A LRCN Approach to Human Activity Recognition

Vishnu Khokhar
Computer Science Department
Patiala, India
vishnukhokhar2412@gmail.com

***Abstract*—Human action recognition is challenging problem in Computer Vision that has received a lot attention in the previous few years. With a lot of new learning techniques such as Convolutional Neural Network (CNN) and Long Short Term Memory (LSTM), the recognition problem has been become very easy. . In this paper, we propose a holistic deep learning-based activity recognition architecture, a Long-Term Recurrent Convolutional Networks (LRCN) which is a mixture of CNN and LSTM. This LRCN approach not only improves the predictive accuracy of Human activities from raw data but also reduces the complexity of the model while eliminating the need for advanced feature engineering. Our proposed model has achieved a 78.67% accuracy on the KTH and Kaggle dataset. This electronic document is a "live" template and already defines the components of your paper [title, text, heads, etc.] in its style sheet. **CRITICAL: Do Not Use Symbols, Special Characters, Footnotes, or Math in Paper Title or Abstract*. (*Abstract*)**

***Keywords*—CNN, LSTM, LRCN, KTH Dataset**, *Kaggle Dataset*

I. Introduction

Human Activity Recognition is complex problem of Computer Vision and is worked upon a lot of places in the world. The evolution of Deep Leaning in the field of Human Activity Recognition has made recognizing human behaviour very trivial. Machine (ML) and deep (DL) learning models are readily made available by frameworks like TensorFlow, PyTorch, Scikit, and others not to mention the Keras API [2] that has made it easy to build and experiment with model. The main task of Human Action Recognition is to differentiate between different actions in video clips captured by the CCTV In other words, it is a problem of predicting the actions in a video for a short-term period. On Deep learning basis the field hasn't received much attention in Human Action recognition side. The action recognition system first detects intrinsic patterns of a captured video sequence consisting of multiple frames and then predicts an activity or action of interest on a spatio-temporal axis. With Deep learning, it is now much simpler and fast to predict activities. Some of the existing machine learning algorithms that have become near obsolete include the support vector machine (SVM) used in the histogram of gradients (HOG) feature extraction with a k-nearest neighbour classifier [5] that have been proposed in the past. Human action recognition is of two categories: hand-crafted learning-based methods and deep learning-based methods. With the new deep learning techniques, it has really become easy to increase accuracy and efficiency of the model. In this paper I have proposed a LRCN Model to predict human actions. In 2016 a group of authors suggested end-to-end trainable class of architectures for visual recognition and description. The main idea behind LRCN is to use a combination of CNNs to learn visual features from video frames and LSTMs to transform a sequence of image embeddings into a class label, sentence, probabilities, or whatever you need. Thus, raw visual input is processed with a CNN, whose outputs are fed into a stack of recurrent sequence models. Even though CNN and LSTM have been extensively studied in the past, they have been studied in isolation. My paper tends to leverage the strengths of combining the models as far as human activity recognition is concerned. The proposed hybrid architecture is designed to model rich multi-domain features by increasing the contextual correlation of spatio-temporal cues and exploiting synergies and computational sharing among processing modules Section III discusses our method and its implementation. Experimental results show that our hybrid network outperforms recent state-of-the-art models on KTH dataset

II. Related Work

There has been a lot research in the field of Human activity recognition using multiple approaches. In this section, I have set a foundation for deep learning based Human Activity Recognition while exploring some of the previous work that relates to our proposed approach and how the works differ from our approach.

A. Handcrafted method

For decades, human activity recognition has been one of the challenging tasks in computer vision and has made a lot of progress in the last few years. Human action recognition consists of interpreting human actions and assigning consistent labels with similar appearance properties to each action. First, a video dataset is captured from an image sensor and pre-processed to create a background model. Next, local and global descriptors can be used to extract low-level features. Finally, a shallow classifier is employed to learn the extracted features and classify potential actions. Most of the methods depend on the modelling of the temporal structure so they can identify frame-level action patterns in each sequence. To perform the final classification of the extracted features, conventional machine learning algorithms such as decision trees [13], support vector machines (SVM) [14], Naive Bayes [15], K-nearest neighbours (KNN) [16], etc., have been mainly used in the literature. There has been effectiveness of such experimental analysis of algorithm but they rely on very complex and time-consuming scheme.

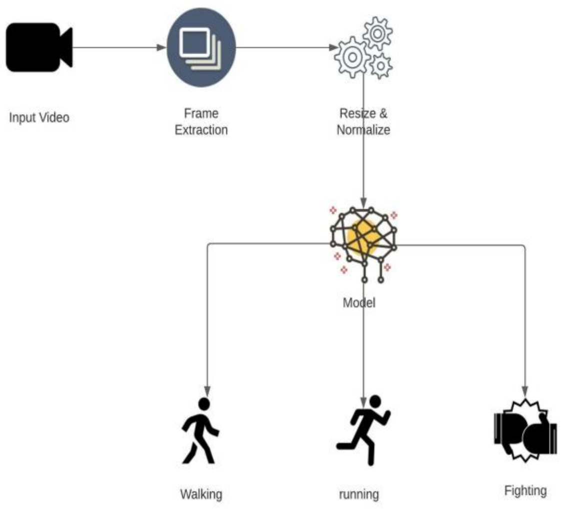
confirm that you have the correct template for your paper size. This template has been tailored for output on the A4 paper size. If you are using US letter-sized paper, please close this file and download the Microsoft Word, Letter file.

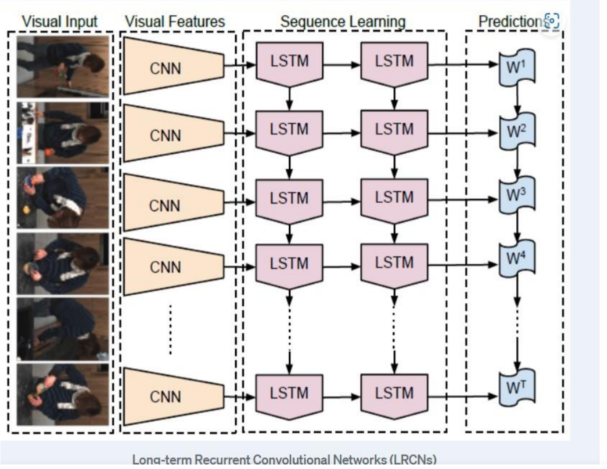
B. Deep learning methods for human activity recognition

Traditional handcrafted-based action recognition methods are time-consuming and labor-intensive and perform poorly on complex scenes and challenging scenarios, such as occlusions, background blur, visual appearance variations, etc., making human behavior recognition analysis extremely difficult. For this reason, the computer vision community has turned to deep learning algorithms to address most of these limitations. If hardware infrastructure and the availability and massive dataset are given, deep learning provides the superior performance in most computer vision task. Hammerla et al. [1] go ahead to suggest that approaches that can exploit the temporal dependencies in time-series data appear as the natural choice for modelling human movement captured with sensor data. Deep recurrent networks, most notably those that rely on Long Short-Term Memory cells (LSTMs) [12], have recently achieved impressive performance across a variety of scenarios. This temporal characteristic of the LSTM architecture and its long-term dependences make it a solid candidate to extract temporal features from our signal. Our work differs from [1] in a way that while Hammerla et al. [1] were comparing the performance of individual networks for HAR, our work seeks to leverage the combined power of both networks.

III. Method and models

The proposed model architecture is composed of action recognition task of three main components: Computational Neural Network (CNN), Long Short-Term Memory (LSTM), and a Long-term Recurrent Convolutional Networks (LRCN) Fig. 1 shows the architectural level of the proposed model. First, frame-level discriminative and informative features are extracted by Computational Neural Network (CNN) as the first pass of recognition. An LSTM network is then used to further model the motion information obtained from the previous representation, helping to capture hidden correlations and long-range dependencies throughout the data sequence. Finally, to further make the system better I have integrated a LRCN model that detect the most relevant features of the object of interest. One of the most widely used computer vision model is CNN, showing impressive results in feature extractionThese systems typically focus on CNNs due to their computational efficiency and accuracy. The temporal classification problem which is a task of recognition of action from the videosis highly dependent on the spatial, temporal, and contextual information inherent in the data stream sequence.





In out experiment I have used KTH and Kaggle Dataset. The Kth dataset is a 2-activity dataset containing walking and running data and Kaggle dataset is a single activity dataset containing fighting data. The datasets are split randomly into test and training data. The train data is 75% of the data and the test data is 25% of the data. The datasets contain the videoclips extracted from the CCTVs. The KTH dataset is a standard dataset which has collection of sequences representing 6 actions and each action class has got 100 sequences. Each sequence has got almost 600 frames and the video is shot at 25 fps. The model is trained on this dataset only with 45 sequences of each class due to hardware Limitations for normal behaviour. (running and walking). Kaggle Dataset, over 100 videos taken from movies and YouTube videos can be used for training suspicious behaviour (fighting). Only 45 of the 100 videos are taken due to hardware limitation.

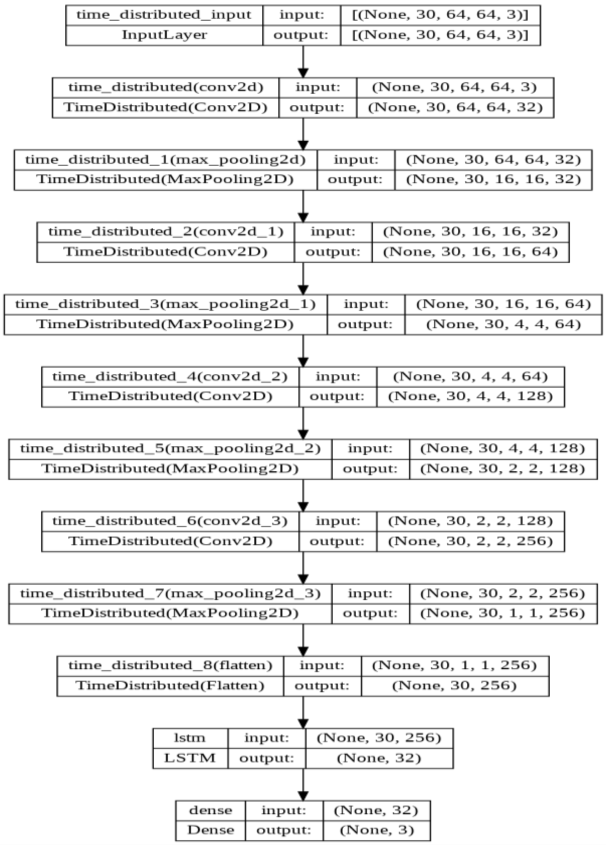
The data pre-processing is done by splitting the video clips into frame to make one sequence. Each Video is read using OpenCV Library, Only 30 frames at equal time intervals are read to form a sequence of 30 frames. The resizing of the frame is done as it is necessary y when we need to increase or decrease the total number of pixels. So, we resized all the frames to width: 224px and height: 224px to maintain the uniformity of the input images to the architecture. Normalization of the frames is done as will help the learning algorithm to learn faster and capture necessary features from the images. So, we normalized the resized frame by dividing it with 255 so that each pixel value lies between 0 and 1.

The normalized data now stored in the numpy arrays. The sequence of 30 resized and Normalized frames are stored in a numpy array to give as Input to the .Model. The data will now be split into train and test. The train data will be 75% of data and the test data will be 25% of the data.

Now we have created the LRCN model which is being used in our proposed system for suspicious activity detection from video surveillance. The model has only 12 Layers and does not take too longg to train. The Frames are now resized to 64 x 64 which makes it possible to load the whole dataset even on low end devices. Raw visual input is processed with a CNN, whose outputs are fed into a stack of recurrent sequence models. LSTM networks are well-suited to classifying, processing and making predictions based on time series data, since there can be lags of unknown duration between important events in a time series. The model is trained to predict over 3 classes – walking, running and fighting. The training set is given to the model for training, with the following hyper parameters : epochs was set to 10, batch size was set to 4, validation split was set to .25

IV. Experimental results and discussion

I performed a few experiments to evaluate the performance of the models explained above to confirm whether the LRCN model performs as anticipated. The best classified activity is fighting achieving accuracy of 99.86% and the worst classified activity is walking with 77.6% accuracy. The dataset of the proposed model includes videos of anomalous behaviour which is Fighting as well as it also contains videos of normal behaviour which is walking and running. Following are the images of result of the proposed model. we can observe that the model converges quickly and stabilizes at a certain level, avoiding the problem of overfitting.



time_distributed_input	input:	[(None, 30, 64, 64, 3)]
InputLayer	output:	[(None, 30, 64, 64, 3)]

time_distributed(conv2d)	input:	(None, 30, 64, 64, 3)
TimeDistributed(Conv2D)	output:	(None, 30, 64, 64, 32)

time_distributed_1(max_pooling2d)	input:	(None, 30, 64, 64, 32)
TimeDistributed(MaxPooling2D)	output:	(None, 30, 16, 16, 32)

time_distributed_2(conv2d_1)	input:	(None, 30, 16, 16, 32)
TimeDistributed(Conv2D)	output:	(None, 30, 16, 16, 64)

time_distributed_3(max_pooling2d_1)	input:	(None, 30, 16, 16, 64)
TimeDistributed(MaxPooling2D)	output:	(None, 30, 4, 4, 64)

time_distributed_4(conv2d_2)	input:	(None, 30, 4, 4, 64)
TimeDistributed(Conv2D)	output:	(None, 30, 4, 4, 128)

time_distributed_5(max_pooling2d_2)	input:	(None, 30, 4, 4, 128)
TimeDistributed(MaxPooling2D)	output:	(None, 30, 2, 2, 128)

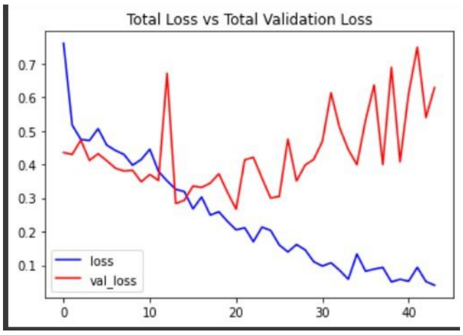
time_distributed_6(conv2d_3)	input:	(None, 30, 2, 2, 128)
TimeDistributed(Conv2D)	output:	(None, 30, 2, 2, 256)

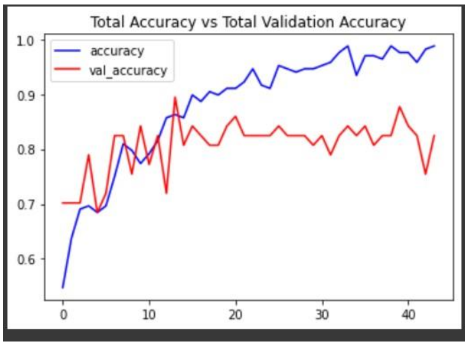
time_distributed_7(max_pooling2d_3)	input:	(None, 30, 2, 2, 256)
TimeDistributed(MaxPooling2D)	output:	(None, 30, 1, 1, 256)

time_distributed_8(flatten)	input:	(None, 30, 1, 1, 256)
TimeDistributed(Flatten)	output:	(None, 30, 256)

lstm	input:	(None, 30, 256)
LSTM	output:	(None, 32)

dense	input:	(None, 32)
Dense	output:	(None, 3)





V. Conclusion

In this paper, I have proposed an end-to-end Human Activity Recognition model based on LRCN. The proposed model combines the CNN features extracted from the video sequences with an LSTM network to model the short and long-term dependencies in the data structure. The experimental results performed on the KTH dataset showed that the proposed method outperforms other baseline work in terms of classification accuracy.

In the future, our challenge is to improve the discriminatory performance of the proposed architecture in terms of accuracy and efficiency by better tuning the hyperparameters and enhancing the architectural design of the model.

References

[1] A. Khan, A. Sohail, U. Zahoora, and A. S. Qureshi, “A survey of the recent architectures of deep convolutional neural networks,” Artif Intell Rev, vol. 53, no. 8, pp. 5455–5516, Dec. 2020, doi: 10.1007/s10462-020-09825-6.

[2] A. Khan, A. Sohail, U. Zahoora, and A. S. Qureshi, “A survey of the recent architectures of deep convolutional neural networks,” Artif Intell Rev, vol. 53, no. 8, pp. 5455–5516, Dec. 2020, doi: 10.1007/s10462-020-09825-6.

[3] D. R. Beddiar, B. Nini, M. Sabokrou, and A. Hadid, “Vision-based human activity recognition: a survey,” Multimed Tools Appl, vol. 79, no. 41, pp. 30509–30555, Nov. 2020, doi: 10.1007/s11042-020- 09004-3.

[4] D. R. Beddiar, B. Nini, M. Sabokrou, and A. Hadid, “Vision-based human activity recognition: a survey,” Multimed Tools Appl, vol. 79, no. 41, pp. 30509–30555, Nov. 2020, doi: 10.1007/s11042-020- 09004-3.

[5] Y. LeCun, Y. Bengio, and G. Hinton, “Deep learning,” Nature, vol. 521, no. 7553, Art. no. 7553, May 2015, doi: 10.1038/nature14539