Heliyon 11 (2025) e42119

Contents lists available at ScienceDirect

Heliyon

journal homepage: www.cell.com/heliyon

Research article

Cloud and IoT based smart agent-driven simulation of human gait for detecting muscles disorder

Sina Saadati *, Abdolah Sepahvand, Mohammadreza Razzazi

Department of Computer Engineering, Amirkabir University of Technology (Tehran Polytechnic), Tehran, Iran

ARTICLE INFO

Keywords:
Agent-based modeling and simulation
Cloud and IoT based simulation
Human gait
Agent-driven simulation
Muscles disorder

ABSTRACT

Motion disorders affect a significant portion of the global population. While some symptoms can be managed with medications, these treatments often impact all muscles uniformly, not just the affected ones, leading to potential side effects including involuntary movements, confusion, and decreased short-term memory. Currently, there is no dedicated application for differentiating healthy muscles from abnormal ones. Existing analysis applications, designed for other purposes, often lack essential software engineering features such as a user-friendly interface, infrastructure independence, usability and learning ability, cloud computing capabilities, and AI-based assistance. This research proposes a computer-based methodology to analyze human motion and differentiate between healthy and unhealthy muscles. First, an IoT-based approach is proposed to digitize human motion using smartphones instead of hardly accessible wearable sensors and markers. The motion data is then simulated to analyze the neuromusculoskeletal system. An agent-driven modeling method ensures the naturalness, accuracy, and interpretability of the simulation, incorporating neuromuscular details such as Henneman's size principle, action potentials, motor units, and biomechanical principles. The results are then provided to medical and clinical experts to aid in differentiating between healthy and unhealthy muscles and for further investigation. Additionally, a deep learning-based ensemble framework is proposed to assist in the analysis of the simulation results, offering both accuracy and interpretability. A user-friendly graphical interface enhances the application's usability. Being fully cloud-based, the application is infrastructure-independent and can be accessed on smartphones, PCs, and other devices without installation. This strategy not only addresses the current challenges in treating motion disorders but also paves the way for other clinical simulations by considering both scientific and computational requirements.

1. Introduction

Movement is an essential factor in everybody's life and affects diverse aspects of people. However, a significant proportion of people are suffering from motion disorders affecting over a billion patients worldwide [1]. Thus, controlling of motion disorders is an important field in science, technology, and clinical studies [2–4]. Motion analysis plays a vital role in detecting motion disorders and in their subsequent treatment and control [4–6].

Motion is a complex phenomenon in the human body which originates from the contribution of three compound systems including

* Corresponding author.
E-mail addresses: sina.saadati@aut.ac.ir (S. Saadati), a.sepahvand@aut.ac.ir (A. Sepahvand), razzazi@aut.ac.ir (M. Razzazi).

https://doi.org/10.1016/j.heliyon.2025.e42119
Received 25 September 2024; Received in revised form 17 January 2025; Accepted 18 January 2025
Available online 20 January 2025

neural, muscular, and skeletal known as the neuromusculoskeletal system [7]. This process begins from the neural system in which motion commands are sent to the muscles through motor neurons [7–10]. Muscles generate a contraction force as a reaction to neural stimulations [5–10]. The muscular force is then applied to the skeletal system which behaves as a mechanical lever and, finally, movement happens [5,6]. Any failure in each segment of the neuromusculoskeletal system causes a motion disorder [7]. For instance, common disorders such as multiple sclerosis (MS) or Parkinson disease are arising from of neural failures in which muscles cannot be stimulated correctly [10–12].

Medications, including chemical drugs, are widely used to treat or control the symptoms of disorders [13–15]. Considering the disadvantages of chemical medications and their side effects, a critical issue in treatment procedures is that while only a small proportion of muscles are affected by disorders, all muscles are uniformly influenced by medications, even the muscles of the eyes [14–17]. These adverse effects include involuntary movements, confusion, hallucinations, decreased short-term memory, blurred vision, and muscle stiffness. Understanding these side effects is crucial for optimizing treatment strategies and improving patient outcomes [16–18]. Consequently, it is necessary to differentiate between healthy and unhealthy muscles in the treatment of motion disorders.

Differentiating between healthy and unhealthy muscles presents several challenges, as the failure of one muscle can significantly alter the behavior of other healthy muscles, potentially leading to misleading conclusions [4,6]. Therefore, this research focuses on patients in the early stages of motion disorders, where the behavior of their healthy muscles is not substantially affected by the failing muscles. The objective of this study is to propose a computer-based method to analyze patients with non-acute motion disorders, assisting clinical experts in distinguishing between healthy and unhealthy muscles and providing valuable insights to them for accurate diagnosis. Therefore, using the proposed method, experts can conduct further investigations on patients to gather more detailed information, ultimately aiding in making a final decision. Although, this research lays the groundwork for automating other clinical processes and developing more comprehensive analytical applications.

The proposed motion-analyzing application encompasses critical procedures, including the signal-capturing phase, signal-processing phase, and result presentation to the user, each with specific requirements. In the first phase, it is essential that the captured signals are accurate. Additionally, the signal-capturing method should be user-friendly, utilizing common devices such as smartphones instead of hard-to-access markers. Therefore, this research proposes an IoT-based method to capture motion signals, ensuring both accuracy and ease of use. Despite some research [19–25] that have used hardly available sensors, the proposed IoT-based method for digitizing motion enhances system usability by requiring only smartphones. Unlike markers and insole-based devices, smartphones are widely available globally, including in remote and underserved areas. This widespread availability significantly enhances the usability of the proposed IoT-based approach. This approach eliminates the need for additional sensors, which can be economically expensive or unavailable in many regions.

In the second phase, it is essential that signal processing is interpretable. Therefore, this research employs a simulation-based method to process the captured signals instead of black box-like deep learning models with ambiguous functioning. It is crucial for the simulation to be natural, meaning that the algorithm executed within the computer must originate from natural phenomena that are under investigation [26–30]. In the simulation of human motion, achieving naturalness in modeling requires a deep representation of neurophysiological and biomechanical details [31–33]. For this purpose, it is recommended to use an agent-based method for modeling and simulation to enhance the naturalness, accuracy, and interpretability of the simulation [26–29,34,35]. Consequently, the causes of any phenomena can be precisely tracked and explained [36,37]. Thus, an agent-driven approach is adopted to model and simulate the neuromusculoskeletal system of the patient during gait, ensuring that scientific details are accurately reflected in the simulation.

To incorporate neurophysiological details, this research employs a motor unit (MU)-based approach to simulate human voluntary muscles. Motor units, the smallest units of movement in the human body, consist of a single neuron and all the muscle fibers it innervates [4–12]. Despite some studies such as [38–44] that model muscles as atomic, indivisible modules, this approach demonstrates that including neurophysiological details such as MUs enhances the accuracy and interpretability of the simulation [45]. Consequently, this research conducts MU-based modeling of the human neuromuscular system, enabling an in-depth analysis of both motion and the neuromuscular system. For example, it is possible to analyze muscles by distinguishing their motor units or by examining the action potentials (APs) sent to each MU. This level of detail is unattainable in electromyography-based simulations, as electromyography signals represent an average weighted sum of numerous action potentials at a specific muscle location and are different from the neural messages in human body [5–12,46].

In this research, the Henneman's size principle is also considered which provides detailed information about the MU recruitment within the muscles. This principle explains how different types of MU are stimulated to perform a motion [4–6,10,47]. Thus, if motor units are not considered as the constituent elements of muscles, a significant portion of the details related to muscle's physiological behavior will be omitted from the modeling process, making it impossible to analyze movement at the level of individual MUs. In addition to distinguishing muscles by their constituent MUs, it is required that the physiological functions within these MUs be accurately modeled to maintain naturalness of the simulation. This requires detailed modeling of all stages of a MU's contraction, from receiving the neural signal to producing mechanical force which includes the pattern of increasing and decreasing contraction force, the effect of increased neural stimulation frequency on contraction force, and the state of tetanus (when the stimulation frequency is intense and the MU produces maximum possible force). Many muscular phenomena, such as the manner of increasing or decreasing contraction intensity, or muscle weakening in cold environments or because of continuous activity, occur due to physiological changes within MUs [5,6,10]. In this research, the physiological details within MUs are considered to ensure that the proposed simulation is scientifically detailed and suitable for other research purposes.

The naturalness of the muscular modeling can also enhance the flexibility of the model. For example, some research [43,44,48–50] has proposed different models for different muscles, often limited by environmental conditions such as the gender of the patients. Despite the differences between various types of voluntary muscles, they all share the same physiological structure [4–6,10]. Therefore, by employing a natural MU-based simulation, different voluntary muscles can be analyzed using a single proposed model.

The link between the neuromuscular system and motion signals is the skeleton, which functions as a complex mechanical lever [5,6]. In the case of lower body movement during gait, which is investigated in this research, it is necessary to convert motion-captured signals into the contraction forces of each muscle group during the gait cycle, a challenging process. This conversion requires solving a system of mathematical equations at each time step of the gait cycle, where joints form the equations and muscle contraction forces are the variables. Since the number of variables exceeds the number of equations in leg modeling, mathematical optimization methods are needed to solve these systems, which can lead to inaccuracies [3,51,52]. In this research, gait is considered as a 2D movement in the sagittal plane, significantly simplifying the biomechanical model by minimizing the need for complex geometric modeling of joints and bones. This simplification is scheduled for further investigation in future works. Additionally, a task-based approach is followed, where each motion task in the joints forms one equation. This results in more equations and, consequently, more accurate results.

Using an agent-driven model of the neuromusculoskeletal system, designed for reverse engineering analysis in this research, gait can be simulated. This allows for the computation of contraction forces of MUs and the action potentials sent to each MU during the gait. At the end of the second phase, the simulation outputs should be analyzed by experts. However, this analysis can also be performed by a computer using a deep learning-based method. Deep learning, a field within artificial intelligence, is widely used in medical signal processing research due to its ability to learn complex patterns more effectively than other machine learning methods [53–58]. This research proposes an ensemble deep learning method composed of artificial neural networks to analyze the simulation outputs and assist clinical experts. For this purpose, the deep learning model must be both accurate and interpretable, meaning that the analysis procedures within the model should be as explainable as possible. Therefore, this research develops different artificial neural network models for different joints, rather than a single complex model that processes all simulation outputs, thereby maximizing interpretability because it is shown that ensemble neural networks with simpler inputs can promote the interpretability [59].

The final phase is another crucial step in the proposed methodology, where the results are provided to the users because software rich in computational facilities is not valuable if users can not work with it [60–62]. Therefore, it is crucial that the analyzing application features a user-friendly graphical interface. The software should be easy to install, learn, use, and maintain. Despite some simulator such as OpenSim, this research proposes a fully cloud-based application, making the software infrastructure independent and easily executable on any internet browser, whether on a PC or a smart mobile device. The proposed cloud-based approach also eliminates the need for installation or maintenance by the users. Unlike some simulation applications that are cluttered with confusing menus, involute windows, and buttons, this research minimizes the number of graphical components to ensure the application is convenient to learn and use. This feature is important, especially for clinical users who are working in stressful conditions. Additionally, as cloud computing is recommended in green computing, the proposed approach offers environmental benefits.

Thus, the contribution of this study is as follows.

- **IoT method for motion signals capturing:** A methodology is proposed in which motion signals can be captured using sensors of mobile devices easily. This approach significantly enhances the usability of the proposed method for movement modeling compared to methods requiring markers or wearable sensors.
- **Agent-driven biomechanical model of the human lower body:** An agent-based model is proposed in which the muscular forces can be calculated from angular displacements of lower body joints during a gait. Lower body joints are modeled as agents to maintain the naturalness of the simulation.
- **Agent-driven model of human voluntary muscles:** An agent-base model of human voluntary muscle is proposed which can be used inside of the proposed biomechanical model. Neural stimulations of each muscle can be calculated by this model. In this paper, neural messages are modeled as APs, and muscles are modeled as a collection of independent MUs. This approach, while maintaining the naturalness of the modeling, enables specialists to analyze movement at the deep level of individual MUs.
- **Deep learning framework for signal processing:** An ensemble neural network framework is proposed in order to analyze the motion of lower body joints and detect if the gait is normal or pathological. In this paper, biomechanical, muscular, and neural patterns during the gait are independently examined by three separate neural network models. This approach, as opposed to using a single complex neural network with ambiguous performance, will bring greater reliability and interpretability to the results.
- **Design and development of user-friendly cloud-based application:** An open-source application is developed that can capture motion signals through a mobile device and analyze it. The application is infrastructure-independent as can be executed on a wide range of computers.

Therefore, we can articulate our research questions as follows.

- **RQ1. Signal Capturing:** Is the proposed IoT-based approach for motion signal capturing accurate? Can this method serve as a viable alternative to markers or other wearable sensors? Can the proposed method extract the patterns accurately? Are the outputs reliable for research or clinical investigations? Is the proposed application infrastructure independent and easy to use?
- **RQ2. Modeling and Simulation:** Is the proposed agent-driven simulation natural? Are biomechanical and neurophysiological details accurately reflected in the proposed modeling? Are the simulation outputs reliable for research and clinical investigations? Can the proposed modeling approach be used to differentiate healthy muscles from abnormal muscular organs in patients with non-acute motion disorders?

- **RQ3. Deep learning models:** Are deep learning-based models utilized to analyze the simulation outputs to assist clinical experts? Is the proposed method interpretable? Are the predictions as accurate and reliable as required for clinical use?
- **RQ4. Software Requirements:** Is the designed graphical interface user-friendly and easy to learn? Are all processes, including the IoT-based signal capturing program, agent-driven simulation, and neural network calculations, executed fully on the cloud to ensure the proposed application is infrastructure-independent and easy to run?

In Section 2, preliminaries are explained which are the basis of our proposed method. Related works are also surveyed briefly in this section. In Section 3, our methodology for analysis of human motion during gait is detailed. In Section 4, our proposed method is analyzed in order to investigate its precision, reliability, and interpretability. In Section 5, the discussion and our future works are described.

2. Literature review

In this section, the preliminaries of our proposed method are described. In Section 2.1, the agent-driven method for modeling and simulation is described. Then, the physiology and biology of the human voluntary muscles as the main motors of the motion are described in Section 2.2. Section 2.3, describes how artificial intelligence models and software techniques can promote analyzer applications. Finally, some related works are investigated in Section 2.4.

2.1. Agent-based modeling and simulation (ABMS)

In many modeling and simulation processes, the visual representation on the computer may closely resemble real-world phenomena. However, there is often a significant disparity between the algorithms driving the simulation and the actual processes occurring in nature [26–28]. The Agent-Based Modeling and Simulation methodology is a modern modeling approach that ensures the naturalness, interpretability, and accuracy of the simulations [27,28]. Unlike conventional methods, agent-based simulation elements are autonomous components known as agents and have an internal decision-making strategy. Various types of agents can be defined in a simulation, and some can even form the structure of others. Additionally, agents interact with each other and their environment [27–29].

Fig. 1 illustrates the difference between agent-driven and traditional methods for modeling a humanized society, with lines representing data streams. In the real world, agents (people, as illustrated in Fig. 1) are independent and completely autonomous. They can make decisions autonomously, using their brains, which are located within the agent's structure. Since each component accesses only a limited proportion of information, they require communication through various channels (such as talking in the real world) to transfer information to one another and complement their knowledge. In contrast, traditional simulations often feature a central decision-making component that controls all the components from outside their structure. This central system is aware of all the necessary information, thereby minimizing or eliminating the need for data transfer between components. In the agent-driven methodology, each agent possesses an autonomous decision-making system within its structure. Each agent can interact with other agents and has its own knowledge base. Consequently, communication is essential for agents to transfer information and enhance their knowledge. The decision-making components are represented by small processors in Fig. 1. The autonomy of agents enhances the naturalness of the simulation, making it more comparable to real-world phenomena. In clinical and medical investigations, where the interpretability of simulation phenomena is critical, an agent-driven approach is recommended [26–28,34,35].

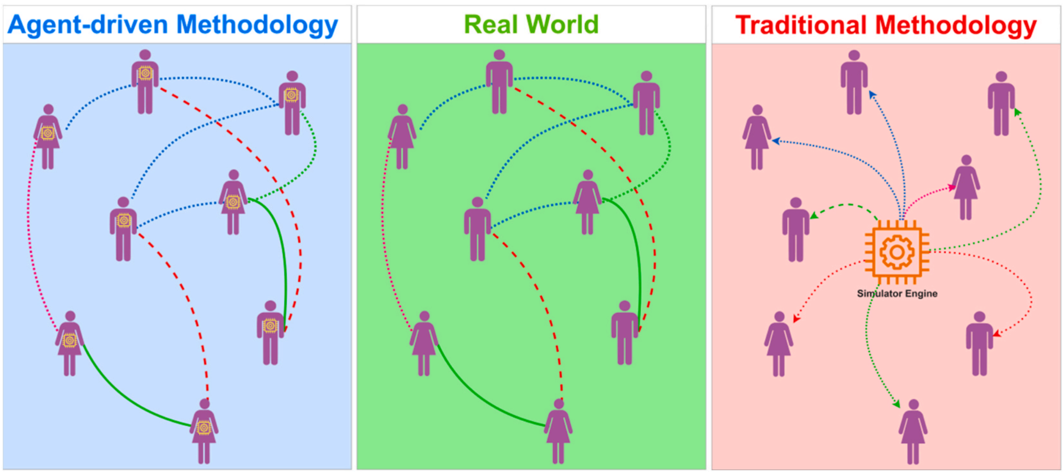


Fig. 1. Comparison between agent-driven and traditional modeling of human society.

2.2. Physiology and biology of muscles and motor control

The motion begins from the neural system in which movement commands are sent to the muscles. The motion commands are electrochemical interactions, known as APs, that are moved forward along the motor neurons [10,63,64]. Each motor neuron is connected to some muscle fibers which are contracted as a reaction to the APs they receive. The complex of one motor neuron and all the muscle fibers connecting to it form a motor unit, which is known as atomic elements in the motion. Muscle fibers are biological cells that contain nuclei, mitochondria, and a number of myofibrils covered by sarcoplasmic reticulum [10,65]. When an AP is transferred to one muscle fibers, a chemical interactions happen which turn the chemical energy of Adenosine Triphosphate (ATP) into mechanical energy [5,6,10,63]. Finally, a physical contraction force is generated and applied to the skeletal system which behaves as a complex mechanical lever [5,6]. Thus, the overall system which controls the motion is neuromusculoskeletal system.

2.3. Artificial intelligence and software requirements

Machine learning is a subfield of artificial intelligence that computers extract knowledge from data [53,55]. Artificial Neural Networks (ANNs) are a subfield of machine learning models that are implemented based on the learning process inside the human brain. ANNs are computationally powerful as they are able to realize complex patterns in compound data [56–58,66,67]. Thus, they are widely used in medical and clinical applications to assist physicians in order to simplify the medical processes, enhance accuracy, and reduce costs [68–71]. ANNs are layered networks of perceptrons which map the input data to the output results. In this architecture, the perceptron is a mathematical model of a single human neuron and plays the main role in the learning of the models [56,57,67].

By increasing the number of perceptrons, the network can realize more complex patterns from data. Thus, this field of computing is also referred to as deep learning [57,58]. However, the computations of ANNs with a huge number of parameters are hard to be explained or interpreted [36,72–76]. Nevertheless, the challenge can be remedied by providing interpretable methods in which ensemble ANN models are developed instead of one compound and ambiguous model [59,72]. In an ensemble model, multiple ANNs are used, each focusing on one aspect of the phenomenon. Following this method, not only interpretability is enhanced, but also the risk of noise detection is minimized as all the ANN models learn equal patterns while learning different noises. This feature is known as cross-training [77,78].

While ANNs are very powerful and advantageous in learning compound patterns, they are complex concepts in computer science and cannot be used by inexpert people. Thus, software engineering techniques are used to cover the complexity of computations and make intelligent applications transparent and useable for users. As a result, it is important that artificial intelligence software be equipped to a user-friendly graphical interface [60–62,79,80].

To develop a user-friendly graphical interface, several factors should be considered. It is important to minimize the number of graphical components, such as menus, buttons, and windows, as they can be confusing for users. Additionally, advanced settings related to networking, sensors, and calculations should be managed automatically to facilitate procedures and increase transparency. It is also recommended to use conceptual images instead of complex texts or diagrams, as they are easier to understand.

The Internet of Things (IoT) is a software technique where various devices, particularly intelligent mobiles rich in sensors, connect to form a powerful computing system with enhanced capabilities. By utilizing the sensors connected to the IoT system, the network can obtain accurate environmental signals. Additionally, since intelligent mobiles are widespread globally, it is feasible to implement an IoT-based system even in regions with minimal facilities, such as some rural areas.

In addition to the user-friendly interface, modern applications should be infrastructure-independent so that they can be easily executed on a wide range of computers, with no difference between a PC or mobile device, and without any compound installation procedure. Therefore, cloud-based applications are getting popular nowadays as they provide infrastructure-independent computations and are also available, and economically effective [81–85]. IoT is another software technology which can promote the power of applications. Mobile devices are significantly considered in IoT methods owing to their precise sensors and computing powers [81]. Hence, applications can be more useful, available, and cost-effective if they connect to mobile devices and use their functionalities [81–87]. We have utilized IoT and cloud computing to enable users to employ the proposed methodology from any location. Unlike wearable sensors, which may not be available even in some metropolises such as Isfahan or Tehran, our proposed IoT method for motion signal capturing is accessible worldwide, including in rural areas with minimal technological infrastructure. The fully cloud-based platform allows users to execute our application without the need for installation, regardless of their hardware or software configurations. Consequently, even residents of rural areas can access our advanced analysis application using a smartphone.

2.4. Related works

The simulation of human muscles and motion analysis of motion represents a captivating area of study and research. Within this section, we conduct a brief comparative analysis of related studies, examining their limitations from simulation, artificial intelligence, and software engineering points of view. Our comparison is detailed in Table 1.

OpenSim and AnyBody are powerful tools built for the investigation of the biomechanical behavior of human motion. They enable experts to analyze different muscles within different motions [19–21]. However, the software engineering requirements such as user-friendly interface or infrastructure-independency are not considered in those works making them challenging for users.

A cloud-based method is proposed by Ref. [22] to analyze and monitor the patient's motions in which neural networks are utilized. Nevertheless, as a subset of signal processing operations are executed in the local computer, the method is not fully cloud-based and, as

Table 1
Comparison of the human motion analysis methods, assuming full satisfactory by ✓, partial satisfactory by F0A8, and failing to satisfactory by × .

Metrics		[19]	[21]	[22]	[23]	[24]	[25]	[38]	[39]	[40]	[41]	[42]	[43]	[44]
Differentiation between healthy and unhealthy muscles		×	×	×	×	×	×	×	×	×	×	×	×	×
Natural Modeling and Simulation	Agent-driven method for modeling	×	×	×	×	×	×	×	×	×	×	×	×	×
	AP-based simulation of neuromuscular system	×	×	×	×	×	×	×	×	×	×	×	×	×
	Physiological simulation of the muscle	✓	×	×	×	×	×	×	×	F0A8	×	F0A8	×	F0A8
	Ability to simulate muscular environmental conditions	✓	✓	×	×	×	×	×	×	×	×	×	×	×
	Flexibility of the model	F0A8	F0A8	×	×	×	×	×	×	×	✓	✓	×	×
Software Requirements	User-friendly graphical interface	×	×	×	×	×	×	✓	×	×	×	×	×	×
	Cloud-based platform	×	×	F0A8	×	×	×	×	×	×	×	×	×	×
	Hardware and software infrastructure independency	×	×	×	×	×	×	×	×	×	×	×	×	×
	IoT-based approach	×	×	×	×	×	×	F0A8	×	×	×	×	×	×
Artificial Intelligence	Pathological gait detection	F0A8	×	✓	✓	✓	✓	×	×	×	×	×	×	×
	Interpretability of the method	×	×	×	×	×	×	F0A8	×	×	×	×	F0A8	×
	Cross-training learning	×	×	×	×	×	×	×	×	×	×	×	×	×

a consequence, not infrastructure-independent.

Machine learning models are proposed by Refs. [23–25] in which motion signals are captured from insole-based methods or mounted accelerometers and then, abnormal movements are detected. Nevertheless, the models are not interpretable as physiological and biological aspects of motion are not considered. Also, the signal analysis is conducted using a complex machine learning model, where the mapping of input signals to output results remains ambiguous. Research has demonstrated that dividing a complex problem into smaller subproblems and developing specialized neural networks for each subproblem significantly enhances the interpretability of the entire artificial intelligence process. This approach allows for more explainable models, as each neural network can be tailored to address specific aspects of the problem, thereby improving overall transparency and understanding [59,88]. Additionally, signal capturing could be facilitated by an IoT-based approach.

An application is proposed by Ref. [38] in order to detect motor impairment severity in patients suffering from Parkinson's disease. Data collection is done by the screen of mobile. However, the method is restricted to be used only on iPhone mobiles. The machine learning models are not interpretable. Other artificial intelligence methods for the analysis of the gait are proposed by Refs. [39,40] that are limited in the classification of gait phases. There is no proposed abnormality detection in these methods. In addition, data are received through insoles or electrogoniometers which are not accessible devices in many regions.

From a flexibility point of view, the models proposed by Refs. [41,42] can be used for a wide range of muscles and animals. However, interpretability is not considered as the biology of the neuromuscular system is not modeled. EMG signals are analyzed in proposed methods by Refs. [43,44]. Although, the algorithms are not natural as APs are the real neural stimulations inside the human body which are far from what can be seen in electromyograms [8,46]. Therefore, they have used signal processing techniques for EMG signals which breaches the principle of naturalness of the modeling. The methods are also restricted to analyzing the motions of only a single joint. The Naturalness of simulation also enables modelers to simulate environmental conditions such as temperature or genetics. This is because the effect of these factors is justified by their influence on the biological features of the muscles [5,6,10].

3. Methodology

In this section, our methodology for gait analysis is detailed. An overview of the proposed method is demonstrated in Fig. 2. Motion signals are first captured by sensors of mobile devices (Step 1). Then, muscular forces are calculated (Step 2). Then, the APs entered into each MU can be computed by the muscle agents (Step 3). Finally, simulation results are analyzed by deep-learning models (Step 4) and displayed to the users by a user-friendly interface (Step 5).

Section 3.1 details the motion signals capturing method. The agent-driven biomechanical model of the human lower body is described in Section 3.2. The agent-based model of human voluntary muscle is explained in Section 3.3 as well as deep learning methods for analysis of results that are detailed in Section 3.4.

3.1. Proposed IoT-based method to capture the motion signals

Due to the complexity, expensiveness, and not availability of wearable sensors, an IoT method is proposed in this study to capture human motion. Mobile devices contain a 3D compass that can detect the orientation of the device on the ground by providing three values for the angles of the device relative to the three axes of rotation. The compass is sensitive to small motions of the device and is widely used in applications such as games, traveling, astronomy, artificial reality, and augmented reality.

In this study, a web-based application is developed which can connect to the 3D compass of the mobile and capture the motions during gait. Since the program is cloud-based, it is infrastructure-independent and can be executed on any type of smartphone. The

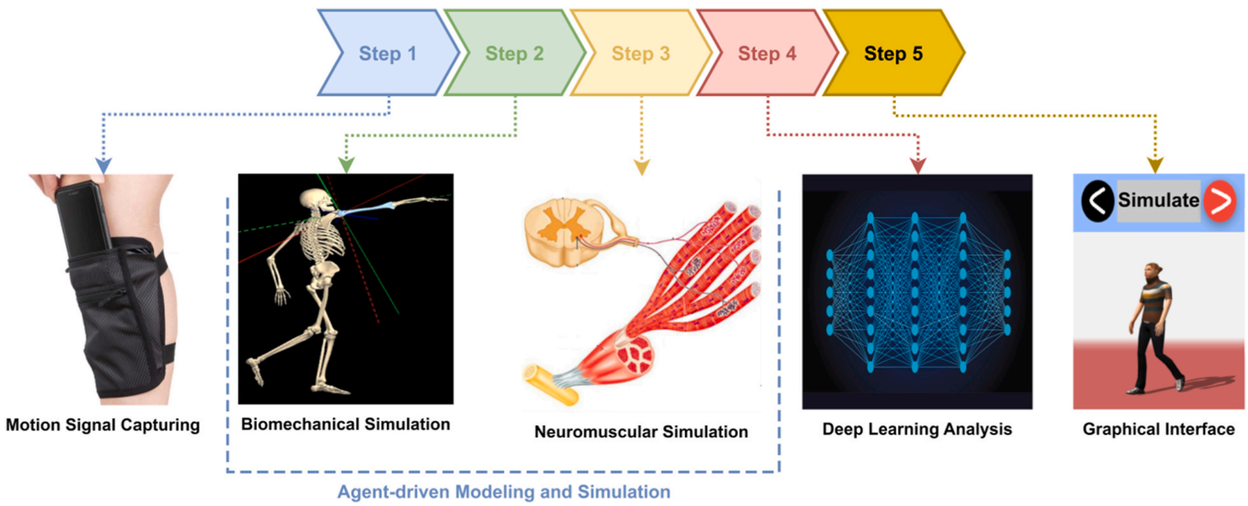


Fig. 2. Overview of the proposed methodology for gait analysis.

application also provides a user-friendly graphical interface which can be even used by inexpert users. Thus, a step-by-step process is designed in which a short clip is presented to the user at each step, demonstrating how to perform specific actions (e.g., how and where to attach the smartphones for each joint). This strategy minimizes user confusion and, consequently, enhances the software's ease of learning and use.

First, the patient is instructed to attach the mobile devices to specific positions on their body. This task is explained to the user during three steps for hip, knee, and ankle joints. The placement of these devices is crucial, as certain positions can yield more accurate data. For instance, the lower anterior position of the leg (just above the ankle joint) is ideal for capturing motion signals from the leg. This location minimizes noise from soft tissues such as muscles and skin due to its proximity to the bone. Once the devices are attached, the patient is required to stand in the anatomical position in the fourth step. Anatomical position is a especial style of standing with the back and hands against a straight wall. This step allows the system to update its initial state and align itself with the patient's anatomical position. By accurately determining the patient's body position at the start of the gait and capturing the orientations of the mobile devices, it is possible to calculate the position of each body part and, consequently, the angles of the lower body joints during gait. Following this setup, the patient is asked to walk for about seven steps in the fifth step to capture the motion signals. This step includes playing a warning sound to inform the patient to start walking, and another warning sound to notify them to stop.

To facilitate the digitization of the patient's motion, all smartphones are attached to his body simultaneously, and he is then asked to walk only once. Since the smartphones are lightweight, the gait will not be significantly affected. The parallel design of the IoT system which is demonstrated in Fig. 3 ensures that these signals are seamlessly merged and integrated. Our findings indicate that delays between the mobile devices and the cloud server are negligible, as the delays tend to be consistent across all devices due to their equal geographic distance from the server computer. Additionally, since data are sent to the cloud server, the simulation can be continued on any computer that can connect to the user's cloud account.

Following the capturing phase, the raw data during the walking is processed. The raw data is a matrix of angles during the movement. First, noisy values in the matrix are detected and removed. Then, a mathematical regression model is used to summarize all the steps into a single gait cycle. For this purpose, the data from all the passed steps are divided into individual steps. This can be achieved by identifying the maximum and minimum values or by using a predefined angle threshold. Subsequently, the angle values from all the steps are plotted on a time-based diagram. A regression model is then employed to summarize the data from all the steps and to derive an equation that describes the angular changes of the organ during gait. Additionally, some features such as minimum and maximum angle values, number of steps, stance/swing time proportion, speed, and time per step are calculated which are valuable in subsequent analysis. The output of each regression model which is a matrix that represents the angles of the lower body joints during the gait cycle, is used for the agent-driven simulation of the motion. In addition, feature extraction output provides additional and valuable information for further deep learning analysis. The process is illustrated in Fig. 4.

3.2. Proposed agent-driven biomechanical model of lower body

To accurately analyze human motion throughout a gait cycle, it is essential to employ a biomechanical model in order to convert angular displacements of the joints into muscular contraction forces. Thus, this section proposes an agent-driven biomechanical model for the lower body, governed by the proposed Boots algorithm. Fig. 5 presents our innovative agent-driven model of the human lower body. Muscle agents that are assembled in this model are described in Section 3.3.

The proposed agent-driven model for the human lower body encompasses three joint agents and six muscle group agents: the Triceps Surae, Tibialis Anterior, Hamstrings, Quadriceps, Gluteus Maximus, and Iliopsoas. These are considered as the ankle dorsiflexor, ankle plantar flexor, both knee flexor and hip extensor, both knee extensor and hip flexor, hip extensor, and hip flexor, respectively. The Triceps Surae also functions as a knee flexor in some cases, as it crosses the knee joint. However, in this research, the gait cycle is investigated for a patient with non-acute motion disorder, where the knee joint remains straightened during gait. Consequently, while the angle between the Triceps Surae muscle and the foot is vertical, maximizing its effective torque, the angle

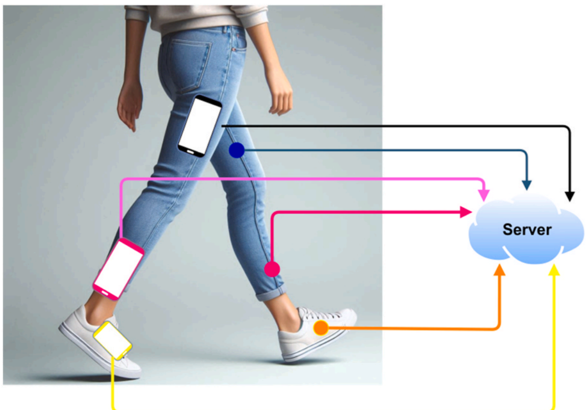


Fig. 3. Mobiles attached to the lower body of a patient.

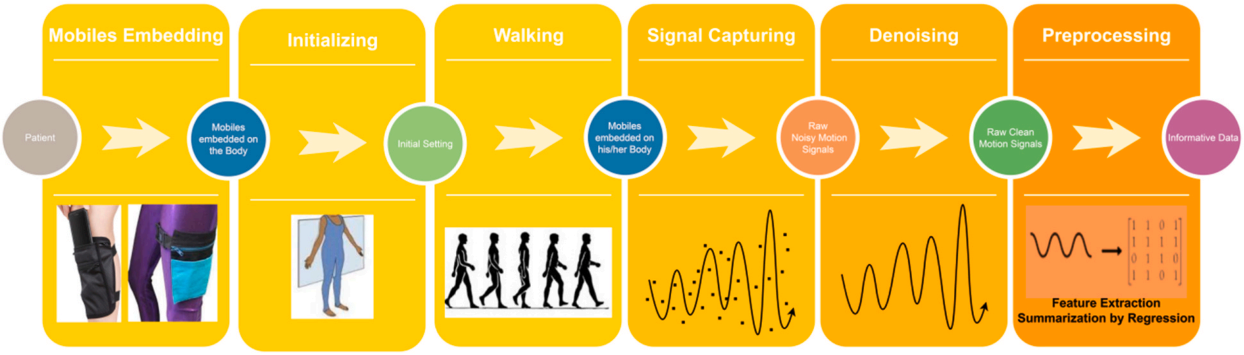


Fig. 4. Proposed IoT-based method for motion signal capturing.

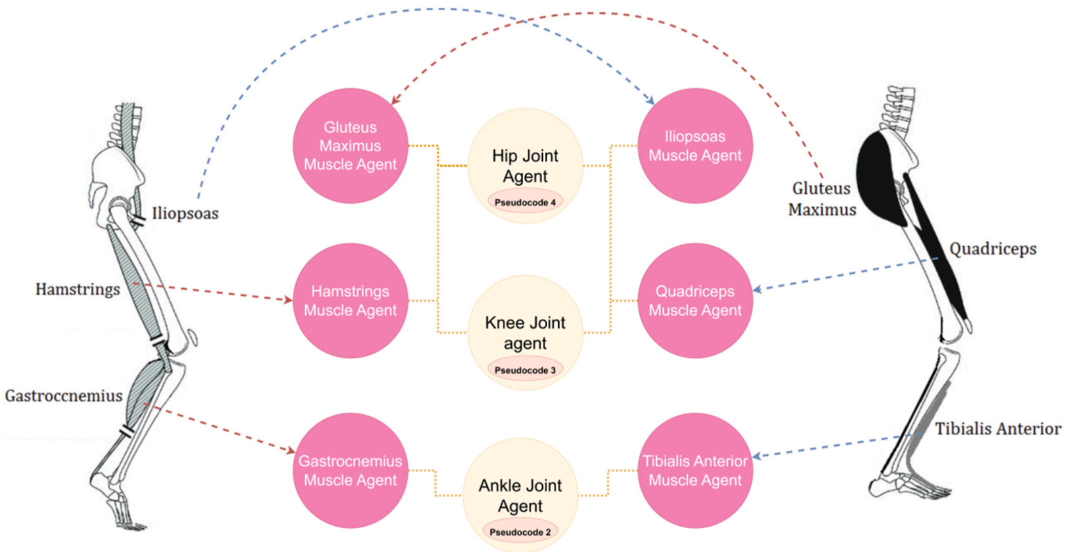


Fig. 5. Our proposed agent-driven biomechanical model for lower body.

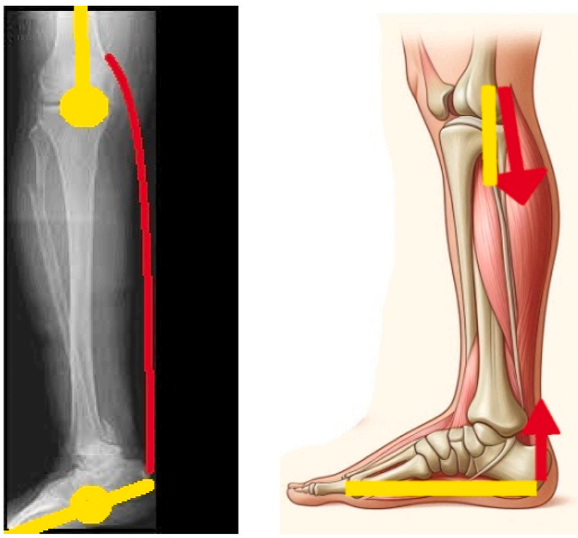


Fig. 6. Influence of Triceps Surae muscle on the ankle joint compared to the knee joint.

between the muscle and the hip bone is approximately zero, minimizing its effective torque. Thus, in this paper, the Triceps Surae is considered only as an ankle dorsiflexor. This case is illustrated in Fig. 6 in which the Triceps Surae and bones are represented by red and yellow lines, respectively. This assumption simplifies the biomechanical model by reducing the number of equations in the optimization system, resulting in more accurate outcomes. To investigate the influence of the Triceps Surae muscle on the knee joint, the patient can be asked to perform additional exercises beyond a single gait cycle. For example, flexing the knee while sitting on a chair can be easily analyzed using a simple linear equation system, without the need for optimization methods.

In the proposed model, gait is considered as a 2D movement occurring in the sagittal plane. This assumption significantly simplifies the biomechanical model by eliminating the need for 3D geometric modeling of the lower body joints, calculation of torques caused by ligaments, and the complexity of a 3D biomechanical model of the human lower body skeleton. Our findings indicate that although most gait abnormalities lead to three-dimensional motion, some of them can still be effectively detected when the motion is reduced to a two-dimensional space. This conversion significantly simplifies the biomechanical calculations without eliminating the informative data necessary for detection.

Additionally, the intelligent motor control of the body by the neural system, which optimizes energy consumption, is considered in converting captured motion signals to contraction forces. For instance, when the body intends to flex the knee without hip extension, the quadriceps muscle (a knee flexor) will not be activated, as it would counteract the effect of the hamstrings muscle (the primary knee flexor). The same principle applies to knee extension without hip flexion. This consideration leads to a task-based analysis of the muscular system during gait, where each motion task is defined by an equation and the number of equations rises. Consequently, the system of equations can be solved more easily and with greater accuracy.

To accurately characterize the biomechanical behavior of our agent-based model, we introduce the Boots algorithm. The anno-

Table 2
Definition of annotations in the Boots algorithm.

Feature	Equation Statement	Derivative	Details
Mass	$M^{(O)}$	$M^{(body)}$	Mass of the patient
		$M^{(feet)}$	$0.0145 \times M^{(body)}$ [5]
		$M^{(leg)}$	$0.0465 \times M^{(body)}$ [5]
		$M^{(thigh)}$	$0.100 \times M^{(body)}$ [5]
Weight	$W_{(t)}^{(J)}$	$W_{(t)}^{(ankle)}$	The felt weight in the ankle joint.
		$W_{(t)}^{(knee)}$	The felt weight in the knee joint.
		$W_{(t)}^{(hip)}$	The felt weight in the hip joint.
Length	$L^{(O)}$	$L^{(feet)}$	Distance from ankle joint to toe.
		$L^{(leg)}$	Distance from ankle to knee.
		$L^{(thigh)}$	Distance from knee to hip joint.
Moment of Inertia	$I^{(O)}$	$I^{(ankle)}$	$M^{(feet)} \times 0.62^2 \times {L^{(feet)}}^2$ [5]
		$I^{(knee)}$	$M^{(leg)} \times 0.528^2 \times {L^{(leg)}}^2$ [5]
		$I^{(thigh)}$	$M^{(thigh)} \times 0.540^2 \times {L^{(thigh)}}^2$ [5]
Angular Acceleration	$\alpha_{(t)}^{(J)}$	$\alpha_{(t)}^{(ankle)}$	$\alpha_{(t)} = \frac{d}{dt} v_{(t)} = \frac{d^2}{dt^2}\,\theta(t)$ [5]
			$\alpha_{(t)} = \frac{\Delta}{\Delta t} v_{(t)} = \frac{\Delta^2}{\Delta t^2}\,\theta(t)$ [5]
Ground Reaction Force	$F_{ground}^{(P)}(t)$	$F_{ground}^{(toe)}(t)$	Computable using $d_{(bodyCentre,toe)}^{(horizon)}$
		$F_{ground}^{(heel)}(t)$	$W_{(t)}^{(ankle)} - F_{ground}^{(toe)}(t)$
Angle	$\theta_{(H)}^{(O)}(t)$	$\theta_{(H)}^{(feet)}(t)$	The horizontal angles for the lines connecting the ankle to the toe, the ankle to the knee, and the knee to the hip joints, respectively.
		$\theta_{(H)}^{(leg)}(t)$	
		$\theta_{(H)}^{(thigh)}(t)$	
	$\theta_{(V)}^{(O)}(t)$	$\theta_{(V)}^{(feet)}(t)$	The vertical angles for the lines connecting the ankle to the toe, the ankle to the knee, and the knee to the hip joints, respectively.
		$\theta_{(V)}^{(leg)}(t)$	
		$\theta_{(V)}^{(thigh)}(t)$	
Distance	$d_{(P_a,\,P_b)}^{(H/V)}(t)$	$d_{(bodyCentre,toe)}^{(H)}(t)$	The horizontal distance between mass centre of the body and toe of the feet.
		$d_{(bodyCentre,hip)}^{(H)}(t)$	The horizontal distance between mass centre of the body and hip joint.
	$d_{(P_a,\,P_b)}$	$d_{(ankle,toe)}$	The distance from ankle to toe.
		$d_{(heel,ankle)}$	The distance from heel to ankle.
		$d_{(ankle,feetCentre)}$	The direct distance from mass centre of the feet to the ankle.
		$d_{(ankle,knee)}$	The distance from knee to ankle.
		$d_{(knee,legCentre)}$	The distance from the centre of the leg and knee.
Torque	$\tau_{(t)}^{(J)}$	$\tau_{(t)}^{(ankle)}$	$I^{(ankle)} \times \alpha_{(t)}^{(ankle)}$
		$\tau_{(t)}^{(knee)}$	$I^{(knee)} \times \alpha_{(t)}^{(knee)}$
		$\tau_{(t)}^{(hip)}$	$I^{(hip)} \times \alpha_{(t)}^{(hip)}$

tations for the algorithm are provided in Table 2 in which O, J, t, H, V, and P represent organ and joint of interest, the current time, horizontal or vertical direction, and an anatomic point, respectively. Annotations listed in Table 2 can be defined as variables or matrix during the development of the application. For example, each annotation that varies over time can be defined as an array of values representing a matrix.

To facilitate the measurement of annotations by the user, this research requires only four annotations to be measured: $L^{(feet)}$, $L^{(leg)}$, $L^{(thigh)}$, and the mass of the patient ($M^{(body)}$). To maintain the accuracy of the analysis, the user measures these four features through a step-by-step process. In each step, a short clip (less than 30 s) is presented to the user, explaining how to measure the feature (e.g., $L^{(leg)}$ that needs to be measured from one special point on the ankle to another special point on the knee). The user then measures the feature and enters the value before proceeding to the next step. This strategy is necessary due to specific requirements for the annotations. For instance, $L^{(thigh)}$ must be delineated by measuring two predefined anatomical positions on the body. Other annotations can be measured automatically using biomechanical equations. For example, $W^{(knee)}_{(t)}$ can be calculated at each time step using the proposed agent-driven biomechanical model, as all the dependent features can be calculated.

Ground reaction force, a crucial factor in our biomechanical model, depends on the phase of the gait cycle and the mass center of the body. During the stance phase, when one foot is on the ground and the other leg is in the swing phase, the ground reaction force corresponds to the reaction of the ground to the patient's body weight, making it relatively straightforward to calculate. Conversely, during the swing phase, when the leg is not in contact with the ground, the ground reaction force is zero. However, during the transition phases between stance and swing, when one leg is being placed on the ground and the other foot is lifting off, calculating the exact value of the ground reaction force becomes challenging [5,6]. In this research, biomechanical model is used to calculate the ground reaction force during the gait. However, we use approximate values for these transition phases to avoid the need for advanced sensors that are not easily accessible. Because of the brief duration of these transitions, the use of approximate values does not compromise the overall accuracy of the simulation. Our findings indicate that this approach does not alter the patterns that need to be analyzed by experts. More details for calculating the ground reaction force is provided in Section 3.2.1.

The Boots algorithm calculates the contraction force generated by each muscle group during the gait cycle. This calculation is achieved by determining the total torque resulting from the joint's angular acceleration. Then, the algorithm differentiates the torque generated by the muscle of interest from other factors, which are defined as environmental torques in this study.

Boots algorithm (Algorithm 1) is the key functionality of joint agents, which takes three inputs: the agent of the joint in question, the agent of the muscle group whose contraction force is being calculated, and the environmental torques present at each step of the gait cycle, represented as a 2D array. Environmental torques encompass all torques caused by forces other than those from the target muscle group, which include the weight of body parts, ground reaction forces, and forces from non-target muscles.

Boots algorithm initiates by creating a zero-filled list (line 2). A 'For' loop then processes each gait cycle time step (lines 3–12), calculating the joint's total torque from its angular acceleration at that time step (line 4). The environmental torques are then deducted to isolate the muscle's contraction torque (lines 5–7). The algorithm concludes by computing the contraction force from the torque, recording the outcome (lines 8–11).

Given that muscles can only contract, a negative torque value calculated in Line 10 suggests interference from other muscles. Therefore, the contraction force for the muscle of interest is set to zero using the *ReLU* function.

Algorithm 1. Calculating Muscular Contraction forces using mechanical torques.

	Boots Algorithm (Joint, MuscleOfInterest, EnvironmentalTorques [ ] [ ])
1	Input: Agent of Joint, Agent of muscle group of interest, and list of all the environmental torques.
	Output: Diagram of contraction force generated by group muscle of interest.
	Begin
2	Results = Array with zero values.
3	**For** *time* **in** Gait Cycle Diagram
	Begin
4	TotalTorque = $\tau^{(J)}_{(t)}$
5	**For** *i* **from** 0 **to** EnvironmentalTorques[*t*].length( )
6	TotalTorque − = EnvironmentalTorques [*t*] [*i*]
7	θ = the angle between MuscleOfInterest contraction force and the bone.
8	*Dist* = The distance between Joint and the ligament of the MuscleOfInterest.
9	ContForce = $ReLU\left(\frac{\text{TotalTorque}}{Dist \times Sin(\theta)}\right)$
10	Results [time] = ContForce
11	**End**
12	**Return** Results
13	**End**

Gastrocnemius Algorithm (Algorithm 2) describes how the Boots algorithm can be applied to the ankle joint agent to calculate contraction forces of the gastrocnemius muscle. The primary function of the gastrocnemius muscle is to support body weight in the ankle and facilitate the body's forward acceleration. Thus, the torque from the ground reaction force—reflecting the body weight in the feet— (Lines 2–4), and the torque caused by the weight of the feet (Line 5) are considered as environmental torques. Finally, the contraction force is calculated using the Boots algorithm (Line 6). In this study, array concatenation is performed using the 'Concat' function.

Algorithm 2. Calculating Gastrocnemius contraction forces during the gait.

Gastrocnemius Algorithm(Ankle Joint)
Input: Agent of the ankle joint.
Output: Diagram of physical force generated by the group muscle of gastrocnemius.

1 Begin
2 $\text{Torques1} = F^{(toe)}_{ground}(t) \times d^{(H)}_{(ankle,toe)} \times Sin\left(\theta^{(feet)}_{(V)}(t)\right)$
3 $\text{Torques2} = F^{(heel)}_{ground}(t) \times d^{(H)}_{(heel,ankle)} \times Sin\left(\theta^{(feet)}_{(V)}(t)\right)$
4 $\text{EnTorques1} = \text{Torques1} - \text{Torques2}$
5 $\text{EnTorques2} = -M^{(feet)} \times 9.8 \times d_{(ankle,feetCentre)} \times Sin\left(\theta^{(feet)}_{V}(t)\right)$
6 **Return** $\text{BootsAlgorithm}\left(\text{AnkleAgent}, \text{GastrocnemiusMuscleAgent}, \text{Concat}\begin{pmatrix}\text{EnTorques1}, \\ \text{EnTorques2}\end{pmatrix}\right)$
7 **End**

The value of $F^{(toe)}_{ground}(time)$ can be determined by analyzing how the center mass of the entire body is maintained throughout the gait cycle. The contraction force generated by the quadriceps muscle agent can be calculated as shown in Quadriceps Algorithm (Algorithm 2). The quadriceps has two tasks: Negating the body weight in the knee joint and holding the position of the body (lines 2–3), and holding the body position at the back and hip joint (lines 4–5). In each case, environmental torques are defined based on biomechanical tasks of the muscle. By considering these tasks, the contraction force of the muscle can be calculated.

Algorithm 3. Calculating Quadriceps contraction forces during the gait.

Quadriceps Algorithm(Knee Joint, Hip Joint)
Input: Agents of the knee and hip joint.
Output: Diagram of physical force generated by the group muscle of quadriceps(Quads).

1 Begin
2 $\text{EnTorques1} = W^{(ankle)}_{(t)} \times d_{(knee,ankle)} \times Sin\left(\theta^{(leg)}_{(V)}(t)\right)$
3 $\text{KneeForces} = \text{BootsAlgorithm}\ (\text{AnkleAgent}, \text{QuadsMuscleAgent}, \text{EnTorques1})$
4 $\text{EnTorques2} = W^{(hip)}_{(t)} \times d^{(H)}_{(bodyCentre,hip)}(t)$
5 $\text{HipForces} = \text{BootsAlgorithm}\ (\text{AnkleAgent}, \text{QuadsMuscleAgent}, \text{EnTorques2})$
6 **Return** (KneeForces + HipForces)
7 **End**

Gluteus Algorithm (Algorithm 4) outlines the calculation of the contraction force of the gluteus maximus muscle in the hip joint agent. The principal function of the gluteus maximus is hip extension. Therefore, it is necessary to account for the torques resulting from the ground reaction force at the feet (Line 2) and the weight of the leg from the hip to the toes (Line 3). Furthermore, the torques generated by the hamstrings and quadriceps muscles are also considered (Lines 4–5), due to their impact on the hip joint. By considering these torques as environmental, the contraction force of the muscle can be calculated using the Boots algorithm (Line 6).

The muscular agents for the Tibialis Anterior, Hamstrings, and Iliopsoas have tasks similar to those of the Gastrocnemius, Quadriceps, and Gluteus Maximus, respectively. Therefore, their behaviors can be modeled using the proposed strategy. For instance, for the tibialis anterior muscle, the variables EnTorque1 and EnTorque2 should be considered with opposite signs as defined in Algorithm 2. As another example, for the hamstrings muscle, the Quadriceps Algorithm can be used by defining the variables in different orientations.

Algorithm 4. Calculating Gluteus Maximus contraction forces during the gait.

Gluteus Algorithm(Knee Joint, Hip Joint)
Input: Agents of the hip and knee joint.
Output: Diagram of physical force generated by the group muscle of gluteus maximus.

1 Begin
2 $\text{EnTorques1} = W^{(hip)}_{(t)} \times \left(L^{(leg)} + L^{(thigh)}\right) \times Sin\left(\theta^{(thigh)}_{(V)}(t)\right)$
3 $\text{EnTorques2} = \left(M^{(thigh)} + M^{(leg)} + M^{(feet)}\right) \times d_{(hip,fullLegCentre)} \times Sin\left(\theta^{(thigh)}_{(V)}(t)\right)$
4 $\text{EnTorques3} = +\ \text{Hamstrings Contraction Forces (Knee Joint, Hip Joint)} \times L^{(thigh)} \times Sin(5^{\circ})$
5 $\text{EnTorques4} = -\ \text{Gastrocnemius Contraction Forces (Ankle Joint)} \times L^{(thigh)} \times Sin(5^{\circ})$

(continued on next page)

(*continued*)

Gluteus Algorithm(Knee Joint, Hip Joint)
Input: Agents of the hip and knee joint.
Output: Diagram of physical force generated by the group muscle of gluteus maximus.

Return BootsAlgorithm $\left(\text{AnkleAgent}, \text{GastrocnemiusMuscleAgent}, \text{Concat}\begin{pmatrix}\text{EnTorques1},\\ \text{EnTorques2},\\ \text{EnTorques3},\\ \text{EnTorques4}\end{pmatrix}\right)$
End

Utilizing the Boots algorithm and the proposed agent-driven biomechanical model, the kinematics of motion can be used to ascertain the tension forces exerted by the lower body's muscle groups. Then, an agent-based model of muscles needs to be assembled in the biomechanical model in order to calculate the muscular neural stimulation (APs) from muscular contraction forces. Thus, the agents of muscles are detailed in Section 3.3.

3.2.1. *Calculation of ground force reaction*

In this study, an agent-driven model to facilitate the biomechanical calculations of the lower body throughout the gait cycle is proposed. Despite the comprehensive description, computing the value of ground force reaction during the gait $\left(F^{(P)}_{ground}(t)\right)$ can pose challenges. However, this can be computed by considering the phase of the gait cycle and angular values of the lower body joints. Using the angular values of the lower body joints, the position of the upper body relative to the feet can be calculated. Consequently, the horizontal distance between the toes and the body's center of mass $\left(d^{(horizon)}_{(bodyCentre,toe)}(time)\right)$ can be determined during gait using the biomechanical model. This metric can then be utilized to calculate the ground reaction force during gait.

Algorithm 5 details the calculation of the ground force reaction during the gait cycle. Initially, it is necessary to verify whether both the heel and toe are in contact with the ground (Line 2). If the horizontal position of the mass center of the body is anterior to the toe, the entire body weight is transferred to the ground via the toe (Lines 3–4). Conversely, if the body mass center's horizontal position is posterior to the heel, the body weight is transferred to the ground through the heel (Lines 5–6). When the body's center of mass is situated between the heel and toe, both points bear the weight, albeit in varying proportions (Lines 7–9). If only one point is in contact with the ground, then all the weight impacting the ankle is distributed through that point (Lines 10–14).

Algorithm 5. Calculating Ground force reaction during the gait.

Ground Force Reaction(Ankle Joint)
Input: The angular displacements of the ankle joint.
Output: The value of Ground Force Reaction at the location of toe of the feet.

1 Begin
2 **IF** $\left(-\varepsilon \le \theta^{(feet)}_{(horizon)}(time) \le +\varepsilon\right)$ **then**
3 **IF** position of body mass centre is at the anterior of the toe **then**
4 **Return** $W^{(ankle)}_{(t)}$
5 **Else IF** position of body mass centre is at the posterior of the organ **then**
6 **Return** 0
7 **Else**
8 **Return** $\left(\frac{d^{(horizon)}_{(bodyCentre,toe)}(time)}{d^{(horizon)}_{(heel,toe)}} \times W^{(ankle)}_{(t)}\right)$
9 **End IF**
10 **Else IF** $\left(\theta^{(feet)}_{(horizon)}(time) \le -\varepsilon\right)$
11 **Return** $W^{(ankle)}_{(t)}$
12 **Else**
13 **Return** 0
14 **End IF**
15 **End**

3.3. *Proposed agent-driven model of human voluntary muscles*

In this section, our proposed agent-based model for human voluntary muscle is delineated. The architecture of our model is meticulously crafted to mirror the biological and physiological intricacies of muscular structure. The model enables us to convert the contraction forces of MUs to the neural stimuli (APs) for further analysis. Fig. 7 depicts the proposed agent-based model for human voluntary muscle.

In this study, each muscle is investigated as an agent. Each muscle agent comprises two inner agents: active and passive agents. The

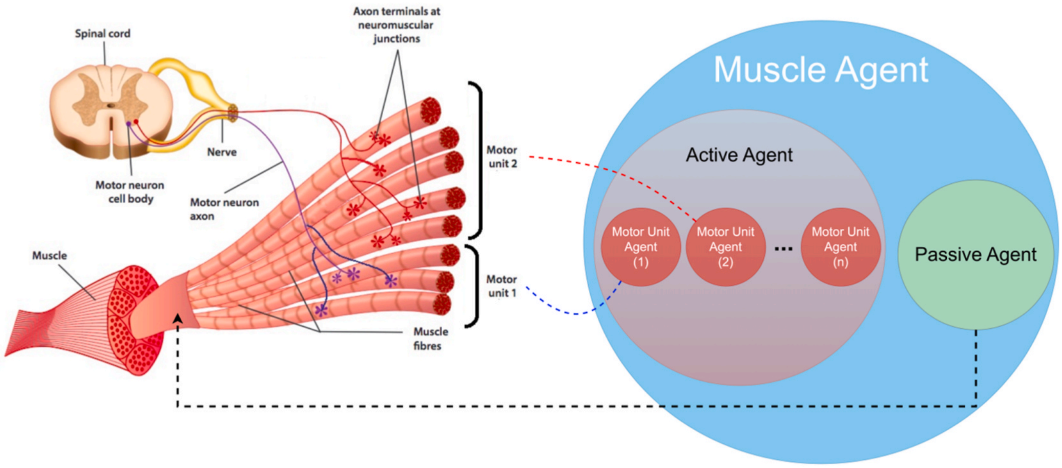


Fig. 7. The proposed agent-based model of the muscle.

active agent, composed of MU agents, acts in response to nervous stimuli, namely APs. The number of MU agents within the active agent is determined by the muscle category. For instance, muscles in the lower body have a greater number of MUs compared to facial muscles.

A formula is introduced by Ref. [5] to elucidate the force generated by a single MU receiving a solitary stimulus or AP. The formula is delineated in Equation (1), where F_0 represents a constant value that depends on the muscle category, T denotes the time interval from the stimulation to the peak tension of the MU, which depends on the MU type and environmental conditions, and t signifies the time.

$$F_{winter}(t) = F_0 \frac{t}{T} e^{\left(\frac{-t}{T}\right)} \tag{1}$$

Equation (1) is derived from the chemical interactions within muscle fibers, which are universally equal across all types of voluntary muscles. Consequently, this equation enables the modeling of contractions for any MU under various environmental conditions. For instance, factors such as ambient temperature or muscular fatigue can affect the variable T in Equation (1), thereby altering contraction dynamics.

MU contractions range from 10 ms to 100 ms. However, Equation (1) suggests that a standard MU in the lower body muscles may sustain contraction for over 600 ms following a single stimulus. To address this discrepancy, we have formulated a new equation presented as Equation (2). This equation offers a refined analysis of MU behavior when receiving a singular AP. The parameters F_0, T, and t retain their definitions from Equation (1).

$$F_{singleStimuli}(t) = F_0 \frac{t}{T} t^{\left(\frac{-t}{T}\right)} \tag{2}$$

Equation (2) quantifies the tension force generated by a MU in response to a single AP. If subsequent stimuli are applied before the muscle has fully relaxed the cycle of calcium ion release within the myofibrils is initiated again. Due to the incomplete obliteration of calcium ions from the previous stimulus, an accumulation occurs within the myofibril, enhancing the interaction between actin and myosin filaments and thereby intensifying the contraction. This effect, termed wave summation, is articulated in Equation (3), with t signifying the current time.

$$F_{multiStimuli}(t) = \sum_{for\ all\ stimuli\ occured\ in\ t_j} F_{singleStimuli}(t - t_j) \tag{3}$$

Elevating the frequency of stimulation does not invariably lead to an increase in tension force. The additional influx of calcium ions becomes ineffective when all actin and myosin filaments within a MU are fully interacting. This state, termed tetanus, is formulated in Equation (4), where $MU_{MaxForce}$ denotes the maximum tension force achievable by an MU of its specific type. Thus, the dynamics of Motor Unit agents shown in Fig. 7, can be characterized and modeled using Equation (4).

$$F_{MU}(t) = Minimum(F_{multiStimuli}(t), MU_{MaxForce}) \tag{4}$$

The length of a muscle is a critical factor that significantly impacts its tension force as it expresses the proportion of interacting actins and myosins in MUs. Equation (5) presents the relationship between muscle length and active tension force in which l denote the length of the muscle in relation to its relaxed state known as its resting length.

$$R_{length\_force}(l) = \begin{cases} 0, l < 0.6 \\ 4l - 2.4, 0.6 \le l \le 0.8 \\ l, 0.8 \le l \le 1.0 \\ 1.0, 1.0 \le l \le 1.2 \\ 3.4 - 2l, 1.2 \le l \le 1.7 \\ 0, 1.7 < l \end{cases} \tag{5}$$

Additionally, the recruitment of a greater number of MUs by the neural system intensifies muscular tension. Consequently, the behavior of a muscle's active agent illustrated in Fig. 7 can be modeled by Equation (6) where l and t denote the length in relation to its relaxed state of the muscle and current time, respectively.

$$F_{active}(t,l) = R_{length\_force}(l) \times \sum_{all\ MU\ agents} F_{MU}(t) \tag{6}$$

The passive component of a muscle produces contraction force through the inherent elasticity of the muscle. The behavior of the passive agent is formulated in Equation (7), where F_{P0} is a constant specific to the type of muscle. The variable l, representing the muscle's length, is a relative variable that expresses the length of the muscle in relation to its relaxed state.

$$F_{Passive}(l) = Maximum\{F_{P0}\ e^{(l-1)} - 1, 0\} \tag{7}$$

The force produced by a muscle is the cumulative result of its active and passive components. Consequently, Equation (8) delineates the functioning of a muscle agent. In this equation, the variables t and l retain the definitions provided in Equations (6) and (7).

$$F_{muscle}(l,t) = F_{Active}(l,t) + F_{Passive}(l) \tag{8}$$

The agent-driven model has been meticulously established and derived through the physiological behavior of the muscle. Our approach, which is known as deductive reasoning, is a robust mathematical method where new insights are systematically deduced from established laws and verified facts. Consequently, the simulation outcomes predicated on this model boast a high degree of interpretability and reliability. The muscle agents can be attached to the joint agents described in Section 3.2 and, as a consequence, motion can be naturally simulated.

3.3.1. Translating total muscular contraction forces to motor unit-specific outputs

In this paper, we presented a biomechanical model of the lower body using an agent-based approach, where each joint is represented as an agent. Additionally, the neuromuscular system is modeled through an agent-driven method, with each muscle and its MUs simulated as individual agents. This approach allows for the simulation of body organs with distinct agents, mirroring the natural processes within the human body. Consequently, this method enhances the naturalness, interpretability, and accuracy of the simulation.

The proposed models are intended for use in a reverse engineering investigation, where motion signals are converted into neural stimulation of MUs. Initially, motion signals are captured using the proposed IoT-based method. These signals are then processed through the proposed agent-based biomechanical model to calculate the contraction forces of lower-body muscle groups. Finally, the proposed agent-driven model of human muscles is employed to determine the action potentials sent to each MU during motion. However, the linkage between skeletal joint agents and muscle group agents requires further investigation. This section details this linkage to maintain the naturalness and accuracy of the simulation.

To link skeletal joint agents to muscular agents, Henneman's size principle must be considered. This principle describes the recruitment order of MUs to move a joint, starting with smaller slow-twitch MUs and progressing to larger faster-twitch MUs as needed. This recruitment order aims to minimizing the recruited MUs in order to optimize energy consumption within the body. Accordingly, four different MU types are defined in this paper: very fast-twitch, fast-twitch, slow-twitch, and very slow-twitch MUs. Table 3 details the modeled MU categories. During muscle contraction, these MUs are recruited in the specified order to achieve efficient and effective movement.

To implement the reverse engineering method using the proposed agent-driven muscle agents, we first need to apply the contraction forces calculated by the Boots Algorithm to Equation (8). In this equation, the passive force ($F_{Passive}(l)$) depends solely on the muscle's length, making the active force ($F_{Active}(l,t)$) the only variable that needs to be solved. Consequently, by utilizing Equations (7) and (6), the problem can be simplified to a MU recruitment problem. This is because relationships such as those in Equation (7) and the length-force relationship ($R_{length\_force}(l)$) depend only on the muscle length, which is known at each time step of the simulation. To address the MU recruitment problem while considering Henneman's size principle, we propose an MU Recruitment Algorithm as is detailed in Algorithm 6.

Table 3
Details of different categories of modeled MUs.

Category	Value of F_0	Value of T	Behavior of MU
Type 1	0.1	25	Very Fast, Minimally Durable Contraction
Type 2	0.1	50	Fast, Moderately Durable Contraction
Type 3	0.1	75	Slow, Durable Contraction
Type 4	0.1	100	Very Slow, Highly Durable Contraction

Algorithm 6. Proposed algorithm to solve MU recruitment problem in each muscle.

MU Recruitment Algorithm
Input: Contraction force of one muscle group in a specific time step.
Output: Number of action potentials sent to each MU category of the muscle group.

1 **Begin**
2 Results [ ] = { 0, 0, 0, 0 };
3 TotalForce = $F_{Active}(l, t)$;
4 **For** i **from** 4 **to** 1
5 $\mathrm{MU}_{\mathrm{force}}$ = Total force that can be generate by the MU with type i ;
6 **While** (TotalForce $-\mathrm{MU}_{\mathrm{force}}$) is greater or equal than 0.0 **do**
7 TotalForce$-=\mathrm{MU}_{\mathrm{force}}$;
8 Results [i] + + ;
9 **End While**
10 **End For**
11 **Return** Results ;
12 **End**

The MU Recruitment Algorithm aims to solve Equation (6) by considering Henneman's size principle to determine how action potentials are sent to different MUs within a time step of the simulation. Initially, an array with four elements, each initialized to zero, is created (Line 2). The active force $\left(\sum_{all\ MU\ agents} F_{MU}(t)\right)$ is then calculated using Equations (7) and (8) (Line 3). The process of determining how different MUs are recruited by the neural system begins next. The algorithm starts by evaluating the larger MUs to determine how many of each type can be activated (Line 4). It is important to note that the for loop begins with the larger MUs because, in scenarios where the contraction force of many smaller MUs equals the contraction force of fewer larger MUs, the neural system prefers to activate the larger MUs. This approach aligns with Henneman's principle, which aims to minimize the number of recruited MUs, thereby optimizing energy consumption and maximizing overall efficiency. The force that can be generated by one MU within a time step is calculated by computing the integral of Equation (2) over the time step (Line 5). Since as many MUs of the specified type as possible are activated, the While loop calculates the action potentials sent to each type of MU (Lines 6–9). After each MU is activated, its generated contraction force is subtracted from the total contraction force (Line 7), and the recruitment is recorded in the results (Line 8). By considering different types of MUs, the recruitment pattern is determined, and the results are returned (Lines 9–11).

By executing the MU Recruitment Algorithm at each time step of the simulation, the number of action potentials, which represent the neural stimuli sent to the muscle agents, can be calculated. Consequently, the neural activity of each muscle agent, the contraction forces, and the captured motion signals can be provided to clinical experts for further investigation and decision-making regarding patient care. Additionally, this research proposes an interpretable deep learning-based method to analyze the simulation's results in order to assist experts in their medical processes.

3.4. Proposed ensemble ANN framework for joint-based disorder detection

In this study, a motion analysis method is proposed in order to differentiate between healthy and unhealthy muscles in a patient. We propose a deep learning method to investigate the output of the agent-driven simulations in each joint. Thus, neuromuscular activity around each joint can be analyzed independently. In this study, three neural networks are trained to analyze the motion in three phases. First, the raw data captured by the IoT method is used to make an initial decision. Also, extracted features such as maximum or minimum angles during the motion are provided to the neural network. Then, using the agent-driven biomechanical model of the lower body, muscular activities are analyzed in the second phase. The neural stimulations of each muscle agent are also calculated using the proposed muscle model and are used in the third phase of pathology detection.

The proposed detection scenario is illustrated in Fig. 8. As can be seen, each aspect of the motion (appearance, biomechanical, and

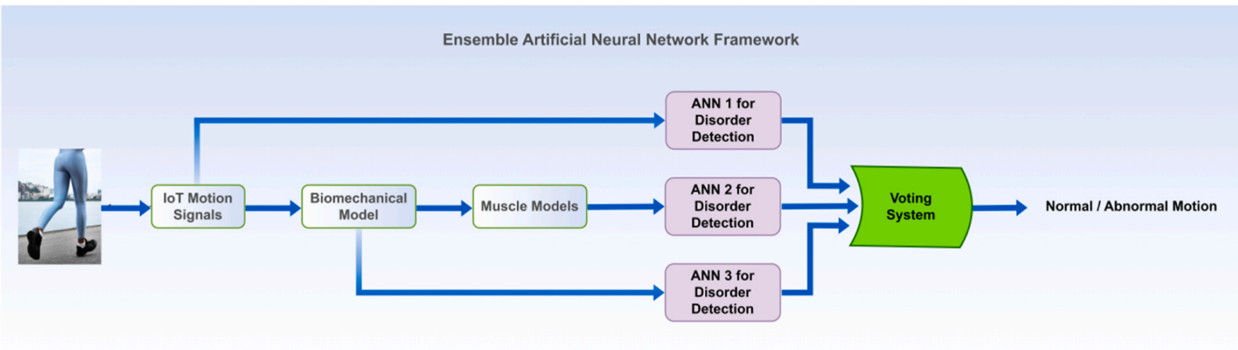


Fig. 8. Proposed ensemble neural network framework for disorder detection.

neural aspects) is investigated by an independent model. Finally, using an averaging method, the probability of healthy or unhealthy motion in each joint is calculated. The method can be repeatedly applied to different joints so that healthy and unhealthy muscles can be differentiated.

The input of each neural network is a matrix including 20 floating point values containing the angles, muscular force generation, and neural stimulations during a gait cycle. As the data is complex, neural networks are used to be trained as they can understand compound patterns from the data.

In this study, a 6 layers architecture is delineated for the ANNs. The number of perceptrons in the layers are 160, 160, 160, 100, 80, and 50 respectively. Relu is considered as the activation function for the perceptrons. A person-based cross-validation method is utilized to evaluate the networks in which the test data from each person is not previously seen by the network. Our findings, which are presented in Section 4, show that the proposed method to analyze the motion is reliable.

4. Analysis

Our methodology includes IoT-based motion signal capturing, followed by agent-based simulation of lower body joints and muscles. The outputs of the simulation are then analyzed using an interpretable ensemble neural network framework to assist the clinical experts in their initial diagnosis. In this study, we aim to respond to the research questions highlighted in Section 1. In this section, we analyze our research questions to determine whether they have been satisfactorily addressed. These questions are investigated in Sections 4.1 through 4.4, respectively.

4.1. IoT-based signal capturing approach

In this research, we proposed an IoT-based method for capturing motion signals from patients. Unlike marker or insole-based methods that require specialized and hardly available sensors, the proposed method is easy to use due to the widespread accessibility of smartphones. Thus, the proposed IoT approach enhances the usability of the signal-capturing method. In this section, the IoT-based approach is evaluated to determine its accuracy and reliability. To this end, 62 participants were asked to attach markers as well as smart mobiles to the special positions, and then they were asked to walk normally. Finally, the values that represented the angles of the lower body joints achieved by markers were compared to those obtained via smartphones. The differences were found to be zero or negligible which shows the reliability of the proposed method.

It is important to note that diagrams of angles of the body joints can vary from step to step and from person to person, as gait can be influenced by many factors, including fatigue, age, gender, and even stress. Therefore, experts focus on angular patterns rather than exact values at each time point [51]. Consequently, the patterns obtained by the proposed IoT-based method from 100 normal gaits were compared with patterns from other research. The differences were negligible. Fig. 9 illustrates these comparisons, where the yellow, red, and blue diagrams represent the minimum, maximum, and average values at each time step from 100 cases investigated in this paper. The green diagram represents the average values reported in other research which is approximately aligned with the blue diagrams [89]. These comparisons demonstrate that the proposed IoT-based method is not only globally accessible and easy to use but also accurate, making its results suitable for analysis and further research or medical investigations.

From the perspectives of availability and infrastructure independence, it is crucial that the proposed method be easy to execute on any type of smartphone, regardless of its software or hardware infrastructure. Therefore, our IoT-based application is developed on cloud-based software and is tested on mobiles with different hardware and software platforms. The application was successfully executed on Sony, Samsung, Xiaomi, and iPhone devices with both iOS and Android operating systems. Table 4 details our execution experiments. These experiments demonstrate that the proposed method is infrastructure-independent, as it is entirely developed on the cloud. Additionally, the software is programmed using an interpreting-based language rather than compile-based languages. Consequently, the program can be easily executed without any installation process on any modern smartphone.

In addition, we used our IoT application to classify each phase of the gait cycle for our captured instances as detailed in Table 5. Compared to existing research, our proposed methodology stands out for its enhanced interpretability and consequent accuracy improvements. For instance, while studies [39,40] have relied on data-driven machine learning models that entail substantial computational overhead and complexity, our study leverages a novel IoT application to precisely capture the angles of lower body joints. This

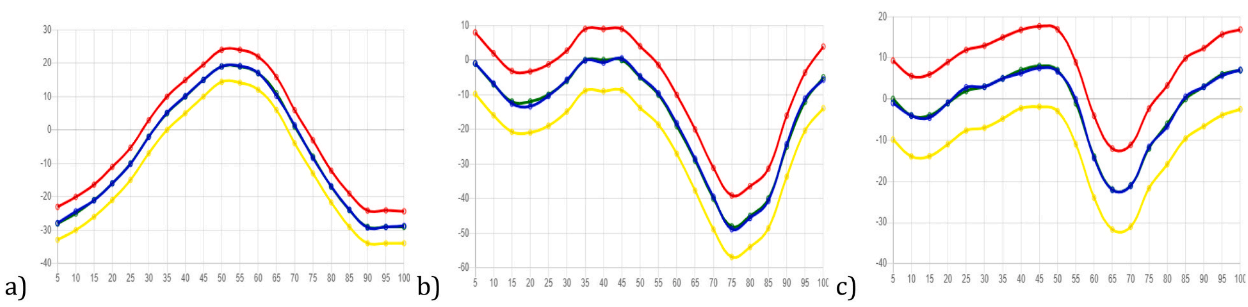


Fig. 9. Comparison of angular diagrams for hip (a), knee (b), and ankle (c) joints.

Table 4
Infrastructure details of smartphones used in the signal capturing experiment.

Experiment	Manufacturer	Brand Name	Operating System	Processor
1	Sony	Xperia XA1 ultra	Android	MediaTek Helio P20
2	Samsung	Galaxy A54 5G	Android	Exynos 1380
3	Xiaomi	Redmi Note 11 S	Android	MediaTek Helio G96
4	Apple	iPhone 16Pro Max	iOS	A18 Pro

Table 5
Comparison of accuracy for different gait phase classifiers.

Method	[39]	[40]	Proposed IoT Method
Accuracy	99.9	90.6	100.0

direct measurement allows for straightforward classification of gait phases by calculating the feet's position. Our empirical results demonstrate a significant reduction in computational load while achieving superior accuracy, making our approach not only more interpretable but also more efficient and practical for real-world applications.

4.2. Agent-driven simulation of neuromusculoskeletal system

An agent-driven modeling method is proposed in this paper to simulate the human Neuromusculoskeletal system during the gait cycle. In this section, the reliability of the results obtained from the simulation process is evaluated. Since many common movement disorders, such as MS or Parkinson's, are caused by neural system disorders, calculating neuromuscular performance in these individuals is essential. To this end, a normal gait is simulated based on the proposed modeling method in which the muscular contraction forces are initially calculated based on the proposed agent-based biomechanical model. Then, using the proposed agent-driven muscle model, the neuromuscular activities of lower-body muscle groups are calculated. This process implements the second and third modeling stages shown in Fig. 2. Finally, the results obtained from the proposed method are compared with other researches.

There are two methods for measuring neuromuscular activity, generally. The first method, known as inverse dynamics or reverse engineering, involves calculating the patterns of muscle contractions based on kinetic characteristics such as angular changes in joints using principles of biomechanics. A common approach is to define one equation for each joint and one variable for each muscle group [5,6]. In modeling the gait of lower limb muscles, three joints, and six or seven muscle groups are assumed. As a result, the defined system of equations is unsolvable. In this case, mathematical optimization methods are used to calculate the variables and contraction forces by the muscles. The proposed method in this paper employs a reverse engineering approach aimed at minimizing errors associated with mathematical optimization solutions. However, this method has the advantage of being non-invasive and painless.

The second method, known as electromyography, is an invasive and painful technique in which electrodes are directly inserted into the muscles to measure all electrochemical changes occurring at the electrode site. Since the electrodes in this method are directly inserted into the muscles, this method is more accurate compared to the first method [46]. However, electromyography signals cannot be differentiated based on the different MUs or the number of action potentials. In this section, we focus on the neuromuscular activity of lower-body muscle groups to evaluate the reliability of the proposed agent-driven model of the neuromusculoskeletal system. The output of the proposed simulation is the number of action potentials sent to different MU agents during gait, providing a deeper

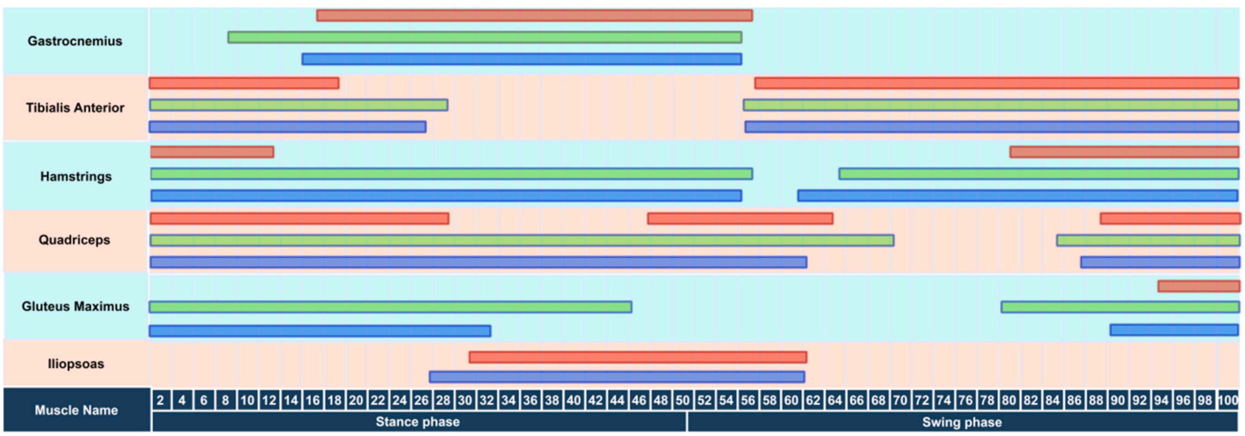


Fig. 10. Comparison of neural stimulations during normal gait.

analysis than electromyography signals. Thus, the total number of action potentials sent to all MUs within a muscle agent is summed and considered as a direct measure of the intensity of neuromuscular activity in a muscle group. The results are then compared with other research, including electromyography signals and inverse dynamics as is illustrated in Fig. 10, where the graphs display the neuromuscular activity of muscle groups during a normal gait. The blue, green, and red graphs represent the results obtained from the proposed method in this paper, electromyography, and inverse dynamics methods, respectively.

Fig. 10's horizontal axis depicts a single gait cycle's stages: initiating with foot placement for the stance phase, lifting off at the cycle's midpoint to end the stance, and then foot placement again to commence the next cycle. The comparison demonstrated in Fig. 10 shows that our proposed simulations for Tibialis anterior, Hamstrings, Quadriceps, and Gluteus Maximus muscles, are more reliable than other inverse dynamics methods as the results for our simulations have more overlapping with electromyography signals than other inverse dynamic methods.

By comparing the results obtained from the proposed method with other methods, it can be observed that the proposed method is reliable. For the gastrocnemius muscle, the proposed method shows 4 % more overlap with electromyography results compared to other inverse dynamics methods. This value is calculated to be 9 %, 61 %, 20 %, and 37 % for the tibialis anterior, hamstrings, quadriceps, and gluteus maximus muscle groups, respectively. Meanwhile, the differences between the results of the proposed method and electromyography for the gastrocnemius, tibialis anterior, hamstrings, quadriceps, and gluteus maximus muscle groups are only 2 %, 2 %, 5 %, 9 %, and 20 %, respectively. Since the iliopsoas muscle is a deep muscle, the electrode must be inserted deep into the muscle, which is very painful. Therefore, electromyography results for this muscle are not reported in some studies. Thus, the results for this muscle in the proposed method are only compared with other inverse dynamics methods. As a result, the comparison demonstrates the reliability of the results obtained from the proposed method. In this comparison, electromyography signals are reported by Ref. [90], while inverse dynamic signals are reported by Refs. [51,91].

The superiority of the proposed method over other inverse dynamics methods stems from the fact that, in this study, biomechanical equations are defined not only for each joint but also for each muscular objective. For example, one of the objectives of the hamstrings muscle is to maintain the upright posture of the body at the waist. Consequently, by calculating the body's center of mass relative to the pelvis, the contraction of the hamstrings to counteract this can be calculated during specific phases of the gait. This phenomenon has not been considered in some studies, resulting in significant differences for the hamstrings and quadriceps muscles between the proposed method and other inverse dynamics methods.

It is important to note that action potentials are inherently different from electromyography signals. Action potentials are single electrochemical pulses sent to various organs, including muscles, carrying specific messages. They can be likened to bits in modern computers. In contrast, electromyography signals are obtained by placing an electrode inside the muscles (intramuscular electromyography) or on the skin (surface electromyography). The electrode detects all the electromagnetic changes generated by the accumulation of action potentials at that location. Thus, electromyography signals can measure the intensity of neuromuscular activity at a specific position. However, it is not possible to differentiate between different MUs or individual action potentials using electromyography. Additionally, since the electrode detects all electromagnetic features, it is susceptible to noise, such as action potentials from distant muscles [5–7].

The proposed agent-driven simulation in this paper enables researchers to conduct a deeper analysis of motion, differentiate between various MUs, and investigate how action potentials are transmitted to the muscles. Since electromyography signals are a result of action potentials, the proposed method also allows experts to use the simulation results to assess the intensity of neuromuscular activity. Therefore, in this section, the results of the proposed methodology are compared with electromyography signals to evaluate neuromuscular activity. The comparison demonstrates that the proposed method is reliable, providing more detailed information compared to other reverse dynamic methods, without losing any informative data compared to electromyography.

As mentioned in Section 2, the naturalness of the algorithms inside the simulations is a key purpose of agent-based modeling and simulation methodology. This feature enhances the interpretability, accuracy, and reliability of the simulation. Despite other researches that assumed electromyography signals as the neural stimulation of the muscles, in this study we simulated action potentials as the real neural stimulations. Considering action potentials and MUs in the modeling of the human body demonstrates that the executed simulation in this research closely mimics the processes occurring within our bodies, thereby enhancing the naturalness of the simulation.

The proposed agent-based model of human voluntary muscle is grounded in the neurophysiological structure of muscle fibers and MUs. Consequently, this method can model the effects of environmental conditions such as temperature or fatigue. Additionally, since all voluntary muscles share the same physiological and biological structure, the proposed agent-based model can be used to simulate a wide range of muscles. Therefore, we believe that our proposed agent-based model accurately reflects the physiological and biological structure of muscles and is also flexible.

The same applies to our proposed biomechanical model of the lower body, where each joint is modeled as an agent based on mechanical principles. The proposed agent-driven biomechanical model is flexible, allowing for the modeling of physical changes in patients. Our comprehensive agent-driven model of the lower body includes both joint agents and muscle agents, mirroring the composition of muscular and skeletal components in the human body. Consequently, our agent-driven model is natural, and the algorithm within our simulation behaves in a manner consistent with our neuromusculoskeletal system.

Since our primary motivation in this research is to analyze different muscles to distinguish between healthy and abnormal muscles, we have also evaluated the effectiveness of the proposed approach in achieving muscle differentiation. To this end, we captured the motion of a male patient suffering from foot drop using the proposed IoT-based methodology. Foot drop is a motion disorder characterized by the weakening of the Tibialis Anterior muscle, preventing proper contraction. This disorder can be observed in patients with muscular dystrophy, amyotrophic lateral sclerosis, Charcot–Marie–Tooth disease, stroke, Parkinson's disease, polio, and other motion disorders [92–94]. As a result, the patient is unable to control his toes, and the ankle cannot perform dorsiflexion effectively. The captured signals from the patient were then used to simulate his gait using the proposed agent-based approach. Our findings indicate that the proposed method is reliable for differentiating between healthy and unhealthy muscles.

In this experiment, which has been conducted under medical supervision, the simulation results showed that the Tibialis Anterior muscle in the patient was unhealthy due to the considerable changes compared to normal gait results as shown in Fig. 11. However, other muscles were healthy as there were no significant changes in their results. This experiment shows that our proposed method for healthy and unhealthy muscles differentiation is reliable as the results were confirmed by clinical experts. In Fig. 11, neural stimulation of the muscle during a gait cycle is simulated in which the vertical axis represents the number of action potentials, and the horizontal axis shows the elapsed time of the gait cycle beginning from the stance phase.

The changes observed in the Tibialis Anterior muscle were not limited to neural stimulations; these changes also included captured signals obtained through the IoT-based method and the output of the biomechanical model, which represents muscle force patterns. All these observations indicated a disorder in the Tibialis Anterior muscle. Since no significant changes were observed in other joints and muscle groups, it was concluded that the other muscles in that patient were healthy. As this research was conducted under the supervision of clinical specialists, the results were also confirmed by clinical specialists, indicating the reliability of the proposed method for distinguishing between healthy and unhealthy muscles. In many cases, muscle differentiation offers significant medical benefits for patients. Medications often have long-term harmful effects on the body, as they influence all tissues when dissolved in the blood. By distinguishing between healthy and unhealthy muscles, treatment can be focused on abnormal muscles, thereby minimizing or even eliminating the negative effects of medication on healthy tissues.

4.3. Deep learning approach for analyzing the simulation results

In this research, an agent-driven simulation method equipped with an IoT-based approach for capturing the necessary signals is proposed. The simulation outputs can be used to investigate various aspects of the body, including skeletal, neural, and muscular systems. Given that artificial neural networks can learn patterns within the output signals, they can serve as valuable assistants for researchers and medical experts. Thus, in this research, they are trained to analyze the simulation outputs as illustrated in Fig. 2. However, it is crucial that the proposed deep learning method is accurate, reliable, and interpretable. In this section, we evaluate the proposed deep learning approach for analyzing the simulation outputs.

In this study, 6 ANN models are developed that analyze different aspects of motion during gait independently and detect intelligently whether the motion is normal. A person-based cross-validation evaluation strategy is employed, ensuring that data from an individual is seen by the networks only once—either during the training phase or the testing phase. In this evaluation, 30 volunteers were investigated, including 24 healthy individuals and 6 patients suffering from motion disorders, such as non-acute Parkinson's disease. The number of gait samples per volunteer ranged from 3 to 35. Each sample includes the raw digitalized signals from volunteers, the muscular contraction forces, and neuromuscular activity of the lower body muscles during a gait cycle. For each experiment, as detailed in Table 6, data from one patient was included in the test set, while the remaining test data were from healthy volunteers. The experiments were repeated 5 times to assess whether the proposed neural network could successfully detect abnormalities in the patient's signals. The proposed method successfully identified the patient by detecting abnormalities in the motion of the joint of interest. Additionally, the reliability of the proposed neural network model was enhanced using an ensemble framework. The experiments are detailed in Table 6 in which for each joint, motion signals from the proposed IoT-based method, muscular contraction forces, and neuromuscular activities are processed in experiment1, 2, and 3, respectively. In the experiment 1, some features such as minimum and maximum angle values, duration of one gait, and rate of gaits per time are also provided to the neural network.

In this study, different ANN models are designed to analyze one joint motion. As each model concentrates on only one aspect of the motion, we believe that the interpretability of the proposed joint-based method is enhanced compared to other researches. For designing the architecture of the ANN models, a trial and test scenario is pursued. The same scenario is followed in Ref. [88] in which one feature is analyzed in each step and the case with the highest accuracy is selected to be checked in further tests. In this scenario,

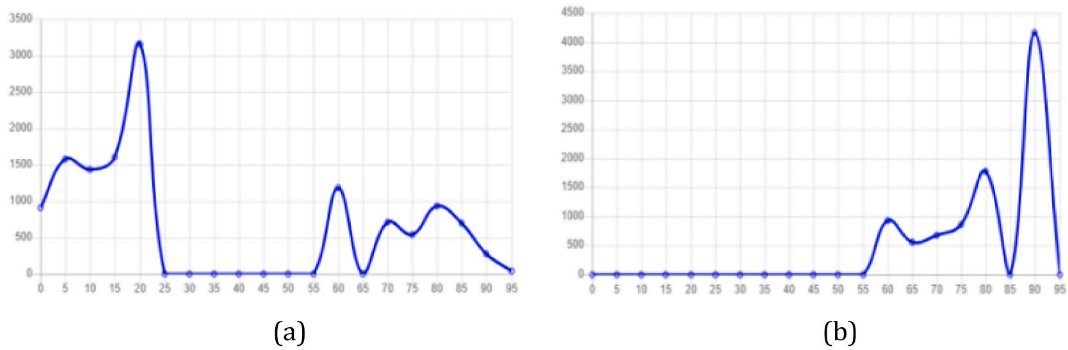


Fig. 11. Comparison between results of gait simulation for healthy (a) and unhealthy (b) tibialis anterior muscle.

Table 6
Accuracy of ANN models for joint-based normal/abnormal motion detection.

	Evaluation Tests	Entered Signals to Neural Network Models			Average Accuracy
Joint of interest	K-Fold Cross Validation	Experiment 1 (Joint Angles and summarized features)	Experiment 2 (Contraction Force)	Experiment 3 (Neuromuscular activities)	
Hip Joint	K-1	96.12	100	92.78	96.42 96.48
	K-2	98.45	98.76	93.05	
	K-3	97.89	97.85	95.34	
	K-4	99.34	96.54	94.56	
	K-5	97.10	98.46	90.12	
Knee Joint	K-1	95.12	99.87	95.67	96.53
	K-2	96.34	100	92.34	
	K-3	97.56	99.76	96.45	
	K-4	94.78	99.03	90.89	
	K-5	95.85	99.55	94.75	

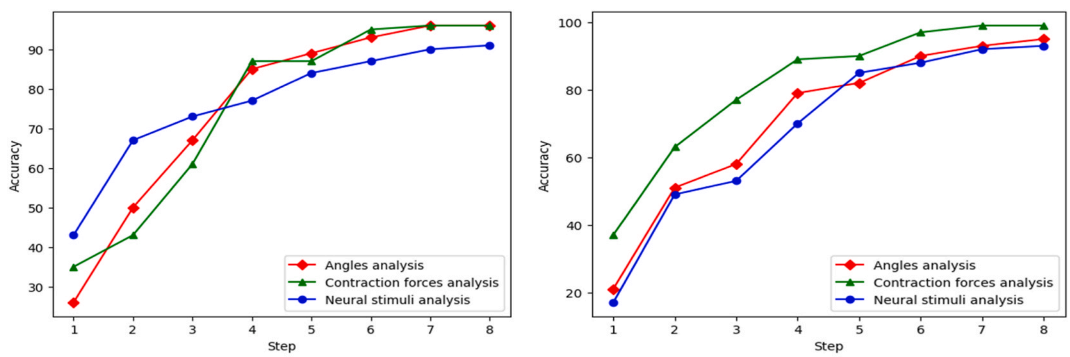
different numbers of hidden layers are investigated in the first step from one to ten hidden layers. Then, for the best case, different numbers of perceptrons are investigated for each layer in the second step. This step was time-consuming as the number of needed tests for a network with 6 hidden layers is 6×20 and the discrepancy between the number of perceptrons for each continuous step is considered as 10. Finally, 4 mathematical functions are investigated as the activation function of the networks including ReLu, Tanh, identity, and logistic. Fig. 12 details our methodology for architectural design in which the accuracy is defined as the proportion of correct predictions made by the machine learning model to the total number of cases evaluated.

Despite other deep learning methods that contain a big ambiguous neural network in which different aspects of the motion are analyzed, in this study for each joint in the lower body, an ensemble neural network framework is proposed in which different independent networks investigate different aspects of the motion including the appearance, muscular forces, and neural stimulations. This strategy is also equivalent to what a physician does to detect a motion disorder. Therefore, we believe that the proposed deep learning method supports the interpretability. Using an ensemble neural network framework not only promotes the interpretability of the system but also enhances the accuracy and reliability of the system. As mentioned in Section 2.3, in this architecture, different networks tend to learn the same patterns while learning different noises. Thus, in a system containing multiple networks for each detection, noises are eliminated.

4.4. Software requirements engineering

In this paper, software is developed that captures motion signals and simulates motion to facilitate scientific and medical investigations. While it is crucial that the algorithms executed in a computer simulator are reliable, this does not diminish the importance of other user-related software requirements. An application rich in computational capabilities is of little value if users cannot effectively interact with it. Therefore, it is essential that simulation applications have a user-friendly graphical interface, making them easy to learn, use, and maintain.

Initially, the installation phase of an application poses significant challenges for users, consuming considerable time, energy, and memory resources. This research introduces a novel, fully cloud-based application that eliminates the need for installation. The proposed solution ensures infrastructure independence by processing computations in the cloud and utilizing interpreted languages for the graphical user interface, rather than compiler-based languages. Consequently, the application operates successfully across diverse



(a) (b)

Fig. 12. Accuracy promotion of ANN models for the hip joint (a) and knee joint (b).

hardware and software platforms, as detailed in Table 7.

In this study, running the proposed web-based application was evaluated on computers with different hardware and software infrastructures. Contrasting with the approach outlined in Ref. [22], where a cloud-based platform application is developed with local processing, our proposed application is entirely cloud-based. This design ensures that all computational tasks are offloaded to the cloud, fostering an infrastructure-independent software that is globally accessible. Moreover, this approach aligns with the principles of green computing, as it leverages the cloud's potential for energy efficiency and reduced environmental impact.

While the elimination of the software installation phase significantly enhances usability, it remains crucial to design a user-friendly graphical interface. A User-friendly graphical interface facilitates the learning of the software. For instance, while OpenSim is a well-known tool for musculoskeletal modeling and simulation, some users have reported challenges with its graphical user interface. These challenges include the steep learning curve for new users and the need for improvements in user-friendliness and visualization capabilities [95]. Despite the OpenSim which contains a high number of windows, buttons, menus, and other interface components making it complex to use, our proposed application includes a surprisingly smaller number of three buttons in a single simple window which promotes the usability of the software.

Some screenshots of the proposed application are demonstrated in Fig. 13. As can be seen, the user can capture his motion signals by clicking a button on the application (Fig. 13a) and then, be informed whether his motion in the joint of interest is normal or abnormal (Fig. 13b and 13c), and also simulate his own captured motion on his mobile (or another computer connected to his cloud account) by a 3D animation (Fig. 13d). The application can also be executed on desktop computers as illustrated in Fig. 13e. For inexpert users, the application is easy to use as the user can start working by only clicking a button and the results from the proposed neural network framework are displayed in an easy-to-understand graphical interface. The interface is powerful as even an inexpert user can learn and use it easily.

For novice users, the processes are designed to be completed step by step. In each step, a short clip is presented to the user, explaining the required actions (e.g., how to position the smartphone on the body during the initial phases of motion signal capture or measuring the anatomical features). These clips are concise and clear to prevent user fatigue or confusion. Following the clip, only one task is assigned to the user (e.g., entering the length of one leg). This strategy helps avoid user fatigue, confusion, or stress. Additionally, the number of steps is minimized to enhance usability. This design makes the proposed application user-friendly, easy to learn, and easy to use.

The simplicity of the graphical user interface of the proposed tool stems from minimizing the number of interactive graphical elements, such as buttons. This feature allows users to quickly learn how to use the tool. The 3D animation display of the user's simulated movement, generated using their movement signals, along with simple and self-explanatory images, and the absence of overwhelming graphical components like texts and windows, facilitate the software's learning process. This design reduces or eliminates the need to refer to software documentation or manuals.

The importance of this feature can be examined from the perspectives of both specialist and non-specialist users. For specialists working in clinical settings, it is essential to consider the stress and psychological pressure they face, which can unintentionally reduce their accuracy. Therefore, the user interface should be as simple as possible to fill this gap. For non-specialist users, a simple user interface increases the tool's accessibility to the public. Consequently, non-specialists can use the proposed method to analyze their

Table 7
Hardware and software infrastructures for executing the proposed cloud-based simulation.

Experiment	Computer Type	Manufacturer	Operating System	Processor
1	Smartphone	Sony	Android	MediaTek Helio P20
		Samsung	Android	Exynos 1380
		Xiaomi	Android	MediaTek Helio G96
		Apple	iOS	A18 Pro
2	Desktop	Dell	Microsoft Windows XP	Intel Core i5
			Microsoft Windows 8.2	
			Microsoft Windows 10	
			Microsoft Windows XP	Intel Core i7
			Microsoft Windows 8	
			Microsoft Windows 10	
			Ubuntu	Intel Core i7
		Lenovo	Microsoft Windows 8	Intel Core i7
			Microsoft Windows 10	
			Ubuntu	
		HP	Microsoft Windows 8	Intel Core i7
			Microsoft Windows 10	
			Ubuntu	
3	Laptop	Acer	Microsoft Windows 8	Intel Core i7
			Microsoft Windows 10	
			Ubuntu	
		Lenovo	Microsoft Windows 8	Intel
			Microsoft Windows 10	Core i7
4	Tablet	Asus	Android	Intel Atom X3
		Lenovo	Android	Qualcomm Snapdragon
5	Smart TV	LG	webOS	Alpha 9 AI processor

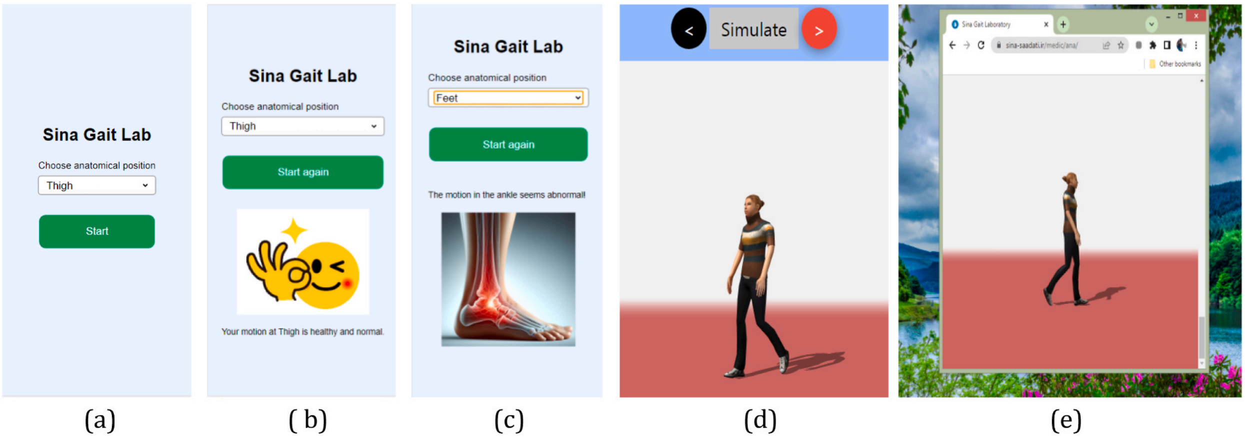


Fig. 13. Graphical Interface of the proposed web-based application.

movements at minimal cost. This approach enhances individual health awareness in society, enabling those with disorders to identify and manage them more quickly.

To assess whether the proposed strategy enhances the usability and learning of the application, a statistical experiment was conducted. In this experiment, two groups of users were asked to simulate their own gait. The first group was asked to use OpenSim as the simulator, while the second group used the proposed agent-driven application. The first group consisted of 2 PhD students, 5 Master's students, and 10 Bachelor's students. The second group comprised 17 high school students. Both groups were given 24 h and were free to use any resources, such as videos, books, or help from friends. Additionally, a small guide clip was provided to the second group. Ultimately, all high school students successfully measured their own motion signals and simulated their gaits resulting in their movement displayed in the 3D animation-based framework of the proposed application. In contrast, none of the members of the first group succeeded in their task. This experiment highlights the user-friendly, easy-to-learn, and easy-to-use features of the proposed application. Even after changing the tasks of the groups, no participant was able to successfully simulate their gait using OpenSim. While advanced computational facilities are essential, software that cannot be easily used by its users loses its value and worth. User-friendliness is a critical factor that determines the practical utility and overall success of any software application.

Table 8 compares the proposed simulation methodology with OpenSim, a biomechanical simulator, focusing on software requirements. Unlike OpenSim, the proposed simulator is user-friendly and can be easily learned by even inexperienced users. While a graphical interface is crucial in software design, infrastructure independence is equally important. Thus, the proposed simulation can be executed as a web-based application without any installation procedures, making it compatible with smartphones, PCs, tablets, and even smart TVs. This feature significantly enhances the usability of the simulation application.

The proposed simulation application utilizes cloud computing for its computations, offering several advantages. It eliminates the need for extensive memory usage on the host computer, preventing the user's computer from being burdened with large files and data. Also, cloud-based computations enable the simulation to be used on computers with limited processing power. Additionally, by aggregating computations on cloud resources, overall computing energy consumption is minimized, contributing to environmental sustainability. The concept of green computing mentioned in Table 8 refers to designing computations to limit their harmful ecological impact. Furthermore, the proposed application ensures transparency by providing only the necessary information to the user. All configurations, such as network settings, are managed automatically to enhance user experience. Unlike OpenSim, which requires expensive and hard-to-access markers, the proposed method relies solely on globally available smartphones. The application guides users through a step-by-step strategy, eliminating the need for additional guidebooks, videos, or lengthy documentation. This approach significantly improves the learning curve of the proposed simulator.

Table 8
Comparison the proposed simulator and OpenSim regarding software requirements.

Requirement	OpenSim	Proposed Simulator
User-friendly graphical interface	×	✓
Infrastructure independence	×	✓
Need for installation	✓	×
Memory Consumption	1 GB	Memory-Free
Cloud-based facilities	×	✓
Attention to green computing	×	✓
Need for configuration (Application Transparency)	Needs user intervention	Automatically managed.
Need for equipment	Markers	Smartphones
Need for additional documents (learning ability)	✓	×

5. Conclusion

In this paper, an agent-based methodology is proposed to simulate human motion for the analysis of neuromuscular activities. Additionally, an IoT-based approach is also proposed to digitalize human motion and neural networks are also employed to analyze the simulation results, thereby assisting clinical experts. The entire methodology is developed as a fully cloud-based application featuring a user-friendly graphical interface. In Section 5.1, we discuss the advantages of the proposed method. Section 5.2 addresses the limitations and potential approaches to mitigate them. Finally, Section 5.3 provides the conclusion.

5.1. Advantages of the proposed methodology

In this section, the advantages of the proposed methodology are outlined as follows:

Interpretability: Naturalness and interpretability are two primary benefits of the proposed method. Unlike neural network-based methods, which often overlook the neurophysiological and biomechanical details governing the human body, this research employs an agent-driven approach to ensure the naturalness, accuracy, interpretability, and reliability of the simulation. The proposed simulation enables researchers and medical experts to analyze various aspects, including the behaviors of motor neurons, motor units, action potentials, Henneman's size principle, classifications of motor units (such as slow-twitch and fast-twitch categories), and biomechanical metrics such as torques, forces, and angular velocities.

Availability: The IoT-based method eliminates the need for wearable sensors such as markers. Consequently, wherever smartphones are available, an advanced laboratory can be established using the proposed method. Additionally, the cloud-based platform of the proposed simulation ensures infrastructure independence. Thus, the application can be executed on various devices, including PCs, laptops, smartphones, and tablets. Since the computations are offloaded to cloud resources, the simulation can be conducted even on computers with limited processing power.

Learning ability and Usability: The proposed methodology enhances the learning ability and usability of the application. As the application is cloud-based, there is no need for installation and maintenance. A step-by-step strategy is proposed to guide the user and prevent confusion. The number of steps, buttons, menus, and windows is minimized, significantly reducing the time and effort required from the user. This, in turn, promotes the learning ability and usability of the application. The simulation results are presented in user-friendly diagrams for medical experts. Additionally, a deep learning-based method is proposed to analyze the simulation outputs and assist the experts, thereby accelerating their analysis and further enhancing the usability of the proposed approach.

5.2. Limitations and future works

In this paper, a methodology is proposed to analyze human motion in patients with non-acute motion disorders, aiming to differentiate between healthy and unhealthy muscles. To achieve this objective, we consider both scientific and technical aspects. Scientifically, we focus on the naturalness of simulations, incorporating neuromuscular and biomechanical details. Technically, we emphasize usability and user-friendliness of the graphical interface, leveraging cloud-based and IoT-based facilities to enhance usability and interpretability of AI-based methods. Additionally, we ensure infrastructure independence to facilitate software usability. In the evaluation section, we demonstrate the reliability of the proposed methodology, showing that the results are as informative, if not more so, than those of similar methods. In this section, the challenges and limitations are presented along with the proposed solutions for each limitation.

Movements Diversity: In this paper, we focus on gait as a crucial exercise that clinical experts use for the initial diagnosis of motion disorders. However, in the clinical centers, patients are asked to perform additional exercises, such as climbing stairs, knee flexion and extension while sitting, pushing their straightened arms to check for tremors, and ankle flexion or extension while lying down. These exercises provide more information for clinical experts to make a precise diagnosis of the motion disorder and, consequently, suggest the appropriate treatment. Additionally, patients are asked about the problems they experience in their daily movements. In this study, we specifically investigated gait, an important and necessary step, to propose a computer-based application to assist clinical experts. The proposed method provides important clues to medical experts. However, the proposed strategy can be extended to other exercises, as it ensures both the naturalness of the simulation and meets software requirements. Further research on other exercises is planned for future work.

Automation of Diagnosis and Treatment: After investigating the diverse movements of patients, clinical experts provide treatment based on their diagnoses. The diagnosis of disorders can be automated using machine learning models such as artificial neural networks, decision trees, and support vector machines to assist clinical experts. However, since diagnosis is critical for patient health, it is vital that AI-based methods are interpretable. Additionally, this system can be equipped with a drug-suggestion application, which is useful for patients living in underserved regions with limited clinical facilities. Developing a digital medic-based system that suggests movements as sports therapy can also be beneficial. It is necessary that the drug and treatment suggestion system be interpretable and highly reliable. The disorder detection system and the drug suggestion system are scheduled for investigation in our future work. These systems can utilize the results of the proposed agent-driven simulation to conduct a deep analysis of the patient and their movements.

Biomechanical Modeling: In this research, an agent-driven biomechanical model is proposed to investigate the mechanical aspects of the human body. The biomechanical modeling of the human skeletal system is simplified by examining gait as a 2D movement, which facilitates mechanical calculations. While analyzing 3D gait presents significant challenges, it is scheduled for future investigation. For this purpose, it is beneficial to examine the geometry of joints, such as the knee and ankle, considering details like ligaments and friction between bones in the joints. Additionally, the vibration characteristics of bones and their geometry must be modeled. An upper body biomechanical model is also necessary to simulate the influence of the upper body on the lower body. These geometric details should be incorporated into the biomechanical framework. Thus, the biomechanical model can be extended for multiple DoF scenarios. Furthermore, investigating simpler exercises, such as single joint flexion and extension, can enhance the accuracy of biomechanical calculations.

Computing Infrastructure: The purpose of the proposed methodology in this paper is to provide a computer-based approach to assist medical and clinical experts in their diagnostic procedures. To enhance the system's usability, a cloud-based application is proposed. Since the simulation computations are processed by cloud-based resources, the application can be executed on computers with limited capabilities. However, for research purposes, modelers may need to run simulations on their own computers to reconfigure certain aspects of the simulation. Given that an agent-driven approach is proposed, the agents are autonomous and their execution is simultaneous. Therefore, the simulation can be performed on a parallel computing platform. Thus, distributed computing and clustering can be employed to improve the performance and speed of the simulation. However, parallel execution presents some challenges, such as synchronizing the agents and managing network traffic between computing nodes. Additionally, despite the widespread availability of fast Internet connections globally, there are still regions where Internet access is unavailable. To address this limitation, a localized application should be developed that utilizes the resources of the host computer, thereby eliminating the need for Internet access. These challenges are scheduled for investigation in our future work.

Data Collection: In this paper, an ensemble deep learning-based method is presented to detect abnormal motions in various lower body joints, aimed at analyzing simulation outputs and assisting clinical experts. The neural network models were trained using data from healthy volunteers and patients with Parkinson's disease, who exhibited motion abnormalities in their lower body joints. Incorporating data from a larger pool of patients can further enhance the proposed method, necessitating additional patient data. Furthermore, more diverse motions are needed to provide detailed insights into different muscular tissues. Investigating simpler motions, rather than focusing solely on gait, could also improve the accuracy and interpretability of the neural network models. Consequently, more data from patients with diverse disorders are required. Collecting this additional data is scheduled for our future work.

5.3. Discussion

In this study, we present a method to analyze human motion, differentiating between healthy and unhealthy muscles in patients with non-acute motion disorders. To facilitate the motion signal capturing process, we introduce an IoT-based methodology that utilizes smart mobile devices to digitize human motion. Subsequently, we present an agent-driven biomechanical model of the human lower body to simulate the gait cycle. In the third step, our proposed agent-based model of human voluntary muscles is used to calculate the neural stimuli for each muscle group.

Our method is rich in neurophysiological and biomechanical details, providing an interpretable and natural simulation with high accuracy and reliability. The IoT-based approach to digitize motion eliminates the need for wearable and insole-based sensors, enhancing the availability and usability of the proposed method. The simulation is conducted on a cloud-based platform, making the application infrastructure-independent and executable on various devices such as PCs, laptops, tablets, and smartphones without requiring installation or maintenance. Additionally, the user-friendly graphical interface is designed to promote ease of learning and prevent user confusion.

In conclusion, our proposed methodology addresses both the scientific aspects of simulation and software requirements. Consequently, the application stands at a high level of interpretability, reliability, and usability. We hope that this methodology will pave the way for improved clinical processes for researchers, clinicians, and patients.

CRediT authorship contribution statement

Sina Saadati: Writing – original draft, Visualization, Validation, Software, Resources, Methodology, Formal analysis, Conceptualization. **Abdolah Sepahvand:** Writing – review & editing. **Mohammadreza Razzazi:** Supervision, Conceptualization.

Ethical aspects of data

In this paper, we employed our proposed signal capturing method to analyze and compare our methodology with existing works. All participants provided consent for their data and analyzed results to be published. Furthermore, personal information, including names, has been removed from our datasets. The data will be made available upon request.

Data availability statement

All data related to this research, including coding, programming, and motion-based datasets, are available at https://github.com/sinasaadati95/IntelligentGaitAnalyzer.

Declaration of competing interest

The authors declare that there are no conflicts of interest regarding the publication of this paper. No financial support was received

for the conduct of this study or preparation of this manuscript. The authors have no relevant affiliations or financial involvement with any organization or entity with a financial interest in or financial conflict with the subject matter or materials discussed in the manuscript apart from those disclosed.

References

[1] World Health Organization, Musculoskeletal conditions [internet], Available from: https://www.who.int/news-room/fact-sheets/detail/musculoskeletal-conditions. (Accessed 26 June 2024).
[2] B. Innocenti, F. Galbusera (Eds.), Human Orthopaedic Biomechanics: Fundamentals, Devices and Applications, Academic Press, 2022 Feb 24.
[3] Jr D.H. Richie, Human walking: the gait cycle, in: Pathomechanics of Common Foot Disorders, Springer, Cham, 2021, https://doi.org/10.1007/978-3-030-54201-6_2.
[4] A. Elvan, S. Ozyurek, Principles of kinesiology, in: InComparative Kinesiology of the Human Body, vol. 1, Academic Press, 2020 Jan, pp. 13–27.
[5] D.A. Winter, Biomechanics and Motor Control of Human Movement, John wiley & sons, 2009 Oct 12.
[6] R.T. Floyd, C.W. Thompson, Manual of Structural Kinesiology, McGraw-Hill, New York, NY, 2009.
[7] M. Shacklock, Clinical Neurodynamics: a New System of Neuromusculoskeletal Treatment, Elsevier Health Sciences, 2005 May 6.
[8] P.J. Atherton, D.J. Wilkinson (Eds.), Neuromuscular Assessments of Form and Function, Springer Nature, 2023 Jul 31.
[9] N.H. Sabah, Neuromuscular Fundamentals: How Our Musculature Is Controlled, CRC Press, 2020 Nov 29.
[10] J.E. Hall, M.E. Hall, Guyton and Hall Textbook of Medical Physiology E-Book: Guyton and Hall Textbook of Medical Physiology E-Book, Elsevier Health Sciences, 2020 Jun 13.
[11] L. Luo. Principles of neurobiology, 2nd, Garland Science, 2020 Sep 5, pp. 330–369.
[12] P. Mason, Medical Neurobiology, Oxford University Press, 2017.
[13] P. Sivanandy, T.C. Leey, T.C. Xiang, T.C. Ling, S.A. Wey Han, S.L. Semilan, P.K. Hong, Systematic review on Parkinson's disease medications, emphasizing on three recently approved drugs to control Parkinson's symptoms, Int. J. Environ. Res. Publ. Health 19 (1) (2021 Dec 30) 364.
[14] J.H. Yang, T. Rempe, N. Whitmire, A. Dunn-Pirio, J.S. Graves, Therapeutic advances in multiple sclerosis, Front. Neurol. 13 (2022 Jun 3) 824926.
[15] M.P. McGinley, C.H. Goldschmidt, A.D. Rae-Grant, Diagnosis and treatment of multiple sclerosis: a review, JAMA 325 (8) (2021 Feb 23) 765–779.
[16] J. Timbrell, F.A. Barile, Introduction to Toxicology, CRC Press, 2023 Feb 27.
[17] A. Paul, N. Anandabaskar, J. Mathaiyan, G.M. Raj (Eds.), Introduction to Basics of Pharmacology and Toxicology: Volume 2: Essentials of Systemic Pharmacology: from Principles to Practice, Springer, Singapore, 2021 Mar 13.
[18] J.P. Rissardo, N. Vora, B. Mathew, V. Kashyap, S. Muhammad, A.L. Fornari Caprara, Overview of movement disorders secondary to drugs, Clinics and Practice 13 (4) (2023 Aug 18) 959–976, https://doi.org/10.3390/clinpract13040087.
[19] S.L. Delp, F.C. Anderson, A.S. Arnold, P. Loan, A. Habib, C.T. John, E. Guendelman, D.G. Thelen, OpenSim: open-source software to create and analyze dynamic simulations of movement, IEEE Trans. Biomed. Eng. 54 (11) (2007 Oct 22) 1940–1950.
[20] J. Rasmussen, The AnyBody modeling system, DHM and Posturography 1 (2019 Jan) 85–96.
[21] M. Damsgaard, J. Rasmussen, S.T. Christensen, E. Surma, M. De Zee, Analysis of musculoskeletal systems in the AnyBody modeling system, Simulat. Model. Pract. Theor. 14 (8) (2006 Nov 1) 1100–1111.
[22] M.Y. Ho, M.C. Kuo, C.S. Chen, R.M. Wu, C.C. Chuang, C.S. Shih, Y. Tseng, Pathological gait analysis with an open-source cloud-enabled platform empowered by semi-supervised learning—PathoOpenGait (P11-3.001), InNeurology 102 (17_supplement_1) (2024 Apr 14) 5156. Hagerstown, MD: Lippincott Williams & Wilkins.
[23] M. Gholami, C. Napier, C. Menon, Estimating lower extremity running gait kinematics with a single accelerometer: a deep learning approach, Sensors 20 (10) (2020 May 22) 2939.
[24] M. Kraus, M.M. Saller, S.F. Baumbach, C. Neuerburg, U.C. Stumpf, W. Böcker, A.M. Keppler, Prediction of physical frailty in orthogeriatric patients using sensor insole–based gait analysis and machine learning algorithms: cross-sectional study, JMIR medical informatics 10 (1) (2022 Jan 5) e32724.
[25] C. Flagg, O. Frieder, S. MacAvaney, G. Motamedi, Real-time streaming of gait assessment for Parkinson's disease. InProceedings of the 14th ACM International Conference on Web Search and Data Mining, 2021 Mar 8, pp. 1081–1084.
[26] S.K. Kasereka, G.N. Zohinga, V.M. Kiketa, R.B. Ngoie, E.K. Mputu, N.M. Kasoro, K. Kyandoghere, Equation-based modeling vs. agent-based modeling with applications to the spread of COVID-19 outbreak, Mathematics 11 (1) (2023 Jan 3) 253.
[27] S.J. Taylor, Introducing agent-based modeling and simulation, in: Agent-based Modeling and Simulation, vol. 27, Palgrave Macmillan UK, London, 2014 Aug, pp. 1–10.
[28] E. Bonabeau, Agent-based modeling: methods and techniques for simulating human systems, Proc. Natl. Acad. Sci. USA 99 (suppl_3) (2002 May 14) 7280–7287.
[29] C.M. Macal, M.J. North, Tutorial on agent-based modeling and simulation, in: Proceedings of the Winter Simulation Conference, vol. 4, IEEE, 2005, p. 14, 2005 Dec.
[30] J. Zhang, Z. Ruan, Q. Li, Z.Q. Zhang, Towards robust and efficient musculoskeletal modelling using distributed physics-informed deep learning, IEEE Trans. Instrum. Meas. 72 (2023 Oct 19) 1–11.
[31] K. Hase, K. Miyashita, S. Ok, Y. Arakawa, Human gait simulation with a neuromusculoskeletal model and evolutionary computation, J. Vis. Comput. Animat. 14 (2) (2003 May) 73–92, https://doi.org/10.1002/vis.306.
[32] S. Song, Ł. Kidziński, X.B. Peng, et al., Deep reinforcement learning for modeling human locomotion control in neuromechanical simulation, J NeuroEngineering Rehabil 18 (2021) 126, https://doi.org/10.1186/s12984-021-00919-y.
[33] J. Shanbhag, A. Wolf, I. Wechsler, et al., Methods for integrating postural control into biomechanical human simulations: a systematic review, J NeuroEngineering Rehabil 20 (2023) 111, https://doi.org/10.1186/s12984-023-01235-3.
[34] N. Montealegre, F.J. Rammig, Agent-based modeling and simulation of artificial immune systems, in: In2012 IEEE 15th International Symposium on Object/Component/Service-Oriented Real-Time Distributed Computing Workshops, vol. 11, IEEE, 2012 Apr, pp. 212–219.
[35] A. Solovyev, M. Mikheev, L. Zhou, J. Dutta-Moscato, C. Ziraldo, G. An, Y. Vodovotz, Q. Mi, SPARK: a framework for multi-scale agent-based biomedical modeling, InProceedings of the 2010 spring simulation multiconference 11 (2010 Apr) 1–7.
[36] V. Shah, S.R. Konda, Neural networks and explainable AI: bridging the gap between models and interpretability, Int. J. Comput. Sci. Technol. 5 (2) (2021 Jun 30) 163–176.
[37] S. Bora, S. Emek, Agent-based modeling and simulation of biological systems, in: Modeling and Computer Simulation, vol. 10, IntechOpen, London, 2019 Apr, pp. 29–44.
[38] B.P. Printy, L.M. Renken, J.P. Herrmann, I. Lee, B. Johnson, E. Knight, G. Varga, D. Whitmer, Smartphone application for classification of motor impairment severity in Parkinson's disease. In2014 36th Annual International Conference of the IEEE Engineering in Medicine and Biology Society, IEEE, 2014 Aug 26, pp. 2686–2689.
[39] A. Vaith, B. Taetz, G. Bleser, Uncertainty based active learning with deep neural networks for inertial gait analysis. In2020 IEEE 23rd International Conference on Information Fusion (FUSION), IEEE, 2020 Jul 6, pp. 1–8.
[40] F. Di Nardo, C. Morbidoni, A. Cucchiarelli, S. Fioretti, Recognition of gait phases with a single knee electrogoniometer: a deep learning approach, Electronics 9 (2) (2020 Feb 20) 355.
[41] S.H. Lee, E. Sifakis, D. Terzopoulos, Comprehensive biomechanical modeling and simulation of the upper body, ACM Trans. Graph. 28 (4) (2009 Sep 8) 1–7.
[42] L.N. Wimalasena, J.F. Braun, M.R. Keshtkaran, D. Hofmann, J.Á. Gallego, C. Alessandro, M.C. Tresch, L.E. Miller, C. Pandarinath, Estimating muscle activation from EMG using deep learning-based dynamical systems models, J. Neural. Eng. 19 (3) (2022 May 19) 036013.

[43] Y. Zhao, Z. Zhang, Z. Li, Z. Yang, A.A. Dehghani-Sanij, S. Xie, An EMG-driven musculoskeletal model for estimating continuous wrist motion, IEEE Trans. Neural Syst. Rehabil. Eng. 28 (12) (2020 Nov 13) 3113–3120.
[44] J. Zhang, Y. Zhao, F. Shone, Z. Li, A.F. Frangi, S.Q. Xie, Z.Q. Zhang, Physics-informed deep learning for musculoskeletal modeling: predicting muscle forces and joint kinematics from surface EMG, IEEE Trans. Neural Syst. Rehabil. Eng. 31 (2022 Dec 5) 484–493.
[45] J.R. Potvin, A.J. Fuglevand, A motor unit-based model of muscle fatigue, PLoS Comput. Biol. 13 (6) (2017 Jun 2) e1005581, https://doi.org/10.1371/journal.pcbi.1005581.
[46] D.I. Rubin (Ed.), Electromyography, an Issue of Neurologic Clinics, E-Book: Electromyography, an Issue of Neurologic Clinics, E-Book. Elsevier Health Sciences, 2021 Oct 6.
[47] L.M. Mendell, The size principle: a rule describing the recruitment of motoneurons, Journal of neurophysiology 93 (6) (2005 Jun) 3024–3026.
[48] D. Kim, S.H. Kim, T. Kim, B.B. Kang, M. Lee, W. Park, S. Ku, D. Kim, J. Kwon, H. Lee, J. Bae, Review of machine learning methods in soft robotics, PLoS One 16 (2) (2021 Feb 18) e0246102.
[49] D.J. Saxby, B.A. Killen, C. Pizzolato, C.P. Carty, L.E. Diamond, L. Modenese, J. Fernandez, G. Davico, M. Barzan, G. Lenton, S.B. da Luz, Machine learning methods to support personalized neuromusculoskeletal modelling, Biomech. Model. Mechanobiol. 19 (2020 Aug) 1169–1185.
[50] E. Kellis, A.J. Blazevich, Hamstrings force-length relationships and their implications for angle-specific joint torques: a narrative review, BMC Sports Science, Medicine and Rehabilitation 14 (1) (2022 Sep 5) 166.
[51] J. Richards, D. Levine, M.W. Whittle (Eds.), Whittle's Gait Analysis-E-Book: Whittle's Gait Analysis-E-Book, Elsevier Health Sciences, 2022 Aug 28.
[52] J. Perry, J. Burnfield, Gait Analysis: Normal and Pathological Function, CRC Press, 2024 Jun 1.
[53] M. Srinivas, G. Sucharitha, A. Matta (Eds.), Machine Learning Algorithms and Applications, John Wiley & Sons, 2021 Aug 24.
[54] E. Alpaydin, Introduction to Machine Learning, MIT press, 2020 Mar 24.
[55] A. Burkov, Machine Learning Engineering, True Positive Incorporated, Montreal, QC, Canada, 2020 Sep.
[56] S. Shanmuganathan, Artificial neural network modelling: an introduction, in: S. Shanmuganathan, S. Samarasinghe (Eds.), Artificial Neural Network Modelling, Studies in Computational Intelligence, vol. 628, Springer, Cham, 2016, https://doi.org/10.1007/978-3-319-28495-8_1.
[57] N. Buduma, N. Buduma, J. Papa, Fundamentals of Deep Learning, O'Reilly Media, Inc., 2022 May 16.
[58] A. Zhang, Z.C. Lipton, M. Li, A.J. Smola, Dive into Deep Learning, Cambridge University Press, 2023 Dec 7.
[59] S. Saadati, M. Amirmazlaghani, Revolutionizing endometriosis treatment: automated surgical operation through artificial intelligence and robotic vision, J Robotic Surg 18 (2024) 383, https://doi.org/10.1007/s11701-024-02139-7.
[60] D. Stone, C. Jarrett, M. Woodroffe, S. Minocha, User Interface Design and Evaluation, Elsevier, 2005 Apr 29.
[61] MacKenzie IS. Human-computer Interaction: an Empirical Research Perspective.
[62] M.S. Alkatheiri, Artificial intelligence assisted improved human-computer interactions for computer systems, Comput. Electr. Eng. 101 (2022 Jul 1) 107950.
[63] R.M. Rangayyan, S. Krishnan, Biomedical Signal Analysis, John Wiley & Sons, 2024 Feb 19.
[64] M.W. Barnett, P.M. Larkman, The action potential, Practical Neurol. 7 (3) (2007 Jun 1) 192–197.
[65] K.A. Taylor, John Squire and the myosin thick filament structure in muscle, J. Muscle Res. Cell Motil. 44 (2023) 143–152, https://doi.org/10.1007/s10974-023-09646-4.
[66] W. Yan, Computational Methods for Deep Learning, vol. 10, Springer, 2021, 978-3.
[67] C.M. Bishop, H. Bishop, Deep Learning: Foundations and Concepts, Springer Nature, 2023 Nov 1.
[68] H.M. Rai, Cancer detection and segmentation using machine learning and deep learning techniques: a review, Multimed. Tool. Appl. 83 (9) (2024 Mar) 27001–27035.
[69] O. Coser, C. Tamantini, P. Soda, L. Zollo, AI-based methodologies for exoskeleton-assisted rehabilitation of the lower limb: a review, Frontiers in Robotics and AI 11 (2024 Feb 9) 1341580.
[70] F.R. Nezhat, M. Kavic, C.H. Nezhat, C. Nezhat, Forward we go, J. Soc. Laparoendosc. Surg.: Journal of the Society of Laparoscopic & Robotic Surgeons. 27 (1) (2023 Jan).
[71] S. Kulkarni, N. Seneviratne, M.S. Baig, A.H. Khan, Artificial intelligence in medicine: where are we now? Acad. Radiol. 27 (1) (2020 Jan 1) 62–70.
[72] F.L. Fan, J. Xiong, M. Li, G. Wang, On interpretability of artificial neural networks: a survey, IEEE Transactions on Radiation and Plasma Medical Sciences 5 (6) (2021 Mar 17) 741–760.
[73] N.A. Wani, R. Kumar, J. Bedi, DeepXplainer: an interpretable deep learning based approach for lung cancer detection using explainable artificial intelligence, Comput. Methods Progr. Biomed. 243 (2024 Jan 1) 107879.
[74] Y. Zhang, P. Tiňo, A. Leonardis, K. Tang, A survey on neural network interpretability, IEEE Transactions on Emerging Topics in Computational Intelligence 5 (5) (2021 Aug 24) 726–742.
[75] X. Li, H. Xiong, X. Li, X. Wu, X. Zhang, J. Liu, J. Bian, D. Dou, Interpretable deep learning: interpretation, interpretability, trustworthiness, and beyond, Knowl. Inf. Syst. 64 (12) (2022 Dec) 3197–3234.
[76] Q. Teng, Z. Liu, Y. Song, K. Han, Y. Lu, A survey on the interpretability of deep learning in medical diagnosis, Multimed. Syst. 28 (6) (2022 Dec) 2335–2355.
[77] Z. Zhang, J. Yang, Z. Zhang, Y. Li, Cross-training deep neural networks for learning from label noise. In2019 IEEE International Conference on Image Processing (ICIP), IEEE, 2019 Sep 22, pp. 4100–4104.
[78] B. Han, Q. Yao, X. Yu, G. Niu, M. Xu, W. Hu, I. Tsang, M. Sugiyama, Co-teaching: robust training of deep neural networks with extremely noisy labels, Adv. Neural Inf. Process. Syst. 31 (2018).
[79] R.C. Martin, Agile Software Development: Principles, Patterns, and Practices, Prentice Hall PTR, 2003 Sep 1.
[80] R. Mall, Fundamentals of Software Engineering, PHI Learning Pvt. Ltd., 2018 Sep 1.
[81] A. Nagaraj, Introduction to Sensors in IoT and Cloud Computing Applications, Bentham Science Publishers, 2021 Feb 1.
[82] P. Bhambri, S. Rani, G. Gupta, A. Khang (Eds.), Cloud and Fog Computing Platforms for Internet of Things, CRC Press, 2022 Jun 7.
[83] A. Sunyaev, A. Sunyaev, Internet Computing, Springer International Publishing, New York, NY, USA, 2020.
[84] P. Flocchini, G. Prencipe, N. Santoro, Distributed Computing by Oblivious Mobile Robots, Springer Nature, 2022.
[85] K. Morris, Infrastructure as Code, O'Reilly Media, 2020 Dec 8.
[86] S.L. Peng, S. Pal, L. Huang (Eds.), Principles of Internet of Things (IoT) Ecosystem: Insight Paradigm, Springer International Publishing, 2020.
[87] A. Gupta, The IoT Hacker's Handbook, Apress, Berkeley, CA, 2019.
[88] S. Hossain, S. Azam, S. Montaha, A. Karim, S.S. Chowa, C. Mondol, M.Z. Hasan, M. Jonkman, Automated breast tumor ultrasound image segmentation with hybrid UNet and classification using fine-tuned CNN model, Heliyon 9 (11) (2023 Nov 1) e21369, https://doi.org/10.1016/j.heliyon.2023.e21369.
[89] K.B. Bumbard, H. Herrington, C.H. Goh, A. Ibrahim, Incorporation of torsion springs in a knee exoskeleton for stance phase correction of crouch gait, Appl. Sci. 12 (14) (2022 Jul 12) 7034, https://doi.org/10.3390/app12147034.
[90] C. Steele (Ed.), Applications of EMG in Clinical and Sports Medicine, BoD–Books on Demand, 2012 Jan 11.
[91] R.M. Howard, R. Conway, A.J. Harrison, Muscle activity in sprinting: a review, Sports BioMech. 17 (1) (2018 Jan 2) 1–7.
[92] J.D. Rose, V.J. Martorana, The Foot Book: the Complete Guide to Caring for Your Feet and Ankles, JHU Press, 2023 Dec 12.
[93] F. Saeed, S. Mukherjee, K. Chaudhuri, et al., Prognostic indicators of surgical outcome in painful foot drop: a systematic review and meta-analysis, Eur. Spine J. 30 (2021) 3278–3288, https://doi.org/10.1007/s00586-021-06936-8.
[94] J.D. Stewart, Foot drop: where, why and what to do? Practical Neurol. 8 (3) (2008 Jun 1) 158–169, https://doi.org/10.1136/jnnp.2008.149393.
[95] OpenSim Community. OpenSim GUI [Internet]. GitHub; [cited 2024 July 3]. Available from: https://github.com/opensim-org/opensim-gui.