AUTOMATIC CRICKET HIGHLIGHTS EXTRACTION USING EVENT DRIVEN AND EXCITATION BASED FEATURES

by

SATHISH KUMAR S 2015103530
SIVARANJANI P 2015103534
BELAL R 2015103007

A project report submitted to the

FACULTY OF INFORMATION AND COMMUNICATION ENGINEERING

in partial fulfillment of the requirements for the award of the degree of

BACHELOR OF ENGINEERING

in

COMPUTER SCIENCE AND ENGINEERING

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

ANNA UNIVERSITY, CHENNAI – 25

APRIL 2019

BONAFIDE CERTIFICATE

Certified that this project report titled **AUTOMATIC CRICKET HIGHLIGHTS EXTRACTION USING EVENT DRIVEN AND EXCITATION BASED FEATURES** is the *bonafide* work of **SATHISH KUMAR S (2015103530)**, **SIVARANJANI P (2015103534)** and **BELAL R (2015103007)** who carried out the project work under my supervision, for the fulfillment of the requirements for the award of the degree of Bachelor of Engineering in Computer Science and Engineering. Certified further that to the best of my knowledge, the work reported herein does not form part of any other thesis or dissertation on the basis of which a degree or an award was conferred on an earlier occasion on these or any other candidates.

Place: Chennai
Date:

Dr. P. Uma Maheswari
Associate Professor
Department of Computer Science and Engineering
CEG, Anna University, Chennai – 25

COUNTERSIGNED

Head of the Department,
Department of Computer Science and Engineering,
Anna University Chennai,
Chennai – 600025

ACKNOWLEDGEMENT

This work would not have been possible without the continued support in terms of discussions, encouragements and suggestions of many people.

First of all, we would like to express our deepest gratitude towards our project guide **Dr. P. Uma Maheswari,** Associate Professor for her guidance, support, motivation and encouragement at every phase of the project. We owe her for harnessing our potential and bringing the best out of us. Without her support through every phase of the project, we could never have it to this extent.

We are grateful to **Dr. S. Valli**, Head of the Department, Computer Science and Engineering, Anna University, Chennai-25, for providing a platform and extending the facilities for our project.

We express our thanks to the panel of reviewers **Dr. S. Chitrakala**, Professor, **Dr. Angelin Gladston**, Associate Professor and **Dr. S. Renugadevi**, Assistant Professor for their valuable suggestions and critical reviews throughout the course of our project.

We wholeheartedly thank our family, parents and friends for their unwavering support during our work.

Sathish Kumar S **Sivaranjani P** **Belal R**

ABSTRACT

In today's world, the rapid development of the digital video has led to the fast growth of the video data. Analysis and consumption of this huge repository urges the sports broadcasters to apply video summarization. Highlights are the summarized video containing only the key events. The huge time and labour consumption of manual highlights extraction motivates the automation of the process. Summarization of sports videos is a challenging task due to variations in camera orientation, game structure and editing effects introduced by the broadcasters.

We propose a model to extract highlights of the cricket match video automatically by recognizing excitation and event driven features. Event tags associated with each live frames are identified using CNN classifiers followed by shot boundary segmentation. Key events such as fours, sixes and wickets are detected by removing advertisements and replays using scorecard detection. Excitement is detected by analyzing the short time audio energy of the input video. All the highlighted video shot segments are concatenated to form the final highlights video.

The results are validated against the manual labels of input video. The precision, recall, accuracy and error rate are calculated. The quantitative assessment shows that the proposed model generates promising results in extracting highlights from the long-duration cricket video.

ABSTRACT

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF ABBREVIATIONS

BBL	Big Bash League
BBN	Bayesian Belief Network
CNN	Convolutional Neural Network
CSV	Comma-Separated Values
GMM	Gaussian Mixture Model
MFCC	Mel-Frequency Cepstral Coefficients
OCR	Optical Character Recognition
SVM	Support Vector Machine
ZCR	Zero Crossing Rate

CHAPTER 1

INTRODUCTION

1.1 PROBLEM DOMAIN

Computer vision is a field of computer science that works on enabling computers to see, identify and process images in the same way that human vision does, and then provide appropriate output. It is like imparting human intelligence and instincts to a computer. Computer vision is closely linked with artificial intelligence, as the computer must interpret what it sees and then perform appropriate analysis or to act accordingly. Computer vision's aim is to enable computers to perform the same kind of tasks as humans with the same efficiency. Computer vision is considered as the base for video summarization.

Video summarization refers to a summary creation of a video where it has to address three main beliefs. First, the video summary should contain the most important scenes and events from the video. Second, the video summary should not show video segments concatenated together in a blindly way. Finally, the video summary should not contain any redundancy. Video summarization is considered one of the most important features which it makes the search away easier and useful than before. The main challenge in sports video summarization resides in detection of all the key events in the video. Sports video contains replays, which are the redundant video shots, should be detected and removed.

In this project, we have developed a system that would automate the sports highlights extraction process using both event and excitation based features. Our interest towards cricket leads us to take cricket video as a test case for sports video summarization.

1.2 PROBLEM STATEMENT

Nowadays digital video plays an important role in everyday's life and due to widely used low cost storage media, the volume of digital video tends to be very large and variety of available video data makes the search and retrieval of content a more and more difficult task. The amount of information generated in today's society is growing exponentially. Videos are voluminous, redundant and their overall contents cannot be captured at a glance. It is essential to help user to provide more compact, interesting video content with narrow bandwidth. In order to meet this need, video summarization is needed. A complete sports video whose duration is much longer like cricket is monotonous which most viewers don't prefer. Hence Sports highlights extraction was brought to light. This work automatically generates highlights from cricket match video by extracting the key event segments and segments with excitement.

1.3 PROBLEM DESCRIPTION

Given a cricket match video as input, the system should generate it's highlights. The highlights generated by the system should contain the key events such as fours, sixes, wickets and other excited segments of the video. The events like fours, sixes, wickets are extracted by comparing the scores and the excited segments are extracted by detecting the excitement in the match.

1.4 OBJECTIVES

1. The main objective is to implement a system that automatically extracts highlights from the cricket match video.
2. To remove the replays in order to avoid the repetitive events in the highlights.
3. To identify and extract the key events such as boundaries, wickets.
4. To identify the video segments containing excitement.
5. To reduce the size required to store voluminous videos.

1.5 SCOPE

A T20 cricket match has an average duration of 4 hours. As the outcome of the match would have been known, people who has missed the match likes to watch only the key events of that match that is the highlights of the match. This emphasizes the importance of highlights. In the fast running world, highlights video will sound better when compared with the longer full match video. Be it any sports, highlights are important for broadcasters in order to increase the viewership. Automatic cricket highlights extraction is very less and hence this system.

1.6 CHALLENGES

1.6.1 Match Timing

The day time match will have high brightness and shadows on the pitch. Night match will have low brightness, less shadows and high contrast. This variations affects the scorecard recognition and event classification.

1.6.2 Replays and Advertisements

Key events like wickets, sixes, fours in cricket match are followed by replay of that event. This replay are the repetitive video shot of the actual event. Since advertisements are unnecessary to the viewers, it should also be removed

1.6.3 Event Detection

Event detection for every frame in the cricket match is highly challenging. Video shot segmentation is primarily dependent on event detection. If not classified properly, it affects further classification in the highlights.

1.6.4 Excitement Detection

Commentators excitement and crowd cheers are one of the major parameters to detect the interesting video segment. Excitement present in the each video segments are calculated and compared with other segments to find the segments that are really exciting to watch.

1.7 ORGANIZATION OF THESIS

Chapter 2 discusses the existing approaches for sports highlights extraction in detail. Chapter 3 gives the requirements analysis of the system. It explains the functional and non-functional requirements, constraints and assumptions made in the implementation of the system. Chapter 4 explains the overall system architecture and the design of various modules along with their complexity. Chapter 5 gives the implementation details of each module, describing the algorithms used. Chapter 6 elaborates on the results of the implemented system and gives an idea of its efficiency. It also contains information about the dataset used for testing and the other observations made during testing. Chapter 7 concludes the thesis. It also states the various extensions that can be made to the system to make it function more effectively.

CHAPTER 2

RELATED WORK

Video browsing technologies developed in recent years are generally divided into two categories: video summarization [3, 19, 21] and video skimming [21]. Video summarization shows a static capsule representation of a long video in terms of a set of key frames, whereas video skimming constitutes a dynamic representation that generates a shorter abstraction of a long video. Video summarization [14, 17, 24] aims at producing short videos or key frames by eliminating redundancy either at signal level or in semantic content. [23] shows the algorithm for video summarization through shot boundary detection, redundant frame elimination and stroboscopic imaging. By eliminating the redundant frames after shot boundary detection, the remaining three consecutive frames are combined using stroboscopic method. Another sub-area of video summarization involves multimodal video analysis such as movie trailer generation [4, 7].

Two main issues in video browsing technologies include

1. how to extract important contents.
2. how to distribute those important contents into limited display duration or space.

One of the possible solutions to the first issue is highlight extraction. The extracted highlight captures most of the important or exciting contents in video and also provides a kind of dynamic video abstraction.

A lot of research has been done for analyzing sports videos[16, 8]. The major aim of sports video analysis is to provide the assistance training. Many works in highlight generation from sports videos is done by player's action analysis and tracking the objects of interest throughout the game. Different approaches are proposed to extract the highlights.

This chapter gives a survey of the possible approaches to summarize the sports video (Cricket, Soccer, Baseball, Tennis, Golf) which is a highlights. We proposed to work with cricket video highlights extraction and we wanted to adopt a hybrid method based on events and excitement. The output should be highlighted events from the given input video. Thus this survey helped us to analyze the various existing approaches and decide on the one which would best cater for our Cricket Highlights Extraction.

2.1 APPROACHES FOR HIGHLIGHTS DETECTION

2.1.1 Shows only Replays

Replays are the repetitive video shots that shows the key events in the video.

In [8], the replays are detected and extracted to form highlights. A replay segment is considered as a clip sandwiched in gradual transitions and absence of score-caption. The proposed method is robust to broadcaster's variation, sports category, score-caption design, camera variations, replay speed, logo design, size and placement. The proposed algorithm does not rely on logo template recognition for replay detection, which makes it computationally efficient. The proposed system consists of two steps i.e., gradual transition and scorecard detection.

In [25], the proposed highlight summarization system was based on replays because the replay is a reliable clue to the highlight and the features is not limited in a specific kind of sports game. First, the replay clips in the sports video are extracted as the highlight candidates. Then the highlight candidates are ranked based on the audio energy arousal level and motion activity. Firstly, event-replay structure is proposed. Secondly a novel highlight model was proposed considering the inter-relation of event replays.

2.1.2 Shows only Important events

The important events in the cricket are the video shot segments containing fours, sixes and wickets and also the segments containing crowd cheers and commentator's excitement. These events are identified by using scorecard recognition[8, 10, 20], event detection. The excitement detection[16, 12, 20] is used to identify key events and milestones. The excitement in the match is done by analyzing the commentator's excitement in voice and crowd's cheer. [13] identifies the key event by scoreboard identification and score change detection then removal of replay and ad is done. The key frame extraction is done by modeling the semantics of the game using state transition model. In [11], events are grouped into rich semantic concepts, using hierarchical classication then selects semantic concepts and the events within the concepts, according to the degree of importance.

2.1.3 Using Logo Transition

A replay in a broadcast sports video is often accompanied with a pair of logo transitions which sweep off at the beginning and the end of it. By using logo transition detection, boundaries of the replay are calculated.

In [25], the logo template was generated automatically from the video. The sweeping effect frames is detected and classified them into several clusters. Then the logo cluster is selected according to a judging criterion. The mean image of the frames in the logo cluster is set to be the logo template.Then all the logos are detected with the logo template.

The distance between logo template and an arbitrary frame in the video was computed. If the distance is smaller than a pre defined threshold, it is recognized as a logo. After all logos are detected and are paired as the boundaries of replays and the wrong logos are eliminated. The replay detection approach proposed above can detect replays in sports video effectively which can be utilized to summarize highlight.

2.1.4 Using Gradual Motion

In [8], replay segments in sports videos include various types of gradual transitions such as dissolves, wipes, fade-in/out etc. It has been observed that replays in sports videos are sandwiched between gradual transitions frames and do not contains scorecards. The characteristics of multiple gradual transitions are therefore used to identify the boundaries of a replay segment by detecting logo frames. Separation between two successive gradual transitions (in number of frames) is used to generate a candidate replay segment. Then a segment between two successive gradual transitions is labeled as a candidate replay segment if it satisfies the condition specified in the equation 2.1.

$$2N_{GT} + N_{RL} \leq E_{(i+1)} - S_i \leq 2N_{GT} + N_{RU} \tag{2.1}$$

where $\mathrm{N_{RL}}$ and $\mathrm{N_{RU}}$ represent lower and upper limits of a replay duration (in number of frames)

2.1.5 Using Scorecard Detection

The scorecards are displayed at fixed location in almost all sports videos. It has been observed through watching extensive amount of sports videos that replay segments do not contain scorecard. Therefore, scorecards are used for replay detection. Input video segments are analyzed to extract scorecards. The presence/absence of scorecard is used to detect replay and live frames.

In [8], the preprocessing stage transforms the candidate replay segments into a sequence of gray scale images. To reduce computational cost, sequence (of gray scale images) is down-sampled by a factor of 2. Each image is processed for illumination adjustment using the top hat filtering. The top hat filter performs morphological opening with a structuring element SE followed by subtraction from the original image. These are expressed in the equations 2.2 and 2.3.

$$I_{thin}^{(i)} = I^{(i)} \bigotimes SE \tag{2.2}$$

$$I_{adj}^{(i)} = I^{(i)} - I_{thin}^{(i)} \tag{2.3}$$

where $I_{thin}{}^{(i)}$, $I_{adj}{}^{(i)}$,$I^{(i)}$,presents the thinned image, illumination adjusted image, and input gray scale image respectively, if ith frame. SE is the disk shaped structuring element of size α , and $\otimes$ is thinning operator. A sliding window of length L frames is used to compute temporal running average sequence expressed in equation 2.4.

$$I_{avg}^{(i)} = I_{avg}^{(i-1)} - I^{(i-1)} + I^{(i+1)}/L \tag{2.4}$$

where $I_{adj}{}^{(i)}{}_{avg}$ represents the average if ith frame.
Then image binarization, morphological thinning is done. OCR is applied on the thinned image. The absence of the scorecard is labelled as the replay frames.

2.2 APPROACHES FOR KEY EVENT DETECTION

2.2.1 Using Scorecard Recognition

Highlights based on scorecard recognition in [1, 15] is done using textual information extraction method. This textual information is extracted from each frame by first detecting scorecard, then converting the text on this score into a sentence like structure based on OCR. If-then rules are used to extract the events associated with the excitement clips. Once textual information is at our disposal, difference in score and wickets is detected to get information about 4, 6 or fall of a wicket. This forms the basis of event detection. If the difference for the runs-caption in the successive frames of the excitement clip is 6, 5, 4, 3, 2, 1, the event associated with the corresponding excitement clip is six-runs, five runs, four-runs, three-runs, two-runs, one-run respectively. If the difference for the wicket-caption in the successive frames is 1, the event associated with the corresponding excitement clip is wicket. If the difference for the runs-caption and wicket-caption is 0, the event associated is no-run/no-wicket (NRNW) event. Now those frames are included in which event has occurred and combined together to generate highlights. We apply caption recognition model to every frame in the excitement clip.

2.2.2 Using Excitement Detection

The excitement detection is done by using the commentator's voice tone analysis or identifying the emotion in the commentator's speech or using the signal processing. [16] extracts the highlights using commentator's excitement. This is detected by a combination of an audio classifier and a salient keywords extractor applied after a speech-to-text component. Excitement in the tone is identified using the trained SVM classifier as in [16]. [10] uses signal processing techniques like audio energy rate, zero crossing rate to identify the excitement in the audio in the sports video. [22] detects highlights by detecting the whistle sound and excitement from crowd and commentators. Whistle sound in certain sports indicates the start and end of the event i.e., sports event boundary.

2.3 APPROACHES FOR EVENT CLASSIFICATION

2.3.1 Using CNN

CNN have outperformed most of the traditional computer vision algorithms for tasks such as image classification and object detection. A CNN is a combination of a feature extractor and a classifier. The convolutional layers of the CNN are the feature extractors where it learns the representations automatically from the input data. The early layers in the CNN learn more generic features such as shapes, edges, and colour blobs, while the deeper layers learn features more specific to that contained in the original dataset. The last fully connected layers of the CNN use these learned features and classify the data into one of the classes. [9] uses CNN for image classification. It uses transfer learning technique called fine tuning technique and uses AlexNet architecture. The network was made up of 5 internal convolutional layers: C1, C2, C3, C4, C5, pooling layers, dropout layers, and 3 fully connected layers : FC6, FC7, FC8. [18] uses CNN for identifying events in cricket videos based on detecting the pose of the umpire. System proposed in [18] is built in two phases. The first phase involves designing classifiers to distinguish images containing an umpire versus no umpire, and also detect the pose of the umpire, if present. During the training stage, the input images are pre–processed by performing intensity normalization on the pixel values and resizing to 299 by 299 pixels for the Inception V3 network and 224 by 224 pixels for the VGG19 network. The features are extracted from different layers of the pretrained networks. Finally, these features are used to train a linear SVM classifier to output the class label of the predicted pose of the umpire. The second phase involves detecting the events from the cricket videos using the saved classifier models, and generating the summary of the videos.

2.3.2 Using SVM

In [18], a linear support vector machine (SVM) classifier is trained on the extracted features for detecting the pose of the umpire. In [16], Cheer samples from 2016 Masters and Wimbledon replay videos as well as examples of cheer obtained from YouTube were used in order to train the audio cheer classifier using a linear SVM on top of deep features. For negative examples, we used audio tracks containing regular speech, music, and other non-cheer sounds found in Masters and Wimbledon replays.

2.3.3 Using BBN

Event-based highlights use more semantically meaningful content than the excitement–based highlights and its accuracy depends upon the richness of the semantic concepts. The main challenge lies in the amount of variation in low–level visual and auditory features and game–specific rules. In order to adequately and flexibly interpret its meaning, they bridged the semantic gap between richness of user semantics and the simplicity of available low-level features. To address this issue, they have proposed BBN in [12] to link low–level features with high-level semantic concepts. BBN consumes much time than CNN but it is effective for detecting the moving events in a sequence of frames when compared to CNN.

2.4 APPROACHES FOR EXCITEMENT DETECTION

2.4.1 Using Speech to Text Synthesis

In [16], excitement score is computed based on audio tone analysis and speech to text analysis. While tone can say a lot about how excited the commentator is while describing a shot, excitement level can also be gauged from another source, that is, the expressions used. A dictionary of 60 expressions (words and phrases) indicative of excitement (e.g. great shot, fantastic) is created and assign to each of them with the excitement scores ranging from 0 and 1. A speech to text service was used to obtain a transcript of commentator's speech and create an excitement score as an aggregate of scores of individual expressions in it.

2.4.2 Using Voice Tone Analysis

In [16], the excitement of the commentator's tone is identified by employing the deep SoundNet audio features. A linear SVM classifier was used for modeling. For negative examples, audio tracks containing regular speech, music, regular cheer (without commentator excitement) and other kinds of sounds which do not have an excited commentator were used. In total, the training set for audio based commentator excitement recognition consisted of 131 positive and 217 negative samples. The leave-one-out cross validation accuracy on the training set was 81.3%.

2.4.3 Using Likelihood Model

In [2], the 14 audio features (i.e., ZCR, pitch period, and 12 MFCCs) extracted from each audio frame constitute a 14-dimensional (14-D) feature vector. Their attempt was to measure the likelihood of a group of feature vectors (e.g., audio clip) belonging to a certain audio types. They have defined five audio types for baseball games:

1. ball hit
2. cheering
3. music
4. speech
5. speech with music background.

Among these audio types, the two types, music and speech with music background, usually appear in the commercial segments, while the other three types usually appear in running commentary and exciting events. They have modelled each audio type using a GMM to describe the distribution of its feature vectors.

2.4.4 Using Signal Processing

During exciting events, spectator's cheer and commentator's speech becomes louder and more rapid. In [10, 15], two popular audio content analysis techniques- short-time audio energy and zero crossing rate (ZCR) are used for extracting excitement clip. A particular video frame is considered as an excitement frame if the product of its audio excitement and ZCR exceeds the threshold. [6] uses similar approach for identifying the excitement in the speech.

2.5 OBSERVATION FROM THE SURVEY

The system is proposed to work on extraction of highlights on cricket video. The idea that replays contain all the highlights of the sports match will not work for all situations. In cricket, key events like milestones don't have any replay, hence extracting the replays as highlights is not advisable. Replays are removed considering them as the repetitive events. Logo transition and the gradual transition is applicable for replay detection and not for the advertisements. The replay and advertisements can be removed by checking the absence of scorecard.

There is no particular method to eliminate unrelated events in the field like commentators, interviews. The boundary is segmented by using the event annotations which is comparatively preferred.

Detecting excitement using Speech to Text and tone analysis is not a good method as it will not work in the crowded ground. Excitement is detected by analyzing the audio by calculating the zero crossing rate is suitable method for cricket video. Removing the non excitement clips (audio based) before without using the visual parameters will not work for sports like cricket. Bayesian networks are used for concept detection which is computationally expensive.

Visual marker detection by detecting player's expression is not suitable for cricket. Identifying the highlights based on umpire's decision is not possible in all situations. Event detection for all frames plays an important role in identifying the highlights.

Key events in cricket can be detected by comparing the scores. This method is suitable for identifying the most of the highlight events.

CHAPTER 3

REQUIREMENTS ANALYSIS

3.1 FUNCTIONAL REQUIREMENTS

The system outputs the highlights video for a given cricket video input. The output video should adhere to the following requirements:

- The output video should contain all the key event segments such as wickets, boundaries.
- The output video should contain all the video segments containing excitement.
- The output video should not contain replay/advertisement segments.

The system should adhere to the following requirements:

- The system must be able to detect the presence and absence of scorecard in the input video.
- The system must be able to classify the input video frames and annotate each frame with the event associated with it.
- The system must be optimized for the time and space complexities.

3.2 NON FUNCTIONAL REQUIREMENTS

3.2.1 User Interface

There must be a simple and easy to use user interface. The intermediate results such as video classification, OCR results are printed in the CSV file for easy perception. The output video clips are concatenated and stored in a new output file by using movie editing tools.

3.2.2 Hardware

A system capable of running a python program with a minimum of 2GB RAM is required.

3.2.3 Software

- Operating System: Windows
- Programming Language: Python
- Tools: Opencv, Keras, Moviepy, Pytesseract.

3.2.4 Performance

The system must be optimized, reliable, consistent and available all the time.

3.3 CONSTRAINTS AND ASSUMPTIONS

3.3.1 Constraints

- The system would work effectively only for the BBL Cricket match videos.
- More than 4000 images are captured manually for CNN classification in order to attain better results.
- The crucial events such as last over balls may not be detected by using the scorecard recognition and excitement.
- The match winning overs will have too much of excitement even for non key events and it is difficult to distinguish the ball segments with excitement.
- If the input video doesn't contain the key events as per the pre coded criteria, no output video will be generated.
- The accuracy of the system is calculated only by the analysis of confusion matrix.

3.3.2 Assumptions

- The fours, sixes, wickets and those shots containing commentator's excitement and crowd cheers are the only highlights
- The fours, sixes and wickets can be classified by using the scorecard alone.
- All the key events are detected only by analyzing the scorecard and the audio.
- OCR detects and recognizes all the scorecards effectively
- The scorecard will display the exact score of that particular frame without any typos.

CHAPTER 4

SYSTEM DESIGN

4.1 SYSTEM ARCHITECTURE

The block diagram of the entire system is shown in the figure 4.1. The system for automatically extracting the highlights based on scorecard and excitement in the audio has been developed.

The system aims at extracting the highlighted clips from the unprocessed cricket match video which is given as the input. Each frames are checked for the presence of scorecard and classified as live frames. The events like batting, bowling, field view, commentators, crowd are identified in the live frames by CNN classifiers. Using the event tagged live frames, the shot boundaries are detected and segmented. The scores of two consecutive shots are recognized and compared for detecting the key events like boundaries and wickets. If detected then it is considered as highlights. The audio from input video is extracted. The average audio energy of the whole input video is calculated. The average audio energy of each shot is calculated and checked with the average audio energy of the whole input match. If it exceeds then that shot is considered as the highlights. Both the highlights are combined to generate the final highlights video.

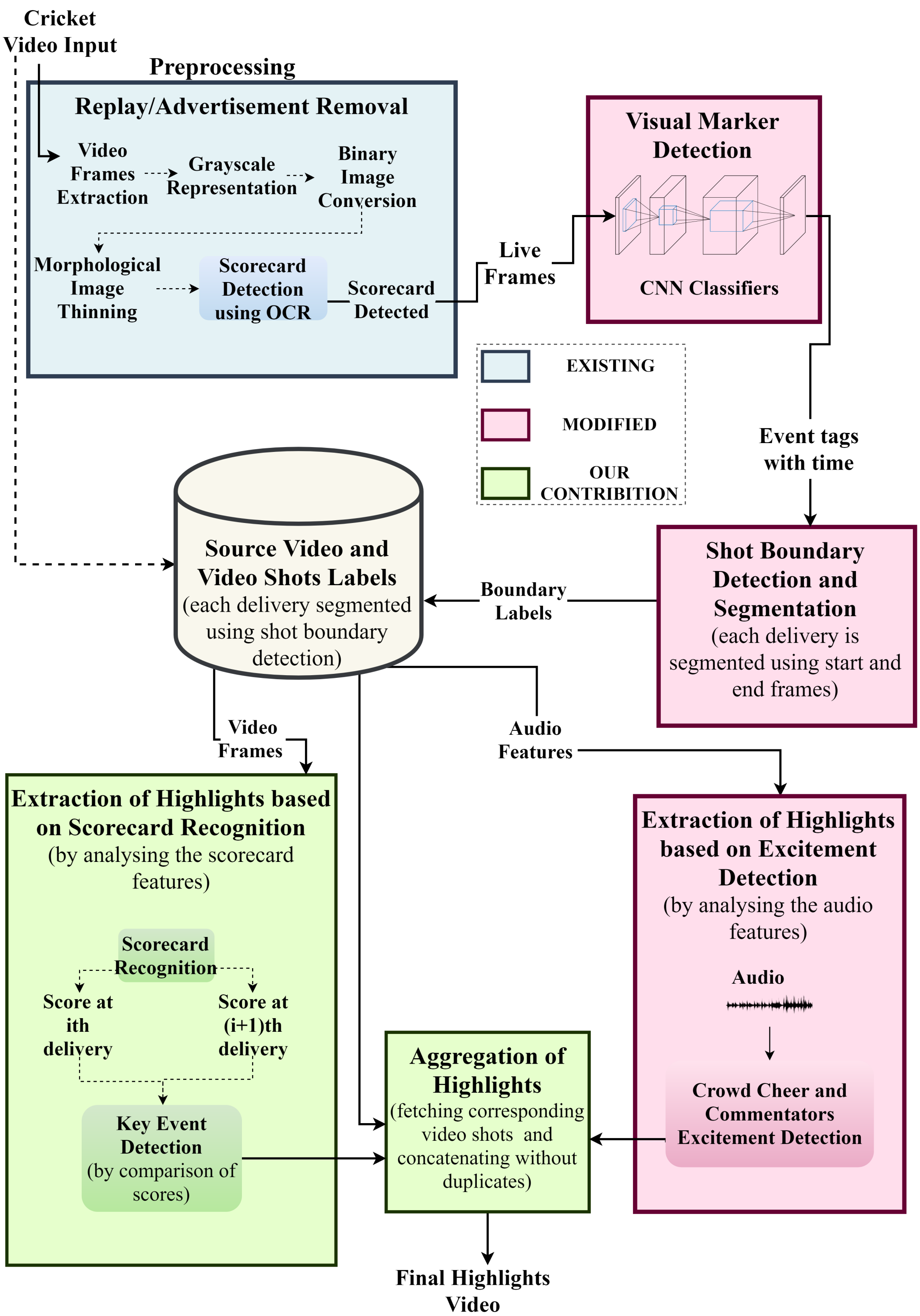

Figure 4.1 System Architecture

4.2 MODULE DESIGN

4.2.1 Preprocessing

Replays are the repetitive video shots of the key events in the cricket match. Advertisements are promotional video segments that are completely unrelated video shots for highlights extraction. Therefore, these shots are removed as they are considered to be unwanted. This module is used for detection and removal of replays and advertisements by checking the scorecard in the video frames. Absence and presence of the scorecard are used to detect replay/advertisement frames and live frames, respectively. The source video is converted into frames. In order to reduce the ambiguity for the OCR to detect the scorecard captions accurately we convert each frame into a morphed image. First, the frames are converted to gray scale representation. Then it is converted to binary image. The brightness and contrast of each frame are increased slightly in order to represent the contour of the captions more precisely. OCR detects and recognizes the scorecard from each frame. Live frames are those which has scorecard are classified and are given to visual marker detection for event classification. Figure 4.2 shows the preprocessing of the input video.

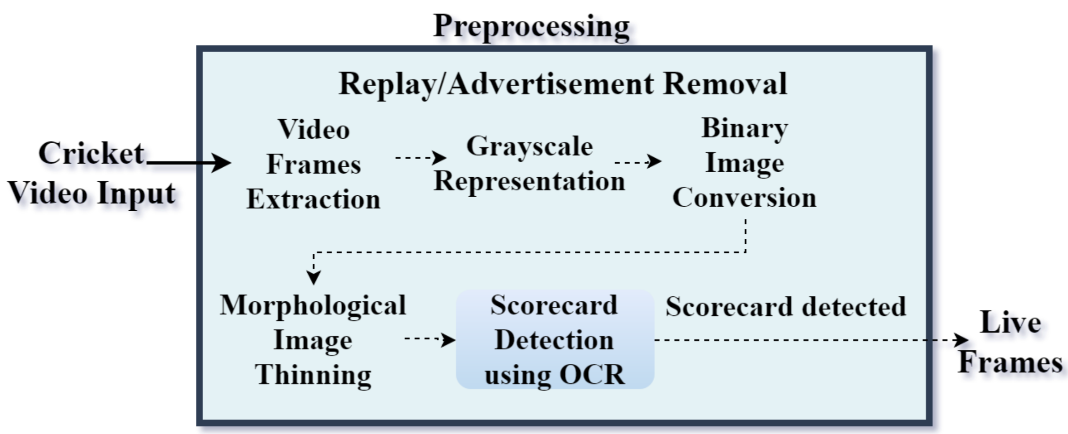

Figure 4.2 Workflow of Preprocessing

4.2.2 Visual Marker Detection

Event detection and classification is important for shot boundary detection and segmentation. Visual marker detection detects events like batting, bowling, crowd, interviews, commentators etc. from the live frames using trained CNN classifiers. The Multiple classes present in the video are pitch view, commentators, interview, field view, crowd, closeup, players gathering, long shot and other undefinable classes. In order to reduce the dimension of the multi class classification problem in detecting the events accurately, ensemble classifiers are used. The One vs All scheme[5] is used here. The complexity of multi class classification is compensated by binarization strategy. The classifier is trained with more than 4000 images with each class containing the training and testing images in the range of 300 and 100 respectively. The trained models are loaded to predict these classes. The image is given to each classifier until it gets the positive output from a classifier. Each negative outputs of the classifier is subjected to the remaining classifiers. If all the classifier gives negative output, the frame is marked as unlabeled. The event tag along with the time of the event is given to the shot boundary detection and segmentation for grouping the frames into video segments. Layout of this module is shown in the figure 4.3.

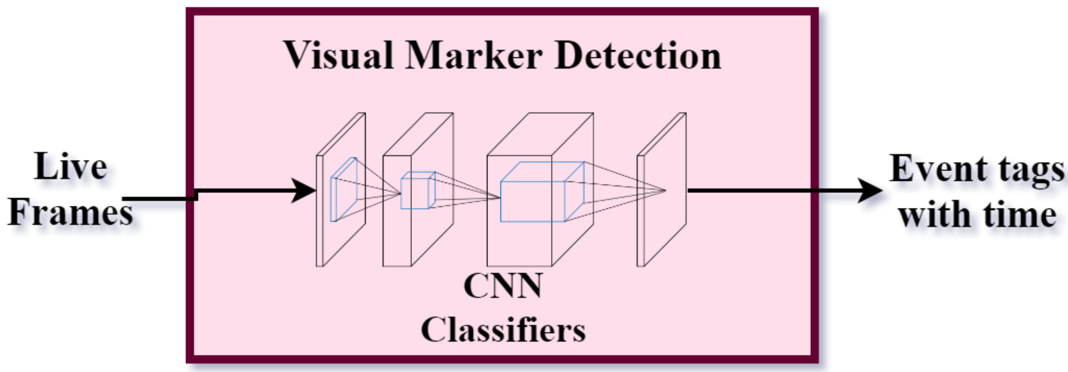

Figure 4.3 Workflow of Visual Marker Detection

4.2.3 Shot Boundary Detection and Segmentation

Each ball starts with a pitch view where a bowler who runs from the pavilion end, bowls the ball and at the other end batsman will be in position to hit the ball. This event is marked as the start boundary of a ball(video segment). The events like commentators view, interviews are shown in a cricket video only during the normal time i.e they are uninteresting parts of the video. The events like crowd view, players celebration are shown after a key event or an interesting event, they are used in detecting the end boundary. The start frame of a ball shot segment can be identified by using the frames annotated with the label as pitch view or bowling view. The end boundary is identified by variety of strategies.

1. Once the start boundary is identified, the subsequent frames starting from the start frame are checked for the labels crowd view, players celebration or gathering, commentators view, interviews, field view. If the frame with above labels are seen, then that frame will be marked as the end boundary of that ball shot segment.
2. When the replay or advertisement is started before detecting the end boundary label, then that is marked as the end boundary of that ball shot segment.
3. When both of the above mentioned strategies are failed, the frame soon before the start boundary of next segment will be marked as the end boundary of the current ball shot segment.

Then the following frames are checked to detect the next ball's start boundary and the process goes on until it reaches the end of the video. The end boundary of the last ball shot segment will be the final frame of the input when the above strategies fail. Using start and end of the delivery, shots are identified and segmented. The labels containing start and end time of the delivery is stored for further use in the following modules. Figure 4.4 shows the layout of the shot boundary detection.

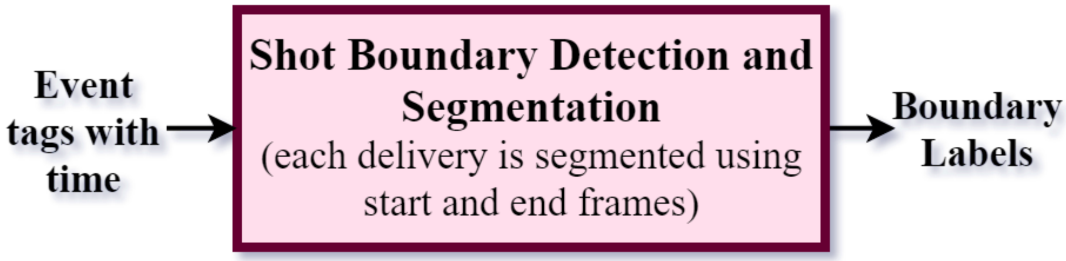

Figure 4.4 Workflow of Shot Boundary Detection and Segmentation

4.2.4 Extraction of Highlights based on Scorecard Recognition

It has been observed that the key events in the cricket video are mostly the events that involves fours, sixes and wickets. The score-card captions contains the score (i.e the combination of runs and wickets) in that particular time, overs, two batsman's name with their score and some other necessary tags associated in that frame. Each frame is cropped to the bottom left part, in order to recognize the score of that particular frame. By using the above mentioned approach the score of the start frame of each video shot segments are recognized from the cropped image using OCR. The difference of the score from the start frame of current video shot segment and the start frame of the next shot is calculated. If run difference exceeds 4, then it is added to highlights or if the wicket difference exceeds 0 then that video shot segment is labeled as highlights. Corresponding delivery labels are given as the output to the next module. Figure 4.5 shows the layout of the extraction based on scorecard recognition module.

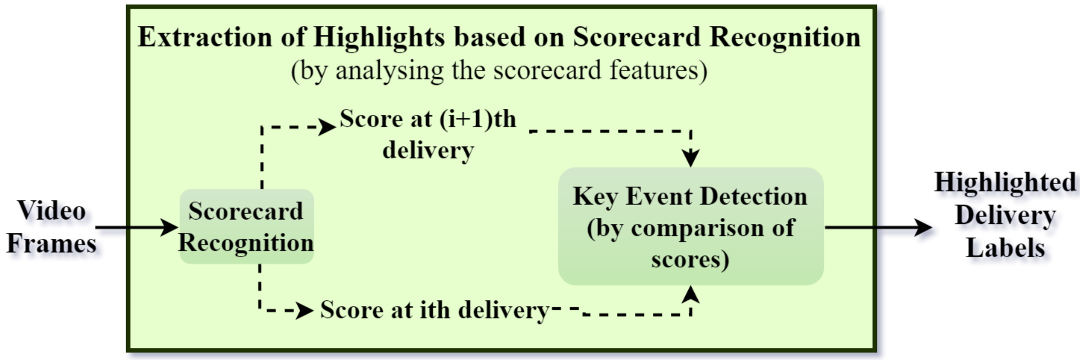

Figure 4.5 Workflow of Extraction of Highlights based on Scorecard Recognition

4.2.5 Extraction of Highlights based on Excitement Detection

Most of the key events in sports especially cricket will have crowd cheers and commentator's excitement in that segment. This module is used for extracting highlights based on excitement using crowd cheer and commentator's excitement from audio. The short time audio energies are used to detect the cheers and excitement. The short time audio energy in a time is computed by calculating the mean of the sum of the squares of all audio sample in that time. The short time audio energies of the whole video are calculated. The average of the short time audio energy for each time with the sliding window of length 5s is calculated. Normalized audio energy NE at a time is calculated by finding the ratio of audio energy at that time to the maximum audio energy present in the video. Then P_{audio} is calculated by finding the mean of the normalized audio energies. Now for each video segments $\Psi(n)$ is calculated by checking NEs with the P_{audio}. The mean of Ψ for each ball shot segments are calculated. The threshold for excitement for the input video is set to the sum of the mean and variance of mean of Ψ . A ball segment contains excitement if its mean of Ψ exceeds the thresold and are classified as highlights. Corresponding labels of the shots are considered as the highlights and given to the next module. Figure 4.6 shows the layout of the extraction based on excitement detection module.

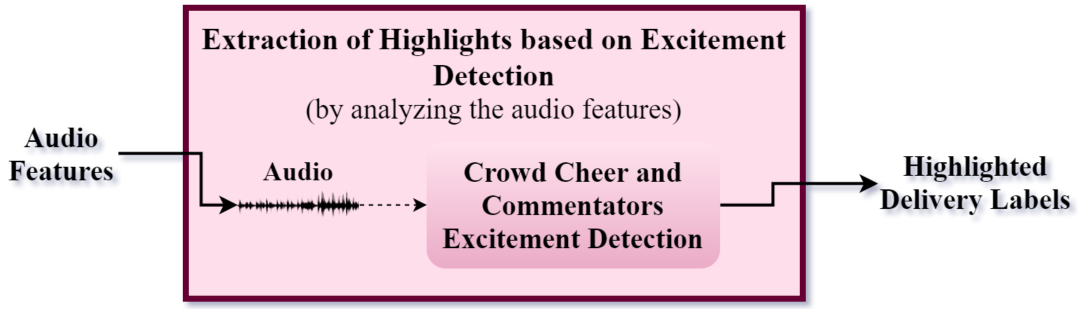

Figure 4.6 Workflow of Extraction of Highlights based on Excitement Detection

4.2.6 Aggregation of Highlights

The two lists from the scorecard recognition module and excitement detection module contains the ball shots labels which are classified as highlights based on scorecard recognition and excitement detection respectively. Each ball label detected from shot boundary detection module is checked for its presence in either of the lists or both. If the ball is present in any of those lists, then it is added to final highlights list. The final list is the combination of the two given lists without duplicates. The corresponding video shots for each highlight labels are fetched from the database by using the start and end boundaries of those ball labels. The extracted video clips are the highlighted clips by the system which are then concatenated to generate the final highlights video. The final highlights video is the summarized version of the input cricket video which has the key events based scorecard recognition and excitement detection. Figure 4.7 shows the layout of the aggregation.

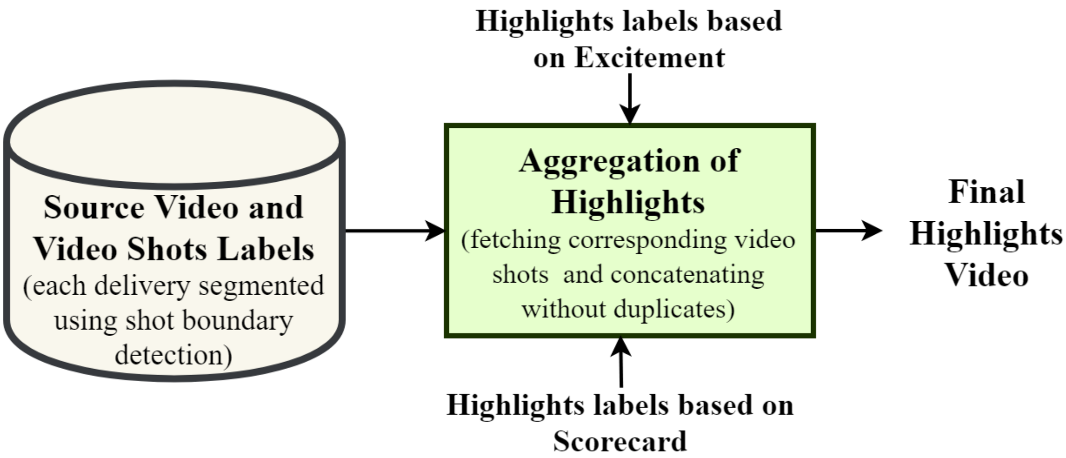

Figure 4.7 Workflow of Aggregation of Highlights

4.3 COMPLEXITY ANALYSIS

4.3.1 Time Complexity

Time complexity of each module of the system is shown in table 4.1.

Table 4.1 Time complexity of various modules

S. No.	Module	Complexity
1	Preprocessing	$O(t_{\text{ocr}})$
2	Visual marker detection	$O(Nt_c)$
3	Shot boundary detection and segmentation	$O(T)$
4	Extraction of highlights based on scorecard recognition	$O(B)$
5	Extraction of highlights based on excitement detection	$O(T+B)$
6	Aggregation of highlights	$O(t_{aggr}t_{out})$

- T is the duration of the input video
- t_{ocr} is the computation time for OCR to detect scorecard in one frame
- t_{class} is the run time of each classifier
- t_{aggr} is the aggregation time for generating 1s video output
- t_{out} is the duration of output video in seconds

- N is the number of classifiers
- B is the number of ball shot segments present in the input

4.3.2 Complexity of the project

- The accuracy of the system lies in event tagging, shot segmentation.
- The accuracy of the shot boundary detection depends on the CNN classifiers and the OCR which does replay/advertisement removal. Shot boundary segmentation can wrongly segment if any frame is falsely classified as start of the ball.
- Milestones such as 100s, 50s, 5 wicket hauls are also important events which the system finds difficult to identify by using score-card. Hence excitement and cheers are used to identify such key events.
- The accuracy of the output depends on shot boundary detection and segmentation. Error in this segmentation may leave out certain key event. So care should be taken in segmenting the shots.
- The efficiency of the visual marker detection also plays an important role. If the start of the ball is wrongly classified as the other event or the other events are classified as the start of the ball error occurs in the output.
- The excitement detection is another complex task. As it sets the threshold as mean variance sum, fails to detect excitement in a video which has more noise and in a video with non varying sound.

CHAPTER 5

SYSTEM DEVELOPMENT

The system removes advertisements, replays for preprocessing. The event in the live frames are detected using ensemble classifiers. Using event tags, shot boundaries are created. Highlights are created based on the scorecard and excitement. The overview of the entire system is given in the algorithm 1.

Algorithm 1: Overall Algorithm of the system

Input : A cricket match video
Output: A highlights video

```
input ← cricketmatch
liveframes := ReplayRemoval(input)
event[] := CNNClassifier(liveframes)
shot[] := ShotDetector(event[])
highlights := HighlightsExtractor(shot[])
```

5.1 PROTOTYPES OF THE MODULES

1. **Replay/Advertisement removal:** Input video is checked for scorecard. Frames with scorecard is live frames and given as output.
2. **Visual Marker Detection:** Live frames are given to CNN classifier for event tagging. Event tags are given as output.
3. **Shot boundary detection:** With event tags, Shot boundary is detected and segmented. Shots are given as output.
4. **Extraction of highlights based on scorecard recognition:**

Scores in 2 consecutive shots are compared and highlights are identified.

5. **Extraction of highlights based on Excitement Detection** The video shots are analyzed for the presence of excitement and then classified as highlights.
6. **Aggregation of highlights:** Both the highlights are combined to form final highlights.

5.2 DEPLOYMENT DETAILS

The deployment of the system requires opencv, tesseract, tensorflow and keras. Any python3.x can be used to deploy the system successfully. Anaconda is preferred to deploy the system in case of windows.

5.3 PREPROCESSING

Algorithm 2 shows the algorithm of the preprocessing step that removes the replay and advertisements.

Algorithm 2: Preprocessing

```
Input  : Source Input Video
Output: Live Frames
frames := generateFrames(Sourcevideo)
for each frame in frames[] do
    image := grayscale(frame)
    image := binaryconversion(image)
    text := OpticalCharacterRecognition(image)
    if is_scorecard(text) then
        mark_as_live(frame);
    else
        mark_as_replay(frame);
    end
end
```

5.4 VISUAL MARKER DETECTION

This module identifies the event in each frame which is shown in algorithm 3.

Algorithm 3: Visual Marker Detection

```
Input   : Live frames
Output: Event tags with time
model :=loadtrainedClassifier()
for each frame in frames[] do
    eventtag :=model.classify(frame)
    time := fetchframetime(frame)
    AppendCSV(eventtag,time)
end
```

5.5 SHOT BOUNDARY DETECTION AND SEGMENTATION

The shot boundary i.e from start of the ball to the commentators, interview, crowd umpire signal is detected and classified using the algorithm 4.

Algorithm 4: Shot Boundary Detection and Segmentation

```
Input   : Event tags
Output: Segmented delivery labels
i ←1
for each event in events[] do
    if is_bowling(event) then
        start[i] := time(event)
        AppendCSV(start)
    else if event in
      commentators,interview,crowd,umpiresignal then
        end[i] := time(event)
        AppendCSV(end))
end
```

5.6 EXTRACTION OF HIGHLIGHTS BASED ON SCORE-CARD RECOGNITION

The scores of 2 consecutive shots is compared to identify the highlights. It is shown in the algorithm 5.

Algorithm 5: Extraction highlights based on Scorecard recognition

Input : Video frames of each segmented deliveries
Output: Delivery labels that contains the highlights

```
i ←1
IsHighlights_scorecard[total_deliveries]
while i < total_deliveries do
    if get_score(get_frames(i)) - get_score(get_frames(i+1))
      >=4 then
        IsHighlights_scorecard[i]=1
    else if get_wicket(get_frames(i)) -
      get_wicket(get_frames(i+1)) >=1 then
        IsHighlights_scorecard[i]=1
    else
        IsHighlights_scorecard[i]=0
end
```

5.7 EXTRACTION OF HIGHLIGHTS BASED ON EXCITE-MENT DETECTION

The short time audio energy, normalised energy, P_{audio} are computed for the whole video and then $\Psi(n)$ is computed. It is shown in the algorithm 6.

Using Ψ, the presence of excitement in each video segments are detected and are given as input to the next module. It is shown in the algorithm 7.

Algorithm 6: Excitement Detection

Input : Audio of the input video and ball shot segments

Output: Ψ

for *each frame n in frames[]* **do**

    Compute audio energy E(n)

$$E(n) = \frac{1}{V} \sum_{(n-1)V+1}^{nV} x(m)^2$$

    where x(m) is audio sample at m and V is the number of audio samples corresponding to one video frame.

end

for *each frame in frames[] n* **do**

    Compute average audio energy AE(n)

$$AE(n) = \frac{1}{L} \sum_{i=0}^{L-1} E(n+L)$$

    where L is the length of sliding window

end

for *each frame in frames[] n* **do**

    Compute normalised audio energy NE(n)

$$NE(n) = \frac{AE(n)}{max_{1 \leq i \leq N} AE(i)}$$

end

for *each frame in frames[] n* **do**

$$\Psi(n) = \begin{cases} 1, & \text{if } NE(n) \geq P_{audio} \\ 0, & \text{otherwise} \end{cases}$$

    where P_{audio} is the is mean of NE(n)

end

Algorithm 7: Detecting the presence of excitement

Input : Ψ and video shot segments

Output: Highlight labels from excitement detection

for *each shot in shots[] m* **do**

$$Ex(m) = mean(\Psi(n))$$

end

for *each shot in shots[] m* **do**

$$IsHighlights\_excitement(m) = \begin{cases} 1, & \text{if } Ex(m) \geq Thres \\ 0, & \text{otherwise} \end{cases}$$

where $Thres := mean(Ex) + variance(Ex)$ is the threshold value to detect the presence of excitement in each ball segment

end

5.8 AGGREGATION OF HIGHLIGHTS

The two lists containing highlights labels are combined and the corresponding video shots are fetched and concatenated to generate final highlights video. It is shown in the algorithm 8.

Algorithm 8: Aggregation of Highlights

Input : Highlight labels from scorecard recognition and excitement detection

Output: Video shot segments with excitement

Labels1[] := IsHighlights_scorecard

Labels2[] := IsHighlights_excitement

Result[] := Label1[] $\cup Label2[]$

ResultVideo[] := FetchVideoFromDatabase(Result[])

FinalVideo := ConcatenateVideo(ResultVideo[])

CHAPTER 6

RESULTS AND DISCUSSION

This chapter presents the experiments designed to evaluate the performance of the proposed system. Each module of the system was also tested separately using the generated intermediate results. The results of the particular module testing as well as the entire system testing are summarized below. The results of these experiments are also reported along with the discussion. The details of the dataset are also provided in the following section.

6.1 DATASET FOR TESTING

The input to the system is the cricket match video. For performance evaluation we selected a dataset comprising of BBL cricket videos from YouTube. The dataset includes 6 videos of 19 hours from BT sport broadcaster. Cricket videos consist of replays, live match, advertisements, interviews. Each video has a frame resolution of 720p and MP4 format. The videos represent different illumination conditions (i.e., daylight, artificial lights). The dataset videos are comprised of different camera shots, i.e., pitch view, long view, field view, pavilion view, close-up view of the players, commentators view, crowd shots, etc. The training data frames are generated from the broadcasters videos and its appropriate class labels are labeled manually. CNNs are trained with 75% frames of our dataset and rest of the 25% frames are kept for testing. The training and testing images are put in separate folders for training and validation purpose. Each classes are trained with the range of

300 images and tested with the range of 100 images. The input videos for all the test cases are manually trimmed for demonstration and evaluation.

6.2 OUTPUT OBTAINED IN VARIOUS STAGES

This section shows the results of each module obtained during execution of the system.

6.2.1 Preprocessing

The output of preprocessing module is 'preprocessing.csv'. It's snapshot is shown in 6.1. The output file marks the time frame

time	scorecard present	scorecard absent
0s-30s	1	0
31s-56s	0	1
57s-144s	1	0
145s-156s	0	1
157s-182s	1	0
183s-229s	0	1
230s-253s	1	0
254s-267s	0	1

Figure 6.1 Output of Preprocessing

6.2.2 Visual Marker Detection

The output of visual marker detection module is 'frame_annotation.csv'. It's snapshot is shown in 6.3. It shows frames with corresponding event tag and time. The figure 6.2 shows the example of how the events are annotated by using ensemble of CNN classifiers.

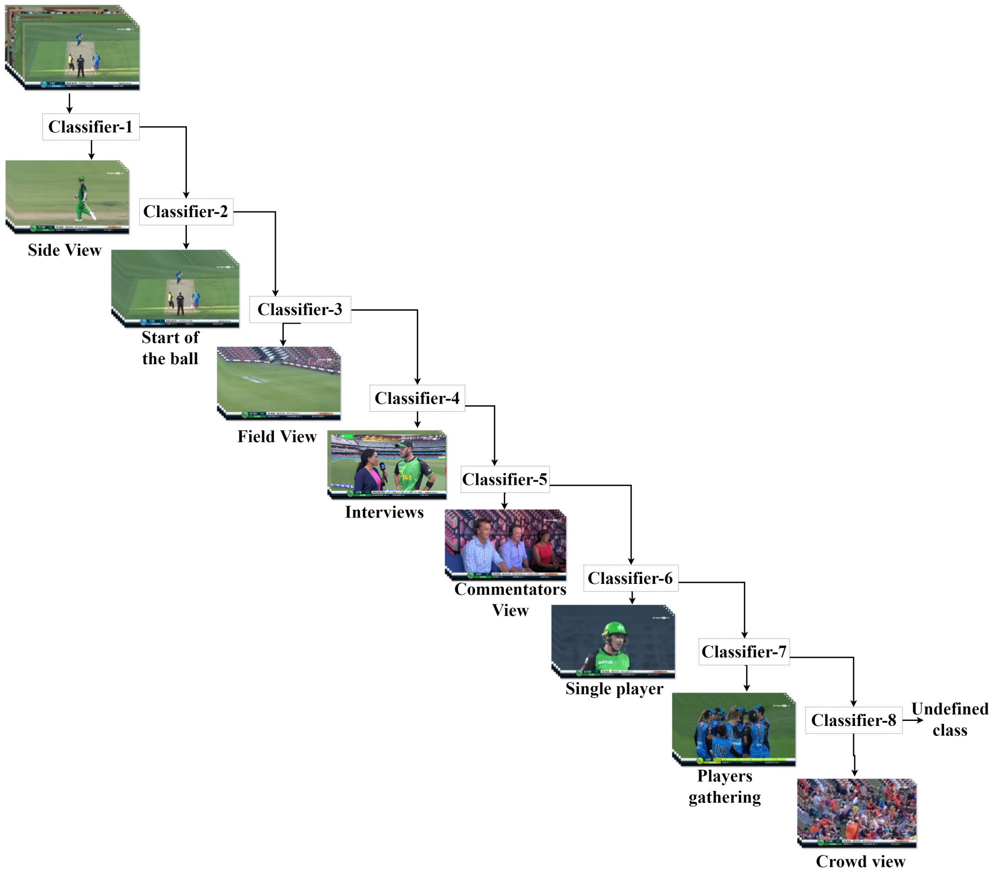

Figure 6.2 An Example of how the events are annotated

time	start of the ball	single_player	commentators	interviews	crowd	field_view	playergathering	others
14s	0	1	0	0	0	0	0	0
15s	1	0	0	0	0	0	0	0
16s	1	0	0	0	0	0	0	0
17s	0	0	0	0	0	0	0	1
18s	0	0	0	0	0	0	0	0
19s	0	0	0	0	0	0	0	0
20s	0	0	0	0	0	0	0	1
21s	0	0	0	0	0	0	0	0
22s	0	0	0	0	0	0	0	0
23s	0	0	0	0	0	0	0	0

Figure 6.3 Output of Visual Marker Detection

6.2.3 Shot Boundary Detection and Segmentation

The output of shot boundary detection and segmentation module is 'ball_boundary.csv'. It's snapshot is shown in 6.4. It shows the boundaries of every delivery in the input video.

count	start of the ball	end of the ball	score
1	15	31	26/1
2	62	89	32/1
3	89	129	33/1
4	129	145	33/1
5	166	183	37/1
6	231	254	41/1
7	270	292	45/1
8	306	325	49/1

Figure 6.4 Output of Shot Boundary Detection and Segmentation

6.2.4 Extraction of Highlights based on Scorecard Recognition

The output of extraction of highlights based on scorecard recognition module is 'scorecard_highlights.csv'. It's snapshot is shown in 6.5. It shows delivery segments with highlights and the type of the event for each segment.

count	start of the ball	end of the ball	type
1	15	31	Boundary
2	129	145	Boundary
3	166	183	Boundary
4	231	254	Boundary
5	270	292	Boundary
6	306	325	Boundary

Figure 6.5 Output of Extraction of Highlights based on Scorecard Recognition

The figure 6.6 illustrates how highlights are generated from the sample input video using key events detected by scorecard recognition. Ads and replays are identifies using OCR in the first step. Event tags are annotated eith each frames in second step. start and end of the ball are identified and then the scores are compared to find the key event.

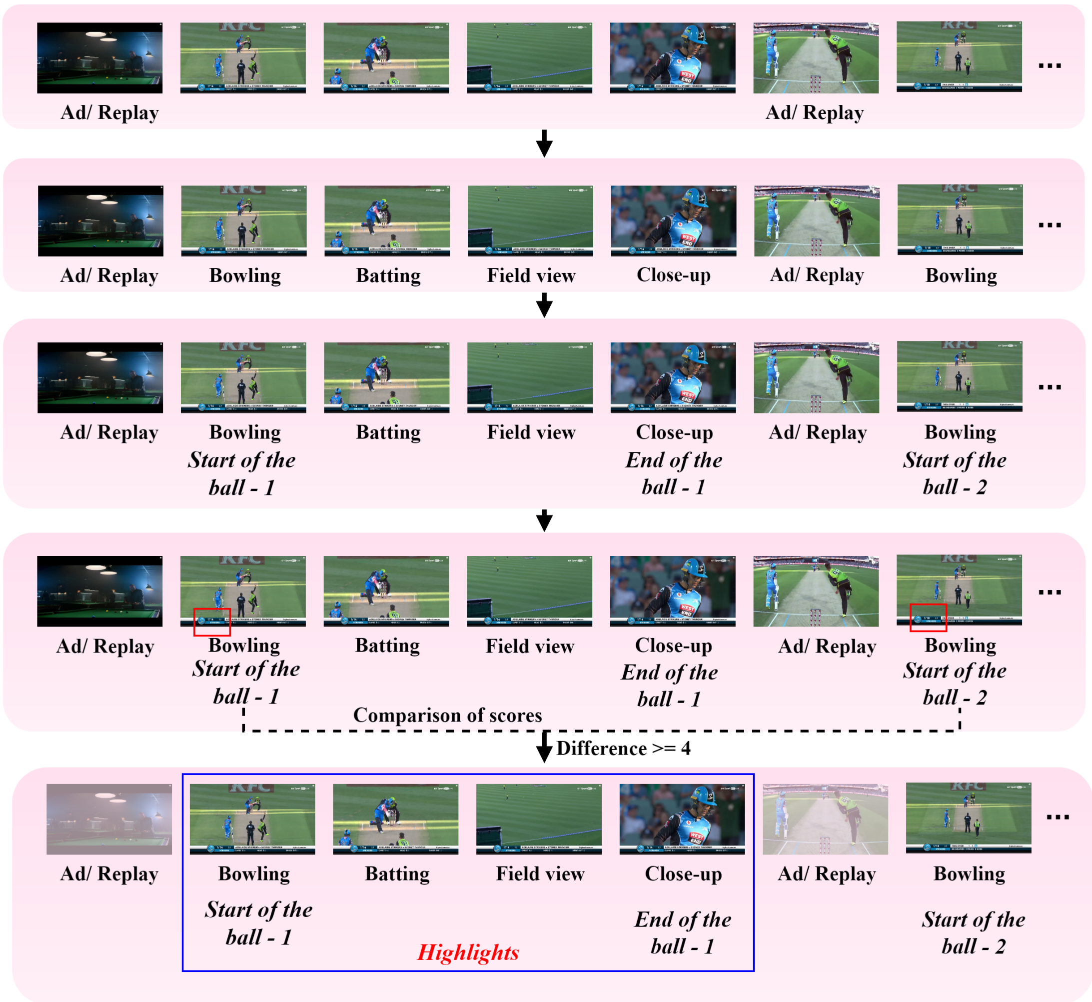

Figure 6.6 An example of how highlights are extracted using scorecard recognition

6.2.5 Extraction of Highlights based on Excitement Detection

The output of extraction of highlights based on excitement detection module is 'excitement.csv'. It's snapshot is shown in 6.7. It shows start and end boundary of each delivery segments which are classified as highlights by excitement present in the segment and the percentage of excitement present in each segment are also shown.

count	start of the ball	end of the ball	excitement
1	15	31	100%
2	231	254	65%
3	306	325	89%

Figure 6.7 Output of Extraction of Highlights based on Excitement Detection

6.2.6 Aggregation of highlights

The output of Aggregation of highlights module is shown in 6.8. It shows the generation of output video highlights containing both key events and excitement.

```
[MoviePy] >>>> Building video C:/FYP/Videos/6_BBL_Trim_highlights.mp4
[MoviePy] Writing audio in 6_BBL_Trim_highlightsTEMP_MPY_wvf_snd.mp3

100%|██████████| 2580/2580 [00:07<00:00, 337.92it/s]

[MoviePy] Done.
[MoviePy] Writing video C:/FYP/Videos/6_BBL_Trim_highlights.mp4

100%|██████████| 1755/1756 [01:14<00:00, 23.57it/s]

[MoviePy] Done.
[MoviePy] >>>> Video ready: C:/FYP/Videos/6_BBL_Trim_highlights.mp4

aggregation Time: 1.4417323311169943 minutes
```

Figure 6.8 Output of Aggregation of highlights

6.3 RESULTS

The system is developed using Python 3x. for programming and Keras package is used for building the CNNs. Moviepy tool is used for clipping and concatenating the highlights video. Six Big Bash League match videos in table 6.1 are used to test the system. Day and night matches are selected to test whether the system can support matches with variations in the contrast. The networks are trained using more than 4000 images for event detection in each frame. The final highlights video which contains the excited and interesting segments are validated against the highlights video generated manually. Precision, recall, accuracy, error rates are used as evaluation metrics.

6.4 PERFORMANCE EVALUATION

The performance of the system is evaluated using two major categories objective evaluation, subjective evaluation.

Objective evaluation criterion relies on metrics such as precision, recall, accuracy, error rates, etc. to measure the performance of the system. Following are some evaluation metrics:

- **True Positive (TP):** It refers to the positive samples that are correctly labeled by the classifier.
- **True Negative (TN):** It refers to the negative samples that are correctly labeled by the classifier.
- **False Positive (FP):** It refers to the negative samples that are incorrectly labeled as positive by the classifier.
- **False Negative (FN):** It refers to the positive samples that are incorrectly labeled as negative by the classifier.
- **Precision Rate (PR):** It is a ratio of number of correctly labeled events (or frames), TP, to the total number of events detected.

$$Precision = \frac{TP}{TP+FP} \tag{6.1}$$

- **Recall Rate (RR):** It is a ratio of true detection rate with respect to the actual events (frames) in the video.

$$Recall = \frac{TP}{TP+FN} \tag{6.2}$$

- **Error Rate (ER):** It is a ratio of the miss labeled events (both false positives and false negatives) to the total number of events examined.

$$ErrorRate = \frac{FP + FN}{TP + TN + FP + FN} \tag{6.3}$$

- **Accuracy Rate (AR):** It is a ratio of the correctly labeled events (both true positives and true negatives) to the total number of events.

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \tag{6.4}$$

The cricket videos which are used for testing is shown in 6.1.

Table 6.1 Cricket videos used for testing

S. No.	Name of the match	Date
1	Adelide Striker v Sydney Thunder	Dec 22, 2017
2	Perth Scorchers v Melbourne Stars	Dec 26, 2017
3	Melbourne Renegades v Sydney Sixers	Jan 03, 2018
4	Hobart Hurricanes v Sydney Sixers	Jan 08, 2018
5	Melbourne Stars v Sydney Sixers	Jan 16, 2018
6	Sydney Sixers v Melbourne Stars	Jan 23, 2018

6.4.1 Preprocessing

The preprocessing module is evaluated and precision, recall, accuracy, error rate are calculated to identify it's performance. Table 6.2 contains result of the preprocessing module. This module identifies the live frames with the an average accuracy of 97%.

Table 6.2 Preprocessing results

S. No.	Precision	Recall	Accuracy	Error
1	95	94	95	5
2	94	94	94	6
3	97	95	95	5
4	96	96	96	4
5	93	91	92	8
6	97	98	97	3

6.4.2 Visual Marker Detection

Table 6.3 shows the performance measures of visual marker detection. Visual marker detection plays a fundamental part in extracting the highlights by detecting the events with decent accuracy of around 80%.

Table 6.3 Visual marker detection overall results

S. No.	Precision	Recall	Accuracy	Error
1	83	81	82	18
2	79	80	79	21
3	81	84	82	18
4	82	85	83	17
5	81	83	82	18
6	83	82	81	19

The Table 6.4 shows the precision, recall, accuracy and error values for each detected classes with the count of each frames involved in evaluation.

Table 6.4 Visual marker detection class wise results

Event type	Frames	Precision	Recall	Accuracy	Error
Start of the ball	7340	87	85	88	12
Single player	7100	80	83	82	18
Commentators	92	73	72	72	28
Interviews	170	75	71	73	27
Crowd view	2703	77	78	77	23
Field view	3681	83	79	81	19
Players Gathering	2492	79	82	80	20

6.4.3 Shot Boundary Detection and Segmentation

Table 6.5 shows the performance measures of shot boundary detection where shots are created using the event tag of the corresponding frames.

Table 6.5 Shot Boundary Detection and Segmentation results

S. No.	Precision	Recall	Accuracy	Error
1	91	90	90	10
2	89	89	89	11
3	92	93	92	8
4	88	89	88	12
5	92	90	92	8
6	89	90	90	10

6.4.4 Extraction of Highlights based on Scorecard Recognition

Table 6.6 shows the result of scorecard recognition where highlights shots are created using score present in the scorecard.

Table 6.6 Highlights based on scorecard recognition results

S. No.	Precision	Recall	Accuracy	Error
1	93	92	92	8
2	92	91	91	9
3	94	95	94	6
4	90	92	90	10
5	94	92	92	8
6	92	93	93	7

6.4.5 Extraction of Highlights based on Excitement Detection

Table 6.7 shows the performance of excitement detection which account for detecting the video shot containing excitement. The results obtained are compared against the excitement shots detected manually for evaluation.

Table 6.7 Performance of excitement detection

S. No.	Total duration	Extracted clip duration	Recall	Precision
1	11040s	2208s	95	96
2	12840s	2105s	98	97
3	12540s	1890s	97	98
4	11220s	2045s	96	96
5	11100s	1935s	95	95
6	12600s	2020s	97	97

6.4.6 Subjective Evaluation

Subjective evaluation is based on user feedback score or rating. The subjective nature of this mechanism makes it difficult to define a benchmark as the quality parameters may vary among different users.

We conducted a user study in order to determine the perceptual quality of the highlights generated by our approach. 10 highlighted clips generated by our algorithm were shown to 20 different cricket fans in the age group of 20-60 years. The participants were asked to rate the clips from 0 to 5. The description of the rating is shown in table 6.8. The

Table 6.8 Scale for user rating

Score	Level	Description
5	Perfect	Good highlights
4	Fair	Good highlights with few mistakes
3	Acceptable	Ok with both highlights and boring segments
2	Bad	Highlights with some unwanted segments
1	Poor	Highlights with many unwanted segments
0	Very poor	Most of the highlight segments are boring

graph in figure 6.9 shows the users ratings for all the key segments. It is seen that the system got an average rating of 4.4 out of 5 overall.

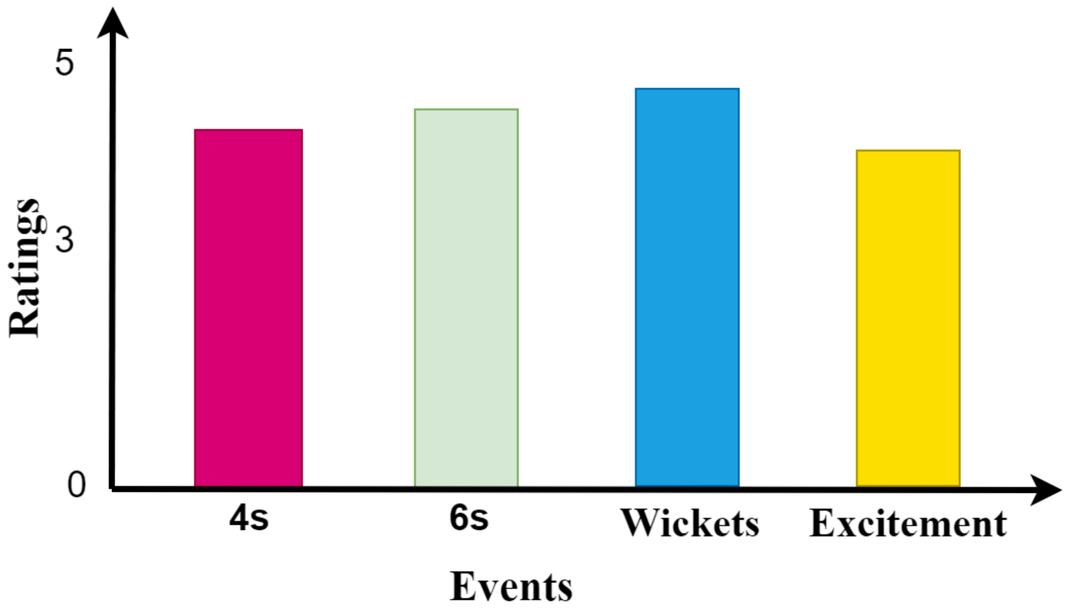

Figure 6.9 User ratings

6.5 DISCUSSION

6.5.1 Precision and Recall for each key event

Each key event in all the innings in the dataset is evaluated against the manually segmented key event segments. The precision and recall for the key events fours, sixes and wickets are calculated for each input video which are shown in table 6.9.

Table 6.9 Performance of each event

	4s		6s		Wickets	
S. No.	Precision	Recall	Precision	Recall	Precision	Recall
1	91	90	91	92	90	92
2	93	92	90	91	92	90
3	90	91	92	93	92	91
4	91	90	91	92	95	92
5	91	92	90	91	93	91
6	90	91	91	92	92	92

6.5.2 Evaluation of Scorecard recognition

The score recognized by the system and the actual score are compared. The score is recognized by the system using OCR. Figure 6.10 shows the evaluation chart which demonstrates the efficiency of OCR in scorecard recognition. The graph is drawn with score in the X axis and frame number in the Y axis. It is seen that the system detects the score appropriate to the actual score. The system struggles in score recognition for the frames occurring before and after the replays/ads are shown. Fading of scorecards and change in contrast are the factors that disturbs the OCR in recognizing the score more accurately. The system replaces the score at those frames by the score present in its neighbouring frames.

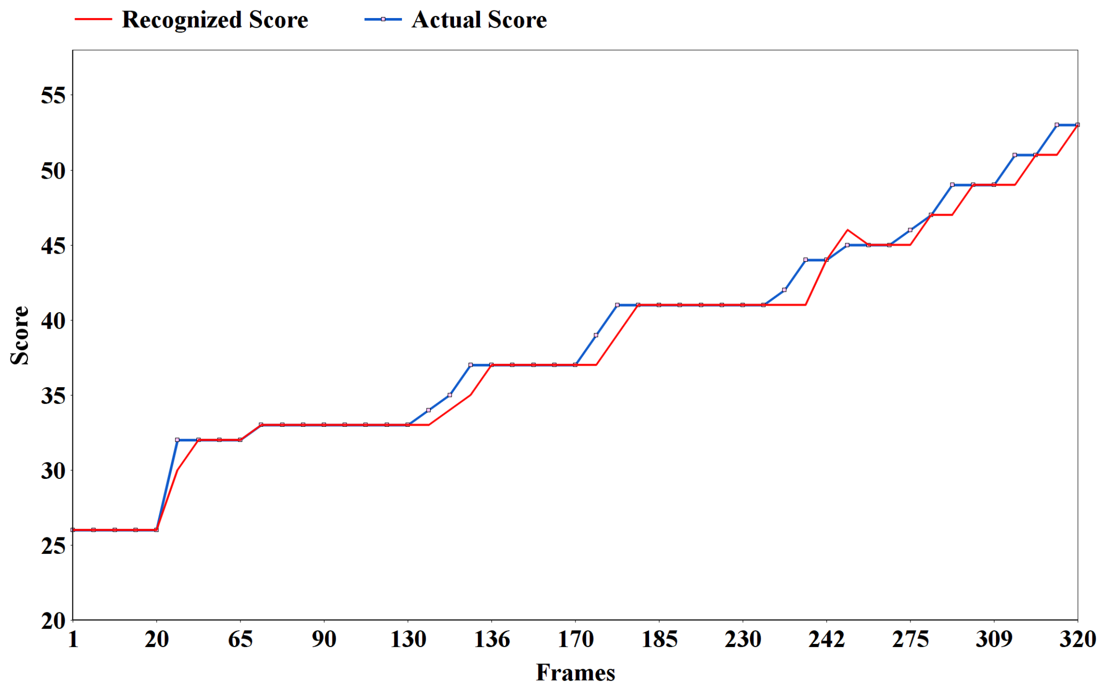

Figure 6.10 Evaluation of scorecard recognition

6.5.3 Comparison with the manually labelled highlights

The system is compared with manually labelled highlights for all the events in all the innings in the dataset. The comparison is given in the table 6.10. It is seen that the system has detected almost all the highlighted shots in comparison with the manually generated highlights.

Table 6.10 Comparison of highlighted video shot counts

	4s		6s		Wickets	
No.	Predicted	Actual	Predicted	Actual	Predicted	Actual
1	9	9	10	10	19	19
2	10	10	8	8	18	18
3	9	9	7	7	16	17
4	9	9	6	6	15	14
5	8	8	9	9	20	20
6	12	12	6	6	18	19

CHAPTER 7

CONCLUSION

7.1 CONCLUSION

We proposed a system to automatically generate cricket highlights, focusing on both event–driven and excitation–based features. Our system can achieve comparable results to manual highlights and that it yields acceptable results for cricket fans. The highlights without human intervention is demonstrated by dividing a cricket match into video shots and cues, such as replays, audio intensity, scoreboard, player celebration, and play field scenarios.

7.2 CONTRIBUTION

The proposed model generates highlights based on the scorecard and excitement features. The replays and advertisements are considered to be undesirable hence they are detected and removed. The highlight video segments will primarily relay on the detecting and segmenting the key events like fours, sixes, wickets. In order to have well defined boundaries, events like bowling, commentators view, interviews view, field view, crowd view are detected and are used in detecting and segmenting the video shot boundaries. The crowd cheering and excitement of the commentators are also used in selecting the video shot segments as highlights. The final highlights video is generated by concatenating all the obtained video shot segments and the intermediate results are shown in separate CSV files. The output video size will be very less when compared to the input video.

7.3 FUTURE WORK

The tasks which needs further exploration are as follows:

- The proposed system is designed for 20 over match. The system can also be generalized for 50 over match. This can be challenging because of its longer duration and the change in the intensity of sunlight.
- Separate highlights video for each key event(fours or sixes or wickets alone) can be done.
- Different broadcasters will have scorecard in different formats. Instead of specifying the scorecard's position, detection of the scorecard can be automated.
- Scorecard detection for the next frame will have to wait until the current frame comes out from the ensemble of classifiers. As both of the above process are independent, this waiting time can be reduced when both of them are executed parallelly using threads.

APPENDIX A

TEST CASES FOR EACH MODULE

This section provides the test cases for each of the module in the system developed.

A.1 PREPROCESSING

A.1.1 Test Pre–requisite

The standard cricket video with the necessity for highlights extraction is given as the input.

A.1.2 Description

Detects the presence and absence of scorecard captions in each frame. Removes replay and advertisement frames from the input video and the remaining frames(live frames) are given to the next module for event classification.

A.1.3 Test Cases

1. **Input:** Video with advertisements or replays.
   Expected output: Frames without advertisements and replays.

2. **Input:** Video with only advertisements or replays.
   Expected output: No live frames.

3. **Input:** Video with only advertisements.
   Expected output: No live frames.

4. **Input:** Video with only replays.
   Expected output: No live frames.

5. **Input:** Video without any advertisements or replays.
   Expected output: All frames as live frames.

The figure A.1 shows the output of for a video containing only replays or advertisements.

Figure A.1 An example of how our system reacts to the video with no live frames

A.2 VISUAL MARKER DETECTION

A.2.1 Test Pre–requisite

All the live frames i.e frames without replays and advertisements from the given input video.

A.2.2 Description

Identifies the event present in each given live frames using the ensemble of CNN classifiers. The event tags are annotated with each of the frames appropriately and are given for shot boundary detection and segmentation.

A.2.3 Test Cases

1. **Input:** Live video frames belonging to the defined classes.
   Expected output: List containing frames with its corresponding event tags.

2. **Input:** Live video frames belonging to the undefined classes.
   Expected output: List containing frames which are annotated with the name $other$.

The figure A.2 shows how our system detects the events(both defined and undefined) present in each frame using ensemble of CNN classifiers where the events, start of the ball, field view, crowd view, closeup shots are identified.

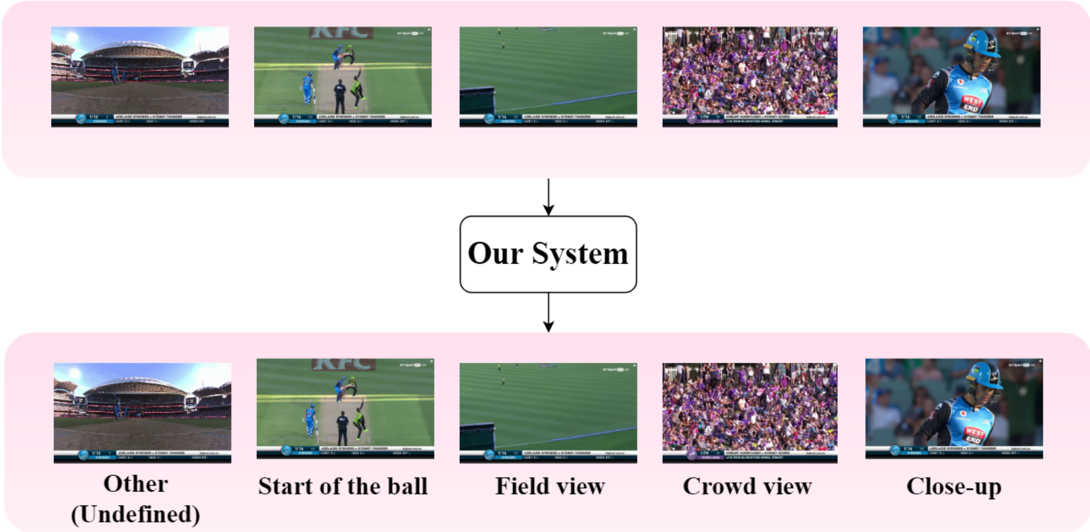

Figure A.2 An example of how our system classifies the events using CNN

A.3 SHOT BOUNDARY DETECTION AND SEGMENTATION

A.3.1 Test Pre–requisite

All the live frames annotated with the corresponding event are taken as input.

A.3.2 Description

Every ball shot segment starts with the bowler bowling the ball and ends with the crowd view or boundary view or until the bowler starts bowling the next ball. Mostly all the shots starts from the start of one ball to start of the next ball.

A.3.3 Test Cases

1. **Input:** List containing frames with bowling event tags.

   Expected output: List containing the boundary times of all the ball shot segments i.e deliveries with its start and end time.
2. **Input:** List containing frames without bowling event tags.

   Expected output: The output indicating the absence of the ball video segments in the given input.

The figure A.3 shows how our system responds to the video input which has no complete ball shot segments.

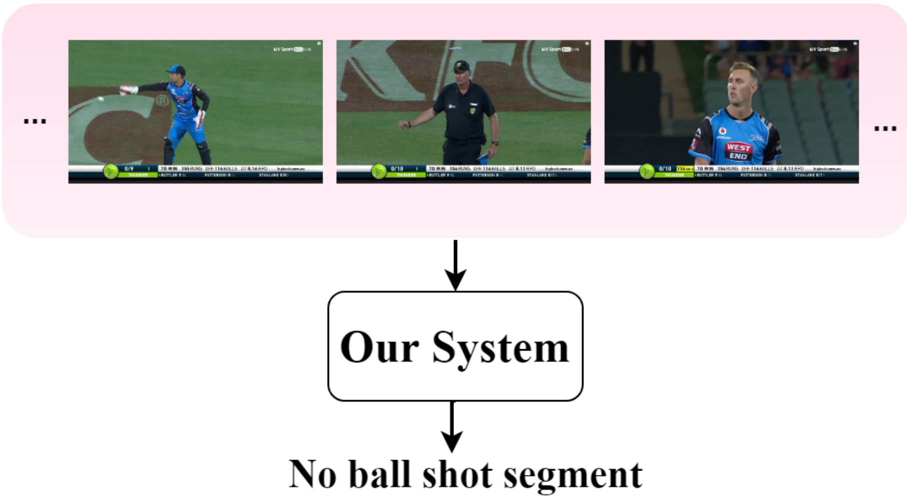

Figure A.3 An example of how our system reacts to the video without bowling events

A.4 EXTRACTION OF HIGHLIGHTS BASED ON SCORECARD RECOGNITION

A.4.1 Test Pre–requisite

Start and end boundaries of all the ball shot segments are given. Start frame of the segment are needed for scorecard recognition.

A.4.2 Description

Scores of the 2 consecutive shots are compared to identify key events. The key event segment labels are appended to the final output list.

A.4.3 Test Cases

- **Input:** Video containing the key events.

  Expected output: Key event segments should be detected accurately.

- **Input:** Video without containing the key events.

  Expected output: The output should convey the absence of key event segments in the input video.

The figure A.4 shows how our system responds to the video input which has no key event segment.

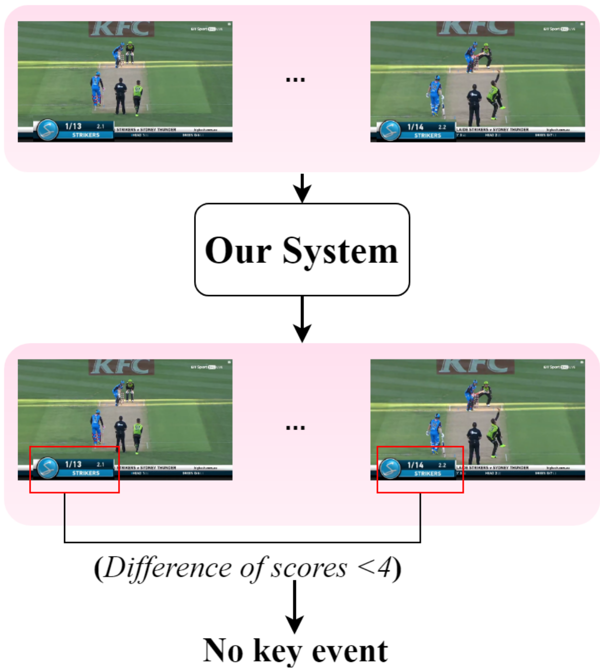

Figure A.4 An example of how our system reacts to the video without key events

A.5 EXTRACTION OF HIGHLIGHTS BASED ON EXCITEMENT DETECTION

A.5.1 Test Pre–requisite

Audio of the input video and start and end boundaries of all the ball shot segments are taken

A.5.2 Description

The excitement counts present in each segment is calculated using the normalized audio energy present in each audio frame. By comparing the excitement counts of all the segments, the segments that contains most excitement are included in the final list which represents the shot segment number.

A.5.3 Test Cases

- **Input:** Video containing the excited segments.
  Expected output: Excited segments should be detected accurately.
- **Input:** Video without containing the excited segments.
  Expected output: The output should convey the absence of excited segments in the input video.

A.6 AGGREGATION OF HIGHLIGHTS

A.6.1 Test Pre–requisite

Two lists which represents the index of the video segments which are detected as highlights by scorecoard recognition and excitement detection respectively.

A.6.2 Description

Corresponding video clips of all the highlighted segments are generated. All the generated clips are concatenated to form final highlights video.

A.6.3 Test Cases

- **Input:** Video containing the key event segments or excited segments.

  Expected output: Highlights segments should be generated accurately.

- **Input:** Video containing both key event segments or excited segments.

  Expected output: Highlights segments should be generated containing both key events and excitement.

- **Input:** Video containing neither excited segments nor key event segments.

  Expected output: The output should convey the absence of excited segments and key event segments in the input video.

APPENDIX B

CRICKET

Cricket is a bat-and-ball game played between two teams of eleven players on a field at the centre of which is a 20-metre (22-yard) pitch with a wicket at each end, each comprising two bails balanced on three stumps. Cricket is played with two teams (say A and B) normally of 11 players a side, one being the batting team while the other one is the fielding team. It is generally played on field with the main playing surface being called a pitch.

Team A will bat first and try to score as many runs as possible while the second team, team B, will bowl and field to make it as hard as possible for the batting team (A) to score these runs and to get them out . Once team A are all out or otherwise their batting is determined closed as per the laws, the teams then swap over. So team B will bat to try and beat the score (number of runs scored) set by team A. Team A will bowl and field and try and restrict Team B from beating their score / getting them all out before they do.

Cricket is a game for all - adults, young people, children, men and women, girls and boys. They play cricket all over the world - on the street, on the beach, in the local park, wherever they can find a place to play. Above all they have fun doing so!

B.1 MAIN ASPECTS OF PLAYING THE GAME

There are 6 key elements of cricket: batting, bowling, fielding, catching, wicket keeping, scoring runs and the appropriate camera view is selected during the broadcast. The camera change will play a major role in the cricket video broadcasting since each camera focuses on each player doing different activity.

B.2 FORMATS OF CRICKET

There are various formats ranging from Twenty20, played over a few hours with each team batting for a single innings of 20 overs, to Test matches, played over five days with unlimited overs and the teams each batting for two innings of unlimited length. Traditionally cricketers play in all-white kit, but in limited overs cricket they wear club or team colours. In addition to the basic kit, some players wear protective gear to prevent injury caused by the ball, which is a hard, solid spheroid made of compressed leather with a slightly raised sewn seam enclosing a cork core which is layered with tightly wound string.

APPENDIX C

VIDEO SUMMARIZATION

There have been tremendous needs of video processing applications to deal with abundantly available & accessible videos. One of the research areas of interest is video summarization that aims creating summary of video to enable a quick browsing of a collection of large video database. It is also useful for allied video processing applications like video indexing, retrieval etc. Video Summarization is a process of creating & presenting a meaningful abstract view of entire video within a short period of time. Mainly two types of video summarization techniques are available in the literature, viz. key frame based and video skimming. For key frame based video summarization, selection of key frames plays important role for effective, meaningful and efficient summarizing process.novel variant of video summarization, namely building a summary that depends on the particular aspect of a video the viewer focuses on. We refer to this as viewpoint. To infer what the desired viewpoint may be, we assume that several other videos are available, especially groups of videos, e.g., as folders on a persons phone or laptop. The semantic similarity between videos in a group vs. the dissimilarity between groups is used to produce viewpoint-specific summaries. For considering similarity as well as avoiding redundancy, output summary should be (A) diverse, (B) representative of videos in the same group, and (C) discriminative against videos in the different groups. To satisfy these requirements (A)-(C) simultaneously, we proposed a novel video summarization method from multiple groups of videos. Inspired by Fishers discriminant criteria, it selects summary by optimizing the combination of three terms (a) inner-summary, (b) innergroup, and (c) between-group variances defined on the feature representation of summary, which can simply represent (A)-(C). Moreover, we developed a novel dataset to investigate how well the generated summary reflects the underlying viewpoint. Quantitative and qualitative experiments conducted on the dataset.

As the name implies, video summarization is a mechanism for generating a short summary of a video, which can either be a sequence of stationary images (key frames) or moving images (video skims). Video can be summarized by two different ways which are as follows.

C.1 KEY FRAME BASED VIDEO SUMMARIZATION

These are also called representative frames, R-frames, still-image abstracts or static storyboard, and a set consists of a collection of salient images extracted from the underlying video source [2]. Following are some of the challenges that should be taken care while implementing Key frame based algorithm 1. Redundancy: frames with minor difference are selected as key frame. 2. When there are various changes in content it is difficult to make clustering.

C.2 VIDEO SKIM BASED VIDEO SUMMARIZATION

This is also called a moving-image abstract, moving story board, or summary sequence [2]. The original video is segmented into various parts which is a video clip with shorter duration. Each segment is joined by either a cut or a gradual effect. The trailer of movie is the best example for video skimming.

APPENDIX D

BIG BASH LEAGUE

The Big Bash League (BBL) is an Australian professional Twenty20 cricket league, which was established in 2011 by Cricket Australia. The Big Bash League replaced the previous competition, the KFC Twenty20 Big Bash, and features eight city-based franchises instead of the six state teams which had participated previously. The competition has been sponsored by fast food chicken outlet KFC since its inception. It is one of the two T20 cricket, alongside the Indian Premier League, to feature among the Top 10 Most Attended Sport Leagues in the world. BBL matches are played in Australia during the southern hemisphere summer, in the months of December, January and February.

D.1 TOURNAMENT FORMAT

Ben Cutting of Brisbane Heat batting against Melbourne Stars in 2014 Since the inception of the BBL in 2011, the tournament has followed the same format every year except the inaugural season.The first BBL season had 28 group stage matches, before expanding to 32 in the following season. Since the 2018-19 season, each team plays all other teams twice during a season, for a total of 56 regular season matches before the finals series..

In previous seasons of the tournament, the group stage matches were divided into eight rounds, with four matches played in each round. Each team played six other teams once during a season, and one team twice. This allowed for both Sydney and Melbourne (which have two teams each) to play 2 derbies within a single season.Each team played eight group stage matches, four at home and four away, before the top four ranked teams progressed to the semi finals.In the 2017/18 Season) the format changed so that there would be 40 group stage matches with each team playing 10 matches before the semi finals.The season was held over a similar time-frame thus resulting in more doubleheaders (one game afternoon, one game night) and teams playing more regularly.

The final of the tournament is played at the home ground of the highest-ranked team. The only exception to this rule was 2014-15 season when the final was played at a neutral venue (Manuka Oval), due to the 2015 Cricket World Cup.

In the 2018-19 season, the league introduced a ’bat flip’ (instead of a coin toss) to decide who would bat/bowl first.

D.2 CURRENT TEAMS

The competition features eight city-based franchises, instead of the six state-based teams which had previously competed in the KFC Twenty20 Big Bash. Each state’s capital city features one team, with Sydney and Melbourne featuring two. The team names and colours for all teams were officially announced on 6 April 2011.The Melbourne Derby and Sydney Derby matches are some of the most heavily attended matches during the league and are widely anticipated by the fans.The Scorchers and Sixers have also developed a rivalry between them over the years and their matches attract good crowds and TV ratings

REFERENCES

[1] M. E. Anjum, S. F. Ali, M. T. Hassan, and M. Adnan, "Video summarization: Sports highlights generation", In *INMIC*, pp. 142–147, Dec 2013.

[2] Chih-Chieh Cheng and Chiou-Ting Hsu, "Fusion of audio and motion information on hmm-based highlight extraction for baseball games", *IEEE Transactions on Multimedia*, vol. 8, num. 3, pp. 585–599, June 2006.

[3] A. Divakaran, R. Radhakrishnan, and K. A. Peker, "Motion activity-based extraction of key-frames from video shots", In *Proceedings. International Conference on Image Processing*, volume 1, pp. I–I, Sep. 2002.

[4] G. Evangelopoulos, A. Zlatintsi, A. Potamianos, P. Maragos, K. Rapantzikos, G. Skoumas, and Y. Avrithis, "Multimodal saliency and fusion for movie summarization based on aural, visual, and textual attention", *IEEE Transactions on Multimedia*, vol. 15, num. 7, pp. 1553–1568, Nov 2013.

[5] Mikel Galar, Alberto Fernández, Edurne Barrenechea, Humberto Bustince, and Francisco Herrera, "An overview of ensemble methods for binary classifiers in multi-class problems: Experimental study on one-vs-one and one-vs-all schemes", *Pattern Recogn.*, vol. 44, num. 8, pp. 1761–1776, August 2011.

[6] Taufiq Hasan, Hynek Bořil, Abhijeet Sangwan, and John H. L Hansen, "Multi-modal highlight generation for sports videos using an information-theoretic excitability measure", *EURASIP*

Journal on Advances in Signal Processing, vol. 2013, num. 1, p. 173, Nov 2013.

[7] Go Irie, Takashi Satou, Akira Kojima, Toshihiko Yamasaki, and Kiyoharu Aizawa, "Automatic trailer generation", In *Proceedings of the 18th ACM International Conference on Multimedia*, MM '10, pp. 839–842, New York, NY, USA, 2010, ACM.

[8] A. Javed, K. B. Bajwa, H. Malik, and A. Irtaza, "An efficient framework for automatic highlights generation from sports videos", *IEEE Signal Processing Letters*, vol. 23, num. 7, pp. 954–958, July 2016.

[9] N. Jmour, S. Zayen, and A. Abdelkrim, "Convolutional neural networks for image classification", In *2018 International Conference on Advanced Systems and Electric Technologies* $(IC_{A}SET), pp.397--402, March2018.$

[10] M. H. Kolekar, K. Palaniappan, and S. Sengupta, "Semantic event detection and classification in cricket video sequence", In *2008 Sixth Indian Conference on Computer Vision, Graphics Image Processing*, pp. 382–389, Dec 2008.

[11] M. H. Kolekar and S. Sengupta, "Event-importance based customized and automatic cricket highlight generation", In *2006 IEEE International Conference on Multimedia and Expo*, pp. 1617–1620, July 2006.

[12] M. H. Kolekar and S. Sengupta, "Bayesian network-based customized highlight generation for broadcast soccer videos", *IEEE Transactions on Broadcasting*, vol. 61, num. 2, pp. 195–209, June 2015.

[13] Y. S. Kumar, S. K. Gupta, B. R. Kiran, K. R. Ramakrishnan, and C. Bhattacharyya, “Automatic summarization of broadcast cricket videos”, In *2011 IEEE 15th International Symposium on Consumer Electronics (ISCE)*, pp. 222–225, June 2011.

[14] Yu-Fei Ma, Lie Lu, Hong-Jiang Zhang, and Mingjing Li, “A user attention model for video summarization”, In *Proceedings of the Tenth ACM International Conference on Multimedia*, MULTIMEDIA ’02, pp. 533–542, New York, NY, USA, 2002, ACM.

[15] Somnath Senguptak Maheshkumar H. Kolekar, “Caption content analysis based automated cricket highlight generation”, 2007.

[16] M. Merler, K. C. Mac, D. Joshi, Q. Nguyen, S. Hammer, J. Kent, J. Xiong, M. N. Do, J. R. Smith, and R. Feris, “Automatic curation of sports highlights using multimodal excitement features”, *IEEE Transactions on Multimedia*, vol. , pp. 1–1, 2018.

[17] A. Rav-Acha, Y. Pritch, and S. Peleg, “Making a long video short: Dynamic video synopsis”, In *2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR’06)*, volume 1, pp. 435–441, June 2006.

[18] Aravind Ravi, Harshwin Venugopal, Sruthy Paul, and Hamid R. Tizhoosh, “A dataset and preliminary results for umpire pose detection using SVM classification of deep features”, *CoRR*, vol. abs/1809.06217, 2018.

[19] X. Shao, C. Xu, and M. S. Kankanhalli, “Automatically generating summaries for musical video”, In *Proceedings 2003 International Conference on Image Processing (Cat. No.03CH37429)*, volume 2, pp. II–547, Sep. 2003.

[20] P. Shukla, H. Sadana, A. Bansal, D. Verma, C. Elmadjian, B. Raman, and M. Turk, "Automatic cricket highlight generation using event-driven and excitement-based features", In *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, pp. 1881–18818, June 2018.

[21] M. A. Smith and T. Kanade, "Video skimming and characterization through the combination of image and language understanding techniques", In *Proceedings of IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pp. 775–781, June 1997.

[22] Dian Tjondronegoro, Yi-Ping Phoebe Chen, and Binh Pham, "Sports video summarization using highlights and play-breaks", In *Proceedings of the 5th ACM SIGMM International Workshop on Multimedia Information Retrieval*, MIR '03, pp. 201–208, New York, NY, USA, 2003, ACM.

[23] J. Varghese and K. N. R. Nair, "An algorithmic approach for general video summarization", In *2015 Fifth International Conference on Advances in Computing and Communications (ICACC)*, pp. 7–11, Sep. 2015.

[24] Ke Zhang, Wei-Lun Chao, Fei Sha, and Kristen Grauman, "Video summarization with long short-term memory", *CoRR*, vol. abs/1605.08110, 2016.

[25] Z. Zhao, S. Jiang, Q. Huang, and G. Zhu, "Highlight summarization in sports video based on replay detection", In *2006 IEEE International Conference on Multimedia and Expo*, pp. 1613–1616, July 2006.