Computers, Materials & Continua

Tech Science Press

DOI: 10.32604/cmc.2024.049512

ARTICLE

Workout Action Recognition in Video Streams Using an Attention Driven Residual DC-GRU Network

Arnab Dey[1,*], Samit Biswas[1] and Dac-Nhuong Le[2]

[1]Department of Computer Science and Technology, Indian Institute of Engineering Science and Technology, Shibpur, Howrah, 711103, India

[2]Faculty of Information Technology, Haiphong University, Haiphong, 180000, Vietnam

*Corresponding Author: Arnab Dey. Email: arnabdeycs@gmail.com

Received: 09 January 2024 Accepted: 09 April 2024 Published: 15 May 2024

ABSTRACT

Regular exercise is a crucial aspect of daily life, as it enables individuals to stay physically active, lowers the likelihood of developing illnesses, and enhances life expectancy. The recognition of workout actions in video streams holds significant importance in computer vision research, as it aims to enhance exercise adherence, enable instant recognition, advance fitness tracking technologies, and optimize fitness routines. However, existing action datasets often lack diversity and specificity for workout actions, hindering the development of accurate recognition models. To address this gap, the Workout Action Video dataset (WAVd) has been introduced as a significant contribution. WAVd comprises a diverse collection of labeled workout action videos, meticulously curated to encompass various exercises performed by numerous individuals in different settings. This research proposes an innovative framework based on the Attention driven Residual Deep Convolutional-Gated Recurrent Unit (ResDC-GRU) network for workout action recognition in video streams. Unlike image-based action recognition, videos contain spatio-temporal information, making the task more complex and challenging. While substantial progress has been made in this area, challenges persist in detecting subtle and complex actions, handling occlusions, and managing the computational demands of deep learning approaches. The proposed ResDC-GRU Attention model demonstrated exceptional classification performance with 95.81% accuracy in classifying workout action videos and also outperformed various state-of-the-art models. The method also yielded 81.6%, 97.2%, 95.6%, and 93.2% accuracy on established benchmark datasets, namely HMDB51, Youtube Actions, UCF50, and UCF101, respectively, showcasing its superiority and robustness in action recognition. The findings suggest practical implications in real-world scenarios where precise video action recognition is paramount, addressing the persisting challenges in the field. The WAVd dataset serves as a catalyst for the development of more robust and effective fitness tracking systems and ultimately promotes healthier lifestyles through improved exercise monitoring and analysis.

KEYWORDS

Workout action recognition; video stream; action recognition; residual network; GRU; attention

1 Introduction

Regular workouts play a vital role in upholding a well-balanced lifestyle, as they promote a robust metabolism, contribute to weight management [1], improve cardiovascular health [2], and preserve muscle mass, ultimately contributing to overall fitness [3]. The recognition of actions holds significant importance in the domain of computer vision as it enables the identification and categorization of human or object actions within video sequences. With the rise of digital video data, accurate action recognition has become increasingly important in areas like surveillance, sports analysis, monitoring workout routines, and robotics. Traditional methods relied on handcrafted features and shallow classifiers, leading to constraints in accuracy and scalability. However, deep learning has revolutionized computer vision, enabling more efficient and precise action recognition [4] systems. This article focuses on applying deep learning-based techniques to identify workout actions in short video sequences, offering immediate insights into exercise form and optimizing workout routines. Many people do exercises without professional guidance, leading to improper movements and hindering progress. Workout action recognition in videos offers advantages such as personalized guidance and motivation, empowering individuals to reach their fitness goals and maintain an active lifestyle. Especially in situations like the pandemic, when access to in-person trainers and gym facilities may be limited, workout action recognition serves as a dynamic virtual trainer, providing guidance and support for individuals practicing their workouts. A significant advantage of workout action recognition lies in its ability to identify and monitor workout routines. Workout action recognition in videos becomes an invaluable tool when access to physical trainers or fitness classes is restricted, enabling individuals to continue their exercise routines at home while receiving support.

In the realm of action recognition, the implementation of sensor-based approaches has been associated with significant costs and technical complexities. These methods often require specialized hardware, intricate setups, and substantial financial investments, posing barriers to widespread adoption and scalability. In light of these challenges, we advocate for a paradigm shift towards vision-based approaches. Leveraging the advancements in computer vision and deep learning, vision-based methodologies offer a more cost-effective and accessible alternative for action recognition tasks. The main benefit of deep learning-based approaches resides in their capability to autonomously acquire representations from raw data, eliminating the necessity for handcrafted features. While deep learning has greatly improved action recognition, existing approaches, such as skeleton-based methods and background subtraction-based techniques, have their limitations. Skeleton-based approaches rely on skeletal joint information, which may not fully capture fine-grained movements or subtle variations in workout actions, leading to limited accuracy. Background subtraction methods, while useful for detecting foreground objects, may struggle with complex scenes or dynamic backgrounds common in workout environments [5]. Additionally, most of the advanced deep learning methods are computationally very expensive, which may pose challenges in real-time processing or deployment on resource-constrained devices, limiting their practical applicability. Despite the remarkable strides made in this domain, certain challenges persist, serving as focal points for ongoing research. Foremost among these challenges is the capacity to identify nuanced and intricate actions that entail the coordination of multiple body parts and complex movements [6]. Notably, the categorization of workout training poses unique challenges due to its inherent complexity, subtle variations, and intricate nature. This research aims to create efficient and accurate deep-learning setups specifically designed for recognizing workout actions in video streams. In this work, we propose novel approaches to modelling spatiotemporal patterns, leveraging a framework that combines Time Distributed Convolution, Dilated Residual convolution, Attention mechanism and Gated Recurrent Unit (GRU) and requires only 0.405 million trainable parameters which can run seamlessly on resource-constrained devices. The proposed method

leverages comparable or even superior performance than various baseline models in action recognition tasks. A novel dataset comprising about 1805 workout videos has been developed in response to the lack of accessible standard workout action video datasets. The proposed Attention driven Residual Deep Convolution-GRU (ResDC-GRU Attention) model is trained with our prepared WAVd dataset, and it mainly classifies the workout videos into eleven categories: *NeckRotation, PunchingBag, HandstandPushups, Pushups, WallPushups, Pullups, JumpingJack, Skipping, Squats, WeightLifting, and GluteBridges*.

The primary contributions of this research encompass:

(1) Development of a new video dataset for Workout Actions (WAVd) which includes meticulously labeled data. This dataset fills a gap in the existing resources by focusing specifically on workout actions.

(2) Introduction of a novel approach for recognizing workout actions in videos, achieved through the proposed Attention-based ResDC-GRU network. This network integrates convolution, sequential attention mechanism with residual dilated convolution and GRU layers, facilitating dynamic frame emphasis and precise spatiotemporal learning for superior video classification.

(3) Evaluation of the proposed model on four benchmark action datasets: HMDB51, Youtube Actions, UCF50, and UCF101, achieving accuracies of 81.6%, 97.2%, 95.67%, and 93.2%, respectively. Its efficacy is thoroughly assessed through comparison with various baseline models.

(4) Utilization of a Custom Weighted Categorical Cross-Entropy (WCCE) loss function to compute the loss during the training process. This tailored loss function aids in optimizing the model's performance and ensuring effective learning from the dataset.

2 Related Works

Action recognition research can be broadly categorized into two main approaches: Approaches that rely on handcrafted features and approaches that leverage deep learning-based features. These categories have been explored and discussed in the literature [4,5]. Handcrafted features-based approaches involve the utilization of manually designed feature descriptors to identify spatial and temporal variations within videos. A study by Peng et al. [6] introduced a novel efficient representation known as the hybrid supervector that combines different Bag of Visual Words (BoVW) frameworks with improved dense trajectories. Yasin et al. [7] used key frame selection to build a crucial technique for action identification in video sequences, which was subsequently applied to the action recognition. These techniques require a significant amount of time and necessitate an end-to-end recognition strategy, heavily relying on human involvement to extract generic and discriminative features. In contrast, deep learning-based methods that leverage learned features have emerged as the predominant approach to tackle the complexities associated with recognizing human actions in videos. Unlike handcrafted features, deep learning models are capable of extracting high-level representations directly from raw video frames, eliminating the need for manual feature engineering. Feichtenhofer et al. [8] introduced a method for identifying actions that utilize spatial and motion data. They have suggested a two-stream Convolutional Neural Network (CNN) model for integrating both spatial and temporal streams, where RGB information (spatial) and optical flow (motion) are modelled separately, and prediction outcomes are averaged in the ultimate layers. Because of optical flow, this network fails to capture long-term motion. Additionally, the spatial CNN stream's performance depends only on one randomly chosen image from the input video. Ullah et al. [9] introduced a unique method for recognizing human actions in video sequences by combining a two-stream convolutional neural net

architecture with a multilayer Long Short Term Memory Module (LSTM). Their method extracts relevant spatial and optical flow features from the videos and employs a multilayer LSTM to accurately identify human actions in the video sequences. Xin et al. [10] proposed a new adaptive recurrent-convolutional hybrid model to handle the challenges of long-term action recognition. This cutting-edge approach proficiently handles variations in both spatial and temporal streams, and also the diverse nature of intra- and inter-class actions. Rangari et al. [11] presented a dynamic solution for identifying six different exercise poses utilizing body skeleton estimation and LSTM based classifier. Verdú et al. [12] have proposed Conv-GRU model to identify anomalies in video streams. Dai et al. [13] have utilized a two-stream attention-based LSTM model. The model consisted of a LSTM stream for encoding temporal sequences and a CNN-LSTM stream for capturing spatial-temporal information. Furthermore, the spatio-temporal attention mechanism was incorporated into the feature learning process to further enhance action recognition at each step of the LSTM. Caballero et al. [14] devised a real-time human action detection method using 3D-CNN, effectively extracting spatio-temporal patterns from unprocessed depth layers and classifying activities based on this data. The proposed 3DFCNN algorithm designed and optimized to achieve high accuracy in event recognition industry and reduce computation costs. Training 3DFCNN models can be computationally intensive and time-consuming due to the large volume of spatio-temporal data. 3DFCNN may struggle with accurately detecting human behavior in the presence of noise or occlusions in the depth layers, leading to misclassifications. Zheng et al. [15] introduced Dynamic Sampling Networks (DSN) setting the backbone network as ResNet-34 to improve video event recognition. DSN has a dynamic sampling and classification module that targets to select the most informative clips and train a clip-level classifier to detect actions and sampling policy to learn which clips should be accessed. Varol et al. [16] have incorporated a long-term transition (LTC) network. It is suggested that the event detection accuracy can be enhanced by using LTC-CNN models with long time extension. They have achieved 67.2% accuracy with HMDB51 data and 92.7% accuracy with UCF-101 data. Verma et al. [17] proposed a new human activity recognition (HAR) method that combines RGB and skeletal data. Their approach consisted of generating motion posture images and skeletal images from RGB video and skeletal information, respectively. These images were then processed using a fusion of CNN and LSTM nets to capture and efficiently capture both spatial and temporal features. The utilization of motion posture and skeletal images can introduce biases in accurately depicting complex human actions, especially in scenarios with occlusions or diverse environmental conditions, while also imposing significant computational complexity. In another study, Zhou et al. [18] have introduced a novel method called Multi-head Attention driven Two-stream EfficientNet (MAT-EffNet) to recognize human actions. This method addresses the difficulty of recognizing similar actions by highlighting crucial information about the actions in various frames. The MAT-EffNet architecture comprises of two streams for spacio-temporal feature extraction from consecutive frames with EfficientNet-B0. The extracted features are then processed using a multi-head attention method to capture essential information about the actions. In the end, a late average fusion method is employed to get the ultimate prediction. While EfficientNet is relatively efficient compared to larger models, the overall architecture with multi-head attention is computationally very expensive and require significant computational resources. Wang et al. [19] have attained action recognition through the adoption of the Temporal Segment Network (TSN). This innovative methodology harnesses distinctive segment-based sampling and aggregation strategies, effectively emulating the intricate patterns and interrelations spanning extended temporal sequences. By partitioning input videos into discrete segments, TSNs perform classification on every segment separately. To derive the final output, the classification scores derived from all the individual segments are fused. TSNs can be computationally expensive since they require processing multiple segments of the input video separately, which can limit their use in real-time applications or

on devices with limited computational resources. Luo et al. [20] employed Dense Semantics-Assisted Networks (DSA-CNNs) to enhance activity recognition in videos. By integrating dense semantic segmentation masks, encoding rich semantic information, they improve network learning, notably boosting action recognition. However, it requires additional semantic information for training the network, which may not always be available or feasible to obtain. The semantic information is obtained from external sources, such as object detection models, which can be computationally expensive and may not generalize well to different types of videos. Gazron et al. [21] presented a rapid action recognition method relying on motion trajectory occurrences. Their approach combines local and mid-level analysis, employing a spatial point process to statistically model the distribution of active points surrounding motion trajectories. By prioritizing neighboring points, it captures local motion distribution effectively. However, it lack robustness in complex scenarios due to its reliance on motion trajectory occurrences. In our prior research focused on recognizing workout actions from images [22], we attained 92.75% validation accuracy on the collected Workout Action Image dataset (WAId) utilizing the suggested WorkoutNet model and later also proposed an Attention driven DC-GRU Network [23] to recognize umpire actions in Cricket game. Akbar et al. [24] have introduced HybridHR-Net which is a neural network model based on EfficientNet-B0 and Entropy controlled optimization technique to identify actions. However, limitations of this study include potential biases in the training data, generalizability issues across diverse action scenarios, and computational resource requirements. In another work [25], we have introduced a novel multi-level attention based AdaptiveDRNet model to recognize various human interactions in images. Wei et al. [26] integrated a self-attention mechanism into graph convolutional neural networks (GCNN) to effectively recognize yoga movements. In another study, Dey et al. [27] proposed an Attention driven AdaptSepCX Network to identify student actions from images. Altaf et al. [28] introduced a convolution-free approach for highly accurate human activity recognition (HAR) is introduced. This approach excels in overcoming prior challenges by adeptly encoding relative spatial information. Leveraging a pre-trained Vision Transformer, it extracts frame-level features, which then passed through a multilayer LSTM to capture intricate dependencies in surveillance videos. However, Vision Transformer and multilayer LSTM architecture demand significant computational resources, limiting practicality in resource-constrained settings. Interpreting learned representations and diagnosing errors pose additional challenges, constraining potential refinement or optimization. In another study, Zhang et al. [29] introduced a novel Physical Fitness Action dataset targeting the identification of three fundamental exercises namely sit-ups, pull-ups, and push-ups. Their approach leveraged an Attention-based Two-Branch Multi-stage CNN and LSTM network architecture for precise classification of these exercise actions. Youssef et al. [30] have proposed a virtual coaching system to identify four distinct workout action categories, specifically free squats, shoulder presses, push-ups, and lunges. Chen et al. [31] devised a sophisticated CNN-LSTM model tailored to discern traditional Chinese exercises from recorded video streams, demonstrating promising results in exercise recognition.

Established benchmark datasets such as KTH [32], HSiPu2 [29], HMDB51 [33], Youtube Actions [34], UCF50 [34], and UCF101 [34], have played a significant role in advancing action recognition research. However, these openly accessible datasets consist of general action categories and lack specific action types. While they encompass numerous actions, the sample size for each action class is typically limited. On the other hand, some datasets may offer a large number of action classes, but with only a few video samples per class. This limited sample size per class can pose challenges for models aiming to generalize and accurately recognize actions in real-world settings. The lack of diverse and sufficient data can hinder the development of robust and effective action recognition models. Recognizing this limitation, we have proactively embarked on developing a dedicated dataset

specifically focused on workout actions. This WAVd dataset encompasses eleven diverse workout activities, providing researchers with comprehensive samples for model development and evaluation. It aims to improve the generalization of action recognition models, specifically targeting workout-related actions in videos.

3 Proposed Method

The method started by collecting and trimming workout videos according to the specific workout actions performed. To enhance the dataset, data augmentation techniques were applied. Following this, a frame extraction function was utilized, that receives the video path as input and meticulously extract individual frames. This function seamlessly employs a Video Capture object, efficiently reading the video and accurately determining the total frame count. It then computes the intervals for frame addition to the frames list, meticulously ensuring extraction aligns with the prescribed sequence length. Each frame is resized to a predetermined height and width during the extraction process. Subsequently, normalization takes place by dividing the pixel values of the frames by 255, which scales them to a range between 0 and 1. After normalization, the frames are added to the frames list. Once the extraction and processing are complete, the Video Capture object is released, and the frames list encompassing the extracted and normalized frames is returned as the final output. The prepared WAVd dataset undergoes partitioning into distinct training and validation subsets. The next step involves training the proposed ResDC-GRU Attention model using callback functions. During this process, video features are extracted. Subsequently, the model's final dense layer equipped with Softmax activation is employed to classify Workout actions. The ultimate trained model is then employed to identify workout actions in video streams.

The overall layout of the proposed method is depicted in Fig. 1. Notably, this research prioritizes resource-friendliness by evaluating it without utilizing Graphics Processing Unit (GPU). Validation with real-world videos confirms the robust performance of the proposed technique in accurately identifying workout actions. The subsequent sections provide detailed explanations for: a) Preparation of the Train dataset b) ResDC-GRU Attention: Model Architecture, c) Loss Function.

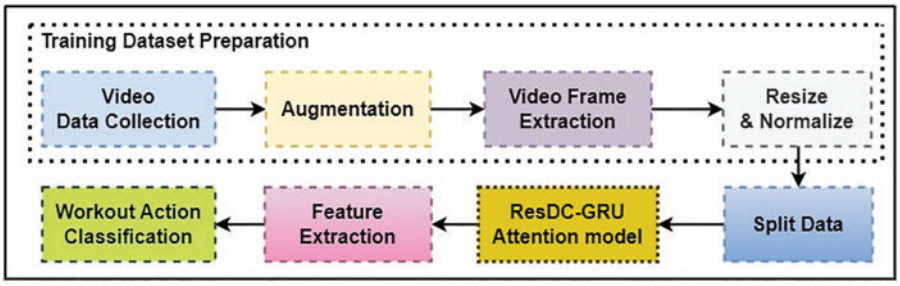


Figure 1: Workflow diagram of the proposed method

3.1 Training Dataset Preparation

Initially, the WAVd dataset was collected and RGB videos were resized to dimensions of 320 × 240. Deep learning models require substantial training data, so to boost the performance of the ResDC-GRU Attention model, data augmentation was employed. This not only expanded the training dataset size but also mitigated overfitting. Augmentation included random alterations like 0–0.15 zoom range, 0–10 degree rotation, brightness in the range of 0.2 to 0.4, and contrast between 0.3

and 0.4. Augmented video data, generated using the Moviepy Python library, increased dataset size by introducing variations. This process increased training data by randomly transforming videos, featuring the model to a broader spectrum of potential data distribution patterns. To facilitate training and testing, the WAVd was partitioned into two subsets. Specifically, 79% of the video data was designated for training purposes, with the remaining 21% earmarked for testing.

3.2 ResDC-GRU Attention: Model Architecture

The proposed model comprises of multiple layers performing different operations to extract features from sequential image data. The model consists of TimeDistributed (T.D.) Conv2D layers for spatial feature extraction, TimeDistributed (T.D.) MaxPooling2D layers for downsampling, Dropout layers for regularization, a TimeDistributed Flatten layer for reshaping, GRU layers for capturing temporal dependencies, a Dropout layer for additional regularization, and a dense layer for final classification output. The proposed ResDC-GRU Attention model comprises 18 layers, considering the Residual block, Attention block and the GRU block as an individual layer. These layers collectively facilitate the model to extract and learn relevant features from sequential frame data, leading to improved performance in tasks such as action recognition or video classification.

We first define the input layer with the specified shape. Input Layer Shape comprises the Sequence length, Frame height, Frame width and the number of channels, i.e., 15, 64, 64, 3. Apply a series of convolutional layers with max pooling and dropout, using the TimeDistributed wrapper to process each frame in the sequence independently thereby allowing the network to learn spatial features from each frame. Apply two dilated residual (Di-Residual) blocks using the Residual Block function. The Di-Residual block comprises of two dilated convolutional layers, one batch normalization layer, a 1×1 convolutional layer for adjusting the shape of inputs before the residual operation, and an element-wise addition with the residual. The activation function used is the Exponential Linear Unit (ELU). Flatten the output of the convolutional layers. And then, an Attention mechanism is incorporated. Following this, three GRU layers with 64 units each are introduced, where the result of the first and second GRU layer is concatenated and passed through the third GRU layer and the third GRU layer does not return sequences. Apply dropout regularization. Then a fully connected output layer with the softmax activation enables multi-class classification. The proposed Attention-based ResDC-GRU network is implemented with a Custom-Weighted Categorical Cross-Entropy loss function and Adam optimizer initialized with a learning rate of 0.0001. The proposed model adopted for workout action recognition in video streams is depicted in Fig. 2. The proposed model boasts exceptional efficiency, demanding a mere 0.405 million (M) trainable parameters. This remarkable characteristic underscores its lightweight design, making it exceedingly well-suited for deployment on resource-constrained devices, where computational resources are limited.

Time Distributed Convolutional Layers: The model starts with a Time Distributed (T.D.) layer that applies a 2D convolutional operation to the input sequence of frames. The initial T.D. convolutional layer consists of 16 filters and the next T.D. convolution layer comprises of 32 filters, and both are having (3, 3) kernel size. To introduce non-linearity, the ELU activation function is employed. The 'same' padding parameter is employed to guarantee that the output feature maps retain the exact spatial dimensions as that of the input. The initial Time Distributed convolutional layers capture spatial features from each frame of the input sequence. This is achieved through a series of convolutional operations followed by an activation function. The utilization of multiple convolutional layers with increasing number of filters allows the model to learn hierarchical representations of the spatial features. The convolution operation in spatial domain is represented as depicted in Eq. (1). In Eq. (1), the activation function f is applied to the sum of the element-wise product between the

convolutional kernel weights ($w_{x,y}$) and the corresponding input matrix elements ($x_{p+s,q+t}$), which are then summed over the spatial dimensions. The bias term (b) is added to this sum to produce the final activation ($a_{p,q}$).

$$a_{p,q} = f\left(\sum_{s=1}^{S}\sum_{t=1}^{T} w_{x,y} \cdot x_{p+s,q+t} + b\right) \tag{1}$$

The ELU activation function is represented as depicted in Eq. (2). Here, z denotes the input to activation function and α denotes a positive constant.

$$f(z) = \begin{Bmatrix} z; & for\ z \geq 0 \\ \alpha\{e^{z} - 1\}; & for\ z < 0 \end{Bmatrix} \tag{2}$$

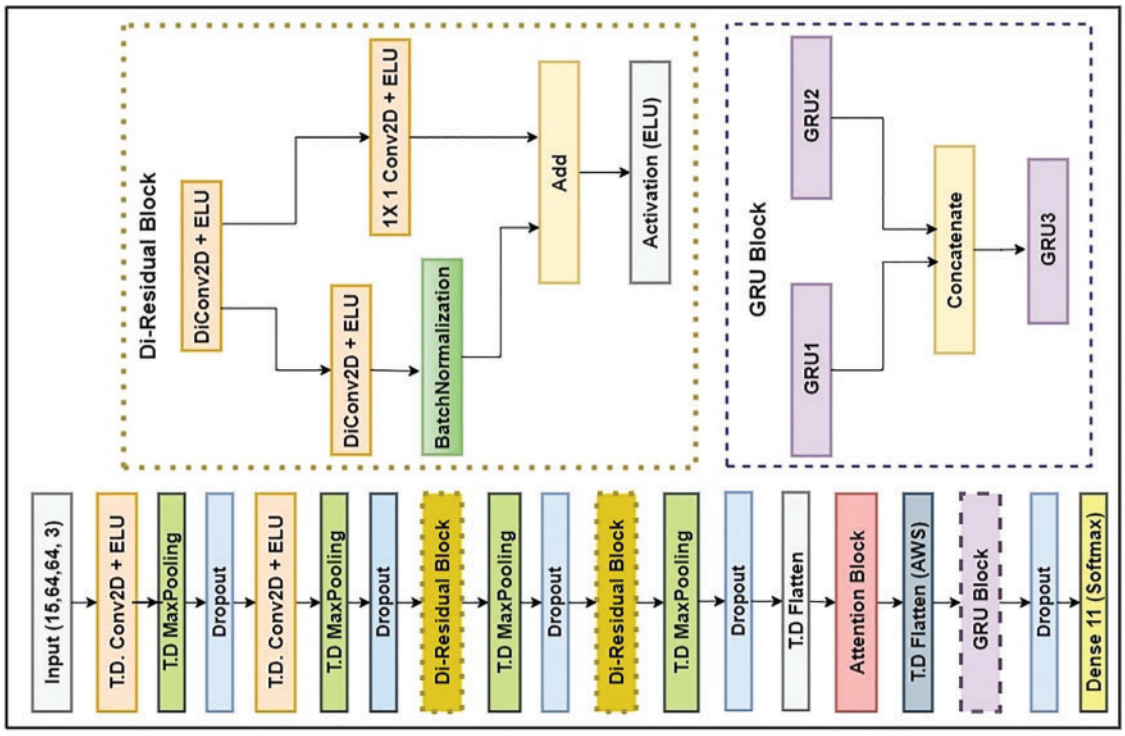


Figure 2: Proposed ResDC-GRU attention model architecture

Time Distributed Max Pooling Layers: Time Distributed (T.D) MaxPooling2D layer is employed after each T.D convolutional layer. Max pooling (MP) serves the purpose of reducing the spatial dimensions of feature maps by selecting the maximum value within a specified pooling window. Here, a pooling window size of (4, 4) is employed in the two initial T.D convolution layers, resulting in downsampling of the feature maps in both height and width dimensions and a (2, 2) pool size is employed after every Residual blocks. This aids in capturing important features while simultaneously reducing the spatial resolution. Subsequently, in Layer 2, max pooling is applied to reduce the dimensions of the feature maps, effectively downsizing them while retaining crucial information. Layer 3 introduces dropout (rate 0.10), a regularization technique, which randomly deactivates units to prevent overfitting during training. The subsequent Layer 4 employs another convolutional

block to further enhance feature extraction. This is followed by Layer 5, which performs additional downsampling using max pooling.

Dropout Layers: The Layer 3 and Layer 6 introduce dropout to regularize the model. Dropout layers are strategically incorporated after each pooling layer to introduce regularization to the model. During the training process, dropout randomly deactivates a fraction of the input units by setting them to zero. This helps prevent overfitting by diminishing the tendency of neurons to overly depend on each other and promoting more robust learning. Here, a dropout rate of 0.13 is added after the T.D MaxPooling (Layer 5).

Dilated Residual Blocks: The dilated residual (Di-Residual) block comprises of two dilated convolution layers (dilation rate 2), one batch normalization layer, a 1×1 convolutional layer for adjusting the shape of inputs before the residual operation, and an element-wise addition with the residual. The Di-Residual Block function applies dilated convolution layers with ELU activation and batch normalization. Dilated convolution layers increase the pixel gaps between kernel elements to capture broader spatial context without increasing the parameters count. In ResDC-GRU network, the dilated convolution operation is mathematically expressed as illustrated in Eq. (3). Let us denote the input sequence as x with spatial dimensions $H_{in} \times W_{in}$ and C_{in} channels. The filter (or kernel) is denoted as w with dimensions K × K and c_{in} channels. The output of the dilated convolution is denoted as y with dimensions $H_{out} \times W_{out}$ and c_{out} channels. Here, r is the dilation rate. The indices i, j iterate over the spatial dimensions of the output, c_{in} iterates over input channels, and c_{out} iterates over output channels.

$$y_{i,j,c_{out}} = \sum_{k=1}^{K} \sum_{l=1}^{K} \sum_{c_{in}=1}^{C_{in}} x_{(i+r.(k-1)),(j+r.(l-1)),c_{in}} \cdot w_{k,l,c_{in},c_{out}} \tag{3}$$

The residual connection in each block helps capture residual features, effectively learn intricate patterns and allows the smooth flow of gradient information throughout the layers. By applying these blocks, the model can learn more complex spatial features and enhance the representation of the input data. In the architecture, Layer 7 implements the first dilated residual block, enabling the network to capture intricate frame features through residual connections. Following this, Layer 8 applies another max pooling operation. The Layer 9 introduces dropout (rate 0.13) for regularization. The second dilated residual block is applied in Layer 10. Subsequently, the Layer 11 employs max pooling with pool size of (2, 2) to further downsample the feature maps. The initial dilated residual block features a pair of dilated convolution layers (dilation rate 2), with 64 filters each and having 3×3 kernel size. Subsequently, the subsequent block similarly incorporates dilated convolution layers (dilation rate 2) with 128 filters and the same 3×3 kernel size. And the Layer 12 applies dropout (rate 0.13).

The Di-Residual block prove valuable for recognizing actions in videos due to their effectiveness in capturing spatial dependencies and learn informative representations from video sequences. It also help to tackle the issue of vanishing gradients. The inclusion of residual connections facilitates the training of deep networks and enhances the efficient propagation of gradients throughout the network, leading to better convergence and improved performance.

Flattening Layer: After the last Time Distributed layer, a Flatten layer is introduced. This layer seamlessly converts the output of the preceding layer into a 1D tensor by flattening the multi-dimensional feature maps. By reshaping the feature maps into a flat vector representation, this layer facilitates feeding the features into subsequent layers of the network.

Attention Block: The integration of an attention mechanism between the Flatten layer and the GRU layers proves to be a valuable enhancement. The attention mechanism selectively assigns weights to different parts of the sequence based on their relevance. It highlights significant regions and de-emphasizes irrelevant ones, enabling the GRU to focus on the most informative aspects of the workout actions. During the training, the attention mechanism adeptly acquires the ability to allocate different weights to each frame in the video sequence. Frames that are more informative and relevant to recognizing workout actions are assigned higher weights, while less important frames receive lower weights. The attention mechanism is introduced in the Layer 14. This involves generating attention scores using a dense layer with a tanh activation, which are then transformed using a softmax activation to create attention weights. These weights are used to compute an attention-weighted sequence that highlights important frames. The attention mechanism is mathematically expressed using Eqs. (4)–(7). Here, $score_{i,j}$ denotes the attention score between the i^{th} element of the input sequence and the j^{th} element of the residual sequence, the transformation of input sequence elements to align with the attention score's dimensions is facilitated by the weight matrix denoted as W_{att}, $x_{residual,i,j}$ denotes the value of the j^{th} element of the residual sequence for the i^{th} element of the input sequence, $\alpha_{i,j}$ represents the attention weight, which determines how much focus the model should give to the j^{th} element of the residual sequence when processing the i^{th} element of the input sequence, Exp denotes the exponential function, $x_{wt,i}$ is the weighted representation of the i^{th} element of the input sequence and x_{att} is the final output sequence after applying the attention mechanism.

Attention Scores:

$$score_{i,j} = Tanh\left(W_{att} \cdot x_{residual,i,j}\right) \tag{4}$$

Attention (att) Weights:

$$\alpha_{i,j} = \frac{Exp(score_{i,j})}{\sum_{k=1}^{T} Exp(score_{i,j})} \tag{5}$$

Weighted (wt) Sequence:

$$x_{wt,i,j} = \sum_{k=1}^{T} \alpha_{i,j} \cdot x_{residual,i,j} \tag{6}$$

Output Sequence:

$$x_{att} = \left\{x_{att,1}, x_{att,2}, \ldots, x_{att,N}\right\} \tag{7}$$

Then, a time-distributed flatten operation is applied after the attention mechanism, preparing the attention-weighted sequence (AWS) for the subsequent recurrent GRU layers. The GRU layers are then utilized to model the temporal dependencies between frames. The recurrent nature of GRU enables the network to capture the sequential patterns and long-term dependencies in the video sequence. The GRU layers leverage the information encoded in the context vector to make informed predictions based on the temporal patterns learned from the video sequence.

Gated Recurrent Unit (GRU): The GRU is a special variant of recurrent neural network which addresses the issues of gradient disappearance and explosion encountered in general Recurrent Neural Networks (RNNs). In contrast to RNNs, GRU has a similar input and output structure. Unlike LSTM, which requires multiple units, the GRU achieves the functions of forgetting and selecting memory using just one unit. Furthermore, GRU exhibits a smaller parameter count compared to

LSTM. GRU is designed to effectively manage and process sequential data. It does this by controlling the transmission state through gate mechanisms. These gates allow the GRU to selectively store sensitive information for longer periods of time, while discarding less relevant information. This process enhances the representation of features, allowing for a more comprehensive and detailed understanding of the data. A CNN-only model struggles with error handling and data tolerance, causing recognition rates to drop as incorrect data increases. GRU networks improve fault tolerance by analyzing multiple feature maps over time, predicting and eliminating errors even within individual channels. The mathematical operations involved in the processing of the GRU are represented in Eqs. (8) to (11).

$$ug_t = \sigma\left(W_{ug}x_t + U_{ug}h_{t-1} + b_{ug}\right) \tag{8}$$

$$rg_t = \sigma\left(W_{rg}x_t + U_{rg}h_t + b_{rg}\right) \tag{9}$$

$$\hat{h}_t = tanh(W_h x_t + U_h(rg_t \odot h_{t-1} + b_h) \tag{10}$$

$$h_t = ug_t \odot \hat{h}_t + (1 - ug_t) \odot h_{t-1} \tag{11}$$

The GRU effectively captures sequential dependencies in time series data. During every time step (t), the activation, h_t, of the j_{th} GRU unit is determined by blending two elements: The previous time step's hidden state, h_{t-1}, and the current hidden state's activation, $\hat{h}_t$. In this particular scenario, h_t comprises 64 features that are derived from two consecutive video frames at time t. Here, W and U are two parameters matrices and b represent the bias vector. The update gate ug_t determines the extent to which each unit's activation needs to be updated. It acts as a controlling mechanism, influencing the flow of information within the GRU. The variable $\hat{h}_t$ denotes the hidden state at the present time step, which is calculated in a manner similar to that of conventional RNNs. The reset gate (rg_t), holds significant importance in the functioning of the model. When the reset gate value is close to 0, it indicates that the model should disregard or forget the information from the previously computed state. The reset gate helps the model to selectively reset or update the information it carries, allowing it to focus on the most relevant and current information. Here, the hidden layer of a GRU at the current time step receives input from the previous hidden state, enabling it to incorporate temporal dependencies and capture relevant information for further processing.

GRU Block: The GRU block comprising three GRU layers is incorporated after the attention block. GRU captures temporal dependencies and sequence information. The proposed model comprises of three GRU layers of 64 units each. The first and second GRU layers are specifically set to return sequences instead of a single output. This design choice enables the ResDC-GRU Attention model to capture temporal information spanning multiple frames effectively. Afterwards, the outputs from the two GRU units are concatenated and passed into another GRU unit. The third GRU layer generates a singular output and does not return sequences. The GRU layers process the features derived from the convolutional layers over time, capturing the temporal dependencies in the sequence. The outputs of the GRU layers form a sequence of hidden states, representing the learned spatiotemporal features.

In this particular GRU block of the proposed model, a unique approach is utilized, wherein the output from the highest-level GRU layer is combined with the output stemming from the subsequent GRU layer through a process of concatenation. Here, the output of the first GRU layer, represented as O1, is obtained as O1 = GRU1(x), where x represents the flatten output. Then, the output of the next GRU layer, O2, is calculated as O2 = GRU2(O1), where O1 is the GRU1 output.

Concatenate and Additional GRU Layer: The concatenation of the outputs from different GRU layers allows the model to capture complementary information from multiple levels of abstraction. Here, the third GRU layer, O3 with 64 units, is obtained as O3 = GRU3 (O1 + O2), where both O1 and O2 are concatenated and provided as input. This approach allows combination of information from multiple layers and facilitates the learning of complex temporal patterns throughout the stacked GRU network. The additional GRU layer further processes the concatenated features and summarizes them into a final hidden state. To prevent overfitting, another dropout layer (0.17 dropout rate) is applied.

Output Layer: This layer assumes the pivotal role of executing the classification task by leveraging the extracted features. Ultimately, the concluding phase involves the employment of a dense layer embedded with a softmax activation function within the output layer, thereby generating class probabilities for every sequence. The softmax activation function produces probability distributions over the classes, indicating the model's confidence for each class. This function serves a critical role in transforming the neural network's raw outputs into a probability vector, meticulously reflecting the likelihood of each input class, where each element in the vector represents the probability of the corresponding input class ensuring that the probabilities sum up to 1. To determine the Softmax function for a specific workout category, denoted as w_j, one can compute it using Eq. (12). Here, $f\left(w_j\right)$ represents the probability value assigned to the j^{th} class through the softmax computation, and the variable T signifies the resultant number of workout action classes, which is eleven (11) in this case.

$$f\left(w_j\right) = e^{w_j} / \sum_{j}^{T=11} e^{w_j} \tag{12}$$

The feature maps (FMap) showcased in Fig. 3, derived from a frame of the 'GluteBridge' workout video, epitomize the intricacies captured by the proposed model.

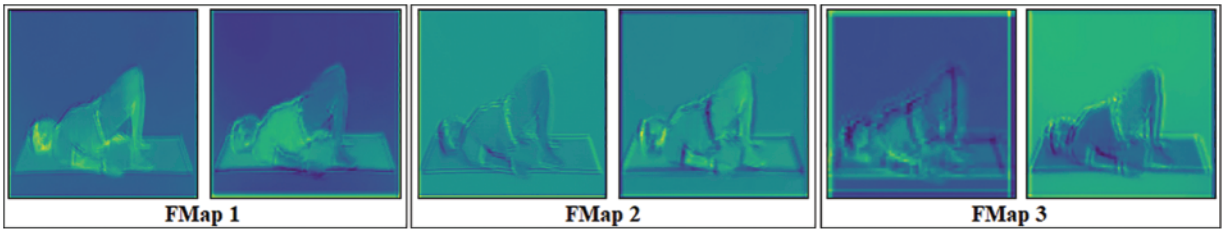


Figure 3: Feature map visualization

Each feature map encapsulates distinctive aspects of the underlying data, offering valuable insights into the convolutional layers' activations. FMap 1 presents an illuminating depiction of the feature map originating from the second activation within the T.D Conv2D layer, with the ELU activation function. In contrast, FMap 2 delineates the feature map arising from the first activation of the Convolution layer, augmented with the ELU activation, within the Residual block. The Residual block produces feature maps rich in spatial and semantic information for the GluteBridge exercise, highlighting hierarchical feature extraction's importance in accurate classification tasks. FMap 3 unveils the activation of the second convolutional layer within the Residual block, emphasizing the model's prowess in capturing hierarchical features across multiple abstraction levels.

3.3 Loss Function

In the context of this study, we address a multi-class challenge by employing a custom loss function derived from weighted categorical cross-entropy (WCCE). This specialized loss function quantifies the discrepancy the model aims to reduce throughout its training process. Through this loss function, the model effectively discerns the allocation of the data samples to their respective categories from the workout categories present in the WAVd dataset. The proposed custom WCCE loss function reduces the disparity between two probability distributions, namely the anticipated distribution and the observed one. The Weighted Categorical Crossentropy with Label Smoothing is particularly useful in scenarios where the training data suffers from class imbalance or when the model tends to overfit on rare classes. It addresses these challenges by assigning different weights to different classes and incorporating label smoothing. In the normal categorical crossentropy loss function, each class is treated equally, which may lead to suboptimal results when certain classes are underrepresented. Weighted Categorical Cross-Entropy addresses this issue by assigning different weights to each class based on their importance. This weight factor empowers the model to prioritize more on learning from the less prevalent classes, thereby improving overall performance. The weighted cross entropy loss is expressed using Eqs. (13)–(15). Here, y_{true} denotes the true target probability distribution for a given example, S_{labels} is the label smoothing parameter. It is a value between 0 and 1 that determines the extent to which the true labels are smoothed, C represents the number of workout categories, N represent the total number of examples in WAVd dataset, $y_{true,i}$ represents the true target probability for the i^{th} example, W_{CE} represents the WCCE loss function, $y_{true,ij}$ denotes the true target probability for the i^{th} example and the j^{th} class, $y_{pred,ij}$ represents the predicted probability for the i^{th} example and the j^{th} class. The weight W_j is calculated based on $y_{true,ij}$, which is used to equalize the impact of each class to the overall loss. The terms S_{labels}, C, N, y_{true}, and y_{pred} are all integral components of this loss function used to guide the training process.

$$y_{true} = (1 - S_{labels}) \cdot y_{true} + \frac{S_{labels}}{C} \tag{13}$$

$$W = \sum_{i=1}^{N} y_{true}, i \tag{14}$$

$$W_{CE} = \frac{1}{N} \sum_{i=1}^{N} \left(\sum_{j=1}^{C} W_j \cdot y_{true,ij} \cdot \log(y_{pred,ij}) \right) \tag{15}$$

Label smoothing is another important aspect of this loss function. In standard crossentropy, the target label for a particular class is represented as a one-hot encoded vector with a value of 1 for true class and 0 for all others. Label smoothing introduces a small amount of uncertainty by replacing the 0 value for the true class with a small positive value (here 0.13). This regularization technique prevents the model from becoming overly confident and encourages it to learn more robust and generalizable representations. The results presented in Table 1 show that when employing the suggested weighted categorical cross-entropy loss function for training the proposed Attention driven ResDC-GRU model on the WAVd dataset, it achieves higher training and validation accuracy and also higher F1-score compared to using the traditional categorical cross-entropy loss function taking Adam optimizer and initial learning rate as 0.0001. Thus, the utilization of the weighted cross-entropy function results in enhanced classification performance.

Table 1: Comparative analysis with application of loss functions used in the proposed model

Loss function	Optimizer	Epochs	Training accuracy	Validation accuracy	F1-score
Standard cross-entropy	Adam	26	99.39%	94.60%	0.95
WCCE	**Adam**	**26**	**99.57%**	**95.81%**	**0.96**

4 Experimental Results

The proposed work is realized using Python programming language in Google collab platform. As there is no standard benchmark dataset for Workout Actions, a new video dataset named Workout Action Video Dataset (WAVd) is created. The proposed ResDC-GRU Attention model's classification performance is evaluated with the train and validation accuracy, and the F1-score. This section provides the details of the available related datasets and developed WAVd dataset, performance assessment, and result analysis.

4.1 Dataset Details

The datasets that are publicly available and are related to this study include KTH Action [32], HMDB51 [33], UCF50 [34], Youtube Actions (YA) [34] and UCF101 [34] data. Furthermore, due to the absence of explicit workout video data, a new dataset named the Workout Action Video dataset (WAVd)[1] is developed by extracting the videos from diverse sources, including TV episodes, YouTube videos and social media sites. This rigorous data collection process was necessary to ensure the research's integrity, given the limited availability of workout action videos. After collecting the data successfully, we manually cut the portion of the video in which the action has been performed and we have named our prepared data as the Workout Action Video dataset (WAVd). Some sample frames of the prepared workout action video dataset are highlighted in Fig. 4. The videos in the dataset were standardized to a resolution of 320 × 240 pixels, captured in RGB color mode, and saved in the AVI video format.

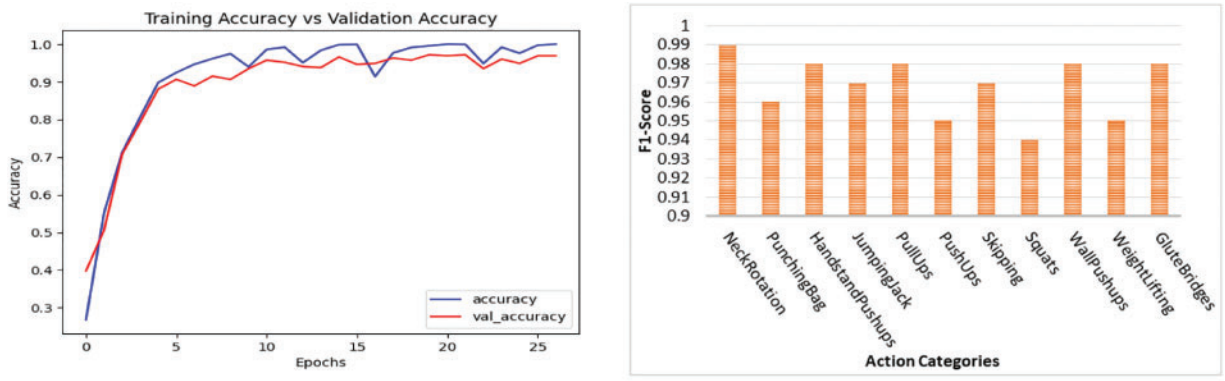
Figure 4: Video frames extracted from WAVd dataset

Diverse workout action classes are labeled by organizing distinct categories of workout videos into separate folders. The workout actions dataset comprises diverse individuals performing workouts in various scenarios, maintaining impartiality. The dataset encompasses a total of 1805 RGB videos, spread across eleven different workout action categories. The considered workout action categories are as follows: *NeckRotation*, *PunchingBag*, *HandstandPushups*, *Pushups*, *WallPushups*, *Pullups*, *JumpingJack*, *Skipping*, *Squats*, *WeightLifting*, and *GluteBridges*.

[1] **WAVd:** https://sites.google.com/view/wavd/home.

4.2 Performance Evaluation

The models were constructed utilizing Python libraries including TensorFlow-Keras, accompanied by specialized modules such as Callbacks and Optimizers within the Google collab platform, without the use of GPU. For analytical tasks, the Sklearn and Matplotlib libraries were employed. The learning curve of the proposed ResDC-GRU Attention model is illustrated in Fig. 5a. The curve reveals that the suggested model exhibits 99.57% training accuracy and 95.81% validation accuracy. As the model's performance showed minimal improvement after 26 epochs, we decided to halt the training process using the callback functions at that stage. The proposed model's effectiveness in categorizing action types is visually represented by the classification scores obtained from the WAVd dataset. These scores are presented in Fig. 5b for the WAVd data, providing a clear illustration of the model's classification performance.



(a) Accuracy Curve (WAVd) (b) Classification Performance (WAVd)

Figure 5: Performance analysis: accuracy curve and classification report of proposed model

The proposed approach has been assessed by comparing it to various state-of-the-art (SOA) deep learning models as depicted in Table 2. The Two-Stream CNN [8], ConvLSTM2D [35], and CNN-LSTM Attention [36] model attained 86.27%, 90.75% and 92.54% accuracy respectively on the WAVd dataset.

Table 2: Performance comparison on different standard models on WAVd dataset

Model	Train accuracy	Validation accuracy	F1-score	TP (in million)
Two stream CNN [8]	89.31%	86.27%	0.86	>97 M
ViT+LSTM [28]	99.84%	95.20%	0.95	>5 M
ConvLSTM2D [35]	96.75%	90.75%	0.91	>1 M
CNN-LSTM Attention [36]	97.82%	92.54%	0.93	>1 M
C3D-BiLSTM [37]	97.20%	93.40%	0.93	>2 M
3DCNN and GRU [38]	98.50%	94.37%	0.94	>1 M
BiHLSTM + Attention [39]	99.46%	94.69%	0.94	>2 M
Proposed model	**99.57%**	**95.81%**	**0.96**	**0.405 M**

Moreover, alongside our proposed model, noteworthy contenders such as the ViT+LSTM [28], C3D-BiLSTM [37], 3DCNN and GRU [38], BiHLSTM + Attention [39], models showcased formidable prowess in classifying workout action videos (WAVd), achieving 95.20%, 93.40%, 94.37% and 94.69% validation accuracies, respectively. Through meticulous experimental analysis, it was discerned that the proposed Attention driven ResDC-GRU model exhibited superior performance compared to alternative approaches with 95.81% accuracy. The ViT+LSTM [28] achieves an F1-score of 0.95, while our proposed model surpasses this performance with an F1-score of 0.96. It is noteworthy that the proposed model stands out with a lightweight architecture, comprising a mere 0.405 million trainable parameters (TP), while all other models boast a considerable number of TP exceeding 1 million.

4.3 Result Analysis

The suggested model is evaluated against several existing video-centric action recognition methods utilizing CNN, LSTM, GRU, and other approaches to assess its effectiveness, as illustrated in Table 3. It illustrates the outcomes of diverse deep learning models using different benchmark datasets, specifically the HMDB51 [33], YA [34], UCF50 [34], and the UCF101 [34] dataset. The proposed model attained an impressive accuracy of 97.2%, followed by the Two-Stream Attention LSTM [13] model and C3D-BiLSTM [37] with 96.9% and 91% accuracy, respectively, on the Youtube actions (YA) dataset. Our methodology demonstrated remarkable classification scores across four prominent action datasets: HMDB51, YA, UCF50, and UCF101, yielding noteworthy accuracies. Among all the models under evaluation, the proposed Attention driven ResDC-GRU model achieved the highest accuracy, reaching 95.6% on the UCF50 benchmark dataset. The UCF50 dataset was evaluated using Improved Dense Trajectories [6], CNN Two Stream Fusion [8], and ARCH [10], achieving accuracies of 92.3%, 86.2%, and 91%, respectively.

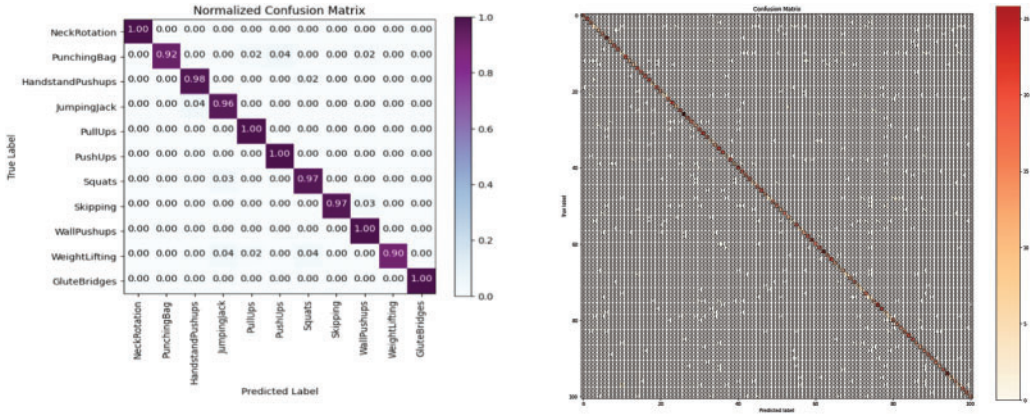
Table 3: Comparative evaluation on standard datasets using various methods

Method(s)	HMDB51	Youtube actions	UCF50	UCF101
Improved dense trajectories [6]	61.1%	–	92.3%	87.9%
Two-stream fusion [8]	76%	88.4%	86.2%	84.6%
ARCH [10]	58.2%	–	91%	85.3%
Two stream attention LSTM [13]	–	96.9%	–	–
3D-FCNN [14]	72.5%	89.1%	90.3%	87.4%
ViT+LSTM [28]	73.7%	–	–	96.1%
C3D-BiLSTM [37]	70.4%	91%	90%	91.2%
BiHLSTM + Attention [39]	71.9%	–	–	94.8%
Video LSTM [40]	56.4%	–	90%	92.2%
Dual 3D-CNN [41]	60.1%	–	89%	87.7%
DS-GRU [42]	72.3%	97.1%	95.2%	95.5%
ST-H ConvLSTM attention (RGB) [43]	52.4%	–	87.8%	85.5%
Proposed model	**81.6%**	**97.2%**	**95.6%**	**93.2%**

The DS-GRU [42] secured the second highest position with 95.2% accuracy on UCF50 dataset. In terms of the HMDB51 data, the ViT+LSTM [28] and C3D-BiLSTM [37] attained an accuracy

of 73.7% and 70.4%, while our proposed technique outperformed all other models with the topmost accuracy of 81.6%. On the same dataset, the CNN Two-Stream Fusion [8], 3D-FCNN [14], Video LSTM [40], and DS-GRU [42], achieved accuracies of 76%, 72.5%, 56.4%, and 72.3%, respectively. However, Improved Dense Trajectories [6] and ARCH [10] achieved 61.1% and 58.2% accuracies, respectively, on the HMDB51 dataset. Lastly, in terms of accuracy on the UCF-101 data, the 3D-FCNN [14] attained an accuracy of 87.4%, while the C3D-BiLSTM [37] and Video LSTM [40] models achieved accuracies of 91.2% and 92.2%, respectively. In contrast, our proposed model achieved a remarkable accuracy of 93.2%, outperforming all others except ViT+LSTM [28], BiHLSTM + Attention [39], and DS-GRU [42], model with accuracy of 96.1%, 94.8%, and 95.5%, respectively. The proposed ResDC-GRU Attention model shows excellent performance on all the benchmark datasets taken for comparison while maintaining a lower computational complexity and having 0.405 million (M) trainable parameters.

Confusion Matrix: The confusion matrix of the proposed Attention driven ResDC-GRU model on the WAVd dataset is portrayed in Figs. 6a while 6b showcases the confusion matrix specifically for the UCF101 dataset, providing a visual representation of the model's performance in classifying actions. We have generated the confusion matrix for these large-scale benchmark datasets by utilizing the proposed model. The confusion matrix provides additional evidence that, barring a small number of misclassified samples, the proposed model adeptly and accurately identifies the majority of samples. The proposed ResDC-GRU Attention model has also demonstrated excellent performance on publicly available benchmark UCF101 action dataset.



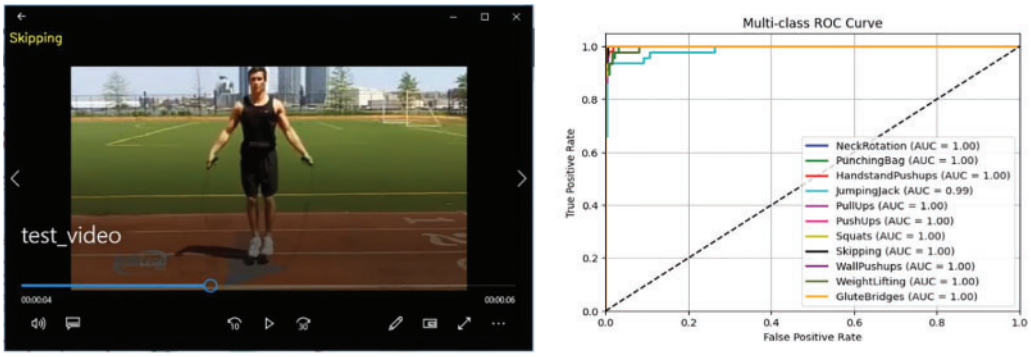
(a) WAVd Dataset (b) UCF101 Dataset

Figure 6: Visualization: confusion matrix of the WAVd and UCF101 datasets

Prediction Result: The proposed model's effectiveness has been assessed using various video samples, and it has successfully classified them accurately. A sample test video labeled as 'Skipping,' but not previously known to the model, is correctly identified by the proposed model, as illustrated in Fig. 7a. Hence, the proposed model stands poised for practical deployment in real-life scenarios, adeptly discerning workout actions from video sequences.

Receiver Operating Characteristic (ROC) Curve: The generated ROC curve of Proposed Attention based ResDC-GRU model on the WAVd dataset is presented in Fig. 7b. The ROC curve offers a visual depiction of the classification model's performance across a range of classification thresholds. In the ROC curve, the horizontal axis denotes the false positive fraction or (1–specificity), while the vertical

axis signifies the true positive fraction or sensitivity. Upon careful examination of the ROC curve, it becomes evident that the majority of the workout classes have achieved an area under curve value of 1, indicating excellent classification performance. Only one class exhibit a slightly lower area under curve value, i.e., 0.99.



(a) Test Result of 'Skipping' video (b) ROC Curve WAVd Test data

Figure 7: Test result and generated ROC curve of the proposed Residual DC-GRU model

5 Conclusion

This research presents a significant advancement in the field of workout action recognition in video streams, offering novel solutions. The introduction of the Workout Action Video dataset (WAVd) fills a crucial gap in existing resources, providing a diverse and meticulously labelled collection of workout action videos. The significant advancements made in deep learning-based techniques have outpaced traditional methods that heavily relied on manually designed features and simple classifiers. The proposed ResDC-GRU Attention model excels in learning spatiotemporal features from video data, leading to more accurate and efficient workout action recognition. The Residual Deep CNN extracts spatial features from video frames, recognizing visual patterns. Subsequently, an attention mechanism refines these spatial features, enabling the model to dynamically allocate varying degrees of importance to specific spatial features across the evolving video sequence, emphasizing the most informative elements. Following the attention-enhanced spatial feature extraction, the GRU efficiently models temporal dependencies, capturing action dynamics across frames. Identifying and categorizing human workout actions in video streams can enhance individuals' workout efficiency by acting as a virtual trainer, offering valuable assistance and guidance during their exercise routines. The proposed Attention-driven ResDC-GRU framework achieves a high accuracy of 95.81% and a Net F1-score of 0.96 in classifying workout actions, which can assist individuals in practising workouts in real life. Furthermore, the utilization of a Custom Weighted Categorical Cross-Entropy (WCCE) loss function enhances the model's learning process, contributing to its efficacy and performance optimization. The efficacy of the proposed model has been comprehensively evaluated across four benchmark action datasets, and it has demonstrated promising results. Future work involves expanding the dataset size and adding diverse workout categories. Refining the ResDC-GRU Attention model will be a priority, as well as optimizing its architecture and network combinations for improved accuracy. The curated WAVd dataset serves as a vital resource for training and evaluating workout action recognition methods, providing researchers with a rich and comprehensive dataset to develop more robust and effective fitness tracking systems.

Acknowledgement: The authors gratefully acknowledge the support received from the Computer Science and Technology Department of IIEST, Shibpur for carrying out the proposed research.

Funding Statement: The authors received no specific funding for this study.

Author Contributions: A.D.: Data curation, Methodology, Analysis and interpretation of results, Writing—original draft; S.B.: Formal analysis and Supervision, Writing—review and editing, Validation; D.N.L.: Writing—review & editing, Visualization, Validation. All authors have read and approved the final version of the manuscript.

Availability of Data and Materials: The data introduced in this research will be made available on reasonable request to: https://sites.google.com/view/wavd/home.

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

References

[1] K. Pinckard, K. K. Baskin, and K. I. Stanford, "Effects of exercise to improve cardiovascular health," *Front. Cardiovasc. Med.*, vol. 6, pp. 69, Jun. 2019. doi: 10.3389/fcvm.2019.00069.

[2] J. P. Thyfault and A. Bergouignan, "Exercise and metabolic health: Beyond skeletal muscle," *Diabetologia*, vol. 63, no. 8, pp. 1464–1474, Jun. 2020. doi: 10.1007/s00125-020-05177-6.

[3] D. Deotale, M. Verma, P. Suresh, and N. Kumar, "Physiotherapy-based human activity recognition using deep learning," *Neural Comput. Appl.*, vol. 35, pp. 11431–11444, 2023. doi: 10.1007/s00521-023-08307-4.

[4] G. Bhola and D. K. Vishwakarma, "A review of vision-based indoor har: State-of-the-art, challenges, and future prospects," *Multimed. Tools Appl.*, vol. 83, no. 1, pp. 1965–2005, May 2023. doi: 10.1007/s11042-023-15443-5.

[5] L. Minh Dang, K. Min, H. Wang, M. Jalil Piran, C. Hee Lee and H. Moon, "Sensor-based and vision-based human activity recognition: A comprehensive survey," *Pattern Recognit.*, vol. 108, pp. 107561, Dec. 2020. doi: 10.1016/j.patcog.2020.107561.

[6] X. Peng, L. Wang, X. Wang, and Y. Qiao, "Bag of visual words and fusion methods for action recognition: Comprehensive study and good practice," *Comput. Vis. Image Underst.*, vol. 150, pp. 109–125, Sep. 2016. doi: 10.1016/j.cviu.2016.03.013.

[7] H. Yasin, M. Hussain, and A. Weber, "Keys for action: An efficient keyframe-based approach for 3D action recognition using a deep neural network," *Sensors*, vol. 20, no. 8, pp. 2226, 2020. doi: 10.3390/s20082226.

[8] C. Feichtenhofer, A. Pinz, and A. Zisserman, "Convolutional two-stream network fusion for video action recognition," in *2016 IEEE Conf. on Comput. Vis. Pattern Recognit. (CVPR)*, Las Vegas, NV, USA, 2016, pp. 1933–1941. doi: 10.1109/CVPR.2016.213.

[9] A. Ullah, J. Ahmad, K. Muhammad, M. Sajjad, and S. W. Baik, "Action recognition in video sequences using deep bi-directional LSTM With CNN features," *IEEE Access*, vol. 6, pp. 1155–1166, 2018. doi: 10.1109/ACCESS.2017.2778011.

[10] M. Xin, H. Zhang, H. Wang, M. Sun, and D. Yuan, "Arch: Adaptive recurrent-convolutional hybrid networks for long-term action recognition," *Neurocomputing*, vol. 178, pp. 87–102, Feb. 2016. doi: 10.1016/j.neucom.2015.09.112.

[11] T. Rangari, S. Kumar, P. P. Roy, D. P. Dogra, and B. G. Kim, "Video based exercise recognition and correct pose detection," *Multimed. Tools Appl.*, vol. 81, no. 21, pp. 30267–30282, 2022. doi: 10.1007/s11042-022-12299-z.

[12] M. Qasim Gandapur and E. Verdú, "CONVGRU-CNN: Spatiotemporal deep learning for real-world anomaly detection in video surveillance system," *Int. J. Interact. Multimed. Artif. Intell.*, vol. 8, no. 4, 2023. doi: 10.9781/ijimai.2023.05.006.

[13] C. Dai, X. Liu, and J. Lai, "Human action recognition using two-stream attention based LSTM networks," *Appl. Soft Comput.*, vol. 86, pp. 105820, Jan. 2020. doi: 10.1016/j.asoc.2019.105820.
[14] A. Sánchez-Caballero *et al.*, "3DFCNN: Real-time action recognition using 3D deep neural networks with raw depth information," *Multimed. Tools Appl.*, vol. 81, no. 17, pp. 24119–24143, Mar. 2022. doi: 10.1007/s11042-022-12091-z.
[15] Y. D. Zheng, Z. Liu, T. Lu, and L. Wang, "Dynamic sampling networks for efficient action recognition in videos," *IEEE Trans. Image Process.*, vol. 29, pp. 7970–7983, 2020. doi: 10.1109/TIP.2020.3007826.
[16] G. Varol, I. Laptev, and C. Schmid, "Long-term temporal convolutions for action recognition," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 40, no. 6, pp. 1510–1517, Jun. 2018. doi: 10.1109/TPAMI.2017.2712608.
[17] P. Verma, A. Sah, and R. Srivastava, "Deep learning-based multi-modal approach using RGB and skeleton sequences for human activity recognition," *Multimed. Syst.*, vol. 26, pp. 671–685, 2020. doi: 10.1007/s00530-020-00677-2.
[18] A. Zhou *et al.*, "Multi-head attention-based two-stream EfficientNet for action recognition," *Multimed. Syst.*, vol. 29, pp. 487–498, 2023. doi: 10.1007/s00530-022-00961-3.
[19] L. Wang *et al.*, "Temporal segment networks for action recognition in videos," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 41, no. 11, pp. 2740–2755, Nov. 2019. doi: 10.1109/TPAMI.2018.2868668.
[20] H. Luo, G. Lin, Y. Yao, Z. Tang, Q. Wu and X. Hua, "Dense semantics-assisted networks for video action recognition," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 32, no. 5, pp. 3073–3084, May 2022. doi: 10.1109/TCSVT.2021.3100842.
[21] G. Garzón and F. Martínez, "A fast action recognition strategy based on motion trajectory occurrences," *Pattern Recognit. Image Anal.*, vol. 29, no. 3, pp. 447–456, Jul. 2019. doi: 10.1134/S1054661819030039.
[22] A. Dey, A. Dutta, and S. Biswas, "WorkoutNet: A deep learning model for the recognition of workout actions from still images," in *2023 3rd Int. Conf. on Intell. Technol. (CONIT)*, Hubli, India, 2023, pp. 1–8.
[23] A. Dey, S. Biswas, and L. Abualigah, "Umpire's signal recognition in cricket using an attention based DC-GRU network," *Int. J. Eng.*, vol. 37, no. 4, pp. 662–674, 2024. doi: 10.5829/IJE.2024.37.04A.08.
[24] M. Naeem Akbar, S. Khan, M. Umar Farooq, M. Alhaisoni, U. Tariq and M. Usman Akram, "HybridHR-Net: Action recognition in video sequences using optimal deep learning fusion assisted framework," *Comput. Mater. Contin.*, vol. 76, no. 3, pp. 3275–3295, 2023. doi: 10.32604/cmc.2023.039289.
[25] A. Dey, S. Biswas, and D. N. Le, "Recognition of human interactions in still images using AdaptiveDRNet with multi-level attention," *Int. J. Adv. Comput. Sci. Appl.*, vol. 14, no. 10, 2023. doi: 10.14569/IJACSA.2023.01410103.
[26] G. Wei, H. Zhou, L. Zhang, and J. Wang, "Spatial-temporal self-attention enhanced graph convolutional networks for fitness yoga action recognition," *Sensors*, vol. 23, no. 10, pp. 4741, Jan. 2023. doi: 10.3390/s23104741.
[27] A. Dey *et al.*, "Attention-based AdaptSepCX network for effective student action recognition in online learning," *Proc. Comput. Sci.*, vol. 233, pp. 164–174, 2024. doi: 10.1016/j.procs.2024.03.206.
[28] A. Hussain, T. Hussain, W. Ullah, and S. W. Baik, "Vision transformer and deep sequence learning for human activity recognition in surveillance videos," *Comput. Intell. Neurosci.*, vol. 2022, pp. 1–10, Apr. 2022. doi: 10.1155/2022/3454167.
[29] C. Zhang, L. Liu, M. Yao, W. Chen, D. Chen and Y. Wu, "HSiPu2—A new human physical fitness action dataset for recognition and 3D reconstruction evaluation," in *2021 IEEE/CVF Conf. on Comput. Vis. and Pattern Recognit. Workshops (CVPRW)*, Nashville, TN, USA, Jun. 2021.
[30] F. F. Youssef, V. Parque, and W. Gomaa, "VCOACH: A virtual coaching system based on visual streaming," *Proc. Comput. Sci.*, vol. 222, pp. 207–216, 2023. doi: 10.1016/j.procs.2023.08.158.
[31] J. Chen, J. Wang, Q. Yuan, and Z. Yang, "CNN-LSTM model for recognizing video-recorded actions performed in a traditional Chinese exercise," *IEEE J. Transl. Eng. Health Med.*, vol. 11, pp. 351–359, 2023. doi: 10.1109/JTEHM.2023.3282245.
[32] KTH Dataset. 2004. Accessed: May 15, 2023. [Online]. Available: https://www.csc.kth.se/cvap/actions/.

[33] H. Kuehne, H. Jhuang, E. Garrote, T. Poggio, and T. Serre, "HMDB: A large video database for human motion recognition," in *2011 Int. Conf. on Comput. Vis.*, Barcelona, Spain, Nov. 2011.
[34] UCF and Youtube Actions Data. 2012. Accessed: Jun. 25, 2023. [Online]. Available: https://www.crcv.ucf.edu/data/
[35] P. Dasari, L. Zhang, Y. Yu, H. Huang, and R. Gao, "Human action recognition using hybrid deep evolving neural networks," in *2022 Int. Joint Conf. on Neural Netw. (IJCNN)*, Padua, Italy, Jul. 2022.
[36] L. Wang, Y. Xu, J. Cheng, H. Xia, J. Yin and J. Wu, "Human action recognition by learning spatio-temporal features with deep neural networks," *IEEE Access*, vol. 6, pp. 17913–17922, 2018. doi: 10.1109/ACCESS.2018.2817253.
[37] W. Li, W. Nie, and Y. Su, "Human action recognition based on selected spatio-temporal features via bidirectional LSTM," *IEEE Access*, vol. 6, pp. 44211–44220, 2018. doi: 10.1109/ACCESS.2018.2863943.
[38] M. Savadi Hosseini and F. Ghaderi, "A hybrid deep learning architecture using 3D CNNs and GRUs for human action recognition," *Int. J. Eng.*, vol. 33, no. 5, pp. 959–965, 2020. doi: 10.5829/IJE.2020.33.05B.29.
[39] H. Yang, J. Zhang, S. Li, and T. Luo, "Bi-direction hierarchical LSTM with spatial-temporal attention for action recognition," *J. Intell. Fuzzy Syst.*, vol. 36, no. 1, pp. 775–786, 2019. doi: 10.3233/JIFS-18209.
[40] Z. Li, K. Gavrilyuk, E. Gavves, M. Jain, and C. G. M. Snoek, "VideoLSTM convolves, attends and flows for action recognition," *Comput. Vis. Image Underst.*, vol. 166, pp. 41–50, Jan. 2018. doi: 10.1016/j.cviu.2017.10.011.
[41] S. Jiang, Y. Qi, H. Zhang, Z. Bai, X. Lu and P. Wang, "D3D: Dual 3-D convolutional network for real-time action recognition," *IEEE Trans. Ind. Inform.*, vol. 17, no. 7, pp. 4584–4593, Jul. 2021. doi: 10.1109/TII.2020.3018487.
[42] A. Ullah, K. Muhammad, W. Ding, V. Palade, I. U. Haq and S. W. Baik, "Efficient activity recognition using lightweight CNN and DS-GRU network for surveillance applications," *Appl. Soft Comput.*, vol. 103, pp. 107102, 2021. doi: 10.1016/j.asoc.2021.107102.
[43] H. Ji, F. Xue, W. Zhang, and Y. Cao, "An attention-based spatial-temporal hierarchical ConvLSTM network for action recognition in videos," *IET Comput. Vis.*, vol. 13, no. 8, pp. 708–718, 2019. doi: 10.1049/iet-cvi.2018.5830.