Computer Modeling in Engineering & Sciences

Tech Science Press

Doi:10.32604/cmes.2025.066984

ARTICLE

A Novel Attention-Based Parallel Blocks Deep Architecture for Human Action Recognition

Yasir Khan Jadoon[1], Yasir Noman Khalid[1], Muhammad Attique Khan[2], Jungpil Shin[3,*], Fatimah Alhayan[4], Hee-Chan Cho[5] and Byoungchol Chang[6,*]

[1]Department of Computer Engineering, HITEC University, Taxila, 47080, Pakistan
[2]Deparment of AI, Prince Mohammad bin Fahd University, Al-Khobar, 31952, Saudi Arabia
[3]Department of Computer Science and Engineering, University of Aizu, AizuWakamatsu, Fukushima, 965-0006, Japan
[4]Department of Information Systems, College of Computer and Information Sciences, Princess Nourah bint Abdulrahman University, P.O. Box 84428, Riyadh, 11671, Saudi Arabia
[5]Center for Computational Social Science, Hanyang University, Seoul, 01000, Republic of Korea
[6]Department of Computer Science, Hanyang University, Seoul, 01000, Republic of Korea

*Corresponding Authors: Jungpil Shin. Email: jpshin@u-aizu.ac.jp; Byoungchol Chang. Email: bcchang@hanyang.ac.kr

Received: 22 April 2025; Accepted: 16 July 2025; Published: 31 July 2025

ABSTRACT: Real-time surveillance is attributed to recognizing the variety of actions performed by humans. Human Action Recognition (HAR) is a technique that recognizes human actions from a video stream. A range of variations in human actions makes it difficult to recognize with considerable accuracy. This paper presents a novel deep neural network architecture called Attention RB-Net for HAR using video frames. The input is provided to the model in the form of video frames. The proposed deep architecture is based on the unique structuring of residual blocks with several filter sizes. Features are extracted from each frame via several operations with specific parameters defined in the presented novel Attention-based Residual Bottleneck (Attention-RB) DCNN architecture. A fully connected layer receives an attention-based features matrix, and final classification is performed. Several hyperparameters of the proposed model are initialized using Bayesian Optimization (BO) and later utilized in the trained model for testing. In testing, features are extracted from the self-attention layer and passed to neural network classifiers for the final action classification. Two highly cited datasets, HMDB51 and UCF101, were used to validate the proposed architecture and obtained an average accuracy of 87.70% and 97.30%, respectively. The deep convolutional neural network (DCNN) architecture is compared with state-of-the-art (SOTA) methods, including pre-trained models, inside blocks, and recently published techniques, and performs better.

KEYWORDS: Human action recognition; self-attention; video streams; residual bottleneck; classification; neural networks

1 Introduction

Human Action Recognition (HAR) has emerged as a pivotal area in pattern recognition and computer vision, revolutionizing the way video data is analyzed and interpreted [1,2]. In recent years, HAR has emerged as a key field of research which greatly improves video analysis [3]. Applications for HAR can be found in many domains, including robots, human-computer interfaces, healthcare monitoring systems, video surveillance, and pedestrian tracking [4,5]. To support automatic action detection systems, HAR entails recognizing actions such as walking, running, punching, leaping, and playing inside video sequences

[6]. Wearable technology, wireless sensor networks, and video-based techniques are only a few of the technologies that are included in HAR [7]. Among these, video-based HAR stands out for its greater accuracy in action recognition and extensive adoption across numerous industries due to its straightforward deployment [8,9].

Due to several elements, including camera views, illumination changes, noise from the environment, and intra and inter-class variances, accurately detecting human behaviors in videos is still a tough challenge [10,11]. Previous methods have been based on handcrafted feature extraction techniques such as Motion Boundary Histograms (MBH) and Scale-Invariant Feature Transform (SIFT) on 2D video frames [12]. Usually, these techniques focus on identifying human appearances and body motions within single frames, but they don't consider the three-dimensional spatial relationships in the action scenes [13]. Consequently, the practical concerns outlined above cannot be fully addressed using only RGB or RGB-D video frames for HAR. Additionally, occlusion, the condition in which a part of the human body is hidden or obscured and causes an incorrect recognition, is a problem for traditional approaches [14]. The dynamic, fine-grained motions and intricate articulations of human movements make the technique more difficult. Recent advancements in depth-sensing technologies have significantly advanced the field by offering detailed 3D data on body parts' motions and postural orientation [15]. Depth data facilitates the computation of joint positions and their relative distances, strengthening the distance vectors between video frames and improving action identification ability [16]. These developments also make it easier to integrate multi-modal data, like 3D motion analysis and skeletal tracking, which enables more thorough and precise interpretations of human activity in complicated contexts [17].

Human action analysis, segmentation, and classification using classical methods usually include recording body component movements to identify activities, extracting human silhouettes from noisy backdrops, and categorizing surroundings based on actions observed [18,19]. For a considerable amount of time, handcrafted features have been used to extract pertinent aspects from video sequences for categorization [20]. Convolutional Neural Networks (CNNs), one of the most recent developments in deep learning, have shown notable gains in recognition accuracy [21]. Research indicates that CNNs perform better than conventional techniques in several fields, such as medical imaging, object identification, and video surveillance [22]. In computer vision, CNNs are a desirable technique due to their efficiency in handling big datasets and complex patterns [23]. Transfer learning is a key component of several cutting-edge CNN-based methods, which use pre-trained models like Nasnet, ResNets, and Inception networks to obtain improved accuracy with shorter training times [24]. The development of HAR systems has progressed even more since transfer learning allows for more rapid adaptation to new datasets while preserving reliable performance [25]. CNNs are also successful in action recognition tasks because of their hierarchical feature extraction capabilities, which enable a more detailed representation of spatial and temporal information in video data [26]. The move toward deep learning architectures has created new avenues for integrating multi-modal data sources, improving the accuracy and adaptability of HAR systems in various demanding scenarios [27].

Although trained CNN models are flexible and helpful in many contexts, there is an increasing need to create customized deep neural network models with specific features adapted to the intricacy of tasks involving the recognition of human actions [28]. Each video frame consists of numerous layers of information, and these customized neural networks need to evaluate and interpret complex human activities using these frames. In order to achieve this task, the contributions are provided below.

- Introduced a unique hybrid neural network architecture with seven separate components, each of which uses parallel bottleneck techniques to extract and analyse features.
- Introduced an attention-oriented feature matrix for precise Human Activity Recognition (HAR) classification by incorporating an attention-based feature extraction process at the end of these components.

- A thorough comparative analysis is performed against cutting-edge methods to confirm the introduced architecture's efficacy and performance.

2 Literature Review

Researchers frequently employ varied methodological frameworks to examine and interpret their findings, resulting in various viewpoints regarding the problems and progress in the field. The primary research findings and innovations in the subject of HAR are presented below, with an emphasis on the developing methods and tools that are propelling advancement. He et al. [29] suggested a novel deep learning model to extract human action patterns from videos. In complex and dynamic video environments, robust human action recognition is required, where traditional methods struggle due to different temporal variations and added visual noise. To extract temporal characteristics through long-range temporal learning and sparse temporal sampling, a sample representation learner was presented. A densely connected bi-directional LSTM (DB-LSTM) network was created to simulate temporal linkages in both directions to enhance long-range action identification. The DB-LSTM model produced encouraging results in experiments on UCF101 and HMDB51, with an accuracy of 81.20% and 97.30% on HMDB51 and UCF101, respectively. The authors showed an intention to enhance work in the future. Uncertain temporal dynamics and static biases made video actions harder to identify in an open-set setting than image data. Bao et al. [30] presented a Deep Evidential Action Recognition (DEAR) approach to address action recognition in open testing sets. A technique for identifying known and unknown human behaviors in recordings. It tackles crucial issues such as static bias, temporal dynamics, and overconfident predictions in conventional models. DEAR employs Contrastive Evidence Debiasing (CED) to lessen scene reliance, Evidential Uncertainty Calibration (EUC) to increase prediction reliability, and Evidential Deep Learning (EDL) for principled uncertainty estimation. To regularize EDL training, the authors presented a novel model calibration technique and posed the problem using an evidential deep learning (EDL) methodology. With an accuracy of 77% on the HMDB51 dataset, experimental findings showed that the DEAR approach consistently outperformed several conventional models and benchmarks.

Identifying human activity in video surveillance presents difficulties, such as handling increasing amounts of streaming data with complex computations in an effective manner. To overcome this, authors Ahmad and Wu [31] presented a multilayer Gated Recurrent Unit (MLGRU) based lightweight spatial-deep features integration. It draws attention to the shortcomings of conventional deep learning models that solely use high-level features and stresses the necessity of integrating temporal and geographical data to enhance recognition performance. Using the MobileNetV2 model, spatial and deep information was retrieved from video frames and combined to improve recognition. Tests conducted on the YouTube11, HMDB51, and UCF101 datasets demonstrated noteworthy results, with HMDB51 demonstrating an accuracy of 80.61% and minimal computing cost. Developing new techniques for automatic comprehension of video data has been an important area of research. Many algorithms were investigated, with the primary goals being the extraction of geographical information and temporal connections.

On the other hand, learning techniques and motion feature extraction were frequently developed independently. Authors draw attention to shortcomings in existing deep convolutional neural network (DCNN) models that prevent practical implementation, such as their dependence on independent variables, computationally demanding motion extraction (i.e., optical flow), their inability to learn spatial and temporal properties end-to-end, and their low resilience to noise in real-world scenarios. To tackle this issue, the author Giveki [32] created a novel Gated Recurrent Unit (GRU) network that can simultaneously record motion characteristics, spatial data, and temporal dynamics. Tests on the YouTube2011, UCF50, UCF101, and HMDB51 datasets proved the model's excellence and generalizability. In particular, it obtained an

accuracy of 82.30% on HMDB51, demonstrating the approach's strong performance and noise resistance in practical applications. The author shows a keen interest in developing more techniques for the HAR domain. Applications for HAR, which identify actions from temporal video material, can be found in several fields, including autism care and video retrieval. In this work, authors Sowmyayani et al. [33] highlighted that accurate and effective action recognition from video data is hampered by high computational cost and efficient temporal feature extraction. They have used temporal feature extraction in residual frames to overcome these problems. For action recognition, both spatial and temporal information were extracted using the Frame Differencing (FD) approach. Keyframes were used to extract spatial characteristics, and residual frames were used to recover temporal features. These were combined to create spatiotemporal features, which were then categorized with the aid of a Multiclass Support Vector Machine (MSVM). The suggested technique outperformed previous methods with an accuracy of 85.8%, 98.83%, and 96.6% on the HMDB51, UCF101, and UCF Sports datasets. To use spatial and temporal information, authors Varshney and Bakariya [34] highlighted the inefficient integration of spatial and temporal data, moreover, discusses the shortcomings of the existing human activity recognition (HAR) models for video frames. Authors draw attention to problems such as distinct motion and appearance processing, ineffective fusion techniques, and weak resilience in differentiating comparable movements because of static or sparse motion representations. They suggested a deep convolutional neural network (CNN) model for human activity recognition in videos by combining multiple CNN streams. The outputs of the spatial and temporal streams were combined using two different fusion strategies: average fusion and convolution fusion. The suggested strategy fared better on the UCF101 and HMDB51 benchmark datasets than other cutting-edge techniques. In particular, the model's test accuracy on UCF101 was 97.2% with convolution fusion and 95.4% with average fusion. Convolution fusion achieved 85.1%, while the average fusion method scored 84.3% for the HMDB51 dataset.

This study presented by Yang and Zou [35] identified that the accuracy of current deep learning models for action recognition is limited because they do not fully utilize spatiotemporal information. The authors suggested a deep learning model based on spatio-temporal feature fusion (STFF) to overcome the shortcomings of existing deep learning networks in fully extracting and fusing spatio-temporal information for action detection, which leads to low accuracy. To extract and integrate spatial and temporal data, the model used two networks: Long Short-Term Memory (LSTM) and Convolutional Neural Networks (CNN). Large-scale video frames were handled using a multi-segment input technique, which addressed long-term dependencies and increased prediction accuracy. Furthermore, an attention mechanism improved the network's emphasis on critical visual components. The success of the strategy was confirmed by experimental findings on the UCF101 dataset, which showed that the suggested FSTFN model outperformed the Two-stream model by 4.7%, achieving 92.7% accuracy. This study by Gowada et al. [36] addresses the problem of identifying immoral human behavior in video, like violence or pornography, which is hampered by occlusion, background clutter, and shifting points of view. It highlights how difficult it is for existing deep learning models to extract high-level characteristics in these kinds of intricate settings efficiently. The authors combined spatio-temporal attention (STA) modules with a two-stream inflated 3D ConvNet (I3D) to develop a deep learning-based hybrid model for identifying unethical human behaviors. While the STA module increased learning by concentrating on spatial and temporal information in each frame, the I3D model improved the performance of 3D CNNs by inflating 2D convolution kernels into 3D kernels. To assess the model's efficacy, a multi-action dataset was created using subsets from several sources, such as Weizmann, HMDB51, UCF101, NPDI, and UCF-Crime. The suggested model performed better on these datasets compared to current methods. The accuracy of the approach on the UCF101 dataset was 96.40%. The authors showed an intention to improve the methodology in the future.

This research, suggested by Dastbaravardeh et al. [37], analyzed the associated difficulty of successfully identifying human activities in low-resolution and low-size video frames. They have demonstrated that typical deep learning models fail due to high computational costs and diminished visual detail. Authors draw attention to the dearth of efficient frameworks that can analyze such limited input with great accuracy, particularly in real-time applications. So, the authors provides a better way to use convolutional neural networks (CNNs) with channel attention mechanisms (CAMs) and autoencoders (AEs) to detect human behaviors in low-size, low-resolution movies. While random frame sampling increased accuracy with less input, enhanced convolutional layers identified important characteristics. The model tackled issues such as uncertainty, computational complexity, and overfitting. Its accuracy was 98.87%, 97.16%, and 77.29% on the UCF50, UCF101, and HMDB51 datasets. The outcomes validated the model's capacity to accurately categorize human actions and its appropriateness for processing low-resolution, low-size videos. The majority of previous approaches, which only used RGB flow or its combination with optical flow, had trouble with background interference, especially when high-activity regions weren't highlighted. This research by Xiong et al. [38] identified that accurately identifying human actions in videos under complicated circumstances, including background interference, redundant frames, and limited temporal modeling, is difficult using traditional techniques. Authors suggested a novel approach that uses a two-stream fusion network and action sequence optimization to overcome these problems. The method highlighted high-activity regions, removed unnecessary intervals, and recorded long-range temporal information by using shot segmentation and dynamic weighted sampling to reconstruct films. A two-stream 3D dilated neural network that combined RGB and human skeletal characteristics was introduced. While the dilated CNN increased the receptive field for better feature extraction, the skeletal information improved human depiction and reduced background interference. The suggested approach performed better or similarly on the HMDB51 dataset and attained 96.15% accuracy on the UCF101 dataset. The presented work in this section motivated me to develop a methodology that can achieve a considerable or comparable classification accuracy to existing techniques. The next section presents the methodology through which considerable improvement was observed.

3 Datasets

Two datasets are employed to check the generalizability of the presented methodology, such as HMDB51 [39] and UCF101 [40]. One well-known database of human actions is the Human Motion Database (HMDB51). It has 51 distinct action classes and 6766 videos, gathered from various sources, including movies, public databases, and YouTube videos. The variety of sources offers a diversity of environments and circumstances for each action class. Seventy-nine thousand one hundred thirty-three videos are taken from these and made available on the Kaggle dataset repository.

The UCF101 dataset constitutes 13,320 videos, which are primarily used for action identification in videos. There are 101 classes where humans are in action while performing different tasks. The videos are in the Audio Video Interleave (.avi) format and typically have dimensions of 320 by 240 pixels. The entire video lasts for 27 h. Among the categories are biking, billiards, bowling, breaststroke, fencing, haircut, high jump, ice dancing, mixing, etc. A three-frame-per-second rate is used to extract frames. Fig. 1 represents a few sample human actions of the selected datasets.

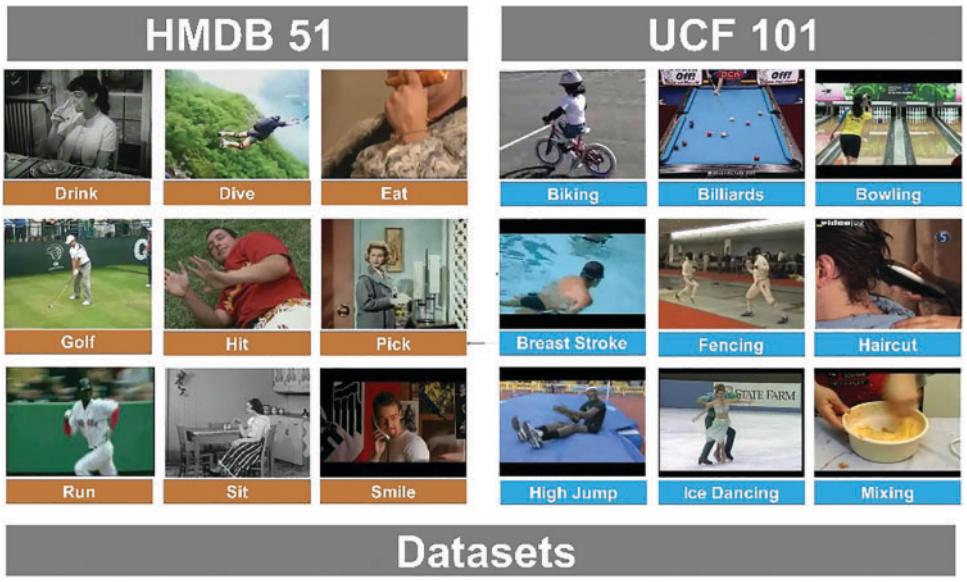

Figure 1: Sample frames of the selected datasets

4 Proposed Methodology

The suggested approach shows a hybrid DCNN specifically designed for HAR tasks. The architecture uses a number of well-optimized architectural elements to extract both spatial and temporal characteristics from video frames efficiently. The efficacy of the model is verified by training and assessing it on two benchmark datasets, HMDB51 and UCF101, which comprise 51 and 101 activity classes, respectively. A stack of seven specially created blocks, each intended to extract multi-scale elements, starts the foundation of the architecture. Parallel convolutional routes with 1×1 and 3×3 convolutional filters are included in each block. The 3×3 convolutions are in charge of capturing local characteristics and spatial correlations, whereas the 1×1 convolutions are mostly used for dimensionality reduction and inter-channel interaction. After combining these parallel serial routes using element-wise addition, the aggregated data is integrated by a second convolution layer. Robust feature learning is facilitated by this design pattern, which enables the network to extract rich and diverse feature representations from several levels of abstraction. Feature discrimination is improved by adding a self-attention mechanism after the seven blocks. In particular, the network uses the extracted features to build query, key, and value matrices. Through a scaled dot-product attention operation, these matrices interact to provide attention scores that suppress noisy or redundant information while emphasizing the most important aspects. An improved attention-weighted feature matrix that captures the most important data required for classification is returned from this step. The attention-enhanced feature representation is then processed through a classification head, which is usually made up of a Softmax activation function after fully connected layers. The class probabilities for the target activity classes in the dataset are produced by this function. The DCNN network is trained from beginning to end, so the feature extraction and classification components can be optimized simultaneously. The suggested architecture achieves competitive or superior performance across a number of HAR benchmarks, according to a thorough examination that includes comparison with state-of-the-art techniques. Fig. 2 illustrates the proposed HAR architecture that is employed for the training on the selected datasets. A detailed description of this model is given in the subsections below.

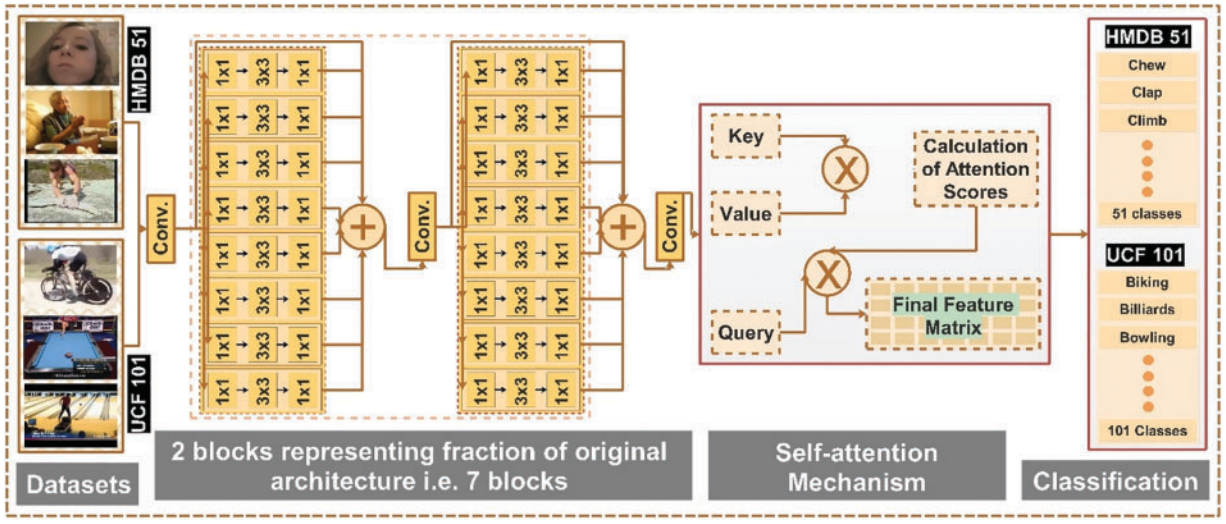

Figure 2: Proposed attention-RB model for human action recognition

4.1 Proposed Attention-RB Net

The proposed customized Attention-RB Net model is visually presented in Fig. 2. The proposed model contains 374 layers, including the classification layer. The model has 11.2 million parameters. The dataset(s) is passed to the model, which traverses through the model, and classification is performed at the end. Traversing the model reveals seven skip connections based on residual blocks. Each block contains 8 parallel bottleneck mechanisms. Each bottleneck mechanism constitutes three series of convolution layers with 1×1, 3×3, and 1×1 filter sizes. A self-attention layer is added at the end of these 7 residual blocks to further improve the feature information. The attention-based feature matrix is passed to the Softmax layer, and final classification is performed.

Description: The network starts with an initial convolutional layer with a depth of 16, which extracts low-level features. It includes edges and textures from input video frames and preserves spatial dependencies crucial for further processing. A Rectified Linear Unit (ReLU) layer follows the convolution layer. This layer is responsible for setting values to zero that are less than zero during the convolutional operation. These low-level features provide a good start to learning about intrinsic features at a higher level. It enables the DCNN to classify human actions provided in datasets.

After the initial step, the network introduces a series of seven residual blocks; at the end of each block, the network's depth doubles until it ends at 1024. Each block has eight parallel bottleneck architectures. In this architecture, convolutional layers with 1×1, 3×3, and 1×1 filter sizes are used in series. Two blocks are depicted in the figure. Inter-channel dependencies are effectively captured using 1×1 convolution layers, which reduce dimensionality and empower feature scaling across different channels.

On the other hand, the spatial relationship is captured using the 3×3 convolution layers between neighboring pixels. This enables the network to learn hierarchical features. Gradient flow during backpropagation is improved by introducing skip connections in the form of residual connections. It will also help in mitigating the vanishing gradient problem. The proposed design will allow the deeper network to train the entire network without compromising performance.

The residual connection bypasses specific layers and directly influences later layers. It retains valuable low-level information that may be lost if no skip connection is used. These connections enrich shallow

and deep features with the utmost learning about action-related features. These connections also make the training process more stable, and faster convergence of gradients during optimization is possible. Residual learning provides a good mix of low-level and high-level features. The expressive power and computational efficiency are achieved using 1×1 and 3×3 convolutions within the residual blocks. The 1×1 convolution acts as a dimensionality reduction layer. They also minimize computational overhead while retaining essential information about features. Contrary-wise, the 3×3 convolution layer captures spatial patterns in the video frames. Rich feature representations are learned using this multi-scale feature extraction capability. These are crucial in the classification of diverse human actions.

Long-range dependencies among features are calculated using a self-attention mechanism. This emphasizes the extraction of the most relevant features within the video frames. Attention scores are calculated based on query, key, and value matrices. The process refines the feature representation. The final refined feature matrix is then passed to the classification layer. It generates the network's predictions for accurate human action classification.

4.1.1 Residual Bottleneck Architecture

The residual bottleneck design is a more sophisticated method [41] used in deep neural networks, namely in ResNet (Residual Networks). To increase depth efficiency and compute performance, it employs a bottleneck architecture and integrates identity mapping through shortcut links. The breakdown and required mathematical expressions are shown below:

Mapping the identity—Adding extra layers to deep networks frequently results in degradation issues, whereby deeper networks perform worse because of optimization difficulties. This is resolved by identity mapping, which bypasses layers with shortcut connections, facilitating gradient flow and maintaining the trainability of deeper networks. For identity mapping, an essential equation is given below:

$$z = G\left(t, \{V_i\}\right) + t \tag{1}$$

In this equation, t denotes the input, $G\left(t, \{\text{and } V_i\}\right)$ represents the residual function used for learning and usually constitutes convolutions and activation functions. The addition of input to the output of residual function by skipping provides a skip connection. The formulation helps residual function to learn as follows:

$$G(x) = Q(t) - t \tag{2}$$

$Q(x)$ denotes output and simplifies optimization by merely understanding the difference in identity. The bottleneck architecture reduces the number of parameters and calculations without compromising representational power, making deep networks more efficient. The bottleneck block reduces the computational effort by compressing and then expanding the input instead of employing huge convolutional layers. Three layers make up a bottleneck block. A 1×1 convolution compresses the input by lowering the dimensionality. The basic computation on the compressed input is carried out using a 3×3 convolution. Regaining the original size of the output requires another 1×1 convolution. In the bottleneck block, the function $G(t)$ is represented as follows:

$$G\left(t\right) = V_3\sigma\left(V_2\sigma\left(V_1t\right)\right) \tag{3}$$

In the equation, V_1, V_2, and V_3 represent weights of 1×1, 3×3, and 1×1 convolutions operation. The activation function is denoted by σ. In this case, it is the Rectified Linear Unit (ReLU). Given below, Fig. 3 represents the procedure pictorially.

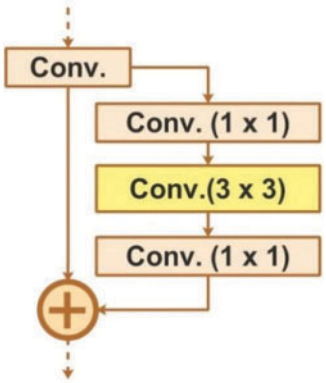

Figure 3: Depiction of residual bottleneck architecture

4.1.2 Self-Attention Mechanism

The growing popularity of self-attention mechanisms can be attributed to their high computational parallelization and flexibility [42]. Every pixel in the input image has a value, key, and query vector allocated to it. These vectors are computed via linear transformations of the pixel embeddings based on learned weight matrices. While the query vector represents the representation of each individual pixel, the key and value vectors are utilized to aggregate data from other pixels and determine attention weights, respectively. Adjacent pixels' attention scores are computed. This is achieved by scaling the result after taking the dot product of a query and the key vector of nearby pixels. The attention weights are then standardized using a Softmax function to obtain attention weights. An attention-weighted pixel is more significant when determining the representation of the current pixel. Next, the weighted sum of the value vectors for every pixel in the image is computed using the attention weights. This aggregation method yields a refined and context-aware representation of the input image, where each pixel's contribution is weighted according to its attention weight.

The output of the self-attention [43] layer is a weighted sum of values that serves as an updated representation of the input image. It records the interplay and interdependencies among all of the image's pixels. In conclusion, a self-attention layer enables a neural network to dynamically focus on various input image regions, efficiently capturing contextual information and spatial connections. Fig. 4 represents the self-attention mechanism:

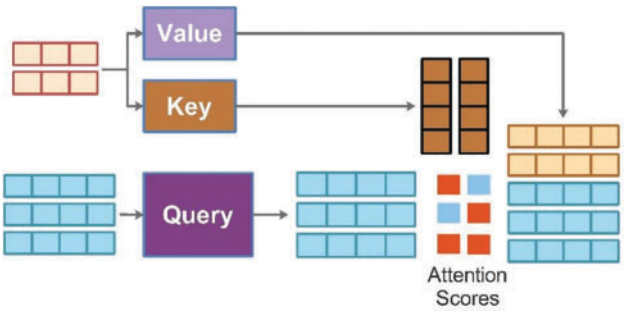

Figure 4: Self-attention mechanism adopted in the proposed architecture

The mathematical representation of the attention mechanism is provided below:

$$a \in \mathbb{R}^{E \times N} \tag{4}$$

$\boldsymbol{A}$ denotes the single input feature matrix, E denotes the total channels, and N denotes the product dimensions. Let us have three convolutions denoted by:

$$f(a), g(a) \,\&\, b(a) \,.f(a) = I_f a \tag{5}$$

$$h(a) = I_h^T a \tag{6}$$

$$b(a) = I_E a \tag{7}$$

$$I_f, I_h, I_E \in \mathbb{R}^{E^* \times E} \tag{8}$$

The activation map of Softmax is given as:

$$G_{j,i} = \frac{e\left(S_{i,j}\right)}{\sum_{i=1}^{N} e\left(S_{i,j}\right)} \tag{9}$$

$$S_{j,i} = f(a_i)^T h\left(a_j\right) \tag{10}$$

$$S_{j,i} = \vec{I_h}(a) \tag{11}$$

$$K_j = U\left(\sum_{i=1}^{N}\left(G_{j,i}\right) \cdot b(a_i)\right) \tag{12}$$

So, the final length of the feature map is:

$$V_a = \mathbb{R}^{E \times E^*} \tag{13}$$

In the end, the output has the same number of channels as the input features to the self-attention layer.

4.1.3 Bayesian Optimization Based Hyperparameters Selection and Training

In this work, we selected the hyperparameters of this model through the Bayesian optimization (BO) algorithm [44] instead of manual selection. BO attempts to minimize the scalar objective function $f(x)$ for input value x in a bound domain. The minimization process is based on the following points:

i. $y_i = f(x)$ is a Gaussian process model.
ii. A Bayesian update procedure for modifying the Gaussian process model at each new evaluation $y_i = f(x)$. The posterior distribution is computed over $f(x)$ at this point.
iii. An acquisition function is adopted to maximize the next point of x in $f(x)$. In this work, we employed an expected improvement acquisition function that is mathematically formulated as follows:

$$EI(x,q) = E_q\left[\text{Max}\left(0, \mu_q\left(x_{best}\right) - f(x)\right)\right] \tag{14}$$

The above process is terminated based on the fixed number of iterations, such as 50 in this work. After the fixed number of iterations, we obtained the best points further utilized for a model's training. For the proposed model, we obtained the following hyperparameter values: best learning rate value of 0.00036, momentum value of 0.663, batch size of 32, dropout value of 0.5, and optimizer is ADAM. Moreover, we performed a total of 100 epochs on each dataset.

After that, the proposed model is trained on the selected datasets. Two different models are saved at the end—one for HMDB51 and another for UCF101. Both trained models are utilized in the final classification and validation testing phase.

4.2 Proposed Model Testing

After the training phase, the next phase tests the proposed model on the selected datasets (testing set). In the testing phase, features are extracted from the self-attention layer using testing frames. The dimension of extracted features on this layer is $N \times 1024$. The extracted features are later passed to the classifiers, who obtain the numerical values and visually labeled output. Visually, the proposed model testing procedure is shown in Fig. 5. This figure shows that the wide neural network (WNN) classifier is picked as the best classifier based on accuracy.

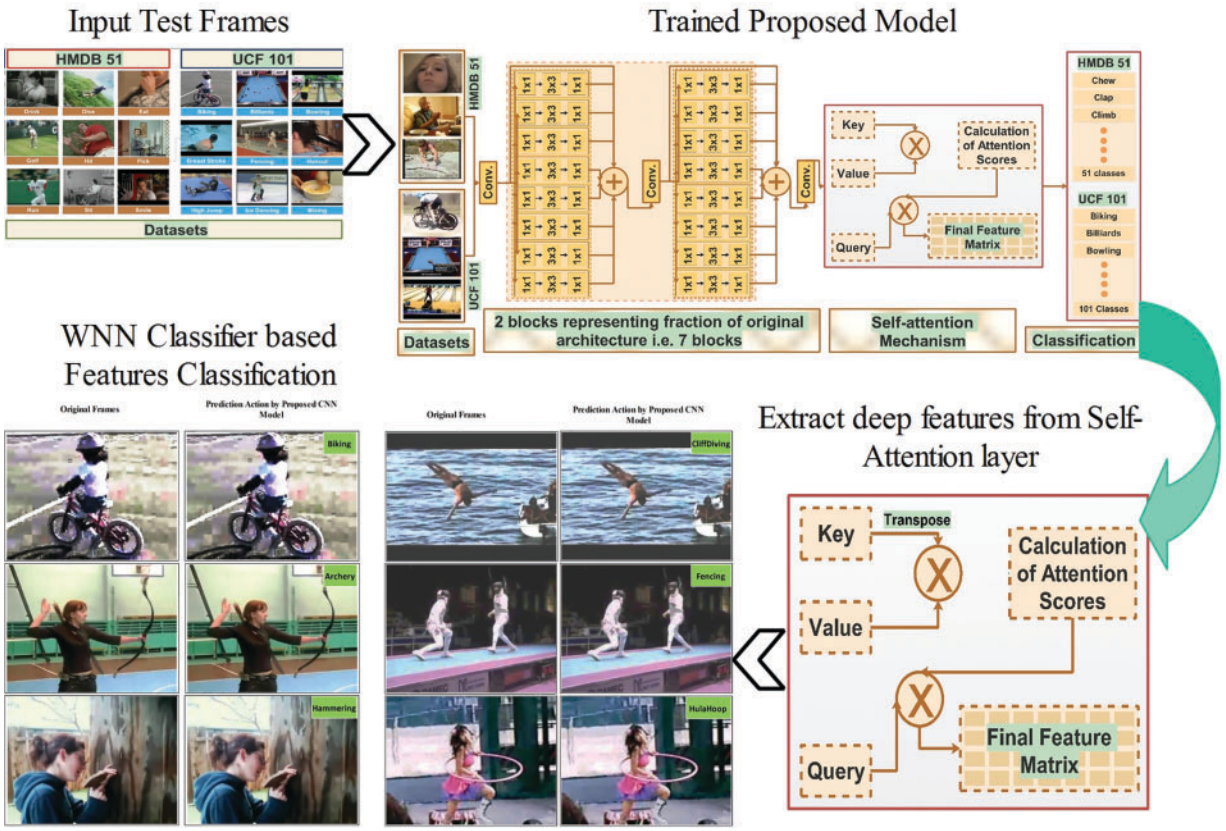

Figure 5: Proposed deep model testing process for action classification

A wide neural network is usually used for continuous learning tasks. A wider network can better capture data complexity and minimize task interference by enhancing gradient orthogonality. Additionally, wider networks are more resilient to catastrophic forgetting, which occurs when knowledge is replaced by fresh information [45]. In addition, the wide neural network classifiers need average training time but excessive computing resources [46].

5 Results and Discussion

5.1 Experimental Setup

Self-attention-based seven parallel residual bottleneck architecture (Attention-RB) is used for the experiments. The optimizer, Stochastic Gradient Descent with Momentum (SGDM), ensures convergence throughout the training process. The dataset was divided 50:50 between training and testing, and a 10-fold cross-validation procedure was used. The key hyperparameters are mentioned above and the best learning rate value is 0.00036, momentum value is 0.663, batch size is 32, dropout value is 0.5, and optimizer is ADAM. Moreover, we performed a total of 100 epochs on each dataset. An NVIDIA GeForce RTX 3060 GPU and a 12th Generation Intel Core i5 processor are used for computing; the experiments are carried out on MATLAB 2024b.

5.2 Dataset and Performance Measures

The HMDB51 and UCF101 datasets are used to validate the generalizability of the model. Both datasets are well-established and used in the scientific community's literature. The datasets are publicly available. The validation of results is tested via different performance measures, such as accuracy, false negative rate, and computational time. These measures check the underlying methodology's efficiency on the provided dataset.

Performance metrics such as false negative rate (FNR), time, and accuracy thoroughly assess a model's efficacy. The model's total performance is evaluated by accuracy, which gives information on how well it can categorize both positive and negative samples from the dataset [47]. For real-time applications or those with constrained computational resources, the model must function rapidly, and time quantifies the computational efficiency of the model. In high-stakes situations like healthcare or security, where missing a real positive (such as a sickness or threat) might have dire repercussions, the false negative rate is crucial [48]. When taken as a whole, these criteria guarantee a fair assessment of the model's precision, speed, and dependability in avoiding important mistakes.

5.3 Classification Results on the HMDB51 Dataset

Completing epochs on the training dataset generates several numeric values depicting the DCNN model efficiency. The training and loss graph for HMDB51 dataset is provided in Fig. 6.

To validate testing accuracy, the testing dataset is used, and five neural network-based classifiers are used to classify the features. The performance measure's value against each classifier is listed below in Table 1. The wide neural network has the highest accuracy value, i.e., 87.70%, the lowest processing time, i.e., 189.50 s, and the lowest number of false negatives, i.e., 12.30%, which proved the significance of the model's performance on the HMDB51 dataset. A tri-layered neural network achieves the worst value, i.e., 51.60% accuracy, maximum processing time, i.e., 697.60 s, and highest false negative rate of 48.40%. A confusion matrix of the highest performing classifier, i.e., wide neural network, is provided in Fig. 7.

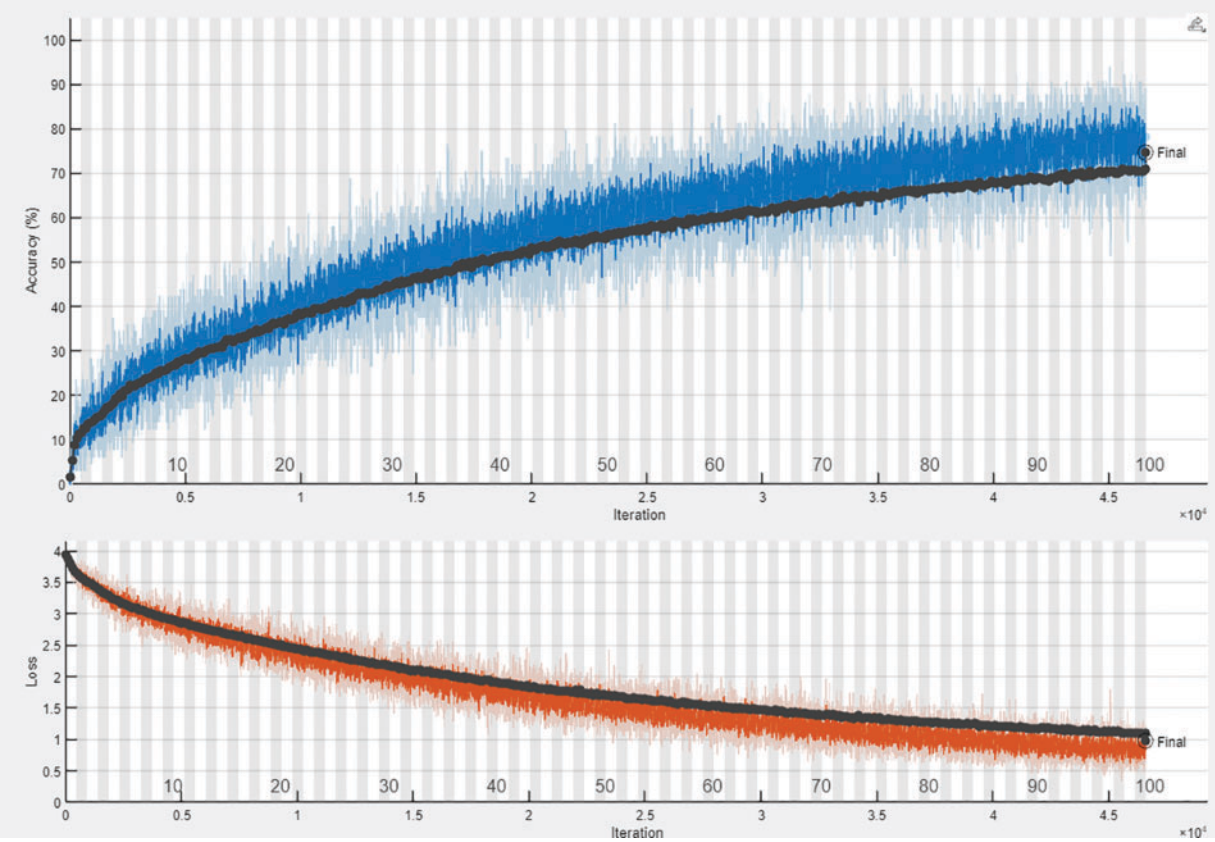

Figure 6: Training and loss graph for the HMDB51 dataset

Table 1: Classification results of the proposed architecture for HMDB51 dataset

Classifier	Accuracy (%)	Processing time (s)	False negative rate (%)
Wide NN	**87.70**	**189.50**	**12.30**
Medium NN	76.10	681.42	23.90
Narrow NN	55.90	655.97	44.10
Bi-Layered NN	53.90	672.30	46.10
Tri-Layered NN	51.60	697.60	48.40

Note: Bold denotes the best values.

5.4 Classification Results on UCF101 Dataset

The DCNN efficiency is represented by several numerical values that are produced by finishing epochs on the training dataset. Fig. 8 shows the training and loss graph for the UCF101 dataset.

Classification is performed using neural network classifiers. The wide neural network proved to be the best-performing classifier, achieving 97.30% accuracy, as per the results depicted in Table 2. It took a minimum of 249.64 s and had the least false negative rate of 2.70%. The tri-layered neural network achieved the lowest accuracy of 65.10%, the highest process. The confusion matrix for a wide neural network is provided in Fig. 9. In this figure, the diagonal shows the correct prediction rate.

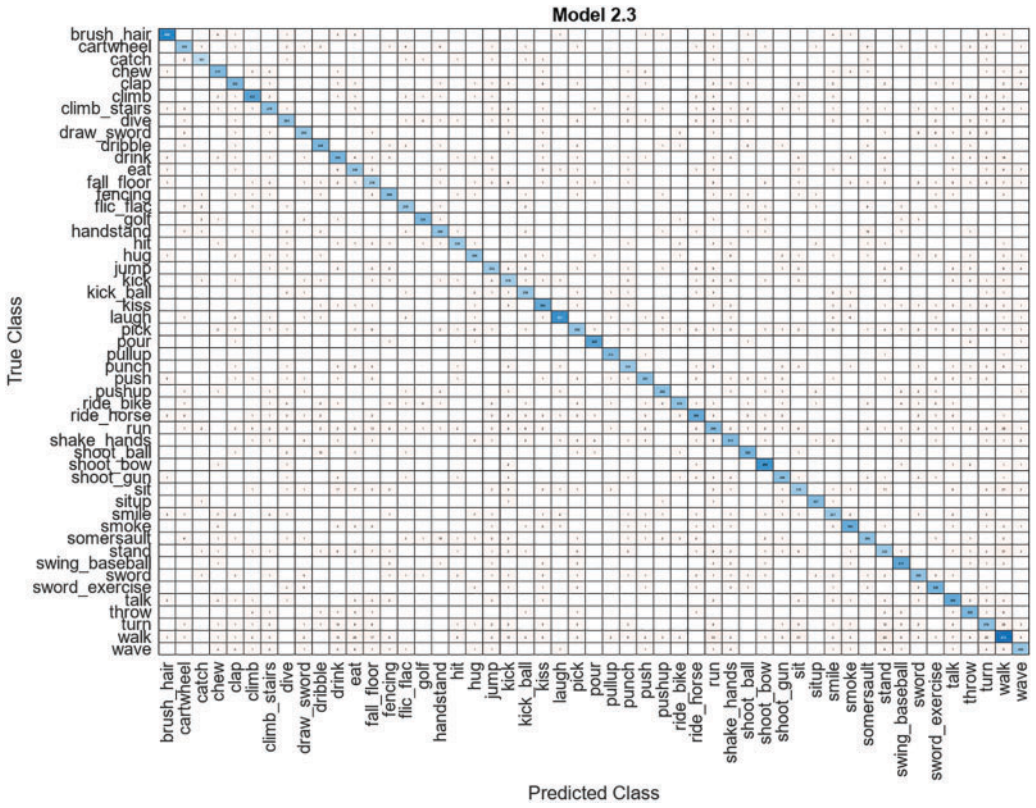

Figure 7: Confusion matrix for a wide neural network classifier

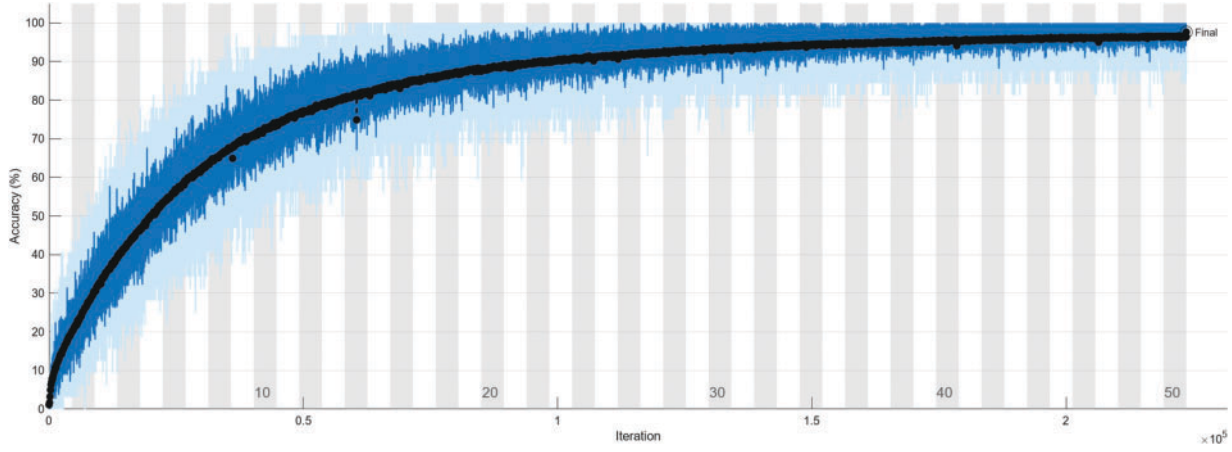

Figure 8: Training and loss graph for the UCF101 dataset

Table 2: Classification results of proposed architecture for UCF101 Dataset

Classifier	Accuracy (%)	Processing time (s)	False negative rate (%)
Wide NN	**97.30**	**249.64**	**2.70**

(Continued)

Table 2 (continued)

Classifier	Accuracy (%)	Processing time (s)	False negative rate (%)
Medium NN	90.90	859.54	9.10
Narrow NN	71.90	2002.80	28.10
Bi-Layered NN	69.40	2134.70	30.60
Tri-Layered NN	65.10	2138.70	34.90

Note: Bold denotes the best values.

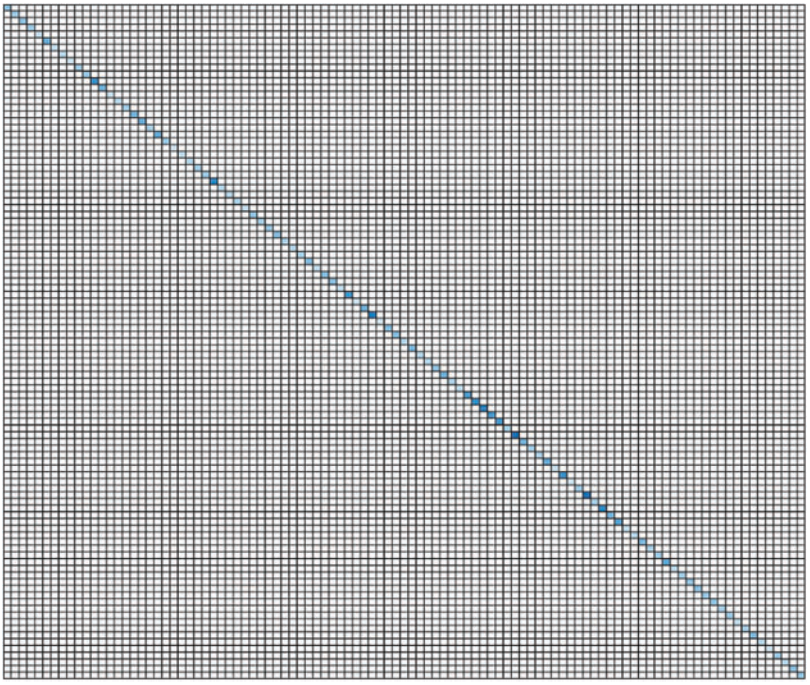

Figure 9: Confusion matrix for wide neural network classifier

6 Discussion

This section has conducted a detailed discussion of the proposed architecture, including ablation studies, comparisons with pre-trained CNN models, and comparisons with state-of-the-art techniques.

6.1 Ablation Studies

Training Accuracy based Analysis—In the first ablation study, we implemented several pre-trained models on the selected datasets using the transfer learning concept, whereas all the models were trained from scratch. After that, the accuracy of the training is noted. In the training process, we selected the same hyperparameters mentioned in Section 4.1.3. After that, the training accuracy is indicated for each model in Table 3. This table shows that the AlexNet model's training accuracy is 82.1% and 93.6% on the selected datasets HMDB51 and UCF101, respectively. After using Google, the accuracy improved by 82.6% and

93.6%, respectively. ResNet architectures' accuracy is improved to 85.9% and 95.8%. The training accuracy of EfficinetNetbo has improved by 87.3% and 97.3%, respectively, which shows this architecture's strength. In addition, we implemented vision transformer (ViT) architecture such as Tiny-16 and obtained an accuracy of 88.9% and 96.5%, respectively. However, the proposed CNN architecture obtained improved training accuracy of 90.6% and 98.2%, respectively, which is better than the listed models in this table.

Table 3: Analysis of proposed CNN model and pre-trained architectures based on the training accuracy on selected datasets

CNN model	HMDB51 accuracy (%)	UCF101 accuracy (%)
Proposed	**90.6**	**98.2**
AlexNet	82.1	93.6
GoogleNet	82.6	94.1
ResNet50	85.9	95.0
ResNet101	84.6	95.8
EfficientNetb0	87.3	97.3
DarkNet19	85.0	97.6
InceptionV3	87.0	97.0
ViT (Tiny-16)	88.9	96.5

Note: Bold denotes the best values.

Testing Accuracy-based Analysis—In the second ablation study, we tested the CNN models, which are presented in Table 3. In the testing process, we used the same testing set frames of the selected datasets and extracted deep features that were later passed to the WNN classifier. The results are given in Table 4. This table shows that the AlexNet model's obtained accuracy is 81.10% and 90.46%, which is further improved by ResNet50 architecture of 83.60% and 93.56%, respectively. The efficient net model obtained an improved testing accuracy of 84.86% and 95.64%, respectively. A ViT (tiny-16) was also tested on the WNN classifier and obtained better accuracy of 85.20% and 95.86%, respectively. The proposed architecture accuracy is 87.70% and 97.30%, which is enhanced than the listed models in this table.

Table 4: Analysis of the proposed CNN model and pre-trained architectures based on the testing accuracy on selected datasets, where we used WNN classifier

CNN model	HMDB51 accuracy (%)	UCF101 accuracy (%)
Proposed	87.70	97.30
AlexNet	81.10	90.46
GoogleNet	80.54	91.26
ResNet50	83.60	93.56
ResNet101	82.50	94.20
EfficientNetb0	84.86	95.64
DarkNet19	83.14	95.03
InceptionV3	84.04	95.17
ViT (Tiny-16)	85.20	95.86

Trainable Parameters based Analysis—This ablation study compared the proposed CNN architecture with pre-trained models based on trainable parameters. Fig. 10 shows the visual illustration of this comparison. This figure demonstrates that the minimum learnable parameters required by EfficientNetb0 and GoogleNet are 5.3 (Million) and 7.0 (Million), respectively. The ViT-16 required the highest number of trainable parameters at 86.8 (Million); however, the number of trainable parameters for the proposed architecture is 11.2 (Million). The proposed architecture is better than pre-trained models except for EfficientNetb0 and VIT based on the learnable parameters. In addition, the accuracy of the designed model is better than those of these pre-trained models on the selected datasets.

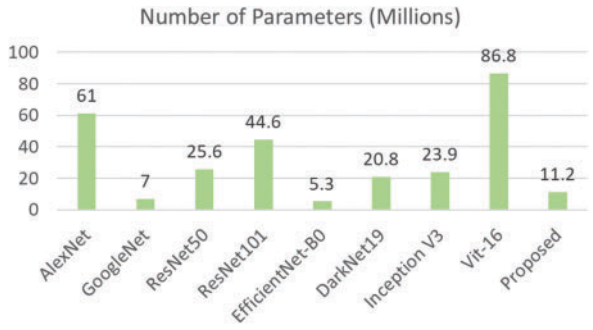

Figure 10: Comparison of the proposed and pre-trained models based on trainable parameters

Inner Residual Blocks based Analysis: In this ablation study, we analyze the performance of the proposed CNN model based on the number of designed blocks and self-attention module. Table 5 presents the output of this ablation study. In this table, initially, we added one residual block in the proposed model (Fig. 2) and performed training on the selected datasets. The obtained accuracy for this experiment is 76.20% and 89.24%, respectively. In the second experiment, we added two blocks and again performed training. The obtained accuracy value for this experiment is 78.54% and 91.36%, respectively. The accuracy gradually increased with the addition of more blocks in the main model and reached a maximum of seven blocks. After eight residual blocks, the accuracy gradually decreased to 89.16% and 96.52%, which was previously 90.10% and 98.20%. Moreover, we also experimented without adding any self-attention layer and obtained an accuracy of 85.34% and 94.80%, respectively. These ablation studies show that the proposed architecture obtained improved performance on seven designed blocks and the self-attention layer on the selected datasets.

Table 5: Analysis of proposed CNN architecture training accuracy based on the number of residual blocks and self-attention (SA) layer

CNN Model	HMDB51	UCF101
Proposed_1 Block + SA	76.20	89.24
Proposed_2 Blocks + SA	78.54	91.36
Proposed_3 Blocks + SA	80.34	91.50
Proposed_4 Blocks + SA	83.69	93.42
Proposed_5 Blocks + SA	85.60	94.89
Proposed_6 Blocks + SA	88.20	96.14
Proposed_7 Blocks + SA	**90.10**	**98.20**
Proposed_8 Blocks + SA	89.16	96.52
Proposed_7 Blocks	85.34	94.80

Note: Bold denotes the best values.

6.2 Comparison with State-of-the-Art (SOTA)

A comparison with existing techniques is performed in Table 4. Comparison is performed using both datasets. In Table 6, recent techniques such as Giveki [32] and Sowmyayani et al. [33] used HMDB51 datasets. They obtained 82.30% and 85.80% accuracy, respectively—the proposed model obtained 87.80% accuracy on this dataset, which shows a 2% improvement. For the UCF101 dataset, Dastbaravardeh et al. [37] and Xiong et al. [38] achieved an accuracy of 97.16% and 96.15%, which is further improved by the proposed architecture to 97.30%. Hence, it is observed that the proposed customized DCNN performed far better or had a comparable performance to these existing techniques. Lastly, the proposed CNN model's visual prediction is shown in Fig. 11.

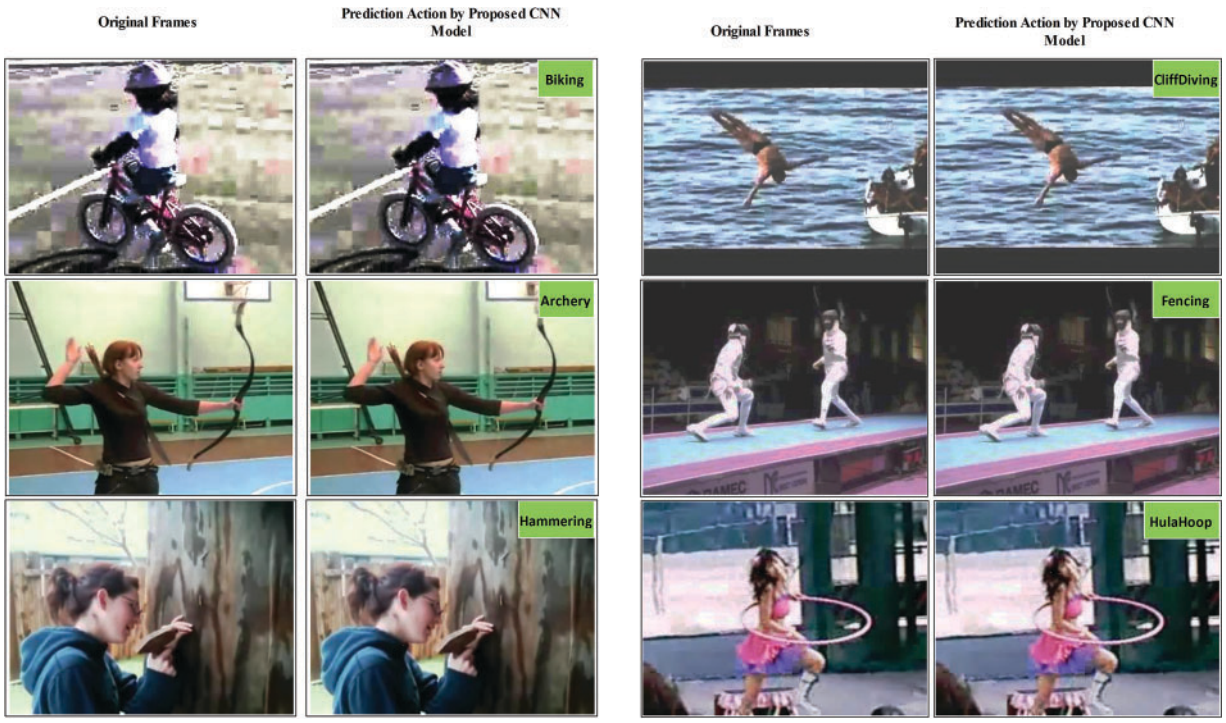

Figure 11: Proposed CNN model action prediction output

Table 6: Comparison with state-of-the-art techniques

Reference	Dataset	Year	Accuracy
He et al. [29]	HMDB51	2021	81.00%
Bao et al. [30]	HMDB51	2021	77.00%
Ahmad and Wu [31]	HMDB51	2023	80.61%
Giveki [32]	HMDB51	2024	82.30%
Sowmyayani et al. [33]	HMDB51	2024	85.80%
Proposed	**HMDB51**	**2025**	**87.70%**

(Continued)

Table 6 (continued)

Reference	Dataset	Year	Accuracy
Varshney and Bakariya [34]	UCF101	2022	97.20%
Yang and Zou [35]	UCF101	2022	92.70%
Gowada et al. [36]	UCF101	2023	96.30%
Dastbaravardeh et al. [37]	UCF101	2024	97.16%
Xiong et al. [38]	UCF101	2024	96.15%
Proposed	**UCF101**	**2025**	**97.30%**

Note: Bold denotes the best values.

7 Conclusion

Human action recognition has gained attention since the evolution of computer vision techniques. Video frames have complex features, so it is a challenging task to classify them from these frames. This study proposes a DCNN model that inputs video frames and classifies each frame into a corresponding class. In the proposed approach, there are seven blocks, and in each block, there are eight residual bottleneck sub-blocks; these are arranged in parallel, and the output of all these is added and provided to the next block. A self-attention layer is appended at the end of these seven blocks. Attention-based feature extraction is carried out and handed over to neural network classifiers. Comparison results with SOTA techniques show a significant improvement in performance measures. Overall, we first introduce new residual blocks with several filters that extract the more important features and are helpful for accurate action classification, even for large and complex datasets such as HMDB51. Secondly, self-attention layers after seven blocks provided better feature information; however, the number of residual blocks up to seven provides better convergence, which is degraded after the addition of a new block. Thirdly, BO-based hyperparameter selection shows better training of the proposed and pre-trained models than selection through random search.

There are a few limitations of this work, including the conversion of video datasets into video frames and the inclusion of several hidden layers inside the parallel blocks. The parallel blocks improved the performance; however, the weight layers increased the parameters that can be resolved in future studies. Future work will focus on developing a deep neural network that can uncover the intrinsic features of video frames in more generalizable settings. In addition, the network can be extended with a few dense ViT blocks to extract more valuable features of the video frames and decrease the weight layers.

Acknowledgement: Authors like to thank Princess Nourah bint Abdulrahman University Researchers Supporting Project number (PNURSP2025R719), Princess Nourah bint Abdulrahman University, Riyadh, Saudi Arabia and the Competitive Research Fund of The University of Aizu, Japan.

Funding Statement: Authors like to thank Princess Nourah bint Abdulrahman University Researchers Supporting Project number (PNURSP2025R719), Princess Nourah bint Abdulrahman University, Riyadh, Saudi Arabia. This work was supported by the National Research Foundation of Korea (NRF) grant funded by the Korea government (*MSIT) (No. 2018R1A5A7059549) and the Competitive Research Fund of The University of Aizu, Japan.

Author Contributions: The authors confirm contribution to the paper as follows: study conception and design, data collection, analysis and interpretation of results, software, draft manuscript preparation, funding: Yasir Khan Jadoon, Yasir Noman Khalid, Muhammad Attique Khan, Jungpil Shin. Review and writing, software, validation, visualization,

project administration: Fatimah Alhayan, Hee-Chan Cho, Byoungchol Chang. All authors reviewed the results and approved the final version of the manuscript.

Availability of Data and Materials: The datasets of this work are publically available for the research purposes. Here are the links of the datasets: https://www.crcv.ucf.edu/research/data-sets/ucf101/ and https://serre-lab.clps.brown.edu/resource/hmdb-a-large-human-motion-database/ (accessed on 15 July 2025).

Ethics Approval: Not applicable.

Conflicts of Interest: The authors declare no conflicts of interest to report regarding the present study.

References

1. Gupta N, Gupta SK, Pathak RK, Jain V, Rashidi P, Suri JS. Human activity recognition in artificial intelligence framework: a narrative review. Artif Intell Rev. 2022;55(6):4755–808. doi:10.1007/s10462-021-10116-x.
2. Khan SI, Dawood H, Khan MA, Issa GF, Hussain A, Alnfiai MM, et al. Transition-aware human activity recognition using an ensemble deep learning framework. Comput Hum Behav. 2025;162:108435. doi:10.1016/j.chb.2024.108435.
3. Pareek P, Thakkar A. A survey on video-based human action recognition: recent updates, datasets, challenges, and applications. Artif Intell Rev. 2021;54(3):2259–322. doi:10.1007/s10462-020-09904-8.
4. Beddiar DR, Nini B, Sabokrou M, Hadid A. Vision-based human activity recognition: a survey. Multimed Tools Appl. 2020;79(41):30509–55. doi:10.1007/s11042-020-09004-3.
5. Thakur D, Dangi S, Lalwani P. A novel hybrid deep learning approach with GWO-WOA optimization technique for human activity recognition. Biomed Signal Process Control. 2025;99:106870. doi:10.1016/j.bspc.2024.106870.
6. Bu X. Human motion gesture recognition algorithm in video based on convolutional neural features of training images. IEEE Access. 2020;8:160025–39. doi:10.1109/access.2020.3020141.
7. Yadav SK, Tiwari K, Pandey HM, Ali Akbar S. A review of multimodal human activity recognition with special emphasis on classification, applications, challenges and future directions. Knowl Based Syst. 2021;223:106970. doi:10.1016/j.knosys.2021.106970.
8. Bukht TFN, Rahman H, Shaheen M, Algarni A, Almujally NA, Jalal A. A review of video-based human activity recognition: theory, methods and applications. Multimed Tools Appl. 2025;84(17):18499–545. doi:10.1007/s11042-024-19711-w.
9. Mehmood F, Guo X, Chen E, Akbar MA, Khan AA, Ullah S. Extended multi-stream temporal-attention module for skeleton-based human action recognition (HAR). Comput Hum Behav. 2025;163:108482. doi:10.1016/j.chb.2024.108482.
10. Dang LM, Min K, Wang H, Jalil Piran M, Lee CH, Moon H. Sensor-based and vision-based human activity recognition: a comprehensive survey. Pattern Recognit. 2020;108:107561. doi:10.1016/j.patcog.2020.107561.
11. Wu H, Ma X, Li Y. Transformer-based multiview spatiotemporal feature interactive fusion for human action recognition in depth videos. Signal Process Image Commun. 2025;131:117244. doi:10.1016/j.image.2024.117244.
12. Zhang C, Xu Y, Xu Z, Huang J, Lu J. Hybrid handcrafted and learned feature framework for human action recognition. Appl Intell. 2022;52(11):12771–87. doi:10.1007/s10489-021-03068-w.
13. Escamilla Pinilla A. Motion-based feature analysis for the design of full-body interactions in the context of computer vision and large volume spaces [dissertation]. Barcelona, Spain: Universitat Oberta de Catalunya; 2024.
14. Saleh K, Szénási S, Vámossy Z. Generative adversarial network for overcoming occlusion in images: a survey. Algorithms. 2023;16(3):175. doi:10.3390/a16030175.
15. Elayaraja C, Rahila J, Velavan P, Rajest SS, Shynu T, Rahman MM. Depth sensing in AI on exploring the nuances of decision maps for explainability. In: Rajest SS, Moccia S, Singh B, Regin R, Jeganathan J, editors. Optimizing intelligent systems for cross-industry application. Hershey, PA, USA: IGI Global; 2024. p. 217–38. doi:10.4018/979-8-3693-8659-0.ch012.
16. Xin C, Kim S, Cho Y, Park KS. Enhancing human action recognition with 3D skeleton data: a comprehensive study of deep learning and data augmentation. Electronics. 2024;13(4):747. doi:10.3390/electronics13040747.

17. Rehman SU, Yasin AU, Haq EU, Ali M, Kim J, Mehmood A. Enhancing human activity recognition through integrated multimodal analysis: a focus on RGB imaging, skeletal tracking, and pose estimation. Sensors. 2024;24(14):4646. doi:10.3390/s24144646.
18. Morshed MG, Sultana T, Alam A, Lee YK. Human action recognition: a taxonomy-based survey, updates, and opportunities. Sensors. 2023;23(4):2182. doi:10.3390/s23042182.
19. Liu F, Wang C, Tian Z, Du S, Zeng W. Advancing skeleton-based human behavior recognition: multi-stream fusion spatiotemporal graph convolutional networks. Complex Intell Syst. 2024;11(1):94. doi:10.1007/s40747-024-01743-2.
20. Kaddar B, Fezza SA, Hamidouche W, Akhtar Z, Hadid A. On the effectiveness of handcrafted features for deepfake video detection. J Electron Imaging. 2023;32(5):053033. doi:10.1117/1.jei.32.5.053033.
21. Nguyen HC, Nguyen TH, Scherer R, Le VH. Deep learning for human activity recognition on 3D human skeleton: survey and comparative study. Sensors. 2023;23(11):5121. doi:10.3390/s23115121.
22. Salehi AW, Khan S, Gupta G, Alabduallah BI, Almjally A, Alsolai H, et al. A study of CNN and transfer learning in medical imaging: advantages, challenges, future scope. Sustainability. 2023;15(7):5930. doi:10.3390/su15075930.
23. Liu Y, Pu H, Sun DW. Efficient extraction of deep image features using convolutional neural network (CNN) for applications in detecting and analysing complex food matrices. Trends Food Sci Technol. 2021;113:193–204. doi:10.1016/j.tifs.2021.04.042.
24. An S, Bhat G, Gumussoy S, Ogras U. Transfer learning for human activity recognition using representational analysis of neural networks. ACM Trans Comput Healthcare. 2023;4(1):1–21. doi:10.1145/3563948.
25. Lacroix K, Gholamiangonabadi D, Luisa Trejos A, Grolinger K. Deep transfer learning for detection of upper and lower body movements: transformer with convolutional neural network. IEEE Sens J. 2024;24(20):33778–90. doi:10.1109/JSEN.2024.3451291.
26. Ren B, Liu M, Ding R, Liu H. A survey on 3D skeleton-based action recognition using learning method. Cyborg Bionic Syst. 2024;5:0100. doi:10.34133/cbsystems.0100.
27. Kumar P, Chauhan S, Awasthi LK. Human activity recognition (HAR) using deep learning: review, methodologies, progress and future research directions. Arch Comput Meth Eng. 2024;31(1):179–219. doi:10.1007/s11831-023-09986-x.
28. Jameer S, Syed H. A DCNN-LSTM based human activity recognition by mobile and wearable sensor networks. Alex Eng J. 2023;80:542–52. doi:10.1016/j.aej.2023.09.013.
29. He JY, Wu X, Cheng ZQ, Yuan Z, Jiang YG. DB-LSTM: densely-connected bi-directional LSTM for human action recognition. Neurocomputing. 2021;444:319–31. doi:10.1016/j.neucom.2020.05.118.
30. Bao W, Yu Q, Kong Y. Evidential deep learning for open set action recognition. In: Proceedings of the 2021 IEEE/CVF International Conference on Computer Vision (ICCV); 2021 Oct 10–17; Montreal, QC, Canada. doi:10.1109/ICCV48922.2021.01310.
31. Ahmad T, Wu J. SDIGRU: spatial and deep features integration using multilayer gated recurrent unit for human activity recognition. IEEE Trans Comput Soc Syst. 2024;11(1):973–85. doi:10.1109/TCSS.2023.3249152.
32. Giveki D. Human action recognition using an optical flow-gated recurrent neural network. Int J Multimed Inf Retr. 2024;13(3):29. doi:10.1007/s13735-024-00338-4.
33. Sowmyayani S, Vivek V, Arulpandy P, Tamilselvi S. Frame differencing based temporal feature extraction in human action recognition. J Comput Anal Appl. 2024;33(5):549–57.
34. Varshney N, Bakariya B. Deep convolutional neural model for human activities recognition in a sequence of video by combining multiple CNN streams. Multimed Tools Appl. 2022;81(29):42117–29. doi:10.1007/s11042-021-11220-4.
35. Yang G, Zou WX. Deep learning network model based on fusion of spatiotemporal features for action recognition. Multimed Tools Appl. 2022;81(7):9875–96. doi:10.1007/s11042-022-11937-w.
36. Gowada R, Pawar D, Barman B. Unethical human action recognition using deep learning based hybrid model for video forensics. Multimed Tools Appl. 2023;82(19):28713–38. doi:10.1007/s11042-023-14508-9.
37. Dastbaravardeh E, Askarpour S, Saberi Anari M, Rezaee K. Channel attention-based approach with autoencoder network for human action recognition in low-resolution frames. Int J Intell Syst. 2024;2024:1052344. doi:10.1155/2024/1052344.

38. Xiong X, Min W, Han Q, Wang Q, Zha C. Action recognition using action sequences optimization and two-stream 3D dilated neural network. Comput Intell Neurosci. 2022;2022:6608448. doi:10.1155/2022/6608448.
39. Kuehne H, Jhuang H, Garrote E, Poggio T, Serre T. HMDB: a large video database for human motion recognition. In: Proceedings of the 2011 International Conference on Computer Vision; 2011 Nov 6–13; Barcelona, Spain. doi:10.1109/ICCV.2011.6126543.
40. Soomro K. UCF101: a dataset of 101 human actions classes from videos in the wild. arXiv: 1212.0402. 2012.
41. He K, Zhang X, Ren S, Sun J. Deep residual learning for image recognition. In: Proceedings of the 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR); 2016 Jun 27–30; Las Vegas, NV, USA. doi:10.1109/CVPR.2016.90.
42. Albarakati HM, Khan MA, Hamza A, Khan F, Kraiem N, Jamel L, et al. A novel deep learning architecture for agriculture land cover and land use classification from remote sensing images based on network-level fusion of self-attention architecture. IEEE J Sel Top Appl Earth Obs Remote Sens. 2024;17:6338–53. doi:10.1109/jstars.2024.3369950.
43. Vaswani A, Shazeer N, Parmar N, Uszkoreit J, Jones L, Gomez AN, et al. Attention is all you need. In: Proceedings of the Neural Information Processing Systems 30 (NIPS 2017); 2017 Dec 4–9; Long Beach, CA, USA.
44. Snoek J, Larochelle H, Adams RP. Practical bayesian optimization of machine learning algorithms. In: Proceedings of the Neural Information Processing Systems 25 (NIPS 2012); 2012 Dec 3–6; Lake Tahoe, NV, USA.
45. Mirzadeh SI, Chaudhry A, Yin D, Hu H, Pascanu R, Gorur D, et al. Wide neural networks forget less catastrophically. Proc Mach Learn Res. 2022;162:15699–717.
46. Goodfellow I, Bengio Y, Courville A. Deep learning. Cambridge, MA, USA: MIT Press; 2016. 800 p.
47. Karim M, Khalid S, Aleryani A, Tairan N, Ali Z, Ali F. HADE: exploiting human action recognition through fine-tuned deep learning methods. IEEE Access. 2024;12:42769–90. doi:10.1109/access.2024.3378515.
48. Ami AS, Moran K, Poshyvanyk D, Nadkarni A. "False negative—that one is going to kill you": understanding industry perspectives of static analysis based security testing. In: Proceedings of the 2024 IEEE Symposium on Security and Privacy (SP); 2024 May 19–23; San Francisco, CA, USA. doi:10.1109/SP54263.2024.00019.