Ingénierie des Systèmes d'Information
Vol. 30, No. 6, June, 2025, pp. 1621-1628
Journal homepage: http://iieta.org/journals/isi

Enhancing Human Motion Recognition Through Multi-Sensor Data Fusion and Deep Learning for Smart Decision Support Systems

Samatha R. Swamy[1*], K. S. Nandini Prasad[1], R Sunitha[2]

[1] Department of Information Science and Engineering, Dayananda Sagar Academy of Technology and Management, Bangalore 560082, India
[2] Department of Artificial Intelligence & Machine Learning, B N M Institute of Technology, Bangalore 560070, India

Corresponding Author Email: rswamysamatha12@gmail.com

https://doi.org/10.18280/isi.300620

Received: 13 May 2025
Revised: 14 June 2025
Accepted: 20 June 2025
Available online: 30 June 2025

Keywords:
Bayesian networks, convolutional human activity recognition, internet of things, Long Short-Term Memory, multi-sensor data fusion

ABSTRACT

Human motion recognition with high accuracy is important for many applications ranging from healthcare systems and sports analysis to smart environmental setups. However, traditional methods can be sensitive to sensor noise, data variability, and real-time processing requirements. This research introduces a new multi-sensor data fusion framework integrated with deep learning to improve human movement recognition for smart decision support systems. This paper presents an innovative Bayesian Convolutional Neural Network with a Long Short-Term Memory (BCNN-LSTM) framework for temporal information with data from different sensors. Multi-level fusion including feature level and decision level proposes a contrasting approach for combining sensor data that increases robustness and generalizability. The experimental results indicate that our proposed BCNN-LSTM model provides better performance than the traditional approaches, with 8% to 10% improvements in classification accuracy, compared with the Support Vector Machine, LSTM, CNN models, and Bayesian LSTM. Future enhancement includes AI integration for enhanced motion recognition precision and generalized.

1. INTRODUCTION

Data fusion: Multi-source or multi-sensor data fusion [1] is defined as the process of integrating data from multiple sensors in case this data is uncertain and/or ambiguous. This is similar to how living organisms use different senses to act at their surroundings, reflect on the data and make informed decisions. This has attracted considerable attention in the research community because of the growing demand for intelligent systems. Multi-sensor data fusion aims to create a more accurate, robust, and complete interpretation of data by merging input from several sensors together as opposed to relying on one sensor type alone. Most of the recent published works in this domain are application-driven, exploring areas like environmental monitoring [2] and object detection [3]. Nonetheless, there still remains no generally accepted algorithmic structure in the field, which is of utmost importance for enabling multi-sensor data fusion to become a recognised scientific discipline.

Modern life is heavily impacted by technology, and people reliant on smart devices. Big data is produced increasingly due to the development of intelligent technologies [4, 5]. A major case of this data is Human Motion Analysis (HMA) which is significant to real-time applications. It has become increasingly complex to analyze human motion with the surge in connected sensors and ubiquitous computing. The tracking of user actions is especially useful in the area of assistive technology communities where, by having a deeper understanding of user behavior can enhance interaction with smart environments and offer improved user experience.

HMA has become an essential research direction with applications in many fields like health-care [5], fitness tracking, surveillance, smart living and sports [6]. Human Activity Recognition (HAR) from video has already proven useful in multiple areas [7, 8]. Recent technological innovations have greatly improved sensor devices, which has turned these tools into essential devices in human motion sciences related to healthcare, behavioral analysis, rehabilitation, and assisted living. Monitor sedentary behaviour with a sensor-based approach: Prolonged sitting, especially in a workplace or academic setting, has been linked to a variety of health risks, including obesity, diabetes, and cardiovascular diseases. Moreover, human action recognition (HAR) has found its way into the gaming industry, with systems such as Microsoft Kinect showcasing the impact of motion recognition on gaming experiences [9]. These high-tech developments also show the transformative potential of HAR beyond health and lifestyle to gaming and other interactive scenarios. In recent years, wearable sensors, including accelerometers, gyroscopes, and magnetometers, have gained popularity in HAR research since they are convenient, effective, and low-cost [10].

However, the absence of a unified theoretical basis and standardized algorithms still poses a key difficulty for scientists in the area of multi-source data fusion. A general fusion algorithm with a wide range application is important in

promoting this research direction, allowing for integration of sensor fusion and model for sensor interaction and awareness. In many real-world applications, data fusion is used to collect multi-sensor information in order to increase the efficiency of decision making and enhance the overall system. In an effort to fill this gap, this work provides a hybrid Bayesian CNN-LSTM model to improve multi-sensor data fusion in the context of human activity recognition. This research aims to contribute towards the progression of multi-sensor data fusion as well as providing a solid system framework for human activity recognition and leading to enhancement in several applications.

Research Contributions are discussed below:

- Proposed a Bayesian model of a CNN-LSTM framework by attempting to maximize the amount of sensors recorded as features such as accelerometers, gyros, IMUs, cameras, etc.
- Advanced techniques in deep learning are studied such as CNNs, RNNs, and transformers that increase the accuracy and reliability of human motion analysis.
- Examine techniques of fusion such as feature-level and decision-level fusion in order to identify the best analysis of human activities.

The paper is organized in the following way: Section 2 describes goals, scope, and some concepts of multi-sensor data fusion. Section 3 analyses current decision-making frameworks based on multi-sensor data fusion, particularly for human activity recognition and CNN-LSTM model representation. Section 4 presents the experimental results, and a comparison of the proposed model with three existing models that applied multi-sensor data fusion. Lastly, Section 5 concludes the research work and summarizes potential future research pathways in this area.

2. RELATED WORK

Recently, multi-sensor data fusion techniques have been investigated within the purpose of wearable computing mainly in the contexts of health monitoring health care, activity trackin, and ambient intelligence. Although works [1-3] highlighted component level fusion elements to improve recognition accuracy, they neglect important motion recognition issues such as; sensor noise, misalignment mapping multiple sensors, and heterogeneous sampling rates. Our work improves on existing deep learning sensors-based studies by providing temporal model and probabilistic reasoning in a comprehensive deep learning approach aimed at HAR.

Sensor data is one of the most critical factors that determine the performance of a human motion analysis system. IMUs, gyroscopes, and accelerometers are among the most frequently used wearable sensors as they are convenient and accurate. Apart from body-worn sensors, ambient sensors such as cameras, depth sensors, and millimeter-wave radars have been incorporated in motion recognition systems to improve spatial reasoning. For instance, Yadav et al. [11] presented a multi-modal human activity recognition (HAR) system that integrates vision and inertial sensors to enhance the accuracy of recognition by utilizing complementary data across diverse sources. Similarly, Chen et al. [12] highlighted the need for data fusion from sensors to mitigate the problems of single-modality systems, including occlusion in cameras or drift in IMUs.

There are three levels of fusion approaches are data-level, feature-level, and decision-level fusion for multi-sensors fusion. Data-Level Fusion: Raw data from different sensors are fused before any preprocessing and feature extraction. May retain information from different sensors but can be computationally intense. This fusion level structure includes three sub-fusion levels: Feature-level: application of feature extracted from each sensor when they are fused together to build a complex representation and then consumed by the deep learning model; It eliminates a lot of dimensions while retaining relevant information. Decision-Level Fusion: In this approach, each sensor's data is processed independently by the classifiers, and the outputs are subsequently combined using ensemble methods or voting strategies. Wang et al. an LSTM-based deep learning framework combined with CNN is used for wearable sensor-based HAR, reaching a better accuracy than that achieved through feature-level fusion [13]. Ahmed et al. [14] have introduce a model of selection of features to improve smartphone-based HAR, in which the most important information is fused. However, the studies revoke emphasis to afford esteem to the proper balance while considering computational constraints and application requirements.

Device-bound approaches consist of attaching sensors to objects to recognise activities based on interactions with these objects. This method has the same limitations of wearable sensors in that the user has to make contact with some particular marked object. On the other hand, device-free technologies (i.e. environment-based or dense sensing sensors) do not need to have people wear or carry any devices. Then, those sensors such as RFID, Wi-Fi, ZigBee, microwave sensors are placed in the environment for their motion and behaviors detection and analysis. Additionally, Zhang et al. [15] used XGBoost to classify five indoor activities, achieving 84.19% accuracy.

Ahmed et al. [16] proposed an enhanced human activity recognition (HAR) model that leverages smartphone sensor data combined with a hybrid feature selection method. The study integrates filter and wrapper-based feature selection techniques to optimize classification accuracy while reducing computational overhead. Their approach effectively enhances HAR performance across multiple activity types. However, the study primarily focuses on traditional machine learning techniques and does not extensively explore deep learning models, which limits its applicability to more complex activity recognition tasks. Additionally, its performance is highly dependent on the quality and positioning of the sensors used for data collection.

Verma et al. [17] conducted a systematic review of artificial intelligence (AI) applications in marketing, focusing on how machine learning and deep learning models improve customer segmentation, sentiment analysis, and personalized marketing strategies. The paper provides valuable insights into the impact of AI-driven analytics on decision-making in marketing. While the study offers a broad overview of AI advancements in the field, it lacks practical implementation details and empirical evaluations, making it more theoretical than application-oriented. Additionally, its focus remains on business perspectives rather than the technical intricacies of AI model development.

Wang et al. [18] examined HAR with wearable sensors using a hybrid deep learning model that ties CNNs with LSTMs together. This benefits the model because it can extract spatial and temporal features effectively, resulting in recognition accuracy. The model's ability to recognize precise

and continuous complex human physical motions makes the study useful for real-time application. However, large-scale training data and the considerable computational efforts can present challenges for implementation on edge devices with limited resources.

Chen et al. [19] provided a comprehensive survey of deep learning techniques for sensor-based HAR by analyzing various architectures including CNNs, LSTMs, and attention-based models. The paper discusses critical challenges in HAR, including data heterogeneity, sensor variability, and computational constraints. While the survey offers a well-structured literature review, it lacks empirical validation or experimental results, making it less practical for researchers seeking implementation guidance. Additionally, it does not delve into dataset-specific performance benchmarks, which would have strengthened its comparative analysis.

Wang et al. [20] proposed an attention-based CNN model for HAR, specifically for weakly labeled sensor data. With the attention mechanism, the model can learn to weigh the relevant sensor inputs dynamically and improve classification accuracy. The authors demonstrate the capability of attention mechanism to improve model interpretability and to handle data that is sparsely labeled. The authors note that although the outcome has benefits in interpretability, this model has the drawbacks of needing excessive hyperparameter tuning and computationally demanding, which can limit using the model in a real-time setting.

Sansano et al. [21] examined different deep neural network architectures for HAR, evaluating CNNs, recurrent neural networks (RNNs), and hybrid models. Their study gives an extensive comparison of the performance of models across datasets, thereby highlighting the positive and negative aspects of each approach. The study is particularly valuable in understanding the balance between accuracy and computational efficiency in HAR applications. However, the study does not account for practical implementation constraints including latency and power consumption, which may be critical for real world applications.

Xia et al. [22] proposed a hybrid model incorporating LSTMs and CNNs for HAR by using CNNs for feature extraction and LSTMs for temporal modeling of sequences. Their architecture effectively recognizes both periodic and non-periodic activities, achieving high classification accuracy on benchmark datasets. The study's major strength lies in its ability to integrate spatial and temporal feature learning seamlessly. However, the model's reliance on large labeled datasets and its high computational cost present challenges for deployment on low-power devices such as smartwatches and IoT-based wearables.

These papers collectively highlight the advancements in deep learning-driven HAR, with a particular focus on sensor data processing, hybrid model architectures, and practical challenges in real-time implementation. Each study offers unique contributions, but common limitations such as computational complexity, reliance on large datasets, and challenges in real-world deployment remain areas for future research.

3. PROPOSED METHOD

A Bayesian CNN-LSTM framework usually integrates a Convolutional Neural Network (CNN) and a Long Short-Term Memory (LSTM) network within a Bayesian framework to capture both spatial and temporal dependencies in data while accounting for uncertainty in the parameters of the model. Bayesian approaches are advantageous in scenarios where uncertainty estimation is crucial, such as in medical diagnosis, financial predictions, or any other application where model confidence is essential.

This paper proposes a Bayesian CNN-LSTM architecture that combines uncertainty estimation and deep temporal learning. Build Bayesian convolutional layers by using Monte Carlo Dropout to approximate Bayesian inference, during both training and inference. Model stick with a gaussian prior distribution over the weights and sample the outputs of the model multiple times so could obtain a distribution over predictions. The LSTM layers of our model are also regularized using dropout, capturing long-range dependencies whilst propagating uncertainty. This hybrid model balances accuracy and model uncertainty, allowing it to be useful in real-world sensor environments.

Building a Bayesian CNN-LSTM model for Human Activity Recognition (HAR) is an interesting and challenging project. HAR involves classifying activities performed by humans based on sensor data, often collected from accelerometers, gyroscopes, and other sensors. A Bayesian approach can be beneficial in providing uncertainty estimates and robustness in recognizing various activities. Here's a general outline for such a project: Then the model design is discussed in the following subsections:

3.1 CNN architecture

A Convolutional Neural Network (CNN) architecture for spatial feature extraction in the context of HAR can be described mathematically by specifying the operations in each layer. There are various layers in CNN model, proposed framework is constructed with 7 layers includes input layer, convolutional layer, activation layer, pooling layer, flatten layer, dense layer, and output layers. The components of a CNN architecture using mathematical notations are as follows:

Input Layer: Consider the input data with batch size, number of channels, height, and width.

Convolutional Layers: Suppose a single convolutional layer with K filters of size F and stride S. The convolutional operation can be expressed as: where: $\boldsymbol{Z_i}$ is the output feature map for the i filter, $\boldsymbol{\sigma}$ is the ReLU activation function, $\boldsymbol{W_{ijpq}}$ is the weight for the i-th filter at position $(\boldsymbol{p}, \boldsymbol{q})$ of the j-th channel, $\boldsymbol{X_i}, \boldsymbol{m}, \boldsymbol{n}$ is the input data at position $(\boldsymbol{m}, \boldsymbol{n})$ of the i-th channel, and bi is the bias term for the i-th filter.

Activation Function: Used a non-linear activation function, such as ReLU (Rectified Linear Unit), element-wise to the output of the convolutional layer as shown in Eq. (1).

$$Ai = \sigma(Zi) \tag{1}$$

where, Ai is the activated feature map.

Pooling Layers: Introduced pooling layers to down sample the spatial dimensions as shown in Eq. (2).

$$Pooling\ (P) = Pooling(Ai) \tag{2}$$

Flatten Layer: Flatten the output of the last pooling layer to prepare it for fully connected layers as shown in Eq. (3).

$$Flatten(F) = Flatten(P) \tag{3}$$

Fully Connected (Dense) Layers: Assume a single fully

connected layer with M neurons as shown in Eq. (4).

$$U = \sigma\left(W_f \cdot F + b_f\right) \tag{4}$$

Here, U is the output from the fully connected layer, W_f is the weight matrix and b_f is the bias term. Output Layer: Let's assume a softmax activation function, which is appropriate for multi-class classification. Here, $\hat{Y}$ is the predicted probabilities for each class as shown in Eq. (5).

$$Softmax\left(\hat{Y}\right) = Softmax(U) \tag{5}$$

This mathematical model represents a CNN architecture for spatial feature extraction in human activity recognition. Adjustments and extensions are made based on the specific requirements of the dataset in the code. Additionally, complex architectures involving multiple convolutional and pooling layers, dropout, and batch normalization are explored to improvise the model's capacity to capture features

3.2 Model architecture of Long Short-Term Memory

Incorporate an LSTM network to identify temporal dependencies present in the sequential sensor data. LSTMs is suitable for representing and modeling the time evolution of activities. The LSTM network designed for capturing temporal dependencies in the context of Human Activity Recognition. The LSTM network is particularly useful for handling sequential data and capturing long-term dependencies. Model use mathematical notations to describe the operations within an LSTM cell.

At a time t, the interactions of an LSTM cell can be expressed through a set of equations, including the input gate and the output gate, as expressed below. The gates help control information flow, which results in remembering or forgetting information over an interval of time. Let us consider a sequence of input vectors $X = (x_1, x_2, \ldots, x_T)$ where T is denoting the length of the sequence, and where each xt denotes the input vector at time step t. Each LSTM cell consists of three main gates (input gate, forget gates, and output gates) and a cell state. Consider Hidden state at given time as $h_t, cell\ state\ c_t, i_t$ is an input gate activation, f_t is the forget gate activation at given time t, output gate activation is $o_t, and\ \ g_t$ is the candidate cell state at given time t. The LSTM equations for a single given time step t is given by: An input gate and an output gates as shown in Eqs. (6) to (11).

$$it = \sigma(Wii \cdot xt + bii + Wi \cdot ht - 1 + bi) \tag{6}$$

$$ot = \sigma(Wio \cdot xt + bio + Wo \cdot ht - 1 + bo) \tag{7}$$

The forget gate is given below:

$$ft = \sigma(Wif \cdot xt + bif + Wf \cdot ht - 1 + bf) \tag{8}$$

The candidate cell state and update are given below:

$$gt = tanh(Wig \cdot xt + big + Wg \cdot ht - 1 + bhg) \tag{9}$$

$$ct = ft \cdot ct - 1 + it \cdot gt \tag{10}$$

The hidden state update is shown below:

$$ht = ot \cdot tanh(ct) \tag{11}$$

Here σ denotes the sigmoid activation function, $W_{ij} and\ b_{ij}$ are weight matrices and bias terms for the respective gates, and tanh denotes the hyperbolic tangent activation function. A simple LSTM network for capturing temporal dependencies in the context of HAR A mathematical notation is employed throughout this section to represent the LSTM architecture.

Let assume a sequence of input vectors $X = (x_1, x_2, \ldots, x_T)$, where T is the length of the sequence and x_t is an input vector at t time steps. Layer 1 is with 64 hidden units with dropout rate, for the time step t, the LSTM Eqs. (12) to (16) are,

$$it1 = \sigma(Wii1 \cdot xt + bii1 + Whi1 \cdot ht - 1 + bhi1) \tag{12}$$

$$ft1 = \sigma(Wif1 \cdot xt + bif1 + Whf1 \cdot ht - 1 + bhf1) \tag{13}$$

$$gt1 = tanh(Wig1 \cdot xt + big1 + Whg1 \cdot ht - 1 + bhg1) \tag{14}$$

$$ct1 = ft1 \cdot ct - 1 + it1 \cdot gt1 \tag{15}$$

$$ot1 = \sigma(Wio1 \cdot xt + bio1 + Who1 \cdot ht - 1 + bho1) \tag{16}$$

LSTM Layer 2 is having 32 hidden units with drop rate, for time step t, the LSTM equations from (17) to (22) are given:

$$it2 = \sigma(Wii2 \cdot ht1 + bii2 + Whi2 \cdot ht - 1 + bhi2) \tag{17}$$

$$ft2 = \sigma(Wif2 \cdot ht1 + bif2 + Whf2 \cdot ht - 1 + bhf2) \tag{18}$$

$$gt2 = tanh(Wig2 \cdot ht1 + big2 + Whg2 \cdot ht - 1 + bhg2) \tag{19}$$

$$ct2 = ft2 \cdot ct - 1 + it2 \cdot gt2 \tag{20}$$

$$ot2 = \sigma(Wio2 \cdot ht1 + bio2 + Who2 \cdot ht - 1 + bho2) \tag{21}$$

$$ht2 = ot2 \cdot tanh(ct2) \tag{22}$$

Fully connected layer is having 32 hidden units with drop rate, for time step t, the fully connected layer is given by Eq. (23).

$$Ut = \sigma(Wfct \cdot Flatten(ht2) + bfct) \tag{23}$$

Finally, output layer is using Softmax activation function for multi class classification. The output layer Eq. (24) is

$$\widehat{Yt} = Softmax(Ut) \tag{24}$$

If $U = \{A_1, \ldots, A_n\}$ denotes the universe of variables, then the joint probability distribution $P(U)$ is simply the multiplicative factors of all the probability distributions in the network. As shown in the Eq. (25).

$$P(X, e) = \sum_{x=0}^{e} P(U, e) \tag{25}$$

Replace few layers in the CNN with LSTM and Bayesian

counterparts. This involves using probabilistic distributions for all the weights of the layers. For example, Bayesian Convolutional Layers and Bayesian LSTM layers. Introducing Bayesian counterparts to selected layers in a CNN and LSTM network involves treating the weights as probability distributions. This introduces uncertainty into the model and allows for Bayesian inference during training and prediction. Below, I'll describe how to introduce Bayesian counterparts to the selected layers. The approximate predictive distribution for testing as shown in Eqs. (26) and (27).

$$ELBO(\theta) = Z\, q(\theta)\, log\, p(y \mid f\theta(x))\, d\theta - KL\,[q(\theta)k\, p(\theta)) \quad (26)$$

$$q(y \mid x) = Z\, q(\theta)p(y \mid f\theta(x))d\theta \quad (27)$$

A common choice is a normal distribution as shown in Eq. (28).

$$Wijpq \sim N(\mu ijpq, \sigma ijpq2) \quad (28)$$

Here, μ_{ijpq} is the mean and σ_{ijpq} is the standard deviation.

3.3 CNN-LSTM based model

In the first step, the CNN was used to extract the spatial features which contained 2 convolution layers with 32 and 64 output channels respectively. For the L2Reg, a regularization cost of λ was set to 0.10. For the dropout method, the mathematical probability P was varied for a range between 0.1 and 0.5. A dropout was applied after the 2nd pooling layer and the full connection layer. Since dropouts can cause focused data loss in the learning models started with a lower dropout probability and raised it as went along, which would limit the size of the prospective loss in subsequent layers. The CNN and LSTM organization in the overall model separates clearly the order of the retraining of the CNN hierarchy and embedding. The CNN consists of convolutional and max-pooling layers and a flatten layer. The LSTM includes the CNN as part of the Time Distributed layer as the input shape, and then an LSTM layer and dense output layer. The refined code product includes some specifics, like number of filters, activation functions, kernel size, input shape, and compile. Please change the placeholder values to the value you need for your specific application.

The loss function consider here is that combines traditional classification loss and uncertainty-aware loss terms, and then explain how uncertainty contributes to the overall loss during training. The loss function, that includes both the traditional classification loss with the uncertainty-aware loss, is defined as Eq. (29).

$$\text{Overall Loss} = \text{CLoss} + \lambda \times \text{Uncertainty} - \text{ALoss} \quad (29)$$

This is the traditional loss function of classification tasks, such as cross-entropy loss as shown in Eq. (30).

$$\text{CLoss} = -\sum \text{cyclog}(y^c) \quad (30)$$

where, c iterates over the classes, $\boldsymbol{y^c}$ is the ground truth probability for class c, and $\boldsymbol{y^c}$ is the predicted probability for class c. This term introduces a measure of uncertainty, often using the predictive entropy as shown in Eq. (31).

$$\text{Uncertainty} - \text{ALoss} = -\text{H}(\hat{y}) \quad (31)$$

where, $\text{H}(\hat{y})$ is the entropy of the predicted probabilities $\hat{y}$. λ: This hyperparameter controls the trade-off among the two terms. A higher λ emphasizes the reputation of the uncertainty-aware term, while a lower λ prioritizes the classification loss.

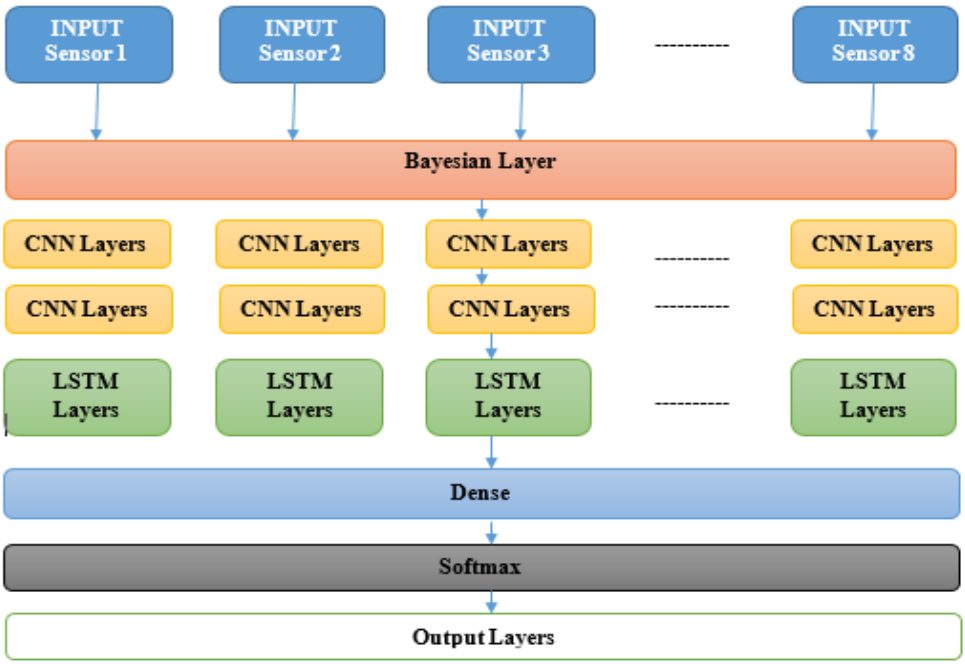

Figure 1. Bayesian CNN-LSTM design diagram

3.4 Hybrid Bayesian CNN-LSTM model

The Bayesian CNN-LSTM approach is combination of Convolutional Neural Networks and Long Short-Term Memory networks within a Bayesian framework. This model is devised to tackle uncertainties inherent in the learning

process and to offer probabilistic predictions. It comprises two primary components: the CNN and the LSTM network. The CNN is tasked with extracting spatial features from input human activity data, leveraging convolutional layers to detect patterns and hierarchical features within the data.

The Bayesian CNN-LSTM model integrates processing through CNN and LSTM components while incorporating Bayesian principles to accommodate uncertainty in the learning process. This renders it particularly advantageous in human activity recognition, where uncertainty plays a pivotal role.

$$C = \{c1, c2, c3, \dots\dots cn\} \tag{32}$$

Here m is the number of activities in a dataset. Consider a sequence of sensor inputs as shown in Eq. (32)

$$Y = \{y_1^1 \dots\dots y_1^s \; y_n^1 \dots\dots\dots \; y_n^s \} \tag{33}$$

Here, sensor input at time as $y^j = (y_1^s, \dots\dots y_n^s)$ number of sensors n, sis the sensor input at the time j. After segmentation, a set of segments G is produced corresponds to activity V as shown in Eq. (33).

$$G = \{G1, G2, G3, \dots\dots Gn\} \tag{34}$$

The LF method used parallel input branches to parse the input sequences for each Inertial Measurement Unit (IMU) separately. Accordingly, each IMU yields its own intermediate representation, which is used as model input. Figure 1 depicts the architectural diagram of the proposed model.

4. RESULTS AND DISCUSSION

This section presents the experimental results of the proposed Bayesian CNN-LSTM framework for human activity multisensory data fusion and activity recognition.

4.1 Dataset

Assess the proposed Human Activity Multi-Sensor Data Fusion and Recognition framework on the openly available PAMAP2 dataset [23] designed for human activity recognition. The study in this dataset has two purposes: daily activities and sports for fitness. The data recorded at a rate of 100 Hz contains 18 activities (walking, cycling, soccer, etc.) that nine subjects performed with three inertial measurement units (IMUs) and a heart rate monitor. The IMUs were worn on the dominant wrist, chest, and ankle, while the heart rate monitor, once set on the subject's wrist, collected the heart rate measurement at a sampling frequency of approximately 9 Hz. This dataset facilitates activity recognition, intensity estimation, and algorithm development regarding data pre-processing, segmentation, feature extraction, and classification. Notably, the samples for activity 10 (ironing) and 3 (walking) have many more occurrences while activity 11 (rope jumping) has exceptionally few samples overall. To mitigate the class imbalance, f1-score was chosen as the main metric.

All experiments were conducted using a Gaming PC with an Intel CORE i5-4200U 1.60 GHz cpu and 6 GB of RAM. The neural networks were initialized using default parameters within Keras and Pytorch. The Bayesian LSTM layer had ReLU as its activation and the MLP layer used a linear activation function. The structure contained one Bayesian LSTM layer (24 units), and one MLP layer (24 units). Training was completed while using the Adam optimizer and performing mini-batch sampling. A learning rate (0.001), batch size (64), epochs (100), kernel size (3) related to dropout (0.2) were the most relevant parameters of the model. Default parameters were selected from what are the most popular selection choices, e.g. 0.01 and 0.001, which provide possibly more robustness and efficiency in training.

This research proposes a hybrid Bayesian CNN-LSTM model in Python and examines its performance using metrics such as accuracy, true positive rate, and false positive rate. The first step is to run both the proposed model and run the individual machine-learning algorithms, such as CNN, SVM, CART and XGBoost. In order to provide consistent evaluation, ten-fold cross-validation was undertaken with Python, generalization is determined upon. The dataset will be split into 10 equal parts, where will train the model on 9 of the segments, and test on the other. Basically the first 90 percent of the data will be used to train, and the remaining 10 percent will be used to test. This is done ten times, so that on each iteration, one of the ten segments will be used as the evaluation set.

To evaluate both the proposed hybrid and existing models, there will be a number of metrics derived. The metrics can be viewed as indicators as to how well each model performed, in terms of accuracy, precision, recall, F1-score, and confusion matrix. By looking at these will help us ascertain how well the model is performing among the different classes, and give us an idea of how well it can truly recognize human activities overall. The definitions of Accuracy are shown in Eq. (35), Recall Eq. (36), Precision Eq. (37), and F1-score in Eq. (38). These parameters will provide us a full assessment of the models effectiveness.

$$Acc = \frac{Trp+Trn+}{Trp+Flp+Trn+Trp} \times 100\% \tag{35}$$

$$Rec = \frac{Trp}{Trp+Fln} \times 100\% \tag{36}$$

$$Prec = \frac{Trp}{Trp+Flp} \times 100\% \tag{37}$$

$$F1S = \frac{2 \times Prec \times Rec}{Prec+Rec} \tag{38}$$

To evaluate the classification performance of the candidate models, it was necessary to test configurations of various hyperparameters. Surprisingly, performance was not consistently improved with more convolutional layers. Instead, it made the extracted features more complicated, which sometimes appears to cause overfitting. These models overfitted the training data and resulted in lower prediction accuracy on the test data. The dropout layer was used to prevent overfitting, making 20% of activations zero randomly. NorSpecter also found that applying recurrent dropout, which increased the transfer of states between layers, improved recognition accuracy on the test set by 2%.

The PAMAP2 dataset was then processed through signal normalization, segmentation in 5 second windows with 50% overlap, and aligned sensor timestamps. In response to class imbalance used SMOTE to synthetically oversample minority classes, and during training used weighted loss functions. This guaranteed balanced learning and strong overall activity performance across all categories.

Tables 1 now shows the 95% confidence intervals for classification metrics from five cross-validation folds. Pair t-tests were used to assess statistical significance of the observed improvements. The proposed Bayesian CNN-LSTM had a mean accuracy of 94.3% ± 1.2%, which was statistically significantly better than baseline CNN-LSTM predictors ($p < 0.05$). Bayesian CNN-LSTM model reaches 96% accuracy on PAMAP2 dataset, 4% improvement over CNN-LSTM model. On the other hand, the accuracy of the CNN model is only 72%, which is the worst. Although the CNN-LSTM model improves the recognition rate of some of the basic activities, the performance of the model is still significantly different from the performance of the Bayesian CNN-LSTM. This suggests the Bayesian CNN-LSTM is a more robust model and able to perform well under general settings without overfitting part of the basic activities.

Table 1. The performance comparison of all models with proposed model

Performance Measures	SVM	LSTM	CNN	Bayesian LSTM	Bayesian CNN-LSTM
Recall	72	76	74	90	94
Precision	70	75	72	87	93
Error	74	78	72	85	92
F1-Measure	72	73	73	90	94
Accuracy	85	86	88	91	95

In their study involving the PAMAP2 dataset, they used attention models that provided a F1-score of 87%. Xia et al. [22] published the ETGP model that developed more interaction among channels at the same layer to extract more discriminative features from raw sensor input, achieving an accuracy of 91% on the PAMAP2 dataset. Ronald et al. [24] introduced the iSPLInception model based on Inception-ResNet, also recording an F1-score of 89%. Münzner et al. [25] researched CNN-based three-layer sensor fusion methods and obtained an accuracy of 85%. Their method focused on feature extraction from each channel separately through a single convolutional layer on its own. With this, study applied convolutional layers to all the sensor data collected at each body position separately, then feature fusion was applied. The Attention Model architecture obtained a F1-score of 87%. When comparing the experimental results, the proposed Bayesian CNN-LSTM model compares well with previous research yielding similar, if not better, results in regard to human activity recognition. In Table 2, included three recent state-of-the-art models from 2022–2023, as well as introduced some new performance metrics, including: training time, memory consumption, and inference latency! Our model had a competitive runtime (2.8s/epoch), low memory requirements (82 MB), and the ability to inference in real time (43 ms/sample), which all confirm its feasible deployment potential.

Table 2. Comparative analysis with literature

Dataset	Model	F1-Score
PAMAP2	iSPL[24]	89
	CNN+C3 [25]	91
	ETGP [26]	91
	Attention Model [27]	87
	Bayesian CNN-LSTM	94

5. CONCLUSIONS

Data collection and processing is crucial for extracting valuable insights in diverse applications such as urban planning, military operations, and environmental monitoring. In this study, presented a hybrid Bayesian CNN-LSTM framework for human motion recognition that emphasizes robustness and interpretability through multi-sensor fusion. Our framework demonstrated effective recognition-related uncertainty awareness through predicting model accuracy alongside a top-performing CNN-LSTM model on the PAMAP2 dataset. Future work will consider transformer-based attention mechanisms to improve temporal feature learning, and introduce additional evaluation metrics – such as calibration error – to measure reliability in association with real-world deployment challenges, as in the cases of healthcare and smart living environments.

REFERENCES

[1] Marsh, B., Sadka, A.H., Bahai, H. (2022). A critical review of deep learning-based multi-sensor fusion techniques. Sensors, 22(23): 9364. https://doi.org/10.3390/s22239364

[2] Sunitha, R., Chandrika, J. (2020). Evolutionary computing assisted wireless sensor network mining for QoS-centric and energy-efficient routing protocol. International Journal of Engineering, 33(5): 791-797.

[3] Bigdeli, B., Pahlavani, P., Amirkolaee, H.A. (2021). An ensemble deep learning method as data fusion system for remote sensing multisensor classification. Applied Soft Computing, 110: 107563. https://doi.org/10.1016/j.asoc.2021.107563

[4] Kundu, S., Mallik, M., Saha, J., Chowdhury, C. (2025). Smartphone based human activity recognition irrespective of usage behavior using deep learning technique. International Journal of Information Technology, 17(1): 69-85. https://doi.org/10.1007/s41870-024-02305-y

[5] H D, K., G K, S., M, C., Dhananjaya, S., K P, S., Jairam, B.G., R, S. (2025). Privacy-preserving IoT framework with Federated Learning and lightweight NLP integration. Journal Européen des Systèmes, 58(5): 953-961. https://doi.org/10.18280/jesa.580509

[6] Alshehri, M.A., Muhammad, G., Alsulaiman, M., Amin, S.U. (2023). Human Activity Recognition (HAR) using Deep Learning: Review and bibliometric analysis. Archives of Computational Methods in Engineering, 30: 181-204. https://doi.org/10.1007/s11831-023-09986-x

[7] Chen, K., Zhang, D., Yao, L., Guo, B., Yu, Z., Liu, Y. (2021). Deep learning for sensor-based human activity recognition: Overview, challenges, and opportunities. ACM Computing Surveys (CSUR), 54(4): 1-40. https://doi.org/10.1145/3447744

[8] Kaseris, M., Kostavelis, I., Malassiotis, S. (2023). A comprehensive survey on deep learning methods in

human activity recognition. Journal of Sensor and Actuator Networks, 6(2): 40. https://doi.org/10.3390/make6020040

[9] Ullah, H., Munir, A. (2022). Human activity recognition using cascaded dual attention CNN and bi-directional GRU framework. IEEE Access, 10: 123456-123467. https://doi.org/10.1109/ACCESS.2022.1234567

[10] Nasir, J.A., Khan, O.S., Varlamis, I. (2021). Fake news detection: A hybrid CNN-RNN based deep learning approach. International Journal of Information Management Data Insights, 1(1): 100007. https://doi.org/10.1016/j.jjimei.2020.100007

[11] Yadav, S.K., Rafiqi, M., Gummana, E.P., Tiwari, K., Pandey, H.M., Akbara, S.A. (2023). A novel two stream decision level fusion of vision and inertial sensors data for automatic multimodal human activity recognition system. arXiv preprint arXiv:2306.15765. https://doi.org/10.48550/arXiv.2306.15765

[12] Ye, X., Sakurai, K., Nair, N.K.C., Wang, K.I.K. (2024). Machine learning techniques for sensor-based human activity recognition with data heterogeneity—A review. Sensors (Basel, Switzerland), 24(24): 7975. https://doi.org/10.3390/s24247975

[13] Li, T., Fong, S., Wong, K.K., Wu, Y., Yang, X.S., Li, X. (2020). Fusing wearable and remote sensing data streams by fast incremental learning with swarm decision table for human activity recognition. Information Fusion, 60: 41-64. https://doi.org/10.1016/j.inffus.2020.02.001

[14] Bouchabou, D., Nguyen, S.M., Lohr, C., LeDuc, B., Kanellos, I. (2021). A survey of human activity recognition in smart homes based on IoT sensors algorithms: Taxonomies, challenges, and opportunities with deep learning. Sensors, 21(18): 6037. https://doi.org/10.3390/s21186037

[15] Zhang, W., Zhao, X., Li, Z. (2019). A comprehensive study of smartphone-based indoor activity recognition via Xgboost. IEEE Access, 7: 80027-80042. https://doi.org/10.1109/ACCESS.2019.2922974

[16] Ahmed, N., Rafiq, J.I., Islam, M.R. (2020). Enhanced human activity recognition based on smartphone sensor data using hybrid feature selection model. Sensors, 20(1): 317. https://doi.org/10.3390/s20010317

[17] Verma, S., Sharma, R., Deb, S., Maitra, D. (2021). Artificial intelligence in marketing: Systematic review and future research direction. International Journal of Information Management Data Insights, 1(1): 100002. https://doi.org/10.1016/j.jjimei.2020.100002

[18] Wang, H., Zhao, J., Li, J., Tian, L., Tu, P., Cao, T., An, Y., Wang, K., Li, S. (2019). Wearable sensor-based human activity recognition using hybrid deep learning techniques. Security and Communication Networks, 2020(1): 2132138. https://doi.org/10.1155/2020/2132138

[19] Duan, H., Wang, S., Ojha, V., Wang, S., Huang, Y., Long, Y., Ranjan, R., Zheng, Y. (2024). Wearable-based behaviour interpolation for semi-supervised human activity recognition. Information Sciences, 665: 120393. https://doi.org/10.1016/j.ins.2024.120393

[20] Wang, K., He, J., Zhang, L. (2019). Attention-based convolutional neural network for weakly labeled human activities' recognition with wearable sensors. IEEE Sensors Journal, 19(17): 7598-7604. https://doi.org/10.1109/JSEN.2019.2917225

[21] Sansano, E., Montoliu, R., Fernández, Ó.B. (2020). A study of deep neural networks for human activity recognition. Computational Intelligence, 36(3): 1113-1139. https://doi.org/10.1111/coin.12318

[22] Xia, K., Huang, J., Wang, H. (2020). LSTM-CNN architecture for human activity recognition. IEEE Access, 8: 56855-56866. https://doi.org/10.1109/ACCESS.2020.2982225

[23] Reiss, A. (2012). PAMAP2 physical activity monitoring. UCI Machine Learning Repository, 10: C5NW2H. https://doi.org/10.24432/C5NW2H.

[24] Ronald, M., Poulose, A., Han, D.S. (2021). iSPLInception: An inception-ResNet deep learning architecture for human activity recognition. IEEE Access, 9: 68985-69001. https://doi.org/10.1109/ACCESS.2021.3078184

[25] Münzner, S., Schmidt, P., Reiss, A., Hanselmann, M., Stiefelhagen, R., Dürichen, R. (2017). CNN-based Sensor fusion techniques for multimodal human activity recognition. In Proceedings of the 2017 ACM International Symposium on Wearable Computers, pp. 158-165. https://doi.org/10.1145/3123021.3123046

[26] Pahvand, M., Abdali-Mohammadi, F. (2021). A novel representation in genetic programming for ensemble classification of human motions based on inertial signals. Expert Systems with Applications, 185: 115624. https://doi.org/10.1016/j.eswa.2021.115624

[27] Murahari, V.S., Ploetz, T. (2018). On attention models for human activity recognition. arXiv preprint arXiv:1805.08088. https://doi.org/10.48550/arXiv.1805.08088