWee Z controller	home or clinic	yes	yes	$1,000 - $1,999
Bimeo [66] [67]	sphere, armbands	wrist and arm	yes	IMUs, force sensor	home or clinic	yes	yes	-
Motus Hand [14] [68]	brace	wrist	yes	angle sensor	home	yes	yes	$399 /month
SaeboReJoyce [69]	controller	fingers and wrist	no	force and position sensors	home or clinic	yes	yes	$10,000 - $13,000

Chapter 4

Methodology

This chapter presents a proposal for the development of a system based on serious games for rehabilitation of the hand and wrist. Section 4.1 describes the general architecture of the system, section 4.2 details the hardware and software used on the data glove and the steps taken in its creation, and section 4.3 relates the development of the application.

4.1 System Architecture

HandRehab is a hand and wrist rehabilitation system. It is composed of a computer app and a data glove. It is designed to allow patients to use it independently, but can also be a helpful tool for therapists. The system can be seen if Figure 4.1.

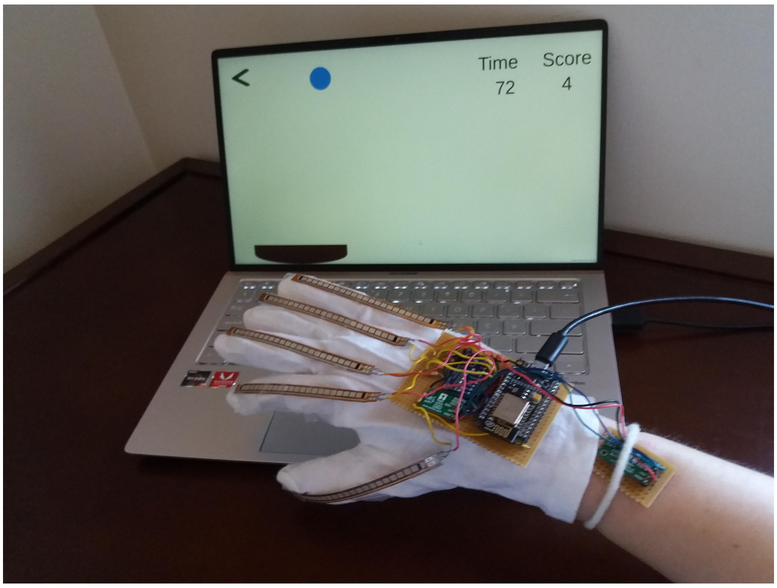


Figure 4.1: HandRehab system

Figure 4.2 shows an architecture diagram of the system. Five flex sensors, in the fingers, send resistance values to the microcontroller, which uses uses them to calculate the angles of the fingers.

The two IMUs send to the microcontroller the acceleration, rotation, and magnetic field values, from which the orientation of the hand is obtained. The values calculated by the microcontroller are then transmitted to the app through Wi-Fi.

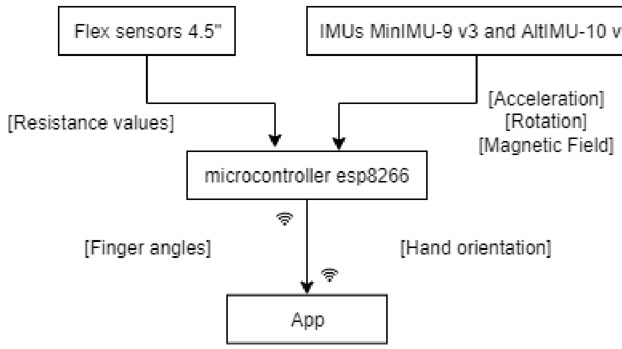


Figure 4.2: Architecture diagram of the system

The app includes two games. In the first one, the player is instructed to extend different numbers or combinations of fingers or open/close the hand. In the second game, balls fall from the top of the screen, and the player must move a basket along the bottom of the screen, using wrist or finger movements, to catch some of them, and avoid others. At the end of each game, the player receives feedback on their performance. It is also possible to review the evolution of the game scores over time. To allow patients with different kinds and degrees of disability to use the app, the games are available in different difficulty levels, and can be adapted to the patient's ROM.

4.2 Data glove

The data glove contains five 4.5" flex sensors, a MinIMU-9 v3 and an AltIMU-10 v5 IMU, an ADS1115 analog-to-digital converter (ADC) and an ESP8266 microcontroller.

Figure 4.3: ESP8266 microcontroller [16]

(a) ADS1115 [72]

(b) Flex sensor [73]

Figure 4.4: Devices used in the data glove

The ESP8266 (Figure 4.3) is a Wi-Fi microcontroller produced by Espressif Systems for mobile devices, wearable electronics and internet of things (IoT) applications. It can be programmed in several ways, including through the Arduino IDE. It has only one ADC pin, making necessary the use of an additional ADC [70].

The ADS1115 (Figure 4.19(a)) is a 16-bit ADC with four input channels. It interfaces via Inter-Integrated Circuit (I2C) and includes a programmable gain amplifier. The I2C address can be changed to allow the use of up to four devices at the same time [71].

Flex sensors, or bend sensors (Figure 5.3), measure the amount of bending. When the sensor is bent its resistance varies. They are often used as goniometers in data gloves [74].

An IMU is a device composed of an accelerometer, a gyroscope and a magnetometer, which measure acceleration, orientation, and magnetic field, respectively. The Pololu AltIMU-10 v5 (Figure 5.4(a)) is an IMU and altimeter which uses the LSM6DS33 3-axis gyroscope and 3-axis accelerometer, the LIS3MDL 3-axis magnetometer, and the LPS25H digital barometer [75]. The Pololu MinIMU-9 v3 (Figure 5.4(c)) combines a L3GD20H 3-axis gyro and an LSM303D 3-axis accelerometer and 3-axis magnetometer. In both cases the sensors share an I2C interface, with alterable addresses [76].

(a) Pololu AltIMU-10 v5 [75]

(b) Pololu MinIMU-9 v3 [76]

Figure 4.5: IMUs used in the data glove

A schematic of the circuit can be seen in Figure 4.6. Each of the flex sensors is connected to a 10k resistor, creating a voltage divider. Four of the sensors are connected to the ADS1115, while the fifth connects directly to the ESP8266 ADC. The ADS1115 and the two IMUs are connected to the ESP8266's SCL and SDA pins, communicationg through I2C. In the Altimu, the SA0 pin is connected to the ground, which changes the I2C address, allowing the two IMUs to be read simultaneously.

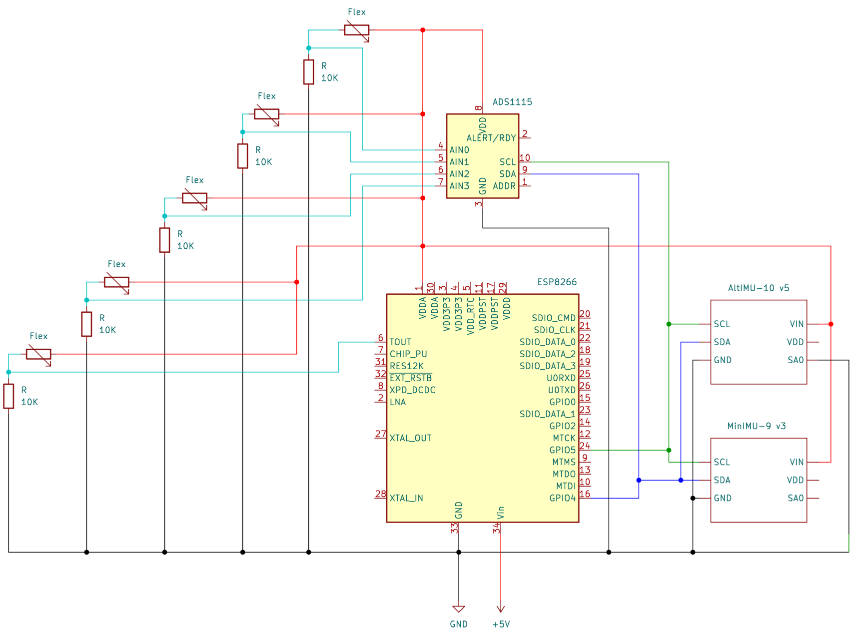


Figure 4.6: Glove electronic circuit

The Arduino IDE was used to program the ESP8266. The IMU data processing was based on an attitude and heading reference system (AHRS) program [77], which uses a Direction Cosine Matrix (DCM) algorithm [78]. A DCM is a matrix that represents the position of an orthogonal frame in relation to a reference frame, and is a very common way of representing rotations [79]. The algorithm returns the position in the form of euler angles: roll (ψ), pitch (θ) and yaw (ϕ).

The Arduino code begins by read the IMU values. Four libraries are used: The LSM6 library for the LSM6DS33 gyroscope and accelerometer, the LIS3MDL library for the LIS3MDL magnetometer, the L3G library for the L3GD20H gyroscope, and the LSM303 library for the LSM303D accelerometer and magnetometer. In the calibration function, some readings are taken and used to calculate the offset. In the loop function, the gyroscopes and the accelerometers are read at 50 Hz, and the magnetometers at 10 Hz. The magnetometer values are calibrated, to adjust to the specific offset values of each board. Then the tilt-compensated values of the magnetic field are calculated, using equation 4.1, in which mag_x, mag_y, and mag_z are the magnetometer values for each of the three axis; and the heading can be obtained through equation 4.2.

$$\begin{aligned} Cmag_x &= mag_x cos(\theta) + mag_y sin(\psi) sin(\theta) + mag_z cos(\psi) sin(\theta) \\ Cmag_y &= mag_y cos(\psi) - mag_z sin(\psi) \end{aligned} \tag{4.1}$$

$$Heading = Atan\left(\frac{Cmag_y}{Cmag_x}\right) \tag{4.2}$$

The gyroscope data is corrected, by adding the proportional and integral compensation values calculated in the previous iteration, and is then used to update the DCM according to equation 4.3, in which ω is the rotation rate vector, R is the DCM, t is the current time and dt is the time between iterations.

$$R(t+dt) = R(t) \begin{bmatrix} 1 & -\omega_z dt & \omega_y dt \\ \omega_z dt & 1 & -\omega_x dt \\ -\omega_y dt & \omega_x dt & 1 \end{bmatrix} \tag{4.3}$$

The following step is the normalization, in which the orthogonality of the DCM is enforced, as shown in equations 4.4 and 4.5.

$$\begin{aligned} X_{ort} &= X - \frac{error}{2} Y & Y_{ort} &= Y - \frac{error}{2} X & Z_{ort} &= X_{ort} \times Y_{ort} \\ X &= \begin{bmatrix} r_{xx} \\ r_{xy} \\ r_{xz} \end{bmatrix} & Y &= \begin{bmatrix} r_{yx} \\ r_{yy} \\ r_{yz} \end{bmatrix} & Z &= \begin{bmatrix} r_{zx} \\ r_{zy} \\ r_{zz} \end{bmatrix} & error &= X \cdot Y \end{aligned} \tag{4.4}$$

$$\begin{aligned} X_{norm} &= \frac{1}{2}(3 - X_{ort} \cdot X_{ort}) X_{ort} \\ Y_{norm} &= \frac{1}{2}(3 - Y_{ort} \cdot Y_{ort}) Y_{ort} \\ Z_{norm} &= \frac{1}{2}(3 - Z_{ort} \cdot Z_{ort}) Z_{ort} \end{aligned} \tag{4.5}$$

It is then necessary to correct the drift, by using orientation reference vectors to detect orientation error. The correction for the yaw is obtained by equation 4.6, while the correction for the roll and pitch is calculated using equation 4.7, in which A is the accelerometer data.

$$Yaw_{corr} = (r_{xx} sin(Heading) - r_{yx} cos(Heading)) Z \tag{4.6}$$

$$RollPitch_{corr} = Z \times \left(A + \omega_{gyro} \times \begin{bmatrix} velocity \\ 0 \\ 0 \end{bmatrix} \right) \tag{4.7}$$

The total correction, the weighted sum of the two corrections, is passed through a proportional integral (PI) controller (equation 4.8, in which K_P is the proportional gain and K_I is the integral gain) to compute the correction rotation rates. These values are then added to the rotation rate in the next iteration.

$$\begin{aligned} \omega_{Pcorr} &= K_P TotalCorr \\ \omega_{Icorr} &= \omega_{Icorr} + K_I dt TotalCorr \\ \omega_{corr} &= \omega_{Pcorr} + \omega_{Icorr} \end{aligned} \tag{4.8}$$

Finally, the DCM is used to calculate the Euler angles, as shown in equation 4.9.

$$\begin{aligned} \theta &= -arcsin(R_{31}) \\ \psi &= arctan\left(\frac{R_{31}}{R_{33}}\right) \\ \phi &= arctan\left(\frac{R_{21}}{R_{11}}\right) \end{aligned} \tag{4.9}$$

Flex sensor data is also read at 50 Hz, using the Adafruit ADS1X15 to read the ADS1115. Equation 4.10, in which Vcc is the voltage in the system and Vo is the output voltage, is used to obtain the resistance of a resistor, in this case a flex sensor, in a voltage divider.

$$R_f = R\left(\frac{V_{cc}}{V_o} - 1\right) \tag{4.10}$$

The resistance value is remapped to a range of 0 to 180, becoming the flexion angle of the sensor. The angle of forearm pronation is the average of the roll for both sensors, and the angle of wrist flexion is the difference between the pitch of the first sensor (placed in the forearm) and the second (placed in the back of the hand). These values are then sent to the computer app using the Uduino plugin.

4.3 Application

The computer application was developed using the Unity game engine. Unity uses C# as its main programming language, and allows the creation of games in 2D and 3D for several platforms, including Windows, Mac, Android, iOS, PlayStation, and virtual reality platforms. It is also used in other fields, such as films and simulations. Several factors contributed to the choice of using Unity in this project, including the fact that it is free of charge, can be easily used by people without much game development experience, allows the same game to be deployed on multiple platforms, and has a wide range of assets available in the Asset Store. Communication between the ESP8266 and the app is accomplished through Uduino, a plugin for Unity-Arduino interaction that also works with ESP boards [80].

4.3.1 Activity diagram

Figure 4.7 shows an activity diagram of the application. When opening the application, the user finds a menu with 4 options: “Calibrate”, “Fingers game”, “Ball catcher”, and “Results”. If the user chooses “Calibrate”, they can select which of the three movements they want to adjust. The app will show a message instructing them to perform the movement, read the data for a few seconds, and calculate the ROM. After this, the message "Done" is displayed, and the patient may repeat the process for another movement or return to the main menu.

Clicking “Fingers game” opens a screen in which it is possible to select the number of repetitions to perform, the time given to perform each repetition, and the level of difficulty. After this, the user can press the “Play game” button, and the game begins. An image or number indicates the hand position. If the player performs it in the time previously chosen, a “Well done” message is displayed and the score increases. It the player fails, a “Keep trying” message is shown instead. If the selected number of repetitions has not yet been reached, a new hand position is shown. If it has, then the final score is presented, along with an appropriate message. The user may decide to play again, in which case they are again asked to define their preferences for the game, or to return to the main menu.

If “Ball catcher” is selected, the user can select the movement the want to train, the duration of the game, and its speed, after which they press “Play game”. The game shows balls falling from the top of the screen, and the player performs the movement, which is used to change the position of the game basket. If a ball is caught the score is changed: increased if the ball is blue, and decreased if it is red. Once the defined time is over the final score is shown, and the user can return to the second game menu or to the main menu.

By pressing “Results”, the user is taken to a screen where they may select which type of result they want to see. The chosen results are displayed, and the user may at any time use the back button to see again the main menu.

4.3.2 App design

Mockups for the app design were created. The first screen (Figure 4.8) is the main menu, with four buttons. The first button will lead to the "Calibration" screen (Figure 4.9), which contains a group of three radio buttons, labeled for the three hand and wrist movements, an image demonstrating the selected movement, and a text message that gives the user instructions and information.

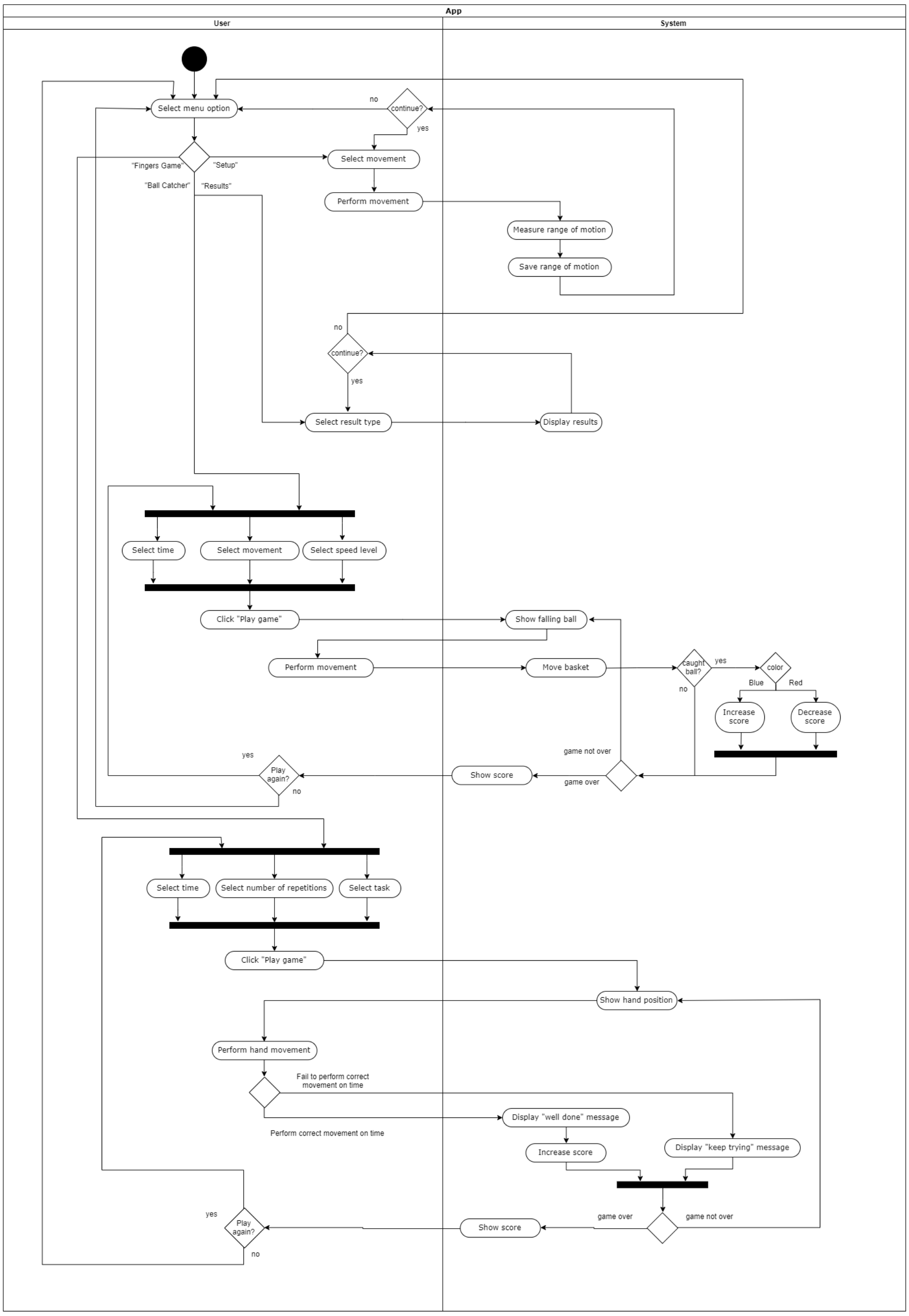


Figure 4.7: Activity diagram of the application

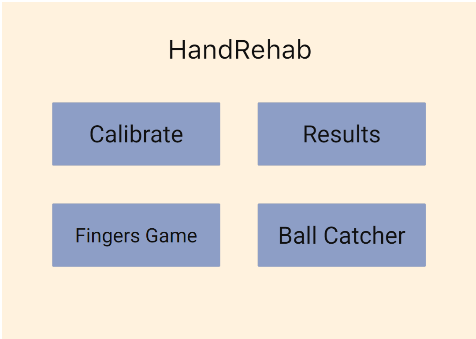


Figure 4.8: Main menu screen

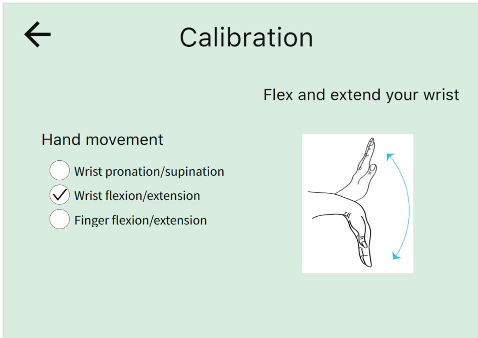


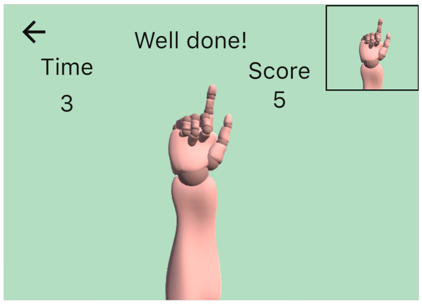
Figure 4.9: Calibration screen

The second button of the menu "Fingers Game", opens the "Fingers Game options" screen (Figure 4.10(a)), with radio buttons for selection between three tasks, and two text fields to input the number of repetitions and the time to perform each of them. There is also a "Play game" button, which opens the "Fingers Game" screen (Figure 4.10(b)). This screen includes a 3D hand model, which repeats the user's finger movement, an image indicating the desired position, an indicator for the remaining time, another for the score, and one that indicates if the movement was correctly performed on time. Once the game ends, the "Fingers Game over" screen (Figure 4.10(c)) is opened, showing the score, a message which varies with the performance, and two buttons: "Play again", and "Main Menu".

The third button of the menu is "Ball Catcher", and it leads to the "Ball Catcher options" screen (Figure 4.11(a)), in which there are two groups of three radio buttons, one for the movements and one for the speed levels, as well as a text field for the total game time, and a "Play game" button. Clicking this, the "Ball Catcher" screen (Figure 4.11(b)) is opened, and it is possible to see blue and red balls falling from the top of the screen, a basket to catch them with, and indicators of the remaining time and the score. When the time is over, the "Ball Catcher over" screen (Figure 4.11(c)) is shown. This screen is similar to that of the previous game, showing the score, an appropriate message, and the "Play again" and "Main Menu" buttons.

The last button of the menu takes the user to the "Results" screen (Figure 4.12), where there are radio buttons for the three types of results "Fingers Game", "Ball Catcher", and "Range of motion". When an option is selected, three graphs are created, separating the results by task for

(a) Fingers Game options screen

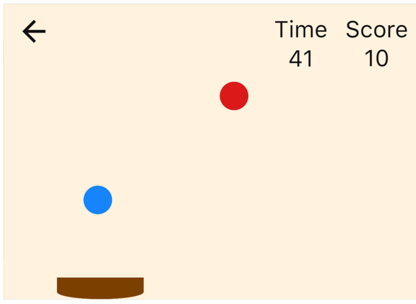


(b) Fingers Game screen

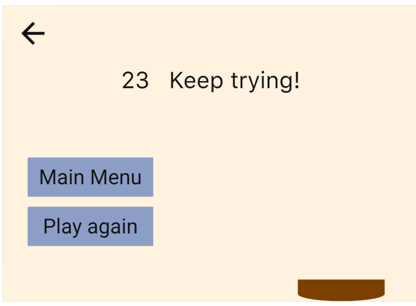
(c) Fingers Game over screen

Figure 4.10: Fingers Game screens

(a) Ball Catcher options screen



(b) Ball Catcher screen



(c) Ball Catcher over screen

Figure 4.11: Ball Catcher screens

Fingers Game, and by movement for Ball Catcher and ROM. All screens except the first include a "Back" button, that opens the previous screen.

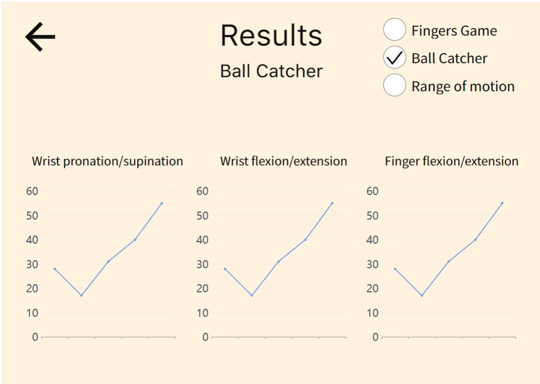


Figure 4.12: Results screen

4.3.3 App development

The game has nine scenes: "Menu", "FingersGameOptions", "FingersGame", "FingersGameOver", "BallCatcherOptions", "BallCatcher", "BallCatcherOver", "Calibration", and "Results". In the "Menu" scene (Figure 4.13), clicking the four buttons opens four scenes:"FingersGameOptions", "BallCatcherOptions", "Calibration", and "Results". The scene transitions are accomplished by adding an onClick Event to the button and using the ChangeScene.MoveToScene method.

Figure 4.13: Menu scene

In the "Calibration" scene (Figure 4.14), the type of ROM to adjust is selected using toggles. The toggles are placed in a toggle group, so that only one of them can be switched on at a time. A script is written with a static variable and a method for each toggle. In the methods, if the toggle is switched on, its value is assigned to the variable. A message and an image instruct the user to perform the selected movement, and a coroutine is used to wait a few seconds while the movement is performed, after which the ROM is saved. A message indicates the process is completed, and the toggle is switched off. These methods are then assigned to onValueChanged events in each toggle. A method is created for reading the sensor data, and is assigned to an onDataReceived event in the Uduino object. In the method, the received string is split into separate variables for each movement. If a toggle is switched on, the static variable changes, and the maximums and minimums for the selected movements are calculated. The ROM data is saved in JSON files, along with the current day, month, and year, which are obtained with the DateTime.Now property. Each time data is saved, the file is read and its information is placed in a list. The new data is added to the list, which is again converted to JSON format and saved in the file. This data will later be used to adapt the games to the user's ROM.

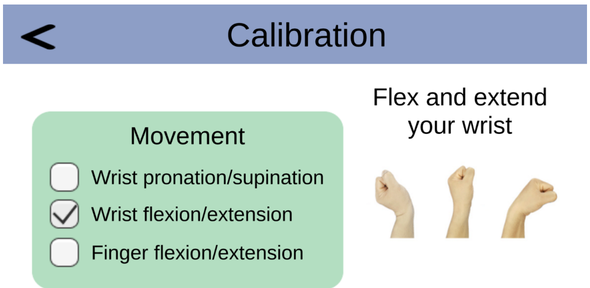


Figure 4.14: Calibration scene

The game options scenes, "FingersGameOptions" (Figure 4.15) and "BallCatcherOptions" (Figure 4.16), contain input fields, toggles, and buttons. In the first one there are two input fields, for the number of repetitions and the time for each repetition, and in the second there is only one, for the game time. In the input fields, the content type is set as "Integer Number", so no other data types are allowed. An onEndEdit event is created, using a method that stores the entered integer in a static variable, so it can later be accessed by the scripts used in the game. "Fingers-GameOptions" has a single toggle group, for task selection, with the options "Task 1", "Task 2", and "Task 3". "BallCatcherOptions" has a toggle group for speed level ("Level 1", "Level 2", or "Level 3"), and one to chose the movement, between "Wrist pronation/supination", "Wrist flexion/extension", and "Finger flexion/extension". The selected options are obtained and assigned to static variables, in the same way described for the "Calibration" scene. For the movement group of the second game options, the sprite of an image is also modified, so that it shows the selected movement. Both scenes have a "Play game" button, which uses an onClick Event and the ChangeScene.MoveToScene method to open scene "FingersGame" or "BallCatcher".

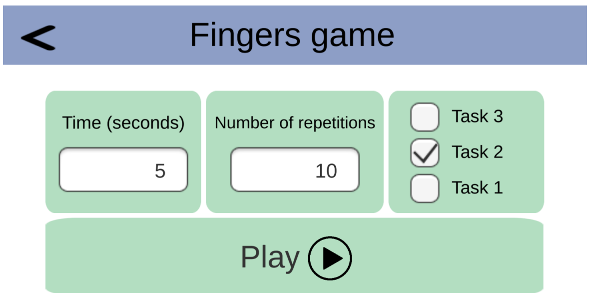


Figure 4.15: Fingers game options scene

The "Fingers Game" scene (Figure 4.17) contains a 3D hand model, which is used in the game. The hand model used is part of the Mixamo X Bot character [81]. Six animations were created for the character: an "idle" animation, in which the character is still, and an animation for the flexion

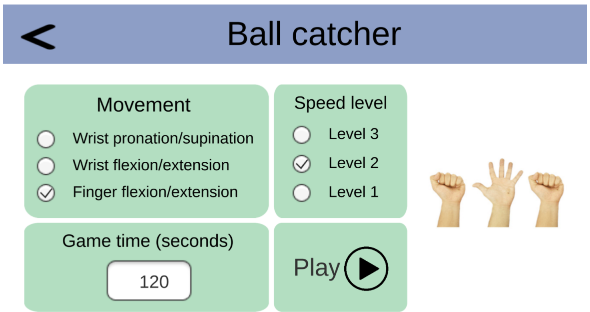


Figure 4.16: Ball catcher options scene

of each finger. In the animator, five layers are added, one for each finger. The blending types are set to "additive", and the weights to one. In each of the layers, there is a transition from the entry node to a blend tree. Each tree is controlled by a parameter representing the finger angle, which varies between 0 (fully extended finger) and 1 (fully flexed finger), and has two motion fields: the idle animation, and the animation created for that finger. When the parameter is 0, the idle animation has 100% of influence and the finger animation 0%, and the opposite happens when it equals 1. The sensor data is obtained in the same way it is done for the calibration, and the finger flexion values are assigned to the animator parameters through the Animator.SetFloat method.

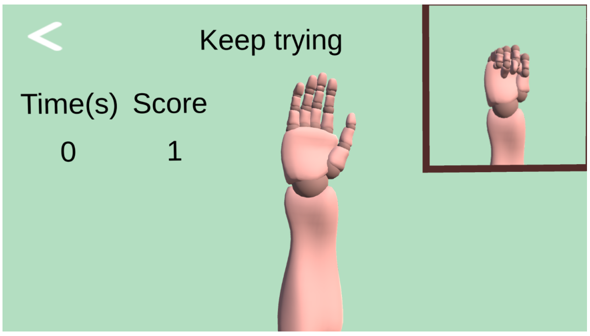


Figure 4.17: Fingers Game scene

The fingers ROM file is read and the values assigned to variables, and the game task is determined by reading the static variable defined in the options scene. The hand position the player must mimic is chosen at random from the positions defined for each task. If the task is 1, there are only two positions (closed hand and open hand), which alternate. In the second task, only the number of raised fingers is indicated, and the player may raise any set of fingers they wish. There are six positions, from no fingers to all fingers raised. For the third task, twelve positions were selected, which can be seen in figure 4.19. The current position is shown to the player through a number, for task two, or an image, for tasks one and three. Once the position is chosen, it is shown on the screen, and the timer is set to the time selected by the player. The Update method is used to decrease this time with each frame and to update its indicator on the screen. The finger angle values are obtained form the animator with the Animator.GetFloat method and it is verified if the current position matches the desired position. A finger is considered flexed or extended if it

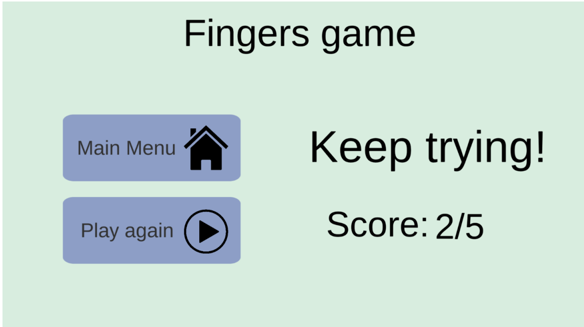
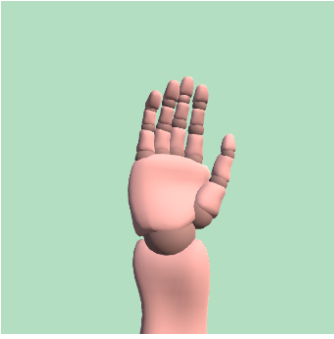


Figure 4.18: Fingers Game over scene

is within 10% of the player's maximum flexion or extension. For the second task, the number of extended fingers is calculated and compared. If the hand position is correct, or the time runs out, a coroutine is used to pause the game for two seconds. During this time, the score is increased, if the player was able to perform the movement, and one of two messages appears: "Well done" or "Keep trying", depending on the result. The score is saved in a static variable and shown in the screen during the game. After that time passes, it is verified if the chosen number of repetitions have been performed. If so, the game ends, and the SceneManager.LoadScene method is used to move to the next scene. If not, a new position is selected, and the timer is reset.

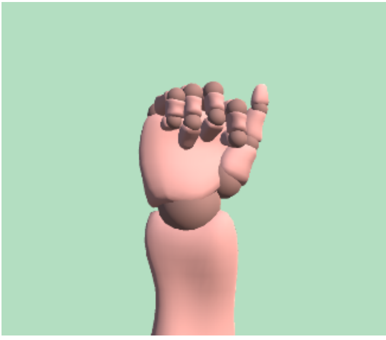
After the game ends, the "FingersGameOver" scene (Figure 4.18) is loaded. This scene displays on the screen the total number of repetitions and how many were correctly performed on time. A message also appears: "Perfect!" if all the repetitions were performed correctly, "Great job!" if 60% or more were , and "Keep trying!" otherwise. The result and the current date are saved in a JSON file, using similar methods to the ones used in the calibration.

In the "Ball Catcher" scene (Figure 4.20), the data obtained from the data glove is used to move a basket along the bottom of the screen, which must be used to catch some of the balls falling from the top of the screen. The selected movement, level, and time are obtained from the static variables mentioned before. The creation of the balls is done in a coroutine. When the scene is opened, a coroutine starts in which a message is displayed instructing the player in how to move the basket. The WaitForSeconds method is used to give the player a few seconds to read the text and familiarize themselves with the basket movement. The message then disappears and the game begins. A second coroutine is started, in which a GameObject is created with a ball sprite. The ball color (red or blue) is randomly selected. After an equally random amount of time, the coroutine starts again. This limits of this amount of time vary with the speed level selected by the player. An update function is used to count down from the game time, display the current time, and leave the game, with the SceneManager.LoadScene method, when it reaches 0. The glove data is read in a method which is assigned to an onDataReceived event in the Uduino object. The value used to move the basket is chosen depending on the movement selected by the player. If the movement is finger flexion/extension, the value used is the average of the five fingers. The data is adjusted according to the player's ROM, and to guarantee the basket doesn't leave the screen limits, and the basket position is defined. The onTriggerExit2D method is used to increase the score when a

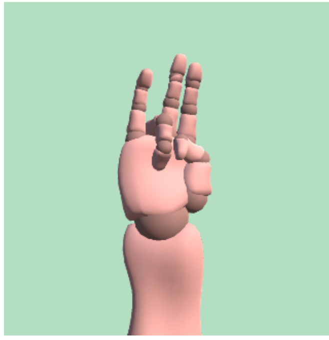

(a) All fingers up

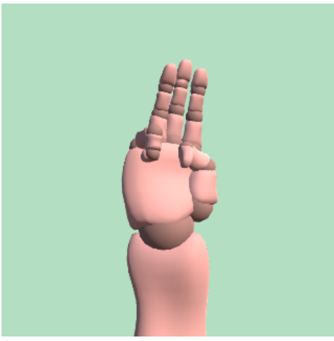

(b) Index and middle fingers up

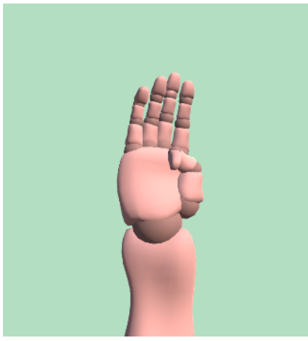

(c) All fingers down

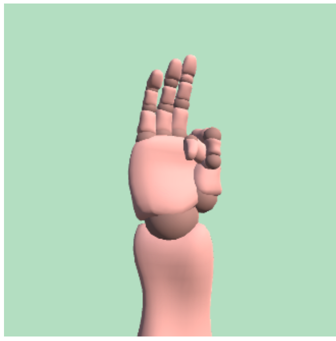

(d) Thumb up

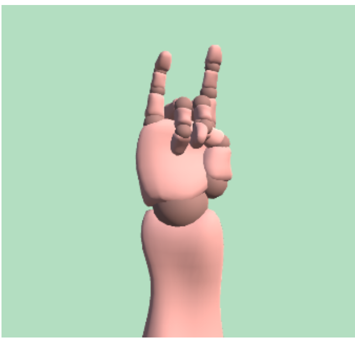

(e) Index finger and thumb up

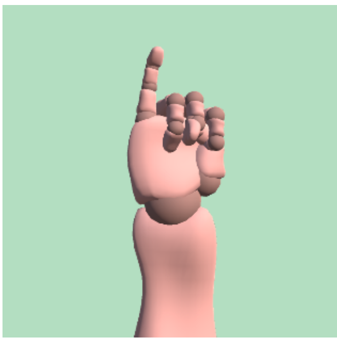

(f) Index, middle and little fingers up


(g) Index, middle, and ring fingers up


(h) Index, middle, ring and little fingers up


(i) Middle, ring, and little fingers up


(j) Index and little fingers up


(k) Little finger up


(l) Index finger up

Figure 4.19: Hand positions selected for task 3 of the first game

blue ball passes through the basket and decrease it when a red one does.

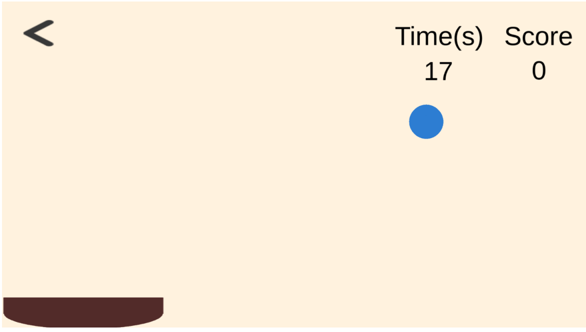


Figure 4.20: Ball Catcher scene

After the game is over, the "BallCatcherOver" scene (Figure 4.21) is opened. This script, like "FingersGameOver", shows the score on the screen, along with a message which varies with the player's performance. This performance is evaluated by the score divided by the number of blue balls in the game. The score and the current date are also saved to a JSON files.

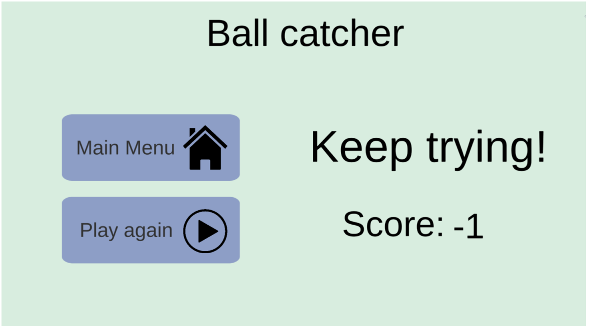


Figure 4.21: Ball Catcher over scene

The "Results" scene (Figure 4.22) allows the player to see their evolution. Three types of results are available: "Fingers Game", "Ball Catcher", and "Range of motion". These types can be selected from a toggle group. The methods attached to the onValueChanged events of the toggles display the result type on the top of the screen, read the data from the JSON files, and create three graphs. For the first game there is a graph for each task, and for the second game and the ROM, one for each type of movement. For each graph, one of the three GameObjects created for the graphs is found, the data taken from the files is used to calculate the average of the scores for each day, and a list is created with the values for the last 10 days. In the case of the ROM graphs, daily averages are not calculated, and the last ten results are shown instead. A method is then called which goes through this list and creates a new GameObject with a small circle image and with a y position proportional to the value in the list. Labels are also created for the x and y axis of the plot. A Line Renderer component is created for each variable that is plotted, and the coordinates of the points created are added to its array. The graph y axis minimum and maximum are defined as 0 and 180 for the ROM and are calculated based on the highest and lowest scores for the games.

For the plot that shows the fingers ROM, a legend is created which indicates the color in which the data for each finger is plotted.

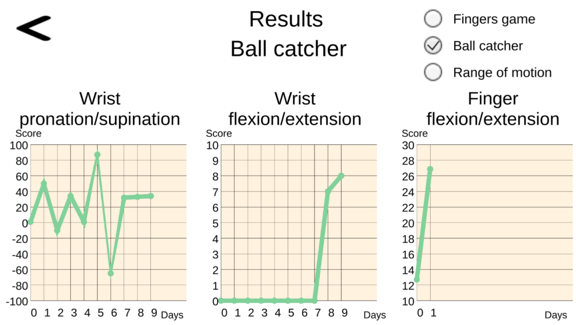


Figure 4.22: Results scene

Chapter 5

Results and discussion

This chapter describes the tests which were carried out in the system prototype validation, and their results. Section 5.1 evaluates the performance of the data glove, section 5.2 the two rehabilitation games, section 5.3 the calibration option, and section 5.4 the results function of the application.

5.1 Angle calculation test

The goal of the first test was to determine if the microcontroller was able to correctly read the sensor data and calculate the hand position values. Each of the movements used in the games (wrist flexion/extension, wrist pronation/supination, and fingers flexion/extension) was performed several times, and the resulting position angles analysed. Instead of the data being sent to the Unity app, the Arduino IDE Serial Plotter tool was used to visualize it.

The resulting plot for the fingers flexion/extension test is shown in figure 5.1. It was observed that the finger angles are approximately 0º when the user keeps their fingers extended, and increase when they are flexed, reaching values between 140º and 200º at the point of total flexion. When the fingers are closed there is a small peak, and the angle value then slowly decreases while they remain closed.

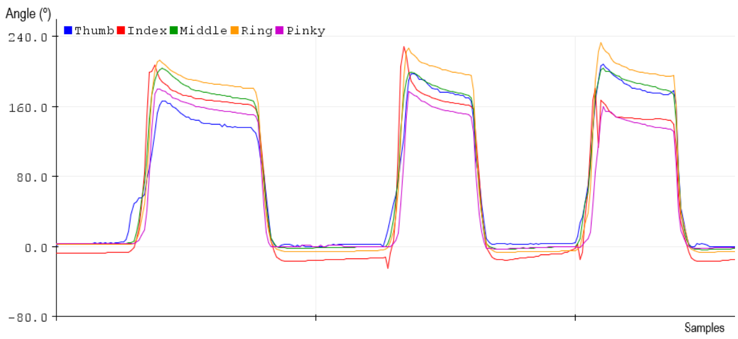


Figure 5.1: Angles test results for fingers flexion/extension

To test the forearm pronation/supination movement, the forearm is placed with the palm of the hand down, and is supinated and pronated. Analysing figure 5.2, it is seen that the angle values vary from 0º, when the forearm is fully pronated, to 180º, when it is fully supinated.

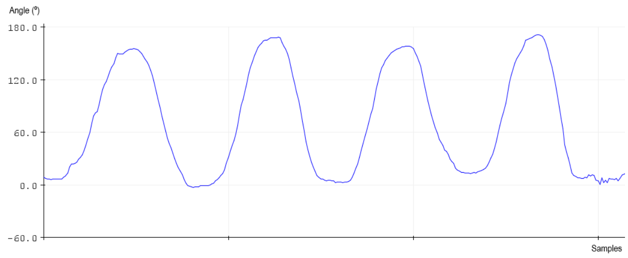
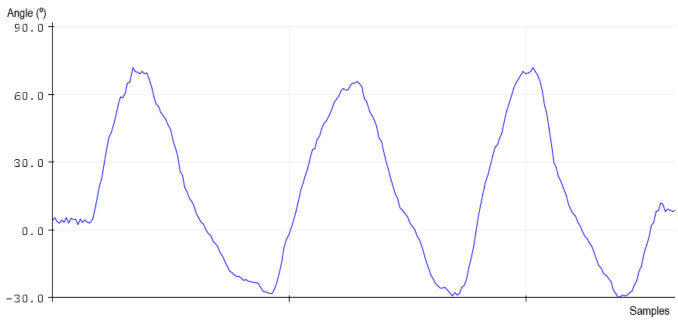
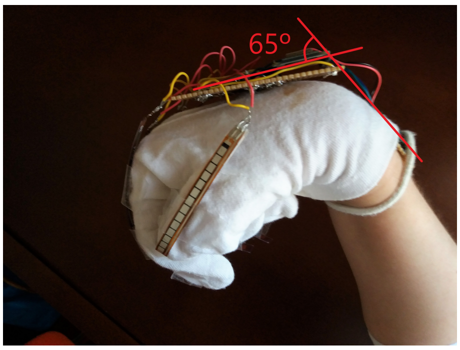


Figure 5.2: Angles test results for forearm pronation/supination

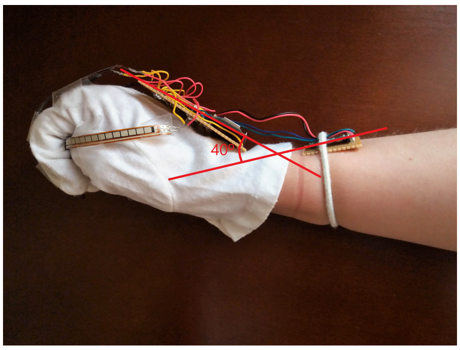
Finally, wrist flexion/extension was tested. Figure 5.3 displays the results. In this part of the test, the calculated angle was 0º when the wrist was straight. When it was fully flexed, its angle was approximately 65º, and the calculated angle was 75º; and when it was fully extended, the correct angle was -40º, and the calculated one was -30º.



(a) Plotted data



(b) Flexed wrist angle



(c) Extended wrist angle

Figure 5.3: Angles test results for wrist flexion/extension

It was, then, possible to verify that the angles obtained were close to those expected, and the data glove can be used to accurately determine the values necessary for the developed games.

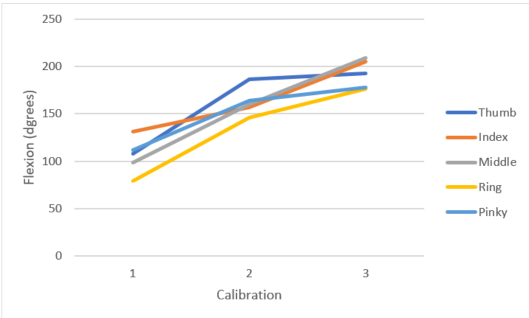
5.2 Games test

In the second test, the two rehabilitation games were evaluated. The games were played fifteen times, selecting different tasks Fingers Game, and different movements and speeds for Ball Catcher. It was verified that the data sent to the app from the data glove was correctly processed by the app, and the games behaved, in general, as expected. There was, however, an issue.

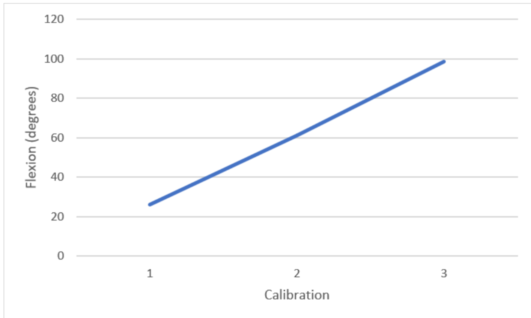
In the Fingers game, closed fingers were sometimes not considered to be fully closed. This is due to the small peak in the finger flexion angle calculations, which causes the maximum flexion angle obtained in the calibration to be superior to the average angle of fully flexed fingers.

5.3 Calibration test

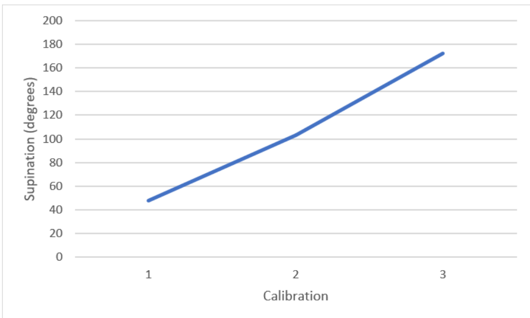
The purpose of this test was to evaluate the calibration function of the app. The calibration for the 3 different movements was performed several times, with increasing motion. The saved data was retrieved from the files and used to calculate the ROM values, which were plotted. The results are in Figure 5.4.



(a) Finger flexion/extension



(b) Wrist flexion/extension



(c) Forearm pronation/supination

Figure 5.4: Angles test results for the three movements

For each calibration, the two games were played. It was observed that the movement necessary to flex/extend the hand model fingers in Fingers Game, and to move the basket in Ball Catcher, increased over the test, proportionally to the saved ROM.

5.4 Results test

The last test assessed the developed results function.

Firstly, it was confirmed that the game results were being correctly recorded in the JSON files, by checking the file data against the results of the games played in the previous test. Next, the plotting was tested. Values were introduced in the files representing ROM data and game results saved over a period of time. The chosen values can be seen in Tables A.11, A.15, A.3, and A.7, and the graphs created by the app are shown in Figures 5.5, 5.6, and 5.7.

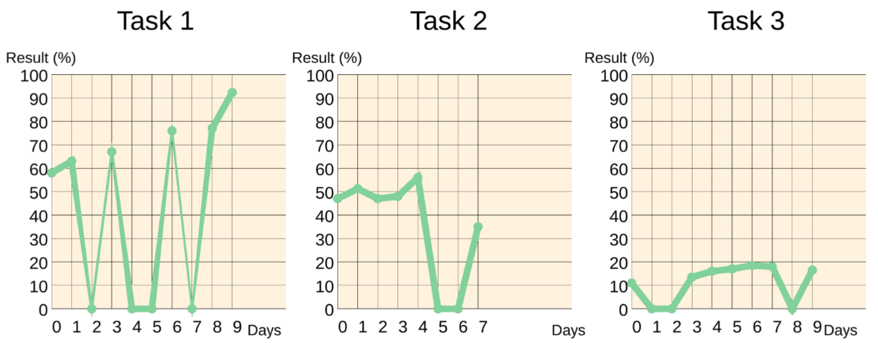


Figure 5.5: Graphs of the Fingers Game results created by the app

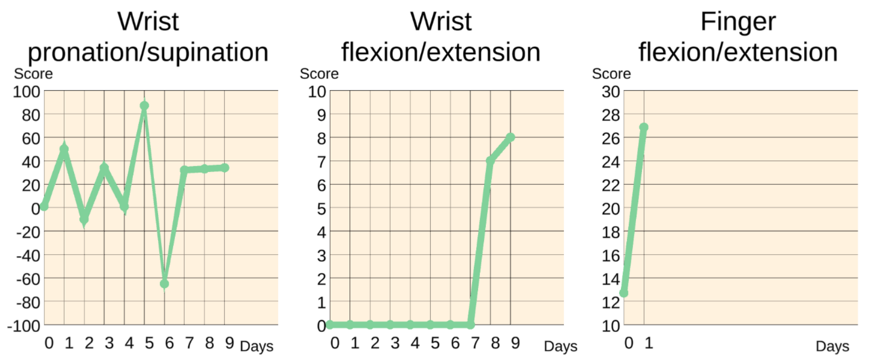


Figure 5.6: Graphs of the Ball Catcher results created by the app

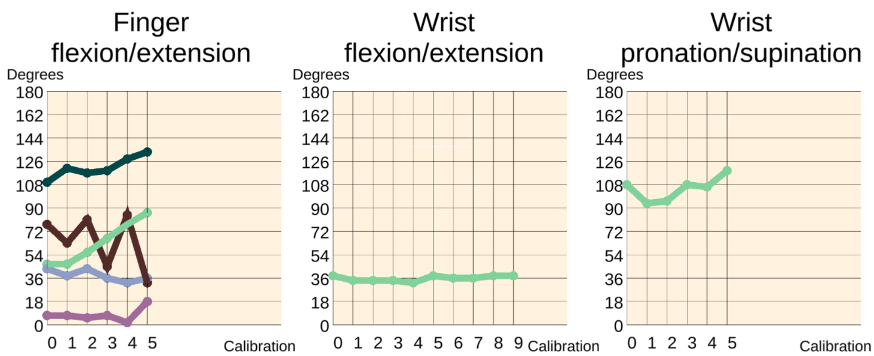


Figure 5.7: Graphs of the ROM results created by the app

By examining the graphs, it is possible to conclude that, for the ROM (Figure 5.7), the last up to ten calibration values are shown. For the two games (Figures 5.5 and 5.6), the values plotted are the average results of the last up to ten days, starting on the last day a game was played, and, in the days when the game was not played, the value is zero. The results function can, then, be considered functional.

Chapter 6

Conclusions and future work

The hand and the wrist are of the most used parts of the human body. This both makes them extremely important and leaves them very exposed to stress and injuries. Hand and wrist rehabilitation is, then, very important for the well being of many people. However, the rehabilitation process can easily become repetitive and tedious, and progress can be hard to perceive. Because of this, it is common that patients quickly lose motivation, start missing appointments, stop doing their recommended exercises, and even give up on the treatment entirely.

Recently, systems have been developed aiming to mitigate these problems, targeting the rehabilitation of several body parts, including hand and wrist. Some of these systems use serious games to increase patient engagement and enjoyment. Several functions are included, such as passive and active assisted training, adaptability to different disabilities, performance feedback, and progress reports. The systems can help patients manage their own rehabilitation, or assist therapists in their work.

This dissertation is a contribution to this field and had the purpose of creating a simple, affordable system for hand and wrist rehabilitation that helps patients stay motivated through their recovery. Initially, a data glove was created which uses IMUs and flex sensors to capture finger flexion/extension, wrist flexion/extension, and forearm pronation/supination, and sends the obtained data to the app. There was, then, the development of an app, which includes two serious games which integrate rehabilitation exercises and can be calibrated to adjust to the any patient's disability level, and allows the user to review their progress. The data glove and the app communicate through Wi-fi. The created system lets people with hand and/or wrist disability be more involved and active in their rehabilitation process.

After the development phase, the system was tested. In the first test, the sensor data reading and processing was evaluated, by performing the different movements the algorithm should be able to detect, and observing the results. After this, the two games were played several times, selecting different options, and their outcomes were examined. There was then an assessment of the calibration function, in which calibration for different ranges of motion was simulated, and the results were observed. Finally, the results function was appraised.

Although the development of this system can be considered successful, some aspects can still be improved. It was not possible to include wrist ulnar/radial deviation in the games, because the IMU algorithm output for the yaw angle, which corresponds to this movement, had too much noise for the small angle variations to be correctly read. Adding the movement to the system would make it more versatile, allowing for a more complete treatment of wrist impairment. A pausing option may be included in the games, allowing the patient to pause the training at any time they need without losing the progress in the game. The data glove can also be improved, for example by adding a battery, so it does not need to be connected to an external power source, allowing for more freedom of movement.

Finally, system validation should be performed with people who have hand and/or wrist impairment, to learn their opinion about it, and evaluate its effect on patient recovery.

References

[1] Cinnamon L. VanPutte, Jennifer L. Regan, Andrew F. Russo, Rod R. Seeley, Trent D. Stephens, and Philip Tate. *Seeley's Anatomy & Physiology: 12th edition*. McGraw-Hill, 2020.

[2] Doyoung Byun and Lee. Implementation of hand gesture recognition device applicable to smart watch based on flexible epidermal tactile sensor array. *Micromachines*, 10:692, 10 2019.

[3] Richard L Drake, Wayne Vogl, and Adam W M Mitchell. *Gray's Atlas of Anatomy*. Elsevier churchill lvgst, 2015.

[4] June Lee. Neofect Smart Glove Helps Congenital Stroke Survivor Janet. Available at https://www.neofect.com/us/blog/neofect-smart-glove-helps-congenital-stroke-survivor-janet, May 2019.

[5] MedicalExpo. HAND OF HOPE - Hand rehabilitation system by Rehab-Robotics. Available at https://www.medicalexpo.com/prod/rehab-robotics/product-77946-472601.html.

[6] Handtutor™. Available at https://constancecare.pl/en/equipment/handtutortm, 2017.

[7] Robots for stroke rehabilitation. Available at https://www.technologynetworks.com/neuroscience/news/robots-stroke-rehabilitation-283022, 2015.

[8] Reogo. Available at https://samcon.be/Rehabilitation/Robotics/ReoGo.

[9] Musicglove - home suite for mac/pc. Available at https://www.rehabtechnology.com.au/product/musicglove-home-suite/, 2018.

[10] Syrebo robotic gloves. Available at https://rebotics.co.uk/syrebo-robotic-gloves.

[11] Radhika Holmstrom. Getting to grips. Available at https://www.imperial.ac.uk/stories/getting-to-grips/.

[12] Ablex. Available at https://ablex.healthcare/, Apr 2021.

[13] Bimeo pro rehabilitation - gies 2021 exhibition products. Available at https://gies.hk/en/expo/exhibition-products/detail/1124, 2021.

[14] Spasticity and tone: Improve flexibility. Available at https://motusnova.com/hand/.

[15] Saeborejoyce - hand rehabilitation system by saebo. Available at https://www.medicalexpo.com/prod/saebo/product-80464-620339.html.

[16] Nodemcu esp8266 esp-12n v1.0 wifi cp2102 iot lua 267. Available at https://www.gmelectronic.com/esp12n-v1-0-nodemcu-lua267-esp8266-cp1202?id_lang=2&id_currency=2.

[17] Chia-Ye Chu and Rita Patterson. Soft robotic devices for hand rehabilitation and assistance: A narrative review. *Journal of NeuroEngineering and Rehabilitation*, 15, 02 2018.

[18] Damien Djaouti, Julian Alvarez, and Jean-Pierre Jessel. Classifying serious games: the g/p/s model. *Handbook of Research on Improving Learning and Motivation through Educational Games: Multidisciplinary Approaches*, 01 2011.

[19] Centers for Disease Control and Protection. Difficulties in physical functioning among adults aged 18 and over, by selected characteristics. Available at https://ftp.cdc.gov/pub/Health_Statistics/NCHS/NHIS/SHS/2015_SHS_Table_A-10.pdf, 2015.

[20] Prompt Therapy Solutions. The patient retention problem in physical therapy. Available at https://promptemr.com/blog/the-patient-retention-problem-in-physical-therapy/, Aug 2022.

[21] Varacallo M Erwin J. Anatomy, shoulder and upper limb, wrist joint. *Treasure Island (FL): StatPearls Publishing*, Sep 2021. Available at https://www.ncbi.nlm.nih.gov/books/NBK534779/.

[22] Aaron Beger BSc. Interphalangeal joints of the hand. Available at https://www.kenhub.com/en/library/anatomy/interphalangeal-joints-of-the-hand, Aug 2022.

[23] Roberto Grujičić MD. Metacarpophalangeal (mcp) joints. Available at https://www.kenhub.com/en/library/anatomy/metacarpophalangeal-mcp-joints, Aug 2022.

[24] Adrian Rad BSc. Carpometacarpal (cmc) joints. Available at https://www.kenhub.com/en/library/anatomy/carpometacarpal-cmc-joints, Jul 2022.

[25] Jana Vasković MD. Trapeziometacarpal joint. Available at https://www.kenhub.com/en/library/anatomy/trapeziometacarpal-joint, Jul 2022.

[26] Arthritis. Available at https://www.cdc.gov/arthritis/index.html, May 2020.

[27] Rheumatoid arthritis. Available at https://www.niams.nih.gov/health-topics/rheumatoid-arthritis, March 2022.

[28] Cerebral palsy (cp). Available at https://www.cdc.gov/ncbddd/cp/, May 2022.

[29] What is parkinson's? Available at https://www.parkinson.org/understanding-parkinsons/what-is-parkinsons.

[30] Parkinson's disease: Causes, symptoms, and treatments. Available at https://www.nia.nih.gov/health/parkinsons-disease, April 2022.

[31] Stroke. Available at https://www.nhs.uk/conditions/stroke/, Aug 2019.

[32] Spinal cord injury. Available at https://www.ninds.nih.gov/health-information/disorders/spinal-cord-injury.

[33] Hand disorders. Available at https://www.bssh.ac.uk/patients/conditions/hand_disorders, 2022.

[34] Tannan SC Satteson E. De quervain tenosynovitis. *Treasure Island (FL): StatPearls Publishing*, Feb 2022. Available at https://www.ncbi.nlm.nih.gov/books/NBK442005/.

[35] Range of motion. Available at https://www.physio-pedia.com/Range_of_Motion?utm_source=physiopedia&utm_medium=search&utm_campaign=ongoing_internal, 2022.

[36] Hand and finger exercises. Available at https://www.assh.org/handcare/condition/hand-finger-exercises.

[37] 12 simple exercises to heal your injured hand. Available at https://centralorthopedicgroup.com/12-simple-exercises-to-heal-your-injured-hand/, Aug 2019.

[38] Stroke: Hand rehabilitation. Available at https://www.physio-pedia.com/Stroke:_Hand_Rehabilitation, 2022.

[39] Soheila Saeedi, Marjan Ghazisaeedi, and Sorayya Rezayi. Applying game-based approaches for physical rehabilitation of poststroke patients: A systematic review. *Journal of Healthcare Engineering*, 2021:1–27, 09 2021.

[40] Hee-Tae Jung, Hwan Kim, Jugyeong Jeong, Bomin Jeon, Taekeong Ryu, and Yangsoo Kim. Feasibility of using the rapael smart glove in upper limb physical therapy for patients after stroke: A randomized controlled trial. volume 2017, pages 3856–3859, 07 2017.

[41] Joon-Ho Shin, Mi-Young Kim, Ji-Yeong Lee, Yu-Jin Jeon, Kim Suyoung, Soobin Lee, Beomjoo Seo, and Younggeun Choi. Effects of virtual reality-based rehabilitation on distal upper extremity function and health-related quality of life: A single-blinded, randomized controlled trial. *Journal of NeuroEngineering and Rehabilitation*, 13, 02 2016.

[42] Neofect. Rapael smart glove. Available at https://forexmedical.hu/wp-content/uploads/2021/11/Brochure-Smart-Glove-A1-EN.pdf.

[43] Maik Hartwig. Fun and evidence-computer-based arm rehabilitation with the pablo ® plus system. 2011.

[44] Marco Tramontano, Laura Conti, Niccolò Marziali, Giorgia Agostini, Sara De Angelis, Giovanni Galeoto, and Maria Grasso. *Hand Robotics Rehabilitation in Patients with Multiple Sclerosis: A Pilot Study*, pages 50–57. 07 2020.

[45] Pablo® upper extremity. Available at https://tyromotion.com/en/products/pablo/, Aug 2021.

[46] Amadeo®. Available at https://www.tradex-services.com/company-product/272/.

[47] Xianwei Huang, Fazel Naghdy, Golshah Naghdy, Haiping Du, and Catherine Todd. The combined effects of adaptive control and virtual reality on robot-assisted fine hand motion rehabilitation in chronic stroke patients: A case study. *Journal of Stroke and Cerebrovascular Diseases*, 27, 09 2017.

[48] Amadeo®: The pioneer in finger-hand-rehabilitation: Tyrotherapy. Available at https://tyromotion.com/en/products/amadeo/, Aug 2021.

[49] Rehab-Robotics. Hand of hope for hand rehabilitation. Available at http://www.rehab-robotics.com.hk/hoh/.

[50] Rehab Robotics Cmp Ltd. How hand of hope can help you in stroke rehabilitation? Available at http://www.rehab-robotics.com.hk/hoh/RM-230-HOH3-0001-7%20HOH_brochure_eng.pdf, journal=Rehab-Robotics.

[51] Crescendo. Available at https://www.gloreha.com/crescendo/, journal=Gloreha, Apr 2022.

[52] Sinfonia. Available at https://www.gloreha.com/sinfonia/, Jan 2022.

[53] Aria - hand rehabilitation system by idrogenet. Available at https://www.medicalexpo.com/prod/idrogenet/product-74722-1021399.html.

[54] Federica Bressi, Fabio Santacaterina, Laura Cricenti, Benedetta Campagnola, Francesca Nasto, Carla Assenza, Daniela Morelli, Francesca Cordella, Martina Lapresa, Loredana Zollo, Silvia Sterzi, and Marco Bravi. Robotic-assisted hand therapy with gloreha sinfonia for the improvement of hand function after pediatric stroke: A case report. *Applied Sciences*, 12:4206, 04 2022.

[55] Aria. Available at https://www.gloreha.com/gloreha-aria/, Dec 2021.

[56] Eli Carmeli, Sara Peleg, Gadi Bartur, Enbal Elbo, and Jean-Jacques Vatine. Handtutortm enhanced hand rehabilitation after stroke - a pilot study. *Physiotherapy research international : the journal for researchers and clinicians in physical therapy*, 16:191–200, 12 2011.

[57] Session 2, handtutor system treatment 2. Available at https://www.youtube.com/watch?v=ZL07W9-ZNm8, Jul 2020.

[58] Handtutor. Available at https://meditouch.co.il/products/handtutor/, Sep 2021.

[59] Farshid Amirabdollahian, Serdar Ates, Angelo Basteris, Alfredo Cesario, Jaap Buurke, Hermie Hermens, Dennis Hofs, E. Johansson, Gail Mountain, Nasrin Nasr, Sharon Nijenhuis, Gerdienke Prange, N. Rahman, Patrizio Sale, F. Schätzlein, Boris Schooten, and AHA Stienen. Design, development and deployment of a hand/wrist exoskeleton for home-based rehabilitation after stroke - script project. *Robotica*, 32:1331–1346, 12 2014.

[60] Federica Bovolenta, Milena Goldoni, Pierina Clerici, Maurizio Agosti, and Marco Franceschini. Robot therapy for functional recovery of the upper limbs: A pilot study on patients after stroke. *Journal of rehabilitation medicine : official journal of the UEMS European Board of Physical and Rehabilitation Medicine*, 41:971–5, 11 2009.

[61] Nizan Friedman, Vicky Chan, Danny Zondervan, Mark Bachman, and David Reinkensmeyer. Musicglove: Motivating and quantifying hand movement rehabilitation by using functional grips to play music. *Conference proceedings : ... Annual International Conference of the IEEE Engineering in Medicine and Biology Society. IEEE Engineering in Medicine and Biology Society. Conference*, 2011:2359–63, 08 2011.

[62] Order the musicglove hand therapy glove for stroke patients. Available at https://www.flintrehab.com/product/musicglove-hand-therapy/, Jul 2022.

[63] Stroke therapy glove. Available at https://www.syrebo.com/rehabilitation-system-for-clinic/stroke-therapy-glove.html.

[64] Gripable. Available at https://gripable.co/, Apr 2022.

[65] Kimberlee Jordan, Michael Sampson, and Marcus King. Gravity-supported exercise with computer gaming improves arm function in chronic stroke. *Archives of physical medicine and rehabilitation*, 95, 03 2014.

[66] Maja Goršič, Imre Cikajlo, Nika Goljar, and Vesna Novak. A multisession evaluation of an adaptive competitive arm rehabilitation game. *Journal of NeuroEngineering and Rehabilitation*, 14, 12 2017.

[67] bimeo. Available at https://kinestica.com/bimeo/.

[68] Libby Rosenstein, Angela Ridgel, Anil Thota, Bridgette Samame, and Jay Alberts. Effects of combined robotic therapy and repetitive-task practice on upper-extremity function in a patient with chronic stroke. *The American journal of occupational therapy : official publication of the American Occupational Therapy Association*, 62:28–35, 01 2008.

[69] Saebo. Saeborejoyce. Available at https://www.saebo.com/wp-content/uploads/2016/01/SaeboReJoyce-Brochure.pdf.

[70] Esp8266. Available at https://www.espressif.com/en/products/socs/esp8266, 2022.

[71] Ads1115 16-bit adc - 4 channel with programmable gain amplifier. Available at https://www.adafruit.com/product/1085, 2022.

[72] Adc 16bits 4 canais i2c c/ amplificador ads1115. Available at https://www.botnroll.com/pt/i2c-spi/3669-adc-16bits-4-canais-i2c-c-amplificador-ads1115.html.

[73] Flex sensor 2.2". Available at https://www.botnroll.com/en/flex-pressure-vibration/389-flex-sensor-22.html.

[74] Sreejan Alapati and Shivraj Narayan Yeole. A review on applications of flex sensors. *International Journal of Emerging Technology and Advanced Engineering*, 7:97–100, 07 2017.

[75] Altimu-10 v5 gyro, accelerometer, compass, and altimeter (lsm6ds33, lis3mdl, and lps25h carrier). Available at https://www.pololu.com/product/2739.

[76] Pololu - minimu-9 v3 gyro, accelerometer, and compass (l3gd20h and lsm303d carrier). Available at https://www.pololu.com/product/2468.

[77] Pololu minimu-9 + arduino ahrs (attitude and heading reference system). Available at https://github.com/pololu/minimu-9-ahrs-arduino, Nov 2018.

[78] William Premerlani and Paul Bizard. Direction cosine matrix imu: Theory. *DIY DRONE: USA*, 01 2009.

[79] Direction cosine matrix. Available at https://ahrs.readthedocs.io/en/latest/dcm.html.

[80] Idrogenet. Gloreha Sinfonia. Available at https://www.omnia-health.com/product/gloreha-sinfonia.

[81] Mixamo. Available at https://www.mixamo.com/#/.

Appendix A

Game results and calibration data

Pronation	Supination	Range	Range (deg)
0,2	0,8	0,6	108
0,24	0,76	0,52	93,6
0,26	0,79	0,53	95,4
0,23	0,83	0,6	108
0,27	0,86	0,59	106,2
0,19	0,85	0,66	118,8

Table A.1: Wrist pronation/supination

Flexion	Extension	Range	Range (deg)
0,73	0,52	0,21	37,8
0,73	0,54	0,19	34,2
0,74	0,55	0,19	34,2
0,74	0,55	0,19	34,2
0,73	0,55	0,18	32,4
0,76	0,55	0,21	37,8
0,75	0,55	0,2	36
0,75	0,55	0,2	36
0,76	0,55	0,21	37,8
0,77	0,56	0,21	37,8

Table A.2: Wrist flexion/extension

Table A.3: Values saved in the wrist pronation/supination and wrist flexion/extension ROM JSON files

Thumb				Index Finger			
Flexion	**Extension**	**Range**	**Range (deg)**	**Flexion**	**Extension**	**Range**	**Range (deg)**
0,94	0,68	0,26	46,8	0,73	0,12	0,61	109,8
0,94	0,68	0,26	46,8	0,78	0,11	0,67	120,6
0,94	0,63	0,31	55,8	0,75	0,1	0,65	117
0,94	0,59	0,35	63	0,76	0,12	0,64	115,2
0,94	0,57	0,37	66,6	0,79	0,13	0,66	118,8
0,94	0,51	0,43	77,4	0,82	0,11	0,71	127,8
0,94	0,46	0,48	86,4	0,83	0,09	0,74	133,2
0,94	0,45	0,49	88,2	0,83	0,1	0,73	131,4

Table A.4: Thumb and index finger

Middle Finger				Ring Finger			
Flexion	**Extension**	**Range**	**Range (deg)**	**Flexion**	**Extension**	**Range**	**Range (deg)**
0,43	0	0,43	77,4	0,55	0,31	0,24	43,2
0,45	0,1	0,35	63	0,55	0,34	0,21	37,8
0,45	0	0,45	81	0,57	0,33	0,24	43,2
0,45	0	0,45	81	0,55	0,33	0,22	39,6
0,45	0,2	0,25	45	0,54	0,35	0,19	34,2
0,47	0	0,47	84,6	0,54	0,36	0,18	32,4
0,48	0,3	0,18	32,4	0,55	0,35	0,2	36
0,48	0	0,48	86,4	0,53	0,37	0,16	28,8

Table A.5: Middle and ring finger

Flexion	**Extension**	**Range**	**Range (deg)**
0,89	0,85	0,04	7,2
0,9	0,86	0,04	7,2
0,87	0,84	0,03	5,4
0,9	0,84	0,06	10,8
0,9	0,86	0,04	7,2
0,88	0,87	0,01	1,8
0,92	0,82	0,1	18
0,91	0,83	0,08	14,4

Table A.6: Little finger

Table A.7: Values saved in the finger flexion/extension ROM JSON files

Day	Result	Day average
18/08/2022	0,61	0,61
21/08/2022	0,58	0,58
22/08/2022	0,63	0,63
24/08/2022	0,67	0,67
27/08/2022	0,8	0,76
27/08/2022	0,72	
29/08/2022	0,75	0,77
29/08/2022	0,79	
30/08/2022	0,84	
30/08/2022	0,93	0,923
30/08/2022	1	

Table A.8: Task 1

Day	Result	Day average
03/09/2022	0,43	
03/09/2022	0,42	0,47
03/09/2022	0,56	
04/09/2022	0,39	
04/09/2022	0,53	0,5125
04/09/2022	0,46	
04/09/2022	0,67	
05/09/2022	0,54	
05/09/2022	0,34	
05/09/2022	0,62	0,47
05/09/2022	0,71	
05/09/2022	0,14	
06/09/2022	0,48	0,48
07/09/2022	0,56	0,56
10/09/2022	0,35	0,35

Table A.9: Task 2

Day	Result	Day average
21/06/2022	0,1	0,1
22/06/2022	0,11	0,11
25/06/2022	0,12	
25/06/2022	0,13	0,135
25/06/2022	0,14	
25/06/2022	0,15	
26/06/2022	0,16	0,16
27/06/2022	0,17	0,17
28/06/2022	0,18	0,18
28/06/2022	0,19	0,19
29/06/2022	0,18	0,18
01/07/2022	0,17	0,165
01/07/2022	0,16	

Table A.10: Task 3

Table A.11: Values saved in the Fingers Game JSON files

Day	Result	Day average
21/06/2022	0,1	0,1
22/06/2022	0,11	0,11
25/06/2022	0,12	0,135
25/06/2022	0,13	
25/06/2022	0,14	
25/06/2022	0,15	
26/06/2022	0,16	0,16
27/06/2022	0,17	0,17
28/06/2022	0,18	0,18
28/06/2022	0,19	0,19
29/06/2022	0,18	0,18
01/07/2022	0,17	0,165
01/07/2022	0,16	

Table A.12: Wrist pronation/supination

Day	Result	Day average
03/09/2022	0,43	0,47
03/09/2022	0,42	
03/09/2022	0,56	
04/09/2022	0,39	0,5125
04/09/2022	0,53	
04/09/2022	0,46	
04/09/2022	0,67	
05/09/2022	0,54	0,47
05/09/2022	0,34	
05/09/2022	0,62	
05/09/2022	0,71	
05/09/2022	0,14	
06/09/2022	0,48	0,48
07/09/2022	0,56	0,56
10/09/2022	0,35	0,35

Table A.13: Wrist flexion/extension

Day	Result	Day average
15/04/2022	10	12,71
15/04/2022	13	
15/04/2022	15	
15/04/2022	14	
15/04/2022	9	
15/04/2022	12	
15/04/2022	16	
16/04/2022	19	26,86
16/04/2022	26	
16/04/2022	22	
16/04/2022	25	
16/04/2022	28	
16/04/2022	35	
16/04/2022	33	

Table A.14: Finger flexion/extension

Table A.15: Values saved in the Ball Catcher JSON files