Federated LSTM-LRP Networks with SMOTE and LIME for Interpretable Gait Classification

O. Pushpalatha
Department of ECE, JAIN (Deemed-to-be University), Bengaluru, Karnataka, India | Department of ECE, Jain Institute of Technology, Davanagere, Karnataka, India
pushpalatha.sonu@gmail.com (corresponding author)

R. Premkumar
Department of ECE, JAIN (Deemed-to-be University), Bengaluru, Karnataka, India
premkumar.r@jainuniversity.ac.in

Received: 8 September 2025 | Revised: 23 September 2025, 17 October 2025, 25 October 2025, and 2 November 2025 | Accepted: 3 November 2025

 DOI: https://doi.org/10.48084/etasr.14598

ABSTRACT

Gait analysis is essential for the diagnosis and monitoring of various biomechanical and neurological conditions. Clinical reliability is compromised due to imbalanced data and lack of transparency in existing models. This study uses vertical GRF features of patients to analyze gait on the publicly available GaitRec dataset. The proposed approach addresses the limitations of traditional models using Borderline SMOTE to balance the dataset and FedAvg-SVRG to stabilize gradient updates across dispersed nodes and enable federated learning with stable convergence without centralizing data. Enhanced LRPs are combined with LSTM to improve transparency in pattern recognition, and LIME offers instance-level interpretability. A deep learning model is used to improve the accuracy of gait classification, achieving 99.72% on single run execution and 99.45±0.14% for 10-fold cross-validation. The proposed deep learning framework holds significant potential for robust gait classification and clinical decision making.

Keywords-federated stochastic variance reduced gradient; interpretable gait analysis network; Local Interpretable Model-agnostic Explanations (LIME)

I. INTRODUCTION

In orthopedics, accurate gait patterns are required to identify and treat fracture patients [1]. Gait analysis examines human motion to reveal biomechanical features and neurological limitations [2]. Advanced technologies, such as deep learning methods, increase the fidelity and efficiency of fracture identification [3]. Managing fractures is a challenge for both patients and medical staff, as it requires careful examination and management [4]. Traditional approaches relying on biomechanical models and observational methods can be laborious and subjective [5], whereas deep learning models can recognize complex gait patterns [6].

Fracture-related gait studies can analyze kinematic and kinetic data obtained from accelerometers, force plates, and motion capture systems [7]. Deep learning methods detect musculoskeletal disorders by determining small changes in gait patterns linked to fractures [8]. This multidisciplinary method combines various fields, such as biomechanics, orthopedics, and computer technology, to advance fracture diagnosis and management [9]. Researchers and physicians can use deep learning gait analysis to assess the severity of the fracture and the progress of rehabilitation [10]. The walking pattern of a human can be explained by LRP using suitable variables by assigning model predictions to input variables [11].

In [12], a deep learning model was proposed for gait analysis, achieving 96.67% detection accuracy for five distinct human activities. In [13], an ANN was used to evaluate gait kinematics in the sagittal plane among 200 children diagnosed with cerebral palsy and spastic diplegia. In [14], an autonomous system was developed to identify participants based on their gait, which is a personal feature that is used in nonverbal communication. In [15], the ankle kinematics of patients with fractures were shown to be considerably different from those of healthy participants. In [16], 20 patients (≥44 years) with intra-articular tibial plateau fractures were assessed twelve weeks post-injury. In [17], multiple machine learning classifiers were used to predict the activities of individuals wearing exoskeletons. In [18], a transfemoral prosthetic sensor was used to evaluate the ground reaction forces below the ankle joint for prosthetic parts.

The research gaps identified from existing models are a lack of transparency and imbalanced datasets that lead to lower accuracy. Clinicians need gait analysis, but the problem is in detecting injuries in particular locations that are responsible for gait impairments. The datasets used contain only a few body

regions. So, it is difficult for clinicians to detect whether gait is normal or abnormal. This study aimed to address these research gaps using the GaitRec dataset. The contributions of the proposed method are as follows:

- Uses Borderline SMOTE to reduce the bias and increase model accuracy by creating synthetic samples close to the decision boundaries for the minority classes.
- Integrates FedAvg-SVRG to enable federated learning by stabilizing gradient updates across distributed nodes, improving convergence and model consistency without centralizing sensitive gait data.
- Uses LSTM-LRP and LIME to provide both global feature relevance and instance-level insights, offering transparency into how specific gait features influence model predictions.

This interpretable federated approach provides a robust tool for gait analysis, suitable for real-world clinical settings, allowing for both accurate classification and actionable insights into gait patterns.

II. METHODOLOGY

The proposed architecture, shown in Figure 1, overcomes the limitations of class imbalance, privacy concerns, and the limited interpretability of traditional methods. The input data is collected from the GaitRec dataset, which is imbalanced. The GRF features of the patients are used for gait analysis. Borderline-SMOTE is used to generate new data points near the decision boundaries to balance the classes. Federated Averaging (FedAvg) is combined with the Stochastic Variant Reduced Gradient (SVRG) technique to address the variance in the gradients that can cause slow and unstable convergence, and also supervises the secure and efficient federated learning across distributed data. LSTM-LRP highlights the importance of the features in the sequential gait patterns for interpretability. The LSTM captures the temporal dependencies in the gait cycle. In this approach, two enhanced LRP rules are applied to the LSTM model, enabling a detailed layer-by-layer visualization of which features contribute most to the system's prediction. Instance-level explanation is provided by LIME, which makes the framework clinically reliable and applicable.

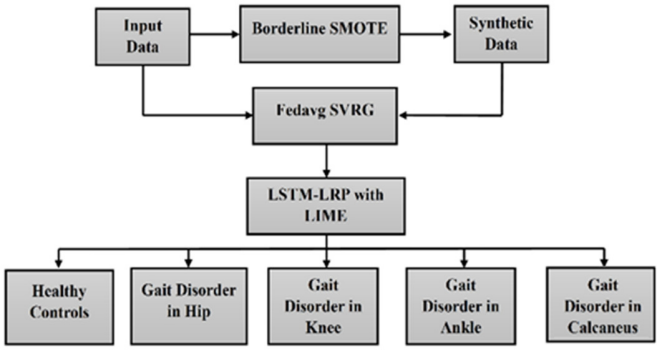

Fig. 1. Proposed architecture.

The proposed model was implemented using three LSTM layers having 64 units for single run training, and LSTM layers with 128, 64, and 64 units for a 10-fold cross-validation with batch normalization and LeakyReLU activations (α = 0.1). Dropout layers (rates of 0.3, 0.2, and 0.2) were used to enhance generalization for a 10-fold cross-validation. An LRP layer of 64 units was used between the 2nd and 3rd LSTM layers to capture important weighted features. A softmax output layer follows the final dense layers, including a 64-unit ReLU activation layer. Adam optimizer, with a learning rate of 0.0005, and categorical cross-entropy loss is used to train the model for computational efficiency. Batch sizes of 32 and 128 are used for single-run execution and 10-fold cross-validation, respectively.

A. Data Preprocessing with Borderline SMOTE

Borderline-SMOTE handles class imbalance by focusing on minority instances located close to the decision boundaries. It generates synthetic samples around these borderline points, preserving class separation and reducing overlap with majority samples, as shown in Figure 2 (minority class is shown in red). For each borderline minority sample, neighbors are selected and synthetic points are created using a random proportion of the distance to these neighbors:

$$sythestic_j = a_j + r_j^* dif_j, \quad j = 1,2,\ldots..,s \qquad (1)$$

where the random number r_j^* (between 0 to 1) multiplies the difference between the original and the selected neighbor instance at the j^{th} feature, a_j is the feature value at j that defines the process of generating synthetic samples in augmentation, balancing class distribution and enhancing model accuracy, precision, and recall for both normal and abnormal gait patterns.

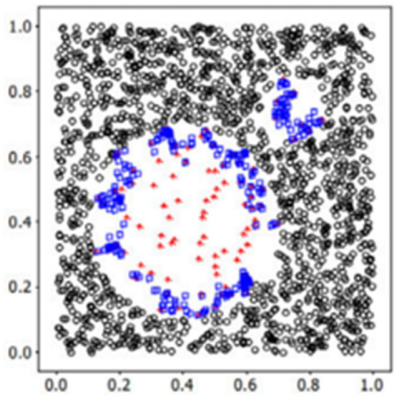

Fig. 2. Solid squares are the borderline minority samples, and hollow squares represent the borderline synthetic minority samples.

B. FedAvg-SVRG

Federated learning with SGD has been used for local resource management updates [19], but suffers from high variance and results in a decaying step size and slow convergence. To overcome this, instead of SGD, the SVRG update was used. The federated learning setup was implemented using TensorFlow 2.15 with Python 3.10 and NumPy. A workstation with an Intel Core i5 CPU at 2.4 GHz with 16 GB of RAM was used for training. The system consists of three nodes, each trained on a separate subset of the gait

dataset for 10 global communication rounds using FedAvg. Every node locally updates its model, and only the resulting weights are sent to the central server. Then the global model is updated by aggregating the weights without exposing raw data. This setup provides a model that protects privacy.

```
Algorithm: FedAvg combined with SVRG under an uneven agent sampling strategy
Input: Δ, L, θ^o, {a_q^l}, S, U
Output: θ^l
for l ← 0, ….., L − 1 do
  Identify the participating agents (sample 1_q^l ~ a_q^l)
  P^l ← set of participating agents
  for q ϵ P^l do
    c̃_{q,0}^l = θ^l
    for s ← 0, … … , S − 1 do
      c̃ = c̃_{q,s}^k
      μ̃ = (1/E_n) Σ_{i=1}^{E_n} ∇h_{q,i}(c̃)
      c̃_{q,s,0}^l = c̃
      for u ← 0, … , U − 1 do
        Randomly pick from ϵ {1, … … , E_n}
        c_{q,s,u+1}^l = c_{q,s,u}^l − Δ v_{q,s,u}^l
        where v_{q,s,u}^l = ∇h_#(c_{q,s,u}^l) − ∇h_#(c̃) + μ̃ is a gradient term used for local updated by the agent
      end for
      set c̃_{q,s+1}^l = c_{q,s,U}^l
    end for
  end for
  θ^{l+1} = θ^l + (1/Q) Σ_{q=1}^{Q} (1_q^l / a_q^l) (c_{q,S−1,U}^l − c̃_{q,0}^l)
end for
```

Here, a_q^l is the activation probability of each active agent at iteration l, and the server initializes the model with θ^0. L is the number of federated learning iterations and Q denotes the total agents. Over L rounds, participating agents P^l receive the current parameters θ^l and perform local updates [20]. c defines the feature weight, v is the correction gradient term, and S and U are the two snapshots that guide SVRG averaging [21, 22]. Unlike SGD, the SVRG allows a larger step size that can facilitate more rapid convergence.

C. LSTM-LRP

A neuron's transformation in a basic recurrent network can be described as:

$$d^{(t)} = H(Ad^{(t-1)} + Bx^{(t)} + b_d), \quad (2)$$

$$e^{(t)} = F(Dd^{(t)} + b_e), \quad (3)$$

These equations define the hidden state $d^{(t)}$ at time t, depending on the prior state $d^{(t-1)}$, the current input $x(t)$, and weight matrices A, B, D with biases b_d and b_e in an RNN. The network uses shared weights and the backpropagation algorithm through time for training [22]. However, due to weight sharing, gradients can either vanish or explode as the time gap increases, depending on $|A| > 0$ or $|A| < 0$ and the activation factor f used.

An LSTM [23] is similar to a standard RNN but replaces each hidden neuron with a memory cell containing a state unit $s^{(t)}$. For every time step t, the present input $x_j^{(t)}$ and past output $d_j^{(t-1)}$ are used to update the values of the forgetting entry unit $h_i^{(t)}$, state unit $s^{(t)}$, input entry unit $f_i^{(t)}$, output entry unit $y_i^{(t)}$ and output $d_i^{(t)}$.

The following equations delineate the procedures for an LSTM system at each stage as follows:

$$h_i^{(t)} = \sigma\left(b_i^h + \sum_j B_{i,j}^h x_j^{(t)} + \sum_j A_{i,j}^h d_j^{(t-1)}\right) \quad (4)$$

$$s_i^{(t)} = h_i^{(t)} s_i^{(t-1)} + f_i^{(t)} \sigma\left(b_i + \sum_j B_{i,j} x_j^{(t)} + \sum_j A_{i,j} d_j^{(t-1)}\right) \quad (5)$$

$$f_i^{(t)} = \sigma\left(b_i^f + \sum_j B_{i,j}^f x_j^{(t)} + \sum_j A_{i,j}^f d_j^{(t-1)}\right) \quad (6)$$

$$y_i^{(t)} = \sigma\left(b_i^o + \sum_j B_{i,j}^o x_j^{(t)} + \sum_j A_{i,j}^o d_j^{(t-1)}\right) \quad (7)$$

$$d_i^{(t)} = \tanh\left(s_i^{(t)}\right) y_i^{(t)} \quad (8)$$

All the gate and state units have their own bias b_i. Input and repetitive weights $B_{i,j}$ and $A_{i,j}$ are enabled by the sigmoid function $\sigma(\cdot)$. The hidden state $d_i^{(t)}$ is modified through these gates, which control when information enters or exits the cell.

The relevance to each input of the target class C is assigned by propagating the importance layer-wise using LRP [24]. For two consecutive layers u and q, the importance scores are redistributed using a specified rule to explain the prediction $h(x)$.

$$\sum_i R_i^{(u)} = \sum_i R_i^{(q)} = h(x) \quad (9)$$

where $R_i^{(u)}$ and $R_i^{(q)}$ are the outcomes of neurons in layers u and q, respectively. A fundamental rule, LRP-0, is given as:

$$R_i^{(u)} = \sum_j \frac{z_{i,j}}{\sum_l z_{l,j}} R_j^{(q)} \quad (10)$$

where $z_{i,j}$ denotes the neuron j in layer q that receives the contribution from the activated neuron i in layer u. $\sum_l z_{l,j}$ denotes the total relevance received by neuron j from all interconnected neurons in layer u before applying the activation factor.

The two upgraded rules used are as follows.

1) Epsilon Rule (LRP-ε):

$$R_i^{(u)} = \sum_j \frac{z_{i,j}}{\varepsilon + \sum_l z_{l,j}} R_j^{(q)} \quad (11)$$

where ε is a minor positive term to determine the numerical stability relation to the fundamental principle.

2) Alpha-Beta Rule (LRP-αβ):

$$R_i^{(u)} = \sum_j \left(\alpha \frac{z_{i,j}^+}{\sum_l z_{l,j}^+} - \beta \frac{z_{i,j}^-}{\sum_l z_{l,j}^-} \right) R_j^{(q)} \quad (12)$$

where $z_{i,j}^+$ and $z_{i,j}^-$ represent the contributions from the upper layer q, while the parameters α and β are used to control their weights to satisfy the conservation principle $\alpha + \beta$. By carefully selecting α and β, the relative importance of positive and negative contributions can be manually adjusted. In this process, two neurons are multiplied to manage the proportion of the signal that goes into the output:

$$n_a = h(z_f) \cdot f(z_s) \quad (13)$$

where $h(\cdot)$ and $f(\cdot)$ are the activation functions for the neuron values z_f and z_s that propagate to the gate and signal units from the previous layers, and n_a is the multiplicative neuron output. A commonly used method for redistributing relevance [25] is known as the "signal-take-all" approach, which involves:

$$(R_f, R_s) = (0, R_a) \quad (14)$$

where R_f and R_s are important scores for the control and data neurons, respectively, and R_a is the total relevance from the redistributed output neuron.

D. LIME for Interpreting the Proposed Model

The perturbed samples are assigned weights based on their similarity to the real instance.

$$\pi_x(z) = exp\left(-\frac{D(x,z)^2}{\sigma^2}\right) \quad (15)$$

where $D(x,z)$ is the distance between the original instance x and the disturbed instance z, and σ controls the width of the neighborhood. Multiple samples are generated near z and the trained model is evaluated for its prediction. Using LIME, the following equation is used to minimize the loss function:

$$\xi(x) = arg\ min_{f \in F} K(h, f, \pi_x) + \Omega(f) \quad (16)$$

where f is the interpretable surrogate system, F is the family of potential interpretable models, and $K(h, f, \pi_x)$ is the loss function that measures how well f approximates h (the original complex model) in the neighborhood defined by π_x. $\Omega(f)$ is a standardization term to ensure that the model g remains explainable, such as keeping the number of features small.

III. RESULT AND DISCUSSION

This section describes the implementation of the proposed method and the results obtained.

A. Dataset Description

The publicly available GaitRec dataset consists of data collected during patients' stay at the Austrian workers' compensation board's rehabilitation center (AUVA). The dataset encompasses one Healthy Control (HC) and four gait disorder classes, such as Hip (H), Knee (K), Ankle (A), and Calcaneus (C). The dataset comprises a total of 75732 bilateral Ground Reaction Force walking trials of 2084 patients with musculoskeletal problems and 211 healthy controls. Raw and processed GRF data and metadata are available. Noise was removed using a second-order Butterworth filter with a frequency of 20 Hz. The dataset contains the patients' data after the fracture, joint replacement, and ligament ruptures, and is mainly used to distinguish gait as normal or pathological, to assess improvement in gait after rehabilitation, and to determine subject-specific gait patterns. The dataset was divided into training and testing subsets in a ratio of 85:15. The dataset is available online from Figshare [26], and its details are described in [27].

B. Performance Evaluation

Before applying Borderline-SMOTE, the dataset was imbalanced, with Class A having the highest count (85,544) and Class HC the lowest (31,020). After applying Borderline-SMOTE, all classes were balanced to 85,544 samples, ensuring equal representation and reducing bias, which improves model performance.

Figure 3 provides the prediction probabilities for a single instance, where the model predicts Class 0 with 100% confidence. The features are split into two categories: those contributing to NOT 1 (Class 0) and those supporting Class 1. Features like 10, 9, and 13 strongly contribute to Class 0, while others like Feature 22 (0.66) and Feature 18 (0.58) are less impactful. The table on the right ranks feature values, highlighting their role in the prediction.

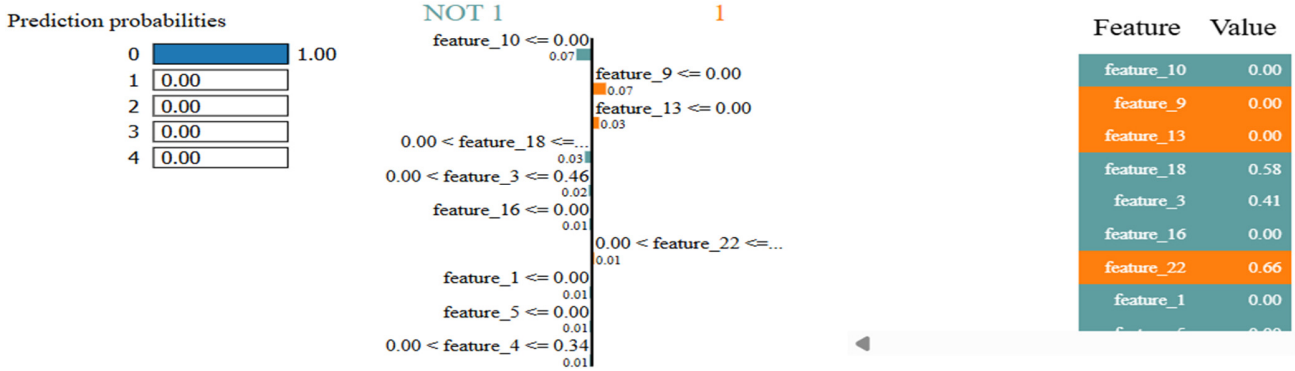

Fig. 3. Prediction probabilities for class 0 using LIME.

Figure 4 visually describes the local contributions of features for predicting Class 1. Features with negative contributions, such as features 10 and 18, push the prediction away from Class 1, while features with positive contributions, such as features 9 and 13, support Class 1. The graph shows that feature 10 has the most significant negative impact, while feature 9 contributes the most positively, offering an interpretable breakdown of the model's decision.

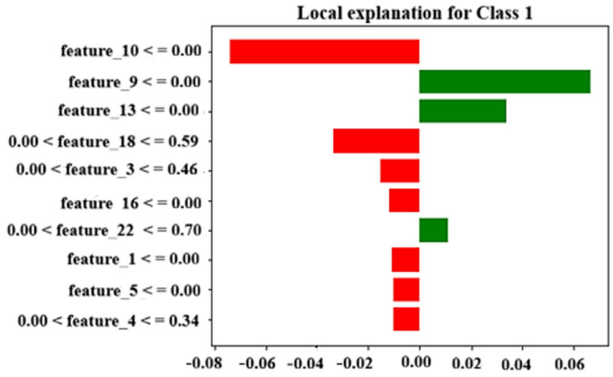

Fig. 4. Contribution by each feature for predicting Class 1.

The confusion matrix in Figure 5 demonstrates the model's performance in classifying gait patterns (HC, C, A, H, K) based on the distribution. For the HC gait pattern, the model achieved perfect classification with 12,934 correct predictions and no misclassifications. For the C gait pattern, the model correctly identified 12,680 instances but misclassified 43 instances as HC. The A gait pattern saw 12,848 correct predictions, with only 8 instances misclassified as K. In the case of the H gait pattern, the model made 12,637 correct predictions, misclassifying 1 instance as K and 77 instances as C. Lastly, for the K gait pattern, the model correctly identified 12,882 instances but misclassified 8 instances as A and 40 as H.

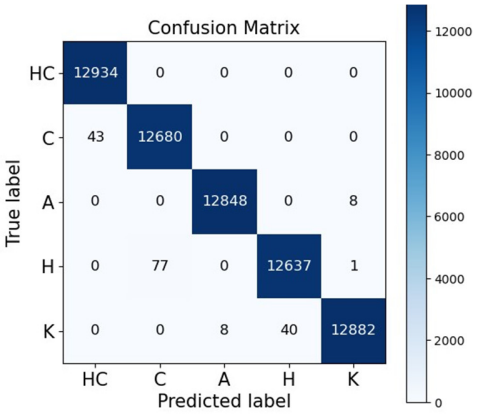

Fig. 5. Confusion matrix of the proposed network on the GaitRec dataset.

Figure 6 demonstrates the system's performance and convergence. The AUC values are 0.99 for Class 0 and 1.00 for the other classes, as shown in the ROC curve with near-perfect discrimination. The training and testing accuracies are stabilized around 99.72% and the strong stabilization is confirmed under FedAvg-SVRG. Training and testing losses of 0.0081 and 0.0078 indicate efficient learning of gait features, as shown in the loss curve. Overall, these results show the model's superior accuracy, robustness, and consistency in gait analysis with 25 epochs.

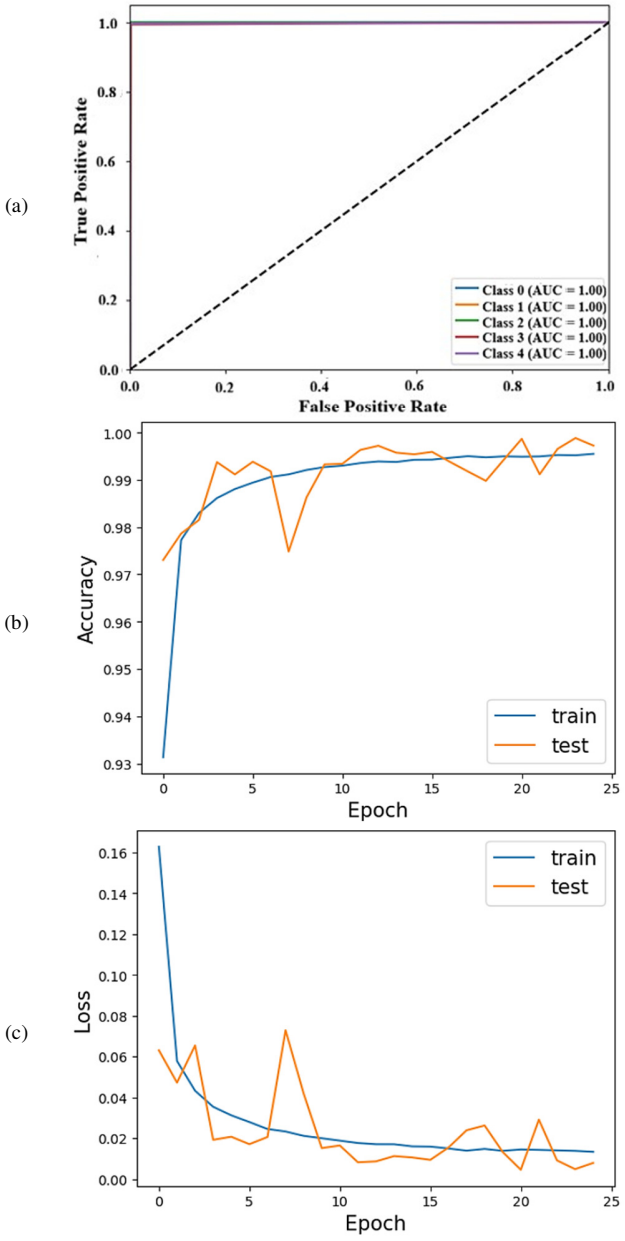

Fig. 6. Performance evaluation of the proposed network: (a) TPR vs FPR, (b) Accuracy, (c) Loss.

C. Results' Comparison

The following equations were used to calculate the performance metrics.

$$\text{Accuracy} = \frac{\text{TP+TN}}{\text{TP+TN+FP+FN}} \quad (17)$$

$$\text{Recall} = \frac{\text{TP}}{\text{TP+FN}} \quad (18)$$

$$\text{Precision} = \frac{\text{TP}}{\text{TP+FP}} \quad (19)$$

$$\text{F1 score} = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision+Recall}} \quad (20)$$

where TP denotes true positives, TN represents true negatives, and FP and FN denote false positives and negatives. Table I demonstrates the model's performance on all gait classes. Classes 0 (HC), 2 (A), and 4 (K) achieved perfect precision and recall. Class 1 (C) maintained very high precision (0.99) with few misclassifications, a perfect F1-score was obtained for all classes, while Class 3 (H) achieved 0.99 recall with minimal misclassifications. Overall, the model shows excellent reliability and consistency in classifying gait patterns with near-perfect accuracy.

TABLE I. CLASS-WISE PERFORMANCE EVALUATION OF THE PROPOSED MODEL

Class	Precision	Recall	F1-score	Accuracy	Support
0	1.00	1.00	1.00	1.0000	12,934
1	0.99	1.00	1.00	0.9966	12,723
2	1.00	1.00	1.00	0.9994	12,856
3	1.00	0.99	1.00	0.9939	12,715
4	1.00	1.00	1.00	0.9963	12,930

As shown in Table II, the proposed model provides higher precision, recall, F1 score, and accuracy in analyzing gait patterns compared to conventional models [28, 29]. These models used the GaitRec dataset to classify the gait based on GRF data, split in a ratio of 80:20. Impairments in the lower limb are detected and classified as healthy or pathological gait.

TABLE II. COMPARISON OF CONVENTIONAL MODELS WITH THE PROPOSED MODEL

Classifiers	Precision	Recall	F1-score	Accuracy
CATBOOST [28]	0.948	0.947	0.948	0.947
Extra Trees [28]	0.884	0.869	0.867	0.869
Random Forest [28]	0.870	0.851	0.849	0.851
LGBM [28]	0.697	0.627	0.656	0.627
Decision Tree [28]	0.592	0.591	0.591	0.591
XGBOOST [28]	0.627	0.569	0.557	0.569
MLP [29]	0.90	0.84	0.86	0.95
Proposed	0.9972	0.9972	0.9972	0.9972

Table III compares the accuracy of the proposed model with existing ones. The Force-plate open-source dataset with a split of 70:30 was used [30], and the gait classification was based on human fall risk assessment. The GaitRec dataset was used in [31-33] with GRF values, and a train/test split of 70:30. Sensors were placed under the surface for force measurement. An Arduino microcontroller and Python were used to find the peak GRF values for gait analysis [31]. In [32], a wearable auditory biofeedback system with sensors was used, while in [33], an automatic deep learning framework was used for gait identification. Table IV shows the results of a 10-fold cross-validation for the proposed model. The mean values of precision, recall, and F1-score were 99.46, 99.45, and 99.45% respectively. A mean accuracy of 99.45±0.14% was achieved, which is very close to the single run execution accuracy of 99.72%.

TABLE III. ACCURACY COMPARISON OF EXISTING MODELS WITH THE PROPOSED

Ref.	Classifier	Accuracy (%)
[30]	CNN, RNN, LSTM	99.3, 96.9, 98.3
[31]	1D-CNN	88.7
[32]	L-SVM	91
[33]	SVM, NB, KNN, GaitRec-Net	89.99, 55.2, 91.29, 91.6
Proposed	LSTM-LRP	99.72

TABLE IV. 10-FOLD CROSS-VALIDATION RESULTS

Fold	Val_Accuracy	Val_Loss	Precision	Recall	F1-score
1	0.9948	0.0148	0.9948	0.9948	0.9948
2	0.9954	0.0149	0.9954	0.9954	0.9953
3	0.9927	0.0173	0.9928	0.9927	0.9927
4	0.9929	0.0164	0.9929	0.9929	0.9929
5	0.9938	0.0235	0.9939	0.9938	0.9938
6	0.9943	0.0132	0.9943	0.9943	0.9943
7	0.9962	0.0135	0.9962	0.9962	0.9962
8	0.9943	0.0166	0.9943	0.9943	0.9943
9	0.9973	0.0085	0.9973	0.9973	0.9973
10	0.9938	0.0154	0.9939	0.9938	0.9938

Figure 7 shows the learning curves for training and validation accuracy and loss for fold 6 as an illustrative example. Similar trends were observed in all other folds.

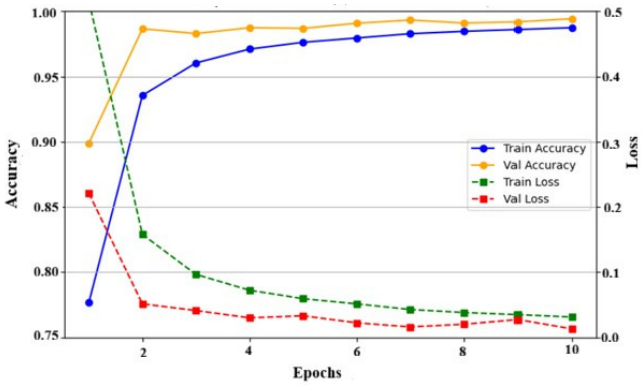

Fig. 7. Learning curves for fold 6.

IV. CONCLUSION

This study integrates GRF features from the GaitRec dataset with a deep learning LSTM-LRP architecture to classify gait as normal or pathological. Instead of SGD, FedAvg is combined with SVRG to reduce variance in gradient updates, accelerating training across distributed nodes. The proposed approach also places a strong emphasis on interpretability, which is critical for clinical applications. The key novelty of this method is that the temporal dependencies in the gait cycle are captured by an LSTM, while a detailed layer-to-layer visualization of which features contribute the most to the model prediction is performed using two enhanced LRP rules, LRP-ε and LRP-αβ. LIME is used to improve the interpretability of gait prediction. Compared to conventional models, the proposed model achieved a superior accuracy of 99.72% for a single run execution and 99.45±0.14% when cross-validated with 10 folds.

REFERENCES

[1] S. P. Lalehzarian, A. K. Gowd, and J. N. Liu, "Machine learning in orthopaedic surgery," *World Journal of Orthopedics*, vol. 12, no. 9, pp. 685–699, Sep. 2021, https://doi.org/10.5312/wjo.v12.i9.685.

[2] R. Das, S. Paul, G. K. Mourya, N. Kumar, and M. Hussain, "Recent Trends and Practices Toward Assessment and Rehabilitation of Neurodegenerative Disorders: Insights From Human Gait," *Frontiers in Neuroscience*, vol. 16, Apr. 2022, https://doi.org/10.3389/fnins.2022.859298.

[3] Y. Matsushita, D. T. Tran, H. Yamazoe, and J. H. Lee, "Recent use of deep learning techniques in clinical applications based on gait: a survey," *Journal of Computational Design and Engineering*, vol. 8, no. 6, pp. 1499–1532, Oct. 2021, https://doi.org/10.1093/jcde/qwab054.

[4] R. Griffiths *et al.*, "Guideline for the management of hip fractures 2020," *Anaesthesia*, vol. 76, no. 2, pp. 225–237, 2021, https://doi.org/10.1111/anae.15291.

[5] B. Su, C. Smith, and E. Gutierrez Farewik, "Gait Phase Recognition Using Deep Convolutional Neural Network with Inertial Measurement Units," *Biosensors*, vol. 10, no. 9, Sep. 2020, Art. no. 109, https://doi.org/10.3390/bios10090109.

[6] A. S. Alharthi, S. U. Yunas, and K. B. Ozanyan, "Deep learning for monitoring of human gait: A review," *IEEE Sensors Journal*, vol. 19, no. 21, pp. 9575–9591, 2019.

[7] J. Cuadrado, F. Michaud, U. Lugrís, and M. Pérez Soto, "Using Accelerometer Data to Tune the Parameters of an Extended Kalman Filter for Optical Motion Capture: Preliminary Application to Gait Analysis," *Sensors*, vol. 21, no. 2, Jan. 2021, Art. no. 427, https://doi.org/10.3390/s21020427.

[8] I. K. Jalata, T. D. Truong, J. L. Allen, H. S. Seo, and K. Luu, "Movement Analysis for Neurological and Musculoskeletal Disorders Using Graph Convolutional Neural Network," *Future Internet*, vol. 13, no. 8, Aug. 2021, Art. no. 194, https://doi.org/10.3390/fi13080194.

[9] J. Kubicek, F. Tomanec, M. Cerny, D. Vilimek, M. Kalova, and D. Oczka, "Recent Trends, Technical Concepts and Components of Computer-Assisted Orthopedic Surgery Systems: A Comprehensive Review," *Sensors*, vol. 19, no. 23, Jan. 2019, Art. no. 5199, https://doi.org/10.3390/s19235199.

[10] N. Kour, S. Gupta, and S. Arora, "A Survey of Knee Osteoarthritis Assessment Based on Gait," *Archives of Computational Methods in Engineering*, vol. 28, no. 2, pp. 345–385, Mar. 2021, https://doi.org/10.1007/s11831-019-09379-z.

[11] F. Horst, S. Lapuschkin, W. Samek, K. R. Müller, and W. I. Schöllhorn, "Explaining the unique nature of individual gait patterns with deep learning," *Scientific Reports*, vol. 9, no. 1, Feb. 2019, Art. no. 2391, https://doi.org/10.1038/s41598-019-38748-8.

[12] Z. Zhang *et al.*, "Deep learning-enabled triboelectric smart socks for IoT-based gait analysis and VR applications," *npj Flexible Electronics*, vol. 4, no. 1, Oct. 2020, Art. no. 29, https://doi.org/10.1038/s41528-020-00092-7.

[13] Y. Zhang and Y. Ma, "Application of supervised machine learning algorithms in the classification of sagittal gait patterns of cerebral palsy children with spastic diplegia," *Computers in Biology and Medicine*, vol. 106, pp. 33–39, Mar. 2019, https://doi.org/10.1016/j.compbiomed.2019.01.009.

[14] M. Hnatiuc, O. Geman, A. G. Avram, D. Gupta, and K. Shankar, "Human Signature Identification Using IoT Technology and Gait Recognition," *Electronics*, vol. 10, no. 7, Jan. 2021, Art. no. 852, https://doi.org/10.3390/electronics10070852.

[15] E. Warmerdam, M. Orth, T. Pohlemann, and B. Ganse, "Gait Analysis to Monitor Fracture Healing of the Lower Leg," *Bioengineering*, vol. 10, no. 2, Feb. 2023, Art. no. 255, https://doi.org/10.3390/bioengineering10020255.

[16] A. Fändriks, R. Tranberg, J. Karlsson, M. Möller, and R. Zügner, "Gait biomechanics in patients with intra-articular tibial plateau fractures – gait analysis at three months compared with age- and gender-matched healthy subjects," *BMC Musculoskeletal Disorders*, vol. 22, no. 1, Aug. 2021, Art. no. 702, https://doi.org/10.1186/s12891-021-04577-y.

[17] Y. L. Ng, X. Jiang, Y. Zhang, S. B. Shin, and R. Ning, "Automated Activity Recognition with Gait Positions Using Machine Learning Algorithms," *Engineering, Technology & Applied Science Research*, vol. 9, no. 4, pp. 4554–4560, Aug. 2019, https://doi.org/10.48084/etasr.2952.

[18] A. I. Bulbul, U. Mayetin, and S. Kucuk, "Development of an Ankle Sensor for Ground Reaction Force Measurement in Intelligent Prosthesis," *Engineering, Technology & Applied Science Research*, vol. 14, no. 4, pp. 15161–15170, Aug. 2024, https://doi.org/10.48084/etasr.7430.

[19] J. Perazzone, S. Wang, M. Ji, and K. S. Chan, "Communication-efficient device scheduling for federated learning using stochastic optimization," in *IEEE INFOCOM 2022-IEEE Conference on Computer Communications*, 2022, pp. 1449–1458.

[20] R. Johnson and T. Zhang, "Accelerating Stochastic Gradient Descent using Predictive Variance Reduction," in *Advances in Neural Information Processing Systems*, 2013, vol. 26.

[21] R. Xin, S. Kar, and U. A. Khan, "Decentralized stochastic optimization and machine learning: A unified variance-reduction framework for robust performance and fast convergence," *IEEE Signal Processing Magazine*, vol. 37, no. 3, pp. 102–113, 2020.

[22] P. J. Werbos, "Backpropagation through time: what it does and how to do it," *Proceedings of the IEEE*, vol. 78, no. 10, pp. 1550–1560, 1990.

[23] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural computation*, vol. 9, no. 8, pp. 1735–1780, 1997.

[24] A. Binder, S. Bach, G. Montavon, K. R. Müller, and W. Samek, "Layer-Wise Relevance Propagation for Deep Neural Network Architectures," in *Information Science and Applications (ICISA) 2016*, vol. 376, K. J. Kim and N. Joukov, Eds. Springer Singapore, 2016, pp. 913–922.

[25] L. Arras *et al.*, "Explaining and Interpreting LSTMs," in *Explainable AI: Interpreting, Explaining and Visualizing Deep Learning*, vol. 11700, W. Samek, G. Montavon, A. Vedaldi, L. K. Hansen, and K. R. Müller, Eds. Springer International Publishing, 2019, pp. 211–238.

[26] B. Horsak, Djordje Slijepcevic, A. M. Raberger, C. Schwab, M. Worisch, and M. Zeppelzauer, "GaitRec: A large-scale ground reaction force dataset of healthy and impaired gait." figshare, 2020, https://doi.org/10.6084/M9.FIGSHARE.C.4788012.V1.

[27] B. Horsak, D. Slijepcevic, A. M. Raberger, C. Schwab, M. Worisch, and M. Zeppelzauer, "GaitRec, a large-scale ground reaction force dataset of healthy and impaired gait," *Scientific Data*, vol. 7, no. 1, May 2020, Art. no. 143, https://doi.org/10.1038/s41597-020-0481-z.

[28] D. Jani *et al.*, "An Efficient Gait Abnormality Detection Method Based on Classification," *Journal of Sensor and Actuator Networks*, vol. 11, no. 3, Jun. 2022, Art. no. 31, https://doi.org/10.3390/jsan11030031.

[29] M. M. Mulwa, R. W. Mwangi, and A. Mindila, "GMM-LIME explainable machine learning model for interpreting sensor-based human gait," *Engineering Reports*, vol. 6, no. 10, Oct. 2024, Art. no. e12864, https://doi.org/10.1002/eng2.12864.

[30] M. Savadkoohi, T. Oladunni, and L. A. Thompson, "Deep neural networks for human's fall-risk prediction using force-plate time series signal," *Expert Systems with Applications*, vol. 182, Nov. 2021, Art. no. 115220, https://doi.org/10.1016/j.eswa.2021.115220.

[31] S. A. Boompelli and S. Bhattacharya, "Design of a telemetric gait analysis insole and 1-D convolutional neural network to track postoperative fracture rehabilitation," in *2021 IEEE 3rd Global Conference on Life Sciences and Technologies (LifeTech)*, 2021, pp. 484–488.

[32] M. Iber *et al.*, "Mind the Steps: Towards Auditory Feedback in Tele-Rehabilitation Based on Automated Gait Classification," in *Audio Mostly 2021*, Sep. 2021, pp. 139–146, https://doi.org/10.1145/3478384.3478398.

[33] C. Pandey *et al.*, "GaitRec-Net: A Deep Neural Network for Gait Disorder Detection Using Ground Reaction Force," *PPAR Research*, vol. 2022, pp. 1–10, Aug. 2022, https://doi.org/10.1155/2022/9355015.