


## Research Article

# Multimodal Classification Technique for Fall Detection of Alzheimer's Patients by Integration of a Novel Piezoelectric Crystal Accelerometer and Aluminum Gyroscope with Vision Data

V. Mohan Gowda,<sup>1</sup> Megha P. Arakeri,<sup>2</sup> and Vasireddy Raghu Ram Prasad <sup>3</sup>

<sup>1</sup>Department of Computer Science and Engineering, GITAM School of Technology, GITAM University, Bengaluru, India

<sup>2</sup>Department of Information Science & Engineering, Center of Imaging Technologies, M.S. Ramaiah Institute of Technology, Bengaluru, India

<sup>3</sup>Faculty of Electrical and Computer Engineering, Arba Minch Institute of Technology, Arba Minch University, Arba Minch, Ethiopia

Correspondence should be addressed to Vasireddy Raghu Ram Prasad; [prasad.raghu@amu.edu.et](mailto:prasad.raghu@amu.edu.et)

Received 19 August 2022; Revised 3 September 2022; Accepted 20 September 2022; Published 10 October 2022

Academic Editor: Samson Jerold Samuel Chelladurai

Copyright © 2022 V. Mohan Gowda et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Smart expert systems line up with various applications to enhance the quality of lifestyle of human beings, such as major applications for smart health monitoring systems. An intelligent assistive system is one such application to assist Alzheimer's patients in carrying out day-to-day activities and real-time monitoring by the caretakers. Fall detection is one of the tasks of an assistive system; many existing methods primarily focus on either vision or sensor data. Vision-based methods suffer from false positive results because of occlusion, and sensor-based methods yield false results because of the patient's long-term lying posture. We address this problem by proposing a multimodal fall detection system (MMFDS) with hybrid data, which includes both vision and sensor data. Random forest and long-term recurrent convolution networks (LRCN) are the primary classification algorithms for sensor data and vision data, respectively. MMFDS integrates sensor and vision data to enhance fall detection accuracy by incorporating an ensemble approach named majority voting for the hybrid data. On evaluating the proposed work on the UP fall detection dataset, accuracy was enhanced to 99.2%, with an improvement in precision, F1 score, and recall.

## 1. Introduction

According to the World Alzheimer's Report-2021, more than 55 million people live with dementia worldwide [1]. Almost 50% of people live in Asia. In India, more than 4.1 million people have dementia. The most prevalent kind of dementia that impairs memory, thinking, and behavior is Alzheimer's disease. The majority of Alzheimer's patients are 65 years old and older, and it affects one in nine adults aged 65 and older [2]. It is one of many different kinds of brain diseases that drastically lowers mental function. Hence, these patients depend more on caretakers. According to Bharucha et al. [3], one such technological tool intended to lessen the strain on Alzheimer's patients and their caregivers is the intelligent assistive technology (IAT). IAT has been

developed to help older people with Alzheimer's disease carry out their daily living activities and compensate for memory loss and executive function. For the safety of the patients, we proposed a multimodal fall detection system.

Falling is one of older people's most common health-related problems [4]. The World Health Organization ranks fall as the second most common global cause of unexpected injuries and fatalities. Hence, the fall requires immediate medical care. The fall detection system warns the caregivers when a fall occurs and reduces the consequences for the patient. In real time, the fall detection system helps to decrease the harmful effects of falls while increasing the patient's access to medical care [5]. Supposing falls are not immediately detected, patients may frequently continue to lie on the floor, leading to significant medical and

psychological issues. Participants sometimes forget the details of a fall when observing people's falls in real-world situations at less frequent intervals. This recall issue is more severe among older or Alzheimer's participants [6]. These systems for detecting falls can assist identify elderly people's falls in real time.

Mubashir et al. [7] describe fall detection as accumulating data in three ways based on wearables, environmental sensors, or visual equipment. Most fall detection wearable sensor devices are made of accelerometers, gyroscopes, and other sensors. Recently, smartphones are also applicable as wearable devices to detect falls. Floor, infrared, thermal, pressure, and other environmental sensors are used by ambient sensors, while cameras are used by vision devices. The objective of this research work is given as follows:

- (1) Random forest classification method for computing raw sensor signals
- (2) LRCN method to handle picture video data from cameras
- (3) The majority voting ensemble method takes aggregation of the results of random forest and LRCN to identify falls and human activity

Recent trends in fall detection systems make use of wearable sensors such as accelerometers and Kinect, according to a recent survey by Xu et al. [8]. The biggest problem of elderly individuals is that they find it uncomfortable to wear wearable sensor gadgets constantly. It could be challenging to alert the fall if they forgot to wear a gadget. Ambient fall detection using sensors also yields inaccurate findings with the patient's physical posture like lying for a long period. On the other hand, reducing false positives is a significant issue for all fall detection systems. However, multimodal techniques are becoming more common to increase precision and reliability. Hybrid models are giving a quick response, the amount of damage is reduced, and the quality of life can be improved significantly. According to the research, machine learning techniques are increasingly being adopted in place of more traditional algorithms for fall detection. Perry et al. [9] predicted that combining accelerometers with other sensors would be more accurate and efficient in their analysis of real-time fall detection methods.

To overcome these problems, based on the UP fall detection public dataset [10], we proposed a multimodal fall detection approach. The dataset is a collection of various sensors like wearable and ambient sensors with vision data generation devices. According to a recent survey of researchers, machine learning and deep learning methods are suitable for classifying human day-to-day activities and fall detection. For this purpose, we used the decision tree classification method for computing raw sensor signals and the LRCN method to handle picture video data from cameras. After that, we used the majority voting ensemble method to identify falls and human activity.

The paper is structured as follows. In Section 2, we look into related fall detection systems, Section 3 discusses the UP fall dataset. The proposed multimodal fall detection approach is described in Section 4. Results are discussed in

Section 5. Finally, the concluding remarks of the work are recorded.

## 2. Literature Survey

In fall detection systems, many preprocessing strategies or procedures are employed. These strategies are based on the data gathered from various sensors based and vision based systems. There are two main types of data preparation techniques used in the majority of fall detection systems. That is, statistical techniques and machine learning.

Zhang et al. [11] developed a method integrating the deep convolutional network and the collection of heuristic visual features to detect falls. The medical Internet of Things (IoT) video surveillance architecture incorporates this technique as well. It provides real-time monitoring and alarms for older people who require it. Finally, they evaluated accuracy with URFD Public dataset, and it achieved 96%. Still, it produces false alarms on account of similar behaviors or occlusion problems.

Dementia patients in Jianan hospital have incorporated an intelligent safety monitoring system for nursing. The system is implemented by Wang et al. [12] with a wearable vest for fall detection with an accelerometer. CCTV systems and RFID have been deployed in living spaces for real-time detection based on body posture, and location services assure enhanced accuracy. Integration of RFID with CCTV results in a low false alert rate.

Makma et al. [13] have compared various machine learning classification algorithm results for the three different groups of sensors. The authors used various sensors, viz., acceleration data sensors, body barometric pressure sensor, and wall barometric pressure sensor. Compared to all three machine learning algorithms, random forest gives a good performance. Usually, the barometric pressure sensor gives noisy signals because of atmospheric change. However, in the group of sensors, they are taking reference barometric pressure sensors of the wall, which improves the fall detection performance.

Sengul et al. [14] implemented deep learning-based fall detection using smartwatches. This method differentiates falls and other daily activities such as sitting, walking, squatting, and running. They begin by utilizing a mobile app to gather the accelerometer and gyroscope sensor data from the smartwatch and then transform it into the cloud. In the cloud, they adopted a bidirectional long short-term memory deep learning approach to classify falls and other activities of the person. It gives better accuracy but consumes more energy in the smartwatch.

Butt et al. [15] used deep learning techniques and evaluated their suitability for extracting characteristics from sensor data, such as accelerometer and gyroscope data that assess fall hazards. They used a senior citizen's publicly available dataset for training. Additionally, they compared two deep learning architectures such as long short-term memory and convolutional neural network (CNN)-based transfer learning. CNN-based transfer learning produced the best performance in terms of quantitative accuracy.

Martinez-Villaseñor et al. [16] developed a multimodal fall detection system for detecting falls based on various sensor data and vision devices. Long short-term memory networks (LSTM) are utilized for sensor data analysis and feature extraction, whereas CNN is employed for video analysis. Both methods are excellent for real-time identification and can extract characteristics from raw input. After training and testing, the multimodal fall detection system is getting 96.4% accuracy. However, they used deep learning techniques for both data.

Gjoreski et al. [17] developed a fall detection and activity recognition method based on wearable sensor data merging and machine learning methods. This method classifies the fall and human activity using five accelerometers and five gyroscope sensors. They trained and tested the sensor data using decision tree, random forest, and XGBoost machine learning algorithms. Finally, they performed the hyperparameter optimization of all these three algorithms and achieved the 98% of accuracy. However, older people find it challenging to wear five sensor devices every time.

The fatal fall that occurs in older people due to Alzheimer's can be monitored and reduced to a certain extent, but one major problem is the detection of false positives in the scheme, which may slow down the process and make a significant thrust on the system calls. This issue is addressed by Galvao et al. [18] who proposed different techniques of multimodal convolutional neural networks, which detect the fall on the trained RGB images and data collected from the accelerometer. For training and evaluating the network, we use the UP fall detection dataset. UP fall dataset is where rigorous comparison is done; hence, the UP fall detection dataset has achieved good accuracy.

Ha et al. [19] proposed different approaches to extract the features from the sensors and cameras, then concatenate all the features, and apply a new CNN model to extract the final feature of the data. For this fall detection purpose, they used the UP fall detection public dataset. It improves the performance of the fall detection algorithm in the UP fall detection dataset.

Table 1 gives the comparison of the different multimodal fall detection datasets. The University of Rzeszow fall detection dataset collects data using one IMU sensor using Bluetooth and two Kinect cameras via USB. Five participants performed 70 falls and ADLs. They used a threshold-based fall detection technique. Berkeley Multimodal Human Action Database simultaneously collects the data of 11 actions of 12 participants using six accelerometer sensors, one motion capture system, four multiview cameras, and one Kinect system. UP fall detection dataset presents 11 activities over 17 participants. The data recording of each activity is performed on three trails using five IMU with an accelerometer and gyroscope, one EEG headset, six infrared sensors, and two cameras. Compared to all UP fall detection datasets, it has extensive data to perform the multimodal operations.

In this paper, to enhance the performance of the fall detection system on the UP fall detection dataset, we propose the MMFDS, which uses random forest for sensor data

and LRCN for the vision data and then combines both results to evaluate with a majority voting strategy.

### 3. Dataset

We employ the UP fall detection dataset to assess the proposed method [10]. It includes time series representations of various sensor data and synchronized RGB pictures. In the UP fall detection dataset, samples have been classified as different falls and activities in daily living. It consists of 11 activities, each activity as three attempts for multimodal fall detection. This multimodal dataset collects data from 17 participants using wearable sensors, ambient sensors, and two cameras.

Activities performed are majorly related to the following human falls: falling forward using hands and knees, falling backward and sideward, falling while sitting in an empty chair, and six human daily living activities such as picking objects, standing, walking, jumping, sitting, and laying. The final dataset activity data distribution is shown in Figure 1.

The data were collected using the wearable sensor and ambient sensor and cameras. The wearable sensors collect information from the ambient light level, 3-axis accelerometer, and 3-axis gyroscope. The wearable sensors are located in five different places in the human body: the left wrist, the center of the waist, under the neck, the left ankle, and the right pocket of the pants. Also, one electroencephalograph (EEG) sensor is located in the forehead and is used to record the brain wave signals. The six infrared sensors are mounted above the room's floor and track changes in optical device interruption. Last but not least, photographs taken with two cameras as the subjects participated in the activities improved the dataset. The sensors used to collect the dataset have a sampling rate of 18 Hz. Figure 2 shows the position of the wearable sensors in the human body and the environmental position of the ambient sensors along with cameras.

The UP fall detection dataset used five Mbientlab Meta sensor wearable sensors and EEG to collect the data. The Mbientlab Meta sensor combines an accelerometer, gyroscope, temperature, light, pressure, and motion sensors. UP fall detection dataset collects the 3-axis accelerometer and 3-axis gyroscope sensor data. Figure 3 shows the Mbientlab Meta sensor.

The 3-axis accelerometer sensor measures the proper acceleration of people or objects. A piezoelectric crystal material is used in accelerometers. It generates a tiny electrical current in response to the motion. An accelerometer's acceleration is relative to free fall and the acceleration of people and objects. The accelerometer is used to measure the acceleration of various fields such as engineering, biological, industry, and health monitoring. We use an accelerometer to measure the motion of the people such as walking, running, sitting, and sleeping.

The 3-axis gyroscope sensor is a spinning disc. It is used to measure the angular velocity of the materials. It is also used as a motion sensor that tracks and measures an object's

TABLE 1: Different multimodal fall detection datasets.

Dataset	Sensor type	Camera type
University of Rzeszow fall detection dataset [20]	One IMU with an accelerometer	Two Kinect cameras
Berkeley multimodal human action database [21]	Six accelerometers	One motion capture system, four multiview cameras, and one Kinect system
UP fall detection dataset [10]	Five IMU with accelerometer and gyroscope, one EEG headset, and six infrared sensors	Front and back two cameras

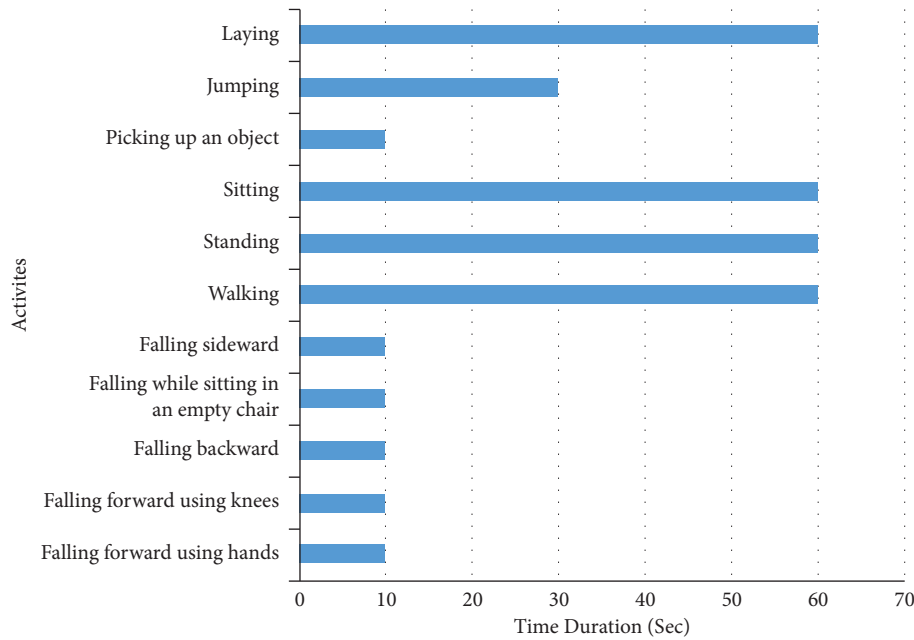


FIGURE 1: Activity data distribution.

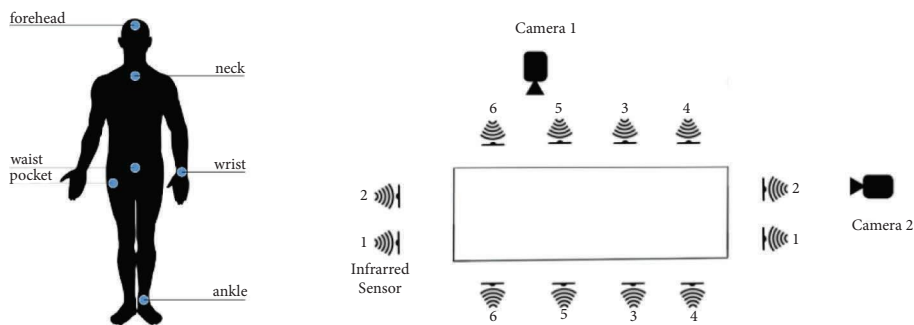


FIGURE 2: Wearable sensor devices, ambient sensor, and camera device distribution.

angular motion. It is a 1-axis, 2-axis, and 3-axis to measure the rate of rotation of the object.

Six infrared sensors are installed above the room's ground to collect the data. The infrared sensor is shown in Figure 4. The infrared sensor is a radiation-sensitivity optoelectric component. Infrared sensors are mainly used to measure the interruption of motion. It measures the changes in interruption of the optical device where 0 is interrupted, and 1 means not interrupted.

#### 4. Multimodal Fall Detection

We proposed a multimodal fall detection using the random forest algorithm and LRCN method shown in Figure 5. Multimodal fall detection contains 3 phases. Firstly, it detects the fall using a machine learning algorithm of the sensor data. Secondly, it detects the fall using the CNN + LSTM technique of the video data. Both data were trained independently using ML and LRCN modules, and in

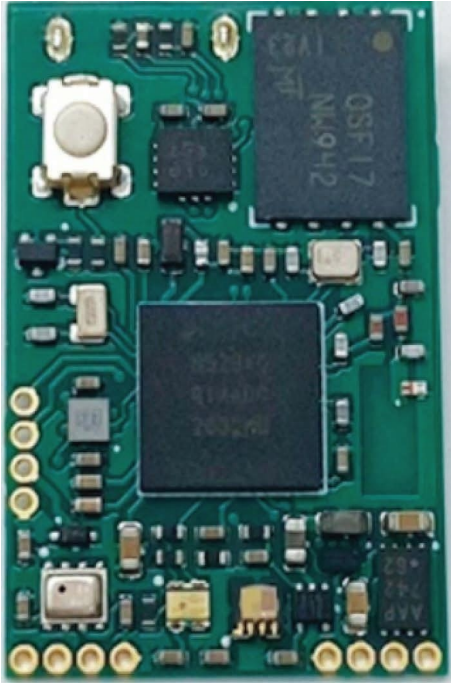


FIGURE 3: Mbientlab Meta sensor.

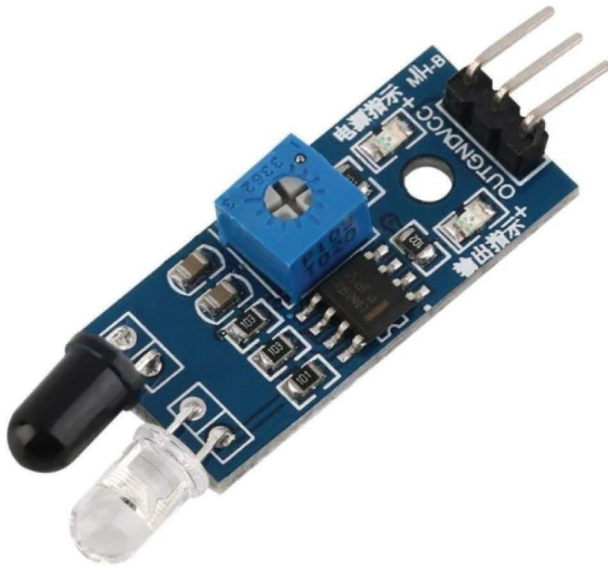


FIGURE 4: Infrared sensor.

the end, we combined both modules with the majority voting strategy. Figure 5 shows the proposed multi model fall detection described as follows.

**4.1. Data Preprocessing.** For data preprocessing related to sensor data, we removed the missing data rows and dropped the duplicate records in each activity. Then, we combine all the activity files into one file. 28 attributes and one label data were used in the present research work. Finally, we normalized the sensor data with the min-max scalar into a range of [0, 1].

All of the information was preprocessed for consolidation purposes. We chose to homogenize the sampling rate in the combined dataset because the devices operated at various sampling rates. In that regard, using its time stamps as a reference, we selected videos of the camera that acquired the fewest frames (18 fps approximately). By deleting duplicate images from both cameras, we could guarantee that all images recovered from Cameras 1 and 2 would be the same size and arranged in the same order in the camera data. We also carefully investigated the timestamp of the sensor data as well as the image timestamp in order to determine the best mapping. Moreover, the images are resized into  $40 \times 40$  pixels and converted into grayscale. We scaled each image by dividing each pixel's value by 255 to make sure that every pixel is in the range [0, 1].

**4.2. Feature Extraction and Modeling.** For sensor data, we compared five machine learning (ML) classification algorithms: logistic regression, decision tree, K-nearest neighbor, support vector machine, and random forest. We examined all five machine learning algorithms, and random forest produced the model's best performance. Therefore, the proposed model classifies the sensor data using the random forest method.

Logistic regression [22] is used when data flow in a linear function, and it uses the sigmoid function.

$$g(z) = \frac{1}{1 + e^{-z}}. \quad (1)$$

This module assumes data flow in linear functions. However, our module sensor data overlapped with each other. Hence, we are unable to get good performance in logistic regression.

A decision tree [23] has trained the module-like tree structure, where the tree structure contains mainly three things such as internal node, branches, and leaf node. The internal node tests the attributes, branches give the output of the test, and finally, the leaf node holds the attribute label. The tree splits the dataset into smaller subsets based on the attribute value test, and at the same time, decision tree starts the incremental learning. The Gini index function determines how well a decision tree was split. Finally, it gives results of multiple branches. Each branch value represents the feature test and leaf nodes represent the activity label. In our case, features are numeric values. Hence, it takes more time and generates a very large tree. The attributes are selected based on the Gini index.

$$\text{Gini}(x) = 1 - \sum_{m=1}^n P\left(\frac{m}{t}\right)^2, \quad (2)$$

where  $n$  represents several classes and  $P$  represents a ratio of the class at the  $m_{\text{th}}$  node.

K-nearest neighbor [24] is classified based on the attributes and training samples using Euclidean distance. We calculated the  $K=3$  nearest learning data and classified the class label to which data belong to the majority vote. KNN is

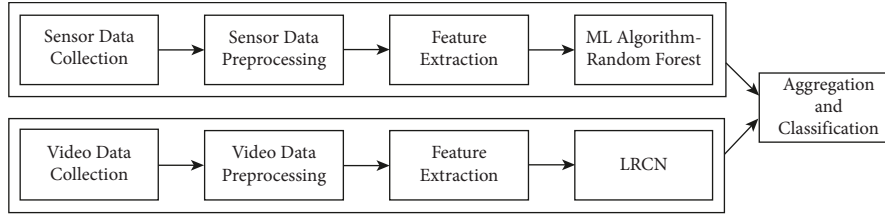


FIGURE 5: Proposed multimodel fall detection architecture.

more efficient in training large datasets. However, it always needs to identify the value of  $K$ , and also computation cost is high because it calculates the distance between the data of each training sample.

Support vector machine (SVM) [25] classification technique can handle both linear and nonlinearly separable data. The SVM module produces a hyperplane that increases the classification margin. The hyperplane will be an  $N-1$  level subspace if  $N$  features are present. Super vectors are the nodes that create a border in the feature space. The maximum margin is calculated based on the relative position and the ideal hyperplane is again drawn in the center. SVM takes the maximum training time of the larger dataset. SVM gives less accuracy to this dataset.

Figure 6 shows a group of decision tree classification algorithms named random forest [26]. Each split of the decision trees is trained using a bootstrapped sample of the initial training data, and the random forest method only searches a random subset of the input variables throughout training. Every tree in the random forest estimates the activity for classification, and the majority of decision trees vote to decide the classifier's final output. This method will determine the final action based on what most decision trees predict. Table 2 gives the parameter setting for machine learning models.

In video data, we proposed the fall detection system based on the deep learning models. The deep learning methods give an outstanding image performance and achieve better performance of state-of-the-art approaches in different applications such as object recognition, object detection, and segmentation. Donahue et al. [27] developed the LRCN method for end-to-end training and it compares state-of-the-art approaches to feature extraction and processing. LRCN architecture takes advantage of the significant process of CNN in video recognition and growing interest in using these models with time-varying inputs and outputs. The CNN outputs are fed to the recurrent sequential model of LSTM. Finally, it produces the variable length predictions. We studied two deep learning methods CNN and LSTM.

Convolutional neural network (CNN): a deep learning algorithm CNN takes the input as an image/video and learns the kernels used to extract the feature map of the input data using convolutional operations [28]. The spatial and temporal relationships in the images are captured by the CNN convolutional filters. CNN has many applications for several real-time problems.

For the same receptive field, each layer filter can extract certain features, resulting in different outputs known as feature maps. The CNN pooling layer decreases the spatial

size of the feature map and passes to another layer of the network layer. Pooling also reduces the computational power of the processing data. Pooling is mainly of two categories max pooling and average pooling. The greatest value from the region of the picture that the kernel has covered is provided by max pooling. The average value from the picture region that the kernel has covered is provided by average pooling. Max pooling conducts dimensionality reduction and denoising in addition to removing noisy activity. Average pooling is performed only for dimensionality reduction. Hence, we used max pooling in our approach.

Long short-term memory (LSTM): a recurrent neural network (RNN) can be used to process temporal data. Because of the vanishing gradient issue, RNNs have short-term memory problems. Hence, RNN is challenging to work with longer data sequences. Standard recurrent neural networks are often not highly effective at processing data in the real world because they cannot handle long-time delays.

The advanced version of RNN is that the LSTM [29] network is appropriate for processing longer data sequences for particular classification and prediction of time series data. In order to efficiently process temporal information and solve the issue of large time lags, the LSTM Network presents a number of gating mechanisms that handle the time delays in a time series. It is also relatively efficient in the real-world processing of data.

The first process information of the forget gate is

$$f_t = \sigma(W_f \cdot [h_t - 1, x_t] + b_f), \quad (3)$$

where  $W_f$  represents the weight of the layers,  $h_t$  and  $x_t$  are the input stimulants,  $t$  is the time of the network context, and  $b_f$  is the bias layer. Once training got over the gate decide that the input stimulants should be retained or omitted, to deal with the time delays.

Network information gate:

$$i_t = \sigma(W_i \cdot [h_t - 1, x_t] + b_i), \quad (4)$$

where  $W_i$  is the weights of layers. Then, we update the context layer:

$$C'_t = \tanh(W_c \cdot [h_t - 1, x_t] + b_c), \quad (5)$$

$$C_t = f_t * C_{t-1} + i_t * C'_t \quad (6)$$

where  $W_c$  represents the weights of the layer. Finally, based on the current cell state, the processed output ( $h_t$ ) is obtained:

$$o_t = \sigma(W_o \cdot [h_t - 1, x_t] + b_o). \quad (7)$$

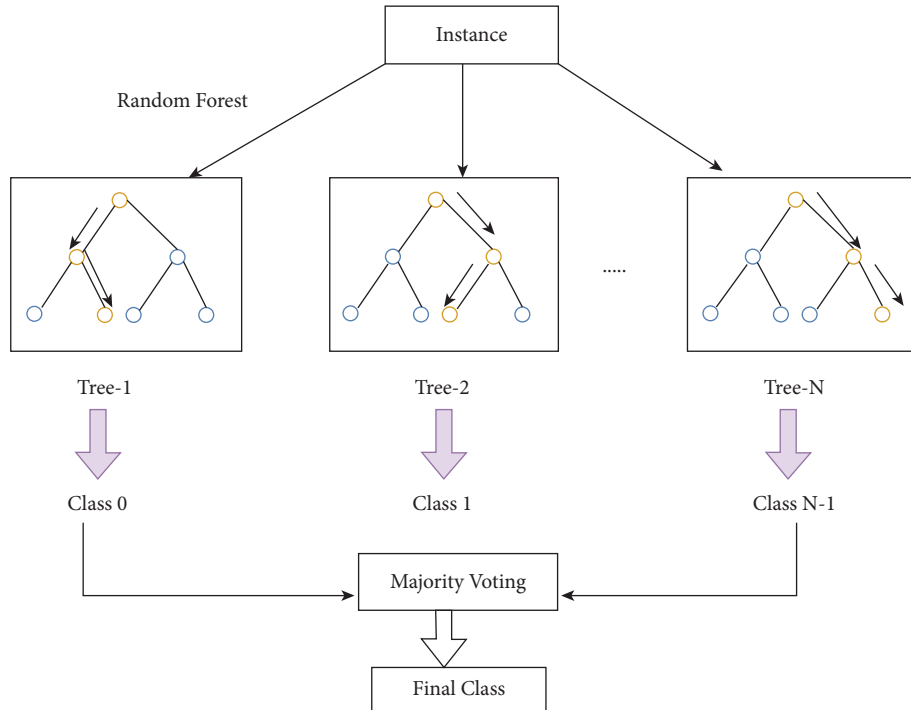


FIGURE 6: Ensemble of decision tree classifiers in random forest.

TABLE 2: Parameter setting for ML models.

ML model	Parameters
Logistic regression	Penalty = l2
	Intercept_scal = 1
	Max_iteration = 100
Decision tree	Criterion = Gini
	Min_samples_split = 2
	Min_samples_leaf = 1
K-nearest neighbors	Neigh = 3
	Leaf_size = 30
	Metric = Minkowski
SVM	C = 1.0 kernel = rbf
	Deg = 3
	Gamma = auto
	Tolerance = 0.001
Random forest	Estimators = 50
	Min_samples_split = 2
	Min_samples_leaf = 1
	Bootstrap = true

Then, input for the tanh function is the cell state, range values between  $-1$  and  $1$ , and the obtained result is multiplied by the sigmoid gate:

$$h_t = o_t * \tanh(C_t). \quad (8)$$

Using input data, we extracted a high-dimension abstraction having low dimension representation by reading the LSTM's output.

Long-term recurrent convolutional networks (LRCN): we adopted the LRCN approach for the proposed model for our video data. The proposed LRCN combines convolutional

layers and LSTM Layers into one model. It receives input as a frame from the video and extracts the spatial feature maps using a convolutional layer. Then, the extracted feature map feeds to LSTM layers at each time stamp. Finally, it estimates falls or other activities performed by the present subject.

The LRCN model's architecture is shown in Figure 7. The LRCN considers the following layer: a time-distributed Conv2D layer is embedded with the rectified linear unit (ReLU) to eliminate the vanishing gradient problem. Conv2D also has 16 filters of size  $3 \times 3$  which is followed by MaxPooling2D of size  $4 \times 4$  layers and dropout layer; then, time-distributed Conv2D layer with 32 filters of size  $3 \times 3$  is with a ReLU which will be followed by MaxPooling2D of size  $4 \times 4$  layers and dropout layer; then, time-distributed Conv2D layer with 64 filters of size  $3 \times 3$  with ReLU will be followed by MaxPooling2D of size  $2 \times 2$  layers and dropout layer; then, time-distributed Conv2D layer is with 64 filters of size  $3 \times 3$  with a ReLU and a MaxPooling2D of size  $2 \times 2$  layers. LSTM (32) accepts only the flatten layer of features extracted from Conv2D. The dense layer will then predict the activities of the frame using the output from the LSTM layer with softmax activation. With a maximum of 70 epochs, we trained the LRCN with an initial learning rate of 0.001.

**4.3. Aggregation and Classification.** We compute an estimation of the fall or other activity conducted using the random forest and LRCN method. We get estimates from both methods over a given temporal window size and combine these results to produce the final fall/activity detection. The majority vote significantly strengthens the classifier's robustness. Hence, we used the majority voting ensemble approach to produce the most frequent class

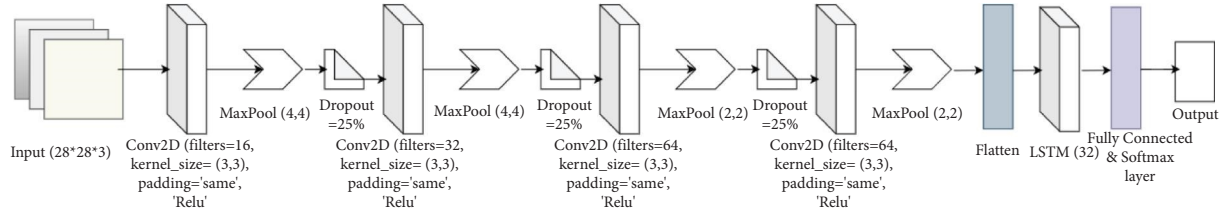


FIGURE 7: Architecture of the LRCN model.

within the window [30]. Each model generates a prediction for each instance, and the output prediction is the one that obtains more than half of the votes. The activity label with the majority of votes is projected after the predictions for each activity label have been added together.

## 5. Results

The proposed model has three steps. Firstly, we train the sensor data; in the next step, we train video data; in step 3, we carry out the majority voting strategy in the estimation of both stage 1 and stage 2.

For the sensor data training, we compared different Machine Learning (ML)-based classification algorithms based on accuracy, F1 score, precision, and recall. For ML algorithms, we have taken 70% of the data samples and 30% of the data samples for training and testing, respectively. Samples are collected from 1-second windows of unedited sensor signals without any overlapping and labeled with the most common fall or other activity that occurred during that

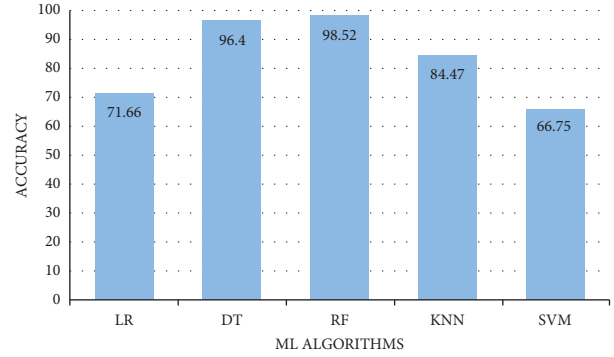


FIGURE 8: Comparison of accuracy.

window. After evaluating, we discovered that random forest is giving better performance. Hence, we use a random forest algorithm in the proposed model for training sensor data. The evaluating metrics are calculated as follows.

$$\text{Accuracy} = \frac{\text{True\_Positive} + \text{True\_Negative}}{\text{True\_Positive} + \text{True\_Negative} + \text{False\_Positive} + \text{False\_Negative}}, \quad (9)$$

$$\text{Precision} = \frac{\text{True\_Positive}}{\text{True\_Positive} + \text{True\_Negative}}, \quad (10)$$

$$\text{Recall} = \frac{\text{True\_Positive}}{\text{True\_Positive} + \text{False\_Negative}}, \quad (11)$$

$$\text{F1 score} = \frac{\text{True\_Positive}}{\text{True\_Positive} + 1/2(\text{False\_Positive} + \text{False\_Negative})}, \quad (12)$$

The evaluation of different machine learning classification algorithms is shown in Figure 8 which shows an accuracy comparison of the logistic regression (LR), decision tree (DT), random forest (RF), K-nearest neighbor (KNN), and support vector machine (SVM) machine learning classification methods. Compared to all ML methods, random forest gives better accuracy. Figure 9 shows a precision comparison of the LR, DT, RF, KNN, and SVM machine learning classification methods. Compared to all ML methods, random forest gives high precision. Figure 10 shows a Recall comparison of the LR, DT, RF, KNN, and

SVM machine learning classification methods. Compared to all ML methods, random forest gives better recall. Figure 11 shows an F1 score comparison of the LR, DT, RF, KNN, and SVM Machine learning Classification methods. Compared to all ML methods, random forest gives a better F1 score. Hence, we used random forest for sensor data training for the proposed method.

For video data training, we use a single model called LRCN (CNN + LSTM). Out of 100%, 30% is testing samples and 70% is training samples in LRCN. Time-distributed Conv2D layers will be used, followed by MaxPooling2D



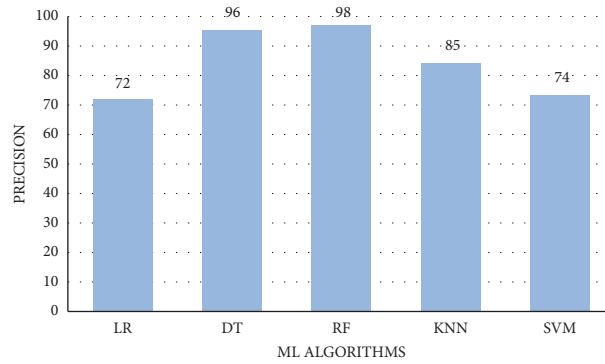


FIGURE 9: Comparison of precision.

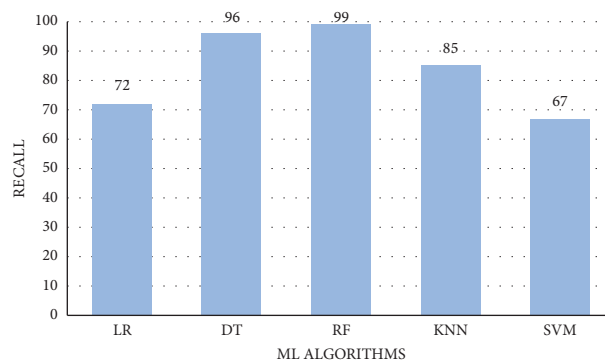


FIGURE 10: Comparison of recall.

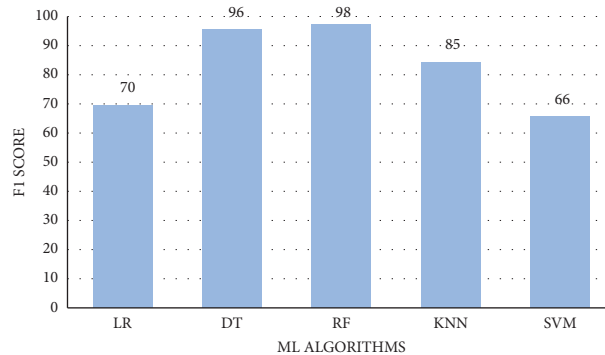


FIGURE 11: Comparison of F1 score.

and Dropout layers. The feature that was taken from the Conv2D layers and flattened using the Flatten layer will then be added to an LSTM layer. The output from the LSTM layer with softmax activation will then be used by the dense layer to forecast the action done over the training data. After that, we validate our LRCN performing 99.8% accuracy and 0.15% of loss of the testing data, as shown in Figure 12.

Finally, majority voting is performed in the estimation of random forest and LRCN model with the window of 1-

second size over the testing data. The overall result of the proposed model (random forest + LRCN + majority voting) achieved 99.2% accuracy. Compared to the single random forest or LRCN approaches, it enhances the classification performance of each activity. Table 3 gives the comparison of the UP fall detection dataset results with the other state-of-the-art methods. The proposed method performs better in accuracy, F1 score, and precision and recall metrics. Hence, we can say that our multimodal method improves fall detection performance.

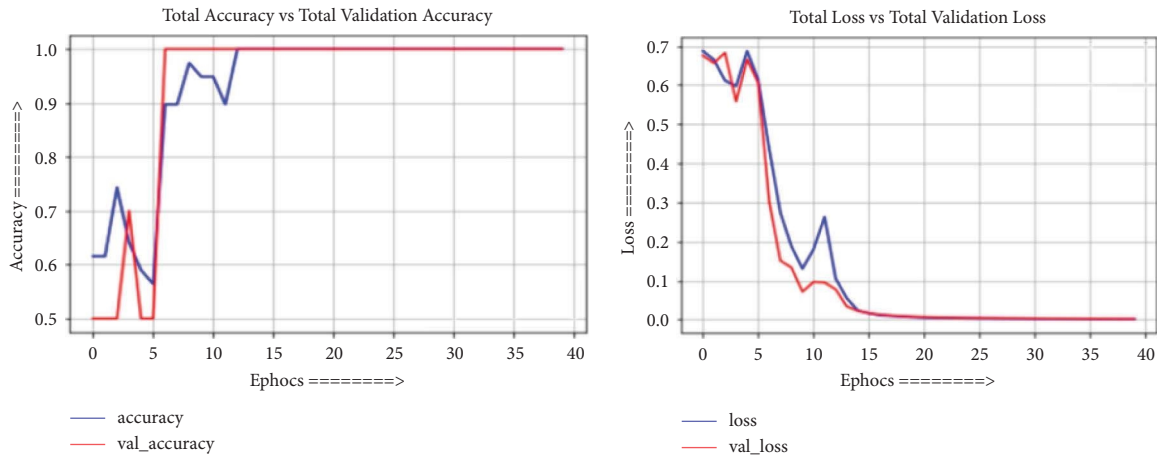


FIGURE 12: LRCN accuracy and loss value.

TABLE 3: Evaluated results of up fall detection dataset.

	Accuracy (%)	Precision (%)	Recall (%)	F1 score (%)
Martinez-Villaseñor et al. [10]	96.4	84.23	81.48	82.31
Gjoreski et al. [17]	98.03	85.77	79.42	82.47
Chahyati and Hawari [31]	98.31	95.64	95.29	95.44
Proposed model	99.2	98.3	99.1	98.4

## 6. Conclusion

This paper implements MMFDS for an intelligent assistive system for Alzheimer’s patients with hybrid data to overcome the issues, particularly with vision-based and sensor-based methods. MMFDS is a three-step algorithm, step 1 implements random forest as the classification Algorithm for sensor data, and in step 2, long-term recurrent convolution network [LRCN] algorithm applies for the classification of vision data. Finally, step 3 incorporates an ensemble approach named majority voting for the result obtained from sensor and vision data classification algorithms to improve fall detection accuracy. The performance of the MMFDS on the UP fall detection dataset records a significant change in the precision, recall, and F1 score with an accuracy of 99.2%. MMFDS extends to the more extensive and real-time data set with hybrid data.

After being trained with data from a sensor device and a primary RGB camera positioned in a room corner, the model can be used for real-world applications. Future work on using the proposed model with real-world scenarios is to assess its generalizability to Alzheimer’s patients.

## Data Availability

The data used to support the findings of this study are included in the article.

## Conflicts of Interest

The authors declare that there are no conflicts of interest regarding the publication of this paper.

## References

- [1] S Alzheimer, “Alzheimer’s disease facts and figures,” *Alzheimer’s Association Report*, 2021, <https://www.alzint.org/u/World-Alzheimer-Report-2021>.
- [2] V. Mohan Gowda and M. P. Arakeri, “Recent advances and future directions of assistive technologies for alzheimer’s patients,” *Emerging Research in Computing, Information, Communication and Applications*, pp. 25–41, 2022.
- [3] A. J. Bharucha, V. Anand, J. Forlizzi et al., “Intelligent assistive technology applications to dementia care: current capabilities, limitations, and future challenges,” *American Journal of Geriatric Psychiatry*, vol. 17, no. 2, pp. 88–104, 2009.
- [4] H. Gjoreski, M. Gams, and M. Luštrek, “Context-based fall detection and activity recognition using inertial and location sensors,” *Journal of Ambient Intelligence and Smart Environments*, vol. 6, no. 4, pp. 419–433, 2014.
- [5] R. Igual, C. Medrano, and I. Plaza, “Challenges, issues and trends in fall detection systems,” *BioMedical Engineering Online*, vol. 12, no. 1, pp. 66–24, 2013.
- [6] J. Klenk, L. Schwickert, L. Palmerini et al., “The FARSEEING real-world fall repository: a large-scale collaborative database to collect and share sensor signals from real-world falls,” *European review of aging and physical activity*, vol. 13, no. 1, pp. 8–7, 2016.
- [7] M. Mubashir, L. Shao, and L. Seed, “A survey on fall detection: principles and approaches,” *Neurocomputing*, vol. 100, pp. 144–152, 2013.
- [8] T. Xu, Y. Zhou, and J. Zhu, “New advances and challenges of fall detection systems: a survey,” *Applied Sciences*, vol. 8, no. 3, p. 418, 2018.
- [9] J. T. Perry, S. Kellog, S. M. Vaidya, J. H. Youn, H. Ali, and H. Sharif, “Survey and evaluation of real-time fall detection approaches,” in *Proceedings of the 2009 6th International*

- Symposium on High Capacity Optical Networks and Enabling Technologies (HONET)*, pp. 158–164, IEEE, Alexandria, Egypt, 2009, December.
- [10] L. Martínez-Villaseñor, H. Ponce, J. Brieva, E. Moya-Albor, J. Núñez-Martínez, and C. Peñafort-Asturiano, “UP-fall detection dataset: a multimodal approach,” *Sensors*, vol. 19, no. 9, p. 1988, 2019.
- [11] Y. Zhang, X. Zheng, W. Liang, S. Zhang, and X. Yuan, “Visual surveillance for human fall detection in healthcare IoT,” *IEEE MultiMedia*, vol. 29, no. 1, pp. 36–46, 2022 Mar 3.
- [12] P. Wang, C. S. Chen, and C. C. Chuan, “Location-aware fall detection system for dementia care on nursing service in evergreen inn of Jianan Hospital,” in *Proceedings of the IEEE 16th International Conference on Bioinformatics and Bioengineering (BIBE)*, pp. 309–315, IEEE, Taichung, Taiwan, 2016 Oct 31.
- [13] J. Makma, D. Thanapatay, T. Isshiki, J. Chinrungrueng, and S. Thiemjarus, “Toward accurate fall detection with a combined use of wearable and ambient sensors,” in *Proceedings of the 2022 Joint International Conference on Digital Arts, Media and Technology with ECTI Northern Section Conference on Electrical, Electronics, Computer and Telecommunications Engineering (ECTI DAMT & NCON)*, pp. 298–301, IEEE, Chiang Rai, Thailand, 2022 Jan 26.
- [14] G. Şengül, M. Karakaya, S. Misra, O. O. Abayomi-Alli, and R. Damaševičius, “Deep learning based fall detection using smartwatches for healthcare applications,” *Biomedical Signal Processing and Control*, vol. 71, Article ID 103242, 2022.
- [15] A. Butt, S. Narejo, M. R. Anjum, M. U. Yonus, M. Memon, and A. A. Samejo, “Fall detection using LSTM and transfer learning,” *Wireless Personal Communications*, vol. 126, no. 2, pp. 1733–1750, 2022.
- [16] L. Martínez-Villaseñor, H. Ponce, and K. Perez-Daniel, “Deep learning for multimodal fall detection,” in *Proceedings of the IEEE International Conference on Systems, Man and Cybernetics (SMC)*, pp. 3422–3429, IEEE, Bari, Italy, 2019 Oct 6.
- [17] H. Gjoreski, S. Stankoski, I. Kiprijanovska et al., “Wearable sensors data-fusion and machine-learning method for fall detection and activity recognition,” in *Proceedings of the Challenges and Trends in Multimodal Fall Detection for Healthcare 2020*, pp. 81–96, Springer, Cham, 2020.
- [18] Y. M. Galvão, J. Ferreira, V. A. Albuquerque, P. Barros, and B. J. Fernandes, “A multimodal approach using deep learning for fall detection,” *Expert Systems with Applications*, vol. 168, Article ID 114226, 2021.
- [19] T. V. Ha, H. Nguyen, S. T. Huynh, T. T. Nguyen, and B. T. Nguyen, “Fall detection using multimodal data,” in *Proceedings of the International Conference on Multimedia Modeling*, pp. 392–403, Springer, Cham, 2022 Jun 6.
- [20] B. Kwolek and M. Kepski, “Human fall detection on embedded platform using depth maps and wireless accelerometer,” *Computer Methods and Programs in Biomedicine*, vol. 117, no. 3, pp. 489–501, 2014.
- [21] Teleimmersion Lab, “Berkeley multimodal human action database (MHAD),” *University of California*, [http://teleimmersion.citris-uc.org/berkeley\\_mhad](http://teleimmersion.citris-uc.org/berkeley_mhad) Available online, 2013.
- [22] D. Maulud and A. M. Abdulazeez, “A review on linear regression comprehensive in machine learning,” *Journal of Applied Science and Technology Trends*, vol. 1, no. 4, pp. 140–147, 2020 Dec 31.
- [23] Y. Wang and I. H. Witten, “Induction of model trees for predicting continuous classes,” *Eur. Conf. Mach. Learn.* vol. 9, pp. 128–137, 1996.
- [24] G. Guo, H. Wang, D. Bell, Y. Bi, and K. Greer, “KNN model-based approach in classification,” in *Proceedings of the InOTM Federated International Conferences On the Move to Meaningful Internet Systems*, pp. 986–996, Springer, Berlin, Heidelberg, 2003 Nov 3.
- [25] C. Cortes and V. Vapnik, “Support-vector networks,” *Machine Learning*, vol. 20, no. 3, pp. 273–297, 1995 Sep.
- [26] P. O. Gislason, J. A. Benediktsson, and J. R. Sveinsson, “Random forests for land cover classification,” *Pattern Recognition Letters*, vol. 27, no. 4, pp. 294–300, 2006 Mar 1.
- [27] J. Donahue, L. Anne Hendricks, S. Guadarrama et al., “Long-term recurrent convolutional networks for visual recognition and description,” in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 2625–2634, IEEE, San Juan, PR, USA, 2015.
- [28] J. Gu, Z. Wang, J. Kuen et al., “Recent advances in convolutional neural networks,” *Pattern Recognition*, vol. 77, pp. 354–377, 2018 May 1.
- [29] J. Liu, G. Wang, P. Hu, L. Y. Duan, and A. C. Kot, “Global context-aware attention lstm networks for 3d action recognition,” in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 1647–1656, IEEE, Honolulu, HI, USA, July 2017.
- [30] A. Bayat, M. Pomplun, and D. A. Tran, “A study on human activity recognition using accelerometer data from smartphones,” *Procedia Computer Science*, vol. 34, pp. 450–457, 2014 Jan 1.
- [31] D. Chahyati and R. Hawari, “Fall detection on multimodal dataset using convolutional neural network and long short term memory,” in *Proceedings of the 2020 International Conference on Advanced Computer Science and Information Systems (ICACSIS)*, pp. 371–376, IEEE, Depok, Indonesia, 2020 Oct 17.