Journal of Intelligent Systems and Internet of Things
Vol. 15, No. 02, PP. 55-75, 2025
DOI: https://doi.org/10.54216/JISIoT.150205

Explainable AI-Driven Gait Analysis Using Wearable Internet of Things (Wiot) and Human Activity Recognition

Ponugoti Kalpana [1,*], Sarangam Kodati [2], L. Smitha[3], Dhasaratham[4], Nara Sreekanth[5], Aseel Smerat[6,7], Muhannad Akram Ahmad[8]

[1]Assistant Professor, Department of Computer Science and Engineering, AVN Institute of Engineering and Technology, Hyderabad, Telangana, 501510, India

[2]Associate Professor, Department of Information Technology, CVR College of Engineering, Hyderabad, Telangana, 501510, India

[3]Assistant Professor, Department of Information Technology, G Narayanamma Institute of Technology and Science, Hyderabad, Telangana, India

[4]Associate Professor, Department of Information Technology, TKR College of Engineering and Technology Hyderabad, Telangana, India

[5]Associate Professor, Department of Computer Science and Engineering, BVRIT Hyderabad College of Engineering for Women, Hyderabad, Telangana, India

[6]Centre for Research Impact & Outcome, Chitkara University Institute of Engineering and Technology, Chitkara University, Rajpura, 140401, Punjab, India

[7]Applied science research center, applied science private university, Amman 11931, Jordan

[8]Faculty of Economics and Administrative Sciences, Al Albayt University, Mafraq, Jordan

Emails: drkalpanacse@gmail.com; k.sarangam@gmail.com; smitha2005sri@gnits.ac.in; dasarath.m@gmail.com; nara.sreekanthap@gmail.com; Smerat.2020@gmail.com; dr.muhannadahmad@aabu.edu.jo

Abstract

Due to the rapid expansion of the Internet of Things (IoT), supportive systems for healthcare have made significant advancements in both diagnosis and treatment processes. To provide optimal support in clinical settings and daily activities, these systems must accurately detect human movements. Real-time gait analysis plays a crucial role in developing advanced supportive systems. While machine learning and deep learning algorithms have significantly improved gait detection accuracy, many existing models primarily focus on enhancing detection accuracy, often neglecting computational overhead, which can affect real-time applicability. This paper proposes a novel hybrid combination of Sparse Gate Recurrent Units (SGRUs) and Devil Feared Feed Forward Networks (DFFFN) to effectively recognize human activities based on gait data. These data are gathered through Wearable Internet of Things (WIoT) devices. The SGRU and DFFFN networks extract spatio-temporal features for classification, enabling accurate gait recognition. Moreover, Explainable Artificial Intelligence (EAI) assesses the interoperability, scalability, and reliability of the proposed hybrid deep learning framework. Extensive experiments were conducted on real-time datasets and benchmark datasets, including WHU-Gait and OU-ISIR, to validate the algorithm's efficacy against existing hybrid methods. SHAP models were also employed to evaluate feature importance and predict the degree of interoperability and robustness. The experimental results show that the method, combining Sparse GRUs and Tasmanian Devil Optimization (TDO)-inspired classifiers, achieves superior accuracy and computational efficiency compared to existing models. Tested on real-time and benchmark datasets, the model demonstrates significant potential for real-time healthcare applications, with an AUC of 0.988 on real-time data. These findings suggest that the approach offers practical benefits for improving gait recognition in clinical settings.

Received: September 25, 2024 Revised: November 20, 2024 Accepted: January 10, 2025

Keywords: Internet of Things; WIoT; Explainable AI; Devil Feared Feed Forward Networks; Sparse Gated Recurrent Units

1. Introduction

Activity recognition (AR) has rapidly expanded in fields like healthcare [1], home automation [2], and crime investigation. These techniques improve quality of life and help people stay independent without external support [3]. In healthcare, AR technology is advancing quickly, focusing on monitoring patient mobility during rehabilitation and tracking recovery after treatment to improve the standard of living.

However, activity data is still complex, prompting continuous research into designing better human activity monitoring systems. Early systems used simple binary sensors [4,5]. Recently, the IoT has been used to collect and analyze human activities and gestures [6,7]. These IoT devices designed for continuous use indoors and outdoors, prioritize data privacy and security.

Due to their versatility, WIoT devices are widely used in AR systems [8-10]. They can capture and process activity data, such as gait signals. Sensors like accelerometers and gyroscopes, embedded in these devices, record and transmit gait sequences for further analysis. This data helps identify physical activities, especially in healthcare applications.

Thus, WIoT devices are essential for collecting data in automatic recognition networks. The gathered information is used to develop efficient recognition systems. Traditional machine learning models, like Decision Trees [11-13], Hidden Markov Models [14-16], and Support Vector Machines (SVM) [17-19], have improved recognition rates but struggle with larger datasets, requiring more advanced learning models.

In recent years, research has focused on deep learning (DL) models, which handle large datasets more effectively. DL techniques like Convolutional Neural Networks (CNN) [20] and Recurrent Neural Networks (RNN) are key in developing AR systems. Hybrid DL models are also gaining attention in AR system design. However, transforming gait data is crucial to enhance classification performance and reduce computational costs, making hybrid approaches necessary for high performance with lower complexity.

A. Motivation:

Considering this, the study introduces a novel hybrid algorithm combining Sparse GRU with Tasmanian Devil Optimization-inspired Classification Systems. The user-specific Sparse GRU extracts spatial features, while another Sparse GRU focuses on temporal features. Such features are subsequently input into complexity-conscious TDO (Tasmanian Devil Optimization) -inspired classification systems to enhance activity recognition (AR) classification with reduced complexity cost.

B. Contribution:

1. This study aims at developing innovative test beds using WIoT devices to efficiently gather raw gait data.
2. It also introduces a method for reorganizing the raw data to make it suitable for training DL models, aiming for improved performance.
3. The paper presents a hybrid DL approach designed for efficient feature extraction with reduced processing overhead and a high gait detection rate.
4. Lastly, the study demonstrates the effectiveness of the suggested method through experimentation with various benchmark datasets and juxtaposing its effectiveness against other existing DL- based AR systems.

C. Organization of the Paper:

The paper is structured as: Section II highlights a survey of relevant studies by various authors. Section III outlines the data collection unit, pre-processing steps, and the suggested hybrid method. Section IV describes the dataset, experimental procedures, outcomes, findings, and evaluation. The paper concludes in Section V with a discussion on future improvements.

2. Related Work

Abdullah et al. utilized a neural network to diagnose human irregularities identified through walking patterns observed in the lower limbs. Real-time data were obtained using Levenberg-Marquardt method, with artifacts removed via Butterworth filters for optimal neural network training. Gait data were recorded from 5 subjects at speeds of “2.4, 3.2, and 5.4 km/h,” with a total of 45 instances used for assessment [21]. While the suggested

neural network attained improved accuracy with the tested data, it is unsuitable for changeable activities, and the tested data is relatively limited.

Aybuke et al. employed three supervised ML algorithms for human AR system: random forest (RF), naïve Bayes, and IB1 classifiers. The hugaDB dataset includes data on standing, sitting, running, and walking, monitored utilising accelerometers and gyroscopes. The RF model surpassed the other methods in classification accuracy whereas demanding lower setup time [22]. Jucheol et al. proposed the multi-model gait recognition classifier combining CNN and RNN with a support vector feature extractor [23].

Jiang and Yin applied Short-time Discrete Fourier Transform (STDFT) for generating time-frequency-spectral images from time-series signals in [24]. Convolutional Neural Networks (CNNs) were then used to process these images for recognizing routine activities including walking and standing. Utilizing a combination of time-frequency-spectral features and CNNs, Laput and Harrison [25] developed a fine-grained hand movements sensing network, achieving a 95.2 percent accuracy in classifying 25 distinct hand movements done by 12 volunteers. Spectral characteristics are utilized beyond wearable sensor-based activity detection extending to device-free AR. To address modality-specific temporal characteristics, Ha and Choi [26] introduced a novel CNN architecture with separate 1D CNNs across every modality. Researchers are exploring various CNN variants to efficiently integrate temporal features.

Shen et al. [27] applied gated CNNs for recognizing daily activities from audio signals, finding them to outperform traditional CNNs in accuracy. Long et al. utilized residual blocks to develop a two-stream CNN structure capable of handling multiple time scales. Guan and Plötz [28] created an ensemble model of several deep Long Short Term Memory models, which showed improved results compared to isolated systems on three benchmark datasets. In addition to variations in RNN structures, other researchers have explored different RNN cells. For instance, Yao et al. constructed a Recurrent Neural Network using GRU instead of LSTM cells for activity recognition. Despite this, several studies have shown that different RNN cells do not considerably outperform conventional LSTM cells in categorizing accuracy. Wang et al. combined CNNs with LSTMs to build a classifier that automatically extracts complex features from audio data and recognizes gestures. For local temporal feature extraction at different scales, Xu et al. employed an advanced Inception CNN architecture, while GRUs are utilized for effective global temporal representation.

To evaluate complicated temporal hierarchies, Yuki et al. [29] utilized a dual-stream ConvLSTM model, where one stream manages short time intervals besides the remaining handles long time intervals. Guo et al. [30] suggested using MLPs to create a base classifier for every sensory system and assigning ensemble weights at the classifier level to integrate all classifiers. They not only assessed detection accuracy when developing the base classifiers as well as emphasizing diversity by incorporating various measures. This approach preserves the variety of distinct systems, which is essential for reducing overfitting problems and improving overall generalization ability.

Most recently, research has shifted towards improving the interpretability and explainability of AI models in human activity recognition and healthcare applications. Kumbhar et al. [31] proposed a novel framework combining feature importance-based methodologies with Local Interpretable Model-agnostic Explanations (LIME) for enhancing the interpretability of predictive healthcare models. Their hybrid approach achieved remarkable accuracy of 0.961 while maintaining transparency in decision-making processes. Similarly, Wang et al. [32] demonstrated the application of XAI techniques in medical IoT systems for detecting age-related macular degeneration using various ophthalmic imaging modalities. Their findings highlighted XAI's potential not only in improving model transparency and trustworthiness but also in enhancing the robustness of AI models in real-world medical applications.

3. Proposed Methodology

A. System Overview

The suggested model comprises four key components: (i) Data Collection Unit, (ii) Data Pre-processing and Filtering, (iii) Spatial and Temporal Feature Extraction utilizing the suggested framework, and (iv) Classification Phase. Figure 1 represents the suggested methodology’s architecture.

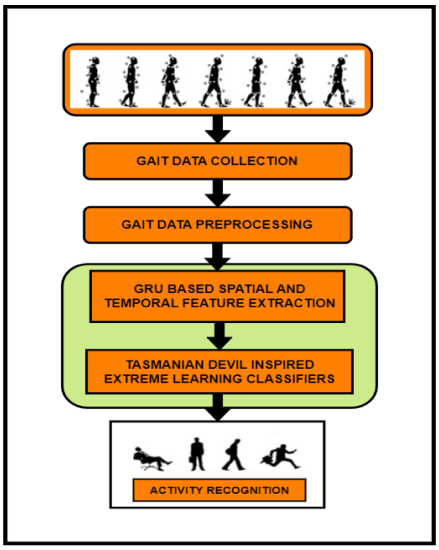


Figure 1. Suggested Methodology's Architecture

B. Material and Methods

a) Data Collection Unit

Real- time data was collected from 29 participants, whose body weights varied between 25 kg and 64 kg. All volunteers are healthy, without any neurological conditions or damage to their legs or feet that could affect gait phase detection. Leveraging advancements in Internet of Things (IoT) technology, six battery-operated IoT devices were utilized for gathering required inertial data. MICOTT boards served as the primary IoT devices for capturing lower limb inertial data. These boards feature an 8-BIT NODEMCU as a primary processor, connected to 10-BIT SPI (Serial Peripheral Interface) MCP3008 analog channels, and ESP8266 Wi-Fi transceivers. The IoT devices used for data collection included ADXL435 three-axis accelerometers and BMG250 three-axis gyroscopes. These sensors were placed on both the lower limbs of the participants to capture precise gait data. Micropython programming was used to manage data collection and transmission to the cloud. A series of Li-Ion batteries, having 3.3 V, powered the board and could be replaced as needed when depleted.

During the experiments, volunteers were instructed to walk normally on a treadmill at speeds ranging from 0.66 m/s to 1.3 m/s for a minimum duration of 180 seconds. Each participant completed two walking sessions at each speed setting. Data were collected at three-minute intervals and transmitted to the cloud for further analysis. To evaluate the effectiveness of the proposed system, various public benchmark datasets, including whuGait and OU-ISIR, were utilized. Detailed information about these datasets is provided in Section 4. Figure 2 illustrates the data gathering approach employed in the suggested framework.

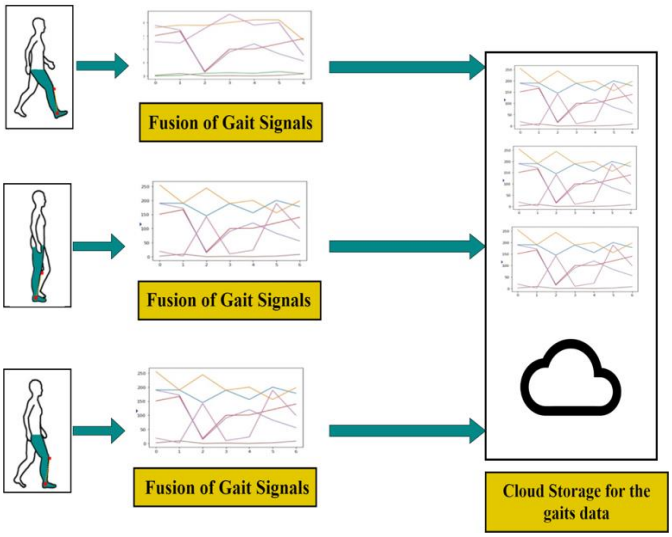


Figure 2. Data Acquisition Setup Implemented for the Recommended Technique

b) Data Pre-Processing Process

The cloud-stored data includes multiple features from six IoT devices, with each entry containing acceleration and angular velocity measurements along the X, Y, and Z axes. The data samples were arranged in sequences, represented as

$$y = \{s1, f2, s2, f1, s3, f3\} \quad (1)$$

Here, y represents the entire data set, with s1, s2, and s3 corresponding to accelerometer data, and f1, f2, and f3 representing angular velocity data, all saved in the cloud. As outlined in the eqn, the aggregated data is archived in the cloud and requires partitioning and extraction to facilitate improved classification. The cloud-stored data are downloaded for offline use, where pre-processing steps are employed to achieve optimal data partitioning and extraction. To obtain reduced computational complexity while maintaining high segmentation accuracy, the study introduces a new Pearson Correlation Sliding Window technique, integrating Pearson Correlation Coefficient with Sliding Window methods [33]**.** 'P' value is crucial in data extraction process, where varying thresholds were applied for optimal extraction throughout the period. Figure 3 illustrates the pre-processed data following the application of the suggested model.

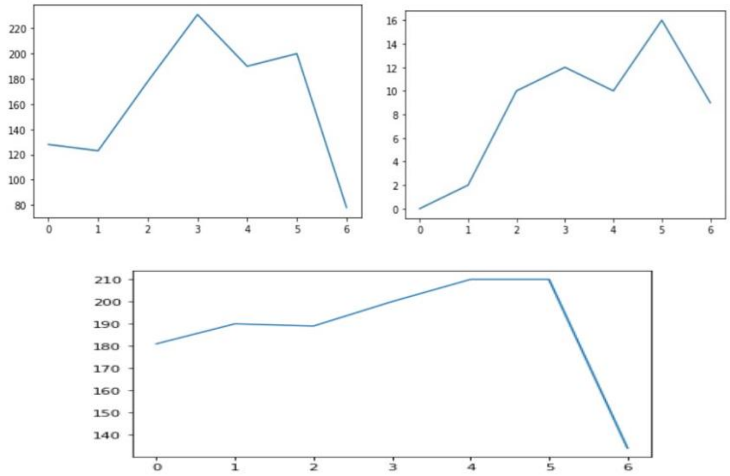


Figure 3. Data PreProcessing Phase Following the Application of the suggested Separation Method

C. Proposed Hybrid Deep Learning Model

Through analyzing the walking abilities of individual models with fused features, we discovered that combining various learning models enhances gait signal recognition and classification while reducing processing overhead. Therefore, we aim to develop a hybrid ensemble of DL and ML methods to effectively learn the combined spatiotemporal features, achieving high accuracy with lower computational complexity. Figure 4 illustrates the Entire Framework for the suggested hybrid method

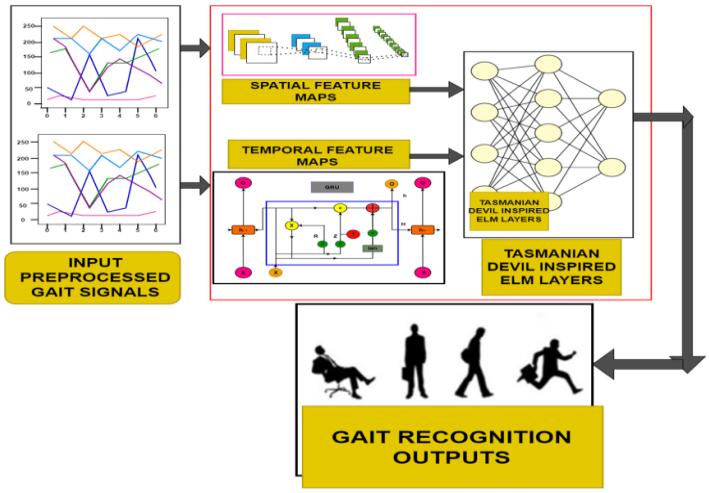


Figure 4. Suggested Framework of the Hybrid Feature Extraction and Classification layer

a) GRU based Temporal Feature Extraction

The primary component employed for extracting temporal features is GRU segment, which analyze the acquired information from IoT-cloud networks. The architecture of the GRU model utilized is depicted in Figure 5.

GRU model comprises two gates and is regarded as more efficient than Long Short Term Memory and Recurrent Neural Network methods. where xt represents the present state's input feature, yt is the output state, ht represents the current output of the module, Zt and rt denotes update and reset gates, W(t) denotes weights, B(t) indicates bias weights at the present state. The formula for extracting the feature maps is represented as:

$$P = GRU(\sum_{t=1}^{n}[x_{t,} h_{t,} z_{t,} r_t(W(t), B(t), \eta(tannh))] \quad (2)$$

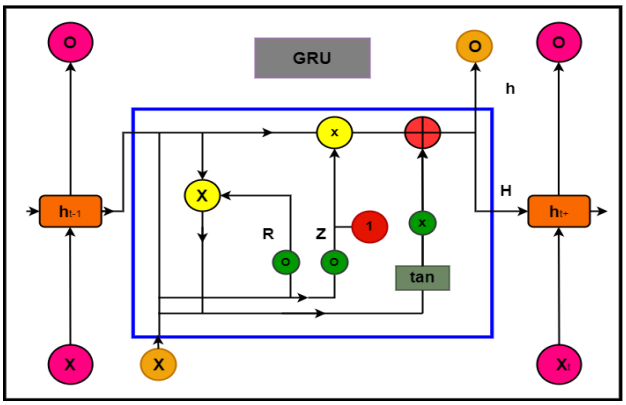


Figure 5. GRU network for Temporal Feature Extractor

b) Sparse GRU Modifications

1. **Sparse Weight Matrices**: In this case, some of the elements in the weight matrices W_z , W_r, U_r, W_h, and U_h are set to zero, reducing the number of computations:

$W_z \cdot x_t \rightarrow S(W_z) \cdot x_t$ (3)

Where $S(W_z)$) is the sparse version of the weight matrix W_z, meaning some weights are zero. The same applies to other weight matrices.

2. **Sparse Connectivity**: In sparse GRUs, some neuron connections are removed, meaning not every neuron in the previous hidden state h_{t-1} interacts with all neurons in the present state. This sparsity can be represented as applying a mask MMM to the hidden state computations:

$Uz \cdot h_{t-1} \rightarrow M \odot (U_z \cdot h_{t-1})$ (4)

Where MMM is a binary mask that removes certain connections, making the computation sparse.

3. **Sparse Activation**: In some cases, after the gates (update or reset) are computed, only a subset of the neurons in z_t are allowed to be active. This can be achieved by applying a sparsity-inducing function, such as:

$z_t = S(\sigma(W_z \cdot x_t + U_z \cdot h_{t-1} + b_z))$ (5)

Where $S(\cdot)$ induces sparsity by setting small values to zero, ensuring that only significant activations remain.

c) Classification Layers:

Following this, we introduce an enhanced single feedforward network that utilizes Extreme Learning Machines (ELM) concept for training the spatiotemporal features extracted from earlier layers. To achieve reduced computational complexity, this study employs an Extreme Learning Network with auto-tuning features, optimized according to TDO-inspired methods. The comprehensive details of the proposed classification layer are outlined below.

1) ELM Decision and Classification layer:

ELM is a type of neural network, which employs a single hidden layer and operates based on auto-tuning's principle. ELM demonstrates improved effectiveness, higher speed, and lower computational cost relative to alternative algorithms like SVM, Bayesian Classifiers, K-Nearest Neighbors, and Random Forests.

Its structure uses a single hidden layer where tuning is not compulsory. In contrast to the alternative algorithms like Support Vector Machine and Random Forest, ELM offers improved performance, faster processing, and reduced computational demands. ELM leverages kernel functions to achieve high accuracy and improved performance. The primary benefits of ELM include minimal training error and enhanced approximation due to its

auto-tuning of weight biases and non-zero activation functions. The comprehensive operational functioning of ELM is described in the input feature mappings of the ELM are represented as

$$X = F(F, P) \tag{6}$$

Here X denotes the fused spatio-temporal features derived from Convolutional Neural Network and Gated Recurrent Unit layers, F indicates CNN spatio features and P is the GRU temporal features

The ELM's output function is represented as

$$Y(n) = X(n)\beta = X(n)\, X^T(\frac{1}{C}XX^T)^{-1}O \tag{7}$$

The complete training procedure for ELM is represented as

$$S = \alpha(\sum_{n=1}^{N}(Y(n), B(n), W(n)) \tag{8}$$

Here X(n) represents the input fused feature maps, $\beta\ is$ temporal matrix that solved by Moore−Penrose generalized inverse theorem indicated by X^T, C represents constant, B and W are weights and bias factors of the model with sigmoidal activation function. The suggested system is trained using features with a sigmoidal activation function. To address computational issues, this paper incorporates TDO-inspired optimizers for adjusting the suggested ELM classifier's hyper parameters. The functioning of the TDO-inspired ELM is described below

d) The basics of TDO algorithm

The framework for the TDO algorithm is outlined in this segment. TDO is one of the most recent optimization techniques developed to mimic the natural behavior of the Tasmanian devil, particularly at the time of its foraging activities. The algorithm models the devil's strategies for finding food, which include either attacking and consuming live prey or scavenging from carcasses. TDO has been tested across 23 benchmark functions to assess its effectiveness and has been further applied to optimize four engineering design problems. Its effectiveness is validated through assessments against eight established optimization techniques, with results affirming its substantial competitive strength. The core phases of the TDO process are detailed below.

1) Initialization

TDO is similar to other population-based methods that begin with an iterative process-involving search agents known as Tasmanian devils. During this process, agents are randomly generated within the established search space. Each agent is represented as a vector, with its elements corresponding to the number of variables in the problem. This initial step can be expressed as follows:

$$X_{ij} = x_j^{min} + rand.(x_j^{max} - x_j^{min}), i = 1,2,\dots,M, j = 1,2,\dots,n \tag{9}$$

In the TDO algorithm, the value of " $rand$ " corresponds to a randomly assigned value following a uniform distribution between [0, 1]. The parameters $x_j^{min} and\ x_j^{max}$ define the minimum and maximum boundaries for the j-th dimension within the search space. Here, M refers to the Tasmanian devils population size, and n represents the variables count. Once the initialization phase is complete, the fitness of each candidate solution is measured using objective function (OF). The best-performing solution, based on OF, is considered the optimal member of the population and is updated during each iteration through the feeding strategies of Tasmanian devils. Each devil has a 50% probability of either scavenging carrion or hunting for food. Thus, during each iteration, one of these two strategies is randomly selected to upgrade the position of each Tasmanian devil.

2) Eating carrion: exploration stage

Tasmanian devils sometimes prefer scavenging for carrion rather than hunting live prey. In their habitat, other predators often hunt large prey but leave behind remains they cannot fully consume. Moreover, such predators could be unable to finish their meal before a Tasmanian devil appears. As a result, the devil takes advantage of these leftovers. This scavenging behavior mirrors the iterative approach in problem-solving algorithms, where Tasmanian devils' method of finding carrion mimics the TDO algorithm's search for an initial optimal solution. Each Tasmanian devil's location is akin to a potential solution in the search space, and other individuals in the population represent different carrion sources. The Tasmanian devil then selects one of these carrion locations at random as its target.

$$C_i = X_k, i = 1,2,\dots,M, k \in \{1,2,\dots,M | k \neq i\} \tag{10}$$

where C_i denotes the carrion chosen by the ith Tasmanian devil, where k is randomly selected between 1 and M.

Based on the selected carrion, the Tasmanian devil's updated position is determined as follows.

$$x_{i,j}^{\text{new},S1} = \begin{cases} x_{ij} + r \cdot (c_{ij} - I \cdot x_{ij}), & F_{C_i} < F_i \\ x_{ij} + r \cdot (x_{ij} - c_{ij}), & \text{otherwise} \end{cases} \quad (11)$$

$$X_i = \begin{cases} X_i^{\text{new},S1}, & F_i^{\text{new},S1} < F_i \\ X_i, & \text{otherwise} \end{cases} \quad (12)$$

In this context, $X_i^{\text{new},S1}$, $x_{i,j}^{\text{new},S1}$ represents updated location of the ith Tasmanian devil, calculated following the initial approach. The variable , $X_i^{\text{new},S1}$, $x_{i,j}^{\text{new},S1}$, refers to the specific element in the jth dimension. Additionally, F_{C_i} corresponds to the objective function value of the selected carrion. The parameter r indicates randomly generated integer between 0 and 1, and I denote randomly chosen integer, limited to the values 1 or 2.

3) Eating prey (exploitation phase)

At this phase, the Tasmanian devil engages in hunting and feeding on its prey, which occurs in two distinct phases. In the initial phase, it surveys the environment to identify potential prey that it can attack. Following this, at the second stage, once the Tasmanian devil approaches its target, it begins the chase, captures, and consumes the prey. The mathematical formulation of the first stage is described by Equations (13) to (15). Secondary, the Tasmanian devil's position is updated by considering the locations of other members in the population, which serve as the prey's positions. A random member (kth) is selected to represent the prey's location during this process of prey selection. This procedure is outlined as follows.

$$P_i = X_k, i = 1,2,\dots,M, k \in \{1,2,\dots,M | k \neq i\} \quad (13)$$

where P_i denotes ith Tasmanian devil chosen prey , k is chosen randomly between 1 and M.

Based on chosen prey, the subsequent step involves determining the updated location of the Tasmanian devil, which is outlined below.

$$x_{i,j}^{\text{new},S2} = \begin{cases} x_{ij} + r \cdot (p_{ij} - I \cdot x_{ij}), & F_{P_i} < F_i \\ x_{ij} + r \cdot (x_{ij} - p_{ij}), & \text{otherwise} \end{cases} \quad (14)$$

$$X_i = \begin{cases} X_i^{\text{new},S2}, & F_i^{\text{new},S2} < F_i \\ X_i, & \text{otherwise} \end{cases} \quad (15)$$

The updated location of the i[th] Tasmanian devil, denoted as $X_i^{\text{new},S2}$, represents its new location using the second strategy. The element corresponding to the jth dimension of this new position is referred to as $x_{i,j}^{\text{new},S2}$. The term Finew, S2represents the objective function value for the updated location, while F_{P_i} represents the chosen prey's objective function value.

In this process, the Tasmanian devil simulates chasing its prey within the targeted location. This behavior is captured through the mathematical formulations in Eqs. (13)– (15), which model the pursuit stage. Currently, the devil's location is regarded as surrounding area's center there it tracks its prey. The radius of this neighborhood, determined by Eq. (16), defines the zone within which the pursuit occurs. Updated Location, reflecting the Tasmanian devil's movement during chase, calculated using the Eq. (17).

$$R = 0.01(1 - t/T), \quad (16)$$

$$x_{i,j}^{\text{new}} = x_{ij} + (2r - 1) \cdot R \cdot x_{ij} \quad (17)$$

$$X_i = \begin{cases} X_i^{\text{new}}, & F_i^{\text{new}} < F_i \\ X_i, & \text{otherwise} \end{cases} \quad (18)$$

Here R specifies the radius of the attack zone. t and T represent the iteration counter and the maximum number of iterations, respectively. X_i^{new} Indicates the updated location of i-th Tasmanian devil within the vicinity, x_{ij} denotes the j-th component or variable of X_i, and F_i^{new} denotes the objective function new value.

In this context, R represents the neighborhood radius around the attack location, while t and T correspond to the current iteration and the maximum number of iterations, respectively. X_i^{new} refers to the updated location of iii-th Tasmanian devil in proximity to X_i, $x_{i,j}^{\text{new}}$denoting the j-th component or variable of X_i^{new}. Additionally, ithindicates the new value of the objective function for X_i^{new} . A visual representation of the TDO algorithm can be found in Figure 6.

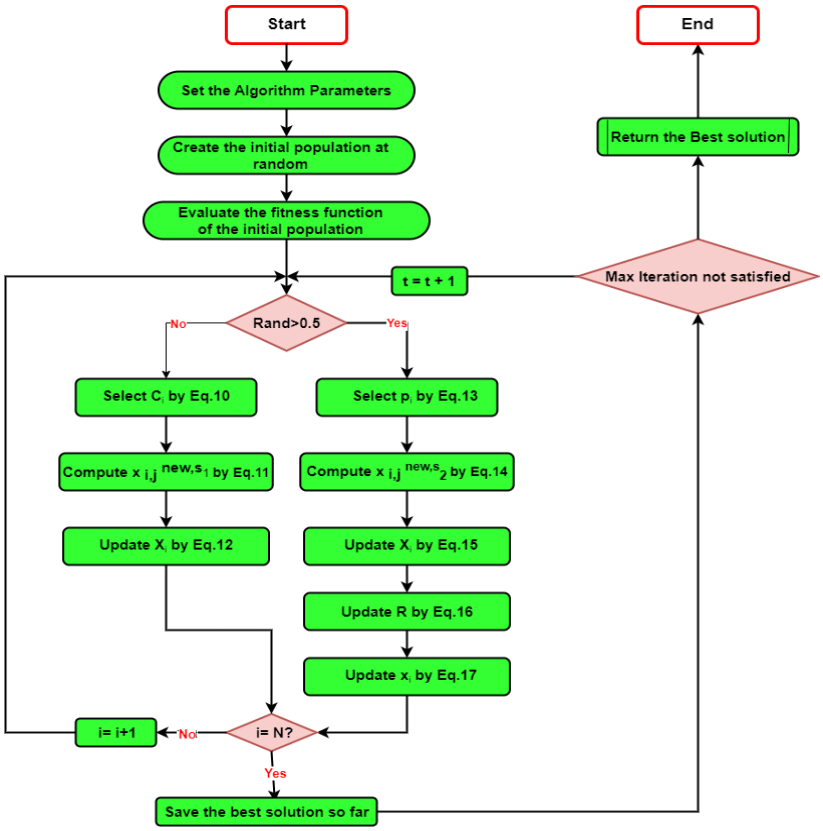


Figure 6. Flowchart of TDO

4) The proposed ITDO

In any optimization process, balancing between exploration and exploitation within the search space is essential for achieving a global optimum. Exploration, often referred to as diversification, involves a broad, global search, while exploitation, or intensification, focuses on refining the search locally around the best-known solution at a given time. Both excessive exploration and overemphasis on exploitation can hinder algorithm performance, leading to local optima entrapment and prolonged convergence times. Traditional Tasmanian Devil Optimization (TDO) relies on randomly selecting a position from the population to represent the prey's location during both exploration and exploitation phases. While this enhances exploration, it weakens exploitation by causing slow convergence and occasionally stagnation at local optima. To address these limitations, an enhanced version of TDO, termed Improved Tasmanian Devil Optimization (ITDO), incorporates two key strategies: (i) a Levy flight mechanism to bolster exploration, and (ii) a spiral elite learning method to enhance exploitation and accelerate convergence.

5) Levy flight strategy

The Levy distribution (LD) was first proposed by the French mathematician Levy as a model for random flight processes built on a probability distribution function. Its effectiveness in enhancing algorithm design has been demonstrated in several recent studies. The use of Levy flights can bolster an algorithm's performance, with small step sizes improving the algorithm's exploration capabilities and larger step sizes increasing the chance of escaping local or misleading optima. The formula for the Levy flight distribution is given below

$$L(s,\gamma,\mu) = \begin{cases} \sqrt{\frac{\gamma}{2\pi}} e^{\left(\frac{-\gamma}{2(s-\mu)}\right)} \left(\frac{1}{(s-\mu)^{3/2}}\right) & 0 < \mu < s < \infty \\ 0 & s \le 0 \end{cases} \quad (19)$$

Here's indicates samples, γ denotes the scale parameter to control the distribution and μ represents transmission factor. The Levy flight distribution is represented as

$$F(k) = e^{(-\alpha|k|^{\beta})}, \beta \in [0,2] \quad (20)$$

where β indicates a distribution index and α represents scaling parameter. The step length, s, is formulated as follows.

$$s = \left(\frac{u}{|v|^{(1/\beta)}}\right) \quad (21)$$

u and v are two parameters having Gaussian distributions as below:

$$u\ N(0, \sigma_u^2), v\ N(0, \sigma_v^2)$$

Where σ_u^2 and σ_v^2 define the standard deviations that are given as follows.

$$\sigma_u = \left\{\frac{\Gamma(1+\beta)\sin(\frac{\pi\beta}{2})}{\beta.\Gamma(\frac{1+\beta}{2})\cdot 2^{\frac{(\beta-1)}{2}}}\right\}^{1/\beta}, \sigma_v = 1 \quad (22)$$

where Γ (.) denotes the Gramma function.

At this phase, the Learning Factor (LF) is employed to refine the current solution, denoted as TDO, to enhance the search around the most optimal solution discovered up to now (x_"best") through the following formula.

$$x_i^{\mathrm{LF}} = x_i + (2 \cdot rand - 1) \cdot levy(\beta) \cdot (x_{\mathrm{best}} - x_i) \quad (23)$$

$$x_i = \begin{cases} x_i^{\mathrm{LF}} & F(x_i^{\mathrm{LF}}) < F(x_i) \\ x_i & \text{otherwise} \end{cases} \quad (24)$$

In Equation (23), x_i^{LF} represents the new position generated through the Levy flight process. Here, xi denotes present solution, and rand refers to a random integer selected within the range [0,1]. According to the equation, if the updated position obtained via the Levy flight analysis is superior to present solution, it will replace it. Otherwise, the current solution will remain as is.

6) Spiralize elite learning strategy

Traditional TDO updates its solutions based on the positions of two randomly generated candidates, which might result in missing potentially beneficial regions. Therefore, it is crucial for enhancing the exploitation capability by conducting additional investigation of the area surrounding the current best solution. To address this, a logarithmic spiral step (LSS) is incorporated close to optimal solution so that the step size diminishes progressively as the iterations increase.

$$\mathrm{LSS} = Ae^{-b\cdot t}\cos(2c\pi t) \quad (25)$$

$$A = \left(\frac{U_i - L_i}{2}\right) * \left(\frac{T-t}{T}\right)^2, U_i = max(X_i), L_i = min(X_i) \quad (26)$$

$$x_i^{\mathrm{LSS}} = x_{\mathrm{best}} + \mathrm{LSS} \quad (27)$$

In the context of the logarithmic spiral step, x_i^{LSS} represents the newly generated position. Here, a defines the radius of reduction, b is a fixed parameter that influences the form of logarithmic spiral, t represents the parameter which controls the proximity of the subsequent position to optimal solution, and c indicates total spirals.

7) TDO-Inspired ELM layers

As detailed in Section 3.3.4, basic TDO system are employed to optimize the weights of ELM networks. During this scenario, the TDO mechanism serves as the primary method for optimizing the weights and hidden layers of ELM. At first, such hyperparameters are chosen randomly and input into the ELM training network. The fitness function for suggested system is described in equation (28). During each iteration, hyperparameters were computed utilsing equations (26) and (27). The process continues until the fitness function satisfies the eqn (28).

$$\boldsymbol{Fitness\ Function\ = Max(Accuracy, Precison, REcall, Specificty\ \&\ F1 - Score)} \quad (\mathbf{28})$$

After the input weights are optimized using the TDO approach, the suggested classification layer performs gait activity classification attaining improved efficiency with minimal computational cost. The operational working of suggested classification layers is outlined in Algorithm 1. The training network employs 30 epochs, a batch size of 40, 150 hidden layers, and a learning rate of 0.001.

Steps	**Algorithm- 1 // Pseudo Code for the Proposed Optimized ELM layers**
01	Input = Bias and weights, Hidden layers, Epochs, Learning Rate
02	Output : Gaits/Human Activity Recognition
03	Randomly allocate bias weights, hidden layers, epochs, learning rate
04	Initialize the Loudness, Frequency, Distance, No of devils and Velocity
05	While (true)

```
06          Evaluate the ELM ‘s output  using equation(7)
07              Calculate the Fitness function  using the  equation(28)
08           For t=1 to Max_iteration
09                                  Assign the  bias weights and input  layers by equation (6)and (7)
10                              Calculate the fitness function  using equation(28)
11                                  If (Fitness function = = Maximum Accuracy)
12                                      Go to Step 17
13                                  Else
14                                      Go to Step 08
15          End
16          End
17            If (output value <=1)
18             / Normal Activity is determined
19           Else if(output value <=2&& output value >1)
20          /  Activity-2 is determined
21          Else if(output value <=3&& output value >2)
22          /  Activity-3 is determined
23          Else
26            Go to step 08
27          End
28          End
29          End
```

e) Explainable AI

Explainable AI (XAI) has gained significant attention in recent years as AI systems become increasingly integrated into critical fields like healthcare. XAI refers to a set of methods and techniques that make AI models more interpretable and understandable to humans, especially when dealing with complex, black-box models such as deep learning networks. This transparency is particularly crucial in healthcare, where clinicians and decision-makers need to understand the rationale behind AI-driven predictions or recommendations.

By leveraging XAI techniques such as SHAP (Shapley Additive exPlanations), our model provides clear insights into the factors contributing to its decisions, making it easier for healthcare professionals to interpret the results. This transparency is critical in clinical decision-making, where the ability to justify predictions can significantly influence patient care and trust in AI systems.

4. Results and Discussion

A. Experimentation and Evaluation Metrics

To evaluate the proposed model's performance, participants walked on treadmills equipped with IoT devices that collected gait data in real-time. This setup simulates the movement tracking typically employed in clinical mobility assessments, allowing for continuous monitoring, which is essential for detecting gait abnormalities during everyday activities in healthcare environments. Table 1 presents the experimental parameters used for training the proposed network. Furthermore, we have calculated the performance metrics such as accuracy, precision, recall, specificity and F1-score using different datasets. Additionally, we have calculated the AUC (Area under ROC) and confusion matrix to prove superiority of the proposed model. The mathematical expression used for calculating the performance metrics is presented in table II. Higher scores of the metrics indicates the better performances. To solve the network’s overfitting problem and improve the generalization problem, early stopping method is used in the paper. This method can be used to end the proposed network training when the validation performance shows no improvement for N consecutive times. The complete model was developed using opensource

TensorFlow version 2.1.0 with keras as backend and implemented on a PC workstation with Intel Xeon CPU, NVIDIA Titan GPU, 16GB RAM and 3.5 GHZ operating frequency

Table 1: Training Parameters used for the Proposed Hybrid Model

Sl.no	Detailed parameters	Specifications
01	No of Epochs	200
02	Batch Size	100
03	Learning Rate	0.0001
04	Training data	70
05	Testing data	30

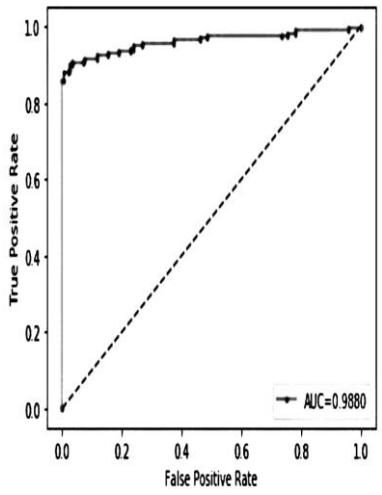
Table 2: Mathematical Expressions for the Performance Metrics' Calculation

SL.NO	Performance Metrics	Mathematical Expression
01	Accuracy	$\frac{TP+TN}{TP+TN+FP+FN}$
02	Recall	$\frac{\mathrm{TP}}{\mathrm{T\,P+FN}}$ x100
03	Specificity	$\frac{TN}{TN+FP}$
04	Precision	$\frac{TN}{TP+FP}$
05	F1-Score	$2.\frac{Precison*Recall}{Precision+Recall}$

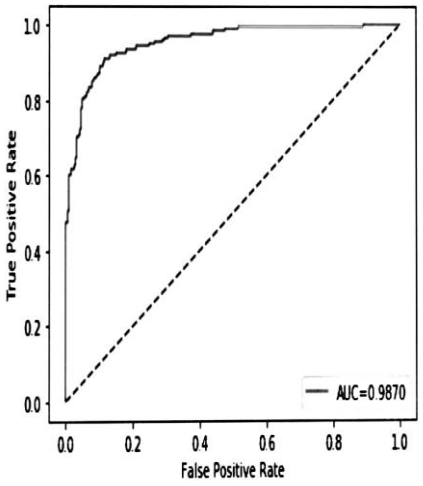
TP is True Positive Values, TN is True Negative Values, FP is False Positive and FN is false negative values

B. Performance Evaluation of the Proposed Model utilizing the distinct datasets:

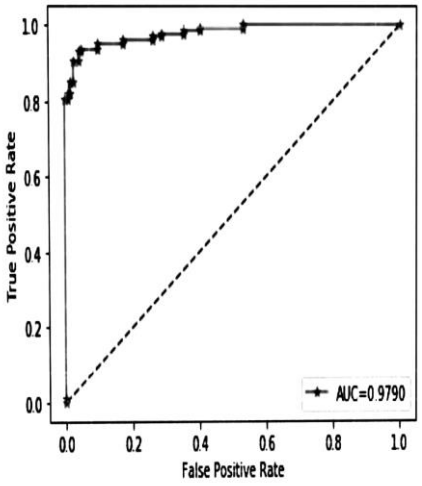
In this section, we performed evaluations utilizing both real-time and benchmark datasets. We computed the ROC curve and confusion matrix for suggested algorithm across various datasets.



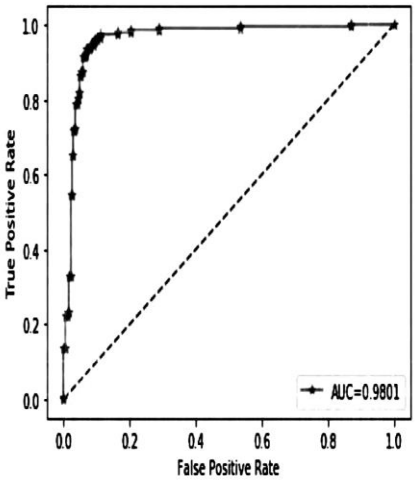
(a)



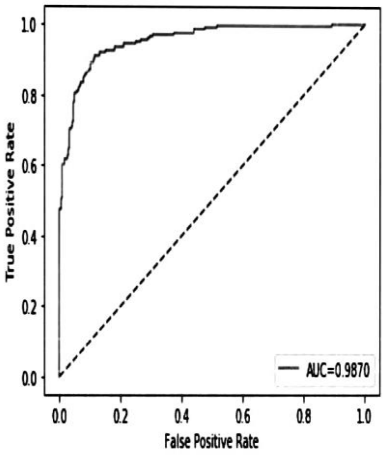
(b)



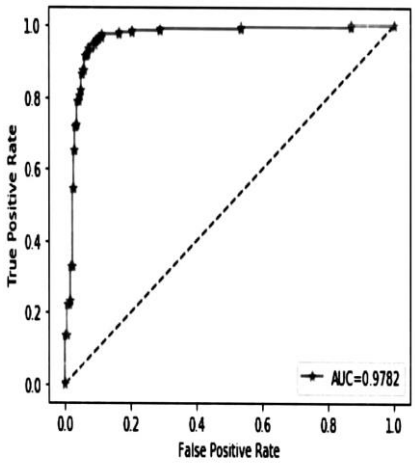
(c)



(d)



(e)



(f)

Figure 7. ROC curves of the suggested model a) Real time Datasets b) Dataset-1(whuGait) c) Dataset-2 d) Dataset-3 e) Dataset-4 f) OU-ISIR datasets.

Figure 7 illustrates the ROC curves for the suggested algorithm across various gait datasets. This indicates that the method achieves an AUC of 0.9880 for the raw data, 0.980 for the whuGait dataset, and 0.9780 for the OU-ISIR dataset. Which indicating that the suggested method consistently performs well on both real-time and public datasets. Figure 8 presents the confusion matrix for suggested method across distinct datasets. The figures demonstrate that the model maintains uniform performance across various datasets.

Figure 7 outlines the ROC curves for the suggested algorithm with real time datasets and with alternative datasets. Table 3 summarizes the performance metrics of the proposed algorithm across various datasets. The table reveals that the network performs exceptionally well with real-time and whuGait datasets. This is due to the higher quality and more diverse gait recordings. These datasets also include larger sample sizes, which improve the training process of the model and enhance overall accuracy. Additionally, the model exhibits a small improvement in results while handling the OU-ISIR datasets.

	1	2	3	4	5	6	7	8	9	10
	98.8%	0	0	1	0.5	0.4	0	0	0	0
	0	98.90%	0	0	0.4	0.5	0	0	0	0
	0	0	98.89%	1	0.5	0.4	0	0	0	0
	0	0	0	98.89%	0.5	0.4	0	0	0	0
	0	0	0	0	98.90%	0.4	0	0	0	0
	0	0	0	0	0.5	98.89%	0	0	0	0
	0	0	0	0	0.5	0.4	98.9%	0	0	0
	0	0	0	0	0.5	0.4	0	99.0%	0	0
	0	0	0	0	0.5	0.4	0	0	98.89%	0
	0	0	0	0	0.5	0.4	0	0	0	98.91%

(a)

	1	2	3	4	5	6	7	8	9	10
	98.8%	0	0	1	0.5	0.4	0	0	0	0
	0	98.90%	0	0	0.4	0.5	0	0	0	0
	0	0	98.89%	1	0.5	0.4	0	0	0	0
	0	0	0	98.89%	0.5	0.4	0	0	0	0
	0	0	0	0	98.90%	0.4	0	0	0	0
	0	0	0	0	0.5	98.89%	0	0	0	0
	0	0	0	0	0.5	0.4	98.9%	0	0	0
	0	0	0	0	0.5	0.4	0	99.0%	0	0
	0	0	0	0	0.5	0.4	0	0	98.89%	0
	0	0	0	0	0.5	0.4	0	0	0	98.91%

(b)

	1	2	3	4	5	6	7	8	9	10
	98.8%	0	0	1	0.5	0.4	0	0	0	0
	0	98.90%	0	0	0.4	0.5	0	0	0	0
	0	0	98.89%	1	0.5	0.4	0	0	0	0
	0	0	0	98.89%	0.5	0.4	0	0	0	0
	0	0	0	0	98.90%	0.4	0	0	0	0
	0	0	0	0	0.5	98.89%	0	0	0	0
	0	0	0	0	0.5	0.4	98.9%	0	0	0
	0	0	0	0	0.5	0.4	0	99.0%	0	0
	0	0	0	0	0.5	0.4	0	0	98.89%	0
	0	0	0	0	0.5	0.4	0	0	0	98.91%

(c)

	1	2	3	4	5	6	7	8	9	10
	98.8%	0	0	1	0.5	0.4	0	0	0	0
	0	98.90%	0	0	0.4	0.5	0	0	0	0
	0	0	98.89%	1	0.5	0.4	0	0	0	0
	0	0	0	98.89%	0.5	0.4	0	0	0	0
	0	0	0	0	98.90%	0.4	0	0	0	0
	0	0	0	0	0.5	98.89%	0	0	0	0
	0	0	0	0	0.5	0.4	98.9%	0	0	0
	0	0	0	0	0.5	0.4	0	99.0%	0	0
	0	0	0	0	0.5	0.4	0	0	98.89%	0
	0	0	0	0	0.5	0.4	0	0	0	98.91%

(d)

	1	2	3	4	5	6	7	8	9	10
	98.8%	0	0	1	0.5	0.4	0	0	0	0
	0	98.90%	0	0	0.4	0.5	0	0	0	0
	0	0	98.89%	1	0.5	0.4	0	0	0	0
	0	0	0	98.89%	0.5	0.4	0	0	0	0
	0	0	0	0	98.90%	0.4	0	0	0	0

0	0	0	0	0.5	98.89%	0	0	0	0
0	0	0	0	0.5	0.4	98.9%	0	0	0
0	0	0	0	0.5	0.4	0	99.0%	0	0
0	0	0	0	0.5	0.4	0	0	98.89%	0
0	0	0	0	0.5	0.4	0	0	0	98.91%

(e)

Label	Recognition of Activity	Non -recognition of Activity
Recognition of Activity	98.9%	1.2%
Non -recognition of Activity	1.3%	98.7%

(f)

Figure 8. Confusion matrix for the suggested Hybrid Methodology a) Real time Datasets b) Dataset-1(whuGait) c) Dataset-2 d) Dataset-3 e) Dataset-4 f) OU-ISIR datasets.

Table 3: Performance metrics of the suggested methodology utilizing Distinct Datasets

Datasets	Performance Metrics				
	Accuracy	Precision	Recall	Specificity	F1-score
Real time Datasets	0.989	0.987	0.986	0.989	0.9902
Datasets-1	0.9889	0.985	0.984	0.978	0.983
Dataset-2	0.9890	0.9856	0.989	0.9902	0.989
Dataset-3	0.9890	0.9879	0.990	0.9901	0.992
Dataset-4	0.9891	0.9890	0.990	0.99	0.988
OU-ISIR datasets	0.990	0.989	0.982	0.992	0.990

C. Comparative Analysis of the Proposed Model with other existing models:

The suggested model's effectiveness is assessed against existing hybrid DL models, including LSTM , ATTENTION , RNN+CNN, GRU, and CNN+GRU, to establish the algorithm's superiority.

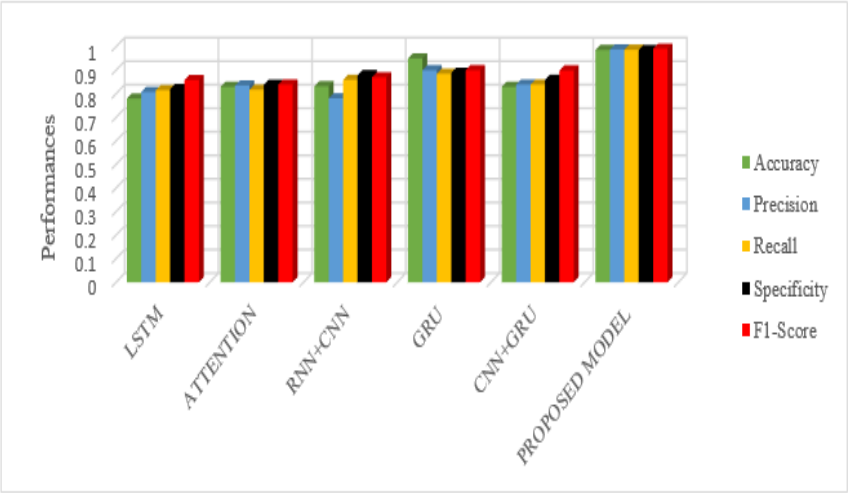


Figure 9. Comparative Assessment of Distinct Hybrid Algorithms utilizing Real time Datasets

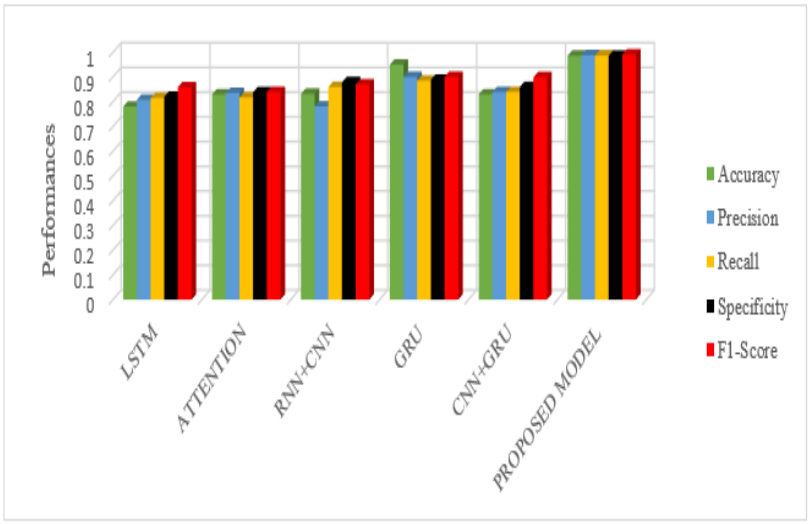


Figure 10. Comparative Assessment of Distinct Hybrid Methods utilizing WhGait Datasets-1

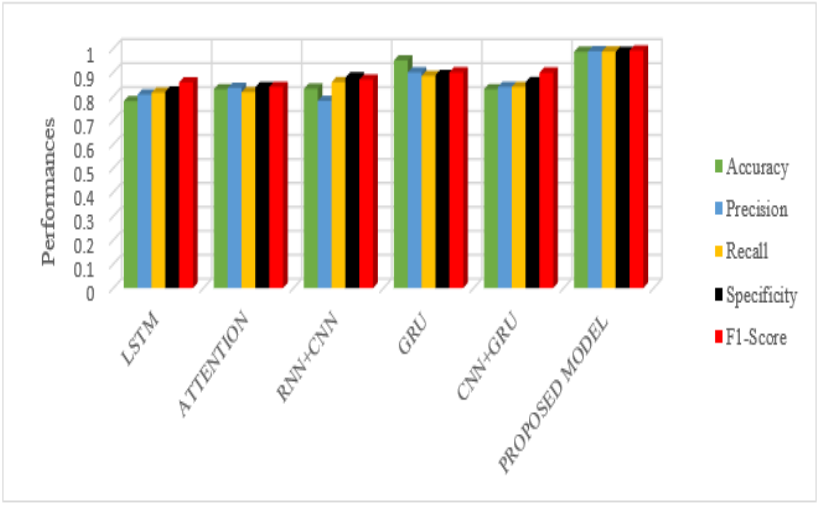


Figure 11. Comparative Assessment of Distinct Hybrid Models utilizing WhGait Datasets-2

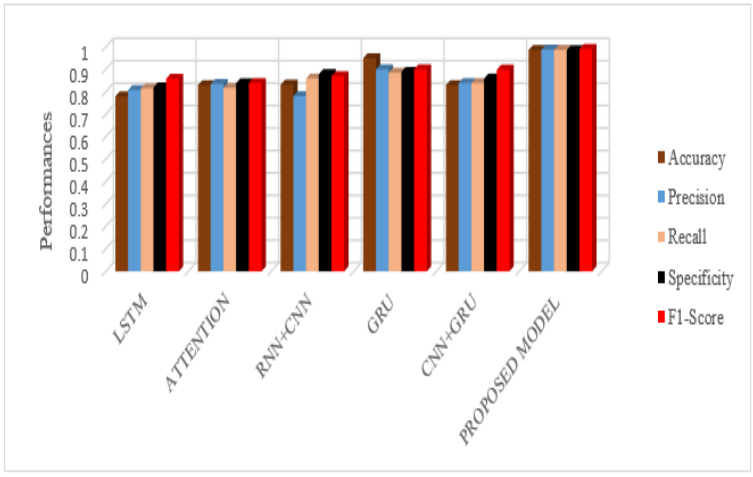


Figure 12. Comparative Assessment of Distinct Hybrid Models utilizing WhGait Datasets-3

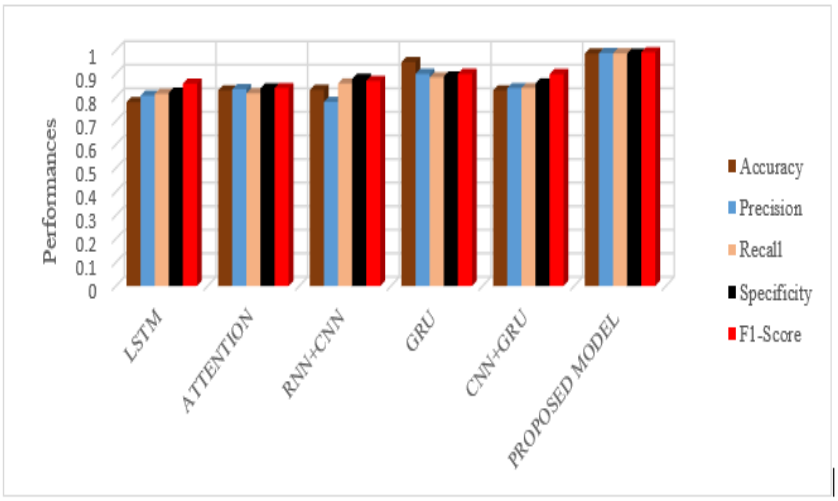


Figure 13. Comparative Assessment of Distinct Hybrid Models utilizing WhGait Datasets-4

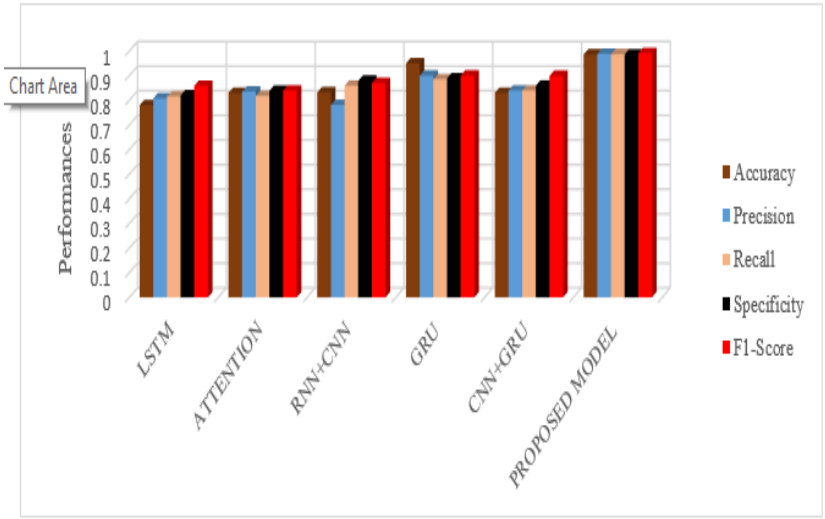


Figure 14. Comparative Assessment of Distinct Hybrid Models utilizing OU-ISIR datasets

Figure 9 illustrates the performance of other hybrid models utilizing real-time IoT datasets. The analysis reveals that the suggested hybrid algorithm and Sparse GRU demonstrate identical performance levels in AR detection systems. Nevertheless, the suggested algorithm exhibits a slight advantage compared to Sparse Gated Recurrent Unit model and surpasses in gait identification. Figures 10 through 13 provide a comparative evaluation of the distinct algorithms utilising WhGait datasets. The suggested algorithm consistently shows superior performance compared to other existing models. Figure 14 depicts the performance of various models using OU-ISIR datasets. From Figures 9 to 14, it is evident that incorporating TDO-inspired ELM methods together with spatio-temporal feature extraction achieves superior results compared to other models. The experiments indicate that the proposed model achieves a higher AR rate across multiple datasets.

D. Computational Complexity

The computational complexity of the suggested method is expressed using Big-O notation. The mathematical formulas for determining processing overhead with Big-O notation are provided as

Time Complexity = O(Convolutaional layers* Pooling Layers *Training Networks) (29)

Table 4: Computational Complexity Assessment between the Hybrid and Suggested Method.

Algorithm Details	Layers required	Big-O-Notations
CNN(without Optimization)	Convolutional Layers count =6 Polling layers count=06 Training layers count=k	$O(n^6,n^6,n^k)$
GRU(without Optimization)	GRU Layers count = 6 Pooling layers count = 6 Training layers count = k	$O(n^6,n^k)$
2DCNN-LSTM	Convolutional Layers count =05 Polling layers count =05 Training layers count =k+3	$O(n^5,n^5,n^{k+3})$
CNN-SVM	Convolutional Layers count =06 Polling layers count =06 Training layers count =k+5	$O(n^6,n^6,n^{k+5})$
CNN+GRU	Convolutional Layers count =06 Polling layers count =06 Training layers count =k+6	$O(n^6,n^6,n^{k+6})$
ATTENTION CNN	Convolutional Layers count =06 Polling layers count =06 Training layers count =k+10	$O(n^6,n^6,n^{k+10})$
Proposed architecture	Convolutional Layers count =06 Polling layers count =06 Training layers count =k-5	$O(n^5,n^5,n^{k-5})$

*k = Upper Limit of Necessary Training Layers

Table IV reveals that the TDO-optimized classification layer exhibits lower computational complexity, being 10% less than that of the existing models, making it more efficient than standard models like CNNs and GRUs.

5. Conclusion and Future Scope

In this paper, a novel Sparse GRU-based fused feature extractor combined with a TDO-inspired classification layer is developed to enhance the detection of human gait patterns, particularly for healthcare applications. Real-time datasets were collected via WIoT devices and stored in the cloud for ongoing assessment and analysis. To enhance classification efficiency, the data were reorganized using the Pearson Correlated Sliding Window technique. This restructured data was processed through two layers of the deep learning algorithm: a custom CNN for extracting spatial features and a Sparse GRU for capturing temporal characteristics. The spatio-temporal features were then input into the TDO-inspired optimized classifiers, achieving superior gait recognition. Extensive experiments conducted on both real-time and public datasets, including the whuGait and OU-ISIR benchmarks, demonstrated that the proposed model significantly improves accuracy and computational efficiency, consistently outperforming existing hybrid learning methods. The model's high performance and reduced computational overhead make it especially suitable for real-time healthcare applications.

Future work will emphasize implementing gait recognition system on devices with limited hardware resources, including smartphones. This will involve optimizing the algorithms to run efficiently with lower memory and processing power, ensuring that real-time gait analysis remains accurate and responsive even on hardware-constrained platforms. Beyond performance metrics, additional factors including energy consumption, resource constraints, and computational capabilities will be considered for real-world applications. Additionally, the gait recognition model could be expanded to predict human behaviors, offering potential applications in psychology and crime investigation.

References

[1] J. Bravo, L. Fuentes, and D. L. de Ipina, "Theme issue: Ubiquitous computing and ambient intelligence," *Pers. Ubiquitous Comput.*, vol. 15, no. 3, pp. 315–316, 2011. doi: 10.1007/s00779-010-0358-9.

[2] A. Rahman, M. S. Hossain, G. Muhammad, et al., "Federated learning-based AI approaches in smart healthcare: Concepts, taxonomies, challenges, and open issues," *Cluster Comput.*, vol. 26, pp. 2271–2311, 2023. doi: 10.1007/s10586-022-03658-4.

[3] J. Medina-Quero, S. Zhang, C. Nugent, and M. Espinilla, "Ensemble classifier of long short-term memory with fuzzy temporal windows on binary sensors for activity recognition," *Expert Syst. Appl.*, vol. 114, pp. 441–453, 2018. doi: 10.1016/j.eswa.2018.07.068.

[4] R. Ali-Hamad, A. Salguero, M. H. Bouguelia, M. Espinilla, and M. Medina-Quero, "Efficient activity recognition in smart homes using delayed fuzzy temporal windows on binary sensors," *IEEE J. Biomed. Health Inform.*, 2019. doi: 10.1109/JBHI.2019.2918412.

[5] F. J. Ordonez and D. Roggen, "Deep convolutional and lstm recurrent neural networks for multimodal wearable activity recognition," *Sensors*, vol. 16, no. 1, pp. 115, 2016. doi: 10.3390/s16010115.

[6] M. Kose, O. D. Incel, and C. Ersoy, "Online human activity recognition on smartphones," in *Proc. 2nd Int. Workshop Mobile Sensing*, Beijing, China, Apr. 2012.

[7] C. A. Ronao and S. B. Cho, "Deep convolutional neural networks for human activity recognition with smartphone sensors," in *Proc. Int. Conf. Neural Inf. Process.*, Istanbul, Turkey, Nov. 2015, pp. 46–53. doi: 10.1007/978-3-319-26561-2_6.

[8] D. Anguita, A. Ghio, L. Oneto, X. Parra, and J. L. Reyes-Ortiz, "A public domain dataset for human activity recognition using smartphones," in *Proc. European Symp. Artif. Neural Networks*, Bruges, Belgium, Apr. 2013.

[9] R. San-Segundo, H. Blunck, J. Moreno-Pimentel, A. Stisen, and M. Gil-Martn, "Robust human activity recognition using smartwatches and smartphones," *Eng. Appl. Artif. Intell.*, vol. 72, pp. 190–202, 2018. doi: 10.1016/j.engappai.2018.04.002.

[10] F. Attal, S. Mohammed, M. Dedabrishvili, F. Chamroukhi, L. Oukhellou, and Y. Amirat, "Physical human activity recognition using wearable sensors," *Sensors*, vol. 15, no. 12, pp. 31314–31338, 2015. doi: 10.3390/s151229858.

[11] A. Malaisé, P. Maurice, F. Colas, F. Charpillet, and S. Ivaldi, "Activity recognition with multiple wearable sensors for industrial applications," in *Proc. 11th Int. Conf. Advances in Computer-Human Interactions*, Rome, Italy, Mar. 2018.

[12] S. Allan, H. Henrik, S. Sourav, T. S. P., M. B. K., A. K. D., T. S., and M. J. Möller, "Smart devices are different: Assessing and mitigating mobile sensing heterogeneities for activity recognition," in *Proc. 13th ACM Conf. Embedded Networked Sensor Systems*, New York, NY, USA, Oct. 2015, pp. 127–140. doi: 10.1145/2809695.2809718.

[13] A. Subasi, M. Radhwan, R. Kurdi, and K. Khateeb, "IoT-based mobile healthcare system for human activity recognition," in *Proc. 15th Learning and Technology Conf. (L&T)*, Jeddah, Saudi Arabia, Feb. 2018, pp. 29–34. doi: 10.1109/LT.2018.8368507.

[14] M. Kheirkhahan, S. Nair, A. Davoudi, P. Rashidi, A. A. Wanigatunga, D. B. Corbett, T. Mendoza, T. M. Manini, S. Ranka, "A smartwatch-based framework for real-time and online assessment and mobility monitoring," *J. Biomed. Inform.*, vol. 89, pp. 29–40, 2019. doi: 10.1016/j.jbi.2018.11.003.

[15] K. M. Masum, A. Barua, E. H. Bahadur, M. R. Alam, M. A. U. Z. Chowdhury, and M. S. Alam, "Human activity recognition using multiple smartphone sensors," in *Proc. 2018 Int. Conf. Innovations in Science, Engineering and Technology (ICISET)*, Chittagong, Bangladesh, Oct. 2018, pp. 468–473. doi: 10.1109/ICISET.2018.8745628.

[16] Y. Tian, X. Wang, W. Chen, Z. Liu, and L. Li, "Adaptive multiple classifiers fusion for inertial sensor-based human activity recognition," *Clust. Comput.*, vol. 22, 2019. doi: 10.1007/s10586-017-1648-z.

[17] A. Peinado-Contreras and M. Munoz-Organero, "Gait-based identification using deep recurrent neural networks and acceleration patterns," *Sensors*, vol. 20, no. 23, pp. 6900, 2020. doi: 10.3390/s20236900.

[18] W. Shen and T. Tan, "Automated biometrics-based personal identification," *Proc. Natl. Acad. Sci. USA*, vol. 96, no. 20, pp. 11065–11066, 1999. doi: 10.1073/pnas.96.20.11065.

[19] R. He, X. Wu, Z. Sun, and T. Tan, "Wasserstein CNN: Learning invariant features for NIR-VIS face recognition," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 41, no. 8, pp. 1761–1773, 2018.

[20] K.-T. Nguyen, T.-L. Vo-Tran, D.-T. Dinh, and M.-T. Tran, "Gait recognition with multi-region size convolutional neural network for authentication with wearable sensors," in *Proc. Int. Conf. Future Data and Security Engineering (FDSE)*, Ho Chi Minh City, Vietnam, Nov. 2017, pp. 14–23. doi: 10.1007/978-3-319-70004-5_14.

[21] A. Alharbi, K. Equbal, S. Ahmad, H. Ur Rahman, and H. Alyami, "Human gait analysis and prediction using the Levenberg-Marquardt method," *Hindawi J. Healthcare Eng.*, vol. 2021, Article ID 5541255, 11 pages, 2021. doi: 10.1155/2021/5541255.

[22] A. M. Saleh and T. Hamoud, "Analysis and best parameters selection for person recognition based on gait model using CNN algorithm and image augmentation," *J. Big Data*, vol. 8, no. 1, 2021. doi: 10.1186/s40537-020-00387-6.

[23] J. Moon, N. A. Le, N. H. Minaya, and S.-I. Choi, "Multimodal few-shot learning for gait recognition," *Appl. Sci.*, vol. 10, no. 24, pp. 7619, 2020. doi: 10.3390/app10217619.

[24] W. Jiang and Z. Yin, "Human activity recognition using wearable sensors by deep convolutional neural networks," in *Proc. 23rd ACM Int. Conf. Multimedia*, 2015, pp. 1307–1310.

[25] G. Laput and C. Harrison, "Sensing fine-grained hand activity with smartwatches," in *Proc. 2019 CHI Conf. Human Factors Comput. Syst.*, 2019, pp. 338.

[26] S. Ha and S. Choi, "Convolutional neural networks for human activity recognition using multiple accelerometer and gyroscope sensors," in *2016 Int. Joint Conf. Neural Networks*, pp. 381–388.

[27] Y.-H. Shen, K.-X. He, and W.-Q. Zhang, "SAM-GCNN: A gated convolutional neural network with segment-level attention mechanism for home activity monitoring," in *2018 IEEE Int. Symp. Signal Process. and Inf. Technol. (ISSPIT)*, pp. 679–684.

[28] H. Guo, L. Chen, L. Peng, and G. Chen, "Wearable sensor-based multimodal human activity recognition exploiting the diversity of classifier ensemble," in *Proc. 2016 ACM Int. Joint Conf. Pervasive and Ubiquitous Comput.*, 2016, pp. 1112–1123.

[29] Y. Yuki, J. Nozaki, K. Hiroi, K. Kaji, and N. Kawaguchi, "Activity recognition using DualConvLSTM extracting local and global features for SHL recognition challenge," in *2018 ACM Int. Joint Conf. and 2018 Int. Symp. Pervasive and Ubiquitous Comput. and Wearable Comput.*, 2018, pp. 1643–1651. doi: 10.1145/3267305.3267533.

[30] B. Guo, U. Chen, D. Zhang, and Z. Yu, "Deep learning for sensor-based human activity recognition: Overview, challenges and opportunities," *ACM Computing Surveys*, vol. 37, no. 4, Article 111, Aug. 2018. doi: 10.1145/1122445.1122456.

[31] U. T. Kumbhar, R. Phursule, V. C. Patil, R. K. Moje, O. R. Shete, and M. A. Tayal, "Explainable AI-powered IoT systems for predictive and preventive healthcare: A framework for personalized health management and wellness optimization," *J. Electrical Systems*, vol. 19, no. 3, pp. 23–31, 2023. doi: 10.52783/jes.648.

[32] M. H. Wang, K. K. Lung Chong, Z. Lin, X. Yu, and Y. Pan, "An explainable artificial intelligence-based robustness optimization approach for age-related macular degeneration detection based on medical IoT systems," *Electronics*, vol. 12, no. 12, pp. 2267, 2023. doi: 10.3390/electronics12122697.

[33] X.-S. Yang and X. He, "Bat algorithm: Literature review and applications," *Int. J. Bio-Inspired Comput.*, vol. 5, no. 3, pp. 141–149, 2013. doi: 10.1504/IJBIC.2013.055093.