CRICKET SHOT DETECTION FROM VIDEOS

Archit Semwal*, Durgesh Mishra*, Vineet Raj†, Jayanta Sharma† and Ankush Mittal†
*College of Engineering Roorkee, India
†Graphic Era University, Dehradun, India.

***Abstract*—Classifying various type of bat strokes played in a cricket match has always been an arduous undertaking while indexing the cricket sport. Identifying the type of shot played by the batsman in a cricket match is a substantial aspect as well as one of the unplumbed subjects in this domain. This paper proposes a novel scheme to recognize and classify different types of bat shots played in cricket. The model relies on the state-of-the-art techniques like saliency and optical flow to bring out static and dynamic cues and on Deep Convolutional Neural Networks (DCNN) for extracting representations. Moreover, a completely new dataset of 429 videos, has been introduced to evaluate the performance of the proposed framework. The model achieves an accuracy of 83.098% for three classes of right-handed shots and 65.186% for three classes of left-handed shots.**

***Index Terms*—Cricket shot detection, Machine vision, Image analysis, Pattern analysis, Neural networks, Support vector machines.**

I. Introduction

Over the recent years, the outbreak in the domain of multimedia content such as images and videos over the cyberspace has led to an explosion of ideas and research work among various disciplines. Sports and athletics is one such field in which a large amount of data is available for analysis. The cricket game, being one of the globally popular sport, is broadcasted widely on television and other media platforms offering a broad range of camera motion shots and events for analysis as well as estimation. Classifying various type of cricket shots is a difficult but crucial aspect in regard to indexing cricket games. There are various types of cricket bat stroke such as Late cut, Pull, Cover drive, Leg glance etc played regularly and sometimes with a slight variation during a live cricket match. Often there are times when the batsman need to improve his game and minimize the deviation of his shots from the standard technique, shot detection algorithms are of great utility in such scenarios. The complication with recognizing different cricket shots lies in the extraction of distinct features and cues from the confined number of data frames. Therefore, this calls for a need to develop automated systems which can efficiently detect cricket shots from real-time videos.

Determining the type of shot played by the batsman has its own set of challenges to be tackled. Playing a particular shot may not always result in a successful hit, therefore we can only perceive the intended shot played. The problem of occlusion, the small size of the ball and the speed of action are other factors responsible for making the job even more complicated. Also, processing only the relevant information from the data is time-consuming and a daunting task, while the data set required is essentially large.

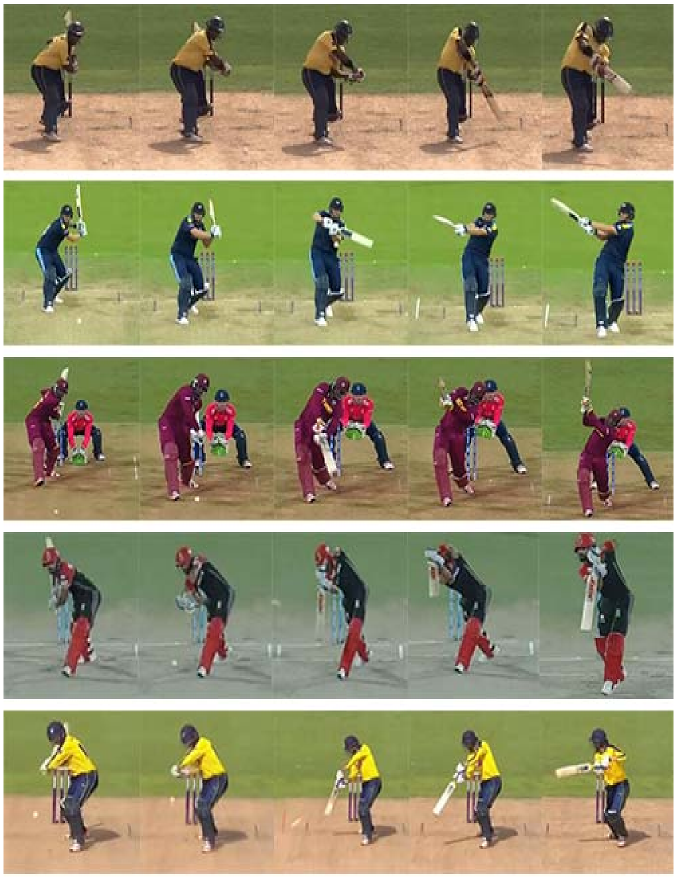

Fig. 1. =
Various cricket shots played during a match

Most of the generic approaches to this subject majorly emphasized on body posture recognition[10] and motion tracking techniques[2][9] to identify the shot from still cameras, with the batsman playing a particular shot within favorable surroundings. The data set used was not from real-time cricket matches where the problem of selecting the object of interest from a continuously moving camera may arise. Also, the descriptors adopted for the estimation were basic ground-level descriptors and thereby did not suggest any sophisticated machine learning methodology.

The major contribution of this paper lies in providing a novel scheme which can identify from a real-time cricket shot

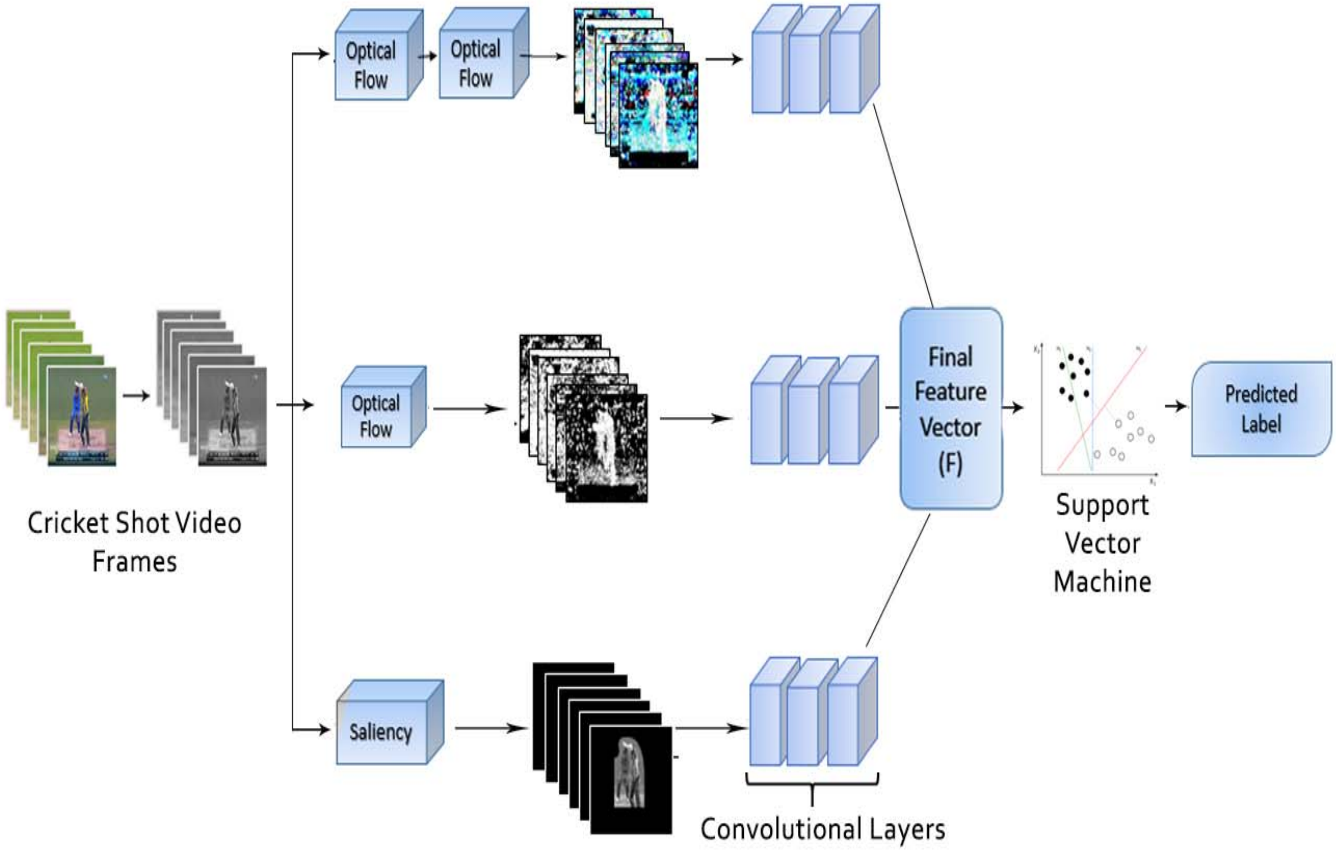

Fig. 2. =

A block diagram representation of the proposed architecture. Different matrices of the frames are supplied as inputs to the convolutional network, output of which is further fed to the multi-class SVM for final categorization.

video, whether the batsman is left handed or right handed and indeed classify the type of intended shot played based on machine learning concepts. The proposed framework aims at establishing a precise stroke detection strategy which makes use of representations extracted from video shots of cricket matches. Besides, a new data-set comprising of a variety of strokes such as lofted on-drive, cover-drive, straight-drive etc have been collected by us from various cricket leagues and all the observations have been carried over this data set only.

The paper has been organized in following sections: A review of the existing work on recognizing cricket shots has been illustrated in section 2. The proposed architecture has been elaborated in Section 3. Data set description has been given in Section 4. The results of the various experiments that were conducted have been presented in Section 5. Section 6 and 7 concludes the paper and discusses the future scope of the work.

II. Literature Review

A study of former approaches to the same problem statement has been presented in this section. Despite being a common issue in this domain of sports, not much work has been done to provide a strategic analysis of cricket shot detection.

Tandon and Mukherjee have proposed a semantic analysis of broadcasting video, involving the use of multi-scale spatiotemporal analysis of color and optical flow features[14] for shot boundary detection and shot classification. Event detection in sports videos, a work proposed by M. Xu et al. gives an effective fusion scheme of visual and auditory modalities to detect events in sports video[16]. Their proposed scheme is built upon semantic shot classification, where they classify video shots into several major interesting classes, each of which has clear semantic meanings.

Karmakar et al. suggested a novel scheme to identify batsmans hand stroke direction along with the type of shot played[5]. Salient features are extracted and tracking of the bat and ball is done using Kalman filter. Then the 3-D skeleton model and motion of articulated body parts are detected, while a vector is derived from bat flow to obtain the particular angle of the shot[1]. Another work on automatic indexing of cricket using camera motion parameters is presented by Lazarescu et al. which uses Incremental Learning Algorithm (ILF)[7], depending on the type of shot the camera typically follows the ball and this is a useful indication of the type of shot.

III. Proposed Methodology

A detailed description of the applied framework has been unfolded in this section.

The key intuition which has been taken into account is that, though the speed and style of the batting stroke may vary from player to player but the basic skill set such as the body posture of the batsman, the direction, and impact of motion as well as the bat movement during the collision with the ball, remains approximately the same.

Having an insight into the framework, the videos are first decomposed into frames which are fed as input to the Convolutional Neural Network(CNN) to determine the shot configuration as described in the fig. 3. For the sake of incorporating the native features, salient regions have been recognized for each considered frame whereas to apprehend the dynamic cues i.e. motion of the batsman and the bat, sequences of continuously ordered frames are subjected to the optical flow technique, which allows the estimation of motion as discrete image displacements, thereby calculating the pixel velocity of every frame. Next pixel acceleration is computed, by again subjecting these frames to the process of optical flow. All the representations extracted from the frames are now fed to the convolutional neural network(CNN) input layer for processing and final classification.

A. Preprocessing

The original videos were manually cropped, so as to limit the data content and reduce the computational time. Only the segment which constituted of the batsman and the bat during the collision with the ball was taken into account. The frames were next extracted from the clipped videos and were supplied as inputs to the model. All of the further processing and techniques has been carried over these video frames.

B. Determining Shot Configuration

Before detecting the intended shot played, the model determines the chirality of the player. This will scale down the count of comparison classes and will give prominent outcomes in the final shot classification algorithm. Therefore, a novel approach has been adopted for the shot configuration detection.

The frames extracted from the dataset are first categorized as left-handed shots or right-handed shots and then individually supplied as input to the CNN layers. The convolutional networks(CNN) are trained over these inputs from which they imbibe the representations extracted from these frames[8][13]. Finally, the arrangement is tested on a fresh data using SVM(Support Vector Machine) as the classifier. The complete scheme has been presented in the block diagram given ahead.

C. Detecting salient regions

A Graph-Based Visual Saliency(GBVS) method[3] has been adopted to bring out the salient regions, which are the object of interest. GBVS, a new bottom-up visual saliency model relies on two stages:

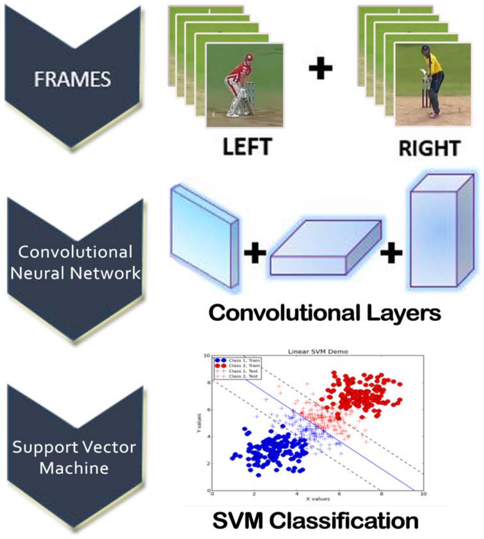

Fig. 3. =
The block diagram presents the approach followed to determine the configuration of the shot.

- Firstly, it forms activation map based on motion intensity of each pixel in the provided frame.
- Secondly, it normalizes them in such a way that it focuses only on the easily distinguishable characteristics, in combination with other maps.

We have applied this approach on the cricket shot videos, frame by frame and extracted the saliency map of every frame providing us with the image showing overlapping of the map at equivalent pixel intensities over our original frame. This mechanism is repeated for the complete series of frames, for all the videos of a particular class in our dataset. As a result of this computational process, we have succeeded in maintaining an intensity map for every pixel through which we can easily draw out the salient region for that particular frame as shown in fig(4).

D. Optical Flow

Optical flow cannot be computed locally due to the availability of only a single component, while the velocity vector has two measurements. Optical flow is, in fact, the division of apparent discrete velocities caused by the motion of brightness patterns in an image. With the objective of apprehending net displacements of individual pixels for a sequence of successive continuous frames, Horn and Schunck in 1981 suggested an algorithm[4] as well as deduced an equation for estimation of optical flow.

Suppose I(x,y,t) denotes the intensity of a given pixel **P(x,y)** at a time instance t. Therefore, for a short time duration (t+Δt) if the pixel shifts to a new locality P'(x+Δx,y+Δy) its intensity relatively varies to I'(x+Δx,y+Δy,t+Δt). Considering the

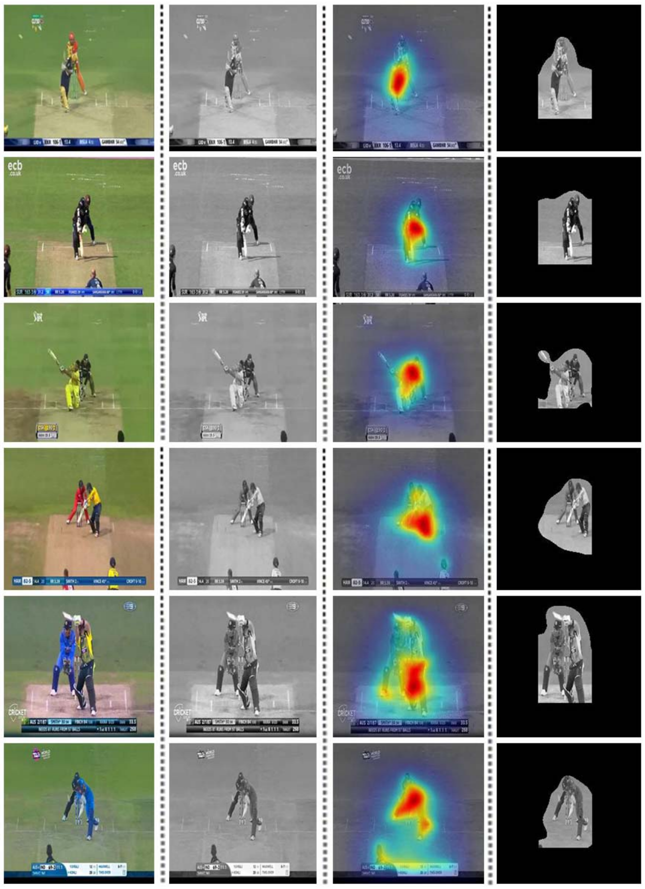

Fig. 4. =
Extracting salient regions from the input frames

assumption that the change in pixel position is very small we can infer :

$$\frac{dI(x,y,t)}{dt} = 0 \tag{1}$$

Hence, we can suppose that :

$$I(x,y,t) = I'(x+\Delta x, y+\Delta y, t+\Delta t) \tag{2}$$

The above equation forms the groundwork for optical flow. The optical flow of a frame constitutes the two orthogonal components u and v along the horizontal axis and vertical axis respectively where

$$u = \frac{dx}{dt}, v = \frac{dy}{dt} \tag{3}$$

Applying Taylor series expansion on equation 2.

$$\frac{\partial I dx}{\partial x dt} + \frac{\partial I dy}{\partial y dt} + \frac{\partial I}{\partial t} = 0 \tag{4}$$

i.e.

$$I_x u + I_y v + I_t = 0 \tag{5}$$

1) Optical FLow Normalization: For a provided pixel Q(x,y), the optic flow can be calculated as :

$$Q(x,y) = u(x,y)^2 + v(x,y)^2 \tag{6}$$

The maximum and minimum values for Q can be expressed as

$$Q_{max} = max(u(x,y)^2 + v(x,y)^2)(x,y) \in C \tag{7}$$

$$Q_{min} = min(u(x,y)^2 + v(x,y)^2)(x,y) \in C \tag{8}$$

Therefore,

$$f(x,y) = \int \frac{Q(x,y)}{(Q_{max} - Q_{min}) * 255} \tag{9}$$

where C represents the set of coordinates for every pixel in the image while f(x,y) denotes the normalized data of optical flow values. The magnitude of f(x,y) is from 0 to 255.

E. Representation using Convolutional Neural Networks

A Convolutional Neural Network (CNN) is a powerful machine learning technique from the field of deep learning which are extensively used for visual imagery[12][6]. Large collections of diverse images are used to train the CNN layers. From these large collections, CNN's can learn rich feature representations for a wide range of images[11] surpassing customary features like HOG, LBP etc. An easy way to leverage the power of CNN, without investing time and effort into training, is to use a pre-trained CNN as a feature extractor. The proposed framework has been fine-tuned on the pre-trained Alex-Net model. In Alex-net architecture, there are five convolution layers and 3 pooling layers after which we obtain two Fully Connected Layers or FC Layers. All the frames are re-sized to the dimension 227 X 227 as per the requirements of Alex Net. The model comprises of 227 X 227 X 3 input layer, 96 X 11 X 11 conv ,3 X 3 max pooling ,128 X 5 X 5 conv, 3 X 3 max pooling, 256 X 3 X 3 conv, 192 X 3 X 3 conv ,192 X 3 X 3 conv ,3 X 3 pooling 4096 X 1 FC-4096 X 1 FC 1000 X 1 FC layers.

Representations were extracted from the fc7 layer of the network and interpreted as a 20480*1 dimension vector. The features of every individual frame in a video are then concatenated to form a single feature matrix F. In order to get a better recognition rate, we have employed 3 CNNs. F = F1, F2, F3 where F1, F2, F3 represent the different representation vectors extracted by each of the three CNN. The dimension of F is 61440 X 1. Vector F is supplied as the input to the support vector machine.

F. Classification using SVM

The representations extracted from section IV are further facilitated to a multi-class SVM. Conventionally, the SVM classifying model was used as a binary classifier but with some modifications and the employment of one vs all strategy, a multi-class SVM model was developed[15]. The multi-class

Cricket Shot	No. of videos
Left Straight Drive	62
Right Lofted On-drive	86
Left Lofted On-drive	63
Right Cover Drive	86
Left Cover Drive	53
Right Straight Drive	79

TABLE I
TOTAL DATA SET COUNT UNDER EACH CLASS LABEL.

	Left	Right
Left	**0.9769**	0.0231
Right	0.0682	**0.9318**

TABLE II
CONFUSION MATRIX FOR LEFT OR RIGHT CONFIGURATION DETECTION OF THE SHOT.

SVM soon became the mainstream classifier for machine learning applications due to its considerable efficiency.

The SVM was trained and tested on a total of 6 classes. A sequential minimal optimization technique with a kernel offset of 0 was used for training the SVM.

IV. DATA-SET DESCRIPTION

Since no public dataset is available for individual cricket shots, video clips of cricket matches and highlights were used. Due to the inadequacy of the desired data set, the data set was manually edited in accordance with the requirements. The cricket matches from various cricket leagues such as T20, IPL, World Championship, Test series have been downloaded from the internet and cropped into small video clips, categorizing each clip into specific shot classes. A video cutter software has been employed for this purpose where only the portion where the shot is being played is retained. The final data set consisted of around a total of 429 cricket shot recordings. Table (1) describes the considered video clip distribution with respect to each class. The experiments were performed on a system Intel (i7 7th generation) processor and an NVIDIA QUADRO-2000 GPU. MATLAB2016a was used as a software for performing these experiments.

V. EXPERIMENTAL SETUP

This section delineates the distinct experimental scenarios that have been conducted for examining the performance of the proposed framework on the considered dataset. In order to measure the performance of the suggested schema three different experimental scenarios have been conducted and their observed outcomes have been illustrated below.

A. Scenario 1

In the first set of experiments, the aim was to determine whether the batsman is left handed or right handed. The framework (CNN + SVM) here attained an overall identification accuracy of 95.4198%. Table (2) shows the recorded observations.

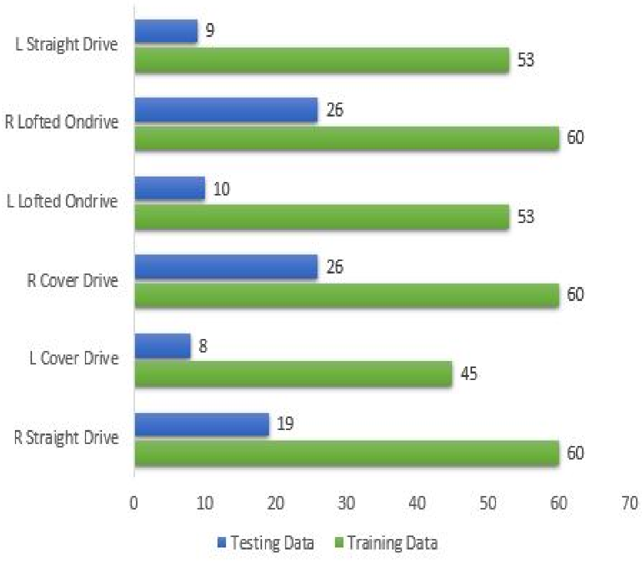

Fig. 5. =

Frequency Distribution of the data set into training and testing data under various class labels

Technique	Right	Left
AlexNet + Optical Flow1 + Multi class SVM	43.6620	40.7407
AlexNet + Optical Flow1 + KNN	32.3944	33.3333
AlexNet + Optical Flow1 + Random Forests	36.6197	48.1481
AlexNet + Optical Flow2 + Multi class SVM	47.8873	48.1481
AlexNet + Optical Flow2 + KNN	39.4366	48.1481
AlexNet + Optical Flow2 + Random Forests	49.2958	51.8519
Frame + AlexNet + Multi class SVM	77.4648	61.020
Frame + AlexNet + KNN	54.9296	37.0370
Frame + AlexNet + Random Forests	69.0141	55.5556
Proposed Approach	**83.0986**	**65.186**

TABLE III
ACCURACY COMPARISON OF THE PROPOSED TECHNIQUE WITH OTHER STATE OF THE REPRESENTATION CLASSIFIER METHODS.

B. Scenario 2

In this part, the experiments were conducted so as to evaluate the performance of each individual part of the considered architecture.

Table (3) demonstrates the accuracy obtained by each subcomponent and the combined proposed model. It is clearly visible that the performance of the model increases when we apply optical flow again over optical flow applied continuous frames. Also, the performance of the framework grows significantly to around 10-15% when we supply salient frames as input to the model. Further, the proposed model yielded a higher accuracy than each of its individual sub-parts.

C. Scenario 3

In the third set of experiments, the major focus lied in examining the efficiency of the proposed framework for distinct classes. After combining each sub-component i.e. left-right classification, optical flow over frames, optical flow over

	Straight drive	Cover drive	Lofted on drive
Straight drive	**0.6316**	0.0000	0.3684
Cover drive	0.0000	**1.0000**	0.0000
Lofted on drive	0.1923	0.0000	**0.8077**

TABLE IV
Confusion Matrix for various right handed cricket strokes .

	Straight drive	Cover drive	Lofted on drive
Straight drive	**0.5556**	0.0000	0.4444
Cover drive	0.1250	**0.5000**	0.3750
Lofted on drive	0.1000	0.0000	**0.9000**

TABLE V
Confusion Matrix for various left handed cricket strokes .

optical flow and salient frames together the model yielded a satisfactory accuracy. Table (4) and Table (5) presents us with the confusion matrix obtained for different shot classes.

VI. Discussion

From the above experimental scenarios, we can infer that the model efficiently recognizes the cricket shots with an accuracy of 83.098% for three classes of right-handed shots and 65.186% for three classes of left-handed shots, therefore the model proves to be quantitatively reliable. Moreover, the experiments conducted on the data set brought out the significance of mapping pixel motion using optical flow in building efficient cricket shot detection systems. It is clearly visible that the model yields a better detection accuracy when we use a multi-class SVM (Support Vector Machine) classifier rather than using any other state-of-the-art classifiers. Also the observation, that incorporation of local features along with the dynamic cues, improves the accuracy of the model was made.

However, we strongly agree with the fact that there are certain limitations in our model, which can be further addressed in the future. According to our user study and precision results, the model miscalculates a few shots like left straight drive, left cover drive and right straight drive to some extent. At present, our model struggles as we increase the number of detection classes due to the limited data set. Besides, due to unsteady camera motion, it is difficult to map the movement of the object of interest and hence such misinterpreted indications lead to false positives in some cases. Another challenging task would be to determine the type of shot when a batsman give immediate variations to the standard shot resulting in an unexpected hit.

VII. Conclusion

The paper offers a novel approach for detection of various shots in the domain of the cricket sport. The proposed framework relies on the pre-trained convolutional neural network for extracting representations from all three auxiliary components of the model. The model performed better than any other formerly used state-of-the-art descriptors and classifiers, with a promising detection accuracy. We also introduced a fresh data set of 429 videos for 6 classes of cricket shots. A more refined model can be of great utility for professional training centers of the sport, recognizing the skill precisely. Cricket shot detection algorithm can be employed for visualization and coaching purpose and a more developed model will find its applicability in automated commentary systems. The future work may lay emphasis on increasing the recognition classes focusing on extending the incorporation of native features for every defined shot.

References

[1] AZM Ehtesham Chowdhury and Abu Umair Jihan. Classification of cricket shots using computer vision.

[2] Harry Collins and Robert Evans. You cannot be serious! public understanding of technology with special reference to ?hawk-eye? *Public Understanding of Science*, 17(3):283–308, 2008.

[3] Jonathan Harel, Christof Koch, and Pietro Perona. Graph-based visual saliency. In *Advances in neural information processing systems*, pages 545–552, 2007.

[4] Berthold KP Horn and Brian G Schunck. Determining optical flow. *Artificial intelligence*, 17(1-3):185–203, 1981.

[5] D Karmaker, AZME Chowdhury, MSU Miah, MA Imran, and MH Rahman. Cricket shot classification using motion vector. In *Computing Technology and Information Management (ICCTIM), 2015 Second International Conference on*, pages 125–129. IEEE, 2015.

[6] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. In *Advances in neural information processing systems*, pages 1097–1105, 2012.

[7] Mihai Lazarescu, Svetha Venkatesh, and Geoff West. On the automatic indexing of cricket using camera motion parameters. In *Multimedia and Expo, 2002. ICME'02. Proceedings. 2002 IEEE International Conference on*, volume 1, pages 809–812. IEEE, 2002.

[8] Yair Poleg, Ariel Ephrat, Shmuel Peleg, and Chetan Arora. Compact cnn for indexing egocentric videos. In *Applications of Computer Vision (WACV), 2016 IEEE Winter Conference on*, pages 1–9. IEEE, 2016.

[9] Ajay K Sarkar, Daniel A James, Andrew W Busch, and David V Thiel. Triaxial accelerometer sensor trials for bat swing interpretation in cricket. *Procedia Engineering*, 13:232–237, 2011.

[10] Jamie Shotton, Toby Sharp, Alex Kipman, Andrew Fitzgibbon, Mark Finocchio, Andrew Blake, Mat Cook, and Richard Moore. Real-time human pose recognition in parts from single depth images. *Communications of the ACM*, 56(1):116–124, 2013.

[11] Pushkar Shukla, Tanu Gupta, Aradhya Saini, Priyanka Singh, and Raman Balasubramanian. A deep learning frame-work for recognizing developmental disorders. In *Applications of Computer Vision (WACV), 2017 IEEE Winter Conference on*, pages 705–714. IEEE, 2017.

[12] Karen Simonyan and Andrew Zisserman. Two-stream convolutional networks for action recognition in videos. In *Advances in neural information processing systems*, pages 568–576, 2014.

[13] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014.

[14] Rashish Tandon. Semantic analysis of a cricket broadcast video. 2009.

[15] Jason Weston and Chris Watkins. Multi-class support vector machines. Technical report, Technical Report CSD-TR-98-04, Department of Computer Science, Royal Holloway, University of London, May, 1998.

[16] Min Xu, Ling-Yu Duan, Chang-Sheng Xu, and Qi Tian. A fusion scheme of visual and auditory modalities for event detection in sports video. In *Multimedia and Expo, 2003. ICME'03. Proceedings. 2003International Conference on*, volume 1, pages I–333. IEEE, 2003.