A Model for Sports Activity as well as Sub Activity Recognition

Dr. Deesha Deotale and Dr. Madhushi Verma
Bennett University

Shubham Kumar Agrawal
IIT Bhubaneswar

Shaik Khalid Naveed
JNTU Hyderabad

Tummala Sai Teja
RVR JC College of Engineering

Christyl Thomas Sunny
Saintgits College of Engineering

Rohit Nuthangi
Aditya Engineeing College

Abstract

Action recognition from a trimmed video is a recent heated research topic in the world which attempts to classify different sports activities as well as sub-activities in a set of trimmed videos. This paper is an attempt to classify sub-activities as well as the main activity associated with a trimmed video. The data has been taken from the UCF 101 dataset (a part of it has been taken for prediction such as videos of cricket, long jump, and high jump). Our model to capture the dependencies between the events in a video introduces a new module that uses contextual information from the dataset to jointly describe all events and hence, the desired output.

Objective of the research

As we know that, with the explosive growth of data and computational resources, Supervised Deep Learning is being used in several scientific fields and is developing day by day. Hence if we somehow use this technique in solving problems related to Engineering, specifically the problems related to activity recognition, there will be a lot of scope of development and its effective usage such as in the fields of video surveillance,human-object interaction, etc. So, the objective of this research was to further develop some ideas which encourage the use of this technique in Engineering especially in the area of Computer Vision.

I. Introduction

The ability to analyze the actions which occur in a video is essential for a deep understanding of sports and its sub-classification. Action as well as subaction recognition in videos are two main topics in this context. In this paper, we have provided a detailed study of the prominent methods devised for these tasks which yield fruitful results for sports videos. We have adopted the UCF 101 Sports dataset, which is a dataset of realistic sports videos collected from youtube channels, as our evaluation benchmark. To provide further details about the existing action recognition methods in this area, we decompose the action recognition framework into two main steps of feature extraction and classification of activities as well as sub-activities, we overview several successful techniques for each of these steps. We studied several methods for action recognition which have shown insightful results on sports datasets which is a collection of trimmed sports videos. Finally, we discussed a few insights drawn from overviewing the action recognition methods. In particular, we argue that performing the recognition of untrimmed videos and attempting to describe an action, instead of conducting a forced-choice classification, are essential for analyzing the sports actions in a realistic environment.

II. About UCF101 dataset

UCF101 is an data set of realistic action videos widely used for action recognition tasks.It has been collected from YouTube having 101 action categories. This data set is a further extension of UCF50 data set which has 50 action categories.With 13320 videos from 101 action categories, UCF101 gives the largest diversity in terms of actions and with the presence of large variations in camera motion, object appearance , object scale, viewpoint, background, illumination conditions, etc, it is the most challenging data set to date. As most of the available action recognition data sets are not realistic and are staged by actors, UCF101 aims to encourage further research into action recognition by learning and exploring new realistic action categories.The videos in 101 action categories are grouped into 25 groups, where each group can consist of 4-7 videos of an action. The videos from the same group may share some common features, such as similar background, viewpoint, etc.The categories of action are divided into five types: 1)Human-Object Interaction 2) Body-Motion Only 3) Human-Human Interaction 4) Playing Musical Instruments 5) Sports. So,We have chosen the Sports Dataset that too for Cricket,high jump and long jump respectively.

III. Related Work

Convolutional Neural Networks(CNN) has always been a research topic since it's first introduced for the action recognition task. Since then, many Data Scientists have adopted 3D CNN model to directly extract spatial features and temporal features from raw video data. Another typical research work named C3D. Based on spatial-temporal features learned by 3D CNN , can achieve leading performance on 4 main-stream benchmarks by just using a simple linear classifier. Recently, with the appearance of large size video datasets like ActivityNet and Kinetics, some other works involving 3D CNN outperform state-of-the-art models after pretraining on a huge video dataset. As we all know, 2D

CNN model almost dominate the domain of image recognition task because it can automatically extract spatial features by performing 2D convolution, which is more accurate and efficient than traditional handcrafted features. However, experiments in research work have already validated that even very shallow 3D CNN models can outperform very deep 2D CNN models to a large extent for the action recognition task. Recently a novel Two-Stream Inflated 3D ConvNet model ,which expand convolution and pooling kernels of Inception module in GoogLeNet into 3D, achieves the best accuracy in UCF-101 dataset.However, there is still the problem of the gradient boosting as well as gradient vanishing while training the model. (especially in the region of deep saddle points in the loss curve).

IV. Our appproach

Our approach uses 2D CNN model for sub activity classification after creating different frames from the sports video dataset. Our approach uses KMeans clustering along with LSTM(Long short-term memory) networks for feature extraction and activity recognition respectively. LSTM is an improved network based on RNN(Long short-term memory), which can prevent gradient vanishing problem and gradient boosting problem during the training process. Despite the fact that LSTM Network and CNN are getting more attention in the area of action recognition, very few current research works make proper use of them. In our work, we find that CNN is better for learning temporal features and finding sub-activities for the main activity between adjacent video frames while LSTM is better for modeling high-level sequence features.

V. Methodology

The work uses the concepts of Convolutional Neural Networks(CNN) and Long Short Term Memory(LSTM) architecture.

A. Convolutional Neural Networks(CNN/ConvNet)

A Convolutional Neural Network (ConvNet/CNN) is a Supervised Deep Learning algorithm that takes in an image as an input, assigns learnable weights and biases to various aspects/objects in the image, and be able to differentiate one from the other. The pre-processing required in a CNN is much lower as compared to other deep learning algorithms. While in primitive methods filters are hand-engineered, with enough training, CNN has the ability to learn these filters/characteristics. The architecture of a Convolutional Neural Network is analogous to that of the connectivity pattern of Neurons in the Human Brain. It was inspired by the organization of the Visual Cortex. Individual neurons respond to stimuli only in a restricted region of the visual field known as the Receptive Field. A collection of such fields overlaps to cover the entire visual area.

Why ConvNets over Feed-Forward Neural Nets? An image is nothing but a matrix of pixel values. So why not just flatten the image (e.g. 256x256 image matrix into a 65536x1 vector) and feed it to a Multi-level Perceptron for classification purposes? In cases of extremely basic binary images, the method might show an average precision score while performing prediction of classes but perhaps would show almost no accuracy when it comes to complex images having pixel dependencies throughout. A Convolutional Neural Network is able to successfully capture the Spatial and Temporal dependencies in an image through the application of relevant filters(Like edge detection filter etc..). The architecture performs a better fitting to the image dataset due to the reduction in the number of hyperparameters involved and the reusability of weights. In other words, the network can be trained to understand the image better. Suppose we have an RGB image that has been separated by its three color planes — Red, Green, and Blue. There are a number of such color spaces in which images exist — Grayscale, RGB, CMYK, etc. You can imagine how computationally intensive things would get once the images reach dimensions, say 8K (7680×4320). The role of the ConvNet is to reduce the images into a form which is easier to process, without losing the critical features for getting a good prediction at the end. This is important when we are to design an architecture that is not only good at learning features but also is scalable to massive datasets.

B. Long Short Term Memory

Long Short Term Memory is a kind of recurrent neural network. In RNN output from the last step is fed as input in the current step. LSTM was designed by Hochreiter Schmidhuber.It is used for processing, predicting and classifying on the basis of time-series data.LSTM has a chain structure that contains four neural networks and different memory blocks called cells. Information is retained by the cells, and the memory manipulations are done by the gates. There are three gates –

1. Forget Gate: The information that no longer useful in the cell state is removed with the forget gate. Two inputs xt (input at the particular time) and h(t-1) (previous cell output) are fed to the gate and multiplied with weights followed by the addition of biases. The resultant is passed through an activation function which gives a binary output, e.g. If for a particular cell state, the output is 0, the piece of information is forgotten and for the output 1, the information is retained for the future purposes.

2. Input gate: The addition of useful information to the cell state is done by the input gate. First, the information is regulated using the sigmoid function and filter the values to be remembered similar to the forget gate using inputs [h(t-1) and xt]. Then, a vector is created using tanh activation function that gives an output from -1 to +1, which contains all the possible values from h(t-1) and xt. At the end, the values of the vector and the regulated values are multiplied to obtain useful information.

3. Output gate: The task of extracting useful information from the current cell state to be presented as output is done by the output gate. First, a vector is generated by applying tanh activation function on the cell. Then, the information is regulated using the sigmoid function and filter the values to be remembered using inputs h(t-1) and xt. At last, the values of the vector and the regulated values are multiplied to be sent as an output and input to the next cell.

C. Technical details and functionality

Initially, we started with Data pre-processing part, we made a CNN based model first to predict the main activity class of the datasets. If we look closely at the dataset, every video has a tag attached to it which refers to the class it belongs to (Eg. a trimmed video of cricket bowling has named as xyz/cricketbowling.avi). We utilized while converting video to frames along with the class/tag names. So, In our first CNN model, we created a list of strings that has the class of every video it belongs to. We did it in a loop(because we had many videos) using the Split method of the python. We then created a data frame containing columns as video names and class/tags it belongs to. We classified this data frame as a train and test for training and validation purposes.We used 30% of the data as test data and remaining70% as the training set.After this, we created the frames from each video in both train and test data by using the OpenCV library of the python. We captured the frames using a suitable frame rate and saved it using cv2.imwrite() function. This was done for each class of the video (we did it for cricket bowling,long jump, and high jump respectively.). To increase the amount of data and thus, to reduce overfitting, the concept of Data Augmentation has been utilized here. Data augmentation is basically a master plan used to increase the amount of data by using techniques like cropping, flipping, zooming etc. Data augmentation makes the model more robust to slight variations, and thus prevents the model from overfitting which we have said earlier. It is neither practical nor efficient to store the augmented data in memory, and that is where the ImageDataGenerator class from Keras (Also included in the TensorFlow's high level api: tensorflow.keras) comes into play. ImageDataGenerator generates batches of tensor image data with real-time data augmentation. The best part of it is it's just one line of code! The output images generated by the generator will have the same output dimensions as the input images. We got the images as (256,256,3) where (256,256) represents height and width of the image and 3 represents the number of channels.(Number of channels for RGB is 3 and that of greyscale image is 1. It is just like a third dimension to the image matrix). Now is the time to do modelling. In the ConvNet part,we introduced why we were using it over normal feed forward network.(Actually, ConvNet has many layers to reduce the hyperparameters of the model building and thus reducing the computational cost. ConvNet has many layers such as Convolutional layer where we do feature mapping using various filters or kernels.Then we have pooling layers to further reduce the image dimensions and hence, hyperparameters. Lastly, we have a fully connected layer where we represent each pixel of the matrix as a feature for training). Then making a normal feed forward network involved hidden layers where we used ReLu as an activation function.We used ReLu because of the non-saturation of its gradient, which greatly accelerates the convergence of stochastic gradient descent compared to the sigmoid / tanh functions. Since we have more than one-two classification at the output,we used the Softmax activation function instead of Sigmoid (Sigmoid is used when we have binary classification problem.). The use of "Adam"optimizer (which is a combination of RMSprop and Stochastic Gradient Descent with momentum) during training because "Adam" optimizer always works better (faster and more reliably reaches a global minimum) when minimizing the cost function in training neural networks. Use of dropout also gives the advantage to reduce overfitting.

Since ConvNet doesn't take care of spatial features which do exists in a dataset containing videos, we tried building an LSTM model over the CNN model.So,In LSTM,we have taken the output of the dense layer(second last layer) of previously built ConvNet model, We split the data into training and test feature set, used 10-time frames from every video, then we got 64 spatial features for training as well as test dataset. After building LSTM model over it by using the same hyperparameters as mentioned above in ConvNet model building, We got a better model than ConvNet ,not in the sense of accuracy but in the sense of spatial featuring . After this, we did hyperparameters tuning like using Kullback–Leibler divergence instead of binary cross-entropy (in statistics, the Kullback–Leibler divergence is a measure of how one probability distribution is different from a second, reference probability distribution.It is also called relative entropy). We also utilized the K Means clustering algorithm to do the temporal featuring of the dataset. As we know,KNN is an unsupervised clustering algorithm used for classification purposes (like image segmentation that we are exploiting here). For temporal featuring, we built another CNN model where we did segmentation of all the spatial features manually along with KNN Model. In KNN, by using different metrics like Inertia(inertia calculates the sum of distances of all the points within a cluster from the centroid of the same cluster.), Dunn Index(It takes into account the distance between two clusters) and Silhouette Index (It measures how similar the data points are within a cluster, It can range between -1 and 1. It is the most robust way to determine the number of clusters compared to the Inertia and the Dunn index). Based on this, we take the output of KNN algorithm as one more feature to the input and build the LSTM model over everything that we have covered. We achieved a remarkable difference between the outputs of the normal sub-classification CNN model and the LSTM model which has been discussed under the result section.

VI. Results

A. Datasets and Implementation

The proposed network in our paper has been trained and evaluated on three sports datasets which is a part UCF-101 human action recognition dataset. 175 trimmed videos of sports activities such as Cricket bowling, long jump, and high jump have been used out of which 122 videos were used for training and 53 as a test dataset.

Table 1 Accuracy score for models used

	Model	Training Accuracy	Test Accuracy
0	ConvNet(for subactivity recognition)	.95	.70
1	LSTM	.88	.52

VII. Conclusion

We can conclude that Convolutional Neural Networks gives better test accuracy while performing subactivity recognition and LSTM(Long-Short Term Memory) model lags behind the ConvNet model in both Training accuracy and test accuracy. In the future, we will keep making efforts to advance the performance of our model with more benchmark datasets.

Acknowledgments

We would like to express our deep and sincere gratitude to supervisor Dr. Deesha Deotale and Dr. Madhushi Verma of Bennett University for the endless encouragement and supervision throughout the internship.

References

Xianyuan Wang, Zhenjiang Miao, Ruyi Zhang and Shanshan Hao:I3D-LSTM: A New Model for Human Action Recognition,Xianyuan Wang et al 2019 IOP Conf. Ser.: Mater. Sci. Eng. 569 032035

Khurram Soomro, Amir Roshan Zamir and Mubarak Shah, UCF101: A Dataset of 101 Human Action Classes From Videos in The Wild, CRCV-TR-12-01, November, 2012.

Itsaso Rodríguez-Moreno ,José María Martínez-Otzeta, Basilio Sierra ,Igor Rodriguez and Ekaitz Jauregi:Video Activity Recognition: State-of-the-Art,Received: 13 June 2019; Accepted: 9 July 2019; Published: 18 July 2019

Zhaofan Qiu,Ting Yao,Chong-Wah Ngo,Xinmei Tian and Tao Mei: Learning Spatio-Temporal Representation with Local and Global Diffusion

https://en.wikipedia.org/wiki/Kullback-Leibler-divergence

https://towardsdatascience.com/tagged/image-classification