**International Journal of Applied Mathematics**
**Volume 38 No. 10s, 2025**
ISSN: 1311-1728 (printed version); ISSN: 1314-8060 (on-line version)

# HUMAN ACTION RECOGNITION NETWORK CLASSIFIER LEVERAGING SVM-DIRECTED MACHINE LEARNING

**Dr.Reshma Banu[1*], Dr G F Ali Ahammed[2], Dr. Prakash V. Parande[3], Dr. Swamy L N[4], Mr. Tammmarguddi Mahmad Husen[5], Dr. Yogesha T[6]**

[1*]Professor & Principal Researcher Computing Sciences-CSE, ISE, MCA, R&D cell VVIET Mysuru, dr.reshmabanu86@gmail.com, ORCID: 0000-0002-3850-041X
[2]Associate Professor, Department of CSE, VTU CPGS, Mysuru, Karnataka, India. ORCID: 0000-0003-3312-5427
[3]Assistant Professor, Department of CSE, VTU CPGS, Belagavi, Karnataka, India. ORCID:0000-0002-1297-7789
[4]Assistant Professor, Department of CSE, VTU CPGS, Mysuru, Karnataka, India. ORCID: 0000-0001-9586-6824
[5]Research Scholar, Department of CSE, VTU CPGS, Mysuru, Karnataka, India. ORCID: 0009-0000-8780-683X
[6]Assistant Professor, Department of CSE, VTU CPGS, Mysuru, Karnataka, India. ORCID: 0000-0002-7662-6064

**Abstract:**
To ensure public safety, successful surveillance video analysis requires both an understanding of human behavior and the ability to recognize human actions. However, because they require a large amount of parameters, existing approaches like neural networks and three-dimensional convolutional neural networks (CNN) have computing challenges. To address these challenges with regard to efficient human action detection, a specific residual CNN based on directed acyclic graphs was created in this work. Effective implicit representational acquisition of human movements is rendered simpler by applying the suggested method's novel pipeline for generating spatial motion data from raw video inputs. The implicit representations, are used as deep learning protocols and are maintained in a fully linked layer. Action recognition is then accomplished by passing these suggested approaches into the Support Vector Machine (SVM) classifier. Simulations were conducted using MATLAB software on widely used behavior recognition statistics to assess the created SVM algorithm.

**Keywords:** Pre-processing, Predictive modelling, convolutional neural network Hybrid optimization, Segmentation, Activity recognition, Machine learning

## I. INTRODUCTION

Numerous computer science and Artificial intelligence communities still put considerable emphasis on human activity recognition research due to its potential to function a wide range of applications, including surveillance, eHealth, and human-computer interaction. In modern times, the human activity recognition system is often categorized into either a deep model or a classical model that employs the feature extraction strategy [1]. To address the issue of human activity recognition, the conventional method employs explicitly developed feature descriptors that fall into one of three categories: local features, global features, or an assortment of either of them. The illustration can be distinguished as its entirety by the global characteristics, which constitute the movements of the human body. To define the representation of areas of a human activity, however, the local traits are taken from an assortment of spatio-temporal interest points (STIPs) [2]. Global strategies are highly sensitive to changes in the setting and partial blockages, nevertheless, can capture more visual information by retaining the spatio-temporal structures of the events that took place within the video

[3, 4]. The image is viewed as minute spots by the local traits such as which is generally computationally expensive. In accordance to the computational complexity, human events can be divided into four distinct groups: gestures, group activity, interactions, and acts. In the digital era, an increasing amount of footage demands professional statistical methods for optimal interpretation. Tracking people's daily activities using a variety of technologies, such as cellphones and wearable sensors, is a significant application scenario for HAR. Smartphones monitor users' movements, which motivates them to move throughout the period of observation. A significant field with many applications, spanning security surveillance, athletics analysis, healthcare, and human-computer interface, is human activity recognition.
A dataset is an essential requirement for a system that identifies human conduct. For the research, the video or image dataset's, which contains several samples to assess the HAR system, is mandatory. Schematic block representation of Human Action Recognition is shown in Fig 1.

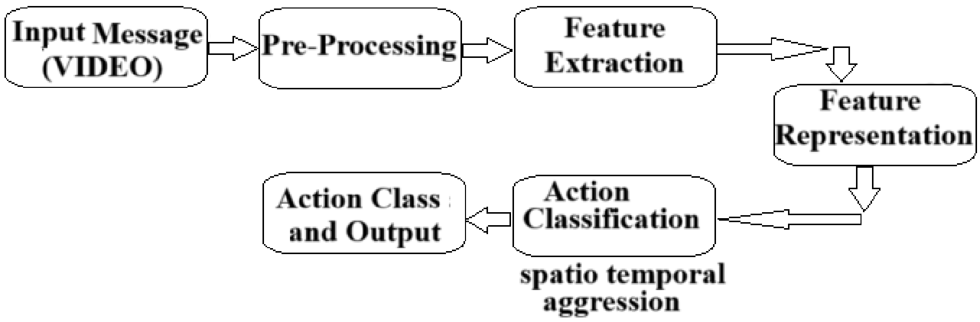



Fig 1:Schematic Representation of HAR

In Recent times, Convolution Neural Network CNN applications has been observed significant enhancements in video activity recognition performance. Owing to the rapid enhancement in deep CNN models' performance, the academic and research community began employing CNNs for human action recognition[5, 6]. Learning accessories and classifiers are used for developing an automated HAR system. Reliable human action modeling and feature representation are offered in a feasible HAR system. With computer vision technique, the representation and selection of features is an important challenge. The HAR system's function is to recognize activities on images, which split up frames of a video, and to extract data about the frames' rapidly shaping effects, such as standing, running, etc[5]. The two reasons for the significant delay are in comparison to still image recognition datasets, the size and range of existing video activity recognition datasets are fragile. Consequently, restricted datasets will over-fit the model and fail to provide the action classification generalized answer. Developing large video collection and training them on deep networks is extremely challenging and expensive. Second, because the video classification task includes additional facts known as temporal data, it requires significant data processing. The remedy to these intricate action categorization problems with videos has recently been the focus of multiple research efforts.

## II. STATE OF ART RELATED WORK

In paper [7] the authors discusses the rank of human activity employing information gathered by smartphone sensors (gyroscope and accelerometer). Various actions performed by humans will be classified as standing, sitting, walking, lying down, going up and down stairs. Machine learning models are created and optimized to set up HAR from data gathered with the best outcomes. The traditional machine learning algorithms that have been used are Random Forest, Decision Tree, Linear and Kernel SVM, and Logistic Regression.

ISSN: 1311-1728 (printed version); ISSN: 1314-8060 (on-line version)

In the paper [8, 9] discusses about author developed a novel based 1D-CNN deep learning approach for recognizing human activities in this work, which focused on sensor-based activity detection. Using raw accelerometer and gyroscope sensor data from three publicly available datasets—UCI-HAPT, WISDM, and PAMAP2 and author validated the model. To specify the model's architecture and adapt the hyper-parameters of the resulting design, parameter optimization was adopted. Using the UCI-HAPT dataset, the author employed 6, 7, and 12 classes of activity recognition, achieving 98%, 96.9%, and 94.8% accuracy rates, respectively. Additionally, the author's accuracy rates on the WISDM and PAMAP2 datasets were 97.8% and 90.27%, respectively. Additionally, author looked at the effects of utilizing each sensor data separately, and outcomes suggest that proposed model performed better when both sensor data were applied.

Authors in paper[10,11] suggests an approach for enhancing the accuracy of HAR by optimizing the one-dimensional (1D-CNN) hyperparameters. The framework incorporates a parametric model for 1D-CNNs with a hyperparameter optimization mechanism. The current research developed the OPTConvNet methodology, which uses Hierarchical Particle Swarm Optimization (H-PSO) for maximizing the hyperparameters of 1D-CNN. The objective of the H-PSO strategy is to optimize 1D-CNN's architecture, layers, and training parameters. The 1D-CNN's architecture has initially optimized using the H-PSO [12]. The following phase will be the optimization of the layer and training hyperparameters. In order to avoid premature convergence to a local optimal solution in the PSO algorithm, the suggested technique uses an exponential-like inertia weight to fine-tune the balance between particle exploration and exploitation. Publicly accessible sensor-human activity recognition (S-HAR) datasets, which include the UCI-HAR, Daphnet Gait, Opportunity, and PAMPA2 datasets, are used to assess the H-PSO-CNN.

The paper [13] author discusses about The most challenging aspect of the pipeline to recognize human behavior using wearable and mobile sensors is extracting pertinent attributes. Feature extraction enhances the algorithm's performance while minimizing computation time and complexity. Nevertheless, nowadays methods for identifying human activity are based on manually constructed traits that are unable to manage complex activities. Deep learning is one of the many existing methods used to recognize individual activities; nevertheless, more has to be created to identify action transitions [14]. A novel deep learning-based approach to identifying human activities is put forth in this research to address these problems. Initially a genetic algorithm (FGA) based on fuzzy logic is employed to extract characteristics from sensor data. The objective of the novel proposed feature selection technique, PSOACO, is to enhance the efficiency of the particle swarm optimizer (PSO) algorithm by utilizing the ant colony optimizer (ACO) operators. To identify the activities, the Deep Convolutional Neural Network and Long Short Term Memory (DCNN-LSTM) deep learning technique is next employed. Leveraging F1 Score, Accuracy, Precision, and Recall, the efficacy of the implementation is evaluated.

## III. PROPOSED METHODOLOGY

### 3.1 Classification models

A. Support vector machines (SVM)

Support vector machines (SVM) represent a supervised machine-learning approach applicable to both classification and regression tasks. While SVMs are fundamentally designed for binary classification, they can be adapted to handle multiclass scenarios through various techniques. The primary objective of SVMs is to identify the optimal hyperplane that maximizes the margin between different classes. Support vectors, which are a limited subset of the training dataset, define the hyperplane's position and are identified upon the completion of the training process. SVMs are capable of executing both linear and nonlinear classification effectively. A kernel technique is required for nonlinear classification, which converts features into a high-dimensional space. If the training data subset are not linearly separable, a kernel function SVM is used to transmit the data to

**International Journal of Applied Mathematics**
**Volume 38 No. 10s, 2025**
ISSN: 1311-1728 (printed version); ISSN: 1314-8060 (on-line version)

a new vector space. SVM yields reliable and pertinent outcomes and performs well with significant training datasets.
Given training data set be
$D\{(x_1, y_1),(x_2, y_2), ...(x_N , y_N)\}$ $\forall \; x \in R^n ; y_i \in \pm 1$ (1)
Where i represents the in the defined data-set in each action.

To determine a decision function for a linear classification, the hyperplane separation is represented by:
$y_i = sng((\omega \Delta x_i) + b)$ (2)
A generic hyperplane is defined by satisfying the condition and if delimited by margins is given by
$w \cdot x_i + b = 0$ (3)
$y_i ((w \cdot x_i) + b) \geq 1$ (4)

B. K-nearest neighbor
K-Nearest Neighbors is sometimes referred to as a lazy learner algorithm because the algorithm safeguards the dataset before it takes an action on it during classification rather than learning from the training set instantaneously [15, 17].
The steps involved in the KNN algorithm are:
1. **Selecting the Optimal Value of K:** This involves experimenting with different values of K where K represents the number of nearest neighbors to consider while making a prediction. Choosing the one that minimizes errors and optimizes performance. Methods like cross-validation can help in this selection process.

Steps taken for selecting the optimal value of K are as follows
a. **Range of K Values:** Selecting a range of K values to test. Typically, this range is from 1 to around 20.
b. **Cross-Validation:** Perform cross-validation for each K value. A common approach is k-fold cross-validation, where your data is divided into k subsets (folds). For each fold, the model is trained on k−1 folds and tested on the remaining fold. This process is repeated k times, each time with a different fold as the test set.
c. **Evaluate Performance:** Calculate the average performance metric (like accuracy, precision, recall, etc.) for each K value across all folds. The goal is to find the K value that yields the best average performance.

2. **Calculating distance:** Distance measurements like Euclidean distance are used to determine the similarity of the target and training data points [16].
$$\mathrm{Dist}(x, X_i) = \sqrt{\sum_{j=1}^{d} \quad (x_j - X_{j_j})^2} \qquad (5)$$
**Syntax:**
def euclidean_distance(p1, p2):
return np.sqrt(np.sum((np.array(p1) - np.array(p2))**2))

3. **Finding Nearest Neighbors:** This involves sorting the entire dataset based on the calculated distances and selecting the top K smallest distances.
**Syntax:**
Consider the samples collected in a set of area
knn = KNeighborsClassifier(n_neighbors='value'
knn.fit(X, y)

**International Journal of Applied Mathematics**
**Volume 38 No. 10s, 2025**
ISSN: 1311-1728 (printed version); ISSN: 1314-8060 (on-line version)

4. **Average for Regression:** Each of the K-nearest neighbors votes for their class, and the majority class wins. Regression: The predicted value is the average (or weighted average) of the target values of the K-nearest neighbors[17].

**Syntax:**

```
knn = KNeighborsRegressor(n_neighbors= 'value'
knn.fit(X, y)
new_point = [['half of the value', 'half of the value']]
prediction = knn.predict(new_point)
```

The **KNeighborsRegressor** is used instead of **KNeighborsClassifier**

C. Random forest

An supervised machine-learning technique that can be employed for issues related to regression and classification is random forest. The conceptual framework of ensemble learning provides the basis for this approach. To solve a complex problem or enhance a model's performance, ensemble learning integrates the outcomes from various classifiers. Randomly selected subsets of the training set are employed to produce a forest of decision trees throughout the Rainforest approach's training phase[15].

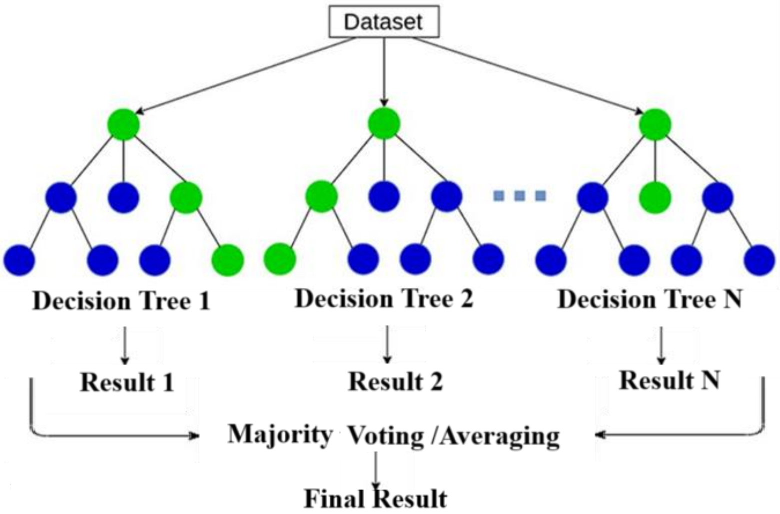



Fig 2: Decision tree using Rainforest approach

Throughout the testing period, the majority vote is employed to gather the forecasts given by every tree in the forest. The Rainforest classifier demands a lot of labeled data to work well, and its accuracy correlates significantly with the number of decision trees [15,18].

KNN is not typically combined with Random Forests directly, they both serve the purpose of classification.

**Syntax:**

```
# Load dataset
data = load_iris()
X_train, X_test, y_train, y_test = train_test_split(data.data, data.target, test_size='value', random_state='value')
# Initialize and fit Random Forest
rf_model=RandomForestClassifier(n_estimators='value2', random_state='value')
rf_model.fit(X_train, y_train)
# Predictions
y_pred = rf_model.predict(X_test)
```

Received: August 11, 2025 

International Journal of Applied Mathematics
Volume 38 No. 10s, 2025
ISSN: 1311-1728 (printed version); ISSN: 1314-8060 (on-line version)

```
# KNN Example
from sklearn.neighbors import KNeighborsClassifier
# Initialize and fit KNN
knn_model = KNeighborsClassifier(n_neighbors='value')
knn_model.fit(X_train, y_train)
OR
knn_model = fitcknn(X_train, Y_train, 'NumNeighbors', k);
# Predictions
y_pred_knn = knn_model.predict(X_test)
% Evaluate accuracy
accuracy_knn = sum(strcmp(pred_Y_knn, Y_test)) / length(Y_test);
```

### D. **Convolutional neural network**

Convolutional neural networks (CNNs) extend findings to unidentified data and automatically learn pertinent characteristics from low-level to high-level patterns without oversight from humans. Several convolutional layers comprise a typical CNN. The convolution layer is in charge of acquiring information. This layer associates the input data employing an assortment of filters in order to create an array of features, which is then fed into the upcoming convolution or pooling layer.

After the convolutional layers, pooling layers are introduced to reduce the spatial dimensions of the feature maps. This process is known as down-sampling. Common down-sampling techniques include first Max Pooling which takes the maximum value from each patch of the feature map.and secondly Average Pooling which takes the average value from each patch of the feature map.Figure 1 Shown the Architecture of an Simple CNN.

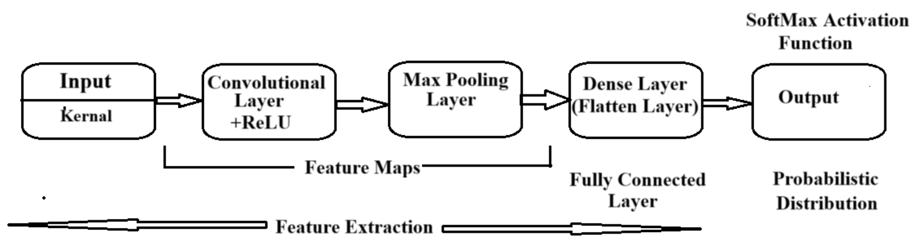



Fig 3: Architecture of an Convolutional Neural Network

Typically, two consecutive convolution layers are encircled by the pooling layer. This layer maintains the most crucial information while decreasing the input feature map's spatial dimensions, which lowers computational complexity [19, 20].

Non-overlapping areas appear when the distance between steps is equal to the filter size when applying the convolution filters. In order to prevent the regions from overlapping, each filter will work on separate, non-overlapping portions of the input image.

### E. Long Short Term Memory

An artificial neural network, more especially a variant of recurrent neural networks (RNNs), is called an LSTM. It is made to work with sequential data and address the issue of long-term memory retention, which regular RNNs have trouble with because to vanishing gradients[21, 22].

Received: August 11, 2025 

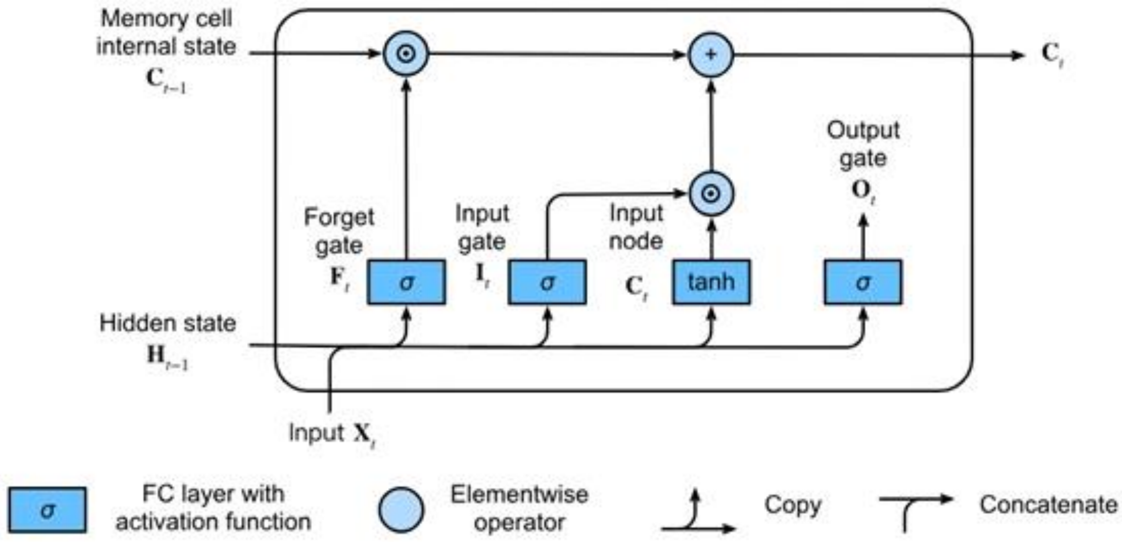



Fig 4. LSTM Cell Architecture

LSTMs are excellent for applications like time-series prediction and natural language processing since they have memory cells that can store information for either short or extended periods of time. LSTM modules can transmit a crucial attribute from the beginning of the input sequence over a long distance,capturing significant long-distance dependencies. One of the main challenges in training the standard RNN, however, is the disappearing gradients issues. Particularly in the early layers, the RNN struggles with small gradient modifications. It consequently becomes unable to remember the data for a long period of time [21]. Internal loops are an outcome of LSTM, which eliminates redundant information and retains only pertinent data.

In LSTM the input gates $I_t$ enables and take the new data into account via $c'_t$ and forget $F_t$ the old cell internal state is given in Eq 6 and retain the data.

$C_{t-1} \in R^{nXh}$ (6)

Using Hadmard producto operator the Eq 6 is modified as follows

$C_t = F_t \odot C_{t-1} + I_t \odot C'_t$ (7)

If the forget gate is always 1 and input gate is always 0 the $c_{t-1}$ will always be constant forever.

## IV. IMPLEMENTATIONS

The Convolutional Neural Network model was trained using Matlab 2021a. Our approach is based on CNN model with a processor 13th Gen Intel(R) Core(TM) i7-1355U 1.70 GHz and 16.0 GB (15.7 GB usable) RAM System type is based on 64-bit operating system, x64-based processor. Our data-set was artificially augmented. This technique allows to avoid the problem of data-set over-fitting. The parameters used for the implementation are shown in Table 1.

Table 1: Parameters used in CNN model

| SL No. | Parameter | Values |
|---|---|---|
| 1 | Frame length | 224 × 224 |

International Journal of Applied Mathematics
Volume 38 No. 10s, 2025
ISSN: 1311-1728 (printed version); ISSN: 1314-8060 (on-line version)

| | | |
|---|---|---|
| 2 | Dimensional vector | 2048 |
| 3 | Sampling | 50 |
| 4 | Initial Learning Rate | 0.0001 |
| 5 | Trained Networl | 6 epochs |

The Linear kernel is a simple kernel function based on the penalty parameter described by the following format

$K(x, y) = x^T y + c$ (6)

### 3.2 Proposed HAR Block Diagram

The proposed "Human Action Recognition Network," is a specialized form of deep learning architecture that was developed specifically for the purpose of recognizing human actions in surveillance footage effectively.

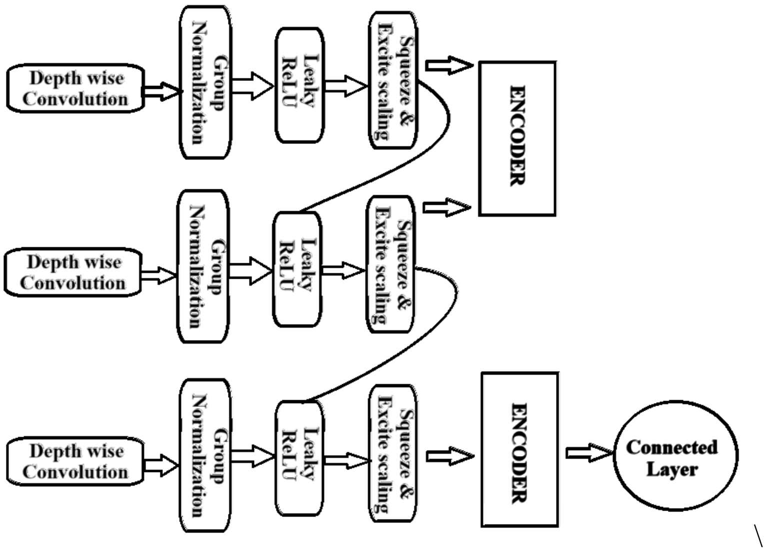



Fig 4: Block Diagram of an HAR Network architecture

The block diagram in Fig 4 shows block diagram of an HAR network architecture. The raw sensor data is received by the input layer. To extract spatial information from the input data, a set of filters is used in the convolutional layer. Each filter accomplishes element-wise summations and multiplications as it traverses across the input.

The Rectified Linear Unit (ReLU) helps introduce non-linearity, which enables the network to learn deeper associations. In this case, the activation function (ReLU) is applied after each convolutional operation. Batch normalization aids in stabilizing and speeding up training. Every neuron in one layer is connected to every other neuron in the subsequent layer via the last dense layers. Squeeze-and-Excitation Mechanisms is used to merge and suppress the dataset's and re-calibrate feature maps to improve HAR Accuracy.

The primary goal is to simultaneously extract spatial and motion cues from video frames in a single stream. HAR's ability to process location and motion data concurrently enables it to achieve greater accuracy in human action detection tasks, rendering it suitable for real-world surveillance applications. The inclusion of the bottleneck concept is critical in the context of the Lightweight HAR network design, driving network efficiency, computational optimization, and streamlined feature extraction.

## V. EXPERIMENTAL RESULTS & DISCUSSION

ISSN: 1311-1728 (printed version); ISSN: 1314-8060 (on-line version)

In this section, experimental results of different classifiers proposed for gender classification, namely KNN,SVM, Decision Tree, CNN and CNN+LSTM, are presented and are compared with the proposed HAR model. The accuracy value obtained on the test data set is used as the evaluation criterion is shown in Table 2.

Table 2: Performance evaluation of classification models.

| Algorithms | KNN | SVM | Decision Tree | CNN | CNN+ LSTM | Proposed HAR model |
|---|---|---|---|---|---|---|
| Accuracy in % | 96.8 | 96.36 | 92.7 | 94 | 95 | 98.5 |

The graphical representation for the performance evaluation for a classical model are shown in Fig 5.

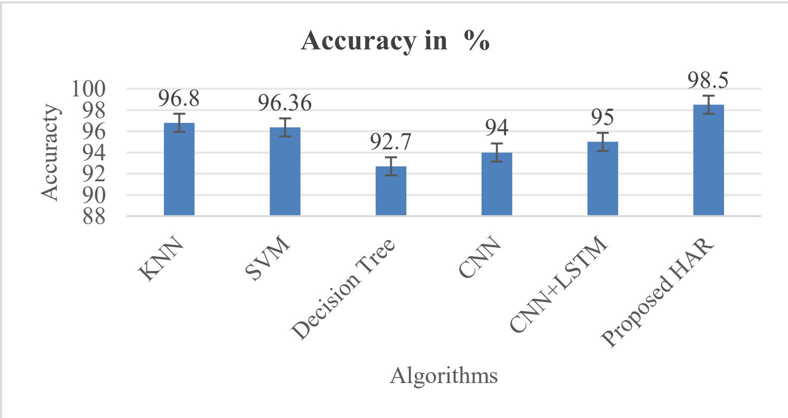
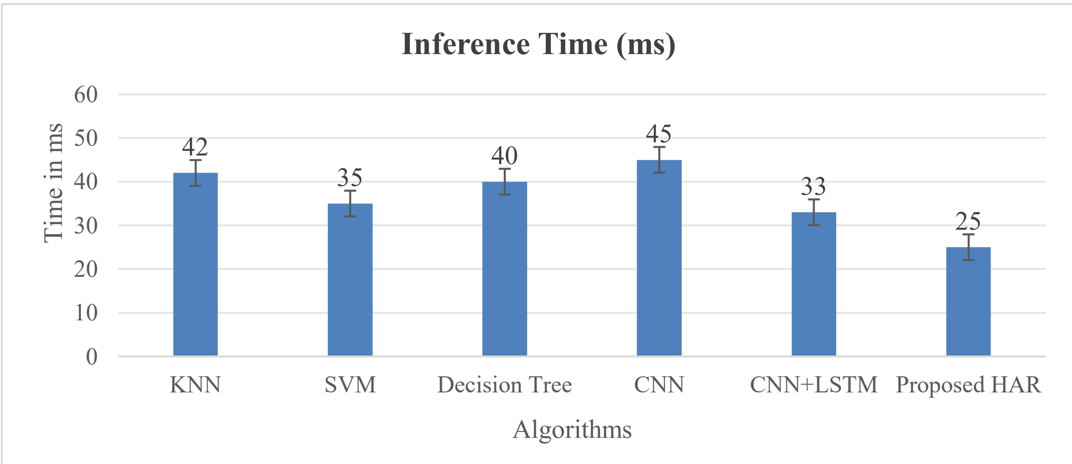



Table 2: Performance evaluation of inference time in (ms).

| Algorithms | KNN | SVM | Decision Tree | CNN | CNN+ LSTM | Proposed HAR model |
|---|---|---|---|---|---|---|
| Inference time (ms) | 42 | 35 | 40 | 45 | 33 | 25 |




To start with the accuracy of Proposed HAR had a discernible decline of 3.5% to 4.5% removal of a component that was based on directed acyclic graphs. The significance of SVM in properly

Received: August 11, 2025

managing the deeply acquired features obtained through proposed HAR model. The inference time of the proposed algorithms was very less as compared to the other existing prototype. To scale the proposed HAR model SVM are more complex to diverse environments. s adaptability to different settings with minimal retraining, possibly through transfer learning. Maintaining high performance in diverse settings may require additional data augmentation techniques and domain adaptation methods.

## VI. CONCLUSIONAND FUTURE SCOPES

In this research paper we presented Human Action Recognition Network Classifier Leveraging SVM-Directed Machine Learning. A pre trained CNN approach based Proposed HAR model has been proposed to extract spatial and temporal features from consecutive frames. In the experimental results, Decision tree algorithms places the least significant of 92% accuracy compare to all other proposed algorithms. Compare to existing algorithms,the proposed HAR model has 98.5% accuracy and inference time was very less compared to other algorithms. But still has a redundancy error for the future scope to improve. For our future works, we propose to use a combination of a genetic algorithm with SVM to optimize the weights of the used CNN model leading to automatically improve the performance.

## REFERENCES

[1]. Basly, H., Ouarda, W., Sayadi, F. E., Ouni, B., & Alimi, A. M. (2020). CNN-SVM learning approach based human activity recognition. In Image and Signal Processing: 9th International Conference, ICISP 2020, Marrakesh, Morocco, June 4–6, 2020, Proceedings 9 (pp. 271-281). Springer International Publishing.

[2]. Dollár, P., Rabaud, V., Cottrell, G., & Belongie, S. (2005, October). Behavior recognition via sparse spatio-temporal features. In 2005 IEEE international workshop on visual surveillance and performance evaluation of tracking and surveillance (pp. 65-72). IEEE.

[3]. Ronao, C. A., & Cho, S. B. (2016). Human activity recognition with smartphone sensors using deep learning neural networks. Expert systems with applications, 59, 235-244.

[4]. Zehra, N., Azeem, S. H., & Farhan, M. (2021, March). Human activity recognition through ensemble learning of multiple convolutional neural networks. In 2021 55th Annual Conference on Information Sciences and Systems (CISS) (pp. 1-5). IEEE.

[5]. Sarabu, A., & Santra, A. K. (2021). Human action recognition in videos using convolution long short-term memory network with spatio-temporal networks. Emerging Science Journal, 5(1), 25-33.

[6]. Rajesh, N., Sarojadevi, H., Nitya, S., Abhishek, B. P., Ananya, B. L., & Akhtar, S. J. (2024, October). Spatio-Temporal Deep Learning Models for Human Activity Recognition-Performance Evaluation and Optimization. In 2024 First International Conference on Software, Systems and Information Technology (SSITCON) (pp. 1-7). IEEE.

[7]. Muralidharan, K., Ramesh, A., Rithvik, G., Prem, S., Reghunaath, A. A., & Gopinath, M. P. (2021). 1D Convolution approach to human activity recognition using sensor data and comparison with machine learning algorithms. International Journal of Cognitive Computing in Engineering, 2, 130-143.

[8]. Li, Y., & Wang, L. (2022). Human activity recognition based on residual network and BiLSTM. Sensors, 22(2), 635.

[9]. Kaya, Y., & Topuz, E. K. (2024). Human activity recognition from multiple sensors data using deep CNNs. Multimedia Tools and Applications, 83(4), 10815-10838.

[10]. Devarakonda, P. G., & Bozic, B. (2022). Particle swarm optimization of convolutional neural networks for human activity prediction. In Optimisation Algorithms and Swarm Intelligence. IntechOpen.

International Journal of Applied Mathematics
Volume 38 No. 10s, 2025
ISSN: 1311-1728 (printed version); ISSN: 1314-8060 (on-line version)

[11]. Ankalaki, S., & Thippeswamy, M. N. (2024). Optimized Convolutional Neural Network Using Hierarchical Particle Swarm Optimization for Sensor Based Human Activity Recognition. SN Computer Science, 5(5), 447.
[12]. Ashwin, M., Jagadeesan, D., Raman Kumar, M., Murugavalli, S., Chaitanya Krishna, A., & Ammisetty, V. (2024). Novel hybrid optimization based adaptive deep convolution neural network approach for human activity recognition system. Multimedia Tools and Applications, 1-25.
[13]. Islam, M. M., Nooruddin, S., Karray, F., & Muhammad, G. (2022). Human activity recognition using tools of convolutional neural networks: A state of the art review, data sets, challenges, and future prospects. Computers in biology and medicine, 149, 106060.
[14]. Jameer, S., & Syed, H. (2023). A DCNN-LSTM based human activity recognition by mobile and wearable sensor networks. Alexandria Engineering Journal, 80, 542-552.
[15]. Boateng, E. Y., Otoo, J., & Abaye, D. A. (2020). Basic tenets of classification algorithms K-nearest-neighbor, support vector machine, random forest and neural network: A review. Journal of Data Analysis and Information Processing, 8(4), 341-357.
[16]. Suguna, N., & Thanushkodi, K. (2010). An improved k-nearest neighbor classification using genetic algorithm. International Journal of Computer Science Issues, 7(2), 18-21.
[17]. Wang, A. X., Chukova, S. S., & Nguyen, B. P. (2023). Ensemble k-nearest neighbors based on centroid displacement. Information Sciences, 629, 313-323.
[18]. Ghasemian, N., & Akhoondzadeh, M. (2018). Introducing two Random Forest based methods for cloud detection in remote sensing images. Advances in Space Research, 62(2), 288-303.
[19]. Kourounis, G., Elmahmudi, A. A., Thomson, B., Hunter, J., Ugail, H., & Wilson, C. (2023). Computer image analysis with artificial intelligence: a practical introduction to convolutional neural networks for medical professionals. Postgraduate medical journal, 99(1178), 1287-1294.
[20]. Abou El-Maged, L. M., AlQaisi, I., Khouqeer, G. A., Elgarayhi, A., & Sallah, M. (2024). Refined Convolutional Neural Networks Automated System for Brain Masses Detection using CT/MRI Diagnostic Scans. Appl. Math, 18(4), 749-759.
[21]. Anandha Praba, R., & Suganthi, L. (2024). HARNet: automatic recognition of human activity from mobile health data using CNN and transfer learning of LSTM with SVM. Automatika: časopis za automatiku, mjerenje, elektroniku, računarstvo i komunikacije, 65(1), 167-178.
[22]. Akinola, D., Oyedemi, A. O., & Ajinaja, M. O. (2024). A Deep Learning-based Hybrid CNN-LSTM Model for Human Activity Recognition. Int. J. Sci. Res. in Computer Science and Engineering Vol, 12(6).