MediaEval 2019: LRCNs for Stroke Detection in Table Tennis

Siddharth Sriraman, Srinath Srinivasan, Vishnu K Krishnan, Bhuvana J, T.T. Mirnalinee
SSN College of Engineering, India
(siddharth18150,srinath18205,vishnukrishnan18200)@cse.ssn.edu.in,(bhuvanaj,mirnalineett)@ssn.edu.in

ABSTRACT

Recognizing actions in videos is one of the most widely researched tasks in video analytics. Sports action recognition is one such work that has been extensively researched in order to make strategic decisions in athletic training. We present a model to classify strokes made by table tennis players as a part of the 2019 MediaEval Challenge. Our approach extracts features into a spatio-temporal model trained on the MediaEval Sports Video Classification dataset to detect the move made.

1 INTRODUCTION

In this paper we have discussed our method to classify strokes in a dataset consisting of various strokes performed by table tennis players during games. [5] The dataset consists of 20 different shot techniques which the classification is based upon, and these moves are shot in natural conditions. Research into this specific domain can improve athletic performance by computer-aided analysis of moves.

The main challenge we faced was to train a model that would take into account the temporal features carefully. We implemented an existing spatio-temporal model for this problem and discussed our results with the given dataset. We applied a Long-term Recurrent Convolutional Network (LRCN) [1]. The work done mainly focusses on testing the implementation and feasibility of using this model in such a paradigm.

2 RELATED WORK

Extensive research has been carried out in the field of action recognition in videos which usually tend to focus on recognising a large number of actions using spatio-temporal models. These videos usually last longer compared to table tennis strokes which are relatively brief.

Our focus is on sports video classification, specifically table tennis. While 2D ConvNets like VGG16[7] have produced outstanding results for image classification, video classification research has focussed on importing this to the temporal dimension using 3D ConvNets [4] and Long-term Recurrent Convolutional Networks.

The application of action recognition to table tennis for stroke detection [6] has been researched, and the closest work uses a 3D ConvNet model along with optical flow data. Our approach does not use optical flow data to detect the moves and instead directly uses the frame sequences. In spite of this reducing the complexity of the model, we found using larger batch sizes more demanding to run.

3 APPROACH

The approach we used had to take into account temporal information in the frames efficiently due to the moves having very subtle differences. The low inter-class variability is the main obstacle we had to face. A vanilla Convolutional Neural Network (CNN) with a rolling average prediction works well enough for highly spatial data since each class is very distinct from the other. Here, due to low inter-class variability the move could not be classified from just a single frame, so we decided to implement a spatio-temporal model. Our basic idea was to implement a Long-term Recurrent Convolutional Network (LRCN) while trying multiple architectures for the CNN used in it and try to find the best hyperparameters to ensure the model performed well on moves of shorter duration.

3.1 Data Pre-processing

Time distributed models take fixed input sizes for each minibatch. The frames were downscaled to 80x80. Since the frame rate of the data (120 fps) is very high, the initial models used 25 frame long sequences for each move. The sequences are then scaled using mean and standard deviation before sending them into the model. A smaller batch size of 64 was taken to train the model as the size of the data was large.

The moves are of varying duration with some being significantly short-lived while others more drawn out. This meant we had to ensure the extracted frames had information on the entire move in it, irrespective of the duration. Initially, We tested a model that took in 25 frames from each sequence sampled at different rates. The rate used was 5 frames as rates higher than this showed significant skips in hand movement from one frame to another. Then we used variable sequence lengths which were padded to ensure the data sent to the network are of uniform lengths. We also tested a variable rate for each video (based on the duration of the move) to ensure each training example had a fixed sequence length, but this did not lead to any improvements.

3.2 Model

Our approach uses an RGB frame sequence of the move sent to a Long-term Recurrent Convolutional Network to classify the frames. The Convolutional Neural Network implemented is Time distributed, meaning the same CNN architecture is applied to each frame in the sequence independently, resulting in a collection of outputs whose length is the frame sequence length. The architecture is a modified version of VGG16 that implements *Batch Normalization* and *dropout* to address overfitting. The parameters were initialised using Glorot initialisation [2]. This sequence is then sent to a vanilla Long Short Term Memory (LSTM) layer [3].

MediaEval'19, 27-29 October 2019, Sophia Antipolis, France

Another variant of this model used a second LSTM layer after that. The LSTM outputs are sent to a standard fully-connected network to map it to the final output of 20 classes namely Defensive Backhand Backspin, Backhand Block, Backhand Push, Forehand Backspin, Forehand Block, Forehand Push, Offensive Backhand Flip, Backhand Hit, Backhand Loop, Forehand Flip, Forehand Hit, Forehand Loop, Serve Backhand Backspin, Serve Backhand Loop, Serve Backhand Sidespin, Serve Backhand Topspin, Serve Forehand Backspin, Serve Forehand Loop, Serve Forehand Sidespin and Serve Forehand Topspin.

Hyperparameters adopted by our approach is listed in Table 1 and training metrics shown in Table 2. Overfitting was a major problem. Usage of multiple dense layers before the final activations caused overfitting. Another reason was due to the depth of the CNN used. Kernel Regularisers were used for all the CNN layers. Using dropout with a 0.3 to 0.5 probability (to retain) for the LSTM layer showed best results.

Table 1: Training Hyperparameters

Hyperparameter	Value
Learning Rate	1e-3
Weight Initialisation	Glorot
CNN Activation Function	ReLU
Final Activation Function	Softmax
Decay (in Adam)	1e-6
Loss Function	Categorical Crossentropy
Regularisation Parameter	0.001
Batch Size	64
Dropout (retention probability)	0.5
Decay (in Adam)	40/354
K-Fold Crossvalidation Splits	8

Table 2: Training Metrics

Metric	Value
Number of epochs	30-40
Validation loss	1.66
Validation accuracy	0.829
Train loss	1.63
Train accuracy	0.839

4 RESULTS AND ANALYSIS

The model was only able to classify 40 out of the 354 moves (11.3%) showing that using LRCN was not a viable option for this dataset. The training set was skewed towards certain classes more than others and Stratified K-Fold Cross Validation was used to ensure the best train and test splits were chosen. The per-class accuracy data shows that the model learnt certain moves very well, and could not learn the moves which had lesser data to work with. We also observed that short and swift actions like table tennis moves are not efficiently learnt by sequence models like LSTMs and that they require data of other forms, in addition to direct RGB frame pixel data.

Table 3: Test Run

Test Run Accuracy	Ratio
0.1130	40/354

The different variants of the model did not show very significant changes in results, with the best run getting a 11.3% accuracy (Table 3). The only difference in the other run we submitted was allowing to the model to train for 5 more epochs, the training results we obtained were very similar to this run. All other hyperparameters used were kept the same. Using only the RGB sequence information (without optical flow data) the model could not generalise on the specific differences between classes of moves on the test set, leading to a low test accuracy. We observed that the model could not predict certain moves at all, while performing reasonably well on other classes of moves.

A closer analysis shows that the model fails to distinguish between the moves belonging to a specific class (such as Serve, Defensive, Offensive) as the differences are very intricate. The model tended to prefer certain moves significantly more than others on the test set, which arose due to the distribution of the training set. Using uniform amounts of data to work with resulted in the number of examples to train on being very low. The model did face significant overfitting issues while training even when the complexity was reduced and regularisation was employed. The difference in accuracy on test and validation data might be due to the frequency of the different classes on the test set being different from the training and validation set.

5 DISCUSSION AND OUTLOOK

The main insight we gained is that data with classes having very low variability require more complex models and features to be extracted. We learnt that data of this kind cannot be generalised even by models known to work well with spatio-temporal data. This was due to the model overfitting and learning the intricacies of the moves too deeply from the training data, hence it failed to reproduce the results on the test data.

We also gained an understanding of why swift action moves are difficult to classify, the limitations of sequence models in this regard and how temporal features with low variability cannot be differentiated easily. Working on the MediaEval Sports Classification dataset helped us to grasp why problems under this domain are critical to solve and how we should choose to approach data of this kind in the future.

REFERENCES

[1] Jeff Donahue, Lisa Anne Hendricks, Marcus Rohrbach, Subhashini Venugopalan, Sergio Guadarrama, Kate Saenko, and Trevor Darrell. 2014. Long-term Recurrent Convolutional Networks for Visual Recognition and Description. (2014). arXiv:arXiv:1411.4389

[2] Xavier Glorot and Yoshua Bengio. 2010. Understanding the difficulty of training deep feedforward neural networks. In *Proceedings of the Thirteenth International Conference on Artificial Intelligence and Statistics (Proceedings of Machine Learning Research)*, Yee Whye Teh and Mike Titterington (Eds.), Vol. 9. PMLR, Chia Laguna Resort, Sardinia, Italy, 249–256. http://proceedings.mlr.press/v9/glorot10a.html

[3] Sepp Hochreiter and Jürgen Schmidhuber. 1997. Long Short-Term Memory. *Neural Comput.* 9, 8 (Nov. 1997), 1735–1780. https://doi.org/10.1162/neco.1997.9.8.1735

[4] S. Ji, W. Xu, M. Yang, and K. Yu. 2013. 3D Convolutional Neural Networks for Human Action Recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 35, 1 (Jan 2013), 221–231. https://doi.org/10.1109/TPAMI.2012.59

[5] Pierre-Etienne Martin, Jenny Benois-Pineau, Boris Mansencal, Renaud Péteri, Laurent Mascarilla, Jordan Calandre, and Julien Morlier. 2019. Sports Video Annotation: Detection of Strokes in Table Tennis task for MediaEval 2019. In *Proc. of the MediaEval 2019 Workshop, Sophia Antipolis, France, 27-29 October 2019.*

[6] P. Martin, J. Benois-Pineau, R. Péteri, and J. Morlier. 2018. Sport Action Recognition with Siamese Spatio-Temporal CNNs: Application to Table Tennis. In *2018 International Conference on Content-Based Multimedia Indexing (CBMI)*. 1–6. https://doi.org/10.1109/CBMI.2018.8516488

[7] Karen Simonyan and Andrew Zisserman. 2014. Very Deep Convolutional Networks for Large-Scale Image Recognition. (2014). arXiv:arXiv:1409.1556