Journal of Theoretical and Applied Information Technology
15th March 2023. Vol.101. No 5
© 2023 Little Lion Scientific

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

A COMPARATIVE STUDY USING IMPROVED LSTM /GRU FOR HUMAN ACTION RECOGNITION

AZHEE WRIA MUHAMAD[1], AREE ALI MOHAMMED[2]

[1] University of Suleimani, College of Basic Education, Department of Computer Science, Iraq
[2]University of Suleimani, College of Science, Department of Computer Science, Iraq
E-mail: [1]azhee.muhamad@univsul.edu.iq, [2]aree.ali@univsul.edu.iq

ABSTRACT

A key focus of this paper is the automatic recognition of human actions in videos. Human action recognition is the automatic understanding of what actions occur in a video implement by a human. Deep learning models can be used to identify and classify objects accurately. The process of detecting human actions in videos is known as human action recognition. One of the deep learning algorithms for sequence data analysis is a recurrent neural network (RNN). In a conventional neural network, the inputs and the outputs are independent of each other. At the same time, RNN is considered a type of Neural Network where the output from the previous step feeds information to the current phase. It has many applications, including video sentiment classification, speech tagging, and machine translation. Recurrent networks are also distributed parameters across each layer of the network. Several layers are stacked together to increase depth in forwarding and backward information of long short-term memory (LSTM) and Gated Recurrent Unit (GRU). This paper proposes two models for various action recognitions using LSTM and GRU, respectively. The first model was improved by increasing the LSTM layers to four and the number of units in each layer to 128 cells. While in the second model, GRU layers were extended to two layers with 128 cells, and the (update and reset) gates are modified based on the previous and current input. A comparative study was conducted during the experimental tests performed on the UCF101 action dataset regarding the accuracy rate for both models. Test results indicate that the accuracy has a significant improvement compared with other state-of-the-arts action recognitions, which are 95.19 % and 92.9 % for both improved LSTM and GRU, respectively.

Keywords: *Action Recognition, Deep learning, LSTM/GRU, Performance Accuracy, and RNN.*

1. INTRODUCTION

Human actions recognition (HAR) in the real-world environment has become increasingly popular in recent years, with applications across a wide range of fields. Developed an intelligent human-computer interaction application that detects human action recognition. The purpose of the HAR is to automatically understand what kind of action is performed in the video. Action recognition in a video file is a challenging problem for computer vision due to the similarity of visual video sequences. A modification in capturing the human action with the action performer, scale, and various lighting conditions in action recognition, detection, and segmentation are components of understanding actions. There are several tasks for interpreting video activities, such as human action recognition, which takes video as input and produces a label for each human action as shown in Figure 1. Most human action recognition methods have difficulty identifying actions in long videos with several scenes and actions. Sequential action localization can split a long video into many short clips. Based on the start/end time of the act, researchers recognize the action and help models of long videos understand them better [1].

The vastness of human action recognition research is one of the key factors that attract researchers working on the range of applications in observation videos [2], robotics, human-computer interaction, sports analysis, and management of web videos [3]. Human motions range from a simple arm or leg movement to complex body activities like legs and arms. For example, kicking a ball requires only a basic leg motion, but jumping to shoot someone in the head needs the action of the legs, arms, head, and the entire body [4].

The models have the building blocks of a linked neuron, consisting of input, internal (or hidden) and output units, each of which is turned on at a time [5]. The LSTM architecture consists of a memory cell, input gate, output gate, and forget gates that maintain their state over time [6], but GRU consists of reset, update and input gates [7].

Researchers have presented a multilayer RNN for processing sequential data and proposed accurate algorithms to identify complex patterns in the visual data, which has not been easy with simple RNN. Alom et al., proposed a method that analyzes video frame features to recognize actions. AlexNet extracts deep features from every sixth video frame [8].

Some previous models have more hyperparameters and limited accuracy in recognition of human action [40] if that is compared to the proposed models. Therefore, several preprocessing algorithms are used in the proposed models to fill this gap, and the models must build methods using a fast and straightforward deep RNN architecture. The proposed models have high accuracy, few RNN layers, and fast convergence speed compared to previous models.

This paper identifies the human behavior in a way that is similar to how we see behaviors in the real world. LSTM and GRU deep learning models are used to consider the information of previous frames when automatically understanding actions in videos. The following are contributions of this research:

1) We propose four layers LSTM and two layers GRU architecture and recurrent unit models to learn sequence information from video frame features in forwarding and backward information.
2) The proposed method can recognize actions in video streams because the video process takes time since video data change very little from frame to frame.
3) The proposed LSTM and GRU models have a high capacity for learning sequences and changing features from frame to frame.
4) Applying some preprocessing techniques to improve the performance of the RNN architecture model for feature extraction and others for normalization of the features before feeding them into the model. Due to these properties, the proposed method is more appropriate for recognizing video actions.

The rest of the paper has been organized into the following sections: Section 2 provides an overview of the related works. Section 3 explains the proposed framework. While in Section 4, experimental results are presented, evaluated the method, and compared it with other state-of-the-art methods. The paper concludes with a discussion of future research directions in section 5.

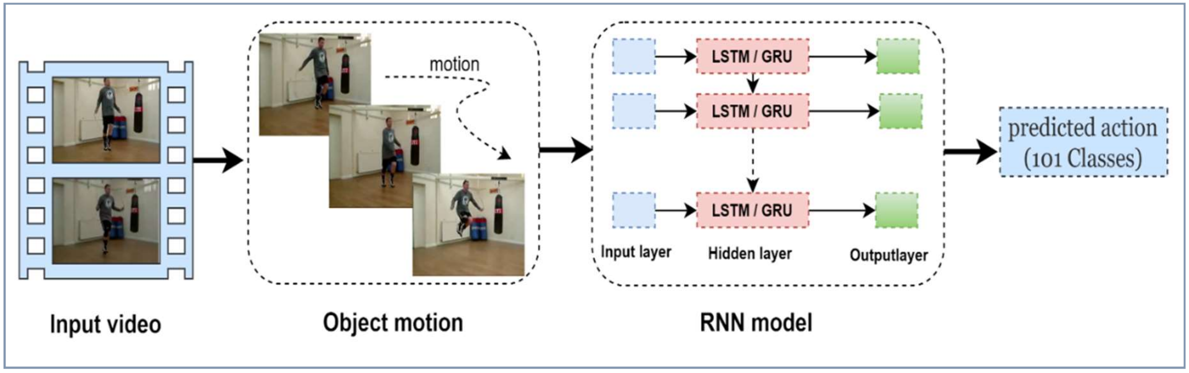


Figure 1. Model for human action prediction overview

2. RELATEDWORKS

Human action recognition has been worked upon and studied well in the current years. In starting years, the methods used to find out interest points of action have mainly been done manually by looking at various feature points for multiple activities. But in the last years, a machine has made efforts to detect those feature points rather automatically. Before moving further, there are also some design issues with systems, such as the selection of different types of neural network models, data collection-related rules, recognition performance in efficiency, preprocessing capacity on the video stream, and flexibility [9].

Several handcrafted and deep-net-based approaches to action recognition have developed over the last decade. A handcrafted feature has been used for non-realistic actions, where an actor would perform some

Yinghua and Zhang [11] analyzed the geometrical properties of space-time volume (STV) called action sketches. The direction, speed, and shape of STV were captured and stacked in time to recognize actions. Angelini et al. [12] presented a three-dimensional representation of the human action using silhouettes from STVs. Analysis of 2D shapes of actions was performed with the Poisson equation method, and Space-time Features (STFs) containing local space-time saliency, action dynamics, shape structure, and orientation. In these two cases, two different actions resulted in the same 2D shapes in STV because of their non-realistic dataset, making it difficult to represent different actions [13].

Deep learning has improved many areas, including image classification, person recognition, object detection, identification, speech recognition, and bioinformatics [14]. The recurrent neural network is an artificial neural network that can process sequential data well. This difference is mainly due to the network architecture, which connects units through directed cycles. The problem of vanishing or exploding gradients occurs. In practical situations or in Real Time, conventional recurrent neural networks are incapable of handling long-term dependencies in Recurrent Teaching and Learning (RTRL) [15]. Dense Trajectories, which consist of Histograms of Oriented (HOG), Histogram of Optical Flow (HOF), and Motion Boundary Histograms (MBH), have recently been identified as a successful method. Improved Dense Trajectories (IDT) features [16] considered camera motion to improve action detection using a D.T.

To enhance accuracy, a method in [17] is proposed to analyze video using a convolutional neural network (CNN) with a deep bidirectional LSTM (DB-LSTM) network. Dengshan et al. used a simple LSTM structure for video detection on UCF101dataset, an LSTM structure with context performed with competitive accuracy. LSTM structure is called Context-LSTM due to its potential for deep processing based on the data, and the context-LSTM reduced the training time [18]. The authors use 3D ConvNet to perform spatiotemporal learning from RGB and optical flow clips and apply Gated Recurrent Units (GRUs) to the spatial-temporal information to recognize actions [19]. The LSTM and GRU models are proposed for actions in a simple environment. Vector Support Machines (SVMs), decision trees, and KNNs were used to recognize actions from video data using low-level features extracted from video frames [10] recognizing human activity with high accuracy. A few parameters in the models helped extract activity features and classify them automatically. The models were enhanced to prevent gradient vanishing and gradient explosion issues when training. Additionally, the above models have many hyperparameters and layers, with an increasing number of memory units.

HAR's design methodology and data collection processes have been studied differently over the past decade. This paper aims to develop methods for recognizing human actions through deep learning and handcrafted techniques. Reviewing deep learning and handcrafted methods to determine the most efficient model for Human action recognition. And finding gaps in state-of-the-art techniques and identifying where contributions are needed. Additionally, Improved accuracy of models and efficiency on standard benchmark datasets over the existing models.

Therefore, to fill this gap, several preprocessing algorithms are used in the proposed models, and the models must build methods using a fast and straightforward deep RNN architecture. The above models generally detect human activities, and structure models. The proposed models have high accuracy and fast convergence speed compared to previous models. Also, testing of the models was done on the most widely available UCF101 dataset.

3. PROPOSED MODEL

RNN is a type of artificial neural network with directed cycles of connections between units. The presented RNN was improved based on the modified LSTM network and GRU models to enhance HAR accuracy in order to recognize human actions automatically on the UCF101 dataset. To increase the quality of dataset using some preprocessing algorithms are used including color-to-grayscale conversions, histogram equalizations, filters, and normalizations. Increasing training data was used to avoid overfitting, then using the data augmentation as shown in Figure 2. This study focused only on three types of RNNs: Simple Recurrent Networks (SRN), LSTMs, and GRUs. These two networks were compared with previous networks using LSTM and GRU models.

3.1 The Improved LSTM

Since the simple RNN system cannot predict well, the LSTM architecture is used in the RNN system instead simple RNN [21].

As a recurrent network unit, LSTMs are excellent at retaining values for long or short periods. In LSTM blocks, three or four gates allow information to enter and exit their memory. Using the logistic function, gates use this method to compute values between 0 and 1.

Information can be allowed to enter or leave the memory in stages by multiplying this value. An "input gate" controls the amount of data flowing into memory. "Forget gates" control how much memory value retains. In addition, an "output gate" determines how much memory value is used in the blocks output [22]. The improved RNN model is based on the modified LSTM architecture.
The traditional LSTM has only one hidden layer, while in the proposed model, LSTM has four layers. Figure3 shows an improved LSTM model with a single cell of the LSTM memory block.

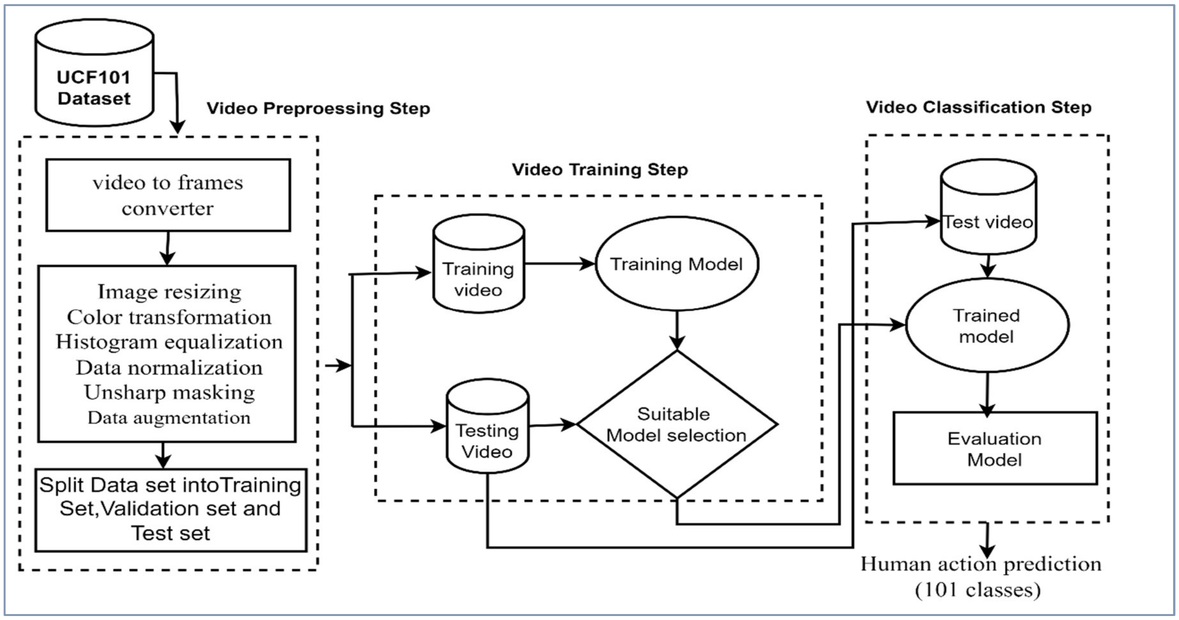


Figure 2: A block diagram of the proposed model.

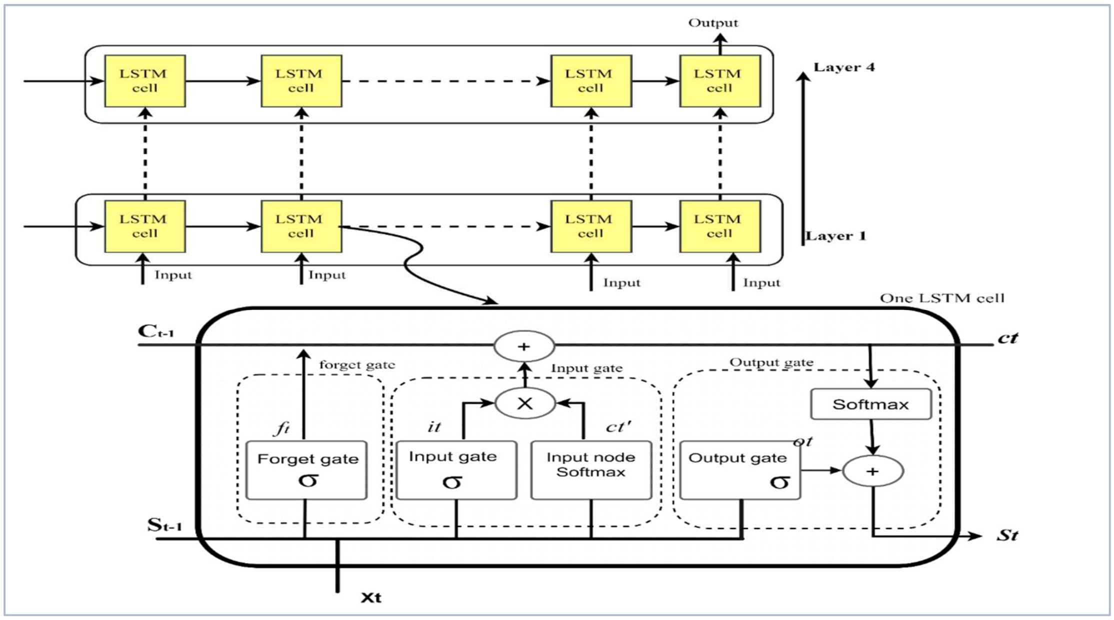


Figure 3. A block diagram of the proposed LSTM model

The forget gate is represented as ft, and the first sigmoid activation function in the network is the forget gate. This gate shall decide which information has to be retained or dropped. The sigmoid function processes the information from the previous hidden state and the current input, and the output is obtained. It is between 0 and 1, thus the closer value to 0 is forgotten, the closer values to one are remembered. The equation is mathematically represented by:

$$ft = \sigma[(W_f * S_{t-1}) + (W_f \ * X_t) + b_f] \quad (1)$$

Where σ = Sigmoid function, W_f= Weights for forget gate, S_{t-1}=old state, X_t=Input, and b_f=forget bias.

The input gate is represented as it. The second sigmoid functions with SoftMax activation function to determines which information is saved to the cell state and which has been deleted. The input gate is represented mathematically as follows:

$$it = \sigma[(W_i * S_{t-1}) + (W_i * X_t) + b_i] \quad (2)$$

Where, σ = Sigmoid function, W_i= Weights for Input Gate,S_{t-1}=old state,X_t=input and b_i= input bias.

The output gate is represented as ot, which highlights which information should go to the next hidden state. The output gate is mathematically represented as follows:

$$ot = \sigma[(W_o * S_{t-1}) + (W_o * X_t) + b_o] \quad (3)$$

Where σ is the Sigmoid function,W_o= Weights for output gate, S_{t-1}= old state, X_t= input, and b_o= output bias.

ct' represents the intermediated cell state and can be calculated using the SoftMax activation and weights for the intermediate cell state (W_c) and multiplied by S_{t-1} plus the weight for the intermediate cell state (W_c) multiplied by input (X_t).

$$ct' = SoftMax\ [(W_c * S_{t-1}) + (W_c * X_t) + b_c] (4)$$

Cell state is calculated by multiplying the input gate by the intermediate cell state plus the forget gate by the privileged cell state.

$$ct = (it * ct') + (ft * c_{t-1}) \quad (5)$$

The new state is calculated through the output gate into the SoftMax activation function for the cell state, so these are all sequential. The following is the formula for calculating the activation of each LSTM unit:

$$S_t = SoftMax\ (ct) + ot \quad (6)$$

The improved LSTM network model over the classical model includes the following:

1). The number of LSTM layers increased. The LSTM algorithm is capable of capturing temporal information from sequential data, it has the problem of the gradient vanishing. In addition to the proposed LSTM hidden layers, the model would be deeper, thus more accurately earning the description of deep learning. The model is the depth of neural networks that is commonly attributed to the success of human action in the video to prediction problems. The proposed LSTM has a significant advantage over simple LSTM in extracting features from sequence data because of its special memory cells. In this model, the input data pass through four layers of LSTM to extract the temporal features from the sequence data, and each layer of LSTM has 128 memory cells.

2). Increasing the number of units in each layer of the LSTM model. In reality, the number of units describes the hidden state (or output). The model of this study has a hidden state of 128 cells. The number of units represents how many neurons are connected to the layer holding the concatenated vector of hidden states and input. There have been several factors that can determine how many layers and cells are presented in modified LSTM:

A. Dataset complexity, which comprised many videos that are divided into number of classes. one of the public dataset used to test models is the UCF101 dataset, which contains huge number of videos.

B. Based on the case study, the accuracy requires a large number of memory cells. Increasing the units of memory LSTM cells, leads to a better performance compared to the state-of-the-art model.

1). Using different optimization methods. Optimizer is used to reduce errors in neural networks. By changing their attributes or parameters. Optimizing neural networks involves reducing the difference between predicted and actual output. The proposed LSTM is used to compare the results obtained through Adam and SGD optimizers.

Stochastic Gradient Descent, one of the most popular optimization algorithms used in neural networks is gradient descent. The algorithm updates argument values to optimize them according to the gradient of the value function. In SGD, error values have been evaluated for each data point during each iteration for the purpose of minimizing errors. Additionally, Adam optimizer popular optimization algorithms to update the learning rate, this algorithm use both the first and second moments of the gradient by combining the advantages of AdaGrad and RMSProp. Adam method offers the best of both optimizers. This method updates both the squared gradientand the gradient exponential moving average, which estimates the first and second moments. There are some steps to optimization according to Adam optimizer [23].

3.2 The improved Gated Recurrent Unit (GRU)

Among the many types of RNNs, GRU network is one of the most common. A recurrent hidden state depends on the previous one for its activation each time. The disappearing gradient effect makes RNNs not easy to train. Some variants of RNNs, such as GRU, have empirically demonstrated their ability to hold data for a long time in various tasks, such as image or video caption generation. Where the input and forget gates have combined into one update gate. The GRU reduces the gating signals to two from three in the LSTM design. Figure 4 describes the GRU design consisting of an update gate z and a reset gate r. The update gate controls how quickly information from the previous one has been allowed to reach the current state.

The update gate is represented by z_t, the input is represented by x_t,the reset gate is represented by r_t, and the parameters representing output are y_t. While the previous h_{t-1} state information is multiplied by respective weights,z_t is an update gate that determines how much the unit updates its activation. In cases when r_t is close to 0, the reset gate forgets the previously computed state, and the unit acts as if it is reading the first symbol. σ Sigmoid activation has been used to derive $z_t$and r_t. On the other hand, the reset gate governs how much status information from the previous moment can be retained or ignored. The forward propagation information can be calculated as follows at time step t [22]:

$$z_t = \sigma(W_z x_t + U_z h_{t-1}) \quad (7)$$

$$r_t = \sigma(W_r x_t + U_r h_{t-1}) \quad (8)$$

$$h'_t = Softmax(W x_t + r_t * h_{t-1}) \quad (9)$$

$$h_t = (1 - z_t) h_{t-1} + z_t h'_t \quad (10)$$

$$y_t = \sigma(W_o h_t) \quad (11)$$

The improved GRU network model over the classical model includes the following:

(1) The GRU has been expanded to include more layers: Due to its special memory cells, the proposed GRU was very effective at extracting features from sequence data. This model extracts temporal features from sequence data by passing the input data through two layers of GRU, each with 128 memory cells.

(2) Increasing the number of GRU units in each model layer. In reality, the number of units describe the hidden state of 128 cells. The number of units represent how many neurons have been connected to the layer holding the concatenated vector of hidden states and input.

(3) Using different optimizers in the GRU model: According to the above formula, the GRU model has the advantage of long-distance preservation of key information because it uses fewer gates, continually discards redundant information, and uses the hidden state to store information dependencies. The optimizer algorithms are the ADAM and SGD techniques (learning rate equal to 0.001). This technique is used to enhance the first-order optimization. Algorithm based on stochastic gradient descent allows relevant parameters to be dynamically updated.

3.2.1 Dataset preparation

There are several datasets accessible on the internet that may be used to pre-trained models while there may be a UCF dataset suitable for human action. The UCF datasets present a project by the Department of Electrical Engineering and Computer Science at the University of Central Florida that has led to the use of unscripted footage that is difficult to compile.

3.2.2 UCF101 Dataset

The UCF101 database has features of 101 human action classes and contains 13320 video clips collected from YouTube; the video clips were timed and had a fixed frame rate of 25 frames per second with 320 x 240 pixels. Each action class has split into 25 groups, each with four to seven video segments. The train and test splits are utilized for action recognition on UCF101 under the literature to guarantee that video clips from the same film have not been used for training and testing. Because current algorithms often attain the 95th percentile or superior accuracy, the data sets do not adequately simplify actual data [15].

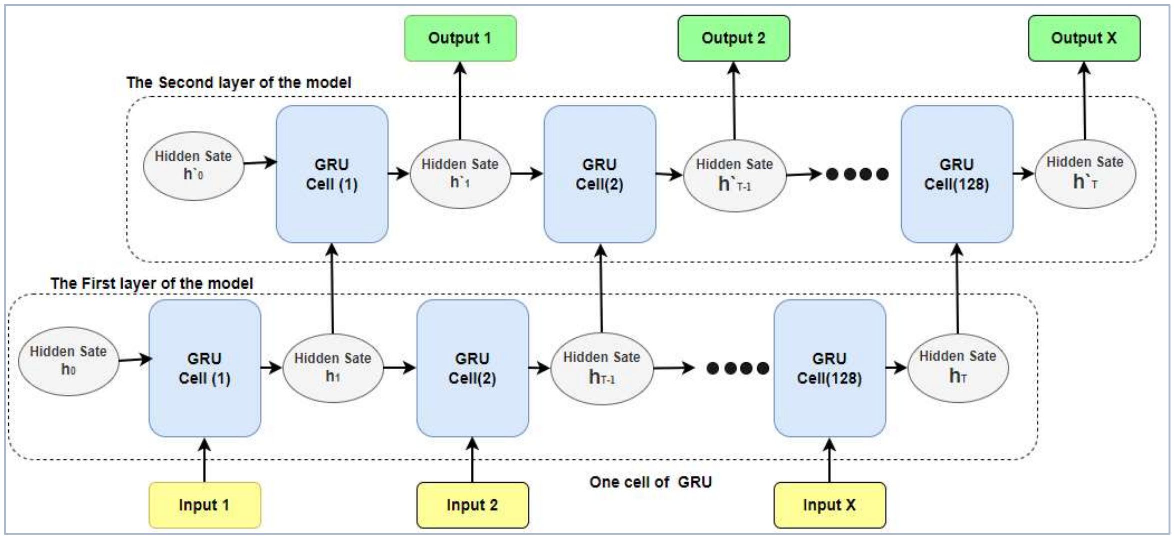


Figure 4. A block diagram of the proposed GRU model

3.2.3 Data preprocessing

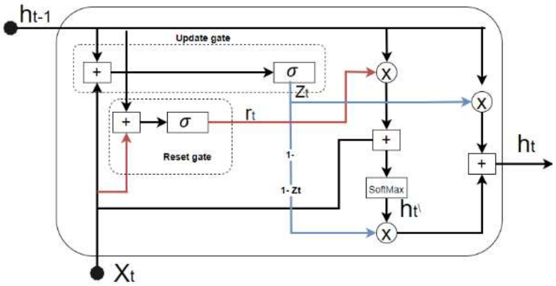
To achieve higher accuracy in human action classification and recognition on videos, the quality of videos is improved before training and testing the proposed model. Following are some of the image processing techniques that have been applied to the proposed model:

3.2.4 Image resizing:

A model can be trained faster with smaller image frames since smaller images take less time to process. The RNN architectures of the deep learning model require input images with the same size, from 320x240 pixels to 128x128 pixels. Therefore, the images in the UCF101 dataset are of size 320x240 pixels but late they are resized to 128x128 pixels according to the RNN model.

3.2.4.1 Color conversion:

Color conversion from RGB to grayscale has been applied to all video frames in the dataset. Pixels R, G, and B values represent the color in an RGB image. Therefore, these three numbers correspond to pixels with values ranging from 0 to 255. Moreover, grayscale images contain one value per pixel, with each value ranging from 0 to 1, corresponding to the pixel's intensity.



3.2.4.2 Histogram equalization:

By using this technique, the image's brightness distribution is equalized, and the grayscale image's uniform pixel distribution is modified so that image details are clearer and the contrast is increased. In this method, training images will be modified in order to make the training process more dynamic [24].

$$x' = T(x) = \sum_{i=0}^{x} n_i \frac{MAXintensity}{N} \qquad (12)$$

In grayscale images, n_i represents the number of pixels with intensity i, and N represents the total number of pixels. A histogram equalization transforms pixels' intensities (x) into new intensities (x'). A new intensity value is achieved, the sum of a cumulative histogram and a scale factor must be fit within a range of intensity values (0-255).

3.2.4.3 Unsharp masking:

Due to a large number of blurry images in the dataset, this technique was used to solve blurry images [33]. In both the training and testing phases, the proposed model could not detect or recognize everything. The enhanced grayscale image $y(n,m)$ is generated based on the input grayscale image $x(n,m)$. The positive scaling factor $z(n,m)$ determines the degree of contrast enhancement. It is calculated as the output of a linear high-pass filter.

$$y(n,m) = x(n,m) + \lambda z(n,m) \qquad (13)$$

3.2.4.4 Data normalization

A data normalization technique is applied within the range 0 and 1 to improve the classification accuracy. Which has a significant impact on feature extraction and classification processes. Training neural networks is faster when input images are normalized. Input images have been normalized by centering their pixel values around a few preprocessing outputs as shown in Figure 5.

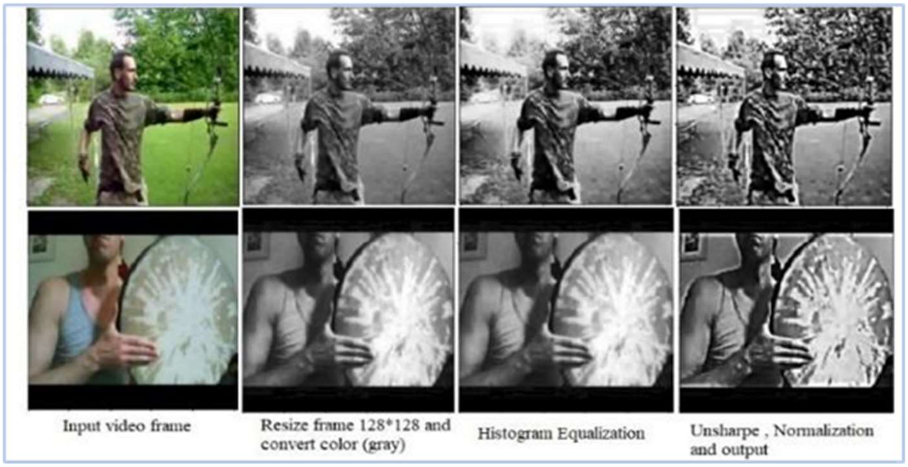

Input video frame
Resize frame 128*128 and convert color (gray)
Histogram Equalization
Unsharpe , Normalization and output

Figure 5. The outcomes of preprocessing algorithms

3.2.5 Data augmentation

Through "data augmentation," new data can be generated from existing data to increase the amount of data. To improve the accuracy of machine learning algorithms, data augmentation techniques generate more versions of a real dataset. This method is used on the training data to avoid overfitting by preventing the model from fitting well on the training data. The images are horizontally-shifted, sheared, rotated, and zoomed-in in order to improve the training images. Additionally, the parameters used for augmentation are randomly selected from [-10.0, 10.0] %, [0.0, 0.2] radian CCW counter-clockwise), and, [0.0,0.2] radian CW (clockwise) with angle 200, and [-20.0, 20.0] for horizontally-shifted, rotated, zoomed-in, and sheared respectively. An example of augmented images can be seen in figure 6.

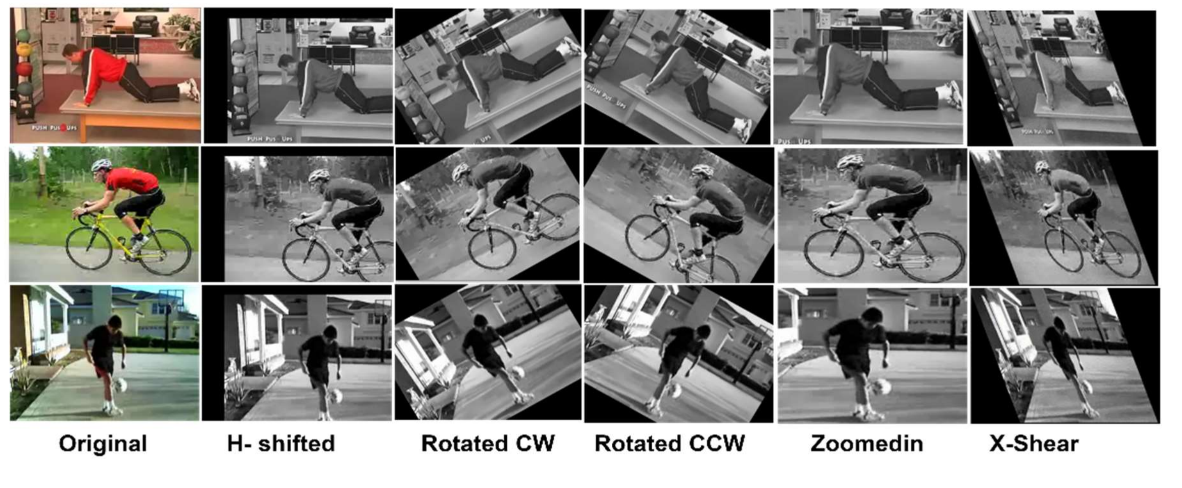


Figure 6. Augmented Human action recognition using UCF101 Dataset

4. EXPERIMENTAL EVALUATION

Experimental evaluation of the proposed technique is done on different benchmark action recognition models. The UCF101 dataset is presented in this section. The LSTM model was built with 4 LSTM layers, and each layer has 128 neurons with different optimizations for each experiment; dropout 0.5, and 1 dense layer in the final layer SoftMax activation. Also, there are two GRU layers in a GRU model. A total of 128 neurons are present in each layer, with different optimizations for each experiment at dropout 0.5, and a dense layer with SoftMax activation at the end of the sequence. The dataset is divided into the training dataset and the validation dataset.

Training data are 80% of the whole dataset, and validation and test data are 20%. As part of the training, validation and testing processes use training data to build the model and verification data to monitor its performance. during training and testing data to evaluate the model's performance. The result of the improved models on UCF101 data without any video preprocessing, data augmentation or optimization are shown in Table 1 for RNN models. Moreover, the evaluation of the accuracy and loss of the improved LSTM model is shown in Figure7-a without performing any preprocessing and augmentation on the UCF101 dataset.

Table 1: The Models Accuracy Without Preprocessing And Data Augmentation Technique With RNN Models

No. of epochs	Batch size	Test accuracy without preprocessing and data augmentation on UCF101dataset by LSTM model	Test accuracy without preprocessing and data augmentation on UCF101dataset by GRU model
10	16	0.722352948188781	0.751176486015319
	32	0.72411768913269	0.753529422283172
	64	0.730576243004532	0.760588243007659
20	16	0.757865433007658	0.760588243007659
	32	0.770337811369902	0.774117653369903
	64	0.799546063732140	0.793647063732147
30	16	0.8010000338976929	0.807654311920929
	32	0.818753717102140	0.809764717102050
	64	0.838566537826536	0.815823537826538
40	16	0.8433454653369672	0.8124117653369903
	32	0.8688541768913255	0.8186411768913269
	64	0.8688794740943901	0.8301764740943900
50	16	0.8666572358551133	0.8550882358551025
	32	0.8766774767943456	0.8571176474094390
	64	0.8891374475094377	0.8691176474094390

Additionally, data augmentation and preprocessing improved the test LSTM and GRU models' performance on the UCF101 dataset. This improves the accuracy of the data during training and testing. And also, Figure 7-b shows the accuracy and loss of the improved LSTM model after implementing some preprocessing and data augmentation on the dataset. The highest accuracy is achieved with an epoch equal to 50 and a batch size of 64, as presented in Table 2.

Table 2. The Model's Accuracy With Preprocessing And Data Augmentation Technique With RNN Models.

No. of epochs	Batch size	Test accuracy with preprocessing and data augmentation on UCF101dataset by LSTM model	Test accuracy with preprocessing and data augmentation on UCF101dataset by GRU model
10	16	0.751176486015456	0.72 41176533699036
	32	0.758529422283223	0.7323529481887817
	64	0.770588243007732	0.74888235378265381
20	16	0.799588243007536	0.75 41176533699036
	32	0.804117653369104	0.77 23529601097107
	64	0.807647063732538	0.79 70588326454163
30	16	0.821000011920990	0.80 88235378265381
	32	0.830764717105090	0.81 76470637321472
	64	0.845823537826678	0.83 05882430076599
40	16	0.8524117653378903	0.84 41176533699036
	32	0.8606411768915369	0.8535294222831726
	64	0.8681764740943680	0.8600588326454163
50	16	0.8770882358551075	0.86 23529481887817
	32	0.8871176474094397	0.87 05882430076599
	64	*0.9161764740943887*	*0.90670588326454163*

The results showed that using Adam with a SoftMax activation function to optimize the LSTM model worked well. This provided a higher level of accuracy than the GRU model, depending on the learning rate. An optimization algorithm uses the learning rate to set the size of each step at each iteration as it moves toward a minimum loss function. Figure 7-c illustrated the accuracy and loss of the improved LSTM model with using optimizer method. Table 3 shows the Adam optimizer achieved the highest accuracy value of 95.19% by LSTM learning rate of 0.001 at 50 epochs. The experiment consists of two training sessions with two optimization methods. The number of epochs trained the number of optimization experiments is 10, 20, 30, 40, and 50, and the learning rates are 0.001, 0.01, and 0.1.Training a deep learning model requires modifying each epoch's weights and minimizing the loss function. The neural network optimizer modifies attributes such as the weights and learning rate of the network.

In this way, accuracy is improved, and overall loss is reduced. Improved LSTM with SGD has the highest accuracy; however, this can be influenced by a number of factors, including the data used, the preprocessing results, and the architecture used, as well as tuning parameters. Table 4 shows that SGD achieved 92.90% accuracy using LSTM learning at a rate of 0.001 at 50 epochs.

Table 3. The Model's Accuracy Of Adam Optimizer Technique

No. of epochs	Learning rate	Test accuracy with preprocessing and data augmentation on UCF101dataset by LSTM model and Adam optimizer	Test accuracy with preprocessing and data augmentation on UCF101dataset by GRU model and Adam optimizer
10	0.001	0.799411792755127	0.7611764860153198
	0.01	0.7629411852359772	0.7441176652908325
	0.1	0.7570588266849518	0.7229411852359772
20	0.001	0.8370588445663452	0.8297522974014282
	0.01	0.8170588445663452	0.8064706134796143
	0.1	0.8109412031173706	0.782843479156494
30	0.001	0.8805882549285889	08685612144470215
	0.01	0.8623529601097107	0.8404058963775635
	0.1	0.85352941632270813	0.8251176533699036
40	0.001	0.9114706015586853	0.8964706134796143
	0.01	0.9058823704719543	0.8877743158340454
	0.1	0.89823530077934265	0.8841764979362488
50	0.001	0.9519412031173706	0.92901477575302124
	0.01	0.9389411911964417	0.90988235378265381
	0.1	0.9195823823928833	0.90667280974388123

In general, there have been several previous implementations of video classification, particularly of videos of human action. These works were chosen based on the use of the UCF101 dataset, which is larger and more public and on the use of RNN architecture, which is more precise. As shown in Table 5, some recent studies are compared to the accuracy of the proposed model. One can find that the proposed approach using LSTM obtained an average accuracy of 95%, which outperforms other deep learning techniques. Each deep neural network approach contains several hidden layers that may store different data types. The overfitting problem decreases as the number of layers increases. As a result, the model's overall accuracy approaches an optimal value. Table 6 displays a comparative analysis of the strengths and weaknesses of some papers on human action recognition by different techniques and models with our models.

Table 4: The Evaluation Models Accuracy Of SGD Optimizer Technique

No. of epochs	Learning rate	Test accuracy with preprocessing and data augmentation on UCF101dataset by LSTM model and SGD optimizer	Test accuracy with preprocessing and data augmentation on UCF101dataset by GRU model and SGD optimizer
10	***0.001***	*0.770411792755127*	*0.7511764860153198*
	0.01	*0.7529411852359772*	*0.7341176652908325*
	0.1	*0.7270588266849518*	*0.7129411852359772*
20	***0.001***	*0.8470588445663452*	*0.7997522974014282*
	0.01	*0.8270588445663452*	*0.7864706134796143*
	0.1	*0.8109412031173706*	*0.762843479156494*
30	***0.001***	*0.8805882549285889*	*08385612144470215*
	0.01	*0.8623529601097107*	*0.8204058963775635*
	0.1	*0.85352941632270813*	*0.8151176533699036*
40	***0.001***	*0.9014706015586853*	*0.8564706134796143*
	0.01	*0.8988823704719543*	*0.8377743158340454*
	0.1	*0.89223530077934265*	*0.8301764979362488*
50	***0.001***	*0.9299412031173706*	*0.91901477575302124*
	0.01	*0.9189411911964417*	*0.90388235378265381*
	0.1	*0.9095823823928833*	*0.90067280974388123*

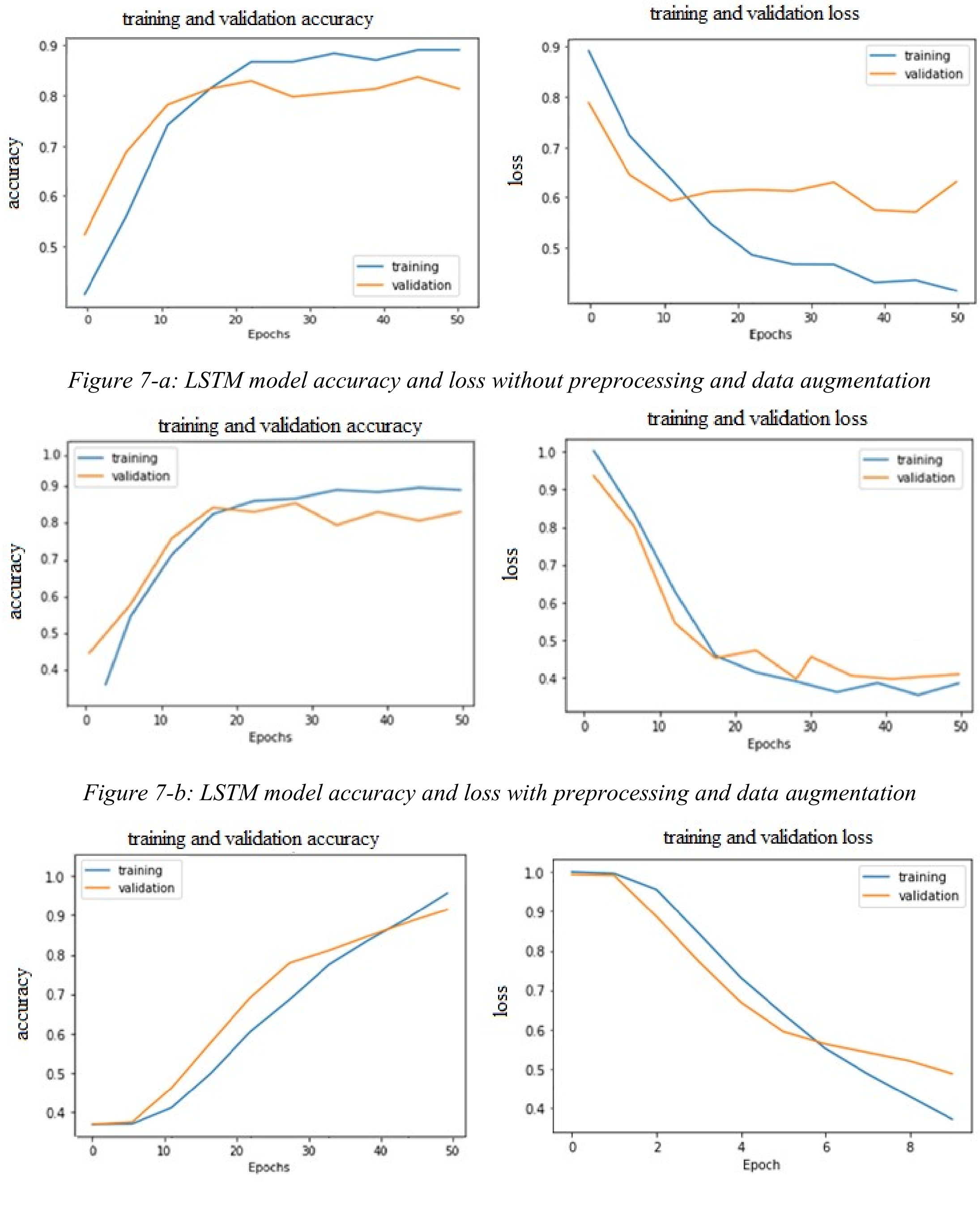


Figure 7-a: LSTM model accuracy and loss without preprocessing and data augmentation

Figure 7-b: LSTM model accuracy and loss with preprocessing and data augmentation

Figure 7-c: LSTM model accuracy and loss with preprocessing, data augmentation and optimizer

Figure 7. Graphs Representing The LSTM Model Accuracy And Loss For UCF101 Dataset.

Table 5. A Comparison Of The Proposed Model's Accuracy With Related Previous Studies.

Reference	Methodology	Dataset	Accuracy %
[25]	Single Frame CNN Model (Optical Flow)	UCF101	73.90
[26]	Spatio–Temporal Differential LSTM (ST-D LSTM)	UCF101	75.70
[27]	Recurrent Neural Networks as Encoding	UCF101	81.90
[28]	Multi-Level Recurrent Residual Networks	UCF101	81.90
[29]	Adaptive Recurrent-Convolutional Hybrid ARCH	UCF101	85.3
[30]	Regularizing Long Short-Term Memory with 3D	UCF101	86.90
[31]	Long Term Recurrent Convolutional Networks	UCF101	87.60
[32]	Recurrent Neural Networks	UCF101	88.50
[33]	Two-Stream LSTM	UCF101	88.60
[34]	Gated Recurrent Unit	UCF101	88.0%
[35]	Convolution Neural Network (CNN), Recurrent Neural Networks	UCF101	89.10
[36]	Recurrent Neural Networks	UCF101	89.20
[37]	GMM + KF + GRNN	UCF101	89.30
[38]	TS-LSTM and temporal-inception	UCF101	91.10
[39]	Deep bi-directional LSTM	UCF101	91.20
Proposed Models	Gated Recurrent Unit (GRU)	UCF101	92.90
	Long Short-Term Memory (LSTM)		95.10

Table 6. Analyzing The Proposed Models With Related Previous Research Using The Plus, Minus, Interesting Method.

Reference	Years	Plus	Min	Interesting
[41]	2015	State-of-the-art research and innovations in human action recognition based on a taxonomy of approaches	It did not cover many deep-learning approaches and followed a specific taxonomy.	Semantic-based human recognition approaches and a short representation of their application. Recognition of human action in video and images.
[42]	2016	The article has an evaluation of different methods for human action recognition and a comparison of their results.	The article covers very few aspects of HAR approaches.	Recognition of human action only in videos. Analysis of different HAR methods
[43]	2016	3D skeleton-based human action recognition approaches.	The study focuses mainly on 3D skeleton-based HAR, ignoring a wide range of other strategies	Emphases mostly on 3D skeleton-based HAR and Recognition of human action in the video only.
[44]	2017	Techniques for recognizing complex events	It focuses on complex event techniques and does not cover HAR approaches for other activities.	Interested to complex event techniques. Recognition of human action in videos and images.
[45]	2017	The article talks about how to identify, track, and understand group interactions and abnormal activity detection in large crowds, including summaries of the available data underlying the wor	The analysis of crowds for video surveillance purposes and the detection of abnormalities. Other applications are not covered.	The article is identifying, tracking and understanding group activities detection in large crowds of people with available datasets.
[46]	2018	The study focuses on three architectures of neural networks that have based on RGB-D graphics for recognizing human motion	A deep learning-based RGB-D motion recognition system that does not include many other approaches	RGB-D-based human motion recognition with deep learning and converging neural networks. Recognition human action in video just.
[47]	2019	Analyze multiple HAR techniques in depth and comprehensively.	These are some of the most recent research works in the fields of Intelligent Video Surveillance, Wireless Sensor Networks-based HAR, and camera-based health monitoring.	Applications of HAR on mobile devices with an emphasis on data fusion.
[48]	2022	Deep learning-based HAR using video and classification of datasets based on level of complexity	The focus of this paper is on deep learning-based HAR rather than handcrafted approaches	Video-based HAR via deep learning and classification of datasets gives different complexity levels. Recognition of human action in videos and images.
Our models		In this research proposes two models for various action recognition using LSTM and GRU.A comparative study was conducted during the experimental tests performed on HAR. The proposed models have a high capacity for learning sequences and changing features from frame to frame. Additionally, applying some preprocessing techniques to improve the performance of the RNN for feature extraction.	Doesn't cover many databases of HAR because the ucf101 one of huge dataset for human action.	Those models emphasize on deep learning-based with handcrafted approaches. Human actions in images and videos are recognized. It is the process of automatically recognizing what actions human actors perform in a video. Models based on deep learning are capable of accurately identifying and classifying objects. Many different methods for preprocessing the data have been used.

5. CONCLUSION AND FUTURE WORK

This study describes practical strategies for classifying human action recognition. The present models are an accurate deep learning technique based on modified LSTM and GRU models. In this paper, the hidden layers of the LSTM deep learning have been modified into four layers and the GRUs model into two layers. Additionally, several preprocessing algorithms have been applied to the UCF101 dataset to reduce the overfitting problem using different transformations such as translation, scaling, and rotation. According to the results, the Adam optimizer with a SoftMax activation function is an effective way to optimize an LSTM model. LSTM model with Adam optimizer has a maximum accuracy value of 95.1 % with a learning rate of 0.001 at 50 epochs. However, the maximum accuracy value for a GRU model with a learning rate of 0.001 at 50 epochs is 92.9%. The proposed models of this study have been compared to those that have used traditional (handcrafted) machines, and deep learning techniques. The model can be effectively fused with another deep learning algorithm. Also, we plan to enhance the proposed model to classify human actions throughout the video more reliably. In terms of classification and recognition, we plan to improve additional RNN architectures, including VGG-16, inception-V4 and others.

REFERENCES

[1] A. Nanda, D. S. Chauhan, P. K. Sa, and S. Bakshi, "Illumination and scale invariant relevant visual features with hypergraph-based learning for multi-shot person re-identification," Multimedia Tools Appl., pp. 1–26, Jun. 2017.

[2] A. Nanda, P. K. Sa, S. K. Choudhury, S. Bakshi, and B. Majhi, "A neuromorphic person re-identification framework for video surveillance," IEEE Access, vol. 5, pp. 6471–6482, 2017.

[3] S. Herath, M. Harandi, and F. Porikli, "Going deeper into action recognition: A survey," Image Vis. Comput., vol. 60, pp. 4–21, Apr. 2017.

[4] J. Y.-H. Ng, M. Hausknecht, S. Vijayanarasimhan, O. Vinyals, R. Monga, and G. Toderici, "Beyond short snippets: Deep networks for video classification," in Proc. IEEE Conf. Comput. Vis. Pattern Recognit., Jun. 2015, pp. 4694–4702.

[5] Z. C. Lipton, J. Berkowitz, and C. Elkan. (2015). "A critical review of recurrent neural networks for sequence learning.".

[6] K. Greff, R. K. Srivastava, J. Koutník, B. R. Steunebrink, and J. Schmidhuber, "LSTM: A search space odyssey," IEEE Trans. Neural Netw. Learn. Syst., vol. 28, no. 10, pp. 2222–2232, Oct. 2017.

[7] Pavithra, M., Saruladha, K., &Sathyabama, K. (2019, March). GRU based deep learning model for prognosis prediction of disease progression. In 2019 3rd International Conference on Computing Methodologies and Communication (ICCMC) (pp. 840-844). IEEE.

[8] Alom, M. Z., Taha, T. M., Yakopcic, C., Westberg, S., Sidike, P., Nasrin, M. S., ... &Asari, V. K. (2018). The history began from alexnet: A comprehensive survey on deep learning approaches. arXiv preprint arXiv:1803.01164.

[9] Lara, O.D.; Labrador, M.A. A survey on human activity recognition using wearable sensors. IEEE Commun. Surv. Tutor. 2013, 15, 1192–1209.

[10] Chen, Y.; Zhong, K.; Zhang, J.; Sun, Q.; Zhao, X. LSTM networks for mobile human activity recognition. In Proceedings of the 2016 International Conference on Artificial Intelligence: Technologies and Applications, Bangkok, Thailand, 24–25 January 2016; pp. 50–53

[11] Fu, Y., Zhang, T., & Wang, W. (2017). Sparse coding-based space-time video representation for action recognition. Multimedia Tools and Applications, 76(10), 12645-12658.

[12] Angelini, F., Fu, Z., Velastin, S. A., Chambers, J. A., & Naqvi, S. M. (2018, April). 3d-hog embedding frameworks for single and multi-viewpoints action recognition based on human silhouettes. In 2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP) (pp. 4219-4223). IEEE.

[13] Ullah, A., Ahmad, J., Muhammad, K., Sajjad, M., &Baik, S. W. (2017). Action recognition in video sequences using deep bi-directional LSTM with CNN features. IEEE access, 6, 1155-1166.

[14] S. K. Choudhury, P. K. Sa, R. P. Padhy, S. Sharma, and S. Bakshi, "Improved pedestrian detection using motion segmentation and silhouette orientation," Multimedia Tools Appl., pp. 1–40, Jun. 2017.

[15] Muhamad, Azhee W., & Mohammed, Aree A. (2022). “Review on recent Computer Vision Methods for Human Action Recognition”.Advances in Distributed Computing and Artificial Intelligence Journal, pp. 361-379,2021, DOI:https://doi.org/10.14201/ADCAIJ2021104361379.

[16] H. Wang and C. Schmid, "Action recognition with improved trajectories," Proc. IEEE Int. Conf. Comput. Vis., pp. 3551–3558, 2013, doi: 10.1109/ICCV.2013.441.

[17] Ullah, A., Ahmad, J., Muhammad, K., Sajjad, M., &Baik, S. W. (2017). Action recognition in video sequences using deep bi-directional LSTM with CNN features. IEEE access, 6, 1155-1166.

[18] Li, D., & Wang, R. (2022). Context-LSTM: a robust classifier for video detection on UCF101. arXiv preprint arXiv:2203.06610.

[19] Yao, G., Liu, X., & Lei, T. (2018, August). Action recognition with 3d convnet-gru architecture. In Proceedings of the 3rd International Conference on Robotics, Control and Automation (pp. 208-213).

[20] Zhang, X., Sun, Y., Jiang, K., Li, C., Jiao, L., & Zhou, H. (2018). Spatial sequential recurrent neural network for hyperspectral image classification. IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing, 11(11), 4141-4155.

[21] Chen, X., Wei, L., & Xu, J. (2017). House price prediction using LSTM. arXiv preprint arXiv:1709.08432.

[22] K. Greff, R. K. Srivastava, J. Koutnik, B. R. Steunebrink, and J. Schmidhuber, “LSTM: A Search Space Odyssey,” IEEE Trans. Neural Networks Learn. Syst., vol. 28, no. 10, pp. 2222–2232, 2017, doi: 10.1109/TNNLS.2016.2582924.

[23] Keskar, N. S., &Socher, R. (2017). Improving generalization performance by switching from adam to sgd. arXiv preprint arXiv:1712.07628.

[24] K.G. Dhal, A. Das, S. Ray, J. Gálvez, and S. Das, “Histogram equalization variants as optimization problems: a review,” Archives of Computational Methods in Engineering, vol. 28, no. 3, pp.1471-1496, 2021.

[25] I. C. Duta, B. Ionescu, K. Aizawa, and N. Sebe, "Spatio-temporal VLAD encoding for human action recognition in videos," Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics), vol. 10132 LNCS, pp. 365–378, 2017, doi: 10.1007/978-3-319-51811-4_30.

[26] Hu, K., Zheng, F., Weng, L., Ding, Y., & Jin, J. (2021). Action Recognition Algorithm of Spatio–Temporal Differential LSTM Based on Feature Enhancement. Applied Sciences , 11(17), 7876.

[27] J. Patalas-maliszewska, D. Halikowski, and R. Damaševičius, "An automated recognition of work activity in industrial manufacturing using convolutional neural networks," Electron., vol. 10, no. 23, pp. 1–17, 2021, doi: 10.3390/electronics10232946.

[28] K. Simonyan and A. Zisserman, "Two-stream convolutional networks for action recognition in videos," Adv. Neural Inf. Process. Syst., vol. 1, no. January, pp. 568–576, 2014.

[29] M.Xin,H.Zhang,H.Wang,M.Sun,D.Yuan,Arch: Adaptive recurrent-convolutional hybrid networks for long-term action recognition, Neurocomputing 178 (2016) 87–102

[30] B. Mahasseni and S. Todorovic, "Regularizing Long Short Term Memory with 3D Human-Skeleton Sequences for Action Recognition," Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit., vol. 2016-December, pp. 3054–3062, 2016, doi: 10.1109/CVPR.2016.333.

[31] Z. Qin, Y. Zhang, S. Meng, Z. Qin, and K. K. R. Choo, "Imaging and fusing time series for wearable sensor-based human activity recognition," Inf. Fusion, vol. 53, no. May 2019, pp. 80–87, 2020, doi: 10.1016/j.inffus.2019.06.014.

[32] Q. Zhang, M. Zhang, T. Chen, Z. Sun, Y. Ma, and B. Yu, "Recent advances in convolutional neural network acceleration," Neurocomputing, vol. 323, pp. 37–51, 2019, doi: 10.1016/j.neucom.2018.09.038.

[33] H. Idrees et al., "The THUMOS challenge on action recognition for videos 'in the wild,'" Comput. Vis. Image Underst., vol. 155, pp. 1–23, 2017, doi: 10.1016/j.cviu.2016.10.018.

[34] Zhang, L., & Xiang, X. (2020). Video event classification based on two-stage neural network. Multimedia Tools and Applications. doi:10.1007/s11042-019-08457-5

[35] L. Wang, Y. Xu, J. Cheng, H. Xia, J. Yin, and J. Wu, "Human Action Recognition by Learning Spatio-Temporal Features with Deep Neural Networks," IEEE Access, vol. 6, pp. 17913–17922, 2018, doi: 10.1109/ACCESS.2018.2817253.

[36] B. Leng, X. Zhang, M. Yao, and Z. Xiong, "A 3D model recognition mechanism based on deep Boltzmann machines," Neurocomputing,

vol. 151, no. P2, pp. 593–602, 2015, doi: 10.1016/j.neucom.2014.06.084

[37] Jaouedi, N., Boujnah, N., &Bouhlel, M. S. (2019). A New Hybrid Deep Learning Model For Human Action Recognition. Journal of King Saud University - Computer and Information Sciences. doi:10.1016/j.jksuci.2019.09.004

[38] C.-Y. Ma, M.-H. Chen, Z. Kira, G. AlRegib, TS-LSTM and temporalinception: Exploiting spatiotemporal dynamics for activity recognition,Signal Process., Image Commun. 71 (2019) 76–87.

[39] A. Ullah, J. Ahmad, K. Muhammad, M. Sajjad, S.W. Baik, Action recognition in video sequences using deep Bi-directional LSTM with CNN features, IEEE Access 6 (2018) 1155–1166

[40] K. Hu, F. Zheng, L. Weng, Y. Ding, and J. Jin, "Action recognition algorithm of spatio–temporal differential lstm based on feature enhancement," Appl. Sci., vol. 11, no. 17, 2021, doi: 10.3390/app11177876.

[41] Ziaeefard, M., & Bergevin, R. (2015). Semantic human activity recognition: A literature review. Pattern Recognition, 48(8), 2329-2345.

[42] Akansha, U. A., Shailendra, M., & Singh, N. (2016, March). Analytical review on video-based human activity recognition. In 2016 3rd International Conference on Computing for Sustainable Global Development (INDIACom) (pp. 3839-3844). IEEE.

[43] Presti, L. L., & La Cascia, M. (2016). 3D skeleton-based human action classification: A survey. Pattern Recognition, 53, 130-147.

[44] Alevizos, E., Skarlatidis, A., Artikis, A., & Paliouras, G. (2017). Probabilistic complex event recognition: A survey. ACM Computing Surveys (CSUR), 50(5), 1-31.

[45] Grant, J. M., & Flynn, P. J. (2017). Crowd scene understanding from video: a survey. ACM Transactions on Multimedia Computing, Communications, and Applications (TOMM), 13(2), 1-23.

[46] Wang, P., Li, W., Ogunbona, P., Wan, J., & Escalera, S. (2018). RGB-D-based human motion recognition with deep learning: A survey. Computer Vision and Image Understanding, 171, 118-139.

[47] Nweke, H. F., Teh, Y. W., Mujtaba, G., & Al-Garadi, M. A. (2019). Data fusion and multiple classifier systems for human activity detection and health monitoring: Review and open research directions. Information Fusion, 46, 147-170.

[48] Pham, H. H., Khoudour, L., Crouzil, A., Zegers, P., & Velastin, S. A. (2022). Video-based human action recognition using deep learning: a review. arXiv preprint arXiv:2208.03775.