International Journal of
INTELLIGENT SYSTEMS AND APPLICATIONS IN ENGINEERING
ISSN:2147-6799 www.ijisae.org Original Research Paper

A Robust LSTM Model for Video Detection and Classification: An Optimization Procedure

Sunitha Sabbu*[1], Dr. Vithya Ganesan[2]

Submitted: 09/01/2024 **Revised**: 15/02/2024 **Accepted**: 23/02/2024

Abstract: The act of human activity recognition plays a vital role in a wide range of fields, such as monitoring and healthcare. Researchers are now using various deep learning and machine learning techniques to analyze and interpret the collected data. With deep learning techniques, such as DL, researchers have been able to improve the performance of various HAR systems by extracting high-level features. In this work, a novel research methodology for human abnormal activities like abuse, assault, arson, arrest, and fighting was introduced. The procedure utilizes the VGG16-based LSTM neural network. The proposed methodology combines the features from the LSTM layers to generate a representation which further supports in enhancing the performance accuracy of the classification task. Due to the complexity of human action recognition and video detection, it can be very expensive to train a model. In this paper, our goal was to minimize the training time and improve the accuracy of our work by implementing a low-cost LSTM structure for video detection. The paper presents a LSTM structure that can perform well after the hyperparameter tuning in validating UCF101's entire dataset. The structure, which is called Context-LSTM, can process the deep temporal features. The use of the proposed LSTM structure can reduce the training time, while maintaining the top-rated accuracy and helps to minimize the memory usage of the GPU. The classification model was able to categorize the "fighting," "arson," "abuse," and "arrest," class labels 95.43%, 95.97%, and 97.37% accuracy, respectively. The proposed model has also tested well and achieved an accuracy rate of 95.81%. with a misclassification error rate of 4.19%.

Keywords: *Activation Function, Batch Normalization, Deep Temporal Features, Human Activity Recognition, Long-Short Term Memory, Video Detection.*

1. Introduction

The field of machine learning has been growing rapidly. This technology is used in various applications such as natural language processing and image recognition. Deep learning has become a main research subject. The development of deep neural networks has led to the improvement of image and video detection. One of the most common applications of this technology is image recognition. The VGG net and AlexNet are two of the most widely used deep neural networks. The depth and width of a deep neural network can be expanded using the GoogLeNet technique. RNN is also proposed for learning temporal series. LSTM is a variation of RNN that is designed to prevent the network from experiencing a gradient explosion.

The development of LSTM is a significant advance in the field of temporal series learning, as it allows the network to learn the features of the series more accurately and faster. In the past, the method used for learning temporal series usually passes through several layers of neural networks. With the new learning architecture, the data feature information can be retrieved from the network without affecting its multi-layer structure.

To minimize the loss of data, feature information, we suggest using ResNet as a shortcut to connect to the multi-layer networks or using RNN with bi-directional capabilities. In this paper, we present a framework called Context-LSTM that is designed to allow the network to extract contextual information from the temporal sequences. It adopts a hierarchical structure and connects the hidden and cell data streams.

The paper presents a framework that is designed to allow the network to extract contextual information from the temporal sequences. It adopts a hierarchical structure and connects the cell data streams. The proposed loss of information transmission during training is analyzed and presented. The deep temporal feature, on the other hand, refers to the extracted information from a temporal sequence. The proposed Context-LSTM classification system can achieve a top-notch accuracy in the evaluation of UCF101's deep temporal feature. It is a robust and reliable model. The paper presents a simulation of the two-stage human recognition system using the Context-LSTM framework.

The remainder of the article is organized as follows: Section 2 summarizes the literature review as background.

[1]*Research Scholar, Department of CSE*
Koneru Lakshmaiah Education Foundation, Vaddeswaram, Andhra Pradesh, India.
ORCID ID : 0000-0003-4027-5212
[2]*Professor,Department of CSE*
Koneru Lakshmaiah Education Foundation,Vaddeswaram, Andhra Pradesh, India.
ORCID ID : 0000-0002-5896-4094
** Corresponding Author Email: sunindra111@gmail.com*

Section 3 presents the dateset preparation and feature extraction with proposed RNN model. Section 4 presents comparative analysis describing the performance comparison of the proposed RNN based LSTM before and after tuning the hyper parameter. Section 5 presents the discussions and future scope followed by conclusion in section 6.

2. Background

Deep learning and machine learning techniques are commonly used in the research for the development of HAR. Most of the time, researchers have used the classical methods for identifying activities such as the Random Forest, XG Boost, Naive Bayes, and others. However, the effectiveness of these methods is often limited by the domain expertise required to extract the features. Due to the increasing number of studies on the effectiveness of deep learning techniques in detecting activity in HAR, researchers have started to adopt them instead of relying on the traditional methods. These techniques have shown to perform better than their conventional counterparts when it comes to identifying difficult activities. Due to the advantages of DL models, researchers have started to adopt them for developing HAR systems. The accuracy of this technology has significantly improved with the emergence of new techniques such as CNNs and RNNs. These techniques have shown to be very useful in developing sensor-based systems [1].

The OmniSource framework has been introduced which was designed to train models for action recognition. Through a teacher network and student network, the framework was able to achieve high-quality detection accuracy [2]. The I3D-LSTM framework has been introduced which is a multi-level classifier that uses the 3D ConvNet framework with Inception as its backbone. It was able to achieve an accurate detection rate of UCF101 [3]. The BQN framework has be introduced which is designed to provide a convenient way to split the redundancy of nearby frames [4]. In a paper, Gowda et.al, presented a framework that is designed to classify videos by selecting the frames with the most relevant object features and temporal information [5]. A method that is designed to train a confidence-based framework for action detection has been presented. The framework was able to perform training on various frames of a video [6].

In one of the studies, Lokesh and colleagues proposed a HAR system that uses LSTM and CNN. The model was able to achieve a 97.5% accuracy on WISDM's dataset, with a low complexity architecture and more than 21,922 trainable variables. The results were presented through learning curves and a convolutional matrix [7]. The researchers proposed a sensor-based system for detecting and monitoring objects in the PAMAP2. It was able to achieve an accuracy of 88%, which is higher than the other standard DL models [8]. The researchers presented a DL model known as DSFNet, which is designed for multi-sensing fusion. It uses acceleration sequences and a Transformer encoder for extraction of features, and it also utilizes angular velocity for pose direction and movement. Its performance compared to other methods is significantly higher [9]. The researchers presented a dynamic network for low-price HAR systems called RepHAR. It uses a combination of structured reparameterization and multibranch topologies for achieving a tradeoff between accuracy and speed. In the tests, it was able to achieve a precision of 0.83–2.18% and a reduction of 12–44% on various datasets. In terms of performance, RepHAR performed better than the MB CNN model on the Raspberry Pi [10]. The researchers led by Challa presented a CNN-BiLSTM model, which is designed to perform extraction of features with minimal per-processing. It can learn long-term dependencies and local features through various filter sizes, which helps improve its performance [11].

Most of the literature review provides competent and novelty-packed works that are designed to detect simple activities. However, some of these works do not perform well in detecting complex activities. Some methods try to improve the recognition rates by using external sensors. In this paper, we present a novel deep ensemble model called Deep-HAR that can detect various complex activities.

3. Methodology

The LSTM framework is suitable for extracting temporal series information, as its structure has the features of a temporal series. In this study, we will use the framework to improve its utilization. The level of semantic information extracted by LSTM is expected to increase, but it might also be lost due to the increasing number of network layers. The use of ResNet-50 as the backbone for the feature extraction process will allow us to solve this issue. The LSTM module and the ResNet-50 backbone are part of this architecture. The latter is utilized as a classifier, and it can use the collected temporal information. It also utilizes a cascading structure to link its sequences. The various LSTM blocks are connected using the Batch Normalization and ReLU activation functions. The hidden output and cell state flow of the framework are then interconnected to improve its temporal characteristics. The complete methodology followed during the present work is illustrated in Figure 1.

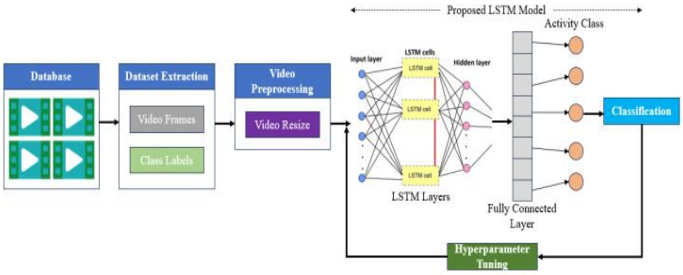


Fig 1. Research Methodology for Video Detection and Classification

3.1 Data set and Feature Extraction

The goal of this study is to analyze the recognition of human activities using movement-based features. It presents six different activities, including fighting, assault, arrest, abuse, arson, and assault. Data for each activity was collected from the Kaggle platform. In various applications, such as categorization and video summarizing, it is usually required to identify significant frames in videos. With just a few essential clips, it can be done without requiring the entire video. Currently, the only method for detecting key frames in videos is through supervised learning, which requires a large amount of data labeling.

The process of labeling involves using human annotations from different backgrounds to identify crucial frames in real-time videos. Although it is not very time-consuming and costly, it can also lead to errors. The frame rate of videos is typically 30 frames per second. This provides a lot of information to be handled by computer vision systems. Some of the applications that are commonly used for detecting key frames include visual mapping and summarization. Processing all the frames requires a lot of power and memory. In most cases, a few crucial frames can be sufficient to recognize certain actions in a video. For instance, video summarizing is a process that involves identifying significant portions of a film and presenting the full story. One method to resolve this issue is to allow human participants to inspect a video and provide annotations on significant frames. Due to the subjectivity of the task, it is impossible to come up with a consensus on the count. A trained model can then identify the key frames in previously viewed videos. The trained model will then use the annotated key frames as the gold standard. Another approach is to use deep neural networks, which can provide exceptional results for various visual applications. But deep models require large datasets that can be hard to handle for humans.

Machine learning methods are used in the recognition of human abnormal activity. Experiments are performed in Google Collaboratory, an online environment that allows users to run Jupyter notebooks. It is ideal for training complex models. The training loss is computed using the cross-entropy loss method and the detection accuracy is provided by Sklearn. The generated model undergoes a top-to-bottom accuracy assessment every epoch. The dataset was divided into two test sets, one randomly and the other training set. PyTorch provided the ResNet model. It utilized the dropout function in the FC and LSTM blocks. There were three LSTM layers in the block. The hidden size of the units in the block was also set to 512. The LSTM system's classification component was completely connected by two FC layers. The output and input dimensions of the first layer are 256 and 512 respectively. The second layer's input and output dimensions are 101 and 256. The epoch, batch size, and learning rate were set at 30, and 0.001. The algorithm known as Adam was utilized for optimization. The results of the experiment revealed that increasing the batch size significantly improved the test's accuracy.

3.2 Proposed LSTM Model Structure

LSTM (Long Short-Term Memory) is a type of recurrent neural network whose output is influenced by its current input as well as the previous inputs. An LSTM later is composed of a cell, an input gate, an output gate and a forget gate. The cell remembers values from the previous iterations (time steps). LSTM can take either individual data points or arbitrary length time series data (sequences) as it’s inputs. The dataset consists of variable-length videos in 5 different classes.

The proposed model consists of a pre-trained VGG16 model followed by some additional LSTM layers. Finally, the model contains a fully connected neural network to perform the classification. During training the layers of VGG16 are not re-trained and instead they utilize the weights of ImageNet. These steps are performed during the training phase. First, for each video in the database, video frames are extracted from a video along with its corresponding class label. Second, each video frame is resized to a fixed dimension and passed through the pre-trained VGG16 layers to extract useful features from it. Third, these features are then passed through LSTM layers, so that the layers can retain the information contained in the frame for future reference. Forth, the output of these LSTM layers is then passed through a fully connected layer to perform the classification. Fifth, the predicted output then incurs a loss using the “categorical-cross entropy” loss function and both the LSTM and fully connected layer weights are adjusted through the ADAM optimizer. Later, hyper parameters are tuned based on the training results of the model until the model achieves desirable results. Finally, the trained model is then evaluated against a test set. The model will perform the classification by finding the relationship between the frames of a video. Figure 2 depicts the architecture of the proposed LSTM-based fully connected network.

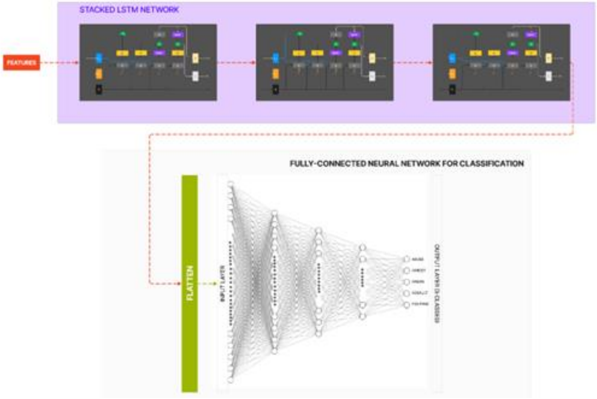


Fig 2. Proposed LSTM Model

4. Experiments and Outcomes

This section provides a template for evaluating and visualizing the results achieved through experiments of a proposed HAR model to perform multi-class abnormal activities recognition and classification using a novel LSTM model. Several experiments were conducted using the proposed LSTM model before and after hyper parameter tuning (HT). TensorFlow and Keras libraries are used for building and training the proposed model and made sure to adapt the code-based on the video datasets abuse, arrest, assault, arson, and fighting and other requirements like model selection, performance metrics, and visualizations.

4.1 Performance Metrics

Developing and evaluating the multi-class classification models demand in understanding their performance with respect to various metrics such as accuracy, precision, recall, and f1-score. These metrics can provide helpful insight into a model's performance, such as how it handles imbalanced datasets and how it can classify samples properly. Metrics are useful, but they must first be understood to fully utilize them. This includes having a good understanding of their limitations and underlying calculations. The accuracy metric is used by ML researchers to measure the proportion of predictions that a model makes that are correct. One of the most frequently used metrics for assessing the classification model's performance is accuracy, which measures how accurately a model predicts. The accuracy metric can be calculated using the equation 1.

$$Accuracy = \frac{Number\ of\ Correct\ Predictions}{Number\ of\ Predictions} \quad (1)$$

The precision metric is used in machine learning to measure the model's performance. The metric refers to the fraction of the predicted values that are part of a positive class. The precision metric value can be calculated using the equation 2.

$$Precision = \frac{\text{TP}}{Predicted\ \text{TP}+Predicted\ \text{FP}} \quad (2)$$

The performance metric recall is used to measure the fraction of the predicted values of a given class that are in a positive class. It differs from the precision metric, which is the fraction of the predicted values that are in a positive class. The recall metric value can be calculated using the equation 3.

$$\text{Recall} = \frac{\text{Predicted True Positives}}{\text{True Positives+False Negatives}} \quad (3)$$

The F1 score is an evaluation measure that combines the recall and precision harmonic mean. It is commonly used in multi-class and binary classification to evaluate a model's performance. The F1-Score metric value can be calculated using the equation 4.

$$\text{F1} - \text{Score} = \frac{\text{Number of TP}}{\text{Number of TP + Number of FP}} \quad (4)$$

4.2 Experiments

First, we have used the ResNet50 for feature extraction from the video data set. The same features are passed through the four layers of LSTM each of 20, 10, 5, and 1 output dimensionalities respectively. The LSTM layers return sequences instead of states to the next LSTM layers in the proposed model. Experiment 1 describes the training and testing phase of the proposed model. The 'Adam' optimizer with an epsilon value of 0.1 and learning rate of 0.1 are used to train the model along with KL diversions as the loss function. After selecting the videos belongs to five different classes, the model is trained for 50 epochs to understand and learn the features of the video. After the training the generalization process of the model is evaluated to understand how far the model has learned and could be able to recognize and understand the class of the video. After 50 epochs, the proposed model achieved 100% accuracy in remembering the training dataset and at the same time the training loss also reduced to almost zero. Figure 3 depicts the performance curves during the training phase. But, during the testing phase the proposed model could be able to achieve only 33.33% of classification accuracy, 1.6% of precision per each class, 2% of recall per class, and an average recall of 8.33%.

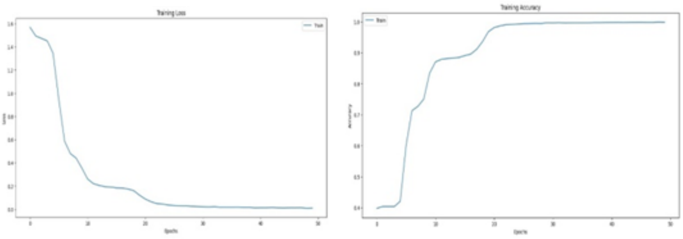
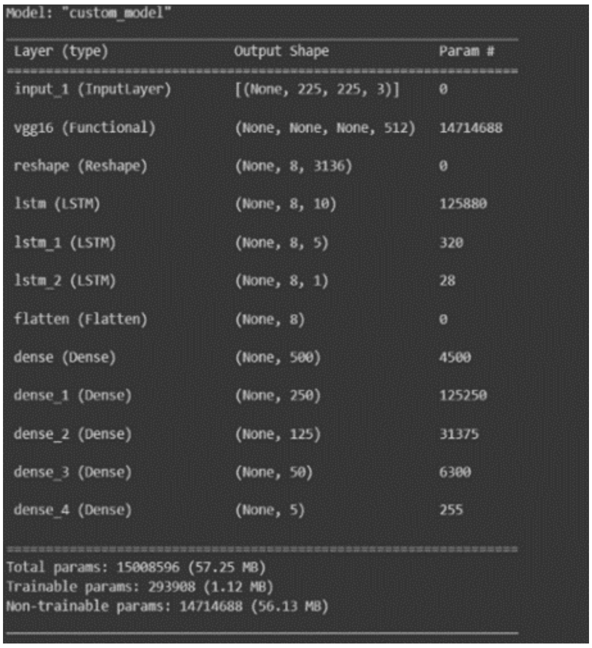


Fig 3. Training and Testing – Accuracy and Loss Curves –

Before HT

Second, we have used VGG16 for feature extraction and the same features are passed through the three layers of LSTM each of different output dimensionality. The output of these layers is connected to the fully connected neural network for the classification. The stochastic gradient descent optimizer is used during the training process with a learning rate 0.005 along with categorical_cross_entropy as a loss function. Figure 4 depicts the structure of the revised proposed LSTM model.



```
Model: "custom_model"
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 input_1 (InputLayer)        [(None, 225, 225, 3)]     0

 vgg16 (Functional)          (None, None, None, 512)   14714688

 reshape (Reshape)           (None, 8, 3136)           0

 lstm (LSTM)                 (None, 8, 10)             125880

 lstm_1 (LSTM)               (None, 8, 5)              320

 lstm_2 (LSTM)               (None, 8, 1)              28

 flatten (Flatten)           (None, 8)                 0

 dense (Dense)               (None, 500)               4500

 dense_1 (Dense)             (None, 250)               125250

 dense_2 (Dense)             (None, 125)               31375

 dense_3 (Dense)             (None, 50)                6300

 dense_4 (Dense)             (None, 5)                 255

=================================================================
Total params: 15008596 (57.25 MB)
Trainable params: 293908 (1.12 MB)
Non-trainable params: 14714688 (56.13 MB)
_________________________________________________________________
```

Fig 4. Proposed LSTM model structure

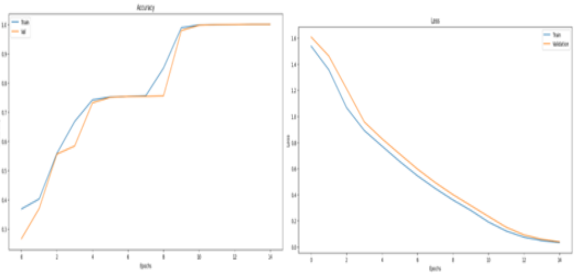
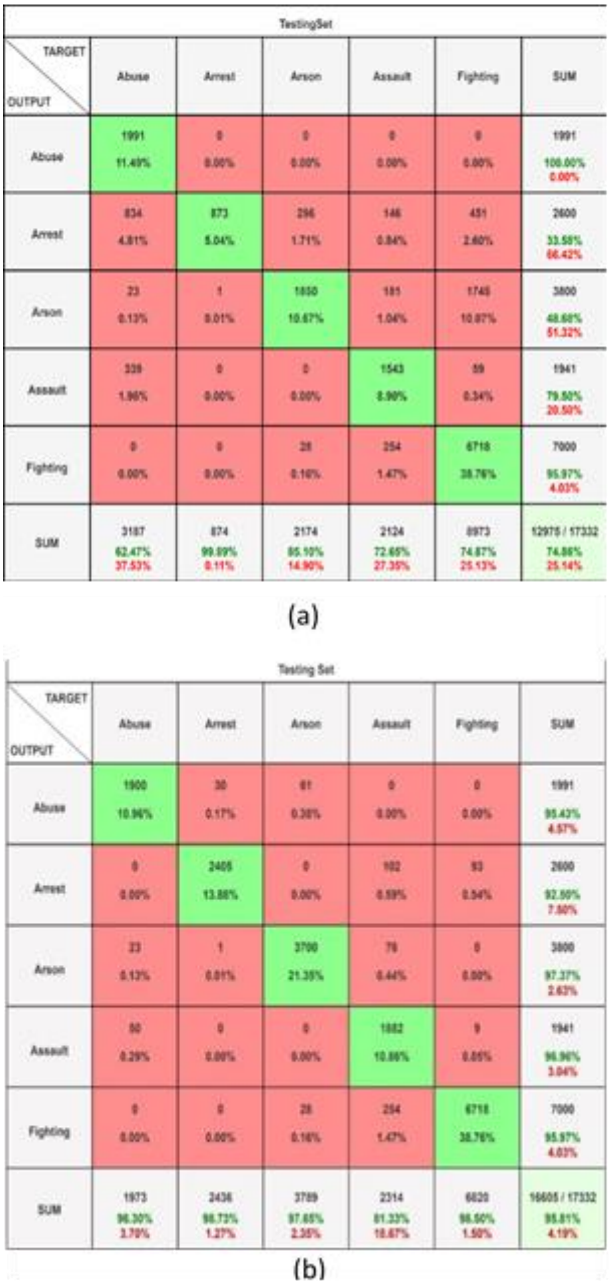


Fig 5. (a) Training and (b) validation – Accuracy and Loss Curves – After HT

Figure 5(a) depicts the training and validation accuracy curves and figure 5(b) depicts the training and validation accuracy curves. The tuned model after running for 13 epochs has achieved 99.91% training accuracy and 99.94% validation accuracy. The training loss obtained is 0.0742% and the validation loss is 0.0920%.

Figure 6(a) is the confusion matrix generated for the testing dataset after the proposed model evaluation. The results show that the model has only generalized the 'abuse' class labels with 100% accuracy. The model also classified the 'Fighting' data set with 95.97% accuracy and a misclassification of only 4.03%. Whereas the model can achieve only 79.50%, 48.68%, and 33.58% classification accuracy for 'Assault', 'Arson', and 'Arrest' class respectively. Finally, the model managed to get only 74.86% testing accuracy with 25.14% misclassification rate.



TestingSet

TARGET / OUTPUT	Abuse	Arrest	Arson	Assault	Fighting	SUM
Abuse	1991 11.49%	0 0.00%	0 0.00%	0 0.00%	0 0.00%	1991 100.00% 0.00%
Arrest	834 4.81%	873 5.04%	296 1.71%	146 0.84%	451 2.60%	2600 33.58% 66.42%
Arson	23 0.13%	1 0.01%	1850 10.67%	181 1.04%	1745 10.07%	3800 48.68% 51.32%
Assault	339 1.96%	0 0.00%	0 0.00%	1543 8.90%	59 0.34%	1941 79.50% 20.50%
Fighting	0 0.00%	0 0.00%	28 0.16%	254 1.47%	6718 38.76%	7000 95.97% 4.03%
SUM	3187 62.47% 37.53%	874 99.89% 0.11%	2174 85.10% 14.90%	2124 72.65% 27.35%	8973 74.87% 25.13%	12975 / 17332 74.86% 25.14%

(a)

Testing Set

TARGET / OUTPUT	Abuse	Arrest	Arson	Assault	Fighting	SUM
Abuse	1900 10.96%	30 0.17%	61 0.35%	0 0.00%	0 0.00%	1991 95.43% 4.57%
Arrest	0 0.00%	2405 13.88%	0 0.00%	102 0.59%	93 0.54%	2600 92.50% 7.50%
Arson	23 0.13%	1 0.01%	3700 21.35%	76 0.44%	0 0.00%	3800 97.37% 2.63%
Assault	50 0.29%	0 0.00%	0 0.00%	1882 10.86%	9 0.05%	1941 96.96% 3.04%
Fighting	0 0.00%	0 0.00%	28 0.16%	254 1.47%	6718 38.76%	7000 95.97% 4.03%
SUM	1973 96.30% 3.70%	2436 98.73% 1.27%	3789 97.65% 2.35%	2314 81.33% 18.67%	6820 98.50% 1.50%	16605 / 17332 95.81% 4.19%

(b)

Fig 6. Confusion Matrix without Normalization at (a) initial stage (b) final stage

Table 1 shows the performance metrics of the model after hyperparameter tuning at an initial stage. The table includes the precision, 1-precision, recall, 1-recall, and f1-score results for each class with an overall classification accuracy and the misclassification rate.

Finally, we have used VGG16 for feature extraction and the same features are passed through the three layers of LSTM each of different output dimensionality. The output of these layers is connected to the fully connected neural network for the classification. The stochastic gradient descent optimizer is used during the training process with a learning rate 0.005 along with categorical_cross_entropy as a loss function. The entire model is evaluated for more epochs and with the balanced datasets. Figure 6(b) is the confusion matrix generated for the testing data set after the proposed model evaluation. The results show that the model generalized the ‘abuse’, ‘arrest’, ‘arson’, ‘assault’, and ‘fighting’ class labels with 95.43%, 92.50%, 97.37%, 96.96%, and 95.97% accuracy respectively. Finally, the model has achieved a high testing accuracy of 95.81% with just 4.19% mis classification rate. Table 2 shows the performance metrics of the model after hyper parameter tuning at an initial stage. The table includes the precision, 1-precision, recall, 1-recall, and f1-score results for each class with an overall classification accuracy and the mis classification rate.

5. Conclusion

A deep feature structure for LSTM was developed to identify temporal features, and a proposed classifier was presented. The proposed framework was able to achieve the highest accuracy in the validation of the UCF101 data set. The proposed framework can reduce the training time and the computational resources required for the model. It also provides a robust and flexible structure that can be used for training. The main advantage of the proposed framework is that it can simulate a human recognition process. It can be used to extract various levels of features from the data and perform deep temporal processing. The experimental outcomes of the proposed framework exhibited competitive performance. Due to the rapid evolution of the field of video analysis and deep learning, it is important to keep up with the latest developments. This will give you valuable insight into the future of LSTM for classification and detection.

6. Discussions and Future Scope

The potential of LSTM networks for identification of abnormal human activities through video classification and detection is immense.

- Developers can enhance the LSTM architecture to better capture the complexity of video sequences' temporal dependencies. This can be done by developing more sophisticated variants or by implementing attention mechanisms that are designed to focus on specific temporal aspects.

- Researchers can develop new methods for detecting anomalous activities in LSTM frameworks. Doing so will enable them to create models that can handle diverse scenarios.

- By integrating information from different sources, such as video data and sensors, researchers can enhance the robustness and accuracy of their models for detecting unusual activities.

- To ensure the practicality of LSTM models in real-world applications, researchers should consider the various factors that affect their scalability, adaptability, and handling dynamic and complex situations.

- To train LSTM models on routine activities and detect anomalous ones, researchers can investigate transfer learning techniques. This can be useful when there is a limited amount of labelled data.

- Developing LSTM models that can learn and adapt over time is important to ensure that the system's capabilities for detecting anomalous activities are continuously improved.

- Researchers should also develop methods that make LSTM models easier to explain and interpret. Doing so will allow end-users to gain a deeper understanding of the model's reasoning. This is especially important in applications that deal with privacy and safety.

- Researchers can also enhance the capabilities of LSTM models by integrating them with the Internet of Things. Doing so will allow them to perform better in detecting unusual activities in diverse environments.

- Researchers must enhance LSTM models' ability to handle ambiguous or noisy situations to do so effectively. This involves implementing methods that aim to filter out irrelevant data and focusing on the signals that are relevant to detecting anomalous activities.

Table 1. Performance metrics of the proposed model at an earlier stage of HT

Class Name	*Precision*	*1-Precision*	*Recall*	*1-Recall*	*F1-Score*	*Accuracy*	*Misclassification Rate*
Abuse	1.0000	0.0000	0.6247	0.3753	0.7690	0.7486	0.2514
Arrest	0.3358	0.6642	0.9989	0.0011	0.5026		
Arson	0.4868	05132	0.8510	0.1490	0.6194		
Assault	0.7950	0.2050	0.7265	0.2735	0.7592		
Fighting	0.9597	0.0403	0.7487	0.2513	0.8412		

Table 2. Performance metrics of the proposed model at a final stage of HT

Class Name	*Precision*	*1-Precision*	*Recall*	*1-Recall*	*F1-Score*	*Accuracy*	*Misclassification Rate*
Abuse	0.9543	0.0453	0.9630	0.0370	0.9586	0.9581	0.0419
Arrest	0.9250	0.0750	0.9873	0.0127	0.9751		
Arson	0.9737	0.0263	0.9765	0.0235	0.9751		
Assault	0.9696	0.0304	0.8133	0.1867	0.8846		
Fighting	0.9597	0.0403	0.9850	0.0150	0.9722		

- Designing systems that can detect anomalous activities should consider the various aspects of human nature, including privacy, cultural sensitivities, and rights. Through collaboration with the developers and end-users, the technology should be socially advantageous and acceptable.

References

[1] Akter, M.; Ansary, S.; Khan, M.A.-M.; Kim, D. Human Activity Recognition Using Attention-Mechanism-Based Deep Learning Feature Combination. Sensors 2023, 23, 5715. https:// doi.org/10.3390/s23125715.

[2] Haodong Duan, Yue Zhao, Yuanjun Xiong, Wentao Liu, and Dahua Lin, "Omni-Sourced Webly-Supervised Learning for Video Recognition," Cham, 2020: Springer International Publishing, in Computer Vision – ECCV 2020, pp. 670- 688.

[3] Xianyuan Wang, Zhenjiang Miao, Ruyi Zhang, and Shanshan Hao, "I3D-LSTM: A New Model for Human Action Recognition," IOP Conference Series: Materials Science and Engineering, vol. 569, no. 3, p. 032035, 2019/07/01 2019, doi: 10.1088/1757-899x/569/3/032035.

[4] Guoxi Huang and Adrian G Bors, "Busy-Quiet Video Disentangling for Video Classification," in Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision, 2022, pp. 1341-1350.

[5] Shreyank N Gowda, Marcus Rohrbach, and Laura Sevilla-Lara, "SMART Frame Selection for Action Recognition," in Proceedings of the AAAI Conference on Artificial Intelligence, 2021, vol. 35, no. 2, pp. 1451-1459.

[6] Shervin Manzuri Shalmani, Fei Chiang, and Rong Zheng, "Efficient Action Recognition Using Confidence Distillation," arXiv preprint arXiv:2109.02137, 2021.

[7] Dhammi, L.; Tewari, P. Classification of Human Activities using data captured through a Smartphone using deep learning techniques. In Proceedings of the 2021 3rd International Conference on Signal Processing and Communication (ICPSC), Coimbatore, India, 13–14 May 2021; pp. 689–694.

[8] Jantawong, P.; Jitpattanakul, A.; Mekruksavanich, S. Enhancement of Human Complex Activity Recognition using Wearable Sensors Data with InceptionTime Network. In Proceedings of the 2021 2nd International Conference on Big Data Analytics and Practices (IBDAP), Bangkok, Thailand, 26–27August 2021; pp. 12–16.

[9] Shi, H.; Hou, Z.; Liang, J.; Lin, E.; Zhong, Z. DSFNet: A Distributed Sensors Fusion Network for Action Recognition. IEEE Sens. J. 2023, 23, 839–848.

[10] Teng, Q.; Tang, Y.; Hu, G. RepHAR: Decoupling Networks with Accuracy-Speed Tradeoff for Sensor-Based Human Activity Recognition. IEEE Trans. Instrum. Meas. 2023, 72, 2505111.

[11] Challa, S.K.; Kumar, A.; Semwal, V.B. A multibranch CNN-BiLSTM model for human activity recognition using wearable sensor data. Vis. Comput. 2022, 38, 4095–4109.