EAI Endorsed Transactions
on Pervasive Health and Technology

Research Article **EAI.EU**

The Future of Fall Prevention: Integrating OpenPose with Cutting-Edge ML Models

Shina Samuel Kolawole[1], Gautam Siddharth Kashyap[2,*], Olamide Emmanuel Kolawole[3], and Miao Yu [4]

[1, 4]University of Lincoln, UK
[2]IIIT Delhi, New Delhi, India
[3]Aston University, UK

Abstract

The research paper aims to assess ML models for video-recorded gaits with an aim of classifying people into high or low risks to fall groups. Several ML algorithms were tried employing OpenPose for CV, with RF showing the best outcomes: 93% accuracy along with F1-score as well as balanced sensitivity (93.50%) as well as specificity (92.50%). Some important determining factors were speed per unit distance, angle among other statistical measures. In comparison to wearables-based DL approaches plus traditional fall detection methods, this study's approach showed higher accuracy and adaptability within health care settings.

Keywords: Deep Learning, Fall, Healthcare, Machine Learning, OpenPose

Received on 26 August 2024, accepted on 01 November 2024, published on 02 April 2025

doi: 10.4108/eetpht.11.9013

1. Introduction

Particularly in old people, slips are a big problem for the public health with severe harm and less life quality. The traditional means for example questionnaires or clinical tests are subjective and usually wrong. Machine Learning (ML) infused with Computer Vision (CV) provides a more objective way of doing this by looking at subtle gait features to determine risk of falling thus allowing for early intervention and better results in patients.CV is one such technology that is facilitating these developments. OpenPose[†] is a CV method used to extract useful data from video recordings. It works by estimating body posture accurately during different frames within a sequence hence pin-pointing all major joints' locations over time so as to capture elaborate details about walking like joint angles, movement paths as well as variations temporally. ML algorithms are designed to recognize patterns contained in complex datasets where most other systems fail. By feeding them with rich gait information obtained through OpenPose—they become aware of slight changes which usually accompany increased falling danger. For instance, 1) Shorter strides could mean weak or unstable lower limbs, 2) Noticeable difference in stride lengths or swing times between left and right legs points towards abnormality while walking which may result balance problems, 3) Slower speeds are associated with frailty syndrome characterised by multiple falls among aged individuals, and 4) Minimum foot clearance during the swing phase indicates impaired balance and higher trip risk. Supervised ML algorithms trained on labeled gait videos can learn the intricate connections between walking features and fall likelihood, enabling them to assess new gait recordings for fall risk accurately.

Because machine learning (ML) is unbiased and can handle large volumes of gait data, it is excellent at predicting falls in older persons. In contrast to subjective evaluations, machine learning (ML) yields objective forecasts, which facilitate the prompt identification of high-risk patients and

[*]Corresponding author. Email: officialgautamgsk.gsk@gmail.com

[†] https://github.com/CMU-Perceptual-Computing-Lab/openpose

the implementation of customized exercises or prescription modifications. This can greatly lower the number of injuries caused by falls and enhance general health. Also, by examining unique gait patterns and creating focused interventions, machine learning (ML) enables the creation of customized fall prevention plans. This project intends to investigate important issues with fall risk prediction by collecting gait data using OpenPose.

1. Researchers aim to compare the effectiveness of different ML algorithms in classifying fall risk based on gait features extracted from videos, evaluating their accuracy and efficacy in identifying individuals at risk.
2. The research seeks to identify the optimal ML model for fall risk prediction using OpenPose data and pinpoint the most significant gait features contributing to the model's effectiveness.

The work enhances trustworthy fall risk prediction models by tackling these issues, which would result in more potent preventative measures and a safer future for the elderly. The organization of the study is as follows: Section 2 contains related works; Section 3 contains materials and methods; Section 4 contains experimental analysis; Section 5 contains result analysis; and Section 6 contains conclusions and future research.

2. Related Works

The devastation caused from mishappenings or accidents is commonly associated with falls that severely affect physical (1) and mental well-being in old-age people (2). Social isolation (3) and decreased physical activity (4) can result from fear of falling—these factors further raise the risk of falling—thereby—creating a vicious circle. While traditional clinical assessments done by healthcare specialists are good at identifying major risk elements—however—they could miss subtle changes related to gait, balance or cognitive function which may indicate an increased dangerousness for dropping down early. Recently, technology has enabled more accurate fall risk predictions using wearable sensors like accelerometers or gyroscopes in wristbands/anklets, which record seniors' real-time movements and activity levels, allowing ML algorithms to analyze vast sensor data and identify hidden trends for precise fall risk predictions(5,6). ML provides powerful tools against falls, especially for seniors, by finding patterns from massive sensor data. It detects subtle changes in gait, like stride variability or speed, which might precede falls, identifying issues humans might miss. Moreover, ML can connect seemingly unrelated data streams to enhance fall risk predictions(7). To illustrate, ML considers various factors like activity patterns and medication consistency, offering a broader perspective on health issues and fall risks compared to traditional methods(8,9). ML enables more accurate fall risk detection by comparing extensive data from at-risk and non-at-risk individuals, continuously improving itself with new information, thereby enhancing fall prediction with each encounter and adapting to live situations through exposure to continuous data(10). Early identification of fall risks allows healthcare providers to intervene before falls occur, offering personalized workouts to enhance strength and balance, adjusting medications to mitigate side effects, improving spatial awareness through vision enhancement methods, and modifying home environments to eliminate potential hazards, thus significantly reducing the frequency and severity of falling accidents and their devastating outcomes.

2.1. Fall Risk Prediction

People with neurological conditions face a heightened risk of falls due to balance and coordination issues, which can lead to life-changing incidents such as hip fractures, resulting in limited mobility, increased dependence on caregivers, and overall deterioration in health status(11). Social isolation is promoted by fear of falling which leads to lower quality life (12). In order to tackle this problem, scientists are engaged in cutting-edge investigations into fall prediction through use of advanced ML particularly DL such as (13,14). This is an evolution from reactive measures taken after a fall to proactive identification of those at high risk even before they take place; hence the paradigm shift in fall prevention. A recent trial exemplifies such possibilities such as (15). By using specialized sensors for monitoring people with neurological problems while they move around, (16)—were able to capture many kinds of information. In predicting falls, these—(17)—achieved an accuracy rate as high as 90% by using DL algorithms. This breakthrough promises next-gen wearable devices for fall prevention, with concealed sensors tracking movement patterns, enabling real-time risk assessments powered by Deep Learning algorithms, although challenges remain, highlighting the importance of translating accurate predictions into actionable interventions, where Machine Learning excels by identifying specific risk-associated motion patterns(16). By providing detailed insights, healthcare providers can design tailored interventions such as physical therapy programs addressing specific weaknesses identified by algorithms and adjusting medications to reduce balance-affecting side effects; however, limitations in current fall risk prediction methods, including small sample sizes and narrow focus on sensor-captured movement patterns, overlook crucial risk factors like drug use and vision impairment, potentially leading to inaccurate predictions and missed prevention opportunities. Moreover, the complexity and opacity of certain ML models hinder physicians' understanding and integration into clinical workflows, posing challenges in patient communication and acceptance.

2.2. Gait Analysis

As a result of CV—gait analysis has taken a new dimension. Gait analysis with video-based CV consists of different approaches that employ advanced image processing, ML, and depth-sensing technologies such as (18). Through these methods, which rely on key-point tracking to monitor joint angles, limb movements, and postural changes during the step cycle, gait mechanics and anomalies can be effectively explored, enhancing patient care and rehabilitation efforts(19). Incorporating depth sensing devices with three-dimensional reconstruction techniques enhances gait analysis by providing detailed information on joint angles, step lengths, and stride width, enabling a comprehensive understanding of how anatomical elements cooperate during walking. Consequently—(20)—are able to get an overall viewpoint of gait physiology important for correct diagnosis as well as effective treatment for different musculoskeletal pathologies. Other motion capture systems used in the assessment of gait are very useful such as (21,22). (23) deploys markers on specific anatomical areas by using synchronized cameras to record the movement of bodies. This meticulous approach results in highly accurate three-dimensional models, facilitating precise quantification and comprehensive study of biomechanical phenomena, aiding in diagnosing and monitoring rehabilitation progress by differentiating between gait phases and categorizing walking styles using ML and pattern detection techniques(24) are able to identify subtle irregularities by training algorithms on different data sets, thus allowing for prompt intervention and tailored treatment strategies. Gait kinematics and kinetics investigate the forces applied during walking as well as the angles used in this process. Such an investigation brings out the complexities of human movement intricacies that may cause issues such as (25). Real-world gait tracking with wearable gadgets enables ecologically valid investigations for evaluation, forming the basis for therapeutic interventions.

2.3. Research Gaps

Current research on fall risk prediction via OpenPose using ML approaches requires validation and real-world deployment of created models. Many studies lack validation with an independent dataset, crucial for confirming the generalizability and precision of the models in practical settings(26). The evaluation of OpenPose-based fall risk prediction models for practical applications is hampered by their limited demonstration in clinical or healthcare settings. Therefore, their deployment and enhanced validation are required before they can be integrated into clinical practice. The necessity for improvements in fall risk prediction techniques is highlighted by previous studies that were divided into three categories: wearables, ambient fusion-based methods, and vision-based systems. These studies use a variety of sensors for data collection and prediction, but they all face difficulties like discomfort, privacy issues, environmental noise, and inaccurate results. For instance, (27) have questioned the reliability of using fall simulation data as a representative of real world falls. Major shortcomings and limitations exist in present research on OpenPose-based machine learning algorithms for fall risk prediction. These include small sample sizes, inadequate consideration of pertinent variables beyond gait-related data, improper validation with independent datasets, disregard for the interpretability of ML models, and a dearth of real-world applications in clinical settings. By solving these problems, fall risk prediction models can become more reliable and effective. This can improve fall prevention programs and ensure the safety of those who are at-risk while also addressing ethical concerns about privacy and subject protection.

3. Materials and Methods

3.1. Data Analysis

To anticipate fall risk using ML—experts performed a systematic gathering and readying of data. The main aim was to use gait analysis for predicting the probability of a person falling. The dataset used in this study was composed of videos showing people walking that were obtained from (28)—Mendeley Data—an open-access repository—and included patients diagnosed with knee osteoarthritis or Parkinson's disease at different stages as shown in Figs. 1 and 2. The study included 107 patients categorized into high and low fall risk groups, with data extracted from videos by analyzing each frame individually using OpenPose software, identifying 25 key body points per frame (excluding head region details) and resulting in rich datasets spatially representing the points across time in each video clip. Prior to utilization in ML models, the extracted data underwent preprocessing stages such as filtering out unnecessary information and normalizing values to address scale differences, followed by the design of feature vectors capturing movement patterns observed during various activities, comprising velocities, distances, and statistical measures to create a mathematical representation understood by ML models. Observing velocity changes across frames along limbs from the shoulder joint down allows for assessment of energy efficiency during movement, while distances between key points and statistical features reveal abnormal gait patterns potentially leading to falls; subsequently, assigning labels of zero for low-risk individuals and one for high-risk individuals based on their likelihood of falling down categorized the feature vectors for analysis. In supervised ML, labeling the dataset allows the model to learn associations between characteristics and outcomes, followed by splitting the labeled dataset into training and testing sets, with the training set typically comprising about 70% of samples to build models and ensure they capture general patterns rather than overfitting specific cases, essential for evaluating model performance on unseen data during the testing phase. Reassembling situations from the majority class until they corresponded to the total count of the minority class in both training and testing datasets was the method for balancing the classes after the initial dataset showed a severe imbalance, with a significantly higher number of records indicating higher fall probabilities, which can lead to overfitting issues. This ensures more dependable leads to for training ML models for fall prediction.

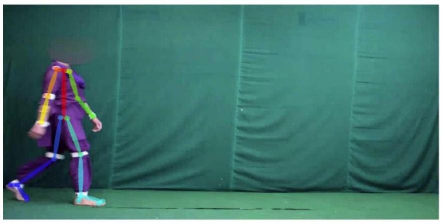

Figure 1. A still image taken from a video showing a person with a low fall risk walking

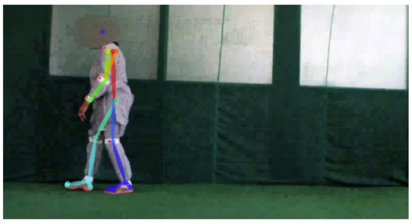

Figure 2. A still image taken from a video showing a person with a high fall risk walking

3.2. Model Analysis

To prevent overfitting and refine ML models, we employed the grid search cross-validation method, testing various parameter combinations for each model until finding the best configuration, such as different kernel types and gamma values for Support Vector Classifier (SVC), utilizing 10-fold cross-validation where the training data is divided into 10 smaller folds for evaluation. Ten iterations of the process guarantee that every fold is evaluated once, lowering the possibility of overfitting and facilitating the assessment of the model's performance on unobserved data. In KNNs, for example, the number of neighbors and the weighting method were changed; in RF, different numbers of trees and maximum depths were tested; in MLP, different hidden layer configurations and alpha values were investigated during grid search. Feature selection plays a crucial role in effective machine learning by leveraging domain knowledge about human movement patterns, including significant indicators like joint angles and velocities, and using a data-driven approach through empirical analysis to identify the most informative features for predicting fall risk across various algorithms through cross-validation. The best-performing model on training data was chosen based on metrics like accuracy, F1-score, specificity, and sensitivity.

4. Experimental Analysis

4.1. OpenPose—Extract 2D Pose from Low Risk Falling Group Video Gaits

We use PowerShell to start the OpenPose library. This will allow us to run a demonstration that examines video recordings. In this demonstration, OpenPose detects keypoints for multiple persons in real-time. What it does is gather information about where people are and what they are doing by looking at the major joints and landmarks on their bodies. All this requires GPU or any other computational resource with high processing power as well as being able to identify videos tagged with "lowfallriskgroup" among other things; PowerShell command lets initiate OpenPose while specifying files consecutively and when executed each command begins analyzing immediately — generating JSON files filled with data about movement and posture which can be used for further research into human kinetics or biomechanics.

4.2. OpenPose—Extract 2D Pose from High Risk Falling Group Video Gaits

The OpenPose software framework is a Windows PowerShell environment instruction that uses video recordings marked as “highfallriskgroup” to assess people’s posture dynamics and walking patterns detected in gait videos. You can use the command line parameters of OpenPoseDemo.exe to identify input and output directories, as well as detect key points through GPU computing, where every command has time-based measurement on how long it took for analysis to finish. In this case, PowerShell employs systematic processes which automate analytical pipelines thereby enabling holistic video evaluation and extraction of biomechanical data while demonstrating potentiality of OpenPose in movement study and biomechanics; this is most applicable during fall risk assessment along with motion analysis.

4.3. Preprocessing—Extracted 2D Pose for Machine Learning

In order to measure fall risk—this method transforms human posture data from JSON files to NumPy[‡] arrays. Data is arranged according to fall risk group. It initially configures the file directories for the NumPy and JSON data. Next, a JSON file is converted to a NumPy array using the function *json_to_numpy*—which only retains the *x* and *y* coordinates of the pose keypoints.The script iterates through JSON files from low and high fall risk categories,

[‡] https://numpy.org/

converting them into NumPy arrays to prepare pose keypoint data for analysis, efficiently organizing them into appropriate folders, and categorizing them based on risk level for further modeling purposes. A function named *json_to_numpy* is used to transform the JSON data into a NumPy array that is saved as a NumPy data file (*.npy*) in corresponding directory within output folder designated by fall risk group. After processing all JSONs for both groups—it continues loading saved NumPies. Loading is done using a function called *load_numpy_files*—which takes folder path as input—reads all *.npy* files found within that folder and its subdirectories then returns list containing loaded data. This function is used to load data for each fall risk group before converting them into NumPy arrays.

Next step entails processing *combined_data* from both groups whereby first we merge arrays 'along' specific axis so as to create single array with low risk followed by high risks contents. Following normalization using the MinMaxScaler function to scale keypoints between 0 and 1, velocity is calculated by taking the difference between consecutive time steps along each dimension, providing feature vectors for analysis or input into ML algorithms for various tasks related to analyzing such information.Finally, script prints velocity data at particular moment like 10000th step. More actions are performed on generated labels and feature vectors such as reshaping, labelling, splitting and printing. Labelling assigns 0 value to "low fall risk" group and 1 to "high fall risk" group. These labels are then concatenated into single array. Feature vectors are reshaped from 2D to 1D array. Data is split into training and testing sets using *train_test_split* function—which allows one specify percentage of data used for testing while ensuring reproducibility through random seed. Afterwards, it displays shapes of training and testing feature vectors along with their corresponding labels—then—may print some data examples so as to give better picture about its structure. In order to understand how fall risks labels are distributed—data visualization techniques are used. The script generates a bar chart to show the number of times each label category ("Low Fall Risk" and "High Fall Risk") occurs.In order to address the potential negative effects of class imbalance on ML model performance, visualization combines training and testing data to analyze the distribution of fall risk labels across the entire dataset through resampling. The script finds the minority and majority categories within training samples, undersamples the majority instances to match the number of cases in the minority group, and uses similar techniques to equalize class distributions in the test dataset to enable appropriate model evaluation. Based on their frequency in the test labels, the minority and majority classes are determined in this step. The instances from the majority class are then replicated using resampling to match the minority class's count. Ultimately, the original minority class samples and the resampled majority class samples are mixed, and the labels are added together appropriately. In order to guarantee unpredictability in the sample and label order inside the balanced test dataset, shuffling is used. The script ends by verifying the attained balance in class depiction by publishing the dimensions of the balanced test dataset.

5. Result Analysis

5.1. Feature Analysis

The study examines challenges with SVM classification using features derived solely from keypoint distances, highlighting the necessity of feature engineering and exploring alternative modeling techniques for successful classification, evaluating SVM classifier performance with optimized hyperparameters on a dataset where features are calculated from keypoint distances.Despite achieving a mean cross-validation score of 0.76, indicating some learning capability, further examination reveals significant underfitting in the model, with uniform predictions across all data folds and no true positives for class 0.0, rendering precision, recall, and F1-score calculations impossible, highlighting the need for model refinement. In other words, if we see someone saying "We can't believe it! We got zero true positives", just know that we had used SVMs to separate classes purely based on keypoints' distance and no wonder why everything failed including confusion matrix showing zero true positives for class 0.0. These findings suggest that using keypoints alone as features for SVM classification does not work well under these specific conditions or criteria used here—may be wrong too—but—whichever way it clearly doesn't give best results or any result at all. Sometimes—hence needs rethinking altogether what were we doing of making such choices anyway? At this point one might ask us—what should we consider then? Here are some possible options such as reassessing features—what other features can we use or combine in order to better represent the data and make them more differentiable between classes?

```
Best parameters for SVM: {'C': 0.1}
Mean cross-validation score for SVM: 0.76
Accuracy scores for SVM: [0.76131572 0.76131572 0.76131572 0.76131572 0.7613157
2 0.76131572
 0.76131572 0.76159685 0.76159685 0.76152981]
              precision    recall  f1-score   support

         0.0       0.00      0.00      0.00      8487
         1.0       0.76      1.00      0.86     27082

    accuracy                           0.76     35569
   macro avg       0.38      0.50      0.43     35569
weighted avg       0.58      0.76      0.66     35569
```

Figure 3. Distance between important locations as the feature in SVM results

Maybe we should stop thinking about distances only but also include angles, geometrical relationships among others. Alternative modeling strategies—if SVM is not working as expected—then—what other classification algorithms could be tried out considering this particular kind of data. RF, GB or even NNs may be worth giving a shot depending on the nature of our problem. There are important things that need to be considered if accuracy is to be improved in similar situations. The study goes ahead by carrying out another experiment which involves using wider range of features consisting of angles, mean, standard deviation and distances between keypoints as shown in Fig. 4. Even though the model gets cross-validation score (0.56)—it still has issues with balanced classification across all the classes. There exists significant imbalance both in terms of accuracy as well as individual class metrics such as precision, recall and F1-score too—all show great imbalance indeed. Suppose there was a confusion matrix for this experiment—its likely going to indicate high misclassification rate for class 0.0 meaning that no matter how we try to add onto existing feature set it will never be enough for good separation between these two groups. Therefore, this study brings out the necessity of being careful when selecting and designing features while using SVMs in keypoint distance classification tasks. These findings suggest trying different feature engineering techniques and investigating alternative models so as to overcome limitations encountered during both experiments by making accurate classifications within such scenarios through careful crafting of feature space coupled with more robust modeling techniques that can greatly enhance the ability.

```
Best parameters for SVM: {'C': 1}
Mean cross-validation score for SVM: 0.56
Accuracy scores for SVM: [0.57067138 0.5606596  0.56419317 0.55889282 0.53859753 0.57159694
 0.55391868 0.56570418 0.55981143 0.54331173]
              precision    recall  f1-score   support

         0.0       0.60      0.35      0.44      8487
         1.0       0.54      0.77      0.63      8487

    accuracy                           0.56     16974
   macro avg       0.57      0.56      0.54     16974
weighted avg       0.57      0.56      0.54     16974

Confusion matrix:
 [[2978 5509]
 [1981 6506]]
Cohen's Kappa: 0.11747378343348647
```

Figure 4. Angles, mean, standard deviation, and the separation between important points are combined to form the features in SVM findings

5.1. Machine Learning Analysis

The SVM, DT, RF, KNN and MLP classifiers that have been implemented are evaluated in this section. These classifiers were tested using 10-fold cross validation, the accuracy on the test set, F1 score, sensitivity and specificity. Also, the hyperparameters were optimized through grid search and then selecting best model with it. Finally predictions were made on a test set by using best model. The features used to build these ML classifiers are key point's velocity. The Fig. 5 shows enhanced performance of DT with best hyperparameters. The optimal parameter for DT is {'max_depth': None}—which allows growing tree without setting any maximum depth beforehand. The model showed good data classifying ability with 83.90% mean accuracy from cross-validation. Its classification performance was indicated by its F1 score of 0.819—which balances precision and recall. According to confusion matrix—this model's performance is represented by having 2161 true negatives, 2087 true positives, 424 false positives and 498 false negatives. Test set accuracy remained at around 82%. Additionally, both classes (positive/negative) had high specificity/sensitivity scores—84% and 81% respectively.

```
Best parameters for DecisionTreeClassifier: {'max_depth': None}
Best accuracy score for DecisionTreeClassifier: 0.8442904605404784
Accuracy per fold: [0.83827265 0.83996613 0.82218459 0.85182049 0.8440678  0.82881356
 0.83305085 0.86186441 0.85338983 0.81694915]
Mean accuracy: 0.839037945435497
F1 Score: 0.8190737833594975
Confusion Matrix:
[[2161  424]
 [ 498 2087]]
Accuracy on the test set: 0.8216634429400387
Specificity (Label 0): 0.8359767891682786
Sensitivity (Label 0): 0.8073500967117988
Specificity (Label 1): 0.8073500967117988
Sensitivity (Label 1): 0.8359767891682786
```

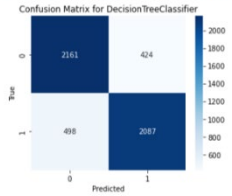

Figure. 5. Results of a DT

Optimized hyperparameter KNN classification model is shown in Fig. 6. Optimal configuration for the KNNs classifier is—{'n_neighbors': 3, 'weights': 'distance'}. Cross-validation mean accuracy of 84.65% showed remarkable results. A harmonious balance between precision and recall was achieved by getting F1 score equalling to 0.842 which confirms success in classification of data. The confusion matrix indicates how well our model performed where we have 2089 true negatives, 2238 true positives, 496 false positives and 347 false negatives. The test set accuracy was high at about 83.69%. Model also had specificity ratings of 81% for both classes showing that it could classify positive and negative cases.

```
Best parameters for KNeighborsClassifier: {'n_neighbors': 3, 'weights': 'distance'}
Best accuracy score for KNeighborsClassifier: 0.8464914106115187
Accuracy per fold: [0.84758679 0.85012701 0.85012701 0.85351397 0.85508475 0.83728814
 0.84322034 0.84491525 0.85       0.83305085]
Mean accuracy: 0.8464914106115187
F1 Score: 0.8415115623237451
Confusion Matrix:
[[2089  496]
 [ 347 2238]]
Accuracy on the test set: 0.8369439071566731
Specificity (Label 0): 0.8081237911025145
Sensitivity (Label 0): 0.8657640232108317
Specificity (Label 1): 0.8657640232108317
Sensitivity (Label 1): 0.8081237911025145
```

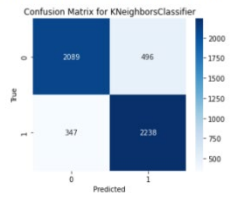

Figure 6. Results of a KNNs

The best hyperparameters for SVM were found to be {'C': 10, 'gamma': 10, 'kernel': 'rbf'} as shown in Fig. 7. Each fold in cross-validation had an average accuracy rating of 86.73%. F1-score is a notable metric for balancing precision against recall which was recorded as 0.8594. Confusion matrix—it correctly predicted class 0 in 2338 instances and class 1 in 2134 instances. Class 0 experienced 247 misclassifications while class 1 went through 451 misclassification errors. The model had another test set accuracy of 86.50%. Specificity measures reveal that when it came to telling apart two different groups or types such as good from bad—then—this algorithm did so with a high degree of reliability—since its values were between 82-90%.

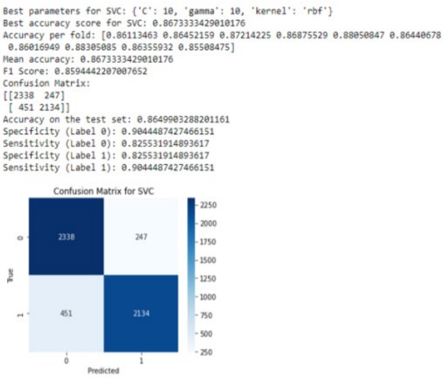

Figure 7. Results of a SVM

MLP performance is examined with adjusted hyperparameters as shown in Fig. 8. The optimal MLP configuration is {'alpha': 0.0001, 'hidden_layer_sizes': (100, 50)}. This choice gave the model a cross-validation mean accuracy of 83.25%—thus demonstrating its ability to classify data well enough. The classification performance of the model was 0.852—measured by precision and recall. True negatives were 2195, true positives were 2209, false positives were 390, and false negatives were 376. The test set accuracy of the model was also equal to 85.18%. Its capability to classify positive and negative cases can be seen where both classes have sensitivity and specificity scores of around 85%.

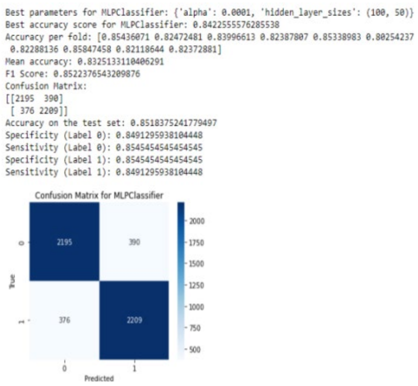

Figure 8. Results of a MLP

RF does a good job of tuning the hyperparameters as shown in Fig. 9. The optimal RF configuration is {'max_depth': None, 'n_estimators': 100}. Data classification ability of the model is high as indicated by its mean cross-validation accuracy which is 92.99%. The classifier's F1 score was 0.930—this shows that it can perform well in classification. The model is robust with 2417 true negatives, 2391 true positives, and 168 false positives along with 194 false negatives. Test set accuracy was also equal to 92.99%. Besides, among all other classifiers tested on our dataset ML algorithms—both positive predictive value and negative predictive value of RF were approximately equal at about 93%. It did exceptionally well in classifying the problem under consideration with given hyperparameters—a very large number indeed.

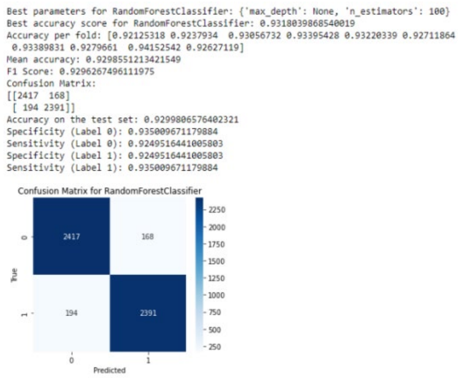

Figure 9. Results of a RF

Each classifier has been tuned using specific hyperparameters detailed in Tables 1 and 2—then performance assessed rigorously over 10-folds cross validation approach was used. The best hyper parameters for each classifier were identified from the table based on mean accuracy obtained through cross validation. RF had maximum average accuracy rate at 0.93 having no specified maximum depth but 100 estimators which makes it most suitable when dealing with large datasets like ours where multiple features may be available simultaneously or sequentially such that they require greater processing power than other models.

Table 1. Illustration Of All Classifiers Evaluation Metrics Results

Classifiers	Accuracy on Test Set	Sensitivity of Each Labels	Specificity of All Labels
SVM	0.86	[L0: 0.82, L1: 0.90]	[L0: 0.90, L1: 0.82]
DT	0.82	[L0: 0.81, L1: 0.83]	[L0: 0.83, L1: 0.81]
RF	0.93	[L0: 0.92, L1: 0.93]	[L0: 0.93, L1: 0.92]
KNN	0.84	[L0: 0.86, L1: 0.80]	[L0: 0.80, L1: 0.86]
MLP	0.85	[L0: 0.85, L1: 0.85]	[L0: 0.85, L1: 0.85]

Table 2. Illustration Of All Classifiers Hyperparameters And Evaluation Results

Classifiers	List of Hyperparameters Tuned	Best Hyperparameter	10 Folds CV Mean Accurac	F1 Scores

			y of Best Hyperparameter	
SVM	{'C': [0.1, 1, 10], 'kernel': ['linear', 'rbf'], 'gamma': [0.1, 1, 10]}	{'C': 10, 'kernel': 'rbf', 'gamma': 10}	0.87	0.86
DT	{'max_depth': [None, 10, 20]}	{'max_depth': None}	0.84	0.82
RF	{'n_estimators': [50, 100, 200], 'max_depth': [None, 10, 20]}	{'max_depth': None, 'n_estimators': 100}	0.93	0.93
KNN	{'n_neighbors': [3, 5, 7], 'weights': ['uniform', 'distance']}	{'n_neighbors': 3, 'weights': 'distance'}	0.85	0.84
MLP	{'hidden_layer_sizes': [(100,), (100, 50), (30, 30, 30)], 'alpha': [0.0001, 0.001, 0.01]}	{'alpha': 0.0001, 'hidden_layer_sizes': (100, 50)}	0.85	0.85

6. Conclusion and Future Works

Preventing fall risks relies on early detection. In this study, the main focus was on predicting fall risks using machine learning (ML) and gait analysis from video recordings. The tool achieved a 93% accuracy rate through Random Forest (RF), which means it can be considered as a dependable system. Among the predictors, gait velocity was identified to be most significant. This method is non-invasive and more convenient than wearable sensors because it is based on videos. With early detection, healthcare providers can suggest preventive actions that enhance health outcomes and independence especially among the aged. Challenges faced were high computational requirements and dataset imbalances that were solved by resampling methods. For higher accuracy and practical applications in hospitals or nursing homes; deep learning models should be deployed alongside ensemble techniques as future directions where necessary by this study so far conducted in various settings including hospitals or nursing homes.

References

[1] Song W, Latham NK, Liu L, Rice HE, Sainlaire M, Min L, et al. Improved accuracy and efficiency of primary care fall risk screening of older adults using a machine learning approach. J Am Geriatr Soc [Internet]. 2024 Apr 1 [cited 2024 Jun 5];72(4):1145–54. Available from: https://onlinelibrary.wiley.com/doi/full/10.1111/jgs.18776

[2] Steckhan GM, Warner LM, Fleig L. Falls prevention is more than just promoting physical health: evaluation of the group-based, out-patient prevention program 'Staying safe and active in old age – falls prevention.' Heal Psychol Behav Med [Internet]. 2024 Dec 31 [cited 2024 Jun 5];12(1). Available from: https://www.tandfonline.com/doi/abs/10.1080/21642850.2024.2358915

[3] Bevilacqua G, Westbury LD, Bloom I, Zhang J, Lawrence WT, Barker ME, et al. General self-efficacy, not musculoskeletal health, was associated with social isolation and loneliness in older adults during the COVID-19 pandemic: findings from the Hertfordshire Cohort Study. Aging Clin Exp Res [Internet]. 2024 Dec 1 [cited 2024 Jun 5];36(1):1–9. Available from: https://link.springer.com/article/10.1007/s40520-023-02676-5

[4] Banerjee S, Mihailovic A, Miller R, Jian-Yu E, Gitlin LN, Xiong Y, et al. Visual Impairment and Real-World Home Physical Activity with Home Environment in an Older Population. JAMA Ophthalmol [Internet]. 2024 Mar 1 [cited 2024 Jun 5];142(3):208–14. Available from: https://jamanetwork.com/journals/jamaophthalmology/fullarticle/2814751

[5] Ayvat E, Doğan M, Ayvat F, Kılınç ÖO, Sütçü G, Kılınç M, et al. Usefulness of the Berg Balance Scale for prediction of fall risk in multiple sclerosis. Neurol Sci [Internet]. 2024 Jun 1 [cited 2024 Jun 5];45(6):2801–5. Available from: https://link.springer.com/article/10.1007/s10072-024-07318-w

[6] Takase M, Kisanuki N, Nakayoshi Y, Uemura C, Sato Y, Yamamoto M. Exploring nurses' clinical judgment concerning the relative importance of fall risk factors: A mixed method approach using the Q Methodology. Int J Nurs Stud. 2024 May 1;153:104720.

[7] Jahangiri S, Abdollahi M, Patil R, Rashedi E, Azadeh-Fard N. An inpatient fall risk assessment tool: Application of machine learning models on intrinsic and extrinsic risk factors. Mach Learn with Appl. 2024 Mar 1;15:100519.

[8] Liang HW, Ameri R, Band S, Chen HS, Ho SY, Zaidan B, et al. Fall risk classification with posturographic parameters in community-dwelling older adults: a machine learning and explainable artificial intelligence approach. J Neuroeng Rehabil [Internet]. 2024 Dec 1 [cited 2024 Jun 5];21(1):1–17. Available from: https://link.springer.com/articles/10.1186/s12984-024-01310-3

[9] Takada H, Yamashita K, Osawa L, Komiyama Y, Muraoka M, Suzuki Y, et al. Assessment of lower limb muscle strength can predict fall risk in patients with chronic liver disease. Sci Rep [Internet]. 2024 Jan 2 [cited 2024 Jun 5];14(1):1–10. Available from: https://www.nature.com/articles/s41598-023-50574-7

[10] Murayama A, Higuchi D, Saida K, Tanaka S, Shinohara T. Fall Risk Prediction for Community-Dwelling Older Adults: Analysis of Assessment Scale and Evaluation Items without Actual Measurement. Int J Environ Res Public

Health [Internet]. 2024 Feb 14 [cited 2024 Jun 5];21(2):224. Available from: https://www.mdpi.com/1660-4601/21/2/224/htm
[11] Olesen AE, Væver TJ, Simonsen M, Simonsen PG, Høj K. Deprescribing in primary care without deterioration of health-related outcomes: A real-life, quality improvement project. Basic Clin Pharmacol Toxicol [Internet]. 2024 Jan 1 [cited 2024 Jun 5];134(1):72–82. Available from: https://onlinelibrary.wiley.com/doi/full/10.1111/bcpt.13925
[12] Aarts J, Saddal SRD, Bosmans JE, de Groot V, de Jong BA, Klein M, et al. Combining Exercise and Cognitive Training to Postpone Cognitive Decline in People with Multiple Sclerosis: Part of the Don't be late! Study Protocol. Mult Scler Relat Disord. 2024 Jun 1;86:105590.
[13] Yi MK, Han KH, Hwang SO. Fall Detection of the Elderly Using Denoising LSTM-based Convolutional Variant Autoencoder. IEEE Sens J. 2024;
[14] Yu J, Yang X, Deng Y, Krefman AE, Pool LR, Zhao L, et al. Incorporating longitudinal history of risk factors into atherosclerotic cardiovascular disease risk prediction using deep learning. Sci Rep [Internet]. 2024 Jan 31 [cited 2024 Jun 5];14(1):1–10. Available from: https://www.nature.com/articles/s41598-024-51685-5
[15] Habib Z, Mughal MA, Khan MA, Shabaz M. WiFOG: Integrating deep learning and hybrid feature selection for accurate freezing of gait detection. Alexandria Eng J. 2024 Jan 1;86:481–93.
[16] Huang D, Wu C, Wang Y, Zhang Z, Chen C, Li L, et al. Episode-level prediction of freezing of gait based on wearable inertial signals using a deep neural network model. Biomed Signal Process Control. 2024 Feb 1;88:105613.
[17] Sruthi P, Udgata SK. Wi-Fi sensing based person identification and activity recognition using two-phase deep learning model. Eng Appl Artif Intell. 2024 Jun 1;132:107904.
[18] Liu PL, Chang CC. Simple method integrating OpenPose and RGB-D camera for identifying 3D body landmark locations in various postures. Int J Ind Ergon. 2022 Sep 1;91:103354.
[19] Hu R, Diao Y, Wang Y, Li G, He R, Ning Y, et al. Effective evaluation of HGcnMLP method for markerless 3D pose estimation of musculoskeletal diseases patients based on smartphone monocular video. Front Bioeng Biotechnol. 2023 Jan 9;11:1335251.
[20] Mora A, Ozeki T, Mouri T, Barber R. A ROS-Integrated Skeleton Detection Framework Enabling Human-Centered Robotic Applications. 2024 IEEE Int Conf Auton Robot Syst Compet [Internet]. 2024 May 2 [cited 2024 Jun 5];3–8. Available from: https://ieeexplore.ieee.org/document/10535944/
[21] Sheng B, Chen L, Cheng J, Zhang Y, Hua Z, Tao J. A markless 3D human motion data acquisition method based on the binocular stereo vision and lightweight open pose algorithm. Meas J Int Meas Confed. 2024 Feb 15;225:113908.
[22] Murugan AS, Noh G, Jung H, Kim E, Kim K, You H, et al. Optimising computer vision-based ergonomic assessments: sensitivity to camera position and monocular 3D pose model. Ergonomics [Internet]. 2024 [cited 2024 Jun 5]; Available from: https://www.tandfonline.com/doi/abs/10.1080/00140139.2024.2304578
[23] Dutt M, Gupta A, Goodwin M, Omlin CW. An Interpretable Modular Deep Learning Framework for Video-Based Fall Detection. Appl Sci 2024, Vol 14, Page 4722 [Internet]. 2024 May 30 [cited 2024 Jun 5];14(11):4722. Available from: https://www.mdpi.com/2076-3417/14/11/4722/htm
[24] Nishiyama D, Arita S, Fukui D, Yamanaka M, Yamada H. Accurate fall risk classification in elderly using one gait cycle data and machine learning. Clin Biomech. 2024 May 1;115:106262.
[25] Jahandideh S, Hutchinson AF, Bucknall TK, Considine J, Driscoll A, Manias E, et al. Using machine learning models to predict falls in hospitalised adults. Int J Med Inform. 2024 Jul 1;187:105436.
[26] Juneau P, Lemaire ED, Bavec A, Burger H, Baddour N. Automated step detection with 6-minute walk test smartphone sensors signals for fall risk classification in lower limb amputees. PLOS Digit Heal [Internet]. 2022 Aug 18 [cited 2024 Jun 5];1(8):e0000088. Available from: /pmc/articles/PMC9931302/
[27] Sellers WI, Cross CF, Fukuhara A, Ishiguro A, Hirasaki E. Producing non-steady-state gaits (starting, stopping, and turning) in a biologically realistic quadrupedal simulation. Front Ecol Evol [Internet]. 2022 Sep 8 [cited 2024 Jun 5];10:954838. Available from: https://tohoku.elsevierpure.com/en/publications/producing-non-steady-state-gaits-starting-stopping-and-turning-in
[28] Kour N, Sunanda, Arora S. A Vision-Based Gait Dataset for Knee Osteoarthritis and Parkinson's Disease Analysis with Severity Levels. In Mendeley Data; 2022. p. 303–17.