Human Activity Recognition Using CNN and LSTM Methods

Ayesha Sarwat[1], Dr. K Vaidehi[2] and M. Sowmya[3]

[1]PG Scholar, Department of Computer Science & Engineering,

[2]Professor, Department of Artificial Intelligence & Data Science,

[3]Research scholar, Department of CSE, Annamalai University

Stanley College of Engineering and Technology for Women, India.

Abstract:

Many researchers have approached human activity recognition using a variety of data sets (photos and videos) to use the network to detect human activity in everyday life. To cluster human task recognition on recorded data a Convolutional Neural Network (CNN) along with a Long- Short Term Memory (LSTM) network is used. Convolutional Neural Networks (CNN) is useful for better understanding of image content and Long-Short Term Memory (LSTM) networks aims to work with sequence data. The extraction of spatial features at specific time steps of a sequence (video) can be done by using CNN. While convolution layer is aimed responsible for evaluating spatial features from the frames, LSTM layer is given authority for temporal sequence modelling with extracted spatial features in place. The data used UCF50-Action Recognition Dataset consists of: 50 Activity Categories, 25 Video Groups per Activity Category, 133 Medium Videos per Activity Category, 199 Frame Rate per Video, 320 Frames per video each, 240 Frames Average length of each video, 26 intermediate frames per second per video. By evaluating the Loss and accuracy curves for the models we conclude the best performing model and then put the model to test on some testing videos. The downside to this paper is that, our vision cannot work for most people doing different tasks.

1.0 Introduction:

Image recognition is used as an essence of Artificial Intelligence (AI), and computer vision. Therefore, the following topics are:

1.1 Image Recognition:

A process designed to read a frame and evaluate or recognize the object in a frame and relate it to the category they belong to is called image recognition. When we visualize an object or a group with our eyes, we automatically identify things as different situations and associate them with individual meanings. However, visual recognition is the difficult task that machines can perform. With basic purpose to cluster the recognized objects into same or different category respectively is called object recognition.

Over the years, machine learning, especially in-depth reading technology, has achieved great success in many computers vision and imaging activities. To conquer best results in image recognition many deep learning methods are used which are known to be flexible as In-depth learning algorithms evaluate best performance for visual or image recognition.

T The words image recognition and image acquisition are synonyms but comes with a technological difference. Image Acquisition detects the spread-out objects in an image after capturing a picture or frame. An example of image recognition is face detection, in which algorithms purpose to detect facial marks in images. If we are serious about acquisition, we do not care if the findings are significant in any way. The goal of image acquisition is to distinguish one thing from another to determine how many different businesses exist within the image. Binding boxes are limited on each item.

However, the traditional method of computer vision requires a high level of expertise, a lot of engineering time, and contains many parameters that need to be determined in person, while the functionality of other functions is limited.

The deep learning method consists of many hidden layers in a model. With image classification, facial recognition, deep learning algorithms not only used for real-time object detection and implements beyond human-level functionality and object detection. In 2017, the RCNN Mask algorithm was the fastest real-time material finder for MS COCO benchmarking, with a cut-off time of 330ms per frame. Compared with the usual computer-assisted visual processing 20 years ago, in-depth study requires only engineering knowledge of machine learning tool, not expertise in specific areas of machine vision in order to create handmade features. Also, specialized implementation of in-depth learning requires only dozens of reading samples. However, in-depth reading to recognize bad and good samples it is important to provide manual labeling of data. The process of reading data labeled by humans is called supervised reading. The process of creating data with such a label training AI model requires the work of a time-consuming person, for example, to define traffic conditions that are occurred frequently in autonomous driving.

1. Image Recognition Systems in organized steps:

Dataset:

Every Visualization approches needs training data either be a video or a photo. Neural networks require those training images from the acquired database to create ideas for what specific classes look like.

Training of Neural Networks for Image Recognition:

Captured frames from organized databases are inserted in the neural network algorithm. This is an in-depth or machine learning feature for creating a photo recognition model. To visualize the convolutional neural network before identifying specific categories image recognition algorithm is to be trained.

Testing the AI model:

Images that are part of the learning database are tested in this module. This is used to determine the usability, performance, and accuracy of the model. Thus, approximately 80-90% of the complete image database is used for model training, while the remaining data is reserved for model testing. Model performance is measured based on a set of parameters that indicate the reliability percentage of each image for testing, inaccurate identification, and more.

1.3 Deep Learning:

Convolutional Neural Networks (CNNs) are popular among image recognition process, also referred as deep image recognition. Not only do CNNs accelerate and deliver the best acquisition results in digital image recognition, but they can also detect many instances of an object inside the image, even if the image is distorted. In Deep Image Visualization, Convolutional Neural Networks transcend even people in activities such as classifying objects into subspecies such as a particular dog. In the training period, a convolution layer or acting filter apprehend to recognize each detail of frame before giving input to another layer. One-layer processes colors, another frame structure, and so on. Finally, the combined effect of all these layers is considered collectively when determining whether the same is found.

1.4 Image Recognition Applications:

Across the industry, AI imaging technology is becoming increasingly needed. Its applications provide economic value to industries such as health care, retail, security, agriculture, and much more.

Face Analysis:

Computer-assisted face recognition allows systems to identify identity, goals, emotional states and health, age, or nationality. Some visual aids even aim to measure the perceived levels of attraction to school. Other tasks related to face recognition include face recognition, face recognition, and facial verification that includes visual processing techniques to identify and match face recognition images on a website.

Medical Image Analysis:

Medical image analysis used in medical field is now established as a small essence for the greater advantage in AI. For example, there are many activities related to the diagnosis of melanoma, a deadly skin cancer. Deep image recognition software allows for the monitoring of the tumor over time, for example, abnormal detection of breast cancer scans.

COVID-NET in-depth image learning algorithm is used to recognize Covid-19 features - AI recognition to combat COVID

Object Detection:

AI image recognition technology and video recognition technology are useful for identifying people, patterns, logos, objects, locations, shapes, and colors. Customization of image recognition permits to use while co-exsisting with many software programs. For example, once a photo-viewing system is specialized to find people, it can be used by counting people, a popular computer vision app in grocery stores.

Automatic Plant Image Identification:

The accuracy of image identification for plant family, growth morphology, health morphology, and local abundance. This software compare the plant image with the photo matching software to question the results and the online images.

Food Image Recognition:

Image recognition of in-depth study of different types of food is used with computer-assisted food testing. Photo recognition software applications are implemented to enhance the accuracy of original dietary measurements by inspecting food images clicked by mobile devices.

Image Search Recognition:

This Image recognition learns visual features using deep neural network to explore the most efficient and easy ways of detecting an image. To solve this stretching problem researchers have organized a vast image dictionary which is a training set of neural network features.

2.0 LITERATURE SURVEY:

A paper [13] was proposed which renders a multiple class activity prediction including individuals as well as groups from video by using the state-of-the-art You Look only Once (YOLOv3) object detector. Any human action either simple or complex ranging from grayscale to RGB image frames captured from video sequences can be detected by Deep Landmark model when considering evaluated account of geographical knowledge of cameras and YOLO object detection framework. This model is tested and compared with various benchmark datasets and found to be the most precise model for detecting human activities in video streams.

A new hybrid deep learning model [14] was proposed which is initialized of Gated Recurrent Neural Networks for sequences data or a video. However, to have an efficient classifier for assigning the class label, it is very necessary to have a strong features vector. With features as the potential data; features extraction can effect on the performance of the algorithm and the computation complexity. The proposed approach is evaluated on the challenging UCF Sports, UCF101 and KTH datasets. An average of 96.3% is obtained when we have tested on KTH dataset.

A paper [15] was proposed in which additional layers are mounted on a VGG16 model-pre trained with 1000 classes to encourage transfer learning for facial expressions on a deployed deep learning-based framework then the model was validated on the reference datasets CK+, JAFFE. Extended Cohn-Kanade (CK+) and Japanese Female Facial Expression (JAFFE) are popular Facial Expression Dataset. The proposed model has shown 94.8% accuracy on CK+ and 93.7% on the JAFFE dataset and found superior to existing techniques.

This manuscript [7] was proposed to predict the human emotions to accomplish this CNN wass used to extract local features and also statistical features which are used as a database. WISDM and UCI datasets with labeled accelerometer data evaluate accuracy, while also cross-examine datasets. The output was phenomenal with low computational cost and didn’t need to be physically featured.

Deep learning methods for recognizing facial expressions [16] This thesis used deep learning techniques to identify seven main emotions: anger, disgust, fear, happiness, sadness, surprise, and neutrality. The following architectures were used in one of his works: VGG-16 and ResNet 50. A performance of 31.8% was determined using the SVM multiclass classifier as baseline accuracy. With improved results and ensemble and transfer learning techniques, the published accuracy was 67.2% for ensemble learning techniques and 78.3% for encapsulated transfer learning.

This paper [18] used five machine learning models which analysis the data given from repository after extracting it and also calculate distinct accuracy with the distinct performances by all five machine learning models.. Used decision tree (DT), random forest (RF), LR, SVM, and hidden layer artificial neural network (ANN) to predict human activity from mobile data.

In this paper [17], a combine version of CNN and LSTM is taken into consideration. This model could automatically extract the activity features and classify them using several model parameters. With LSTM being a copy of (RNN) makes it convenient for data with time sequences. Architecture is given the power of not only gathering the real-time data from sensors but also given a responsibility to pass them as inputs to Convolutional layers which then is passed to the double LSTM.

In this Paper [5] a questionable prediction is formulated with spatio-temporal sequences common in both input and target. Fully linked LSTM (FC-LSTM) has been extended to consist of convolutional structures for both enter-to-state and state-to-state transactions. Experiments prove that ConvLSTM networks outperform FC-LSTM in detection and performance.

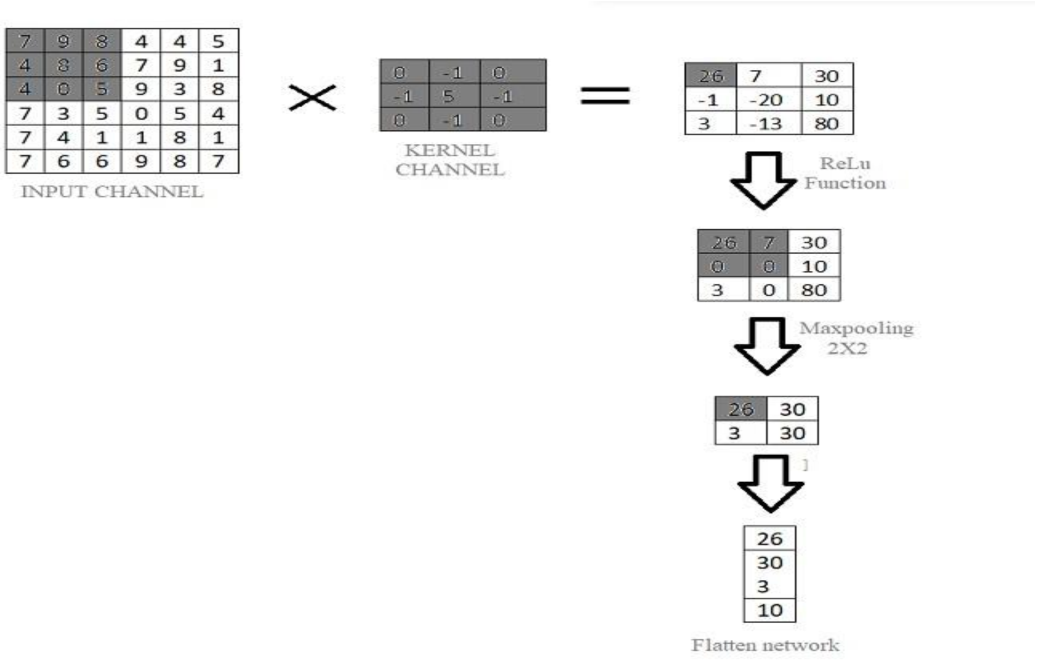
In this paper [12] a raw depth video-based RNNs are used with two CNN-LSTM models for detecting human action recognition with different architectures but the same long-term learning strategy. One uses a stateless generator and the other emphasizes the stateful capabilities of her RNN, but especially applies to the specific case of human action detection. With this approach, the proposed model achieved accuracies of 75.26% (CS) and 75.45% (CV) for stateless models and 80.43% (CS) and 79.91% (CV) for stateful models.

3.0 Convolution Neural Networks:

Convolutional neural networks refer to a sub-category of neural networks and have all the characteristics of neural networks. As image is a matrix of nxn dimension, an input image of nxn dimension and filter which can move along the image matrix calculate the convolution of the matrix to get output which provides a 3x3 matrix for the same and is called kernel or filter.

Input image is processed with the kernel. The kernel is sided over by 1 pixel which is also called a stride. And after every slide, compute the multiplication of part of matrix under the kernel and sum up the output which gives us integer which makes an element of the Convolved Feature matrix called a Feature Map. Convolved Feature matrix is also known as Feature Map.

The 3x3 matrix is a Kernel/Filter and the same sliding matrix is known as Filter or Kernel. In real world data the input image can be quite big and will contain lots of information most of which is not important for our deep learning or classification problem. To solve a problem at our hands Kernel or Filters are used.



7	9	8	4	4	5
4	8	6	7	9	1
4	0	5	9	3	8
7	3	5	0	5	4
7	4	1	1	8	1
7	6	6	9	8	7

0	-1	0
-1	5	-1
0	-1	0

26	7	30
-1	-20	10
3	-13	80

26	7	30
0	0	10
3	0	80

26	30
3	30

26
30
3
10

Fig 3.1 Working of kernel with single input image

Epochs can be defined as the time during which our learning algorithm has seen the complete training data i.e. with the accuracies as the proof, the backward pass which gives a minimum loss function value. This complete one forward and one backward pass of all the instances of training data is termed as an Epoch.

The difference between an iteration and epoch is that the any neural network first divide the whole training dataset into batches e.g. If there are 1000 images in our training dataset which is divided into a batch of 4 i.e. 500 each. Then every time a batch of 500 is processed which completes one iteration. So, in our example to complete an epoch 4 iterations are required.

Multiple epochs are needed to converge any deep learning neural network, as using a single epoch will train a model that is under-fit and will be of no use. The reason behind why multiple epoch or why one epoch is needed will not have much effect on weight is because generally we use Gradient Descent algorithm for learning the weights, so with one epoch might not end up with weights that minimize our loss function, that may not have the best minima and there can be other minima's which can reduce the loss function even further, so requirement of more epochs make gains in achieving minimization or till our algorithm converges.

3*3 convolution

3x3 is the size of kernel which is used to do convolution over the input image, it is called 3x3 convolution. Benefits of using a 3 x 3 Convolution is that it is very much effective to detect the local feature in the image. Being of smaller size the computation cost is also relatively less then the other let's say 5 x 5 convolution. When 3x3 convolution is applied over a N x N image, as aresult output is feature map of (N-2) x (N-2).

Feature Maps

The output obtained after repeatedly applying convolution over the sub-regions of an entire image is called Feature Maps. Previous layer is an input for initial pass. While doing convolution with help of nxn kernel and with a stride of 1, one pixel is moved at time and perform a convolution over a particular sub region of the input image, while doing that neurons get activated and the output is generated which contains the information about the features like edge, curve this information is then collected into a feature map.

Let's take an example of a 32x32 image, and using a receptive field of 5 x 5 over the whole image with stride equal to 1 will end up of having 28 x 28 sized feature map or distinct activation totaling to 784. This number is the times a neuron is fired off with the data of features.

An objective when involved in convolution operation is to extract features of high quality from an image frame. You can always add multiple transformation layers when creating a neural network. The first layer of convolution rounds off the gradients while the second limits the edges. Layers There is no magic number for how many layers to add, as the layers depend on the complexity of the image. Note that using the 3x3 filter effect on the original image will apply the effect to the dynamic 3x3 element. As such, images are usually overlapped at both end values to maintain their actual size.

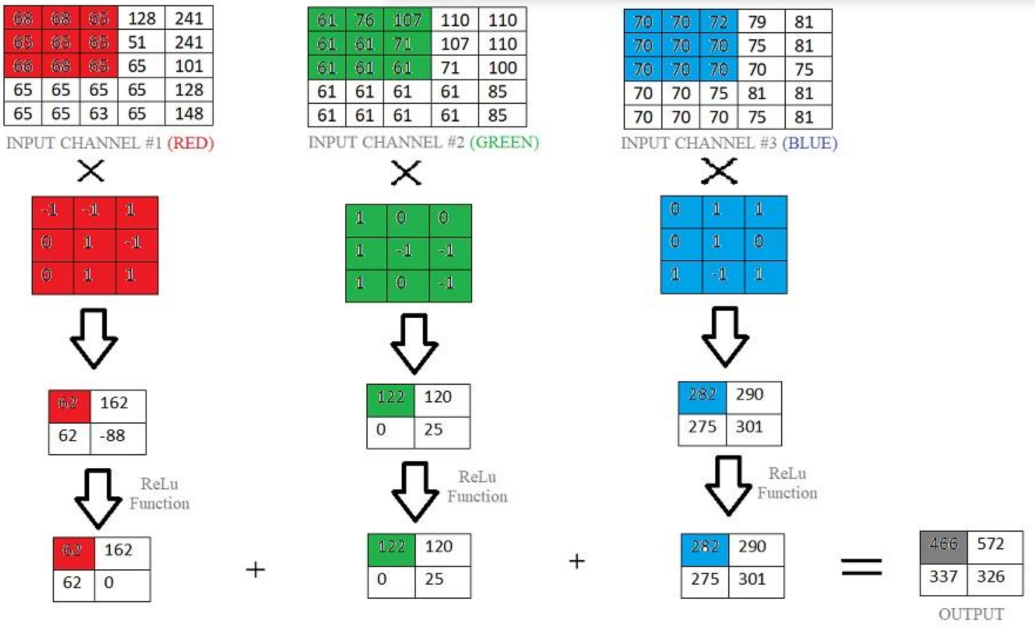


Fig 3.2 Working of Kernel with an R-G-B Images

Activation Function

The activation function is a node that is put at the end of or in between Neural Networks. They help to decide if the neuron would fire or not. We have different types of activation functions. Rectified Linear Unit (ReLU) is one among them. ReLU function is the most widely used activation function in neural networks. One of the greatest advantage of ReLU has over other activation functions is that it does not activate all neurons at the same time. From the image for

ReLU function above, it converts all negative inputs to zero and the neuron does not get activated. This makes it very computational efficient as few neurons are activated per time. It does not saturate at the positive region. In practice, ReLU converges six times faster than tanh and sigmoid activation functions.

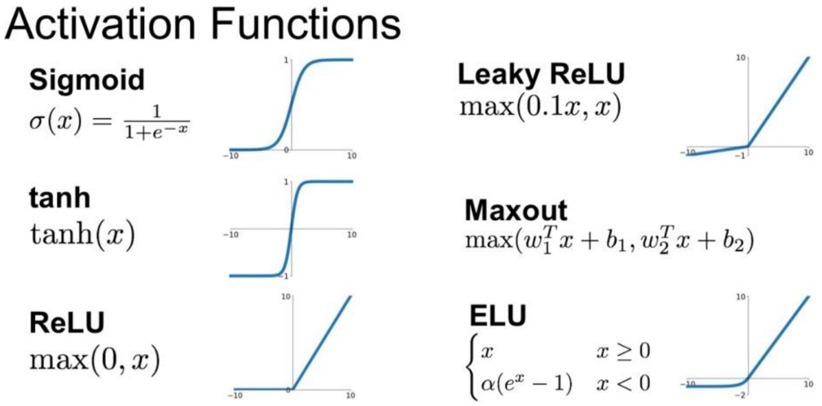


Fig: 3. Activation function

3.4 Long-Short Term Memory (LSTM):

Long Short-term Memory Networks; a special form of RNN [11], which is known to adapt long-term dependencies. They have been delivered by means of Hochreiter & Schmidhuber (1997), and had been refined and popularized within the next paintings.They work very well on a variety of major problems, and are now widely used. LSTMs are specifically designed to avoid the problem of long-term dependence. All emerging neural networks have a kind of repetitive series of neural network modules.

LSTMs [6] interacts with all the layers involved within a network with the repeating module constructed with distinction.

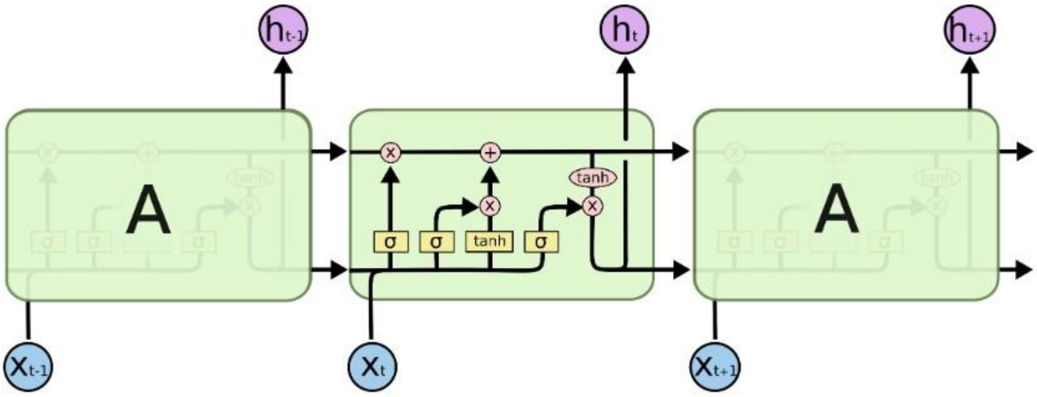


Fig 3.4 Communicating Layers in LSTM Cell

[4] In the diagram above, each line indicating a whole vector, i.e., from the output to the input of the other. The pink circles dictate independent operation the vector addition, while the

yellow boxes are for aid of the layers of the neural networks. Line integration means merging, while the fork line displays its copied content and copies to different locations.

- Step 1: decide what data to discard within the network. This choice turned into made by using a sigmoid layer called the "gate oblivion layer". Draw a number between 0 and 1 by looking at ht - 1 and xt for each number in the Ct-1 cell state. output 1 means "stop doing this", output 0 means "get rid of it completely".
- The subsequent step is to determine what new information to keep within the cell environment. There are two components to this. First, a sigmoid layer referred to as the "input gate layer" determines the price to check. The responsibility of tanh layer is to deliver the produced vector of new candidate values C~t this subsequent step will combine the two into one international update.
- Now update the status of the old cell Ct-1 to the new status of Ct. The previous step determined what to do. you really just have to do it.
- We repeat old situations with ft and forget what we wanted to forget before. Then add ∗ C ~ t.
- Finally, you need to decide what to release. This output is based on the cell status but is a filtered version. First, we use a sigmoid layer that determines which parts of the cell structure are freed. To get the desired state of the cell set it to tanh and then let the sigmoid gate to multiply and extract the desired part.

4.0 Proposed System:

Convolutional Neural Networks (CNNs) are very good at image data and Long-Short Term Memory (LSTM) networks are good at working with sequential data, but when you combine them, you get the best of both worlds and solve difficult computer vision problems. like video split. We will actually use two different structures and methods in TensorFlow to do this. Finally, we will take the most effective model and make predictions about it in YouTube videos.

In the first step, we download and visualize the data and labels to get an idea of what we will be dealing with. We will use UCF50 - An Activity Data Recognition Set, which includes real-time videos taken from YouTube that separates this data set from many other data recognition systems that are not real and played by players. Dataset contains:

50 Action Classes with a total video of 6671

25 Video Groups per Activity Category

133 Medium Videos per Activity Category

199 Average number of frames per video

320 Frame Range for Each Video

240 Average Frame Length Each video

Medium frames 26 per second per video

To visualize, we will select all the random categories in the database and the random video in each selected category and see the first draft of the selected videos written by related labels. In this way we will be able to visualize all sections of the database.

Next, we will do a preliminary processing of the database. First, we will study video files in the database and change the frame size of the videos to a specified width and length, in order to reduce the figures and make the data normal to distance [0-1] by dividing 255pixel values, enabling faster integration while training pre-specified network IMAGE_HEIGHT, IMAGE_WIDTH and SEQUENCE_LENGTH. These static conditions can be increased for best results, although increasing the sequence length only applies to a certain area, and increasing the values will result in a more costly process for the computer.

The frames_extraction () function is defined as creating a list containing framed and standard frames for the video in which the channel is transmitted as an argument. The function will read the video file frame by frame, although not all frames are added to the list as we will only need an evenly distributed sequence length. Then create a create_dataset () function that will duplicate in all classes specified in CLASSES_LIST regularly and call frame_extraction () function in all video files for selected categories and restore features, class labels, and video_files_paths. Released class labels (class references) are converted into vectors with one hot code.

From now on, we have the necessary features (NumPy list containing all video output frames) and label_hot_encoded_one (and Numpy array containing all class labels in one hot code format). So now, we're going to break down our data to build training and testing sets. We will also manipulate the database prior to fragmentation to avoid any bias and find variations that represent the distribution of data as a whole.

In the fourth step, we apply the first method using a ConvLSTM cell combination. A ConvLSTM cell is a type of LSTM network that contains convolution functions in the network. is an architecture-centric convolutional LSTM that allows local regions of data to be identified while considering temporal relationships. With video segmentation, this method effectively captures local relationships in individual frames and temporary interactions in all different frames. As a result of this transformation structure, ConvLSTM is capable of taking 3-dimensional inputs (width, length, number_ of channels) while simple LSTM only captures 1-dimensional inputs which is why LSTM does not integrate to create a Spatio-temporal data model in it. we build the model, with repetitive layers of Keras ConvLSTM2D. The ConvLSTM2D layer also captures the number of filters and kernel size needed for the convolutional operation. Layer output is flat at the end and is given a dense layer by activating the softmax that eliminates the possibilities for each stage of the action.

By using maxpooling3D layers we reduce frames, avoid nonessential calculations and pull layers to prevent over-modeling of data. Architecture is simple and has a small number of professional parameters. This is because we are only dealing with a small data set that does not require a large model. We will now use the create_convlstm_model () function created above, to create the required convlstm model.

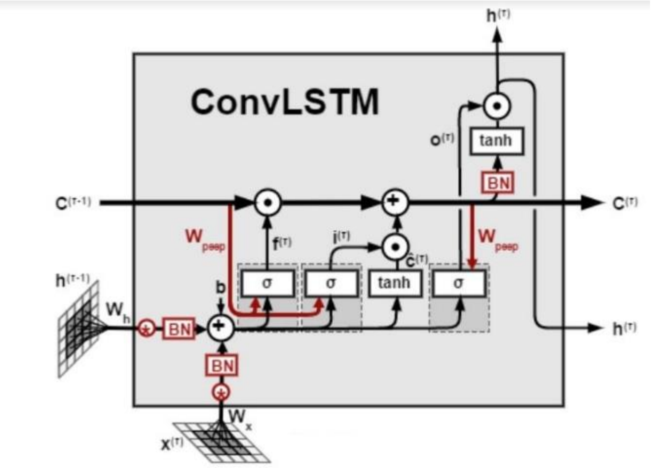


Fig 4.1 ConvLSTM model Network

Next, we will add an early stopping callback to prevent overfitting and start the training after compiling the model with 50 epochs and batch size as 4.

In the fifth step, the LRCN Approach is used to combine the Convolution and LSTM layers into a unique model. Another similar approach would be to use the CNN model and the separately trained LSTM model. A CNN model can be used to extract local features from frames in a video, and a pre-trained model can be used for this purpose, which can be better prepared for the problem while the LSTM model can predict performed action on video. But here, we will use another method known as the Long-term Recurrent Convolutional Network (LRCN), which integrates CNN and LSTM layers into a single model. Convolutional layers are used to exclude a local element from frames, and extruded local features are given to the LSTM layer for each step of the sequence modeling time. In this way, the network learns the spatiotemporal features directly in the final training, resulting in a stronger model.

We'll also use the TimeDistributed wrapper layer, which allows the same layer to be applied independently to an entire video frame. Therefore, create a layer (rounded) that can take input input (none_from_mames, width, length, number_ of channels) if initially the layout of the layer layout was (width, length, number_ of channels) which is very useful as it allows you to embed the entire video in one shot.

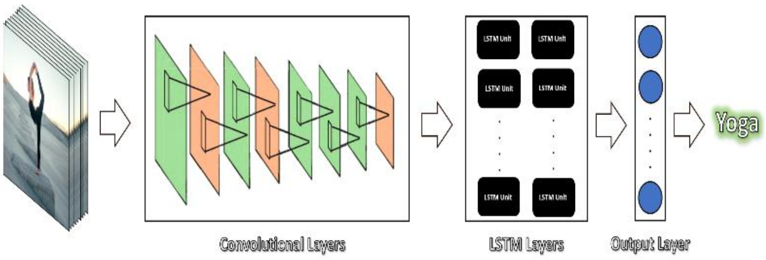


Fig 4.2 LRCN Model Network

In order to use our LRCN architecture, we will be using Conv2D layers over time that will be followed by MaxPooling2D and Dropout layers. The feature extracted from the Conv2D layers will be flattened using the Flatten layer and will be provided with the LSTM layer. A dense layer with a softmax implementation then uses the output from the LSTM layer to predict the action being taken. Now we will use the create_LRCN_model() function created above to create the desired LRCN model. After inspecting the structure, we will assemble and begin training the model with 70 epoch and batch size as 4.

After each model has been tested and the models saved for losses and precision curves are sorted using the plot_metric () function. This fuction will visualize training and certification metrics for each model. Model loss curves are a strategy that uses a complete loss associated with a total loss of validation. Similarly the precision curve of the models is sorted using complete perpendicular accuracy to complete verification accuracy.

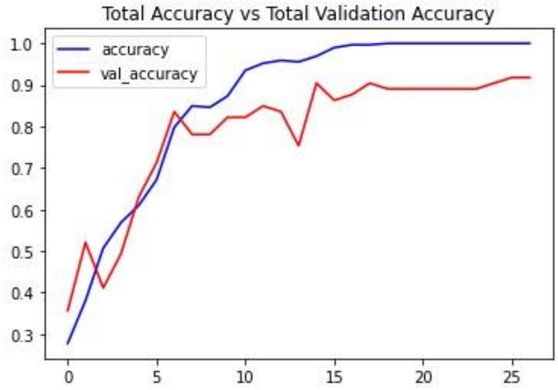


Fig 4.3 Total Accuracy Vs Total Validation Accuracy for ConvLSTM Model Network

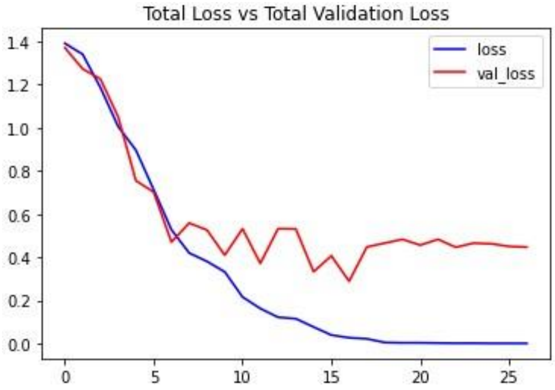


Fig 4.4 Total Loss Vs Total Validation Loss for ConvLSTM Model Network

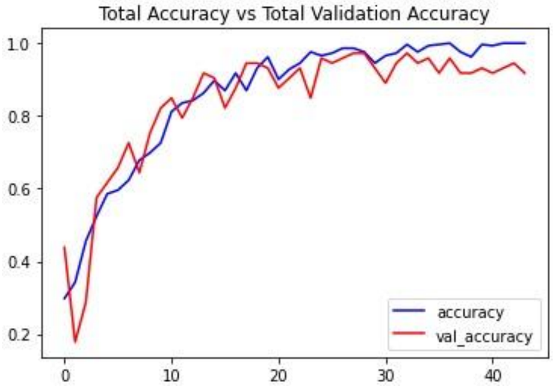


Fig 4.5 Total Accuracy Vs Total Validation Accuracy for LRCN Model Network

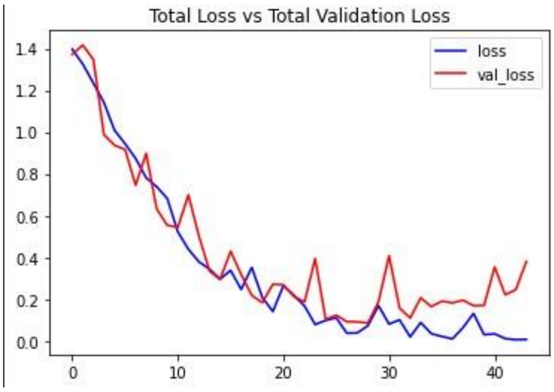


Fig 4.6 Total Loss Vs Total Validation Loss for LRCN Model Network

From the structural metrics we can check that LRCN performed better than ConvLSTM with 92% accuracy and 20% loss. While ConvLSTM provided 80% accuracy.

From the results, it seems that the LRCN model has performed very well in a small number of classes. so in this step, we will set up an LRCN model to test in other youtube videos. We will create a download function for_youtube_videos () to download YouTube videos first using the pafy library. The library only needs the URL in the video to download it and its related metadata as the title of the video. Next, we will create a predict_on_video () function that will automatically read the frame of the video frame in a path that has been transmitted as an argument and will perform action recognition on the video and save the results.

5.0 Conclusion:

In this paper, we proposed a CNN-LSTM approach to human activity recognition using UCF50-Human Activity Recognition dataset. This approach helps in frame extraction from a video robustly while taking advantage of both the CNN and LSTM models which works great with Spatial and temporal feature extraction respectively. The ConvLSTM and LRCN model achieved better performance when it was compared to other deep learning approaches that uses videos as a dataset. We evaluated the model metric using total loss vs total validation loss and total accuracy vs total validation accuracy for each of these architectures of CNN-LSTM model. The LRCN model out performed with an accuracy of 92% while ConvLSTM performed with an accuracy of 80%. The training of LRCN took less time when compared to ConvLSTM model. This model is applied with complex activities as dataset categories to tackle the recognition of the activity.

For future work, the model is to be trained to recognize action on multiple people performing different activities in the frame. To train this asset in the model the data used should be annotated for more than one person’s activity and also provide the bounding box coordinates of the person along with the activity he or she is performing or a hacky way is to crop out each person and perform activity recognition separately on each person but this will be very expensive.

6.0 References:

1. Joy Zhang, Le T. Nguyen, Ming Zeng, Ole J. Mengshoel, Jiang Zhu, Pang Wu, Bo Yu,: *Convolutional Neural Networks for Human Activity Recognition using Mobile Sensors*, page 198-199, 2014.

2. Horst Kloeden, Christoph Stiller, and Eike Rehder: *Head Detection and Orientation Estimation for Pedestrian Safety,* page 2292-2297. **IEE**, 2014.

3. Jiang Zhu, Joy Zhang, Le T. Nguyen, Ole J. Mengshoel, Ming Zeng, Bo Yu, Pang Wu: "*Convolutional Neural Networks for Human Activity Recognition using Mobile Sensors",* page 5-8. **ResearchGate**, 2015.

4. *Understanding LSTM Networks.* Colah's blog, 2015.

5. Xingjian Shi Zhourong Chen Hao Wang Dit-Yan Yeung and Wai-Kin Wong Wang-chun Woo: "*Convolutional LSTM Network: A Machine Learning Approach for Precipitation Nowcasting",* page 4-5. 2015.

6. *Essentials of deep learning: Introduction to Longshort Term memory.* Analytics Vidhya, 2017.

7. Ignatov Andrey: *Real-time human activity recognition from accelerometer data using Convolutional Neural Networks,* page 13-15. ***Swiss Federal Institute of Technology in Zurich (ETHZ)***, 2017.

8. Jason Brownlee, *LSTMs for Human Activity Recognition Time Series Classification.* Machine Learning Mastery, 2019.

9. Adrian Rosebrock, *Video classification with Keras and Deep Learning*. Pyimagesearch, 2019.

10. Dshahid, *Convolutional Neural Network.* Towards Data Science, 2019.

11. Usman Malik, *Solving Sequence Problems with LSTM in Keras.* StackAbuse, 2019.

12. David Fuentes-Jimenez, Cristina Losada-Gutierrez Adrian Sanchez-Caballero, ´ Universidad de Alcala *"Exploiting the ConvLSTM: Human Action Recognition using Raw Depth Video-Based Recurrent Neural Networks"*, Pages 5-17. arXiv:2006.07744v1, 2020.

13. Madhu Bala Myneni, Budi Padmaja and Epili Krishna Rao Patro: *A comparison on visual prediction models for MAMO (multi activity-multi object) recognition using deep learning,* pages 5-10. **Journal Big Data**, 2020.

14. Med Salim Bouhlel, Neziha Jaouedi, Noureddine Boujnah: *A new hybrid deep learning model for human action recognition*, page 449. **Journal of King Saud University – Computer and Information Sciences,** 2020.

15. Vanita Jain, Arun Kumar Dubey: *Automatic facial recognition using VGG16 based transfer learning model*, page 3. **Taylor & Francis Group,** 2020.

16. James Wong, Alexandru Savoiu: *Recognizing Facial Expressions Using Deep Learning*, page 2-5. 2020.

17. Kun Xia, Jianguang Huang, And Hanyu Wang: "*LSTM-CNN Architecture for Human Activity Recognition",* page 56859. **IEEE**, 2020

18. Md. Tahmid Hasan Fuad, Jakaria Rabbi, Md. Abdul Awal: *Human Activity Analysis and Recognition from Smartphones using Machine Learning Techniques*, pages 2-5. 2021.

19. Mihir Verma, Dr. M. Baskar, Palash Sharma: *Action Recognition Using Deep Learning And CNN*, pages 819-820. **Turkish Journal of Computer and Mathematics Education**, 2021.