POLITECNICO DI TORINO

Department of Control and Computer Engineering
Master's Degree in Data Science and Engineering

Machine Learning-Driven Vision System for Detecting and Evaluating Compensatory Movements in Robot-Based Stroke Rehabilitation

Supervisors
Prof. Danilo Demarchi
Prof. Paolo Bonato
Dr. Giulia Corniani

Candidate
Federica Lupo

October 2024

"Strength does not mean you are not tired,
it does not mean you are not in pain,
it does not mean you are not exhausted,
it just means that you have the strength to keep going"

Abstract

A stroke, also known as a cerebrovascular accident (CVA), occurs when blood flow to a part of the brain is interrupted or when a blood vessel in the brain bursts. Stroke remains the second-leading cause of death and the third-leading cause of death and disability combined in the world. Only 12% of stroke survivors achieve complete upper limb functional recovery. Robotic rehabilitation has become a pivotal tool in improving the recovery process for stroke survivors, offering targeted support to enhance motor function. However, many patients develop compensatory movements—unintended patterns that allow them to work around mobility limitations but reinforce incorrect motor behaviors. This thesis aims to develop a machine learning framework to detect and evaluate compensatory movements in stroke patients using 3D skeletal data extracted from video frames. By identifying and correcting compensatory movements, our approach promotes more personalized and effective rehabilitation, potentially easing the burden on therapists and healthcare facilities with a scalable solution. A previous study was conducted using 2D reconstruction through OpenPose, in this work we advanced the analysis by exploiting the third dimension. We collected data from 22 subjects, including 8 therapists and 14 non-specialized subjects. Therapists were tasked with simulating compensatory movements, similar to those exhibited by patients, while interacting with a robotic end-effector arm, which was used to assist in performing the movements. Normative data were gathered by considering all the subjects. By extracting biomechanically relevant features from this data, and selecting them, we developed hierarchical models that initially identify compensatory movements and subsequently classify them into specific categories. Three scenarios were considered: (1) differentiation between Normative and Compensatory Movements, (2) Single Compensatory Movements vs Multiple Compensatory Movements, and (3) distinction between seven Single Compensatory Movements (e.g. Trunk Extension, Trunk Flexion, Trunk Rotation, Shoulder Elevation). As first approach, we used Random Forest with K Fold Cross-Validation and Leave One Subject Out (LOSO) methods to evaluate the models' performance. Our goal was to develop robust models that could accurately detect and classify compensatory movements, for this reason we conclude our study by considering a further approach with the introduction of an uncertain class. Our models delivered promising results. Considering the F1 score metric for the LOSO per target study, we reached: (1) a value of 0.83 for the classification of Normative vs. Compensatory Movements, (2) a score of 0.74 for the classification of Single vs. Multiple Compensatory Movements, and (3) a value of

0.79 for the classification of Single Compensatory Movements. This thesis presents an innovative approach to automatically detect compensatory movements using 3D skeletal reconstruction and machine learning algorithms. It provides a framework for improving compensatory movement detection in rehabilitation scenarios and could be extended for real-time analysis to support automated therapeutic assistance.

Acknowledgements

I would like to extend my deepest thanks to Professor Danilo Demarchi for the incredible, life-changing thesis opportunity, to Professor Paolo Bonato for hosting me at the Motion Analysis Lab, to Dr. Giulia Corniani for being an inspiration, for her encouragement and support, and for believing in my abilities, and to conclude to Benito, thanks for your engaging curiosity, for always challenging yourself, and for the example you set.

Finally, thank you to all the clinicians and the people who helped me with the data collection, with whom it was a pleasure to spend hours and hours collecting videos and performing compensatory movements.

The nine months dedicated to this thesis have taught me so much—thank you for fueling my curiosity and pushing me to always seek the "why" behind things.

With this achievement, an important chapter that began with my Bachelor's degree comes to an end. During these years at Politecnico di Torino, I have learned a lot, faced many challenges, and shared moments, doubts, and fears with many people.

To *you*, whose example always pushed me to do better, even when I doubted myself and I was tired, I learned that curiosity and passion matter more than results and performance. Your encouragement to become the best version of myself has been invaluable.

To *you*, who supported me despite my rough edges, sometimes difficult to understand. To you who, despite the distance, have always been a constant presence, who patiently listened and stood by me, no matter how long the voice notes or messages.

To *you*, who encouraged me not to take myself too seriously, to smile at life, and let our inner child always have fun. Thank *you* for teaching me lightness, for reminding me to stop and appreciate everything along the way.

To my extended family in Sicily, for the serenity they bring, and to my aunts and cousins spread across Switzerland (I would love to thank everyone by name, but that would take the focus away from the thesis! :))

To two strong women who hold a special place in my heart, Grandma Maria and Grandma Michelina, who saw the beginning of this journey but will not be here for its conclusion—you were my breath of fresh air during stressful times. A part of you will always be with me.

To my Aunt Laura, who was part of my endless study sessions, the laughs, the Nutella snacks, and the words of encouragement when I faltered. Thank you for always welcoming me with open arms and, of course, for the baked pasta and endless roasted potatoes.

To my Aunt Patrizia, thank you for your example of love, for the conversations that make me question who I am, and for the laughs—it's always a pleasure to come back home.

Thanks to my Dad Piero and my Mom Cristi, for always indulging, with varying degrees of lightheartedness, my hunger for knowledge and new experiences. You have given me incredible opportunities that have taught me so much. I'm done for now :)

Thanks to Carla, always ready to support me, even if she doesn't appreciate long voice notes—even the short two-minute ones. Thank you for being such an example to me.

Thanks to my family for sharing both the hardships and the joys with me.

Thanks to the inspiring people I met along the way. All the experiences that enriched me as a person, made me marvel at life, and always pushed me further.

Thanks to the support from my friends in Alessandria, and to Chiara and Elena, for a friendship that has adapted over time. Thank you for always being there, unconditionally.

Thanks to the friendships that have withstood the test of distance, especially Antonella, Maria Elisa, Ilenia, Giacinto, and Salvatore, for every word of encouragement, for your patience in listening to my worries, for indulging my passion for outings, and for the carefree evenings we spent together.

Thanks to the friendships I made outside of my home country, in Germany and the USA, which I hope to rekindle throughout this journey.

Thanks to the people I met in Stuttgart, for being an inspiration, for their encouragement, for the revelations, the meaningful talks, the late nights, and the playful moments. It was a pleasure share a part of this journey with you.

Thanks to my Boston crew for the fun, the hard work, and the countless parties. To my travel companions, Marta, Ste, Ale, Mattia, and Daniel—thank you for the laughs and for rekindling my ability to be a child and always be amazed. I will never forget the moments we shared.

Thanks to all the challenges I faced, even when I doubted my ability to succeed. I didn't always respond in the best ways, but I learned something through each hit. This is just a small step toward becoming more patient with myself—learning that it's okay to make mistakes and not feel capable at first, but remembering that with perseverance and determination, you can continue to grow. That failure will never define a person; it's just a redirection, and even in difficult moments, there's

always something to learn. That we can always change. That sometimes, things shouldn't be overthought, and if we're not where we think we should be, maybe there's a reason—we might have something else to learn before we get there. That multiple parts of ourselves can coexist, and we don't need to identify with just one thing. To do so would be to limit ourselves. Live life to the fullest.

And remember: "The only way out is through".

Ringraziamenti

Vorrei esprimere i miei più sentiti ringraziamenti al Professor Danilo Demarchi per l'incredibile opportunità di tesi che ha cambiato la mia vita, al Professor Paolo Bonato per avermi ospitato al Motion Analysis Lab, alla Dottoressa Giulia Corniani per essere un'ispirazione, per il suo incoraggiamento e supporto, e per aver creduto nelle mie capacità. E per concludere, a Benito, grazie per la tua curiosità coinvolgente, per metterti sempre alla prova e per l'esempio che dai.

Infine, grazie a tutti i clinici e le persone che mi hanno aiutato nella raccolta dei dati, con i quali è stato un piacere passare ore e ore a collezionare video e a fare movimenti compensatori.

I nove mesi dedicati a questa tesi mi hanno insegnato moltissimo—grazie per aver alimentato la mia curiosità e per avermi spinto a cercare sempre il "perché" delle cose.

Con questo traguardo si conclude un capitolo importante iniziato con la triennale. In questi anni al Politecnico di Torino ho imparato moltissimo, ho affrontato tante sfide e condiviso momenti, dubbi e paure con diverse persone.

A *te*, il cui esempio mi ha sempre spinto a migliorare, anche quando dubitavo di me stessa e mi sentivo stanca, ho imparato che la curiosità e la passione sono più importanti dei risultati e delle prestazioni. Grazie per il tuo incoraggiamento a essere la miglior versione di me stessa.

A *te*, che hai supportato i miei lati più spigolosi e difficili da comprendere. A *te*, che nonostante la distanza, sei stato/a una presenza costante, sempre pronto/a a farmi sentire ascoltata, nonostante gli audio chilometrici o i lunghi messaggi.

A *te*, che mi hai insegnato a non prendermi troppo sul serio, a sorridere alla vita e a lasciare che il nostro bambino interiore si diverta sempre. Grazie per avermi insegnato la leggerezza e a fermarmi per apprezzare tutto ciò che incontro lungo il cammino.

Alla famiglia allargata della Sicilia, per la serenità che trasmette e per gli zii e cugini sparsi in Svizzera (vorrei ringraziare tutti per nome, ma si perderebbe il focus della tesi! :)).

A due donne forti che abitano un posto speciale nel mio cuore: Nonna Maria e Nonna Michelina, che hanno visto l'inizio di questo percorso ma non saranno qui alla sua conclusione. Siete state una boccata d'aria fresca nei momenti più stressanti, e porterò sempre con me un pezzo di voi.

A mia zia Laura, che ha condiviso con me infinite sessioni di studio, risate, merende e parole di incoraggiamento quando vacillavo. Grazie per la tua accoglienza e per la pasta al forno (e infinite patate al forno!).

A mia zia Patrizia, grazie per essere un esempio d'amore, per le conversazioni che mi spingono sempre a riflettere su chi sono e per le risate. È sempre un piacere tornare a casa.

Grazie a mio papà Piero e a mia mamma Cristi per aver sempre assecondato, con più o meno leggerezza, la mia fame di conoscenza ed esperienze nuove. Mi avete regalato esperienze incredibili che mi hanno insegnato moltissimo. Ho finito, per ora :)

Grazie a Carla, sempre pronta a supportarmi, anche se non apprezza gli audio, nemmeno quelli corti di due minuti. Grazie per essere stata un esempio per me.

Grazie alla mia famiglia per aver condiviso con me sia le difficoltà che le gioie.

Grazie alle persone che mi hanno ispirato lungo il mio cammino. Tutte le esperienze che mi hanno arricchito come persona, insegnandomi a gioire e stupirmi della vita, e a spingermi sempre oltre.

Grazie al supporto degli amici di Alessandria, e a Chiara ed Elena, per un'amicizia che ha saputo adattarsi nel tempo. Grazie per esserci sempre, incondizionatamente.

Grazie alle amicizie che hanno resistito alla distanza, in particolare ad Antonella, Maria Elisa, Ilenia, Giacinto e Salvatore, per ogni parola di incoraggiamento, per la pazienza nell'ascoltare le mie preoccupazioni, per aver condiviso la mia passione per le scampagnate e per le serate spensierate trascorse insieme.

Grazie alle amicizie nate fuori dal mio paese d'origine, in Germania e negli Stati Uniti, che spero di rincontrare lungo questo viaggio.

Grazie a chi ho conosciuto a Stuttgart, per essere stati un'ispirazione, un incoraggiamento, una rivelazione. Grazie per le conversazioni profonde, le nottate e i momenti di leggerezza.

Grazie al gruppo di Boston per la spensieratezza, la dedizione al lavoro e le infinite feste. Grazie ai miei compagni di viaggio: Marta, Ste, Ale, Mattia e Daniel, per le risate e per aver risvegliato la mia capacità di essere bambina e stupirmi sempre. Non dimenticherò mai i momenti passati insieme.

Grazie a tutte le sfide che ho affrontato, anche quando non pensavo di farcela. Non ho sempre reagito nel migliore dei modi, ma col tempo ho imparato qualcosa da ogni ostacolo. Questo è solo un piccolo passo verso l'essere più paziente con me stessa, imparando che sbagliare fa parte del processo e che, anche se non ci si sente all'altezza, con costanza e tenacia si può continuare a crescere. Un fallimento non può mai definire una persona, è solo un reindirizzamento, e anche nei momenti difficili c'è sempre qualcosa da imparare. Possiamo sempre cambiare. E se non siamo dove pensiamo di dover essere, forse c'è un motivo: forse dobbiamo imparare

qualcos'altro prima di arrivare lì. Che diverse parti di noi possono coesistere, e non dobbiamo identificarci solo con una cosa. Limitarci significherebbe non vivere appieno la vita. Ricorda: "The only way out is through".

Table of Contents

List of Tables xiv

List of Figures xv

Acronyms xix

The Motion Analysis Lab 1

1 Introduction 3
- 1.1 Post-Stroke Rehabilitation 5
- 1.2 Detection of Compensatory Movements 6
- 1.3 Automatic Detection of Compensatory Movements 7
- 1.4 Human-Pose Estimation - State-of-the-Art 10
  - 1.4.1 2D Human-Pose Estimation 13
  - 1.4.2 3D Human-Pose Estimation 14
- 1.5 Posture Check - Phase I 14
- 1.6 Aims, Objectives, and Research contributions of the project 16

2 Materials and Methods 19
- 2.1 Preliminary stage Phase II 20
- 2.2 Experimental Setup and Workstation Configuration 22
- 2.3 Data Collection 23
- 2.4 Data Preprocessing 27
- 2.5 Feature Extraction 31
- 2.6 Feature Selection 36
- 2.7 Model Development 37
  - 2.7.1 Random Forest 42
  - 2.7.2 Leave-One-Subject-Out Study - LOSO 42
  - 2.7.3 LOSOxTarget 43
  - 2.7.4 Uncertainty in Model Predictions 44
  - 2.7.5 Model Evaluation 44

3 Results 46
3.1 Normative vs Compensatory Movements . . . 47
3.1.1 Random Forest (RF) . . . 47
3.1.2 LOSO study . . . 48
3.1.3 LOSO x Target movement study . . . 49
3.1.4 LOSO x Target movement study with uncertain class . . . 50
3.2 Single vs Multiple Compensatory Movements . . . 52
3.2.1 Random Forest (RF) . . . 52
3.2.2 LOSO study . . . 53
3.2.3 LOSO x Target study . . . 54
3.2.4 LOSO x Target study with uncertain class . . . 56
3.3 Single Compensatory Movements . . . 58
3.3.1 Random Forest (RF) . . . 59
3.3.2 LOSO study . . . 60
3.3.3 LOSO x Target study . . . 62
3.4 Recap of Results . . . 66

4 Discussion 68
4.1 Preliminary stage Phase II . . . 68
4.2 Limitations and future steps of the study . . . 71

5 Conclusion 73

Bibliography 75

List of Tables

1.1 F1-Scores for different compensatory movements using ML-KNN and ML-DT classifiers. 9

2.1 Single Compensatory Movements Description. 27

3.1 Single Compensatory Movements: F1-score from RF for each class. 60
3.2 Single Compensatory Movements: F1-score from LOSO study for each class. 61
3.3 Single Compensatory Movements: F1-score from LOSOxTarget study for each class. 62

List of Figures

1.1 Taxonomy of HPE methodologies [25]. 11
1.2 An illustration of top down pipeline. (a) Input image, (b) two persons detected by human detector, (c) cropped single person image, (d) single person pose detection result, and (e) multi-person pose detection result [25]. 12
1.3 An illustration of bottom-up pipeline. (a) Input image, (b) keypoints of all the person, and (c) all detected keypoints are connected to form human instance [25]. 13

2.1 Pipeline. 19
2.2 Skeleton of the subject provided by LightBuzz. 21
2.3 Outliers in skeleton reconstruction with Lightbuzz. 21
2.4 Upper Limb Robotic Device and workstation. 22
2.5 Upper Limb Robotic Device Setup. 23
2.6 Illustration of the 11 target movements. (a) Center, (b) Reaching backward, (c) Reaching forward, (d) Reaching level and affected (0 degrees), (e) Reaching up and affected (45 degrees), (f) Reaching up and center (90 degrees), (g) Reaching up and non-affected (135 degrees), (h) Reaching level and non-affected (180 degrees), (i) Reaching down and non-affected (225 degrees), (l) Reaching down and center (270 degrees), (m) Reaching down and affected (315 degrees). 25
2.7 Reference position: RGB image and skeleton reconstruction. 28
2.8 Skeleton reconstruction before and after normalization for center (first row) and 45° targets (second row). 29
2.9 Examples of inaccurate skeleton reconstruction on coronal, transverse, sagittal planes. 30
2.10 Overview of the cleaned dataset (Normative data -Label 0, Compensatory Movements - Label 1). 31
2.11 Example of angle between body segments. 33

2.12 Example of features. 33
2.13 Description of the spine metrics features. 34
2.14 Description of the upper and lower axis metrics features. 34
2.15 Description of the joint coordinates features. 35
2.16 Three proposed scenarios. 38

3.1 Normative vs Compensatory Movements: overview of the results obtained for the three methods. 47
3.2 Normative vs Compensatory Movements: RF feature ranking. . . . 47
3.3 Normative vs Compensatory Movements: RF Confusion Matrix. . . 48
3.4 Normative vs Compensatory Movements: Confusion Matrix for LOSO Classification. 48
3.5 Normative vs Compensatory Movements: LOSOxTarget Confusion Matrix. 49
3.6 Normative vs Compensatory Movements: global Confusion Matrix for LOSOxTarget Classification. 50
3.7 Normative vs Compensatory Movements: Performance Metrics for LOSOxTarget with uncertain class classification. 51
3.8 Normative vs Compensatory Movements: LOSOxTargetxUncertain class confusion matrix per target. 51
3.9 Normative vs Compensatory Movements: LOSOxTargetxUncertain class global confusion matrix. 52
3.10 Single vs Multiple Compensatory Movements: overview of the results obtained for the three methods. 52
3.11 Single vs Multiple Compensatory Movements: RF feature ranking. . 53
3.12 Single vs Multiple Compensatory Movements: RF Confusion Matrix. 53
3.13 Single vs Multiple Compensatory Movements: Confusion Matrix for LOSO Classification. 54
3.14 Single vs Multiple Compensatory Movements: LOSOxTarget Confusion Matrix per target. 55
3.15 Single vs Multiple Compensatory Movements: global Confusion Matrix for LOSOxTarget classification. 55
3.16 Single vs Multiple Compensatory Movements: Performance Metrics for LOSOxTarget with uncertain class classification. 56
3.17 Single vs Multiple Compensatory Movements: LOSOxTargetxUncertain class Confusion Matrix per target. 57
3.18 Single vs Multiple Compensatory Movements: LOSOxTargetxUncertain class global Confusion Matrix. 58
3.19 Distribution of Single Compensatory Movements. 58
3.20 Single Compensatory Movements: overview of the results obtained for the three methods. 59

3.21 Single Compensatory Movements: RF feature ranking. 59
3.22 Single Compensatory Movements: RF Confusion Matrix. 60
3.23 Single Compensatory Movements: Confusion Matrix for LOSO Classification. 61
3.24 Single Compensatory Movements: Confusion Matrix per target in the LOSOxTarget study. 63
3.25 Single Compensatory Movements: global Confusion Matrix for LOSOxTarget Classification. 64
3.26 Single Compensatory Movements: projections in 2D and 3D using t-SNE. 65
3.27 RF accuracy for the three scenarios. 66
3.28 F1-micro score LOSOxTarget approach for the three scenarios. . . . 67

4.1 Neutral position. 69
4.2 OpenCap Pipeline [37]. 70
4.3 Scheme of the final setup. 70
4.4 Some single compensatory movements frames obtained with OpenCap. 71

Acronyms

AVG
Active Video Games

CNN
Convolutional Neural Networks

CP
Cerebral Palsy

CST
Corticospinal Tract

CVA
CerebroVascular Accident

DNN
Deep Neural Network

HPE
Human Pose Estimation

LOSO
Leave-One-Subject-Out

LSTM
Long Short-Term Memory

MAL
Motion Analysis Lab

ML
: Machine Learning

OOB
: Out-Of-Bag

RB
: Rule-Based

RF
: Random Forest

RNNs
: Recurrent Neural Networks

SAR
: Socially Assistive Robot

t-SNE
: t-Distributed Stochastic Neighbor Embedding

The Motion Analysis Lab

I had the honor of developing this project at the Motion Analysis Lab (MAL) of Harvard Medical School, based at Spaulding Rehabilitation Hospital in Boston, Massachusetts. The MAL is a cutting-edge research facility focused on advancing biomechanics and exploring the most effective methods for restoring mobility. Their research incorporates innovative technologies such as wearable sensors and robotics and has a profound impact on the treatment of neurological conditions, including stroke, cerebral palsy, spinal cord injuries, and Parkinson's disease.

During my experience at the MAL, I had the unique opportunity to actively contribute to multiple phases of various projects, gaining valuable hands-on experience in this dynamic field.

Rehab-Pal

The REHAB-PAL (Rehabilitation Engagement at Home with a socially Assistive roBot for Pediatric Adherence) project aims to enhance adherence to home-based physical therapy for children with Cerebral Palsy (CP) through the use of a Socially Assistive Robot (SAR). Cerebral Palsy is a prevalent motor disorder in children, affecting over 500,000 individuals in the U.S.

Traditional therapy relies on extensive repetition and is often limited by time and logistical challenges, making at-home engagement crucial for effective rehabilitation.

Research shows that Active Video Games (AVGs) improve engagement in rehabilitation but lack the interactive and motivational features necessary for full adherence. In contrast, SARs provide non-contact support through social interactions, such as speech and gestures, offering motivation and accountability that can lead to better patient engagement compared to screen-based solutions. SARs have been effective in various rehabilitation scenarios, such as weight loss and post-stroke recovery.

REHAB-PAL introduces a SAR that works in conjunction with an AVG to create a more interactive and engaging therapy environment for children. The project aims to observe the reaction of children comparing three conditions: 1) therapy with an AVG alone, 2) therapy with the SAR, and 3) therapy with a virtual agent. The goal is to assess whether the robot can lead to greater adherence and better quality of therapeutic performance at home.

I initially contributed to this project by comparing 2D and 3D skeleton reconstruction, determining which type of camera was more suitable for accurately tracking patient movements during therapy. This analysis was crucial in selecting the right technology for the preliminary study.

Posture Check - Phase I and Phase II

My thesis is part of the Posture Check project, which focuses on detecting compensatory movements in stroke patients during robot-assisted upper limb rehabilitation therapy. During Phase 1, we utilized 2D skeleton reconstruction and OpenPose framework to analyze video data of patients performing rehabilitation exercises with the assistance of a robotic arm. This allowed for effective classification of compensatory movements. The results of this phase, along with the introduction of Phase 2, were presented in a poster at the AI Cures Conference, co-hosted by Mass General Brigham (MGB) and the Massachusetts Institute of Technology (MIT), in which I had the honor to participate.

Chapter 1
Introduction

A stroke, also known as a cerebrovascular accident (CVA), occurs when blood flow to a part of the brain is interrupted or when a blood vessel in the brain bursts. A stroke can result in permanent brain damage, long-term impairment, or death. Stroke remains the second-leading cause of death and the third-leading cause of death and disability combined in the world, responsible for over 6.55 million deaths annually [1]. Every year, over 12.2 million people suffer from a stroke, with more than 101 million survivors globally. These numbers are expected to double by 2050, making stroke rehabilitation an increasingly pressing priority [1].

One of the most debilitating consequences of stroke is upper limb motor impairment. It is estimated that between 50% and 80% of stroke survivors suffer from upper limb deficits immediately following the event, with around 50% of these patients continuing to experience functional limitations six months later [2]. These impairments significantly hinder a patient's ability to perform essential daily tasks, such as dressing, eating, and personal hygiene, ultimately reducing independence and overall quality of life. Despite advances in rehabilitation techniques, only 12% of stroke survivors achieve complete upper limb functional recovery [3]. The pathophysiology of motor deficits in stroke patients primarily arises from damage to the corticospinal tract (CST), which disrupts neural communication to the affected limbs [4][5]. The middle cerebral artery, which supplies blood to the upper limb motor cortex, is most frequently involved in strokes, leading to impairments such as weakness, paralysis, and loss of dexterity [6]. For many patients, this loss of function persists despite conventional therapy, necessitating innovative and intensified rehabilitation approaches [4]. Contemporary stroke rehabilitation focuses on personalized, high-intensity interventions, such as task-specific training, strength training, and electrical stimulation. It is recommended that stroke patients receive at least three hours of therapy per day, five days a week, to optimize motor recovery

[7]. Emerging therapies, such as robotic rehabilitation and virtual reality, are gaining traction as adjunctive methods to enhance recovery outcomes [3].

Despite the pivotal role of robotic rehabilitation in improving the recovery process, many patients develop compensatory movements, i.e., unintended patterns that allow them to work around mobility limitations but reinforce incorrect motor behaviors [8][9][10]. In the clinic, methods to determine whether a patient performs compensatory movements have traditionally relied on visual observation, facilitated through one-on-one interaction between the therapist and the patient [11]. However, this approach demands a significant amount of time from therapists, who must guide stroke survivors through repetitive exercises and functional activities while providing corrective feedback. Given the increasing number of stroke survivors and the shortage of trained therapists and caregivers capable of assisting individuals with disabilities, society faces a substantial challenge in providing adequate care [11]. Robotic devices, meanwhile, offer the potential to enhance motor re-learning by providing high-dosage, variable-intensity therapy tailored to the individual needs of patients. These devices hold the promise of enabling more efficient rehabilitation by delivering intensive, personalized therapy while reducing the burden on healthcare professionals [12].

In this thesis, the focus is on leveraging machine learning and 3D skeleton reconstruction from video frames to detect and evaluate compensatory movements in stroke survivors during robotic-assisted upper limb rehabilitation. The goal is to offer a scalable solution that can augment current therapeutic practices by promoting a more personalized and effective recovery pathway. The thesis starts with a brief exploration of the fundamental concepts of stroke, rehabilitation, and human-pose estimation and finally presents the contributions of this work in developing machine-learning algorithms for the automatic detection of compensatory movements during robot-assisted upper limb therapy.

In the first chapter, a thorough introduction is provided on the impact of stroke, particularly focusing on upper limb motor impairment. Additionally, it covers skeletal reconstruction techniques using video data, offering a comparison between 2D and 3D methods and their respective advantages.

The second chapter dives into the materials and methods of the presented study. This chapter describes how data was collected, how features were extracted, and the techniques used to develop machine-learning models to detect and classify compensatory movements.

Chapter three then presents the results derived from the analysis of the 3D skeletal data. This section evaluates the performance of the developed machine learning models, assessing their ability to accurately detect compensatory movements.

Finally, chapters four and five bring the thesis to a conclusion by summarizing the key findings and discussing the clinical implications of the work. The chapters also outline potential future directions for this research, highlighting areas where further development could enhance both the detection of compensatory movements and the overall effectiveness of rehabilitation strategies.

1.1 Post-Stroke Rehabilitation

Post-stroke rehabilitation is a complex and multidisciplinary process aimed at recovering motor, cognitive, and sensory functions that have been compromised due to brain lesions. Neural plasticity, the brain's capacity to reorganize and adapt after injury, is crucial for motor recovery following a stroke. However, not all forms of neural adaptation are beneficial. Takeuchi and Izumi [13] introduced the concept of maladaptive plasticity, a phenomenon where certain neural changes post-stroke impede rather than facilitate recovery, resulting in incomplete or abnormal motor function.

One of the primary mechanisms contributing to maladaptive plasticity is the development of compensatory movements [13]. Stroke patients often rely on the unaffected side or use proximal muscles on the affected side to compensate for their deficits. While this helps with daily tasks, these compensatory strategies reinforce abnormal movement patterns and prevent the full recovery of normal motor control in the affected limbs. Another maladaptive mechanism involves ipsilateral motor projections, where increased activity from the same side of the brain as the affected limb attempts to compensate for the damage [14]. This strategy, however, is often insufficient, particularly in restoring fine motor control in distal muscles, leading to suboptimal motor outcomes.

The imbalance between the two hemispheres of the brain, known as interhemispheric inhibition, further exacerbates motor dysfunction [15]. The unaffected hemisphere can inhibit the recovery of the affected hemisphere too much, preventing the relearning of normal motor patterns. This inhibition also contributes to competitive interactions within the affected hemisphere, where different muscle groups, such as those controlling the hand and proximal arm, compete for dominance. This competition, driven by disinhibition, often results in impaired motor coordination [15].

To counteract these maladaptive processes, Takeuchi and Izumi [13] propose targeted rehabilitation strategies. Rehabilitation programs should aim to minimize compensatory movements, encouraging the restoration of normal motor patterns in the affected limbs. Additionally, noninvasive brain stimulation (NIBS) techniques, such as repetitive transcranial magnetic stimulation (rTMS) and transcranial direct current stimulation (tDCS), may help modulate cortical activity to foster more adaptive plasticity. These techniques can either enhance excitability in the affected hemisphere or reduce excessive activity in the unaffected hemisphere, helping to restore interhemispheric balance and improve motor recovery outcomes.

In addition to medical and surgical treatments in the acute phase of stroke, rehabilitation is a key factor in improving patient outcomes. Conventional therapies, including physical therapy (PT), occupational therapy (OT), and speech therapy (ST), target different areas of recovery [16]. PT focuses on improving limb movement, gait, and balance, while OT helps with daily living tasks and cognitive functions such as memory and attention. ST addresses communication challenges like aphasia. Despite these well-established methods, many stroke survivors continue to experience significant disabilities. This is often due to insufficient therapy dosage, lack of patient motivation, and the absence of real-time, objective feedback—factors crucial for driving functional improvements and long-term recovery [16].

Recently, advanced technologies such as robotic rehabilitation have emerged, offering mechanical support to perform repetitive and precise movements, thereby enhancing cortical plasticity and promoting motor recovery [17]. These robotic devices can adapt to the specific needs of each patient and allow for longer and more repetitive exercise sessions compared to manual therapy alone.

1.2 Detection of Compensatory Movements

The detection of compensatory movements is a crucial area of research, particularly in rehabilitation settings where patients, such as stroke survivors, develop compensatory strategies to perform tasks that their impaired limbs can no longer do effectively. These compensations, while useful for immediate functionality, often lead to long-term negative consequences by reinforcing maladaptive movement patterns and limiting motor recovery [13].

In the literature, different approaches to detection have been proposed.

Marker-based Systems

Traditional compensatory movement detection has relied on marker-based motion capture systems like VICON. These systems track reflective markers placed on specific body parts, capturing detailed kinematic data. This data can then be used to identify compensatory movements such as trunk lean, scapular elevation, or abnormal gait patterns in stroke patients. While highly accurate, these systems are expensive, require specialized environments, and are not easily accessible for continuous monitoring outside of clinical settings [18].

Inertial Measurement Units (IMUs) and Accelerometers

More recent methods leverage inertial sensors and accelerometers worn by patients [19]. These sensors detect deviations from normal movement patterns by measuring acceleration, angular velocity, and other parameters. For example, wearable systems have been used to detect compensatory trunk movements during upper limb reaching tasks, achieving high precision in detecting compensations by analyzing standard deviation of acceleration and angular velocity [19]. These systems are portable, relatively low-cost, and can provide real-time feedback, making them suitable for home-based rehabilitation.

In the context of low back pain (LBP) rehabilitation, the system developed by Asaad Sellmann et al. [20] was used to monitor and detect compensatory movements during exercises like Prone-Rocking, Bird-Dog, and Rowing. The goal was to differentiate between correctly performed movements (CPE) and compensatory movements (TCM). Using wearable inertial sensors placed on the back and head, the system successfully identified compensatory movements with high accuracy (up to 98.9%), aiming to enhance autonomous rehabilitation by providing feedback on movement quality.

Robotic Systems

Robotic systems are increasingly used in rehabilitation to both assist with movement and monitor for compensatory behaviors [21]. For example, robots used in upper limb rehabilitation can detect compensatory patterns such as trunk leaning or shoulder elevation through integrated sensors that track force and movement trajectories, as demonstrated in the study by Laut et al. [21]. Real-time feedback from the robot can then guide the patient to correct these compensations, promoting proper movement patterns and improving recovery outcomes.

Machine Learning for Automated Detection

Machine learning (ML) is transforming compensatory movement detection by allowing for more automated, scalable, and precise analyses. Deep learning algorithms like Recurrent Neural Networks (RNNs) and Long Short-Term Memory (LSTM) networks have been used to classify compensatory motions based on data from sensors or vision-based systems [22]. These models learn from labeled datasets of human motion to automatically detect compensatory patterns, offering a powerful tool for monitoring in both clinical and home environments.

1.3 Automatic Detection of Compensatory Movements

The paper *"Towards Personalized Interaction and Corrective Feedback of a Socially Assistive Robot for Post-Stroke Rehabilitation Therap"* presents an interactive approach that integrates machine learning (ML) and rule-based (RB) models. The goal is to automatically assess rehabilitation exercises and provide personalized corrective feedback. The system was evaluated using a dataset of upper-limb rehabilitation exercises from 15 post-stroke patients. The method included:

1. Feature Extraction: joint coordinates were collected from a Kinect sensor, and various kinematic features such as joint angles, speed, and smoothness were computed.

2. Machine Learning Model: Neural Networks (NNs) and other ML algorithms were applied to predict the quality of patient movements.

3. Rule-Based Model: based on knowledge from therapists, rules were used to assess specific aspects of the exercises.

4. Hybrid Model: a combination of ML and RB models, using a weighted average, was used to improve the overall accuracy of the assessment.

5. Ensemble Voting: to improve frame-level assessment, predictions from multiple frames were combined using an ensemble voting method.

The study found that combining ML and RB models significantly improved the system's ability to provide personalized feedback. The average F1-score of the hybrid model increased from 0.7447 to 0.8235 after tuning with the patients' unaffected motions. Additionally, using an ensemble voting method further improved the accuracy of detecting compensatory movements, especially at the frame level.

One limitation noted is that the rule-based model initially performed poorly because the rules were generic and not tuned for individual patients' specific physical conditions. The performance improved when the model was personalized using data from the patient's unaffected side. However, there is still the challenge of fully adapting the system to handle diverse patient conditions and physical limitations dynamically.

Another relevant study in the topic is the work *"Vision- based Automatic Detection of Compensatory Postures of after-Stroke Patients During Upper-extremity Robot-assisted Rehabilitation: A Pilot Study in Reaching Movement"* [23]. The study investigates the feasibility of using vision-based detection systems to automatically detect compensatory postures during upper-limb rehabilitation of stroke patients. The main elements of the methodology include:

1. Participants: ten stroke survivors (aged 59.4 ± 5.44 years) performed scripted reaching movements with robot assistance.

2. Data Collection: two RGB cameras recorded upper-body joint movements using the OpenPose software to track skeletal joint positions in 2D. This data was transformed into 3D coordinates using a reconstruction method.

3. Feature Extraction: relevant joint angles were computed for trunk flexion, trunk rotation, shoulder flexion, and elbow extension. These features were used to detect compensatory postures.

4. Compensatory Postures: forward trunk displacement, trunk rotation, shoulder elevation, insufficient elbow extension were the compensatory postures monitored.

5. Classification Algorithms: two machine learning models were used, Multi-label k-Nearest Neighbor (ML-KNN), and Multi-label Decision Tree (ML-DT).

Both models were trained to classify the compensatory postures based on the extracted features. In table 1.1 an overview of the results obtained.

Movement	**ML-KNN F1-Score**	**ML-DT F1-Score**
Forward Trunk Displacement	0.69	0.68
Trunk Rotation	0.67	0.60
Shoulder Elevation	0.53	0.50
Insufficient Elbow Extension	0.73	0.80

Table 1.1: F1-Scores for different compensatory movements using ML-KNN and ML-DT classifiers.

Some limitations arose:

1. Feature Selection: only certain joint angles were used as features (e.g., trunk and shoulder angles). This limited the ability to detect more complex compensatory movements, as the influence of other upper body joints was not considered.

2. Dataset Size: the dataset was relatively small, with only ten participants. This limited the generalizability of the models.

3. 3D Reconstruction Errors: the use of 3D reconstruction based on 2D camera data introduced some errors, potentially affecting the accuracy of the feature extraction process.

4. Manual Labeling: the compensatory movements were labeled manually by raters, which may have introduced subjective bias and errors in the ground truth data.

5. First-order Strategy: the classifiers did not consider the correlation between multiple compensatory movements occurring simultaneously. High-order classification strategies might improve detection performance.

The study concludes that it is feasible to detect compensatory postures in stroke patients during robot-assisted rehabilitation using RGB cameras and machine learning classifiers, though improvements are needed in feature selection and model strategies to enhance accuracy.

Finally, the last work that is worth mentioning is *"Automatic Detection of Compensation During Robotic Stroke Rehabilitation Therapy"* [24]. The methods involved using a Kinect v2 camera to track the 3D joint positions of stroke patients and healthy participants during rehabilitation exercises. The participants performed a series of movements, and compensatory strategies such as shoulder elevation, trunk rotation, and leaning forward were simulated by the healthy participants and naturally exhibited by the stroke survivors. The researchers employed machine learning models, including Support Vector Machines (SVM) and Recurrent Neural Networks (RNN), to detect compensatory movements. They used Leave-One-Participant-Out Cross-Validation (LOPOCV) to evaluate the classifier performance.

The findings revealed that the SVM model performed well in detecting compensatory movements in healthy participants, particularly for lean-forward compensations, with F1-score of 0.82. However, the performance dropped significantly when applied to stroke survivors, with much lower F1-scores. This indicated that detecting compensatory movements in stroke survivors is more challenging than in healthy individuals.

The limitations of the study included the small dataset size and imbalanced class distribution, particularly between compensatory and non-compensatory movements, which affected the classifier's performance. Another limitation was the use of a single Kinect sensor placed at an oblique angle, which caused partial occlusion of body parts and impacted the accuracy of detecting certain compensations, like trunk rotation. Furthermore, the study's focus on simulated compensations by healthy participants, rather than real compensations from stroke survivors, highlighted the need for a more robust dataset involving real-world scenarios.

1.4 Human-Pose Estimation - State-of-the-Art

Human Pose Estimation (HPE) is a rapidly evolving field that focuses on determining the configuration of a human body in a given image or video by estimating the spatial coordinates of key body joints (keypoints) [25]. Over the years, significant progress has been made in this area, driven largely by advances in deep learning and computer vision techniques.

The task of pose estimation can be classified into 2D and 3D pose estimation, where the former focuses on detecting keypoints in the 2D plane of an image, and the latter aims to estimate the 3D coordinates of the human body joints.

In the early days of HPE, classical methods such as pictorial structures and deformable part models were prevalent [25]. These methods relied heavily on hand-crafted features and optimization techniques to predict the location of body parts. While effective in controlled environments, they often struggled with challenges like occlusion, variations in lighting, and complex or cluttered backgrounds, which limited their robustness in real-world applications.

The introduction of deep learning, particularly Convolutional Neural Networks (CNNs), marked a turning point for HPE. Deep learning-based models, such as DeepPose [26] and OpenPose [27], significantly improved accuracy by learning rich, high-level feature representations directly from image data. These models transformed the field by mapping image pixels to joint coordinates, allowing for more precise and scalable pose estimation even in challenging settings, thus surpassing classical approaches.

We can identify different groups of approaches [25]:

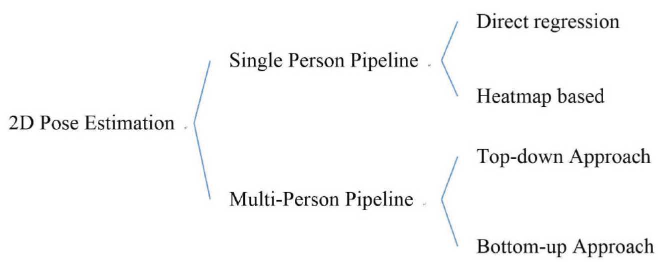

Figure 1.1: Taxonomy of HPE methodologies [25].

- Single-Person Pose Estimation: this method assumes that the position of a person in the image is known, focusing solely on predicting the keypoints. It is a constrained problem where the number of keypoints is predetermined, and the task involves solving a regression problem to locate these points. Single-person approaches can be further categorized into two pipelines [25]:
  - Direct Regression-Based Approaches: these methods directly regress the keypoint coordinates from feature maps. Toshev and Szegedy's [26] cascaded deep neural network (DNN) is a key example, predicting keypoints in a straightforward fashion but facing challenges with mapping complex features.
  - Heatmap-Based Approaches: these approaches generate heatmaps, where each pixel's value reflects the likelihood of a keypoint existing at that location. Techniques such as Convolutional Pose Machines (CPMs) or Stacked Hourglass networks use heatmaps to iteratively refine keypoint predictions [25].

- Multi-Person Pose Estimation: the challenge in multi-person estimation is greater because the number and location of people in the image are unknown. The task requires solving both detection and keypoint localization, which leads to two primary strategies [25]:
  - Top-down approaches: the model first detects the person (bounding box) and then estimates the pose within that region. These methods, like HRNet [28], generally produce more accurate results but require significant computational resources. Figure 1.2 illustrates the pipeline of the approach.
  - Bottom-up approaches [29][30]: on the other hand, directly detect keypoints across the entire image, followed by grouping them into individual poses. Models like OpenPose have demonstrated excellent real-time performance and multi-person pose estimation capabilities but may struggle with fine-grained accuracy in crowded scenes [18]. Figure 1.3 illustrates the pipeline of the approach.
- Transformers in Pose Estimation: more recently, transformer-based models, originally developed for natural language processing, have been adapted for human pose estimation. Transformers allow for global attention mechanisms, enabling the model to consider the relationship between distant body parts, which is critical for complex poses and occlusion handling [31][32].

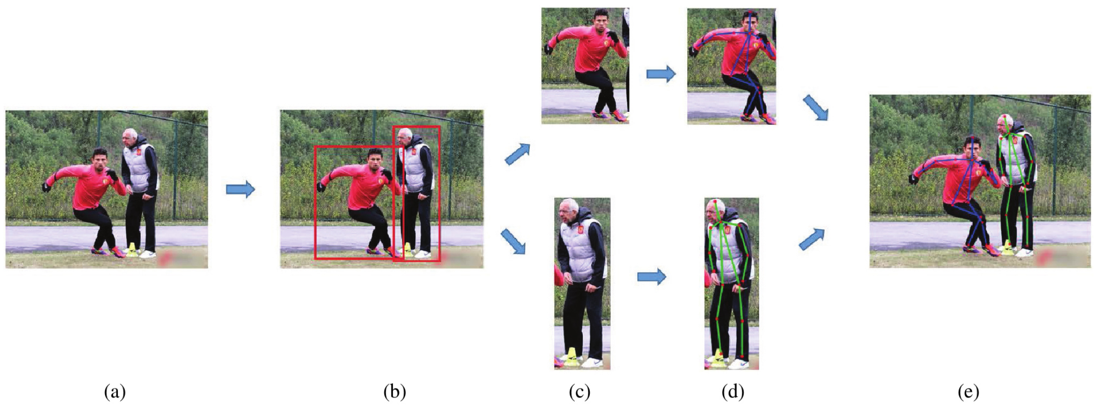

Figure 1.2: An illustration of top down pipeline. (a) Input image, (b) two persons detected by human detector, (c) cropped single person image, (d) single person pose detection result, and (e) multi-person pose detection result [25].

HPE is widely used in fields such as augmented reality, sports analysis, animation, and healthcare. In healthcare, for instance, pose estimation can assist

Figure 1.3: An illustration of bottom-up pipeline. (a) Input image, (b) keypoints of all the person, and (c) all detected keypoints are connected to form human instance [25].

with monitoring rehabilitation exercises or detecting fall risks in elderly patients. In animation and gaming, accurate pose tracking enables more lifelike character movements [25].

Despite remarkable progress, several challenges remain in achieving robust HPE:

- Occlusion and Crowded Scenes: in real-world applications, occlusion, i.e. where body parts are blocked by objects or other people, remains a significant hurdle [33].
- Computational Efficiency: achieving real-time, high-accuracy pose estimation on edge devices like mobile phones is still challenging due to the high computational demands of deep learning models [34].
- Generalization: models trained on specific datasets often struggle to generalize to unseen environments, lighting conditions, or human poses not represented in the training data [28].

Future research is likely to focus on improving efficiency, handling occlusions more effectively, and enhancing the generalization of pose estimation models to work across diverse settings and tasks [25].

1.4.1 2D Human-Pose Estimation

As mentioned, 2D human pose estimation aims to identify the spatial coordinates of human body keypoints (joints like the elbows, knees, and shoulders) from a single image or video. The challenge in this field comes from occlusions, varying lighting conditions, body pose variations, and complex backgrounds. Despite these challenges, significant advances have been made, particularly through the use of deep learning and convolutional neural networks (CNNs).

The bottom-up approach is one common method used in 2D pose estimation. Algorithms like OpenPose [27] and HRNet [28] have set benchmarks in this space. OpenPose, for instance, uses Part Affinity Fields (PAFs) to estimate keypoints for multiple individuals in an image, offering real-time, multi-person pose detection. High-Resolution Networks (HRNet) stand out for their ability to maintain high-resolution representations throughout the network, significantly improving accuracy in keypoint detection even in challenging scenarios. This method has set benchmarks for accuracy in several public datasets like COCO and MPII [28].

1.4.2 3D Human-Pose Estimation

3D human pose estimation extends 2D keypoint detection into the third dimension, reconstructing the spatial positions of keypoints in 3D space. This technique is crucial for applications in virtual reality, animation, and sports analytics, where depth information is essential.

3D pose estimation can be performed using monocular (single camera) or multi-view (multiple cameras) systems. Monocular approaches are inherently more challenging because depth information is lost in a single 2D image, which requires sophisticated algorithms to infer 3D structures. Approaches like DeepPose [26] and HMR (Human Mesh Recovery) [35] use deep learning to infer 3D joint locations directly from images.

However, one of the biggest challenges in monocular 3D pose estimation is resolving depth ambiguity—objects at different distances can appear similarly in 2D [36]. To address this, techniques such as lifting 2D poses to 3D have been explored, where a 2D pose estimate is first obtained, and then a model predicts the 3D joint coordinates based on the 2D inputs. This has been the foundation for methods like Martinez et al. (2017) [36], which introduced a simple yet effective deep neural network for this task.

Multi-view systems, on the other hand, use multiple cameras to capture different angles of a scene and merge these perspectives to generate accurate 3D poses. Although more reliable, these systems are expensive and require complex setups, limiting their use in everyday applications [37][38].

Recent innovations include RGB-D cameras that combine RGB (color) images with depth information to improve 3D accuracy [39]. Additionally, the integration of temporal data from videos has enabled models to consider motion information, leading to smoother and more accurate 3D pose sequences.

1.5 Posture Check - Phase I

Posture Check Phase I was the clinical research study which preceded the study presented in this manuscript [40]. The overall goal of the Phase-I study was to assess the suitability of image processing techniques and 2D skeleton reconstruction, based on Deep Learning, to detect compensatory movements performed by stroke survivors during robot-assisted upper-limb training.

Materials and Methods

- **Data Acquisition:**
  - Data was collected from 50 stroke survivors and 10 physical therapists. The subjects engaged in a series of games designed to elicit various upper limb movements. Therapists simulated compensatory movements of varying severity.
  - Videos of the sessions were analyzed by six clinical researchers who annotated compensatory movements, such as the use of non-affected limbs, shoulder elevation, and trunk rotation. In total, 6585 examples of compensatory strategies were identified.
  - The compensatory movements considered are: Assist, Inclination (affected side and non-affected side), Shoulder Elevation, Trunk Extension, Trunk Flexion, Trunk Rotation (affected side and non-affected side), Feet adjustment.
- **Skeleton Tracking:**
  - OpenPose was used to track the 3D positions of key body joints using the BODY_25 model, which estimates 25 key points (e.g., head, neck, shoulders).
- **Feature Extraction:**
  - A total of 46 features were extracted, including angles between segments (e.g., shoulders and neck) and segment lengths (distances between key joints). Additional features were scaled relative to normative posture.
- **Model Development:**
  - A hierarchical model comprising multiple Random Forest classifiers was created. The model had three stages: (1) classification of Normative vs. Compensatory Movements, (2) distinction between Single and Multiple Compensatory Movements, and (3) classification of specific compensatory strategies (e.g., trunk extension, shoulder elevation).

Results

The model achieved 90% accuracy in distinguishing Compensatory from Normative Movements and in classifying Single Compensatory Movements. The model effectively identified specific compensatory movements, such as trunk rotation and shoulder elevation, though some movements were harder to classify due to their subtlety or lack of training data. Classifying Single vs Multiple Compensatory Movements proved more difficult, with an accuracy of about 77.8%, mainly due to the limitations of 2D human pose estimation and the complexity of compensatory strategies occurring simultaneously.

Limitations

- **Data Limitations:** the dataset lacked a sufficient number of examples involving multiple compensatory movements, which hindered the model's ability to generalize in those scenarios.
- **Technical Challenges:** 2D human pose estimation was used in the pilot study, which struggled to capture movements in 3D space accurately, especially compensations involving the z-axis (depth). The OpenPose model faced issues with occlusion from the robotic arm during the therapy, which caused some inaccuracies in skeletal tracking.

1.6 Aims, Objectives, and Research contributions of the project

The aims, objectives, and research contributions of the project are outlined and discussed.

Aims

1. Develop an automated framework for detecting compensatory movements during robot-assisted stroke rehabilitation using 3D skeletal data derived from video.
2. Improve the efficiency of therapy by providing real-time feedback to patients and therapists to correct undesirable compensatory movements.
3. Enhance rehabilitation outcomes by personalizing treatment strategies based on detected movement patterns.

Objectives

1. **Data Collection and Preprocessing:**
    - Collect 3D skeletal data from video footage of therapists simulating compensatory movements.
    - Collect 3D skeletal data from video footage of therapists and non-specialized subjects performing normative movements.
    - In the future, collect data from stroke patients during rehabilitation sessions to expand the dataset and test developed models.
    - Normalize and preprocess the data to ensure consistency across different subjects, movements, and simulated scenarios.
2. **Feature Extraction:**
    - Identify and extract biomechanically relevant features such as joint angles.
3. **Model Development:**
    - Train machine learning models to automatically classify movements as compensatory or normative, while also distinguishing between single and multiple compensations, as well as identifying individual compensatory movement types.
4. **Validation and Evaluation:**
    - Assess the model's performance through a basic stratified KFold cross-validation approach.
    - Additionally, evaluate the system's performance on unseen data through a Leave-One-Subject-Out cross validation technique.
    - Evaluate model's performance on patients data (ongoing).
5. **Implementation in Real-Time (ongoing):**
    - Implement the framework in a clinical environment to provide real-time feedback during rehabilitation sessions.

Research contributions

In this work, several improvements are proposed compared to existing research. First, we aim to create a dataset that includes a sufficient number of normative, single, and multiple compensatory movements, allowing for a more comprehensive analysis of different types of movements. To enhance the accuracy of the data, we use a 3D camera, specifically the Intel RealSense, which provides better depth perception. Our approach is designed to be independent of the subject, with a robust cross-validation technique using Leave-One-Subject-Out (LOSO) to ensure that the model generalizes well across different individuals. We have also developed a more robust model by incorporating an "uncertain" class to handle cases where the prediction is unclear.

The model focuses on seven compensatory movements: shoulder elevation, trunk extension, trunk flexion, trunk inclination on affected and non-affected sides, and trunk rotation on affected and non-affected sides. Additionally, we consider multiple targets in space, with a setup of 11 targets, and multiple types of compensations, expanding the range of scenarios.

The study explores three different scenarios: Normative vs Compensatory Movements, Single vs Multiple compensations, and Single Compensatory Movements alone. Unlike previous research, which mainly focused on the identification of upper body features [23], our approach takes into account various angles from the entire body, providing a more detailed analysis of movement patterns.

Chapter 2

Materials and Methods

In this chapter, a detailed description of the materials and methods applied throughout this study is provided, focusing on several crucial steps, including data collection, processing, feature extraction, and model development.

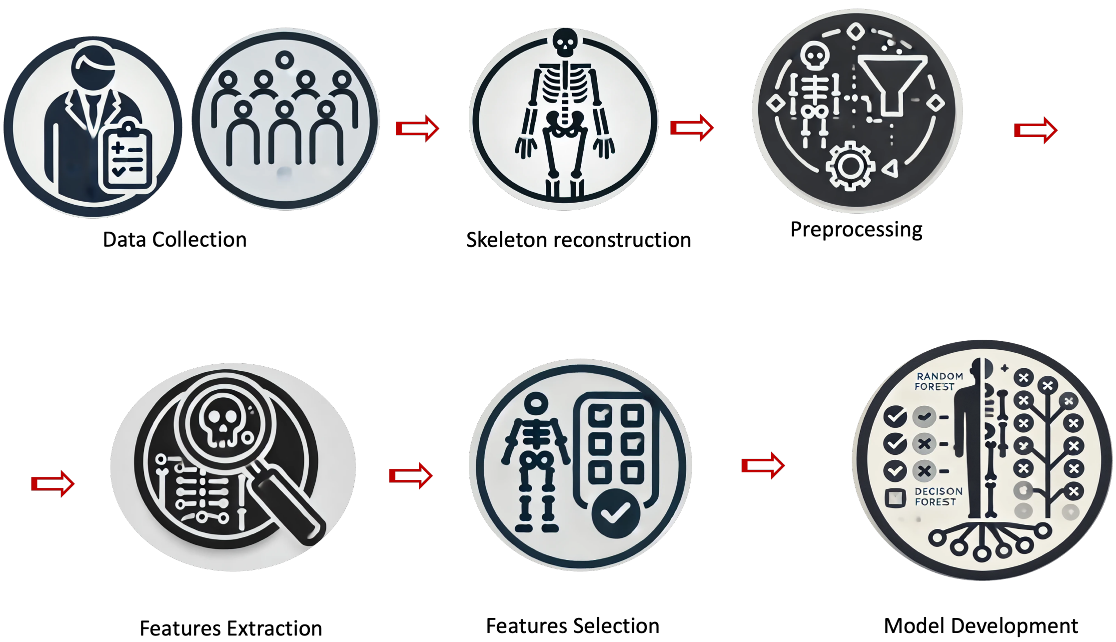

Figure 2.1: Pipeline.

2.1 Preliminary stage Phase II

As mentioned, different reasons have motivated this study. First, 3D HPE offers greater potential than 2D methods in understanding the biomechanics of body movements and positions. Additionally, with a three-dimensional estimation of the subject's skeleton, it becomes possible to capture features that 2D approaches either inaccurately estimate or are unable to detect. The ability to estimate these new features may prompt us to potentially allow for the identification of new compensatory movement strategies. Lastly, 3D HPE could enable a more comprehensive analysis of multiple compensatory movement strategies, an area where 2D estimation has shown limitations.

For this step, we decided to leverage LightBuzz's potential for the skeleton reconstrution.

LightBuzz

LightBuzz [41] is a company offering body-tracking software compatible with various platforms and devices. One of their products is the LightBuzz SDK, a cross-platform software development kit for creating motion capture applications. It supports a wide range of cameras, including smartphone cameras, USB webcams, Apple LiDAR, Intel RealSense, Azure Kinect, Structure Core, Luxonis OAK-D, and more. The LightBuzz SDK can track up to 20 human bodies with 35 joints each (Figure 2.2) and analyze human motion across different planes.

These devices provide the (x, y) coordinates of the subject's keypoints, and with depth cameras, the z-coordinate can also be captured. However, relying solely on LiDAR technology for 3D pose reconstruction has limitations, such as potential obstructions between the patient's body and the camera, like the robotic arm in our case. LightBuzz compensates for these limitations by post-processing the depth data, improving the accuracy of human pose estimations.

The robotic arm presents a challenge, as it can obstruct accurate predictions of the subject's skeleton. In some instances, the predictions exhibit significant errors (examples in Figure 2.3). To improve skeleton prediction, we focused on two key strategies:

1. Stabilizing the hip position: upon reviewing the side view of the skeleton, we noticed that the hip position often deviated from its original alignment. Given the seated posture of the subject, we chose to fix the hip position. While not an ideal solution, it is acceptable for this preliminary phase, though future efforts will focus on finding a more robust approach.

2. Estimating the subject's arm position using the robotic arm's position (ongoing implementation): marker points were placed on the robotic arm to track its

spatial position. Using geometric calculations, the aim is to improve the accuracy of the subject's keypoints, leading to a more accurate skeleton estimation.

While improvements are noticeable, particularly in the lower body, LightBuzz's predictions still deviate from reality, prompting us to explore alternative solutions.

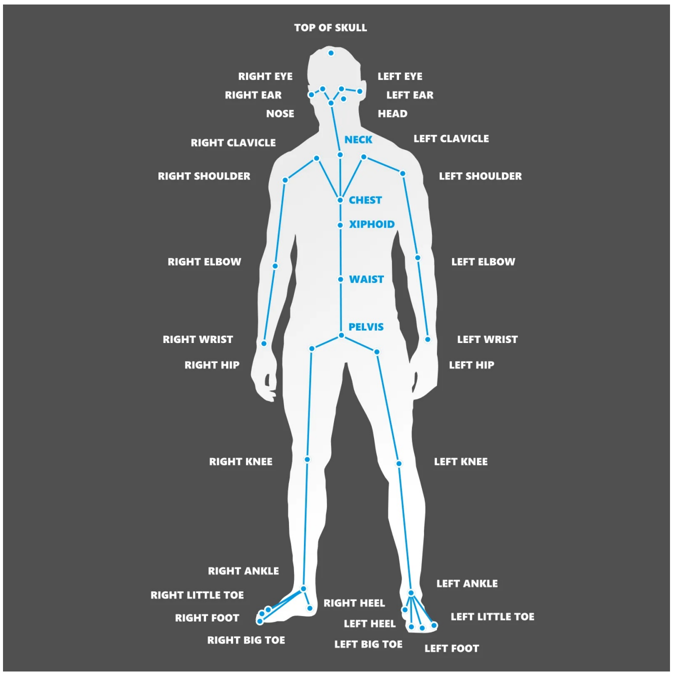

Figure 2.2: Skeleton of the subject provided by LightBuzz.

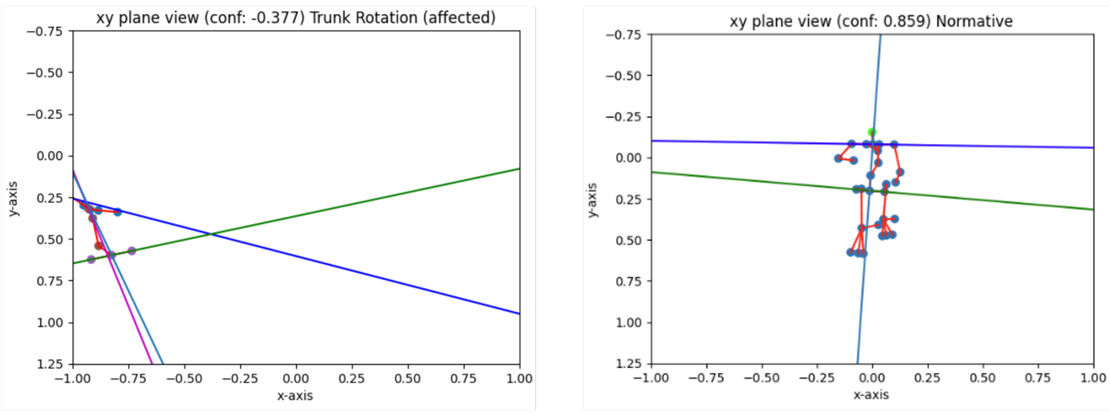

Figure 2.3: Outliers in skeleton reconstruction with Lightbuzz.

2.2 Experimental Setup and Workstation Configuration

Three components make up the complete system used (Figure 2.4):

1. Upper Limb Robotic Device (BURT - Barrett Upper-Extremity Robotic Trainer) [42]: a cohesive hardware and software rehabilitation system (Figure 2.4).

2. Cameras to capture compensatory movements (Intel RealSense).

3. A tool to support data collection, to assist during therapy sessions and (ongoing) provide feedback (PostureCheck Tool).

The setup used for this study is depicted in Figure 2.5, which illustrates the configuration of the upper limb robotic device. This device can be positioned on either the right or left side, depending on the affected limb of the subject. To capture the movements, we used an Intel RealSense camera placed approximately two meters in front of the subject, aligned with their chest.

Figure 2.4: Upper Limb Robotic Device and workstation.

A custom-developed interface in Unity was used to initiate the recording process. This interface triggered the motion capture and skeleton reconstruction framework, which automatically ran during each session. For each frame, the system produced three output files:

1. Body file: containing 35 joints with 3D coordinates (x, y, z) and a confidence measurement for each joint.

2. Color file: providing the RGB image of the scene.

3. Depth file: representing the depth map of the captured area.

The data was captured at a rate of 30 frames per second and automatically stored in folders corresponding to each specific movement, facilitating the labeling process for the classification model.

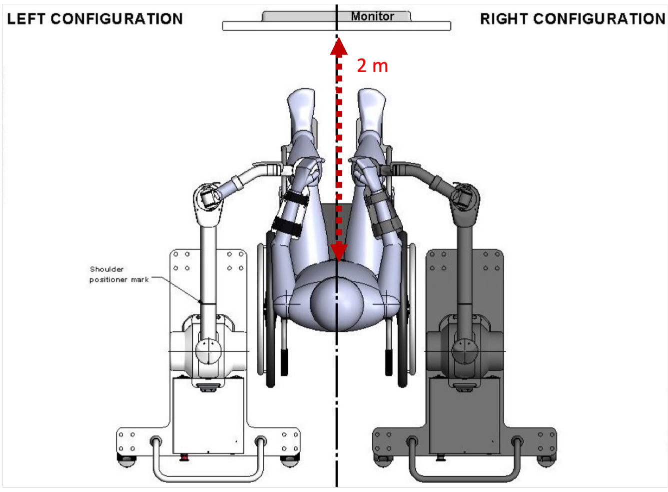

Figure 2.5: Upper Limb Robotic Device Setup.

2.3 Data Collection

The starting point of this project foresees the creation of the dataset, by gathering data from 22 participants, which included:

- 8 experienced therapists, who simulated compensatory movements as part of a preliminary phase, other than contributing to the normative movement data.
- 14 non-specialized individuals, who contributed to the normative movement data.

All movements were performed using the Upper Limb Robotic Device, which provided support and ensured consistent motion patterns across subjects. It is also possible to adjust the level of support provided by the system; however, this functionality was not utilized during data collection. To establish a foundation for the models, we initially collected simulated compensatory movements from therapists, as their expertise provided a reliable representation of these movements. This phase was designed to test and refine the model before moving on to patient data. In future stages, we aim to reduce the need for extensive patient data collection, using patient-specific data primarily to validate and fine-tune the model.

Using simulated data in stroke rehabilitation offers two key advantages. First, post-stroke fatigue limits patients' ability to consistently perform therapy exercises, making it hard to gather sufficient training data from them. Second, stroke patients' varying levels of physical impairment can prevent them from completing specific or difficult tasks, further complicating data collection [24].

For this study, we identified 11 target movements across both compensatory and normative categories (Figure 2.6):

1. Reaching forward
2. Reaching backward (towards the body)
3. Reaching level and affected (0 degrees)
4. Reaching up and affected (45 degrees)
5. Reaching up and center (90 degrees)
6. Reaching up and non-affected (135 degrees)
7. Reaching level and non-affected (180 degrees)
8. Reaching down and non-affected (225 degrees)
9. Reaching down and center (270 degrees)
10. Reaching down and affected (315 degrees)
11. Center position

In total, the ontology of the collected movements is composed by 852 unique combinations, encompassing both single compensatory movements and combinations of up to four simultaneous compensatory movements. This reflects the common occurrence of multiple compensations during movement, which often overlap in real-world scenarios. Each movement was performed with varying levels of severity, categorized as either severity 2 or 3. For multiple compensatory movements, all possible combinations of severities were recorded—such as both movements performed at severity 2 or 3, or a combination of severity 2 for one movement and severity 3 for another. This detailed ontology was developed in close collaboration with clinical experts, ensuring the dataset's clinical relevance and robustness.

Due to time constraints and the availability of therapists, it was not feasible to collect the entire ontology. Collecting data for every possible combination of compensatory movements would have required a significant amount of time and resources. Furthermore, given the early stage of this project, it would have been difficult to fully utilize all the data. Therefore, we prioritized a subset of the

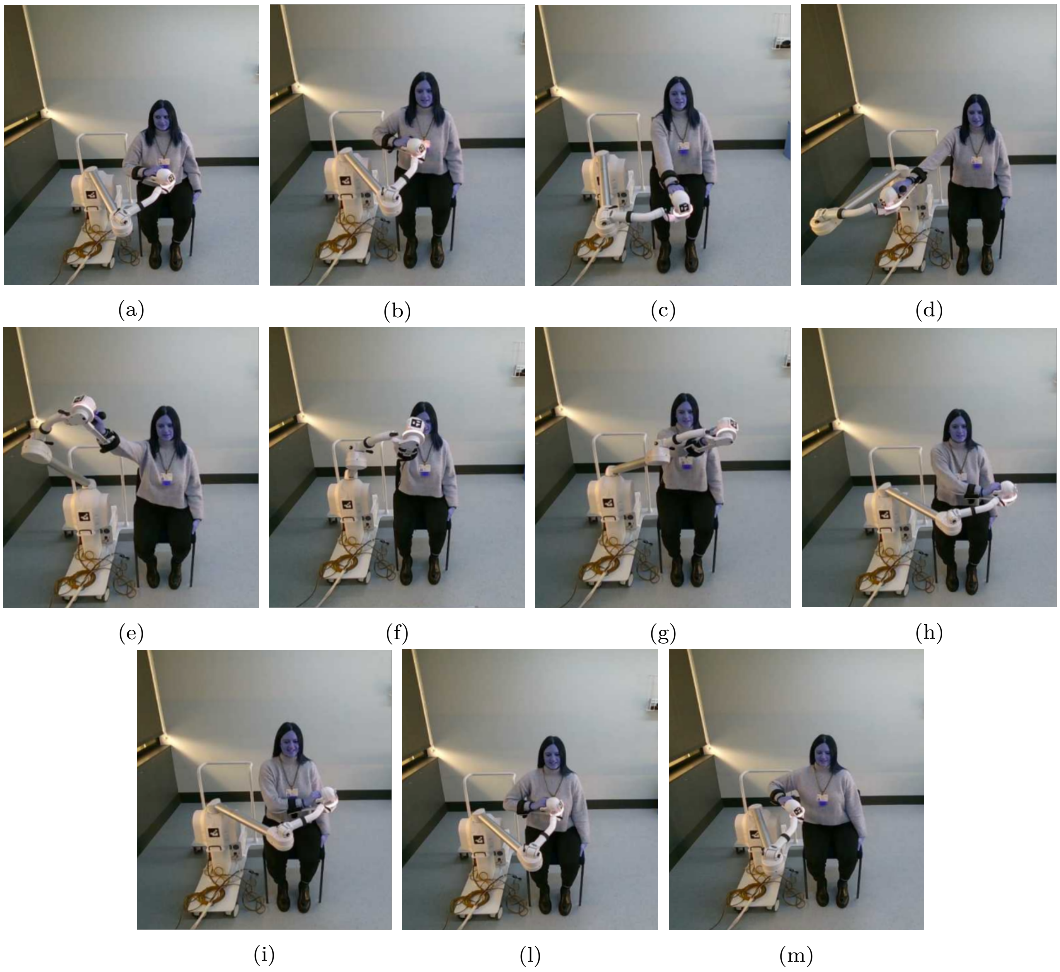

Figure 2.6: Illustration of the 11 target movements. (a) Center, (b) Reaching backward, (c) Reaching forward, (d) Reaching level and affected (0 degrees), (e) Reaching up and affected (45 degrees), (f) Reaching up and center (90 degrees), (g) Reaching up and non-affected (135 degrees), (h) Reaching level and non-affected (180 degrees), (i) Reaching down and non-affected (225 degrees), (l) Reaching down and center (270 degrees), (m) Reaching down and affected (315 degrees).

ontology, allowing us to focus on key aspects that are manageable at this point, while still laying the groundwork for further expansion in future phases.

In this phase of the project, we focused on collecting data for single compensatory movements associated with each specific target. It is worth noting that certain compensatory movements, which were deemed unlikely to occur with specific targets, were excluded. For instance, trunk flexion was only recorded in association with targets where it was most likely to be observed, based on clinical expectations.

Additionally, only three therapists contributed in the collection of multiple compensatory movements, with a focus on cases involving two simultaneous compensatory movements, with all the combinations of severities.

For each subject, we recorded five repetitions of every battery of compensatory movements: a single battery of single compensatory movements was composed of 76 movements, while, for multiple compensatory movements, we collected 244 movements for each repetition. On the other hand, all participants contributed to the collection of normative movement data, performing up to 30 repetitions for each of the 11 target movements. This provided a comprehensive dataset of 330 normative movements per subject.

The result is a robust and well-structured dataset that captures a wide range of compensatory and normative movements. The variety of compensations, coupled with the inclusion of multiple severity levels, allows for a detailed analysis of compensatory strategies. This dataset lays the groundwork for training models to accurately detect and differentiate between normative and compensatory movements, and to handle the complexity of overlapping compensations in real-world scenarios.

Overview of Compensatory Movements

In this study, we focused on seven specific compensatory movements, which were identified as key for analyzing compensatory strategies in upper limb rehabilitation. These movements include:

1. Trunk extension
2. Trunk flexion
3. Trunk inclination towards the affected side
4. Trunk inclination towards the non-affected side
5. Trunk rotation towards the affected side
6. Trunk rotation towards the non-affected side

7. Shoulder elevation

A detailed description in the Table 2.1.

Table 2.1: Single Compensatory Movements Description.

Movements	***Biomechanical Description***
Trunk Rotation	Rotational angle of trunk and/or alignment of clavicles/shoulders relative to pelvis
Trunk Inclination	Angle and/or displacement of trunk segment to their left or right. May display appropriate shortening/lengthening of sides
Trunk Flexion	Angle and/or displacement of trunk segment to the front
Trunk Extension	Angle and/or displacement of trunk segment to the back
Shoulder Elevation	Angle and/or displacement of clavicles/shoulder upwards

2.4 Data Preprocessing

In our data collection and processing workflow, the primary focus was on body files, as they contain the essential data for skeleton reconstruction, with the 35 joint coordinates. Each frame captured during the motion sequence resulted in three output files: a body file with 3D joint positions, a color file for RGB images, and a depth file for distance measurements. For each movement, we selected 10 key body files corresponding to frames where the subject was closest to the target position. The rationale behind this was to capture the motion as it approaches the target and reaches its peak extension, and to avoid to rely on only one frame. The division of the movement into two parts—one from the start to the target and the other from the target back to the initial position—allowed us to focus on the most critical and insightful part of the motion.

Before analyzing the data, we needed to ensure consistency in the orientation of the body files, especially when the robotic device was placed on the left side of the subject. To account for this, we flipped the images so that the affected arm was always positioned on the right side, ensuring uniformity across the dataset.

In addition, we excluded the target center movement from our analysis. This movement did not involve significant motion, as the subject was required to stay relatively still, and thus, it did not provide meaningful compensatory behavior data.

Normalization of the Data

Before outlier removal step, we normalized the data to ensure that all measurements were comparable across subjects. For normalization, we used a reference position called Reaching Level and Affected (0°) for each subject. In Figure 2.7, the RGB image and skeleton reconstruction of the reference position are depicted. For this reference position, as with all movements, we considered 10 frames. Instead of calculating the normalization trunk length based on a single frame, we averaged the trunk length across all 10 frames of that position. All subsequent data was normalized by dividing each coordinate by the average trunk length to account for variations in body size and posture among subjects. In Figure 2.8, we can see a comparison between the skeletons reconstructed before and after the normalization.

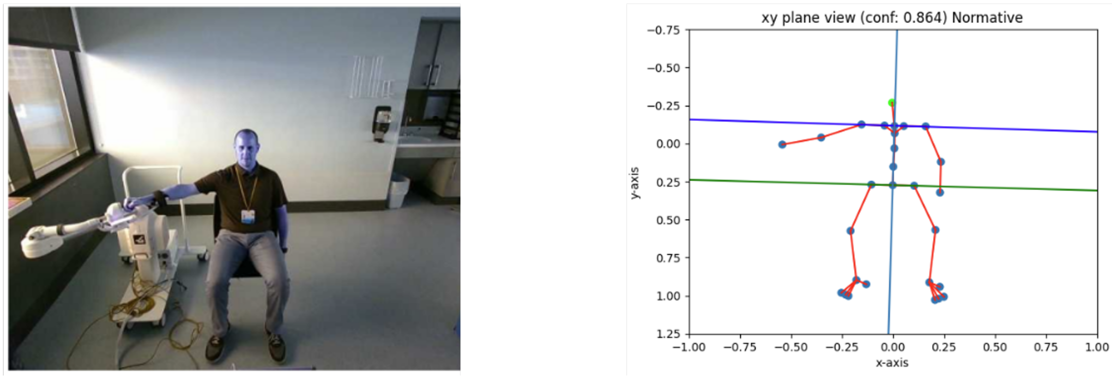

Figure 2.7: Reference position: RGB image and skeleton reconstruction.

Outlier Removal

We then performed an outlier removal process to enhance data quality. One of the main causes of inaccurate reconstruction is the occlusion caused by the robotic device. In certain movements, particularly when the device passes in front of the subject or very close to the limbs, it may be mistaken for part of the body. As a result, the system might incorrectly identify the robotic arm as the actual limb, leading to errors in the reconstruction process.

First of all, in the body file reading phase, we removed body files with more than 35 joints (due to other subjects in the scene) and empty body files. Afterwards, outliers were defined as skeleton configurations where specific segments (i.e. left arm joint segments, neck-chest-xiphoid joint segments, chest-waist-pelvis joint segments, trunk length, right leg segments, and left leg segments) showed abnormal lengths. We computed the mean (μ) and standard deviation (σ) for the segments considered over multiple movements per subject, and excluded any

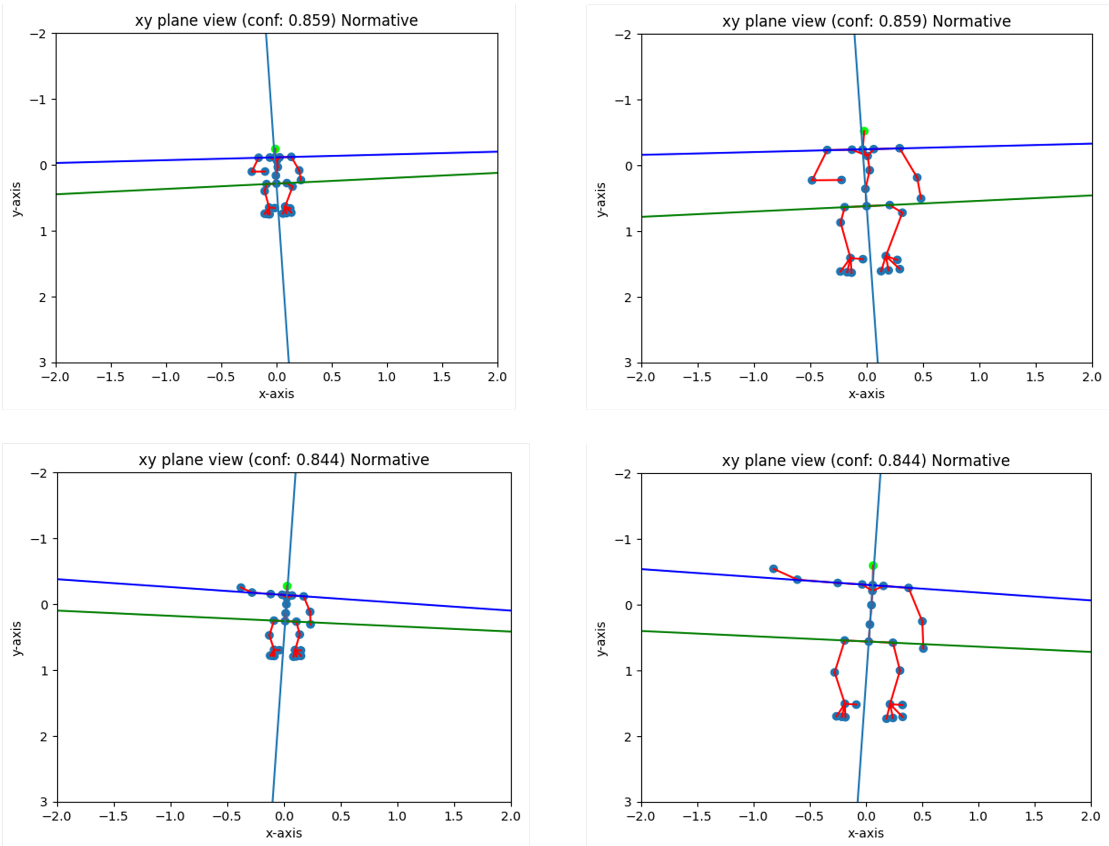

Figure 2.8: Skeleton reconstruction before and after normalization for center (first row) and 45° targets (second row).

skeletons where the measure considered exceeded the $\mu \pm \sigma$. This process helped to remove inconsistencies caused by noisy sensor data or erroneous joint estimations. Some outliers examples in Figure 2.9.

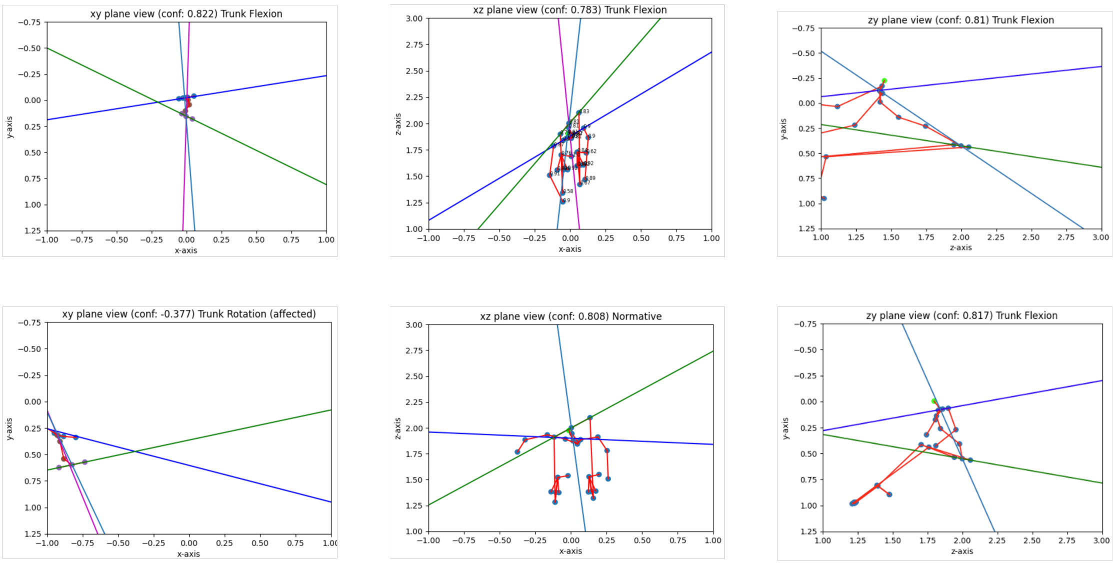

Figure 2.9: Examples of inaccurate skeleton reconstruction on coronal, transverse, sagittal planes.

The described preprocessing steps, including flipping for consistency, normalization, and outlier removal, ensured that the dataset was ready for further analysis, providing a stable basis for understanding compensatory movement patterns in upper limb rehabilitation.

In Figure 2.10, an overview of the cleaned dataset.

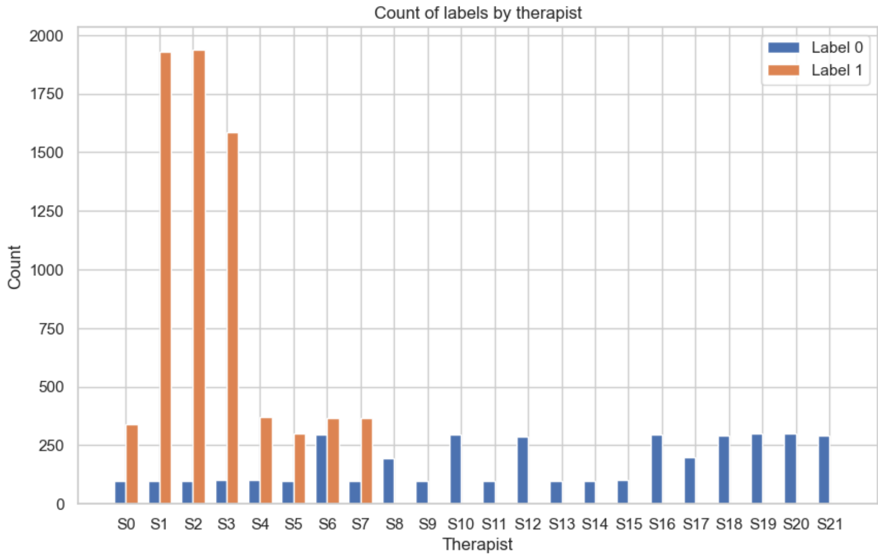

Figure 2.10: Overview of the cleaned dataset (Normative data -Label 0, Compensatory Movements - Label 1).

The next step in the project involves feature extraction and selection, aimed at identifying the most relevant characteristics from the skeletal data, and the model development, where machine learning algorithms are trained on the cleaned dataset to classify and predict compensatory behaviors. By focusing on both the identification of key features and the construction of accurate models, we aim to enhance the system's ability to effectively analyze and detect compensatory movements, ultimately contributing to more targeted rehabilitation strategies.

2.5 Feature Extraction

In the feature extraction phase, following clinicians suggestions, we identified three main groups of biomechanical features: spine metrics, upper and lower axis metrics, and joint coordinates. A total of 60 features were extracted, along with the target information, to capture the compensatory movement strategies of the patients. An overview of the features considered with an accurate description in Figure 2.13, Figure 2.14, and Figure 2.15.

Specifically, we focused on three axes: the upper axis, lower axis, and spine axis. The upper axis covers joints in the upper part of the body, including the neck, clavicles, and shoulders, while the lower axis includes the pelvis and hips.

To model the relationship between these points, we calculated the coefficients of a regression line for both the upper and lower axes using the linear regression formula [43]:

$$y = b_0 + b_1 x \tag{2.1}$$

where

$$b_1 = \frac{SS_{xy}}{SS_{xx}}, \quad b_0 = m_y - b_1 \cdot m_x$$
$$m_x = \mathrm{mean}(x), \quad m_y = \mathrm{mean}(y)$$
$$SS_{xy} = \sum(y \cdot x) - n \cdot m_y \cdot m_x, \quad SS_{xx} = \sum(x^2) - n \cdot m_x^2$$

These axes were projected in the form of linear regression lines using customized linear regression techniques, which allowed for a more nuanced and more accurate understanding and projection of the movements being performed. After deriving the coefficients of the regression lines, we computed the angles between these lines and a reference, by calculating the $\theta = \arctan(\mathrm{slope})$, which were evaluated across three different planes: the xy (Coronal) plane, xz (Transverse) plane, and zy (Sagittal) plane.

Additional features were derived by calculating the average of distances between points and lines using the following distance formula:

$$\mathrm{distance} = \frac{\mathrm{abs}(-m \cdot x + y - b)}{\sqrt{m^2 + 1}} \tag{2.2}$$

where m is the slope of the line.

For the spine metrics, instead of applying regression, we computed the line passing through two key points, the neck and pelvis, using the equation:

$$m = \frac{y_2 - y_1}{x_2 - x_1}, \quad q = y_1 - m \cdot x_1 \tag{2.3}$$

In a similar way to what was done for the upper and lower regression lines, we also calculated angles for the spine line using $\theta = \arctan(\mathrm{slope})$, where the slope is represented by m.

Furthermore, to calculate the angles between body segments (Figure 2.11), the segments must first be identified. Each pair of body segments shares a common joint, denoted as joints[0]. Using this common joint, we define two vectors, $\mathbf{v_1}$ and $\mathbf{v_2}$, representing the segments. The angle between the segments is then calculated using the cosine of the angle between the two vectors [44]:

$$\mathbf{v_1} = \mathbf{joints[1]} - \mathbf{joints[0]}$$
$$\mathbf{v_2} = \mathbf{joints[2]} - \mathbf{joints[0]}$$
$$\mathrm{cosine}\theta = \frac{\mathbf{v_1} \cdot \mathbf{v_2}}{\|\mathbf{v_1}\| \|\mathbf{v_2}\|}$$
$$\theta = \arccos(\mathrm{cosine}\theta)$$

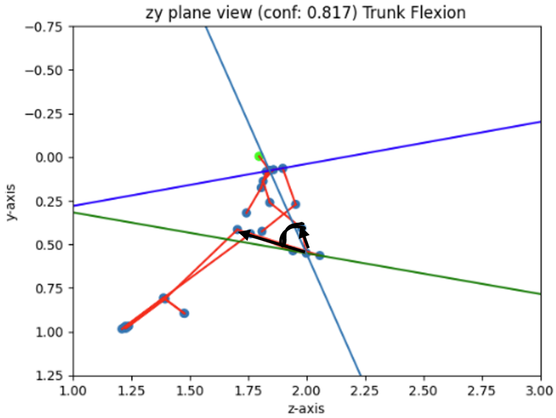

Figure 2.11: Example of angle between body segments.

Finally, we exploited the Euclidean distance to compute the distance between two points, using the standard formula [44]:

$$\text{distance} = \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2 + (z_2 - z_1)^2} \tag{2.4}$$

Additionally, we measured the differences in angles and distances between joints, capturing variations in movement patterns across different planes. By incorporating these biomechanically meaningful features, we aimed to accurately identify compensatory movements and provide a detailed characterization of patient posture during therapy. In Figure 2.12, some features illustrated.

Finally, after calculating the features for each frame, we grouped the 10 frames per movement per subject, and we averaged across the features.

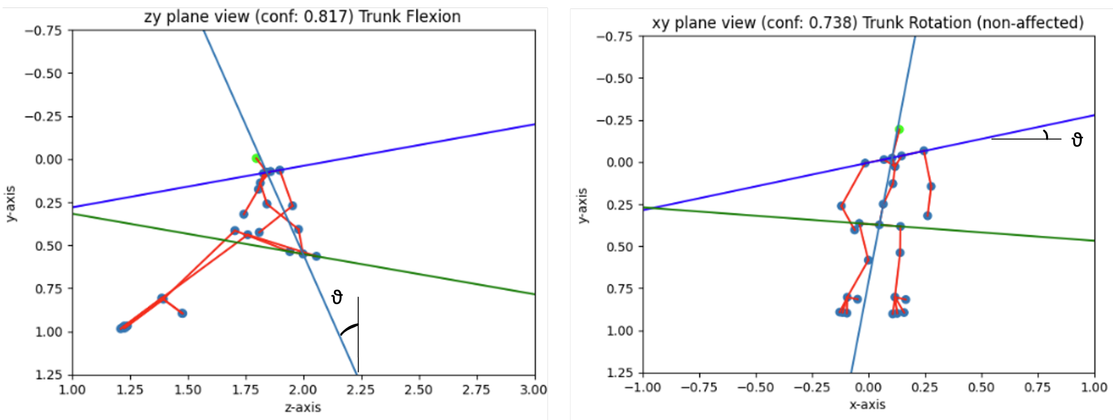

Figure 2.12: Example of features.

Spine Metrics	Description
zy_spine_axis_complementary **xy_spine_axis_complementary**	Complementary angle of the spine inclination along the zy and xy planes (sagittal and coronal).
zy_spine_axis **xy_spine_axis**	Inclination of the spine along the zy and xy planes (sagittal and coronal).
xy_necknose_angleCompl **zy_necknose_angleCompl**	Complementary angle of the neck-to-nose line inclination along the zy and xy planes (sagittal and coronal).
xy_necknose_angle **zy_necknose_angle**	Angle of the neck-to-nose line inclination along the zy and xy planes (sagittal and coronal).
trunkLengthNeck3D	Length of the neck-pelvis segment in 3D.
skullNeckDistance3D	Length of the skull- neck segment in 3D.
distanceTrunkJointstoTrunk	Mean of distances of joints along the trunk to spine line (zy view - lateral view).
ratioTrunkShoulder3D	Ratio between trunk length and shoulders distance in 3D.

Figure 2.13: Description of the spine metrics features.

Upper and Lower Axis metrics	Description	Upper and Lower Axis metrics	Description
xy_upper_axis **xz_upper_axis**	Inclination of the upper axis along the xy and xz planes (coronal and transverse).	**xz_lower_axis** **xy_lower_axis**	Inclination of the lower axis along the xy and xz planes (coronal and transverse).
xyUpperSlope **xzUpperSlope**	Slope of the upper axis along the xy and xz planes (coronal and transverse).	**xyLowerSlope** **xzLowerSlope**	Slope of the lower axis along the xy and xz planes (coronal and transverse).
shlength3D	Shoulders distance in 3D.	**lengthHip3D**	Hips distance in 3D.
xz_upper_lower_angle	Angle between the upper and lower axis along the xz-axis (transverse).	**xy_hipshoulder_difference**	(xy_upper_axis - xy_lower_axis)
xy_upper_lower_distance **xz_upper_lower_axis_distance**	Mean of distances of upper axis joints to lower axis along the xy and xz planes.	**xy_upper_lower_slope_difference** **xz_upper_lower_slope_difference**	(xyUpperSlope - xyLowerSlope) (xzUpperSlope - xzLowerSlope)
ratiodistancesUpperLower	(xy_upper_lower_distance/xz_upper_lower_axis_distance)	**kneePelvisWaist_angle**	Angle formed between waist, pelvis, and knees (zy – sagittal plane)
distance_chestRightWrist **distance_chestLeftWrist**	Distance between the chest and right/left wrist joint.	**sum_of_means_upper_lower**	(xy_mean_upper_lower_slope + xz_mean_upper_lower_slope)
xy_LeftShoulder_inclination **xz_LeftShoulder_inclination**	Inclination of the line linking left shoulder to left clavicle along the xy and xz planes.	**ratioShoulderHipLength**	(shlength3D/lengthHip3D)
xy_RightShoulder_inclination **xz_RightShoulder_inclination**	Inclination of the line linking right shoulder to right clavicle along the xy and xz planes.	**ratioshLength_axisDistance_xy**	(shlength3D/xy_upper_lower_distance)
leftShoulderArmAngle (3D) **rightShoulderArmAngle**	Angle between the left/right shoulder and left/right arm.	**kneeDistance**	Distance between the knees.
trunkRightHipAngle **trunkLeftHipAngle**	Angle between the trunk and right/left hip.	**neckLeftShoulderAngle** **neckRightShoulderAngle**	Angle between the neck and left/right shoulder.
trunkRightShoulderAngle **trunkLeftShoulderAngle**	Angle between the trunk and right/left shoulder.		

Figure 2.14: Description of the upper and lower axis metrics features.

Joints coordinates	Description
xNose, yNose, zNose	3D coordinates of the nose.
xChest, yChest, zChest	3D coordinates of the chest.
xShoulderR, yShoulderR, zShoulderR	3D coordinates of the right shoulder.
xShoulderL, yShoulderL, zShoulderL	3D coordinates of the left shoulder.

Figure 2.15: Description of the joint coordinates features.

2.6 Feature Selection

Feature selection is a crucial step in machine learning that aims at improving model performances by identifying the most relevant features and discarding irrelevant or redundant ones. This process not only reduces overfitting but also speeds up training time, enhances model interpretability, and lowers computational cost, allowing models to generalize better [45].

Feature selection techniques can be broadly categorized into two main approaches: supervised and unsupervised feature selection [46].

Supervised Feature Selection

Supervised feature selection relies on labeled data to guide the selection process and is typically divided into three main methods:

- Wrapper Methods [45]: wrapper methods evaluate different subsets of features by training models and assessing their performance. Although highly accurate, they are computationally expensive due to the multiple iterations of model training required. A common wrapper method is Recursive Feature Elimination (RFE), which recursively removes the least important features to optimize model performance.
- Filter Methods [45]: filter methods evaluate each feature independently of model training, using statistical techniques to score the relevance of features based on their relationship with the target variable. These methods are computationally efficient and fast, although they may overlook feature interactions. Common statistical techniques include: Pearson's correlation for numerical input and output to assess linear relationships, ANOVA (Analysis of Variance) for numerical input and categorical output, Chi-squared tests for categorical input and output, and mutual information, which works well with both numerical and categorical data.
- Embedded Methods [45]: embedded methods integrate feature selection directly into the model training process. Algorithms like Lasso regression and decision trees perform automatic feature selection by assigning importance scores to features during the learning phase. These methods balance accuracy and efficiency, as feature selection occurs simultaneously with model fitting.

Unsupervised Feature Selection

Unsupervised feature selection does not rely on labeled data. It focuses on identifying the most informative features based on the inherent structure of the dataset, often using techniques such as clustering or Principal Component Analysis (PCA) to reduce dimensionality without regard to the target variable [45].

Choosing the appropriate statistical measure is crucial for accurately capturing the relevant relationships between features and the target variable. Filter-based methods are efficient for handling high-dimensional datasets and are often used during the initial stages of model development due to their speed and simplicity.

In this study, we employed a combination of the correlation matrix, feature importance given by the Random Forest, and the forward selection method to refine the features set. Initially, a correlation study has been conducted by exploiting *pandas* function [47][48]. Each value in the matrix represents the strength and direction of the linear relationship between two variables, ranging from -1 to 1. A value of 1 indicates a perfect positive correlation (both variables increase or decrease together), -1 indicates a perfect negative correlation (as one variable increases, the other decreases), and 0 suggests no linear relationship [47]. We applied the standard Pearson's correlation with a threshold of 0.80 to eliminate features that were highly correlated with each other, reducing redundancy.

Then, features were ordered by importance using the Random Forest algorithm.

Next, we applied the Forward Selection method, adding one feature at a time from the ordered list and retaining it only if it improved the model's performance by more than 0.01. This process enabled us to build a model using the most significant features while avoiding overfitting and eliminating unnecessary complexity.

In the first step of the feature selection, instead of splitting the dataset into training and testing sets, we optimized feature selection on the entire dataset. Although substantial, our dataset was not large enough to risk degrading performance by reducing the available data for training. By working with the full dataset, we maximized the utility of the data and ensured optimal feature selection. To further enhance model robustness, we balanced the dataset using downsampling, which involved reducing the size of the majority class to match that of the minority class. This prevented bias in model predictions. Subsequently, before ordering the features, we splitted the dataset in train/test set (80/20), we downsampled the training set, we prepared it (further detail in the next section), and we trained the model and the Forward Selection technique to identify the final relevant features.

2.7 Model Development

The model development process is essential for crafting a successful machine learning solution. It encompasses various stages, including dataset preparation, selecting the most suitable model, and fine-tuning it for optimal performance.

We developed and tested models for each of the three distinct scenarios (Figure

2.16):

- Normative vs Compensatory Movements (binary classification): this scenario aimed to differentiate between Normative and Compensatory movements during therapy sessions.
- Single vs Multiple Compensatory Movements (binary classification): in this scenario, we tested the model's ability to identify a single compensatory movement versus multiple compensatory movements within the same session.
- Single Movements (multiclass classification): lastly, we focused on detecting the presence of a single compensatory movement, testing the model's sensitivity to even the smallest deviation from normative behavior.

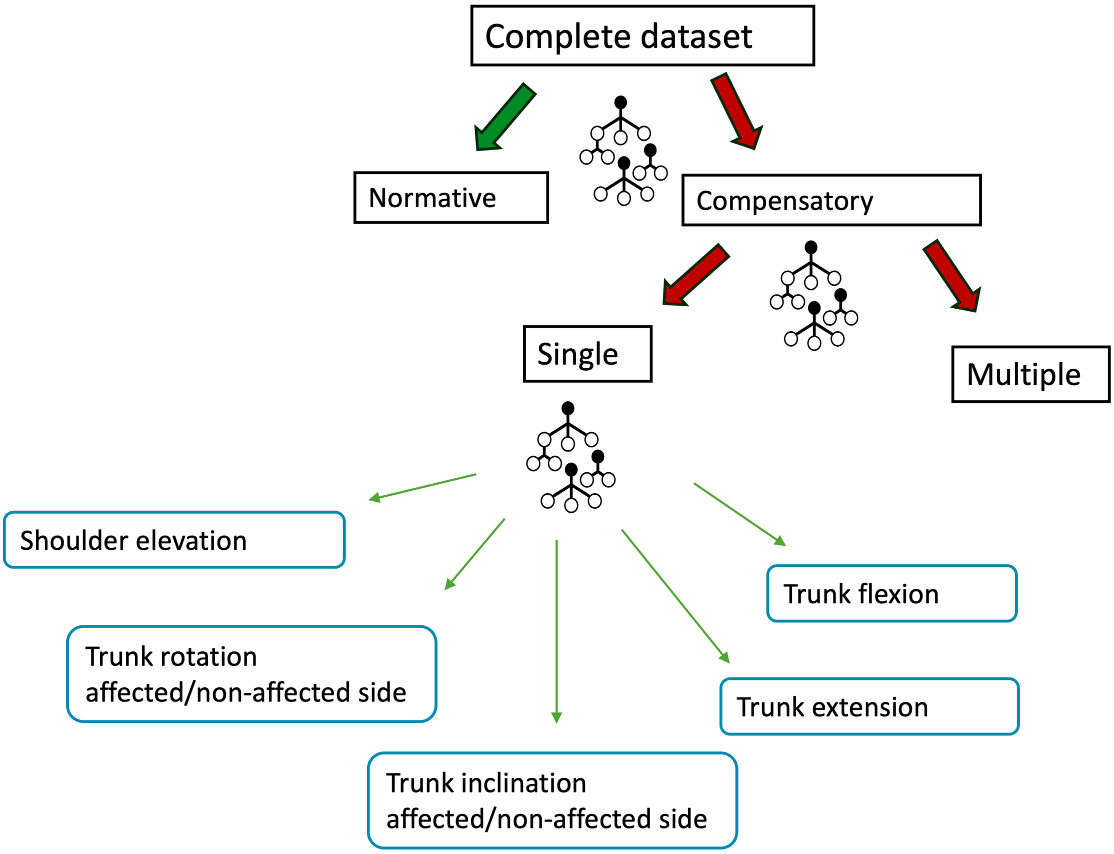

Figure 2.16: Three proposed scenarios.

We can identify some key steps shared by the three scenarios.

Preparation of the dataset

For our analysis, we utilize different subsets of the dataset across three scenarios. The dataset we collected consists of Normative Movements, Single Compensatory Movements, and Multiple Compensatory Movements.

In the first scenario, we focus on all available data, where Normative Movements are labeled as 0, and both Single and Multiple Compensatory Movements are combined under the label 1.

For the second scenario, we narrow the analysis to only Single and Multiple Compensatory Movements, excluding normative ones; in this case, Single Compensatory Movements are labeled as 0, and Multiple Compensatory Movements as 1.

Finally, in the third scenario, we focus exclusively on Single Compensatory Movements, further refining our analysis to capture these movements in isolation.

Preparation of the training set

Before selecting the relevant features and training the model, we focused on preparing the input data to ensure a balanced and representative training set. As said, in order to select the relevant features, after applying the correlation matrix, we divided the dataset in train/test (80/20) and before proceeding with the selection, we considered the following steps: balancing the dataset, normalizing it, and categorical features encoding.

Balancing the dataset

Balancing the dataset is a critical step in machine learning, especially when dealing with classification problems where the number of instances in each class is highly imbalanced. In such cases, the model tends to favor the majority class, resulting in poor predictive performance for the minority class [49]. To address this issue, balancing techniques are applied to ensure that each class contributes equally during training, improving both model accuracy and generalization.

There are several techniques for balancing a dataset [49]:

1. Undersampling/Downsampling: this method involves reducing the size of the majority class by randomly selecting a subset of its instances. While it helps achieve balance, it risks discarding potentially valuable data, which may affect model performance.

2. Oversampling: in this technique, additional samples are generated for the minority class, often by duplicating existing instances or using more advanced

methods like SMOTE (Synthetic Minority Over-sampling Technique). This approach helps balance the dataset without losing information but may lead to overfitting if not used carefully.

3. Hybrid Methods: a combination of both undersampling and oversampling can also be employed to maintain a balance between the majority and minority classes while mitigating the risks of overfitting or information loss.

To create a balanced dataset, we relied on downsampling to reduce the size of the majority class. By doing so, we ensured that each class had a similar number of samples, preventing the model from being biased toward the majority class.

Normalization

Normalization is a data preprocessing technique used to scale the features of a dataset so that they fall within a specific range, typically between 0 and 1 [50]. This process ensures that all features contribute equally to the model training process, preventing features with larger numerical values from dominating those with smaller values.

Normalization is achieved by applying a transformation to each feature in the dataset. The two most common methods are [50]:

1. **Min-Max Normalization**: this method scales each feature to a specified range, typically between 0 and 1, using the formula:

$$X' = \frac{X - X_{\text{min}}}{X_{\text{max}} - X_{\text{min}}}$$

Here, X_{min} and X_{max} represent the minimum and maximum values of the feature, respectively. This technique is useful when you want to preserve the relationships between the original data points but scale them down to a comparable range.

2. **Z-Score Normalization (Standardization)**: unlike Min-Max normalization, this method standardizes the features by transforming them into a distribution with a mean of 0 and a standard deviation of 1, using the formula:

$$X' = \frac{X - \mu}{\sigma}$$

where μ is the mean of the feature and σ is its standard deviation. This method is commonly used when the features have varying units or when the dataset follows a normal distribution.

Normalization can help speed up the training process by providing a well-conditioned dataset that is easier for optimization algorithms to handle.

In this study, we applied StandardScaler as part of the data preprocessing pipeline. The StandardScaler standardizes the features by removing the mean and scaling them to unit variance. This ensures that each feature has a mean of 0 and a standard deviation of 1, making the features comparable and improving the efficiency of algorithms that are sensitive to the scale of input data.

In this project, the StandardScaler helped normalize the input data, ensuring that the model training was not influenced by the varying scales of the features and that the optimization process converged more smoothly.

For the fourth approach—LOSO by target with an uncertain class, presented later in 2.7.4—we used the Min-Max Scaler instead, which scales the data to a range between 0 and 1. However, the overall performance remained similar compared to the StandardScaler.

Encoding of categorical features

Another essential preprocessig step is the encoding of categorical features [51], to convert non-numerical data into a numerical format that can be interpreted by algorithms. Many models, such as decision trees, random forests, and neural networks, require numerical input to process data efficiently. Categorical encoding techniques vary depending on the type of data and the requirements of the algorithm.

The most common encoding methods include [52]:

- One-Hot Encoding: converts each category into a new binary feature (0 or 1) for every unique category value. It is ideal when dealing with nominal categorical variables that do not have an inherent order.
- Label Encoding: assigns a unique integer to each category. This method is more compact but can mislead the model into thinking that the categories have an ordinal relationship when they don't.
- Binary Encoding: a hybrid technique that first converts categories into integers and then applies binary encoding, representing these integers in a binary format. This method reduces dimensionality compared to one-hot encoding, making it useful for datasets with high cardinality (many unique categories).

In this study, the target variable representing 11 distinct movement categories was transformed into a binary format, by applying the binary encoding method. This method compresses the categorical data while retaining the meaningful distinctions between the different types of movements. Binary encoding is particularly efficient when dealing with multiple categories, as it reduces dimensionality compared to one-hot encoding.

Diving in the model development, for all three scenarios mentioned, we considered four approaches:

- Random Forest with StratifiedKFold cross validation.
- Leave-One-Subject-Out (LOSO) cross validation.
- Leave-One-Subject-Out (LOSO) cross validation with a model per target (LOSO x target).
- LOSO x target, with a classification model based on three classes: 0, 1, and uncertain class (2). This model was applied only to the binary scenarios.

2.7.1 Random Forest

For the initial and basic step of our classification task, we employed the Random Forest Classifier, a simple yet powerful ensemble learning method. Random Forest [53] is a popular choice for classification due to its ability to construct multiple decision trees and then aggregate their results, which enhances both accuracy and model stability. This approach is particularly well-suited for handling high-dimensional data, as it mitigates overfitting by using bagging (bootstrap aggregating). In our case, we selected this model to provide a straightforward starting point for the classification task.

To ensure the model could generalize well and not be biased by specific therapist-related information, we trained it without including any therapist-related features. The features used were those identified in the feature selection phase, ensuring that only the most relevant characteristics were considered.

We also used Stratified K-Fold Cross-Validation to validate our model (with 5 folds). Cross-validation [54] helps ensure that the model is not overfitting to the training data, and by using the stratified version of K-Fold, we ensured that each fold maintained the same proportion of labels, making the validation process more balanced and reliable.

2.7.2 Leave-One-Subject-Out Study - LOSO

The motivation for the second approach arises from concerns regarding the generalizability of the Random Forest model when using simple cross-validation, as all subjects were included in the training set. This raised doubts about the model's ability to perform well on unseen data. To address this, we opted for a more robust validation technique: Leave-One-Subject-Out (LOSO) cross-validation [54]. LOSO is particularly effective for datasets with inherent group dependencies, such as data collected from different subjects. It ensures that the model is tested on data from a subject who was entirely excluded from training, offering a stronger measure of the model's generalization capability.

LOSO cross-validation is advantageous in scenarios like ours, where individual differences between subjects can introduce variability in the data. By excluding one subject per iteration, we simulate real-world conditions where the model encounters previously unseen data, better assessing its performance on new subjects. This method is especially important in medical and biomechanical datasets where subject-specific variations are common and critical to evaluate.

In each LOSO iteration:

- One subject's entire data is left out as the test set, ensuring that the model has never seen the subject's data during training.

- We apply downsampling on the training set to address class imbalance, followed by standard scaling and binary encoding of the categorical target variable to ensure consistency across iterations.

- The Random Forest Classifier is trained on the remaining subjects' data, without exposure to the left-out subject.

- The model is then evaluated on the held-out subject, and this process is repeated for each subject in the dataset.

By using LOSO, we obtain a more reliable estimate of the model's performance, as it mimics the scenario of applying the model in real-world clinical settings, where the goal is to generalize well across unseen subjects.

This version enhances the description of LOSO's utility, especially in terms of group dependencies and subject-specific variability, and how it simulates real-world testing conditions.

2.7.3 LOSOxTarget

The third cross-validation scenario involves applying the previously described LOSO approach but with a separate model trained for each of the 10 movement targets (since the center target was discarded). Since these targets are distributed around the subject, some movements experienced robot occlusion, potentially affecting the model's performance. Training a model for each target allows us to evaluate whether the model remains robust across all targets or performs better on specific ones. During each iteration, we apply downsampling and normalization, but remove information related to the target itself, as it does not contribute meaningfully to the classification task.

2.7.4 Uncertainty in Model Predictions

The final approach adopts a more conservative strategy by focusing on reducing misclassifications, specifically false positives and false negatives. This method introduces an additional challenge: predicting an uncertain class when it was unclear whether a movement belonged to one of the two primary classes (e.g., Compensatory vs Normative, or Single vs Multiple Compensatory Movements). We leveraged the probability scores [53] provided by the Random Forest classifier using the tools available in scikit-learn [50]. A threshold of 0.65 was set for both classes—if the probability for a given class exceeded this threshold, the movement was classified accordingly; otherwise, it was assigned to the uncertain class. This approach was applied only to the binary classification scenarios. In each step, we applied downsampling and normalization, as we used the LOSOxtarget method, where one subject was left out during training. While this conservative approach reduces errors, it also results in the exclusion of potentially correct classifications, which further contributes to its conservative nature.

2.7.5 Model Evaluation

For classification tasks, several evaluation metrics are commonly used [55]:

- Accuracy: the percentage of correct predictions out of all predictions. It works well when the dataset is balanced, but can be misleading for imbalanced datasets.
- Precision: the proportion of true positive predictions out of all positive predictions made by the model.
- Recall (Sensitivity): the proportion of true positives out of all actual positives in the dataset.
- F1 Score: the harmonic mean of precision and recall, providing a single measure of performance, especially useful for imbalanced datasets. There are two types of F1 scores, F1-macro and F1-micro. F1-macro calculates the F1 score for each class separately and then averages them, giving equal weight to all classes, regardless of size. This is useful for imbalanced datasets where smaller classes need equal consideration. F1-micro, on the other hand, computes precision and recall globally, treating all samples equally and emphasizing larger classes, making it suitable for evaluating overall performance in balanced datasets.

For each of the four approaches, different sets of metrics are employed to evaluate the model's performance. These metrics are tailored to the specific characteristics and objectives of each approach, ensuring a comprehensive assessment of the classification tasks in each scenario:

- Random Forest with Stratified K-Fold Cross-Validation: accuracy.
- LOSO: F1-micro score.
- LOSOxtarget: F1-micro score.
- LOSOxtarget with uncertain class: we introduced several performance metrics to evaluate the classification results when incorporating the uncertain class. These include:
  - Correct Class 0: it measures the recall of class 0 (Normative Movements/Single Compensatory Movements, first and second scenario).
  - Correct Class 1: which reflects the recall of class 1 (Compensatory Movements/Multiple Compensatory Movements).
  - Percentage of Decision: which represents the specificity of class 2 (uncertain).
  - Class 0 Wrongly Classified: which shows the percentage of true class 0 instances incorrectly classified as class 1.
  - Class 0 Without Decision: which captures the percentage of true class 0 instances classified as uncertain.
  - Class 1 Wrongly Classified: reflects the percentage of true class 1 instances incorrectly labeled as class 0.
  - Class 1 Without Decision: indicates how often true class 1 samples are assigned to the uncertain class.

Results and confusion matrix showing the true positive, false positive, true negative, and false negative predictions, of the performances achieved are highlighted in the next chapter.

Chapter 3
Results

In this chapter, we present the results of the four selected methods across the three proposed scenarios. For each scenario, an overview of the results is provided in terms of accuracy and F1-micro score. Following this, we present the ranked features extracted from the data, the overall confusion matrix for all four methods, and individual confusion matrices for each target, where a separate model was trained for each target.

Each scenario is analyzed using the following methods: the Random Forest model, Leave-One-Subject-Out (LOSO) cross-validation, LOSO by target, and LOSO by target with uncertain classes. The Random Forest model was chosen with 100 trees and a random seed of 42. Feature selection was performed using a correlation matrix, followed by feature ranking through Random Forest, forward selection method, and a final Random Forest model trained on the selected features to provide the official feature ordering.

Ten targets were considered, with the “center” target removed from the analysis. While we draw conclusions from the confusion matrices for each target, it is important to note that the distribution of compensatory movements across targets is not uniform. This reflects the real-world conditions of data collection, where certain targets are associated with more likely movements, as suggested by clinicians. Therefore, the number of movements per target varies naturally, which could influence the model’s performance for certain targets.

The fourth approach—LOSO by target with uncertain classes—was applied only in the binary classification scenarios, and not for the final proposed scenario.

3.1 Normative vs Compensatory Movements

In this section, we dive into the results obtained for the first scenario, which aims to discriminate between Normative and Compensatory movements. The original dataset before downsampling was composed of 3931 Normative Movements and 7193 Compensatory Movements.

Figure 3.1 provides an overview of the accuracy and F1-micro scores obtained for the three proposed approaches. In the Subsection 3.1.4 the metrics associated with the LOSO by target method, combined with the uncertain class approach, are also presented.

Threshold Features Selection	N features	Accuracy RF	LOSO F1 micro	LOSOxtarget F1 micro
0.01	4	0.88	0.82	0.83

Figure 3.1: Normative vs Compensatory Movements: overview of the results obtained for the three methods.

3.1.1 Random Forest (RF)

In Figure 3.2, the ranking given by the RF of the features selected.

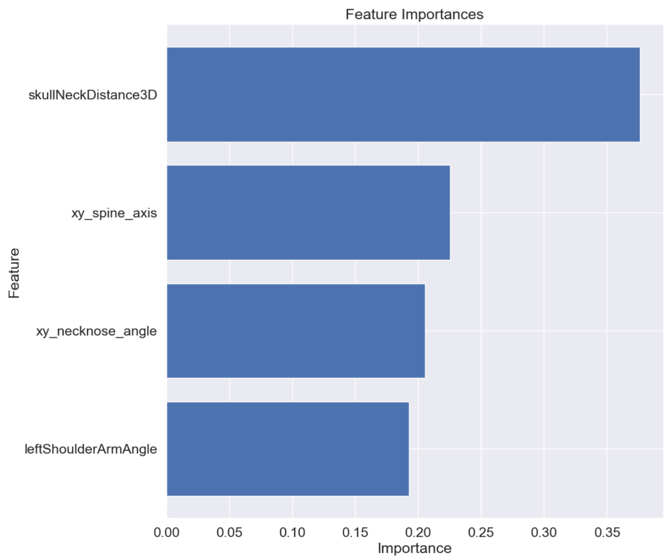

Figure 3.2: Normative vs Compensatory Movements: RF feature ranking.

The accuracy achieved in this scenario is 0.88, and Figure 3.3 highlights the confusion matrix of the test set classification.

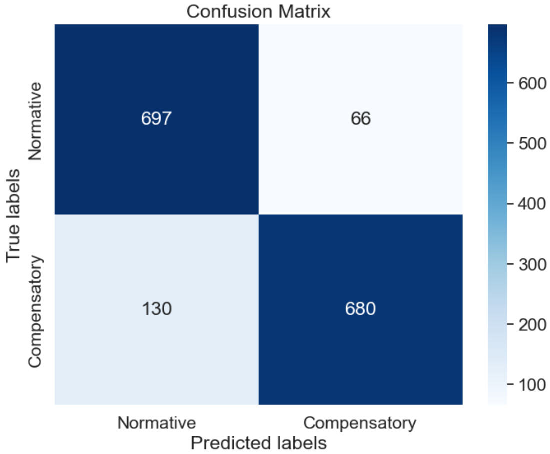

Figure 3.3: Normative vs Compensatory Movements: RF Confusion Matrix.

3.1.2 LOSO study

In the LOSO study, we generated a global confusion matrix (Figure 3.4) by aggregating the confusion matrices obtained during each iteration. In each iteration, one subject was left out, and the corresponding confusion matrix was computed. These individual matrices were then combined to form the overall confusion matrix.

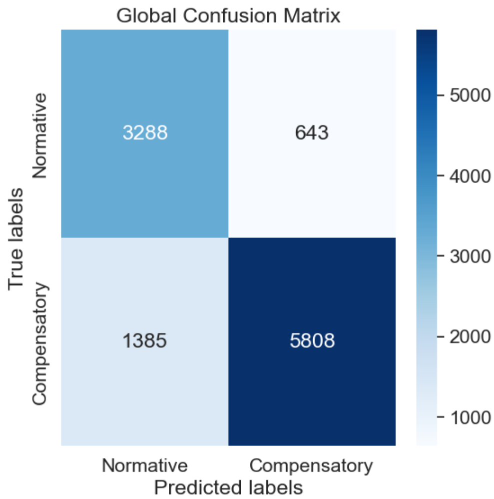

Figure 3.4: Normative vs Compensatory Movements: Confusion Matrix for LOSO Classification.

3.1.3 LOSO x Target movement study

In the LOSO by target movement study, we first present the results for each individual target (Figure 3.5) and then a global confusion matrix (Figure 3.6) where we combined the results across all subjects for every target.

We observe some performance drops, which can likely be attributed to the small dataset size. This is particularly evident for the Reaching down and center (270°) target, where only a limited number of compensatory movements were recorded, reflecting the real-worlds scenario. Additionally, some targets, such as 135°, 180°, 225°, involve occlusions caused by the robotic arm; however, this does not seem to affect the model's performance significantly. The model remains robust despite the occlusions.

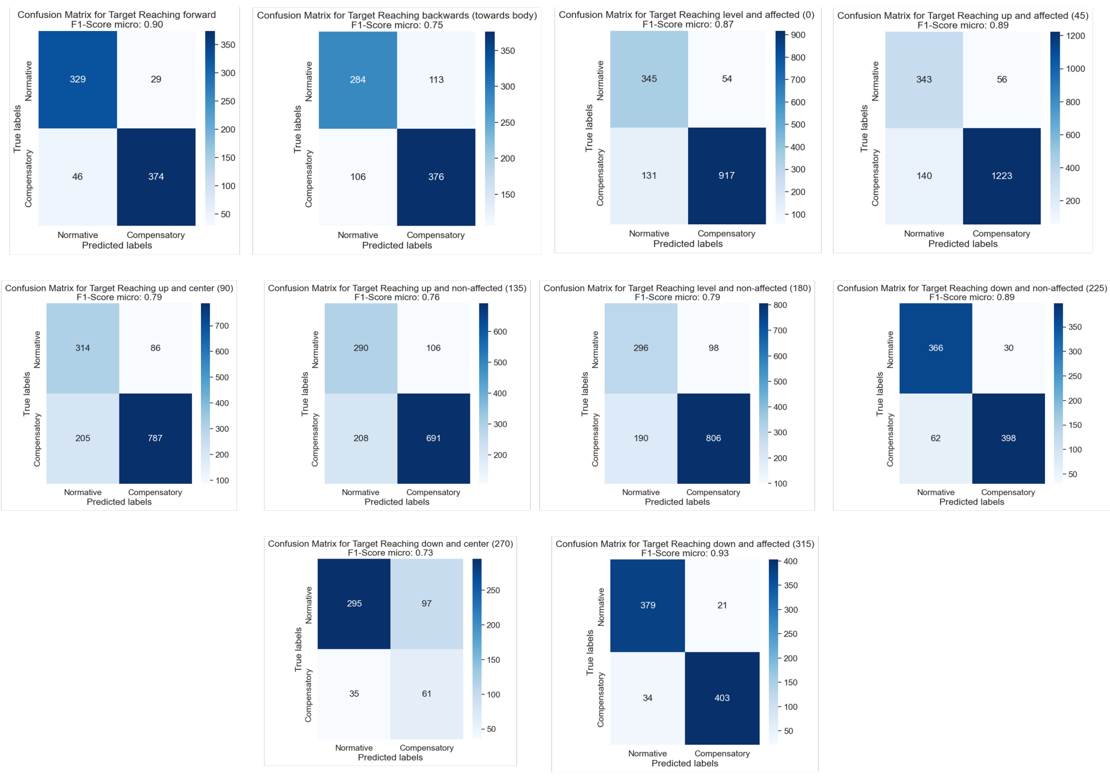

Figure 3.5: Normative vs Compensatory Movements: LOSOxTarget Confusion Matrix.

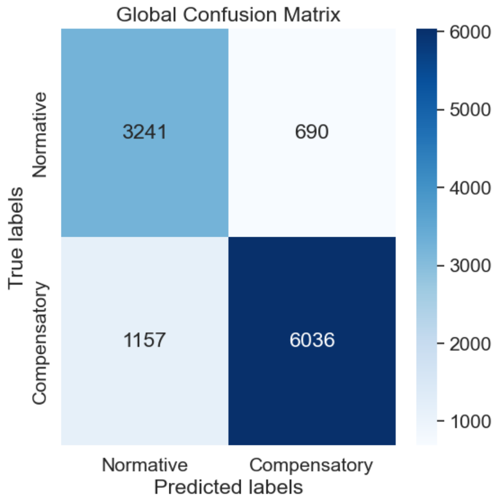

Figure 3.6: Normative vs Compensatory Movements: global Confusion Matrix for LOSOxTarget Classification.

3.1.4 LOSO x Target movement study with uncertain class

In the LOSO by target with the uncertain class study, as outlined in the theoretical section 2.7.4, we introduced an uncertain class to the standard LOSO by target study. We set a threshold of 65% for the probability decision of the Random Forest model. The results are presented both per target and globally (Figure 3.8 and Figure 3.9).

We experimented with adjusting the threshold, but its optimal value depends on which metrics we aim to improve. Ideally, we want to increase the number of correctly classified samples, reduce the number of wrongly classified samples, and raise the percentage of decisions made by the model. A balanced trade-off was achieved with the 0.65 threshold. The metrics are displayed in Figure 3.7.

When examining the performance per target, we observe a similar conclusion to the standard LOSO by target study: the results are comparable between targets with and without occlusion, demonstrating the robustness of the model. Additionally, we observe that for certain targets, such as 90° and 135°, there are many unclassified samples, highlighting, as previously mentioned, how this approach is conservative. It is important to note that these unclassified samples include both false classifications and potentially correct classifications, which means that we may be losing some valuable information.

Metric	Value
Correct Normative	70.03%
Correct Compensatory	78.45%
Percentage of Decision	86.29%
Normative Wrongly Classified	10.79%
Normative Without Decision	19.18%
Compensatory Wrongly Classified	10.83%
Compensatory Without Decision	10.72%

Figure 3.7: Normative vs Compensatory Movements: Performance Metrics for LOSOxTarget with uncertain class classification.

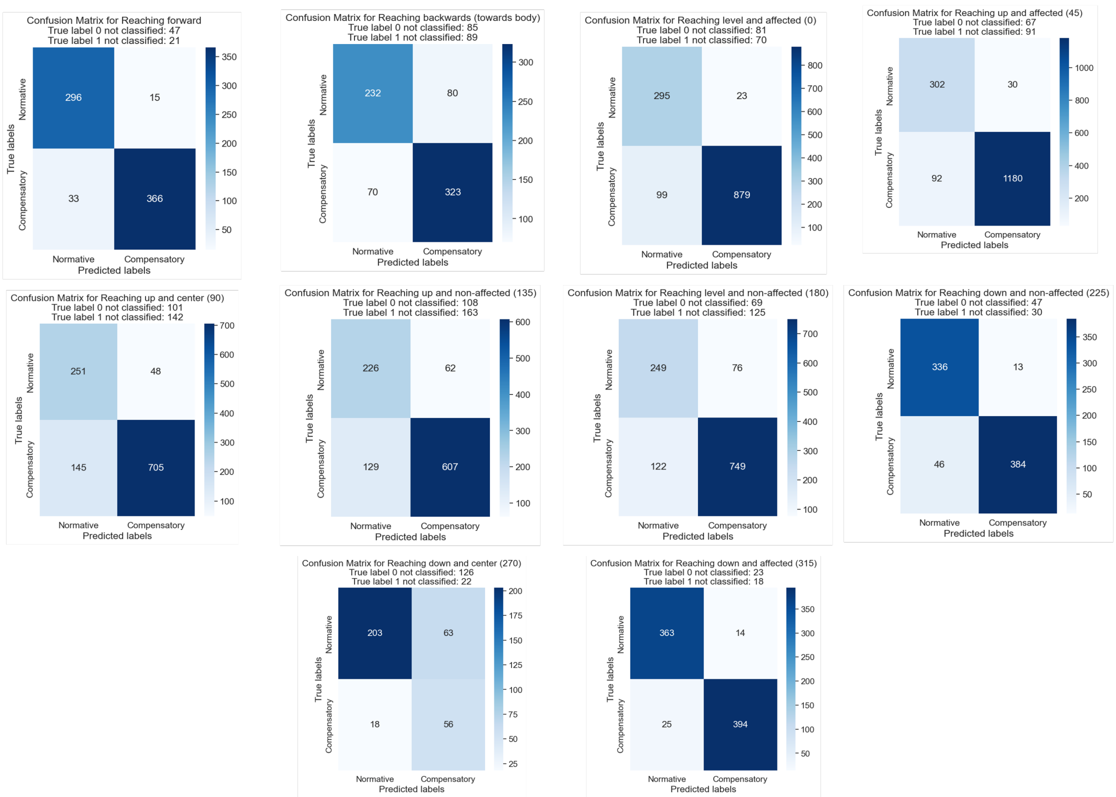

Figure 3.8: Normative vs Compensatory Movements: LOSOxTargetxUncertain class confusion matrix per target.

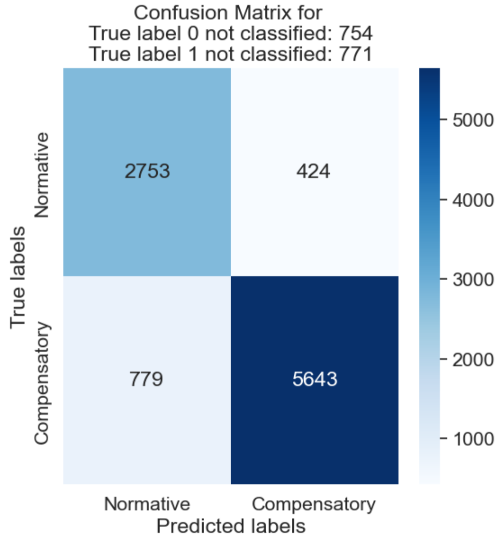

Figure 3.9: Normative vs Compensatory Movements: LOSOxTargetxUncertain class global confusion matrix.

3.2 Single vs Multiple Compensatory Movements

In this section, we present the results for the Single Compensatory vs Multiple Compensatory Movements scenario. The original dataset before downsampling was composed of 3533 Simple Compensatory Movements and 3668 Multiple Compensatory Movements.

The structure of this section follows a similar outline to the previous section on Normative vs Compensatory Movements. We begin by presenting the overall results in a summary table (Figure 3.10).

Threshold Features Selection	N features	Accuracy RF	LOSO F1 micro	LOSOxtarget F1 micro
0.01	6	0.82	0.74	0.74

Figure 3.10: Single vs Multiple Compensatory Movements: overview of the results obtained for the three methods.

3.2.1 Random Forest (RF)

In this subsection, we present the selected features (Figure 3.11) and the results obtained from the Stratified K-Fold cross-validation (Figure 3.12).

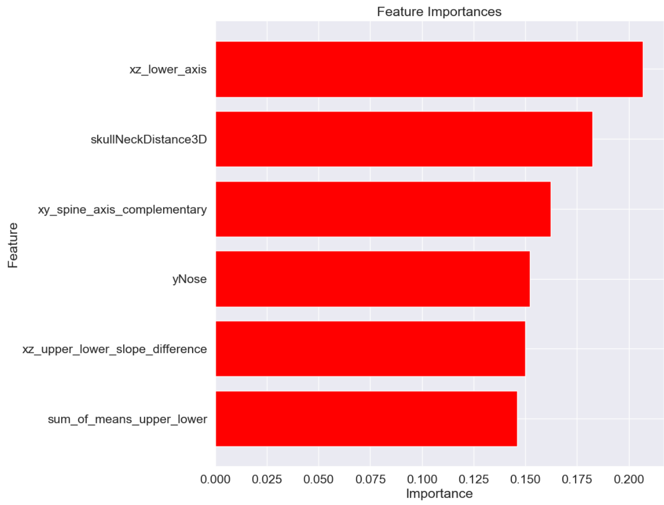

Figure 3.11: Single vs Multiple Compensatory Movements: RF feature ranking.

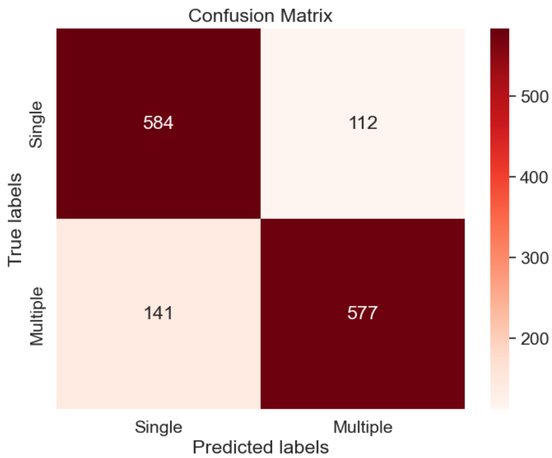

Figure 3.12: Single vs Multiple Compensatory Movements: RF Confusion Matrix.

3.2.2 LOSO study

We present the aggregated confusion matrices for the LOSO study, where each subject was left out in each iteration (Figure 3.13).

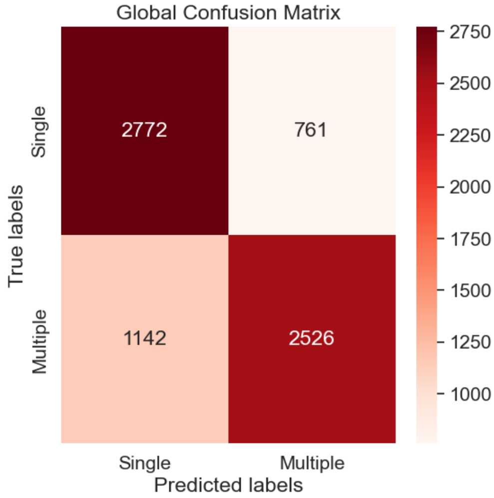

Figure 3.13: Single vs Multiple Compensatory Movements: Confusion Matrix for LOSO Classification.

3.2.3 LOSO x Target study

In this study, a consistent pattern emerged, showing a decrease in performance when the dataset was small, although the model remained robust across occlusion and non-occlusion targets (Figure 3.14).

The confusion matrix for the 270° target is particularly interesting, as it shows no false positives or incorrect predictions. This is because no multiple compensatory movements are associated with that target.

One of the challenges in this task is that the features do not have a one-to-one correspondence with compensatory movements. It is common for patients to perform multiple compensatory movements, which may resemble single compensatory movements. This may explain the observed drop in performance, with the model achieving 74% accuracy.

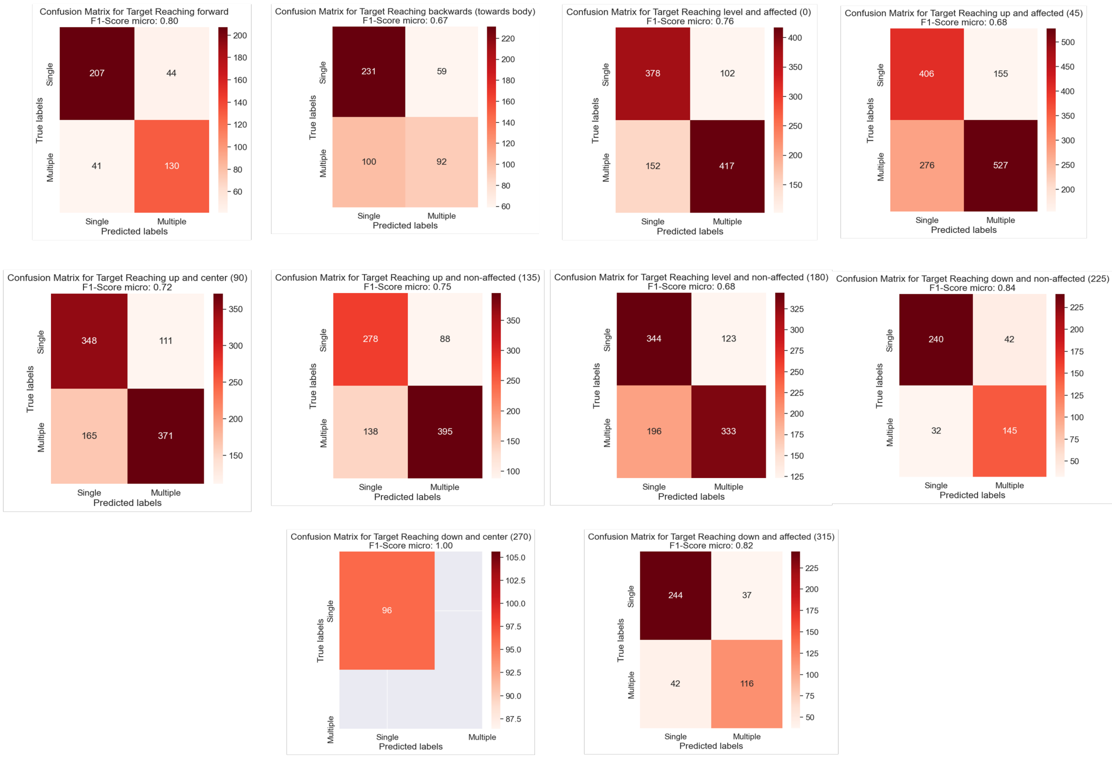

Figure 3.14: Single vs Multiple Compensatory Movements: LOSOxTarget Confusion Matrix per target.

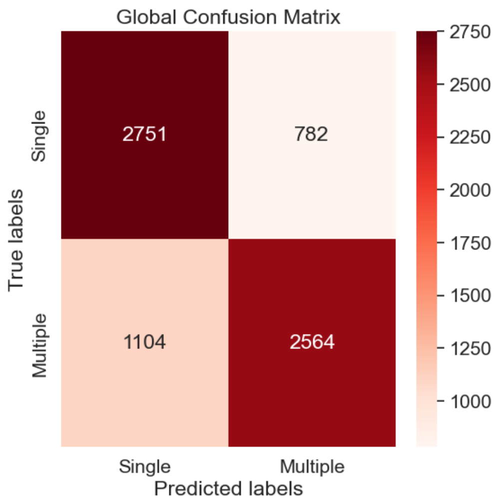

Figure 3.15: Single vs Multiple Compensatory Movements: global Confusion Matrix for LOSOxTarget classification.

3.2.4 LOSO x Target study with uncertain class

Finally, we present the study incorporating the uncertain class, including the associated metrics (Figure 3.16), target-specific confusion matrices (Figure 3.17), and the global confusion matrix (Figure 3.18). Notably, the 270° target is excluded from this analysis since no multiple compensatory movements were associated with it, so classification was not performed.

Metric	Value
Correct Normative	61.04%
Correct Compensatory	56.24%
Percentage of Decision	72.45%
Normative Wrongly Classified	12.29%
Normative Without Decision	26.67%
Compensatory Wrongly Classified	15.47%
Compensatory Without Decision	28.30%

Figure 3.16: Single vs Multiple Compensatory Movements: Performance Metrics for LOSOxTarget with uncertain class classification.

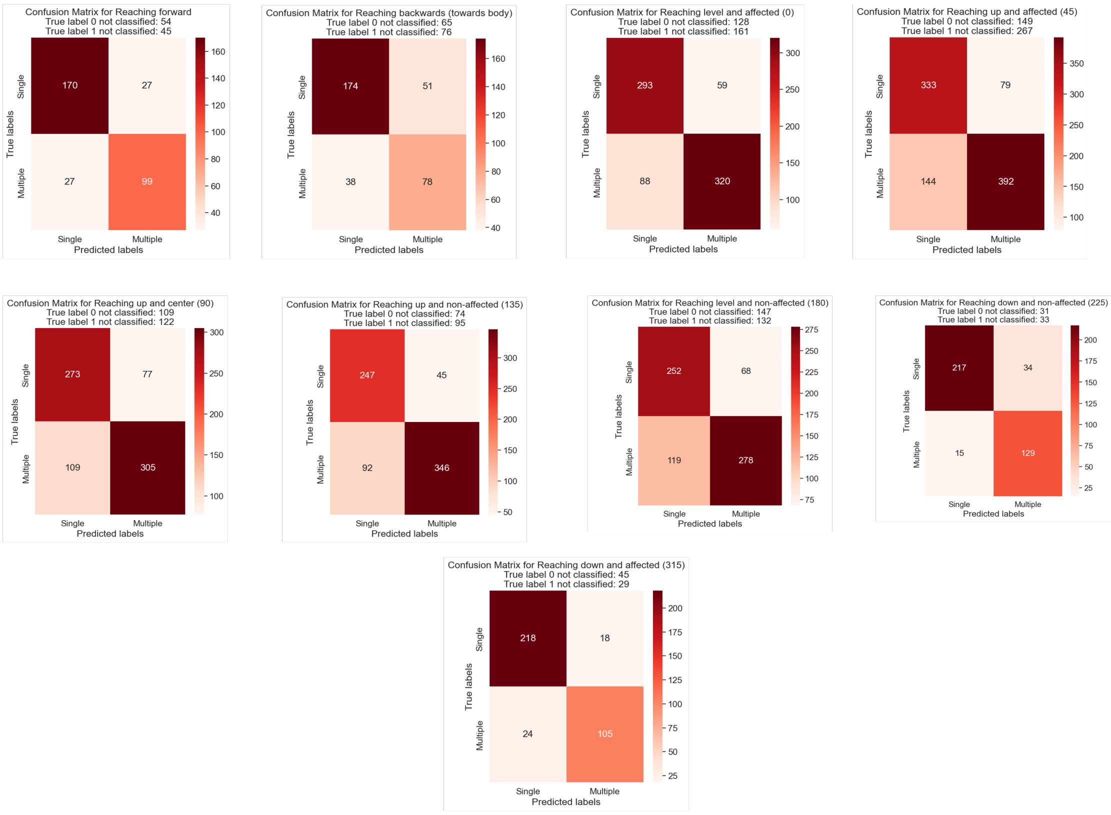

Figure 3.17: Single vs Multiple Compensatory Movements: LOSOxTargetxUncertain class Confusion Matrix per target.

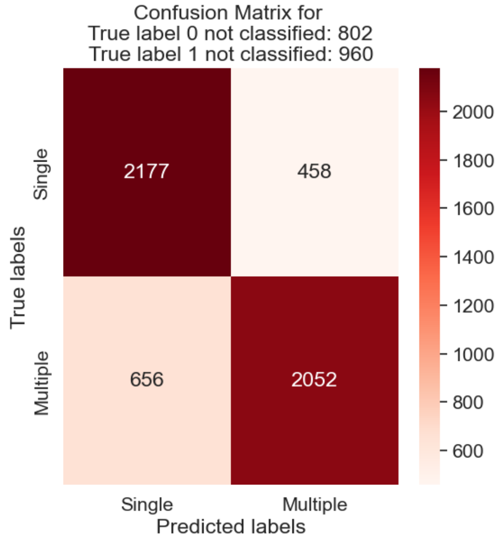

Figure 3.18: Single vs Multiple Compensatory Movements: LOSOxTargetxUncertain class global Confusion Matrix.

3.3 Single Compensatory Movements

In this paragraph of the results, we discuss the Single Compensatory Movements scenario performance. The dataset was composed of 3533 instances of Single Compensatory Movements. Figure 3.19 provides the data distribution according to these Single Compensatory Movements. After that, we present a figure displaying the overall results for the Random Forest, LOSO, and LOSOxTarget study (Figure 3.20).

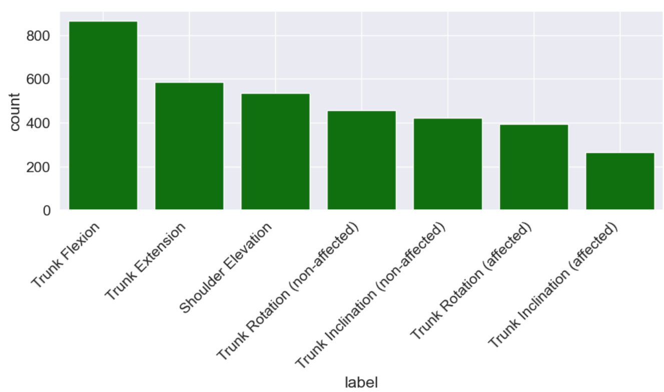

Figure 3.19: Distribution of Single Compensatory Movements.

Threshold Features Selection	N features	Accuracy RF	LOSO F1 micro	LOSOxtarget F1 micro
0.01	7	0.84	0.76	0.79

Figure 3.20: Single Compensatory Movements: overview of the results obtained for the three methods.

3.3.1 Random Forest (RF)

In Figure 3.21 and Figure 3.22, we show the ranked selected features and the corresponding confusion matrix. Following this, we provide a table summarizing the performance obtained for each movement (Table 3.1).

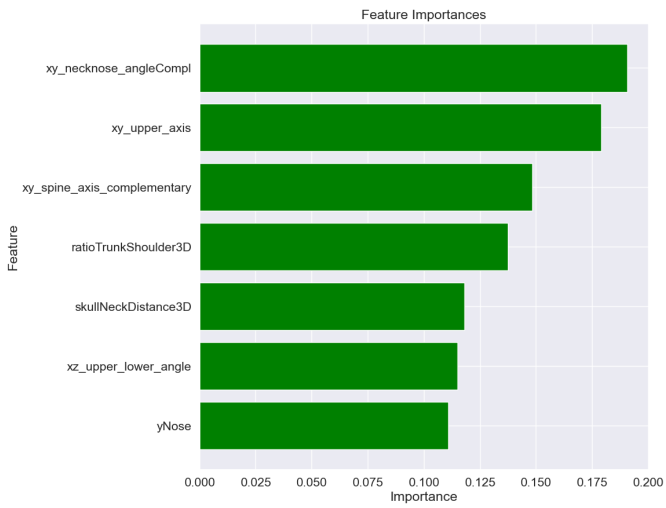

Figure 3.21: Single Compensatory Movements: RF feature ranking.

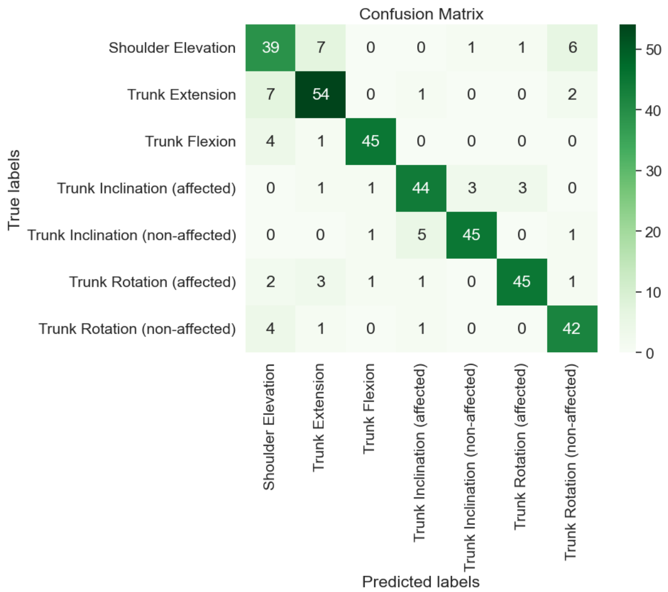

Figure 3.22: Single Compensatory Movements: RF Confusion Matrix.

Class	**F1-Score (%)**
Shoulder Elevation	71%
Trunk Extension	82%
Trunk Flexion	92%
Trunk Inclination (affected)	85%
Trunk Inclination (non-affected)	89%
Trunk Rotation (affected)	88%
Trunk Rotation (non-affected)	84%

Table 3.1: Single Compensatory Movements: F1-score from RF for each class.

3.3.2 LOSO study

In the case of the LOSO study, we observe a drop in performance, especially with difficulties discriminating movement types like extension and shoulder elevation, as we can see in the Figure 3.23. In Table 3.2, a detail of the F1-scores obtained for each class.

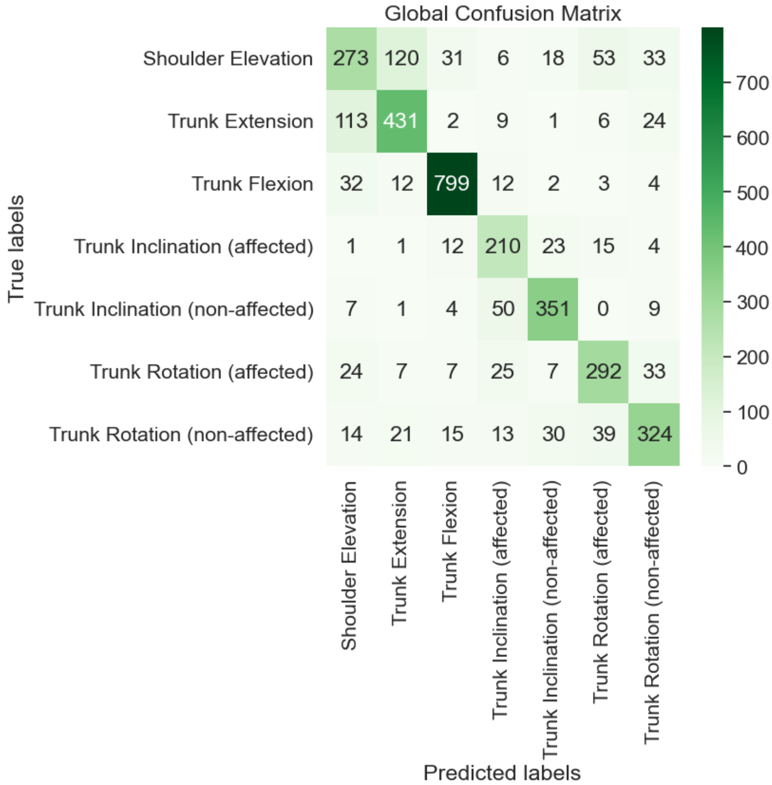

Figure 3.23: Single Compensatory Movements: Confusion Matrix for LOSO Classification.

Class	**F1-Score (%)**
Shoulder Elevation	55%
Trunk Extension	73%
Trunk Flexion	92%
Trunk Inclination (affected)	71%
Trunk Inclination (non-affected)	82%
Trunk Rotation (affected)	73%
Trunk Rotation (non-affected)	73%

Table 3.2: Single Compensatory Movements: F1-score from LOSO study for each class.

3.3.3 LOSO x Target study

For the LOSOxTarget study, we see a 3% improvement compared to LOSO, although there is a small drop from the Random Forest results. We then present the confusion matrix for the targets (Figure 3.24). As seen earlier, not all targets are associated with the same movements, as some movements are less frequent for certain targets. The 270° target is particularly noteworthy, even in the case of Single Compensatory Movements scenario, as only the flexion movement is associated with this target.

Lastly, we display the global confusion matrix (Figure 3.25). The difficulty in discriminating movements can be explained by the fact that different compensatory movements can result in similar biomechanical patterns.

To conclude, Table 3.3 offers a recap of the performance achieved for each class.

Class	**F1-Score (%)**
Shoulder Elevation	60%
Trunk Extension	68%
Trunk Flexion	93%
Trunk Inclination (affected)	78%
Trunk Inclination (non-affected)	88%
Trunk Rotation (affected)	82%
Trunk Rotation (non-affected)	78%

Table 3.3: Single Compensatory Movements: F1-score from LOSOxTarget study for each class.

We also include 2D and 3D projections of the movements (Figure 3.26). Regarding the dimensionality reduction technique, t-SNE (t-distributed Stochastic Neighbor Embedding) was used [56]. This method, t-SNE, is widely adopted for visualizing high-dimensional data by projecting it into a lower-dimensional space (2D or 3D). It preserves local similarities by mapping nearby points in high-dimensional space to nearby points in the lower-dimensional representation, making it an excellent choice for visualizing complex biomechanical movements [56]. However, it is not without limitations, such as difficulties with global structure and computational intensity for larger datasets. The projections also reflect the conclusions made regarding the difficulty in distinguishing some movements, particularly for those that exhibit similar biomechanical patterns, confirming the challenges highlighted in the classification results.

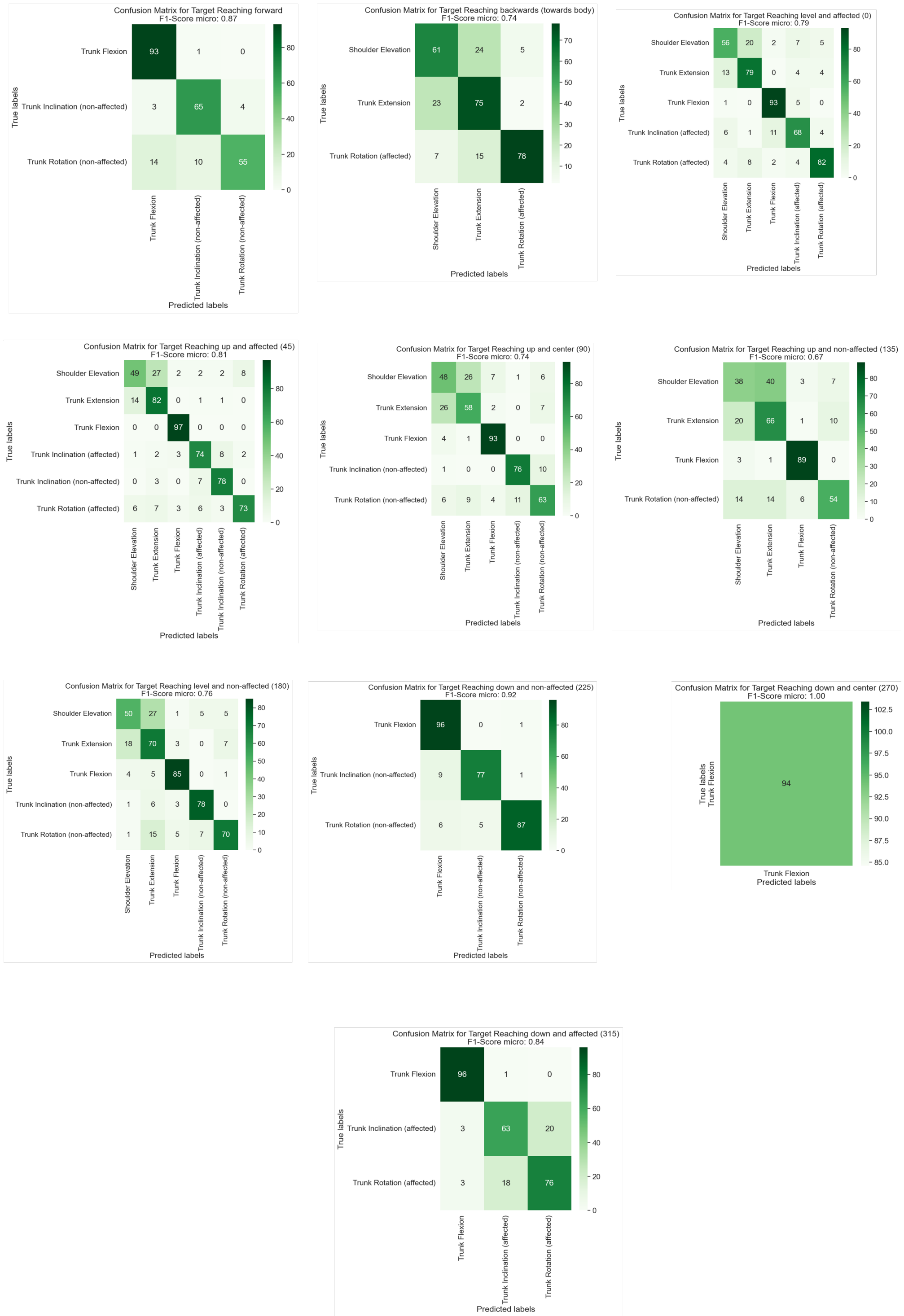

Figure 3.24: Single Compensatory Movements: Confusion Matrix per target in the LOSOxTarget study.

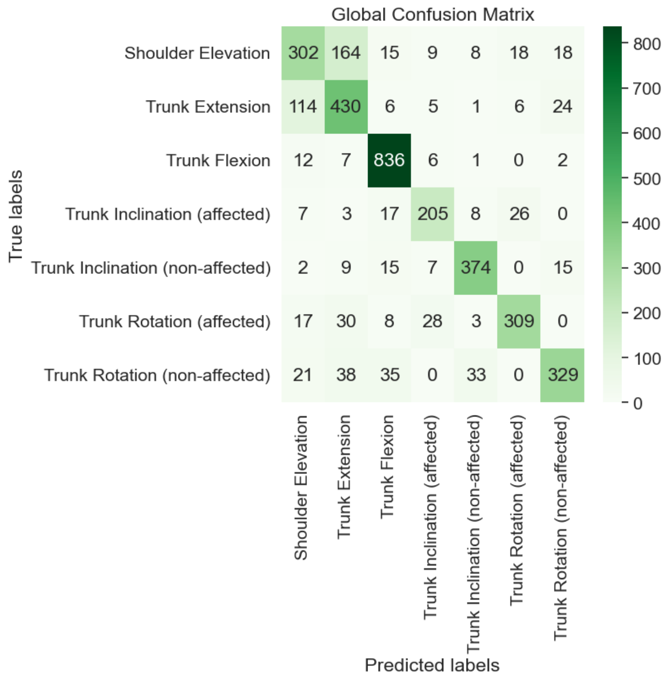

Figure 3.25: Single Compensatory Movements: global Confusion Matrix for LOSOxTarget Classification.

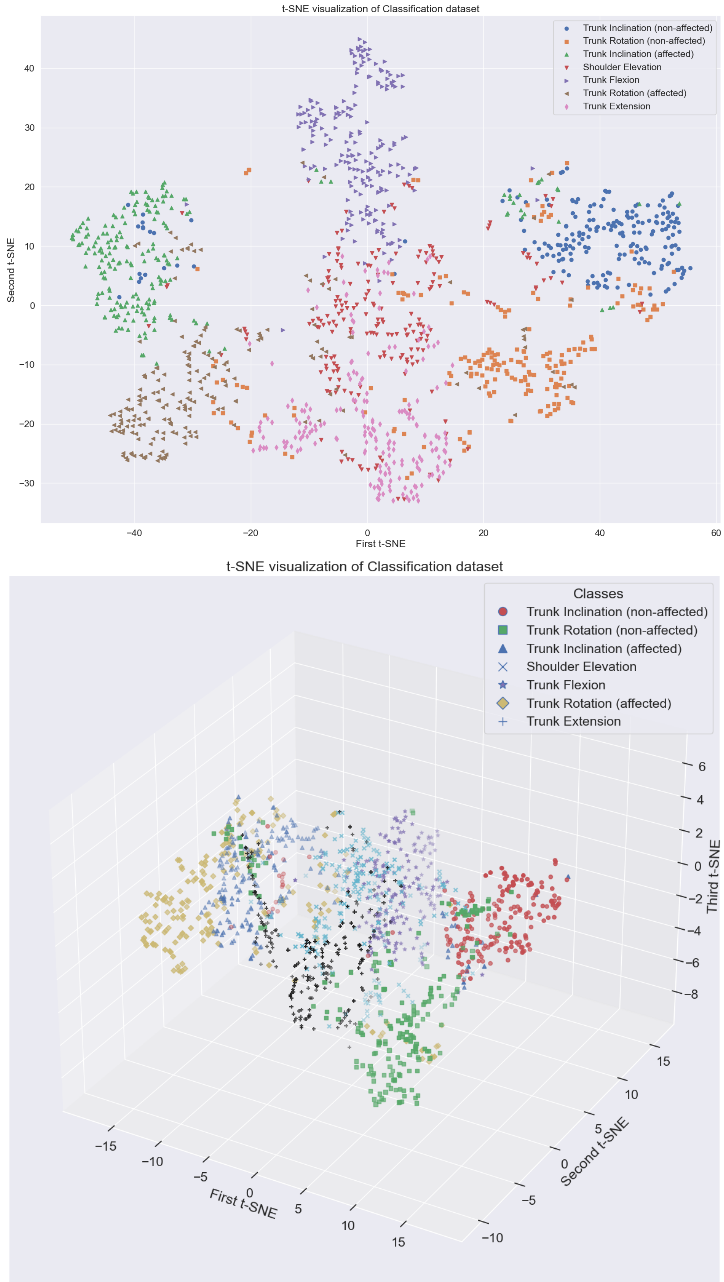

Figure 3.26: Single Compensatory Movements: projections in 2D and 3D using t-SNE.

3.4 Recap of Results

In this section, we present two images summarizing the key results of the study. The first figure below presents a comparison across three scenarios (Figure 3.27): Normative vs Compensatory Movements, Single vs Multiple Compensatory Movements, and Single Compensatory Movements. Overall, the results demonstrate that the model consistently achieves a performance exceeding 80% accuracy across the different scenarios.

The second figure focuses on the F1 micro-scores obtained in the three scenarios using the LOSO (Leave-One-Subject-Out) approach by target (Figure 3.28). The trends observed in the F1 micro-scores closely follow those seen in the accuracy from the first figure, showing a similar pattern of performance across the different scenarios. This consistency reinforces the robustness of the model across varying movement types and target-specific analyses.

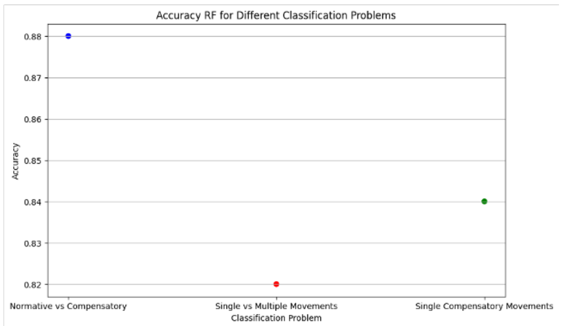

Figure 3.27: RF accuracy for the three scenarios.

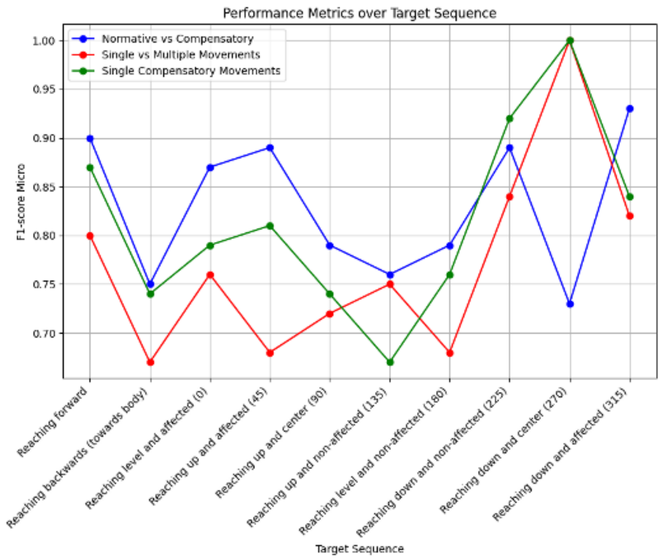

Figure 3.28: F1-micro score LOSOxTarget approach for the three scenarios.

Chapter 4
Discussion

This chapter will outline the preliminary phase that led to the selection of the system used for the project's implementation, specifically the choice of the Lightbuzz platform. This initial phase involved exploring various solutions to determine the most suitable one based on the project's requirements and objectives. After careful consideration, the chosen system was used to conduct the analysis and validate the results.

Following this, the chapter will present final observations regarding the outcomes, highlighting the limitations encountered during the development and testing process, and proposing potential improvements and future steps for the project. In particular, it will address the challenges faced in distinguishing certain movements and the practical implications of these difficulties.

4.1 Preliminary stage Phase II

A preliminary exploration was conducted during Phase II, we conducted a focused evaluation of two primary systems, OpenCap and Lightbuzz, for addressing the research problem of skeleton reconstruction. This comparative exploration allowed us to test both frameworks under the same conditions. Lightbuzz ultimately emerged as the preferred solution based on its overall suitability for our research goals.

OpenCap

OpenCap [37] is a platform designed to assess human movement kinematics and dynamics using videos captured from multiple smartphones.

Before recording begins, camera details are captured using a checkerboard visible to all cameras. Then, a neutral pose recording (Figure 4.1) is done to adjust a musculoskeletal model using OpenSim's scaling tool [57]. Recordings can then begin, using at least two iOS devices (released after 2018). The subject's skeleton is tracked in each video using open-source Human Pose Estimation algorithms like OpenPose [27] and HRNet [28]. After extracting keypoints, OpenCap employs a triangulation process using Direct Linear Transformation. A weighted least-squares approach then adjusts each camera's contribution based on keypoint confidence scores [37]. Since HPE algorithms estimate keypoints frame by frame, this can lead to inconsistencies in the 3D trajectory, which are addressed by training an LSTM network to improve temporal inconsistencies. The 3D kinematics are calculated by applying inverse kinematics to marker trajectories, using a musculoskeletal model with biomechanical constraints [58][59]. This process determines joint angles and movements in 3D space, providing a realistic representation of the subject's movement (Pipeline illustrated in Figure 4.2).

During the initial phase of the project, various camera setups were tested to ensure a robust skeleton representation, while minimizing the number of cameras to avoid bulkiness. The final setup, shown in Figure 4.3, employs three cameras on the subject's unaffected side for OpenCap recordings and one RealSense camera for LightBuzz recordings. Some single compensatory movements frames obtained with OpenCap are shown in Figure 4.4.

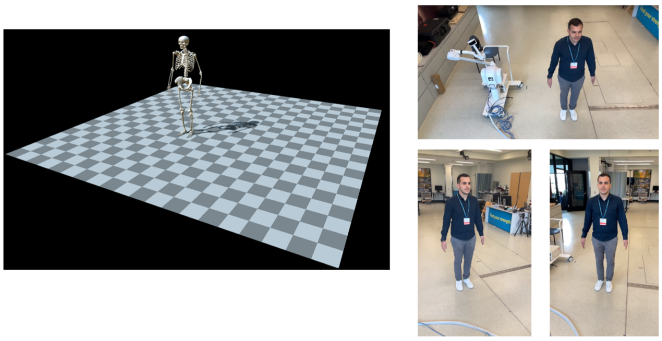

Figure 4.1: Neutral position.

The two frameworks, LightBuzz and OpenCap have been tested in parallel.

Comparison between OpenCap and LightBuzz

Despite their shared goal of capturing 3D skeletons, OpenCap and LightBuzz differ significantly. LightBuzz uses a single RGB-D camera, while OpenCap requires at least two iOS devices, making LightBuzz easier to set up in smaller labs. Additionally, LightBuzz offers real-time detection of compensatory movements,

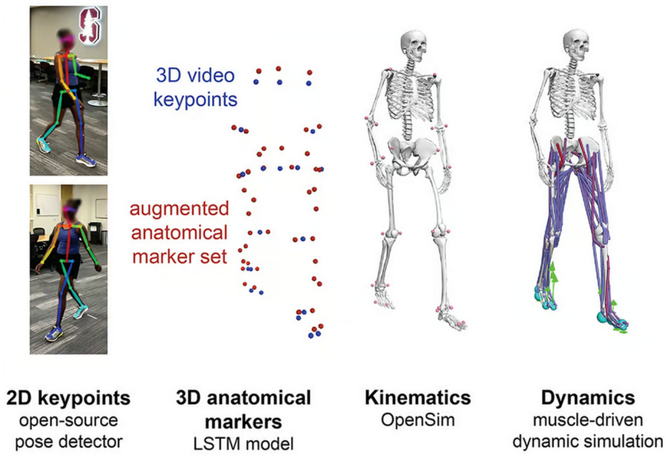

Figure 4.2: OpenCap Pipeline [37].

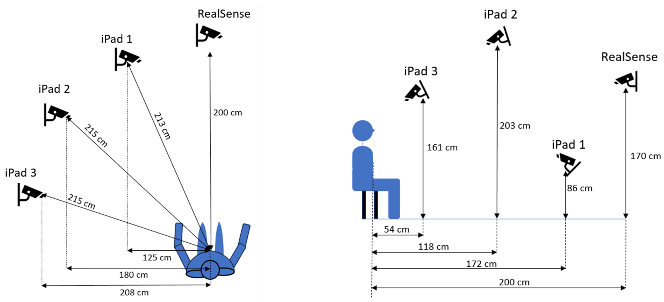

Figure 4.3: Scheme of the final setup.

unlike OpenCap, which requires post-recording processing. However, OpenCap offers greater precision in 3D reconstruction and occlusion handling, thanks to multiple cameras and a downstream biomechanical model. Despite these advantages, OpenCap's requirement for a neutral pose recording can be a limitation, especially for stroke patients unable to stand or walk. Ultimately, the choice between the two systems relied on the specific use case, as each excels in different aspects.

OpenCap addressed several issues, improving certain aspects of the study. However, it also introduced new limitations. One of the main challenges was the requirement for two cameras. This conflicted with the limited space available in the study setup, making implementation more difficult. We decided to leverage LightBuzz's potential, as it is better suited for real-time detection.

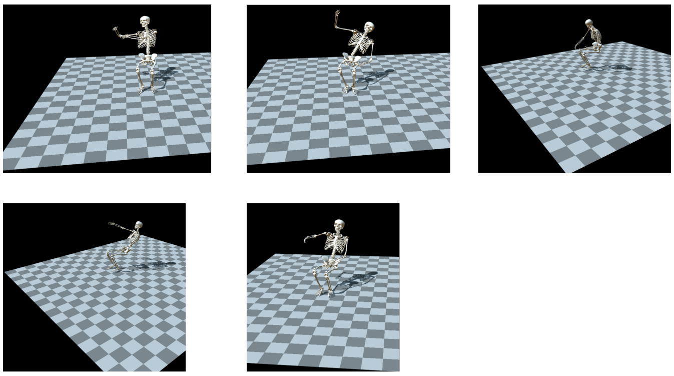

Figure 4.4: Some single compensatory movements frames obtained with OpenCap.

4.2 Limitations and future steps of the study

This study has made significant strides in improving 3D pose estimation techniques and expanding the dataset to enhance the generalization capabilities of the models. A rigorous cross-validation method was applied, exploring both the model's performance when excluding specific subjects and analyzing its behavior across various target positions. However, no fine-tuning of hyperparameters was conducted due to the limited size of the dataset. In general, this technique involves adjusting the parameters of the model, such as learning rate or batch size, to optimize performance and it represents a good practice in the model development [60]. This process would be more appropriate with a larger dataset, where there is a greater range of variability to properly adjust the model parameters.

One limitation of this study is that the models are tested on data generated by therapists, meaning the dataset comprised simulated rather than real-world patient data. The next step is to apply the models to patient data. Preparing this data is a time-intensive process, as therapy sessions are still ongoing, and a selected team of clinicians will need to label the relevant frames for training and evaluation. Two additional challenges arise in this real-world setting: first, the robot may be activated, altering the force that supports the patient, and second, the range of motion (ROM) varies between individuals, particularly in stroke patients. Stroke survivors often experience restricted ROM, which can impact their movement patterns. These factors must be considered to ensure the model's adaptability in clinical applications. A possible next step involves fine-tuning the model with real-world patient data to improve its adaptability and precision [61]. Fine-tuning becomes particularly crucial when initial predictions are inaccurate, allowing the model to be refined for specific patient populations, such as stroke survivors with restricted range of motion (ROM). By calibrating the model to accommodate these movement limitations, it can effectively learn new patterns while still leveraging the general knowledge from earlier stages of training.

Implementing transfer learning can make this process more efficient, enabling the model to incrementally update itself as it encounters fresh data without losing prior knowledge [61]. This approach not only ensures that the model becomes more responsive to patient-specific conditions but also enhances its ability to generalize across diverse cases by dynamically adapting to the constraints of each patient. Ultimately, this refinement makes the model more robust, bridging the gap between simulated data and real-world clinical applications.

One issue that emerged during the analysis was that different compensations can lead to similar biomechanical movements, complicating the classification. Additionally, we aim to implement robot tracking through ArUco markers, which are widely used in computer vision for pose estimation and object tracking [62]. This approach could improve the precision of compensatory movement detection by providing more detailed biomechanical data for analysis.

Another factor to consider is the uneven distribution of compensatory movements across targets. For instance, the model may perform better on well-represented targets (e.g., 0° or 90°) and struggle with less frequent ones (e.g., 270°), potentially skewing precision, recall, and accuracy metrics. While gathering more data for underrepresented targets could address this imbalance, it may not reflect real-world conditions. In clinical practice, certain compensatory movements occur less frequently or are not applicable to specific targets. Thus, we acknowledge this natural variability, ensuring that the dataset accurately represents observed behaviors during rehabilitation. While this may introduce some variability in model performance, it mirrors the variability seen in real-world scenarios.

Despite these challenges, the dataset created for this study is both consistent and extensive. As additional data were collected throughout the study, performance improvements were observed, further validating the model's robustness. If the results from patient testing are promising, the ultimate goal is to develop a real-time system capable of alerting clinicians to the detection of compensatory movements, providing timely feedback during therapy sessions.

Chapter 5
Conclusion

This thesis focused on the detection of compensatory movements in stroke patients during robot-assisted upper limb rehabilitation therapy, with the goal of developing a robust system capable of identifying such movements during rehabilitation exercises. The research began by collecting a comprehensive dataset of both normative and compensatory movements, simulated by therapists and non-therapists across 22 subjects. Simulating movements allowed us to overcome the challenges of directly using patient data, particularly considering the variability in post-stroke recovery. Post-stroke fatigue is a significant challenge for data collection, as it affects patients' physical endurance, preventing them from consistently performing therapy exercises. This fatigue, coupled with varying levels of physical impairment, makes it difficult to gather adequate training data solely from patients [24]. Some patients may even struggle to perform certain exercises altogether. Simulating these motions provided a valuable alternative source of data to train our classifiers, which can later be tested on actual patient data.

The goal of using simulated movements was to create a foundation for real-time detection of compensatory movements in patients, with the ultimate aim of developing a system capable of prompting users to correct their posture when necessary. Such a system has the potential to significantly improve rehabilitation outcomes by reducing harmful compensatory behaviors early in the recovery process.

The next stage involved extracting skeletal data using the Lightbuzz framework, followed by normalizing the data, removing outliers, extracting and selecting features, and preparing the dataset for model development. The classification problem was approached from three perspectives: (1) Normative vs Compensatory Movements, (2) Single vs Multiple Compensatory Movements, (3) Single Compensatory Movements.

In all three scenarios, the Random Forest classifier with Stratified K-Fold cross-validation achieved an accuracy exceeding 80%, demonstrating the model's potential for reliably identifying compensatory movements. While there are certainly areas for improvement, the results are promising, paving the way for future tests on real patient data.

In conclusion, this study lays the groundwork for a system that can be tested with patient data and, if successful, could be developed into a real-time solution for detecting compensatory movements during therapy. This system would alert patients and therapists to postural issues, helping to optimize recovery and improve the overall effectiveness of rehabilitation exercises.

Bibliography

[1] Valery L. Feigin et al. «Global, regional, and national burden of stroke and its risk factors, 1990–2019: A systematic analysis for the Global Burden of Disease Study 2019». In: *The Lancet Neurology* 20.10 (2021), pp. 793–820. DOI: 10.1016/S1474-4422(21)00252-0 (cit. on p. 3).

[2] H. Nakayama, H. Stig Jørgensen, H. Otto Raaschou, and T. Skyhøj Olsen. «Recovery of upper extremity function in stroke patients: The Copenhagen stroke study». In: *Archives of Physical Medicine and Rehabilitation* 75 (1994), pp. 394–398 (cit. on p. 3).

[3] Daniel O'Flaherty and Khalid Ali. «Recommendations for Upper Limb Motor Recovery: An Overview of the UK and European Rehabilitation after Stroke Guidelines (2023)». In: *Healthcare* 12.14 (2024), p. 1433. DOI: 10.3390/healthcare12141433. URL: https://doi.org/10.3390/healthcare12141433 (cit. on pp. 3, 4).

[4] C. Hui, P. Tadi, and L. Patti. *Ischemic Stroke*. Available online: https://www.ncbi.nlm.nih.gov/books/NBK499997/. Treasure Island, FL, USA: StatPearls Publishing, 2021 (cit. on p. 3).

[5] L Huang, L Yi, H Huang, S Zhan, R Chen, and Z Yue. «Corticospinal tract: a new hope for the treatment of post-stroke spasticity». In: *Acta Neurol Belg* 124.1 (Feb. 2024). Epub 2023 Sep 13, pp. 25–36. DOI: 10.1007/s13760-023-02377-w (cit. on p. 3).

[6] TE Nogles and MA Galuska. «Middle Cerebral Artery Stroke». In: *StatPearls [Internet]* (Apr. 2023). Treasure Island (FL): StatPearls Publishing; 2024 Jan–. PMID: 32310592 (cit. on p. 3).

[7] National Institute for Health and Care Excellence. *Stroke Rehabilitation in Adults*. Available online: https://www.nice.org.uk/guidance/ng236. 2023. URL: https://www.nice.org.uk/guidance/ng236 (cit. on p. 4).

[8] J. H. Carr and R. B. Shepherd. *A motor relearning programme for stroke*. Rockville, MD, USA: Aspen Publishers, 1987 (cit. on p. 4).

[9] B. Bobath. *Adult hemiplegia: evaluation and treatment.* London, UK: Heinemann Medical Books, 1990 (cit. on p. 4).

[10] M. C. Cirstea and M. F. Levin. «Compensatory strategies for reaching in stroke». In: *Brain* 123.5 (2000), pp. 940–953. DOI: `10.1093/brain/123.5.940` (cit. on p. 4).

[11] Association of American Medical Colleges. *The Complexities of Physician Supply and Demand: Projections from 2013 to 2025.* 2015. URL: `https://www.aamc.org/download/426242/data/` (cit. on p. 4).

[12] S. E. Fasoli, H. I. Krebs, and N. Hogan. «Robotic technology and stroke rehabilitation: translating research into practice». In: *Topics in Stroke Rehabilitation* 11.4 (2004), pp. 11–19 (cit. on p. 4).

[13] Naoyuki Takeuchi and Shin-Ichi Izumi. «Maladaptive Plasticity for Motor Recovery after Stroke: Mechanisms and Approaches». In: *Neural Plasticity* 2012 (2012), pp. 1–9. DOI: `10.1155/2012/359728`. URL: `https://doi.org/10.1155/2012/359728` (cit. on pp. 5, 6).

[14] J Netz, T Lammers, and V Hömberg. «Reorganization of motor output in the non-affected hemisphere after stroke». In: *Brain* 120.Pt 9 (Sept. 1997), pp. 1579–1586. DOI: `10.1093/brain/120.9.1579` (cit. on p. 5).

[15] N Murase, J Duque, R Mazzocchio, and LG Cohen. «Influence of interhemispheric interactions on motor function in chronic stroke». In: *Ann Neurol* 55.3 (Mar. 2004), pp. 400–409. DOI: `10.1002/ana.10848` (cit. on p. 5).

[16] R. Hong, B. Li, Y. Bao, L. Liu, and L. Jin. «Therapeutic robots for post-stroke rehabilitation». In: *Med Rev* 4.1 (Feb. 2024), pp. 55–67. DOI: `10.1515/mr-2023-0054` (cit. on p. 6).

[17] Albert C. Lo, Peter D. Guarino, Linda G. Richards, Jon K. Haselkorn, Glenn F. Wittenberg, David G. Federman, and Hermano I. Krebs. «Robot-assisted therapy for long-term upper-limb impairment after stroke». In: *New England Journal of Medicine* 362.19 (2010), pp. 1772–1783. DOI: `10.1056/NEJMoa0911341`. URL: `https://doi.org/10.1056/NEJMoa0911341` (cit. on p. 6).

[18] Shaohua Cai, Xiang Wei, Enhong Su, Yingyi Ye, and Ying Zhang. «Online compensation detecting for real-time reduction of compensatory motions during reaching: a pilot study with stroke survivors». In: *Journal of NeuroEngineering and Rehabilitation* 17.58 (2020), pp. 1–12. DOI: `10.1186/s12984-020-00687-1`. URL: `https://doi.org/10.1186/s12984-020-00687-1` (cit. on pp. 6, 12).

[19] Xinyi Wang, Yitong Fu, Bichen Ye, Jonathan Babineau, Ying Ding, and Alex Mihailidis. «Technology-Based Compensation Assessment and Detection of Upper Extremity Activities of Stroke Survivors: Systematic Review». In: *Journal of Medical Internet Research* 24.6 (2022), e34307. DOI: 10.2196/34307 (cit. on pp. 6, 7).

[20] Asaad Sellmann, Désirée Wagner, Lucas Holtz, Jörg Eschweiler, Christian Diers, Sybele Williams, and Catherine Disselhorst-Klug. «Detection of Typical Compensatory Movements during Autonomously Performed Exercises Preventing Low Back Pain (LBP)». In: *Sensors* 22.1 (2022), p. 111 (cit. on p. 7).

[21] Jeffrey Laut, Maurizio Porfiri, and Preeti Raghavan. «The Present and Future of Robotic Technology in Rehabilitation». In: *Current Physical Medicine and Rehabilitation Reports* 4.4 (2016), pp. 312–319. DOI: 10.1007/s40141-016-0139-0. URL: https://doi.org/10.1007/s40141-016-0139-0 (cit. on p. 7).

[22] S. Cai, G. Li, S. Huang, H. Zheng, and L. Xie. «Automatic Detection of Compensatory Movement Patterns by a Pressure Distribution Mattress Using Machine Learning Methods: A Pilot Study». In: *IEEE Access* 7 (2019), pp. 80300–80309. DOI: 10.1109/ACCESS.2019.2923077 (cit. on p. 7).

[23] Yan Fu, Xiaoyi Wang, Zeqiang Zhu, Jie Tan, Yan Zhao, Yong Ding, and Weiya Chen. «Vision-based Automatic Detection of Compensatory Postures of after-Stroke Patients During Upper-extremity Robot-assisted Rehabilitation: A Pilot Study in Reaching Movement». In: *2020 International Conference on Assistive and Rehabilitation Technologies (iCareTech)*. 2020, pp. 62–66. DOI: 10.1109/iCareTech49914.2020.00019 (cit. on pp. 8, 18).

[24] Ying Xuan Zhi, Michelle Lukasik, Michael H. Li, Elham Dolatabadi, Rosalie H. Wang, and Babak Taati. «Automatic Detection of Compensation During Robotic Stroke Rehabilitation Therapy». In: *IEEE Journal of Translational Engineering in Health and Medicine* (2017), pp. 1–7. DOI: 10.1109/JTEHM.2017.2782741 (cit. on pp. 10, 24, 73).

[25] Qi Dang, Jianqin Yin, Bin Wang, and Wenqing Zheng. «Deep Learning Based 2D Human Pose Estimation: A Survey». In: *Tsinghua Science and Technology* 24.6 (2019), pp. 663–676. DOI: 10.26599/TST.2018.9010100. URL: https://doi.org/10.26599/TST.2018.9010100 (cit. on pp. 10–13).

[26] Alexander Toshev and Christian Szegedy. «DeepPose: Human Pose Estimation via Deep Neural Networks». In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2014, pp. 1653–1660. DOI: 10.1109/CVPR.2014.214 (cit. on pp. 11, 14).

[27] Zhe Cao, Tomas Simon, Shih-En Wei, and Yaser Sheikh. «Realtime Multi-Person 2D Pose Estimation using Part Affinity Fields». In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2017, pp. 7291–7299. DOI: 10.1109/CVPR.2017.143 (cit. on pp. 11, 14, 69).

[28] Ke Sun, Bin Xiao, Dong Liu, and Jingdong Wang. «Deep High-Resolution Representation Learning for Human Pose Estimation». In: *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 2019, pp. 5686–5696. DOI: 10.1109/CVPR.2019.00584 (cit. on pp. 12–14, 69).

[29] Leonid Pishchulin, Eldar Insafutdinov, Siyu Tang, Björn Andres, Mykhaylo Andriluka, Peter V Gehler, and Bernt Schiele. «Deepcut: Joint subset partition and labeling for multi person pose estimation». In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. IEEE. Las Vegas, NV, USA, 2016, pp. 4929–4937 (cit. on p. 12).

[30] Eldar Insafutdinov, Leonid Pishchulin, Björn Andres, Mykhaylo Andriluka, and Bernt Schiele. «Deepercut: A deeper, stronger, and faster multi-person pose estimation model». In: *European Conference on Computer Vision*. Springer. 2016, pp. 34–50 (cit. on p. 12).

[31] Kai Lu and Dugki Min. «AiPE: A Novel Transformer-Based Pose Estimation Method». In: *Electronics* 13.5 (2024), p. 967. DOI: 10.3390/electronics13050967 (cit. on p. 12).

[32] Ce Zheng, Sijie Zhu, Matias Mendieta, Taojiannan Yang, Chen Chen, and Zhengming Ding. «3D Human Pose Estimation with Spatial and Temporal Transformers». In: *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*. 2021, pp. 11656–11665. DOI: 10.1109/ICCV48922.2021.01145 (cit. on p. 12).

[33] Andrei Zanfir, Elisabeta Marinoiu, and Cristian Sminchisescu. «Monocular 3D Pose and Shape Estimation of Multiple People in Natural Scenes: The Importance of Multiple Scene Constraints». In: *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition* (2018), pp. 2148–2157. URL: https://api.semanticscholar.org/CorpusID:52846817 (cit. on p. 13).

[34] Jinrui Zhang, Deyu Zhang, Xiaohui Xu, Fucheng Jia, Yunxin Liu, Xuanzhe Liu, Ju Ren, and Yaoxue Zhang. «MobiPose: real-time multi-person pose estimation on mobile devices». In: *Proceedings of the 18th Conference on Embedded Networked Sensor Systems*. SenSys '20. Virtual Event, Japan: Association for Computing Machinery, 2020, pp. 136–149. ISBN: 9781450375900. DOI: 10.1145/3384419.3430726. URL: https://doi.org/10.1145/3384419.3430726 (cit. on p. 13).

[35] Angjoo Kanazawa, Michael J Black, David W Jacobs, and Jitendra Malik. «End-to-end recovery of human shape and pose». In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2018, pp. 7122–7131 (cit. on p. 14).

[36] Julieta Martinez, Rayat Hossain, Javier Romero, and James J Little. «A simple yet effective baseline for 3d human pose estimation». In: *Proceedings of the IEEE International Conference on Computer Vision*. 2017, pp. 2640–2649 (cit. on p. 14).

[37] Scott D. Uhlrich, Antoine Falisse, Łukasz Kidziński, Julie Muccini, Michael Ko, Akshay S. Chaudhari, Jennifer L. Hicks, and Scott L. Delp. «OpenCap: Human movement dynamics from smartphone videos». In: *PLOS Computational Biology* 19.10 (Oct. 2023). cit. on pp. 14, 48, pp. 1–26. DOI: `10.1371/journal.pcbi.1011462`. URL: `https://doi.org/10.1371/journal.pcbi.1011462` (cit. on pp. 14, 68–70).

[38] Hongwen Tu, Chunyu Wang, and Wenjun Zeng. «VoxelPose: Towards Multi-camera 3D Human Pose Estimation in Wild Environment». In: *European Conference on Computer Vision (ECCV)*. Springer-Verlag, 2020, pp. 197–212. DOI: `10.1007/978-3-030-58452-8_12` (cit. on p. 14).

[39] Iuri de Medeiros Esper, Oleg Smolkin, Mikhail Manko, Alexander Popov, Pål Johan From, and Andrew Mason. «Evaluation of RGB-D Multi-Camera Pose Estimation for 3D Reconstruction». In: *Applied Sciences* 12.9 (2022), p. 4134. DOI: `10.3390/app12094134` (cit. on p. 14).

[40] William Bennardo. «Posture Check: Detecting and Evaluating Compensatory Movements in Robot-based Stroke Rehabilitation through Human Pose Estimation and Machine Learning». Available at: `https://webthesis.biblio.polito.it/29949/`. MA thesis. Politecnico di Torino, 2023 (cit. on p. 14).

[41] LightBuzz. *LightBuzz SDK: Body Tracking Software*. `https://lightbuzz.com/`. Accessed: September 26, 2024 (cit. on p. 20).

[42] Barrett Technology. *Barrett Technology - Robotic Rehabilitation for Upper Limbs*. `https://barrett.com/`. Accessed: 2024-09-27. 2024 (cit. on p. 22).

[43] Douglas C Montgomery, Elizabeth A Peck, and G Geoffrey Vining. *Introduction to Linear Regression Analysis*. John Wiley & Sons, 2006 (cit. on p. 31).

[44] Gilbert Strang. *Introduction to Linear Algebra*. Wellesley-Cambridge Press, 2009 (cit. on pp. 32, 33).

[45] Jason Brownlee. *Feature Selection with Real and Categorical Data*. `https://machinelearningmastery.com/feature-selection-with-real-and-categorical-data/`. 2019 (cit. on pp. 36, 37).

[46] Max Kuhn and Kjell Johnson. *Applied Predictive Modeling*. Springer, 2013 (cit. on p. 36).

[47] The pandas development team. *pandas-dev/pandas: Pandas*. Version latest. Feb. 2020. DOI: 10.5281/zenodo.3509134. URL: https://doi.org/10.5281/zenodo.3509134 (cit. on p. 37).

[48] Wes McKinney. «Data Structures for Statistical Computing in Python». In: *Proceedings of the 9th Python in Science Conference*. Ed. by Stéfan van der Walt and Jarrod Millman. 2010, pp. 56–61. DOI: 10.25080/Majora-92bf1922-00a (cit. on p. 37).

[49] Haibo He and Yunqian Ma. *Imbalanced Learning: Foundations, Algorithms, and Applications*. John Wiley & Sons, 2013 (cit. on p. 39).

[50] F. Pedregosa et al. «Scikit-learn: Machine Learning in Python». In: *Journal of Machine Learning Research* 12 (2011), pp. 2825–2830 (cit. on pp. 40, 44).

[51] Daniele Micci-Barreca. «A preprocessing scheme for high-cardinality categorical attributes in classification and prediction problems». In: *SIGKDD Explor. Newsl.* 3.1 (July 2001), pp. 27–32 (cit. on p. 41).

[52] F. Pargent, F. Pfisterer, and J. et al. Thomas. «Regularized target encoding outperforms traditional methods in supervised machine learning with high cardinality features». In: *Comput Stat* 37 (2022), pp. 2671–2692 (cit. on p. 41).

[53] Leo Breiman. «Random Forests». In: *Machine Learning* 45.1 (2001), pp. 5–32. DOI: 10.1023/A:1010933404324. URL: https://doi.org/10.1023/A:1010933404324 (cit. on pp. 42, 44).

[54] Ron Kohavi. «A Study of Cross-Validation and Bootstrap for Accuracy Estimation and Model Selection». In: *Proceedings of the 14th International Joint Conference on Artificial Intelligence (IJCAI)*. 1995, pp. 1137–1143. URL: https://dl.acm.org/doi/10.5555/1643031.1643047 (cit. on p. 42).

[55] Christopher M. Bishop. *Pattern recognition and machine learning*. Springer, 2006 (cit. on p. 44).

[56] L.J.P. van der Maaten and G.E. Hinton. «Visualizing high-dimensional data using t-SNE». In: *Journal of Machine Learning Research* 9 (2008), pp. 2579–2605 (cit. on p. 62).

[57] Scott L. Delp, Frank C. Anderson, Allison S. Arnold, Peter Loan, Ayman Habib, Chand T. John, Eran Guendelman, and Darryl G. Thelen. «OpenSim: Open-source software to create and analyze dynamic simulations of movement». In: *IEEE Transactions on Biomedical Engineering* 54.11 (2007). cit. on p. 48, pp. 1940–1950. DOI: 10.1109/TBME.2007.901024 (cit. on p. 69).

[58] Apoorva Rajagopal, Christopher L. Dembia, Matthew S. DeMers, Denny D. Delp, Jennifer L. Hicks, and Scott L. Delp. «Full-Body Musculoskeletal Model for Muscle-Driven Simulation of Human Gait». In: *IEEE Transactions on Biomedical Engineering* 63.10 (2016). cit. on p. 49, pp. 2068–2079. DOI: `10.1109/TBME.2016.2586891` (cit. on p. 69).

[59] Adrian K. M. Lai, Allison S. Arnold, and James M. Wakeling. «Why are antagonist muscles co-activated in my simulation? A musculoskeletal model for analysing human locomotor tasks». In: *Annals of Biomedical Engineering* 45 (2017). cit. on p. 49, pp. 2762–2774 (cit. on p. 69).

[60] Ian Goodfellow, Yoshua Bengio, and Aaron Courville. *Deep Learning*. MIT Press, 2016 (cit. on p. 71).

[61] Jason Yosinski, Jeff Clune, Yoshua Bengio, and Hod Lipson. «How transferable are features in deep neural networks?» In: *Advances in neural information processing systems*. 2014, pp. 3320–3328 (cit. on pp. 71, 72).

[62] Sergio Garrido-Jurado, Rafael Muñoz-Salinas, Francisco J Madrid-Cuevas, and Rafael Medina-Carnicer. «Automatic generation and detection of highly reliable fiducial markers under occlusion». In: *Pattern Recognition* 47.6 (2016), pp. 2280–2292 (cit. on p. 72).