AIMS Mathematics

https://www.aimspress.com/journal/Math

AIMS Mathematics, 9(6): 15486–15504.
DOI: 10.3934/math.2024749
Received: 30 January 2024
Revised: 25 March 2024
Accepted: 07 April 2024
Published: 29 April 2024

Research article

Artificial Rabbit Optimizer with deep learning for fall detection of disabled people in the IoT Environment

Eatedal Alabdulkreem[1,*], Mesfer Alduhayyem[2], Mohammed Abdullah Al-Hagery[3], Abdelwahed Motwakel[4], Manar Ahmed Hamza[4] and Radwa Marzouk[5]

[1] Department of Computer Sciences, College of Computer and Information Sciences, Princess Nourah Bint Abdulrahman University, P.O. Box 84428, Riyadh 11671, Saudi Arabia
[2] Department of Computer Science, College of Computer Engineering and Sciences, Prince Sattam bin Abdulaziz University, Al-Kharj 16273, Saudi Arabia
[3] Department of Computer Science, College of Computer, Qassim University, Saudi Arabia
[4] Department of Management Information Systems, College of Business Administration in Hawtat Bani Tamim, Prince Sattam bin Abdulaziz University, Saudi Arabia
[5] Department of Mathematics, Faculty of Science, Cairo University, Giza 12613, Egypt

* **Correspondence:** Email: Eaalabdulkareem@pnu.edu.sa.

Abstract: Fall detection (FD) for disabled persons in the Internet of Things (IoT) platform contains a combination of sensor technologies and data analytics for automatically identifying and responding to samples of falls. In this regard, IoT devices like wearable sensors or ambient sensors from the personal space role a vital play in always monitoring the user's movements. FD employs deep learning (DL) in an IoT platform using sensors, namely accelerometers or depth cameras, to capture data connected to human movements. DL approaches are frequently recurrent neural networks (RNNs) or convolutional neural networks (CNNs) that have been trained on various databases for recognizing patterns connected with falls. The trained methods are then executed on edge devices or cloud environments for real-time investigation of incoming sensor data. This method differentiates normal activities and potential falls, triggering alerts and reports to caregivers or emergency numbers once a fall is identified. We designed an Artificial Rabbit Optimizer with a DL-based FD and classification (ARODL-FDC) system from the IoT environment. The ARODL-FDC approach proposes to detect and categorize fall events to assist elderly people and disabled people. The ARODL-FDC technique comprises a four-stage process. Initially, the preprocessing of input data is performed by Gaussian filtering (GF). The ARODL-FDC technique applies the residual network (ResNet) model for feature extraction purposes.

Besides, the ARO algorithm has been utilized for better hyperparameter choice of the ResNet algorithm. At the final stage, the full Elman Neural Network (FENN) model has been utilized for the classification and recognition of fall events. The experimental results of the ARODL-FDC technique can be tested on the fall dataset. The simulation results inferred that the ARODL-FDC technique reaches promising performance over compared models concerning various measures.

Keywords: fall detection; Internet of Things; deep learning; parameter tuning; fully Elman Neural Network
Mathematics Subject Classification: 11Y40

1. Introduction

The IoTs have a comparatively modern technology that can be an excellent capability for emerging an FD system [1]. IoT also gives robust storage and processing capacity and facilities to other layers of edge and cloud computing (CC). Edge and fog computing must be employed for identifying falls [2]. Edge devices are processed as well as positioned near other users and devices. Alternatively, fog nodes are positioned near local networks and other system architecture [3]. FD devices utilize alert systems technology for recognizing and providing emergency assistance. Once the user falls, these models can rapidly actuate the sensor. The embedded technology is positioned near the neck, on the waist, or wrist depending upon the devices [4]. For superior service costs, most medical alert corporations combine the FD ability within their emergency alert system. A few companies sell FD devices, which must be worn individually from one's health alert button. Another device's price is included in the monthly subscription strategy [5]. FD devices, for instance, can support identifying anomalies and transfer real-time signals to medical and social service experts on irregularities. Acceding to wearable devices, FD techniques are extremely prevalent recently due to numerous benefits like non-intrusive, lightweight lower-cost, and energy-saving [6].

Over the years, the development of FD and prevention methods must be a widespread research domain. Several approaches have been implemented for the expansion of these techniques. Three major types of technologies in FD namely IoT, artificial intelligence (AI), and CC-based methods [7]. Detection is the function of identifying the existing specific event or object in a particular context while recognition is defined as resolving the membership of the event or occurrence in a certain class [8]. Recently, DL has been employed extensively in the majority of domains globally. In FD, DL approaches have been more efficiently utilized in past years than other techniques like threshold-based methods. Machine learning (ML) methods can be also highly popular in this domain [9]. DL and ML exist as a subcategory of AI. ML performs under structured data to classify the difficulties. In ML-based approaches, the feature requires identification, which has been programmed by the supervisor [10]. Alternatively, DL-based models commonly execute the essential pattern recognition tasks without implementing some explicit feature extraction techniques.

We design an Artificial Rabbit Optimizer with a DL-based FD and classification (ARODL-FDC) approach in the IoT environment. The ARODL-FDC technique proposes to detect and categorize fall events to assist elderly people and disabled people. The ARODL-FDC technique comprises a four-stage process. Initially, the preprocessing of input data is performed by Gaussian filtering (GF). The ARODL-FDC technique applies the residual network (ResNet) model for feature extraction purposes.

Besides, the ARO algorithm has been utilized for better hyperparameter choice of the ResNet algorithm. At the final stage, a fully Elman Neural Network (FENN) model can be utilized for the classification and recognition of fall events. The experimental results of the ARODL-FDC technique can be tested on the fall dataset.

2. Related works

Alabdulkreem et al. [11] projected a Chameleon Swarm Algorithm with Improved Fuzzy DL for FD (CSA-IDFLFD) method that includes 2 stages of processes. In the primary stage, the method contains IDFL architecture for recognition and classification. Then, the parameters can be optimally chosen by the CSA method. Vaiyapuri et al. [12] designed an architecture using the optimum DCNN (IMEFD-ODCNN) method. Mainly, the input videos taken by the IoT devices were pre-processed. Moreover, the SqueezeNet technique was employed for the feature extractor. Besides, the salp swarm optimization (SSO) method was deployed for hyperparameter tuning. In [13], a DL-based pre-impact FD method is developed. This method presented an automatic feature extraction technique, which could remove temporal features in each category of human fall data gathered by employing wearable sensors. A deep neural model depends on the ensemble of CNNs and LSTM was trained under removed temporal features. Alotaibi et al. [14] designed an innovative Arithmetic Optimizer Algorithm with LSTM-AE (AOA-LSTMAE) approach. Primarily, the P-ResNet architecture is used to extract features. Additionally, the AOA-LSTMAE technique employs the LSTMAE classification method for recognizing various activities. AOA was employed as a hyperparameter optimizer model. Xu et al. [15] introduced a wearable device named TTXFD that depends on MPU6050 and could gather tri-axial acceleration signals. Further, this analysis also developed a 2 stage FD method, which combines ML and threshold-based method (TBM). This model utilizes the TBM phase with lower computational complexity for choosing as well as transmitting assumed fall data. The ML phase of the 2-step method was applied with a server depending on CNN.

In [16], a DL Enabled-FD (DL-FD) approach utilizing Gait Analysis was developed. Primarily, an architecture was developed for the FD technique. Then, the authors considered the developed DLFD approach that implements non-fall and falls RGB video for removing gait features through the MediaPipe model, employs the normalization method, and categorizes with the help of the bi-directional-LSTM (Bi-LSTM) algorithm. In [17], an automatic human FD technique was projected employing multi-stream CNNs with combination. This method was dependent upon a multi-level image-fusion model. Al Duhayyim [18] suggested an Automated Disabled People FD employing the CSO with Mobile Networks (ADPFD-CSOMN) method. This approach integrates the development of the MobileNet framework for the feature extractor method. Afterward, the CSO-based parameter tuning method was performed in the MobileNet. Aboutalebi et al. [19] present MAGID, a framework that augments text-only dialogues with diverse, high-quality images via a diffusion method, integrating a feedback loop between image description generation and image quality modules. In [20], a model SLR-AROSNN by employing Artificial Rabbits Optimizer and Siamese Neural Network for Sign Language Recognition is developed. The MobileNet model is used for feature extraction and the Siamese neural network is utilized for classification, improved by the ARO model.

3. The proposed model

In this article, an ARODL-FDC technique is designed in the IoT environment. The ARODL-FDC approach aims to detect and classify fall events to assist elderly people and disabled people. The ARODL-FDC technique comprises four-stage processes, namely GF-based preprocessing, ResNet-based feature extraction, ARO-based hyperparameter tuning, and FENN-based classification model. Figure 1 depicts the workflow of the ARODL-FDC methodology.

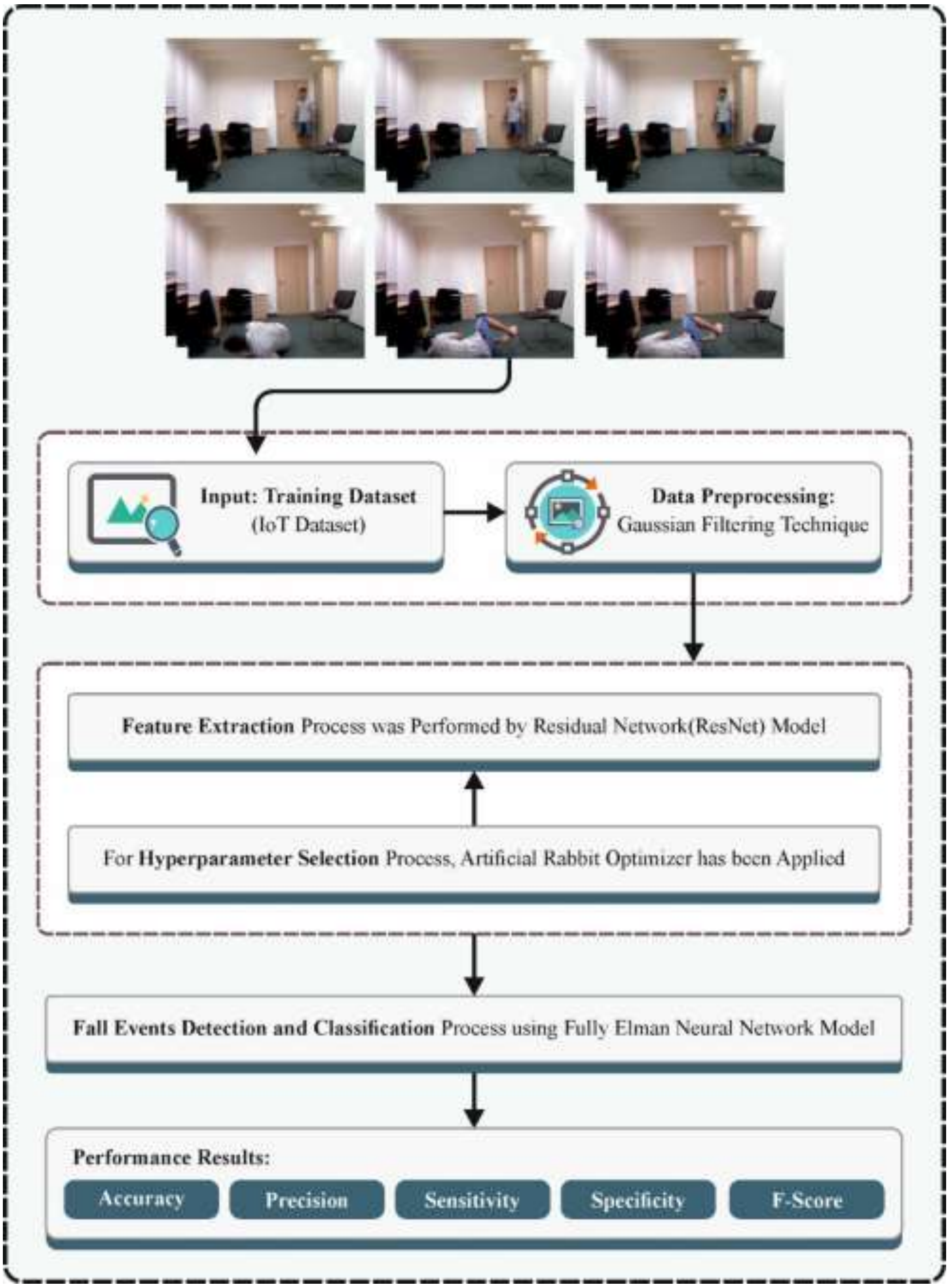
Input: Training Dataset
(IoT Dataset)
Data Preprocessing:
Gaussian Filtering Technique
Feature Extraction Process was Performed by Residual Network(ResNet) Model
For Hyperparameter Selection Process, Artificial Rabbit Optimizer has been Applied
Fall Events Detection and Classification Process using Fully Elman Neural Network Model
Performance Results:
Accuracy
Precision
Sensitivity
Specificity
F-Score

Figure 1. Workflow of ARODL-FDC technique.

3.1. GF-based preprocessing

Initially, the preprocessing of input data is performed by GF. GF has been extremely utilized in image pre-processing systems that comprise convolving an image with a Gaussian kernel for reducing noise and smoothing out differences from the intensity [21]. This linear filter offers further weight to pixels near the kernel center, slowly reducing the effect of pixels far away. Accordingly, GF effectually blurs the image while maintaining its entire design. This method has been appreciated for tasks like edge recognition, noise reduction, and improving subsequent image processing stages by generating a smoother representation of new images. The size of blurring has been organized by the standard deviation (SD) of the Gaussian kernel, permitting flexibility in fine-tuning the filter's strength that relies on the features of the image and the particular pre-processing objectives.

3.2. ResNet feature extractor

For the feature extractor process, the ARODL-FDC technique applies the ResNet model. ResNet is measured as an extension of deeper systems which optimum model for training deeper networks developed [22]. The typical ResNet network contains 101layers, 50layers, and 152layers. Among them, 152-layer deep CNN has earned the 2015-ILSVRC battle. Also, ResNet attained a 28 percent growth on a leading image detection sample database termed cOco132. ResNet mostly feats the concept of bypass networks in "road network" by employing below mentioned mathematical Eqs (1) and (2):

$$g(x_i) = f(x_i) + x_i \tag{1}$$

$$f(x_i) = g(x_i) - x_i \tag{2}$$

In Eq (3), f denotes the transformed signal, and x means the input of the original. An original input further to $f(x)$ through the path of bypass. In Eq (4), $g(x)$ is employed to execute the residual process. ResNet presents shortcut networks to understand links among dissimilar layers; however, these gates are parameter-free and data-independent when equated to highway networks. In the network of highways, these layers signify a non-residual function once the shortcut way is locked. A sub-module was collected of dual portions such as non-linear mapping $F(x)$ and linear direct mapping $x \to x$. If the direct mapping of $x \to x$ is optimum, then the learning technique effortlessly sets all load limits of non-linear mapping $F(x)$ to 0. If no direct mapping, let non-linear mapping $f(x)$ learn a linear $x \to x$, mapping is problematic. However, in ResNet, residual data constantly passed, and the shortcut network never closed.

ReLU is nothing but a function of activation. The main reason for employing the activation function is to stop gradient dispersion, extend the propagation depth of the function, and decrease gradient attenuation produced by deep convolutional. A ReLU expression has been mentioned below:

$$R(x) = \max(O, x) \tag{3}$$

when > 0, $R(x) = x$, derivative will be one, while $x \leq 0$, $R(x) = 0$, then derivative will be zero.

3.3. Hyperparameter tuning using ARO

In this stage, the ARO algorithm has been utilized for the optimum hyperparameter choice of the ResNet system. ARO is one of the present and powerful metaheuristic methods that attract stimulation from existence strategies shown by rabbits [23]. The technique matches nuanced performances detected in rabbit populaces like their operation of arbitrary hiding as well as diversion hunting plans. The core idea of this method solve rounds imitating rabbits' hunting ways where these individuals tactically method plant consumption near neighboring holes to beat potential hunters as well as defend their surroundings. Differing from the conventional tactic of instant consumption, the ARO technique coordinates rabbits to spread their hunt for food to distant locations. This technique uses a populace size similar to a rabbit swarm with every rabbit in the population allocated an eating area decorated with plant life as well as greeneries in combination with numerous holes. At the time of the hunting procedure, rabbits are involved in the accidental examination of other rabbits' holes in the hunt for food. This effort includes gathering food as well as disturbing consistent routines by upgrading their places in unity with particular rabbits; thus a disturbance level or expectable course of action is presented.

$$\vec{\Delta}(t+1) = \vec{z}_j \rightarrow (t) + \rho.(\vec{z}_i \rightarrow (t) - \vec{z}_j \rightarrow (t)) + round(0.5.(0.05 + g_1)).n_1, i, j = 1,2 \ldots\ldots, M \; and \; i \neq j \tag{4}$$

$$E = \left(e - e^{(\frac{1-t}{T})^2}\right).\sin(2\pi g_2) \tag{5}$$

$$c(k) = \begin{cases} 1 \; if \; k == h(u) \\ 0 \; else \end{cases}, k = 1, \ldots, d \& u = 1,2, \ldots, g_3.d \tag{6}$$

where $\rho = E.c$, $h = randperm(d)$, $n_1 \sim N(0,1)$ while $\vec{z}_i(t)$, $\Delta_i(t+1)$, M, T, d, $round$, $randperm$, and E denotes the existing candidate position of i^{th} at time t, i^{th} updated candidate at time $t+1$, dimension of populations of rabbits, entire iteration sizes, problem size, rounding to adjacent integer outcome, arbitrary permutation function extended from 1 to problem size, and running length under the foraging, respectively. At this point, n_1 signifies the normal distribution function, and $g_1, g_2, \; and \; g_3$ refer to the uniform random numbers from the range of 0 and 1. During the exploitation method, rabbits use an arbitrary hiding plan to avoid hunters. It can generate a few holes in close nearness to their unique place. In all the iterations of ARO, a rabbit reliably makes b burrows from all the measurements. Afterward, it arbitrarily picks one of these holes to hide in as well as cheat predators. The i^{th} rabbit with j^{th} burrow is accurately expressed as follows:

$$BU_{ij} \rightarrow (t) = \vec{z}_i(t) + H.h.\vec{z}_i(t), i = 1,2 \ldots\ldots, M \; and \; j = 1,2 \ldots.., d \tag{7}$$

$$H = \frac{1-t+T}{T} g_4 \tag{8}$$

$$h(k) = \begin{cases} 1 \; if \; k == j \\ 0 \; else \end{cases}, k = 1, \ldots, d \tag{9}$$

where $n_2 \sim N(0,1)$, H, and d signify hiding purpose and designed warrens in the region of rabbit, correspondingly. Chiefly, a bunny's big region is anywhere holes can be produced. A random hiding manner is presented below.

$$\Delta_j \wedge (t+1) = \vec{z}_i(t) + \rho.\left(g_4.BU_{ir}(t) - \vec{z}_i(t)\right), i = 1,2, \ldots\ldots M \tag{10}$$

$$h_r(k) = \begin{cases} 1 \; if \; k == [g_5.d], k = 1, \ldots, d \\ 0 \; else \end{cases} \tag{11}$$

$$BU_{ir}(t) = \vec{z}_i(t) + H.h_r.\vec{z}_i(t) \tag{12}$$

where $BU_{ir}(t)$ signifies the chosen hole by the rabbit via hiding mode, g_4, and g_5 refer to random amounts inside the interval [0,1]. $i^{th}rabbit$'s location upgraded after detour hunting mode or random hiding procedure as below.

$$\vec{z}_i(t+1) = \begin{cases} \rightharpoonup z_s(t) & f\big(\rightharpoonup z_s(t)\big) \leq f\big(\rightharpoonup \varDelta_s(t+1)\big) \\ \rightharpoonup \varDelta_s(t+1) & f(\rightharpoonup z_s(t))\}f(\rightharpoonup \varDelta_s(t+1)) \end{cases} \tag{13}$$

If s^{th} rabbit's candidate fitness is superior to the position's existing fitness, the bunny leaves its present place and remains in the candidate place defined by either Eqs (4) or (10). The rabbit's energy decreases with iteration development that aids in the changeover from explorative to exploitative method expressed as mentioned below:

$$EA(t) = 4\left(1 - \frac{t}{T}\right)\ln\left(\frac{1}{\alpha}\right) \tag{14}$$

where α means the random number. When $EA(t) > 1$, the technique searches worldwide for exploration performance; while $EA(t) \leq 1$, the algorithm searches nearby for exploitation solutions.

The ARO system improves a fitness function (FF) to accomplish a larger classifier solution. It explains a positive integer to signify the best efficiency of the candidate outcomes. During this case, the decrease of the classify errors has assumed that FF is formulated as:

$$fitness(x_i) = ClassifierErrorRate(x_i) = \frac{No.of\ misclassified\ instances}{Total\ no.of\ instances} * 100 \tag{15}$$

3.4. FENN-based classification model

Finally, the FENN model has been deployed for the classification and recognition of fall events. In typical ENN, there is single feedback among the hidden layer (HL) and context layer at time t [24]. This feedback is unable to employ spatial, temporal, and long-term data of input as well as output variables. Therefore, to enhance the performance of ENN, a FENN structure was generated in this effort, which links among dual consecutive time points for HL and output, input to output, as well as output layers to HL. Figure 2 demonstrates the infrastructure of ENN.

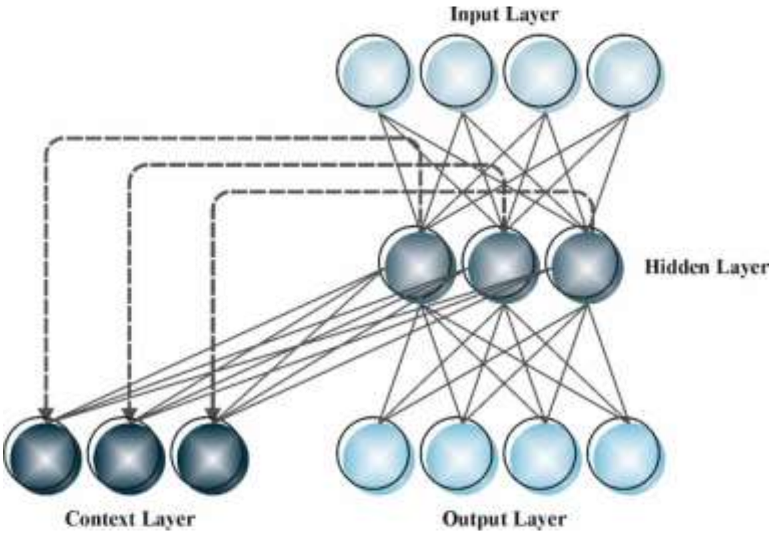

Figure 2. ENN structure.

A measured technique of FENN is expressed as:

$$y(t) = f(w_2x(t) + w_3u(t-1) + y_{c2}(t) + b_2) \tag{16}$$

$$x(t) = g(x_c(t) + w_1u(t-1) + y_{c1}(t) + b_1) \tag{17}$$

$$x_c(t) = w_4x_c(t-1) + w_5x(t-1) \tag{18}$$

$$y_{c1}(t) = w_6y_{c1}(t-1) + w_7y(t-1) \tag{19}$$

$$y_{c2}(t) = w_8y_{c2}(t-1) + w_9y(t-1) \tag{20}$$

where w_2 denotes connection weights among HL to the output layer, w_3 signifies the input layer to HL, and w_1 refers input layer to HL. $y_{c1}(t)$ signifies the outcome context layer 1, $y_{c2}(t)$ denotes outcome context state 2 and $x_c(t)$ signifies context state for HL at t. b_2 and b_1 represent bias for the output layer and HL. w_4, w_6 , and w_8 signify recurrent connection weights amid t and $t-1$ of x_c, y_{c1} and y_{c2}, separately. w_5, w_7 , and w_9 specify connection weights of $x(t-1)$ to $x_c(t)$, $y(t-1)$ to $y_{c1}(t)$ and $y(t-1)$ to $y_{c2}(t).u(t-1)$ means the vector of the input layer at $t-1$ and $y(t)$ and $x(t)$ symbolize the vector of HL and output layer at t, correspondingly. f and g denote the activation function of HL and the output layer that is preferred as hyperbolic tangent and softmax sigmoid, respectively.

4. Performance validation

The experimental validation of the ARODL-FDC methodology has been examined under two databases. The multiple cameras fall (MCF) database [25] includes 192 instances with 2 classes as represented in Table 1.

Table 1. Details of MCF database.

Multiple cameras fall Dataset	
Class	**No. of Samples**
fall event	96
no-fall event	96
Total Samples	**192**

Figure 3 displays the classifier performances of the ARODL-FDC system under the MCF database. Figure 3a and 3b represents the confusion matrices accomplished by the ARODL-FDC technique with 70:30 of the training phase (TRPH)/testing phase (TSPH). The result signified that the ARODL-FDC methodology can be precisely identified and categorized with two class labels. Next, Fig. 3c displays the PR analysis of the ARODL-FDC algorithm. The simulation value describes how the ARODL-FDC technique gets higher PR effectiveness in every class. Besides, Figure 3d shows the ROC outcome of the ARODL-FDC system. This outcome revealed that the ARODL-FDC method provides efficient outcomes with greater ROC values with diverse class labels.

The FD outcomes of the ARODL-FDC system are reported on 70:30 of TRPH/TSPH in Table 2 and Figure 4. The accomplished outcome displayed the ARODL-FDC method gains fall and no-fall events. According to 70% of TRPH, the ARODL-FDC system offers an average $accu_y$ of 99.23%, $prec_n$ of 99.29%, $sens_y$ of 99.23%, $spec_y$ of 99.23%, and F_{score} of 99.25%. Moreover, with 30% of

TSPH, the ARODL-FDC algorithm provides an average $accu_y$ of 98.39%, $prec_n$ of 98.21%, $sens_y$ of 98.39%, $spec_y$ of 98.39%, and F_{score} of 98.27%, respectively.

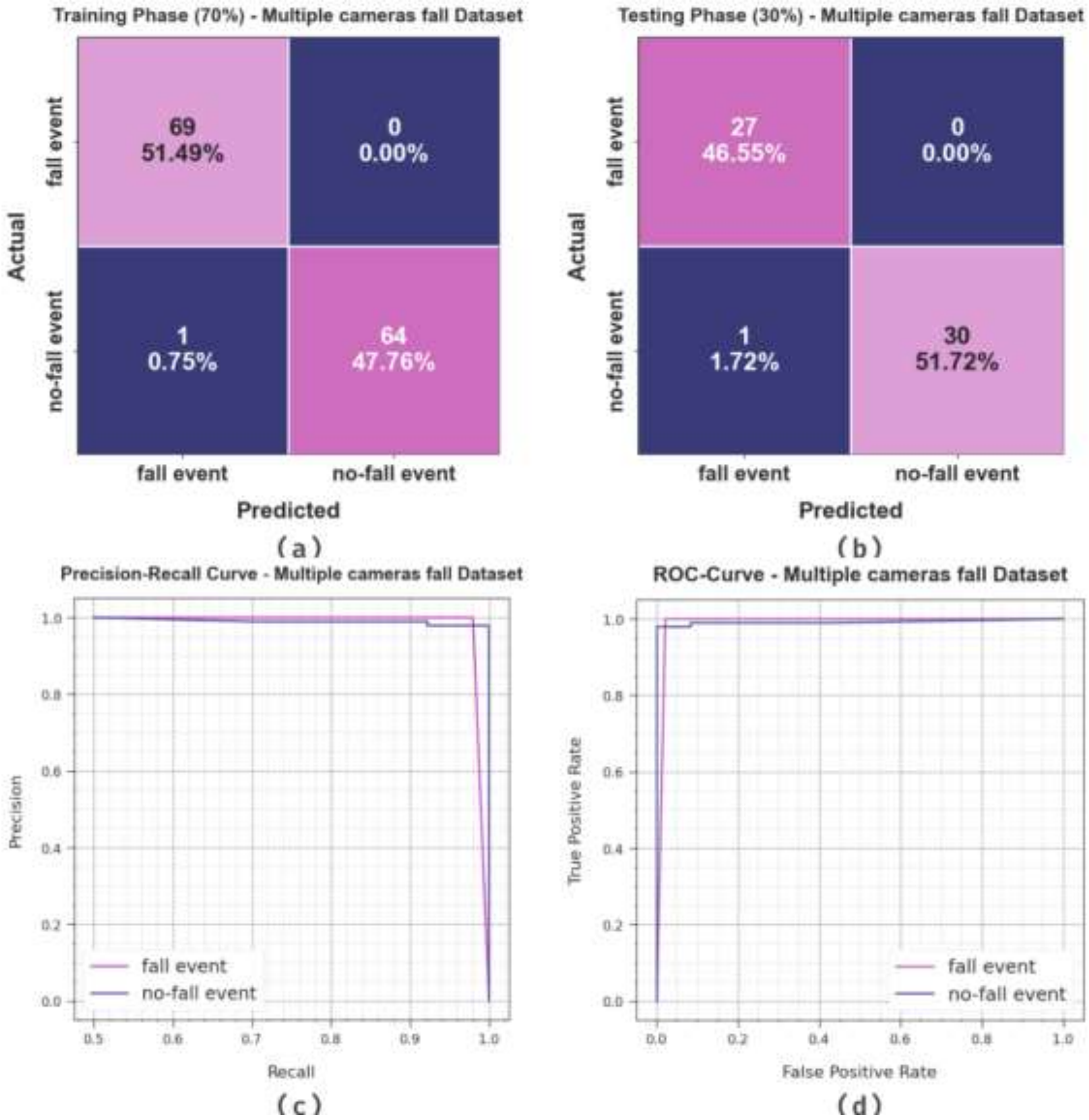

Figure 3. MCF database (a-b) confusion matrices (c-d) PR and ROC curves.

Table 2. FD outcome of the ARODL-FDC system under the MCF dataset.

Classes	$Accu_y$	$Prec_n$	$Sens_y$	$Spec_y$	F_{score}
TRPH (70%)					
fall event	100.00	98.57	100.00	98.46	99.28
no-fall event	98.46	100.00	98.46	100.00	99.22
Average	**99.23**	**99.29**	**99.23**	**99.23**	**99.25**
TSPH (30%)					
fall event	100.00	96.43	100.00	96.77	98.18
no-fall event	96.77	100.00	96.77	100.00	98.36
Average	**98.39**	**98.21**	**98.39**	**98.39**	**98.27**

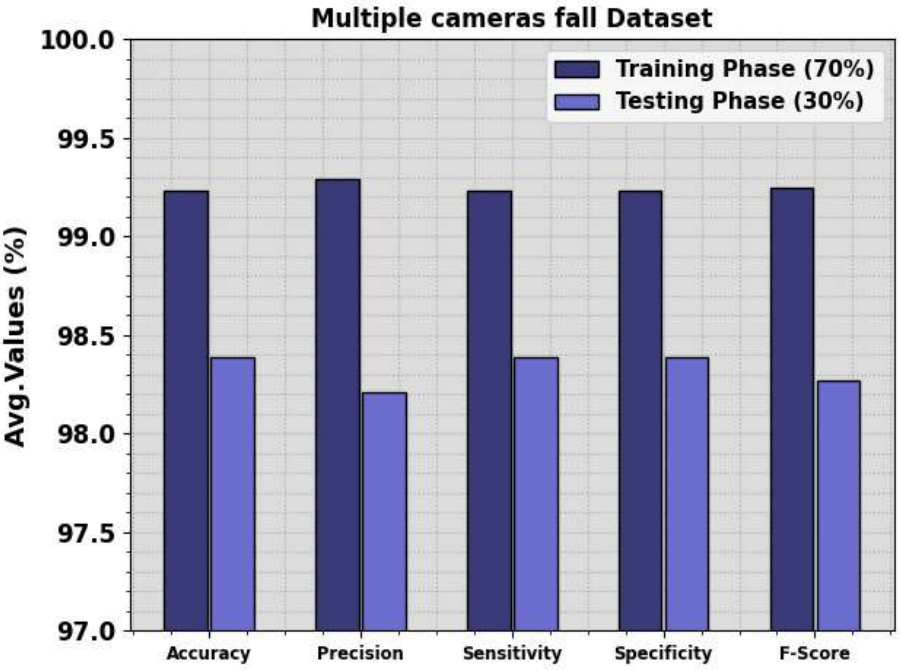

Figure 4. Average outcomes of the ARODL-FDC model on the MCF dataset.

The $accu_y$ curves for TR and validation (VL) displayed in Figure 5 for the ARODL-FDC algorithm under the MCF dataset give valued insights into its performance with numerous epochs. Particularly, it has a constant improvement in both TR and TS $accu_y$ to raising epochs, exhibiting the model's ability to learn and identify patterns in data of both TR and TS. The upward trend in TS $accu_y$ highlights the model's flexibility to the TR dataset and its capacity to generate accurate predictions on unnoticed data, emphasizing better-generalized proficiencies.

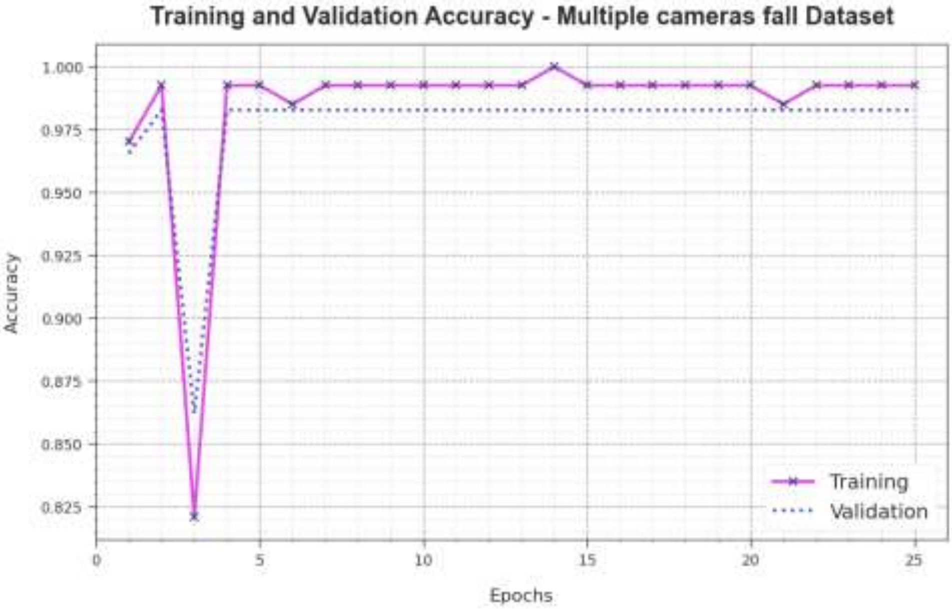

Figure 5. $Accu_y$ curve of the ARODL-FDC model under the MCF dataset.

Figure 6 represents a wide-ranging overview of the TR and TS loss values for the ARODL-FDC method under the MCF dataset through diverse epochs. This TR loss reliably decreases as the model increases their weights for diminishing classification errors at two datasets. The loss curves represent the model's alignment with the dataset of TR, highlighting its ability to competently capture forms in both datasets. Moreover, this can be a continuous enhancement of parameters in the ARODL-FDC methodology, which targets decreasing discrepancies among predictions and actual TR labels.

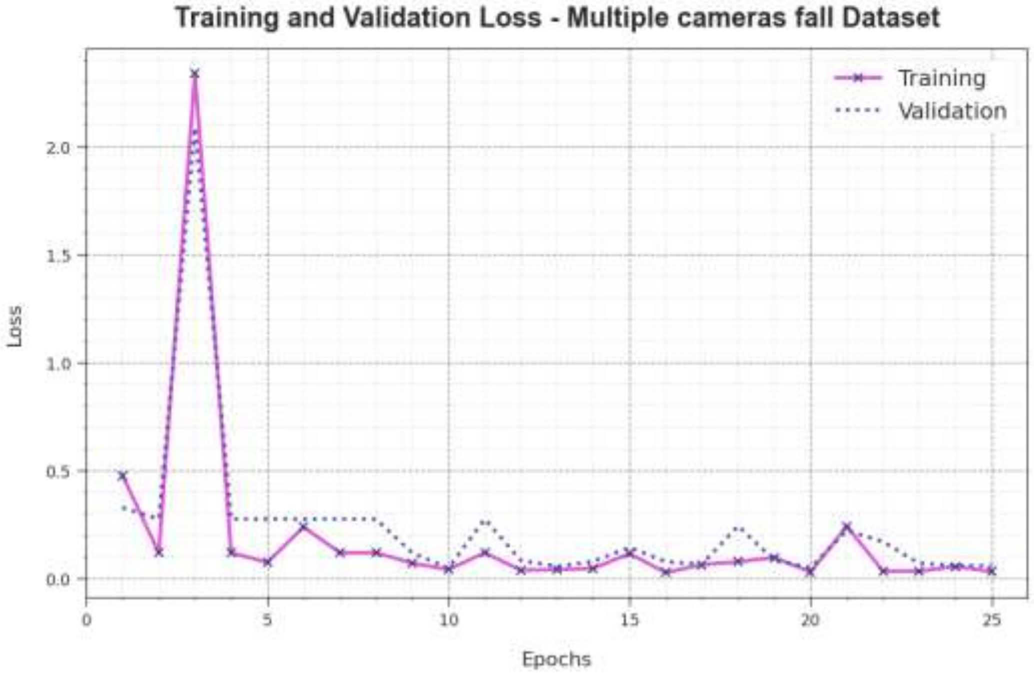

Figure 6. Loss curve of the ARODL-FDC algorithm under MCF database.

In Table 3 and Figure 7, the comparison outcome of the ARODL-FDC algorithm with the MCF database is given [12]. The achieved outcome represented that the 2D-ConvNN model obtained the least $accu_y$ of 95.68%. Furthermore, the ResNet50, ResNet101, VGG16, and Depthwise models accomplish closer $accu_y$ values. Although the IMEFDOD-CNN model gets near-optimal performance, the ARODL-FDC technique exhibits superior outcomes with a maximum $accu_y$ of 99.23%.

Table 3. $Accu_y$ analysis of the ARODL-FDC model with other techniques on the MCF database.

Multiple cameras fall Dataset	
Methods	**Accuracy (%)**
VGG16	98.16
Depthwise	97.95
2D-ConvNN	95.68
ResNet50	96.26
ResNet101	96.67
IMEFDOD-CNN	99.03
ARODL-FDC	99.23

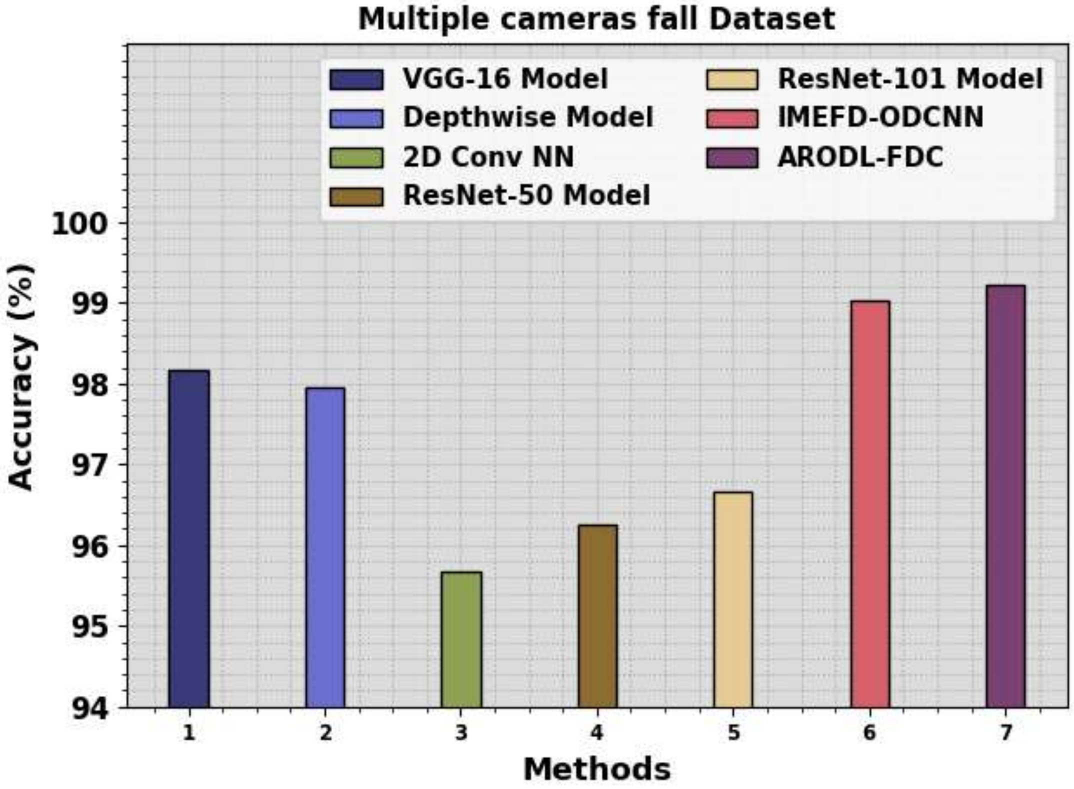

Figure 7. $Accu_y$ outcome of the ARODL-FDC model with other techniques under the MCF dataset.

The UR FD (URFD) dataset [26] comprises 314 instances with 2 classes as demonstrated in Table 4.

Table 4. Details of the URFD dataset.

UR FD (URFD) Database	
Class	**No. of Samples**
fall event	74
no-fall event	240
Total Samples	**314**

Figure 8 represents the classifier performances of the ARODL-FDC system at the URFD dataset. Figure 8a and 8b displays the confusion matrices attained by the ARODL-FDC system in 70:30 of TRPH/TSPH. This simulation value shows that the ARODL-FDC algorithm can be correctly recognized and categorized into two classes. Moreover, Figure 8c represents the PR analysis of the ARODL-FDC system. The outcome described the ARODL-FDC technique to attain great PR effectiveness with every class label. Then, Figure 8d shows the ROC analysis of the ARODL-FDC technique. This result revealed the ARODL-FDC system offers effective outcomes with increased ROC outcomes with different classes.

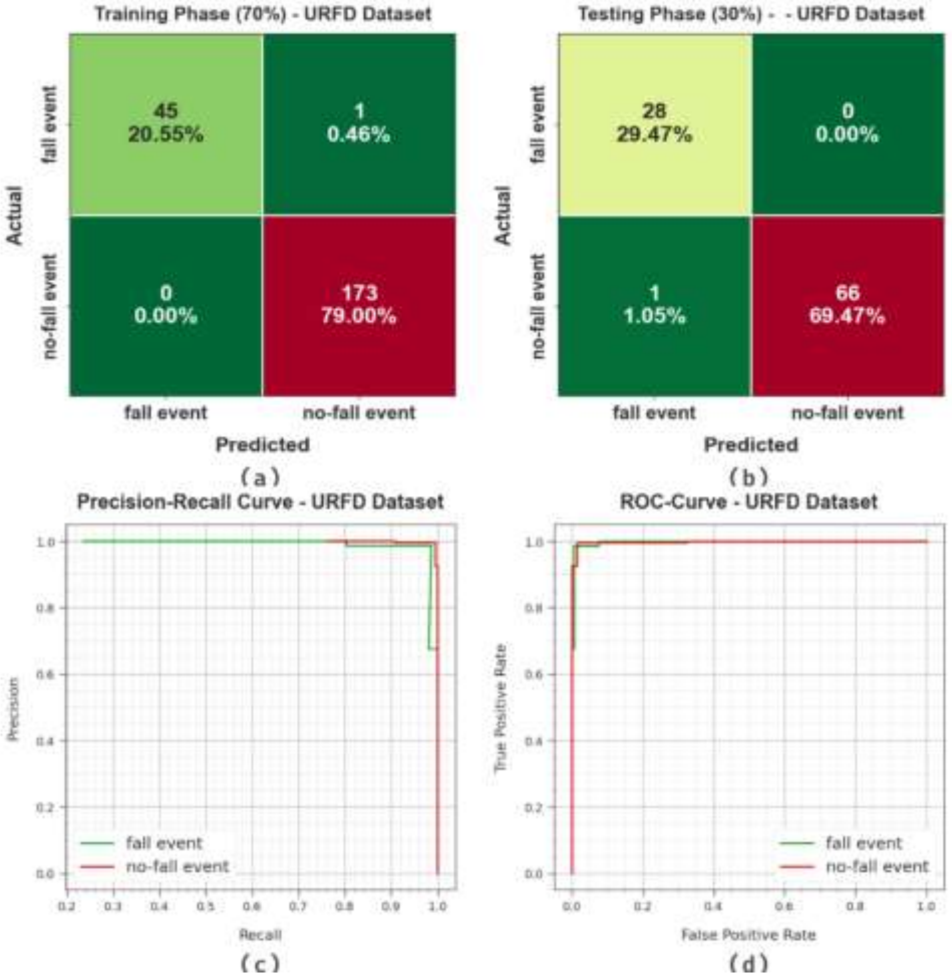

Figure 8. URFD database (a-b) confusion matrices and (c-d) PR and ROC curves.

The FD analysis of the ARODL-FDC approach is defined in Table 5 and Figure 9. The achieved outcome revealed the ARODL-FDC method acquires fall and no-fall events. Based on 70% of TRPH, the ARODL-FDC algorithm gives an average $accu_y$ of 99.54%, $prec_n$ of 99.71%, $sens_y$ of 98.91%, $spec_y$ of 98.91%, and F_{score} of 99.31%. Also, for 30% of TSPH, the ARODL-FDC methodology offers an average $accu_y$ of 98.95%, $prec_n$ of 98.28%, $sens_y$ of 99.25%, $spec_y$ of 99.25%, and F_{score} of 98.75% correspondingly.

Table 5. FD analysis of the ARODL-FDC system under the URFD dataset.

Classes	$Accu_y$	$Prec_n$	$Sens_y$	$Spec_y$	F_{score}
TRPH (70%)					
fall event	99.54	100.00	97.83	100.00	98.90
no-fall event	99.54	99.43	100.00	97.83	99.71
Average	**99.54**	**99.71**	**98.91**	**98.91**	**99.31**
TSPH (30%)					
fall event	98.95	96.55	100.00	98.51	98.25
no-fall event	98.95	100.00	98.51	100.00	99.25
Average	**98.95**	**98.28**	**99.25**	**99.25**	**98.75**

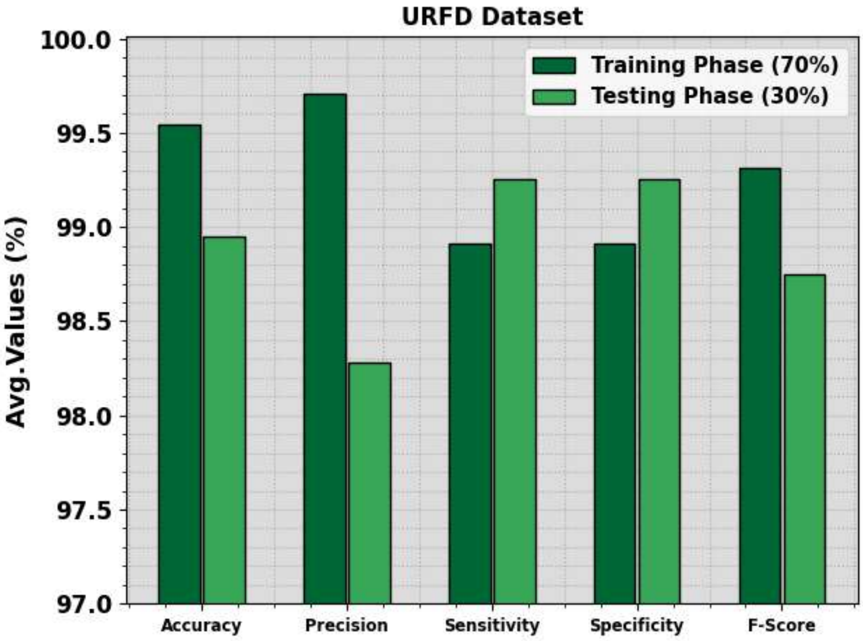

Figure 9. Average outcomes of the ARODL-FDC model with URFD dataset.

The $accu_y$ curves for TR and VL exhibited in Figure 10 for the ARODL-FDC algorithm at the URFD dataset provide respected insights into its effectiveness with different epochs. In particular, this can be a reliable improvement in both TR and TS $accu_y$ to raising epochs, showing the model's ability to recognize and learn patterns in both data of TR and TS. The upward trends in TS $accu_y$ highlight the model's adaptableness to the TR dataset as well as its capabilities to create exact predictions on unobserved data, emphasizing optimum generalization proficiencies.

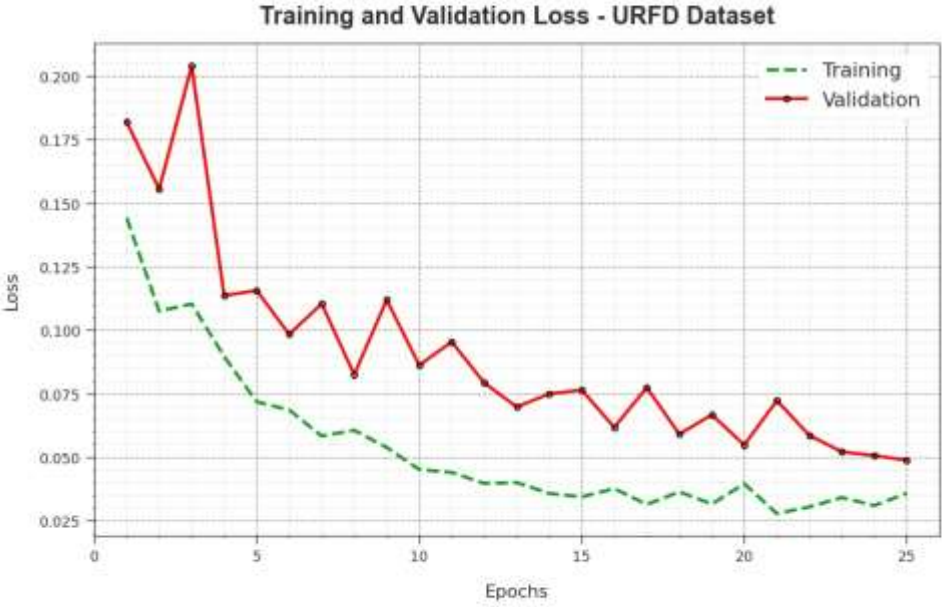

Figure 10. $Accu_y$ curve of the ARODL-FDC algorithm on URFD database.

Figure 11 offers a wide-ranging overview of the TR and TS loss values to the ARODL-FDC system under the URFD dataset in numerous epochs. The TR loss is consistently reduced as the model enriches its weights to minimize classification errors with these datasets. These loss curves noticeably represent the model's alignment with the dataset of TR, underscoring proficiencies in capturing patterns effectively in these databases. The constant modification of parameters in the ARODL-FDC system is intended to minimize discrepancies between predictions and actual TR labels.

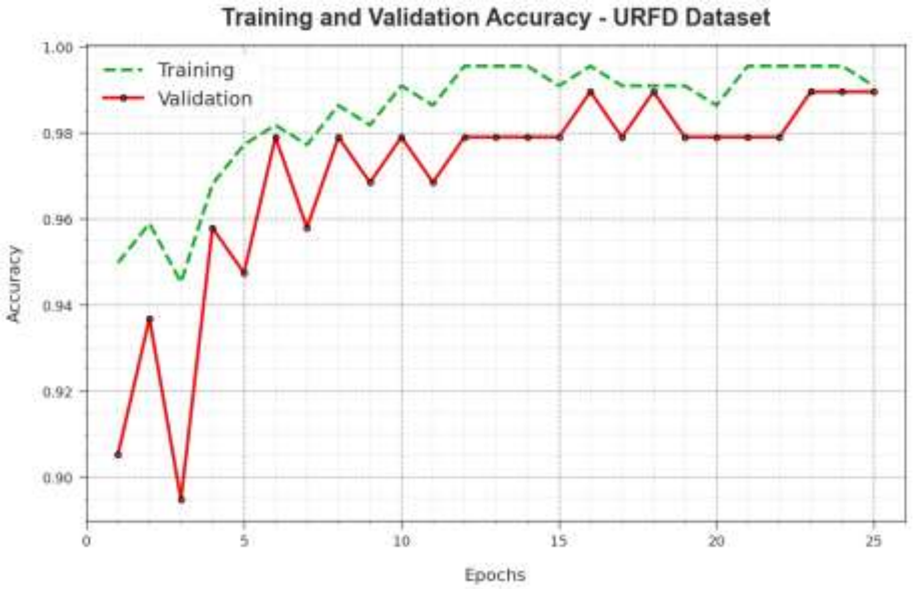

Figure 11. Loss curve of the ARODL-FDC approach with URFD database.

In Table 6 and Figure 12, the comparative outcome of the ARODL-FDC technique on the URFD database can be described. The accomplished outcome displayed that the 2D-ConvNN technique gets a minimum $accu_y$ of 95.18%. Then, the ResNet50, ResNet101, VGG16, and Depthwise systems achieve nearer $accu_y$ values. However, the IMEFDOD-CNN algorithm achieves nearby optimum performance, and the ARODL-FDC technique exhibits superior outcomes with a higher $accu_y$ of 99.54%, respectively.

Table 6. $Accu_y$ analysis of the ARODL-FDC model with other techniques under the URFD dataset.

UR FD (URFD) Dataset	
Methods	**Accuracy (%)**
VGG16	97.76
Depthwise	98.15
2D-ConvNN	95.18
ResNet50	95.55
ResNet101	96.36
IMEFDOD-CNN	99.21
ARODL-FDC	99.54

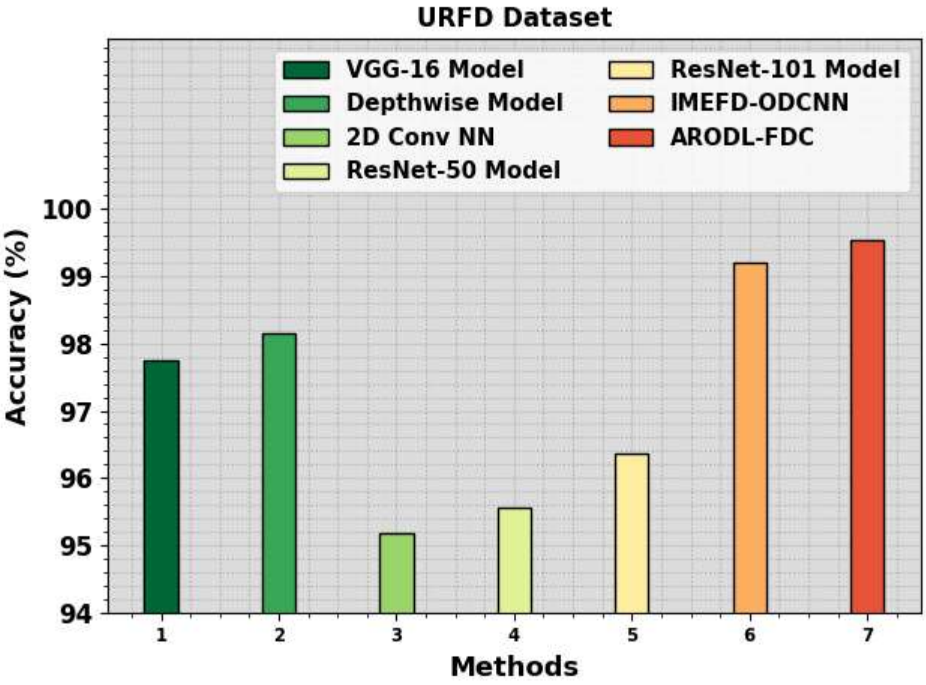

Figure 12. $Accu_y$ analysis of the ARODL-FDC model with other techniques on the URFD dataset.

Thus, the ARODL-FDC technique can be applied for enhanced FD processes in the IoT environment**.**

5. Conclusions

In this study, an ARODL-FDC technique is designed in the IoT environment. The ARODL-FDC approach proposes to detect and categorize fall events to assist elderly people and disabled people. The ARODL-FDC technique comprises four-stage processes, namely GF-based preprocessing, ResNet-based feature extractor, ARO-based hyperparameter tuning, and FENN-based classification model. Initially, the preprocessing of input data is performed by GF. The ARODL-FDC technique applies the ResNet approach for feature extractor purposes. In addition, the ARO algorithm has been employed for better hyperparameter choice of the ResNet algorithm. Eventually, the FENN model can be utilized for the classification and recognition of fall events. The experimental results of the ARODL-FDC technique can be tested on the fall dataset. The simulation results inferred that the ARODL-FDC technique reaches promising performance over compared models concerning various aspects. The ARODL-FDC technique portrays efficiency but may be restricted in complex environments. Future studies may be focus on improvising its adaptiveness in multi-person scenarios and incorporating further sensor modalities for enhanced accuracy.

Acknowledgment

The authors extend their appreciation to the King Salman Center for Disability Research for funding this work through Research Group no KSRG-2022-117

Conflict of interest

The authors declare that they have no conflicts of interest.

Author contributions

The manuscript was written through the contributions of all authors. All authors have approved the final version of the manuscript.

Use of AI tools declaration

The authors declare they have not used Artificial Intelligence (AI) tools in the creation of this article.

References

1. H. El Zein, F. Mourad-Chehade, H. Amoud, Leveraging Wi-Fi CSI Data for Fall Detection: A Deep Learning Approach, In *2023 5th International Conference on Bio-engineering for Smart Technologies (BioSMART)*, 2023, IEEE, 1–4. https://doi.org/10.1109/BioSMART58455.2023.10162090
2. J. Wang, W. C. Wang, K. W. Chau, L. Qiu, X. X. Hu, H. F. Zang, et al., An improved Golden Jackal Optimization Algorithm based on multi-strategy mixing for solving engineering optimization problems, *J. Bionic Eng.*, 2024, 1–24. https://doi.org/10.1007/s42235-023-00469-0
3. B. M. Sundaram, B. Rajalakshmi, R. K. Mandal, S. Nair, S. S. Choudhary, Fall Detection Among Elderly Using Deep Learning, In *2023 International Conference on Intelligent and Innovative Technologies in Computing, Electrical and Electronics (IITCEE)*, 554–558. IEEE, 2023. https://doi.org/10.1109/IITCEE57236.2023.10090887
4. Z. Lian, W. Wang, Z. Han, C. Su, Blockchain-based personalized federated learning for internet of medical things, *IEEE T. Sust. Comput.*, 2023. https://doi.org/10.1109/TSUSC.2023.3279111
5. F. Ahamed, S. Shahrestani, H. Cheung, Privacy-Aware IoT Based Fall Detection with Infrared Sensors and Deep Learning, In *International Conference on Interactive Collaborative Robotics, Cham: Springer Nature Switzerland*, 2023, 392–401. https://doi.org/10.1007/978-3-031-35308-6_33
6. W. C. Wang, L. Xu, K. W. Chau, Y. Zhao, D. M. Xu, An orthogonal opposition-based-learning Yin-Yang-pair optimization algorithm for engineering optimization, *Eng. Comput.*, 2021, 1–35.
7. J. E. Rivadeneira, M. B. Jiménez, R. Marculescu, A. Rodrigues, F. Boavida, J. Sá Silva, A Blockchain-Based Privacy-Preserving Model for Consent and Transparency in Human-Centered Internet of Things, In *Proceedings of the 8th ACM/IEEE Conference on Internet of Things Design and Implementation*, 2023, 301–314. https://doi.org/10.1145/3576842.3582379
8. M. Jarrah, H. Al Hamadi, A. Abu-Khadrah, T. M. Ghazal, IoMT-Based smart healthcare of elderly people using deep extreme learning machine, *Comput. Mater. Con.*, **76** (2023). https://doi.org/10.32604/cmc.2023.032775
9. W. Wang, W. Tian, K. W. Chau, Y. Xue, L. Xu, H. Zang, An improved bald eagle search algorithm with Cauchy mutation and adaptive weight factor for engineering optimization, *Comput. Model. Eng. Sci.*, **136** (2023), 1603–1642. https://doi.org/10.32604/cmes.2023.026231

10. B. Ozkaya, S. Duman, H. T. Kahraman, U. Guvenc, Optimal solution of the combined heat and power economic dispatch problem by adaptive fitness-distance balance based artificial rabbits optimization algorithm, *Expert Syst. Appl.*, **238** (2024), 122272. https://doi.org/10.1016/j.eswa.2023.122272
11. E. Alabdulkreem, R. Marzouk, M. Alduhayyem, M. A. Al-Hagery, A. Motwakel, M.A. Hamza, Chameleon Swarm Algorithm with improved fuzzy deep learning for fall detection approach to aid elderly people, *J. Disabil. Res.*, **2** (2023), 62–70. https://doi.org/10.57197/JDR-2023-0020
12. T. Vaiyapuri, E. L. Lydia, M. Y. Sikkandar, V. G. Díaz, I. V. Pustokhina, D. A. Pustokhin, Internet of things and deep learning enabled elderly fall detection model for smart homecare, *IEEE Access*, **9** (2021), 113879–113888. https://doi.org/10.1109/ACCESS.2021.3094243
13. R. Jain, V. B. Semwal, A novel feature extraction method for preimpact fall detection system using deep learning and wearable sensors, *IEEE Sens. J.*, **22** (2022), 22943–22951. https://doi.org/10.1109/JSEN.2022.3213814
14. F. Alotaibi, M. M. Alnfiai, F. N. Al-Wesabi, M. Alduhayyem, A. M. Hilal, M. A. Hamza, Internet of Things-driven Human Activity Recognition of elderly and disabled people using Arithmetic Optimization Algorithm with LSTM autoencoder, *J. Disabil. Res.*, **2** (2023), 136–146. https://doi.org/10.1109/JSEN.2022.3213814
15. T. Xu, H. Se, J. Liu, A two-step fall detection algorithm combining threshold-based method and convolutional neural network, *Metrol. Meas. Syst.*, **28** (2021).
16. A. R. Anwary, M. A. Rahman, A. J. M. Muzahid, A. W. U. Ashraf, M. Patwary, A. Hussain, Deep Learning enabled Fall Detection exploiting Gait Analysis, In *2022 44th Annual International Conference of the IEEE Engineering in Medicine & Biology Society (EMBC)*, IEEE, 2022, 4683–4686. https://doi.org/10.1109/EMBC48229.2022.9871964
17. T. Alanazi, G. Muhammad, Human fall detection using 3D multi-stream convolutional neural networks with fusion, *Diagnostics*, **12** (2022), 3060. https://doi.org/10.3390/diagnostics12123060
18. M. Al Duhayyim, Automated disabled people fall detection using Cuckoo Search with mobile networks, *Intell. Autom. Soft Co.*, **36** (2023). https://doi.org/10.32604/iasc.2023.033585
19. H. Aboutalebi, H. Song, Y. Xie, A. Gupta, J. Sun, H. Su, et al., MAGID: An Automated Pipeline for Generating Synthetic Multi-modal Datasets, arXiv preprint arXiv:2403.03194, 2024.
20. R. Marzouk, F. Alrowais, F. N. Al-Wesabi, A. M. Hilal, Sign language recognition using Artificial Rabbits Optimizer with Siamese Neural Network for persons with disabilities, *J. Disabil. Res.*, **2** (2023), 31–39. https://doi.org/10.57197/JDR-2023-0047
21. V. Nyemeesha, B. M. Ismail, Implementation of noise and hair removals from dermoscopy images using hybrid Gaussian filter, *Netw. Model. Anal. Health*, **10** (2021), 1–10. https://doi.org/10.1007/s13721-021-00318-2
22. Y. Chen, Y. Lin, X. Xu, J. Ding, C. Li, Y. Zeng, et al., Classification of lungs infected COVID-19 images based on inception-ResNet, *Comput. Meth. Prog. Bio.*, **225** (2022), 107053. https://doi.org/10.1016/j.cmpb.2022.107053
23. D. Izci, R. M. Rizk-Allah, V. Snášel, S. Ekinci, F. A. Hashim, L. Abualigah, A novel control scheme for automatic voltage regulator using novel modified Artificial Rabbits Optimizer, e-Prime-Advances in Electrical Engineering, *Electron. Energy*, 100325, 2023. https://doi.org/10.1016/j.cmpb.2022.107053

24. M. Fetanat, M. Stevens, P. Jain, C. Hayward, E. Meijering, N. H. Lovell, Fully Elman neural network: A novel deep recurrent neural network optimized by an improved Harris Hawks algorithm for classification of pulmonary arterial wedge pressure, *IEEE T. Biomed. Eng.*, **69** (2021), 1733–1744. https://doi.org/10.1109/TBME.2021.3129459
25. E. Auvinet, C. Rougier, J. Meunier, A. St-Arnaud, J. Rousseau, Multiple cameras fall dataset, Technical report 1350, DIRO - Université de Montréal, July 2010.
26. http://fenix.ur.edu.pl/~mkepski/ds/uf.html