Video Understanding : Video Classification of UCF101 Human Action Dataset

Team Members:
Anusha Nemilidinne: 1619484
Teja Salapu: 1634379

1.Introduction

With the advent of ImageNet, computers are able to efficiently detect the objects for thousands of classes including human emotions,postures and also popular landmarks.Next challenge is video analysis, so that we can build efficient robots that can not only detect the objects in a image but can learn what is happening in the video as shown in Fig1.Popular datasets like UCF,Youtube8M have emerged making video understanding a reality.We worked on classification of human actions videos .

Fig1:What is really happening here- Football game right?

One of the ultimate goals of artificial intelligence research is to build a machine that can accurately understand humans actions and intentions, so that it can better serve us. We can simulate robots that can mimic human actions and also build home robots that can see and understand human actions.Imagine that a patient is undergoing a rehabilitation exercise at home, and his/her robot assistant is capable of recognizing the patient's actions, analyzing the correctness of the exercise, and preventing the patient from further injuries. Such an intelligent machine would be greatly beneficial as it saves the trips to visit the therapist, reduces the medical cost, and makes remote exercise into reality. Other important applications including visual surveillance, entertainment, and video retrieval also need to analyze human actions in videos. In the center of these applications is the computational algorithms that can understand human actions. Similar to human vision system, the algorithms ought to produce a label after observing the entire or part of a human action execution.

Biggest challenge in video classification is retaining the true meaning of the combined meaning of the frames in a video as it is quite different from image classification.We have to fuse all the video frames to predict its classification.Our report provides an intuition on different approaches and our implemented architecture- 3DCNN and selection of frame count-depth i.e third dimension of our cnn.Finally, we compare different methods and factors and analyze their effects on model performance.

2.Related work

In video classification, we have to relay continuous frame information unlike image classification and predict based on both temporal and spatial information.We have different approaches for this.

Conv2D- Majority voting

First one, Classifying using conv2d, in this we consider videos as a set of frames, looking at each frame independently, and classifying the entire video based solely on that one frame. We aren't

looking at all the frames and doing any sort of averaging or maxing.This performs poorly because of lack of temporal information retaining.
Accuracy-65%

Time-distributed CNN, passing the features to an RNN, in one network

In order to consider the temporal information,keras provides Time- distributed model which allows us to distribute layers of a CNN across an extra dimension time and passing this to RNN-LSTM layer. Similar to temporal feature pooling, LSTM networks operate on frame-level CNN activations as well as integrate information over time. LSTM also outputs a hidden vector for each input activation frame. The first LSTM layer accept the inputs from the pretrained CNN model, and then the second LSTM receives sequential outputs from the previous LSTM layer. In the end, we apply the output into fully-connected layer and calculate the final softmax score.
Accuracy-45%

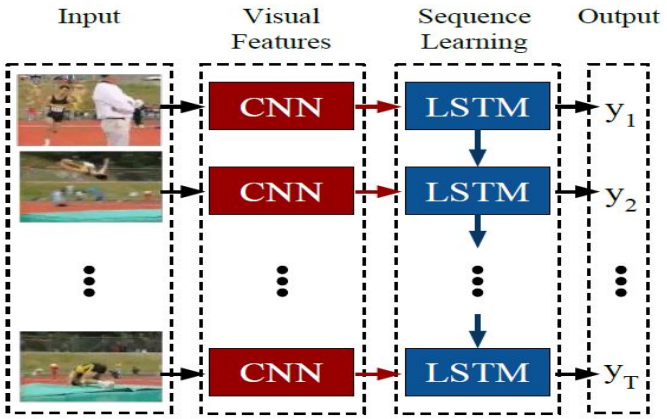


Figure 2: CNN-LSTM Model

3.Approach

3.1 Video classification

We performed classification by considering frames of video- we experimented with 18 and 30 frames for our model. We pass the 3D input to our 3DCNN model and label is decided by majority based prediction of frames.

3.2 Our Method

3D Convolution Neural network(C3D):

Compared to 2D ConvNet, 3D ConvNet has the ability to model temporal information better owing to 3D convolution and 3D pooling operations. In 3D ConvNets, convolution and pooling operations are performed spatio-temporally while in 2D ConvNets they are done only spatially. 2D convolution applied on an image will output an image, 2D convolution applied on multiple images (treating them as different channels also results in an image. Hence, 2D ConvNets lose temporal information of the input signal right after every convolution operation. Only 3D convolution preserves the temporal information of the input signals resulting in an output volume. The same phenomena is applicable for 2D and 3D polling.

Deep 3-dimensional convolution neural network trained on a large scale video dataset achieves state-of-the-art performance on video classification. C3D operates on stacked video frames and

extends the original 2D convolution kernel and 2D pooling kernel into 3D kernel to capture both spatial and temporal information.We used 3 × 3 × 3 convolution kernel size for best performance.

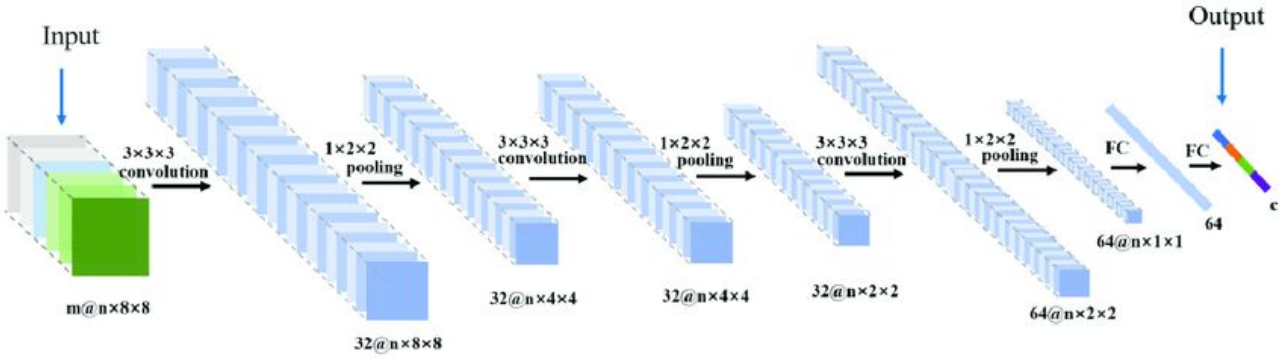


Figure 3: C3D model (convolution on temporal and spatial data)

4. Real-World Applications

Video analysis is not only used for commercial video tagging purpose,it has several applications. we can build efficient robots that can recognise human actions and more than that using simulation of actions they can also mimic like OpenAI Robot hand mimicking Cube dextering.

4.1 Visual Surveillance

We can ensure security using video analysis of CC camera.Places under surveillance typically allow certain human actions, and other actions are not allowed. With the input of a network of cameras, a visual surveillance system powered by action recognition and prediction algorithms may increase the chances of capturing a criminal on video, and reduce the risk caused by criminal actions. For example, in Boston marathon bombing site, if we had such an intelligent visual surveillance system that can forewarn the public by looking at the criminal's suspicious action, the victims' lives could be saved. The cameras also make some people feel more secure, knowing the criminals are being watched.

4.2 Video Retrieval

Managing and retrieving videos according to video content is becoming a tremendous challenge as most search engines use the associated text data to manage video data. The text data, such as tags, titles, descriptions and keywords, can be incorrect, obscure, and irrelevant, making video retrieval unsuccessful. An alternative method is to analyze human actions in videos, as the majority of these videos contain such a cue. For example,researchers created a video retrieval framework by computing the similarity between action representations, and used the proposed framework to retrieve videos of children with autism in a classroom setting. Compared to conventional human action recognition task, the video retrieval task relies on the retrieval ranking instead of Classification.

4.3 Home Robots and simulation of human actions

Now a days, home robots are emerging like osmo,we can incorporate them with video understanding ability for better human robot interaction. Hospitals and children with special care units are employing robots, they can analyse human actions and even mimic them in future. It is

still a research area with a lot of potential to move our technology forward. OpenAI has succeeded in simulating robots actions to some extent with incredible performance.

4.4 Autonomous Driving Vehicle

Action prediction algorithms can predict a person's intention.In an urgent situation, a vehicle equipped with an action prediction algorithm can predict a pedestrian's future action or motion trajectory in the next few seconds, and this could be critical to avoid a collision. By analyzing human body motion characteristics at an early stage of an action using convolutional neural network. Action prediction algorithms can understand the possible actions by analyzing the action evolution without the need to observe the entire action execution.

5.2 Research Challenges

5.2.1 Intra- and inter-class Variations

As we all know, people behave differently for the same actions.For a given semantic meaningful action, for example, "running",a person can run fast, slow, or even jump and run. That is to say, one action category may contain multiple different styles of human movements. In addition, videos in the same action can be captured from various viewpoints.They can be taken in front of the human subject, on the side of the subject, or even on top of the subject.These variations will be even larger on real-world action datasets like UCF101 which we are working on.

5.2.2 Cluttered Background and Camera Motion

It is interesting to see that a number of human action recognition algorithms work very well in indoor controlled environments but not in outdoor uncontrolled environments. This is mainly due to the background noise.Other environment-related issues such as viewpoint changes, dynamic background will also be the challenges that prohibit action recognition algorithms from performing well on real time data.

5.2.3 Uneven Predictability

A lot of frames in a video can be very redundant, this poses challenges in training of the model and results in uneven predictions.

Implementation

Architecture:

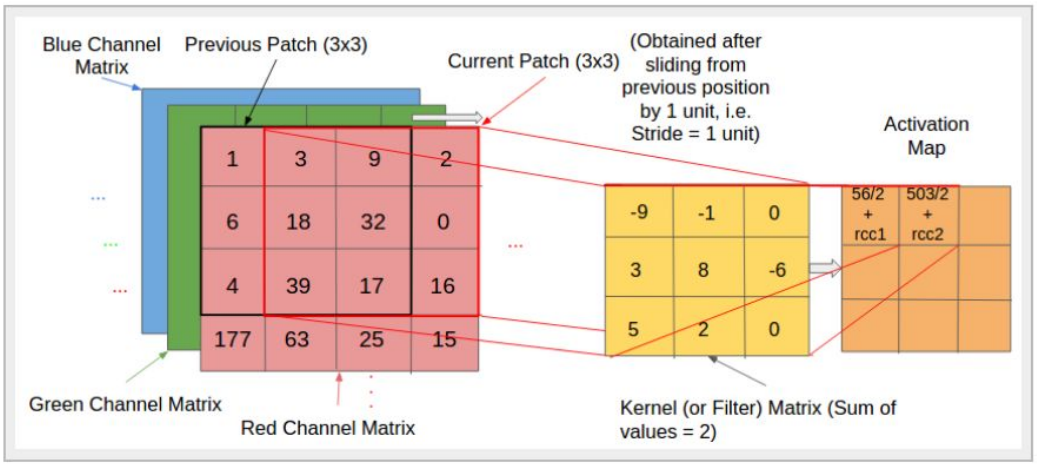
We have implemented 3D Convolution Neural Network using Keras conv3D package. The architecture consists of 16 layers including the input. There is a alternating convolutional and rectification layers followed by subsampling which is max pooling layer. Next comes the dropout layer.This is again followed by 2 alternating convolutional and rectification layers followed by max pooling layer. Next is a dropout layer. The next layer performs flattening of the input to convert high dimensional input array into 1D array. Then a fully connected layer consisting of 2048 neurons is present which is connected to an output layer consisting of neurons equal to the number of classes used for video classification. Between the fully connected network and output layer , is a dropout layer. Dropout is a regularization technique used to prevent overfitting. Dropout rates of 0.25,0.25 and 0.5 are used in the architecture. That means the probability that an input node is kept is given by 0.5.

The architecture looks like the following:

The first convolutional layer uses 64 filters or kernels , the second and third convolutional layers uses 128 filters and the last convolutional layer uses 256 filters.
ReLU and Softmax activation functions are used for rectification layers.

Kernels:

A filter/Kernel is an integral component of the layered architecture.The kernel size used here is 3 by 3 by 3 . It is convolved with the input image to generate feature maps. The following figure shows how an input image is convolved.

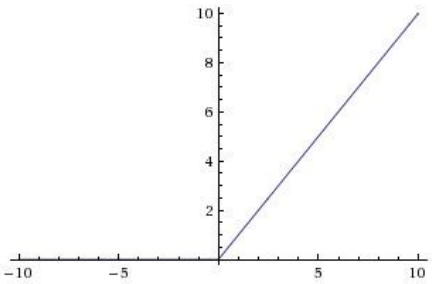


Activation Layers:

Activation layers are used to introduce non-linearity in the input. Some of the popular activation functions are ReLU and Softmax.

ReLU:

It helps to reduce vanishing gradient descent problem .
It is simply defined as:f(x)=max(0,x)
Graphically it looks like this:

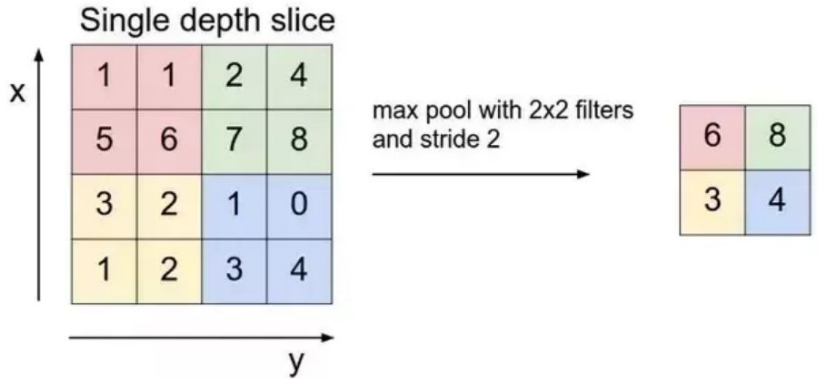


Softmax:

This function calculates the probabilities of each target class over all possible target classes.The formula computes the **exponential (e-power)** of the given input value and the **sum of exponential values** of all the values in the inputs. Then the ratio of the exponential of the input value and the sum of exponential values is the output of the softmax function.

Max Pooling:

The objective of pooling is to downsample the input image to reduce the dimensionality of feature space. It helps to prevent overfitting by providing abstract representation.It reduces the computational cost by reducing the number of parameters to learn.



Dataset:

We have used UCF101 dataset for our analysis and a glimpse of the UCF101 dataset looks like:

UCF101 dataset comprises of realistic videos collected from Youtube. It contains 101 action categories, with 13320 videos in total. UCF101 gives the largest diversity in terms of actions and with the presence of large variations in camera motion, object appearance and pose, object scale, viewpoint, cluttered background, illumination conditions, etc.

Fig4:UCF101 dataset

Implementation Details:

We have used 3D CNN to perform video classification on UCF101 video action dataset.

Software:

Python 3.6.7

Keras 2.2.4 with tensorflow as the backend

Scikit learn 0.19.2

Hardware:

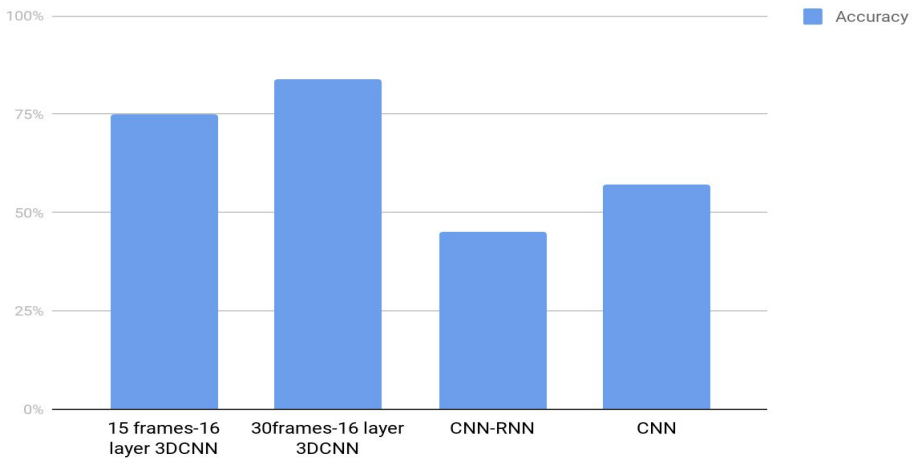
The project is executed on Google's Colaboratory which offers TESLA GPU support for implementation and analysis of large scale deep learning models.

Video preprocessing :
The videos are varying length sequences and 30 frames of a video are considered. Since some frames of the video are empty which means there is no action taking place, some frames are skipped according to a defined criteria i.e., taking frames in fixed intervals apart.
Each frame is resized to a size of 50 by 50 pixels and converted to grayscale for reducing the computational complexity since it is reported that luminance is by far more important in distinguishing visual features than chrominance. We tested several configurations for example, 15 frames with image sizes varying from 32 by 32 to 50 by 50 pixels. We have also tested the model with RGB color coding.

Evaluation Metrics:

Classification Accuracy:
The basic metric for classification problems is to measure the accuracy, which is to calculate the percent of correct predictions.

We have got an accuracy of



Precision-Recall:
Precision-recall is a common method in classification problems. Precision measures result relevancy, while recall measures how many relevant results are returned 6 . F1-score is weighted harmonic mean of the precision and recall, and it also measures the degree of balance between false positives and false negatives.

$$Precision = \frac{TP}{TP + FP}$$

$$Recall = \frac{TP}{TP + FN}$$

$$F1 = 2 \times \frac{Precision \times Recall}{Precision + Recall}$$

Where TP = True Positive
FN = False Negative
FP = False Positive

Below is the precision recall report for the configuration using 50 by 50 image size and 30 frames.

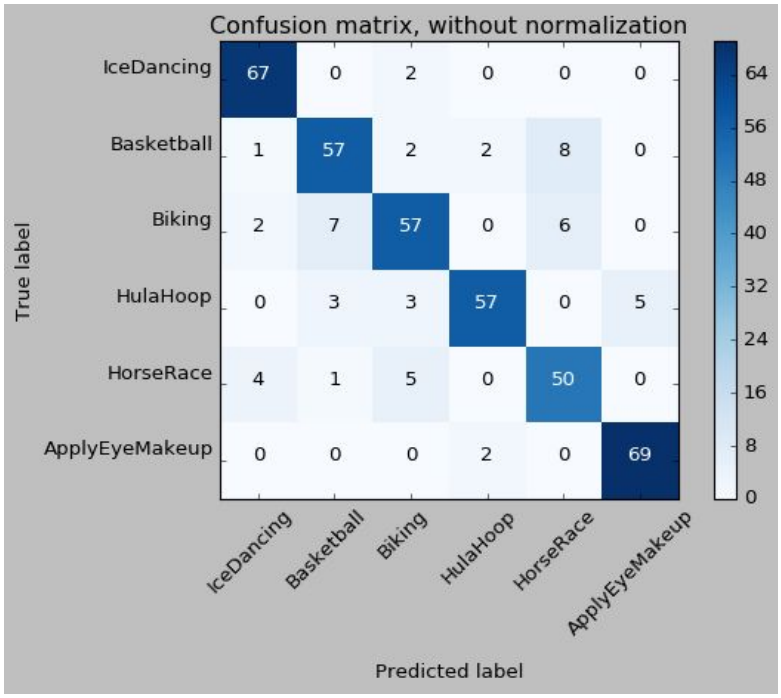
	precision	recall	f1-score	support
IceDancing	0.91	0.97	0.94	69
Basketball	0.84	0.81	0.83	70
Biking	0.83	0.79	0.81	72
HulaHoop	0.93	0.84	0.88	68
HorseRace	0.78	0.83	0.81	60
ApplyEyeMakeup	0.93	0.97	0.95	71
avg / total	0.87	0.87	0.87	410

Confusion Matrix:

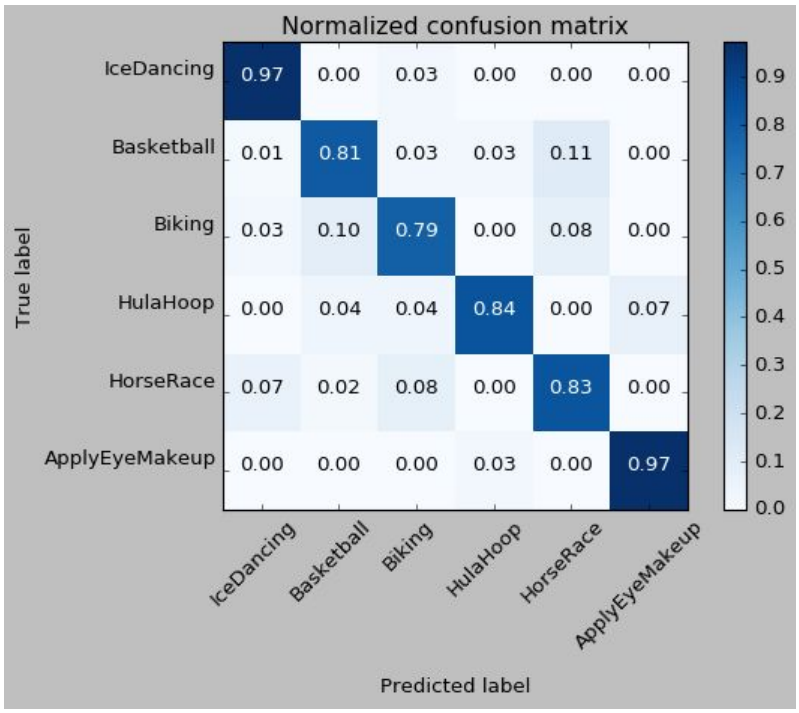
A confusion matrix is a table that is often used to describe the performance of a classification model (or "classifier") on a set of test data for which the true values are known

	Positive	Negative
Predicted Positive	True Positive	False Positive
Predicted Negative	False Negative	True Negative

Below is the unnormalised confusion matrix obtained :

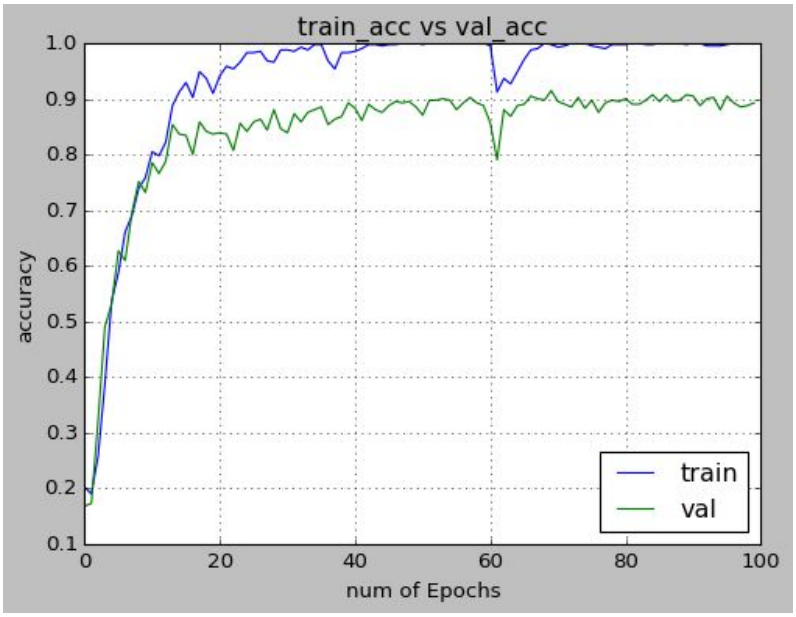


The normalised confusion matrix is :

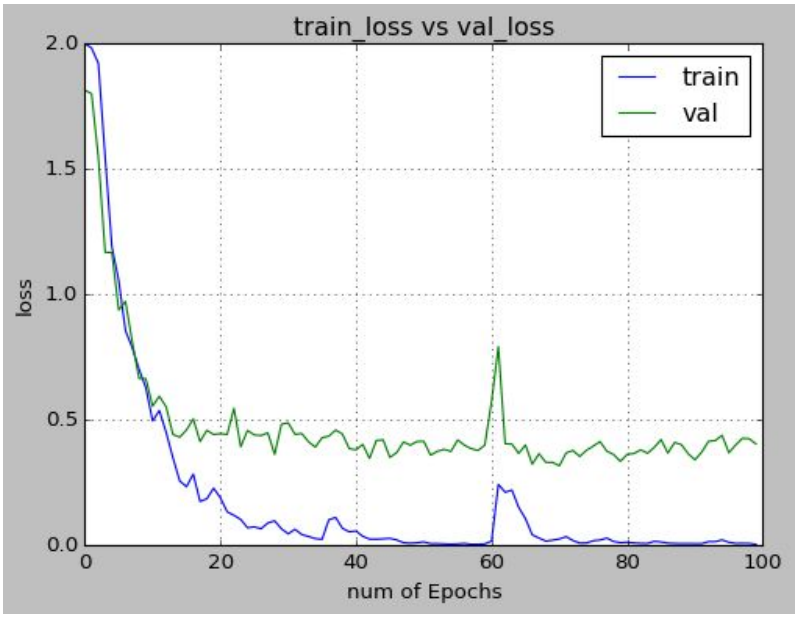


Accuracy and Loss Curves:

We have plotted the accuracy and loss for both training and validation data. The graphs are shown below:



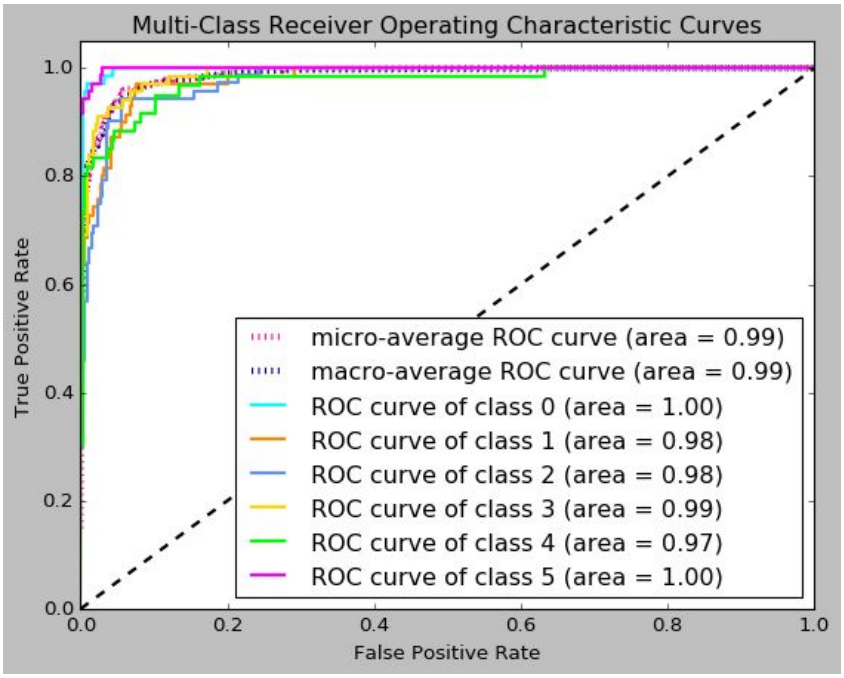
The above accuracy curves clearly show that the accuracy has improved over the epochs.



The above loss curves indicate that the loss for both training and validation data is greatly reduced over the epochs.

ROC curve:

It is one of the most important evaluation metrics for checking any classification model's performance. The ROC curve is created by plotting the true positive rate (TPR) against the false positive rate(FPR) at various threshold settings. The true-positive rate is also known as sensitivity, recall or probability of detection in machine learning. The false-positive rate is also known as the fall-out or probability of false alarm and can be calculated as (1 − specificity). Below is the ROC curve obtained on the test dataset.



Results and Interpretation:

The model is then used to predict unknown test data for each class. Shown below are some of the frames sampled randomly from the videos and here are the predictions made by the model :

Figure : 1 , 2 ,3

The above is the prediction made for a video belonging to Basketball class. When we were analysing the predictions of the model by considering different set of frames in the video each time,

we found that , for this video , the model was able to predict that it belongs to Basketball class after it sees the Basketball court as shown in the figures 1 and 2 but predicts it incorrectly as Hula Hoop in the absence of a court as seen in the figure 3.

Consider the below classification result,

Figure : 4

Figure: 5

After observing the above output, we see that for one video, the model predicted correctly that it belongs to Biking class as shown in figure 4. But for the other video, it predicted incorrectly as Hula Hoop or Basketball. This may be because the wheels of the bike and the Hula Hoop both have identical shapes which is circular as shown in figures 5.

Rest of the results can be found in Appendix.

Conclusions:

Deep neural networks are dominating the research in the field of video understanding and classification. However training deep networks on videos is difficult and computationally expensive. Thus benefiting from deep models pre-trained on images or other sources would be a better solution to explore.We have presented a complete survey of state-of-the-art techniques in video understanding.

Future work:

We want to extend our model by adding Bidirectional LSTM to retain the spatio-temporal information of the previous frames to result in better classification of the video and also extend the training set of our model with Youtube 8M dataset.

References:

1. http://www.cs.utexas.edu/~jsinapov/teaching/cs309_spring2017/readings/A_survey_on_vision-based_human_action_re.pdf
2. https://ieeexplore.ieee.org/document/6165309
3. https://ieeexplore.ieee.org/document/6909619
4. https://blog.coast.ai/five-video-classification-methods-implemented-in-keras-and-tensorflow-99cad29cc0b5
5. https://arxiv.org/pdf/1805.11790.pdf
6. http://crcv.ucf.edu/data/UCF101.php

Appendix: