Efficient Hybrid Algorithm for Human Action Recognition

Mostafa A. Abdelrazik [1,*], Abdelhaliem Zekry [2], and Wael A. Mohamed [1]

[1]Benha Faculty of Engineering, Benha University, Benha, Egypt; Email: wael.ahmed@bhit.bu.edu.eg (W.A.M.)
[2]Ain Shams Faculty of Engineering, Ain Shams University, Cairo, Egypt; Email: aaazekry@hotmail.com (A.Z.)
*Correspondence: mostafa.ahmed15@beng.bu.edu.eg (M.A.A.)

***Abstract*—Recently, researchers have sought to find the ideal way to recognize human actions through video using artificial intelligence due to the multiplicity of applications that rely on it in many fields. In general, the methods have been divided into traditional methods and deep learning methods, which have provided a qualitative leap in the field of computer vision. Convolutional neural network CNN and recurrent neural network RNN are the most popular algorithms used with images and video. The researchers combined the two algorithms to search for the best results in a lot of research. In an attempt to obtain improved results in motion recognition through video, we present in this paper a combined algorithm, which is divided into two main parts, CNN and RNN. In the first part there is a preprocessing stage to make the video frame suitable for the input of both CNN networks which consist of a fusion of Inception-ResNet-V2 and GoogleNet to obtain activations, with the previously trained wights in Inception-ResNet-V2 and GoogleNet and then passed to a deep Gated Recurrent Units (GRU) connected to a fully connected SoftMax layer to recognize and distinguish the human action in the video. The results show that the proposed algorithm gives better accuracy of 97.97% with the UCF101 dataset and 73.12% in the hdmb51 data set compared to those present in the related literature.**

***Keywords*—human action recognition, GRU, RNN, CNN, video classification, activity recognition**

I. Introduction

Throughout the ages, man has been searching for a way to simulate the human mind with its superior capabilities. These efforts have been translated into the science of artificial intelligence, machine learning, and neural networks. Many methods and algorithms have been invented that make the machine work like the human mind and acquire some of its capabilities, such as recognition of (action, speech, face, objects, etc.) [1−10], forecast prediction [11], decision-making [12], etc.

Video-based action recognition remains a difficult challenge because of the complex background, the object's appearance, and the range of behaviour patterns. Video may be considered a collection of image sequences that include temporal and spatial domain information. As a result, the fundamental challenges of action recognition are the variety of behaviour scales in the time domain and the appearance of moving objects in the spatial domain. Because of the addition of the time dimension, the intra-class variation of behaviour samples is greater than in image recognition. Feature extraction from video data is very complex due to varying action durations, particularly in the temporal domain. Spatiotemporal feature extraction is a critical and fundamental phase in visual recognition. Several traditional approaches sought to extract additional features from the local spatiotemporal cube. Recent years have seen an explosion in the usage of Convolutional Neural Networks (CNN) [13] based methods across the board in computer vision applications such image classification, segmentation, and understanding [14−19], face recognition [20, 21], foreground detection [22], target tracking [23, 24], etc.

In this paper, we create an algorithm to improve video human action recognition; we first extract the frames from the video and prepare them to fit the input of both Inception-ResNet-V2 and GoogleNet as Convolutional Neural Networks (CNN). CNNs have been shown to effectively extract spatial characteristics from static images, so we used them to get the activations from each video frame and then pass them into deep Gated Recurrent Units (GRU) as the RNN stage and finally into the full connected SoftMax layer to get the classified action.

This combination of fusion of Inception-ResNet-V2 and GoogleNet as the CNN stage and deep GRU as the RNN stage succeeded in improving the accuracy of the action recognition in videos compared to those present in the related literature.

The remaining of this work is structured as follows. Section II introduces the related work and background. Section III describes in detail the proposed algorithm for action recognition. Section IV describes the results of the experiments. The final section shows our conclusion.

II. Related Work and Background

Many deep network-based systems for action recognition have been presented throughout the last decade by researchers. Traditional data mining involves manually extracting several sets of features from time-series signals and mapping these features to different human activities.

Manuscript received June 22, 2022; revised October 13, 2022; accepted October 19, 2022.

doi: 10.18178/joig.11.1.72-81

For action recognition, these systems take low-level features from video input and pass them to a classifier like a support vector machine (SVM), decision tree, or KNN. Yin *et al*. [25] used a single-class SVM for activity recognition for the first time.

Deep learning starts to penetrate our studies and lives as hardware computing capability improves. CNN and RNN are popular deep learning frameworks. Chen *et al*. [26] suggested a human-based activity recognition model, which is a deep neural network architectural model. Yang [27] and colleagues proposed the deep CNN-based learning method for HAR to automate feature learning from primary inputs systematically. Learning features were viewed as high-level, low-level abstract representations of original time-series signals via deep architecture. Zibin *et al*. [28] introduced a system for learning and researching various important meta-parameters like the number of convolutional layers and kernel size on CNN efficiency. Zeng *et al*. [29] suggested the CNN partial weight sharing strategy, which improved accuracy significantly. For multimodal data, Ha and Choi [30] proposed a sharing of partial weight and full weight sharing mechanism. Ronau and Cho [31] compared the performance of RF and CNN, and the results revealed that CNN had a higher recognition accuracy than RF.

Many attempts to use deep learning approaches for process recognition have been made, spurred on by the enormous success of the CNN model for image recognition. A 3D CNN was developed by Ji *et al*. [32] by convoluting the local space-time of multiple frames. An excellent deep learning model for behavior recognition, this method achieved excellent results on real-world scene datasets. For spatiotemporal features, Tran *et al*. [33] modified the traditional 2D kernels and used 3D CNNs. To capture single-scale spatiotemporal features, the 3D CNN model uses a single-scale image sequence as input. To integrate distinct images into the video and develop a CNN model for visual sequences, to fuse video images, Karpathy *et al*. [34] used a slow fusion model. With this method of fusion, video sequence information can be added to the network effectively, and behavior characteristics can be expressed more effectively. However, the model only receives a single image from a video as input. The optically flowing video has been used as a motion indicator in several algorithms developed in recent years. Using a single RGB image (spatial information) and a stack of optical flow images (temporal information), Simonyan *et al*. [35] developed a two-stream network for action recognition. Two-stream network with new fusion method proposed by Feichtenhofer *et al*. [36] And each stream is still the standard CNN 2D format, as well. A limitation of the optical flow was discovered by [37]. A lighting change can produce optical flow, which is the apparent motion of intensity values without any actual motion. This means that there are certain limitations to the optical flow representation of real motion information The spatial-temporal Laplacian pyramid coding method proposed by Shao *et al*. [38] was designed for video representation. Using CNN, Wang *et al*. [39] proposed a method for expressing trajectory features using a dense trajectory. A multi-scale trajectory pooling 3D convolutional descriptor for action recognition was developed by Lu and colleagues [40]. By stacking motion features, atoms, and phrases, Wang *et al*. [41] have developed a multi-level video representation. “Line pooling”, a method for efficiently pooling stacked features along the timeline, was developed by Zhao and colleagues [42]. To capture long-term temporal information, Varol and colleagues [43] proposed a long-term 3D CNN. Using Weber's law-based Volume Local Gradient Ternary Pattern (WVLGTP) and a new convolutional network, Uddin *et al*. [44] proposed a handcrafted feature descriptor for deep spatial features. Then, combining the handcrafted spatiotemporal feature with a deep spatial feature for action recognition is necessary. In the following subsections we will introduce the utilized deep learning networks:

A. Convolutional Neural Network (CNN)

Convolutional Neural Networks are made up of several layers of artificial neurons. Artificial neurons, like their biological counterparts, are mathematical functions that assess the weighted number of numerous inputs and produce an activation value. The weight of each neuron determines its behavior. When given pixel values, the artificial neurons of a CNN pick out numerous visual characteristics.

When you feed it an image, each ConvNet Layer generates many activation maps. Activation maps highlight the image's most important features. Each neuron takes a pixel patch as input, multiplies the colour values by the weights, adds them all together, and then runs them through the activation algorithm.

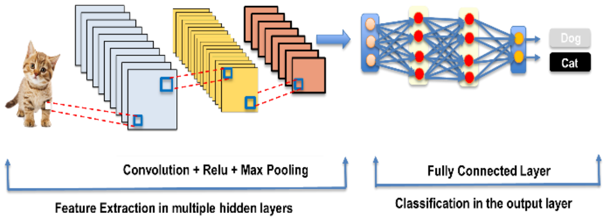
Dog
Cat
Convolution + Relu + Max Pooling
Fully Connected Layer
Feature Extraction in multiple hidden layers
Classification in the output layer

Figure 1. Convolutional neural network steps.

The CNN's first sheet detects basic characteristics such as horizontal, vertical, and diagonal edges. The output of the first layer is passed into the second layer, which eliminates more complicated characteristics such as corners and edge combinations. As you move further into the convolutional neural network, the layers identify higher-level characteristics such as objects, faces, and more. Convolution is the process of multiplying and summing pixel values by weights. A CNN is generally composed of multiple convolution layers, but it may also have additional components. The final layer of a CNN is the classification layer, which receives the output of the final convolution layer as input.

Based on the activation map of the last convolution layer, the classification layer provides a set of confidence scores (numbers ranging from 0 to 1) that indicate how probable the picture is to belong to a “class.” For instance, if a ConvNet identifies cats, and dogs, the final layer's output is the likelihood that the input picture contains any of those animals. Fig. 1 shows a sample of CNN which classify dogs and cats pictures.

B. Recurrent Neural Network (RNN)

RNNs are artificial neural networks that deal with time series or sequential data. These deep learning algorithms are frequently employed for ordinal or temporal issues like as language translation, natural language processing (NLP), speech recognition, and picture captioning, and they may be found in popular apps such as Siri, voice search, and Google Translate. Feedforward and convolutional neural networks (CNNs) are examples of recurrent neural networks that learn from training input. They are distinguished by their "memory", which enables them to impact current input and output by drawing on knowledge from prior inputs. Although traditional deep neural networks believe that inputs and outputs are independent of one another, recurrent neural networks' performance is dependent on the sequence's prior elements. Although future events can be useful in deciding a sequence's performance, unidirectional recurrent neural networks cannot account for them in their predictions.

LSTM: The long short-term memory (LSTM) architecture is a deep learning architecture that is built on an artificial recurrent neural network (RNN). Unlike traditional feedforward neural networks, LSTM contains feedback connections. It is capable of processing not just single data points (such as pictures), but also whole data sequences (such as speech or video). LSTM may be used for tasks such as unsegmented, linked handwriting recognition, speech recognition, and anomaly detection in network traffic or intrusion detection systems, to name a few (intrusion detection systems). A cell, input and output gates, and a forget gate comprise an LSTM unit. These gates regulate the cell's information flow, and the cell stores values for arbitrary time periods.

Since there may be lags of uncertain length between important events in a time series, LSTM networks are well-suited for time series data classification, processing, and prediction. LSTMs were developed to address the issue of vanishing gradients that can occur when conventional RNNs are trained. In many applications, LSTM has an advantage over RNNs, secret Markov models, and other sequence learning methods due to its relative insensitivity to gap length.

GRU: The Gated Recurrent Unit (GRU) is a form of recurrent neural network (RNN) that has advantages over long short-term memory (LSTM) in some situations. GRU is faster and requires less memory than LSTM. GRU will be explained in Section III.C.

III. Research Methodology

The proposed algorithm is divided into two main parts, CNN and RNN, where we rely on CNN to extract the activations from each frame of the video and then pass it to the RNN to classify the human action.

In the first part, after trying several methods, we found that the fusion of GoogleNet and Inception-ResNet-V2 gives the best results in terms of accuracy compared to other methods, so we will explain this method in the following.

This strategy was used to obtain the best results for videos recognition in this paper. The frames from the video are splitted and cropped to the center and then resized to (299×299) and (224×224) to suit the two networks.

Our deep GRU network consists of two steps we configured the output mode as a sequence in the first step to pass the complete sequence to the second one which the output mode configured as last to get the last time step of the sequence.

Fig. 2 shows the general block diagram of the proposed human action recognition algorithm.

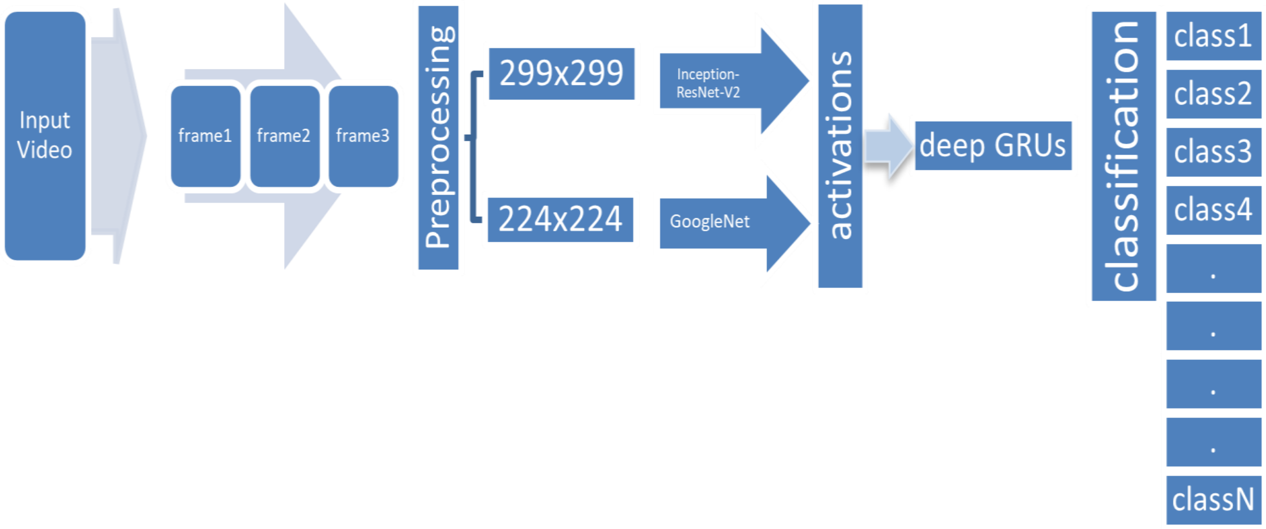

Figure 2. Proposed human action recognition algorithm block diagram.

A. GoogleNet

The Inception Network was a key development in the study of Neural Networks, namely CNNs. Inception Networks is available in three versions: Inception Version 1, Inception Version 2, and Inception Version 3, beside Inception-ResNet. In the ILSVRC, this network was responsible for establishing a new state-of-the-art for classification and detection. GoogleNet is the name given to the first iteration of the Inception network.

Overfitting is a problem that can occur when a network is designed with numerous deep layers. The authors of the research study and provide a solution to this difficulty. Going deeper into convolutions, the GoogleNet architecture was presented, which consists of multiple-size filters that can function on the same level. The network becomes larger rather than deeper as a result of this concept.

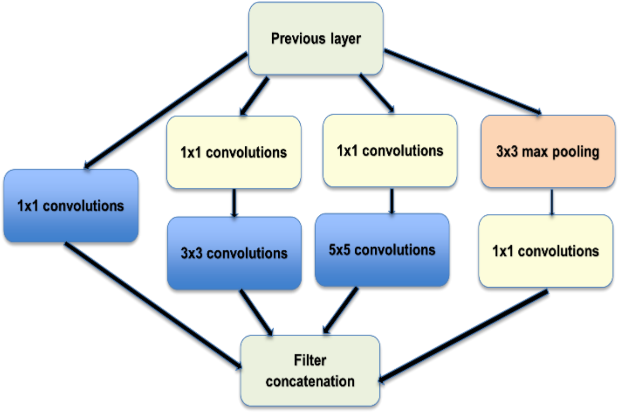

Figure 3. GoogleNet convolution procedure.

The convolution procedure is conducted on inputs with three filter sizes, as shown in Fig. 3 (1×1), (3×3), and (5×5). The convolutions are additionally subjected to a max-pooling procedure before being transferred to the next inception module.

Because training neural networks takes time and resources, the authors reduce the number of input channels by inserting an extra (1×1) convolution before the (3×3) and (5×5) convolutions to decrease the network's dimensions and speed up computations.

The GoogleNet Architecture has a total of 22 layers, including 27 pooling layers. There are a total of 9 inception modules layered linearly. The global average pooling layer is connected to the ends of the inception modules.

- Residual Inception Blocks:

Inception and ResNet have been at the heart of the most significant advances in image recognition performance in recent years, providing great results at a low computational cost. The Inception-ResNet architecture combines Inception with residual connections. For the residual versions of the Inception networks, they used cheaper Inception blocks than the original Inception. Following each Inception block is a filter-expansion layer (1×1 convolution) without activation, which is used to scale up the dimensionality of the filter bank before adding it to match the depth of the input. This is necessary to compensate for the Inception block's reduced dimensionality.

Another minor technical distinction between the residual and non-residual Inception variations is that they employed batch-normalization solely on top of the standard layers in Inception-ResNet, but not on top of the summations. Although it is reasonable to think that extensive batch normalization would be beneficial, they wanted each model replica to be able to be trained on a single GPU. The memory footprint of layers with large activation sizes turned shown to be utilizing an excessive amount of GPU RAM. they were able to significantly increase the overall number of Inception blocks by eliminating the batch normalization on top of those layers Fig. 4 shows the residual inception blocks used in GoogleNet.

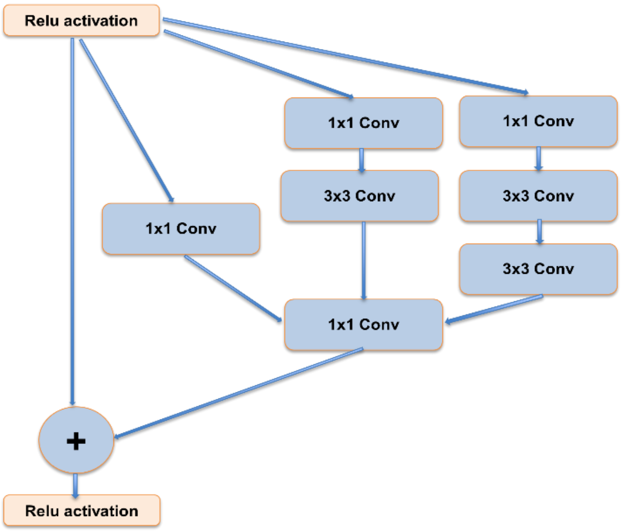

Figure 4. GoogleNet residual inception blocks.

- Scaling of the Residuals:

After more than one thousand filters are used, the instabilities of the residual variants began to appear, and the network simply ‘died’ early in the training, which means the last layer before the average pooling began to produce just zeros after a few tens of thousands of iterations. This could not be avoided, regardless of whether the learning rate was reduced, or an additional batch normalizing layer was added.as shown in Fig. 5.

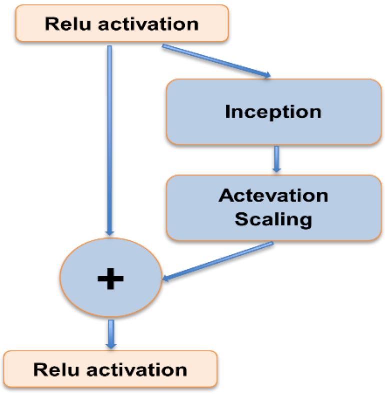

Figure 5. GoogleNet scaling of the residuals.

B. *Inception-ResNet-V2*

Over a million images from the ImageNet collection were used to train the CNN Inception-ResNet-V2. The 164-layer network is capable of classifying images into 1000 different item categories. Consequently, the network has learnt a wide range of rich feature representations for a wide range of pictures. The network is fed a 299×299 picture and returns a list of predicted class probabilities. It is based on the Inception architecture and the Residual connection idea. In the Inception-ResNet block, several sized convolutional filters are combined with residual connections. The introduction of residual connections eliminates the degradation problem caused by deep structures while significantly reducing training time. Fig. 6 shows the architecture of Inception-ResNet-V2.

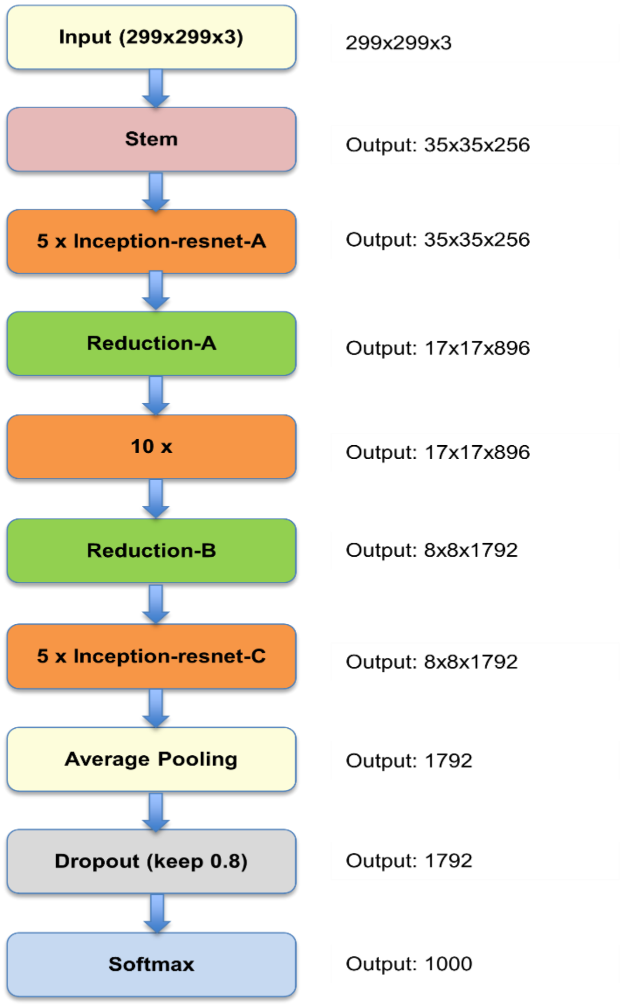

Figure 6. Inception-ResNet architecture.

In the second part of the proposed algorithm, we pass the obtained activations from Inception-ResNet-V2 and GoogleNet to a deep GRU, and finally, there is the softmax layer to identify the action in the sequences.

C. *Gated Recurrent Unit (GRU)*

GRU is faster and requires less memory than LSTM. GRUs also solve the vanishing gradient problem (values used to change network weights), which is a problem with traditional recurrent neural networks. If the grading shrinks as it back propagates over time, it can become too small to impact learning, rendering the neural net untrainable.

RNNs will effectively “forget” longer sequences if a layer in a neural net is unable to remember. The update gate and the reset gate are used by GRUs to solve this problem. These gates control what information is allowed to pass through to the output and can be programmed to remember information for a longer period. This enables it to move specific data down a chain of events to make more accurate predictions.

The update gate works similarly to the LSTM's forget and input gateway. It decided what new information to add and what information to get rid of.

The reset gate is used to determine how much past information should be forgotten. Fig. 7 shows the architecture of GRU.

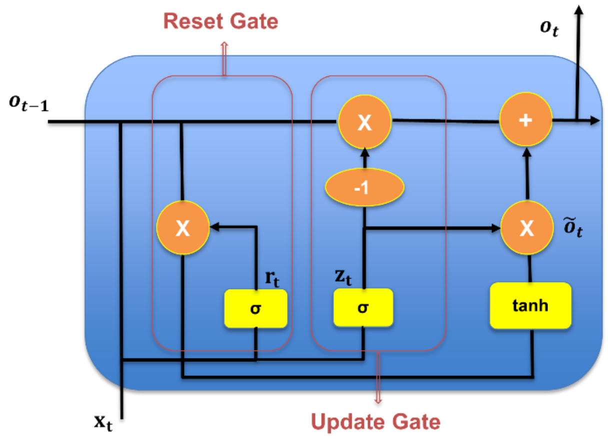

Figure 7. GRU unit.

$$r_t = \sigma(W_{xr}^T \,.\, x_t + W_{or}^T \,.\, o_{t-1} + b_r) \tag{1}$$

$$z_t = \sigma(W_{xz}^T \,.\, x_t + W_{oz}^T \,.\, o_{t-1} + b_z) \tag{2}$$

$$\tilde{o}_t = tanh(W_{x\tilde{o}}^T \,.\, x_t + W_{o\tilde{o}}^T \,.\, (r_t \otimes o_{t-1}) + b_{\tilde{o}}) \tag{3}$$

$$o_t = z_t \otimes o_{t-1} + (1 - z_t) \otimes \tilde{o}_t \tag{4}$$

$$\sigma = \frac{1}{1+ e^{-t}} \tag{5}$$

$$tanh(t) = \frac{1- e^{-2t}}{1+ e^{-2t}} \tag{6}$$

where W_{xr}, W_{xz}, $W_{x\tilde{o}}$ are the weights of the matrices for the corresponding connected input vector, W_{or} , W_{oz} , $W_{o\tilde{o}}$ represent the weight matrices of the previous time step and b_r , b_z , $b_{\tilde{o}}$ are bias

Our deep GRU network consists of two steps; each one has 1000 hidden layers, and the state activation function is tanh, and the gate activation function is sigmoid.

The final step in the algorithm is the SoftMax layer which shows the result of the classification.

IV. Experiment Results

A. *Experimental Setup*

The video datasets UCF-101 and HMDB-51 were used to evaluate the performance of the proposed two-stream network. There are 13320 video clips in the UCF-101, which are divided into 101 categories and 5 action groups (i.e., Human-Object Interaction, Sports, Playing Musical Instrument, Body Motion, and Human-Human Interaction). The UCF-101 has a low noise level and mostly contains motion-related frames. According to the dataset's official publication, there are three types of train/test splits, and the final accuracy is calculated by averaging the results of the three divides. The HMDB-51 has 6766 video clips organized into 51 categories and five action groupings (i.e., face actions, face actions with object manipulation, human body movements, human body movements with object interaction, and human body movements for human interaction). As summarized in Table I.

TABLE I. VIDEO DATA SETS

dataset	No. of clips	No. of categories	No. of groups
UCF-101	13320	101	5
HMDB-51	6766	51	5

In the training procedure, we utilized a learning rate of 0.001 and 30 epochs.

The experiments of this paper have been carried out using the following experimental setup: Intel Core i7 9th Gen processor of 2.60 GHz, a memory of 16 GB, and Nvidia Geforce 1660Ti with 6 GB RAM as the hardware platform and Windows 10 operating system and Matlab 2020a as a software platform.

B. Performance Evaluation and Comparisons

In this section, we illustrate the results from the experiments in quantitative and qualitative means. We first quantitatively compare the results of the proposed algorithm to the results of othet algorithms. All these algorithms are applied to the same data sets. The performance evaluation is conducted in terms of accuracy.

To quantitatively compare our proposed algorithm to the exceptional state-of-the-art algorithms, concerning the accuracy of testing sequences and the confusion matrix. Table II shows the results of applying the proposed algorithm to UCF-101 and HMDB-51 datasets in terms of accuracy. Moreover, it compares these results to the results of applying state-of-the-art algorithms to the same datasets. The results clearly show that the proposed algorithm outperforms the othet algorithms in both datasets.

The next step is to evaluate the performance of the proposed algorithm with the two CNN networks separately. In Table II and Table III, we show the accuracy results of using each method individually and the proposed fusion method which displays the enhancement of the accuracy in both datasets as the accuracy was 97.97% for the UCF-101 and 73.12% for HDMB-51, while the accuracy of the GoogleNet was 92.23% for the UCF-101 and 64.53% for HDMB-51and of the Inception-ResNet-V2 was 95.51% for the UCF-101 and 67.09% for HDMB-51.

TABLE II. APPLIED ALGORITHMS RESULTS

network	dataset	accuracy	Time(min)
GoogleNet	UCF-101	92.23	115
Inception		95.51	135
proposed		97.97	184
GoogleNet	HMDB-51	64.53	40
Inception		67.09	45
proposed		73.12	60

TABLE III. ALGORITHMS ACCURACY

Method	UCF-101 Accuracy	HMDB-51 Accuracy
C3D(3net) (2015)	85.2%	-
Two-stream CNNs (2014)	88.0%	59.4%
EMV+RGB-CNN (2016)	86.4%	-
RLSTM-g3 (2016)	86.9%	55.3%
Multiple dynamic images (2016)	89.1%	65.2%
Factorized spatio-temporal CNNs (2015)	88.1%	59.1%
Temporal pyramid CNNs (2015)	89.1%	63.1%
DTMV+RGB-CNN (2018)	87.5%	72.5%
LSF-CNN (2020)	94.8%	70.2%
proposed method	97.97%	73.12%

Fig. 8 and Fig. 9 show the accuracy and losses of the validation videos of the UCF-101 dataset during each of the thirty epochs of the training process for the three networks. These results show that the proposed method is more accurate and has fewer losses than the Inception-ResNet-V2 and GoogleNet, while Fig. 10 and Fig. 11 show the accuracy and losses of the HMDB-51 dataset with the same parameters. The accuracy of the HMDB-51dataset is always less than the accuracy of the UCF-101 dataset because of the challenges in the HMDB-51 dataset (camera motion, the small size of the object, and illumination issues, …) which make it difficult to classify the actions in the videos, but the proposed method is also more accurate than the two individual networks.

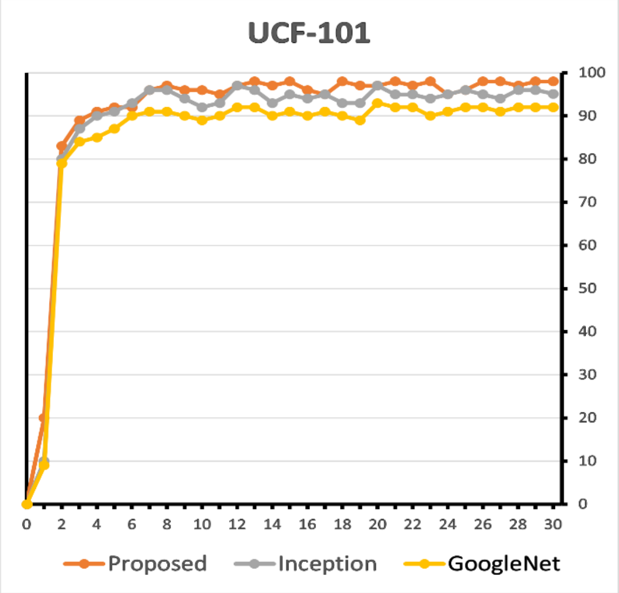

Figure 8. Applied algorithms accuracy for UCF-101 data set.

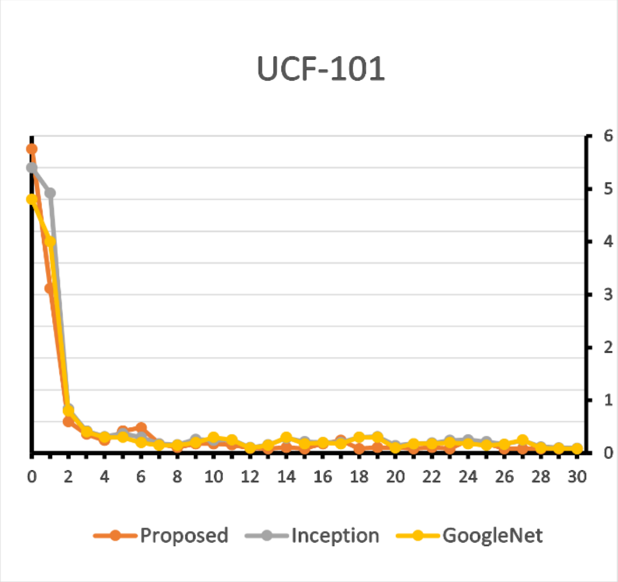

Figure 9. Applied algorithms losses for UCF-101 data set.

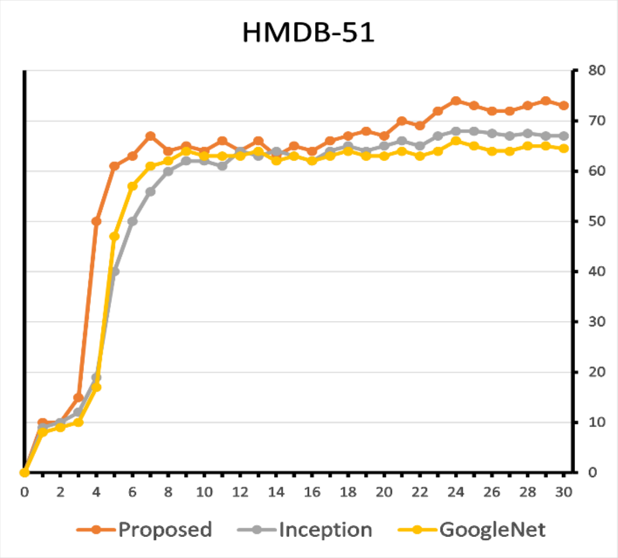

Figure 10. Applied algorithms accuracy for HMDB-51 data set.

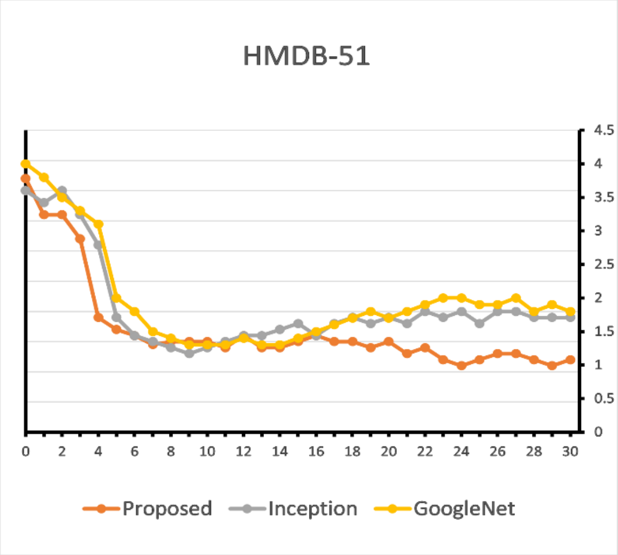

Figure 11. Applied algorithms losses for HMDB-51 data set.

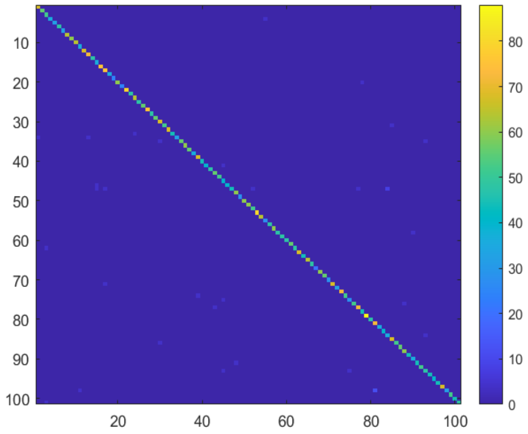

Figure 12. Fusion matrix for proposed algorithm with UCF-101 dataset.

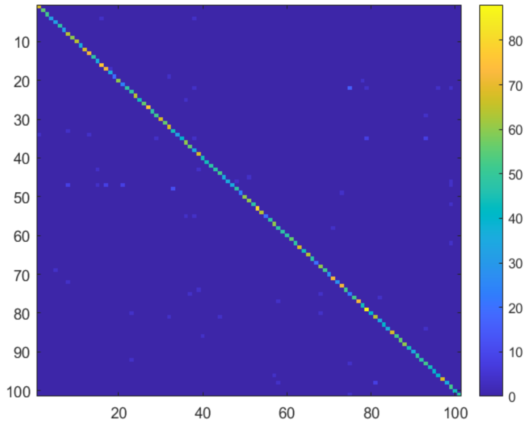

Figure 13. Fusion matrix for Inception with UCF-101 dataset.

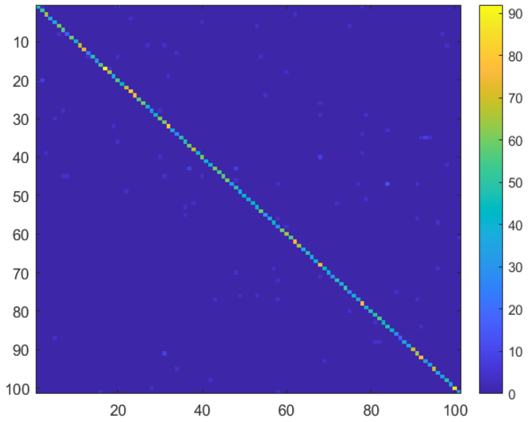

Figure 14. Fusion matrix for GoogleNet with UCF-101 dataset.

We utilized the UCF-101 dataset's whole 101 activities and reported their performance using the confusion matrices given in (Figs. 12−14). The confusion matrices clearly show that most of the activities in Fig. 12 are correctly classified when compared to Fig. 13 and Fig. 14. Furthermore, we can see that our method outperforms the individual networks. Similarly, we utilized all 51 actions in the HMDB-51 and reported the performance with the confusion matrices in (Figs. 15−17), and the result outperforms the proposed technique as in the UCF-101 dataset. In a few cases, combined Inception and GoogleNet provide the same results; otherwise, our proposed approach produces greater overall performance.

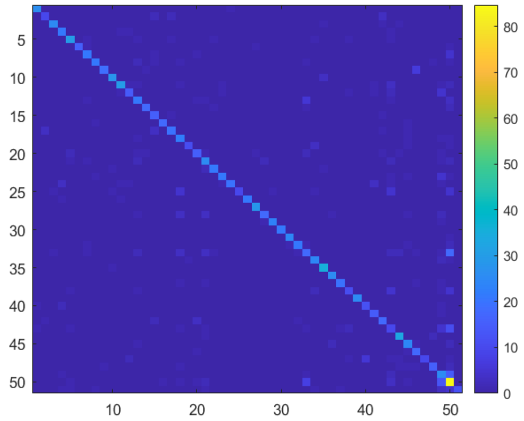

Figure 15. Fusion matrix for proposed algorithm with HMDB-51 dataset.

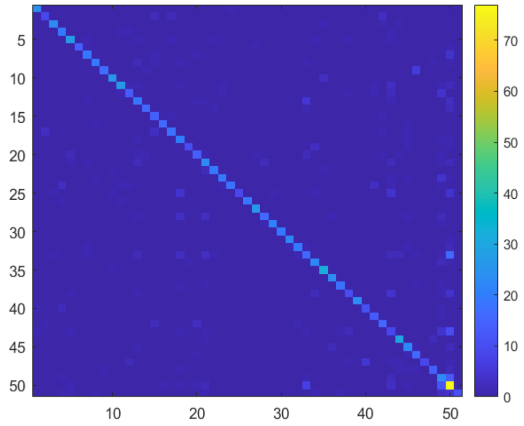

Figure 16. Fusion matrix for Inception with HMDB-51 dataset.

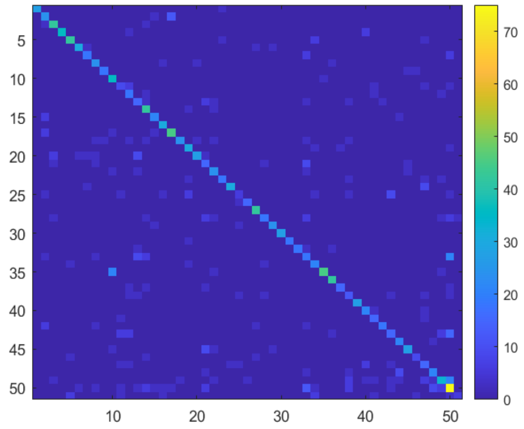

Figure 17. Fusion matrix GoogleNet with HMDB-51 dataset.

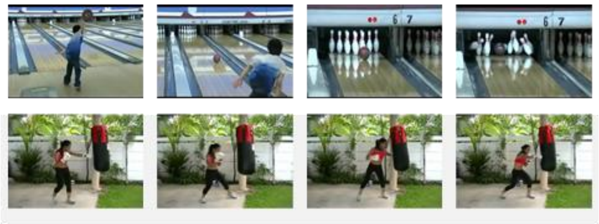

Figure 18. Samples of correct classified actions.

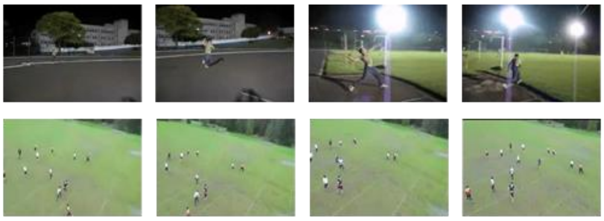

Figure 19. Samples of missed classified actions.

In Fig. 18, we show samples of the best-recognized categories and its confusion matrix like Bowling at the first row with 39 correct recognitions, and Boxing Punching Bag at the second row with 25 correct recognitions. On the other hand, Fig. 19 shows samples of missed classified actions like Javelin throw at the first row, and Frisbee catch at the second row, and this is because of many challenges like the illumination and very small size of objects.

V. Conclusion

This article proposes a new method for human action recognition based on CNN and Deep GRU using only raw video frames. To begin, deep features of video frames are extracted using the Inception-ResNet-V2 and GoogleNet pre-trained Convolutional Neural Networks architecture, which speeds up the learning process and improves performance. The sequence information of the frames is then learned using the DB-GRU recurrent network in both forward and backward transitions, and the final classification is completed.

In comparison to state-of-the-art approaches, simulated results using Matlab show that the proposed method performs exceptionally well in HAR from video on the UCF101 and HMDB-51 datasets. This investigation's accuracy was improved through proper input parameter adjustments and the fusion of Inception-ResNet-V2 and GoogleNet with deep GRU. In future work, this method could be applied in human action recognition applications (power consumption reduction in smart homes depending on the mode of human actions, the control of the drones by hand signs, and sign language recognition programs).

Conflict of Interest

The authors declare no conflict of interest

Author Contributions

Mostafa A. Abdelrazik and Wael A. Mohamed conducted the research and analyzed the data. Mostafa A. Abdelrazik and Abdelhaliem Zekry wrote the paper; all authors had approved the final version.

REFERENCES

[1] A. Sánchez-Caballero, S. D. López-Diz, D. Fuentes-Jimenez, *et al*. "3DFCNN: Real-time action recognition using 3D deep neural networks with raw depth information," *Multimed. Tools Appl.*, vol. 81, pp. 24119–24143, 2022.

[2] P. Lakkhanawannakun and C. Noyunsan, "Speech recognition using deep learning," in *Proc. 34th International Technical Conference on Circuits/Systems, Computers and Communications (ITC-CSCC)*, 2019, pp. 1–4.

[3] K. Teoh, R. Ismail, S. Naziri, R Hussin, M. Isa, and M. Basir, "Face recognition and identification using deep learning approach," *Journal of Physics: Conference Series*, vol. 1755, no. 1, Feb. 2021.

[4] R. Goel, A. Sharma, and R. Kapoor, "Object recognition using deep learning," *Journal of Computational and Theoretical Nanoscience*, vol. 16, no. 9, pp. 4044–4052, 2019.

[5] G. Cheron, I. Laptev, and C. Schmid, "P-CNN: Pose-based CNN features for action recognition," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, 2015.

[6] T. Brox, A. Bruhn, N. Papenberg, and J. Weickert, "High accuracy optical flow estimation based on a theory for warping," in *Proc. ECCV*, 2004.

[7] Y. Wan, Z. Yu, Y. Wang, and X. Li," Action recognition based on two-stream convolutional networks with long-short-term spatiotemporal features," *IEEE Acsess*, vol. 8, pp. 85284–85293, 2020.

[8] I. Jegham, A. B. Khalifa, I. Alouani, and M. A. Mahjoub, "Vision-based human action recognition: An overview and real world challenges," *Forensic Science International: Digital Investigation*, vol. 32, 2020,

[9] N. Dua, S. Singh, and V. Semwal, "Multi-input CNN-GRU based human activity recognition using wearable sensors," *Computing*, vol. 103. pp. 1–18, 2021.

[10] V. Sharma, M. Gupta, A. K. Pandey, D. Mishra, and A. Kumar, "A review of deep learning-based human activity recognition on benchmark video datasets," *Applied Artificial Intelligence*, vol. 36, no. 1, 2022.

[11] X. Ren, Xiaoyong Li, K. Ren, J. Song, Z. Xu, K. Deng, and X. Wang, "Deep learning-based weather prediction: A survey," *Big Data Research*, vol. 23, 2021.

[12] Y. R. Shrestha, V. Krishna, and G. V. Krogh, "Augmenting organizational decision-making with deep learning algorithms: Principles, promises, and challenges," *Journal of Business Research*, vol. 123, pp. 588-603, 2021.

[13] Y. Lecun, L. Bottou, Y. Bengio, and P. Haffner, "Gradient-based learning applied to document recognition," *Proc. IEEE*, vol. 86, no. 11, pp. 2278–2324, Nov. 1998.

[14] J. Wang, Y. Yang, J. Mao, Z. Huang, C. Huang, and W. Xu, "CNN-RNN: A unified framework for multi-label image classification," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognition*, Jun. 2016, pp. 2285–2294.

[15] H. Lee and H. Kwon, "Going deeper with contextual CNN for hyperspectral image classification," *IEEE Trans. Image Process.*, vol. 26, no. 10, pp. 4843–4855, Oct. 2017.

[16] L. C. Chen, G. Papandreou, I. Kokkinos, K. Murphy, and A. L. Yuille, "Semantic image segmentation with deep convolutional nets and fully connected CRFs," *Comput. Sci.*, vol. 4, pp. 357–361, 2014.

[17] M. Oquab, L. Bottou, I. Laptev, and J. Sivic, "Learning and transferring mid-level image representations using convolutional neural networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2014.

[18] E. S. A. El-Dahshan, H. M. Mohsen, K. Revett, and A. B. M. Salem, "Computer-aided diagnosis of human brain tumor through MRI: A survey and a new algorithm," *Expert Syst. Appl.*, vol. 41, no. 11, pp. 5526–5545, 2014.

[19] L. Chang, X. M. Deng, M. Q. Zhou, Z. K. Wu, Y. Yuan, and S. Yang, "Convolutional neural networks in image understanding," *Acta Autom. Sinica*, vol. 42, no. 9, pp. 1300–1312, 2016.

[20] R. He, X. Wu, Z. Sun, and T. Tan, "Wasserstein CNN: Learning invariant features for NIR-VIS face recognition," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 41, no. 7, pp. 1761–1773, Jul. 2019.

[21] S. Banerjee and S. Das, "Mutual variation of information on transfer CNN for face recognition with degraded probe samples," *Neurocomputing*, vol. 310, pp. 299–315, Oct. 2018.

[22] Y. Wang, Z. Yu, and L. Zhu, "Foreground detection with deeply learned multi-scale spatial-temporal features," *Sensors*, vol. 18, no. 12, p. 4269

[23] N. Mahmoudi, S. M. Ahadi, and M. Rahmati, "Multi-target tracking using CNN-based features: CNNMTT," *Multimedia Tools Appl.*, vol. 78, no. 6, pp. 7077–7096, Mar. 2019.

[24] L. Leal-Taixé, C. Canton-Ferrer, and K. Schindler, "Learning by tracking: Siamese CNN for robust target association," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. Workshops (CVPRW)*, Jun. 2016, pp. 33–40.

[25] J. Yin, Q. Yang, and J. J. Pan, "Sensor-based abnormal human-activity detection," *IEEE Trans. Knowl. Data Eng.*, vol. 20, pp. 1082–1090, Sep. 2007.

[26] Y. Chen and Y. Xue, "A deep learning approach to human activity recognition based on single accelerometer," in *Proc. IEEE International Conference on SMC*, 2015, pp. 1488–1492.

[27] J. B. Yang, M. N. Nguyen, P. P. San, X. X. Li, and S. Krishnaswamy, "Deep convolutional neural networks on multichannel time series for human activity recognition," in *Proc. IJCAI*, 2015, pp. 3995–4000.

[28] T. Zebin, P. J. Scully, and K. B. Ozanyan, "Human activity recognition with inertial sensors using a deep learning approach," in *Proc. IEEE Sensors*, pp. 1–3, 2016.

[29] M. Zeng, L. T. Nguyen, B. Yu, O. J. Mengshoel, J. Zhu, P. Wu, and J. Zhang, "Convolutional neural networks for human activity recognition using mobile sensors," in *Proc. MobiCASE*, pp. 197–205, 2014.

[30] S. Ha and S. Choi, "Convolutional neural networks for human activity recognition using multiple accelerometer and gyroscope sensors," in *Proc. IJCNN*, pp. 381–388, 2016.

[31] C. A. Ronao and S. B. Cho, "Human activity recognition with smartphone sensors using deep learning neural networks," *Expert Syst. Appl.*, vol. 59, pp. 235–244, 2016.

[32] S. Ji, W. Xu, M. Yang, and K. Yu, "3D convolutional neural networks for human action recognition," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 35, no. 1, pp. 221–231, 2013.

[33] D. Tran, L. Bourdev, R. Fergus, L. Torresani, and M. Paluri, "Learning spatiotemporal features with 3D convolutional networks," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, 2015.

[34] A. Karpathy, G. Toderici, S. Shetty, T. Leung, R. Sukthankar, and L. Fei-Fei, "Large-scale video classi_cation with convolutional neural networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2014.

[35] K. Simonyan and A. Zisserman, "two-stream convolutional networks for action recognition in videos," in *Proc. Int. Conf. Neural Inf. Process. Syst.*, 2014.

[36] C. Feichtenhofer, A. Pinz, and A. Zisserman, "Convolutional two-stream network fusion for video action recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, 2016.

[37] L. Sevilla-Lara, "On the integration of optical _ow and action recognition," in *Proc. German Conf. Pattern Recognit.*, 2017.

[38] L. Shao, X. Zhen, D. Tao, and X. Li, "Spatio-temporal Laplacian pyramid coding for action recognition," *IEEE Trans. Cybern.*, vol. 44, no. 6, pp. 817–827, 2014.

[39] L. Wang, Y. Qiao, and X. Tang, "Action recognition with trajectory-pooled deep-convolutional descriptors," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, 2015.

[40] X. Lu, H. Yao, S. Zhao, X. Sun, and S. Zhang, "Action recognition with multi-scale trajectory-pooled 3D convolutional descriptors," *Multimedia Tools Appl.*, vol. 78, no. 1, pp. 507–523, 2019.

[41] L. Wang, Y. Qiao, and X. Tang, "MoFAP: A multi-level representation for action recognition," *Int. J. Comput. Vis.*, vol. 119, no. 3, pp. 254–271, 2016.

[42] S. Zhao, Y. Liu, Y. Han, R. Hong, Q. Hu, and Q. Tian, "Pooling the convolutional layers in deep ConvNets for video action recognition," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 28, no. 8, pp. 1839–1849, 2018.

[43] 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, 2018.

[44] M. Uddin and Y. K. Lee, "Feature fusion of deep spatial features and handcrafted spatiotemporal features for human action recognition," *Sensors*, vol. 19, no. 7, p. 1599, 2019.

Mostafa A. Abdelrazik was born in Eldakahliya, Egypt, in 1980. He received the B.S. degree in electrical engineering from Alexandria University, Alexandria, Egypt, in 2002 and the M.S. degree in electrical engineering from Benha University, Benha, Egypt, in 2015. He is currently pursuing the Ph.D. degree in electrical engineering at Benha University, Benha, Egypt.

Abdelhaliem Zekry graduated from Cairo University Egypt in 1969. He was offered the MSc degree in 1973 from the same university. He worked as a scientific coworker at TU Berlin, where he got his Ph.D. in 1981. He worked as an assistant prof. at Ain Shams University (ASU), Egypt 1982. He moved to King Soud University in 1988 and stayed there for 6 years, where he became a professor of Electronics. Now he is a professor of electronics at the faculty of Egypt, ASU. Dr. Zekry made intensive research on semiconductor materials, devices, and circuits. He published more than 70 papers in specialized conferences and periodicals in addition to two books in Electronics. Now, he is driving research on electronics for communication especially the implementation of advanced communication standards using DSP platforms., Dr. Zekry has been awarded several prizes for outstanding research as well as the Decoration of distinction from the Egyptian President. He has been an IEEE member since 1991.

Wael A. Mohamed received his B.Sc. and M.Sc. degrees in electric engineering from Benha Faculty of Engineering, Benha, Egypt, in 1999 and 2005, respectively, and his Ph.D. in 2010 from Cairo Faculty of Engineering - Biomedical Branch. He won the prize for the best research from the Association of Egyptian Scientists in Canada and America in 2009. His experience and research interests in medical imaging, signal and image processing, deep learning, fusion algorithms, and bioinstrumentation. He is currently an Associate professor at Benha Faculty of Engineering from May 2020. He is the coordinator of the Electromechanical Engineering Program. He is a role player in Benha Faculty of Engineering Consultation Unit.