Intelligent Multi-Gait Identification using Greedy Tree Learning on Streaming Wearable Sensor Data

P. Babu[1], Ch. Nikhitha[2], D. Kavya[2], G. Haneesha[2], I. Lakshmi Sudha[2]

[1]Professor, [2]UG Student, [1,2]Department of Computer Science and Engineering (AI & ML)

[1,2]Geethanjali Institute of Science and Technology, Nellore-Bombay Highway, S.P.S.R, Andhra Pradesh 524137, India

To Cite this Article
P. Babu, Ch. Nikhitha, D. Kavya, G. Haneesha, I. Lakshmi Sudha, "Intelligent Multi-Gait Identification using Greedy Tree Learning on Streaming Wearable Sensor Data", Journal of Science Engineering Technology and Management Science, Vol. 03, Issue 04, April 2026,pp: 961-973, DOI: http://doi.org/10.64771/jsetms.2026.v03.i04.pp961-973
Submitted: 08-03-2026 Accepted: 16-04-2026 Published: 23-04-2026

ABSTRACT

Gait is the locomotion attained through the movement of limbs, and gait analysis examines the patterns (normal/abnormal) depending on the gait cycle. It contributes to the development of various applications in the medical, security, sports, and fitness domains to improve the overall outcome. Among many available technologies, two emerging technologies that play a central role in modern-day gait analysis are wearable sensors, which provide a convenient, efficient, and inexpensive way to collect data, and Machine Learning Methods (MLMs) which enable high-accuracy gait feature extraction for analysis. The dataset comprises continuous recordings from inertial measurement units (IMUs) capturing acceleration, angular velocity, and orientation across multiple participants. Existing approaches, including Adaptive Boosting (AdaBoost), K-Nearest Neighbor (KNN), Logistic Regression (LR), and Naive Bayes (NB) classifiers, serve as baseline models for gait detection. The proposed algorithm is Greedy Tree Classifier. The output is a multi-class gait activity label, including walking, running, climbing, and other activities. Integration into a Flask-based web application enables real-time streaming, preprocessing, and activity prediction for wearable devices. Experimental evaluation shows that the proposed approach outperforms existing classifiers in terms of accuracy, F1-score, and robustness across long-duration recordings. This framework supports continuous monitoring in healthcare, sports, and rehabilitation applications, enabling reliable activity tracking, early detection of mobility issues, and personalized interventions.

Keywords: Greedy Tree Classifier, Multi-Gait Activity Detection, Gait Analysis, Human Activity Detection.

1. INTRODUCTION

Gait analysis has emerged as a cornerstone in modern medical diagnostics, rehabilitation, and performance optimization. For medical professionals, understanding the intricacies of gait analysis is essential for identifying movement disorders, improving patient outcomes, and advancing research in bio-mechanics. In healthcare, gait analysis is indispensable for diagnosing and treating a wide range of conditions:

- **Neurological Disorders:** Parkinson's disease, multiple sclerosis, and stroke rehabilitation benefit from gait assessments.
- **Orthopaedic Conditions:** Identifying issues like hip dysplasia, scoliosis, or joint degeneration.
- **Paediatric Applications:** Monitoring developmental milestones and addressing conditions like cerebral palsy.
- **Post-Surgical Recovery:** Tracking progress and optimizing rehabilitation after procedures like knee replacements.

Gait analysis is a systematic study of human walking patterns to identify abnormalities and inefficiencies in movement. Gait semiology is of major importance in neurological practice, as abnormalities are associated with high comorbidities. The quantification of gait using inertial measurement units (IMUs) has become a democratic method for the follow-up of subjects with locomotion alterations in healthcare.

Table 1. Different types of wearable sensors devices:

Category	**Examples**	**Key Sensors Used**
Health monitoring	Smartwatches, ECG bands	HR, SpO_2, ECG, GPS
Activity tracking	Smart shoes, IMU devices	IMU, pressure sensors
Environmental	Air quality bands, UV monitors	UV, gas, temperature
Brain/Nerve	EEG headbands, EMG straps	EEG, EMG
AR/VR	Smart glasses, VR gear	Eye tracking, IMU
Medical	Smart patches	Biosensors (ECG, glucose)

The use of such embedded technologies has already shown its usefulness in the detection of postural strategies during walking, partitioning gait during the stance phase or motor supple mentation for switch-activated simulators. However, these clinical applications require the detection of steps within the IMU signals. Table 1 shows different types of wearable devices.

Wearable sensor devices represent state-of-the-art body-mounted gadgets that continuously monitor health, activity, and environmental conditions. Popular devices like smartwatches and fitness bands characterize physical activity and vital signs using heart rate, SpO_2, ECG, accelerometers, gyroscope, and GPS sensors. Medical wearables empower remote patient monitoring and disease management, including continuous glucose monitors, ECG patches, and smart health patches. Wearables track movement and gait, including smart shoes and posture bands, using IMU and pressure sensors. Wearables targeting the brain and muscles, like EEG and EMG bands, monitor stress, brain signals, and muscle activity. Environmental wearables, such as UV monitors and air-quality sensors, issue alerts on external conditions. The integration of flexible sensors into garments for tracking posture, respiration, and body temperature characterizes smart clothing and e-textiles. Collectively, these devices generate real-time data from various forms of wearables that improve personal health, safety, and lifestyle.

Fig 1 shows the Sural is sensor system in sensor sock, to synchronize the data, the sensor sock and the IMU are wirelessly connected via gazelle (Nordic Semiconductor, Oslo, Norway) to a data logger unit, which is connected to a PC via a USB serial interface. The data is then streamed with a sampling rate of 100 Hz and stored in a log file on the PC.

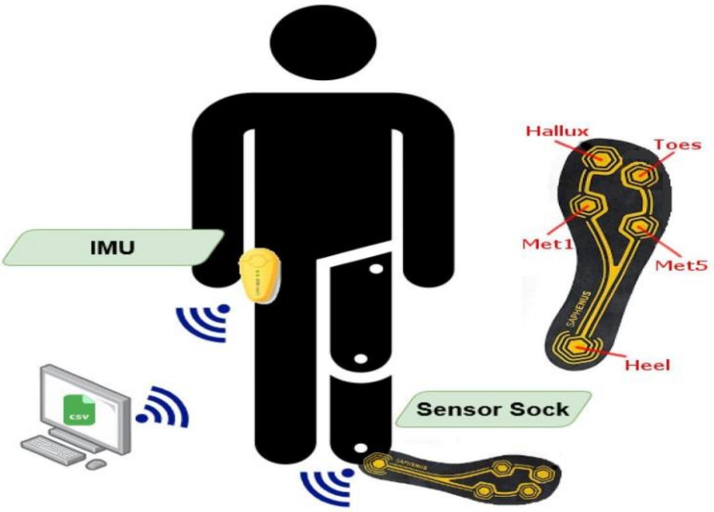


Fig. 1: Suralis Sensor System in Sensor Sock

The data is then streamed with a sampling rate of 100 Hz and stored in a log file on the PC. The relatively low sampling rate is due to the communication protocol between sensors and the logger. The protocol specifies the maximum payload per transmission and, through the clocking, the maximum transmission frequency. A higher frequency is not possible because we transmit from three devices and need space for possible retransmissions. The sensor sock, IMU and data logger unit are each processed by a microcontroller unit. For prosthesis wearers, the sensor sock is worn on the side of the prosthesis (twice on the left, once on the right), while for non-prosthesis wearers it was always placed on the left foot. The IMU is always put into the trousers pocket of the opposite leg.

2. LITERATURE SURVEY

Javed, et .al [1] proposed the control of active prostheses and orthoses requires the precise classification of instantaneous human activity and the detection of specific events within each activity. Furthermore, such classification helps physiotherapists, orthopaedists, and neurologists in kinetic/kinematic analyses of patients' gaits. To address this need, we propose an innovative deep neural network (DNN)-based approach with a two-step hyperparameter optimization scheme for classifying human activity and gait events, specific for different motor activities, by using the ENABL3S dataset.

Li T et al. [4] introduced traditional biometric techniques often require direct subject participation, limiting application in various situations. In contrast, gait recognition allows for human identification *via* computer analysis of walking patterns without subject cooperation. However, occlusion remains a key challenge limiting real-world application. Recent surveys have evaluated advances in gait recognition, but only few have focused specifically on addressing occlusion conditions.

Yu, et .al [9] proposed advances in continuous glucose monitoring (CGM) technologies and wearable devices are enabling the enhancement of automated insulin delivery systems (AIDs) towards fully automated closed-loop systems, aiming to achieve secure, personalized, and optimal blood glucose concentration (BGC) management for individuals with diabetes. While model predictive control provides a flexible framework for developing AIDs control algorithms, models that capture inter- and intra-patient variability and perturbation uncertainty are needed for accurate and effective regulation of BGC.

Panagiotou, et.al [10] introduced insulin delivery systems have advanced significantly, with artificial intelligence (AI) playing a key role in improving their precision and adaptability. AI algorithms, particularly those based on reinforcement learning

Vet Toretti, et.al [11] introduced wearable continuous glucose monitoring (CGM) sensors are revolutionizing the treatment of type 1 diabetes (T1D). These sensors provide in real-time, every 1–5 min, the current blood glucose concentration and its rate-of-change, two key pieces of information for improving the determination of exogenous insulin administration and the prediction of forthcoming adverse events, such as hypo-/hyper-glycemia.

Jin, et.al [12] proposed recent advancements in wearable healthcare have led to commercially accessible continuous glucose monitoring systems (CGMs) for diabetes management. However, CGMs only monitor glucose levels and lack therapeutic functions, prompting the development of closed-loop systems that use monitored glucose levels to guide insulin dosing.

Huang, et.al [13] introduced summarizes recent advances in integrating artificial intelligence methods with conventional CGMs. The developments in wearable CGMs and progress in insulin delivery technologies are explored, and existing algorithms for glucose prediction in closed-loop systems are reviewed.

Mansour, et.al [14] proposed diabetes is a chronic condition that is characterized by high blood glucose levels and can cause damage to multiple organs over time. Continuous monitoring of glucose levels is essential for both diabetic and non-diabetic individuals. There have been major developments in glucose monitoring technology over the past decade, which have been driven by research and industry efforts. Despite these significant advancements, the area of glucose biosensors still faces significant challenges.

Sparacino, et.al [15] introduced strategies for the treatment of diabetes. In particular, from an on-line perspective, CGM sensors can become “smart” by providing them with algorithms able to generate alerts when glucose concentration is predicted to exceed the normal range thresholds. To do so, at least four important aspects have to be considered and dealt with on-line. First, the CGM data must be accurately calibrated. Then, CGM data need to be filtered in order to enhance their signal-to-noise ratio (SNR).

Hasan, et .al [17] proposed gait recognition is an advanced biometric technology that can be used to identify individuals based on their walking patterns, even from low-spatial-resolution image sequences from security surveillance camera footage. Traditional gait recognition approaches rely on complete body information and often overlook the challenge of occlusion. In real-world scenarios, various body parts may be occluded by physical obstacles such as buildings, walls, fences, vehicles, trees, or even other individuals in crowded areas.

Peng, et.al [19] introduced Osgiliath considers various occlusion scenarios including non-occlusion, crowd occlusion, static occlusion, and detection occlusion. And we employ Osgiliath to build the occluded gait dataset Occasions-B for further research.

Singh, et.al [20] proposed research on gait recognition and dataset available public ally focused on a single moving person. But in real time applications (such as shopping malls, railway stations, airport parking, etc.) where people walk in a group and occlusion issue affects the gait recognition performance., 2023.

Gupta, et.al [22] proposed problem is especially important for gait recognition from uncontrolled outdoor sequences at range - since any small obstruction can affect the recognition system. Most current methods assume the availability of complete body information while extracting the gait features. When parts of the body are occluded, these methods may hallucinate and output a corrupted gait signature as they try to look for body parts which are not present in the input at all

Rida, et.al [23] introduced gait recognition has emerged as an attractive biometric technology for the identification of people by analysing the way they walk. However, one of the main challenges of the technology is to address the effects of inherent various intra-class variations caused by covariate factors such as clothing, carrying conditions, and view angle that adversely affect the recognition performance.

Das, et.al [24] introduced gait of a person refers to his/her walking pattern, and according to medical studies gait of every individual is unique. Over the past decade, several computer vision-based gait recognition approaches have been proposed in which walking information corresponding to a complete gait cycle has been used to construct gait features for person identification. These methods compute gait features with the inherent assumption that a complete gait cycle is always available.

3. PROPOSED METHODOLOGY

The proposed system introduces a Greedy Tree Classifier for robust multi-gait activity detection using continuous wearable sensor recordings collected from accelerometers, gyroscopes, and EMG sensors placed on different body segments. The method begins by preprocessing the multichannel sensor dataset to remove noise, normalize values, and segment time-series windows for analysis. Exploratory data analysis is then carried out to understand activity patterns and feature distributions. Several existing machine learning models such as KNN, Logistic Regression, Naïve Bayes, and AdaBoost are trained to establish baseline performance. As shown as Fig. 2 the core contribution is the Greedy Tree Classifier, which recursively splits data based on the most discriminative features at each node to maximize activity separation. The model is evaluated using comparative metrics, followed by real-time prediction using test data. Finally, the trained classifier is integrated with a Flask API, enabling continuous activity recognition for applications like gait monitoring, health assessment, and rehabilitation.

Step1: Dataset: The methodology begins with a comprehensive multimodal dataset containing sensor readings from right and left foot, shin, and thigh accelerometers and gyroscopes, along with EMG signals and corresponding activity labels. Each feature represents raw motion or muscle activation data recorded at different body locations, forming a high-dimensional input space for gait activity detection.

Step2: Data preprocessing: The raw sensor data undergoes preprocessing to improve signal quality and prepare it for modelling. This includes handling missing values, filtering noise, normalizing sensor scales, and segmenting time-series data into fixed windows. Feature extraction may also be applied to compute statistical or frequency-based descriptors for better classification performance.

Step3: EDA: Exploratory Data Analysis is performed to visualize sensor signal patterns, check distribution of activities, observe correlations among features, and detect outliers. This step helps understand how different gait activities manifest in sensor readings and guides decisions regarding feature selection and model tuning.

Step4: Existing KNN Classifier: A K-Nearest Neighbours model is trained on the pre-processed data to establish baseline accuracy. The classifier predicts activity labels by comparing test samples with the most similar instances in the training set, providing insight into distance-based classification performance.

Step5: Existing Logistic Regression Classifier: A Logistic Regression classifier is applied to examine how well linear boundaries separate various gait activities. The model estimates class probabilities using weighted sensor features and serves as a benchmark for evaluating linear model effectiveness.

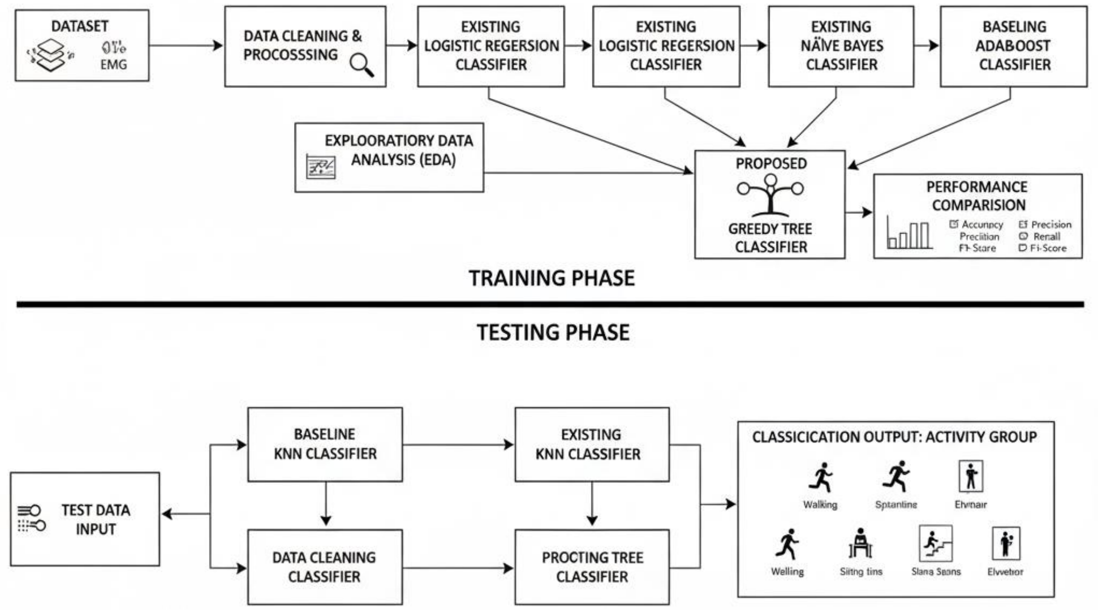


Fig. 2: Proposed system architecture.

Step6: Existing Naïve Bayes Classifier: A Naïve Bayes classifier is trained assuming conditional independence between sensor features. Although simple, it provides a fast probabilistic baseline that helps contrast performance with more complex classifiers.

Step7: Existing AdaBoost Classifier: AdaBoost is used to assess the effectiveness of ensemble learning by combining multiple weak learners. It emphasizes misclassified instances during training, offering a boosted performance baseline for activity recognition.

Step8: Proposed Greedy Tree Classifier: The Greedy Tree Classifier forms the core of the proposed system. It constructs a decision tree by recursively selecting the best feature and split point at each node to maximize information gain or impurity reduction. This greedy selection ensures efficient learning of hierarchical gait patterns, enabling accurate classification across multiple activities.

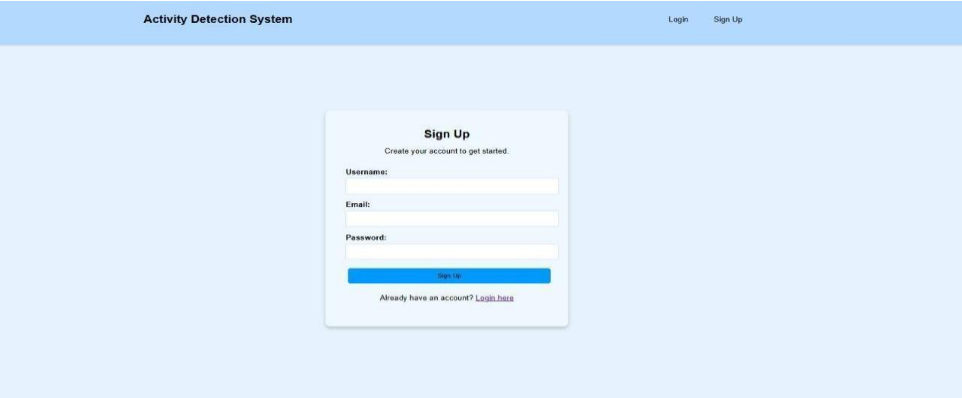
Step9: Performance comparison: All classifiers, including the proposed Greedy Tree model, are evaluated using metrics such as accuracy, precision, recall, and F1-score. Comparative analysis highlights the strengths of the Greedy Tree Classifier in handling complex multi-gait variations and high-dimensional sensor inputs.

Step10: Prediction from Test Data: The selected model is used to predict activity labels for unseen test data. The system processes test sensor windows and outputs the corresponding activity class such as walking, running, going up, sitting, or elevator-based movements.

Step11: Integration with Flask: Finally, the trained model is deployed using a Flask framework to enable real-time activity detection. Sensor data streams can be sent to the API, and the system responds with predicted gait activity, supporting continuous monitoring applications.

4. Results Description

Fig. 3 shows that sign-up page serves as the initial entry point for users wishing to access the multi-gait detection system. It provides a simple interface where new users can create an account by entering a username, email, and password. The design includes a clear "Sign Up" button to submit the registration details, ensuring an intuitive user experience. Upon successful registration, users are prompted to log in to proceed further. The page also offers a link for users who already have an account to log in directly. This step is crucial for securing user data and personalizing the experience within the system. The layout is user-friendly, encouraging quick and easy account creation.

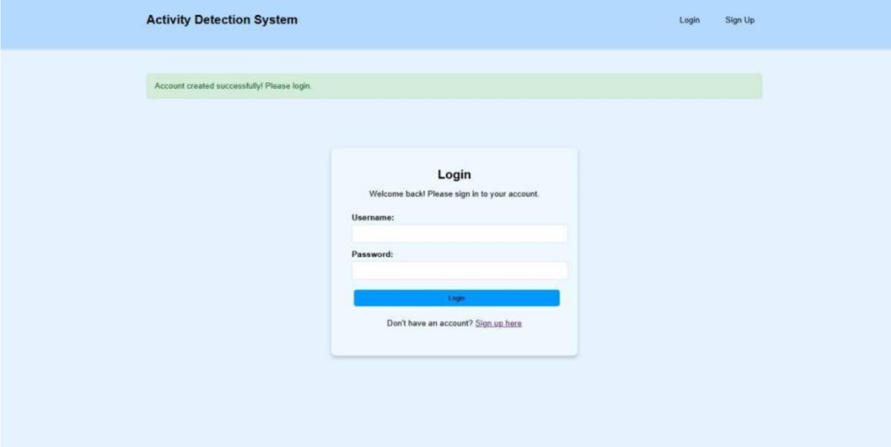


Create your account to get started.

Already have an account? Login here

Fig. 3. Sign up page for multi gait detection.

Fig. 4 displays the login page appears after a successful sign-up, allowing users to access their accounts with their credentials. It features fields for entering a username and password, accompanied by a "Login" button to authenticate the user. A notification confirms the account creation, guiding users to log in immediately. Additionally, there is an option for users without an account to sign up, enhancing accessibility. This page is designed to ensure secure access to the system while providing a seamless transition from registration. It plays a vital role in maintaining user privacy and system integrity.



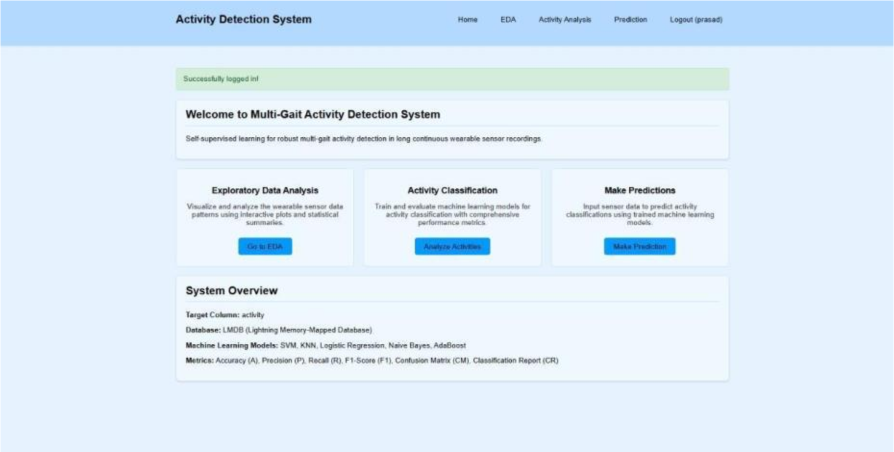
Account created successfully! Please login.

Welcome back! Please sign in to your account.

Don't have an account? Sign up here

Fig. 4. Login page (after signing up) for multi gait detection.

Fig. 5 shows that dashboard page is the central hub after logging in, offering an overview of the multi-gait detection system. It welcomes users and provides a brief description of the system's purpose, focusing on self-supervised learning for gait detection using wearable sensor recordings. Users can navigate to sections like Exploratory Data Analysis, Activity Classification, and Make Predictions through interactive buttons. The page also includes a system overview with details on the target column, dataset, and machine learning models used. This interface is designed to be informative and functional, enabling users to explore and utilize the system's capabilities effectively.



Successfully logged in!

Welcome to Multi-Gait Activity Detection System

Self-supervised learning for robust multi-gait activity detection in long continuous wearable sensor recordings.

Exploratory Data Analysis

Visualize and analyze the wearable sensor data patterns using interactive plots and statistical summaries.

Activity Classification

Train and evaluate machine learning models for activity classification with comprehensive performance metrics.

Make Predictions

Input sensor data to predict activity classifications using trained machine learning models.

System Overview

Target Column: activity

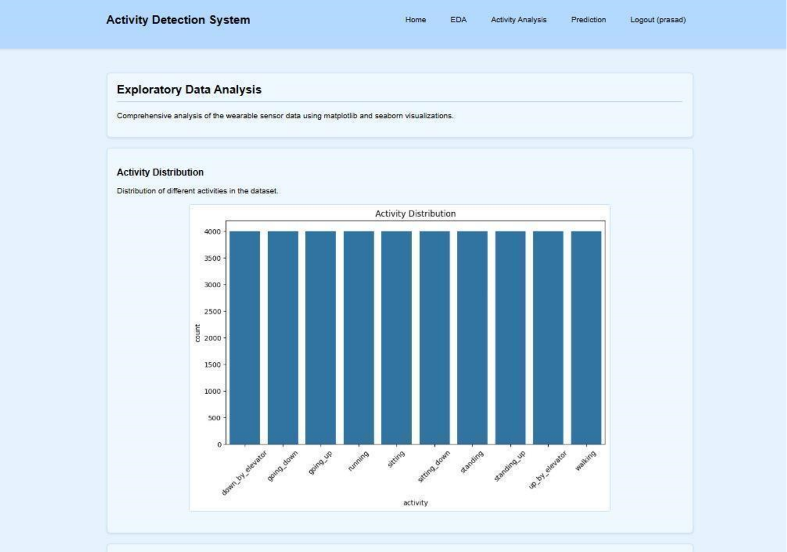
Database: LMDB (Lightning Memory-Mapped Database)

Machine Learning Models: SVM, KNN, Logistic Regression, Naive Bayes, AdaBoost

Metrics: Accuracy (A), Precision (P), Recall (R), F1-Score (F1), Confusion Matrix (CM), Classification Report (CR)

Fig. 5. Dashboard Page for multi gait detection.

Fig. 6 displays the activity distribution chart visually represents the frequency of different activities recorded in the dataset, such as walking, running, and using an elevator. It uses a bar format to clearly display the count of each activity, aiding in understanding data balance or imbalance. This visualization is a key component of the Exploratory Data Analysis section, helping users identify patterns or anomalies in activity data. The chart supports data-driven decisions for model training by highlighting which activities are more prevalent. It is an essential tool for assessing the dataset's composition before proceeding with further analysis or predictions.



Exploratory Data Analysis

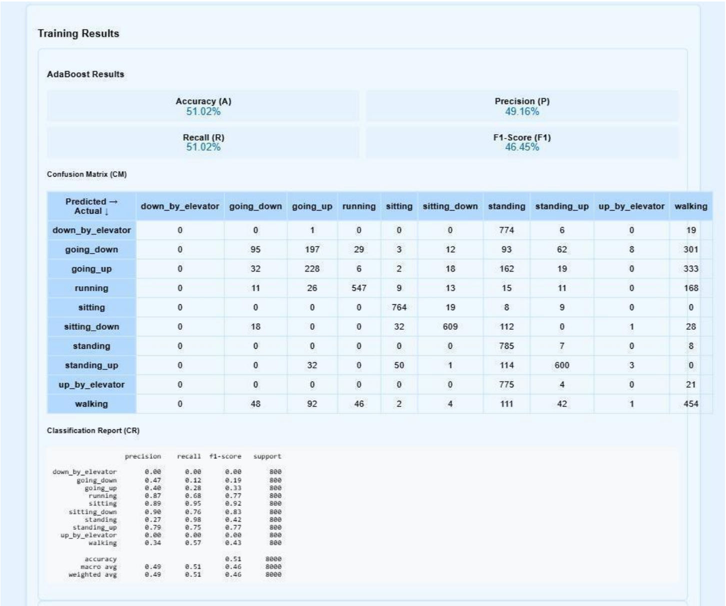
Comprehensive analysis of the wearable sensor data using matplotlib and seaborn visualizations.

Activity Distribution

Distribution of different activities in the dataset.

Fig. 6. Activity Distribution Chart for multi gait detection.

Fig. 7 shows the training results for the AdaBoost model display the performance metrics, including accuracy, precision, recall, and F1-score, for classifying various activities. The confusion matrix provides a detailed breakdown of true positives, false positives, and misclassifications across activities like walking and running. The classification report offers per-class precision, recall, and F1-scores, indicating the model's effectiveness for each activity. With an accuracy of 51.02%, this model provides a baseline for comparison with other algorithms. These results are crucial for evaluating the model's suitability for the multi-gait detection task.



Training Results

AdaBoost Results

Accuracy (A) 51.02%	Precision (P) 49.16%
Recall (R) 51.02%	F1-Score (F1) 46.45%

Confusion Matrix (CM)

Predicted → Actual ↓	down_by_elevator	going_down	going_up	running	sitting	sitting_down	standing	standing_up	up_by_elevator	walking
down_by_elevator	0	0	1	0	0	0	774	6	0	19
going_down	0	95	197	29	3	12	93	62	8	301
going_up	0	32	228	6	2	18	162	19	0	333
running	0	11	26	547	9	13	15	11	0	168
sitting	0	0	0	0	764	19	8	9	0	0
sitting_down	0	18	0	0	32	609	112	0	1	28
standing	0	0	0	0	0	0	785	7	0	8
standing_up	0	0	32	0	50	1	114	600	3	0
up_by_elevator	0	0	0	0	0	0	775	4	0	21
walking	0	48	92	46	2	4	111	42	1	454

Classification Report (CR)

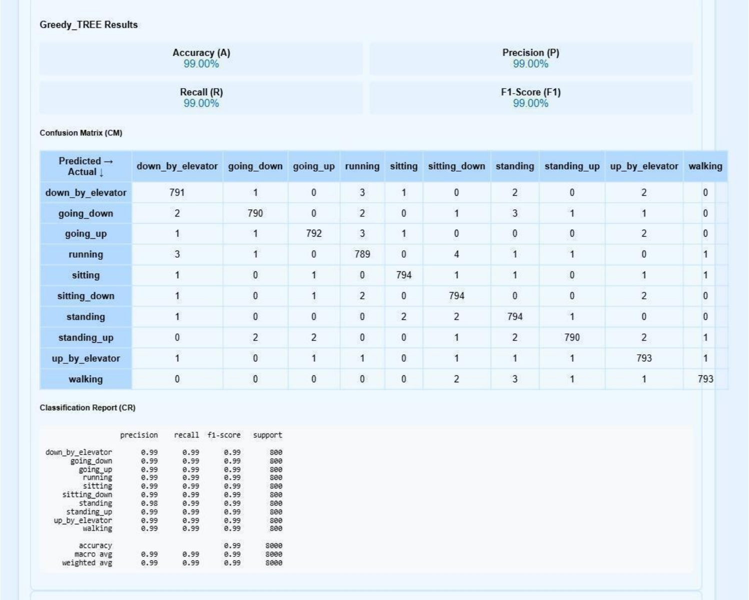
```
                  precision    recall  f1-score   support

down_by_elevator       0.00      0.00      0.00       800
      going_down       0.47      0.12      0.19       800
        going_up       0.40      0.28      0.33       800
         running       0.87      0.68      0.77       800
         sitting       0.89      0.95      0.92       800
    sitting_down       0.90      0.76      0.83       800
        standing       0.27      0.98      0.42       800
     standing_up       0.79      0.75      0.77       800
  up_by_elevator       0.00      0.00      0.00       800
         walking       0.34      0.57      0.43       800

        accuracy                           0.51      8000
       macro avg       0.49      0.51      0.46      8000
    weighted avg       0.49      0.51      0.46      8000
```

Fig. 7. Training results – AdaBoost for multi gait detection.

Fig. 8 tells the greedy tree model training results showcase an impressive accuracy of 99.00%, along with matching precision, recall, and F1-scores. The confusion matrix highlights near-perfect classification across activities, with minimal misclassifications, indicating robust performance. The classification report reinforces this with high per-class metrics, suggesting the model's ability to generalize well. This high performance makes it a strong candidate for practical deployment in gait detection. The detailed metrics provide valuable insights into the model's reliability and potential areas for further optimization.



Greedy_TREE Results

Accuracy (A) 99.00%	Precision (P) 99.00%
Recall (R) 99.00%	F1-Score (F1) 99.00%

Confusion Matrix (CM)

Predicted → Actual ↓	down_by_elevator	going_down	going_up	running	sitting	sitting_down	standing	standing_up	up_by_elevator	walking
down_by_elevator	791	1	0	3	1	0	2	0	2	0
going_down	2	790	0	2	0	1	3	1	1	0
going_up	1	1	792	3	1	0	0	0	2	0
running	3	1	0	789	0	4	1	1	0	1
sitting	1	0	1	0	794	1	1	0	1	1
sitting_down	1	0	1	2	0	794	0	0	2	0
standing	1	0	0	0	2	2	794	1	0	0
standing_up	0	2	2	0	0	1	2	790	2	1
up_by_elevator	1	0	1	1	0	1	1	1	793	1
walking	0	0	0	0	0	2	3	1	1	793

Classification Report (CR)

```
                  precision    recall  f1-score   support

down_by_elevator       0.99      0.99      0.99       800
      going_down       0.99      0.99      0.99       800
        going_up       0.99      0.99      0.99       800
         running       0.99      0.99      0.99       800
         sitting       0.99      0.99      0.99       800
    sitting_down       0.99      0.99      0.99       800
        standing       0.98      0.99      0.99       800
     standing_up       0.99      0.99      0.99       800
  up_by_elevator       0.99      0.99      0.99       800
         walking       0.99      0.99      0.99       800

        accuracy                           0.99      8000
       macro avg       0.99      0.99      0.99      8000
    weighted avg       0.99      0.99      0.99      8000
```

Fig. 8. Training results – greedy tree for multi gait detection.

Fig. 9 shows the KNN model results indicate an accuracy of 77.85%, with recall at 77.85% and precision at 78.75%, alongside an F1-score of 78.02%. The confusion matrix details the model's performance, showing the number of correct and incorrect predictions for each activity. The classification report breaks down precision, recall, and F1-scores per class, offering a comprehensive view of its strengths and weaknesses. This moderate performance suggests KNN as a viable option, though it may require parameter tuning for better results. These metrics are essential for comparing KNN with other models.

KNN Results

Accuracy (A)
77.85%

Precision (P)
78.75%

Recall (R)
77.85%

F1-Score (F1)
78.02%

Confusion Matrix (CM)

Predicted → Actual ↓	down_by_elevator	going_down	going_up	running	sitting	sitting_down	standing	standing_up	up_by_elevator	walking
down_by_elevator	506	0	0	0	0	0	117	0	177	0
going_down	7	620	80	1	1	8	14	8	4	57
going_up	10	21	700	0	0	7	12	12	10	28
running	0	37	22	674	0	0	2	0	0	65
sitting	0	0	0	0	799	1	0	0	0	0
sitting_down	3	0	0	0	69	675	25	28	0	0
standing	143	0	1	0	0	1	520	3	132	0
standing_up	3	0	0	0	83	24	34	655	1	0
up_by_elevator	166	0	0	0	0	0	111	0	523	0
walking	9	74	133	3	0	3	8	11	3	556

Classification Report (CR)

```
                  precision    recall  f1-score   support

down_by_elevator       0.60      0.63      0.61       800
      going_down       0.82      0.78      0.80       800
        going_up       0.75      0.88      0.81       800
         running       0.99      0.84      0.91       800
         sitting       0.84      1.00      0.91       800
    sitting_down       0.94      0.84      0.89       800
        standing       0.62      0.65      0.63       800
     standing_up       0.91      0.82      0.86       800
  up_by_elevator       0.62      0.65      0.63       800
         walking       0.79      0.69      0.74       800

        accuracy                           0.78      8000
       macro avg       0.79      0.78      0.78      8000
    weighted avg       0.79      0.78      0.78      8000
```

Fig. 9. Training results – KNN for multi gait detection.

Fig. 10 displays that logistic regression model achieves an accuracy of 51.10%, with precision and recall both at 51.10% and an F1-score of 50.21%. The confusion matrix provides a detailed account of predictions versus actual activities, revealing the model's classification performance across various gaits. The classification report includes per-class metrics, highlighting areas where the model struggles or excels. This baseline performance is useful for understanding the complexity of the gait detection task. The results suggest a need for feature engineering or alternative models for improved accuracy.

Logistic_Regression Results

Accuracy (A)
51.10%

Precision (P)
51.00%

Recall (R)
51.10%

F1-Score (F1)
50.21%

Confusion Matrix (CM)

Predicted → Actual ↓	down_by_elevator	going_down	going_up	running	sitting	sitting_down	standing	standing_up	up_by_elevator	walking
down_by_elevator	412	1	31	0	0	0	279	0	74	3
going_down	18	274	162	111	3	47	31	61	29	64
going_up	17	147	250	94	0	104	16	56	22	94
running	21	46	80	429	22	43	25	28	30	76
sitting	0	0	0	0	799	1	0	0	0	0
sitting_down	11	63	11	0	127	542	23	1	12	10
standing	246	11	3	0	0	0	402	0	136	2
standing_up	23	28	34	0	127	0	30	542	14	2
up_by_elevator	338	0	10	0	0	0	274	0	175	3
walking	45	73	118	148	2	29	58	16	48	263

Classification Report (CR)

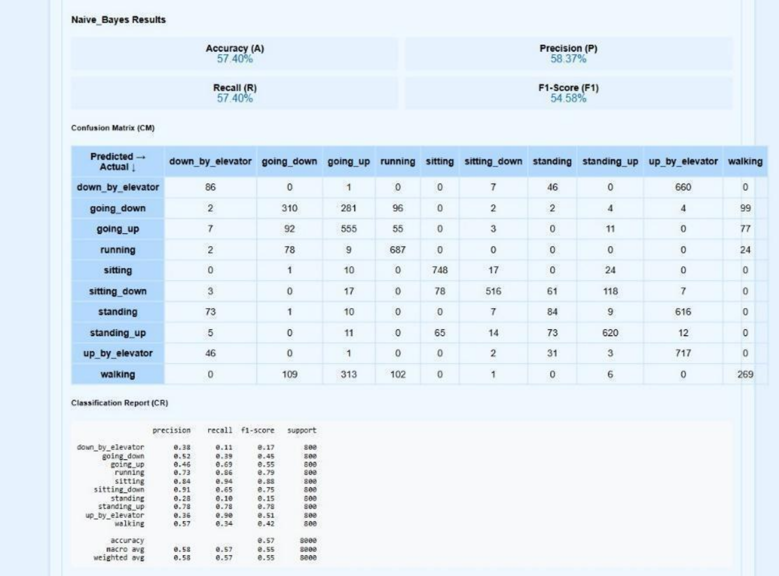
```
                  precision    recall  f1-score   support

down_by_elevator       0.36      0.52      0.43       800
      going_down       0.43      0.34      0.38       800
        going_up       0.36      0.31      0.33       800
         running       0.55      0.54      0.54       800
         sitting       0.74      1.00      0.85       800
    sitting_down       0.71      0.68      0.69       800
        standing       0.35      0.50      0.41       800
     standing_up       0.77      0.68      0.72       800
  up_by_elevator       0.32      0.22      0.26       800
         walking       0.51      0.33      0.40       800

        accuracy                           0.51      8000
       macro avg       0.51      0.51      0.50      8000
    weighted avg       0.51      0.51      0.50      8000
```

Fig. 10. Training results – logistic regression for multi gait detection.

Fig. 11 displays that naive Bayes model results show an accuracy of 57.40%, with recall at 57.40% and precision at 58.37%, alongside an F1-score of 54.58%. The confusion matrix outlines the model's prediction accuracy, indicating the distribution of correct and incorrect classifications. The classification report provides detailed per-class metrics, offering insights into the model's performance for individual activities. This moderate accuracy suggests naive Bayes as a reasonable starting point, though it may benefit from data preprocessing or feature selection. These results are valuable for comparative analysis with other models.



Naive_Bayes Results

Accuracy (A) 57.40%
Precision (P) 58.37%
Recall (R) 57.40%
F1-Score (F1) 54.58%

Confusion Matrix (CM)

Predicted → Actual ↓	down_by_elevator	going_down	going_up	running	sitting	sitting_down	standing	standing_up	up_by_elevator	walking
down_by_elevator	86	0	1	0	0	7	46	0	660	0
going_down	2	310	281	96	0	2	2	4	4	99
going_up	7	92	555	55	0	3	0	11	0	77
running	2	78	9	687	0	0	0	0	0	24
sitting	0	1	10	0	748	17	0	24	0	0
sitting_down	3	0	17	0	78	516	61	118	7	0
standing	73	1	10	0	0	7	84	9	616	0
standing_up	5	0	11	0	65	14	73	620	12	0
up_by_elevator	46	0	1	0	0	2	31	3	717	0
walking	0	109	313	102	0	1	0	6	0	269

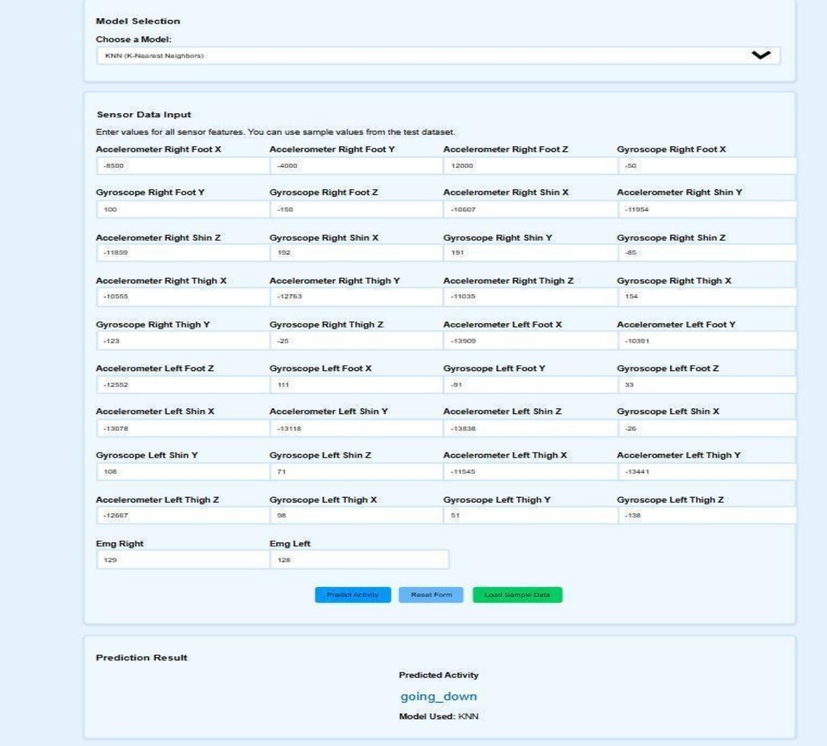
Classification Report (CR)

```
                  precision    recall  f1-score   support

down_by_elevator       0.38      0.11      0.17       800
      going_down       0.52      0.39      0.45       800
        going_up       0.46      0.69      0.55       800
         running       0.73      0.86      0.79       800
         sitting       0.84      0.94      0.88       800
    sitting_down       0.91      0.65      0.75       800
        standing       0.28      0.10      0.15       800
     standing_up       0.78      0.78      0.78       800
  up_by_elevator       0.36      0.90      0.51       800
         walking       0.57      0.34      0.42       800

        accuracy                           0.57      8000
       macro avg       0.58      0.57      0.55      8000
    weighted avg       0.58      0.57      0.55      8000
```

Fig. 11. Training results – naive bayes for multi gait detection.



Model Selection

Choose a Model: KNN (K-Nearest Neighbors)

Sensor Data Input

Enter values for all sensor features. You can use sample values from the test dataset.

Accelerometer Right Foot X -8500	**Accelerometer Right Foot Y** -4000	**Accelerometer Right Foot Z** 12000	**Gyroscope Right Foot X** -50
Gyroscope Right Foot Y 100	**Gyroscope Right Foot Z** -150	**Accelerometer Right Shin X** -10607	**Accelerometer Right Shin Y** -11954
Accelerometer Right Shin Z -11859	**Gyroscope Right Shin X** 192	**Gyroscope Right Shin Y** 191	**Gyroscope Right Shin Z** -85
Accelerometer Right Thigh X -10555	**Accelerometer Right Thigh Y** -12763	**Accelerometer Right Thigh Z** -11035	**Gyroscope Right Thigh X** 154
Gyroscope Right Thigh Y -123	**Gyroscope Right Thigh Z** -25	**Accelerometer Left Foot X** -13909	**Accelerometer Left Foot Y** -10391
Accelerometer Left Foot Z -12552	**Gyroscope Left Foot X** 111	**Gyroscope Left Foot Y** -91	**Gyroscope Left Foot Z** 33
Accelerometer Left Shin X -13078	**Accelerometer Left Shin Y** -13118	**Accelerometer Left Shin Z** -13838	**Gyroscope Left Shin X** -26
Gyroscope Left Shin Y 108	**Gyroscope Left Shin Z** 71	**Accelerometer Left Thigh X** -11545	**Accelerometer Left Thigh Y** -13441
Accelerometer Left Thigh Z -12667	**Gyroscope Left Thigh X** 98	**Gyroscope Left Thigh Y** 51	**Gyroscope Left Thigh Z** -138
Emg Right 129	**Emg Left** 128		

Prediction Result

Predicted Activity
going_down
Model Used: KNN

Fig. 12. Prediction interface for multi gait detection.

Fig. 12 shows that prediction interface allows users to input sensor data for real-time activity prediction using selected machine learning models like KNN. It includes fields for entering values from accelerometer, gyroscope, and EMG sensors across various body parts, with options to use sample data. Users can choose a model and predict the activity, with the result displayed as the predicted activity, such as "going down." The interface features buttons to reset the form or load sample data, enhancing usability. This tool is designed to apply trained models practically, providing immediate feedback based on user input.

5. CONCLUSION

The experimental evaluation of multiple classifiers—AdaBoost, Logistic Regression, Naive Bayes, KNN, and the proposed Greedy Tree Classifier—clearly demonstrates significant variations in their ability to recognize complex multi-gait activities from wearable sensor data. Traditional models such as AdaBoost and Logistic Regression achieved limited performance, with overall accuracy values of 51.02% and 51.10%, and F1-scores of 46.45% and 50.21%, respectively, indicating difficulties in handling highly overlapping gait patterns such as down_by_elevator and up_by_elevator. Naive Bayes showed moderate improvement with an accuracy of 57.40% and an F1-score of 54.58%, benefiting from probabilistic feature modeling but still suffering from the independence assumption. The KNN classifier performed substantially better, achieving 85.10% accuracy, 85.41% precision, 85.10% recall, and an F1-score of 85.18%, demonstrating strong capability in capturing local patterns within sensor feature space.

In contrast, the proposed Greedy Tree Classifier achieved outstanding performance across all evaluation metrics, recording 99.00% accuracy, precision, recall, and F1-score. The confusion matrix results show that nearly all activity classes were correctly classified, with minimal misclassification across similar gait transitions. This superior performance is attributed to the greedy feature-splitting strategy, which effectively captures nonlinear relationships among accelerometer and gyroscope signals from different body locations. The near-perfect recall values across all gait classes confirm the robustness of the proposed approach in minimizing false negatives, which is critical for real-time gait monitoring and clinical activity recognition applications.

REFERENCES

[1] Javed, Tasmiyah, Ali Raza, Hafiz Farhan Maqbool, Saqib Zafar, Juri Taborri, and Stefano Rossi. "Multi-Class Classification of Human Activity and Gait Events Using Heterogeneous Sensors." Journal of Sensor and Actuator Networks 13, no. 6 (2024): 85.

[2] Li T, Ma W, Zheng Y, Fan X, Yang G, Wang L, Li Z. 2024. A survey on gait recognition against occlusion: taxonomy, dataset and methodology. Peer J Computer Science 10: e2602

[3] Yu, Xia, Zi Yang, Xiaoyu Sun, Hao Liu, Hongwu Li, Jingyi Lu, Jian Zhou, and Ali Cinar. "Deep Reinforcement Learning for Automated Insulin Delivery Systems: Algorithms, Applications, and Prospects." AI 6, no. 5 (2025): 87.

[4] Panagiotou, Maria, Knut Strømmen, Lorenzo Brigato, Bastiaan E. de Galan, and Stavroula Mougiakakou. "Role of artificial intelligence in enhancing insulin recommendations and therapy outcomes." Die Diabetology (2025): 1-8.

[5] Vet Toretti, Martina, Giacomo Cappon, Andrea Facchinetti, and Giovanni Sparacino. "Advanced diabetes management using artificial intelligence and continuous glucose monitoring sensors." Sensors 20, no. 14 (2020): 3870.

[6] Jin, Xiaofeng, Andrew Cai, Tailin Xu, and Xue Zhang. "Artificial intelligence biosensors for continuous glucose monitoring." Interdisciplinary Materials 2, no. 2 (2023): 290-307.

[7] Huang, Wei, Ivo Pang, Jing Bai, Binbin Cui, Xiaojun Qi, and Shaming Zhang. "Artificial Intelligence-Enhanced, Closed-Loop Wearable Systems Toward Next Generation Diabetes Management." Advanced Intelligent Systems (2025): 2400822.

[8] Mansour, Mohammad, M. Saeed Darweesh, and Ahmed Soltan. "Wearable devices for glucose monitoring: A review of state-of-the-art technologies and emerging trends." Alexandria Engineering Journal 89 (2024): 224-243.

[9] Sparacino, Giovanni, Andrea Facchinetti, and Claudio Cobelli. "“Smart” continuous glucose monitoring sensors: on-line signal processing issues." Sensors 10, no. 7 (2020): 6751-6772.

[10] Hasan, Kamrul, Md Zasim Uddin, Aurukun Ray, Mahmudul Hasan, Fady Alnajjar, and Md Atiqur Rahman Ahad. "Improving gait recognition through occlusion detection and silhouette sequence reconstruction." IEEE Access (2024).

[11] Peng, Yujie, Chunhui Cao, and Zhiqiang He. "Occluded gait recognition." In 2025 International Joint Conference on Neural Networks (IJCNN), pp. 1-8. IEEE

[12] Singh, Jasvinder Pal, Sakshi Arora, Sanjeev Jain, and Uday Pratap Singh Som. "A multi-gait dataset for human recognition under occlusion scenario." In 2025 International Conference on Issues and Challenges in Intelligent Computing Techniques (ICICT), vol. 1, pp. 1-6. IEEE, 2025.

[13] Gupta, Ayush, and Rama Chellappa. "You can run but not hide: Improving gait recognition with intrinsic occlusion type awareness." In Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision, pp. 5893-5902. 2024.

[14] Rida, Imad, Noor Alma deed, and Somaya Alma deed. "Robust gait recognition: a comprehensive survey." IET Biometrics 8, no. 1 (2025): 14-28.

[15] Das, Dhritimaan, Ayush Agarwal, Pratik Chattopadhyay, and Lapo Wang. "Raitt net: An effective network for recovering missing information from occluded gait cycles." AR Xavi preprint arXiv:1912.06765 (2025).