



Article

Human Action Representation Learning Using an Attention-Driven Residual 3DCNN Network

Hayat Ullah  and Arslan Munir * 

Department of Computer Science, Kansas State University, Manhattan, KS 66506, USA; hayatu@ksu.edu

* Correspondence: amunir@ksu.edu

Abstract: The recognition of human activities using vision-based techniques has become a crucial research field in video analytics. Over the last decade, there have been numerous advancements in deep learning algorithms aimed at accurately detecting complex human actions in video streams. While these algorithms have demonstrated impressive performance in activity recognition, they often exhibit a bias towards either model performance or computational efficiency. This biased trade-off between robustness and efficiency poses challenges when addressing complex human activity recognition problems. To address this issue, this paper presents a computationally efficient yet robust approach, exploiting saliency-aware spatial and temporal features for human action recognition in videos. To achieve effective representation of human actions, we propose an efficient approach called the dual-attentional Residual 3D Convolutional Neural Network (DA-R3DCNN). Our proposed method utilizes a unified channel-spatial attention mechanism, allowing it to efficiently extract significant human-centric features from video frames. By combining dual channel-spatial attention layers with residual 3D convolution layers, the network becomes more discerning in capturing spatial receptive fields containing objects within the feature maps. To assess the effectiveness and robustness of our proposed method, we have conducted extensive experiments on four well-established benchmark datasets for human action recognition. The quantitative results obtained validate the efficiency of our method, showcasing significant improvements in accuracy of up to 11% as compared to state-of-the-art human action recognition methods. Additionally, our evaluation of inference time reveals that the proposed method achieves up to a 74× improvement in frames per second (FPS) compared to existing approaches, thus showing the suitability and effectiveness of the proposed DA-R3DCNN for real-time human activity recognition.

**Citation:** Ullah, H.; Munir, A.

Human Action Representation Learning Using an Attention-Driven Residual 3DCNN Network.

Algorithms **2023**, *16*, 369. <https://doi.org/10.3390/a16080369>

Academic Editor: Frank Werner

Received: 16 June 2023

Revised: 15 July 2023

Accepted: 28 July 2023

Published: 31 July 2023



Copyright: © 2023 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

Keywords: human activity recognition; 3DCNN; 3D spatial attention; 3D channel attention; residual convolutional neural network; pattern recognition

1. Introduction

Convolutional neural networks (CNNs) have become one of the most widely used deep learning architectures in computer vision due to their ability to effectively capture the spatial features of image and video data. In recent years, CNNs have shown remarkable success in a variety of applications, including object detection and recognition [1], image segmentation [2], and scene understanding [3]. The design of deep neural networks is crucial for their efficiency, including the depth and structure of the network layers, depending on the task. In some cases, such as object recognition and video analytics, over-parameterization is necessary to ensure the model captures complex hidden patterns and generalizes well. However, this comes at the cost of increased computational complexity, making them unsuitable for real-time environments and resource-limited devices, and requiring high-end GPUs for training [4,5]. The design of network architecture is task-specific which varies from problem-to-problem. For instance, recognizing objects in still images demands a plain 2DCNN network composed of convolutional layers for spatial feature extraction and classification layer for classification task. Recognizing human actions

in video stream cannot be handled with single 2DCNN networks, in view of the fact that videos are composed of large sequence of frames presenting the temporal flow of video across the frames in the temporal dimension (t).

To cope with the challenge of human action recognition task in video streams, researchers have introduced different solutions that include two-stream 2D convolutional neural network (CNN) architectures [6], 3D CNN (2D + 1D) architectures [7], and CNN with a recurrent neural network (RNN) [8]. Typically, two-stream 2DCNNs [6] use two different CNN architectures to extract two different kinds of features from the input video. The first CNN model extracts the spatial features from the input video frames, whereas the second network extracts the temporal optical flow features with respect to their corresponding spatial features. The extracted features from both the models can then be combined as a single latent representation vector for the activity recognition task. On the other hand, 3DCNNs [7] use single CNN architecture having 3D convolution kernels, where the first two dimensions capture the spatial features and the last dimension of the 3D kernel captures temporal flow of the spatial features across the frames. The CNN with RNN architecture frameworks include two different type of models (i.e., CNN followed by RNN). The CNN model in CNN + RNN architectures extracts the spatial features from the video frames and converts it to a one-dimensional latent representation such as feature vectors. The extracted latent representation from the CNN models can then be fed to the RNN model for activity classification task using sequential pattern learning. Typically, an activity recognition framework with two different network architectures increases the parameters space (computational complexity) of the entire framework as well as the time complexity of the model for the task under the consideration. Considering the parameter space and time complexity of two-stream architectures, 3DCNNs are considered a suitable candidate for human activity recognition task.

Therefore, in this paper, we propose a computationally efficient residual 3DCNN architecture called dual-attentional residual 3D convolutional neural network (DA-R3DCNN) with channel and spatial attention for human activity recognition task. The proposed DA-R3DCNN has channel and spatial attention layers after each residual block which helps our model to propagate salient features from the early layers to later layers. This propagation of salient information significantly improves the performance of our model for human activity recognition task. More precisely, the major contributions of this paper are as follows:

1. To overcome the issue of over-parameterization, we present a computationally efficient yet robust end-to-end residual 3DCNN model coupled with dual 3D attention and residual 3D convolution mechanism, learning object and motion-centric spatio-temporal representations of human actions in video sequence;
2. To prevent gradient vanishing, this work proposes a 3D residual convolution mechanism that allows the flow of learned representations from the early layers to the later layers of the network. Moreover, instead of using plain shortcut path, we use convoluted shortcut path having a 3D convolution layer of kernel size $1 \times 1 \times 1$;
3. To efficiently extract spatial saliency from video frames, we utilize a dual 3D channel-spatial attention mechanism along with residual 3D skip connections. Our approach integrates the dual-attentional module after every two consecutive 3D convolutional layers within the 3DCNN model. This enables the extraction of discriminative features that are sensitive to object saliency, allowing for precise localization of action-specific regions in the video frames.

The remaining sections of this paper are organized as follows. Section 2 presents a concise overview of related works in the field of human activity recognition. Section 3 delves into a comprehensive discussion of the proposed DA-R3DCNN framework and its key components. The detailed experimental evaluation of the DA-R3DCNN framework, along with comparisons to the state-of-the-art human action recognition methods, is presented in Section 4. Finally, Section 5 concludes this paper, and also highlights potential future research directions in this domain.

2. Related Works

Over the past decade, there has been significant research on human activity recognition, with several advanced methods proposed to effectively tackle the problem of recognizing human actions. These approaches include two-stream 2DCNNs [9–13], CNN + LSTM [14–18], and 3DCNN-based methods [7,19–23]. Typically, the two-stream 2DCNN architecture paradigm uses two different CNN architectures for modeling human actions in video data. Both CNN architectures operate on the same input data, however, they extract different representations from the input video. One model extract the discriminative spatial features (i.e., encoded visual representations), while the other network extract temporal features (i.e., temporal flow of spatial features) from the input video. For instance, Wang et al. [11] have proposed a two-stream CNN architecture approach for human activity recognition. Their approach utilizes two separate CNNs to extract spatial and temporal features from the input video frames. They have also introduced a video frames segmentation strategy, which involves segmenting the input video into three segments. The two-stream CNN architecture is then applied to these extracted segments to perform segment classification. The segment classification scores are then combined using average pooling to perform video-level classification. Karpathy et al. [9] have proposed a dual-stream 2DCNN framework, to model both spatial and temporal features from the given video frames. To expedite the computation process, they operate their dual-stream CNN model on two different resolution of video frames. The extracted features from both the models are then fused together to obtain spatial-temporal representation of human actions in video. In another work, Zhang et al. [13] have introduced a multi-task learning approach for human activity recognition in low-resolution videos. To improve the resolution of the input video, they have proposed two super-resolution techniques that transformed the low-resolution input video into a high-resolution video. The transformed high-resolution video frames are then fed to their dual-stream classification network for human activity recognition task.

Unlike the two-stream 2DCNN approaches, the CNN + LSTM paradigm uses two different types of networks for spatial and temporal features representation learning. The first part of this paradigm uses 2DCNN architecture to extract discriminative spatial features and convert it to latent representation, where the later part operates the RNN model on the extracted latent representation and learns the temporal hidden patterns in the spatial features. For instance, Srivastava et al. [15] have proposed unsupervised encoder and decoder long short-term memory (LSTM) networks for learning temporal modeling of human actions. The authors initially transformed the input video into a fixed-length representation of temporal features using an encoder LSTM network. Subsequently, they employed a decoder network to reconstruct the video from the latent representation, which facilitated human action predictions. Donahue et al. [14] have presented a recurrent convolution driven approach called long term recurrent convolutional network (LRCN) for recognizing human actions in videos. They have used a 2DCNN architecture to transform the input video frames to 1D latent representation of spatial features. The extracted latent representations are then fed to an LSTM network to capture the temporal changes in the extracted spatial features across array of frames. In the work presented in [18], Sudhakaran et al. have utilized a task-specific recurrent unit that incorporates a spatial attention mechanism. This mechanism enables the capture of salient features across sequences of video frames. The extracted salient features are then processed by an LSTM network to learn the temporal relations of the salient information, facilitating video-level activity recognition. Sharma et al. [16] have proposed the utilization of a deep multi-layer LSTM for the recursive estimation of visual attention maps. Their approach involves applying the multi-layer LSTM to RGB video frames, allowing for the computation of weighted attention maps through recursive operations. They have claimed that their proposed weighted attention maps mechanism greatly helps the model in enhancing feature representation, which turns in better performance of the model for the activity recognition task.

Both the two-stream 2DCNN and CNN + LSTM based methods use two distinct architectures for capturing spatiotemporal features in video frames. The utilization of two different networks makes these approaches computationally inefficient, thereby, increasing the overall computational complexity of the model for the activity recognition task. To alleviate the computational burden of the model, numerous studies have proposed unified end-to-end 3DCNN approaches that encapsulate the learning objective of both spatial and temporal features through a single model. Typically, 3DCNNs utilized the first two channel of the learnable kernels for capturing spatial features, where the last channel captures the temporal flow of the spatial features across the sequence of input data samples (i.e., video frames in case of human activity recognition). For example, Diba et al. [19] have introduced a modified version of the DenseNet [24] architecture called DenseNet-3D or temporal 3DCNN (T3D). They have achieved this by replacing 2D convolution and pooling kernels with 3D convolution and pooling kernels. Through their experiments, they have claimed that their T3D model demonstrates the potential to capture both short and long-term spatiotemporal features within video sequences. In another study, Varol et al. [20] have presented a specialized variant of CNN known as long-term temporal convolution (LTC). They have extended the temporal depth of their 3DCNN convolutional layers and reduced the receptive field of feature maps. These modifications have allowed the model to effectively learn long-term spatiotemporal patterns in video streams. To capture temporal-specific features, Hussain et al. [22] have proposed a multi-scale 3DCNN called Timeception that is designed to handle significant fluctuations in the temporal dimension by accommodating different temporal extents, which helps to effectively recognize long and intricate actions. The advent of 3DCNN concept has allowed researchers to solve the sequential learning task using unified approach instead of using two different architectures. Although, the reported 3DCNN-based approaches have shown noticeable improvement over two-stream 2DCNN and RNN-based approaches, these models are usually over-parameterized and can be optimized in terms of task-specific parameters reduction.

To address the limitations of existing 3DCNNs for the human activity recognition task, this paper proposes a residual 3DCNN architecture with encapsulated 3D channel and spatial attention mechanisms. The proposed DA-R3DCNN framework uses a residual 3DCNN architecture, where each convolution layers is stacked with a channel and spatial attention module that helps our backbone model to progressively learn salient features during training. This way, the 3D channel and spatial attention module encourages the backbone residual 3DCNN to enhance the representation of salient information across the multiple 3D convolution layers and eliminates the contribution of sparse parameters in the learning process. By eliminating the sparsity of parameter space, the proposed framework learns robust features while having a small parameters space.

3. Proposed DA-R3DCNN Human Activity Recognition Framework

This section presents the detailed overview of the proposed DA-R3DCNN architecture and its sub-components. The proposed framework incorporates three essential components: a 3DCNN architecture for learning spatiotemporal representations, a 3D convolution residual block, and a dual-attention module (comprising channel and spatial attention). These components work together to enable the residual 3DCNN to effectively capture salient features within video frames. For better understanding, we divided the discussion on these components in separate sections. First, we provide insights of the proposed residual 3DCNN, focusing on architecture details, and then present the technical details of the 3D convolution residual block. Finally, we present the detailed technical aspects of channel and spatial attention. The visual overview of our proposed DA-R3DCNN framework and its workflow is depicted in Figure 1.

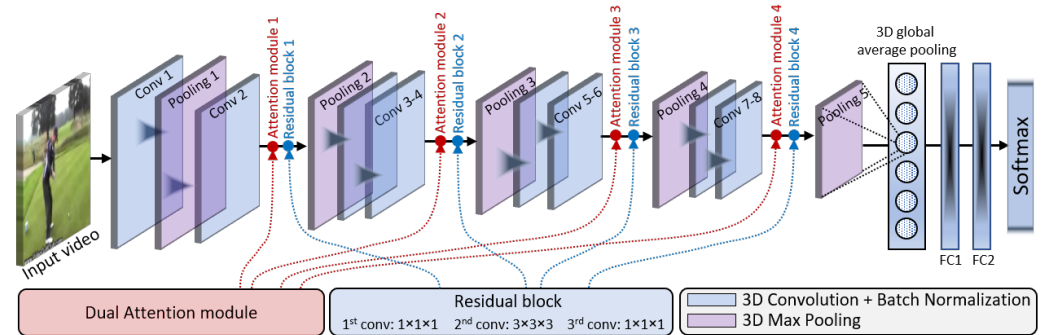


Figure 1. The graphical abstract of our proposed DA-R3DCNN network architecture.

3.1. DA-R3DCNN Architecture

In this work, we propose a residual 3DCNN model coupled with dual-channel spatial attention mechanism. The proposed DA-R3DCNN network consists of eight 3D convolutional + batch normalization layers, four 3D max pooling layers, three residual blocks, and four dual-attention modules. The formation of convolution layers in our proposed DA-R3DCNN model is determined through empirical assessments. It is important to note that we maintained a consistent number of filters across all standard 3D convolution and residual 3D convolution layers, with a fixed value of 128 kernels per layer. Furthermore, all 3D convolution layers, including residual 3D convolution layers, are coupled with batch normalization layers. The architectural details of the proposed DA-R3DCNN model are given in Table 1. As given in Table 1, the first 3D convolutional layer operates 128 kernels of size $3 \times 3 \times 3$ on input frames, which are then down-sampled by the first 3D max pooling layer having a kernel size of $3 \times 3 \times 3$. The second 3D convolution layer operates 128 kernels of size $3 \times 3 \times 3$ on the output feature maps from the first 3D max pooling layer. The convoluted feature maps are then operated by the first attention module that computes channel and spatial attention in input feature maps from the second 3D convolution layer, followed by the first residual block enhancing feature representations using residual convolution connection. The output feature maps from the first residual block are then down-sampled by the second 3D max pooling layer, followed by two consecutive 3D convolution layers (i.e., third and fourth 3D convolution layers) which operate 128 kernels of size $3 \times 3 \times 3$ on the output feature maps from the second 3D max pooling layer. The convoluted feature maps are then operated by the second attention module, followed by the second residual block. The output feature maps from the second residual block are then down-sampled by the third 3D max pooling layer having kernel size of $3 \times 3 \times 3$.

The intermediate pooled feature maps from the third 3D max pooling layer are then operated by two consecutive 3D convolution layers (i.e., fifth and sixth 3D convolution layers) using 128 kernels of size $3 \times 3 \times 3$. The convoluted feature maps from the fifth and sixth 3D convolution layers are then operated by the third attention module, followed by the third residual block. The resultant feature maps from the third residual block are further down-sampled by the fourth 3D max pooling layer having kernel size of $3 \times 3 \times 3$. The down-sampled feature maps from fourth 3D max pooling layer are further convoluted by the seventh and eighth 3D convolution layers, having 128 kernels of size $3 \times 3 \times 3$. The output convoluted feature maps from the seventh and eighth 3D convolution layers are then operated by the fourth attention module. The feature maps generated by the fourth attention module are further improved by passing them through the fourth residual block. Subsequently, these feature maps are down-sampled using the fifth 3D max pooling layer. The pooled feature maps are then converted to 1D (i.e., $1 \times n$ size, where n represents the number of feature values) latent representation by 3D global average pooling layer. The resultant 1D feature values are then operated by 2 consecutive fully connected layers (i.e., FC1 and FC2 layers) having dimensions of 1×512 . Finally, the output (i.e., logits having negative and positive values) of FC1 and FC2 layers are passed to softmax layer which converts it to final probabilities (values between 0 and 1).

Table 1. Architectural overview of our proposed DA-R3DCNN Framework.

Layer	Input Channels	Number of Kernels	Kernel Size	Activation	Padding	Output Channels
Conv 1 + BN	3	128	$3 \times 3 \times 3$	ReLU	1	128
3D Max pooling Layer 1						
Conv 2 + BN	128	128	$3 \times 3 \times 3$	ReLU	1	128
Channel + Spatial Attention Module 1						
3D Residual Block 1						
3D Max pooling Layer 2						
Conv 3 + BN	128	128	$3 \times 3 \times 3$	ReLU	1	128
Conv 4 + BN	128	128	$3 \times 3 \times 3$	ReLU	1	128
Channel + Spatial Attention Module 2						
3D Residual Block 2						
3D Max pooling Layer 3						
Conv 5 + BN	128	128	$3 \times 3 \times 3$	ReLU	1	128
Conv 6 + BN	128	128	$3 \times 3 \times 3$	ReLU	1	128
Channel + Spatial Attention Module 3						
3D Residual Block 3						
3D Max pooling Layer 4						
Conv 7 + BN	128	128	$3 \times 3 \times 3$	ReLU	1	128
Conv 8 + BN	128	128	$3 \times 3 \times 3$	ReLU	1	128
Channel + Spatial Attention Module 4						
3D Residual Block 4						
3D Max pooling Layer 5						
3D Average Pooling Layer						
FC1-(512)						
FC2-(512)						
Softmax (Number of classes)						

3.2. 3D Residual Convolution Block

To limit the propagation of vanishing gradients across the network layers, we used a 3D residual convolution mechanism inspired by the 2D residual convolution in [25], with convoluted shortcut path. As shown in Figure 2, the utilized 3D residual convolution block consists of three convolution layers, with one additional convolution layer over the shortcut path, where each convolution layer is binned with a batch normalization layer. The second convolution layer in the residual block operates a 3×3 size kernel, where the first, third, and the shortcut path convolution layers operate 1×1 size kernels. Unlike, the original residual block presented in [25], in this paper, we used 3D convolution block containing 3D convolution layers instead of 2D convolution layers. Further, instead of using a plain shortcut path as used in [25], in this paper, we used a 3D convoluted shortcut path to ensure the compatibility of input and output dimensions. The 3D residual convolution block used in this paper consists of two key components, the residual mapping and the shortcut path (skip connection). Mathematically, the utilized 3D residual convolution block can be expressed as follows:

$$y = g(x, w) + x', \quad (1)$$

where x is the input of the residual block and g represents the mapping function (convolution layers of the residual block), which learns the mapping (transforming input to output) between input and output using a set of weights represented by w . The variable x' represents the convoluted shortcut path having $1 \times 1 \times 1$ convolution, which enables the gradients to flow more easily through the network layers resulting in better performance. Finally, variable y denotes the weighted mapping of input to output of the residual block.

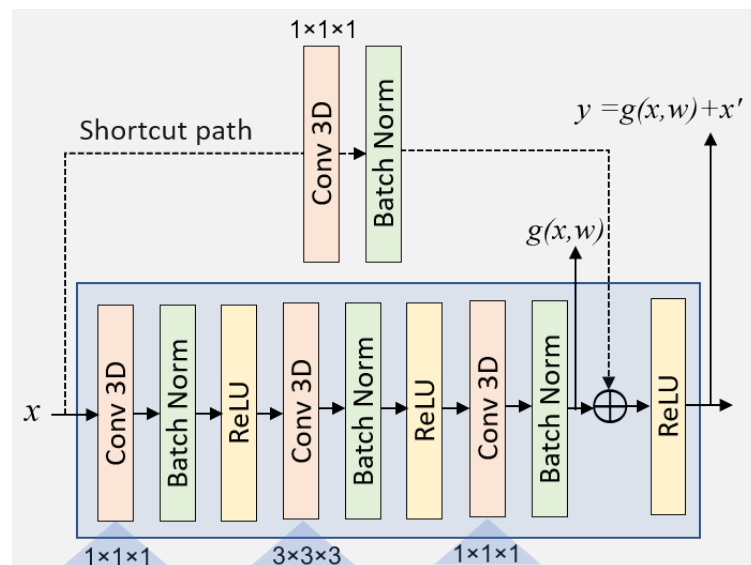


Figure 2. The visual overview of the utilized 3D residual convolutional block used in this study, with convoluted shortcut path.

3.3. Dual Channel-Spatial Attention Module

Our proposed framework utilizes an attention-driven CNN architecture to selectively concentrate on the most significant regions within video frames. This method enables efficient and accurate localization of the salient regions while also enhancing the quality of the feature representation. The proposed attention mechanism is the modification of the convolutional block attention module (CBAM) [26]. To achieve this modification, the 7×7 convolution layer in CBAM is replaced with a 3D convolution layer having a kernel size of $3 \times 3 \times 3$. Additionally, the spatial attention module is fused with the intermediate output of the channel attention module using an element-wise product operation. The resulting dual-attention block is visually represented in Figure 3. Our proposed approach employs a fusion of channel and spatial attentions to efficiently extract important features from video frames while minimizing the number of parameters required. This design not only improves the representation of features but also reduces overhead. To implement this approach, we incorporated a stacked dual-attention module after every two consecutive convolutional layers in our network. This construction strategy optimizes the extraction of salient features, resulting in a highly efficient and accurate model. The channel attention module in our proposed architecture calculates the weighted contribution of RGB channels by applying intermediate channel attention A_C to the output feature maps F_M from the previous convolutional layer. This process results in the channel attention Att_C , which is used to enhance the overall feature representation. Once the channel attention module computes the channel attention feature maps Att_C , they are then passed into the spatial attention module for further processing. The spatial attention module uses the channel attention maps to identify relevant object-specific regions within the video frames. To generate the refined feature maps $F_{M'}$, we fused the spatial attention feature maps Att_S with the input feature maps F_M using a residual skip connection by employing an element-wise addition operation. This approach significantly enhances the quality of the feature representation, enabling more precise localization of salient regions. Mathematically, channel attention, spatial attention, and refined attention feature maps can be expressed as follows:

$$Att_C^{H \times W \times C} = \mathcal{A}_C(F_M^{H \times W \times C}) \otimes F_M^{H \times W \times C}, \quad (2)$$

$$Att_S^{H \times W \times C} = \mathcal{A}_S(Att_C^{H \times W \times C}) \otimes Att_C^{H \times W \times C}, \quad (3)$$

$$F_{M'}^{H \times W \times C} = Att_S^{H \times W \times C} \oplus F_M^{H \times W \times C} \tag{4}$$

In the above equations, H , W , and C represent the height, width, and the number of channels of the feature maps, respectively, and A_C and A_S represent the intermediate channel and spatial attentions. The refined feature maps, denoted as $F_{M'}$, are obtained by fusing the spatial attention feature maps A_S with the input feature maps F_M . This process allowed us to enhance the representation of features and improve the quality of the final output.

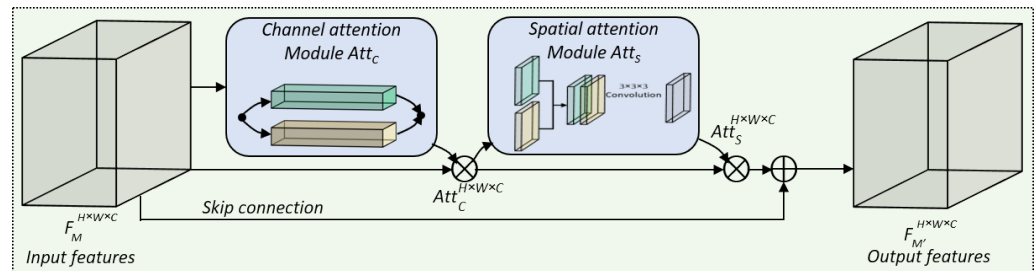


Figure 3. The visual overview of the dual channel-spatial attention module.

3.3.1. Channel Attention

In the context of image/object recognition problems, the contribution of each color channel is crucial in achieving accurate pattern recognition. CNN models leverage this information by constructing feature maps from the input image data and extracting deep discriminative features over multiple convolutional layers. However, certain color channels may be more important than others in the recognition process, and often the object recognition model takes this into account during training. This approach allows an object recognition model to capture the most important visual features of the input images and improve recognition accuracy. Prior attention-based approaches in video analysis utilized either global max pooling or global average pooling layers. However, the proposed DA-R3DCNN model surpasses this limitation and combines both pooling methods to extract more effective features. The global max pooling layer selects the maximum value from the receptive field, emphasizing highly activated values, while the global average pooling layer estimates equally weighted feature maps for each channel. By leveraging the strengths of both pooling techniques, the model can capture and highlight the most important and discriminative features in videos. This results in an improved performance in various video analysis tasks, including action recognition and spatio-temporal localization.

Once the feature maps have been computed, they are fed into a shared multilayer perceptron (MLP), which comprises two fully connected layers, each with 512 nodes. The MLP leverages a rectified linear unit (ReLU) activation function to learn the non-linearity between the two fully connected layers. The MLP then produces two distinct feature vectors— $V_{C-max}^{1 \times 1 \times C}$ and $V_{C-avg}^{1 \times 1 \times C}$ —through global max pooling and global average pooling, respectively. These feature vectors play a critical role in capturing the most salient and essential information present in the feature maps. This approach can significantly enhance the performance of the model across various video analysis tasks. After computing feature vectors from global max pooling and global average pooling, they are fused through elementwise addition and passed through a sigmoid activation function σ to obtain intermediate channel attention features $A_C^{1 \times 1 \times C}$. These features are then fused with the input feature maps $F_M^{H \times W \times C}$ through a residual skip connection using element-wise multiplication operation, resulting in the final channel attention feature maps $Att_C^{H \times W \times C}$. Figure 4 provides a visual representation of this process. Mathematically, the channel attention and its key components can be formulated as follows:

$$V_{C-max}^{1 \times 1 \times C} = fc2(ReLU(fc1(maxpool(F_M^{H \times W \times C})))), \tag{5}$$

$$V_{C-avg}^{1 \times 1 \times C} = fc2(R_{eLU}(fc1(avgpool(F_M^{H \times W \times C})))), \quad (6)$$

$$A_C^{1 \times 1 \times C} = \sigma(V_{C-max}^{1 \times 1 \times C} \oplus V_{C-avg}^{1 \times 1 \times C}), \quad (7)$$

$$Att_C^{H \times W \times C} = A_C^{1 \times 1 \times C} \otimes F_M^{H \times W \times C}, \quad (8)$$

where $fc1$ and $fc2$ denote the first and the second fully connected layer, respectively.

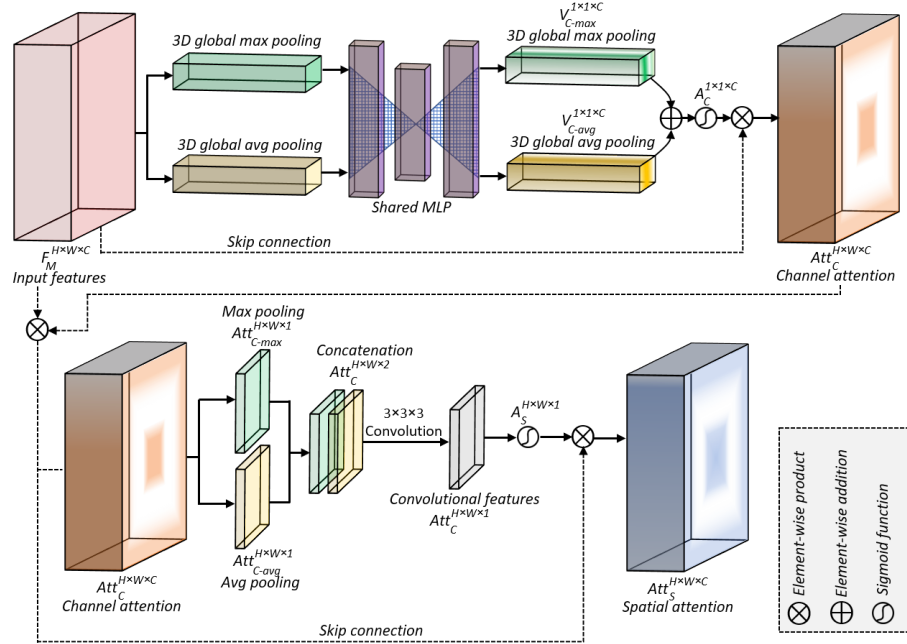


Figure 4. Architecture of the dual channel-spatial attention module.

3.3.2. Spatial Attention

The spatial attention mechanism involves learning a weighting mechanism that assigns importance scores to different spatial locations within an image. These importance scores indicate the relevance or saliency of each location in relation to the task at hand. The mechanism typically consists of trainable parameters that are optimized during the training process. To highlight the salient object-specific regions in the feature maps, DA-R3DCNN takes advantage of inter-spatial features and their relationship between channels. This allows for more accurate tracing of the target object in the feature maps. DA-R3DCNN achieves this by computing the relation of inter-spatial features between channels through max pooling and average pooling applied to the input channel attention feature maps, resulting in max-pooled channel attention $Att_{C-max}^{H \times W \times 1}$ and average-pooled channel attention $Att_{C-avg}^{H \times W \times 1}$, respectively. The concatenated max-pooled channel attention $Att_{C-max}^{H \times W \times 1}$ and average-pooled channel attention $Att_{C-avg}^{H \times W \times 1}$ are passed through a 3×3 convolutional layer $Conv^{3 \times 3}$ to form single-channel convoluted feature maps. The resulting maps are then normalized by a sigmoid activation function to produce intermediate spatial attention features $A_S^{H \times W \times 1}$. These intermediate features are fused with the input channel attention feature maps $Att_C^{H \times W \times C}$ using a residual skip connection through element-wise multiplication operations to obtain the final spatial attention feature maps $Att_S^{H \times W \times C}$ as illustrated in Figure 4. Mathematically, spatial attention and its key components can be expressed as follows:

$$Att_{C-max}^{H \times W \times 1} = maxpool(Att_C^{H \times W \times C}), \quad (9)$$

$$Att_{C-avg}^{H \times W \times 1} = avgpool(Att_C^{H \times W \times C}), \quad (10)$$

$$\mathcal{A}_S^{H \times W \times 1} = \sigma(Conv^{3 \times 3}(Att_{C-max}^{H \times W \times 1} \uplus Att_{C-avg}^{H \times W \times 1})), \quad (11)$$

$$Att_S^{H \times W \times C} = \mathcal{A}_S^{H \times W \times 1} \otimes Att_C^{H \times W \times C}, \quad (12)$$

In the above equations, \uplus denotes the concatenation operation, fusing $Att_C^{H \times W \times C}$ and $Att_S^{H \times W \times C}$.

4. Results and Discussion

In this section, we provide a comprehensive experimental evaluation of our proposed framework on various human activity recognition datasets. This section begins by providing a brief overview of the datasets used, and the implementation details and tools utilized in this study. Afterwards, we present a detailed analysis of the experimental results obtained from the proposed framework, including a comparative analysis with the state-of-the-art human action recognition methods. Additionally, an ablation study is presented, where the proposed method was analyzed with different modifications to the network architecture. Lastly, we assess the runtime performance of our proposed framework using metrics such as seconds per frame (SPF) and frames per second (FPS). We compare the obtained runtime results with the runtime results of state-of-the-art methods.

4.1. Datasets and Implementation Details

In this paper, we evaluate the performance of our DA-R3DCNN method on four publicly available benchmark datasets for human activity recognition tasks: UCF11 [27], HMDB51 [28], UCF50 [29], and UCF101 [30]. These datasets are exclusively created for human activity recognition task, and contain videos collected from different sources and have different lengths, resolutions, and viewpoints of humans actions in the videos. The UCF11 [27] dataset comprises 1640 videos collected from YouTube which are then categorized into 11 distinct action classes of human actions. All videos in the dataset are annotated by action appearance, where each video has a spatial resolution of 320×240 . The HMDB51 [28] dataset is a relatively large dataset, containing 6849 videos, categorized into 51 categories. This dataset has a wide range of variation in camera motion, object scale, view point, and background clutter, which makes it challenging for human action recognition tasks. Videos in this dataset are collected from different sources, including movies, YouTube, Prelinger archive, and Google videos. The UCF50 [29] dataset consists of 6676 realistic videos collected from YouTube, containing human actions performed by different subjects in different environments with varying viewpoints. Videos in this dataset are divided into 50 distinct actions by action appearance in the video. Finally, the UCF101 [30] is the largest dataset amongst the above mentioned datasets, containing 13,320 videos of different human actions. This dataset is the extended version of the UCF50 [29], having comparatively more videos and large variation in actions, categorized into 101 action classes. The number of videos per class in each dataset is approximately 100 to 200, and the duration of video clips is in between 2 and 3 s, with a frame rate of 25 FPS.

For implementation, we used Python version 3 utilizing Keras with a TensorFlow 2.0 backend. We performed the experiments on a computer system equipped with an Intel(R) Xeon(R) CPU E5-2640, operating at a frequency of 2.50 GHz, and 32 GB of dedicated main memory (RAM). Additionally, we employed two dedicated Tesla GPUs with compute capabilities of 7.5 as hardware resources along with the Nvidia CUDA 11.0 library. To train the proposed DA-R3DCNN model, we used 70% of data for training, 20% for validation, and 10% for testing the model performance after training. The same data splitting ratio was considered for each dataset used in the experiments of this paper. It is worth mentioning here that each set of data (including training, validation, and test sets) contained

all classes, where each class consisted of videos as per their corresponding split ratios (training 70%, validation 20%, and test 10%). Further, for model's weights adjustment and convergence, we employed the Adam optimizer with a fixed learning rate of 0.0001 and utilized categorical cross-entropy loss to adjust the network weights. We set the input sequence length to 16 frames, allowing the DA-R3DCNN model to extract spatiotemporal information by sliding multiple 3D kernels over the sequence of frames. To obtain and compare the performance of our DA-R3DCNN method with the state-of-the-art methods, we used two different evaluation metrics: model accuracy performance evaluation and runtime performance evaluation. For accuracy comparison, we compared the average accuracy of our model for each dataset with the state-of-the-art methods, whereas, for runtime performance comparison, we used two metrics: FPS and SPF.

4.2. Quantitative Evaluation

In this section, we present the performance evaluation of the proposed DA-R3DCNN framework. To evaluate the performance of our proposed framework, we conducted quantitative performance evaluation experiments on the four benchmark datasets: UCF11, UCF50, HMDB51, and UCF101. To better analyze the model performance for a specific class in each dataset, we computed the confusion matrix (reflecting true positive, false positive, true negative, and false negative predictions) based on the model predictions for each dataset. The obtained confusion matrices for UCF11, UCF50, HMDB51, and UCF101 datasets are depicted in Figure 5. Further, the obtained quantitative results for both with and without the dual-attention mechanism are listed in Table 2. Based on the listed values in Table 2, it is evident that the proposed method demonstrates strong performance when combined with the dual-attention module. For instance, when applied to the UCF11 dataset, the proposed method achieved an accuracy of 98.6% with the dual-attention module, while achieving an accuracy of 93.1% without the dual-attention module. When applied to the HMDB51 dataset, the proposed method achieved an accuracy of 82.5% when coupled with the dual-attention module and attained an accuracy of 77.2% without the module. Similarly, on the UCF101 dataset, the proposed method obtained an accuracy of 97.8% with the dual-attention module and had an accuracy of 93.6% without the module. The listed accuracy values demonstrate that the proposed method with the dual-attention module achieved improvements of 5.5%, 5.6%, 5.3%, and 4.2% for the UCF11, UCF50, HMDB51, and UCF101 datasets, respectively. Thus, the obtained noticeable improvements in accuracies for each dataset validate the effectiveness of the dual-attention module for the activity recognition task.

4.3. Comparison with the State-of-the-Art Methods

This section presents a comprehensive quantitative comparison between our proposed DA-R3DCNN model and state-of-the-art methods for human action recognition. The comparisons were based on average accuracy and were conducted on the UCF11, UCF50, HMDB51, and UCF101 datasets, as shown in Tables 3, 4, 5, and 6, respectively. Table 3 showcases the results that indicate that our proposed DA-R3DCNN achieved the highest accuracy of 98.6%, surpassing all other methods. The Fusion-based discriminative features method [31] came in second place, with an accuracy of 97.8%. Among the comparative methods, the lowest accuracy on the UCF11 dataset was obtained by the Local-global features + QSVM method [32], which achieved an accuracy of 82.6%. The rest of the comparative methods included Multi-task hierarchical clustering [33], BT-LSTM [34], Deep autoencoder [35], Two-stream attention LSTM [36], Weighted entropy-variances-based feature selection [37], Dilated CNN + BiLSTM + RB [38], DS-GRU [39], Squeezed CNN [40], BS-2SCN [41], and 3DCNN [42]. These methods achieved accuracies of 89.7%, 85.3%, 96.2%, 96.9%, 94.5%, 89.0%, 97.1%, 87.4%, 90.1%, and 85.1%, respectively. Based on the comparative assessment, the proposed DA-R3DCNN achieved an average accuracy improvement of 8.47% as compared to the average results of state-of-the-art methods on the UCF11 dataset.

For the UCF50 dataset, the results presented in Table 4 validate that the proposed

DA-R3DCNN framework achieved the best results by attaining an accuracy of 97.4%, followed by the Deep autoencoder [35] method, which obtained a runner-up accuracy of 96.4%. Among all the comparative methods on the UCF50 dataset, the Local-global features + QSVM method [32] achieved the lowest accuracy of 69.4%. The other methods in the comparison included Multi-task hierarchical clustering [33], Ensemble model with swarm-based optimization [43], DS-GRU [39], Hybrid Deep Evolving Neural Networks [44], ViT + Multi Layer LSTM [45], and 3DCNN [42], which achieved accuracies of 93.2%, 92.2%, 95.2%, 77.3%, 96.1%, and 82.6%, respectively. Upon analyzing the comparative results presented in Table 4, it is evident that the proposed DA-R3DCNN exhibited an average accuracy improvement of 10.93% over the average results of the state-of-the-art methods on the UCF50 dataset.

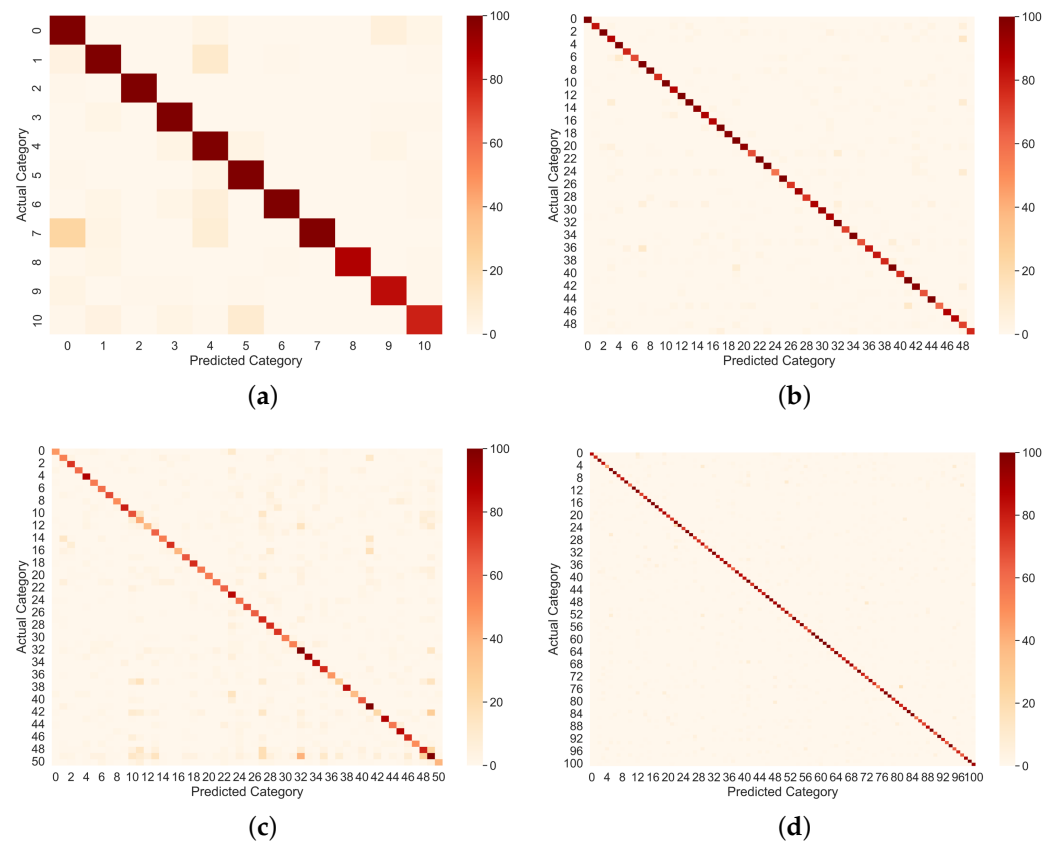


Figure 5. Confusion matrices computed for the proposed DA-R3DCNN framework for the test sets of four experimented datasets: (a) UCF11 dataset, (b) UCF50 dataset, (c) HMDB51 dataset, and (d) UCF101 dataset.

Table 2. The average accuracies obtained by our proposed framework with and without the dual-attention module on UCF11, UCF50, HMDB51, and UCF101 datasets.

Dataset	Accuracy (%)	
	Without Dual Attention	With Dual Attention
UCF11	93.1	98.6
UCF50	91.8	97.4
HMDB51	77.2	82.5
UCF101	93.6	97.8

Table 3. Comparative analysis of the proposed DA-R3DCNN with the state-of-the-art methods on the UCF11 dataset.

Method	Year	Accuracy (%)
Multi-task hierarchical clustering [33]	2017	89.7
BT-LSTM [34]	2018	85.3
Deep autoencoder [35]	2019	96.2
Two-stream attention LSTM [36]	2020	96.9
Weighted entropy-variances based feature selection [37]	2021	94.5
Dilated CNN+BiLSTM+RB [38]	2021	89.0
DS-GRU [39]	2021	97.1
Local-global features + QSVM [32]	2021	82.6
Squeezed CNN [40]	2022	87.4
Fusion-based discriminative features [31]	2022	97.8
BS-2SCN [41]	2022	90.1
3DCNN [42]	2022	85.1
DA-R3DCNN (Proposed)	2023	98.6

Bold value represents the best accuracy, where the italic value indicates the runner up accuracy.

Table 4. Comparative analysis of the proposed DA-R3DCNN with the state-of-the-art methods on the UCF50 dataset.

Method	Year	Accuracy (%)
Multi-task hierarchical clustering [33]	2017	93.2
Deep autoencoder [35]	2019	96.4
Ensemble model with swarm-based optimization [43]	2021	92.2
DS-GRU [39]	2021	95.2
Local-global features + QSVM [32]	2021	69.4
Hybrid Deep Evolving Neural Networks [44]	2022	77.3
ViT + Multi Layer LSTM [45]	2022	96.1
3DCNN [42]	2022	82.6
DA-R3DCNN (Proposed)	2023	97.4

Bold value represents the best accuracy, where the italic value indicates the runner-up accuracy.

Table 5. Comparative analysis of the proposed DA-R3DCNN with the state-of-the-art methods on the HMDB51 dataset.

Method	Year	Accuracy (%)
Multi-task hierarchical clustering [33]	2017	51.4
STPP+LSTM [46]	2017	70.5
Optical flow + multi-layer LSTM [47]	2018	72.2
TSN [48]	2018	70.7
IP-LSTM [49]	2019	58.6
Deep autoencoder [35]	2019	70.3
TS-LSTM + temporal-inception [50]	2019	69.0
HATNet [51]	2019	74.8
Correlational CNN + LSTM [52]	2020	66.2
STDAN [53]	2020	56.5
DB-LSTM+SSPF [54]	2021	75.1
DS-GRU [39]	2021	72.3
TCLC [55]	2021	71.5
Evidential deep learning [56]	2021	77.0
Semi-supervised temporal gradient learning [57]	2022	75.9
BS-2SCN [41]	2022	71.3
ViT + Multi Layer LSTM [45]	2022	73.7
MAT-EffNet [58]	2023	70.9
DA-R3DCNN (Proposed)	2023	82.5

Bold value represents the best accuracy, where the italic value indicates the runner up accuracy.

In the case of the challenging HMDB51 dataset, the proposed DA-R3DCNN achieved the highest accuracy of 82.5%, surpassing all other comparative methods considered in our assessments. The Evidential deep learning method [56] emerged as the runner-up with an accuracy of 77.0%. Among the comparative methods, Multi-task hierarchical clustering [33] achieved the lowest accuracy of 51.4% on the HMDB51 dataset. Other comparative methods included STPP + LSTM [46], TSN [48], Deep autoencoder [35], TS-LSTM + temporal-inception [50], HATNet [51], Correlational CNN + LSTM [52], STDAN [53], DB-LSTM + SSPF [54], DS-GRU [39], TCLC [55], Semi-supervised temporal gradient learning [57], BS-2SCN [41], ViT + Multi Layer LSTM [45], and MAT-EffNet [58]. These methods achieved accuracies of 70.5%, 72.2%, 70.7%, 58.6%, 70.3%, 69.0%, 74.8%, 66.2%, 56.5%, 75.1%, 72.3%, 71.5%, 75.9%, 71.3%, 73.7%, and 70.9%, respectively. From the list of comparative assessments in Table 5, the proposed DA-R3DCNN achieved an average improvement of 19.01%, in terms of accuracy over the average results of the state-of-the-art methods on the HMDB51 dataset.

Table 6. Comparative analysis of the proposed DA-R3DCNN with state-of-the-art methods on the UCF101 dataset.

Method	Year	Accuracy (%)
Multi-task hierarchical clustering [33]	2016	76.3
Saliency-aware 3DCNN with LSTM [59]	2016	84.0
Spatio-temporal multilayer networks [60]	2017	87.0
Long-term temporal convolutions [20]	2017	82.4
CNN + Bi-LSTM [8]	2017	92.8
OFF [61]	2018	96.0
TVNet [62]	2018	95.4
Attention cluster [63]	2018	94.6
Videolstm [17]	2018	89.2
Two stream convnets [64]	2018	84.9
Mixed 3D-2D convolutional tube [65]	2018	88.9
TS-LSTM + Temporal-inception [50]	2019	91.1
TSN + TSM [66]	2019	94.3
STM [67]	2019	96.2
Correlational CNN + LSTM [52]	2020	92.8
ResCNN-DBLSTM [68]	2020	94.7
SC-BDLSTM [69]	2021	94.2
Ensemble model with swarm-based optimization [43]	2021	96.3
BS-2SCN [41]	2022	90.1
TDS-BiLSTM [70]	2022	94.7
META-RGB+Flow [71]	2022	96.0
Spurious-3D Residual Network [72]	2023	95.6
DA-R3DCNN (Proposed)	2023	97.8

Bold value represents the best accuracy, where the italic value indicates the runner up accuracy.

Finally, for the UCF101 dataset, the results listed in Table 6 demonstrate that the proposed DA-R3DCNN surpassed all other comparative methods by achieving the highest accuracy of 97.8%. The Ensemble model with swarm-based optimization method [43] secured the runner-up position with an accuracy of 96.3%. On the UCF101 dataset, the Multi-task hierarchical clustering [33] obtained the lowest accuracy of 76.3% among all the comparative methods. Additional comparative methods included Saliency-aware 3DCNN with LSTM [59], Spatio-temporal multilayer networks [60], Long-term temporal convolutions [20], CNN + Bi-LSTM [8], OFF [61], TVNet [62], Attention cluster [63], Videolstm [17], Two stream convnets [64], Mixed 3D-2D convolutional tube [65], TS-LSTM + Temporal-inception [50], TSN + TSM [66], STM [67], Correlational CNN + LSTM [52], ResCNN-DBLSTM [68], SC-BDLSTM [69], BS-2SCN [41], TDS-BiLSTM [70], META-RGB + Flow [71], and Spurious-3D Residual Network [72]. These methods achieved accuracies of 84.0%, 87.0%, 82.4%, 92.8%, 96.0%, 95.4%, 94.6%, 89.2%, 84.9%, 88.9%, 91.1%, 94.3%, 96.2%, 92.8%, 94.7%, 94.2%, 90.1%, 94.7%, 96.0%, and 95.6%, respectively. Furthermore, it is evident

from the results listed in Table 6 that the proposed DA-R3DCNN exhibited an average accuracy improvement of 7.17% as compared to the average results of state-of-the-art methods on the UCF101 dataset. Additionally, for clear understanding of comparative assessment, we also present the visual overview of comparative analysis of our proposed DA-R3DCNN with the state-of-the-art human action recognition methods on UCF11, UCF50, HMDB51, and UCF101 datasets in Figure 6. The quantitative comparisons depicted in Figure 6 illustrate the performance of comparative methods published up to 2023.

Furthermore, to assess the performance generalization of our method, we conducted an analysis of confidence intervals, following the methodology outlined in [73]. This analysis was performed on each dataset used in this study, and a comparison was made between the confidence intervals of our proposed method and those of the state-of-the-art approaches. It is worth noting that a confidence level of 95% was employed for estimating the confidence intervals of both our method and the state-of-the-art methods. The resulting confidence interval values for our proposed method and the state-of-the-art methods are given in Table 7. Upon examining these values, we observe that our proposed method exhibited higher confidence levels with narrower intervals on all the dataset when compared to the state-of-the-art methods. For instance, for the UCF11 dataset, the confidence interval of our proposed method spanned from 97.31 to 99.10, with a range of only 1.79. In contrast, the average confidence interval of the state-of-the-art methods ranged from 87.74 to 94.20, showing a comparatively larger range of 6.46. Similarly, for the UCF50 dataset, our proposed method achieved a confidence interval between 96.78 and 98.42, with a small range of 1.64, while the state-of-the-art methods had an average confidence interval ranging from 79.86 to 95.74, indicating a larger range of 15.88. Analyzing the HMDB51 dataset, we observe that our proposed method had a confidence interval of 92.21 to 94.16, with a narrow range of 1.95. In contrast, the state-of-the-art methods exhibited an average confidence interval ranging from 65.97 to 72.67, demonstrating a comparatively larger range of 6.70. Lastly, for the UCF101 dataset, our proposed method demonstrated a confidence interval between 96.89 and 98.46, with a small range of 1.57, whereas the state-of-the-art methods have an average confidence interval ranging from 88.93 to 93.57, indicating a larger range of 4.64. It is worth mentioning here that our proposed method consistently achieved higher confidence levels across all datasets, with narrower intervals, in comparison to the state-of-the-art methods. This observation serves to verify the effectiveness of our proposed method in surpassing existing approaches in terms of performance generalization.

The conducted comparative assessments validate the effectiveness of the proposed DA-R3DCNN based on the obtained improvement in the results, across each dataset used in this work. These results verify the robustness of our proposed DA-R3DCNN framework over the state-of-the-art methods for human action recognition task.

Table 7. The obtained confidence interval values (with 95% confidence) for our proposed method and state-of-the-art mainstream methods.

Dataset	State-of-the-Art Methods	Ours
UCF11	[87.74–94.20]	[97.31–99.10]
UCF50	[79.86–95.74]	[96.78–98.42]
HMDB51	[65.97–72.67]	[92.21–94.16]
UCF101	[88.93–93.57]	[96.89–98.46]

First value in the square brackets represents the lower bound and the second value represents the upper bound. Together, the lower and upper bounds represent the confidence interval.

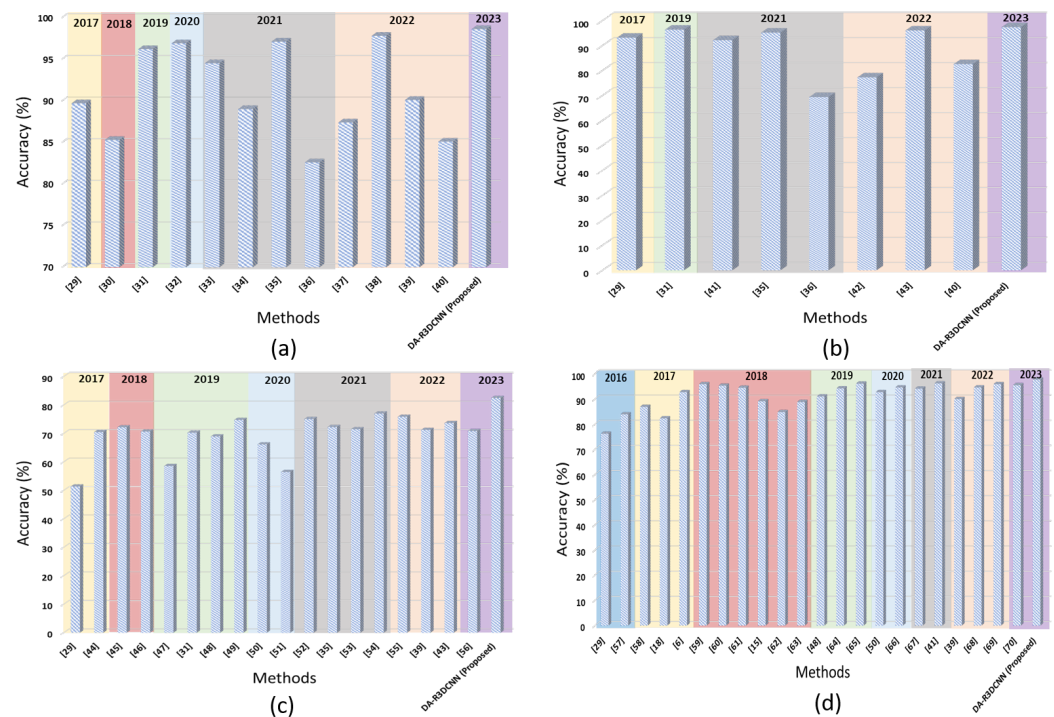


Figure 6. The graphical overview of the conducted comparative analysis of our proposed DA-R3DCNN with the state-of-the-art methods on (a) UCF11 dataset, (b) UCF50 dataset, (c) HMDB51 dataset, and (d) UCF101 dataset.

4.4. Run Time Analysis

In this section, we examine the inference time of our proposed DA-R3DCNN framework and assess its suitability for real-time human activity recognition tasks, considering metrics such as SPF and FPS. To evaluate the overall run time performance, we conducted inference time measurements of the proposed DA-R3DCNN model on both GPU and CPU computing platforms. These measurements were then compared with the runtime results of the state-of-the-art human activity recognition methods, and the findings are presented in Table 8. This analysis provides a comprehensive perspective on the run time efficiency of our proposed DA-R3DCNN model across different computing platforms. The results presented in Table 8 highlight the superior inference efficiency of the proposed DA-R3DCNN model as compared to state-of-the-art methods, as demonstrated by SPF and FPS metrics on both GPU and CPU computing platforms. The findings indicate that, when utilizing GPU resources, our proposed DA-R3DCNN achieved the best SPF of 0.0045 and an FPS of 240. The runner-up method, OFF [61], achieved an SPF of 0.0048 and an FPS of 215. Conversely, the Videolstm [17] method exhibited the highest SPF of 0.0940 and the lowest FPS of 10.6, indicating the least favorable run time performance among all the comparative methods. These results underscore the exceptional inference efficiency of our proposed DA-R3DCNN model when compared to existing approaches. On the CPU computing platform, the proposed DA-R3DCNN framework demonstrated significant superiority over existing methods, achieving an SPF of 0.0058 and an FPS of 187. In comparison, the second-best performing method, Optical-flow + Multi-layer LSTM [47], achieved an SPF of 0.18 and an FPS of 3.5. Conversely, the Deep autoencoder [35] method exhibited the poorest runtime performance with an SPF of 0.43 and an FPS of 1.5. These results further validate the exceptional run time efficiency of our proposed DA-R3DCNN framework when compared to alternative approaches on the CPU computing platform.

Further, to ensure a fair comparison of the run time results obtained for both GPU and CPU platforms, we scaled the run time results (as in [74]) of the state-of-the-art methods to match the hardware resources utilized in our study (i.e., a 2.5 GHz CPU and a

585 MHz GPU). The scaled run time results are provided in the second section of Table 8, enabling an equitable assessment and comparison of the performance of the proposed DA-R3DCNN framework against existing methods. Analyzing the scaled results presented in Table 8, it becomes evident that scaling amplifies the advantages of the proposed DA-R3DCNN model in terms of SPF and FPS metrics for both GPU and CPU computing platforms. When utilizing GPU resources, the proposed DA-R3DCNN outperformed other methods with the best SPF of 0.0045 and an FPS of 240. The STPP + LSTM [46] method secured the second-best position, with SPF and FPS values of 0.0063 and 154.6, respectively. These findings highlight the enhanced performance of the proposed DA-R3DCNN model when considering the scaled runtime results, solidifying its superiority over alternative approaches. The Videolstm [17] method had the highest SPF of 0.1606 and lowest FPS of 6.2, indicating the worst run time results amongst all the comparative methods. When running on CPU computing platform, the proposed DA-R3DCNN framework had the lowest SPF of 0.0058 and highest FPS of 187, indicating the best results obtained on CPU resources as compared to other comparative methods. Among the scaled results in Table 8, the Optical-flow + Multi-layer LSTM [47] emerged as the runner-up with an SPF of 0.23 and an FPS of 2.6 on the CPU computing platform. On the other hand, the Deep autoencoder [35] method exhibited the least favorable performance on CPU resources, achieving an SPF of 0.56 and an FPS of 1.1. These findings further solidify the superior run time performance of the proposed DA-R3DCNN model when compared to alternative methods on the CPU computing platform.

Table 8. Comparison of the run time performance between our proposed DA-R3DCNN framework and state-of-the-art human action recognition methods, considering both scaled and unscaled results.

Method	Seconds per Frame (SPF)		Year	Frames per Second (FPS)	
	GPU	CPU		GPU	CPU
Without Scaling					
STPP + LSTM [46]	0.0053	-	2017	186.6	-
CNN + Bi-LSTM [8]	0.0570	-	2017	20	-
OFF [61]	<i>0.0048</i>	-	2018	215	-
Videolstm [17]	0.0940	-	2018	10.6	-
Optical-flow + Multi-layer LSTM [47]	0.0356	<i>0.18</i>	2018	30	3.5
Deep autoencoder [35]	0.0430	0.43	2019	24	1.5
TSN + TSM [66]	0.0167	-	2019	60	-
IP-LSTM [49]	0.0431	-	2019	23.2	-
STDAN [53]	0.0075	-	2020	132	-
DS-GRU [39]	0.0400	-	2021	25	-
DA-R3DCNN (Proposed)	0.0045	0.0058	2023	240	187
With Scaling					
STPP + LSTM [46]	<i>0.0063</i>	-	2017	154.6	-
CNN + Bi-LSTM [8]	0.0974	-	2017	11.7	-
OFF [61]	0.0082	-	2018	125	-
Videolstm [17]	0.1606	-	2018	6.2	-
Optical-flow + Multi-layer LSTM [47]	0.0608	<i>0.23</i>	2018	17.5	2.6
Deep autoencoder [35]	0.0735	0.56	2019	14	1.1
TSN + TSM [66]	0.0458	-	2019	21.8	-
IP-LSTM [49]	0.0736	-	2019	13.57	-
STDAN [53]	0.0128	-	2020	77.2	-
DS-GRU [39]	0.0683	-	2021	14.6	-
DA-R3DCNN (Proposed)	0.0045	0.0058	2023	240	187

The best and runner-up SPF and FPS scores for GPU and CPU are highlighted in bold and italic text, respectively.

It is evident from the listed scaled and non-scaled results in Table 8 that the proposed DA-R3DCNN provides significant improvement for both GPU and CPU computing platforms. For instance, for non-scaled run time results, the proposed DA-R3DCNN provided an improvement of up to 7× for SPF and 3× for FPS metric when running on GPU resources. When running on CPU resources, the proposed DA-R3DCNN achieved an improvement of

up to $52\times$ for SPF and $74\times$ for FPS for the non-scaled run time results. Similarly, for the scaled run time results, the proposed DA-R3DCNN provided an improvement of $13\times$ for SPF and $5\times$ for FPS when running on GPU resources. When running on CPU resources, the proposed DA-R3DCNN framework achieved an improvement of $68\times$ for SPF and $100\times$ for FPS. These results show the efficiency and applicability of the proposed DA-R3DCNN method for real-time human activity recognition in resource constraint environments.

5. Conclusions and Future Research Directions

In this work, we have proposed an attention-driven 3DCNN with residual skip connections for recognizing human activities in videos. The proposed method combines the powerful characteristics of dual channel-spatial attention and residual 3D convolutional neural network (3DCNN) into a unified framework for efficient modeling of human actions and single instance training. The utilized dual channel-spatial attention mechanism incorporates both channel and spatial attentions, enabling the extraction of highly discriminative features from regions of interest related to the objects involved in the activities. This results in the generation of high-quality feature maps containing object saliency-aware features boosting the overall learning process of the proposed residual 3DCNN network. By employing residual 3DCNN coupled with dual attention, our method, known as DA-R3DCNN, effectively captures the temporal dynamics of human actions through the use of multiple 3D kernels. By leveraging the knowledge acquired from the immediately preceding frames within the input sequence, the model becomes capable of learning the spatial and temporal relationships within unseen frames. This enables the model to grasp the connections and patterns existing between different frames. The incorporation of attention-guided learning further enhances our method's capability to acquire spatial and temporal understanding of human actions, leading to improved learning performance during training and enhanced prediction accuracy during inference.

We have extensively evaluated the performance of our proposed DA-R3DCNN method on four widely recognized benchmark datasets for human action recognition: UCF11, UCF50, HMDB51, and UCF101. These datasets are well-established in the research community and serve as reliable benchmarks for comparison. Through rigorous experimentation and comparison with the state-of-the-art approaches, we have demonstrated the superiority of our method in terms of model robustness and computational efficiency. The obtained results validate the efficacy of our approach in tackling the challenges of human action recognition across diverse datasets. Further, we have assessed the run time performance of our proposed framework in terms of seconds per frame (SPF) and frames per second (FPS) on both CPU and GPU execution environments. This analysis has allowed us to measure the computational efficiency of our method, and to provide valuable insights into the speed of frame processing and overall video processing capabilities across different hardware configurations. The run time assessment results clearly indicate that the proposed DA-R3DCNN method exhibits remarkable improvements when leveraging GPU resources. It demonstrates a significant enhancement of up to $13\times$ in SPF and $5\times$ in FPS metrics as compared to the state-of-the-art methods. Additionally, even when limited to CPU resources, our approach achieves substantial advancements, with SPF improving by $68\times$ and FPS by $100\times$ as compared to existing approaches. These findings establish that the proposed DA-R3DCNN method is exceptionally well-suited for real-time human activity recognition on resource-constrained devices.

While our current implementation of the DA-R3DCNN method leverages the spatial attention mechanism (channel and spatial attention), which has proven to be highly effective, in our future work, we plan to incorporate a temporal attention mechanism for precise temporal localization of human activities within video scenes. Additionally, we are actively exploring the integration of multi-modal data, which holds significant potential for recognizing complex human activities in uncertain environments. These future advancements aim to further enhance the capabilities of our method in capturing both spatial and temporal dynamics for improved activity recognition performance.

Author Contributions: Conceptualization, H.U. and A.M.; methodology, H.U. and A.M.; software, H.U.; validation, H.U.; formal analysis, H.U. and A.M.; investigation, A.M.; resources, A.M.; data curation, H.U.; writing—original draft preparation, H.U. and A.M.; writing—review and editing, H.U. and A.M.; visualization, H.U. and A.M.; supervision, A.M.; project administration, A.M.; funding acquisition, A.M. All authors have read and agreed to the published version of the manuscript.

Funding: This research was supported in part by the Air Force Office of Scientific Research (AFOSR) Contract Number FA9550-22-1-0040. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the Air Force, the Air Force Research Laboratory (AFRL), and/or AFOSR.

Data Availability Statement: Not applicable.

Acknowledgments: The authors would like to acknowledge Erik Blasch from the Air Force Research Laboratory (AFRL) for his guidance and support on the project.

Conflicts of Interest: The authors declare no conflict of interest.

Abbreviations

DNNs	Deep neural networks
CNN	Convolutional neural network
3DCNN	3D Convolutional neural network
CBAM	convolutional block attention module
DA-R3DCNN	dual-attention residual 3D convolutional neural network
RNN	Recurrent neural network
LSTM	Long short-term memory
SPF	Seconds per frame
FPS	Frames per second

References

- Mahmoud, A.; Hu, J.S.; Waslander, S.L. Dense Voxel Fusion for 3D Object Detection. In Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision, Waikoloa, HI, USA, 3–7 January 2023; pp. 663–672.
- Muhammad, K.; Ullah, H.; Khan, S.; Hijji, M.; Lloret, J. Efficient Fire Segmentation for Internet-of-Things-Assisted Intelligent Transportation Systems. *IEEE Trans. Intell. Transp. Syst.* **2022**, *early access*. [\[CrossRef\]](#)
- Muhammad, K.; Hussain, T.; Ullah, H.; Del Ser, J.; Rezaei, M.; Kumar, N.; Hijji, M.; Bellavista, P.; de Albuquerque, V.H.C. Vision-Based Semantic Segmentation in Scene Understanding for Autonomous Driving: Recent Achievements, Challenges, and Outlooks. *IEEE Trans. Intell. Transp. Syst.* **2022**, *23*, 22694–22715. [\[CrossRef\]](#)
- Munir, A.; Blasch, E.; Kwon, J.; Kong, J.; Aved, A. Artificial Intelligence and Data Fusion at the Edge. *IEEE Aerosp. Electron. Syst. Mag.* **2021**, *36*, 62–78. [\[CrossRef\]](#)
- Munir, A.; Kwon, J.; Lee, J.H.; Kong, J.; Blasch, E.; Aved, A.; Muhammad, K. FogSurv: A Fog-Assisted Architecture for Urban Surveillance Using Artificial Intelligence and Data Fusion. *IEEE Access* **2021**, *9*, 111938–111959. [\[CrossRef\]](#)
- Tran, A.; Cheong, L.F. Two-Stream Flow-Guided Convolutional Attention Networks for Action Recognition. In Proceedings of the IEEE International Conference on Computer Vision Workshops, Venice, Italy, 22–29 October 2017; pp. 3110–3119.
- Tran, D.; Bourdev, L.; Fergus, R.; Torresani, L.; Paluri, M. Learning Spatiotemporal Features with 3D Convolutional Networks. In Proceedings of the IEEE International Conference on Computer Vision, Santiago, Chile, 7–13 December 2015; pp. 4489–4497.
- Ullah, A.; Ahmad, J.; Muhammad, K.; Sajjad, M.; Baik, S.W. Action Recognition in Video Sequences Using Deep Bi-Directional LSTM with CNN Features. *IEEE Access* **2017**, *6*, 1155–1166. [\[CrossRef\]](#)
- Karpathy, A.; Toderici, G.; Shetty, S.; Leung, T.; Sukthankar, R.; Fei-Fei, L. Large-Scale Video Classification with Convolutional Neural Networks. In Proceedings of the IEEE conference on Computer Vision and Pattern Recognition, Columbus, OH, USA, 23–28 June 2014; pp. 1725–1732.
- Chéron, G.; Laptev, I.; Schmid, C. P-CNN: Pose-based CNN Features for Action Recognition. In Proceedings of the IEEE International Conference on Computer Vision, Santiago, Chile, 7–13 December 2015; pp. 3218–3226.
- Wang, L.; Xiong, Y.; Wang, Z.; Qiao, Y.; Lin, D.; Tang, X.; Van Gool, L. Temporal Segment Networks: Towards Good Practices for Deep Action Recognition. In Proceedings of the European Conference on Computer Vision, Amsterdam, The Netherlands, 11–14 October 2016; pp. 20–36.
- Girdhar, R.; Ramanan, D.; Gupta, A.; Sivic, J.; Russell, B. Actionvlad: Learning Spatio-Temporal Aggregation for Action Classification. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 971–980.
- Zhang, H.; Liu, D.; Xiong, Z. Two-Stream Action Recognition-Oriented Video Super-Resolution. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Seoul, Republic of South Korea, 27–28 October 2019; pp. 8799–8808.

14. Donahue, J.; Anne Hendricks, L.; Guadarrama, S.; Rohrbach, M.; Venugopalan, S.; Saenko, K.; Darrell, T. Long-Term Recurrent Convolutional Networks for Visual Recognition and Description. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Boston, MA, USA, 7–12 June 2015; pp. 2625–2634.
15. Srivastava, N.; Mansimov, E.; Salakhudinov, R. Unsupervised Learning of Video Representations Using LSTMs. In Proceedings of the International Conference on Machine Learning, PMLR, Lille, France, 6–11 July 2015; pp. 843–852.
16. Sharma, S.; Kiros, R.; Salakhutdinov, R. Action Recognition Using Visual Attention. *arXiv* **2015**, arXiv:1511.04119.
17. Li, Z.; Gavriluyk, K.; Gavves, E.; Jain, M.; Snoek, C.G. VideoLSTM Convolves, Attends and Flows for Action Recognition. *Comput. Vis. Image Underst.* **2018**, *166*, 41–50. [[CrossRef](#)]
18. Sudhakaran, S.; Escalera, S.; Lanz, O. LSTA: Long Short-Term Attention for Egocentric Action Recognition. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Long Beach, CA, USA, 15–20 June 2019; pp. 9954–9963.
19. Diba, A.; Fayyaz, M.; Sharma, V.; Karami, A.H.; Arzani, M.M.; Yousefzadeh, R.; Van Gool, L. Temporal 3D Convnets: New Architecture and Transfer Learning for Video Classification. *arXiv* **2017**, arXiv:1711.08200.
20. Varol, G.; Laptev, I.; Schmid, C. Long-Term Temporal Convolutions for Action Recognition. *IEEE Trans. Pattern Anal. Mach. Intell.* **2017**, *40*, 1510–1517. [[CrossRef](#)]
21. Diba, A.; Fayyaz, M.; Sharma, V.; Arzani, M.M.; Yousefzadeh, R.; Gall, J.; Van Gool, L. Spatio-Temporal Channel Correlation Networks for Action Classification. In Proceedings of the European Conference on Computer Vision (ECCV), Munich, Germany, 8–14 September 2018; pp. 284–299.
22. Hussein, N.; Gavves, E.; Smeulders, A.W. Timeception for Complex Action Recognition. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Long Beach, CA, USA, 15–20 June 2019; pp. 254–263.
23. Li, X.; Shuai, B.; Tighe, J. Directional Temporal Modeling for Action Recognition. In *Proceedings of the Computer Vision—ECCV 2020: 16th European Conference, Glasgow, UK, 23–28 August 2020*; Proceedings, Part VI 16; Springer: Berlin/Heidelberg, Germany, 2020; pp. 275–291.
24. Huang, G.; Liu, Z.; Van Der Maaten, L.; Weinberger, K.Q. Densely Connected Convolutional Networks. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 4700–4708.
25. He, K.; Zhang, X.; Ren, S.; Sun, J. Deep Residual Learning for Image Recognition. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 27–30 June 2016; pp. 770–778.
26. Woo, S.; Park, J.; Lee, J.Y.; Kweon, I.S. CBAM: Convolutional Block Attention Module. In Proceedings of the European conference on computer vision (ECCV), Munich, Germany, 8–14 September 2018; pp. 3–19.
27. Liu, J.; Luo, J.; Shah, M. Recognizing Realistic Actions From Videos “in the Wild”. In Proceedings of the 2009 IEEE Conference on Computer Vision and Pattern Recognition, Miami, FL, USA, 20–25 June 2009; pp. 1996–2003.
28. 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, Barcelona, Spain, 6–13 November 2011; pp. 2556–2563.
29. Reddy, K.K.; Shah, M. Recognizing 50 Human Action Categories of Web Videos. *Mach. Vis. Appl.* **2013**, *24*, 971–981. [[CrossRef](#)]
30. Soomro, K.; Zamir, A.R.; Shah, M. UCF101: A Dataset of 101 Human Actions Classes from Videos in the Wild. *arXiv* **2012**, arXiv:1212.0402.
31. Karuppanan, K.; Darmanayagam, S.E.; Cyril, S.R.R. Human Action Recognition Using Fusion-Based Discriminative Features and Long Short Term Memory Classification. *Concurr. Comput. Pract. Exp.* **2022**, *34*, e7250. [[CrossRef](#)]
32. Al-Obaidi, S.; Al-Khafaji, H.; Abhayaratne, C. Making Sense of Neuromorphic Event Data for Human Action Recognition. *IEEE Access* **2021**, *9*, 82686–82700. [[CrossRef](#)]
33. Liu, A.A.; Su, Y.T.; Nie, W.Z.; Kankanhalli, M. Hierarchical Clustering Multi-Task Learning for Joint Human Action Grouping and Recognition. *IEEE Trans. Pattern Anal. Mach. Intell.* **2016**, *39*, 102–114. [[CrossRef](#)] [[PubMed](#)]
34. Ye, J.; Wang, L.; Li, G.; Chen, D.; Zhe, S.; Chu, X.; Xu, Z. Learning Compact Recurrent Neural Networks With Block-Term Tensor Decomposition. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–22 June 2018; pp. 9378–9387.
35. Ullah, A.; Muhammad, K.; Haq, I.U.; Baik, S.W. Action Recognition Using Optimized Deep Autoencoder and CNN for surveillance Data Streams of Non-Stationary Environments. *Future Gener. Comput. Syst.* **2019**, *96*, 386–397. [[CrossRef](#)]
36. Dai, C.; Liu, X.; Lai, J. Human Action Recognition Using Two-Stream Attention Based LSTM Networks. *Appl. Soft Comput.* **2020**, *86*, 105820. [[CrossRef](#)]
37. Afza, F.; Khan, M.A.; Sharif, M.; Kadry, S.; Manogaran, G.; Saba, T.; Ashraf, I.; Damaševičius, R. A Framework of Human Action Recognition Using Length Control Features Fusion and Weighted Entropy-Variations Based Feature Selection. *Image Vis. Comput.* **2021**, *106*, 104090. [[CrossRef](#)]
38. Muhammad, K.; Mustaqeem; Ullah, A.; Imran, A.S.; Sajjad, M.; Kiran, M.S.; Sannino, G.; de Albuquerque, V.H.C. Human Action Recognition Using Attention Based LSTM Network With Dilated CNN Features. *Future Gener. Comput. Syst.* **2021**, *125*, 820–830. [[CrossRef](#)]
39. Ullah, A.; Muhammad, K.; Ding, W.; Palade, V.; Haq, I.U.; Baik, S.W. Efficient Activity Recognition Using Lightweight CNN and DS-GRU Network for Surveillance Applications. *Appl. Soft Comput.* **2021**, *103*, 107102. [[CrossRef](#)]
40. Nasaoui, H.; Bellamine, I.; Silkan, H. Human Action Recognition Using Squeezed Convolutional Neural Network. In Proceedings of the 2022 11th International Symposium on Signal, Image, Video and Communications (ISIVC), El Jadida, Morocco, 18–20 May 2022; pp. 1–5.

41. Wang, Z.; Lu, H.; Jin, J.; Hu, K. Human Action Recognition Based on Improved Two-Stream Convolution Network. *Appl. Sci.* **2022**, *12*, 5784. [[CrossRef](#)]
42. Vrskova, R.; Hudec, R.; Kamencay, P.; Sykora, P. Human Activity Classification Using the 3DCNN Architecture. *Appl. Sci.* **2022**, *12*, 931. [[CrossRef](#)]
43. Zhang, L.; Lim, C.P.; Yu, Y. Intelligent Human Action Recognition Using an Ensemble Model of Evolving Deep Networks with Swarm-Based Optimization. *Knowl.-Based Syst.* **2021**, *220*, 106918. [[CrossRef](#)]
44. Dasari, P.; Zhang, L.; Yu, Y.; Huang, H.; Gao, R. Human Action Recognition Using Hybrid Deep Evolving Neural Networks. In Proceedings of the 2022 International Joint Conference on Neural Networks (IJCNN), Padua, Italy, 18–23 July 2022; pp. 1–8.
45. Hussain, A.; Hussain, T.; Ullah, W.; Baik, S.W. Vision Transformer and Deep Sequence Learning for Human Activity Recognition in Surveillance Videos. *Comput. Intell. Neurosci.* **2022**, *2022*, 3454167. [[CrossRef](#)] [[PubMed](#)]
46. Wang, X.; Gao, L.; Wang, P.; Sun, X.; Liu, X. Two-Stream 3-D Convnet Fusion for Action Recognition in Videos with Arbitrary Size and Length. *IEEE Trans. Multimed.* **2017**, *20*, 634–644. [[CrossRef](#)]
47. Ullah, A.; Muhammad, K.; Del Ser, J.; Baik, S.W.; de Albuquerque, V.H.C. Activity Recognition Using Temporal Optical Flow Convolutional Features and Multilayer LSTM. *IEEE Trans. Ind. Electron.* **2018**, *66*, 9692–9702. [[CrossRef](#)]
48. Wang, L.; Xiong, Y.; Wang, Z.; Qiao, Y.; Lin, D.; Tang, X.; Van Gool, L. Temporal Segment Networks for Action Recognition in Videos. *IEEE Trans. Pattern Anal. Mach. Intell.* **2018**, *41*, 2740–2755. [[CrossRef](#)] [[PubMed](#)]
49. Yu, S.; Xie, L.; Liu, L.; Xia, D. Learning Long-Term Temporal Features with Deep Neural Networks for Human Action Recognition. *IEEE Access* **2019**, *8*, 1840–1850. [[CrossRef](#)]
50. Ma, C.Y.; Chen, M.H.; Kira, Z.; AlRegib, G. TS-LSTM and Temporal-Inception: Exploiting Spatiotemporal Dynamics for Activity Recognition. *Signal Process. Image Commun.* **2019**, *71*, 76–87. [[CrossRef](#)]
51. Diba, A.; Fayyaz, M.; Sharma, V.; Paluri, M.; Gall, J.; Stiefelwagen, R.; Van Gool, L. Holistic Large Scale Video Understanding. *arXiv* **2019**, arXiv:1904.11451.
52. Majid, M.; Safabakhsh, R. Correlational Convolutional LSTM for Human Action Recognition. *Neurocomputing* **2020**, *396*, 224–229. [[CrossRef](#)]
53. Zhang, Z.; Lv, Z.; Gan, C.; Zhu, Q. Human Action Recognition Using Convolutional LSTM and Fully-Connected LSTM With Different Attentions. *Neurocomputing* **2020**, *410*, 304–316. [[CrossRef](#)]
54. He, J.Y.; Wu, X.; Cheng, Z.Q.; Yuan, Z.; Jiang, Y.G. DB-LSTM: Densely-Connected Bi-Directional LSTM for Human Action Recognition. *Neurocomputing* **2021**, *444*, 319–331. [[CrossRef](#)]
55. Zhu, L.; Fan, H.; Luo, Y.; Xu, M.; Yang, Y. Temporal Cross-Layer Correlation Mining for Action Recognition. *IEEE Trans. Multimed.* **2021**, *24*, 668–676. [[CrossRef](#)]
56. Bao, W.; Yu, Q.; Kong, Y. Evidential Deep Learning for Open Set Action Recognition. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Montreal, QC, Canada, 10–17 October 2021; pp. 13349–13358.
57. Xiao, J.; Jing, L.; Zhang, L.; He, J.; She, Q.; Zhou, Z.; Yuille, A.; Li, Y. Learning from Temporal Gradient for Semi-Supervised Action Recognition. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, New Orleans, LA, USA, 18–24 June 2022; pp. 3252–3262.
58. Zhou, A.; Ma, Y.; Ji, W.; Zong, M.; Yang, P.; Wu, M.; Liu, M. Multi-Head Attention-Based Two-Stream EfficientNet for Action Recognition. *Multimed. Syst.* **2023**, *29*, 487–498. [[CrossRef](#)]
59. Wang, X.; Gao, L.; Song, J.; Shen, H. Beyond Frame-Level CNN: Saliency-Aware 3-D CNN with LSTM for Video Action Recognition. *IEEE Signal Process. Lett.* **2016**, *24*, 510–514. [[CrossRef](#)]
60. Feichtenhofer, C.; Pinz, A.; Wildes, R.P. Spatiotemporal Multiplier Networks for Video Action Recognition. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 4768–4777.
61. Sun, S.; Kuang, Z.; Sheng, L.; Ouyang, W.; Zhang, W. Optical Flow Guided Feature: A Fast and Robust Motion Representation for Video Action Recognition. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–22 June 2018; pp. 1390–1399.
62. Fan, L.; Huang, W.; Gan, C.; Ermon, S.; Gong, B.; Huang, J. End-to-End Learning of Motion Representation for Video Understanding. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–22 June 2018; pp. 6016–6025.
63. Long, X.; Gan, C.; De Melo, G.; Wu, J.; Liu, X.; Wen, S. Attention Clusters: Purely Attention Based Local Feature Integration for Video Classification. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–22 June 2018; pp. 7834–7843.
64. Han, Y.; Zhang, P.; Zhuo, T.; Huang, W.; Zhang, Y. Going Deeper with Two-Stream ConvNets for Action Recognition in Video Surveillance. *Pattern Recognit. Lett.* **2018**, *107*, 83–90. [[CrossRef](#)]
65. Zhou, Y.; Sun, X.; Zha, Z.J.; Zeng, W. Mict: Mixed 3D/2D Convolutional Tube for Human Action Recognition. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–22 June 2018; pp. 449–458.
66. Song, X.; Lan, C.; Zeng, W.; Xing, J.; Sun, X.; Yang, J. Temporal-Spatial Mapping for Action Recognition. *IEEE Trans. Circuits Syst. Video Technol.* **2019**, *30*, 748–759. [[CrossRef](#)]
67. Jiang, B.; Wang, M.; Gan, W.; Wu, W.; Yan, J. STM: Spatiotemporal and Motion Encoding for Action Recognition. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Seoul, Republic of South Korea, 27–28 October 2019; pp. 2000–2009.

68. Mihanpour, A.; Rashti, M.J.; Alavi, S.E. Human Action Recognition in Video Using DB-LSTM and Resnet. In Proceedings of the 2020 6th International Conference on Web Research (ICWR), Tehran, Iran, 22–23 April 2020; pp. 133–138.
69. Arif, S.; Wang, J. Bidirectional LSTM with Saliency-Aware 3D-CNN Features for Human Action Recognition. *J. Eng. Res.* **2021**, *9*. [[CrossRef](#)]
70. Tan, K.S.; Lim, K.M.; Lee, C.P.; Kwek, L.C. Bidirectional Long Short-Term Memory with Temporal Dense Sampling for Human Action Recognition. *Expert Syst. Appl.* **2022**, *210*, 118484. [[CrossRef](#)]
71. Ye, Q.; Tan, Z.; Zhang, Y. Human Action Recognition Method Based on Motion Excitation and Temporal Aggregation Module. *Heliyon* **2022**, *8*, e11401. [[CrossRef](#)]
72. Chen, B.; Meng, F.; Tang, H.; Tong, G. Two-Level Attention Module Based on Spurious-3D Residual Networks for Human Action Recognition. *Sensors* **2023**, *23*, 1707. [[CrossRef](#)] [[PubMed](#)]
73. Brownlee, J. Confidence Intervals for Machine Learning. 2018. Available online: <https://machinelearningmastery.com/confidence-intervals-for-machine-learning/> (accessed on 29 March 2023).
74. Munir, A.; Gordon-Ross, A.; Lysecky, S.; Lysecky, R. A Lightweight Dynamic Optimization Methodology and Application Metrics Estimation Model for Wireless Sensor Networks. *Sustain. Comput. Inform. Syst.* **2013**, *3*, 94–108. [[CrossRef](#)]

Disclaimer/Publisher’s Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.