

Stroke Rehabilitation Therapy for the Upper Extremity with a Virtual Coach for Compensation Reduction

Ana Rita Mendes Córias

Thesis to obtain the Master of Science Degree in
Electrical and Computer Engineering

Supervisors: Prof. Dr. Alexandre José Malheiro Bernardino
Prof. Dr. José Alberto Rosado dos Santos-Vitor

Examination Committee

Chairperson: Prof. Dr. João Fernando Cardoso Silva Sequeira
Supervisor: Prof. Dr. Alexandre José Malheiro Bernardino
Member of the Committee: Prof. Dr. João Miguel Raposo Sanches

February 2021

Declaration

I declare that this document is an original work of my own authorship and that it fulfills all the requirements of the Code of Conduct and Good Practices of the Universidade de Lisboa.

For my loved ones,

Acknowledgments

This master thesis represents the end of a great journey as a student. It represents the most happy and pleasant years of my life and also the most challenging ones. I would not be able to achieve this moment without the most wonderful people with whom I am fortunate to share my life. To them, I want to express my sincere and special acknowledgments.

First, I want to thank my supervisors, Professors Alexandre Bernardino and José Santos-Vitor, for the opportunity they gave me to work on the thematic of assistive systems for rehabilitation after stroke. I especially want to thank Professor Alexandre for his immense support and patience during the development of this thesis, with his wise advice and suggestions, and for giving me the freedom to follow my ideas and accomplish my ambitions regarding this work.

I want to thank my colleagues from ISR, João Avelino, Heitor Cardoso, and Min Lee, for their precious advice and help, especially João Avelino for his availability to help me simulate my work with Vizzy cameras and kind patience.

I would like to thank rehabilitation therapists Mariana Mateus and Carolina Cardoso from NeuroSer for letting me attend therapy sessions and giving me insights into therapeutic approaches.

A special thanks to my brother for always believe in me and for being my role model. It was due to him that I start to study engineering. I want to thank my parents for all their support, especially my mother, who does everything for me, is always by my side, and provided me with the best values.

I want to thank all my amazing friends for their unconditional support and all the great moments they provided me during my academic journey: Bukkatuna, Luís Rodrigues, Eduardo Crespo, Sara Barroso, and Catarina Rocha for being so awesome. A special thanks to my friend José Cordeiro for being such an inspiration. My friend Sebastião Quintas for all the advice and brainstorm. And my ladies Joana Baleiras, Catarina Ferreira, Catarina Dente, and Marília Barata for always being there for me.

Last, but extremely important, I want to thank my friend Francisco Patrocinio and my grandmother, a star in the sky, whose most difficult phase of their lives was the inspiration and motivation for this work. Thank you!

Abstract

Stroke is one of the leading causes of death and long-term disability in the western world. The increasing demands concerning stroke rehabilitation and in-home exercise promotion increased the need for adequate, affordable, and accessible Assistive Technology (AT) to promote patients' compliance in therapy while exercising autonomously. Independent exercise with an assistive system requires objective methods to assess patients' quality of movement and track their progress. In this work, we develop an image-based Virtual Coach (VC) capable of monitoring upper extremity rehabilitation exercises focused on motor compensation reduction. The VC proposes three exercises and assesses users' compensation patterns from images acquired with a webcam. It provides proper visual and audio feedback and instructions through a User Interface (UI). We propose objective measures and classification approaches - a Rule-based (RB) and a Neural Network (NN) based - to assess motor compensation patterns from Two-Dimensional (2D) positional data for the three exercises. For exercise 1, the RB approach assessed different compensation patterns with F_1 score of 76.69%. For exercise 2 and 3, the NN based approach revealed F_1 score of 72.56% and 79.87%, respectively. A group of seven volunteers exercised with the VC in a small experimental session. The group found the system enjoyable and relevant for rehabilitation administration. These results give evidence about the value of this kind of system to aid stroke patients under rehabilitation and accurate performance assessment from 2D data. This latter enables to automate rehabilitation programs monitorization in any device with a 2D camera, such as tablets, smartphones, or robotic assistants.

Keywords

Stroke; Virtual Coach; Performance Assessment; 2D Positional Data.

Resumo

O Acidente Vascular Cerebral (AVC) é uma das principais causas de morte e invalidez de longo prazo no mundo ocidental. As crescentes exigências em relação à reabilitação pós-AVC e recomendações de exercício em casa, aumentaram a necessidade de tecnologia assistiva adequada e acessível para promover a adesão dos pacientes à terapia durante o exercício autônomo. O exercício independente com um sistema assistivo requer métodos objetivos para avaliar a qualidade do movimento dos pacientes e monitorizar o seu progresso. Neste trabalho, desenvolvemos um terapeuta virtual, baseado em imagem, capaz de monitorizar exercícios de reabilitação dos membros superiores com foco na redução de movimentos compensatórios. O terapeuta virtual propõe três exercícios e avalia os padrões de compensação dos utilizadores a partir de imagens adquiridas com uma *webcam*. Ele fornece *feedback*, visual e de áudio, e instruções adequados por meio de uma interface de utilizador. Propomos medidas objetivas e métodos de classificação - um baseado em regras e outro baseado em rede neuronal - para avaliar os padrões de compensação motora a partir de posição 2D para os três exercícios. Para o exercício 1, o método de regras avaliou diferentes padrões de compensação com um *score* F_1 de 76,69%. Para os exercícios 2 e 3, a abordagem baseada em rede neuronal revelou um *score* F_1 de 72,56% e 79,87%, respectivamente. Um grupo de sete voluntários exercitou com o terapeuta virtual numa pequena sessão experimental. O grupo considerou o sistema agradável e relevante para a administração de reabilitação. Estes resultados fornecem evidências sobre o valor deste tipo de sistema para ajudar pacientes com AVC em reabilitação e precisão na avaliação de desempenho a partir de dados 2D. Este último permite automatizar a monitorização de programas de reabilitação em qualquer dispositivo com câmara 2D, como tablets, smartphones ou assistentes robóticos.

Palavras Chave

Acidente Vascular Cerebral (AVC); Terapeuta Virtual; Avaliação da Performance; Dados posicionais 2D.

Contents

1	Introduction	1
1.1	Motivation	2
1.2	Objectives and Contributions	3
1.3	Organization of the Document	4
2	Background & State of the Art	5
2.1	Background	5
2.1.1	Rehabilitation Therapy Process	5
2.1.2	Motor Compensation	7
2.1.3	Virtual Coaches	8
2.2	State of the Art	8
2.2.1	Virtual Assistive Systems for Upper Extremity Rehabilitation After Stroke	8
2.2.2	Quality of Movement Assessment During Exercise Performance	13
3	Methodologies	22
3.1	Virtual Coach	22
3.1.1	Requirements	22
3.1.2	The Intelligent Agent	23
3.1.3	System Architecture	25
3.1.4	User Interface	28
3.2	Compensation Assessment Methods	30
3.2.1	Feature Extraction	31
3.2.1.A	OpenPose	32
3.2.1.B	OpenFace	33
3.2.2	Feature Selection	34
3.2.3	Data Normalization	35
3.2.3.A	Transformation	35
3.2.3.B	Normalization	38
3.2.3.C	Mirror	38

3.2.3.D	Kinematic Variables	38
3.2.4	Classification Approaches	42
3.2.4.A	Rule-based Classification Method	45
3.2.4.B	Neural Network Based Classification Method	47
3.2.5	Classification Result Filtering	48
4	Experiments	49
4.1	Dataset	49
4.1.1	Labeling Process	49
4.1.2	Data Cleansing	51
4.1.3	The Three Upper Extremity Exercises	53
4.1.4	Multilabel Dataset Characteristics	53
4.2	Classification Methods	55
4.2.1	Evaluation Metrics	56
4.2.2	Validation Method	57
4.2.3	Rule-based Classification Method	58
4.2.3.A	Hypotheses to Assess Movements in Depth	58
4.2.3.B	Kinematic Variables to Assess Motion Patterns Observed in the Dataset	60
4.2.4	Neural Network Classification Method	62
4.2.5	Filtering of the Classification Results	62
4.3	Virtual Coach	63
4.3.1	Volunteers	63
4.3.2	Experimental Setup	64
4.3.3	Experimental Procedure	64
4.3.4	Data Collection	64
5	Evaluation	65
5.1	Classification Results Analysis	65
5.1.1	Exercise 1	65
5.1.1.A	Rule-based Classification Method	66
5.1.1.B	Neural Network Classification Method	67
5.1.2	Exercise 2	68
5.1.2.A	Rule-based Classification Method	68
5.1.2.B	Neural Network Classification Method	69
5.1.3	Exercise 3	70
5.1.3.A	Rule-based Classification Method	70
5.1.3.B	Neural Network Classification Method	71

5.1.4	Filtering of the Classification Results	71
5.1.5	Result Comparison	73
5.2	Virtual Coach Validation Results	74
5.2.1	Quantitative Results	75
5.2.1.A	Stroke Survivor vs. Other Volunteers	76
5.2.1.B	The Age Factor	76
5.2.1.C	Volunteers Experience with the Virtual Coach	77
5.2.2	Qualitative Results and Discussion	79
6	Conclusions & Future Work	81
6.1	Future Work	82
A	Appendix A	87
A.1	Compensation Patterns	87
A.2	Virtual Coach Speech	88
A.3	Web App	89
A.4	The Dataset	91
A.5	Kinematic Variables Validation	93
A.5.1	Hypotheses to Assess Movements in Depth	93
A.5.2	Kinematic Variables to Assess Motion Patterns Observed in the Dataset	96
A.5.3	Kinematic Variables to Assess Motion Patterns Not Observed in the Dataset	96
A.6	Classification Results	97
A.7	Questionnaire	99

List of Figures

2.1	Model of the recommended process for the upper extremity management. Taken from [1]. . .	6
2.2	Patient (left) exercising with the AMRR and the supervisor therapist (right). Taken from [2]. . .	10
2.3	A patient exercising (A), HAMSTER setup (B), and games' graphical representation (C-G). Taken from [3].	10
2.4	The technical infrastructure of the rehabilitation system (A), set of smart objects (B), and the combination of two objects in a task (C). Taken from [4].	12
2.5	Setup for the drinking task with the 5-camera capture system: the participant's arm is in its initial position, and the black dots represent the retroreflective markers. Taken from [5].	15
2.6	Joints acquired with a Kinect Sensor. Taken from [6].	16
2.7	A. Illustration of the analyzed movements; red lines and arrows represent movement direction and joint angles. B. Keypoints detected by both motion capture system, Kinect (red dots) and LED markers (blue dots). Adapted from [7].	17
2.8	Flow diagram of the proposed method. Taken from [6].	20
3.1	State transition diagram.	26
3.2	Web application pages.	26
3.3	Virtual Coach workflow.	27
3.4	Virtual Coach Init web page.	29
3.5	Virtual Coach Menu web page.	29
3.6	Virtual Coach Main web page - Patient positioning.	29
3.7	Virtual Coach Main web page - Target position.	30
3.8	Virtual Coach Main web page - Display compensation markers.	30
3.9	Work flow diagram of the compensation assessment method.	31
3.10	OpenPose keypoints. Adapted from [8].	32
3.11	Examples of OpenFace facial landmark detection and bounding box around the head. Taken from [9].	34
3.12	Patient selection method in the case of multi-person detection.	34

3.13	Keypoints' selection for both scenarios S1, S2, and S3.	35
3.14	Determine Y axis for the body 2D Cartesian space, $\{B\}$, for the different scenarios.	36
3.15	${}^B X$ axis for the body 2D Cartesian space, B , for left and right affected sides in all scenarios (S1, S2, and S3). Translation vector between $\{I\}$ and $\{B\}$ coordinate systems, ${}^B P_I$	37
3.16	Trunk moving forward and backward in the case of S2 and S3.	39
3.17	The patient's head area, and triangle defining it.	40
3.18	Trunk Rotation hypothesis for S1.	41
3.19	Trunk Rotation hypothesis for S2 and S3.	41
3.20	Shoulder Elevation for S2 and S3.	42
3.21	Trunk tilt for S1.	42
3.22	Rules to assess Trunk Rotation (TR) and Shoulder Elevation (SE) in S1.	46
3.23	Rules to assess Trunk Forward (TF) and trunk backward - Other (O) - in S2.	46
3.24	NN based approach to assess compensation from 2D positional data.	47
3.25	Filtering of the classification results.	48
4.1	Illustration of the method adopted to select the significant subject present in a video frame.	51
4.2	Examples of extra skeletons detected by OpenPose.	51
4.3	OpenPose keypoint misdetection with a reasonable confidence score value and for patient P12 in E1.	52
4.4	Patients in and perpendicular (S2) and oblique (S3) position regarding the recording camera.	53
4.5	Head area over time, revealing trunk moving backward (Other) observed in the dataset for E2.	59
4.6	Head translation over time in ${}^W Z$, revealing trunk moving backward (Other) observed in the dataset for E2.	59
4.7	Patient shoulders' elevation angles over time describing Trunk Rotation for E1.	60
4.8	Patient affected shoulder elevation angle revealing Shoulder Elevation for E2.	61
4.9	Patient tilted angle of the torso describing a trunk tilt (Other) for E2.	61
4.10	Patient P06 tilted and of the spine and shoulder displacement over time, describing Trunk Forward in E3.	61
4.11	Patient P05 shoulder displacement over time, describing Shoulder Elevation in E3.	61
4.12	Movement pattern transitions described in a set of frames which we call borders	63
5.1	E1 Rule-Based Multilabel Classifier Evaluation Results.	66
5.2	E1 NN Based Multilabel Classifier Evaluation Results.	67
5.3	E2 Rule-Based Multilabel Classifier Evaluation Results.	68
5.4	E2 NN Based Multilabel Classifier Evaluation Results.	69
5.5	E3 Rule-Based Multilabel Classifier Evaluation Results.	70

5.6	E3 NN Based Multilabel Classifier Evaluation Results.	71
5.7	Evaluation Results for the three exercises.	73
5.8	Perceptions of the Virtual Coach.	76
5.9	Stroke Survivor and Other Volunteers Perceptions of the Four Dimensions.	77
5.10	Perceptions among younger and older adults.	78
A.1	Examples of Motor Compensation.	87
A.2	Virtual Coach Menu web page and warnings.	89
A.3	Virtual Coach Demo web page.	89
A.4	Virtual Coach Main web page - Target position.	90
A.5	Head area over time of a Trunk Forward simulation in similar conditions of the dataset.	93
A.6	Head area over time of a trunk moving backward (Other) simulation in similar conditions of the dataset.	93
A.7	Head area over time of a trunk tilt (Other) simulation in similar conditions of the dataset for E3.	93
A.8	Head area over time, revealing trunk moving backward (Other) observed in the dataset for E1.	93
A.9	Head translation over time in $^W Z$ of a Trunk Forward simulation. Video with a resolution of 640×480 pixels.	94
A.10	Head translation over time in $^W Z$ of a trunk moving backward (Other) simulation. Video with a resolution of 640×480 pixels.	94
A.11	Head translation over time in $^W Z$ of a trunk to tilt (Other) simulation. Video with a resolution of 640×480 pixels (E3).	94
A.12	Head translation over time in $^W Z$, revealing trunk moving backward (Other) observed in the dataset for E1.	94
A.13	Head translation over time in $^W Z$ of a Trunk Forward simulation. Video with a resolution of 1920×1080 pixels.	95
A.14	Head translation over time in $^W Z$ of a trunk moving backward (Other) simulation. Video with a resolution of 1920×1080 pixels.	95
A.15	Head translation over time in $^W Z$ of a trunk to tilt (Other) for E3. Video with a resolution of 1920×1080 pixels.	95
A.16	Patient shoulders' elevation angles over time describing Trunk Rotation for E2.	96
A.17	Patient affected shoulder elevation angle revealing Shoulder Elevation for E1.	96
A.18	Patient tilted angle of the torso describing a trunk tilt (Other) for E1.	96
A.19	Tilted angle of the spine and <i>Neck</i> displacement over time acquired from a simulation of trunk moving backward (Other) in E3.	97
A.20	Length between both shoulders for an oblique position and shoulder displacement for a perpendicular position describing trunk to tilt (Other).	97

List of Tables

2.1	Examples of upper extremity exercises. Adapted from [1].	6
2.2	Performance components under evaluation. Adapted from [6].	19
2.3	Upper extremity exercises and respective Fugl-Meyer Assessment (FMA) and Wolf Motor Function Test (WMFT) tasks items. Adapted from [6].	19
2.4	Normalized features for the compensation performance component. Joints' names are in accordance with figure 2.6. Adapted from [6].	19
2.5	NN Explored Hyperparameters. Adapted from [6].	20
3.1	Environment States \mathcal{S}	24
3.2	Agent Actions \mathcal{A} , respective condition-action rules, and action description.	25
3.3	Diagram Function Blocks (figure 3.3).	28
3.4	Compensation Assessment Procedures.	31
3.5	Mathematical notation.	31
3.6	OpenPose body keypoints and applied notation.	33
3.7	Transformation Components.	36
3.8	Hypotheses and Kinematic Variables to Assess Trunk Forward for all the Scenarios.	39
3.9	Hypotheses and Kinematic Variables to Assess Trunk Rotation for all the Scenarios.	40
3.10	Hypotheses and Kinematic Variables to Assess Shoulder Elevation for all the Scenarios.	42
3.11	Hypotheses and kinematic variables to assess Other trunk compensation behaviors for all the scenarios.	43
3.12	Kinematic variables for Rule-based classification.	43
4.1	The number of video frames per exercise that compose the dataset before data cleansing.	50
4.2	Label occurrence before data cleansing.	51
4.3	Dataset after data cleansing.	53
4.4	Label occurrence after data cleansing (and without patient P12 for E1).	54
4.5	Dataset characteristics.	55

4.6	<i>IRLbl</i> for each label $l \in \mathcal{L}$.	55
5.1	Evaluation results concerning F_1 score and <i>HammingLoss</i> after label filtering.	72
5.2	Evaluation results concerning F_1 score and <i>HammingLoss</i> after borders removal.	72
5.3	Volunteers' profiles: (a) Knows what a stroke is (b) Had a stroke (c) Some relative or close friend had a stroke (d) Followed the rehabilitation process closely. *Volunteer ID, **Non-dominant/Affected side.	75
5.4	Descriptive Statistics and Pearson Correlation.	78
A.1	Virtual Coach Speech.	88
A.2	Three upper extremity exercises with respective initial and target arm positions. The images of initial and target positions are displayed with the body skeleton extracted with OpenPose.	91
A.3	Labels to label the dataset video frames and respective examples of the observed compensation behavior.	92
A.4	Classification Report for P06 E1 data as validation set for the NN Based approach.	98
A.5	P11 classification report NN Based approach for E1.	98
A.6	Questionnaire to Volunteers to evaluate the Virtual Coach.	99

Acronyms

VC	Virtual Coach
WHO	World Health Organization
AT	Assistive Technology
CIMT	Constraint-Induced Movement Therapy
WMFT	Wolf Motor Function Test
FMA	Fugl-Meyer Assessment
3D	Three-Dimensional
2D	Two-Dimensional
PCA	Principal Component Analysis
SD	Standard Deviation
RB	Rule-based
NN	Neural Network
MSE	Mean Squared Error
UI	User Interface
ML	Machine Learning
MLC	Multilabel Classification
MLD	Multilabel Dataset

1

Introduction

Virtual Coaches are intelligent agents designed to provide adequate and necessary assistance, training, or therapy to their users. A Virtual Coach (VC) regularly monitors user's activities and behaviors, and the environment where one is inserted. It is aware of the user's physical or cognitive conditions. It can determine when should intervene, providing instructions, or offering assistance. When it turns to aid people with a higher or lower level of incapacity, these agents can positively intervene in providing rehabilitation, learning strategies, and illness management [10, 11].

According to the World Health Organization (WHO), every person will experience some level of disability, directly or indirectly, due to disease, injury, or age factors [12, 13]. As the world's population is growing older [14] and there is a high increase of people with disability, or some limitation in functioning [15], the attention to Assistive Technology (AT), such as virtual coaches, has risen, aiming to improve these people's quality of life and ease their caregivers' work. Along with this huge benefit, it promises to provide physicians and therapists with extra information about patients' health state and evolution, and quantitative measures to quantify and track disability to bring concrete consensus to the subjective evaluation usually made towards disability [12, 13, 15]. Due to the clear evidence of AT's positive impact, the WHO has made some efforts regarding its access and regulations [15]. The WHO's documents state that AT is a fundamental tool to aid people and improve their functioning. They highlight the importance of affordable, accessible, and appropriate AT [12, 15].

After a stroke, patients lose part of their physical capabilities and see themselves with troubles when performing daily activities [16]. Faced with this new condition, patients tend to develop new movement behaviors during task performance, like trunk displacement and shoulder elevation, commonly called compensatory movement patterns [17, 18]. To diminish the impact of physical impairments and reduce compensatory patterns, rehabilitation poses an essential strategy [1, 17], in which the upper extremity needs special attention. During the rehabilitation process, patients face many challenges that compose barriers [16, 19], which highlight the need for assistive agents.

The present project aims to provide solutions for stroke survivors' main challenges during the therapy process. We propose a Virtual Coach (VC) capable of assessing compensatory behaviors using supervised learning methods and provide proper and engaging therapy for the upper extremity. We present a method to assess quantitatively motor compensation from video frames during upper limb exercise performance based on 2D pose data enabling this kind of analysis with widely available RGB cameras.

1.1 Motivation

Stroke is one of the leading causes of death and adult disability in the western world [16, 19, 20, 21]. Due to stroke effects, survivors see themselves with physical/cognitive impairments [16] and often experience a more weakened body side (right or left) [17, 21]. Such disablement has a significant impact on their lives since they are no longer capable of accomplishing their daily tasks and perform their pre-stroke life roles [1, 19]. To reduce this impact, prevent disability and stroke recurrence, rehabilitation poses a crucial and effective strategy [1, 19, 20].

Rehabilitation “starts at the time of the stroke event and continues as long as required for each individual to achieve their maximum potential recovery” [1]. In this process, therapists actively promote physical activity and provide specific exercises to improve stroke survivors’ functional abilities and, consequently, their quality of life. Upper limb recovery, or partial recovery, is fundamental for patients, allowing them to execute the same tasks they used to do or, at least, simple and necessary ones [1, 22]. The exercises conducted can be passive or active [23]. In passive exercises, commonly applied in a primary stage, therapists physically support and stimulate arm movements. In active exercises, which have greater results, patients move and exercise their limbs independently without physical support or assistance, only following the therapist’s instructions. For the upper extremity, Constraint-Induced Movement Therapy (CIMT), based on a task-oriented approach, has revealed to be really effective [20, 24]. This therapy consists of an interaction “between many systems in the brain and is organized around a goal and constrained by the environment,” [20] in which the patient has to exercise repeatedly, giving special attention to the affected limb [24].

To have the desired outcome, rehabilitation poses a serious commitment, in which patients need to be engaged. It demands a lot of time investment in training. Therefore, in addition to the exercises carried out during therapy, CIMT exercises can be prescribed or recommended to fill gaps in the schedule, e.g., between therapy sessions, or to be performed at home [1, 22].

Rehabilitation Therapy and Recovery Challenges

During the therapy process, both patients in a post-stroke status and therapists identify facilitators and barriers to exercise and go through a high-quality rehabilitation process and with the best outcome possible [16, 19, 25]. With the increasing number of patients with different therapeutic demands and therapists’ time availability, the latter cannot often give as much attention as their patients need, leading to a failure in the rehabilitation administration, with a rupture in the performance monitoring and feedback transfer [19, 25]. For the same reason, allied with the associated costs, public hospital facilities and rehabilitation centers suffer high pressures around patient discharge, which can misdirect rehabilitation priorities. With this target, lower limb functional improvement stands as a priority, leaving patients prepared to execute basic tasks and meet discharge requirements [16, 17]. This strategy leads to a lack of essential rehabilitation for the upper

extremity, which requires more time and specialized therapy [1, 16]. This fact implies that a patient has the opportunity to have access to supervised and diversified therapy for a limited period of time. The continuous availability of professional and multidisciplinary rehabilitation comprises a great economic effort made by the patients and their families [16]. In this situation, exercising at home with the support of AT may actually be the only solution [6].

To help patients exercise, whether in a hospital facility or at home, without supervision or just with partial supervision, AT can be very helpful. Nevertheless, both patients and therapists have reported problems with its availability and accessibility and difficulties in getting engaged with these technologies [16]. In fact, to keep themselves motivated and engaged in training, to relearn the lost capabilities due to stroke, is arguably the biggest challenge for stroke survivors. This motivation can be affected when patients see themselves physically impaired and incapable of accomplishing rehabilitation exercises, discouraging engagement in the prescribed activities [1, 16, 19]. Patients can also go through periods of emotional instability, depression patterns, and denial, disturbing their predisposition to engage in treatment [16, 19, 21].

Accompanying the escalation of the demands regarding rehabilitation, the need to have new means to evaluate patients' performance has risen. With the increase of in-home exercise training prescriptions and extra exercise recommendations without therapists' guidance and monitoring, these latter require methods to track patients' progress accurately [6, 7]. Also, conventional assessment methods are based on therapists' observation [6], and experience, which can lead to a mismatch between different therapists' evaluations and consequently distinct treatment approaches [7, 26]. These assessment tests are considerably subjective and qualitative, making them less valuable to track patients' slight progress [5, 26]. Thus, more objective assessment measures are required to trace patients' progress [5, 6, 7], understand impaired movements' strategies and characteristics [5, 26], and provide standard consensus among therapists' concerning their evaluation, applied treatments [7], and adaptation of the rehabilitation therapy regimens according to users' improvements [6].

1.2 Objectives and Contributions

Objectives

To provide a solution that allows patients to exercise their affected limb without supervision, or with partial supervision, and continue to have a planned, enjoyable, and encouraging therapy, this work proposes an affordable and accessible image-based VC capable of providing proper and engaging upper limb rehabilitation. The VC suggests three adequate upper limb exercises. It monitors the user's performance and determines the compensation patterns observed during training. It gives useful instructions for posture amelioration and encourages the patient. To gather perceptions about our VC and test the system, we aim to conduct small therapy sessions with a group of volunteers.

To assess different compensation patterns described while exercising, we explore two Multilabel Classification (MLC) approaches - a Rule-based (RB) method, which works as a baseline, and a Neural Network (NN) based approach - based on subjects' Two-Dimensional (2D) positional data. Performance assessment from 2D data automates rehabilitation programs monitorization with any device with a 2D camera, such as tablets, smartphones, or robotic assistants.

Contributions

Given the project objectives, this work provides a set of contributions:

- Assessment of motor compensation based only on the patient's 2D body pose, using a simple RGB camera, e.g., a webcam;
- Two MLC methods to detect different types of compensation patterns and, of course, the existence of compensation at all;
- Labeling of a dataset of upper extremity exercise trials video frames, indicating different compensation patterns and good-quality movements, enabling the application of supervised learning methods;
- A VC based on simple technical infrastructure composed only by a laptop with a built-in webcam;
- A VC which monitors exercise performance and establishes an interaction to keep the users' engaged;
- Collection of perceptions on the system among a group of volunteer users'.

1.3 Organization of the Document

The present thesis has the following structure:

- Chapter 1 introduces the project motivations and explains why a VC could be a valuable tool;
- Chapter 2 presents the background knowledge on the rehabilitation process commonly adopted to achieve upper limb recovery after stroke, and compensation definition. It exposes a VC's main features as an AT capable of fulfilling its purpose. It gives an overview of previous works on VC and computer-based systems to assess stroke survivors' movement quality and monitor exercise training;
- Chapter 3 describes VC's architecture and implementation. It details compensation detection methodology - feature extraction, selection, and normalization, classification methods, and result filtering;
- In the Chapter 4 we present the used dataset, labeling process for supervised learning practice, data preparation, evaluation metrics, and experiments to evaluate the compensation assessment methods. We detail the experimental procedure to evaluate the VC through exercise sessions with volunteers;
- Chapter 5 presents the experimental results, comparison between the two assessment methods, and VC behavior in an experimental setting;
- Finally, Chapter 6 exposes the project conclusions and future work suggestions.

2

Background & State of the Art

This chapter presents the background knowledge and important related work. Section 2.1 presents the common therapy process for upper extremity rehabilitation after stroke, including frequently used assessment tests aiming to evaluate patients' impairments, and defines and describes compensatory movement patterns. Additionally, we specify the desired features of a Virtual Coach (VC). In section 2.2, we display the relevant related work regarding virtual coaches and computer-based systems for upper extremity rehabilitation, assessment methods to monitor movement quality and exercise performance of stroke survivors.

2.1 Background

2.1.1 Rehabilitation Therapy Process

Rehabilitation should begin immediately after the stroke event [1]. Despite not yet existing consensus about when it should start, strong evidence suggests that motor function improvements achieve their peak in the first months after the stroke [20, 22]. In the first stage, the acute phase, therapists begin mobilizing patients' limbs with hands-on exercises to initiate damaged brain sections' reactivation. Once the patient reaches a stable medical state, the exercise training program to recover prior capabilities occurs. At this stage, patients go through extensive physical and occupational therapy sessions to improve motor function, upper extremity motion and activity, and muscle strength. Therapeutic exercises can also be conducted at home or in a community setting, with or without total supervision. After the stroke rehabilitation, it is essential to provide tools and means to the stroke survivor to keep an active and healthy lifestyle [1, 20, 21].

The upper extremity is significant for participation in daily activities. However, it often does not have the necessary attention during the rehabilitation process [16, 17, 27]. To manage the upper limbs following stroke and maximize their function, Semenko *et al.* [1] proposed a toolkit to guide therapists, based on the Canadian Best Practice Recommendations for Stroke Care. This toolkit gives a model to manage the upper extremity and guidelines to approach the process steps. Figure 2.1 illustrates the model and its fundamental elements: **Screening, Assessment and Treatment.**

To implement the best therapy strategy, therapists must perform an initial evaluation to determine the patient's level of disability, choose suitable assessment tests, set therapy goals and elect the appropriate treatments to prevent undesirable complications and get the best outcome achievable. Screening implies a set of questions and tests to identify the affected upper extremity and compare both limbs' function and

identify subluxation, pain, and edema situations. A first evaluation test categorizes the affected limb into low, intermediate, and high levels, which works as a base knowledge for the remaining process [1].

Once the upper extremity functional level is known, in the assessment phase, therapists should use the related assessment tests, such as the Wolf Motor Function Test (WMFT) and the Fugl-Meyer Assessment (FMA) [1], which are described later, to adequately evaluate motor skills: function, coordination, strength, range of motion and other parameters, as sensitivity, injury and discomfort potential. The assessment allows therapists to define the appropriate treatments and track patients' progress [28, 29]. Afterward, the therapist must define objectives related to upper limb function, to be achieved in the treatment phase [1].

Therapists select the treatments to apply according to screening, assessment results, and defined goals [1]. For upper limb recovery, Constraint-Induced Movement Therapy (CIMT) based on a task-oriented approach has proven very effective [1]. CIMT refers to regular, repeated, and intensive training of the affected limb, constraining the less affected one, promoting its participation and boosting its recovery [1, 20, 24]. The exercises must be oriented to a meaningful task for the patient, with an objective, keeping patients engaged and motivated [1, 20, 21]. The right amount of exercise intensity is not something established or agreed, however intense and prolonged exercising, as well as extra exercising, resulted in positive progress of the upper limb function [20, 22, 27]. Another relevant exercise is mental imagery. In this practice, patients receive instructions to imagine a task or activity and simulate ones' participation in it with the affected limb [1, 20]. Table 2.1 displays some examples of these exercise treatments. Along with these treatments, there are other approaches to stimulate limb's participation in various activities, improve functioning, and prevent possible injuries, such as functional electric stimulation, mirror therapy, and edema management.

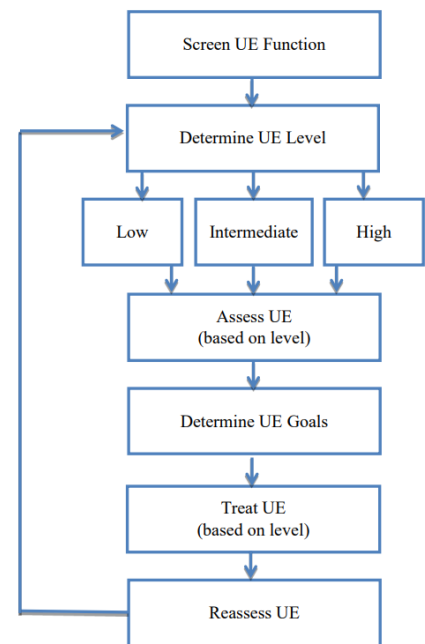


Figure 2.1: Model of the recommended process for the upper extremity management. Taken from [1].

CIMT task-related	Mental Imagery
"Use a fork / spoon to eat"	"Throwing a ball"
"Brush teeth"	"Grabbing a tissue and bringing it up to the nose"
"Drink from a cup"	"Reaching for a towel and drying the other arm with it"
"Brush hair"	"Picking up a pen and positioning it in the hand for writing"
"Turn on light switches"	"Reaching for the cup"

Table 2.1: Examples of upper extremity exercises. Adapted from [1].

Once a sequence of treatment is applied, it is fundamental to reassess patients' upper extremity performance. This reassessment will verify one's progress, achieved goals, and how the treatment should be

adapted. Therapists repeatedly apply this process in a loop until patients reach a desirable state, intending to get the best outcome possible [1]. It is also important to refer that the used rehabilitation strategy and approach must take into consideration not only patients' physical and cognitive state but also their context, psychological and emotional state, familiar and social environments [1, 20, 21].

Assessment Tests

For the upper extremity, the Wolf Motor Function Test (WMFT) and the Fugl-Meyer Assessment (FMA) are assessment tests often referred in literature [1, 6, 17, 20]. Both are performance tests based on therapists' direct-observation of high reliability and consistency, and highly used in a clinical context [28, 29].

The WMFT addresses upper extremity motor capabilities and quantifies them while performing a functional task in a certain period. Its modified version, the most widely used, is composed of a set of 17 exercises - e.g., 'forearm to table', 'reach and retrieve', and 'lift can'. It allows the evaluation of functional ability, strength, and movement quality [28, 30], mainly smoothness, movements' precision, and the existence of compensation [6]. Therapists classify each item on a 6-point scale called Functional Ability Scale [28, 30], which considers limb participation, need for assistance, task accomplishment, and movement quality.

The FMA, unlike WMFT, is not upper limb exclusive. It has 155 items divided into five domains: motor function, sensation, balance, joint range of motion and joint pain. Each task item is scored on a 3-point scale ('0: cannot perform', '1: performs partially', and '2: performs fully'). Examples of FMA task items are: 'Shoulder Elevation', 'Shoulder Abduction', and 'Elbow Extension' [29, 31].

2.1.2 Motor Compensation

In a functional and performance domain, the definition of compensation is the presence of new movement patterns derived from the adaptation of old motion patterns or substitution of these for alternative motion strategies, which might help task accomplishment [1, 17, 18, 32]. These new patterns can include the use and activation of additional or new body joints and muscle groups. Figure A.1 (appendix A) illustrates typical compensation patterns: trunk displacement and rotation, and shoulder elevation. For example, during both reaching and grasping tasks, these compensation patterns are often observed and can be considered a new strategy that allows the patient to perform the activity effectively, achieving its goals [17, 18].

Although the use of compensation mechanisms is authorized in a primary phase to enable patients' participation, these can later become damaging and obstruct patients' real motor function recovery [1, 17, 18]. Moreover, the excessive use of compensatory movement patterns is highly related to severe limb impairments. Taking again reaching and grasping tasks as examples: patients with acute limb weakness and paralysis tend to have a reduced elbow extension and excessive use of trunk forward movement and shoulder elevation [17, 18]. Given this, studies have been conducted to distinguish these compensation patterns, adopted to carry out tasks and have a higher participation rate, from real functional recovery and

old movement patterns relearning. Rehabilitation should not only focus on the capability to perform a task successfully but also on the quality of movement during task execution [17].

Among the different methods used to reduce compensatory patterns, exercises with clear instructions, appropriate feedback, and trunk displacement restriction, promoting the correct use of body joints to execute a task, pose an approach with excellent outcomes [17, 18, 32].

2.1.3 Virtual Coaches

VCs are intelligent agents designed to provide suitably and required support, training, or therapy to their users. Since our project goals include a VC for rehabilitation post-stroke, understanding and getting familiar with its predominant features is essential. Siewiorek *et al.* [11] declare that a VC should monitor the user's performance while executing an exercise or task, provide proper feedback and instructions, and encourage one to keep engaged in the activity. It has to be aware of the user's context, update its actions according to the user's performance and progress, and a caregiver or therapist could customize its features.

Recently, technological advances enabled the emergence of alternative solutions requiring a limited intervention of the therapist. Robotic and computer-based therapy providers and games can work as a tool of primary treatment or as additional support to traditional treatment strategies [3, 4, 11]. Smartphones and computer tablets allow easy and fast access to users' relevant data, and their touch screens and voice control systems ease the interaction with the user [4, 11]. Low-cost microelectromechanical sensors, RGB and depth cameras, such as Kinect [33], allow users' environment awareness and recognize and track users' behaviors [3, 4, 11]. According to user context and actions, these systems can intervene properly. Nevertheless, interactive systems for rehabilitation therapy confront many challenges [4, 11]:

- Propose and manage appropriate exercises regarding patients' diagnosis;
- Imitate therapists' functions and provide an adaptive therapy program;
- Suit to domestic environments being the less invasive as possible;
- Be affordable, and user-friendly, with a simple technical infrastructure, promoting its accessibility;
- Accomplish a reliable, accurate, and significant user performance evaluation;
- Have an interaction model to keep the users motivated and engaged in therapy and improving their physical capabilities.

2.2 State of the Art

2.2.1 Virtual Assistive Systems for Upper Extremity Rehabilitation After Stroke

This section presents a set of computer-based solutions combining desired attributes into a complex system that engages patients in upper limb rehabilitation therapy and findings on their impact on patients' recov-

ery. Unlike these works, we mainly focus on compensation behaviors, one component of upper extremity movement performance.

Adaptive Mixed Reality Rehabilitation (AMRR)

Trying to address upper extremity rehabilitation therapy challenges, such as address a significant variety of impairments and motivate patients to execute repetitive exercises promoting motor function recovery, Duff *et al.* [2] developed the Adaptive Mixed Reality Rehabilitation (AMRR). The AMRR is an interactive system that trains the user to perform repetitive reaching tasks with smart objects. It assesses users' movement quality and impairments through motion kinematic analysis. It generates audio and visual feedback based on the kinematic data to give the users information on performance. A screen displays the feedback [34].

Duff and her team [2] analyzed the functional and kinematic results from two groups of stroke survivors. One group received traditional therapy and the other held therapy with the AMRR (figure 2.2). Participants were evaluated with conventional assessment tests, such as FMA and WMFT, and reaching tasks before and after the therapy. The latter was a set of trained and untrained reaching tasks. In the trained tasks, participants reached and grasped cones in different known locations. In the untrained tasks, patients reached and touched one of nine buttons. Each button lit up in a random order (unknown location) to stimulate the reach and touch. Once the button was successfully touched, the light turned off. Both groups received the same amount of therapy time. The control group performed pegboard, bead threading, cone reaching tasks, and range-of-motion and coordination exercises. For this group, therapists provided all the feedback and verbal instructions. The AMRR group executed reaching tasks with three objects: a virtual point, a physical button, and a cone. AMRR gave real-time audio and visual feedback and cues, performance and progress evaluation, and therapy adaptation through kinematic variables. Cues provide information on incorrect arm movements and successful task completion.

As a study result, Duff *et al.* [2] verified that the AMRR group participants presented significant progress in most kinematic categories. The results revealed that therapy with AMRR enhances movement quality improvement. AMRR group progressed performance in the untrained reaching tasks, which means that subjects executed learned motor strategies in this post-therapy evaluation. On the contrary, traditional therapy had no significant influence in both trained and untrained tasks results. Duff and her team mention as a possible reason that traditional therapy is commonly directed to motor function recovery and not so focused on movement quality improvement - thus on diminishing compensatory movement patterns - and the recovery of pre-stroke movement patterns. Both groups showed identical relevant improvements in the WMFT, possibly because both had the same amount of therapy time and went through repetitive and varied exercise training. The two groups also had an increase in FMA Motor Function scores. With the results obtained with the clinical assessment tests, the authors consider that the AMRR would become better with a graphical interface to display information about the therapy procedure and motivate the participant.

Although the really positive results, Duff *et al.* [2] assume that the full added value of the AMRR only can be proved with an evaluation of its impact in long-term therapy, and with more participants to reach a larger sample of individuals with distinct impairments and characteristics.



Figure 2.2: Patient (left) exercising with the AMRR and the supervisor therapist (right). Taken from [2].

Home Arm Movement Stroke Training Environment for Rehabilitation (HAMSTER)

The Home Arm Movement Stroke Training Environment for Rehabilitation (HAMSTER) [3] is a Kinect game for upper extremity rehabilitation after stroke, aiming to provide adequate and engaging therapy at home for motor recovery. Commercially available rehabilitation games are not well suited to impaired subjects due to their complexity, fast progress, and lack of compensation patterns restriction, impacting the full recovery negatively. In opposition, the HAMSTER promises to be fully directed to stroke impaired individuals, focused on the limitation of compensation behaviors.

The games follow a specific order promoting evolution in training, going from single joint training to multiple joint training and speed control exercises. To create a direct visual interaction with the user, researchers developed a graphical interface. It displays each game's graphical representation and triggers audio effects to keep patients' interest. Figure 2.3 shows the game set up and a stroke survivor exercising.

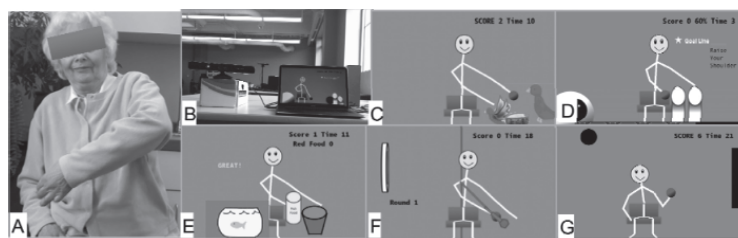


Figure 2.3: A patient exercising (A), HAMSTER setup (B), and games' graphical representation (C-G). Taken from [3].

The games restrict compensation movements by considering incorrect task accomplishment involving compensation - trunk displacements and aid with the unaffected arm - when assessing subjects' joint angles and provides error messages. Thus, in the games, impaired subjects are prone to use compensation [3].

The initial clinicians' system evaluation reported good usability and high relevance for in-home therapy since it could benefit stroke patients. Brokaw *et al.* [3], evaluated HAMSTER usability and its safety for autonomous home use. In an initial test phase, 10 stroke survivors evaluated the system in interviews. Subjects provided information about their exercise habits, rehabilitation goals, and stroke impact in their lives. According to a 5-point Likert scale, they gave answers about their motivation regarding a home rehabilitation program and compliance level if they were enrolled in it. Participants played the HAMSTER games with the affected arm for some minutes. At the end of the session, they provided feedback, a score on game usability, and opinion on the system's application as an in-home therapy system.

For deeper evaluation, one subject in a post-stroke status received a month of home therapy with the HAMSTER and an additional hand training device. Before the treatment, participant's impairments were assessed by a therapist with the FMA for the upper extremity and other popular assessment tests. Also, the impact of the stroke on the subject's life was determined. As in the HAMSTER initial evaluation, the participant provided information about one's motivation to enroll in a home therapy program and the perceived compliance in it. The researcher supervised the first therapy sessions.

The initial tests with stroke survivors revealed that stroke impairments interfered with their daily activities. All participants mentioned a large set of affected tasks, home exercises, and rehabilitation objectives. Subjects exposed an interest in therapy at home and believed they would comply in a treatment program. They considered the system suitable for home therapy. For the HAMSTER's long-term use, the subject, who suffered a severe stroke impact, expressed no interest in therapy at home but revealed that she would feel complied in such treatment. The participant reported some struggles using the HAMSTER independently and satisfaction towards the system. Additionally, she reported improvements in her arm movements with the treatment, which were supported with clinical tests [3].

Semi-automated home-based therapy

Rikakis *et al.* [4] highlight the challenges faced when developing computer-based therapy providers for home use - the need for affordable systems to administer engaging therapy with an interaction model and the lack of standard quantitative measures to assess motor performance. Moreover, the authors state as a major challenge the supply of a continuously adaptive therapy at home without therapists' regular intervention to promote patients' compliance. To provide a suitable solution for home-based rehabilitation therapy, Rikakis *et al.* [4] developed a system for in-home rehabilitation.

Based on the findings from their previous works, Rikakis *et al.* [4] developed a more proper and adaptable solution to different home environments and lower cost with a less complex technical setup. Figure 2.4 shows the system composed of a laser etched mat, six smart objects, a Kinect camera, a mini-computer, and a tablet, all of adjustable positioning. With lightweight and easy to handle components, this technical setup can be placed in any typical table. The mat works as a platform where the patient performs the exer-

cises, manipulating one or more objects. Visual markers are displayed on the mat to guide the exercise, and Kinect allows motion capture. The tablet runs a web application with therapy protocol and task instructions.



Figure 2.4: The technical infrastructure of the rehabilitation system (A), set of smart objects (B), and the combination of two objects in a task (C). Taken from [4].

Through Kinect Rikakis *et al.* [4] assessed hand activity and trunk movements providing information on major movement components: end-effector movements over space and time, hand shape for grasping analysis, trunk compensation from shoulders' Three-Dimensional (3D) positional data, and object manipulation. They aimed to assess performance errors: low speed, lengthy trajectory, object dropping, object misplacement, lack of task completion, and trunk compensation observation.

Rikakis *et al.* [4] evaluated the system through two pilot studies with unimpaired and impaired subjects. The training protocol was based on standard clinical tests analysis, such as the WMFT. Through this analysis, researchers identified 12 tasks involving object reaching, grasping, transportation, and manipulation.

The first pilot study seized to evaluate system functionality, participants' ability to use and comprehend it, and movement assessment efficiency. In the sessions, unimpaired participants repeated each task four times. In the third and fourth repetitions, they simulated performance errors. Researchers reported good functionality, without technical failures and correct motion detection. Precision and recall metrics revealed accurate performance errors' detection for the recorded videos. Participants demonstrated growing confidence using the system while stepping forward through the exercises, progressing quickly and actively.

The second pilot study, with stroke survivors, had semi-supervision. Participants followed the same protocol but without error simulation. In this phase, the team included simple and direct feedback, giving information about users' performance and encouragement. Thus, the feedback system consisted of a performance rating displayed on the tablet. For small and few performance errors, task completion errors and object misplacement, and significant errors, it displays "excellent", "very good", and "nice try", respectively.

The study demonstrated accurate performance evaluation. As with healthy participants, patients revealed growing confidence exercising with the system. They reported that task understanding could increase with audio directions in video instructions. The smart objects positively promoted reasoning and creativity during exercise performance since their usability purpose was not intuitive. Since impaired patients had difficulties executing tasks and resorted to the unaffected limb, occlusion situations in motion detection sometimes occurred. Patients found performance ratings a motivating feature for task compli-

ance. When the system provided erratic ratings, such as a good rating in occlusion situations, therapists intervened to diminish these inconsistencies. This situation emphasizes the need for robust motion detection and performance evaluation in systems for in-home use, where therapists' intervention is impossible.

Summary and Work Contributions

Previous works investigated solutions for in-home rehabilitation therapy for the upper extremity with recently available devices. They tried to give answers to rehabilitation problems, which motivated their work, and to overcome the main challenges regarding such systems' development - manage proper training given patients' diagnosis, replicate therapists' role, be affordable, adaptable to a domestic environment, easy to use, perform a reliable and accurate performance assessment, and include an efficient interaction model to keep patients' motivation and interest in therapy. Duff *et al.* [2] presented the AMRR, a system capable of produce visual and audio feedback from patients' movements' kinematic analysis. Brokaw *et al.* [3] introduced the HAMSTER, a Kinect game for upper limb rehabilitation, focused on compensation behaviors restriction, with a graphical interface displaying activities' representation and providing error messages with audio cues. Rikakis *et al.* [4] developed a Kinect-based system, with a tablet computer, a mat, and smart objects for reaching and grasping tasks. The tablet displays task instructions and direct performance ratings.

To assess their systems' usability and impact in real patients, the researchers conducted studies with unimpaired and impaired subjects with semi supervision [2, 3, 4]. These studies intensified the importance of systems suited to home use, with a simple technical setup, and a highly robust and reliable motor assessment to avoid errors in occlusion situations in independent use. They also highlight the relevance of proper interaction structures with visual and audio feedback and instructions, providing performance self-assessment. The studies' results revealed that patients' improved their motor function and movement quality. However, to determine the real impact of training with these systems in motor skill learning systems, which influences patients' daily activities, more studies need to be conducted with long-term treatment periods and more stroke-impaired individuals.

In our work, we present an interactive web application to monitor upper limb exercises. Our system provides audio, visual, and written instructions and feedback to keep subjects interested. The produced feedback follows patients' compensation movements performance. Our motion analysis is based on 2D body positional data, acquired from RGB images. This way, the technical infrastructure is only a laptop, and it tracks patients' movements with the laptop's webcam.

2.2.2 Quality of Movement Assessment During Exercise Performance

This section reveals kinematic variables relevance in describing movement quality and assess impairment. It briefly presents motion capture devices that powered these variables study. The section describes per-

minent previous works in two parts: **Kinematic Variables to Describe Post-Stroke Impairments** and **Automated Methods to Assess Movement Quality During Exercise Performance**. The first part presents articles that prove kinematic variables' utility to measure impairment and some motion capture devices' reliability and feasibility. The second part exposes previous works that relate conventional assessment measures with the proposed automatic methods, revealing their clinical relevance and reliability. Finally, we provide a section summary and present our contributions.

Given the lack of objective assessment methods, the kinematic study of 3D body positional data has been upon the table to characterize stroke patients' biomechanical behavior and develop feasible solutions [5, 6, 7, 26] to track their motor function improvement. These solutions can enhance in-home training and provide an objective and evidence-based assessment to aid therapists' evaluation [26]. Kinematics delineate body movements over space and time, giving insights about linear and angular displacements, velocity, and acceleration [5]. Joint angular motion - shoulder flexion/extension, shoulder abduction/adduction, elbow flexion/extension, and wrist flexion/extension - is commonly explored [5, 6, 7, 26] since it is highly correlated with arm degrees of freedom and encompasses relevant data about patients' movement restrictions [7]. Additionally, due to its clinical significance, trunk displacement patterns are also considered to describe motor limitations. Studies suggest that the excessive use of trunk shifts during task performance is highly related to impairment severity and is very significant when monitoring patients' progress [5, 26].

Nowadays, kinematic analysis is possible due to motion capture systems like "visual markers based sensors (optoelectronic system based on active and passive markers), robotic devices, electromagnetic sensors, and inertial sensors," [26] such as accelerometers, and cameras. Microsoft Kinect sensor is popular in this research domain due to its low cost, compactness, and unrequired use of markers for body skeleton tracking [6, 26, 33], and thus considered proper for a clinical or home setting [7, 26, 33]. To be accepted as a rehabilitation tool to assess patients' motor function, these systems should be very accurate, reliable, and acquired data should have proven clinical relevance [7, 26].

Therefore, it is appropriate and of extreme importance to:

- Prove motion capture systems reliability and feasibility, which contribute to the acceptance of these systems in the rehabilitation context [5, 6, 7, 26];
- Explore different movement categories, identify kinematic variables and determine which better describe movement restrictions, differentiating healthy subjects from post-stroke patients [5, 26], and thus prove kinematic variables' utility and relevance [5, 6, 7, 26];
- Find a relation between these quantitative metrics and conventional qualitative assessment to help interpret kinematic variables and provide additional evidence of their clinical pertinence [6, 7].

Kinematic Variables to Describe Post-Stroke Impairments

To identify valuable and precise kinematic variables to describe upper limb motor function, and discriminate between healthy subjects from stroke survivors, Murphy *et al.* [5] and Ozturk *et al.* [26] conducted a study with participants from both groups. These works present a factor analysis resorting to Principal Component Analysis (PCA), determining which kinematic variables better describe motor impairment and how they relate with each other in a meaningful way. In addition to distinguishing between healthy participants and subjects in a post-stroke status, Murphy *et al.* [5] aimed to differentiate distinct levels of impairment among stroke patients, mild (FMA scores of 58-64; less severe), and moderate (FMA scores of 39-57; more severe).

Murphy *et al.* [5] instructed participants to perform the “drinking from a glass” task without movement constraint, enabling the occurrence of any movement pattern such as compensation. This task had five stages: reaching the glass, moving it towards the mouth, drinking, placing the glass back to its initial position, and return the hand to its starting point. Participants were initially seated near a table with the drinking glass on top, surrounded by five motion capture cameras and with retroreflective markers on relevant body joints to compute their 3D coordinates (figure 2.5). The markers were set on the hand and elbow of the arm under evaluation, both shoulders, thorax, face, and also on the glass. Throughout the experiment, researchers used kinematic variables to describe movement strategy, initial motion effort in reaching, movement smoothness, joints’ angular behavior, trunk compensation, and inter-joint coordination.

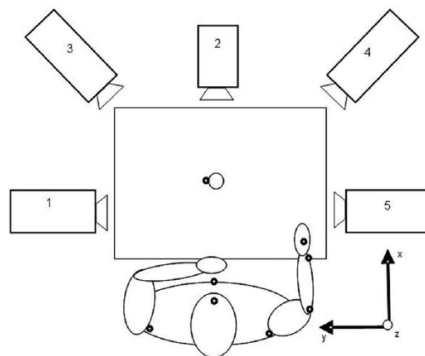


Figure 2.5: Setup for the drinking task with the 5-camera capture system: the participant’s arm is in its initial position, and the black dots represent the retroreflective markers. Taken from [5].

To describe movement strategy and initial effort in reaching, the authors calculated the time and percentage of time to peak and first peak hand and elbow joint velocity. Murphy *et al.* [5] quantified movement smoothness by repetitive hand acceleration and deceleration - number of movement units (NMUs). Joints’ angular motion described maximal elbow extension and shoulder flexion during reaching and maximal shoulder abduction and flexion during drinking. Thorax displacements, regarding its initial position, determined trunk compensatory behaviors. Shoulder and elbow inter-joint coordination was analyzed through cross-correlation. The authors also identified movement onset and offset and defined different task phases. Onset and offset corresponded to the time that hand velocity achieved 2% of its peak velocity in reaching

and returning phases. The beginning of the glass lift and the returning phase corresponds to the moment that glass's tangential velocity surpasses or goes beneath a 15 mm/s and 10 mm/s, respectively. The drinking step agrees with an increase or decrease of the face-glass distance in steady-state.

Having identified all the relevant measures in task execution, Murphy *et al.* [5] performed a PCA, which provides the number of necessary kinematic variables (components) to seize kinematic variance over time. The PCA's correlation matrix gives insight into how the variables meaningfully cluster together and quantify elements of the same dimension (factor). Researchers selected a group of variables, which clustered together into two major factors identified within the result analysis: movement time and smoothness, compensation patterns (arm flexion and elbow extension in reaching, and arm abduction and flexion in drinking) and inter-joint coordination. Trunk displacement was also extracted due to its clinical relevance.

With an independent-samples *t*-test or Mann-Whitney *U*-Test, Murphy *et al.* [5] compared both groups of healthy and chronic stroke participants. The results validated their clinical significance. Briefly, they showed that stroke survivors revealed slower and oscillatory movement profiles (many NMUs). They had lower inter-joint coordination, mainly among moderate impaired, expressing difficulties moving the elbow and shoulder simultaneously in a continuous movement, as observed in healthy participants. Stroke patients manifested pronounced compensation patterns in the drinking - considerable shoulder abduction and elevation angles - and reaching phases - reduced maximal elbow extension and more evident forward trunk displacement.

Similarly, Ozturk *et al.* [26] determined which kinematic variables better describe motor impairment and differentiated healthy subjects from individuals in a post-stroke status. Ozturk *et al.* [26] used a Microsoft Kinect sensor to acquire joints' 3D pose data and pursued to find evidence of its easy usability, robustness, and reliability to delineate and monitor motor limitations in a clinical context. In their work, participants' motor ability was previously assessed with WMFT (section 2.1.1). Participants had to perform a reaching movement from a neutral position to a random point in a table placed next to them, executing an elbow extension with the arm under evaluation.

From 3D positional data, illustrated in figure 2.6, Ozturk *et al.* [26] selected eight joints to describe movement - spine shoulder, spine mid, and both shoulders, elbows, and wrists. Then, to overcome noise, they filtered the raw signal through singular spectrum analysis. To evaluate motor control and the wrist's ability to follow a straight path to its reaching point, the authors calculated the wrist's speed profile and curvature index to quantify its deviation from the path. To describe movement smoothness, the researchers used the spectral arc-length metric based on the Fourier magnitude spectrum. Trunk displacement, calculated from the 3D data of the spine shoulder joint, and inter-joint coordination identified compensation patterns. Shoulder and elbow coordination was assessed

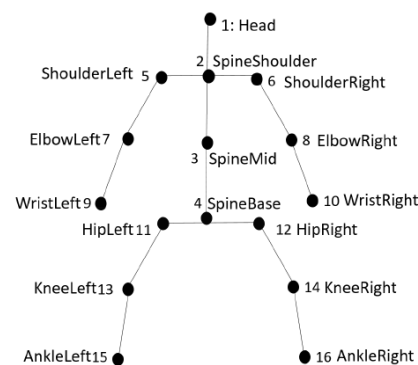


Figure 2.6: Joints acquired with a Kinect Sensor. Taken from [6].

by the temporal coordination index and through PCA. PCA was used to convert several correlated variables into a reduced number of principal components, which comprise the main kinematic variability, describing movement accurately.

In their analysis, Ozturk *et al.* [26] observed that the metrics describing smoothness and straightness were pretty close to each other. They recognized that stroke survivors had lower maximum speed and more compensatory behaviors with pronounced trunk displacement. The authors also observed more complex elbow movements in patients revealing disturbance in inter-joint coordination. Additionally, with PCA, researchers verified that healthy subjects and stroke survivors clustered together into two separate groups and inter-joint coordination reflected the main difference between their physical capabilities. For stroke patients, they compared the WMFT with the kinematic metrics projected on the principal components. They noticed that the resultant scores were able to distinguish different levels of impairment. By demonstrating that this analysis assesses accurately motor impairment, Ozturk *et al.* [26] also prove that the Kinect sensor is a reliable and feasible system and thus useful in a clinical setting.

Automated Methods to Assess Movement Quality During Exercise Performance

Aiming to develop an automated method to assess movement quality during exercise performance, to standardize clinical impairment assessment, and apply it to home-based therapy, Olesh *et al.* [7] conducted a study with a group of chronic stroke patients. Participants had to perform several trials of 10 unconstrained arm movements, with the affected and non-affected limbs, from tests such as FMA (section 2.1.1). To record participants' movements and acquire positional data, the authors simultaneously used a high-definition Samsung camera aided by LED markers in relevant joints (standard system) and a Microsoft Kinect sensor. Figure 2.7 shows the studied movements and the body points tracked by both systems.

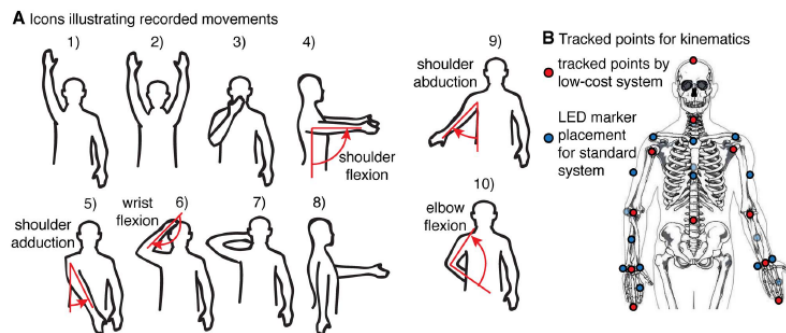


Figure 2.7: A. Illustration of the analyzed movements; red lines and arrows represent movement direction and joint angles. B. Keypoints detected by both motion capture system, Kinect (red dots) and LED markers (blue dots). Adapted from [7].

Olesh *et al.* [7] filtered the recorded data with a Butterworth low-pass filter, calculated joints' angles (shoulder, elbow, and wrist), and completed a temporal alignment of the analogous movements of affected and non-affected arms for all movements. To quantify movement quality accurately, they estimated the

minimum number of movement repetitions needed for precise motion capture. This way, the authors could have their evaluation supported by many movement trials, contrary to therapists' traditional qualitative assessment, based only on a unique movement. To do so, the authors bootstrapped the data to estimate the errors of averaging one or more trials of the same movement for each exercise and participant. The minimum number of movement repetitions of the same movement necessary to evaluate movement performance accurately corresponds to the number of trials with a lower bootstrapped error.

The authors applied PCA to the averaged unaffected arm joints' angles across trials and determined principal components. Based on the principal components, Olesh *et al.* [7] reconstructed affected arm joint angles' temporal profiles, which revealed that the chosen components described very well the kinematic behavior. When comparing the reconstructed profile with the affected limb's original profile with the coefficient of determination, they determined how the unaffected arm kinematic pattern can describe the affected arm's kinematic behavior. Thus, the coefficient of determination served as a score to quantify impairment.

To determine quantitative scores' reliability, a group of therapy students performed FMA assessment of each recorded movement. The authors used linear regression to relate quantitative scores from both motion capture systems and the Pearson correlation coefficient to relate quantitative and qualitative scores. Regression was also used to develop a performance decoding model, which converted the obtained quantitative scores into the students' corresponding qualitative scores.

Olesh *et al.* [7] verified a strong correlation between the scores from the standard system motion data and Kinect data, meaning that both systems are equivalent. The researchers also proved Kinect's clinical feasibility estimating that is needed nearly one movement trial to compute accurately kinematic metrics, which relates to therapists' qualitative evaluation. Additionally, the analysis of the root squared errors between the joints angles, calculated from systems' data, determined Kinect's accuracy in motion detection. Although Kinect's sensitivity to noise, the quantitative assessment with PCA was successful since the quantitative scores strongly correlated with qualitative scores. Olesh and her team concluded that this automated method is a good tool to track patients' progress during unsupervised therapy.

Identically, Lee *et al.* [6] explored automated methods to assess movement quality in rehabilitation therapy. This work seizes to overcome the difficulty found by therapists to interpret kinematic variables, promote in-home rehabilitation, and provide patients with a performance score, enhancing their motivation and engagement during exercise practice. Lee and his team proposed a method to evaluate stroke rehabilitation exercises with Machine Learning (ML) algorithms and a threshold model providing a quantitative and qualitative assessment. To validate their approach, the authors collected a dataset of 3D positional data, acquired with a Kinect, in an experiment with 11 healthy subjects and 15 post-stroke survivors.

After a discussion with therapists, Lee *et al.* [6] identified three performance components to provide more comprehensive evaluation and feedback instead of only a global performance score. As in previous works, sets of kinematic variables characterize these components. Table 2.2 introduces them.

Performance Component	Description
Range of Motion (ROM)	Evaluates task accomplishment performing a particular movement pattern and represents a particular joint activity
Smoothness	Investigates the presence of trembling motion patterns and abrupt transitions
Compensation	Verifies the existence of compensatory patterns, commonly used to achieve target positions during upper extremity training, such as torso inclination and shoulder elevation

Table 2.2: Performance components under evaluation. Adapted from [6].

Lee *et al.* [6] explored three upper extremity exercises similar to usual daily activities. Exercise 1 (E1) is 'Bring a Cup to the Mouth', in which the participants simulate holding a cup and brings it to the mouth as drinking. Exercise 2 (E2) is 'Switch a Light On', in which the participants simulate turning on a light switch. In exercise 3 (E3), 'Move a Cane Forward', participants move a cane forward and move it back to its initial position. The team selected these exercises due to their characteristics, and high relation with FMA and WMFT items (section 2.1.1). Table 2.3 relates the three exercises and the task items from FMA and WMFT.

Exercises	FMA	WMFT
E1. Bring a Cup to the Mouth	Elbow Extension	Lift can
E2. Switch a Light On	Shoulder Flexion	Hand to a box
E3. Move a Cane Forward	Elbow Extension	Extend Elbow

Table 2.3: Upper extremity exercises and respective FMA and WMFT tasks items. Adapted from [6].

Figure 2.6 illustrates the joints detected by the Kinect sensor. Lee *et al.* [6] applied a moving average filter with a window of five frames to overcome noise in the acquired keypoints with Kinect. With the filtered data, Lee *et al.* [6] processed joints' position and normalized them to diminish any physical variabilities, such as the patient-camera distance, body parts dimensions, and subject's placement in the image. For the ROM performance component, the authors computed limbs' joint angles, normalized relative trajectory, and normalized projected trajectory. For smoothness, they determined wrist and elbow speed, acceleration, and jerk. They also used normalized and mean arrest period ratio speed and jerk, and zero-crossing ratio [6] Table 2.4 presents the kinematic features for compensation.

Compensation Normalized Features	
Joint Angle	Angles between vectors defined by three joints: $ShoulderSpine_{init} \rightarrow SpineBase_{init} \rightarrow ShoulderSpine$, $Shoulder_{init} \rightarrow ShoulderSpine_{init} \rightarrow Shoulder$, $Hip \rightarrow Shoulder \rightarrow Elbow$
Projected Trajectories	Difference between the current and initial distance of two joints in X, Y, Z axis: $Head_{init} \rightarrow Head$, $SpineShoulder_{init} \rightarrow SpineShoulder$, $Shoulder_{init} \rightarrow Shoulder$

Table 2.4: Normalized features for the compensation performance component. Joints' names are in accordance with figure 2.6. Adapted from [6].

Lee *et al.* [6] used as baseline regression and multi-class classification methods, which predict categorical performance score (e.g. '0: no movement perform', '1: limited movement', '2: normal movement')

as the FMA. The authors proposed approach, 'BinToMulti', is composed of a threshold model with binary classification to evaluate movement quality. Threshold models consider that each category has an inherent value that distinguishes and separates different categories. Threshold models learn a function capable of predicting these inherent values (thresholds) and estimate the categorical response based on them. Figure 2.8 shows that binary classifiers determine correct and incorrect movements (qualitative assessment). The threshold model transforms the binary classification confidence score into a performance score (quantitative assessment). To evaluate the proposed approach, the authors compared it with the baseline methods.

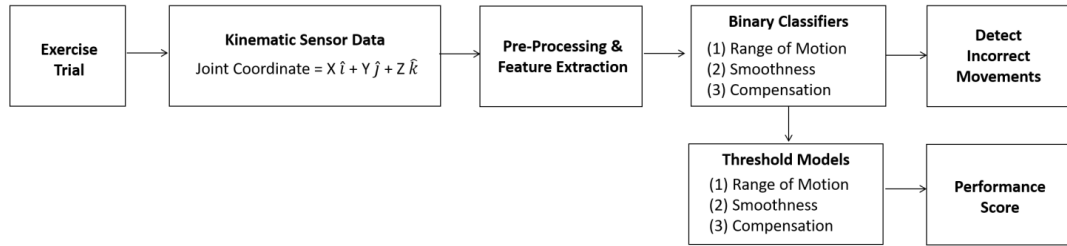


Figure 2.8: Flow diagram of the proposed method. Taken from [6].

The team explored different ML algorithms, non-sequential - Decision Trees, Linear Regression, Support Vector Machine, and NN - and sequential models - Long Short Term Memory (LSTM) Network -, with cross-entropy loss for classification and Mean Squared Error (MSE) loss for regression. Each algorithm was trained for each performance component and each exercise. For the NN, researchers explored the hyperparameters from table 2.5. The LSTM model was chosen given that therapists assign a score after observing patients' entire movement. Given this, the chosen architecture for the LSTM was many-to-one, which produces a unique output for a sequential set of frames.

Architecture	one to three layers of {16, 32, 64, 128, 256, 512} hidden units
Learning Rate	Adaptive with {0.005, 0.001, 0.01, 0.1} as initial learning rate
Activation Function	'Relu'
Solver	'AdamOptimizer' with mini-batch size of 5

Table 2.5: NN Explored Hyperparameters. Adapted from [6].

To validate the proposed method, the authors needed to create ground truth scores. Two therapists (primary and secondary) assessed post-stroke participants' recorded movements with FMA and then discussed their evaluations to generate ground truth scores. Since, in a real context, a patient is only evaluated by a therapist, Lee *et al.* [6] only tried to reach a score agreement with the primary therapist's assessment scores. The authors determined the agreement level of all methods - regression, multi-class classification, and 'BinToMulti' - and then compared them. To do this evaluation, the authors handled Leave-One-Subject-Out (LOSO) cross-validation. It consists of using all the post-stroke patients' data, except one, to train the models and use the left-out patient's data to test the method.

Lee *et al.* [6] approach, 'BinToMulti', had better performance, specially with the NN. It performed better compared with other methods and secondary therapist's assessment. Researchers used linear regression to determine the relationship between the predicted and FMA scores. It showed a high relation between both scores, concluding that it is possible to estimate FMA scores with this approach. The authors conclude that their method is feasible to replicate closely primary therapist's assessment and to estimate FMA scores.

Summary and Work Contributions

With the growth of in-home rehabilitation, the need for objective metrics to track patients' progress over time enhanced the development of quantitative and automated methods to evaluate movement quality. These methods can bring more consensus between therapists' assessments and provide standard knowledge to aid in the selection of rehabilitation treatment. Explored objective automatic assessment methods are based on body joints' kinematic study, which significantly describes motion patterns. This study is possible due to technological advances concerning motion capture devices. These are more suitable for clinical or home settings, such as Kinect. For objective assessment to gain clinical acceptance, researchers need to prove motion capture systems and methods' reliability and feasibility.

Murphy *et al.* [5] and Ozturk *et al.* [26] identified the kinematic variables that best describe motion patterns and distinguish healthy participants from stroke survivors. Olesh *et al.* [7] and Lee *et al.* [6] provided automated methods to produce assessment scores highly correlated with FMA scores. These works explored the shoulder abduction and elevation angles, and trunk displacement from its initial position to describe motor compensation. The authors verified that stroke survivors demonstrate lower elbow and shoulder inter-joint coordination, more severe shoulder elevation, and pronounced trunk displacement. Lee *et al.* [6], whose work we mainly follow, also explored *head*, *spine*, *shoulder*, and *shoulder* joints (figure 2.6) projected trajectory, which is the distance of these joints to their initial position, at each timestamp.

Despite relevant analysis and great results, these works do not provide a detailed assessment and feedback about different types of compensation. They only give a global performance score and indicate the existence of compensation through joint angles, which describe it. Also, 3D analysis is still dependent on sensors capable of acquiring 3D pose data and implies higher processing complexity. We conduct a 2D analysis to assess compensation using a simple RGB camera in our work. We explore Multilabel Classification (MLC) to determine different compensation patterns, such as trunk displacements and shoulder elevation, allowing the Virtual Coach (VC) to provide the user comprehensive and directed feedback.

3

Methodologies

In this chapter, we present the methodologies used to develop the Virtual Coach (VC) and assess motor compensation. First, in section 3.1, we describe the VC intelligent agent, its features and architecture, and the User Interface (UI) developed to establish an interaction with the user. Second, in section 3.2, we detail all the workflow to assess compensation during exercise performance. This workflow includes feature extraction, feature selection, data normalization, classification, and results filtering. Regarding the classification phase, we introduce the Multilabel Classification (MLC) domain, inherent issues, and approaches to learning from a Multilabel Dataset (MLD). Additionally, we present three different algorithms to perform MLC to determine the distinct compensation patterns.

3.1 Virtual Coach

In this work, we propose a computer-based Virtual Coach (VC) to monitor upper extremity exercise training. The VC proposes three appropriate exercises (table 2.3) and monitors user compensation behaviors during their execution. First, it verifies if the patient is correctly positioned to enable motion capture. Once the user is well placed, the exercise begins and it starts evaluating one's movements. It gives verbal and visual instructions about the exercise and target position the user has to reach. When the patient exhibits compensatory, the VC suggests posture improvement. It also encourages movement repetition and praises the user when the target position is reached.

This section details all the considerations and procedures to develop our VC for upper extremity rehabilitation. We introduce project requirements, define the coach intelligent agent and its behavior in the environment where it is inserted, and describe our system architecture and implementation details.

3.1.1 Requirements

In addition to previous works' findings, to define the requirements for our VC we followed physical and occupational therapy sessions with real patients and sought rehabilitation professionals' advice. We learned that patients require clear and detailed task instructions and even an exercise demonstration. Patients with cognitive deficits, such as reduced attention or slowed functional execution, need extra and repetitive directions and constant help redirecting their focus to the exercise. The exercise approach needs to be aligned with patients' physical and cognitive state to avoid injury or accidents, like falls. Patients may fear

falling or hurting themselves, limiting their participation in the exercise, or slows down execution. During the sessions, patients' necessity for motivation is permanent. Additionally, we were able to record some of the instructions, feedback, and encouragement that therapists gave. From our meeting with the therapists, a series of potential features for the VC were discussed. Patients should be able to watch themselves while exercising, like when they are looking at a mirror. In their displayed image, the VC should provide visual markers to guide the patient to reach the desired positions and indicate compensation. Additionally, since repetitive movements and tasks are fundamental, the system should stimulate movement repetition.

Given these discoveries, we list a set of requirements that our system should include:

- Present an exercise demonstration;
- Propose adequate exercises;
- Give patients the possibility of exercising sitting in a chair, contributing to their confidence and safety;
- Display of the patient's image while exercising as if looking at a mirror;
- Provide clear and repetitive audio instructions, cues for posture correction, encouragement, and suggest task repetition;
- Display visual markers indicating the arm target position and the existence of compensation

3.1.2 The Intelligent Agent

In our work, the Virtual Coach (VC) perceives its environment completely through an RGB camera (*sensor*) and gathers image data on patient's body movements, more precisely, user's arms and torso motor patterns. We can describe the VC environment as a regular space, such as a domestic or hospital room, with a patient performing a set of upper extremity exercises, maybe accompanied by a caregiver. Given an image, the coach builds an idea of the environment *state*. The agent establishes an interaction with the user giving qualitative feedback on performance - concerning compensation and target reaching - to keep one engaged. Section 3.2 details the motor performance assessment methods proposed in this work. VC actions are managed through a UI, described in section 3.1.4, which has the role of an *actuator* to interact with the user.

We define our VC as a *Simple Reflex Agent* [35]. The agent selects the *action* to take based on the current environment's *state*, previous state, and time interval, through a condition-action rule. The set of condition-action rules that holds the action selection procedure compose the agent's built-in knowledge.

The VC action selection procedure is considered a decision process problem. We detail this process introducing the set environment *states*, the coach *actions* and respective condition-action rules, and state transition conditions. Table 3.1 presents the system space state, S , and a description of each state, which are intrinsically correlated with patients' motion behaviors.

Given the states, it is fundamental to detail what we mean by patient correct position, movement beginning, target position, and how these different stages are determined.

State Space S	Description
$o \rightarrow out$	Patient not placed in the correct position
$i \rightarrow in$	Patient placed in the correct position
$e \rightarrow exercise$	Exercise and movement trial beginning
$n \rightarrow normal$	Normal movement pattern
$tr \rightarrow trunk rotation$	Patient rotates the torso as compensation movement
$se \rightarrow shoulder elevation$	Patient elevates the shoulder as compensation movement
$td \rightarrow trunk displacement$	Patient displaces the trunk as compensation movement
$tg \rightarrow target$	Patient reaches the target position

Table 3.1: Environment States S .

A patient is correctly positioned when has the hips, shoulders, and head visible. A rectangle in the patient's image center bounds the correct position, concerning a distance of approximately $2.5 m$ between the subject and camera. When the patient's relevant joints are outside the rectangle, the system state is *out*. Individual correct positioning in the acquired image enables accurate motion capture and motor performance assessment. Once the patient is well located, the system transits to state *in*.

Every single time the user executes a movement, it is considered a movement trial. To distinguish different movement repetitions, we track the patient's wrist displacement regarding its initial position. Once the wrist moves significantly away from its starting point, a new movement trial begins. When the subject returns the wrist to its initial position, the movement trial ends. Euclidean distance defines the distance between the wrist current and initial positions.

Once the exercise begins, the VC indicates the exercise target position, i.e., the position that the hand should reach. This position is conveniently determined concerning the patient's positioning in the image and arm length. When the patient reaches the target, the system state is *tg*. This is monitored by the euclidean distance between the patient's wrist and the target.

Table 3.2 presents the VC actions, corresponding condition-action rules, and action description. VC actions are conducted through the UI, described later in section 3.1.4. The actions include:

- Display of position markers - the rectangle indicating patient's valid positions;
- Display of the target position marker;
- Display of compensation indicator markers - shoulder and trunk markers;
- Audio speech and respective subtitles - instructions, suggestions, encouragement, and praise.

Figure 3.1 describes the system's state transition. For each video frame acquired with the RGB camera, there is a corresponding state. When the patient is outside the rectangle defining correct body positioning, the system state is '*o*'. Every state can transit to '*o*' state, except state '*e*', as depicted by the green dashed arrows. Once the patient is correctly positioned, the system transits to '*i*' state, in which it must remain for 60 frames before transiting to '*e*' and with the position rectangle in green. State '*e*' defines exercise and

Action Space \mathcal{A}	Condition-Action Rules	Action Description
$pos : 'position'$	$state_{prev} = o, state = o,$ $time > th_{pos}$	Patient not well-positioned for $time > th_{pos}$: VC suggests body repositioning; position rectangle in red color.
$ins : 'instruction'$	$state_{prev} = o,$ $state = i$	Patient well-positioned: position rectangle in green color; VC gives exercise directions.
$mar : 'marker'$	$state_{prev} = i, state = e$	Exercise begins: VC displays target position marker (green).
$tri : 'trials'$	$state_{prev} = S/\{i\},$ $state = e$	Patients stops moving: VC proposes movement repetition.
$tar : 'target'$	$state_{prev} = e,$ $state = n$	The VC starts evaluating patient's performance and asks one to reach the target position.
$enc :$ 'encouragement'	$state_{prev} = \{tr, se, td, n\}$ $state = \{tr, se, td, n\},$ $time > th_{tg},$	Patient takes too much time reaching the target position: VC encourages patient to reach the target.
$con :$ 'congrats'	$state_{prev} = \{tr, se, td, n\},$ $state = tg$	Patient reaches the target: VC praises the patient; target position marker in blue color.
$trr :$ 'trunk rotation'	$state_{prev} = \{tr, se, td, n\},$ $state = tr$	Patient describes trunk rotation: VC suggests posture correction;it displays trunk compensation marker (red).
$she :$ 'shoulder elevation'	$state_{prev} = \{tr, se, td, n\},$ $state = se$	Patient describes shoulder elevation: VC suggests correction;VC displays shoulder compensation marker (red).
$trd : 'trunk$ $displacement'$	$state_{prev} = \{tr, se, td, n\},$ $state = td$	Patient describes displaces the torso: VC suggests posture correction; VC displays trunk compensation marker (red).

Table 3.2: Agent Actions \mathcal{A} , respective condition-action rules, and action description.

moment trial beginning, and thus performance evaluation, $t = 1$ (t denotes the number of frames included in the performance evaluation process). State 'e' transits directly to state 'n', which corresponds to observed normal movement patterns. Once the movement evaluation begins, the system jumps between states corresponding to performance classification results - states 'tr', 'se', 'td', and 'n'. Red dotted bi-directional arrows represent these transitions. When the patient reaches the target position, the system transits to state 'tg' (blue dotted arrows). When the patient moves the hand from this position, the movement evaluation is resumed. When the patient is finishing a movement trial, with the wrist returning to its initial position, the motor pattern is considered 'n' (rest position). At this point, the system shifts to state 'e', initiating a new movement trial.

3.1.3 System Architecture

To establish an interaction between the user and the Virtual Coach (VC), we develop a UI through a web application, whose implementation is detailed in section 3.1.4. In this section, we describe the architecture of the entire system. First, we introduce the web app pages and their purpose. Second, we describe the architecture of the VC.

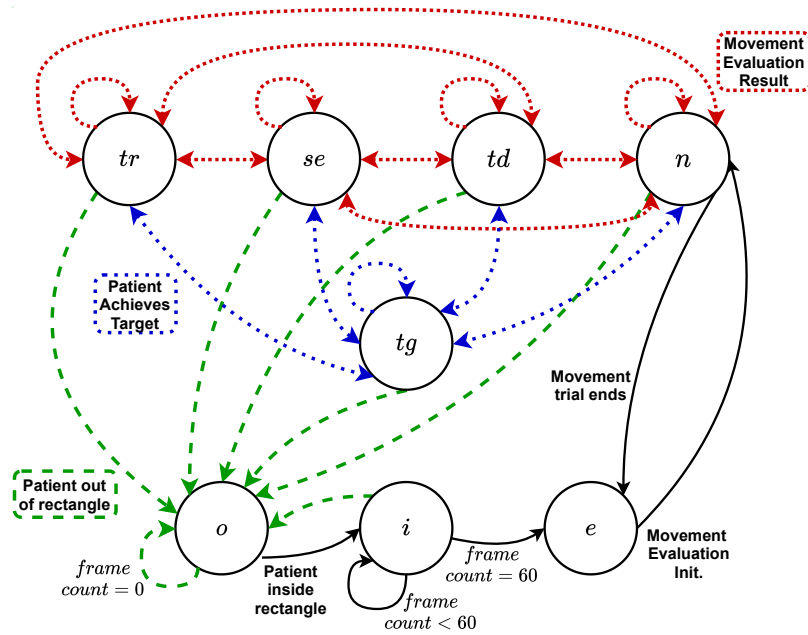


Figure 3.1: State transition diagram.

Figure 3.2 illustrates the web application pages and the connection between them. The **Init** page has the role of application opening screen, and connects directly to the **Menu** page. In the **Menu**, the user chooses the training exercise and can watch each exercise demonstration in the **Demo** page. Once in the **Demo** page, the user can return to the **Menu**. In the **Menu** page, along with the exercise preference, the user should select the affected side due to stroke, which is the one that needs to be exercised and go under performance evaluation. After selecting these two parameters, the exercise and the affected side, the user can skip to the **Main** page where the VC operates. In the **Main** page the user can rewatch the selected exercise demonstration in the **Demo** page and then return back to the **Main** page.

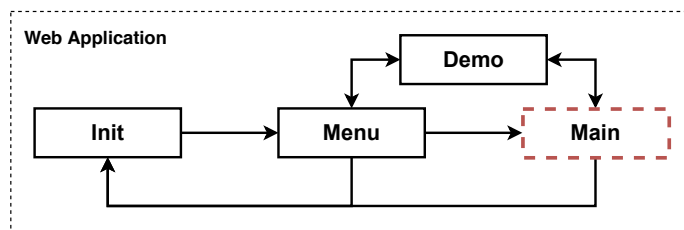


Figure 3.2: Web application pages.

Figure 3.3 details the VC architecture, operating in the **Main** page. The VC operates in a loop cycle concerning the video acquired with the RGB camera. Each cycle corresponds to a single frame acquisition and its analysis. In figure 3.3, the blocks with dashed and dotted boundaries represent the coach visual and verbal cues, respectively. The block with solid boundaries correspond to the VC main functions. Table 3.3 describe the diagram function blocks.

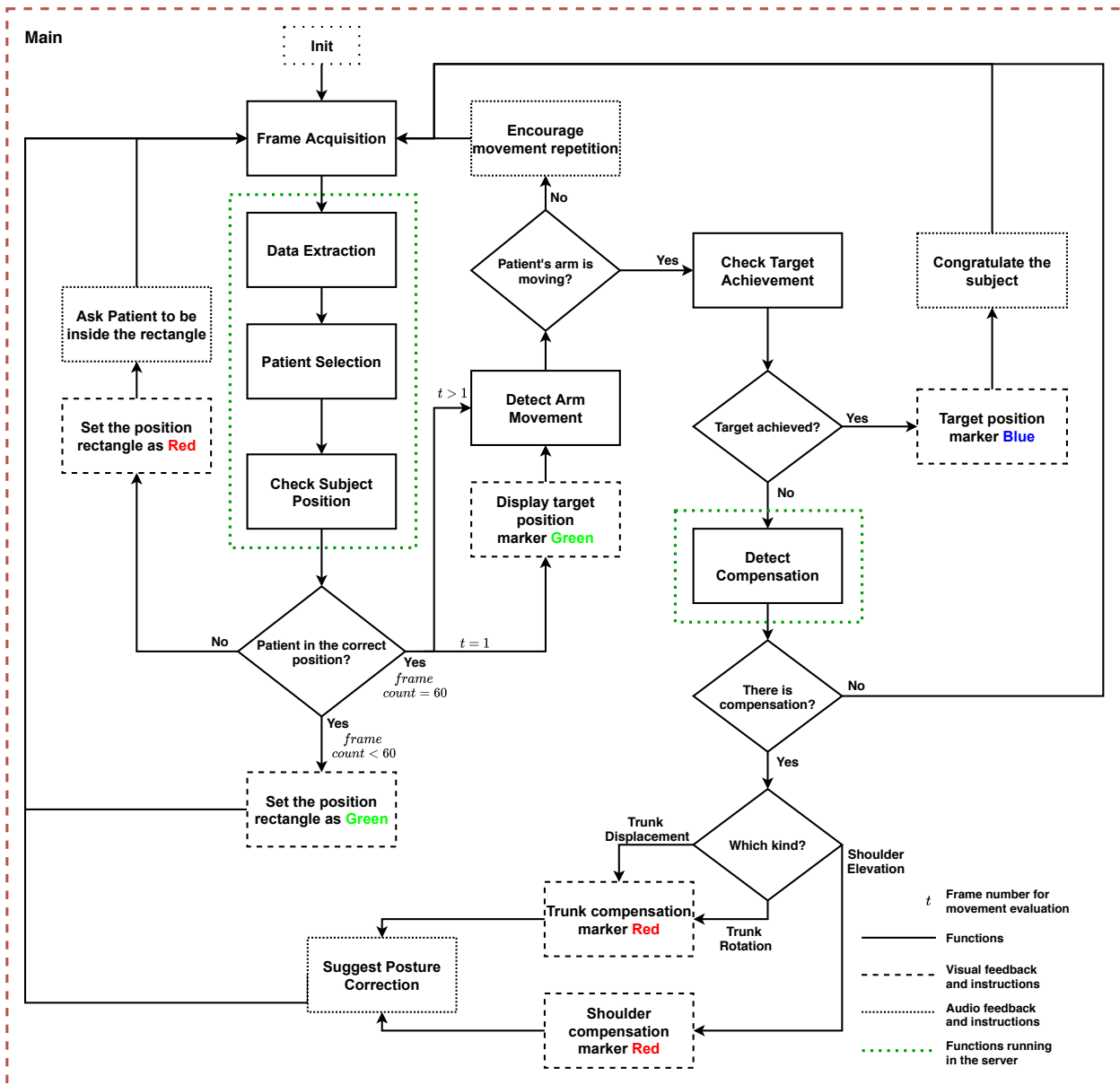


Figure 3.3: Virtual Coach workflow.

Function Blocks	Description
Frame Acquisition	Through the RGB camera the system acquires video frames
Data Extraction	Given the acquired image, the system extracts the observed subjects' body joints 2D positional data (described in section 3.2.1)
Patient Selection	In a multi-person setting (a therapist or caregiver might accompany the patient), the VC selects the patient, which should be positioned in the image center (section 3.2.2)
Check Subject Position	The VC verifies if the patient is correctly positioned as described before in section 3.1.2
Detect Arm Movement	Through the euclidean distance between patient's wrist and its initial position the VC determines when a movement trial begins or ends
Check Target Achievement	The VC checks if the patient achieved the target position through the euclidean distance between the wrist and the target
Detect Compensation	While the patient is exercising the VC continuously evaluates one's performance through the methods described in section 3.2 and through RB classification method

Table 3.3: Diagram Function Blocks (figure 3.3).

3.1.4 User Interface

As a Virtual Coach (VC) actions actuator to establish a proper interaction with the user, we developed a web application with a User Interface (UI). To develop our web application, we used Flask microframework [36]. Flask is written in Python and depends on the Jinja template engine and Werkzeug WSGI toolkit. We created a dynamic web application to run locally in a personal computer with four web pages: **Init**, **Menu**, **Demo**, and **Main**. As any web app, it handles HTTP methods when accessing the URLs, such as GET and POST. Along with the functions that respond to URL requests, Flask uses templates to render HTML containing static data, which the browser will display. Additionally, static CSS files add style to the HTML layout. Next, we present and describe our web application web pages. Figure 3.2 already details the connection between the pages. Figure 3.4 presents our app **Init** page, which has the role of opening page and redirects the user to the **Menu** page.

In the **Menu** web page, shown in figure 3.5, the user can choose the exercise and affected side. This option selection is done through a HTML form with POST request method. When the user tries to submit the form without the required fields filled, the app pops up an error message asking the user to fill all the fields (figure A.2(a)). With the fields filled, the user can submit the form and access the **Main** page. For both **Menu** and **Main** pages, when the user tries to exit the app, it asks for confirmation (figure A.2(b)). Also in **Menu**, the user can watch each exercise video demonstration accessing the **Demo** web page. Figure A.3 presents each exercise **Demo** pages. Once in the **Demo** page the user can return to **Menu** page.

The most important web page is **Main**, in which the VC operates. On this page, the user can watch the selected exercise demonstration again, accessing the exercise's **Demo** page and then return to **Main** page. On **Main** page, the system acquires the image captured with the RGB camera and streams video frames. The displayed image is horizontally flipped to show up to the user as a mirror.

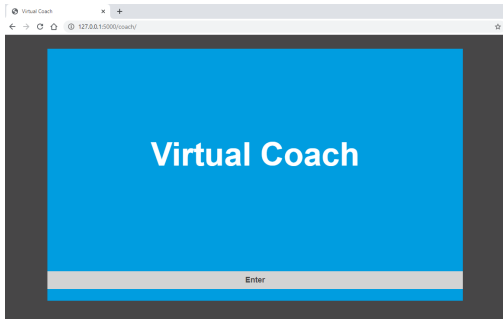


Figure 3.4: Virtual Coach **Init** web page.

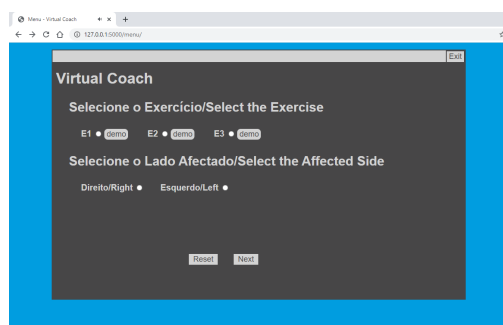
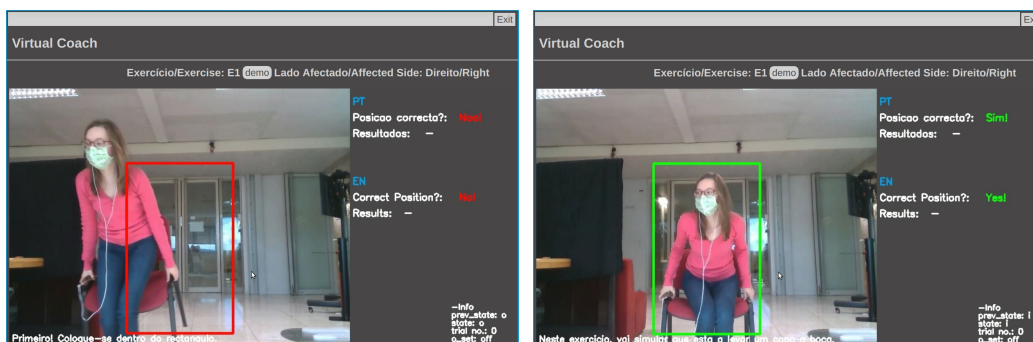


Figure 3.5: Virtual Coach **Menu** web page.

At this stage, image acquisition and any image manipulation process is conducted utilizing the OpenCV library. Image manipulation includes the display of markers, speech subtitles, qualitative results, and information regarding system state and motion monitoring. Figure 3.6 shows the position rectangle when the system is in state *out* and *in*. Figures 3.7 and A.4 (appendix A) present each exercise target position and how this marker looks like when is reached, respectively. Figure 3.8 shows the displayed compensation markers. These markers indicate compensation existence and the body part requiring pose correction.

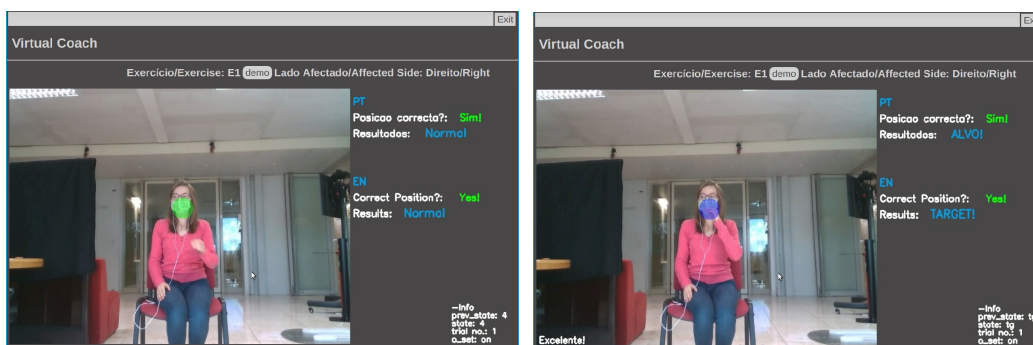
As detailed previously in section 3.1.3, in figure 3.3, the main functions to process the acquired image (green dotted bosh), to determine the existence of compensation and track patient's movements are handled in a remote server, accessed via WiFi, for faster processing and result extraction, since the laptop in which the web app runs has low processing capacity to run all the required tools for image processing. Our Flask application sends, to a simple Flask application in the server through Python `requests` library, an HTTP request with a `json` request containing an acquired image at a time and extra metadata, such as the exercise, the affected side, and the system current state.

The VC speech was generated with a text-to-speech tool available online. We chose a female voice with a neutral tone. In appendix A, we present the VC speech for every web app page or action. For each situation, we created different sentences providing the user with the same information to avoid boredom.



(a) Incorrect position, state *out* - red rectangle. (b) Correct position, state *in* - green rectangle.

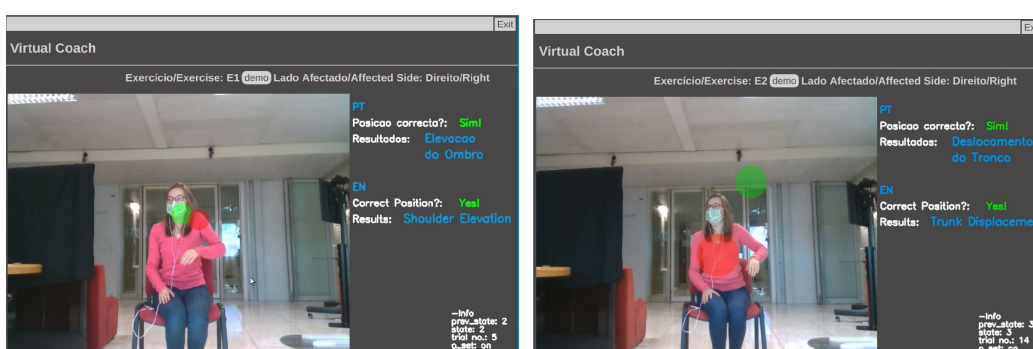
Figure 3.6: Virtual Coach **Main** web page - Patient positioning.



(a) Exercise 1 target position.

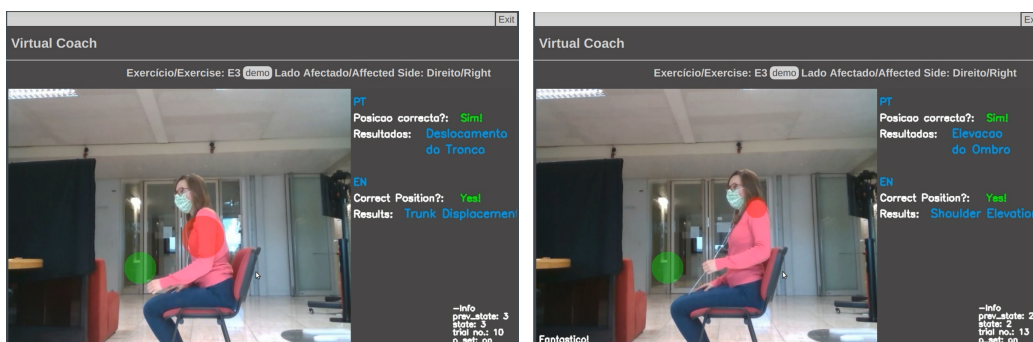
(b) Exercise 1 target position reached.

Figure 3.7: Virtual Coach Main web page - Target position.



(a) Shoulder elevation in exercise 1 - display shoulder compensation marker.

(b) Trunk displacement in exercise 2 - display trunk compensation marker.



(c) Trunk displacement in exercise 3 - display trunk compensation marker.

(d) Shoulder elevation in exercise 3 - display trunk compensation marker.

Figure 3.8: Virtual Coach Main web page - Display compensation markers.

3.2 Compensation Assessment Methods

This section describes the proposed methods to assess compensation patterns during exercise performance from 2D body keypoints. We establish the data processing steps needed to do this assessment, define a Multilabel Classification (MLC) problem, and propose two approaches to handle this task and determine different types of compensation detected in a movement trial. We propose a Rule-based (RB)

classification approach, which works as a baseline method, based on usually studied kinematic variables, and a Neural Network (NN) based approach. Similarly to related work from section 2.2.2, we follow the workflow from figure 3.9. Table 3.4 briefly describes the workflow procedures.



Figure 3.9: Work flow diagram of the compensation assessment method.

Procedures	Description
Feature Extraction	Detect and acquire body joints' 2D positional data
Feature Selection	From the set of acquired data, select the data of the subject of interest, in the case of a multi-person setting, and select its relevant keypoints
Data Normalization	Normalize keypoints to overcome physical variabilities and compute kinematic variables
Classification	Determine classification approaches that enable the detection compensation patterns, based on body keypoints and kinematic variables
Result Filtering	Filter classification results to produce a final decision

Table 3.4: Compensation Assessment Procedures.

To describe the main mathematical calculations, we use the notation described in table 3.5, where j is a joint of a set of joints, defined further on 3.2.2, and t is the video frame number.

Equation	Description
$p_j^t = [x_j^t \ y_j^t]'$	2D coordinates of a joint j ; $[x \ y]'$ denotes a transposed vector
$P^t(j_1, j_2) = p_{j_2}^t - p_{j_1}^t = [x_{j_2}^t - x_{j_1}^t \ y_{j_2}^t - y_{j_1}^t]'$	Vector directed from joint j_1 to joint j_2
$\ P^t(j_1, j_2)\ $ $d^t(j_1, j_2) = \ p_{j_1}^t - p_{j_2}^t\ $	$\ P^t(j_1, j_2)\ $ is the euclidean norm of vector $P^t(j_1, j_2)$ and, alternatively, $d^t(j_1, j_2)$ is the euclidean distance between two selected joints, j_1 and j_2
$\Delta x^t(j_1, j_2) = x_{j_1}^t - x_{j_2}^t$ $\Delta y^t(j_1, j_2) = y_{j_1}^t - y_{j_2}^t$	Displacement between two selected joints, j_1 and j_2 , in the X (Δx) and Y (Δy) axis
$a^t(j_1, j_2, j_3) = \arccos\left(\frac{P^t(j_2, j_1) \cdot P^t(j_2, j_3)}{\ P^t(j_2, j_1)\ \cdot \ P^t(j_2, j_3)\ }\right)$	Angle between two vectors, $P^t(j_2, j_1)$ and $P^t(j_2, j_3)$, defined by two points, j_2 to j_1 and j_2 to j_3

Table 3.5: Mathematical notation.

3.2.1 Feature Extraction

The first step to assess compensation is the extraction of relevant body joints' 2D positional data, which enable us to have the raw information about the performed movements. This process is called **Feature Extraction**. To accomplish body keypoints extraction from a video or sequence of images, we explore two motion detection libraries OpenPose and OpenFace, which we present and describe below.

3.2.1.A OpenPose

To extract body joints' 2D pose data, we use the OpenPose. OpenPose is the "first open-source realtime system for multi-person 2D pose detection, including body, foot, hand, and facial keypoints." Cao *et al.* [37] developed it to enable machines' comprehension of people's bodies and movements in an image or video and push forward computer vision and Machine Learning (ML) research. Taking as input a color image, OpenPose [37] detects different body parts (keypoints) of each person in the image, giving their 2D pose. It provides confidence score maps of body keypoints, which depict how confident the system is about a particular body part location in any given pixel. OpenPose predicts Part Affinity Fields [37] to associate body parts and build one's skeleton. These are a set of 2D vector flow fields, which keep limbs' position and orientation, encoding body parts association.

Unlike other 2D body pose estimation libraries, OpenPose [37] stands out by its compactness and traits. It consists of three detection blocks: body and foot detection (core block), hand detection, and face detection. The user can select the desired detection block or combine different blocks. OpenPose allows the user to select the desired input between images, videos, webcam, and IP camera streaming. The user can visualize the detected body skeletons and save keypoints' data. Additionally, OpenPose can also perform a 3D realtime single-person detection through 3D triangulation out of multiple synchronized cameras. Due to multiple cameras and the single-person requirements, this work does not consider this 3D reconstruction.

Along with the OpenPose [37] user-friendly characteristic and robustness, it is important to acknowledge its failure cases and drawbacks. In a multi-person setting, people's interaction and physical contact may lead to body occlusion, making difficult keypoints' identification and the association between body parts. Also, even in a single-person setting, one's abrupt and fast movements can hinder keypoint detection. These situations can lead to higher detection errors.

In this work, we only focus on body keypoints detection from a sequence of images. In this case, OpenPose [37] provides 2D positional data of body keypoints and a confidence score on detection correctness. Figure 3.10 illustrates a skeleton detected by OpenPose with its body keypoints. OpenPose gives a JSON file for each video frame sequence, containing keypoints with the format $\{x_0, y_0, s_0, x_1, y_1, s_1, \dots, x_{24}, y_{24}, s_{24}\}$, according to the OpenPose library documentation [8], where x and y are the 2D keypoint coordinates in the image Cartesian space, s is the confidence score, and the numbers in subscript correspond to the keypoint number. In this work, we denote a

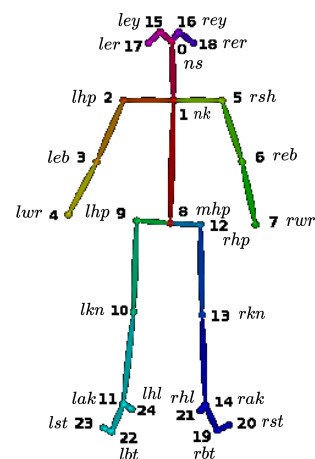


Figure 3.10: OpenPose keypoints. Adapted from [8].

OpenPose keypoint positional data as $o_j^t = [p_j^t \ s_j^t]' = [x_j^t \ y_j^t \ s_j^t]'$, in the image coordinate system $\{I\}$. The subscript j specifies a joint in the set of keypoints extracted with OpenPose, $J_o = \{ns, nk, lsh, leb, lwr, rsh, reb, rwr, mhp, lhp, lkn, lak, rhp, rnk, rak, ley, rey, ler, rer, rbt, rst, rhl, lbt, lst, lhl\}$, and s_j^t denotes a confidence score. Table 3.6 presents the complete set of keypoints, their joint number and name, taken from the library documentation [8], and the notation adopted by us. Figure 3.10 illustrates this set of keypoints mapped in a body skeleton.

Since we work with horizontally flipped images, in this work, we assume right and left body sides contrary to the ones presented in OpenPose documentation [8]. This way that the right and left sides in the image are the same sides for the observer, as a regular mirror.

Joint Number	Joint Name	Notation	Joint Number	Joint Name	Notation
0	<i>Nose</i>	<i>ns</i>	13	<i>RightKnee</i>	<i>rkn</i>
1	<i>Neck</i>	<i>nk</i>	14	<i>RightAnkle</i>	<i>rak</i>
2	<i>LeftShoulder</i>	<i>lsh</i>	15	<i>LeftEye</i>	<i>ley</i>
3	<i>LeftElbow</i>	<i>leb</i>	16	<i>RightEye</i>	<i>rey</i>
4	<i>LeftWrist</i>	<i>lwr</i>	17	<i>LeftEar</i>	<i>ler</i>
5	<i>RightShoulder</i>	<i>rsh</i>	18	<i>RightEar</i>	<i>rer</i>
6	<i>RightElbow</i>	<i>reb</i>	19	<i>RightBigToe</i>	<i>rbt</i>
7	<i>RightWrist</i>	<i>rwr</i>	20	<i>RightSmallToe</i>	<i>rst</i>
8	<i>MidHip</i>	<i>mhp</i>	21	<i>RightHeel</i>	<i>rhl</i>
9	<i>LeftHip</i>	<i>lhp</i>	22	<i>LeftBigToe</i>	<i>lbt</i>
10	<i>LeftKnee</i>	<i>lkn</i>	23	<i>LeftSmallToe</i>	<i>lst</i>
11	<i>LeftAnkle</i>	<i>lak</i>	24	<i>LeftHeel</i>	<i>lhl</i>
12	<i>RightHip</i>	<i>rhp</i>	-	-	-

Table 3.6: OpenPose body keypoints and applied notation.

3.2.1.B OpenFace

To overcome the lack of 3D pose data, and without the need to resort to another sensor to acquire 3D keypoints, like an RGB-depth camera, we explore the OpenFace toolkit [38]. OpenFace emerged to give answers to the increased attention to facial behavior analysis and comprehension in computer vision and ML research and for the development of facial interactive applications. This interest is because facial behavior gives cues about people’s emotional and cognitive state and social interaction signs since people communicate through facial expression, along with speech and body gesture. The authors of OpenFace [38] define facial traits as the set: facial landmark location; head pose; eye gaze; and facial expressions.

Like OpenPose, OpenFace is an easy-to-use toolkit that can process videos in realtime from a simple webcam, video files, image sequences, or single images, in multi or single person settings. It provides CSV files with the required data - facial landmarks, shape parameters, head pose, and gaze vectors [38].

OpenFace landmark detection is done through the Convolutional Experts Constrained Local Model [38] and, additionally to this detection, the head pose is extracted. The model uses a 3D depiction of facial

landmarks, which are projected to the image through orthographic camera projection. Head pose is derived after facial landmark detection by solving the problem of n point perspective [38]. Given this, we may assume that with images of lower resolution, thus lower detail, OpenFace might have increased difficulty in detecting facial landmarks, mostly if the subjects are distant from the camera.

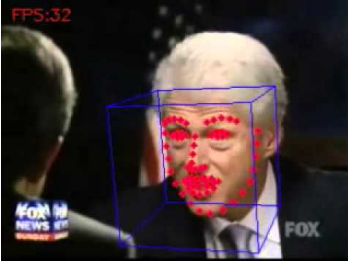


Figure 3.11: Examples of OpenFace facial landmark detection and bounding box around the head. Taken from [9]

In this project, we resort to head translation in Z , to track trunk displacements in depth. OpenFace provides the head's position in millimeters concerning the camera, which is the origin of the world coordinate system, $\{W\}$ [9]. Figure 3.11 illustrates OpenFace facial landmarks and head's bounding box.

3.2.2 Feature Selection

From the broad set of keypoints provided by OpenPose, it is imperative to select the ones that better describe the upper limb movement.

Additionally, in a real setting, we consider that, besides the patient, other subjects can be visible in the image, such as a therapist, a family member, or a caregiver. Therefore, we need to accomplish two procedures: select the subject under evaluation (the patient) and select one's significant keypoints. This process is called **Feature Selection**.

To remove extra people detected by OpenPose, we consider that the patient is placed right in front of the camera, i.e., in the center of the image. Thus, we consider an imaginary disk in the image center, denoting the region where the patient should be detected. When comparing the patient's and the extra subject positions, the patient should be closer to the disk or closer to its center. This assumption is illustrated in figure 3.12. Given that an acquired image has a resolution of 640×480 , in the image Cartesian space, the disk is defined by $D(c_d, r)$, which is a set of points in the image of the form $\{p \in [0, 640] \times [0, 480] : \|p - c_d\| = r\}$, where $c_d = [320 \ 240]'$ is the disk center, $p \in [0, 640] \times [0, 480]$ and r is the disk radius. The patient has to have its *MidHip* joint inside the disk, $d(p_{mhp}, c_d)_{patient} < r$. If both subjects are inside the disk, the patient is the one closer to its centre, $d(p_{mhp}, c_d)_{patient} < d(p_{mhp}, c_d)_{person2}$.

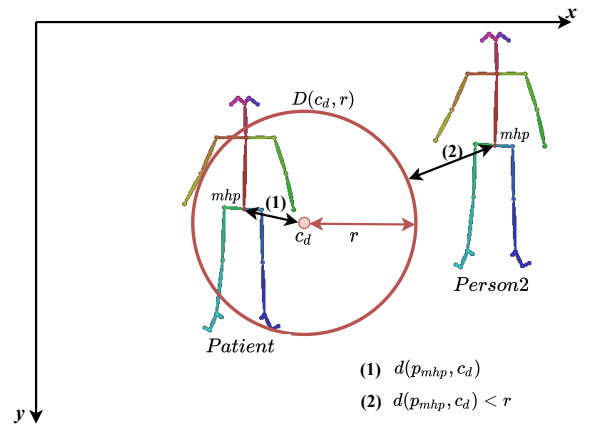


Figure 3.12: Patient selection method in the case of multi-person detection.

With the desired subject selected and having discarded the extra individual, now it is time to select the patient's relevant keypoints. To describe upper extremity motor function, we select the same keypoints as in *et al.* to detail compensation behaviors, through joint linear and angular displacement describing a tilted

torso, shoulder elevation, and shoulder abduction. We consider three scenarios (S1, S2, and S3) concerning patients' placement in front of the camera: the patient facing the recording camera (S1), and the patient's affected arm facing the camera, in a perpendicular (S2) or oblique (S3) position. Accordingly, for scenario S1, we select the set of keypoints denoted as $J_s = \{ns, nk, lsh, leb, lwr, rsh, reb, rwr, mhp, lhp, rhp\}$. This set contains the joints of the patient's affected side and the opposite side, which works as a reference. For scenarios S2 and S3, we select the keypoint set $J_{s_{right}} = \{ns, nk, lsh, rsh, reb, rwr, mhp, rhp\}$, when the affected side is the right, and $J_{s_{left}} = \{ns, nk, lsh, leb, lwr, rsh, mhp, lhp\}$, when the affected side is the left. Figure 3.13 illustrates these keypoints for all the scenarios.

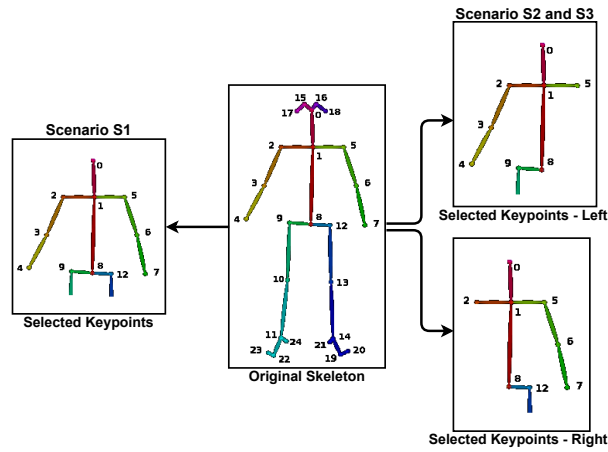


Figure 3.13: Keypoints' selection for both scenarios S1, S2, and S3.

3.2.3 Data Normalization

Given the selected body keypoints, it is crucial to perform feature **Normalization** to overcome physical variabilities. Our feature normalization approach consists of three steps: **transformation**, **normalization**, **mirror**. Transform the keypoints from the image coordinate system $\{I\}$ to the body coordinate system $\{B\}$, assuming the *MidHip* (*mhp*) joint as the origin (**transformation**), helps us handle variations in patient positioning. Even positioned in the image center, different patients are not positioned exactly at the same point in the image. Also, patient positioning may vary between movement trials. Normalizing the keypoints to a convenient body part length enables us to deal with different patients' body part dimensions and their distance to the camera (**normalization**). Additionally, we **mirror** negative joints in X , of $\{B\}$, to align both sides of body joints and provide the unaffected side as a reference for the NN classification approach.

3.2.3.A Transformation

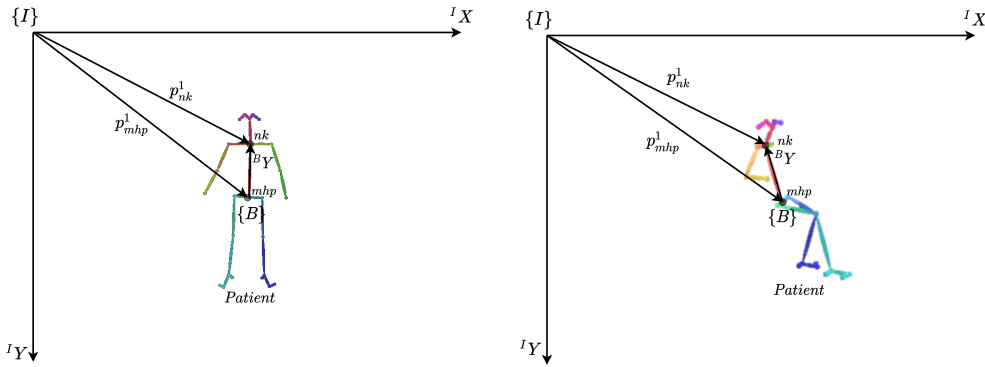
The transformation from the image coordinate system, $\{I\}$, to the body coordinate system, $\{B\}$, for each video frame, is computed with the data provided in the first frame, $t = 1$. This transformation is applied to every keypoint in every frame and is given by the expression (3.1). Table 3.7 describes its components.

$${}^B p_j^t = {}^B_I R \cdot {}^I p_j^t + {}^B P_I \quad (3.1)$$

	Description
${}^B_I R$	Rotation matrix from $\{I\}$ to $\{B\}$
${}^B P_I$	Translation vector between $\{I\}$ and $\{B\}$
${}^I p_j^t$	Any joint j in $\{I\}$, which is transformed to $\{B\}$
${}^B p_j^t$	Transformed joint in the body coordinate system, $\{B\}$

Table 3.7: Transformation Components.

To compute this transformation, first, we compute the direction vector, or versor, used to describe Y axis spacial direction in the body coordinate system, $\{B\}$, denoted ${}^B Y$. Considering the patient might not have a completely vertical spine position, the ${}^B Y$ axis is aligned with the patient's spine as described in figure 3.14, which represents one's initial position. Figure 3.14 illustrates the Y axis for scenarios S1, S2, and S3.



(a) ${}^B Y$ axis in a patient with straight spine (S1). (b) ${}^B Y$ axis for left affected side (S2 and S3).

Figure 3.14: Determine Y axis for the body 2D Cartesian space, $\{B\}$, for the different scenarios.

To compute the ${}^B Y$ vector for all scenarios, we consider it is a result from the subtraction of two vectors, as shown in figure 3.14 - a vector directed from the image coordinate system origin to the *Neck* joint, and a vector from the same origin to the *MidHip* joint. Since the image Cartesian space origin is a point with coordinates $[0 \ 0]'$, these vectors are described by the same coordinates of the joints *Neck* and *MidHip*, respectively. Thus, the vectors are notated as follows, for $t = 1$: $p_{nk}^1 = [x_{nk}^1 \ y_{nk}^1]'$, and $p_{mhp}^1 = [x_{mhp}^1 \ y_{mhp}^1]'$, both in the image coordinate system, $\{I\}$. The Y axis versor, in the body coordinate system, $\{B\}$, is given by ${}^B Y = [y_B^x \ y_B^y]'$.

Equation (3.2) describes the operation used to compute the ${}^B Y$ versor. Since a versor is a unit vector, i.e., a vector with a unit norm, vector subtraction is divided by its norm.

$${}^B Y = \frac{p_{nk}^1 - p_{mhp}^1}{\|p_{nk}^1 - p_{mhp}^1\|} \quad (3.2)$$

The X axis direction vector, in $\{B\}$, is denoted by ${}^B X' = [x_B^x \quad x_B^y]'$. Its coordinates are given solving the (3.3) system of equations according with following the considerations below:

- The versors ${}^B X$ and ${}^B Y$ are perpendicular, i.e., they form a 90° angle between them, which means their inner product is zero, ${}^B X \cdot {}^B Y = 0$;
- ${}^B X$ is a unit vector, i.e., its has unit norm $\|{}^B X\| = 1$.

$$\begin{cases} x_B^x \cdot y_B^x + x_B^y \cdot y_B^y = 0 \\ \sqrt{x_B^x^2 + x_B^y^2} = 1 \end{cases} \quad (3.3)$$

Since patients have one affected side due to stroke, which is the one that needs more attention during rehabilitation training, the ${}^B X$ is directed to the affected side, as illustrated in figure 3.15, for scenario S1 (patient facing the camera). For scenarios S2 and S3 (patient's affected arm facing the camera), ${}^B X$ is directed to the patient's front. Thus, solving the (3.3) equations' system we obtain ${}^B X$. More specifically, for S1, ${}^B X$ is given by (3.4) and (3.5) for right and left affected sides, respectively. For S2 and S3, is the opposite, ${}^B X$ is given by (3.5) and (3.4) for right and left affected sides, respectively.

$$\begin{cases} x_B^x = -y_B^y \\ x_B^y = y_B^x \end{cases} \quad (3.4)$$

$$\begin{cases} x_B^x = y_B^y \\ x_B^y = -y_B^x \end{cases} \quad (3.5)$$

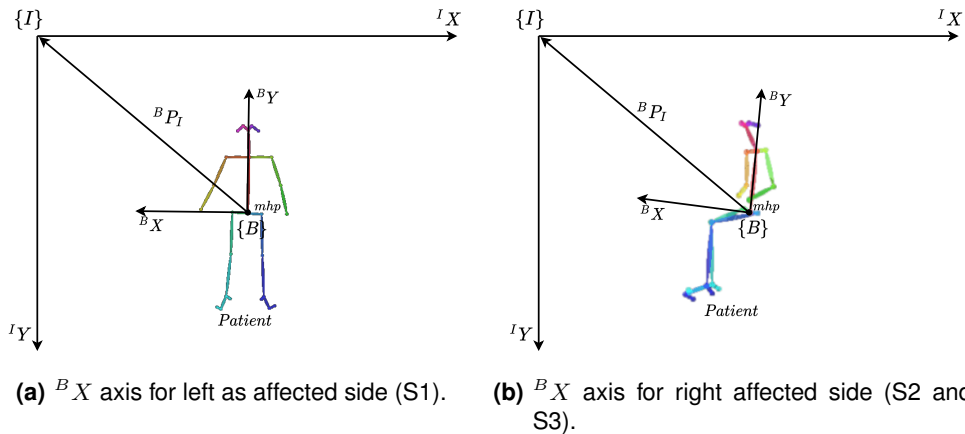


Figure 3.15: ${}^B X$ axis for the body 2D Cartesian space, B , for left and right affected sides in all scenarios (S1, S2, and S3). Translation vector between $\{I\}$ and $\{B\}$ coordinate systems, ${}^B P_I$.

Given the ${}^B X$ and ${}^B Y$ vectors coordinates, we can calculate the rotation matrix, ${}^B R$, and translation vector, ${}^B P_I$. The rotation matrix is given by the matrix (3.6). Figure 3.15 illustrates the translation vector, which is given by equation (3.7).

$${}^B_I R = \begin{bmatrix} x_B^x & x_B^y \\ y_B^x & y_B^y \end{bmatrix} \quad (3.6)$$

$${}^B P_I = {}^B_I R \cdot (-p_{mhp}^1) \quad (3.7)$$

3.2.3.B Normalization

To avoid physical variabilities concerning patient-camera distance and body parts' dimensions, we normalize the transformed body keypoints. Keypoints' coordinates are normalized to the patient's spine length in the first video frame, $t = 1$. Spine length is given by the distance between the *Neck* and *MidHip* joints, $d^1(p_{nk}, p_{mhp})$, which is the same in $\{I\}$ or $\{B\}$.

Equation (3.8) gives a keypoint normalized to the spine length. Since for S2 and S3, we only consider visible joints, this equation gives the final normalized keypoints, for these scenarios.

$${}^B \hat{p}_j^t = \frac{{}^B p_j^t}{d^1({}^I p_{nk}, {}^I p_{mhp})} = \frac{\begin{bmatrix} {}^B x_j^t & {}^B y_j^t \end{bmatrix}'}{d^1({}^I p_{nk}, {}^I p_{mhp})} \quad (3.8)$$

3.2.3.C Mirror

Since only the affected side intervenes in the exercise training, i.e., only this side describes clear movements to perform a task, we give the healthy side as a reference to the classifier in S1. We align the opposite and affected sides, mirroring the joints in the negative ${}^B X$ axis to the positive ${}^B X$. For S2 and S3, we do not perform this step since we cannot give as reference an invisible body side. In this case, the spine joints work as a reference. Equation (3.9) gives the normalized and mirrored keypoints in S1.

$${}^B \tilde{p}_j^t = \frac{\begin{bmatrix} -{}^B x_j^t & {}^B y_j^t \end{bmatrix}'}{d^1({}^I p_{nk}, {}^I p_{mhp})} \quad (3.9)$$

3.2.3.D Kinematic Variables

For the RB classification approach, which works as our baseline method, we consider as features kinematic variables as in related work [6]. However, since we only have 2D positional data, we need to find alternative measures to assess some compensation patterns. We intend to identify four types of compensation: **Trunk Forward**, **Trunk Rotation**, **Shoulder Elevation**, and **Other** trunk compensation patterns, such as trunk moving backward and trunk to tilt. Thus, we formulate hypotheses for the three scenarios regarding patients' positioning in front of the camera (S1, S2, and S3) to detect these movement patterns. It is important to mention that, to calculate the kinematic variables - joints' angles and displacements -, the mirror keypoint normalization step is not applied, i.e., normalized keypoints are given by the expression (3.8). These

variables describe linear and angular displacements over time. Thus, we do not need to provide the RB classifier the healthy side as a reference.

Trunk Forward Since we believe that OpenFace might have issues detecting patients more distant to the camera, given the image resolution, to detect the trunk moving forward, **Trunk Forward**, we state two hypotheses for S1 and consider one kinematic variable for S2 and S3, which we describe in table 3.8.

Scenarios	Hypotheses and Kinematic Variables
S1	<ul style="list-style-type: none"> • <i>Hypothesis 1</i>: detect trunk moving forward through the observed changes in subject head size, i.e., when the subject moves forward, one's head will seem bigger in the image • <i>Hypothesis 2</i>: use OpenFace to assess head's translation in Z - in the world coordinate system, $\{W\}$, - i.e., when the subject moves forward will be closer to the camera
S2 & S3	Detect the trunk moving forward (illustrated in figure 3.16) with the tilted angle of the spine, $a^t(p_{mhp}^1, p_{nk}^1, p_{nk}^t)$, and a positive displacement of the <i>Neck</i> joint in X , $\Delta x^t(p_{nk}^t, p_{nk}^1)$

Table 3.8: Hypotheses and Kinematic Variables to Assess **Trunk Forward** for all the Scenarios.

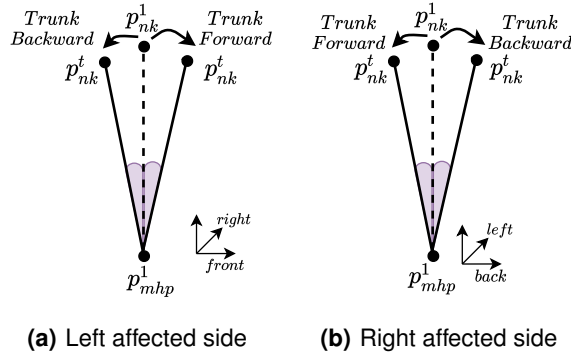


Figure 3.16: Trunk moving forward and backward in the case of S2 and S3.

In the case of S1, *Hypothesis 1* includes the head keypoints as significant ones. These keypoints are $J_{sr} = \{J_s, ley, rey, ler, rer\}$ (see figure 3.10 and table 3.6). To detect changes in the observed head size, we define the patient's head area as a set of four triangles, which figure 3.17(a) illustrates. Three joints define each triangle. Thus, we denote a triangle's area as $A^t(j_1, j_2, j_3)$.

To compute the triangle's area, we define a triangle, as shown in figure 3.17(d). Triangle vertices are body joints and its edges are the Euclidean distance between those joints. We consider an isosceles triangle since we assume the head is symmetric. To calculate the area, we perform the following calculations, which result in the expression (3.10):

- $a^t(j_2, j_1, j_3)$ is one of the triangle's internal angle, and $\cos a^t(j_2, j_1, j_3) = \frac{m}{d^t(j_1, j_2)} \Leftrightarrow$
 $\Leftrightarrow m = d^t(j_1, j_2) \cdot \cos a^t(j_2, j_1, j_3);$
- $\cos a^t(j_2, j_1, j_3) = \frac{P^t(j_1, j_2) \cdot P^t(j_1, j_3)}{\|P^t(j_1, j_2)\| \cdot \|P^t(j_1, j_3)\|};$
- h the triangle's height is given following the Pythagoras' theorem $h^2 = d^t(j_1, j_2)^2 - m^2 \Leftrightarrow$
 $\Leftrightarrow h = \sqrt{d^t(j_1, j_2)^2 (1 - \cos^2 a^t(j_2, j_1, j_3))}.$

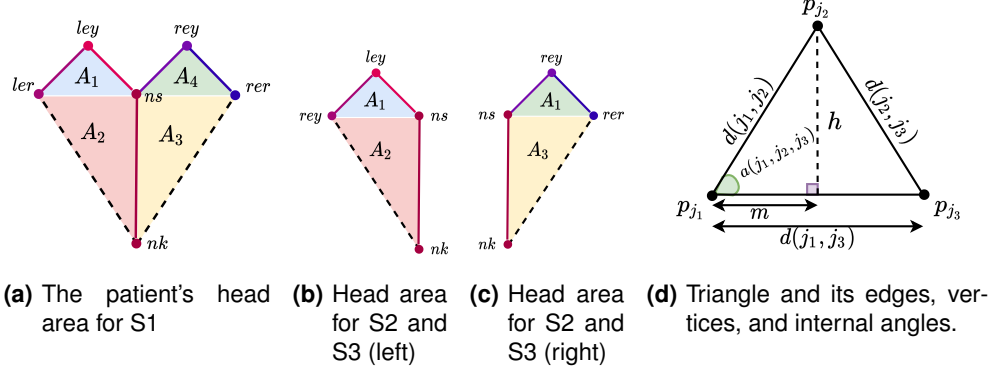


Figure 3.17: The patient's head area, and triangle defining it.

$$A^t(j_1, j_2, j_3) = \frac{d^t(j_1, j_3) \cdot h}{2} = \frac{d^t(j_1, j_3) \cdot d^t(j_1, j_2)}{2} \cdot \sqrt{1 - \cos^2 \alpha(j_2, j_1, j_3)} \quad (3.10)$$

The head's area is denoted $H^t = \sum_{i=1}^4 A_i^t(j_1, j_2, j_3)$, where i denotes the different head triangles. More specifically, head's area is the sum of $A_1^t(ler, rey, ns)$, $A_2^t(ler, ns, nk)$, $A_3^t(nk, ns, rer)$, and $A_4^t(ns, ley, rer)$. Head area variations are described by the difference between the head area at any timestamp t and at first frame, $t = 1$, which is expressed by the system (3.11).

$$\Delta H^t = \begin{cases} H^t - H^1, & \text{if } t > 1 \\ 0, & \text{otherwise} \end{cases} \quad (3.11)$$

S1 *Hypothesis 2* suggests the track of patient head displacement in Z in the world coordinate system, $\{W\}$, in which the camera is the origin. We denote the translation component in Z in any frame t as T_z^t , and the conditions (3.12) give the changes in trunk position.

$$\Delta T_z^t = \begin{cases} T_z^t - T_z^1, & \text{if } t > 1 \\ 0, & \text{otherwise} \end{cases} \quad (3.12)$$

Trunk Rotation To detect **Trunk Rotation** and since we do not have a manner of computing the rotation angle in 3D, we formulate the hypotheses from table 3.9 for the three scenarios.

Scenarios	Hypotheses and Kinematic Variables
S1	<ul style="list-style-type: none"> <i>Hypothesis</i>: when the patient rotates the torso to the affected side while moving the affected limb, the shoulder of the affected side elevates while the opposite shoulder decays
S2	<ul style="list-style-type: none"> <i>Hypothesis</i>: given the patient is in a perpendicular position, when the patient rotates the torso to the affected side, the shoulder is displaced regarding the <i>Neck</i> joint
S3	<ul style="list-style-type: none"> <i>Hypothesis</i>: given the patient oblique position, when the patient rotates the torso to the affected side, the observed distance between both shoulders decreases

Table 3.9: Hypotheses and Kinematic Variables to Assess **Trunk Rotation** for all the Scenarios.

For S1, figure 3.18 represents the hypothesis stated for both affected sides. The rotation is measured by the angles formed by the shoulder in the initial ($t = 1$) and current positions ($t > 1$), given by $a^t(p_{rsh}^1, p_{nk}^1, p_{rsh}^t)$ and $a^t(p_{lsh}^1, p_{nk}^1, p_{lsh}^t)$ for right left shoulders, respectively.

For S2 and S3, we distinguish them by assessing the length between the patient's shoulders in $t = 1$, which is given by $d^1(p_{lsh}, p_{rsh})$. If this length is above a specified threshold, the subject's chest is visible, and one is in an oblique position in the image. In the case of S2, figures 3.19(a) and 3.19(b) illustrates trunk rotation for both sides and it is given $\Delta x^t(p_{rsh/lsh}, p_{nk}) = x_{rsh/lsh}^t - x_{nk}^t$. For S3, trunk rotation is assessed by the variation over time of the length between both shoulders, given by the equation (3.13). We distinguish S2 and S3 because the different patient's positioning leads to a different relation between *Neck* and shoulder joints. While for S2 the shoulder moves in BX regarding *Neck* over time, for S3, we cannot observe the same. Section 4.2.3 validates these hypotheses.

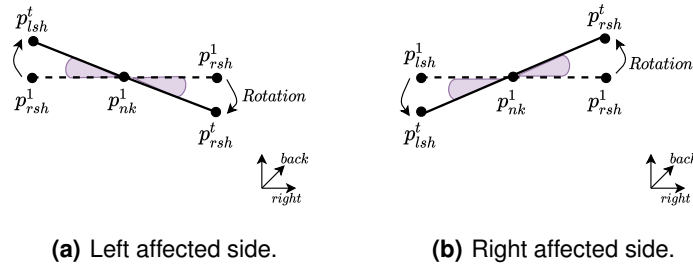


Figure 3.18: Trunk Rotation hypothesis for S1.

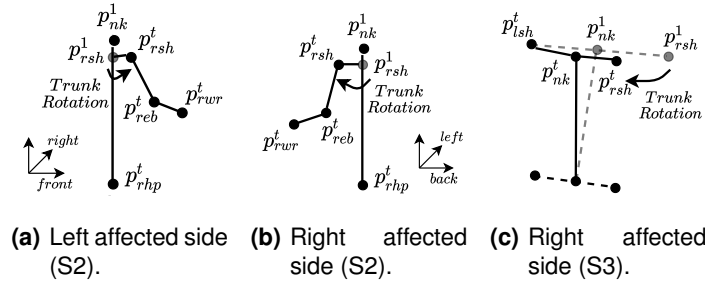


Figure 3.19: Trunk Rotation hypothesis for S2 and S3.

$$|\Delta d^t(p_{rsh}, p_{lsh})| = \begin{cases} |d^t(p_{rsh}, p_{lsh}) - d^1(p_{rsh}, p_{lsh})|, & \text{if } t > 1 \\ 0, & \text{otherwise} \end{cases} \quad (3.13)$$

Shoulder Elevation The **Shoulder Elevation** compensation behavior is detectable in 2D by the shoulder elevation angle of the affected side in the case of S1 as described in figure 3.20. For S2 and S3, we formulate a hypothesis to assess this pattern. Table 3.10 describes shoulder elevation assessment for the three scenarios.

Scenarios	Hypotheses and Kinematic Variables
S1	Shoulder elevation is detected by the angle described by the affected side shoulder, given by $a^t(p_{rsh/lsh}^1, p_{nk}^1, p_{rsh/lsh}^t)$
S2 & S3	<ul style="list-style-type: none"> <i>Hypothesis</i>: when the patient elevates the shoulder during exercise performance, the shoulder is displaced regarding the <i>Neck</i> joint in <i>Y</i>, described by $\Delta y^t(p_{rsh/lsh}, p_{nk}) = y_{rsh/lsh}^t - y_{nk}^t$

Table 3.10: Hypotheses and Kinematic Variables to Assess **Shoulder Elevation** for all the Scenarios.

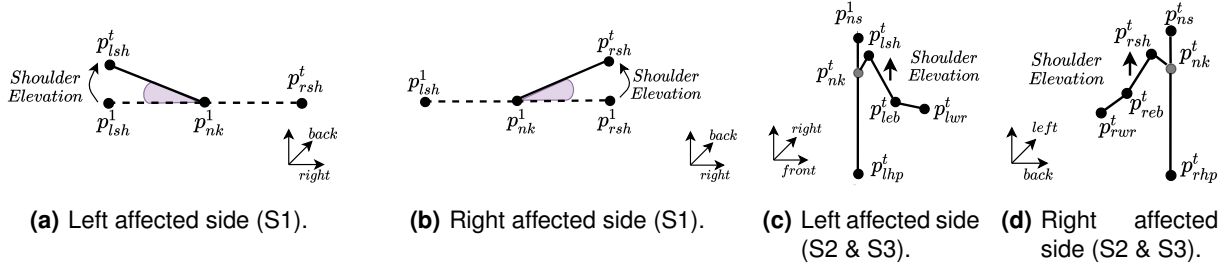


Figure 3.20: Shoulder Elevation for S2 and S3.

Other Compensation Patterns The **Other** compensation behaviors concern additional trunk displacement patterns. Here we focus on trunk tilt and trunk moving backward. Table 3.11 delineates the established hypotheses and kinematic variables assumed to assess these patterns.

Concerning the hypotheses raised to assess trunk tilt for S2 and S3, positive and negative changes in the patient's head area (equation (3.11)) and in head translation in ${}^W Z$ (equation (3.12)) correspond to the same movement pattern. Thus, trunk tilt is assessed by these variables' absolute values, $|\Delta H^t|$ and $|\Delta T_z^t|$, respectively. For *Hypothesis 1* since in these scenarios the unaffected side is not visible or is partially hidden, to calculate the head area we only consider the affected side joints as illustrated in figures 3.17(b) and 3.17(c). It is given by $H^t = \sum_{i=1}^2 A_i^t(j_1, j_2, j_3)$. Assessing trunk moving backward for S1 is similar to detect the trunk moving forward through equations (3.11) (*Hypothesis 1*) and (3.12) (*Hypothesis 2*).

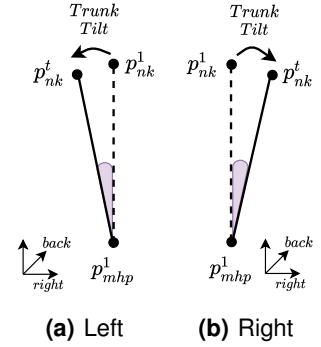


Figure 3.21: Trunk tilt for S1.

In summary, for the RB classification approach, we use the kinematic variables presented in the table 3.12 for scenarios S1, S2, and S3 regarding the patient position in front of the camera.

3.2.4 Classification Approaches

Classification is a predictive task, usually carried out through supervised learning methods, which predicts an output from an input dataset. In supervised methods, the input data samples have to be labeled, ideally by a specialist in the problem domain, who assigns a discrete value, a *label*, or *class*, to each sample. The classifier learns from each data sample characteristics and respective label, analyzing the correlation between them, becoming able to predict the class of new, never observed before, data samples [39, 40].

Scenarios	Hypotheses and Kinematic Variables - Trunk Tilt
S1	When the patient tilts the torso (illustrated in figure 3.21), the tilted angle of the spine, $a^t(p_{mhp}^1, p_{nk}^1, p_{nk}^t)$, describes this movement
S2 & S3	<ul style="list-style-type: none"> • <i>Hypothesis 1</i>: detect the trunk tilt through the observed changes in the patient's head size • <i>Hypothesis 2</i>: use OpenFace to assess head's translation in Z, in $\{W\}$
Scenarios	Hypotheses and Kinematic Variables - Trunk Backward
S1	<ul style="list-style-type: none"> • <i>Hypothesis 1</i>: detect the trunk moving backward through the observed changes in the patient's head size, i.e., when the subject moves backward, the observed head area decreases • <i>Hypothesis 2</i>: use OpenFace to assess head's translation in Z, i.e., when the subject moves backward will be farther away from the camera, revealing a bigger translation in Z, in $\{W\}$
S2 & S3	The tilted angle of the spine, $a^t(p_{mhp}^1, p_{nk}^1, p_{nk}^t)$, and a negative displacement of the <i>Neck</i> joint in $^B X$, $\Delta x^t(p_{nk}^t, p_{nk}^1)$ detects the trunk moving backward (illustrated in figure 3.16)

Table 3.11: Hypotheses and kinematic variables to assess **Other** trunk compensation behaviors for all the scenarios.

Type of Compensation	Kinematic Variables S1	Kinematic Variables S2 & S3
Trunk Forward	<p><i>Hypothesis 1</i>: $\Delta H^t = \begin{cases} H^t - H^1, & \text{if } t > 1 \\ 0, & \text{otherwise} \end{cases}$</p> <p>in the $\{B\}$ coordinate system</p> <p><i>Hypothesis 2</i>: $\Delta T_z^t = \begin{cases} T_z^t - T_z^1, & \text{if } t > 1 \\ 0, & \text{otherwise} \end{cases}$</p> <p>in the $\{W\}$ coordinate system</p>	$a^t(p_{mhp}^1, p_{nk}^1, p_{nk}^t)$ and $\Delta x^t(p_{nk}^t, p_{nk}^1)$
Trunk Rotation	<p><i>Hypothesis</i>: $a^t(p_{rsh}^1, p_{nk}^1, p_{rsh}^t)$</p> <p>and $a^t(p_{lsh}^1, p_{nk}^1, p_{lsh}^t)$</p>	<p>(S2) <i>Hypothesis</i>: $\Delta x^t(p_{lsh/rsh}^t, p_{nk}^t)$</p> <p>(S3) <i>Hypothesis</i>: $\Delta d^t(p_{rsh}, p_{lsh})$</p>
Shoulder Elevation	$a^t(p_{rsh}^1, p_{nk}^1, p_{rsh}^t)$ or $a^t(p_{lsh}^1, p_{nk}^1, p_{lsh}^t)$	<i>Hypothesis</i> : $\Delta y^t(p_{lsh/rsh}^t, p_{nk}^t)$
Other	<ul style="list-style-type: none"> • Trunk Inclination: $a^t(p_{mhp}^1, p_{nk}^1, p_{nk}^t)$ • Trunk Backward: <p><i>Hypothesis 1</i>: $\Delta H^t = \begin{cases} H^t - H^1, & \text{if } t > 1 \\ 0, & \text{otherwise} \end{cases}$</p> <p>in the $\{B\}$ coordinate system</p> <p><i>Hypothesis 2</i>: $\Delta T_z^t = \begin{cases} T_z^t - T_z^1, & \text{if } t > 1 \\ 0, & \text{otherwise} \end{cases}$</p> <p>in the $\{W\}$ coordinate system</p>	<ul style="list-style-type: none"> • Trunk Inclination: <p><i>Hypothesis 1</i>: $\Delta H^t = \begin{cases} H^t - H^1 , & \text{if } t > 1 \\ 0, & \text{otherwise} \end{cases}$</p> <p>in the $\{B\}$ coordinate system</p> <p><i>Hypothesis 2</i>: $\Delta T_z^t = \begin{cases} T_z^t - T_z^1 , & \text{if } t > 1 \\ 0, & \text{otherwise} \end{cases}$</p> <p>in the $\{W\}$ coordinate system</p> <ul style="list-style-type: none"> • Trunk Backward: <p>$a^t(p_{mhp}^1, p_{nk}^1, p_{nk}^t)$ and $\Delta x^t(p_{nk}^t, p_{nk}^1)$</p>

Table 3.12: Kinematic variables for Rule-based classification.

Multilabel Classification (MLC) is a specific type of classification task in which the output is not a unique output value but an array of outputs. This aspect poses the main difference between multilabel and traditional classification problems. The output array has a length equal to the number of labels. It is composed of binary values suggesting the active and significant labels for the data sample. Thus, the chosen model for handling a MLC performs multiple predictions at once [40]. As any other Machine Learning (ML) algorithm, the multilabel classifier requires a quantitative performance measure to evaluate its prediction capability. Some of the performance measures employed for this kind of classification problem are defined in section 4.2.1. Based on available literature [40], we formally define the MLC problem as follows:

- Let \mathcal{L} denote the set of all possible labels, and $\mathcal{P}(\mathcal{L})$ the set of all \mathcal{L} subsets, with all the possible label

combinations $l \in \mathcal{L}$. $k = |\mathcal{L}|$ is the total number of labels;

- Let \mathcal{X} denote the input space with data samples $X \in \mathcal{X}$, and \mathcal{Y} is the output space with all possible arrays, $Y \in \mathcal{P}(\mathcal{L})$;
- Let \mathcal{D} denote a MLC, in which each component $(X, Y) \in \mathcal{D}$ is a data sample. $n = |\mathcal{D}|$ denotes the number of samples;
- Let $\mathcal{F} : \mathcal{X} \rightarrow \mathcal{Y}$ define a classifier. Any sample $X \in \mathcal{X}$ is the classifier input, and its output is a prediction $Z \in \mathcal{Y}$. Thus, a prediction of an array of labels for any data sample is given by $Z = \mathcal{F}(X)$.

Learning from multilabel data is handled by applying two main approaches: data transformation and method adaptation. The former approach consists of a problem simplification by transforming the Multilabel Dataset (MLD) into one or more binary or multiclass datasets, making the problem solvable with the traditional classification algorithms. The latter approach resides on the adaptation of the conventional classifiers to make them able to predict an array of output values instead of a single value. Additionally, emerging from the data transformation approach, the use of sets of classifiers (ensembles), a technique that intends to enhance individual classifiers performance, also poses a way of dealing with this kind of tasks [40].

While dealing with a MLD, it is essential to consider some conditions. We mention label dependency and imbalance. When applying a problem transformation method, it is important to recognize label dependency. E.g., in image labeling, if the label garden is active is highly probable that the labels flower or tree are also relevant. Learning from imbalanced data is also a problem of MLC. Disparities between label distribution stand from the MLD characteristics, mainly if it holds many labels. For instance, in a problem of document categorization, is expected the occurrence of some categories to be more frequent than others [40].

This work focuses on problem transformation approaches to learn from a MLD. Among these methods, we highlight the binarization techniques. The most popular is *Binary Relevance*, which consists of training k classifiers, one for each label, taking data samples in which a label is active, and the others are inactive, providing a binary result per label. Applying this method to the original dataset generates as many predictions as the number of labels, which are then combined to produce the multilabel response. A well-known technique is the *One-vs-Rest* approach, which trains an individual classifier for each class against all others. However, there are drawbacks associated with this binary approach. Since it considers an independent classifier for each label, one label prediction does not influence the other, leading to the loss of any possible existent label dependency. Additionally, each label classifier has to deal with greatly imbalanced data, which is very obvious for classifiers learning from very frequent or scarce labels [40].

Considering the present in section 3.2.3.D, our problem comprises a set of categories towards the compensation patterns we desire to assess: **Trunk Forward**, **Trunk Rotation**, **Shoulder Elevation**, **Other** trunk displacements, and **Normal** motor patterns.

3.2.4.A Rule-based Classification Method

Rule-based (RB) classification models, used for predictive tasks, are a set of *if-then* rules applied to a collection of features and providing a predicted label. This kind of model is pretty easy to comprehend and the produced results are of uncomplicated interpretation, which constitutes their main advantage [40, 41].

In this work, we apply a set of independent rules to the hypothesized kinematic variables in section 3.2.3.D to detect the compensation patterns from 2D positional data. We define rules for each pattern and each scenario S1, S2, and S3. First, we define the rules for scenario S1 and then for scenario S2 and S3.

In S1, to assess trunk moving forward and backward movements, we stated two hypotheses. As stated previously, for *Hypothesis 1*, there is compensation if the observed head size increases when the patient moves the torso forward. For *Hypothesis 2*, compensation exists if the patient does a positive displacement in ${}^W Z$. Thus, for both hypotheses, if ΔH^t and ΔT_z^t , respectively, are above a specified threshold value, the patient is moving the torso forward when trying to reach the desired position. Similarly, when the patient is moving the torso backward, for *Hypothesis 1*, compensation exists if ΔH^t decreases. In the case of *Hypothesis 2*, we consider that the patient is moving the trunk backward if there is a negative displacement in ${}^W Z$. These hypotheses are detailed by the equation (3.14).

$$Predicted\ Label = \begin{cases} 'Other', & \text{if } \Delta H^t < th_{TB_H} / \Delta T_z^t > th_{TB_T_z} \\ 'Trunk\ Forward', & \text{if } \Delta H^t > th_{TF_H} / \Delta T_z^t < th_{TF_T_z} \\ 'Normal', & \text{otherwise} \end{cases} \quad (3.14)$$

To assess **Trunk Rotation** and **Shoulder Elevation** we focus in the angles described by both shoulders concerning their initial position, $a^t(p_{rsh}^1, p_{nk}^1, p_{rsh}^t)$ and $a^t(p_{lsh}^1, p_{nk}^1, p_{lsh}^t)$. As schemed in figure 3.22, if the shoulder of the affected side describes an elevation angle above a particular value, we need to look at the opposite shoulder to determine its displacement. If both shoulders describe a similar angle, the patient rotating the trunk. On the other hand, if the affected side shoulder describes an elevation angle higher than the fall described by the unaffected side shoulder, we consider that the patient elevates the shoulder and rotates the torso simultaneously.

To assess patient tilting the torso for S1, we look at the tilted angle of the spine, $a^t(p_{mhp}^1, p_{nk}^1, p_{nk}^t)$. As described in equation (3.15), if this angle reaches a certain value, we consider that the patient tilts the torso as an aid to reach the goal position.

$$Predicted\ Label = \begin{cases} 'Other', & \text{if } a^t(p_{mhp}^1, p_{nk}^1, p_{nk}^t) > th_{TI} \\ 'Normal', & \text{otherwise} \end{cases} \quad (3.15)$$

For scenarios S2 and S3, since we formulated different hypotheses towards the different subject positioning, we also define distinct rules. To detect the trunk moving forward or backward, in this case, we focus in the tilted angle of the spine, $a^t(p_{mhp}^1, p_{nk}^1, p_{nk}^t)$, and in the displacement in ${}^B X$ of the joint *Neck*,

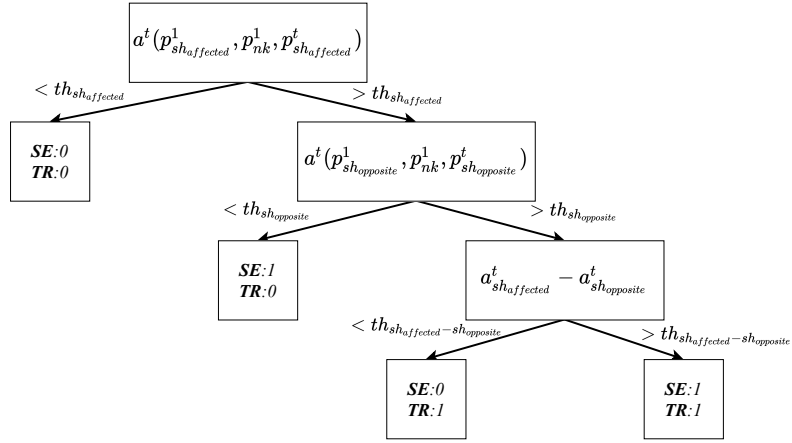


Figure 3.22: Rules to assess **Trunk Rotation (TR)** and **Shoulder Elevation (SE)** in S1.

$\Delta x^t(p_{nk}^t, p_{nk}^1)$. As shown in figure 3.23, if the tilted angle of the spine is above a specified value, there is a compensation of the torso. To complement, if the displacement in ${}^B X$ of the *Neck* joint is positive, the corresponding pattern is **Trunk Forward**. Otherwise, the subject is moving the trunk backward (**Other**).

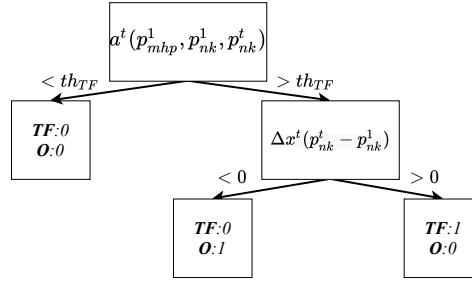


Figure 3.23: Rules to assess **Trunk Forward (TF)** and trunk backward - **Other (O)** - in S2.

When a patient rotates the torso, we assume a hypothesis for S2 and another for S3, since in these cases, the shoulder joint does not relate with the spine in the same way. For S2, we hypothesize that this can be detected through the shoulder displacement in ${}^B X$ regarding the *Neck* joint, $\Delta x^t(p_{lsh/rsh}, p_{nk})$. This way, as defined by equation (3.16), if this displacement is significant enough there is **Trunk Rotation**. For S3, we hypothesize that when the patient rotates the torso, the distance between both shoulders decreases since this patient has the chest visible. This assessment is given by the expression (3.17).

$$Predicted\ Label = \begin{cases} 'Trunk\ Rotation', & \text{if } \Delta x^t(p_{lsh/rsh}, p_{nk}^t) > th_{TR} \\ 'Normal', & \text{otherwise} \end{cases} \quad (3.16)$$

$$Predicted\ Label = \begin{cases} 'Trunk\ Rotation', & \text{if } |\Delta d^t(p_{lsh}, p_{rsh}^t)| > th_{TR} \\ 'Normal', & \text{otherwise} \end{cases} \quad (3.17)$$

To evaluate shoulder elevation, we compare shoulder position with the *Neck* joint position, described by a displacement in ${}^B Y$, $\Delta y^t(p_{lsh/rsh}, p_{nk})$. If this displacement is above a specified value, the patient is

describing **Shoulder Elevation** (equation (3.18)).

$$Predicted\ Label = \begin{cases} 'Shoulder\ Elevation', & \text{if } \Delta y^t(p_{lsh/rsk}^t, p_{nk}^t) > th_{SE} \\ 'Normal', & \text{otherwise} \end{cases} \quad (3.18)$$

For this scenario, S2, if the subject tilts the torso, is describing a movement in depth as when the subject is moving the trunk forward and backward in scenario S1. So, this pattern can also be detected by the absolute variations in the observed head size, $|\Delta H^t|$ (*Hypothesis 1*) or by head's dislocation in ${}^W Z$, $|\Delta T_z^t|$ (*Hypothesis 2*). Equation (3.19) describes these rule.

$$Predicted\ Label = \begin{cases} 'Other', & \text{if } |\Delta H^t| > th_{TIH} / |\Delta T_z^t| > th_{TIZ} \\ 'Normal', & \text{otherwise} \end{cases} \quad (3.19)$$

3.2.4.B Neural Network Based Classification Method

Neural Network (NN), know as Multilayer Perceptron, is a supervised learning algorithm able to learn a function, \mathcal{F} , from a given dataset, through a procedure called *training*. For classification tasks, it maps a set of provided input feature samples, \mathcal{X} , to target values or class labels, \mathcal{Y} . Once the model is trained, it can predict the label for new input samples. NN models are commonly trained by *Backpropagation* [39, 42, 43].

Since we are dealing with a multilabel problem, we can infer firsthand that we can come across a label dependency problem. Many times, patients do not describe motor compensation while performing a task. When we appraise **Normal** movement behaviors, we desire that our multilabel classifier is robust enough to avoid assigning a label to a video frame denoting the existence of compensation as it confirms the good movement quality. With this desire, we divide our problem into two problems, a binary and a multilabel. First, we use a binary classifier (C1) to assess compensation existence from each video frame keypoints. Second, we apply a multilabel classifier (C2), which concludes the described compensation patterns from the frames with compensation detected by C1. This way, our model will not classify a movement pattern as **Normal** and detect compensation simultaneously. Figure 3.24 represents our proposed approach. It is relevant to mention that the model considers the positional data extracted from a video frame as an independent data sample, without acknowledging frames' sequential order, which composes the entire exercise trial video.



Figure 3.24: NN based approach to assess compensation from 2D positional data.

3.2.5 Classification Result Filtering

After classifying the input data and determine the different compensation patterns presented in each frame, our final decision for a specific moment could be based on a set of frames' classification results instead of on only a unique frame. This assumption comes mainly because classifying frames with a frame rate of 30 *fps* is very exhaustive and incompatible with our human perception to process this flow of results. Thus, producing a final decision for a specific moment based on the classification results of a set of frames could pose a solution to produce a final decision with a timing fitting our human perception and still accurate.

To perform this final decision based on the classification results of a set of frames, which we call **filtering** of the predicted labels, we establish a window of frames with a specific size, from which we compute its median result. The computed median result will be our final decision, the compensation behavior detected, as illustrated in figure 3.25. In the case of multilabel classification, the median is computed for each label.

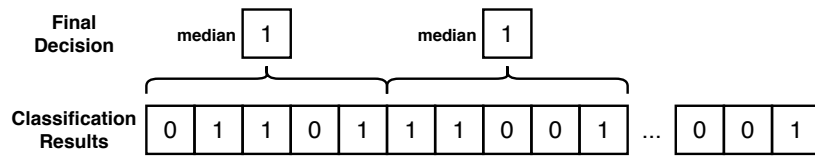


Figure 3.25: Filtering of the classification results.

4

Experiments

This chapter presents all the experimental procedures conducted to evaluate the Virtual Coach (VC) and validate the compensation assessment approaches. First, we describe the dataset used to train and validate our classification models and the three upper limb exercises under study. We detail the dataset labeling process, which allows us to perform Multilabel Classification (MLC), and the dataset cleansing process. Given we are dealing with a Multilabel Dataset (MLD), we present its main characteristics. We validate the proposed kinematic variables, which are the features for our Rule-based (RB) method, and describe the explored architectures and hyperparameters for the NN based approach. Finally, we detail the experimental procedure that enables the validation of the developed VC and UI.

4.1 Dataset

In this work, to train and validate our classification models, we use the dataset from Lee *et al.* [6] work. This dataset is a set of videos of 15 post-stroke survivors performing three exercises, to which the authors attributed a patient ID from P01 to P15. Lee and his team collected movement trials' videos using a Microsoft Kinect sensor with an average frame rate of $F = 30 \text{ Hz}$ and a resolution of $640 \times 480 \text{ pixels}$ [6].

The 15 participants, with an average and Standard Deviation (SD) age of 63 ± 11.43 years old, suffered a stroke and were left with a more affected body side (left or right) [6]. Each participant performed an average of 10 trials for each exercise, which resulted in a dataset with 448 videos. Section 2.2.2 already details the three upper limb exercises (E1, E2, and E3), when we describe Lee *et al.* [6] work. Table A.2 from appendix A illustrates each exercise the patient's arm initial and target position.

4.1.1 Labeling Process

One of this work's objectives is to assess patients' movements in realtime and determine the different compensation behaviors patients might describe. With this intent, and to perform MLC, we labeled the entire dataset, which poses one of this work's contributions. To have a MLD for realtime classification, we converted the videos into sequences of images regarding their frame rate. We developed a Python program, resorting to the FFmpeg multimedia framework to convert the videos into images. This way, we have a set of 79963 video frames as described in table 4.1, which we individually labeled.

Exercise	#Patients	#Videos/Trials	#Frames
E1	15	150	28687
E2	15	149	25329
E3	15	149	25947
	Total	448	79963

Table 4.1: The number of video frames per exercise that compose the dataset before data cleansing.

Acknowledging the distinct compensation patterns mentioned in section 2.1.2 and the categories specified in 3.2.4 - **Trunk Forward (TF)**, **Trunk Rotation (TR)**, **Shoulder Elevation (SE)**, **Other (O)** trunk displacements, and **Normal (N)** - we specify five labels. Each label is decoded by an integer and designates a compensation pattern. According to the compensation behaviors observed while trying to reach the target position, a label or a set of labels is assigned to every frame. Label '0: *Trunk Forward*' is assigned to frames in which the patient moves the trunk forward. '1: *Trunk Rotation*' is for torso rotation. Label '2: *Shoulder Elevation*' is for subjects describing an excessive shoulder elevation angle. '3: *Other*' is for other trunk displacements, such as trunk moving backward and torso tilt. Finally, '4: *Normal*' is assigned to the frames in which the patient does not perform any compensation behavior or is resting. Figure A.3 from appendix A shows an example for each compensation behavior observed in the dataset and their corresponding labels.

For this labeling process, we consulted a Physical therapist and an Occupational therapist. They helped us recognize compensation patterns and identify the moments when those behaviors begin or end. Therapists also advised us, in cases of labeling uncertainty, to choose the more noticeable pattern over another. Thus, the labeling assignment only considers movements in which the compensation is very pronounced. When we observed more than one compensation behavior, the corresponding labels were assigned if those patterns are visibly distinct. In doubt, the label assigned corresponds to the most evident movement. For the '3: *Other*' label, could be considered compensation movements with the head, commonly visible in E1 during the "drinking" phase. However, due to this pattern's lack of representation, this compensation behavior was ignored. We assigned labels to every frame in which the compensation patterns were visible, even to the frames in which the behavior was only beginning or ending. In these frames, in which compensation is beginning or ending, which we call **borders**, the patterns are not extremely obvious.

In summary, in E1, patients only demonstrate **Trunk Rotation**, **Shoulder Elevation**, and **Other** compensation patterns. **Shoulder Elevation** is the most frequently observed pattern. In E2, patients describe compensation behaviors similar to E1. For both exercises, subjects do not reveal the **Trunk Forward** pattern. In E3, we can witness **Trunk Forward** behaviors when subjects move the cane forward and **Shoulder Elevation** when they return it to the initial position. To enhance certainty concerning the observed patterns, we repeatedly observed the exercises' videos, video frames, and OpenPose body skeletons. Table 4.2, exposes the label occurrences for each exercise at this point.

	TF	TR	SE	O	N
E1	0	1059	10330	5822	17946
E2	0	910	5795	3162	17631
E3	5538	0	1241	0	19647

Table 4.2: Label occurrence before data cleansing.

4.1.2 Data Cleansing

After keypoint extraction with OpenPose, it is important to consider three different situations: the presence of other people in the image beside the patient, extra skeletons detected by OpenPose, which do not necessarily belong to a person, and keypoint misdetection.

As illustrated in figure 4.1, the presence of other people in the image is easily overcome with the method proposed in section 3.2.2. As explained before, we consider a disk with its center in the image center. If more than one person is detected in the image, the patient should be placed inside the disk. If both subjects are inside the disk, the patient is the one closest to its center.

Another situation that deserves attention is the detection of false skeletons by OpenPose. Since OpenPose detects body parts in an image, it can detect body parts in other objects, not necessarily a person, or detect body parts from people that partially appear. Figure 4.2 shows three examples of these situations.

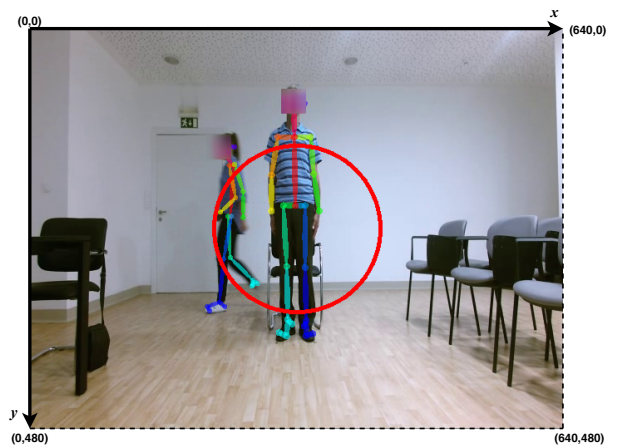
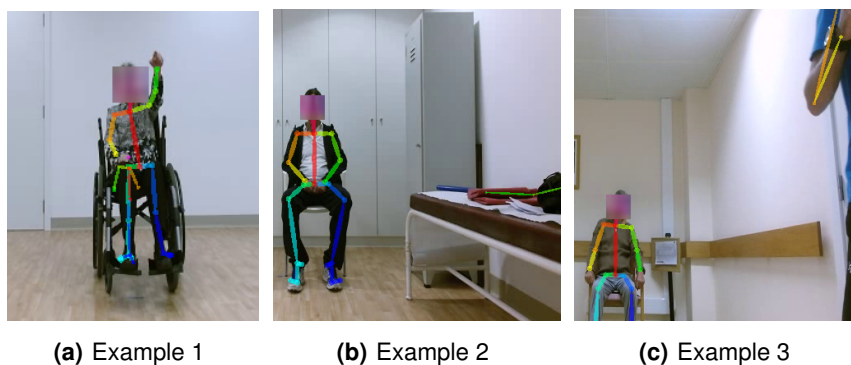


Figure 4.1: Illustration of the method adopted to select the significant subject present in a video frame.



(a) Example 1

(b) Example 2

(c) Example 3

Figure 4.2: Examples of extra skeletons detected by OpenPose.

In the data provided by OpenPose, every extra skeleton is like another person in the image. A person of interest, the patient, is someone whose spine is detected by OpenPose, which means that the three spine keypoints *Nose*, *Neck*, and *MidHip* must exist, i.e., must have a confidence score higher than zero:

$s_{ns}^t > 0$, $s_{nk}^t > 0$, and $s_{mhp}^t > 0$. Every extra skeleton whose spine is not detected is discarded.

The third situation is the keypoint misdetection or poor detection. Keypoints' confidence score reveals the more obvious misdetections. Thus, we consider a set of relevant joints among the selected joints, without which we cannot perform the desired processing. We remember the set of selected joints (figure 3.13) - J_s for scenario S1 and $J_{S_{right}}$ and $J_{S_{left}}$ for scenario S2 and S3 (section 3.2.2). These relevant joints are $\{ns, nk, rsh, reb, rwr, lsh, mhp, rhp\}$ and $\{ns, nk, rsh, lsh, leb, lwr, mhp, lhp\}$ for right and left affected sides, respectively. For these joints, their confidence score has to be higher than a defined threshold. This threshold was determined through image observation, which revealed that all the keypoints with a confidence score below this value are significantly displaced from their true location. For the remaining joints, it is a required condition a positive confidence score, $s_j^t > 0$. Every frame not fitting these conditions is removed from the data because they make the desired processing impossible.

Still, in a keypoints' misdetection situation, there are cases with keypoints really badly detected but with a fairly reasonable confidence score value. Figures 4.4(a) and 4.4(b) show two examples of this situation, in which OpenPose provided a wrist keypoint in a different position from its real position. For these instances, we correct the keypoints using the MATLAB `imshow` function, which allows displaying an image, analyze its properties, and determine the position in the image of any desired point.

Another observed issue worthy of mention is the extreme keypoint misdetection for patient P12 [6] in E1. In this particular case, the subject is wearing a light color sweatshirt, quite close to white, placed in front of a white wall. As OpenPose detects body parts regarding image characteristics, it has extreme difficulty detecting body joints correctly under these circumstances. Figure 4.3(c) and 4.3(d) give two examples of this very poor keypoint detection. Due to the high number of keypoints badly detected in this case, patient P12 was removed from E1's data.

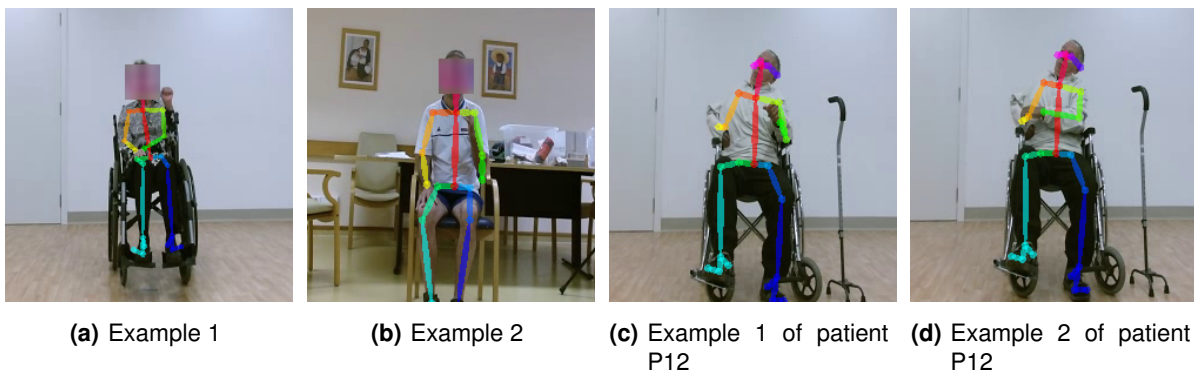


Figure 4.3: OpenPose keypoint misdetection with a reasonable confidence score value and for patient P12 in E1.

4.1.3 The Three Upper Extremity Exercises

In section 3.2.3, we consider three possible scenarios S1, S2, and S3 regarding patients' positioning in the image that should be respected in this processing step. In scenario S1 a subject is placed in a parallel position in front of the recording camera, facing it. In scenarios S2 and S3, a subject in a perpendicular or oblique position, respectively, with the affected arm facing the camera. It is fundamental to relate these scenarios with the three exercises and cases included in Lee *et al.* dataset [6].

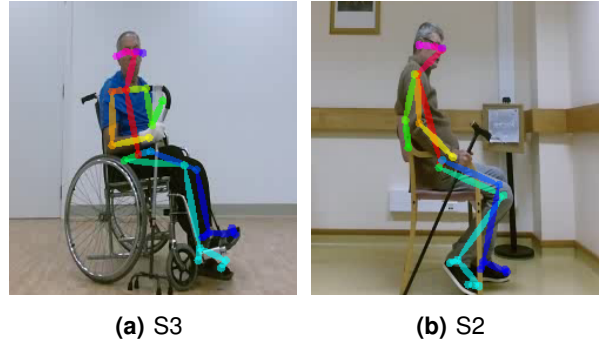


Figure 4.4: Patients in and perpendicular (S2) and oblique (S3) position regarding the recording camera.

In exercises E1 and E2, the study participants are positioned according to S1. In E3, the patients are commonly completely aside with their affected arm facing the camera, according to S2. Also, for E3, some patients are in an oblique position with their chest visible, suiting S3. This way, the **Data Normalization** step is applied to each exercise following the respective patient positioning scenario. For the RB classification method, the kinematic variables used as features are also distinct for each scenario, as described in 3.2.3.D.

4.1.4 Multilabel Dataset Characteristics

After all the data cleansing, we have to look at the final dataset obtained from this process. This final dataset is the one used to train and validate our Multilabel Classification (MLC) models. At this point, we have a reduced set of videos and frames, as shown in table 4.3, and consequently a lower number of labels occurrences summarized in table 4.4. As stated in [40], before developing a classification model and training it, it is crucial to know our dataset better to decide which model we should build. With this purpose, we present a set of metrics that characterize a Multilabel Dataset (MLD) concerning its multilabel nature, verifying the extent to which the data samples have more than one label assigned. While defining these metrics, we use the same notation introduced in section 3.2.4 to define a multilabel problem.

Exercise	#Patients	#Videos/Trials	#Frames
E1	14	140	25584
E2	15	149	25207
E3	15	149	25866
Total		438	76657

Table 4.3: Dataset after data cleansing.

The first metric is called cardinality, or *Card*, which is given by expression (4.1). This metric counts the number of active labels per sample. It then averages the sum to the total number of data samples in the

	TF	TR	SE	O	N
E1	0	1059	7989	3488	17184
E2	0	910	5783	3160	17521
E3	5530	0	1241	0	19574

Table 4.4: Label occurrence after data cleansing (and without patient P12 for E1).

dataset. A high *Card* value means more relevant labels for each sample, i.e., the dataset is surely multilabel. High cardinality is commonly associated with a dataset with a high number of labels per sample. A low *Card*, near 1, means that one label is active for most data samples, i.e., the dataset is poorly multilabel.

$$Card(\mathcal{D}) = \frac{1}{n} \sum_{i=1}^n |Y_i| \quad (4.1)$$

As established in section 3.2.4, when we defined a multilabel problem, \mathcal{D} denotes the MLD, n is the total number of data samples, and Y_i is the labelset assigned to the i th sample.

Another metric is the density, or *Dens*, a normalized version of the *Card*. It is the *Card* divided by the number of labels in the dataset, k . Thus, *Dens* is given by the expression (4.2). This metric expresses how representative the labels are in each sample in the dataset. High *Dens* values imply that the labels are well represented for each sample. On the contrary, low *Dens* values mean more label dispersion, i.e., only some labels are significant for most samples.

$$Dens(\mathcal{D}) = \frac{1}{k} \frac{1}{n} \sum_{i=1}^n |Y_i| \quad (4.2)$$

Additionally, we also mention the P_{min} metric, given by the expression 4.3. This measure is a percentage of the data samples that have only one label active. A high P_{min} percentage means that most samples are single labeled, and the dataset is not greatly multilabel.

$$P_{min}(\mathcal{D}) = \sum_{y' \in Y / |y'|=1} \frac{|y'|}{n} \quad (4.3)$$

One issue associated with the MLDs is label imbalance. Imbalanced MLDs have labels much more prevalent than others. This issue poses a challenge when building and training multilabel classifiers and influences the classification outcomes. To assess label imbalance we use the metrics *IRLbl*, given by expressions (4.4). In the expression, the operator $\llbracket expression \rrbracket$ returns 1 if the *expression* inside is true and 0 otherwise. By computing the ratio between the number of occurrences of the most frequent label and the other labels, we can know the dataset's imbalance level concerning each label. For the most frequent label, $IRLbl = 1$, and for the less frequent, it will be bigger than 1. Rarer the label higher the *IRLbl* value.

$$IRLbl(l) = \frac{\max_{l' \in \mathcal{L}} (\sum_{i=1}^n \llbracket l' \in Y_i \rrbracket)}{\sum_{i=1}^n \llbracket l \in Y_i \rrbracket} \quad (4.4)$$

With the described metrics, we determine the characteristics of our MLD and assess its multilabel nature and label imbalance level. Table 4.5 compiles the metrics for each rehabilitation exercise, in which n is the number of samples, in this context the number of video frames, and k is the number of labels. These calculi do not include labels with no occurrences. As we can see in table 4.5 each exercise datasets has a low $Card$, near to 1, which means that most data samples are single labeled. The $Dens$ metric also has a low value for the three exercises, meaning labels are not well represented for each frame. Additionally, the three exercises have a high percentage of the P_{min} metric, which indicates that most samples only have one active label. Thus, with these metrics, we conclude that our dataset has a poor multilabel nature.

Regarding label imbalance, the $IRLbl$ metric for each label in table 4.6 indicates for E1 and E2 ‘1: Trunk Rotation’ is the less frequent label, being poorly represented in the dataset. For E3, the label ‘2: Shoulder Elevation’ has a low number of occurrences. For the three exercises the label ‘4: Normal’ is the most common one thus $IRLbl = 1$.

	n	k	$Card$	$Dens$	$P_{min}\%$
E1	25584	4	1.1617	0.2904	83.8336
E2	25207	4	1.0860	0.2715	91.4032
E3	25866	3	1.0185	0.3395	98.1481

Table 4.5: Dataset characteristics.

	‘0: TF’	‘1: TR’	‘2: SE’	‘3: O’	‘4: N’
E1	–	16.2266	2.151	4.9266	1
E2	–	19.2538	3.0297	5.5446	1
E3	3.5396	–	15.7728	–	1

Table 4.6: $IRLbl$ for each label $l \in \mathcal{L}$.

4.2 Classification Methods

Once we have our Multilabel Dataset (MLD), data cleaned, keypoints normalized, and we have perfect knowledge about its characteristics, we can set the threshold values for the Rule-based (RB) method and train the NN based classifier, and validate our models. This section describes the adopted metrics to evaluate our models’ performance and validate them. For the RB method, we validate the kinematic variables presented and hypothesized in section 3.2.3.D, i.e., we check if these kinematic variables can detect compensation patterns as we hypothesized. For the NN, we describe the explored hyperparameters. Additionally, we describe two experiments to apply to the obtained classification results.

We apply the classification methods to the normalized keypoints raw signal from the video frames and filtered signal. To filter the body keypoints signal, we applied a moving average filter with a window of five frames as in [6]. This procedure reduces the noise from keypoint acquisition.

4.2.1 Evaluation Metrics

To evaluate our classification models' performance on predicting compensation patterns from video frames, we need a set of performance metrics appropriate to a MLC problem.

While in the binary context, the output result from a classifier can only be considered correct or incorrect, in the multilabel field, the provided output is a set of labels, being considered completely correct, partially correct, or totally incorrect [40]. This way, we need adequate metrics that acknowledge these possibilities. In the multilabel field, evaluation metrics are categorized concerning two criteria "how the prediction is computed" (1) and "how the result is provided" (2) [40]. The (1) refers to the application of the measure by sample or by label. The (2) concerns the output provided as a binary bipartition or a label ranking. In this work, we only consider binary bipartition, which consists of an output provided as a vector of 0s and 1s, indicating the active labels for a data sample. These metrics work with the counts of *True Positives (TP)*, *True Negatives (TN)*, *False Positives (FP)*, and *False Negatives (FN)*.

According with the criteria (1), two different groups of metrics emerge: *example-based* and *label-based metrics*. *Example-based metrics* are calculated individually for each sample and averaged. *Label-based metrics* are computed for each label and then averaged according with two strategies *macro-averaging* and *micro-averaging*. In *macro-averaging* the metric is computed for each label and the result is averaged dividing by the number of labels, k . This averaging method gives each label the same weight whether the label is extremely frequent or infrequent, highlighting low performance on rare labels. In *micro-averaging*, the counters of correct and incorrect predictions are joined together and then the metric is calculated, this way infrequent labels are diluted between the most frequent ones [40, 43]. Equations (4.5) and (4.6) define these averaging strategies, respectively, where *EvalMet* stands for any selected evaluation metric [40].

$$MacroMet = \frac{1}{k} \sum_{l \in \mathcal{L}} EvalMet(TP_l, FP_l, TN_l, FN_l) \quad (4.5)$$

$$MicroMet = EvalMet\left(\sum_{l \in \mathcal{L}} TP_l, \sum_{l \in \mathcal{L}} FP_l, \sum_{l \in \mathcal{L}} TN_l, \sum_{l \in \mathcal{L}} FN_l\right) \quad (4.6)$$

For the multilabel context we use *Precision*, *Recall*, F_1 score, F_β score, and *HammingLoss* [40, 43] to evaluate our models. These are example-based metrics, and the first four are calculated according the two averaging strategies - *macro-averaging* and *micro-averaging*. To evaluate binary classifiers we use *Accuracy*, *Precision*, *Recall*, F_1 score, and the *Mean Squared Error* metrics. These evaluation metrics are computed utilizing Python machine learning library, 'Scikit-learn' [43].

Precision expresses the classifier's capability of not to label as positive a negative sample. This metric is defined by the ratio between the number of correct predictions and total predictions given by (4.7). In multilabel, it can be understood as the percentage of predicted labels truly significant for the sample [40, 43]. In this context, we give this metric special importance. We consider more valuable a classifier capable of accurately predict true compensatory patterns and, in error, assume a normal movement (even if the subject

is describing compensation), then indicate compensation for a normal movement.

$$Precision = \frac{1}{n} \sum_{i=1}^n \frac{|Y_i \cap Z_i|}{|Z_i|} = \frac{TP}{TP + FP} \quad (4.7)$$

Recall refers to the classifier's ability to detect all positive samples. It is the percentage of correctly predicted labels in between all the truly active labels. It is defined by the ratio (4.8) [40, 43].

$$Recall = \frac{1}{n} \sum_{i=1}^n \frac{|Y_i \cap Z_i|}{|Y_i|} = \frac{TP}{TP + FN} \quad (4.8)$$

F_1 and F_β scores are an weighed harmonic mean that combines *Precision* and *Recall*. In F_1 , *Precision* and *Recall* contribute equally (equation (4.9)). In F_β score, given by (4.10), the β parameter regulates the weight of *Recall*. For $\beta < 1$, *Precision* has a higher contribution, while $\beta > 1$ *Recall* has more weight in the metric [40, 43]. Since *Precision* is more importance to us, we give *Recall* a weight of $\beta = 0.5$.

$$F_1 = 2 \times \frac{Precision \times Recall}{Precision + Recall} \quad (4.9)$$

$$F_\beta = (1 + \beta^2) \frac{Precision \times Recall}{\beta^2 Precision + Recall} \quad (4.10)$$

Hamming Loss is the portion of mispredicted labels, given by equation (4.11). The operator $|\cdot|$ counts the number of differences between the predicted and true labels. The number of prediction errors is accumulated and normalized by the number of samples and labels [40, 43].

$$HammingLoss = \frac{1}{n} \frac{1}{k} \sum_{i=1}^n |Y_i \neq Z_i| \quad (4.11)$$

4.2.2 Validation Method

After defining our classification methods, we need to select the best model hyperparameters that regulate our learning model's behavior. Model hyperparameters include NNs' topology, number of layers, and neurons. Learning algorithm hyperparameters are the learning rate, optimization method, and mini-batch size. These settings are difficult to optimize through the training procedure. Also, training them on the training set would lead to overfitting. This way, splitting the data set into a *training* set for learning model weights and *validation* set to guide hyperparameters selection is the recommended approach. Once the learning process is finished, we need to evaluate the algorithm to determine if it fulfills its purpose of identifying compensation motor patterns with a good performance level and generalization capacity [39, 43].

Since we have a pretty small dataset in which each stroke survivor executes a task with practically the same compensation patterns among movement trials with some labels poorly or nothing represented, we resort to *cross-validation* to evaluate our models' predictive ability and ensure generalization. Cross-

validation consists of partitioning the dataset into small subsets. In the validation loop, all the sets except one are used for training, and the remaining set is used for validation [39, 43]. In the end, the performance measure determined in each loop, i.e., for each small subset, is averaged. Since all patients in a post-stroke status present their own motor pattern, we apply *Leave-One-Subject-Out* (LOSO) cross-validation. Validating the models on each patient compensation pattern enables a better understanding of their classification performance and generalization capacity.

4.2.3 Rule-based Classification Method

At this stage, it is essential to validate the formulated hypotheses regarding the kinematic variables defined in section 3.2.3.D. This step is crucial to prove our hypotheses' utility and efficiency to assess motor compensation from 2D data. Additionally, the kinematic variables' analysis allows us to determine the thresholds values that condition the existence of compensation for the defined rules in section 3.2.4.A.

We split the kinematic variables' validation splits into three parts. First, we validate the stated hypotheses to assess movements in depth since we only work in 2D. More specifically, the hypotheses to assess **Trunk Forward** and the **Other** compensation behavior, trunk moving backward, for S1 and S2, and trunk tilt for S3 (section 3.2.3.D). We intend to approve both or one hypothesis and choose the best one. Second, we present the kinematic variables' variation over time regarding their initial value ($t = 1$), proving they can assess the expected motion pattern with the dataset's data. Third, we present additional experiments validating the kinematic variables that assess motion behaviors not observed in the dataset.

4.2.3.A Hypotheses to Assess Movements in Depth

As mentioned, we formulated two hypotheses to detect movements performed in depth to overcome the lack of 3D positional information. To validate these hypotheses we recorded a video in similar conditions of the dataset videos [6], with the camera placed nearly 2.5 m away from the subject, and with a resolution of $640 \times 480\text{ pixels}$. In these videos, the subject simulated all the compensation patterns implying a translation in depth of the torso for the three exercises. For E1 and E2, the subject simulated **Trunk Forward** and trunk moving backward (**Other**) during reaching. For E3, the subject tilted the torso (**Other**) while moving the cane forward and backward. In this experiment, the subject is a researcher with knowledge of this thematic and indirect experience with rehabilitation after stroke.

The first hypothesis is to assess depth movements tracking the changes in the subject's head size. Thus, we look at the head's area variation over time for each movement pattern and exercise. Figures A.5 to A.7, from appendix A.5.1, expose the changes in the head size while the subject is moving the trunk forward and backward for E1 and E2, and tilting the torso for E3. We infer that our hypothesis is not correct. When the subject moves forward, the head's area (figure 3.17(a)) decreases due to a reduction in the distance between the *Nose* and *Neck* joints. Similarly, when the subject moves the torso backward, the head's area

gets smaller, validating our hypothesis. In E3, when the subject tilts the torso while moving the cane forward and backward to its initial position, the absolute head size gets bigger, confirming our hypothesis.

To support this hypothesis's validation, it is relevant to examine head area changes in dataset cases since we are working with them and following the recording conditions [6]. However, in the dataset, only trunk moving backward (**Other**) is observed. Figure 4.5 exposes an example for E2 (A.8 for E1). Also, in these cases, the head size reduces when the patient moves the trunk backward. Then, we can affirm that through head size variations, we can assess this compensation pattern.

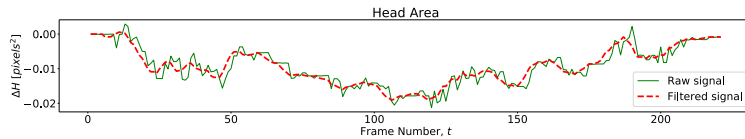


Figure 4.5: Head area over time, revealing trunk moving backward (**Other**) observed in the dataset for E2.

The second formulated hypothesis implies assessing the head's translation in ${}^W Z$ provided by the OpenFace toolkit. Figures A.9 to A.11, from appendix A.5.1, show the variation of the head's pose when the subject moves the trunk and backward for E1 and E2, and tilts the torso for E3. When the subject moves the trunk forward and backward, the head performs a negative and positive translation in ${}^W Z$, respectively, confirming the hypothesis. However, in figure A.10 for E2, we can notice that OpenFace could not detect head translation under the recording conditions. Another example is the one from figure A.11, for which OpenFace also could not detect head displacements, being the signal provided null.

Again, to support these results, we use the same example of a trunk moving backward from the dataset for E2 (A.12 for E1). Equally, as shown in figure 4.6, OpenFace provides a null signal, making impossible for us to assess head's pose in depth.

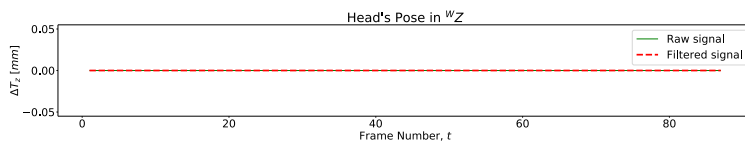


Figure 4.6: Head translation over time in ${}^W Z$, revealing trunk moving backward (**Other**) observed in the dataset for E2.

Based on these results, we consider that OpenFace cannot detect head pose from an image with low resolution, 640×480 pixels, and the subject placed at a significant distance away from the camera. Also, OpenFace has increased difficulty in detecting head's pose when the subject's face is not facing the camera, which makes sense since OpenFace derives head pose from the detected facial landmarks, which is proven with the example exposed for E3 in figure A.11.

To support this theory, we recorded videos in the same conditions (subject ≈ 2.5 m from the camera), with a higher recording resolution, 1920×1080 pixels, to verify if in this case there are signal loss. Figures A.13 and A.14 show head's translation in ${}^W Z$ when the subject moves the torso forward and backward for

E1 and E2. As observed, with a higher resolution video, OpenFace could measure the head pose and the changes in the pose support our formulated hypothesis. Additionally, figure A.15 shows the translation in ${}^W Z$ for E3 in which the subject tilt the trunk while moving the cane forward and backward to its initial position. In this case, OpenFace successfully detected translation since the displayed curve presents a noticeable absolute change in the head pose in ${}^W Z$.

With all the presented outcomes to validate the formulated hypotheses to assess movements in depth, or choose the one that best fits our demands, we decided to move on with the *hypothesis 1* - detection of the compensation patterns through the head area. We chose this hypothesis mainly because it is the one that enables us to detect movements in depth with the data provided in the dataset. Also, it prevents us from confronting situations of signal loss in face landmark detection.

4.2.3.B Kinematic Variables to Assess Motion Patterns Observed in the Dataset

Having chosen the hypothesis that best assesses movements in depth, it is now important to validate the kinematic variables used to detect other compensation patterns with data from the dataset. When we evaluated the hypotheses to assess movements in depth, kinematic variables to assess torso moving backward (**Other**) behaviors were validated. Now, we need to validate the other compensation patterns observed in the dataset - **Trunk Rotation** and trunk to tilt (**Other**) in E1 and E2, **Trunk Forward** in E3, and **Shoulder Elevation** in the three exercises.

The formulated hypothesis to assess torso rotation is detailed in section 3.2.3.D. Figure 4.7 illustrates an example from the dataset of **Trunk Rotation** for E1 (A.16 for E2). The patient rotates the torso while trying to reach the target position. As it is observable, the displayed curve shows exactly what we hypothesized, with the elevation angle of both shoulders (falling for the opposite shoulder) exhibiting a similar pattern.

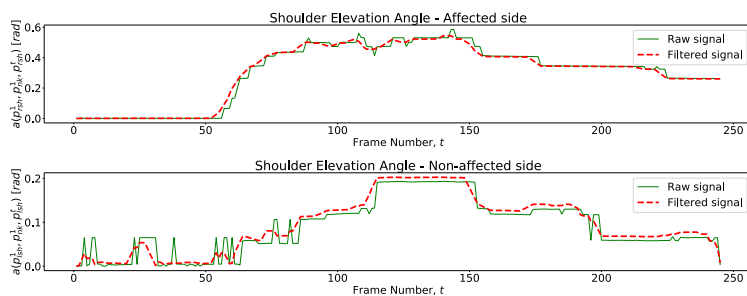


Figure 4.7: Patient shoulders' elevation angles over time describing **Trunk Rotation** for E1.

As in previous works, shoulder elevation is measured in E1 and E2 with the shoulder elevation angle. Figure 4.8 illustrates an example of a patient executing this compensation behavior for E2 (A.17 for E1), in which the presented curves show clearly the elevation.

For E1 and E2, we also validate the trunk tilt compensation pattern (**Other**). Figure 4.9 illustrates this compensation pattern measured with the tilted angle of the spine for E2 (A.18 for E1).

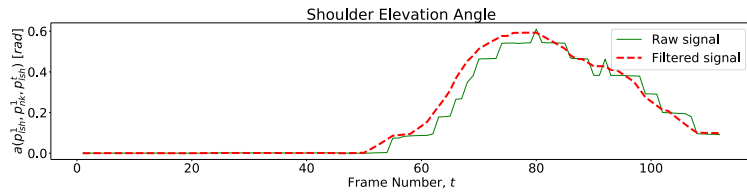


Figure 4.8: Patient affected shoulder elevation angle revealing **Shoulder Elevation** for E2.

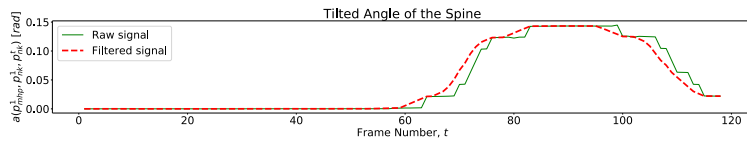


Figure 4.9: Patient tilted angle of the torso describing a trunk tilt (**Other**) for E2.

In E3, only **Trunk Forward** and **Shoulder Elevation** are observed. Figure 4.10 shows the tilted angle of the spine and a positive displacement of the *Neck* joint regarding its initial position, which proves the **Trunk Forward** movement occurrence since the ${}^B X$ axis is directed to the front of the subject. Figure 4.11 presents the variable used to assess **Shoulder Elevation**, the displacement of the shoulder in ${}^B X$ regarding the *Neck* joint. As we can see, the elevation is noticed from frame 120, which makes sense since shoulder elevation is mainly visible when the patient is trying to take the cane back to its initial position.

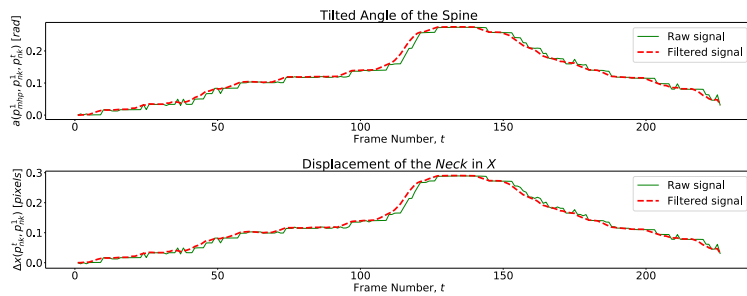


Figure 4.10: Patient P06 tilted and of the spine and shoulder displacement over time, describing **Trunk Forward** in E3.

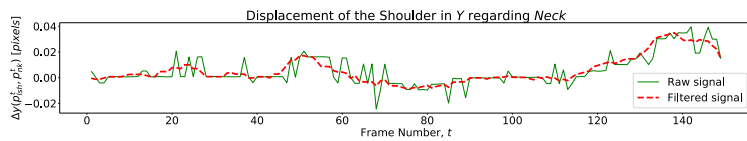


Figure 4.11: Patient P05 shoulder displacement over time, describing **Shoulder Elevation** in E3.

Since not all the compensation patterns considered in this work are observed in the dataset for each exercise, we recorded new videos to validate the kinematic variables that assess them. These videos follow the dataset video collection conditions and properties. At this stage we only investigate torso moving backward and torso rotation patterns in the scenario of E3. Regarding this experiment, the results are displayed in section A.5.3 from appendix A.

4.2.4 Neural Network Classification Method

In section 3.2.4.B, we describe our NN approach to assess compensation motor patterns, which consists of an ensemble of a binary (C1) and a multilabel (C2) classifiers. First, with C1, we apply binary classification to determine if compensation behaviors exist in video frames. Second, we perform MLC with C2 to specify the described compensation patterns in the frames inferred from C1 results. Following Lee *et al.* [6] work and since we are working with the same dataset, for both classifiers we explore diverse model architectures with one to three hidden layers with several hidden units (i.e. {16, 24, 32, 48, 64, 96, 128, 192, 256, 384, 512} hidden units). We apply an adaptive learning rate with values for the initial learning rate of {0.0001, 0.0005, 0.001, 0.005, 0.01, 0.05, 0.1}. We also adopt the 'Adam Optimizer' and a mini-batch size of 5. With a small series of experiments, we selected a number for the maximum iterations of 550, in which the solver converges. Regarding hidden layer activation function, for C1 we selected 'ReLU' function and for C2 we tested with 'ReLU' and 'tanh' activation functions. Additionally, for C2, we apply *One-vs-Rest* multilabel strategy to deal with this multilabel problem, mentioned in section 3.2.4. We implement and train our classifiers using 'Scikit-learn' Python library [43].

4.2.5 Filtering of the Classification Results

Section 3.2.5 expresses the assumption of needing to filter the classifier results. As described, the filtering step determines the median value of a set of video frames' classification results, which leads to the final classification decision. Given the results obtained with our classifiers for each video frame, we examine this filtering process for diverse frame window sizes - {5, 7, 9, 11} frames - in which we compute the median predicted label. Thus, with the evaluation metrics from 4.2.1, we evaluate the median predicted labels concerning the median true labels for each window size.

From the labeling process described in 4.1.1, another issue emerges regarding the results that we might obtain in classification. While labeling the dataset, we assign labels to frames indicating compensation patterns, in which those are visible, and thus including frames in which a patient is starting or ending describing a compensation. In these, patterns are not very pronounced. We call these sets of frames of **borders**. Given this, we assume a high probability of low classification performance when identifying motion patterns' transitions - from 'Normal' movement pattern to compensation patterns and vice-versa.

Figure 4.12 schemes what we consider as borders in the set of frames of a movement trial video. In "case 1", the chart illustrates a subject describing a 'Normal' movement without pattern transitions and thus without borders. In "case 2" is shown a transition from a 'Normal' movement to a compensation behavior. In "case 3" and "case 4", there are two transitions. Suppose the patient describes an isolated compensation pattern during a small set of frames, below a defined number of frames that compose a border. In that case, we consider a border the first transition set of frames. Otherwise, a border is every transition at the beginning and end of a compensation description. We experiment with borders of 5 and 10 frames. To

verify if classifying the frames composing the borders affect performance metrics, we remove the borders from the predicted labels and true labels and apply the evaluation metrics again.

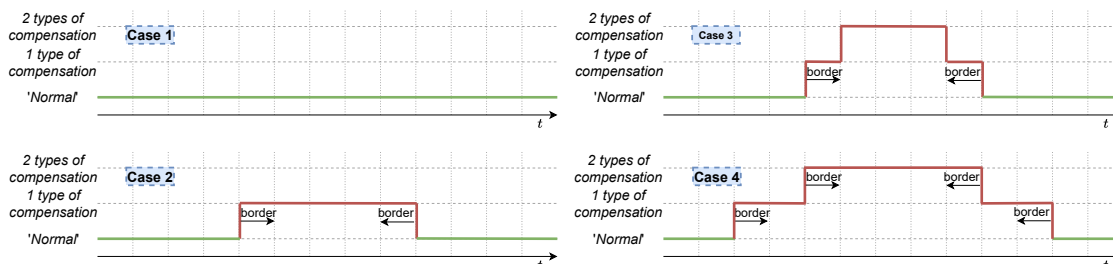


Figure 4.12: Movement pattern transitions described in a set of frames which we call **borders**.

4.3 Virtual Coach

We evaluate our VC through a set of experiments with a group of volunteers. We aim to investigate the system's hedonic and utilitarian value, systems' performance, and users' use intention. Thus, we introduce the following hypotheses:

- H1** There is a disparity between stroke survivors volunteers and the other volunteers among the different perceptions about the system;
- H2** There is a difference in the perceived virtual coach utilitarian value between older adults and younger adults since stroke is common among older adults and the elderly;
- H3** Hedonic value perceptions are affected by the virtual coach performance monitoring exercise performance, detecting compensation, and interaction model awarenesses;
- H4** Hedonic and utilitarian value perceptions affect users' intention to exercise with the virtual coach.

The initial experimental design was a field experiment with patients in a post-stroke status under rehabilitation in a hospital facility. Due to the current global situation of the COVID-19 pandemic, we had to adapt our experimental protocol. The experimental procedure was thought not to harm the participants. The data acquisition and storage process is in agreement with the General Data Protection Regulation (GDPR). To ensure these conditions, the Instituto Superior Técnico Ethics Committee reviewed and approved our experimental protocol. This protocol followed Čaić *et al.* previous work [44].

Before the sessions with the volunteers the VC was tested. Later, it was tested in the laboratory connected to the same WiFi network of the server.

4.3.1 Volunteers

We recruited a total of seven volunteers to exercise their limbs with our system. Given the pandemic situation, volunteers are among our closest social groups, such as family and close friends. When selecting the participants, we aimed to gather a diverse group concerning age, sex, and stroke rehabilitation experience. At least one participant had a stroke and undergone a rehabilitation process. More than one participant

had close contact with a patient under rehabilitation. All participants are over 18 years old and own their cognitive capabilities' full potential, which guarantees the inexistence of obstacles in understanding experiment directions, system instructions and cues, and the final questionnaire. Volunteers signed an Informed Consent authorizing the recording of their image necessary to the normal system operation.

4.3.2 Experimental Setup

With the will to provide an affordable and accessible solution with a simple technical infrastructure, we only use a laptop with a built-in webcam in this experiment. This is possible due to the developed UI to interact with the user and the proposed methods to assess compensation from 2D positional data extracted from the captured image by the webcam. By convenience, we use the RB classification algorithm, which enables an easy result interpretation and adjust the rules' threshold values to increase or decrease system sensibility on detecting compensation.

The sessions took place in a domestic environment spacious enough to assure experimental conditions similar to the data collection process described in [6]. The laptop was placed in front of the volunteer with an ideal distance of 2.5 m and an adequate position to capture the participant's relevant body joints.

4.3.3 Experimental Procedure

At the beginning of the session, the responsible researcher introduced the study and experimental purpose. The researcher properly explained the entire procedure and introduced the UI to the volunteer giving an overview of its functionalities. The volunteers were asked to perform the three upper extremity exercises with their arm - from their affected side due to stroke if it was the case, or from their non-dominant body side. During the exercises, patients had to simulate the different compensation strategies mentioned in the previous sections. Volunteers had to repeat the proposed movements at least five times. The entire session should not exceed 30 minutes. In the end, each participant answered a questionnaire giving their feedback about the VC and the interaction with it.

4.3.4 Data Collection

To investigate the VC perceived hedonic and utilitarian value, system's performance, and users' intention to use the system, we gather information from the volunteers through a set of questions. With the questionnaire we aim to gather quantitative and qualitative data. The volunteers responded to each question on a 5-point Likert scale (quantitative) - from '1 = *Strongly Disagree*' to '5 = *Strongly Agree*' - and a question with open answer for each item - e.g., "tell me more about it" - to gather more information on volunteers opinion and ask a deeper promote a deeper reflection (qualitative). Table A.6 in appendix A presents the questionnaire items. Due to their experience concerning stroke and stroke rehabilitation process, some questions are only answered by volunteers that followed closely a post stroke rehabilitation process. Those questions are Q4 on utilitarian value and Q1 on intention to use.

5

Evaluation

This chapter presents the experimental results. First, we evaluate the two proposed Multilabel Classification (MLC) approaches and analyze experiments applied to both methods' classification results. Second, we present the information collected from questionnaire respondents, who exercised with the Virtual Coach (VC) through the UI, and a statistical analysis which validates the hypotheses raised in section 4.3. Finally, we also present volunteers' opinions and suggestions on the VC system. These results validate our system for autonomous upper extremity rehabilitation therapy, mainly its efficiency and relevance.

5.1 Classification Results Analysis

In this section, we lean over each exercise at a time and evaluate each classification method individually with the evaluation metrics introduced in section 4.2.1. Additionally, we present the experiments performed over classification results detailed in section 4.2.5. In the end, we compare the methods for each exercise and relate the differences noticed between them.

In the previous chapter, in section 4.1.4, we characterize our Multilabel Dataset (MLD) with proper measures concerning its multilabel properties and label imbalance. Given the dataset attributes, we verify that most of the dataset samples are single labeled, i.e., in most cases, patients only describe one kind of motor compensation behavior. We can also affirm that some labels are poorly represented in the dataset. Since this indicates label imbalance, we support our decisions regarding classifiers performance in *micro-averaging* strategy for metric calculus, mentioned in 4.2.1. With *macro-averaging* strategy (section 4.2.1), we can prove how the classifiers behave when trying to determine less frequent labels.

The classifiers are applied to two types of signals: the keypoints signal acquired with OpenPose and then normalized (raw signal), and a filtered signal, which is also normalized (filtered signal). The signal is filtered using a moving average filter. We compare the results for both signals and perform the experiments on the predicted labels with the signal with which the classifiers revealed a better performance.

5.1.1 Exercise 1

As inferred from table 4.4, in section 4.1.4, for E1, **Trunk Forward** pattern is not observed, being excluded from the video frames classification process to access compensation. As noticed, most of the samples are singly labeled, with some labels poorly distributed and represented. These aspects are inferred from

the measures characterizing the E1 MLD. It has a label cardinality pretty close to 1, $Card(\mathcal{D}) = 1.1617$, revealing that most video frames have only one label assigned. Density value of $Dens(\mathcal{D}) = 0.2904$, shows poor representation of dataset labels. Additionally, the dataset's poor multilabel nature is also uncovered by P_{min} value. This measure indicates that 83.8336% of the samples are singly labeled. Regarding label imbalance, looking at each label through $IRLbl$ metric, we notice that the label '1:Trunk Rotation' is the less frequent one and label '4: Normal' is the most frequent, which makes sense since '4: Normal' is assigned to every frame in which the patient is describing a good quality movement pattern or is at rest. These findings on the E1 dataset should reflect in classification results.

5.1.1.A Rule-based Classification Method

After we have defined the threshold values that regulate compensation detection, we apply the rules, from section 3.2.4.A, directly to the kinematic variables, computed from both raw and filtered signals. Figure 5.1 displays the evaluation metrics values expressing classifier's performance for both signals.

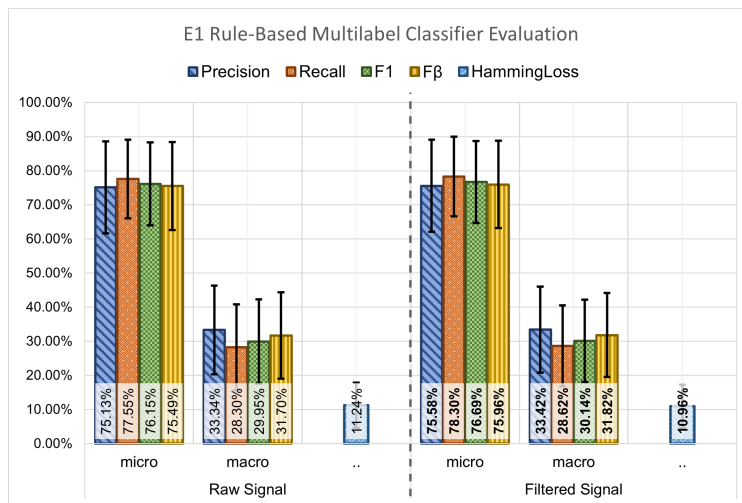


Figure 5.1: E1 Rule-Based Multilabel Classifier Evaluation Results.

In figure 5.1, we can see that our classifier fulfills its purpose with a pretty good performance for both raw and filtered signals, with a F_1 score of 76.15% and 76.69% respectively. Nevertheless, we can easily verify that the classifier performs slightly better for the filtered signal. This result is expected since the signal noise was reduced, and abrupt changes in the acquired keypoints' positions were smoothed. Without sudden and sharp variations in the signal, the application of the rules based on a threshold value limiting the existence of compensation works better. Although this difference is not significantly big, with a difference in F_1 score of 0.54%, we look at the classification results for the filtered signal in the remaining analysis. Concerning *HammingLoss*, which shows the portion of labels mispredicted, it takes a value of 10.96% for the filtered signal, which is pretty acceptable.

Looking at the results obtained with the macro-averaging strategy, we can clearly notice that metrics' values reveal a poor performance classifying rarer labels with a F_1 score of 30.14%, differing 46.55% from the F_1 score obtained through micro-averaging. This reveals that the dataset has obviously a set of rare labels poorly represented, which are badly classified, confirming the label imbalanced inferred from the dataset characterization metrics. Concerning *Recall*, this metric is very affected when we give the same weight to every label, which means that labels with few occurrences affect classifiers' ability to detect positive labels. Although this was not expected for this method because a training process is not involved - in which label frequency and distribution have a great impact that reflects on classifiers' performance - given that macro-averaging highlights rarer labels mispredictions and we are working with a tiny number of labels, we can recognize the penalty in the metrics computed through macro-averaging.

5.1.1.B Neural Network Classification Method

In this section, we evaluate the NN based approach. In this case, the classifier is trained to fit the training data and predicted the correct labels. The resulting model performance is highly influenced by the number of data samples, the number of labels, and labels' frequency and distribution.

Figure 5.2, displays the better results obtained from C1 and C2 classifiers ensemble for both raw and filtered signals. To perform binary classification, classifier C1 has one hidden layer of 16 units and an initial value for the learning rate of 0.001. Classifier C2 has one hidden layer of 64 hidden units, an initial value for the learning rate of 0.001 and uses the 'tanh' activation function.

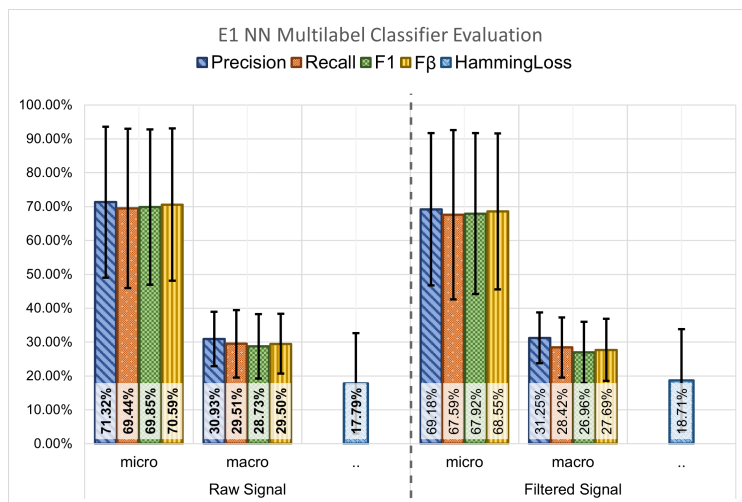


Figure 5.2: E1 NN Based Multilabel Classifier Evaluation Results.

Figure 5.2 shows that for the NN based approach we got better performance for the raw signal with a small difference of 1.93% on the F_1 score. Again, for this method, there is a big difference between the micro and macro averaging strategies to compute *Precision*, *Recall*, F_1 , and F_β , revealing the existence of rare labels and many mispredictions for these when compared with the most common labels, confirming

the dataset characteristics. However, this is an expected outcome for this method since the classifier takes as input very few samples of some labels, making difficult its learning process on how to identify those few compensation patterns properly.

5.1.2 Exercise 2

Exercise 2 has similar characteristics to E1. As described in section 4.1.4, label '1: *Trunk Rotation*' is the less frequent one, expressed by an $IRLbl$ value for this metric of 19.2538 and label '4: *Normal*' the most frequent. This exercise as a label cardinality of $Card(\mathcal{D}) = 1.0860$ and a density of $Dens(\mathcal{D}) = 0.2715$, indicating that most exercise samples have one single label assigned and rarer labels are badly represented among samples. The percentage of single labeled samples confirms this, $P_{min} = 91.4032\%$. As for E1, for E2, we verify how the label imbalance issue reflects in classifiers performance.

5.1.2.A Rule-based Classification Method

For E2, we applied the same rules defined in section 3.2.4.A, but with adequate threshold values conditioning the existence of compensation in E2 scenario. Figure 5.3 presents the evaluation results obtained with the RB classifier.

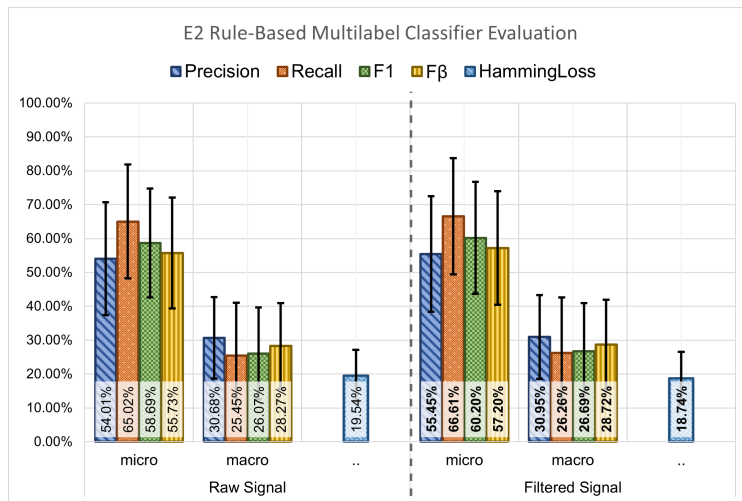


Figure 5.3: E2 Rule-Based Multilabel Classifier Evaluation Results.

As we can see in figure 5.3, for E2 the rules had a lower classification performance, with a F_1 score of 58.69% for raw signal and 60.20% for filtered signal, performing better for the latter. For the filtered signal, the classifier reveals a value of *Precision* of 55.45%, meaning that the classifier has a poorer ability to identify truly positive labels for this exercise and with its thresholds. Also, it has a percentage of mispredictions of $HammingLoss = 18.74\%$. In comparison to E1, this lower performance reveals that for E2 it was harder to find threshold values good enough to fit all the patients' movement patterns while executing the exercise.

When looking at the differences between the metrics computed through micro and macro averaging, there is a big difference between both, revealing the low number of occurrences for some labels in the dataset, which are also poorly represented. Also, it is worth noting that in macro-averaging, the value of *Precision* is higher than the value of *Recall*, giving the idea that label imbalance affects mostly classifier's capability to find all the positives labels. These differences between values from micro and macro averaging confirm the findings of the dataset characteristics previously described.

5.1.2.B Neural Network Classification Method

For E2, we figured out the best configuration for classifiers C1 and C2 for both raw and filtered signals after trying distinct combinations of hyperparameters. For binary classifier C1, we used a network of two hidden layers of 16 units and an initial value for the learning rate of 0.001. For the multilabel classifier C2, we used one hidden layer of 16 units, an initial value for the learning rate of 0.01, and *'tanh'* activation function.

With this method we obtained the results displayed in figure 5.4. It is clearly observable that, for both raw and filtered signals, this classifier had much better performance detecting compensation than the RB one, which is expected since through the training process, the classifier can fit the model's parameters to the provided input data, i.e., adapt the parameters to the different movement patterns described by the patients while executing the exercise. The model had better performance for the raw signal with a $F_1 = 72.56\%$ and *HammingLoss* = 15.02%.

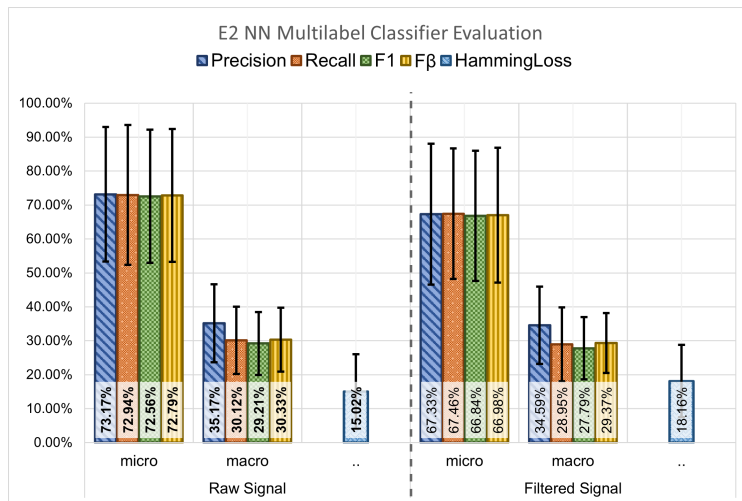


Figure 5.4: E2 NN Based Multilabel Classifier Evaluation Results.

Analyzing the values from micro and macro averaging, we see a big difference between them. For example, F_1 score shows a difference of 43.35% for the raw signal, confirming the existence of rare labels badly represented and confirming the dataset characteristics, which greatly impact classifiers' performance.

5.1.3 Exercise 3

Exercise 3 has very different characteristics from E2 and E3, starting with patients positioning scenario and kinematic variables used to assess motor compensation. For this exercise, only labels '0: Trunk Forward', '2: Shoulder Elevation', and '4: Normal' occur as detailed in table 4.4 (section 4.1.4). Thus, only three labels were considered in the model development and the classification results' analysis. This exercise has a label cardinality of 1.0185 and a label density of 0.3395, which means that the dataset is poorly multilabel. However, labels are better represented in the E3 dataset in comparison with E1 and E2. The P_{min} measure confirms the dataset poor multilabel nature indicating for E3 98.1481% of the samples are single labeled, i.e., for the entire dataset in most video frames, patients are describing only one compensation behavior. From table 4.6, we can see that as for E1 and E2 label '4: Normal' is the most frequent one, followed by '0: Trunk Forward' and '2: Shoulder Elevation', which is the less frequent with $IRLbl = 15.7728$.

5.1.3.A Rule-based Classification Method

For E3, we apply a set of rules (section 3.2.4.A) to specific kinematic variables from section 3.2.3.D which enable the assessment of motor compensation in scenario S2 concerning patient positioning in front of the camera. After we specify the threshold values for the rules that carry out compensation detection and apply them to each patient pattern, we obtain the figure 5.5 results.

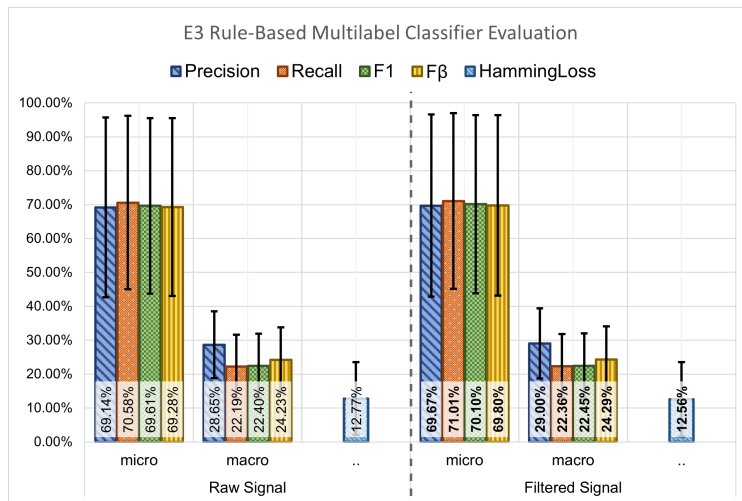


Figure 5.5: E3 Rule-Based Multilabel Classifier Evaluation Results.

In figure 5.5, evaluation metrics reveal classifiers' good performance on detecting compensation for both raw and filtered signals. However, it behaves better for the filtered signal with $F_1 = 70.10\%$, which is expected due to noise reduction. For this exercise, $HammingLoss = 12.56\%$ is lower than for E1 and E2, revealing that this classifier produces fewer mispredictions.

The difference between the metrics computed through micro and macro averaging strategies exposes

that some labels are rare and poorly represented in the dataset. This difference for F_1 score is 47.65%, which makes sense since we only have three labels for this exercise. Besides **Normal** motor patterns only **Trunk Forward** occurs with higher frequency.

5.1.3.B Neural Network Classification Method

For E3, C1 is a network with one hidden layer with 96 units and an initial value for the learning rate of 0.01. Multilabel classifier C2 has one hidden layer of 16 units, an initial value for the learning rate of 0.001, and 'tanh' as the activation function. Through LOSO cross-validation we obtain the results displayed in figure 5.6 concerning model performance on assessing compensation.

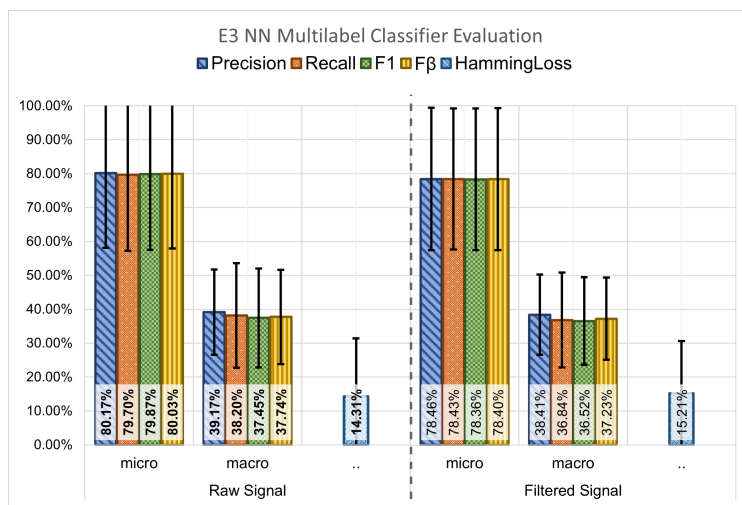


Figure 5.6: E3 NN Based Multilabel Classifier Evaluation Results.

From figure 5.6, we can understand that this model had better performance than the RB. These results are expected since a learning algorithm fits its parameters to the input data, resulting in a better prediction capacity than a simple RB algorithm. As opposed to what is observed previously for the NN based approach for E1 and E2, the classifier shows better results for the filtered signal with $F_1 = 79.87\%$. However, for this classifier, *HammingLoss* has a higher value of 14.31%, which means that even though this model has better performance, it provides more mispredictions.

Similar to what is observed previously, a big disparity between the results obtained through micro and macro averaging is visible. When all the labels have the same weight, these metrics are penalized.

5.1.4 Filtering of the Classification Results

This section analyzes the impact of filtering the classification results, i.e., filtering the predicted labels, in the evaluation metrics that reveal the models' classification performance. Classification results filtering experiment is described in section 4.2.5.

Tables 5.1 and 5.2, present F_1 score and *HammingLoss* resulting from the predicted labels filtering and borders (frames corresponding to movement patterns transitions) removal respectively, for all the exercises and classification approaches.

From table 5.1, we can see that for almost all the exercises and classification approaches, and all window sizes in which we filter the labels, the evaluation metrics increase their values, indicating a benefit from the filtering step in classification accuracy. Filtering a window of 11 frames clearly provides a better result for the RB and NN approaches for E1 and E3, respectively. Filtering a window of 9 frames gives good results for the NN for E2. For E1 NN approach, a window of 9 frames revealed an increase in the value of F_1 score, and a window of 11 frames provided a better value concerning *HammingLoss*. For E3 RB approach, the filtering produced a decrease on the F_1 score value.

From table 5.2, removing the borders result on better values for the metrics. This fact supports the assumption that the classifiers might have greater difficulty predicting correctly the labels of frames in which the patients are describing movement pattern transitions, in which compensation patterns are not much pronounced, penalizing the metrics. This assumption suits the RB classifier since it is improbable that a simple threshold is robust enough to detect a compensation pattern described for different subjects with distinct motor behaviors when the patient begins or finishes describing it. Removing borders of 10 frames provides an improvement of the evaluation metrics values.

		Result Filtering					
		E1		E2		E3	
W. Size	Metrics	RB	NN	RB	NN	RB	NN
5	F_1	76.71 %	69.87 %	60.15 %	72.99 %	69.94 %	79.92 %
	<i>H.Loss</i>	10.96 %	17.78 %	18.77 %	14.80 %	12.62 %	14.28 %
7	F_1	76.91 %	69.85 %	60.59 %	72.96 %	69.90 %	79.86 %
	<i>H.Loss</i>	10.85 %	17.82 %	18.57 %	14.79 %	12.64 %	14.31 %
9	F_1	76.95 %	70.23 %	60.25 %	73.46 %	70.08 %	79.66 %
	<i>H.Loss</i>	10.84 %	17.59 %	18.70 %	14.53 %	12.57 %	14.43 %
11	F_1	77.85 %	70.22 %	60.30 %	73.08 %	69.83 %	80.23 %
	<i>H.Loss</i>	10.48 %	17.57 %	18.67 %	14.72 %	12.67 %	14.11 %

Table 5.1: Evaluation results concerning F_1 score and *HammingLoss* after label filtering.

		Without Borders					
		E1		E2		E3	
W. Size	Metrics	RB	NN	RB	NN	RB	NN
5	F_1	77.83 %	70.86 %	60.87 %	74.42 %	71.26 %	81.47 %
	<i>H.Loss</i>	10.43 %	17.21 %	18.44 %	13.99 %	12.06 %	13.18 %
10	F_1	78.62 %	71.44 %	61.30 %	75.88 %	71.99 %	82.61 %
	<i>H.Loss</i>	10.05 %	16.95 %	18.20 %	13.17 %	11.74 %	12.40 %

Table 5.2: Evaluation results concerning F_1 score and *HammingLoss* after borders removal.

5.1.5 Result Comparison

Having evaluated each proposed approach to assess motor compensation for each rehabilitation exercise, in this section, we compare the results obtained in the evaluation metrics among the two methods and exercises. To do so, we gather, in figure 5.7, F_1 score and $HammingLoss$ measures for the signal with which classifiers better performed and highlights the better approach for each exercise.

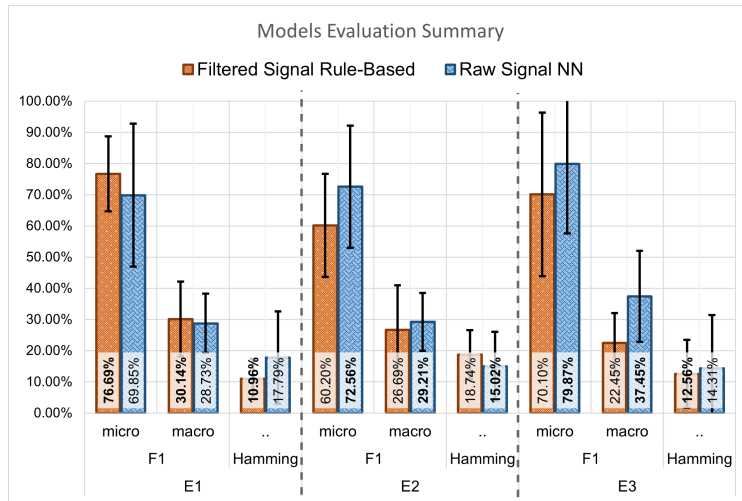


Figure 5.7: Evaluation Results for the three exercises.

Looking at figure 5.7 we can immediately notice that for E1 RB approach performs better, with a $F_1 = 76.69\%$, and for E2 and E3 the NN based approach presents better results, with $F_1 = 72.56\%$ and $F_1 = 79.87\%$ respectively. An evident difference between these exercises' datasets is their percentage of single labeled samples, P_{min} . For E1, P_{min} indicates 83.8336% of the samples are single labels. For E2 and E3, 91.4032% and 98.1481% of the samples are single labeled. This makes us believe that the RB method handles better a scenario with a not so poor multilabel nature. On the other hand, the NN based approach is more efficient when the problem is closer to a binary classification problem.

A particular case is E3. Although for this exercise the NN based approach performs better, it has a higher value of $HammingLoss$, meaning that this approach provides more mispredictions than the RB approach.

For all methods, the differences between the metrics computed through micro and macro averaging reflect the existence of extremely rare labels badly represented in the dataset, reflecting classifiers' difficulty detecting less frequent compensation patterns, which penalizes the evaluation metrics values.

Another detail noticed is the huge values of the SD. This means that the classifiers determine the compensation patterns described by some stroke survivors with better performance than others. One example that has influence in the SD values is described in appendix A.6. Another situation, also detailed in appendix A.6, is the fact that for E1 and E2, only one patient performs **Trunk Rotation**, which leads to the failure on evaluating classifiers' capacity to detect this pattern with cross-validation support the need for

more data samples, mainly representing rarer movement patterns.

In general, we obtained better results for the NN based approach, which was expected besides label imbalance since this approach automatically learns models weights to fit the input data and be prepared to detect new motor patterns. In contrast, the RB approach only works with threshold values that limit compensation existence. Since the NN based shows evidence of better performance, extremely related to the fact that our multilabel problem is very close to a binary problem, this approach would perform even better with more data, more examples of motor patterns, in particular of rarer compensation behaviors.

Concerning the RB method, although it revealed better performance just for E1, it gathers a set of benefits. This approach has the advantage of not be dependent on the number of data samples with a balanced label representation, which enables the accurate detection of compensation even when we have a small dataset. Additionally, this approach has the advantage of interpretability. Due to its simplicity, we can easily interpret the obtained results and the procedure to reach those results, a feature that is pretty relevant to this application problem - assessment of movement quality of patients performing rehabilitation exercises - since for therapists is important to know why the system identified specific motor behaviors to perform adequate patient evaluation and apply proper therapy approaches.

5.2 Virtual Coach Validation Results

In section 4.3 we detail the empirical experiment procedure conducted with a group of volunteers to investigate the perceptions of the Virtual Coach (VC) hedonic and utilitarian value, system's performance, and users' use intention. Additionally, we establish a set of hypotheses on volunteers' perceptions of the VC concerning their experience with stroke, age, and short experience with the system. This section presents the quantitative and qualitative data gathered through the questionnaires about the system and statistical analysis to investigate each raised hypothesis.

Table 5.3 displays the volunteers' profiles, with three females and four males. Three volunteers are between 25-34 years old, two are 55-64 years old, and two are 65-74 years old, which means that most volunteers are older adults.

Only volunteer V01 had a stroke and directly experienced a rehabilitation process five years ago. Before the stroke, this subject was left-handed. However, the stroke left his left side more affected, so it was evaluated during the experiment. Another subject had a rupture of the shoulder rotator tendons, restricting her movements. Therefore, this subject naturally performs compensation movements with the arm when executing reaching tasks. Other volunteers used their non-dominant side to perform the exercises.

Additionally, three volunteers experienced a rehabilitation process indirectly through a friend or relative. The remaining volunteers did not have much contact with a stroke rehabilitation process.

VID*	Age	Sex	ND/A side	(a)	(b)	(c)	(d)
V01	25-34	Male	Left	Yes	Yes	Yes	Yes
V02	55-64	Female	Left	Yes	No	Yes	Yes
V03	65-74	Female	Left	Yes	No	Yes	Yes
V04	65-74	Male	Left	Yes	No	Yes	Yes
V05	25-34	Male	Left	Yes	No	Yes	No
V06	55-64	Male	Left	Yes	No	Yes	No
V07	25-34	Female	Left	Yes	No	No	No

Table 5.3: Volunteers' profiles: (a) Knows what a stroke is (b) Had a stroke (c) Some relative or close friend had a stroke (d) Followed the rehabilitation process closely. *Volunteer ID, **Non-dominant/Affected side.

5.2.1 Quantitative Results

The questionnaire provided quantitative data illustrating volunteers' perceptions about the system's hedonic and utilitarian values, system performance, and users' intention to use. Since we only have data from seven volunteers first, we display the numeric results obtained for each item of each domain concerning the 5-point Likert scale with a violin plot. In figure 5.8, the violin plot shows the distribution of answers to each category's questions concerning the 5-point Likert scale and a kernel density estimation of the distribution.

Concerning the perceptions of the system's hedonic value (figure 5.8(a)), the violin plots reveal that most volunteers enjoyed exercising aided by the VC, felt motivated and interested while performing the exercises with the system, and found the interaction established with them - through the coach speech, provided feedback, recommendations on posture correction, and encouragement - extremely pleasurable. One volunteer revealed a more neutral opinion concerning the system interaction features, and another felt a bit bored while exercising. From 5.8(b) we infer that most volunteers find the system strongly useful to help patients improve their upper extremity movement quality. They find it valuable to help the patients exercise when they cannot have therapists' supervision and exercise independently. Study participants who answered Q4 believe that this system can help overcome some of the obstacles found in rehabilitation clinics and hospital facilities. Only one subject expressed a less agreeing opinion about the system's utility in aiding patients exercising alone. Regarding user's intention to use (figure 5.8(c)), all subjects revealed that they would confidently keep using the system to help them exercise in the case of being under a rehabilitation process. Figure 5.8(d), shows that volunteers' perception of the system's performance is that it performed really well and fulfills its purpose of monitoring the exercise and detecting compensation behaviors. Generally, volunteers felt that they could trust the system evaluation of their motor performance.

To investigate the raised hypotheses in section 4.3 on the perceptions about the system of users with different experiences regarding stroke (H1), how user's perceptions differ with age (H2), and their experience in the training session with the VC (H3 and H4), we perform statistical analysis.

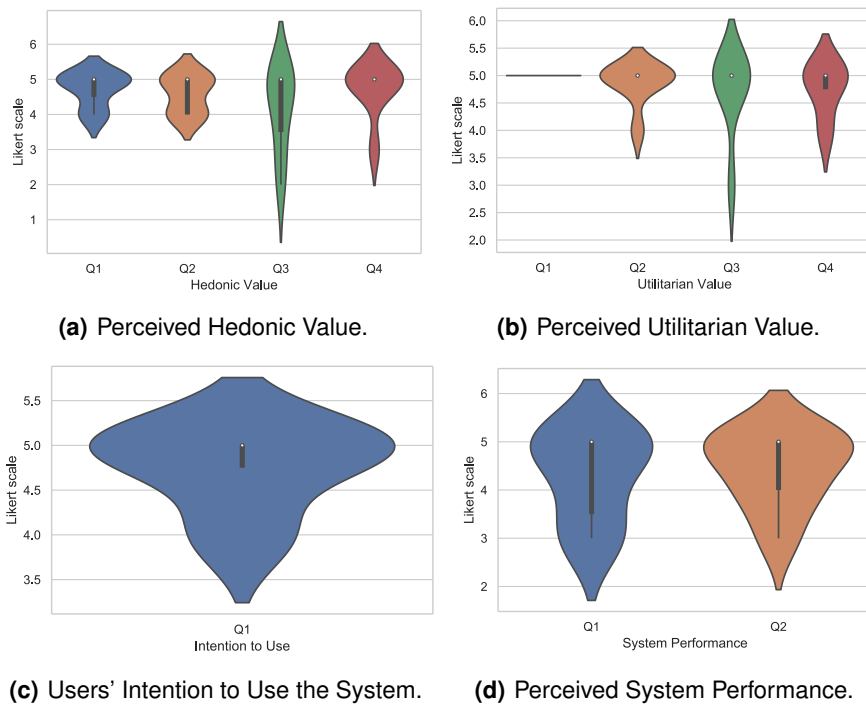


Figure 5.8: Perceptions of the Virtual Coach.

5.2.1.A Stroke Survivor vs. Other Volunteers

To investigate H1, since we only have one volunteer who is also a stroke survivor, we show the mean perceptions for each evaluation dimension of the stroke survivor volunteer and the other subjects. Figure 5.9 exposes that stroke survivor was more critical with the system, giving a lower mean score for the utilitarian ($mean = 4$) and system performance ($mean = 3$) and showing a less intention to keep using the system ($mean = 4$). However, this difference is tolerably small. Other volunteers present a mean score for system utilitarian value of 5.0 ± 0.0 , for intention to use of 5.0 ± 0.0 , and for system performance of 4.5833 ± 0.5845 . Concerning hedonic value perception, the stroke survivor and the remaining volunteers equally enjoyed the training and interaction with the system.

5.2.1.B The Age Factor

Hypothesis H2 raises the assumption that older adults have a different perception of the system utilitarian value from younger adults. We believe older adults find the system more valuable, having a more positive perception about VC utilitarian value. Mean scores for this evaluation category between both groups should reveal this assumption. Before we investigate this hypothesis, we look at the mean scores for each age group (figure 5.10(a)) and observe how these values vary with age among the different dimensions. Older adults, with age over 55 years old, find the system more enjoyable and consider that the system performs very well and fulfills its purpose. They find the system more relevant and useful ($mean = 5.0 \pm 0.0$), and

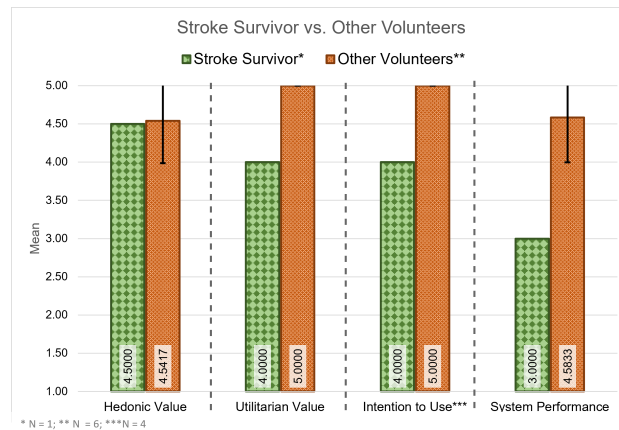


Figure 5.9: Stroke Survivor and Other Volunteers Perceptions of the Four Dimensions.

express a clearer intention to keep using the system ($mean = 5.0 \pm 0.0$).

To support the idea of different perceptions of utilitarian value between older and younger adults and investigate H2, we perform an independent-sample t-test and determine if the mean scores between both groups are statistically significantly different. We consider older adults the group of volunteers over 54 years old (four subjects), and younger adults the remaining volunteers. More specifically, the volunteers with 25-34 years old (three subjects).

Before running an independent-sample t-test, it is important to note that both groups have a small and unequal number of subjects. This condition can lead to a not trustworthy $p - value$. First, we perform a Levene's test to verify the equality of variances of both groups. With this test we get Levene's test statistic $F = 22.857$ and $p - value = 0.005$. Since F reveals a huge value and the $p - value$ is really small, we assume that both groups' variance is unequal. This fact influences how the test statistic and the degrees of freedom are calculated in the independent-sample t-test.

With the independent-sample t-test we verify that younger adults have a mean perception for the utilitarian value of 4.6667 ± 0.5774 and older adults of 5.0 ± 0.0 (figure 5.10(b)). The groups have a mean score difference of 0.3333. The test statistic of the null hypothesis - younger and older adults have equal mean score for the utilitarian value - is $t(2) = 1$ and $p - value = 0.423$. This means that we cannot reject the null hypothesis and accept H2, with any significance level under 0.423. This test also gives us a 95% confidence interval $[-1.1009, 1.7676]$, which contains 0, meaning that these results are not very significant. Consequently, we conclude that the difference in the utilitarian value between both groups is not significant.

5.2.1.C Volunteers Experience with the Virtual Coach

Now, we investigate hypothesis H3 - hedonic value perceptions are affected by the perceived system performance - and H4 - hedonic and utilitarian value perception affects users' intention to use. To investigate these hypotheses, we correlate the dimensions and examine how they influence each other.

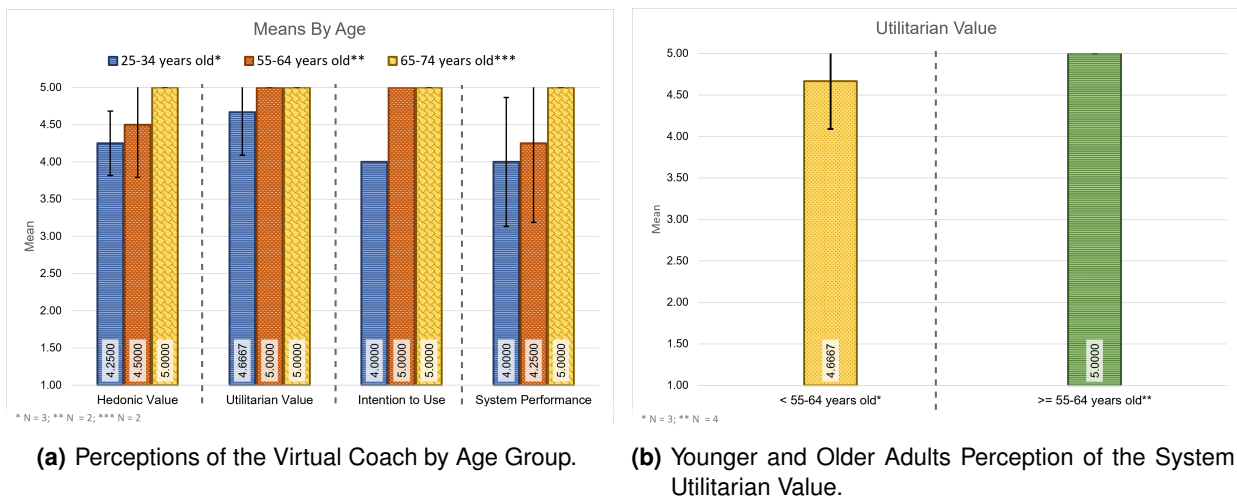


Figure 5.10: Perceptions among younger and older adults.

Table 5.4 presents a summary of descriptive statistics and the Pearson correlation between hedonic and utilitarian values, users' intention to use, and system performance. The Pearson correlation coefficient (ρ) measures the degree of correlation of the distinct dimensions. A $\rho = 1$ means a perfect positive correlation between two dimensions, i.e., if one increases, the other increases. A $\rho = 0$ indicates the nonexistence of correlation between two variables, i.e., they do not linearly depend on each other. Moreover, a $\rho = -1$, two variables have a perfect negative correlation, meaning that if one increases the other decreases.

Table 5.4 shows a correlation between hedonic value and system performance with a Pearson correlation coefficient of $\rho = 0.525$, revealing that these dimensions are linearly correlated. If the mean value of the perceived system performance increases, it positively influences the perceived hedonic value, supporting H3. Intention to use has a perfect linear correlation with all the other dimensions ($\rho = 1$). If the mean perceived hedonic and utilitarian values increase, users' intention to use the system grows, supporting H4.

	H	U	IU	SP
Hedonic Value (H)	1	0.031	1.000	0.525
Utilitarian Value (U)	0.031	1	1.000	0.746
Intention to Use (IU)	1.000	1.000	1	1.000
System Performance (SP)	0.525	0.746	1.000	1
<i>Mean</i>	4.54	4.86	4.75	4.36
<i>SD</i>	0.51	0.38	0.50	0.80

Table 5.4: Descriptive Statistics and Pearson Correlation.

5.2.2 Qualitative Results and Discussion

Before the empirical experiment, we conducted some tests with the system respecting the established condition - laptop with the webcam 2.5 m away from the subject - and to ensure it behaved as expected. During these previous tests, while simulating motor compensation patterns during exercise performance, the system behaved as predicted and detected compensation as expected.

The qualitative information gathered in the questionnaire confirms volunteers' quantitative answers to each question with the 5-point Likert scale. All subjects expressed how much they enjoyed the therapy session with the system. They appreciated how the system monitors their movements and its reactions, mainly its suggestions on their compensation behaviors. Also, they expressed that the displayed visual markers and coach instructions, make the therapy with the VC very pleasant. Additionally, they found the compensation detection pretty precise and the coach recommendations for posture correction very relevant.

Volunteers find the system pretty appropriate and useful to help patients perform rehabilitation exercises, principally when they cannot have professional supervision and assistance. Some think that this kind of VC can be valuable to overcome problems found in hospital facilities, essentially the lack of therapists to answer the needs and demands of an extremely high number of patients. Some participants expressed that this system would help them keep practicing and exercising their limbs with continuous use.

Concerning system performance, the subjects confirmed accurate compensation detection and its reliability and synchronized suggestion of a new movement trial. During some experiments, due to space conditions, it was not possible to keep the laptop 2.5 m away from the subject, which influenced the system performance. The system was sometimes unable to detect the body keypoints correctly. Also, the rectangle that delineates subjects' correct positioning revealed to be too small, constraining volunteers positioning since they kept the body very close to the rectangle limits. Faced with this problem, a volunteer suggested a bigger rectangle to give patients more space to move freely without giving the system the idea of incorrect positioning. Also, the subject revealed feeling bored with this issue.

Another situation that made us violate the ideal conditions was older adults' difficulty seeing well what was displayed on the laptop screen. This highlights the need for a bigger screen to display the UI or a camera independent from the laptop, which could be placed at the required distance to the subject.

The system had a slower performance in some sessions due to internet connection conditions, having a slower response to volunteers' movements. Some study participants highlighted the system's slow response to their movements. This aspect was overcome during the laboratory tests since the system was connected to the same network of the server - in which the system performs the image processing necessary to detect motor compensation and the subject's arm positioning - having a faster response.

Concerning the coach exercise instructions, when the coach encourages the participants to reach the target position (table A.1), some participants found misleading it referring to take their "arm" upper for E1 and E2, and not their hand, enabling them to accomplish the movement the system was expecting. Also,

about coach speech, a volunteer expressed that it could be more emotive and expressive.

Other subjects expressed that the VC's interaction with them should be more stimulating and found its instructions quite repetitive. These perceptions meet the VC requirements presented in section 3.1.1. Real patients perform slow movements and need time to execute them. They also need to hear repetitive instructions and encouragement to keep training. Another volunteer referred that this kind of independent exercising only is possible with patients without extremely limiting physical conditions, which is correct since this kind of therapy can only be prescribed to patients already able to execute it.

Since the stroke survivor volunteer has a direct experience of a long rehabilitation process, his opinion about the system and suggestions are very relevant to our study. This volunteer commented that the system should not be too sensitive to detect compensation, giving the patient time and opportunity to perform the necessary movements to accomplish the proposed tasks and correct oneself without being constantly and immediately corrected. This subject finds the system extremely relevant, diminishing therapists intervention, and highlights patients' need to keep practicing to improve their physical abilities. He states that this kind of system is a good tool to stimulate patients' motivation to get better and keep training. Additionally, he supports the significance and accuracy of the compensation detection and posture correction suggestions. However, he affirms some corrections may be difficult to perform autonomously.

As mentioned in section 4.3.2, we used the RB approach for the VC to assess compensation. During the experimental sessions with the volunteers, we simulated the compensation behaviors considered and identified some situations. For example, when a subject tilts his torso, the system sometimes assumes that the subject elevates his shoulder. This assumption is understandable considering the classifier fails to detect trunk tilt and assumes shoulder elevation since the subject's shoulder is actually elevated compared with its initial position. To overcome this issue, we could consider as a priority detect trunk compensation behaviors. Since we consider that shoulder elevation and torso rotation can occur simultaneously, we could first assess trunk rotation and then shoulder elevation. For other trunk compensation behaviors, once detected, we could exclude the possibility of shoulder elevation and not proceed with its assessment. Of course, these assumptions should be confirmed with therapists to ensure our RB approach updates would really improve and not exclude the ability to detect some patterns.

6

Conclusions & Future Work

This chapter summarizes all work components' final results and conclusions concerning the evaluation of the proposed methods. Additionally, we give some suggestions for future work.

This work aims to provide a solution to give answers to the increasing demands concerning stroke rehabilitation therapy. The rise of in-home exercise recommendations leads to the necessity of proper, affordable, and accessible assistive systems to aid autonomous exercising and objective assessment measures to track patients' motor improvements.

In this work, we develop an image-based Virtual Coach (VC) capable of monitoring upper extremity rehabilitation exercises focused on motor compensation reduction. The VC proposes three exercises (E1, E2, and E3), assess users compensation patterns from video frames, and provides proper visual and audio feedback and instructions through a UI. This only requires a simple technical setup composed of a laptop and a webcam. We propose a method and objective measures to assess compensation from 2D positional data. We investigate two Multilabel Classification (MLC) approaches - a Rule-based (RB) and a NN based - to assess various compensation patterns for three exercises and compare them. The RB takes the kinematic variables as input features and the NN approach normalized body keypoints.

In a previous step, we validate the hypothesized kinematic variables to assess compensation from 2D body keypoints. Given variables' behavior during exercise performance, we concluded that most of them behaved as expected, enabling applying the RB method rules.

For exercise E1, the RB method performed better with a $F_1 = 76.69\%$ than the NN and for exercises E2 and E3 the NN approach reveals better results with $F_1 = 72.56\%$ and $F_1 = 79.87\%$. Given the datasets' poor multilabel nature, we conclude that the RB approach handles better a multilabel scenario and the NN a binary one. This conclusion comes from the fact that the E1 dataset has 83.8336% single labeled samples, and E2 and E3 are tendentiously binary, with 91.4032% and 98.1481% single labeled samples, respectively. Label imbalance also had a big influence on the evaluation metrics, mostly when calculated through macro-averaging. Since NN based approach showed evidence of greater performance in general, we believe we could get greater results with more data samples. Thus, we would benefit from a larger dataset with more examples and a balanced representation of each motor pattern.

To evaluate our VC we recruited seven volunteers to exercise with it and gather their perceptions about the system through a questionnaire. In general, the subjects found the system enjoyable with pleasant interaction features. They found the VC really valuable to aid patients to exercise autonomously and compensation assessment very accurate and following the feedback provided. However, improving some details

could lead to better results, such as the system response speed and flexibility regarding patient positioning in the image. Since the VC used the RB method to assess compensation, we noticed some issues during this experiment. In some cases, when users' performed torso compensation, the system detected shoulder elevation incorrectly because the shoulder was actually elevated even though this was not the corresponding compensation behavior.

Given these results, we can affirm that we achieved the project objectives. We were able to assess motor compensation from 2D positional data with pretty good accuracy. Our VC fulfilled its purpose and met all the requirements concerning this kind of system.

6.1 Future Work

With the presented results and all the work developed during this process, we formulated some improvements and future work suggestions.

To continue the investigation of ML assessment methods to detect motor compensation from 2D positional data, we suggest the development of a Recurrent Neural Network based method. Since this kind of network processes sequential data, giving the model the body keypoints in sequential order, it could be interesting to investigate how this approach benefits compensation assessment.

Concerning the assessment methods, our RB approach could be improved, giving priority to trunk displacements over shoulder elevation patterns, to overcome the observed issue of shoulder elevation misdetection when a subject is displacing the torso.

To improve our VC, a more emotive and expressive speech would provide a more engaging therapy experience to patients. The implementation of exercise adaptations, such as different exercise levels, which update given user performance, would make the exercises more challenging, promoting compliance. Also, VC could be adjusted to enable more flexibility regarding users' initial position.

To evaluate the real impact of the VC in real stroke survivors under a rehabilitation process and gather more solid perception about the system, this study could benefit from an empirical experiment of VC long-term use. Also, evaluation of the system from therapists would give a clearer notion of its significance and reliability and determine more improvements.

Bibliography

- [1] B. Semenko, L. Thalman, E. Ewert, R. Delorme, S. Hui, H. Flett, and N. Lavoie, “An Evidence Based Occupational Therapy Toolkit for Assessment and Treatment of the Upper Extremity Post Stroke,” 2015.
- [2] M. Duff, Y. Chen, L. Cheng, S.-m. Liu, P. Blake, S. L. Wolf, and T. Rikakis, “Adaptive Mixed Reality Rehabilitation Improves Quality of Reaching Movements More Than Traditional Reaching Therapy Following Stroke,” *Neurorehabilitation and Neural Repair*, vol. 27, no. 4, pp. 306–315, 2012.
- [3] E. B. Brokaw, E. Eckel, and B. R. Brewer, “Usability evaluation of a kinematics focused Kinect therapy program for individuals with stroke,” *Technology and Health Care*, vol. 23, no. 2, pp. 143–151, 2015.
- [4] T. Rikakis, J. B. Huang, A. Kelliher, K. Kitani, S. L. Wolf, J. Choi, and S. Zilevu, “Semi-automated home-based therapy for the upper extremity of stroke survivors,” *ACM International Conference Proceeding Series*, pp. 249–256, 2018.
- [5] M. A. Murphy, C. Willén, and K. S. Sunnerhagen, “Kinematic variables quantifying upper-extremity performance after stroke during reaching and drinking from a glass,” *Neurorehabilitation and Neural Repair*, vol. 25, no. 1, pp. 71–80, 2011.
- [6] M. H. Lee, D. P. Siewiorek, A. Smailagic, A. Bernadino, and S. Bermúdez I Badia, “Learning to assess the quality of stroke rehabilitation exercises,” in *24th International Conference on Intelligent User Interfaces (IUI'19)*. Association for Computing Machinery, 2019, pp. 218–228.
- [7] E. V. Olesh, S. Yakovenko, and V. Gritsenko, “Automated assessment of upper extremity movement impairment due to stroke,” *PLoS ONE*, vol. 9, no. 8, 2014.
- [8] Z. Cao, G. Hidalgo Martinez, T. Simon, S. Wei, and Y. A. Sheikh. (2019) OpenPose. Accessed 1-Nov-2019. [Online]. Available: <https://github.com/CMU-Perceptual-Computing-Lab/openpose>
- [9] T. Baltrusaitis, A. Zadeh, Y. C. Lim, and L. Morency. (2019) OpenFace 2.2.0: a facial behavior analysis toolkit. Accessed 15-Feb-2020. [Online]. Available: <https://github.com/TadasBaltrusaitis/OpenFace/wiki>

- [10] R. Schulz, *Quality of Life Technology Handbook*. CRC Press, 2012.
- [11] D. P. Siewiorek, A. Smailagic, and A. Dey, "Architecture and applications of virtual coaches," *Proceedings of the IEEE*, vol. 100, no. 8, pp. 2472–2488, 2012.
- [12] S. Federici and M. J. Scherer, *Assistive technology assessment handbook*. CRC Press, 2017.
- [13] World Health Organization (WHO), "WHO Global Disability Action Plan 2014-2021," 2015.
- [14] UN, *World Population Ageing 2019*, 2019.
- [15] Executive Boardroom , WHO Headquarters, "Rehabilitation 2030: a call for action," 2017.
- [16] K. L. Meadmore, E. Hallett, C. Freeman, and A. M. Hughes, "Factors affecting rehabilitation and use of upper limb after stroke: views from healthcare professionals and stroke survivors," *Topics in Stroke Rehabilitation*, vol. 26, no. 2, pp. 94–100, 2019.
- [17] M. F. Levin, J. A. Kleim, and S. L. Wolf, "What do motor "recovery" and "compensating" mean in patients following stroke?" *Neurorehabilitation and Neural Repair*, vol. 23, no. 4, pp. 313–319, 2009.
- [18] M. F. Levin, D. G. Liebermann, Y. Parmet, and S. Berman, "Compensatory Versus Noncompensatory Shoulder Movements Used for Reaching in Stroke," *Neurorehabilitation and Neural Repair*, vol. 30, no. 7, pp. 635–646, 2016.
- [19] T. M. Damush, L. Plue, T. Bakas, A. Schmid, and L. S. Williams, "Barriers and facilitators to exercise among stroke survivors," *Rehabilitation Nursing*, vol. 32, no. 6, 2007.
- [20] M. Rensink, M. Schuurmans, E. Lindeman, and T. Hafsteinsdóttir, "Task-oriented training in rehabilitation after stroke," *Journal of Advanced Nursing*, vol. 65, no. 4, pp. 737–754, 2009.
- [21] S. A. Billinger, R. Arena, J. Bernhardt, J. J. Eng, B. A. Franklin, C. M. Johnson, M. Mackay-Lyons, R. F. Macko, G. E. Mead, E. J. Roth, M. Shaughnessy, and A. Tang, "Physical activity and exercise recommendations for stroke survivors: A statement for healthcare professionals from the American Heart Association/American Stroke Association," *Stroke*, vol. 45, no. 8, pp. 2532–2553, 2014.
- [22] I. Serrada, M. N. McDonnell, and S. L. Hillier, "What is current practice for upper limb rehabilitation in the acute hospital setting following stroke? A systematic review," *NeuroRehabilitation*, vol. 39, no. 3, pp. 431–438, 2016.
- [23] A. Tapus, M. Mataric, and B. Scassellatti, "The Grand Challenges in Socially Assistive Robotics," *IEEE Robotics and Automation Magazine*, vol. 14, no. 1, 2007.
- [24] D. Feil-Seifer and M. J. Matarić, "Defining socially assistive robotics," pp. 465–468, 2005.

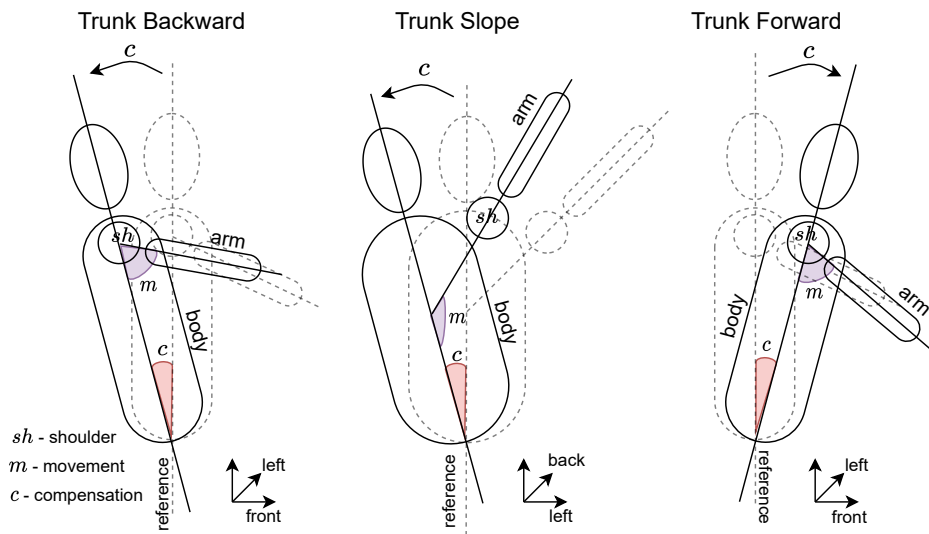
- [25] A. S. Pollock, L. Legg, P. Langhorne, and C. Sellars, "Barriers to achieving evidence-based stroke rehabilitation," *Clinical Rehabilitation*, vol. 14, no. 6, pp. 611–617, 2000.
- [26] A. Ozturk, A. Tartar, B. Ersoz Huseyinsinoglu, and A. H. Ertas, "A clinically feasible kinematic assessment method of upper extremity motor function impairment after stroke," *Measurement: Journal of the International Measurement Confederation*, vol. 80, pp. 207–216, 2016.
- [27] E. J. Schneider, L. Ada, and N. A. Lannin, "Extra upper limb practice after stroke: A feasibility study," *Pilot and Feasibility Studies*, vol. 5, no. 156, pp. 1–7, 2019.
- [28] D. M. Morris, G. Uswatte, J. E. Crago, E. W. Cook, and E. Taub, "The reliability of the wolf motor function test for assessing upper extremity function after stroke," *Archives of Physical Medicine and Rehabilitation*, vol. 82, no. 6, pp. 750–755, 2001.
- [29] D. J. Gladstone, C. J. Danells, and S. E. Black, "The Fugl-Meyer Assessment of Motor Recovery after Stroke: A Critical Review of Its Measurement Properties," *Neurorehabilitation and Neural Repair*, vol. 16, no. 3, pp. 232–240, 2002.
- [30] Canadian Partnership for Stroke Recovery. (2019) Wolf Motor Function Test. Accessed 19-May-2020. [Online]. Available: https://www.strokengine.ca/en/indepth/wmft_indepth/
- [31] ——. (2019) Fugl-Meyer Assessment of Sensorimotor Recovery After Stroke (FMA). Accessed 19-May-2020. [Online]. Available: <https://www.strokengine.ca/en/assess/fma/>
- [32] G. Alankus and C. Kelleher, "Reducing compensatory motions in motion-based video games for stroke rehabilitation," *Human-Computer Interaction*, vol. 30, no. 3-4, pp. 232–262, 2015.
- [33] D. Webster and O. Celik, "Systematic review of Kinect applications in elderly care and stroke rehabilitation," *Journal of NeuroEngineering and Rehabilitation*, vol. 11, no. 108, pp. 1–24, 2014.
- [34] N. Lehrer, Y. Chen, M. Duff, S. L. Wolf, and T. Rikakis, "Exploring the bases for a mixed reality stroke rehabilitation system, Part II: Design of Interactive Feedback for upper limb rehabilitation," *Journal of NeuroEngineering and Rehabilitation*, vol. 8, no. 54, pp. 1–21, 2011.
- [35] S. J. Russell and P. Norvig, *Artificial Intelligence: A Modern Approach*. Pearson Education London, 2010.
- [36] Pallets. (2020, May) Flask Documentation. Accessed 4-Aug.-2020. [Online]. Available: <https://flask.palletsprojects.com/en/1.1.x/>
- [37] Z. Cao, G. Hidalgo Martinez, T. Simon, S. Wei, and Y. A. Sheikh, "Openpose: Realtime multi-person 2d pose estimation using part affinity fields," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2019.

- [38] T. Baltrusaitis, A. Zadeh, Y. C. Lim, and L. Morency, "Openface 2.0: Facial behavior analysis toolkit," in *2018 13th IEEE International Conference on Automatic Face Gesture Recognition (FG 2018)*, 2018, pp. 59–66.
- [39] I. Goodfellow, Y. Bengio, and A. Courville, *Deep Learning*. MIT Press, 2016, <http://www.deeplearningbook.org>.
- [40] F. Herrera, F. Charte, A. J. Rivera, and M. J. Jesus, *Multilabel Classification: Problem Analysis, Metrics and Techniques*. Springer, 2016.
- [41] O. Biran and C. Cotton, "Explanation and Justification in Machine Learning: A Survey," *IJCAI-17 workshop on explainable AI (XAI)*, vol. 8, no. 1, 2017.
- [42] T. Mitchell, *Machine Learning*. Mcgraw-Hill, 1997.
- [43] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay, "Scikit-learn: Machine learning in Python," *Journal of Machine Learning Research*, vol. 12, pp. 2825–2830, 2011.
- [44] M. Čaić, J. Avelino, D. Mahr, G. Odekerken-Schröder, and A. Bernardino, "Robotic Versus Human Coaches for Active Aging: An Automated Social Presence Perspective," *International Journal of Social Robotics*, vol. 12, no. 4, pp. 867–882, 2020.

A

Appendix A

A.1 Compensation Patterns



(a) Trunk Displacement

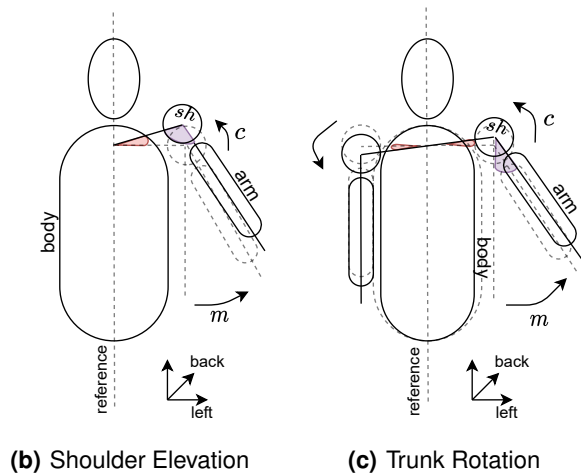


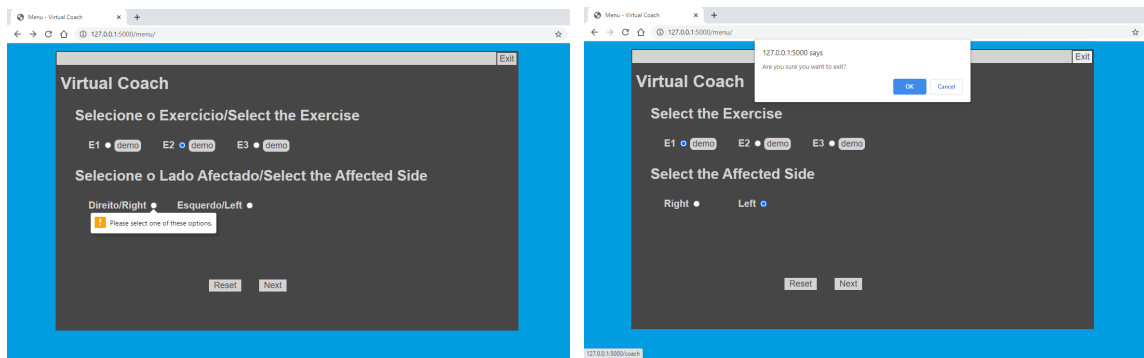
Figure A.1: Examples of Motor Compensation.

A.2 Virtual Coach Speech

Web App	PT	EN
Init web page	> Olá! Bem vindo à sessão de terapia!	> Hello! Welcome to the therapy session!
Menu web page	> Selecione o exercício e o lado afetado!	> Select the exercise and the affected side!
Main web page	> Vamos exercitar o braço um pouco! > Bora lá! Isto vai ser divertido!	> Let's exercise the arm a little! > Let's go! This is going to be fun!
When the user exits Menu and Main	> Foi um prazer ajudá-lo a exercitar! > Espero voltar a vê-lo em breve!	> It was a pleasure to help you exercise! > I hope to see you again soon!
Coach Actions	PT	EN
<i>pos</i> 'position'	> Primeiro! Coloque-se dentro do retângulo. > Posicione-se dentro do retângulo.	> First! Place yourself inside the rectangle. > Position yourself inside the rectangle.
<i>inr</i> 'instruction'	> [E1] Neste exercício, vai simular que está a levar um copo à boca. > [E1] Imagine que está a levar o copo à sua boca. > [E2] Neste exercício, vai simular que está a acender uma luz, ao tocar no interruptor. > [E2] Bora lá, vamos acender a luz! > [E3] Neste exercício, vai mover uma vara para a frente e para trás.	> [E1] In this exercise, you will simulate that you are taking a glass to your mouth. > [E1] Imagine that you are holding the glass in your mouth. > [E2] In this exercise, you will simulate that you are turning on a light by touching the switch. > [E2] Let's go, turn on the light! > [E3] In this exercise, you will move a stick forward and backward.
<i>tar</i> 'target'	> Tente alcançar a posição indicada a verde. > Vamos tentar chegar só até ao ponto verde.	> Try to reach the position indicated in green. > We will try to reach only the green point.
<i>tri</i> 'trials'	> Vamos lá mais uma vez! > Vamos fazer outra vez. > Mais uma vez! > Novamente, realize o movimento.	> Come on one more time! > Let's do it again. > Once again! > Again, perform the move.
<i>enc</i> 'encouragement'	> [E1 & E2] Tente ir mais acima com o braço. Está quase lá! > [E1 & E2] Leve o braço mais acima. > [E3] Leve o braço mais à frente.	> [E1 & E2] Try going higher with your arm. It's almost there! > [E1 & E2] Take the upper arm. > [E3] Take your arm forward.
<i>con</i> 'congrats'	> Está a ir muito bem! Continue! > Excelente! > Está a ficar em forma! > Fantástico!	> It's doing great! Continues! > Excellent! > You are getting in shape! > Fantastic!
<i>trr</i> 'Trunk Rotation'	> Tente não rodar tanto o tronco. > Não mexa o tronco. > Mexa apenas o braço, sem rodar o tronco. > Não rode o tronco.	> Try not to rotate the trunk so much. > Do not move the trunk. > Move the arm only, without rotating the trunk. > Do not rotate the trunk.
<i>she</i> 'Shoulder Elevation'	> Mova apenas o seu braço. > Não eleve o ombro. > Tente sem elevar o ombro. > Não puxe tanto pelo ombro.	> Move your arm only. > Do not raise your shoulder. > Try without raising your shoulder. > Don't pull so hard on the shoulder.
<i>trd</i> 'Trunk Displacement'	> Mantenha o seu tronco direito. > Endireite o tronco. > Tente endireitar o tronco. > Tente manter o tronco direito.	> Keep your torso straight. > Straighten the trunk. > Try to straighten the trunk. > Try to keep the trunk straight.

Table A.1: Virtual Coach Speech.

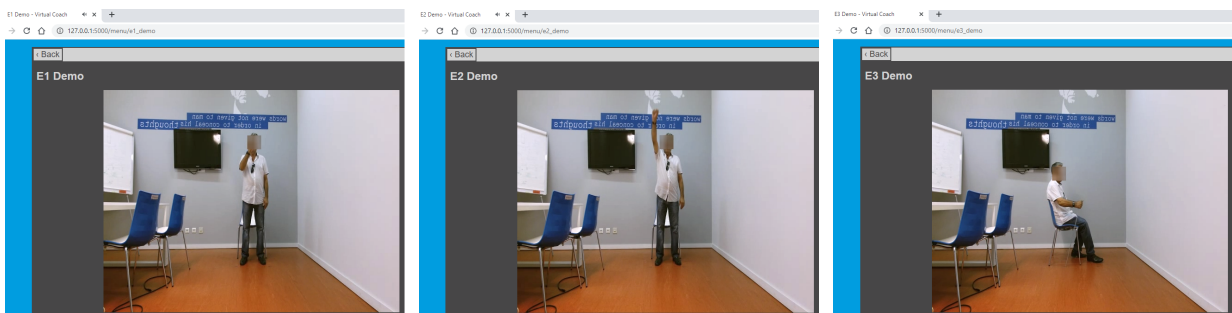
A.3 Web App



(a) Error message indication that required form fields are unfilled.

(b) Exit confirmation.

Figure A.2: Virtual Coach Menu web page and warnings.

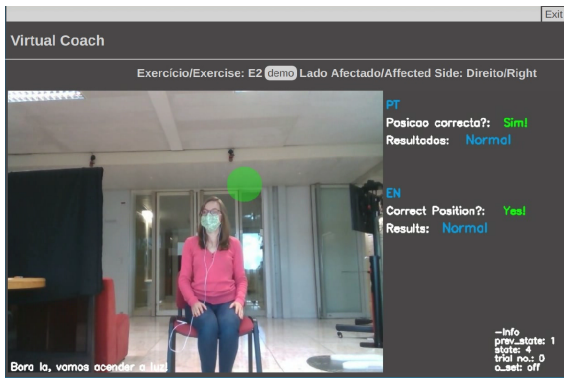


(a) Exercise 1.

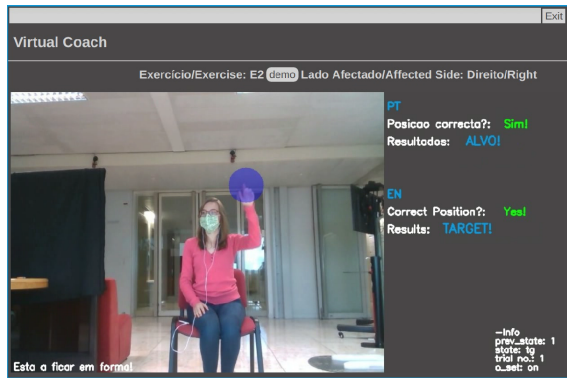
(b) Exercise 2.

(c) Exercise 3.

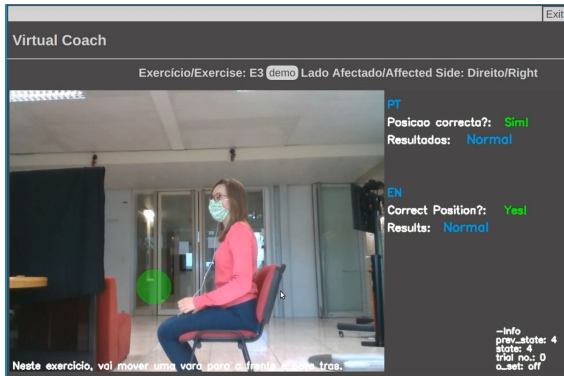
Figure A.3: Virtual Coach Demo web page.



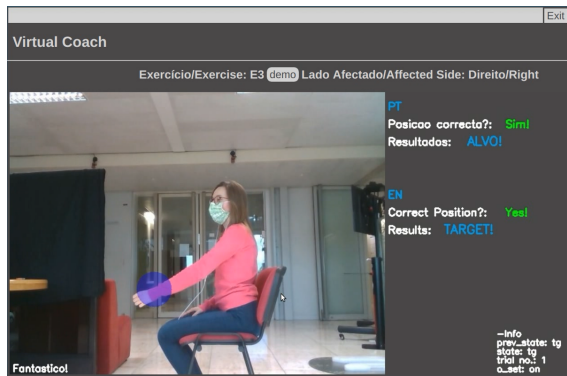
(a) Exercise 2 target position.



(b) Exercise 2 target reached.



(c) Exercise 3 target position.



(d) Exercise 3 target reached.

Figure A.4: Virtual Coach Main web page - Target position.

A.4 The Dataset







Exercise	Initial and Target Positions
E1. <i>'Bring a Cup to the Mouth'</i>	<div style="display: flex; justify-content: space-around;"> <div style="text-align: center;"> <p data-bbox="815 801 943 824">Initial Position</p>  </div> <div style="text-align: center;"> <p data-bbox="1018 801 1145 824">Target Position</p>  </div> </div>
E2. <i>'Switch a Light On'</i>	<div style="display: flex; justify-content: space-around;"> <div style="text-align: center;"> <p data-bbox="815 1090 943 1113">Initial Position</p>  </div> <div style="text-align: center;"> <p data-bbox="1018 1090 1145 1113">Target Position</p>  </div> </div>
E3. <i>'Move a Cane Forward'</i>	<div style="display: flex; justify-content: space-around;"> <div style="text-align: center;"> <p data-bbox="815 1438 943 1460">Initial Position</p>  </div> <div style="text-align: center;"> <p data-bbox="1018 1438 1145 1460">Target Position</p>  </div> </div>

Table A.2: Three upper extremity exercises with respective initial and target arm positions. The images of initial and target positions are displayed with the body skeleton extracted with OpenPose.


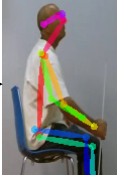







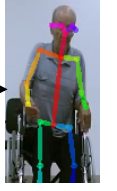


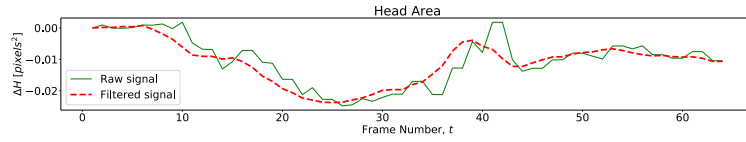
Label	Compensation Pattern	
'0: Trunk Forward'	<p data-bbox="770 297 887 320">Initial Position</p> 	<p data-bbox="986 297 1102 320">Compensation</p> 
'1: Trunk Rotation'	<p data-bbox="770 515 887 537">Initial Position</p> 	<p data-bbox="986 515 1102 537">Compensation</p> 
'2: Shoulder Elevation'	<p data-bbox="770 745 887 768">Initial Position</p> 	<p data-bbox="986 745 1102 768">Compensation</p> 
'3: Other'	<p data-bbox="770 976 887 999">Initial Position</p> 	<p data-bbox="986 976 1102 999">Compensation</p> 
	<p data-bbox="770 1193 887 1216">Initial Position</p> 	<p data-bbox="986 1193 1102 1216">Compensation</p> 
'4: Normal'	<p data-bbox="770 1417 887 1440">Initial Position</p> 	<p data-bbox="986 1417 1102 1440">Compensation</p> 

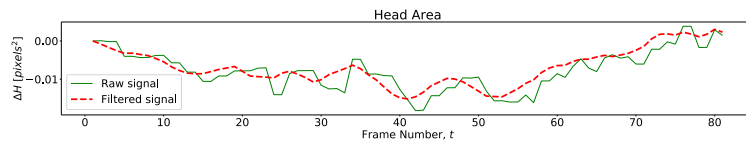
Table A.3: Labels to label the dataset video frames and respective examples of the observed compensation behavior.

A.5 Kinematic Variables Validation

A.5.1 Hypotheses to Assess Movements in Depth

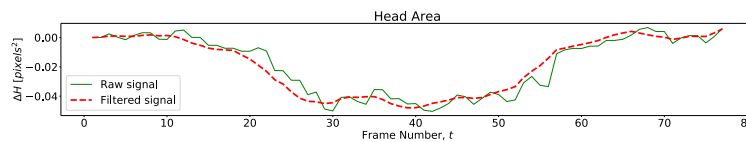


(a) E1

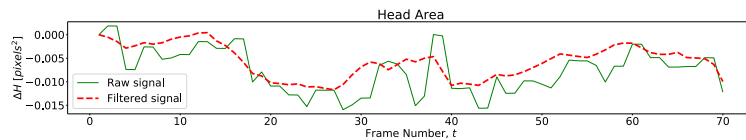


(b) E2

Figure A.5: Head area over time of a **Trunk Forward** simulation in similar conditions of the dataset.



(a) E1



(b) E2

Figure A.6: Head area over time of a trunk moving backward (**Other**) simulation in similar conditions of the dataset.

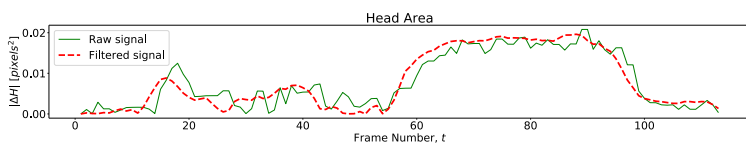


Figure A.7: Head area over time of a trunk tilt (**Other**) simulation in similar conditions of the dataset for E3.

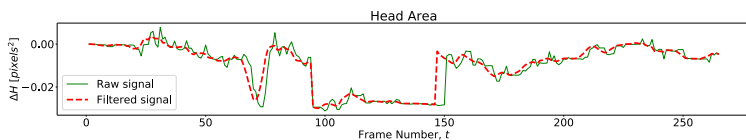
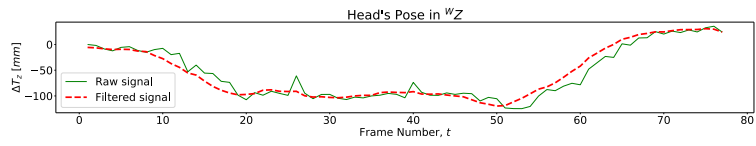


Figure A.8: Head area over time, revealing trunk moving backward (**Other**) observed in the dataset for E1.

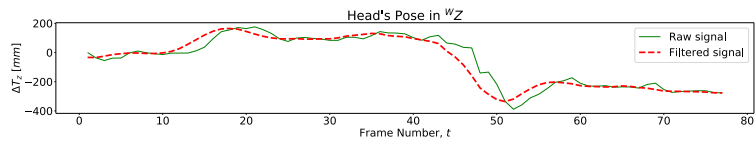


(a) E1

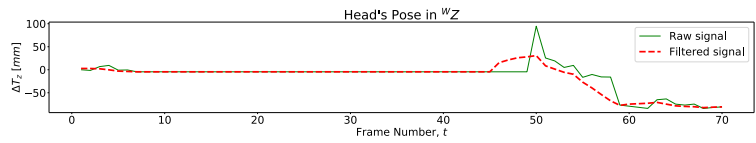


(b) E2

Figure A.9: Head translation over time in ${}^W Z$ of a **Trunk Forward** simulation. Video with a resolution of 640×480 pixels.



(a) E1



(b) E2

Figure A.10: Head translation over time in ${}^W Z$ of a trunk moving backward (**Other**) simulation. Video with a resolution of 640×480 pixels.

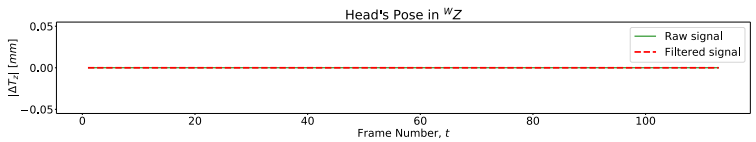


Figure A.11: Head translation over time in ${}^W Z$ of a trunk to tilt (**Other**) simulation. Video with a resolution of 640×480 pixels (E3).

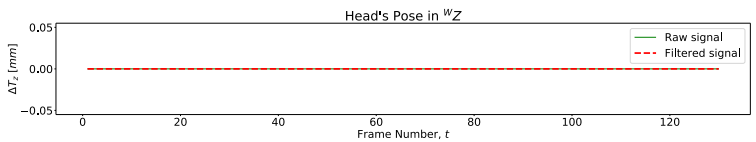
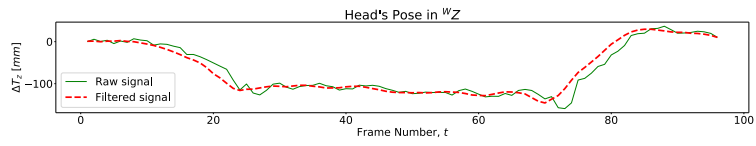
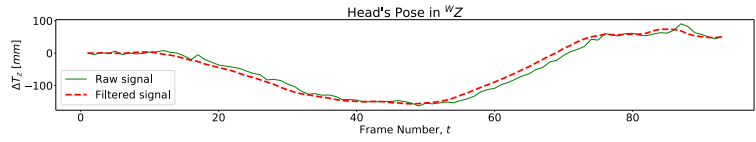


Figure A.12: Head translation over time in ${}^W Z$, revealing trunk moving backward (**Other**) observed in the dataset for E1.

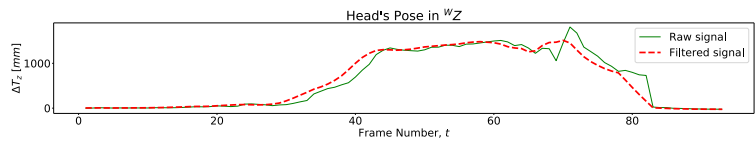


(a) E1



(b) E2

Figure A.13: Head translation over time in ${}^W Z$ of a **Trunk Forward** simulation. Video with a resolution of 1920×1080 pixels.



(a) E1



(b) E2

Figure A.14: Head translation over time in ${}^W Z$ of a trunk moving backward (**Other**) simulation. Video with a resolution of 1920×1080 pixels.

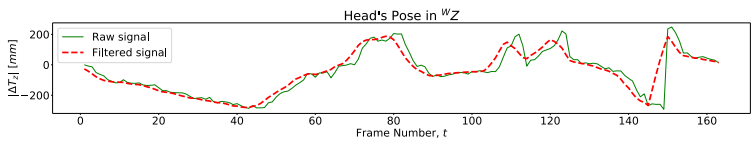


Figure A.15: Head translation over time in ${}^W Z$ of a trunk to tilt (**Other**) for E3. Video with a resolution of 1920×1080 pixels.

A.5.2 Kinematic Variables to Assess Motion Patterns Observed in the Dataset

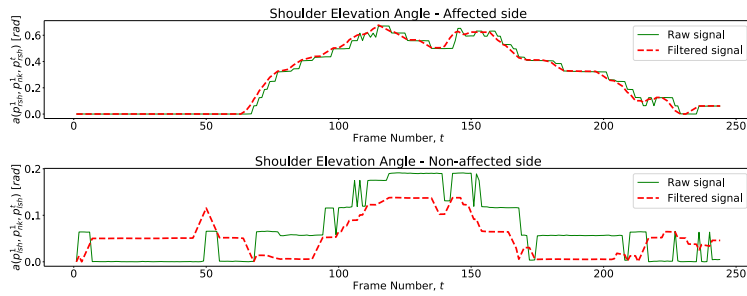


Figure A.16: Patient shoulders' elevation angles over time describing **Trunk Rotation** for E2.

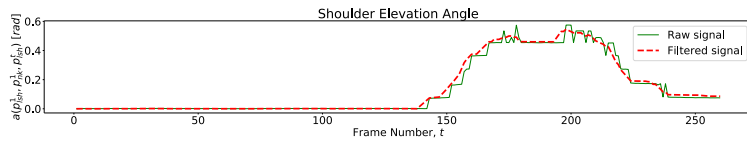


Figure A.17: Patient affected shoulder elevation angle revealing **Shoulder Elevation** for E1.

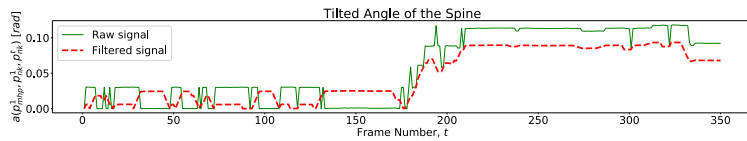


Figure A.18: Patient tilted angle of the torso describing a trunk tilt (**Other**) for E1.

A.5.3 Kinematic Variables to Assess Motion Patterns Not Observed in the Dataset

Here we investigate the patterns not detected in the dataset. As mentioned, we recorded new videos to assess torso moving backward and torso rotation patterns for E3.

Figure A.19 illustrates the variation of the tilted angle of the torso and the displacement of the *Neck* joint regarding its initial position. Similarly to what is observed in figure 4.10 for the trunk moving forward, there is an acute inclination of the trunk but a negative displacement of the *Neck* joint since it does not follow the B_X direction. In figure A.20 we show the two variables used to assess **Trunk Rotation**, one for a patient in an oblique position, and other for a patient in a perpendicular position. Both hypotheses display a clear change in the variables over time, proving that both can accurately assess trunk rotation while reaching the target position.

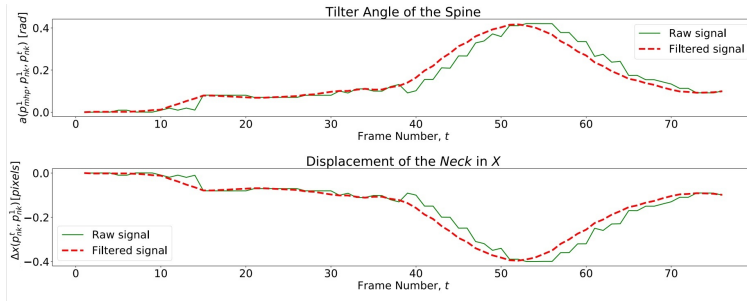
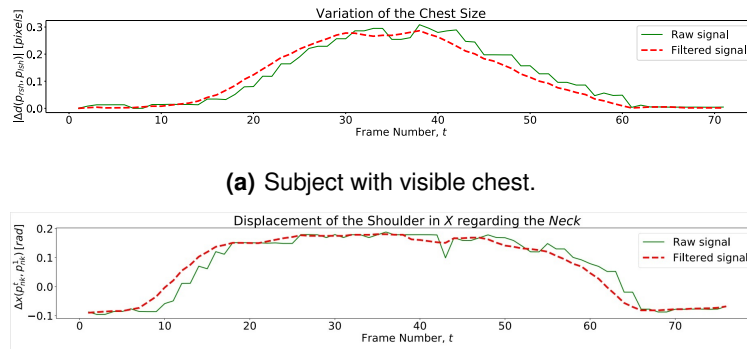


Figure A.19: Tilted angle of the spine and *Neck* displacement over time acquired from a simulation of trunk moving backward (**Other**) in E3.



(a) Subject with visible chest.

(b) Subject completely aside.

Figure A.20: Length between both shoulders for an oblique position and shoulder displacement for a perpendicular position describing trunk to tilt (**Other**).

A.6 Classification Results

This section presents two examples of classification results from cross-validation that illustrate the NN based ensemble's incapacity to detect some patterns.

One example is patient P06 for E1, for whom the NN based classifier reveals a low classification performance. When this patient data is used for validation during cross-validation, the classification report in table A.4 supports this low performance. For this patient, the classifier had a performance of about 27% of F_1 score, which is rather low. Additionally, the classifier is unable of detecting **Other** compensation patterns, which is understandable since this patient describes a movement pattern when trying to reach the target position and in the "drinking" phase very particular, which we considered torso moving backward.

A specific case while performing cross-validation for the NN based approach for E1 and E2 is that the pattern **Trunk Rotation** only occurs for one patient. During cross-validation, when we use this subject video trials data for validation, we remain without any examples of **Trunk Rotation** in the training set, which makes it impossible for the classifier to predict this compensation pattern. This particular case is confirmed by the classification report for E1, when we use P11 for validation, in table A.5. When we include this patient in the training set, we can never verify if the ML approach can predict this compensation pattern. This case

is clear evidence that we would benefit from more data to achieve more accurate predictions, with more examples of the different motor patterns, mainly of the less frequent.

	<i>Precision</i>	<i>Recall</i>	F_1	Support
Trunk Rotation	0.00	0.00	0.00	0
Shoulder Elevation	0.57	0.13	0.22	1780
Other	0.00	0.00	0.00	1649
Normal	0.31	0.80	0.44	849
micro avg	0.35	0.22	0.27	4278
macro avg	0.22	0.23	0.17	4278

Table A.4: Classification Report for P06 E1 data as validation set for the NN Based approach.

	<i>Precision</i>	<i>Recall</i>	F_1	Support
Trunk Rotation	0.00	0.00	0.00	1059
Shoulder Elevation	1.00	0.33	0.50	1501
Other	0.00	0.00	0.00	0
Normal	0.50	1.00	0.67	1004
micro avg	0.60	0.42	0.50	3564
macro avg	0.38	0.33	0.29	3564

Table A.5: P11 classification report NN Based approach for E1.

A.7 Questionnaire

<p>Hedonic Value</p> <p>Q1: Overall, did you like the therapy session with the system? I hated it (1) ... I enjoyed it (5)</p> <p>Q2: Did the system make you feel motivated to perform the exercise? I did not feel motivated at all (1) ... I felt extremely motivated (5)</p> <p>Q3: Did you ever feel bored during the exercise? I felt bored (1) ... I felt interested (5)</p> <p>Q4: Did you find the system's interactive features (feedback, suggestions, and encouragement) pleasant/interesting? I found it unpleasurable (1) ... I found it pleasurable (5)</p>
<p>Utilitarian Value</p> <p>Q1: Do you find the system and this type of interaction harmful or useful for health? I find it harmful (1) ... I find it very useful (5)</p> <p>Q2: Do you think that these exercises with the system's support can help patients improve their physical condition? I do not think it helps at all (1) ... I think it can help a lot (5)</p> <p>Q3: Do you find this system useful when you have to exercise alone without the supervision of professionals? I do not find it useful at all (1) ... (5) I find it useful a lot</p> <p>Q4: In your opinion, can this system help to overcome or reduce the difficulties existing in a rehabilitation center in responding to your needs or the population in recovery in general? I think it can not help at all (1) ... I think it can help a lot (5)</p>
<p>Intention to Use</p> <p>Q1: If you could have/have had this system at the rehabilitation center or home, would you use it to exercise? I would always use it again (1) ... I would always use it (5)</p>
<p>System Performance</p> <p>Q1: Did you find the system was competent to perform its task? I found it not competent at all (1) ... I found it very competent (5)</p> <p>Q1: Do you find the information given by the system about its performance and suggestions reliable? I find it not reliable at all (1) ... I find it very reliable (5)</p>

Table A.6: Questionnaire to Volunteers to evaluate the Virtual Coach.

