VFAST Transactions on Software Engineering
https://vfast.org/journals/index.php/VTSE@ 2024, ISSN(e): 2309-3978, ISSN(p): 2411-6246
Volume 12, Number 1, January-March 2024 *pp:46-57*

Action Recognition in videos using VGG19 pre-trained based CNN-RNN Deep Learning Model

Fayaz Ahmed Memon [1], **Majid Hussain Memon** [2], **Imtiaz Ali Halepoto** [1*], **Rafia Naz Memon** [1], **Ali Raza Bhangwar** [1]

[1]Software Engineering Department, Quaid-e-Awam University of Engineering, Science & Technology, Nawabshah, Pakistan; [2]Electronic Engineering Department, Quaid-e-Awam University of Engineering, Science & Technology, Nawabshah, Pakistan

Keywords: *Automatic identification, Classification, Human actions, Computer vision, Deep learning, Transfer learning.*

Journal Info:
Submitted:
January 15, 2024
Accepted:
March 24, 2024
Published:
March 26, 2024

Abstract

Automatic identification and classification of human actions is one the important and challenging tasks in the field of computer vision that has appealed many researchers since last two decays. It has wide range of applications such as security and surveillance, sports analysis, video analysis, human computer interaction, health care, autonomous vehicles and robotic. In this paper we developed and trained a VGG19 based CNN-RNN deep learning model using transfer learning for classification or prediction of actions and its performance is evaluated on two public actions datasets; KTH and UCF11. The models achieved significant accuracy on these datasets that are equal to 90% and 88% respectively on KTH and UCF11 which beats some of the accuracy achieved by handcrafted feature based and deep learning based methods on these datasets.

***Correspondence author email address:** halepoto@quest.edu.pk
DOI: 10.21015/vtse.v12i1.1711

1 Introduction

Action recognition refers to the process of automatically detecting and classifying the actions based on visual data in the form of videos or images. The main goal is to train computers so that computers can understand and interpret the human actions and behaviors in videos. Human actions in videos can be predicted or classified successfully with higher accuracy by using deep learning models that have been trained on a dataset which contains several action classes. Unlike, the human's ability to predict and recognize actions, the training of deep learning model for classification of actions is difficult and challenging due to various factors such as temporal variability, viewpoint and scale variations, background clutter, data variability, large scale datasets, fine-grained actions, action context, multi-person scenarios, real time processing and long term dependencies. Moreover, classifying human actions is one of important and fundamental tasks in the field of computer vision with

numerous real world applications such as security and surveillance [1], sports analysis [2], video analysis [3], human computer interaction [4], health care [5], autonomous vehicles [6] and robotics [7]. These applications are the core motivations for this research.

The conventional machine learning based methods depend on manually engineering features, whereas the deep learning models automatically learn hierarchical representations of features directly from raw data which enables better performance in capturing complex patterns and relationships in the data. Various deep learning models have been developed, trained and used by the researchers in literature for the classification or prediction of action videos. The earliest and simplest approach uses Convolution Neural Network (CNN)[8] for the classification of action videos.Using this approach, the action video is classified on frame by frame basis i.e. a video is firstly converted into video frames and then each video frame is predicted by CNN model individually. The frame level predictions are then merged by average or max pooling to make a prediction of a video.This technique of classifying of videos frame by frame is conceptually straightforward, computationally efficient, flexible, easy to implement, and we can get reasonable classification results; however, this technique does not intrinsically capture temporal dependencies between frames which make it difficult to classify the actions that heavily depend on sequence and timing of frames.Therefore most of the techniques for classifying or predicting actions in videos use special kind of Recurrent Neural Network (RNN)[9] network called Long Short-term Memory (LSTM)[10] network to model the long-term dependencies between the video frames.

In technique [11] for human action recognition,a pre-trained CNN model is used that extracts the features from the action dataset followed by SVM-KNN hybrid classifier for classification of actions. This study proved that the features learnt by CNN based on a large scale dataset are successfully transferable for a new task such as action recognition with a small training dataset. The authors assessed their proposed technique on two renowned public action KTH and UCF sports datasets and their results showed significant improvements in terms of accuracy as compared to handcrafted feature based methods.

In order to achieve higher prediction accuracy, a deep learning model known as CNN-RNN model, not only learns visual information in video frames but also it incorporates temporal information between video frames. This model can achieve much better prediction results as compared to CNN model due to its capability to learn motion information between consecutive video frames.The CNN-RNN model can also be used as single CNN-RNN model that combines both CNN and RNN networks within a single architecture.In work [12], CNN-LSTM deep learning architecture is proposed for action recognition. A VGG16 pre-trained based CNN model is firstly trained for extracting the spatial features from input video. Then a LSTM model is trained for classification of input video into a particular class. The performance of this model is evaluated on three public KTH, UCF-11 & HMDB-51 datasets and achieved accuracy equal to 93%, 91% and 47% respectively on these datasets.

In some recent deep learning based approaches [13], [14], [15], [16], the authors have used complex networks such as Conv3D, ConvLSTM and two-stream networks and attention mechanism in their networks for enhancing performance of action recognition tasks. These networks incorporate temporal and other modalities such as optical flow between successive video frames for motion information.The 3D-CNN model for classification of actions was first presented by Tran et al. [17]. The 3D-CNNs are the extension of 2D-CNNs to directly process spatio-temporal data. These networks use 3D convolutions instead 2D convolutions for capturing both spatial and information simultaneously.T. Wang et al. [18] developed a 3DCNN-ConvLSTM deep learning model by modifying CNN-RNN model for classification of human actions. In this work, they used 3D-CNN as a part of CNN for capturing spatial features and ConvLSTM as a part of RNN for temporal features. The performance of this model is assessed on two bench mark datasets and achieved significant accuracy that are equal to 94.8% and 91.2% respectively on KTH and UCF11 datasets.

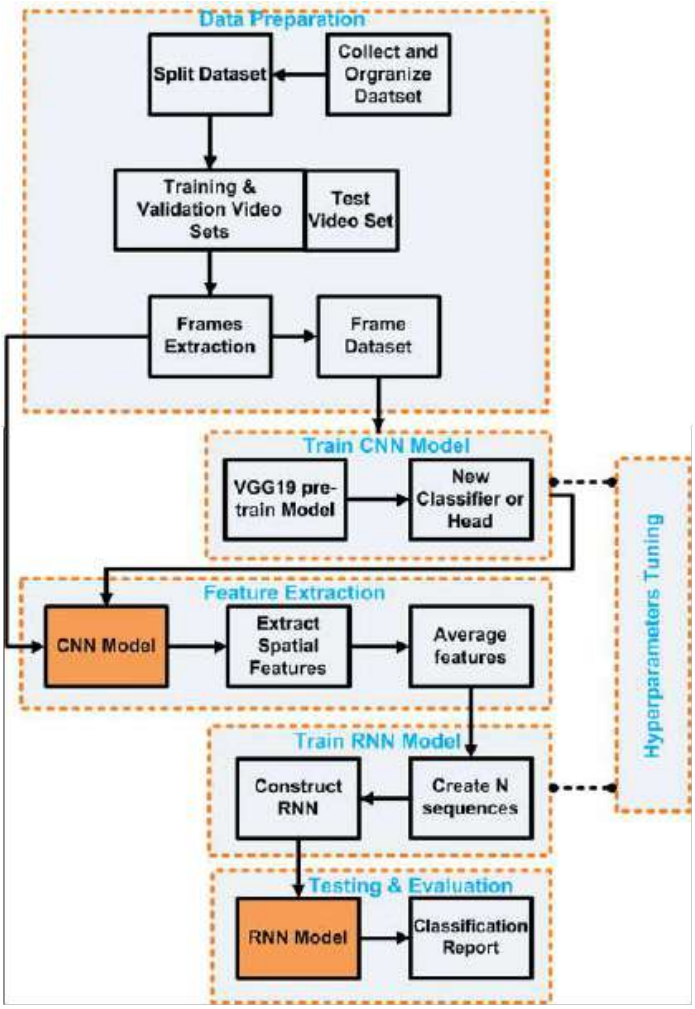

Figure 1. Proposed Methodology

Several kinds of two-stream CNNs [19], [20], [21] are also proposed and developed by the researchers for enhancing the classification accuracy of action recognition systems. A two stream network uses two stream CNNs; one for visual information and other for temporal information. The outputs from these two streams are then merged to make a final prediction. Chakraborty and Mukhopadhyay [22] developed a novel unsupervised heterogeneous recurrent spiking neural (HRSNN) model for the classification of human actions. The performance of this model is evaluated on three public benchmark UCF101, UCF11 KTH action datasets and one event-based DVS128 Gesture dataset and achieved 77.53%, 79.58% and 94.32% accuracy respectively on UCF11, UCF101 and KTH datasets and 96.54% accuracy on even-based DVS Gesture dataset.

In these days deep learning based methods are used for action classification tasks. The CNN-RNN model outclass in capturing spatial and temporal information from video data which enables accurate action recognition. The use of transfer learning by utilization of pre-trained models not only reduces data requirements as well as it improves generalization and can achieve faster training convergence. Some complex networks such as 3D CNNs, ConvLSTM and attention based models are also used for enhancing classification performance. The active research is conducted on addressing challenges such as dataset variability and real time processing that aim to further improve the robustness and efficacy of deep learning methods in action recognition. In this paper, CNN & CNN-RNN deep learning models are developed based on VGG19 [23] pre-trained network using transfer learning. It is a commonly used method for classification and detection tasks [24], [25]. Transfer learning is a technique that allows us to transfer and apply the knowledge gained from one task to another task. The main reason behind using transfer learning is the fact that pre-trained networks are trained on large datasets such as ImageNet [26]. The learned general features by these pre-trained networks are mostly relevant to many tasks, so instead training a CNN model from scratch we can take the advantage of using transfer learning which not only speeds up the training process as well as it reduces computational resources needed for training and improves performance and accuracy [27].Some of most popular pre-trained CNNs exists that are used for transferring knowledge into your own model includes VGG19 [23],ResNet [28], MobileNet [29], GoogleNet [30], Xception [31] and Inception V3 [32]. The main contributions in this paper are summarized as follows:

- CNN-RNN deep learning architecture for classification of actions is described in detail.
- Developed, trained and evaluated the performance of VGG19 based CNN-RNN deep learning models on KTH and UCF11 datasets.
- Compared the performance of developed mod-

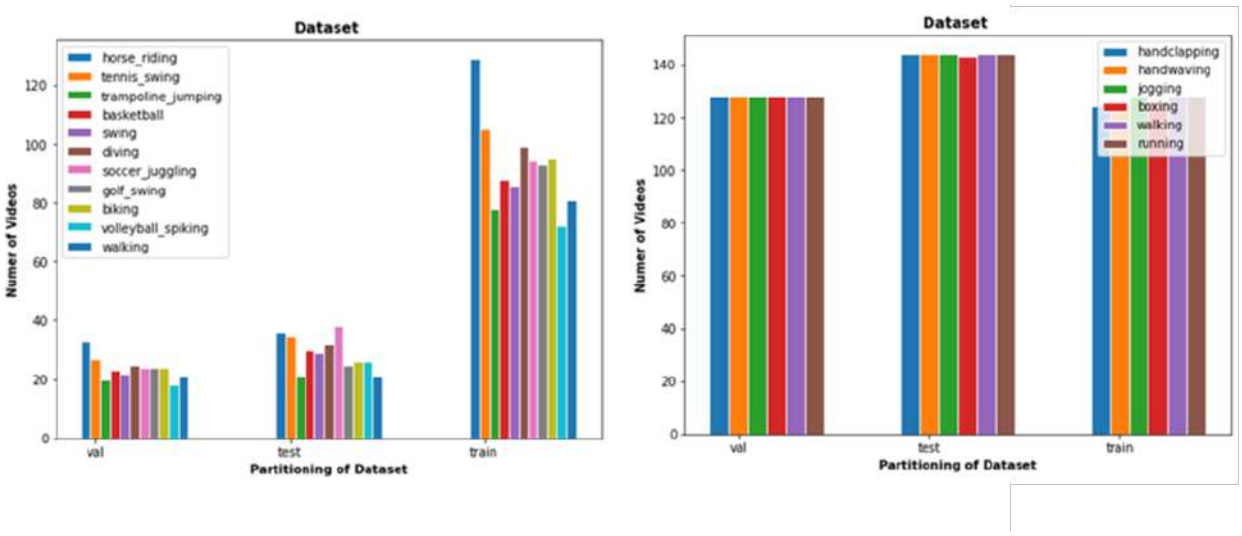

Figure 2. Figure 2: UCF11 & KTH Datasets Split

els with other works.

2 Proposed Methodology

In this paper, we use CNN-RNN architecture for classification of actions.This architecture for classification of actions uses two separately trained CNN & RNN models as shown in Figure 1. The CNN model is used for extraction of spatial features from individual video frames and RNN model is used to capture the temporal dependencies between these frame-level spatial features. This section describes the methodology used for creation of dataset, development of CNN RNN models and assessment of CNN-RNN model.

2.1 Preparation of KTH & UCF11 datasets

The dataset is split into training and testing sets. The training dataset is used to train a model. The part of training dataset called validation set which is used to tune the hyper-parameters and monitor the performance of model during training. The performance of the model is evaluated on the testing dataset to assess its generalization ability to unseen data. In this paper, we trained and assessed the performance of CNN-RNN architecture on two benchmark actions datasets; i.e. UCF11 & KTH. We randomly split UCF11 dataset by considering a ratio of 80% and 20% respectively for train and test sets based on description [33] so that these sets contain videos of different groups. Moreover, 20% of train set is used for validation during training. Similarly, we split KTH dataset into train, test and validation sets according to person-IDs [34] which contain human action videos acted by 8, 8 and 9 different persons respectively. These dataset splits are shown in Figure 2. After splitting KTH and UCF11 datasets into train, test and validation video sets, we obtain frame datasets of these video sets by extraction of frames. We extracted 10 frames at regular intervals for each video in these video sets. The frame datasets are used for training CNN models.

2.2 Training CNN Model

The CNN model is developed by utilizing VGG19 pre-trained network using transfer learning for extraction of frame level features. The model structure is shown in Figure 3. This pre-trained based CNN model is usually created by firstly selecting a pre-trained network as the base model. Then custom layers (i.e. a new classifier) on top of this base model are added to adopt it for your own task. The model is trained on training dataset of videos frames to learn the spatial features. This training set of video frames is obtained by converting each of the videos in each of classes of the training video dataset in to video frames. When framing each video of training set, we can set the value of k where k is the number of frames in video. On the basis of k, the

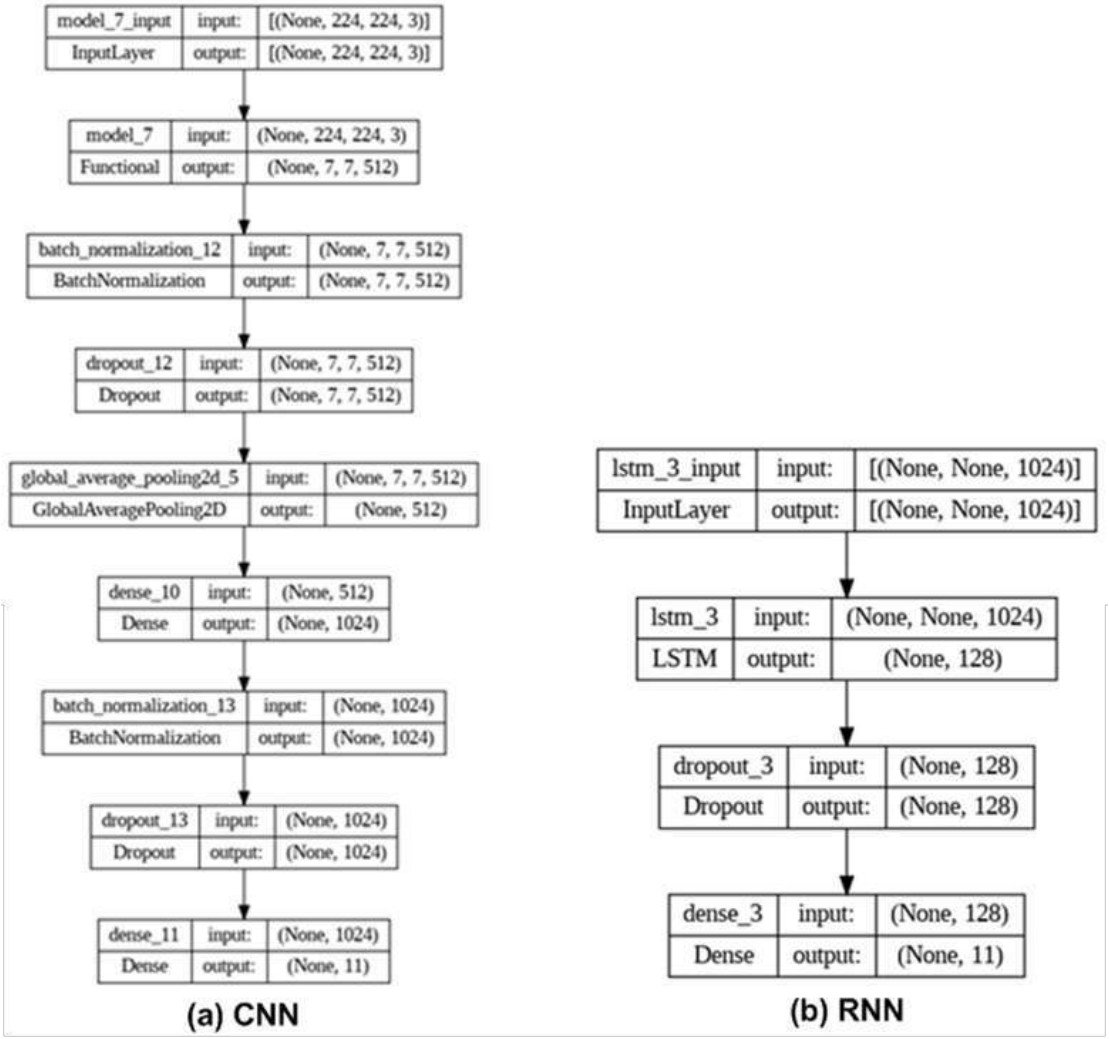

Figure 3. Model Structures

uniform gap between frames i.e. h is to be calculated and k frames with a gap of h will be extracted for each video.

2.3 Training RNN Model

We develop RNN model based on a single LSTM layer, a dense and a dropout layer. The structure of this model is shown in Figure 3. The RNN model isthen developed by training RNN network on sequences of n (where n is number of frames in video) spatial features to learn the temporal features between video frames. Each of these sequences is obtained by firstly converting video into n frames and passed to CNN model to extract spatial features. These spatial features are then combined to create a sequence of n features. In this way T (where T is total number of sequences) such sequences of n spatial features are created for training RNN model.

2.4 Testing and Evaluation of CNN-RNN Model

The performance of CNN-RNN deep learning model is assessed on several performance metrics such as accuracy, precision, recall and F1-score values and PR ROC plots. The performance of CNN-RNN model can further be improved by identifying key frames instead extracting k frames with a gap of h in videos. The extraction of key frames removes the redundancy between successive videos frames particular when training the CNN model and improves the prediction

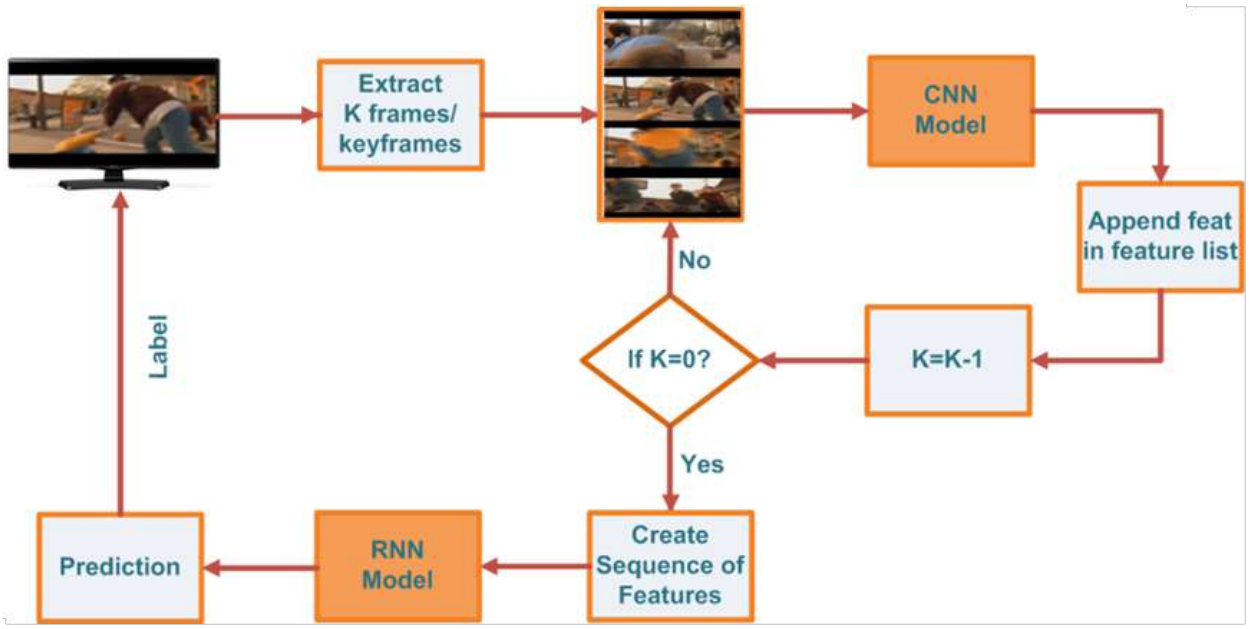

Figure 4. Prediction of an action video

accuracy. The proposed methodology for classification of action videos using CNN-RNN deep learning architecture is presented in Figure 4. An input video using this deep learning architecture is classified by firstly converting a video into in to k frames or key frames. Each frame is then individually fed through CNN model to obtain a feature vector that encodes spatial information. Finally sequence of these feature vectors is generated and directed to the RNN model to classify or predict a video.

2.5 Hyperparameter Tuning

We experiment several hyperparameters such as batch size, learning rate, dropout and the specific parameters related to CNN and RNN architectures to optimize the performance of CNN-RNN model. The hyperparameter values where the model achieves the optimal performance are selected and are used for training CNN and RNN models.We conducted experiments in python language usingGoogle Colab platform which provides us free GPU for training models. We use Adam optimization with default learning rate, a batch size of 32, a dropout equal to 0.2 and 50 epochs for training CNN model and 200 epochs for training RNN model.

3 Results and Discussions

This section presents experiments results of CNN-RNN model on KTH and UCF11 datasets. We use various performance metrics such as accuracy, confusion & normalized confusion matrices, precision, recall and F1score and Precision-Recall & Roc Curves to assess the performance of CNN-RNN model.

3.1 Training Graphs

Various benchmark datasets such as UCF101, HMDB-51 and YouTube-8M, were used to evaluate the performance of CNN-RNN model. The reported accuracy is achieved on these benchmark datasets can vary depending on factors such as model architecture, training procedure, hyper-parameters, and the complexity of the dataset. The model accuracy and loss of CNN-RNN model on KTH & UCF11 datasets are described by learning curves shown in Figure 5 and Figure 6. These curves show how well a deep learning model is learning from data and help us to identify under fitting and over fitting issues. From Figure 5 and 6, it can be seen that the CNN-RNN models achieved 100% training accuracy which means that the models have successfully learned the all training data and can make predictions perfectly on training data. However

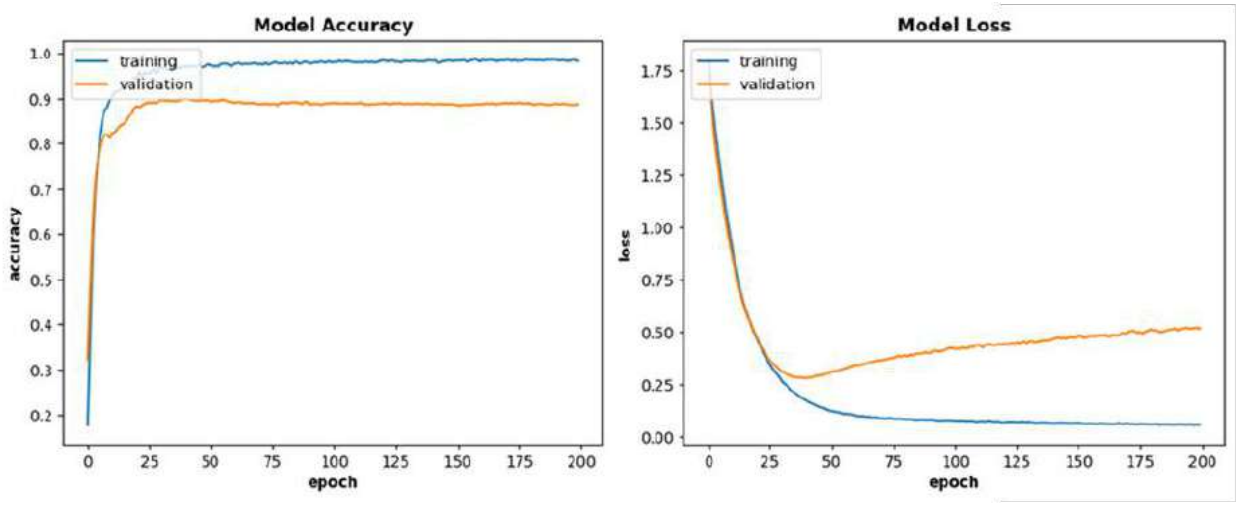

Figure 5. Model Accuracy & Loss of CNN-RNN model on KTH Dataset

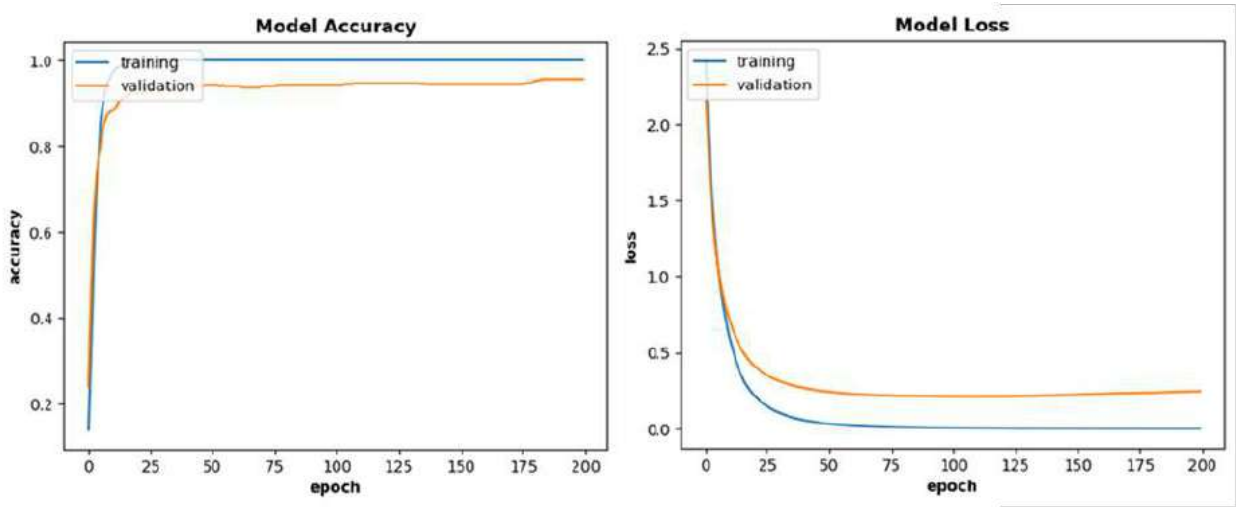

Figure 6. Model Accuracy & Loss of CNN-RNN model on UCF11 Dataset

there may a chance of over fitting of data. The CNN-RNN model achieved near 90% validation accuracy on KTH dataset and 95% validation accuracy on UCF11 dataset.

3.2 Confusion and Normalized Confusion Matrices

The performance of CNN-RNN model on test sets of KTH and UCF11 datasets is described using confusion and normalized confusion matrices as shown in Figure 7 and Figure 8. The confusion matrix allows us to see the performance of a deep learning model in terms of number of correctly and incorrectly classifying samples of each class while a normalized confusion matrix provide a clear picture of how well a deep learning model is performing for each class with regards to class imbalances particularly for multi-classification tasks. On KTH dataset as shown in Figure 7, the model achieved more than 90% accuracy on each class except Jogging for which 30% samples are miss-classified as Running due to resemblance between these two classes. The overall accuracy achieved on this dataset by the model is around 90%. The accuracy achieved on each class of UCF11 dataset by the CNN-RNN model are shown in Figure 8. On this dataset the

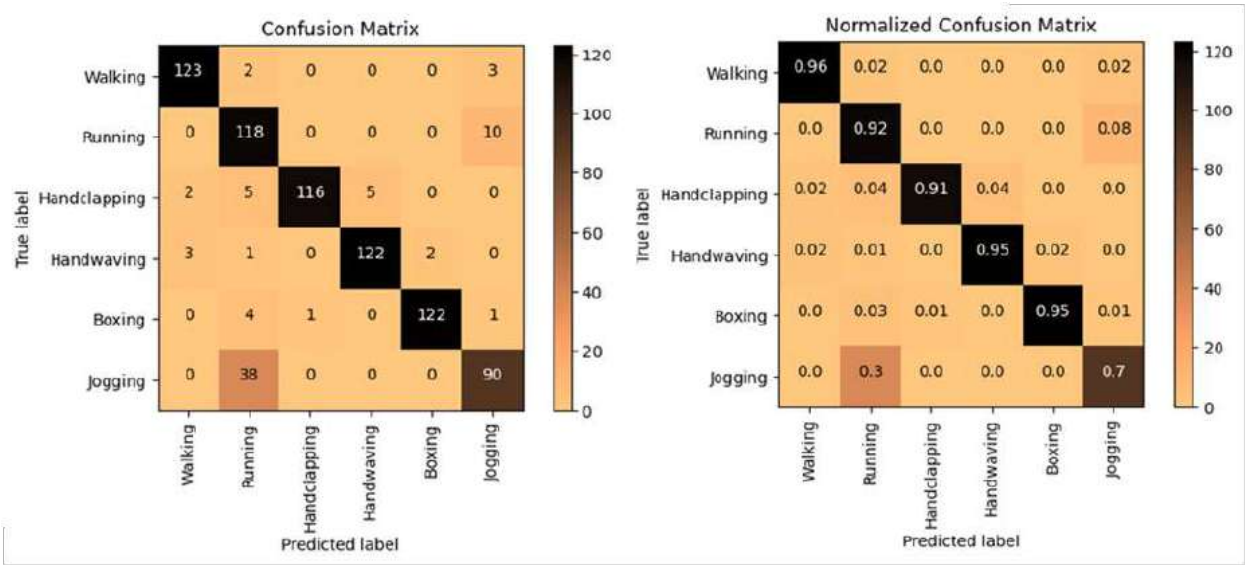

Figure 7. Confusion and Normalized Confusion Matrices for KTH Dataset

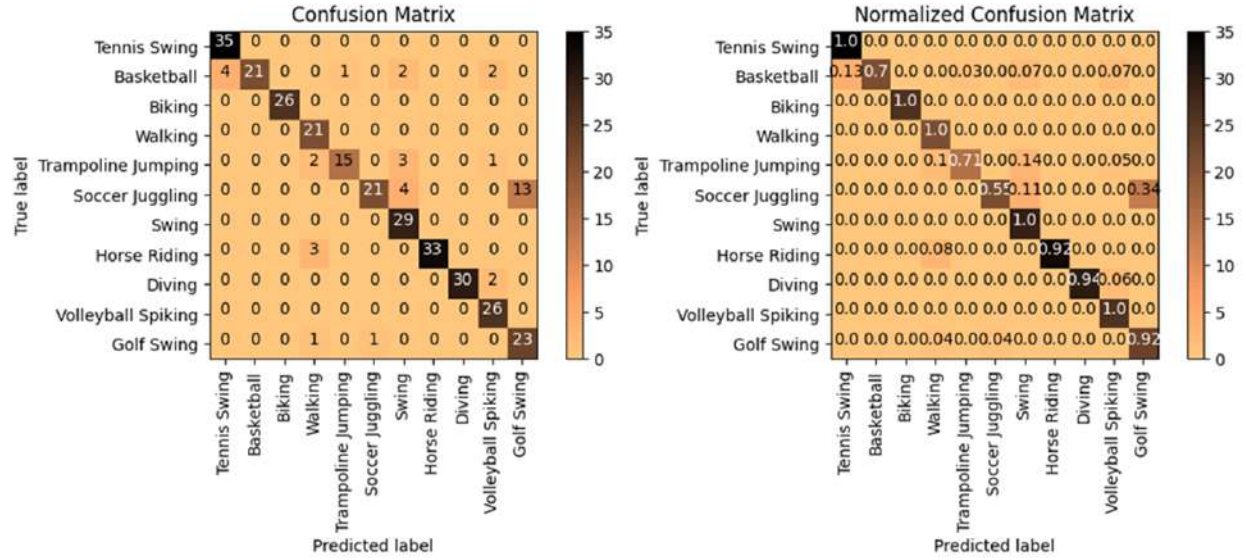

Figure 8. Confusion and Normalized Confusion Matrices for UCF11 Dataset

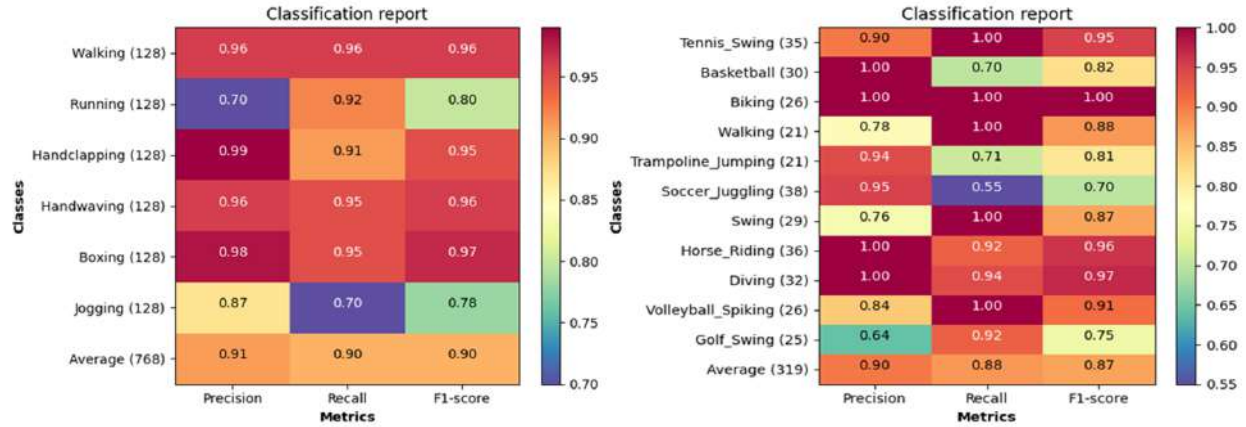

Figure 9. Classification Reports for KTH and UCF11 Datasets

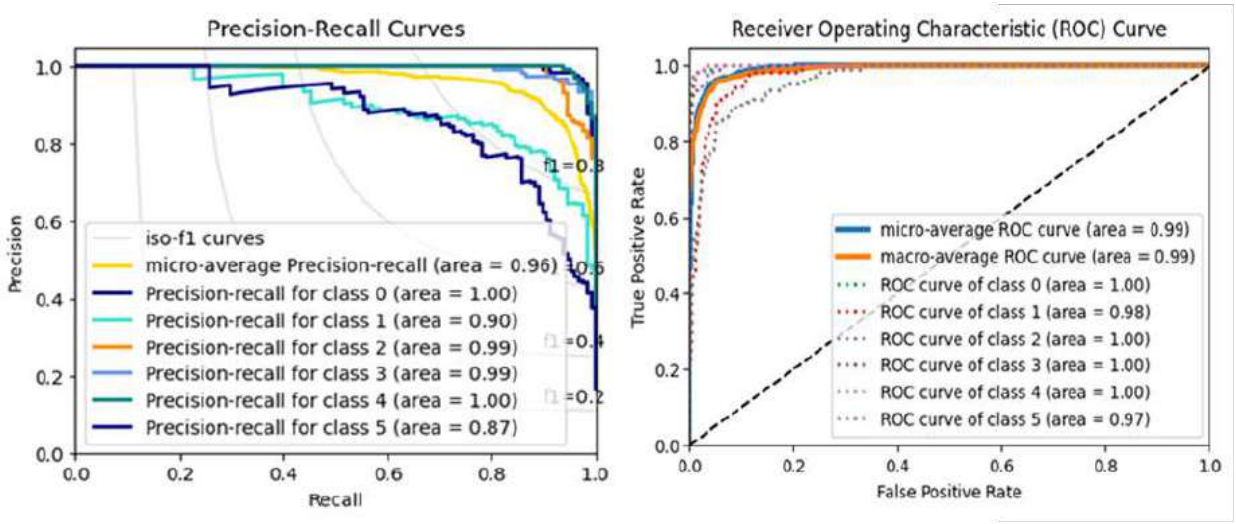

Figure 10. PR & ROC Plots on KTH Dataset

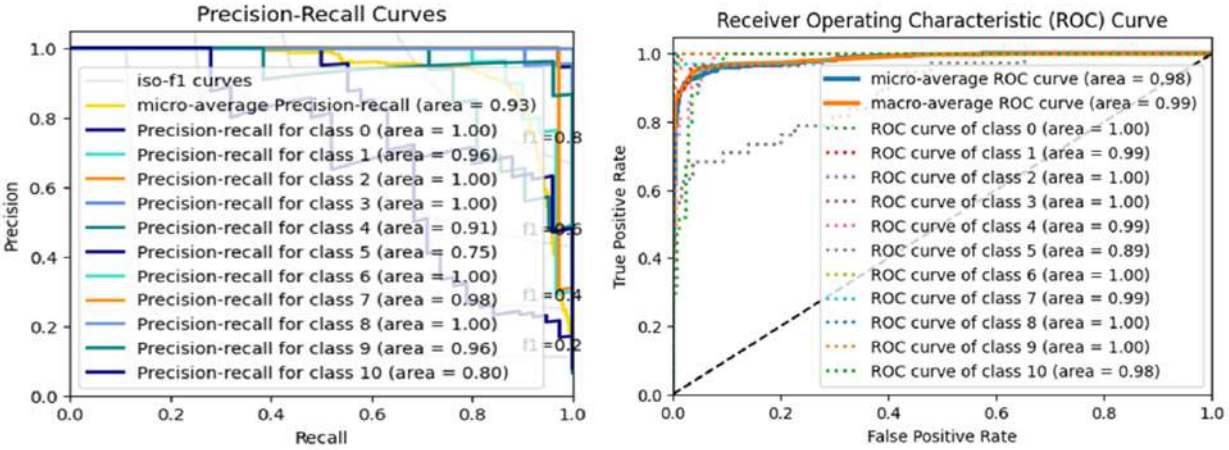

Figure 11. PR & ROC Plots on UCF11 Dataset

model achieved overall accuracy near to 88%. The accuracy are significant and are achieved on these dataset without using any optical flow between video frames for motion information.

3.3 Classification Reports

The classification reports of KTH and UCF11 datasets are shown in Figure. These reports shows the precision, recall and F1score values on each class and their mean values. The precision shows the capability of a model of finding correct samples, recall shows the capability of a model of finding true or positive samples and F1score is weighted harmonic mean of precision and recall values. The model achieved 90% and 88% F1scores respectively on KTH and UCF11 datasets.

3.4 PR & ROC Curves

We also measure the model's capability of making accurate positive predictions and distinguishing between the positive and negative classes of CNN-RNN model on KTH & UCF11 datasets using PR and ROC plots as shown in Figure 10 and Figure 11. A PR plot is usually used when we are dealing with imbalanced datasets and our emphasis is on positive class prediction, while ROC plot is normally used for balanced datasets and

our interest is trade-off between true positive rate and false positive rate. From Figure 10 & Figure 11, it can be seen that PR curves are closer to the top right corner and ROC curves are closer to the top left corner and larger AUC values in ROC curves indicate the better performance of CNN-RNN model on KTH and UCF11 datasets.

3.5 Comparative Analysis of CNN-RNN Architecture with Other Works on UCF11 & KTH Datasets

Table 1, shows comparative analysis of presented work in this paper and some other works by the researchers on KTH and UCF11 datasets. We employed a simple CNN-RNN model that does not use any modality for additional motion information which can be trained with limited GPU and memory resources. Using this deep learning architecture, we achieved 90% and 88% accuracies respectively on KTH and UCF11 datasets as shown in Table 1.

4 Conclusions

In this paper, CNN-RNN deep learning architecture is used for classification of action videos. The methodology for developing, training and classifying actions using this architecture is described in detail and the performance of this deep learning architecture is evaluated on two benchmark KTH and UCF11 action datasets. Substantial accuracy have been achieved using CNN-RNN architecture on these two datasets that beats some of the accuracy achieved by other works on these datasets using handcrafted features based and deep learning based methods. The overall accuracy achieved by the model on KTH dataset is not more than 90% because this dataset consists of similar type of actions which makes difficult for a model to predict or classify the actions in this class with higher accuracy without having additional motion information. In future our aim is to utilize complex deep learning models and use optical flow information between video frames for motion information in-order to enhance the accuracy on these datasets. Moreover, we aim to collect more data relevant to KTH dataset and combine with KTH dataset for increasing generalization ability and improving the classification accuracy of the model.

Table 1. Comparison of the proposed model with the existing models for UCF1 & KTH Datasets

Authors	Year	UCF11(%)	KTH (%)
(Grushin et al)[35]	2013	Item 3	90.70
(Figueiredo et. al) [36]	2014	59.50	87.70
Hasan et. al.) [37]	2014	54.50	90.00
(Maia et. al.)[38]	2015	64.00	91.50
(Figueiredo et al.) [39]	2016	65.80	91.40
(J. Arunnehru et. al.) [40]	2018	—	94.90
(Orozco, et al.)[12]	2020	90.00	93.00
T. Wang et. al.) [18]	2021	91.20	94.80
(Chakraborty et al.) [22]	2023	79.58	94.32
In the paper	—	88.00	90.00

Author Contributions

Fayaz: Conceptualization, Methodology, Software **Majid**: Data curation, Writing- Original draft preparation. **Imtiaz**: Visualization, Investigation. **Rafia**: Supervision and validation. **Ali Raza**: Software, Validation, Reviewing and Editing

Compliance with Ethical Standards

It is declare that all authors don't have any conflict of interest. It is also declare that this article does not contain any studies with human participants or animals performed by any of the authors. Furthermore, informed consent was obtained from all individual participants included in the study.

References

[1] M. Zahrawi and K. Shaalan, "Improving video surveillance systems in banks using deep learning techniques," Sci Rep, vol. 13, no. 1, Art.no. 1, May 2023.

[2] M. M. Afsaret al., "Body-Worn Sensors for Recognizing Physical Sports Activities in Exergaming via Deep Learning Model," IEEE Access, vol. 11, pp. 12460–12473, 2023.

[3] L. Romeo, R. Marani, T. D'Orazio, and G. Cicirelli, "Video Based Mobility Monitoring of Elderly People Using Deep Learning Models," IEEE Access, vol. 11, pp. 2804–2819, 2023.

[4] W. Alsabhan, "Human–Computer Interaction with a Real-Time Speech Emotion Recognition with Ensembling Techniques 1D Convolution Neural Network and Attention," Sensors, vol. 23, no. 3, p. 1386, Jan. 2023.

[5] N. D. Kathamuthuet al., "A deep transfer learning-based convolution neural network model for COVID-19 detection using computed tomography scan images for medical applications," Advances in Engineering Software, vol. 175, p. 103317, Jan. 2023.

[6] J. D. Choi and M. Y. Kim, "A sensor fusion system with thermal infrared camera and LiDAR for autonomous vehicles and deep learning based object detection," ICT Express, vol. 9, no. 2, pp. 222–227, Apr. 2023.

[7] K. You, C. Zhou, and L. Ding, "Deep learning technology for construction machinery and robotics," Automation in Construction, vol. 150, p. 104852, Jun. 2023.

[8] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner, "Gradient-based learning applied to document recognition," undefined. Accessed: Jan. 29, 2021.

[9] D. E. Rumelhart, G. E. Hinton, and R. J. Williams, "Learning representations by back-propagating errors," Nature, vol. 323, pp. 533–536.

[10] Hochreiter, S. and Schmidhuber, J., 1997. Long short-term memory. Neural computation, 9(8), pp.1735-1780.

[11] A. B. Sargano, X. Wang, P. Angelov, and Z. Habib, "Human action recognition using transfer learning with deep representations," in 2017 International Joint Conference on Neural Networks (IJCNN), May 2017, pp. 463–469.

[12] Orozco, C.I., Xamena, E., Buemi, M.E. and Berlles, J.J., 2020. Human action recognition in videos using a robust cnn lstm approach. Ciencia y Tecnología, pp.23-36.

[13] R. Vrskova, R. Hudec, P. Kamencay, and P. Sykora, "Human Activity Classification Using the 3DCNN Architecture," Applied Sciences, vol. 12, no. 2, p. 931, Jan. 2022.

[14] K. J. Naik and A. Soni, "Video Classification Using 3D Convolutional Neural Network," in Advancements in Security and Privacy Initiatives for Multimedia Images, IGI Global, 2021, pp. 1–18.

[15] R. Singh, S. Saurav, T. Kumar, R. Saini, A. Vohra, and S. Singh, "Facial expression recognition in videos using hybrid CNN ConvLSTM," Int. j. inf. tecnol., vol. 15, no. 4, pp. 1819–1830, Apr. 2023.

[16] C. Dai, X. Liu, and J. Lai, "Human action recognition using two-stream attention based LSTM networks," Applied Soft Computing, vol. 86, p. 105820, Jan. 2020.

[17] Tran, Du, Lubomir Bourdev, Rob Fergus, Lorenzo Torresani, and Manohar Paluri. "Learning spatiotemporal features with 3d convolutional networks." In Proceedings of the IEEE international conference on computer vision, pp. 4489-4497. 2015.

[18] T. Wang, J. Li, M. Zhang, A. Zhu, H. Snoussi, and C. Choi, "An enhanced 3DCNN-ConvLSTM for spatiotemporal multimedia data analysis," Concurrency and Computation: Practice and Experience, vol. 33, no. 2, p. e5302, 2021.

[19] Diba, A., Fayyaz, M., Sharma, V., Karami, A.H., Arzani, M.M., Yousefzadeh, R. and Van Gool, L., 2017. Temporal 3d convnets: New architecture and transfer learning for video classification. arXiv preprint arXiv:1711.08200.

[20] L. Wang et al., "Temporal Segment Networks: Towards Good Practices for Deep Action Recognition," in Computer Vision – ECCV 2016, B. Leibe, J. Matas, N. Sebe, and M. Welling, Eds., in Lecture Notes in Computer Science. Cham: Springer International Publishing, 2016, pp. 20–36.

[21] J. Carreira and A. Zisserman, "Quo Vadis, Action Recognition? A New Model and the Kinetics Dataset," in 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Honolulu, HI: IEEE, Jul. 2017, pp. 4724–4733.

[22] B. Chakraborty and S. Mukhopadhyay, "Heterogeneous recurrent spiking neural network for spatio-temporal classification," Frontiers in Neuroscience, vol. 17, 2023, Accessed: Sep. 22, 2023.

[23] K. Simonyan and A. Zisserman, “Very Deep Convolutional Networks for Large-Scale Image Recognition,” arXiv:1409.1556 [cs], Apr. 2015, Accessed: Dec. 08, 2020. [Online]. Available: http://arxiv.org/abs/1409.1556

[24] S. Ismail, B. Ismail, I. Siddiqi, and U. Akram, “PCG classification through spectrogram using transfer learning,” Biomedical Signal Processing and Control, vol. 79, p. 104075, Jan. 2023.

[25] M. Zinnen, P. Madhu, P. Bell, A. Maier, and V. Christlein, “Transfer Learning for Olfactory Object Detection.” arXiv, Jan. 24, 2023.

[26] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, “ImageNet: A large-scale hierarchical image database,” in 2009 IEEE Conference on Computer Vision and Pattern Recognition, Jun. 2009, pp. 248–255.

[27] S. Khan, N. Islam, Z. Jan, I. Ud Din, and J. J. P. C. Rodrigues, “A novel deep learning based framework for the detection and classification of breast cancer using transfer learning,” Pattern Recognition Letters, vol. 125, pp. 1–6, Jul. 2019.

[28] Shafiq, Muhammad, and Zhaoquan Gu. "Deep residual learning for image recognition: A survey." Applied Sciences 12, no. 18 (2022): 8972.

[29] Haase, Daniel, and Manuel Amthor. "Rethinking depthwise separable convolutions: How intra-kernel correlations lead to improved mobilenets." In Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, 2020, pp. 14600-14609.

[30] Sunkara, Raja, and Tie Luo. "No more strided convolutions or pooling: A new CNN building block for low-resolution images and small objects." In Joint European Conference on Machine Learning and Knowledge Discovery in Databases, 2022, pp. 443-459. Cham: Springer Nature Switzerland.

[31] McNeely-White, David, J. Ross Beveridge, and Bruce A. Draper. "Inception and ResNet features are (almost) equivalent." Cognitive Systems Research 59 (2020): 312-318.

[32] Ding, Xiaohan, Xiangyu Zhang, Jungong Han, and Guiguang Ding. "Diverse branch block: Building a convolution as an inception-like unit." In Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, 2021, pp. 10886-10895.

[33] J. Liu, J. Luo, and M. Shah, “Recognizing realistic actions from videos ‘in the wild,’” in 2009 IEEE Conference on Computer Vision and Pattern Recognition, Jun. 2009, pp. 1996–2003.

[34] C. Schuldt, I. Laptev, and B. Caputo, “Recognizing human actions: a local SVM approach,” in Proceedings of the 17th International Conference on Pattern Recognition, 2004. ICPR 2004., Cambridge, UK: IEEE, 2004, pp. 32-36 Vol.3.

[35] A. Grushin, D. D. Monner, J. A. Reggia, and A. Mishra, “Robust human action recognition via long short-term memory,” in The 2013 International Joint Conference on Neural Networks (IJCNN), Aug. 2013, pp. 1–8.

[36] A. M. O. Figueiredo, H. A. Maia, F. L. M. Oliveira, V. F. Mota, and M. B. Vieira, “A Video Tensor Self-descriptor Based on Block Matching,” in Computational Science and Its Applications – ICCSA 2014, B. Murgante, S. Misra, A. M. A. C. Rocha, C. Torre, J. G. Rocha, M. I. Falcão, D. Taniar, B. O. Apduhan, and O. Gervasi, Eds., in Lecture Notes in Computer Science. Cham: Springer International Publishing, 2014, pp. 401–414.

[37] M. Hasan and A. K. Roy-Chowdhury, “Incremental Activity Modeling and Recognition in Streaming Videos,” presented at the Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2014, pp. 796–803.

[38] H. A. Maia, A. M. D. O. Figueiredo, F. L. M. D. Oliveira, V. F. Mota, and M. B. Vieira, “A VIDEO TENSOR SELF-DESCRIPTOR BASED ON VARIABLE SIZE BLOCK MATCHING,” Journal of Mobile Multimedia, pp. 090–102, Aug. 2015.

[39] A. M. de Oliveira Figueiredo, M. Caniato, V. F. Mota, R. L. de Souza Silva, and M. B. Vieira, “A Video Self-descriptor Based on Sparse Trajectory Clustering,” in Computational Science and Its Applications – ICCSA 2016, O. Gervasi, B. Murgante, S. Misra, A. M. A. C. Rocha, C. M. Torre, D. Taniar, B. O. Apduhan, E. Stankova, and S. Wang, Eds., in Lecture Notes in Computer Science. Cham: Springer International Publishing, 2016, pp. 571–583.

[40] J. Arunnehru, G. Chamundeeswari, and S. P. Bharathi, “Human Action Recognition using 3D Convolutional Neural Networks with 3D Motion Cuboids in Surveillance Videos,” Procedia Computer Science, vol. 133, pp. 471–477, Jan. 2018.