Advancing Human Activity Recognition: A Novel WAECN-BO Approach for Distinguishing Highly Correlated Actions

Ghalan, Mamta
Computer Science Department, National Institute of Technology

Rajesh Kumar Aggarwal
Department of Information Technology, MSIT

https://doi.org/10.5109/7160930

出版情報：Evergreen. 10 (4), pp.2398-2411, 2023-12. 九州大学グリーンテクノロジー研究教育センター
バージョン：

Advancing Human Activity Recognition: A Novel WAECN-BO Approach for Distinguishing Highly Correlated Actions

Mamta Ghalan[1,2*], Rajesh Kumar Aggarwal[1,]

[1]Computer Science Department, National Institute of Technology, Kurukshetra

[2]Department of Information Technology, MSIT, New Delhi

*Author to whom correspondence should be addressed:
E-mail: mamtaghalan@gmail.com

(Received September 30, 2023; Revised December 11, 2023; accepted December 12, 2023).

Abstract: Human actions, when gauged through the lens of Human Activity Recognition (HAR), find numerous applications across healthcare, sports, and security sectors. Nonetheless, the intricacy of HAR becomes apparent when distinguishing akin actions poses a challenge. To tackle this issue, the present article introduces a pioneering method known as the Weighted Average Ensemble of Convolutional Neural Networks with Bayesian Optimization (WAECN-BO), which amalgamates five distinct Convolutional Neural Network (CNN) layer configurations. Notably, this method incorporates a fresh CNN layer designed to enable more intricate abstraction and optimizes its hyperparameters through Bayesian optimization. The evaluation of this method transpires on the UniMiB SHAR Database, a well-recognized benchmark dataset for HAR, focusing on actions with considerable resemblance. The findings reveal a remarkable accuracy rate of 99.98% across the entire dataset, surpassing established state-of-the-art approaches. Additionally, an analysis of the individual contributions made by each CNN layer configuration to the model's performance is conducted. This method, poised to enhance the accuracy of HAR systems across diverse domains, especially those dealing with actions that closely resemble each other, emerges as a promising advancement.

Keywords: Accelerometer-based systems, Heterogeneous Parallel Ensemble Learning, Human Activity Recognition (HAR), Multi-sensor, Convolutional Neural Network

1. Introduction

In our everyday lives, humans engage in a variety of routine and essential tasks, such as cleaning, walking, and regular exercise. These tasks are essentially combinations of fundamental activities like standing, sitting, and bending. Human activity recognition, often referred to as HAR, involves equipping the body with numerous sensors to detect and monitor these daily activities. Recent years have witnessed the proliferation of pervasive sensing, primarily focused on extracting valuable insights from data collected by these ubiquitous sensors[1)]. HAR has garnered significant interest from both industries and academia due to its potential to play a crucial role in a wide range of applications, including video surveillance[2)], healthcare[3)], context-aware computing[4)], multiplayer video games, Internet-connected appliances, and public-use surveillance cameras[5)]. Broadly speaking, there are two main categories of HAR systems currently in use: accelerometer-based and computer vision-based systems. Wearable sensor systems comprise sensors like gyroscopes, accelerometers, magnetometers, among others. These foundational sensors, along with inertia, acoustic, and environmental sensors, contribute to a more comprehensive understanding of an individual's activities[6,7)]. Figure 1.1 provides a comprehensive overview of accelerometer-based systems, spanning from data acquisition to data analysis. However, working with such data presents challenges, including inherent noise due to sensing modalities, issues with missing and erroneous sensor readings, data annotation, and imbalances in class distribution.

Deep learning (DL) methods have attracted tremendous success in HAR with the ability to address the above challenges. A lot of complex and intuitive models exist in the state-of-the-art, and these methods have proven significant potential to overcome all crucial needs in feature engineering[8,9)]. Convolutional Neural Netwoks (CNN) is conventionally used Strategies for Deep Neural Networks, in the HAR study. It analyzes data sequentially and also demonstrated the dual advantage of learning and representing the data effectively and efficiently[10,11)]. The CNN networks work on fixed-size input, in contrast, the other models like Recurrent Neural Networks (RNN) flexibly adjust to the change in the input size. Auto-encoders are adaptive, but they require extensive, expensive metadata annotation[12]. While it comes to temporal characteristics, CNN and LSTM both fall short, while the former has trouble with the latter[13].

In addition to these considerations, numerous unresolved issues persist in this field. Despite progress in sensor fusion techniques, the integration of data from multiple sensors and modalities to achieve accurate activity recognition remains a formidable challenge[14]. While there have been advancements in the development of real-time activity recognition models, the pursuit of low latency and high efficiency without compromising accuracy continues to be a research hurdle[15]. Real-time applications necessitate more efficient architectures and optimization techniques[16–19].

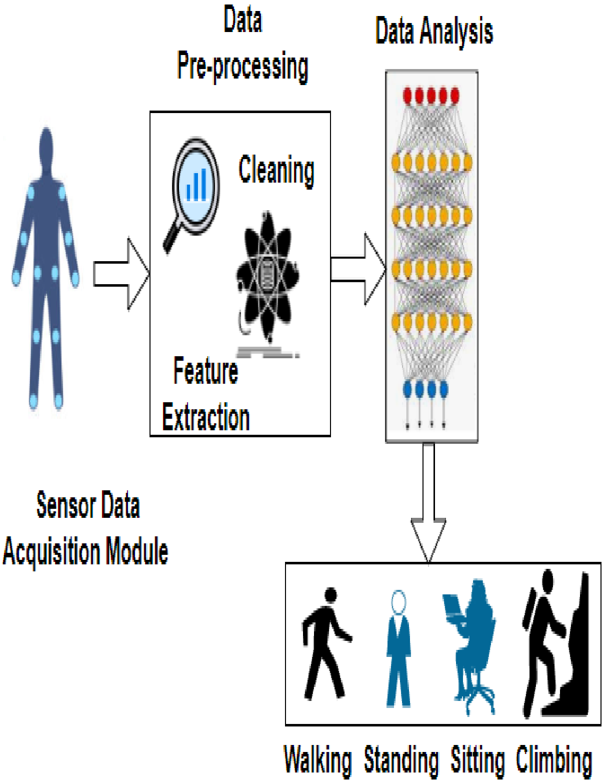


Fig.1.1: A Complete Process: Data Acquisition to Activity.

The domain of activity recognition grapples with various sources of variability in human activities, encompassing inter-subject and intra-subject variations, as well as the challenge of dealing with ambiguous activities that share similar motion patterns[5]. Addressing these issues requires the development of robust models capable of discerning subtle differences and disambiguating activities with resemblances[15]. Despite strides in generalization techniques, the task of creating models with strong generalization capabilities across diverse individuals, environments, and sensor setups remains an ongoing research focus. Transfer learning approaches hold promise, but further exploration is essential to enhance their transferability and adaptability to different domains[20]. Another persistent challenge in human activity recognition revolves around imbalanced datasets. While techniques exist to manage imbalanced and scarce data, addressing this issue remains an active research area[21]. The development of effective methods to handle limited samples for rare activities and enhance recognition performance in such scenarios remains a pressing concern [22].

1.1 Motivation

Detecting highly correlated human activities within human activity recognition (HAR) systems has posed an enduring challenge. Previous research has acknowledged the intricacies of accurately discerning and distinguishing closely related activities. In the realm of video-based human activity detection, the challenge lies in the fact that subtle distinctions between activities can result in misclassifications[23]. Likewise, the imperative of employing feature selection techniques becomes apparent to enhance the recognition of these closely intertwined activities[24]. Surmounting these hurdles necessitates innovative approaches capable of effectively optimizing ensemble learning models employed in HAR systems. These primary challenges stem from overlapping features and data variations, with existing methods struggling to capture the nuanced disparities between these activities. Reducing data has implications for human activity recognition via wearable technology[25], with similar[26] ramifications observed in activity transition mining. The presence of inherent uncertainties and variations within highly correlated human activities presents a substantial impediment to accurate recognition. Existing deep learning models, such as CNN-GRU architectures[27], may grapple with encapsulating the complexities and subtle differences among activities. To address this limitation, ensemble learning methods have displayed promise in bolstering recognition accuracy. Nonetheless, enhancing the efficiency of HAR structures can be achieved by incorporating Bayesian optimization into the entire ensemble learning framework. As previously mentioned, prevailing accelerometer-based HAR systems confront challenges encompassing noise, missing and erroneous sensor readings, data annotation, and imbalanced class distributions. Deep learning techniques, including CNN and RNN, exhibit promise but also exhibit constraints in effectively capturing temporal and spatial features. To mitigate the limitations of individual models and elevate the performance of HAR systems, ensemble learning has surfaced as a promising avenue. Ensemble learning amalgamates multiple machine learning algorithms, capitalizing on their strengths and offsetting their weaknesses[28]. In the context of HAR, ensemble methods offer distinct advantages over standalone models. By extracting features from multiple models and amalgamating them, ensemble methods attain a more holistic comprehension of activity data, culminating in enhanced recognition accuracy and resilience.Moreover, ensemble learning furnishes a more steadfast solution for managing noisy and heterogeneous characteristic data, inherent to HAR. The diverse array of models employed in ensembling can capture various facets of the data, rendering them effective in addressing both linear and non-linear characteristics. Additionally, ensemble methods aid in mitigating the intricacies and uncertainties associated with deep learning models[29]. By amalgamating the predictive outcomes of multiple models, ensemble learning diminishes the risk of overfitting and augments generalization performance. It offers a more dependable and robust solution by amalgamating the individual strengths of each model. The contributions of present paper are following:

1. The highly correlated human actions are detected with the proposed Weighted Average Ensemble of Convolutional Neural Networks with Bayesian Optimization (WAECN-BO).

2.The proposed model implements heterogeneous Stacked Ensemble Methods with the Bagging technique and cross-validation for considerable enhancements in the orchestrated results. The different models included in the Stacked Ensemble model consisting of three CNN model layers, they provide very high accuracy with automatic detection of important features without any or minimal human invention. Also, CNN has the ability to share the weights. This reduces the training and computational complexity of the designed model.

3.With the integration along with the Bayesian Optimisation, enables adaptive learning and dynamic adjustment of the ensemble weights based on the characteristics of the highly correlated activities are facilitated.

The paper continues with a brief overview of the associated research in HAR ensemble technique in Section 2, a discussion of the suggested structure and the intended architecture in Section 3, an examination of the laboratory environment and the different settings for assessment and a comparison of the model's efficacy compared to the current state of the field in Section 4, and finally, a summary and recommendations for further research in Section 5.

2. Related Work

The state-of-the-art related to HAR is extensive. There are various ways in which the existing rich literature can be securitized. In this work, the author mainly focuses on the study of various ensemble-based models. These can be mainly classified into two board categories: ML-based and DL-based ensembles. The brief literature survey on the various methods in ML-based is tabulated in Table. I while for DL are listed in Table. II. In the case of ML-based ensembles, simple supervised methods are combined to get the model. As in[30], the authors have combined ten classifiers to design Adaboost. The work proposed in[14], has basically designed a weighted majority voting ensemble method of the classifier. The accuracy achieved 90% for only the accelerometer data for 10 users. The individual training of these classifiers is done for small datasets and also only for signal data. This limits its universal applicability and generality. Similar work is proposed in[31], cascade ensemble learning concept is developed combining Extremely Gradient Boosting Trees (XGBoost), Random Forest, and Extremely Randomized Trees. Most of these conventional methods heavily depend on hand-crafted feature mining followed by various data mining techniques for classification. The work done in[32] has collected data from three sources wireless sensor data mining, human activity recognition utilizing cellphones, and Kaggle. Followed by the pre-processing steps. The authors have employed different statistical methods like short-time Fourier transform, and t-Distributed Stochastic Neighbor Embedding techniques. Then, handcrafted method is employed to get the critical features using the hybrid coyote Jaya optimization (HCJO). The three classifiers-a support vector machine (SVM), a deep neural network (DNN), and a fuzzy classifier-are used in this meta-heuristic-based ensemble learning method for classification.

The next category of the methods employed is the DL-based ensemble. The scope of methods creeps in to remove the feature extraction manually and try to automate the system for better efficiency and analysis. Howsoever, the DL methods mainly employ the CNN or LSTM structure of the design. As the training and time complexity of the architecture adopted is already too much, authors refrain from themselves with small and concentrated sensor datasets. As in[33] only [32] activities are sorted to be analyzed while in[34] the sample test is only 8 subjects. Through this study, one can analyze that there are many methods developed previously on machine learning-based ensembles. But, the ensemble used was mostly the majority voting system. This voting scheme does not assume prior knowledge about the problem at hand or the classifiers used.

While in the case DL-based approach, the model is tested on small datasets, also the training and testing time required is high. Hyperparameter tuning is also an unanswered issue. With this, the author proposes to build a model lighter on the hyper-parameters and an ensemble model that can provide appreciable results in a small dataset without any bias.

Table I: ML-Based Ensembles

Year	Dataset	Model	Feature Extraction Domain (Time /Frequency)	Ensemble Approach	Accuracy	Remark
2018[30]	REALDISP	Random Forests , Support Vector Machines , Naive Bayes, K- Nearest Neighbors , ANN , C4.5 Decision Tree	-	Adaboost	99.98%	To achieve better accuracy large dataset is required.
2018[14]	Smartphone Collected	DT, Linear Regression (LR),Multilayer Perceptron (MLP), K-NN	T/F	Weighted Majority Voting	90%	A limited set of activities are tested only the accelerometer and gyroscope of a smartphone only by 10 subjects.
2019[35]	MHELTH/ USCHAD	LR,SVM,MLP,K-NN,RF ,NB	T/F	Majority Voting	94% /86%	The approach combine the basic ML algroithms.
2019[31]	-	XGBoost, RDT,Softmax Regression	T	Majority Voting	96.04%	Only triaxial accelerometer and gyroscope data are tested.
2021[32]	Smartphone Collected, Wireless sensor Kaggle	SVM, DL, Fuzzy System	T	wireless body area network	90%, 91% 92%	It helps to extract significant and robust features.

Table. II: DL-Based Ensembles

Year	Dataset	Model	Feature Extraction Domain (Time /Frequency)	Ensemble Approach	Accuracy	Remark
2021[33]	MHEALTH, Self collected	CNN-LSTM, CNN-GRU	T	Hybrid deep Learning Model	99.3%	7 walking data using IMU sensor for 50 subjects Needs for layers in the ensemble for better accuracy.
2022[36]	Self Collected Data	ResNet18, ResNet50, ResNet101	T	ReliefF	99.92%	Millions parameters should be optimized
2022[21]	UCI-HAR UCI-WIDSM OPPORTUNITY	CNN, GRU	T/F	DNN	81.7%	Difficult in real time application
2022[34]	Photoplethysmography (PPG) Electrocardiogram (ECG)	Resnet50V2, MobileNetV2, Xception	T	DNN	97%	8 subjects
2022[37]	UCI HAR	Fuzzified deep convolutional neural network (FDCNN)	T	Fuzzy λ_{max}	97.98%	Mini-batch size hyper parameter selection
2022[13]	HASC UCI Smartphone WISDM ,UniMiB SHAR PAMAP2	VGG	T	Group Ensemble (GE)	97%	Accuracy depends on hyperparameter tuning
2022[38]	Self MHEALTH USC-HADWHARF OPPORTUNITY	Fussed CNN-LSTM ResNet-50	T	Fuzzy-Ensemble Approach	96.52%	Refined fine tune is required of Hyper-parameters

3. Proposed Model

3.1 System Model

Deep learning has revolutionized various domains, including activity recognition. The CNN_DenseNet (Activities Recognition Dense Network) previously proposed by the author addresses the challenge of accurately distinguishing similar activities and optimally tuning hyperparameters through Bayesian optimization and SMOTE for data augmentation. In this article, an enhanced version of the CNN_DenseNet model[22)] using heterogeneous Stacked Ensemble Methods with the Bagging technique and cross-validation is presented. Our approach leverages the strengths of multiple layered CNN models to achieve improved accuracy and automatic detection of important features, while minimizing human intervention. Specifically, the focus of this work is on the Weighted Model Averaging form of Ensemble convolutional network (WAECN), which allows for efficient integration of diverse models by assigning weights based on their performance. Weighted

bagging technique in the proposed ensemble network reduces the sensitivity of the models to the stochasticity of the training process. The proposed model for c class recognition employing a weighted ensemble of the four models is depicted as a schematic representation in Figure 3.1. The input signal is fed to the respective interconnected CNN layers and then their weighted average is accordingly computed and fed as an input to the fully connected layer. Each CNN classifier initialized with seed value i and utilizing action bank features (AB) as input is denoted as CNN_i. Additionally, CNN_i' stands for the CNN classification algorithm trained with input characteristics from the supplementary action bank (AB'). CNN_i and CNN_i' are considered as part of the weighted ensemble of models with the same initial weights in order to calculate the effectiveness of the combined model CNN_i', that streamlines the explanation and evaluation of the suggested model.

Then, the combined models' results are used to assess the ensemble model's efficacy. To make predictions, CNN classifiers undergo training to produce binary decoded outputs, where a_1 represents the predicted class index and 0 represents all other classes. By using the highest possible value function as the weighted ensemble function, the model selects the outputs f_j that have a confidence value close to 1 (i.e., accurate predictions). For the purpose of calculating the expected class label of the ensemble model, the results of the fusion functions $f_1, f_2, \dots, f_c$ are considered as binary decoded results.

By assigning higher weights to more accurate or reliable models, the ensemble can achieve better overall performance. The strategy is to train multiple models with the same architecture and dataset, but with different number of layers, initial weights or random seeds, and then averages their different abstraction features. This reduces the sensitivity of the models to the stochasticity of the training process.

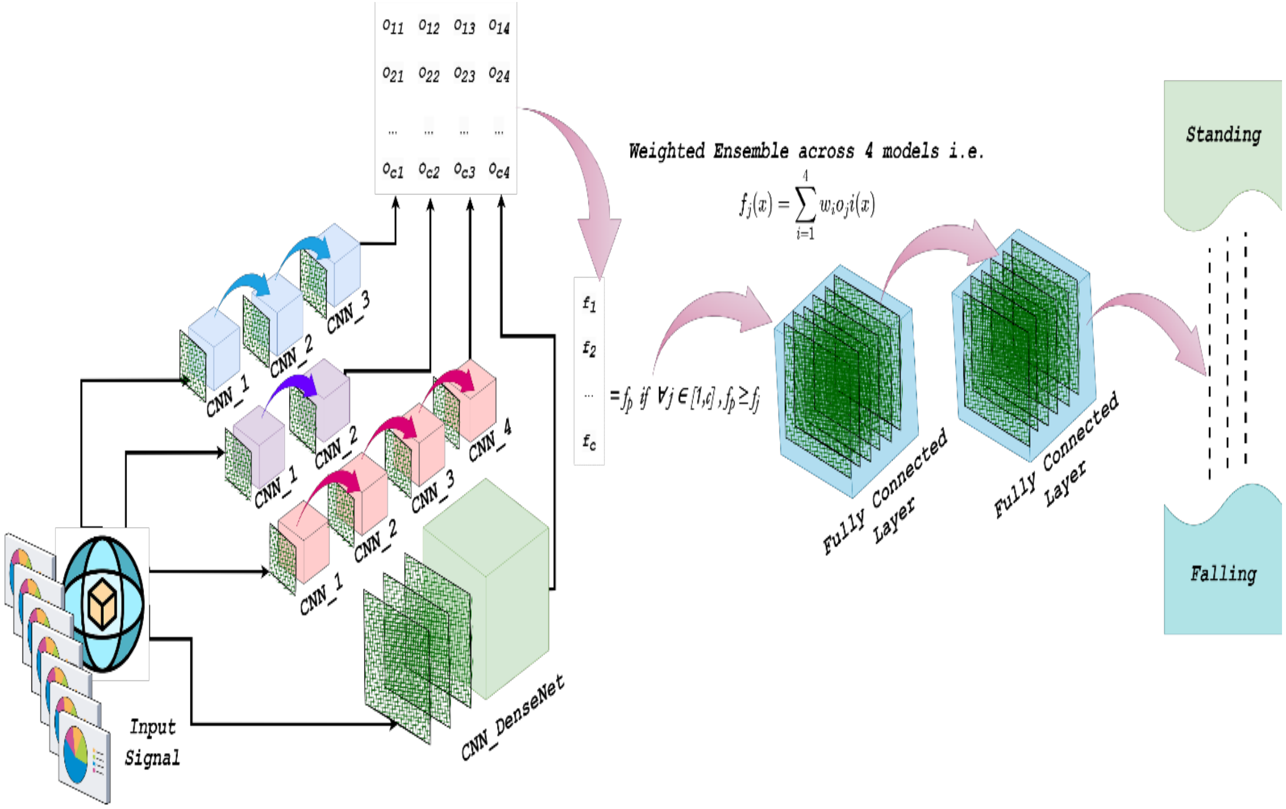

o_{11} o_{12} o_{13} o_{14}
o_{21} o_{22} o_{23} o_{24}
... ...
o_{c1} o_{c2} o_{c3} o_{c4}
Weighted Ensemble across 4 models i.e.
$f_j(x) = \sum_{i=1}^{4} w_i o_j i(x)$
f_1
f_2
...
f_c
$= f_p$ if $\forall j \in [1,c], f_p \geq f_j$
CNN_1
CNN_2
CNN_3
CNN_1
CNN_2
CNN_1
CNN_2
CNN_3
CNN_4
CNN_DenseNet
Fully Connected Layer
Fully Connected Layer
Standing
Falling
Input Signal

Fig. 3.1: CNN Dense Network Architecture Complete with Cross-Validated Stacked Ensemble

To mathematically formulate the ensembling of deep learning networks, the following definitions are needed in accordance to figure 3.2:

The base models: These are the individual convolutional neural network models that are trained on the data and make predictions. They can be denoted as $o_i(x)$, where i is the index of the model and x is the input.

Network model: Network can be perceived as the assembly of the base models forming multiple layers each generating its own convoluted prediction $o_i(x)$ which are then ensembled according to the combination function.

The ensemble approach is a more complex system that takes into account the forecasts of individual base models. whereas x represents a certain input, it can be written as $f_j(x)$.

The combination function: This is the function that determines methodology to combine the predictions of the base models. It can be denoted as $f_j(x)$, where this is a function that maps from $\mathbb{R}^n \; to \; \mathbb{R}^n$ (for regression) or $\{0, 1\}$ (for classification). This work has suggested the bagging as the weighted average to ensemble the outcome from the different convolutional networks.

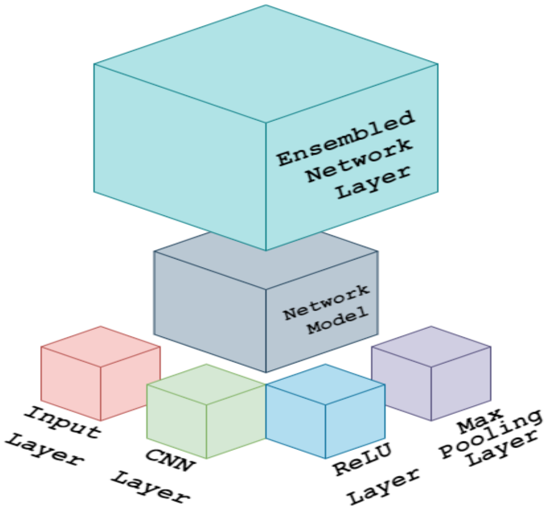


Fig. 3.2: Architectural basics of the WAECN-BO

To formulate the Weighted Model Averaging ensemble (WAE), a heterogeneous Stacked Ensemble has been considered with 4 CNN model layers. Each layer represents a distinct base model denoted as $o_i(x)$, where i is the index of the model and x is the input. The ensemble modelled prediction, denoted as $f_j(x)$, combines the predictions of the base models using weighted averaging. Depending on the type of ensembling method, the combination function can be different. The proposed Weighted Model Averaging can be mathematically expressed as: For bagging, $f_j(x)$can be a weighted mean expressed as:

$$f_j(x) \;=\; \sum_{\{i=1\}}^{4} w_i \, o_i(x) \qquad (3.1)$$

where w_i are non-zero, additive weights. The ensemble has four different baseline models. The values of w_i show how much emphasis is placed on the forecasting abilities for every base model. However, the final optimal prediction op is calculated as:

$$op = f_p \; if \; \forall j \; \in [1, c], f_p \geq f_j \qquad (3.2)$$

To ensemble the features from the last layer and classify closely related human activities in deep learning, the presented approach involves training each base model individually. Each base model is trained using a specific architecture and hyperparameters, allowing them to capture distinct representations of the input data. After training the base models, the features are extracted from the last layer of each model. These features capture high-level representations that are informative for classifying human activities. By ensembling the features from the last layer, the strengths of each base model are effectively leveraged to improve the overall classification performance for closely related human activities.

Further, a crucial aspect that contributes to the unique performance of Weighted Model Averaging form of Ensemble convolutional network with Bayesian Optimization (WAECN-BO) is the utilization of powerful Bayesian optimization technique for fine-tuning the hyperparameters. This technique helps to efficiently search the vast hyperparameter space by combining prior knowledge and observes results. By leveraging Bayesian optimization, the presented model identifies the most promising hyperparameter configurations, leading to improved performance and robustness. This optimization process is particularly effective for the four stacked grouped CNN layers in our model, allowing them to extract and leverage meaningful features from the input data. This method not solely improves the offered model's efficacy but additionally cuts down on tedious experimentation and failure procedures. The integration of Bayesian optimization in previously published CNN_DenseNet [31] architecture enables to achievement superior accuracy and generalization in activity recognition tasks.

3.2 Proposed Ensembled Network: WAECN-BO

A key contribution of the presented work is the introduction of distinct CNN layers that enable more complex abstraction of features from the input data. This approach enhances the model's ability to capture intricate patterns and representations, leading to more accurate action recognition. Bayesian optimization shows an important place in optimizing the hyperparameters of the respective layer, allowing them to adapt and learn from the data effectively. By combining the strengths of ensemble learning, multiple CNN layer arrangements, and Bayesian optimization, our model achieves exceptional accuracy and robustness in human action classification tasks.

In order to decrease the size of extracted features and increase their stability, the convolutional layers for feature extraction are supported by the ReLU layers and are then followed by max-pooling layers. After the extracted features have been integrated by the fully connected layers, the network model is completed by adding an output layer that averages the weighted models to convey a categorical distribution of the different operations.

3.2.1 Architecture of the Ensembled model

To understand the structure of the involved convolutional neural network (CNN) model, it comprises of several layers that are arranged sequentially to form the network. Each layer performs specific operations on the input data to extract features and transform them. To further reflect upon the details of this approach, Fig 3.3 illustrates the intricacies of each distinct network model involved in the process.

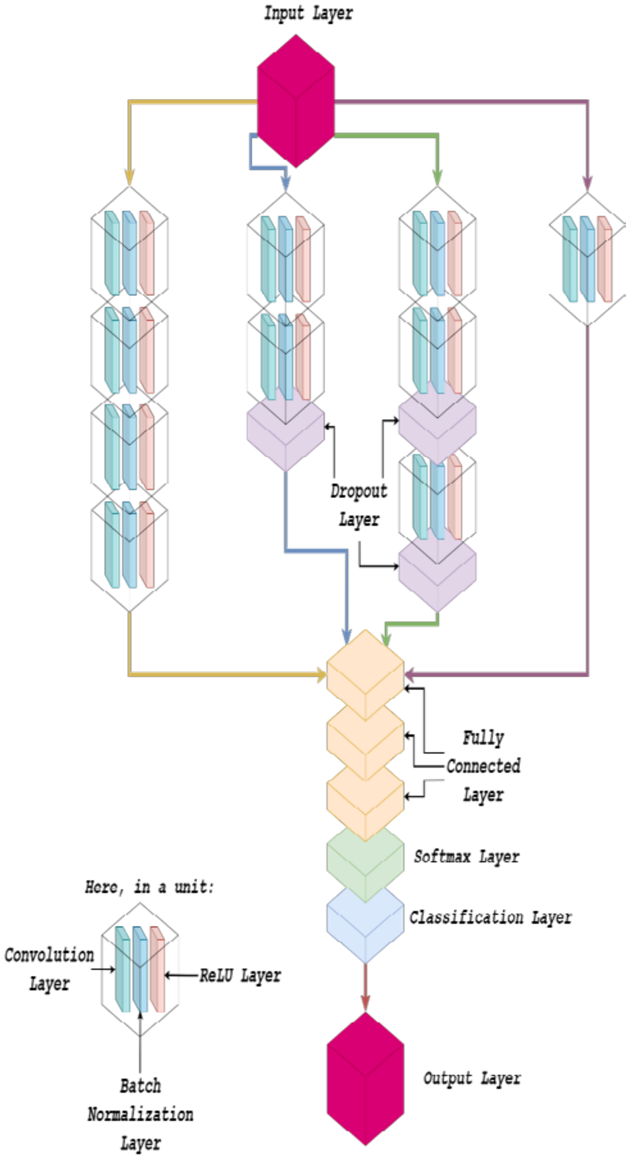


Fig. 3.3: Intricacies of the respective Network models.

As shown in the figure, each model in the presented approach utilizes an `*imageInputLayer*` to define the input dimensions of the network as equal number of rows and columns through a single channel. Then a `*convolution2dLayer*` convolves the input with a set of learned filters to extract local features. Following this, a `*batchNormalizationLayer*` normalizes the activations of the previous layer, which helps in improving the training process and generalization. There's a `*reluLayer*` which applies the Rectified Linear Unit (ReLU) activation function element-wise to introduce non-linearity to the network. The `*dropoutLayer*` randomly sets a fraction of the input elements to zero during training. It helps in preventing overfitting by introducing some level of regularization. Finally, the `*maxPooling2dLayer*` reduces the spatial dimensions of the previous layer's output, which helps in minimizing the computational complexity and providing translation invariance.

In this method, CNN serves as a foundational layer in all involved network models, executing an inner product functioning among local filters and the data from input to produce the appropriate dimension of the convolutional output matrix while employing input-space regional filters for extraction of features. The result will be composed of $(N - m + 1)$ units if the input is a layer with N units and the next layer is a convolutional layer with a filter of size m. After breaking down every network model into its component parts, the convolutional layer's output features are as follows:

$$x_i^{l,j} = \boldsymbol{g}(\textstyle\sum_{a=1}^{m} w_a^j x_{i+a-1}^{l-1,j} + b_j) \qquad (3.3)$$

where $x_{i(l,j)}$ is the result of the jth feature map on the ith unit of the lth convolutional layer, $w_{a(j)}$ is the convolutional kernel matrix, and $b(j)$ is the bias of the jth convolutional feature map. The resultant feature map from the layer before it is convoluted with the weights, and then the sum of that and the bias is calculated. The activation function $g(\dots)$ is then used to execute the non linear mapping.As an activation function, $relu()$ is used in the proposed model. The initial hidden section of the first local filter, for instance, can be computed as for the first network model depicted in Fig 3.1.

$$x_1^{1,1} = relu(w_1^1 x_1^{0,1} + w_2^1 x_2^{0,1} + w_3^1 x_3^{0,1} + b_1 \qquad (3.4)$$

The outcomes for the upper layers can be derived by extrapolating these computations. The activity recognition issue is also tackled by the proposed model via the Max-pooling tactics. After features have been identified in the convolutional layer of the algorithm, the max-pooling layer can shrink the size of the extracted features and strengthen certain characteristics with no ruining the information's internal connections. In CNN, the activation function of the max-pooling layer is denoted as

$$x_i^{l,j} = \overset{r}{\max}_{i,j=1}(x_{i,j}) \qquad (3.5)$$

In this case, the dimension of the pooling kernel, r, determines the location of the final output, $x_{i(l,j)}$ Features retrieved in the convolutional layer are partitioned into subsets in the max-poling layer. In each breakdown, the highest possible values are displayed.

The aforementioned layers are typically repeated several times throughout the network architecture, each time with tweaked parameters and an increased amount of convolutional filters. Overlaying layers follow, with $'fullyConnectedLayer1$ connecting all the neurons from the preceding layer to the current layer in the same way that a conventional neural network would. The $'fullyConnectedLayer2'$ is the network's final output layer, which comes after it. By employing the activation function (softmax), the $'softmaxLayer'$ converts the result of the preceding layer into a distribution of probability of the classes. Last but not least, the $'classificationLayer'$ makes the ultimate classification using the softmax layer's calculated probabilities. The accuracy of its predictions is evaluated in relation to the true labels via the cross-entropy loss function.

```
Algorithm 1: Stacking with 5-fold cross-validation
Input: Training Data : $T_D = A_i \in (A_{AF17}, A_{AF8}, A_{AF9})$, K=5, $\mathbb{W} = \{\mathbb{W}_1, \dots \dots \mathbb{W}_4\}$
Output: An ensemble Classifier $H_i$ and Ensembled Output $O$
Repeat for each classifier $H_i \in (1,5)$
Step 1:    Train the classifier $H_i$ for the adopted cross-validation
Randomly split $T_D$ into 5 equal-size subsets: $T_D = \{T_1, \dots \dots T_5\}$
For i=1: $T_D$
Step 1.1 Learn the classifier $H_i$
Step 1.2 Construct the data set for training the classifier $H_{i+1}$
End For
Step 2: $H_i = \sum H_{T_D}$
The weighted average value of the model for each action   $O = \sum_{i=1}^{4} \mathbb{W} H_i$
```

The complete steps of the designed ensemble model are presented in Algorithm.1, which is aimed at creating an ensemble classifier and generating an ensembled output. This is achieved by multiplying the weights WH_i by the corresponding classifier outputs and summing the results over i from 1 to 4. The data UniMiB SHAR Database is divided into three basic bifurcations. Each classifier H_i is trained and tested on the different chunks of the data (T_D). The final detection of the activity is based on the weighted average output of each classifier. This algorithm leverages input training data and a set of classifiers to achieve this goal through a systematic process. It starts by considering the input training data, denoted as T_D, which consists of three datasets. Additionally, K is set to 5, representing the desired number of folds for cross-validation. A set of weights, $W = \{W_1, \dots \dots W_4\}$, is provided to account for the importance of individual classifiers. For each classifier H_i in the range of 1 to 5, the training data T_D is randomly split into 5 equal-size subsets as $T_D = \{T_1, \dots \dots T_5\}$.For each subset T_i in T_D, the classifier H_i is trained using the data from subset T_i. The data set for training the next classifier, H_{i+1}, is constructed by combining the predictions of the previous classifiers H_1 to H_i. The ensemble classifier H_i is obtained by summing the individual classifiers' outputs for each action, represented as H_{T_D}. The predicted final action is the one with the high O, for the given set of sensor inputs to the network. The method designed is tested and verified.

3.2.2 Integration of Bayesian Optimization WAECN-BO

To direct the hunt for optimal arrangements, Bayesian Optimization uses a probabilistic surrogate model (typically a Gaussian Process) in conjunction with an acquisition function. The process involves iteratively sampling configurations, evaluating their performance, updating the surrogate model, and selecting the next configuration based on the acquisition function. In our model, a set of hyperparameters can be denoted as H. The objective is to determine the value of h^*, the optimum hyperparameter, by which the objective function, $o_i(h)$, can be minimized. Bayesian Optimization formulates this as a surrogate model, $p(y|X,H)$, where y represents the observed performance and X denotes the hyperparameter configurations. The surrogate model captures the uncertainty in the relationship between hyperparameters and performance by modeling the joint distribution $p(y, X \mid H)$. It is implemented using a Gaussian Process (GP), which provides a flexible and principled framework for Bayesian inference.

The key idea is to use Bayesian Optimization to automatically search for the optimal hyperparameter configuration for each individual model in the ensemble. The ensemble weight w_i is determined based on the performance of each model during the Bayesian Optimization process. The higher the performance, the higher the weight assigned to the corresponding model.

The acquisition function guides the search towards configurations that are likely to improve the ensemble's overall performance. The mathematical formulation of the weighted model averaging process using Bayesian Optimization can be represented in equation 3.5 as follows:

$$w_i = \rho * exp(-\beta * y_i) \quad (3.5)$$

where y_i is the performance of model i, and ρ and β are hyperparameters that control the weighting scheme. The value of ρ ensures that the weights sum up to one, while β controls the sensitivity of the weights to the performance. The optimization process for BO in WAECN-BO involves initializing the surrogate model $p(y|X,H)$ using an initial set of hyperparameter configurations. A new configuration h is generated by selecting the upcoming point to considering using the acquisition function. For training and evaluating the model with hyperparameters h, the surrogate model has to be consistently updated with the new observations (h,y) and the weights w_i of the ensemble models using the performance y_i. By iteratively optimizing the ensemble weights and hyperparameters, the BO in WAECN-BO framework effectively explores the hyperparameter space and learns an ensemble of models that collectively achieves superior performance.

4. Results and Discussion

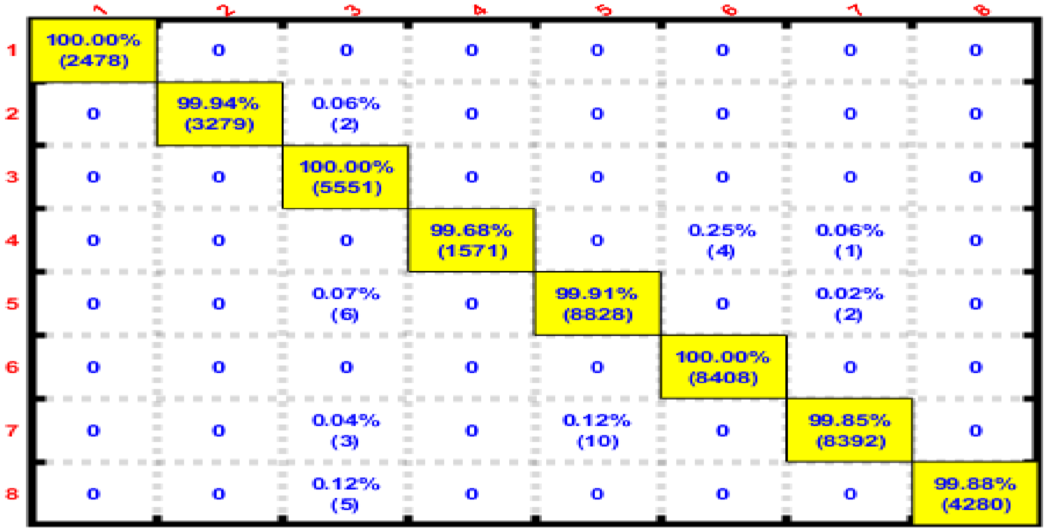
UniMiB SHAR Database: There are several datasets available to the public, especially for fall detection techniques. This work primarily focuses on sensor-based datasets from wearable sensors. Five sensors were employed, consisting two through phones and 3 wearable sensors that were attached to various body areas. The UniMiB SHAR dataset comprises of movement traces from both falls and typical ADL activity. Public access to the raw data, which is free of gravitational constant influence and noise (such as EMG noise), is provided[20]. Because it only includes accelerator data, the UniMiB SHAR dataset was gathered from a real-time environment with little power consumption. The lack of null values in the UniMiB SHAR dataset balances the data. This database contains test data from the 19 objects' everyday activities. For data collected, only accelerometer observations were employed from data collection via Smartphone and sensor nodesse's 30 individuals do 11771 activities. Everyone ranged in age from 18 to 60. It includes 17 actions, nine of which are normal and eight of which are important (like falling).

Simulation Background: The proposed and other similar clustering algorithms are simulated in Windows-8 (64-bit) using MATLAB version R2012a on an Intel core(TM) i7 processor 2.40 GHz Central Processing Unit (CPU) with 8 Gigabyte (GB) of inbuilt Random Access Memory (RAM).

Discussion: Since the data is a 1-D signal by definition, it must be converted before being sent to the network. The network receives an input in the form of a 1×453×1. There are 96 batch normalizing layers in a convolution layer. The classifier is assessed using the 5-fold cross-validation method. The three types of training that are used are daily activities only (AF17), all activity courses (AF8), and fall activities (AF8) (AF9) The correlation plot between the AF8 is plotted before the 5-fold and after application on the model designed in Figure 4.1 and Figure 4.2 respectively. The classification accuracy achieved for cases like falling forward and backwards being too identical is still nearly 100% distinguished.

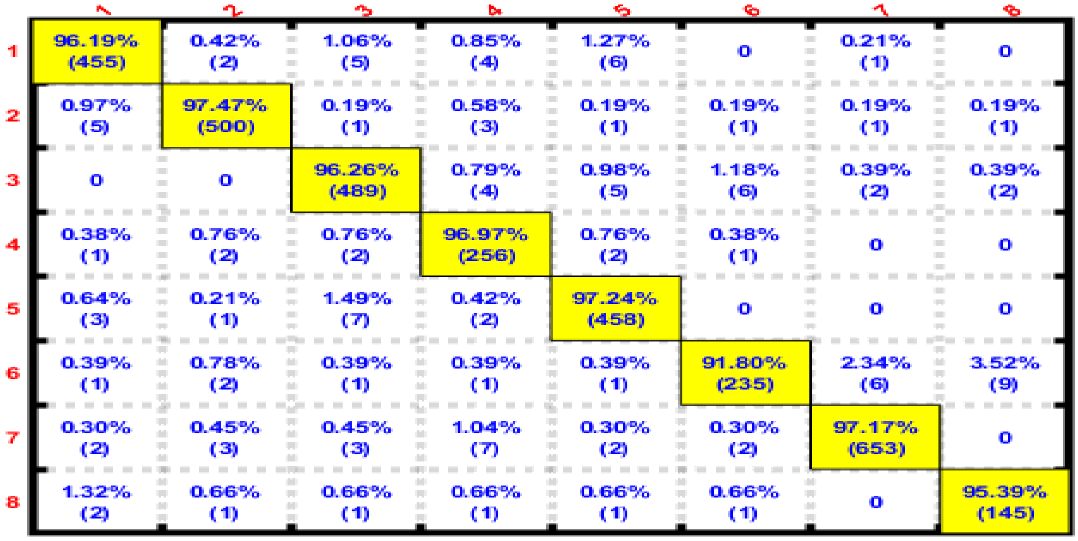
Table. III Board Classification of the various schemes in the UniMiB SHAR dataset.

Action	Class	Proportion (%)	Action	Class	Proportion (%)
Falling	Forward	4.49	**Standing**	Up from sitting	1.30
	Left	4.54		Up from Lying	1.83
	Right	4.34		Down	1.70
Backwards	Falling	4.47	**Common**	Walking	14.77
	Sitting Chair	3.69		Running	16.86
	Protection Strategies	4.11		Jumping	6.34
	Hitting Obstacle	5.62	**Going**	Up	7.82
Syncope		4.36		Down	11.25
			Lying and Standing	Down	2.51



	1	2	3	4	5	6	7	8
1	100.00% (2478)	0	0	0	0	0	0	0
2	0	99.94% (3279)	0.06% (2)	0	0	0	0	0
3	0	0	100.00% (5551)	0	0	0	0	0
4	0	0	0	99.68% (1571)	0	0.25% (4)	0.06% (1)	0
5	0	0	0.07% (6)	0	99.91% (8828)	0	0.02% (2)	0
6	0	0	0	0	0	100.00% (8408)	0	0
7	0	0	0.04% (3)	0	0.12% (10)	0	99.85% (8392)	0
8	0	0	0.12% (5)	0	0	0	0	99.88% (4280)

Fig. 4.1: Correlation plot for basic 8 activities for the Ensemble Model. (It includes Falling, Backwards and Syncope)



	1	2	3	4	5	6	7	8
1	96.19% (455)	0.42% (2)	1.06% (5)	0.85% (4)	1.27% (6)	0	0.21% (1)	0
2	0.97% (5)	97.47% (500)	0.19% (1)	0.58% (3)	0.19% (1)	0.19% (1)	0.19% (1)	0.19% (1)
3	0	0	96.26% (489)	0.79% (4)	0.98% (5)	1.18% (6)	0.39% (2)	0.39% (2)
4	0.38% (1)	0.76% (2)	0.76% (2)	96.97% (256)	0.76% (2)	0.38% (1)	0	0
5	0.64% (3)	0.21% (1)	1.49% (7)	0.42% (2)	97.24% (458)	0	0	0
6	0.39% (1)	0.78% (2)	0.39% (1)	0.39% (1)	0.39% (1)	91.80% (235)	2.34% (6)	3.52% (9)
7	0.30% (2)	0.45% (3)	0.45% (3)	1.04% (7)	0.30% (2)	0.30% (2)	97.17% (653)	0
8	1.32% (2)	0.66% (1)	0.66% (1)	0.66% (1)	0.66% (1)	0.66% (1)	0	95.39% (145)

Fig. 4.2: Correlation plot for basic 8 activities for the alone CH4. (It includes Falling, Backwards and Syncope)

Table.IV Overall results of an individual action set.

HAR SUB SET ACTIONS	Casilari, E.et.al SVM (24)	Casilari, E.et.al KNN (24)	Casilari, E.et.al Random forest (24)	Frédéric Li.et .al (25)	Designed Method Ensemble	
					With (CN4)	
A1: AF-17	78.75	82.86	81.48	93.4	97.98	99.98
A2: AF-9	83.1	87.77	88.41	98.40	99.98	99.99
A3: AF-8	75.63	79.1	78.6	89.9	97.2	98.2

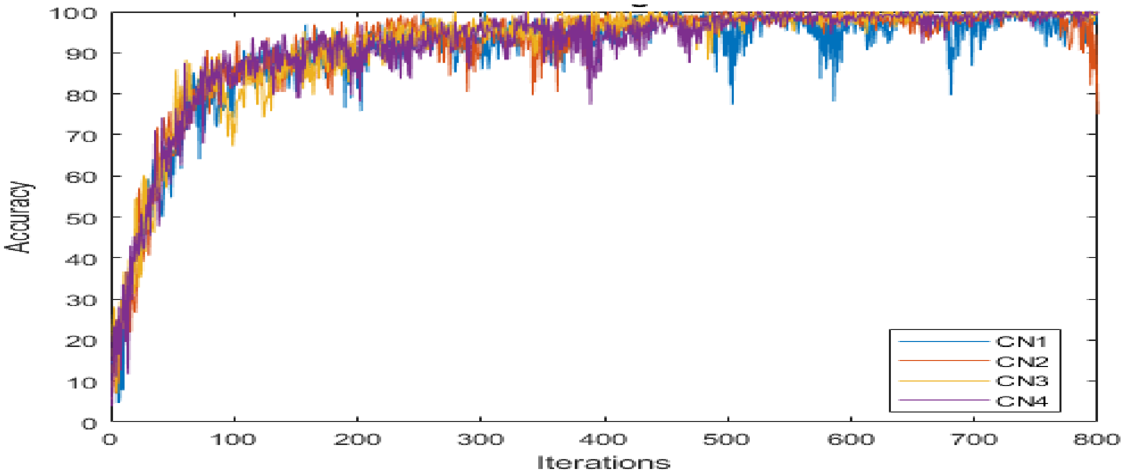


Figure.4.3: The accuracy of each model incorporated in the Stacked Ensemble.

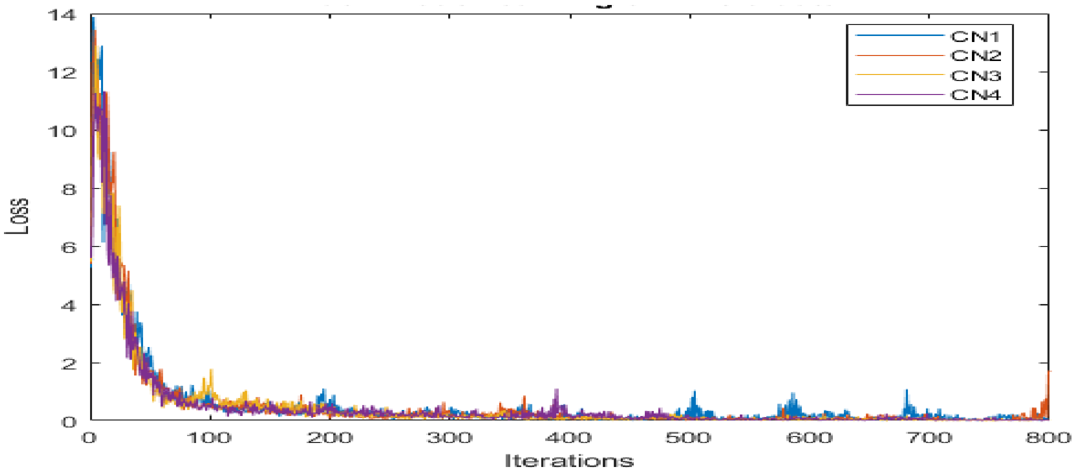


Figure.4.4: The loss of each model incorporated in the Ensemble (Stacked).

Tabel.IV compares the total accuracy of every sequence of actions to the current state of the art techniques. The model designed is reported with the highest accuracy of individual models and the ensemble also. The understanding of techniques like KNN, SVM and the Random Forest (RF) is too basic for understanding the hidden pattern or structure of similarity. On the other hand, the work proposed in[39] is an amalgamation of various different advanced techniques that have boosted classification accuracy. The proposed method with ensemble design outperforms with appreciable accuracy. Accuarcy of cascade model along with loss is also plotted for the AF8 activities. In Figure 4.3, the model CN1 which is just a simple cascade of three blocks has a varying accuracy. While the model CN4 had continuously changing accuracy from 200-500 iterations and then settles. Thus, the parameters take time to tune themselves. The same thing also gets validated from the loss in Figure 4.4, where CN1 has more spike in the loss at 700 iterations also.

Table V: Proposed Ensemble Comparison with other methods using UniMIB SHAR Dataset

	Wan et al. (26)	Challa et al. (27)	Dua et al. (28)	Ankalaki et al. (29)	Tan et al. (30)	Ensemble (proposed)
Acc(%)	92.71	96.37	96.20	96.86	96.7	99.98
Pre(%)	93.21	-	-	-	96.8	97.86
R(%)	92.82	-	-	-	96.8	96.83
F1-Score(%)	92.93	96.31	96.19	-	96.8	97.84

Traditional DL models for activity recognation served as the basis for the suggested architecture. Similarly, CNN, LSTM, BiLSTM, multilinear progression, and SVM were all used in Wan et al work[15] to assess classification accuracy using the UNiMIB dataset. Author claims maximum overall accuracy (92.71%) comes from CNN model. Author's accuracy employing layered structure architecture was 7.6 percentage points higher than that attained with CNN[15)].

A hybrid CNN-BiLSTM deep learning model has been suggested[16)] due to the efficacy of CNN in obtaining features and choosing the features offered by the forgetting gate within BiLSTM when using backward and forward sequencing. In the proposed model, After every pair of convolutional neural network (CNN) layers is a dropout layer and a maximum pool layer. Using CNN layering, the BiLSTM network learned the traits of each of these three sub-branches. The ensemble architecture is made up of three different CNN forks with varying filter lengths. A total accuracy of 96.37% was obtained on the same dataset, which is 3.6% lower than what the model predicted. The linked studies in[27)] used several deep learning network architectures to address the underlying research problems. Each of the three recurrent routes in an arranged convolutional neural network (CNN) receives an additional two Gated Recurrent Unit (GRU) layers. Once the dropout layer has been applied, the features can be combined. The system that is suggested is 3.86 percentage points more precise than the existing one.

A stack of deep learning layers of varying depths is used to categorize the signal properties as proposed in[20)]. The paper used a conventional CNN method to categorize the signals as either stationary or active. Next, created and trained a modified ML in CNN (EML-CNN) algorithm that can account for both stationary and moving objects. Using an ensemble learning strategy, Tan et al.[21)] propose a CNN ensemble with a GRU (Gated Recurrent Unit) layered model that incorporates time-frequency features that have not been processed at the fully linked layer. The UNiMIB SHAR dataset is broken down into six different types of actions: sitting, standing, lying down, ascending and descending stairs, and walking. The accuracy that Tan et al. achieved for the six everyday tasks that they categorised was 96.8%, while the network that they proposed in the study achieved an accuracy of 99.98% across the board.

Table.V provides a comparison of the present state-of-the-art methods. Recall, precision, and F1 score are used to assess the dataset. The positive activity forecast with the highest degree of precision is called recall, while the positive prediction with the largest degree of recall is called precision. Due to the inequitable distribution of the data classes, the accuracy parameter may drive up classification costs. The F1-score, which demonstrates an optimal trade-off between recall and precision, is a useful metric here. Each of these 17 activities' findings can be found in their respective rows of the UniMiB SHAR results table. Because of the large number of imbalanced class samples in the UniMiB data, the ensemble has been praised for its high level of accuracy.

5. Conclusion

This paper presents an innovative approach, the Weighted Average Ensemble of Convolutional Neural Networks with Bayesian Optimization (WAECN-BO), to address the challenging task of distinguishing highly correlated actions in Human Activity Recognition (HAR). The incorporation of five distinct CNN layer configurations, including a novel CNN layer, enhances the model's ability to discern closely related actions through more intricate abstraction. The utilization of Bayesian optimization further refines the ensemble model, fostering adaptive learning and dynamic adjustment of weights, thereby increasing versatility in handling highly correlated activities. By adopting a heterogeneous Stacked Ensemble with the Bagging technique and cross-validation, the proposed approach significantly improves the model's performance. The integration of three CNN model layers within the Stacked Ensemble facilitates automatic feature detection with minimal human intervention. Additionally, the weight-sharing capability of CNN reduces training complexity, enhancing computational efficiency. The experimental evaluation on the UniMiB SHAR Dataset demonstrates the effectiveness of the proposed WAECN-BO method. The ensemble model, featuring four networks in parallel connection, achieves a remarkable accuracy rate of 99.98% across the entire dataset. This surpasses current best practices, highlighting the potential of WAECN-BO to enhance the accuracy of HAR systems, especially in scenarios involving similar actions. In essence, the WAECN-BO method introduces a robust and adaptive approach to HAR, showcasing superior accuracy and robust performance. The successful experimental outcomes underscore it's potential for real-world applications, particularly in domains where distinguishing highly correlated actions is paramount.

References

1) J. Wang, Y. Chen, S. Hao, X. Peng, and L. Hu, “Deep learning for sensor-based activity recognition: a survey,” *Pattern Recognit Lett*, **119** 3–11 (2019).

2) L. Shao, L. Ji, Y. Liu, and J. Zhang, “Human action segmentation and recognition via motion and shape analysis,” *Pattern Recognit Lett*, **33** (*4*) 438–445 (2012).

3) V. Osmani, S. Balasubramaniam, and D. Botvich, “Human activity recognition in pervasive health-care: supporting efficient remote collaboration,” *J Netw Comput Appl*, **31** (*4*) 628–655 (2008).

4) X. Wang, D. Rosenblum, and Y. Wang, “Context-aware mobile music recommendation for daily activities,” in: Proc 20th ACM Int Conf Multimed, 2012: pp. 99–108.

5) J.-L. Reyes-Ortiz, L. Oneto, A. Samà, X. Parra, and

D. Anguita, "Transition-aware human activity recognition using smartphones," *Neurocomputing*, **171** 754–767 (2016).

6) S.K. Yadav, K. Tiwari, H.M. Pandey, and S.A. Akbar, "A review of multimodal human activity recognition with special emphasis on classification, applications, challenges and future directions," *Knowledge-Based Syst*, **223** 106970 (2021).

7) I.G.D. Nugraha, and D. Kosasih, "Evaluation of computer engineering practicum based-on virtual reality application," *Evergreen*, **9(1)** 156–162 (2022). https://doi.org/10.5109/4774233

8) B. Zhou, J. Yang, and Q. Li, "Smartphone-based activity recognition for indoor localization using a convolutional neural network," *Sensors*, **19** (*3*) 621 (2019).

9) P. Panwar, P. Roshan, R. Singh, M. Rai, A.R. Mishra, and S.S. Chauhan, "DDNet-a deep learning approach to detect driver distraction and drowsiness," (2022).

10) S. Münzner, P. Schmidt, A. Reiss, M. Hanselmann, R. Stiefelhagen, and R. Dürichen, "CNN-based sensor fusion techniques for multimodal human activity recognition," in: Proc 2017 ACM Int Symp Wearable Comput, 2017: pp. 158–165.

11) H.K. Chaudhary, K. Saraswat, H. Yadav, H. Puri, A.R. Mishra, and S.S. Chauhan, "A real time dynamic approach for management of vehicle generated traffic," (2023).

12) S. Zhang, Y. Li, S. Zhang, F. Shahabi, S. Xia, Y. Deng, and N. Alshurafa, "Deep learning in human activity recognition with wearable sensors: a review on advances," *Sensors*, **22** (*4*) 1476 (2022).

13) S. Ghosal, M. Sarkar, and R. Sarkar, "NoFED-net: nonlinear fuzzy ensemble of deep neural networks for human activity recognition," *IEEE Internet Things J*, **9** (*18*) 17526–17535 (2022).

14) J. Saha, C. Chowdhury, I. Roy Chowdhury, S. Biswas, and N. Aslam, "An ensemble of condition based classifiers for device independent detailed human activity recognition using smartphones," *Information*, **9** (*4*) 94 (2018).

15) S. Wan, L. Qi, X. Xu, C. Tong, and Z. Gu, "Deep learning models for real-time human activity recognition with smartphones," *Mob Networks Appl*, **25** 743–755 (2020).

16) S.K. Challa, A. Kumar, and V.B. Semwal, "A multibranch cnn-bilstm model for human activity recognition using wearable sensor data," *Vis Comput*, **38** (*12*) 4095–4109 (2022).

17) D. Singh, and A. Singh, "Role of building automation technology in creating a smart and sustainable built environment," (2023).

18) V. Aggarwal, A. Ranjan, S. Shaurya, and S.K. Garg, "To determine the futures pricing of metal commodities using deep learning," (2023).

19) M.S. Sumathi, V. Jain, G.K. Kumar, Z.Z. Khan, and others, "Using artificial intelligence (ai) and internet of things (iot) for improving network security by hybrid cryptography approach," (2023).

20) S. Ankalaki, and M.N. Thippeswamy, "Static and dynamic human activity detection using multi cnn-elm approach," in: Emerg Res Comput Information, Commun Appl ERCICA 2020, Vol 1, 2022: pp. 207–218.

21) T.-H. Tan, J.-Y. Wu, S.-H. Liu, and M. Gochoo, "Human activity recognition using an ensemble learning algorithm with smartphone sensor data," *Electronics*, **11** (*3*) 322 (2022).

22) & R.K.A. Mamta Ghalan, "Multifold Classification for Human Action Recognition," in: 2021 IEEE Bombay Sect Signat Conf, 2021: pp. 1–6.

23) S.O.& P.D. V.Japtap P.Gawande Y. Rathore A. Rao, "Video-Based Human Activity Detection," in: 2nd Int Conf Intell Technol, 2022: pp. 1–5.

24) M.M. Manca, B. Pes, and D. Riboni, "Exploiting feature selection in human activity recognition: methodological insights and empirical results using mobile sensor data," *IEEE Access*, **10** 64043–64058 (2022).

25) H. Nourani, E. Shihab, and O. Sarbishei, "The impact of data reduction on wearable-based human activity recognition," in: 2019 IEEE Int Conf Pervasive Comput Commun Work (PerCom Work, 2019: pp. 89–94.

26) H. El Hafyani, K. Zeitouni, Y. Taher, and M. Abboud, "Leveraging change point detection for activity transition mining in the context of environmental crowdsensing," in: Actes La Conférence BDA, 2020: p. 64.

27) N. Dua, S.N. Singh, and V.B. Semwal, "Multi-input cnn-gru based human activity recognition using wearable sensors," *Computing*, **103** 1461–1478 (2021).

28) X. Dong, Z. Yu, W. Cao, Y. Shi, and Q. Ma, "A survey on ensemble learning," *Front Comput Sci*, **14** 241–258 (2020).

29) M. Sewell, "Ensemble learning," *RN*, **11** (*02*) 1–34 (2008).

30) A. Subasi, D.H. Dammas, R.D. Alghamdi, R.A. Makawi, E.A. Albiety, T. Brahimi, and A. Sarirete, "Sensor based human activity recognition using adaboost ensemble classifier," *Procedia Comput Sci*, **140** 104–111 (2018).

31) S. Xu, Q. Tang, L. Jin, and Z. Pan, "A cascade ensemble learning model for human activity recognition with smartphones," *Sensors*, **19** (*10*) 2307 (2019).

32) J. Boga, and others, "Human activity recognition in wban using ensemble model," *Int J Pervasive Comput Commun*, (*ahead-of-print*) (2022).

33) V.B. Semwal, A. Gupta, and P. Lalwani, "An optimized hybrid deep learning model using ensemble learning approach for human walking

activities recognition," *J Supercomput*, **77** (*11*) 12256–12279 (2021).

34) O.R.A. Almanifi, I.M. Khairuddin, M.A.M. Razman, R.M. Musa, and A.P.P.A. Majeed, "Human activity recognition based on wrist ppg via the ensemble method," *ICT Express*, **8** (*4*) 513–517 (2022).

35) H.D. Nguyen, K.P. Tran, X. Zeng, L. Koehl, and G. Tartare, "Wearable sensor data based human activity recognition using machine learning: a new approach," *ArXiv Prepr ArXiv190503809*, (2019).

36) T. Tuncer, F. Ertam, S. Dogan, E. Aydemir, and P. Pławiak, "Ensemble residual network-based gender and activity recognition method with signals," *J Supercomput*, **76** 2119–2138 (2020).

37) V. Gomathi, S. Kalaiselvi, and others, "Sensor-based human activity recognition using fuzzified deep cnn architecture with $λ$max method," *Sens Rev*, **42** (*2*) 250–262 (2022).

38) E. Casilari, J.A. Santoyo-Ramón, and J.M. Cano-Garc¥'¥ia, "Umafall: a multisensor dataset for the research on automatic fall detection," *Procedia Comput Sci*, **110** 32–39 (2017).

39) F. Li, K. Shirahama, M.A. Nisar, L. Köping, and M. Grzegorzek, "Comparison of feature learning methods for human activity recognition using wearable sensors," *Sensors*, **18** (*2*) 679 (2018).