International Journal of Innovative Research in Engineering
Volume 4, Issue 5(September-October 2023), PP: 30-38.
www.theijire.com ***ISSN No: 2582-8746***

Detection of Cricket Shots Using Deep Learning

Manan Pruthi[1], Ashish Katyal[2], Sanyam[3], Rishabh Semwal[4], Vijay Kumar[5]
[1,2,3,4,5]Department of Computer Science and Engineering, Bharati Vidyapeeth's College of Engineering, New Delhi-110063,India

How to cite this paper:
MananPruthi[1], AshishKatyal[2], Sanyam[3], RishabhSemwal[4], VijayKumar[5] "Detection of Cricket Shots Using Deep Learning", IJIRE-V4I03-30-38.

https://www.doi.org/10.59256/ijire.20230405004

Abstract: *Classifying the different types of bats hots played in cricket has always been a challenging task in the field of cricket indexing .Identifying the type of shot bats man played during a match is a crucial aspect that has not been thoroughly studied. This information can be used for context-based advertisements for cricket viewers, creating sensor-based commentary systems, and coaching assistants. However, manually identifying the different hots from video frames is difficult due to the similarity between them. This project presents a new approach for recognizing and categorizing different crickets hots by using state-of-the-art techniques such as saliency and optical flow to capture both static and dynamic information, and Long Short Term Memory (LSTM) for representation extraction. Additionally, a new data set of120 videos has been introduced to evaluate the performance of the model, with 4 classes of shot search having 30videos.Themodelachievedanaccuracyof83.34%for the four classes of crickets' hots.*

Key Word: *Cricket, Convolution Neural Network, LSTM, Media Pipe*

I.INTRODUCTION

Cricket is a game played with a bat and ball and various technologies are used to visualize and coach it. However, recent research has not yet produced satisfactory results for detecting shots in the game. The challenge lies in extracting features from video footage captured from different cameras. Some methods utilize information such as field position, camera position, boundary limits, commentary, and texts. However, machines are unable to visualize moothly like humans do. The issues involved in detecting shots include recognizing the movements of the human body parts, the cricket bat and the ball swing. Machines and etect human body parts in images, but tracking and estimating he movement of these body parts is difficult using a single video. Semantic analysis for cricket broadcasting video requires alarge dataset, but its accuracy level is still not satisfactory. Action classification based on templates is used to distinguish between different sport activities, but it is challenging to classify cricket shots due to the obscuration of the bat and body parts. This study proposes a unique approach to classify cricket shots by using motion vector detection for each frame using optical flow and transforming the vectors intoangles.

The recent surge in multimedia content, such as images and videos, has led to a boominre search cross various disciplines, including sports and athletics where a vast amount of data is available for analysis. Cricket, as a globally popular sport, is widely broadcasted, providing a variety of camerashotsandeventsforanalysisandestimation.Classifyingdifferent types of cricket shots is a difficult but crucial a spect of indexing cricket games. There are various standard cricket bat strokes such as Late Cut, Pull, Cover Drive, Leg Glance,etc., that are played during a match and sometimes with slight variations. Inorder to improve abats man's game and minimize deviation from the standard technique, shot detectional gorithms are highly valuable. The difficulty in recognizing different cricket shots lies in extracting unique features and cues from a limited number of data frames, whichcallsforthedevelopmentofautomatedsystemsthatcaneffectivelydetectcricketshotsfromreal-timevideos. Classifying the type of bat stroke played in a cricket match is a difficult task. The success of a shot is not always guaranteed, making it challenging to accurately determine the intended shot played. Factors like occlusion, the small size of the ball ,and the fast pace of the game only add to the complexity. Sifting through the datato find only the relevant information is a time-consuming and challenging process, requiring a large dataset.

1.1 Problems in Current Scenario

One of the major hurdles is detecting the movements and posture of body parts from just a single video. Furthermore, semantic analysis of cricket broadcasting video requires a big dataset, with accuracy being a concern. Distingu is hing between different ports activities relies on template-based action classification which makes it hard to classify cricket shots because of the occlusion of the bat and body parts.

Fig1: *Various cricket shots play enduring match[15]*

II.OBJECTIVES

The task of categorizing different types of batting shots in a cricket game has been a challenging task in the indexing ofcricket. Knowing the type of shot played by a batsman in a cricket match is an important and untapped aspect in this field.The goal is to introduce a new method for recognizing and classifying various bats hoots played in cricket.

In this project, the aim is to identify the cricket shot a bats man is playing as it happens. Previously, other projects have succeeded in doing so with recorded videos, but this project focuses onusingreal-time videos captured from the front camera.

This breakthrough can lead to the creation of an auto matic commentary system, a significant advancement for the sports industry. Additionally, it can be used to develop an automatic highlight generation system and provide a bats man with valuable statistics to improve their game play.

III.LITERATUREREVIEW

Recently, there has been an increase in research in the field of sports due to its commercial value and widespread popularity. Different methods have been developed to classify various actions in sports videos. Some of these methods have been used in the analysis of cricket videos, categorizing different types of shots played in the game.

Classification of eight cricket shots was performed using deep convolution neural networks (CNNs) in [1]. A 2D CNN, 3DCNN, and an RNN with LSTM were utilized to categorize the shots. The classification was improved through transfer learning from a pre-trained VGG model. The highest accuracy of 90% was obtained with 800 shots, but the hook shot had thelowest accuracy at 85%. In [2], a new method for classifying cricket shots was proposed using motiones timation. The classification was done based on eight different an gleranges.

The highest accuracy of 63.57% was recorded for the off drive, while the hook shot had the lowest accuracy of 53.32%.In [3], wearable technology was utilized to categorize cricket shots with the goal of creating a quality evaluation system for these shots. The classification was based on specific aspectssuchasfootpositionandshotdirectionusinghierarchicalrepresentation.Fivelevelsofgroupingactionsweredetermined using DT, SVM, and k-NN algorithms. The best classifiers per level gave a class-weighted F1 score of 88.30%.This work only used videos collected from a training session, not actual match videos. [4] Presented a CNN approach to classify 6 different cricket shots based on a custom-developed dataset. [5] Introduced a model to identify straight drive, cover drive, and lofted on drive for both left-and right-handed batsmen using a pre trained CNN and multiclass SVM. The model achieved 83.098% accuracy for right-handed shots and65.186% accuracy for left- handed shots. M is classifications occurred for some shots such as left cover drive and straight drive due to the limited dataset.

Several efforts have been made to identify different events in cricket videos [6,7]. Harikrishnaetal [6] presented a technique that divided cricket video into shots and used SVM to classify these shots into four events, including run, four, six, and out. The recorded accuracy was 87.8%, but it had low accuracy for the six events. Another study was conducted by Kolekar et al. [7]. They presented a hierarchical framework for identifying different events in cricket videos, including excitement detection, replay detection, field-view detection and classification, close-up detection and classification, and crowd classification. They mentioned that there was a problemwithmisidentificationneartheboundariesoftheshots.Premaratneetal.[8]proposed as tructured method for recognizing different events in a full cricket match. They used k-nearest-neighbor, sequential-minimal optimization, decision-tree, and naive Bayes classifiers for event detection. However, the authors noted the need for human interpretation to remove replay frames and highlights between deliveries Several studies were carried out for automatically extracting highlights from cricket videos [9-14]. In one study [9], video and audio features were combined. The researchers started by using rule-based induction to detect exciting audio clips, followed by a video summarization step using a decision tree. This method achieved an average accuracy of 90%.In [10] ,the authors suggested a hierarchical approach for automatically generating highlights from cricket videos. They used association-rule mining to extract semantic concepts and noted the importance of features such as detecting the umpire's gesture and estimating camera motion as future work. In [11], methods for summarizing cricket games based on video-shot detection, sound detection, and scoreboard recognition were proposed. In [12], a new method for identifying highlights in sports videos was presented. The approach use dacolorhistogramandah is

to gramo for ientedgradients.[13]introduced auniquesummarizationstrategy that extracted boundary and wicket events based on bowling and score changes, and removed adsandr eplay events throughcalculation of the temporal derivative. In [14], a hierarchical approach was proposed forreal- time extraction of highlightsin sports videos, using a combination of motion and colorfeatures within a multi-spatial framework. Ring is et al. [15]proposed a method to automatically detect bowlerrun- upsequences (BRSs) in cricket to generate highlights, using the oriented fast and rotated brief (ORB) method. However, their proposed method has some limitations, such as the requirement of training at the start of the match, detection of players as key points only, no use of orientation parameters, and only training and detection of BRS views. Rafiq et al.proposed a method in [16] to generate video summarization in sports videos using transfer learning. The approach use da CNN-based Alex net model and achieved anaccuracyof99.26%when tested on a smaller data set of only crickets cenes. So far, only a few experiments involving sports other than cricket have been conducted. Steel setal. [17]conducted experiments in sports other than cricket and propose da method that uses a convolution neural network and accelerometer data to recognize 9 different badminton actions. The accuracy of the proposed method was lower compared tothe average accuracy, with a weighted accuracy of 93% and96% for sensor placement sint hewrist and upperarm,respectively.Rangasamy etal.[18] proposeda model that used the pre-trained VGG16 network to classify four types of hockey activities: freehit,goal ,penaltycorner, and long corner. After 300 epochs, the model achieved an accuracy of98%, but it struggled to distinguish between a free hit and along corner because they had similar visual patterns with regards to player positioning and appearance. In [19], Junju net al. put forth a method to identify three basketball actions: shoot, pass, and catch. The model utilized an adaptive multi-levelclassificationtechniqueandachievedanaccuracyof92.3%. However, this study was limited in scope, covering nly a small number of examples.In[20],a method for recognizingfine-grained video actions in basketball games was proposed by Guetal. The study focusedon 3broadactions, dribbling, passing, and shooting, which were further divided into 26 fine-grained actions. However, the a vera geprecisio no fsubclasses within the broadcate goriesvarie dsignificantly duet otheir uniquecharacteristics.

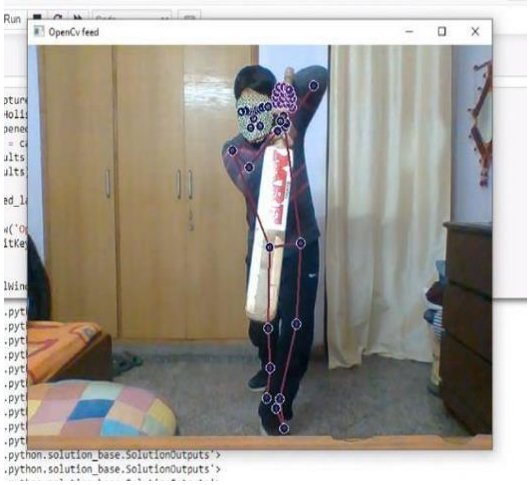
IV.METHODOLOGY

4.1 Data set Description

Since there was no publicly available dataset for individual cricket shots, a data set was created manually by recording various shots in real time using Open CV and a front camera. The data set was constructed to meet he desired specifications. A series of cricket shots were recorded with a front-facing camera, which were then divided into four shot categories ,each with 30 video clips. The final data set consisted of around a total of 120 cricket shot recordings. The video clip distribution fore a chcategory is described in Table 3.1.The experiments were conducted using an AMD Ryzen5 Hexacore5500U processor and NVIDIAGeForceGTX1650GPU,and were run using Jupyter Note book as the software.

CRICKETSHOT	NO.OFVIDEOS
Pre-Stance	30
Stance	30
StraightDrive	30
Pull Shot	30

TABLE1: Total Data Set Count Under Each Class Label



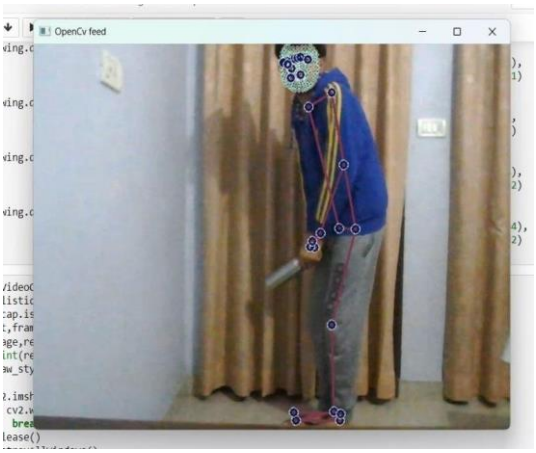


Figure 2: Examples from the dataset, (a) Straight-drive, (b) Stance

4.1 Media Pipe

Object detection is a widely used and popular application in the field of computer vision .Many object detection models are applied globally for various specific purposes. Many of the remodels are designed to best and alone solutions to individual computer vision problems and are fixed to particular use case. Combining all of the set asks into a single end-to-end solution, in real- time,is basically the job of Media Pipe.

Media Pipe is an open-source, cross-plat form machine learning framework that allows for the creation of complex, multi-modal machine learning pipe lines. It can be used to develop advanced machine learning models for tasks such as face detection, multi-hand tracking, object detection and tracking, and more. Media Pipe acts a same diator for imlementing models on any platform, freeing developer's to focus on experimenting with models rather than the underlying system.

4.1.1 Media Pipe Holistic

Media pipe Holistic is a pipeline that has been optimized to track faces, hands, and poses all at once, which is referred toas "holistic tracking." This pipeline enables detection of hand and body poses along with facial landmarks. It is mainly used for detecting faces and hands and extracting key points for use in computer vision models.

4.1.2 Detect face, hand, pose land marks and extract key points

The Media Pipe Holistic pipeline brings together optimized models for human pose, face, and hand detection, allowing for the simultaneous tracking of hand and body poses along with face landmarks. This pipeline can be used for various applications that require human pose estimation from video, such a smea suring physical exercises, recognizing sign language, and controlling full-body gestures. These capabilities can be applied in fields such as yoga, dance, and fitness.

The Media Pipe Pose pipeline uses machine learning to accurately track body pose, including 33 3D landmarks that consist of x, y, z and visibility coordinates. The Media Pipe Face Mesh, on the other hand ,is capable of estimating 4683D face landmark sin real- time. The Face Landmark Model performs face landmark detection from a single camera and expresses the X-and Y-coordinates as normalized screen coordinates, with the Z coordinate being relative and scaled relative to the X coordinate based on the weak perspective projection camera model.

Media Pipe Hands is a machine learning-based solution for tracking hands and fingers with high precision. It uses ML to identify the 21 3D landmarks of a hand from just one frame, and each hand is represented as a list of 21 landmarks, each with x, y, and z coordinates.

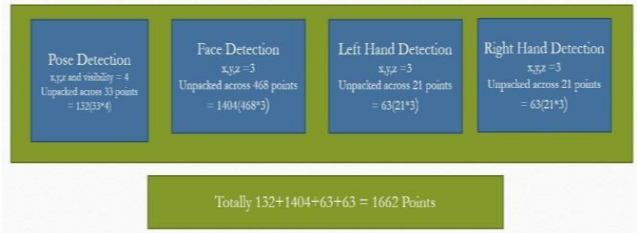


Fig3: Model Architecture

A video is divided into 30 frames, and the coordinates of 1662points are calculated for each frame. The 30 frames from one video are then sent to a model for prediction of a cricket shot action (Pre-Stance, Stance, Straight Drive, Pull Shot) based on these 1662 points. The input for the model is 30 sets of 30frameseach,with 1662points perframe,foratotalof30*30*1662 per action.

4.2 Pre- processing of Data

Initially, a large Num Py array was created to hold all the data, which is a common first step in the pre-processing of imported data before building a model. So, we have 4 actions labeled as:
[Pre-Stance:0,Stance:1,Straight Drive:2,Pull Shot:3]

The data set consisted of 120videos of cricket's hoots, each action contributing 30 videos. Each video had 30 frames and each frame had 1662 key-points, all of which were stored in a single Num Pyarray.

Therefore, the final result was an array of 120 arrays, each with 30 frames, and each frame having 1662 values which represents the key points. This resulted in an array shape of (120,30,1662).

Thelabelarray,containing120values,with30foreachaction,wascreated.TheTrueCategoricalfunctionwasutilized to convert the label array into a binary class matrix As a result, the Pre-Stance was depicted by [1,0,0,0] instead of0, Stance was represented by [0,1,0,0] instead of 1, Straight Drive was represented through [0,0,1,0] instead of 2, and Pull Shot was shownvia [0,0,0,1]insteadof3.

The dataset was divided into training and testing sets using the train_test_split function. The training partition consists of 95%of the total data, while the test partition consists of 5% of the data.

4.3 LSTM Model

LST Ms are a type of Recurrent Neural Network that is designed to handle sequence prediction problems. They work by using the output from a previous step a sin put for the current step, allowing them tolearnthe ordered pendencie sinase quence. LSTM s were developed by Hochr eiter and Schmidhuber. The solution tackles the problem of long-term dependency in RNNs, where the network struggles to recall information from its memory but performs better in predicting based on current input. RNNs struggle to predict words from memory as the gap length increases, whereas LSTMs can store information for a longer time and are therefore more effective for time-series data processing, prediction, and classificationtasks.LSTMisdistinctfromtraditionalfeed-forwardneural networks due to its feedback connections, which allow it to process not only individual data points (e.g. images) but also continuous sequences of data (such as audio or video). This makes LSTM suitable for tasks such as unbroken handwriting recognition and speech recognition.

4.3.1 Structure of LSTM

The LSTM is composed of four neural networks and many memory units called cells arranged in a chain structure. A standard LSTM unit comprises a cell, an input gate, an output gate, and a forget gate. The input, output, and information flow into and out of the cell is regulated by the three gates,allowing the cell to retain values over extended periods of time. The LSTM model is well-suited to classify, examine and for ecast time-series data of indefinite length.
The cells retain information, while the gates control memory. There are three entrances:

Input Gate:

The LSTM decides which of the input values should affect the memory ,using the sigmoid function to determine whether to pass on 0or1values and the tanh function to assign significance to the input data on a range of -1to 1.

$$i_t = \sigma(W_i.[h_t - 1, x_t] + bi)$$
$$C_t = \tanh(W_C.[h_t - 1, x_t] + b_C)$$

Forget Gate:

The LSTM determines which details to discard from the memory block, as decided by the sigmoid function. For each element in the cell state at time Ct-1, it considers the previous state (at time ht-1) and the current input (at time X t), and outputs a value between 0(to discard)and 1 (to keep).

$$f_t = \sigma(W_f.[h_t - 1, x_t] + b_f)$$

Output Gate:

The output is determined by the block's input and memory, with the sigmoid function controlling which 0or1 values are permitted to pass and the tanh function regulating which values can pass through 0 or 1. The tanh function also assigns importance to the input data on a range of -1 to 1 and multiplies this with the output from the sig moidfunction.

$$O_t = \sigma(W_o[h_t - 1, x_t] + b_o$$
$$h_t = o_t * \tanh(C_t)$$

Their current neural network employs long short- term memory (LSTM) blocks to give context to how the software handles inputs and produces outputs. By using a design inspired by short-term memory processes to construct long-term memory, the unit is known as a long short-term memory block. In natural language processing, the sesys tem sarewidelyutilized. **For instance,** the recurrent neural network utilizes long short-term memory blocks to examine a single word or sound in the context of the other sinase quence,as thememorycanassistinthefilteringandclassificationofspecifictypesofinformation. Overall, LSTM is a well-established and widely used concept in the designofre currentneural networks.

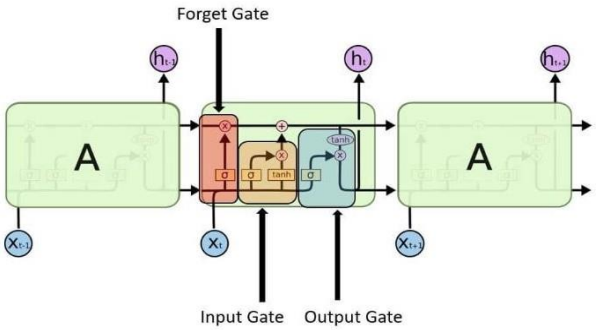


Fig4: *Structure of LSTM [11]*

4.1.1 LSTM Networks

Recurrent neural networks are composed of a series of repeating neural network modules. In conventional RNNs, this repeating module has a straightforward structure, such as a single tanh layer.The output from onetime step becomes the input for the next, which is referred to as being "recurrent." A teach element in the sequence, the model takes into account not only the current input but also its knowledge of previous inputs. The cell state, depicted by the horizontal line at the top of the diagram, is a critical component of LSTMs.

The cell state operates similarly to a conveyor belt in some ways, with only a few small line a inter actionsa long its length. Data can easily flow down it without being changed. The LSTM can choose to remove or add information to the cell state, which is precisely regulated by structures known a sgates. Gates are a mechanism for selectively letting information pass through and a recompress do fa sigmoidneuralnet work layer and a point wise multiplication operation.

The sigmoid layer generates numbers between zero and one, showing how much of each component should be permitted to pass. A value of zero means "nothing" should be allowed, while a value of one means "everything" should be allowed.An LSTM has three of these gates to control and manage the cellstate.

4.3.2 LSTM Cycle

The LSTM process consists of four phases. The forget gate identifies information to be discarded from the previous step. The input gate and tanh identify new information for updating the cell state. The information from the previous two gates is then used to update the cells tate.

The LSTM cycle is comprised of four steps, including the identification of information to be forgotten using the for getgate, and the update of the cell state using the input gate and tanh. The output of an LSTM cell is passed through a denselayerandthenprocessedthroughthesoftmaxactivationfunctioninthe output stage.

4.1.1 Bidirectional LSTM s

Bidirectional LSTM s are an improvement on traditional LSTMs, which is often discussed. In a Bidirectional Recurrent Neural Network (BRNN), the same output layer is connected to two independent recurrent networks that process seach training sequence forwards and backwards. This provides the BRNN with comprehensive sequential knowledge about all points before and after each point in a sequence. With BRNN, there is no need to specify a time window size or goal delay as it has the free dom to use a smuch ora slittle contextasneeded.

Traditional RNNs have the limitation of only being able to utilize prior context. To overcome this, Bidirectional RNNs(BRNNs) process data in both directions using two hidden layers that both feed into the same output layer. By combining LSTMs with BRNNs, a bidirectional LSTM is created that can access on text from both input directions over alon grange.

4.1.1 Proposed LSTM model

In this project, we have constructed an LSTM model with 3LSTM layers followed by 3 Dense layers, as shown in the accompanying figure.

```
Model: "sequential"
_________________________________________________________________
Layer (type)                 Output Shape              Param #
=================================================================
lstm (LSTM)                  (None, 30, 64)            442112
_________________________________________________________________
lstm_1 (LSTM)                (None, 30, 128)           98816
_________________________________________________________________
lstm_2 (LSTM)                (None, 64)                49408
_________________________________________________________________
dense (Dense)                (None, 64)                4160
_________________________________________________________________
dense_1 (Dense)              (None, 32)                2080
_________________________________________________________________
dense_2 (Dense)              (None, 4)                 132
=================================================================
Total params: 596,708
Trainable params: 596,708
Non-trainable params: 0
_________________________________________________________________
```

FIG5: LSTM MODEL

V.RESULTS

To determine the effectiveness of the LSTM model in this project, the performance was measure du sing categorical accuracy. Graphs for both categorical accuracy and categorical cross entropy were also analyzed.

Since there was no publicly available dataset for individual cricket shots, short video clips of cricket shots were made to create a dataset. The dataset was then created manually to meet the specific needs, due to the lack of sufficient data. In the end, the dataset was comprised of around 120 cricket shot recordings, divided into 4 categories, with 30 recordings for each shot type. These experiments were conducte dona system with an AMD Ryzen 5 Hexa core 5500U processor and anNVIDIAGeForceGTX1650GPU,using Jupyter Notebook as the software.

The dataset was then divided into training and testing sets using the train_test_split function, with the training set consisting of 95% of the total data andt he test set consistingof5%ofthe data.

The label array was created with 120 values, each representing one of the four shot actions(Pre-Stance,Stance, Straight Drive, Pull Shot). The true categorical function was applied to convert the label array into a binary class matrix, where Pre-Stancewasrepresentedby[1,0,0,0],Stanceby[0,1,0,0],StraightDrive by[0,0,1,0], and Pull Shotby[0,0,0,1].

In our project, we employed the LSTM algorithm and built a model with three LSTM layers followed by three dense layers. The model was trained on shots made with both the left and right hand.

The model was evaluated using real-time video input from a camera and video clips from international cricket matches.

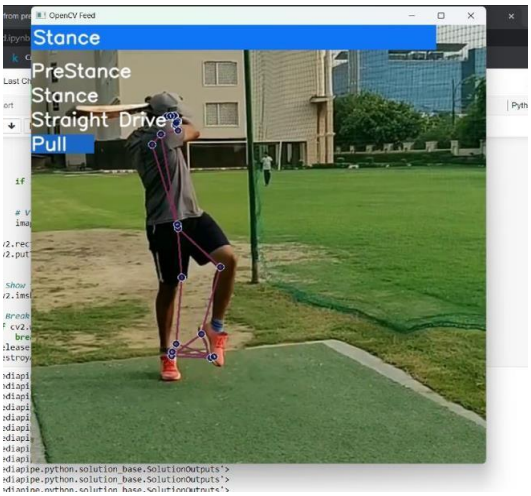


Fig6: *Pull Shot detection*

Fig7: *Straight Drive detection*

Our model achieved an overall accuracy of 83.33%,sur passing there sults of many previous models.

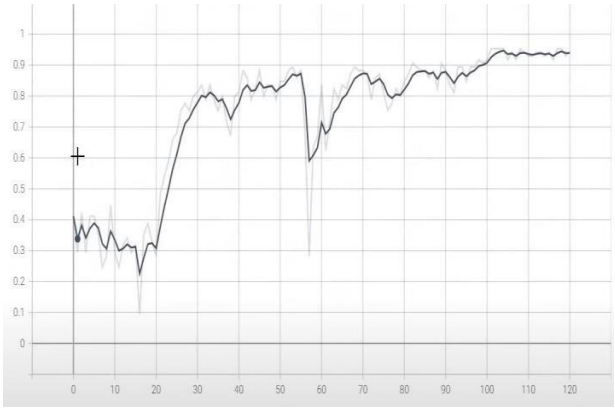


Fig8: Categorical Accuracy v/s Epochs

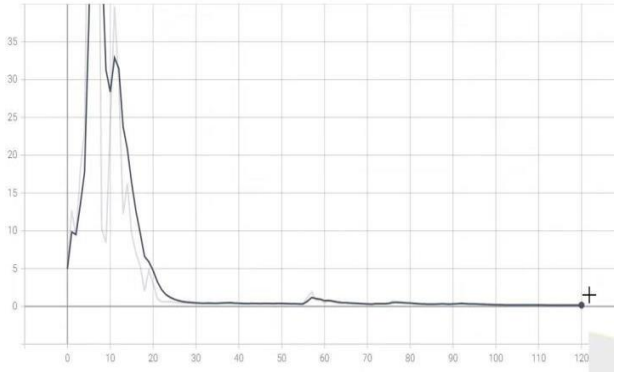


Fig9: Categorical Cross-entropy v/s epochs

VI.CONCLUSION

The project introduces a new approach for detecting various shots in cricket. The framework proposed utilizes Long Short Term Memory (LSTM) Neural Network to extract information from the three components of the model. The proposed method performed better than prior state-of-the-art models and achieved a promising detection accuracy. Additionally, a new data set of 120 videos for 4 classes of cricket shots was introduced. A more sophisticated model can be highly be neficial for professional cricket training centers,as it can accurately recognize skills. The cricket shot detection algorithm can be utilized for visualization and coaching, and a more advanced model could be applied in automated commentary systems. Future work could focus on expandingtherecognitionclassesandincorporatingmorespecificfeaturesfor each definedshot.

With the increasing popularity of shorter format cricket matches likeT20,more unconventional shots are being played. Our future work aims to incorporate these unorthodox shots. Furthermore, classifying umpire actions and boundary events will greatly improve the accuracy of automatic highlight extractions. These actions will also be addressed in future work. Our goal is to expand the dataset and develop a custom model with higher accuracy. Another challenge we plan to tackle is detecting key frames from a video, as this would allow us to work with fewer frames and parameters. Some experiments combining acoustic and image features will be conducted. It would be intriguing to combine traditional machine learning methods like SVM, SIFT, and visual bag of words with deep learning algorithms to study the performance of the model. The proposed architecture can be applied to solve various video classification problems. However, GPU memory limit ation shinderthe combinationofcomplex modelsandlarge datasets.

References

1. *Khan,M.Z.;Hassan,M.A.;Farooq, A.;Khan,M.U.G.Deep CNN Based Data-Driven Recognition of Cricket Batting Shots.In Proceedings of the International Conference on Applied and Engineering Mathematics(ICAEM),Taxila,Pakistan,4–5 September2018;pp.67–71.[CrossRef]*
2. *Karmaker,D.;Chowdhury,A.;Miah,M.;Imran,M.;Rahman, M. Cricket shot classification using motion vector. In Proceedings of the 2nd International Conference on Computing Technology and Information Management(ICCTIM), Johor, Malaysia, 21–23 April 2015; pp. 125–129.[Cross Ref]*
3. *Khan, A.; Nicholson, J.; Plötz, T. Activity recognition for qualit yassessmentofbatting shotsincricketusingahierarchicalrepresentation.Proc.ACMInteract.Mob.WearableUbiquitousTechnol.2017,1,1–31.[CrossRef]*
4. *Foysal, M.F.A.; Islam, M.S.; Karim, A.; Neehal, N. Shot-Net: A convolutional neural network for classifying differentcricket shots. In International Conference on Recent Trends inImage Processing and Pattern Recognition; Springer:*

Berlin,Germany,2018;pp.111–120.

5. *Semwal, A.; Mishra, D.; Raj, V.; Sharma, J.; Mittal, A.Cricket shot detection from videos. In Proceedings of the 9thInternational Conference on Computing, Communication andNetworking Technologies(ICCCNT),Bengaluru,India,10–12 July 2018; pp. 1–6. [CrossRef] Sensors 2021, 21, 2846 19of19*
6. *Harikrishna,N.;Satheesh,S.;Sriram,S.D.;Easwarakumar, K. Temporal classification of events in cricketvideos.InProceedingsoftheSeventeenthNationalConference on Communications (NCC), Bangalore, India, 28–30 January2011;pp. 1–5.[CrossRef]*
7. *Kolekar, M.H.; Palaniappan, K.; Sengupta, S. Semanticeventdetectionandclassificationincricketvideosequence.InProceedingsoftheSixthIndianConferenceonComputerVision,Graphics&ImageProcessing,Bhubaneswar,India,16–19 December 2008;pp.382–389.[CrossRef]*
8. *Premaratne, S.; Jayaratne, K. Structural approach for eventresolutionincricketvideos.InProceedingsoftheInternationalConferenceonVideoandImageProcessing,Singapore,27–29 December2017;pp.161–166.[CrossRef]*
9. *Javed,A.;Bajwa,K.B.;Malik,H.;Irtaza,A.;Mahmood,*
10. *M.T.A hybrid approach for summarization of cricket videos.InProceedingsoftheIEEEInternationalConferenceonConsumer Electronics-Asia (ICCE-Asia), Seoul, Korea, 26–28October 2016;pp. 1–4.[CrossRef]*
11. *Kolekar, M.H.; Sengupta, S. Semantic concept mining incricket videos for automated highlight generation. Multimed.ToolsAppl. 2010,47,545–579. [CrossRef]*
12. *Bhalla, A.; Ahuja, A.; Pant, P.; Mittal, A. A MultimodalApproachforAutomaticCricketVideoSummarization.InProceedingsofthe6thInternationalConferenceonSignalProcessingandIntegratedNetworks(SPIN),Noida,UttarPradesh,India,7–8 March2019;pp.146–150.[CrossRef]*
13. *Tang, H.; Kwatra, V.; Sargin, M.E.; Gargi, U. Detectinghighlightsinsportsvideos:Cricketasatestcase.InProceedingsoftheIEEEInternationalConferenceonMultimedia and Expo (ICME), Barcelona, Spain, 11–15 July2011;pp.1–6.[CrossRef]*
14. *Kumar, Y.S.; Gupta, S.K.; Kiran, B.R.; Ramakrishnan,K.; Bhattacharyya, C. Automatic summarization of broadcastcricket videos. In Proceedings of the IEEE 15th InternationalSymposium on Consumer Electronics (ISCE), Singapore, 14–17 June 2011;pp.222–225. [CrossRef]*
15. *Ramsaran, M.; Pooransingh, A.; Singh, A. AutomatedHighlightGenerationfromCricketBroadcastVideo.InProceedingsofthe8thInternationalConferenceon ComputationalIntelligenceandCommunicationNetworks(CICN), Tehri, India, 23–25 December 2016; pp. 251–255.[CrossRef]*
16. *Ringis,D.;Pooransingh,A.Automatedhighlightgeneration from cricket broadcasts using ORB. In ProceedingsoftheIEEEPacificRimConferenceonCommunications,Computers and Signal Processing (PACRIM), Victoria, BC,Canada, 24–26 August2015;pp. 58–63.[CrossRef]*
17. *Rafiq, M.; Rafiq, G.; Agyeman, R.; Jin, S.I.; Choi, G.S.Sceneclassificationforsportsvideosummarizationusingtransferlearning.Sensors2020,20,1702.[CrossRef][PubMed]*
18. *Steels,T.;VanHerbruggen,B.;Fontaine,J.;DePessemier, T.; Plets, D.; De Poorter, E. Badminton ActivityRecognitionUsingAccelerometerData.Sensors2020,20,4685.[CrossRef] [PubMed]*
19. *Rangasamy,K.;As'ari,M.A.;Rahmad,N.A.;Ghazali,*
20. *N.F.Hockeyactivityrecognitionusingapre-traineddeeplearningmodel.ICTExpress 2020,6,170–174.[CrossRef]*
21. *Junjun, G. Basketball action recognition based on FPGAandparticleimage.Microprocess.Microsyst.2021,80,103334.[CrossRef]*
22. *Gu,X.;Xue,X.;Wang,F.Fine-GrainedActionRecognition on a Novel Basketball Dataset. In Proceedings ofthe ICASSP 2020—2020 IEEE International Conference onAcoustics,SpeechandSignalProcessing(ICASSP),Barcelona, Spain, 4–8May2020;pp.2563–2567*