

Cairo University

Faculty of Engineering

Department of Computer Engineering

**LA**خ**SLY**



A Graduation Project Report Submitted

to

Faculty of Engineering, Cairo University

in Partial Fulfillment of the requirements of the degree

of

Bachelor of Science in Computer Engineering.

**Presented by**

Ahmed Salama Hamed Ahmed Maher

Moamen hassan attia Mohamed Talaat Mohame

**Supervised by**

Dr. MagdaFayek

2020

All rights reserved. This report may not be reproduced in whole or in part, by photocopying or other means, without the permission of the authors/department

**General Guidelines**

* The color of the cover page should be green
* Throughout your text, use the font type, font size, and spacing, as in this template. In general, Arial font should be used. Chapter headings should be of size 24. Sections should be of size 18, and regular text should be of size 12. Your text should be justified on both left and right sides.
* The reference list should be written using a font size of 10. Ensure that the references are written correctly and all fields are included. References should be ordered according to their appearance in the text “[1], [2], [3] … etc”
* The table of content is a tentative one. You could add more sections as required. However, the mentioned sections should be included in your report
* For the appendices, add any appendix you see necessary. Remove any appendix that is not applicable to your project. However, the feasibility study and user guide should be included
* Ensure that the report is clear and self-contained, such that any future interested reader could completely understand your project “to the extent of building another one similar to yours”
* Use figures as much as possible to clarify and enrich your discussion. You have to draw all figures yourself. Ensure that the figures are clear and their size is suitable.
* Any figure caption should be inserted below the figure. Figures within any chapter should be numbered starting from 1. For example, the first figure of chapter 2 should be “Figure 2.1”. Similarly, the fourth figure of chapter 3 should be “Figure 3.4”
* Any table caption should be inserted above the table. Tables within any chapter should be numbered starting from 1. For example, the first table of chapter 4 should be “Table 4.1”. Similarly, the seventh table of chapter 5 should be “Table 5.7”
* Copy and paste from any other source is not allowed by any shape. Even for the background knowledge, you have to use your own wording.
* The complete report should be submitted 48 hours before the final project demonstration day. Ensure that you would meet this deadline to avoid any late penalty

**Abstract**

Soccer is one of the most popular team sports all over the world. Most sports games are naturally organized into successive and alternating plays of offence and defence, Resulting in events such as goals or dangerous attacks. If a sport videos can be segmented according to these meaningful events, it then can be used in various applications to enhance their values and enrich the user's viewing experiences and also save the user’s time. According to this, soccer video summarization and analysis has recently attracted much research and a wide spectrum of possible applications have been considered.

Soccer video summarization and analysis is concerned with the extraction of valuable events by efficient and effective processing of combination of visual, audio and text information. However, one of the major limitations of current soccer analysis is the gap between the low level features such as (color, texture, shape and motion) and high level representation such as (shot types, shot length, and shot replays).

This thesis presents an automatic soccer video summarization system using a variety of techniques in the fields of image processing, machine learning. The proposed system is composed of five phases. Namely; in the pre-processing phase, the system reads the video stream into patches of frames. Then, in the shot Boundary phase is applied on every patch of frames to segment the video into seprate shots i.e. finding the transition from a camera into another. Then each shot is processed seperatly in the following three phases, the shot classification phase in concerned with classifiying each shot for example a wide shot is when the whole pitch is visible, A close shot is when the players faces are visible etc. then the event detection phase in which each shot is analyside to determine if a goal happens in it or not or if it the audio level in this shot is high with respect to the whole video or if the goal mouth appears in the shot . And lastly, the summarization phase in which the output of all previous phases is analysized to determine which shot is important and should include in the final video and which shots to discard

**الملخص**

**ACKNOWLEDGMENT**

We would like to express our sincere gratitude to Dr. Magda fayek for letting us

work on the exciting topic of sports video processing and giving us so much freedom to

explore and investigate new areas of video processing, providing invaluable personal and

professional guidance, being accessible all the time, and also convey our love for our parents, family and friends

**Table of Contents**

Abstract ...........................................................................................................…........................ iii

الملخص ........................................................................................................................................... iv

Acknowledgment ......................................................................................................................... v

Table of Contents .....................................................................................................................… vi

List of Figures ............................................................................................................................. ix

List of Tables .............................................. .............................................................................. x

List of Abbreviations .............................................. ..................................................................... xi

List of Symbols ............................................................................................................................ xii

Contacts ..................................................................................................................................... xiii

**Chapter 1: Introduction**

1.1. Motivation and Justification…..……………………………………………..……………………1

1.2. Project Objectives and Problem Definition………………………………….………………….1

1.3. Project Outcomes ............................................................................................................... 1

1.4. Document Organization ...................................................................................................... 1

**Chapter 2: Visibility Study**

2.1. Target Customers .............................................................................................................. 2

2.2. Market Survey ……………………......................................................................................... 2

2.2.1. Example of Subsection 1 .............................................................................................. 2

2.2.2. Example of Subsection 2 ............................................................................................. 2

2.2.2.1. Example of subsubsection .................................................................................. 2

2.3 Business Case and Financial Analysis ……………………………………………………………………. 2

**Chapter 3: Literature Survey**

3.1. Background on Topic1 ….................................................................................................... 4

3.2. Background on Topic 2 .........................................................................................................4

3.3. Comparative Study of Previous Work …......................................................................................................4

3.4. Implemented Approach .............................................................................. 4

**Chapter 4: System Design and Architecture**

4.1. Overview and Assumptions ................................................................................................ 5

4.2. System Architecture ………………………………...………………………………………....… 5

4.2.1. Block Diagram .............................................................................................................. 5

4.3. Module 1 …………………………………………………………………………………..……… 6

4.3.1. Functional Description .................................................................................................. 6

4.3.2. Modular Decomposition ................................................................................................ 6

4.3.3. Design Constraints ....................................................................................................... 6

4.3.4. Other Description of Module 1 ...................................................................................... 6

4.4. Module 2 …………………………………………………………………………………..…....… 6

4.4.1. Functional Description .................................................................................................. 6

4.4.2. Modular Decomposition ................................................................................................ 6

4.4.3. Design Constraints ....................................................................................................... 6

4.4.4. Other Description of Module 2 ...................................................................................... 6

**Chapter 5: System Testing and Verification**

5.1. Testing Setup ……………................................................................................................... 7

5.2. Testing Plan and Strategy................................................................................................... 7

5.2.1. Module Testing ............................................................................................................. 7

5.2.2. Integration Testing ........................................................................................................ 7

5.3. Test Schedule.......................................................................................................................7

5.4. Comparative Results to previous work……………………………….......................................7

**Chapter 6: Conclusions and Future Work**

6.1. Faced Challenges ................................................................................................................8

6.2. Gained Experience ............................................................................................................... 8

6.3. Conclusions ..............................................................................................................8

6.4. Future Work ........................................................................................................................ 8

References .................................................................................................................................. 9

Appendix A: Development Platforms and Tools ..…………………................ 10

Appendix B: Use Cases ............................................................................................................ 11

Appendix C: User Guide ............................................................................................................ 12

Appendix D: Code Documentation ............................................................................................ 13

Appendix E: Feasibility Study .................................................................................................... 14

**List of Figures**

Figure 1.1: Same caption as in the text ................................................................................. page

Figure 2.1: Same caption as in the text ................................................................................. page

**List of Tables**

Table 1.1: Same caption as in the text .................................................................................. page

Table 2.1: Same caption as in the text .................................................................................. page

**List of Abbreviation**

[The abbreviations should be put in an alphabetical order]

AI Artificial Intelligence

EA Evolutionary Algorithms

GA Genetic Algorithms

SA Simulated Annealing

VLSI Very Large Scale Integration

**List of Symbols**

[The symbols should be put in an alphabetical order. Greek symbols come first, followed by English symbols]

σ Noise standard deviation

B Buffer size

fop Operating frequency

**Contacts**

**Team Members**

|  |  |  |
| --- | --- | --- |
| Name | Email | Phone Number |
| Ahmed Salama Hamed | [asivo.ahmed@gmail.com](mailto:asivo.ahmed@gmail.com) | +2 01144697687 |
| Ahmed Maher | Demha.ahmed@yahoo.com | +2 01065725338 |
| Moamen Hassan Attia | moamenattia@outlook.com | +2 01200370750 |
| Mohamed Talaat Mohamed | [mohamedtalaat0111790@gmail.com](mailto:mohamedtalaat0111790@gmail.com) | +2 01115598525 |

**Supervisor**

|  |  |  |
| --- | --- | --- |
| Name | Email | Number |
| Dr. Magda Fayek | [magdafayek@gmail.com](mailto:magdafayek@gmail.com) | +2 01xxxxxxxxx |

This page is left intentionally empty

**Chapter 1: Introduction**

This chapter presents the importance of soccer video analysis and summarization. Soccer video matches always attract major sports audience. Recently, the amount of digitized video content has been increasing rapidly and watching a soccer match needs a lot of time, many TV fans of sport competitions prefer to watch a summary of soccer video matches. Presenting the problems that are facing the automatic soccer video summarization and the proposed solutions. Finally an overview about the organization of the thesis is shown at the end of the chapter.

* 1. **Motivation and Justification**

Automatic soccer video summarization systems extracts the most important events to produce general summaries for the important moments in which soccer viewers may be interested. Researchers have proposed many techniques to take full advantage of the fact that sport videos have typical and predictable structure, recurrent events, consistent features, and fixed number of camera views and angles.

The multimedia analysis tool, which could automatically parse soccer video and output required video clips or the most interesting events such as goals, dangerous attacks and free kicks, fans could go though many more games without spending much time. This can certainly entertain these fans and in turn popularize the sport itself. So, soccer video analysis is absolutely necessary.

Event detection in soccer video is a high level analysis, which needs an effective description of soccer video information and approaches to bridge the gap between low level features (color, texture, shape and motion) and high level representation such as (shot types, shot length, and shot replays) However, research in this field is far from enough. Shot is commonly used as an intermediate representation, but the standard for soccer video analysis has yet to be reached and other high level representations should be explored. This work has been inspired by this motivation.

We have two propellants which motivated us to develop an automated system for soccer match summarization. First, most people cannot watch the whole matches which are played on same time within different time zones because of lack of time. Second, coaches need to view the highlighted events to truly developing plans and evaluate the team players. From this point we concluded the importance of our proposed program to put a solution for the mentioned problems.

* 1. **Project Objectives and Problem Definition**

Analyzing general sport games is still an open problem because of the variance and

diversity of different games and different sports. Some former researchers have proposed many highlight summarization methods both for general sports game and for a specific kind of sports game (2). detected the play and break event in sports videos to generate the summary. Some other researchers summarize sports videos using slow motion replays (3). On the other hand, another group of researchers turn to study specific sport games such as soccer, basketball or diving (4).

The summarization phase is an essential part in almost any sytem trying to retrieve the important parts in a video. This field is undergoing rapid change, as computers are now prevalent in virtually every application, from games for children through the most sophisticated planning tools for governments and multinational firms.

When we are talking about a soccer game, we can refer to it as a continuous sports which means that if there is an existence of replay during the match, it can be an indicator of the occurrence of important event such as (goal, penalty shot and red / yellow card). Therefore the summarization process which we aim for can be recognized by a combination of these events.

The input is a soccer video match needed to be summarized using a computer based application, our concern here is to extract the most exciting events in the soccer game such as (goal, attacks, and other events) using our proposed application then output those events into summarized video.

In this thesis, we are going to highlight the most important events such as (goals, attacks, and the other events) to save the viewer's time, and try to improve on the research that has benn done in the technology of computer based summarization of sports.

* 1. **Project Outcomes**

Proposed solutions have been presented to avoid most of the problems discussed in the previous section, these solutions are involved in each of the stages of the proposed system . In the preprocessing stage, reading all frames of a video is clearly an overhead on the sytem as most soccer matches are two hours long and also the differences between two consequtive frames doesn’t carry much information because very little has changed between the frames so frame skipping is done to ensure that there is no redundancy in the consequitve fremes and also make the system much faster.

shot boundary detection phase is proposed to segments the whole input video stream into small video shots by applying a combination of image processing techniques and histogram comparison techniques. In the shot classification phase , the system applies different algorithms, namely; Grass dominant ratio extraction, Face Detection ,Deep learning and image processing techniques to classify the shot into one of the following classes :wide – medium- close – close out – logo. The logo classification is used for replay detection which is an important factor in determining which of the shots are important and which are not.

In The event detection phase each is shot in sconsedered seperatly and compared with the previous shot to determine if a goal has happened between the two shots by comparing the scoreboard in the upper left corner. Two methods are proposed to detect a goal event, Optical character recognition (OCR) and structural similarity image index (SSIM).

Lastly, the summarization phase which takes into account the output of the previous phases to decide if he shot is important and shoul be included in the final video or should be discarded. For example, if a shot contains a goal then it is important or if a shot has a goal mouth and the audio level is high then it is important or it’s a replay shot and a replay of course is for something important. Then combining the important shots only in a video and that is the output of the system

* 1. **Document Organization**

In this section, you have to give the organization of the report and a quick description of the following chapters.

**Chapter 2: Market Visibility Study**

There are many platforms where a user can watch a summary of a soccer match such as youtube and filgoal but in these platforms the summaries are produced manulay and even some online platforms have only a test summary not a video which is also done manulay.

An automated system for summarizing may be available to corporation or clubs but there isn’t a product that is available for the mass public to use and that was an additional motivation for embarking on this project

**2.1. Targeted Customers**

As mentioned before such a system’s output is intended for the dedicated sports fans especially soccer fans who are not able to able to keep up with all the matches they want to whats so they resort to watching summaries of the matches they misses on such platforms such as youtube

**2.2. Market Survey**

In this section, list the competitive products to your work. Similar commercial tools/platforms should be mentioned and discussed. Write a subsection for everyone of them and explain its pros and cons in that subsection

**2.2.1. Competitive Project 1**

Explain and discuss each competitive project

**2.2.2. Competitive Project 2**

Explain and discuss each competitive project

**2.3**. **Business** **Case** **and** **Financial** **Analysis**

In this section you describe the success of establishing a company to sell your product (or service)

Two Aspects must be addressed

Business Case: Based on Market survey above you should anticipate how many products you will sell over the next 5 years and how will you set your price to counter the competition.

Financial Analysis: Based on the business case we must anticipate

1. The Capex (Capital Expenditure): These are one-time spending that you pay for development and buying things for the company
2. The Opex (Operational Expenses): These are recurring payments for salaries and marketing and … etc.

Then you create what we call a cash flow table (on an excel sheet). In this sheet you put down your monthly capex and opex on a set of rows and your reveneus (money you get back from selling product/services) on another set of rows.

The difference between both sums is your profit before tax.

It is likely that this difference is negative at beginning until your sales increase and counter the expenses.

From this cash flow analysis you find the date of the break even point wbich is the date at which all the money you get back equals the money you spent. From that date onward you will be making true profit ☺.

**Chapter 3: Literature Survey**

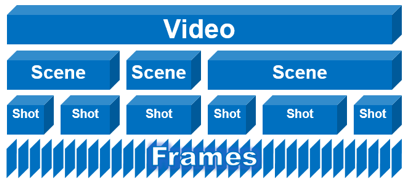
In this chapter we review the background on how a generic video is structured to be able to properly understand the processing that happens in the following chapters and also discuss

**3.1. Background on soccer analysis**

There are many works for detection of camera breaks in the past few years. Some typcal methods for the detection of camera breaks could be found in (8)(9)(10)(11). Recent published papers for shot change detection could be found In(12)(13)(14)(15)(16)(17).

Most work has been focusing on pixel difference, intensity statistics comparison, histogram distance, edge difference, and motion information. Among these methods, histogram basal ones have been consistently reliable, while DCT coefficient based ones give the lowest precision. Motion information based methods are somewhere in between. Some work for performance evaluation of shot detection could be found in (19)(20).

Some work has been done on detecting these special effects. Related works can be found in (21)(22)(23)(24). In a recent review paper, Lienhart (17) compares four major shot boundary detection algorithms, which include fade and dissolve detection. Extensive experimental results also favor the color histogram based method (25) for shot boundary detection, instead of the computationally expensive edge change ratio method (26).



**3.2. Background on video structure**

To probely understand the phases of the sytem, first we must understand a Video structure. Video data are typically organized in a typical hierarchical structure as shown in Figure, some elementary units such as scenes, shots, frames, key frame and objects are generated. A successful structure parsing is an important step in video classification and summarization. In the past, many works have been done in video structure parsing, especially in shot detection, motion analysis and video segmentation.

As discussed above, video data are structured into many shot units. Shot changes should be detected before dividing video data into shot units. shot change can be viewed as detection of a camera break. Normally, there are three major editing types of camera breaks: cut, wipe and dissolve. A cut is an immediate change from a shot to another shot; a wipe is a change where first frame of a shot is replaced with last frame of another shot gradually; a dissolve is a change where one shot gradually appears (fade-in) and another shot slowly disappears (fade-out). A cut can be detected by comparing two adjacent frames. While wipe and dissolve are difficult to detect since they are change gradually. The transition between shots usually corresponds to a change of subject, scene, camera angle, or view. Therefore, it is very natural to use shots as the unit for video analysis.

**3.3. Comparative Study of Previous Work**

Y.H. Gong et al. in (40) proposed a system that can automatically parse soccer video programs using domain knowledge. The parsing process was mainly built upon line mark recognition and motion detection. They categorized the position of the play into several predefined classes by recognizing the compound line pattern with signature method. The motion vectors field is used to infer the play positions for those scenes without line marks. Despite the strong semantic indexes from the categorization of play positions, they have to address these two problems:

• How to identify different camera angle and shooting scale

• How to determine reasonable segment for processing.

D. Yow et al. in (41) presents techniques to automatically detect and extract the soccer highlights by analyzing the image contents, and to present these shots of action by the panoramic reconstruction of selected events. The analysis include the recognition of prominent features of the game, tracking of ball, camera movement compensation for effective recognition, and construction of the panoramic views. The authors pointed out a direction for application of soccer video summarization.

Utsumi et al. in (43) proposed a novel object detecting and tracking method in order to detect and track objects necessary to describe contents of a soccer game.

A. Ekin et al. in (2) presented an automatic and computationally efficient framework for analysis and summarization of soccer videos using cinematic and object based features. In this paper, algorithms of dominant color region detection are robust

A. Ekin et al. in (2) presented a fully automatic and computationally efficient framework for analysis and summarization of soccer videos using cinematic and object based features.

1..Y. Duan et al. in (45) presented a unified framework for semantic shot classification in sports video.

**3.4. Implemented Approach**

The Proposed system is presented to avoid most of the problems discussed in the previous sections, these solutions are involved in each of the stages of the proposed system . In the preprocessing stage, reading all frames of a video is clearly an overhead on the sytem as most soccer matches are two hours long and also the differences between two consequtive frames doesn’t carry much information because very little has changed between the frames so frame skipping is done to ensure that there is no redundancy in the consequitve fremes and also make the system much faster.

shot boundary detection phase is proposed to segments the whole input video stream into small video shots by applying a combination of image processing techniques and histogram comparison techniques. In the shot classification phase , the system applies different algorithms, namely; Grass dominant ratio extraction, Face Detection ,Deep learning and image processing techniques to classify the shot into one of the following classes :wide – medium- close – close out – logo. The logo classification is used for replay detection which is an important factor in determining which of the shots are important and which are not.

In The event detection phase each is shot in sconsedered seperatly and compared with the previous shot to determine if a goal has happened between the two shots by comparing the scoreboard in the upper left corner. Two methods are proposed to detect a goal event, Optical character recognition (OCR) and structural similarity image index (SSIM).

Lastly, the summarization phase which takes into account the output of the previous phases to decide if he shot is important and shoul be included in the final video or should be discarded. For example, if a shot contains a goal then it is important or if a shot has a goal mouth and the audio level is high then it is important or it’s a replay shot and a replay of course is for something important. Then combining the important shots only in a video and that is the output of the system

**Chapter 4: System Design and Architecture**

**4.1. Overview and Assumptions**

The proposed system is composed of five phases. Namely; in the pre-processing phase, the system reads the video stream into patches of frames. Then, in the shot Boundary phase is applied on every patch of frames to segment the video into seprate shots i.e. finding the transition from a camera into another. Then each shot is processed seperatly in the following three phases, the shot classification phase in concerned with classifiying each shot for example a wide shot is when the whole pitch is visible, A close shot is when the players faces are visible etc. then the event detection phase in which each shot is analyside to determine if a goal happens in it or not or if it the audio level in this shot is high with respect to the whole video or if the goal mouth appears in the shot . And lastly, the summarization phase in which the output of all previous phases is analysized to determine which shot is important and should include in the final video and which shots to discard

In this section, introduce how you design you system and develop its underlying architecture. Any employed assumptions should be clearly enumerated and justified.

**4.2. System Architecture**

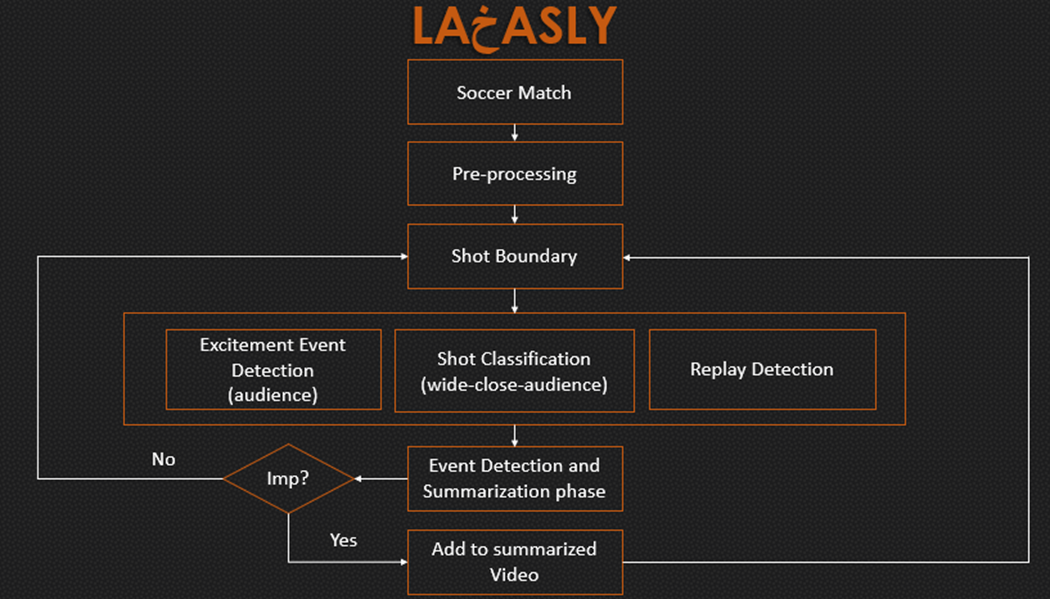
The Proposed system is presented to avoid most of the problems discussed in the previous sections, these solutions are involved in each of the stages of the proposed system . In the preprocessing stage, reading all frames of a video is clearly an overhead on the sytem as most soccer matches are two hours long and also the differences between two consequtive frames doesn’t carry much information because very little has changed between the frames so frame skipping is done to ensure that there is no redundancy in the consequitve fremes and also make the system much faster.

shot boundary detection phase is proposed to segments the whole input video stream into small video shots by applying a combination of image processing techniques and histogram comparison techniques. In the shot classification phase , the system applies different algorithms, namely; Grass dominant ratio extraction, Face Detection ,Deep learning and image processing techniques to classify the shot into one of the following classes :wide – medium- close – close out – logo. The logo classification is used for replay detection which is an important factor in determining which of the shots are important and which are not.

In The event detection phase each is shot in sconsedered seperatly and compared with the previous shot to determine if a goal has happened between the two shots by comparing the scoreboard in the upper left corner. Two methods are proposed to detect a goal event, Optical character recognition (OCR) and structural similarity image index (SSIM).

Lastly, the summarization phase which takes into account the output of the previous phases to decide if he shot is important and shoul be included in the final video or should be discarded. For example, if a shot contains a goal then it is important or if a shot has a goal mouth and the audio level is high then it is important or it’s a replay shot and a replay of course is for something important. Then combining the important shots only in a video and that is the output of the system

**4.2.1. Block Diagram**



**4.3. pre-processing phase**

As the first phase of the proposed system it’s functions are to read the video to be summarized form storage, skip unnecessary and redundant frames to avoid any unnecessary calculations and store the frames in a proper data structure for subsequent phases to process.

. In the phase, reading all frames of a video is clearly an overhead on the sytem as most soccer matches are two hours long and also the differences between two consequtive frames doesn’t carry much information because very little has changed between the frames so frame skipping is done to ensure that there is no redundancy in the consequitve fremes and also make the system much faster.

**4.3.1. Functional Description**

**4.3.2. Modular Decomposition**

Explain the modular decomposition of the coarse module into smaller fine ones

**4.3.3. Design Constraints**

Explain the constraints that affect the design of the module

**4.3.4. Other Description of Module 1**

Give any other necessary discussion of the module to ensure that it is clearly described.

**4.4. Shot Boundary**

We will start to define the term of shot which will be used in this chapter, and then

we are going to identify the basic concepts and techniques applied in shot- boundary

detection.

What is meant by shot?

It is separated view which comes from multiple cameras views positioned at different

locations along the pitch; see Figure (4.1) for more details. You can realize that when the

changing from camera to another (camera 1 to camera 2) done, the resulted view indicates

that it is a new shot and it must be marked as the boundary of the new shot.

Llllllllllllllllllll

Before discussing the types of transitions occur in soccer we must first mention that

the false recognition of new shot resulted from the object and camera motion. We have two

types of transition depending on the camera movement and transition; 1) instant (cut)

transition and 2) gradual transition. When there is a special editing effects during the match

to illustrate the transition we said; gradual transition while the instant transition haven’t any

effects and it is more accurate than the gradual one [1]

lllllllllllllllllllll

there is two types of transition depending on the camera movement and transition; illustrated in Figures 1,2

1) instant (cut) transition

2) gradual transition. When there is a special editing effects during the match  
to illustrate the transition, we said; gradual transition while the instant transition hasn’t any effects and it is more accurate than the gradual one



Figure 5 example of an instant cut



Figure 6 example of a gradual cut

Lllllllllllllllllllllll

After discussing the previous basic concepts that are very important to deal with in  
this section, we present the hierarchal steps of the shot-boundary detection  
algorithms simply by following the flowchart in Figure (4.4).  
We use two main features for shot-boundary detection:  
• The intersection of color histogram of two consecutive frames.

**Intersection of two histograms h1, h2**

d(H_1,H_2) =  \sum _I  \min (H_1(I), H_2(I))

• The correlation of color histogram of two consecutive frames.

**Correlation of tow histograms h1, h2**

d(H_1,H_2) =  \frac{\sum_I (H_1(I) - \bar{H_1}) (H_2(I) - \bar{H_2})}{\sqrt{\sum_I(H_1(I) - \bar{H_1})^2 \sum_I(H_2(I) - \bar{H_2})^2}}

where

\bar{H_k} =  \frac{1}{N} \sum _J H_k(J)

Since color histograms are robust to moderate object and camera motions, we  
represent a frame by its color histogram, which is defined in the RGB space.  
The similarity between the two histograms is measured by histogram intersection and histogram correlation.

The proposed shot-boundary detection algorithm is able to detect both instant and  
gradual transitions with a high detection and low false alarm rates.  
The detection of gradual transitions in sports video is particularly difficult because of the  
high color correlation between two shots so Instead of computing the difference vector for every consecutive frame pair, the comparisons are performed between frame i and frame i + placement K: Unless there is a significant difference, the comparisons are only defined for frames that are placement K apart. This is done to transform the gradual cut to an instant cut.

We have observed that the placement K has an upper bound of k = 5.

|  |  |
| --- | --- |
| Histogram intersection | Histogram correlation |
|  |  |
| This method simply compares, for each bin, the two values in each histogram, and keeps the minimum one. The similarity measure is then simply the sum of these minimum values. Consequently, two images having histograms with no colors in common would get an intersection value of 0, while two identical histograms would get a value equal to the total number of pixels. | is based on the normalized cross-correlation operator used in signal processing to measure the similarity between two signals,  cross-correlation is a [measure of similarity](https://en.wikipedia.org/wiki/Similarity_measure) of two series as a function of the displacement of one relative to the other. This is also known as a sliding [dot product](https://en.wikipedia.org/wiki/Dot_product) or sliding inner-product. |

For both Correlation and Intersection methods, the higher the metric, the more accurate the match.

There are other metrics that we have tried but discarded, they will be discussed in the final project document

The two frames below are obviously consecutive frames from the same shot i.e. there is no cut.

that should be reflected in the values of histogram intersection and correlation



*Histogram Intersection = 9.04120311407678*

*Histogram correlation = 9.971235345958373*

The values are higher than the thresholds mentioned in the above algorithm (the thresholds are based on trial and error) so no cut

Here is another example to fully explain how the shot boundary algorithm works.

These are also consecutive frames but there is a transition (cut) between them



*Histogram intersection = 0.24404063194742776*

*Histogram correlation = 0.002219072222348816*

Both values are smaller than the thresholds so we perform step 4 of the algorithm in which dividing the two frames into blocks, each block 150px x 150px and calculate intersection and correlation between individual blocks then, according to the values of intersection and correlation of the blocks to consider whether that block is considered full changed or half changed or not changed at all.

Then, step 9 in which we count how many blocks have changed and calculate the percentage with respect to all block

In the above two frames the percentage of changed blocks is 100% i.e. all frame blocks have has changed so, there is a cut.

Another example to full demonstrate the importance of having two methods.

The following two frames are consecutive but in the midst of a logo transition (gradual transition)



*Histogram intersection = 2.1293090697099615*

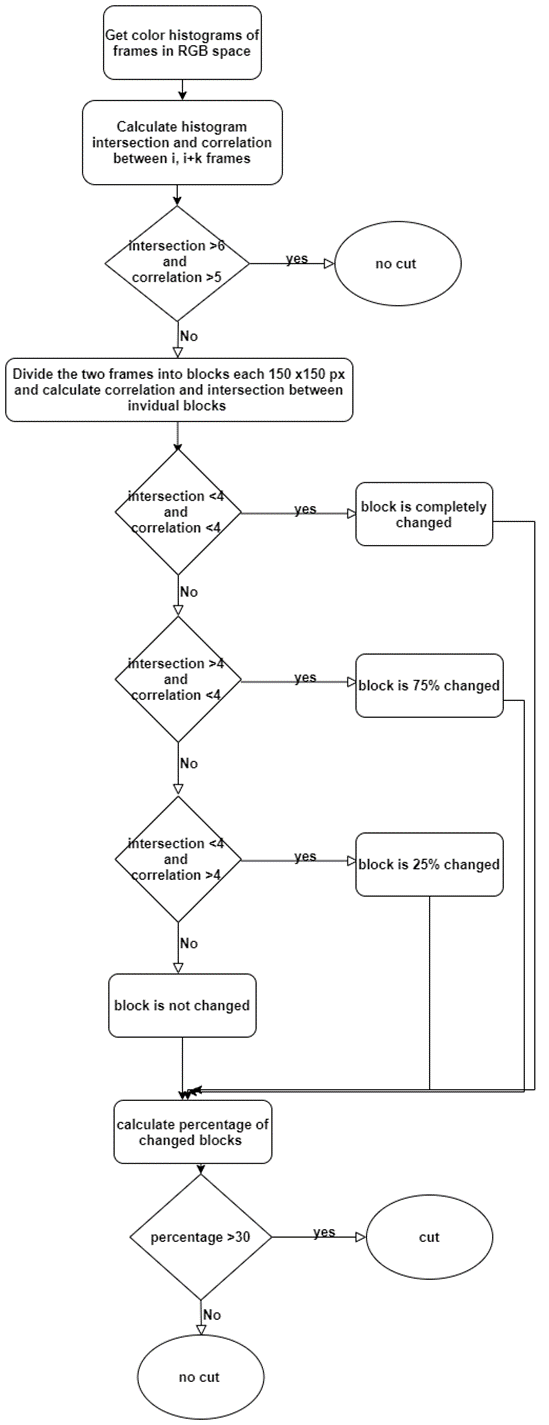
*Histogram correlation = 9.844120057485238*

Here the two methods disagree, the intersection indicates that there are minor similarities in the two frames but the correlation indicates that that’s they are variations of each other i.e. the histogram is a shifted version of the other histogram for example.

So, we resort to the blocks process once again to determine if there is a cut or not. The percentage of changed block is 37% which makes sense because there are areas that didn’t change but according to the threshold this is still considered a cut and that is logical because it’s a logo transition.

**Algorithm:**

* Get color histograms of the two frames in RGB space
* Calculate histogram intersection and correlation between the two histograms
* If intersection >6 and correlation >5 then no cut
* Divide the two frames into blocks, each block 150px x 150px and calculate intersection and correlation between individual blocks.
* If intersection <4 and correlation <4 then block is 100% changed
* If intersection >4 and correlation <4 then block is 75% changed
* If intersection <4 and correlation >4 then block is 25% changed
* If none of the above conditions are met then the block is not changed
* Count changed block and calculate percentage of changed blocks. 10- If percentage of changed blocks > 30% then Cut else no cut



We tested the proposed algorithms on 6 short videos of different durations extracted from different football matches

|  |  |  |  |
| --- | --- | --- | --- |
| Test | Duration in minutes | Number of cuts | efficiency |
| Test1.mp4 | 5 | 15 | 100% |
| Test2.mp4 | 2:22 | 21 | 80.9% |
| Test3.mp4 | 1:36 | 19 | 89.4% |
| Test4.mp4 | 3:12 | 38 | 84.2% |
| Test5.mp4 | 3:18 | 38 | 92% |
| Test6.mp4 | 4:41 | 37 | 86% |

**4.4.1. Functional Description**

Explain the functional description of the module

**4.4.2. Modular Decomposition**

Explain the modular decomposition of the coarse module into smaller fine ones

**4.4.3. Design Constraints**

Explain the constraints that affect the design of the module

**4.4.4. Other Description of Module 2**

Give any other necessary discussion of the module to ensure that it is clearly described.

**4.5. Exitment Event detection**

# Excitement Event Detection (Audio Processing)

## Introduction

Loudness, silence and pitch generated by the commentator and/or crowd are effective measurements for detecting excitement. The volume level is the most frequently used and simplest audio features as an indication of the loudness of the sound, so in this module we get the video times in seconds where the volume is high which is an indication of an important event in the match.

## Methodology

Input: video clip

Output: times in video having volume level > 90% of the video volumes

Algorithm:

1. Read video clip:
2. Extract audio from the video clip:
3. Get average volume of each 10 seconds
4. Get the difference between every two averages then detect the increases and decreases in volume:
5. Determine peaks indices of volumes:
6. Get peaks volumes:
7. Get Times of peaks having volume level > 90%

90% of the values in the averaged volumes array.

for example: averaged volumes = [1,2,3,4,5,6,7,8,9,10,11,…,97,98,99,100]

so, having the averaged array, we take the largest 10% of it

so, values > 90% are [91, 92,…,99,100] i.e. the larger 10% of the array.

**Flow Chart**

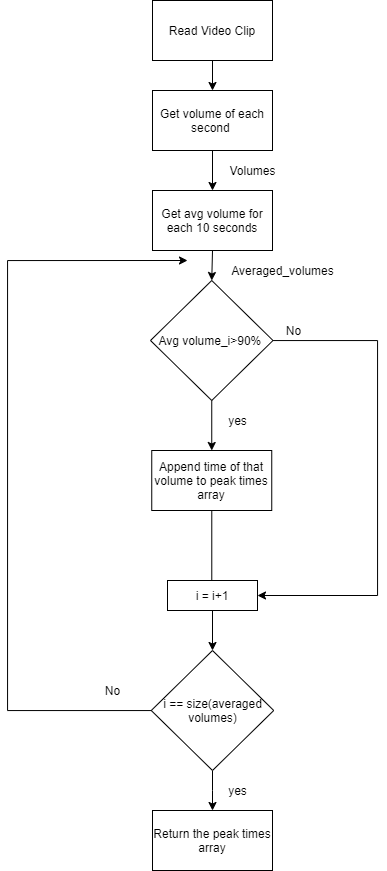


Figure Flow chart for the audio processing

# Excitement Event Detection (Goal Mouth Detection)

## Introduction

For soccer video, the goal-mouth scenarios can be selected as the highlighted candidates, for the reason that most of the exciting events occurs in the goal mouth area such as the goal, shooting, penalty, direct free kick, etc.

On the other hand, the non-goal-mouth scenarios often consist of the dull passes in the midfield, defense and offense or some other shots to the audiences or coaches, etc., which are not considered as exciting as the former.



Figure Examples of important events containing goal post

## Goal mouth detection algorithm

***For each frame***

***Convert image form RGB then to gray***

***Perform edge detection using canny***

***Get lines from detected edges using Hough transform***

***Skip lines whose magnitude is less than 200***

***Check if a line is parallel to 2 other lines then there is a potential goal mouth in the image and return true***

**Accuracy** is about 70% - 80%.

## Flow Chart

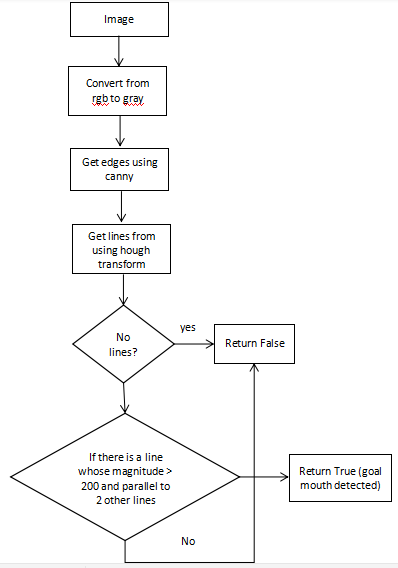


Figure Flowchart of goal mouth detection

# Excitement Event Detection (Goal detection)

## Introduction

The score board is a caption region (usually at the top left) distinguished from the surrounding region, which provides the information about the score of the game and the match time. The score board dimensions and position is always constant in a certain league so detection of the score board itself is not needed



Figure Examples of Scoreboards

## Goal detection using OCR

The goals are detected by the help of scoreboard with optical character recognition



Figure Example Before and after a goal

**La5asly** uses **Tesseract** engine for **OCR**, it is free software, developed by **Google.**

### Algorithm

***1: while reading the input video.  
2: for each 5 seconds do.  
 2.1: apply the mask to get the scoreboard.  
 2.2: run the tesseract engine to check if the results changed or not.  
 2.3: if changed and stable:  
 2.3.1: save the results (error free).  
 2.4: if changed:  
 2.4.1: save the results locally and go to 2.***

We need tesseract OCR to get the number in different shots to see of the number changed

## Goal Detection with structural similarity image index (SSIM)

### Abstract

In this approach, Goals are detected with the structural similarity image index (**SSIM**), It is an enhanced way to detect if the image changed or not. It uses the difference between the scoreboard and get the mean, variance and illuminance between the two scoreboards and get if the scoreboard changes or not. If scoreboard changes then it is an event (goal or substitution).

### Flow Chart

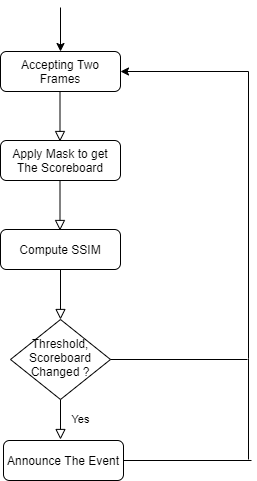


Figure Flowchart for ssim

tesseract has an advantage over the ssim method, it returns only the goals.

we see the results as strings, so comparing is easy.

But for SSIM method, the results are percentages of change so it’s threshold dependent

And this is not good because in substitutions the scoreboard changed also to declare the substitution

example: [https://ibb.co/LNXP8v6](https://ibb.co/LNXP8v6?fbclid=IwAR2E-R86nooKpmq59BPLWlNTbrKGpFXpt92ogU0piWovq_LpgOfMz27zqkI)

**4.5.1. Functional Description**

Explain the functional description of the module

**4.5.2. Modular Decomposition**

Explain the modular decomposition of the coarse module into smaller fine ones

**4.5.3. Design Constraints**

Explain the constraints that affect the design of the module

**4.5.4. Other Description of Module 2**

Give any other necessary discussion of the module to ensure that it is clearly described.

**4.6. shot classification**

Production crews use different shot types in broadcasting a soccer match, which is  
be used for high-level video analysis in a particular domain.

Cinematographers classify a shot into one of four categories Wide, medium, close-up and audience (out-of-field) shot classes, the definitions of which are usually domain-dependent.

In the following, we define these four classes for sports videos:

1. **Wide Shot:** A long shot displays the global view of the field; wide shots almost always display some part of the stadium, which decreases the dominant colored pixel ratio.
2. **Medium (In-field) shot:** A medium shot, where a whole human body is usually visible, is a zoomed-in view of a specific part of the field.
3. **Close-up:** A close-up shot usually shows the above-waist view of a player or referee.
4. **Audience (Out-of-field) Shot:** The audience, coach, and other shots are denoted as out of field shots.

The sequence occurrence of a close-up shots and audience (out-of-field) indicates an important event such as (goal, goal attempts …etc.) during the match.

As we illustrated before in our meetings, after the classification we observe the sequence of shots to determine if an important event occurred or not.

For example, when a goal is scored in the wide shot the director of the match usually cuts to the player who scored the goal celebrating (i.e. close) then a shot of the audience (i.e. out of field) then a replay of the goal.

**

Figure Sequence of shot to declare a goal

## Shot Classification Approaches

The above definitions consider shot-types as functions of field region. Because field  
region information is available after **Grass dominant ratio extraction** (discussed before in progress report-1) but using only the grass dominant ratio didn’t yield great results in shot classification.

In the following section we discuss four approaches to classification and comparing between them:

1. Shot classification using Grass dominant ratio extraction
2. Shot Classification using Face Detection
3. Shot classification using Deep learning
4. Shot classification using image processing techniques

### Shot classification using Grass dominant ratio extraction

The proposed algorithm is based on a specific threshold (range) for grass ratio (G); which was developed by observing different matches in many lighting and weather conditions to appropriately define a range that will cover the four shot types.

Classification of a shot into one of the discussed three classes is based on spatial features.  
Therefore, shot class can be determined from a single key frame or from a set of frames  
selected according to a certain criterion.

G is the grass ratio in the frame as mentioned in the first line of the above paragraph

For example: if the frame is in a wide shot that means that the whole filed is visible which means that G would be high (> ~70%)

Thinking intuitively, G would be almost zero for out-of-field frames, a low G value in a frame corresponds to a close-up, while high G value indicates that the frame is a long view, and in between, a medium view is selected.

Frame Type

***For each shot***

***Choose a set of key frames in the shot***

***Compute grass ratio G of these frames***

***Classify frames***

***Determine the majority type of the frames and assign it to the shot***

The set of the key frames is based on the length of the shot, if the shot is small (less than 50 frame) then the whole shot is included and if the shot is larger then sample 50 frames uniformly from the shot from start to end.

Choosing the set of frames also depends on the computation power running the code.

Due to the computational simplicity of the proposed algorithm, computing the grass ratio of all frames in the shot would not be a big overhead.

Although the accuracy of the above simple algorithm is sufficient but it does not meet the desired outcome.

The accuracy of the above algorithm is roughly 60% (i.e. it classifies 60% of the shots correctly)

### Shot classification using Face detection

In the observation phase mentioned above it can be said that the Wide shot does not contain any clear faces (large enough to be recognized by a human or a computer), in the medium shot a face can or cannot be recognized depending on the angle of the camera filming the shot, in the close shot a face can be clearly recognized as it covers a large area of the frame.

With that said, combining face detection model and the Ratio G to obtain better results in classification.

Frame type

***For each shot***

***Choose a set of key frames in the shot***

***Compute grass ratio G of these frames***

***Determine whether the frames contain face or not and if it does get the***

***Area of the bounding box around the face***

***Classify frames***

***Determine the majority type of the frames and assign it to the shot***

The obtained results are better than using grass ratio only but the computational time and complexity is much worse because the face detection model is very complex and time consuming.

This approach yielded good results (~70%) but we didn’t spend much time refining it because the face detection model is pretrained and we wanted our classification method to be implanted from scratch

### Shot Classification using deep learning

In this approach, Frames are Extracted from Matches and filtered (assigned labels) manually to construct a data set, Then A model **(CNN)** is used to train on the constructed data.

By ‘filtered’ we mean assigning different labels manually.

For example, we extract the frames from the match then separate the wide frames form the close from the medium.

It’s an exhaustive process to get labeled data, the model is trained on these labeled frames to be able to predict the type of a given input frame.

The model trained with 8,000 images (Total 40,000) per each class, 20 epochs. The input to model is the raw image 28x28x3.

Firstly, the model suffered from overfitting. We overcame this problem by adding dropout in layers with probability 0.6 and the results got better.

This approach is much better than the previous techniques and has a great classification accuracy 86%.

### Layers Explanation

**Convolution** is the first layer to extract features from an input image. Convolution preserves the relationship between pixels by learning image features using small squares of input data. It is a mathematical operation that takes two inputs such as image matrix and a filter or kernel. Convolution of an image with different filters can perform operations such as edge detection, blur and sharpen by applying filters.

**Pooling** layers section would reduce the number of parameters when the images are too large. Spatial pooling also called subsampling or down sampling which reduces the dimensionality of each map but retains important information.



Figure CNN Model

### Shot Classification using image processing

In this approach, an image, after a dominant color mask is applied i.e. green is white and any other color is black, is divided into 3:5:3 grid.

The frame is divided into a 3:5:3 grid as illustrated in the figure below

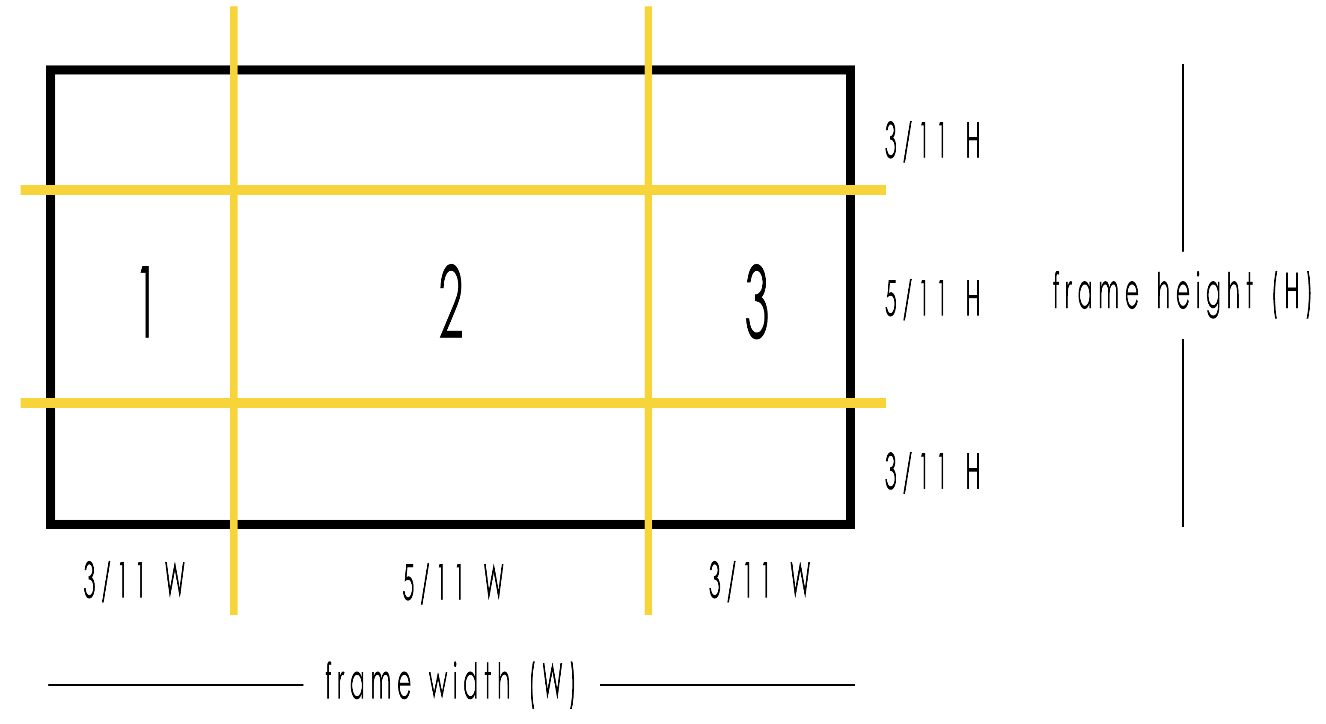


Figure frame after splitting into 3:5:3

The image classes are determined using the green pixels in 1, 2, 3 regions and the absolute difference between 1, 2 and 2, 3 to estimate is there a close shot or not.

The close out is pretty simple, if green ratio is tiny (less than 10%) then it is close-out.

But this approach has a good accuracy, actually near 60%, but compared to the above its subpar.

The accuracy is great for an image processing technique and no learning.

The advantages of this technique are that its computationally inexpensive (no models) and its implemented from scratch

**Comparison between shot classification approaches**

|  |  |  |  |
| --- | --- | --- | --- |
| Grass dominant color | Face detection | deep learning | image processing |
| **60%** | **70%** | **86%** | **60%** |

**4.6.1. Functional Description**

Explain the functional description of the module

**4.6.2. Modular Decomposition**

Explain the modular decomposition of the coarse module into smaller fine ones

**4.6.3. Design Constraints**

Explain the constraints that affect the design of the module

**4.6.4. Other Description of Module 2**

Give any other necessary discussion of the module to ensure that it is clearly described.

**Chapter 5: Audio version**

In this version the video summarization is based only on the audio feature i.e. when the audio level is high that gives an indication of an important event in the football match

**Reason to make this version**: the initial results of the main version showed that theaudio feature is contributing significantly in the summarized final video so making a version with audio is worth the effort and would be much faster than the original version, with that said it’s not without its drawbacks

**Advantages**:

* Very less computations compared to the main version.
* Depends only on the Audio module and the shot boundary module but the main version has more modules to take into considerations.
* Much faster runtime

**Disadvantages**:

* The summarized output of this version is relatively longer in minutes than the main version as in many matches the audiences sometimes have loud voice while cheering even if it’s not really an important event in the game.
* The summarized video can get longer depending on audience and commentator attitude in every match.
* In rare situation, some important events are not included because the audio level is not larger than the audio threshold

**Used Modules Details**

**Used modules:** Audio, Shot Boundary

1. **Audio module Input:** video clip

**Output:** times in video having volume level > 90% of the video volumesAlgorithm:

* Read video clip:
* Extract audio from the video clip:
* Get average volume of each 10 seconds
* Get the difference between every two averages then detect the increases and decreases in volume:
* Determine peaks indices of volumes:
* Get peaks volumes:
* Get Times of peaks having volume level > 90% [largest 10%]

1. **Shot Boundary module Input:** two frames **Output:** cur or no cut

**Algorithm:**

* Get color histograms of the two frames in RGB space
* Calculate histogram intersection and correlation between the two histograms
* If intersection >6 and correlation >5 then no cut
* Divide the two frames into blocks, each block 150px x 150px and calculate intersection and correlation between individual blocks.
* If intersection <4 and correlation <4 then block is 100% changed
* If intersection >4 and correlation <4 then block is 75% changed
* If intersection <4 and correlation >4 then block is 25% changed
* If none of the above conditions are met then the block is not changed
* Count changed block and calculate percentage of changed blocks. 10- If percentage of changed blocks > 30% then Cut else no cut

**Input:** Video Clip

**Output:** Summarized video clip

**Algorithm:**

1- Read video clip

2- Get Peaks times using Audio module, peaks [i…n]

3- For each peak time peaks[i], get its frame number

4- For each frame get its shot using the shot boundary module.

1- Starting from this frame explore frames descending tell finding a shot cut

= shot start

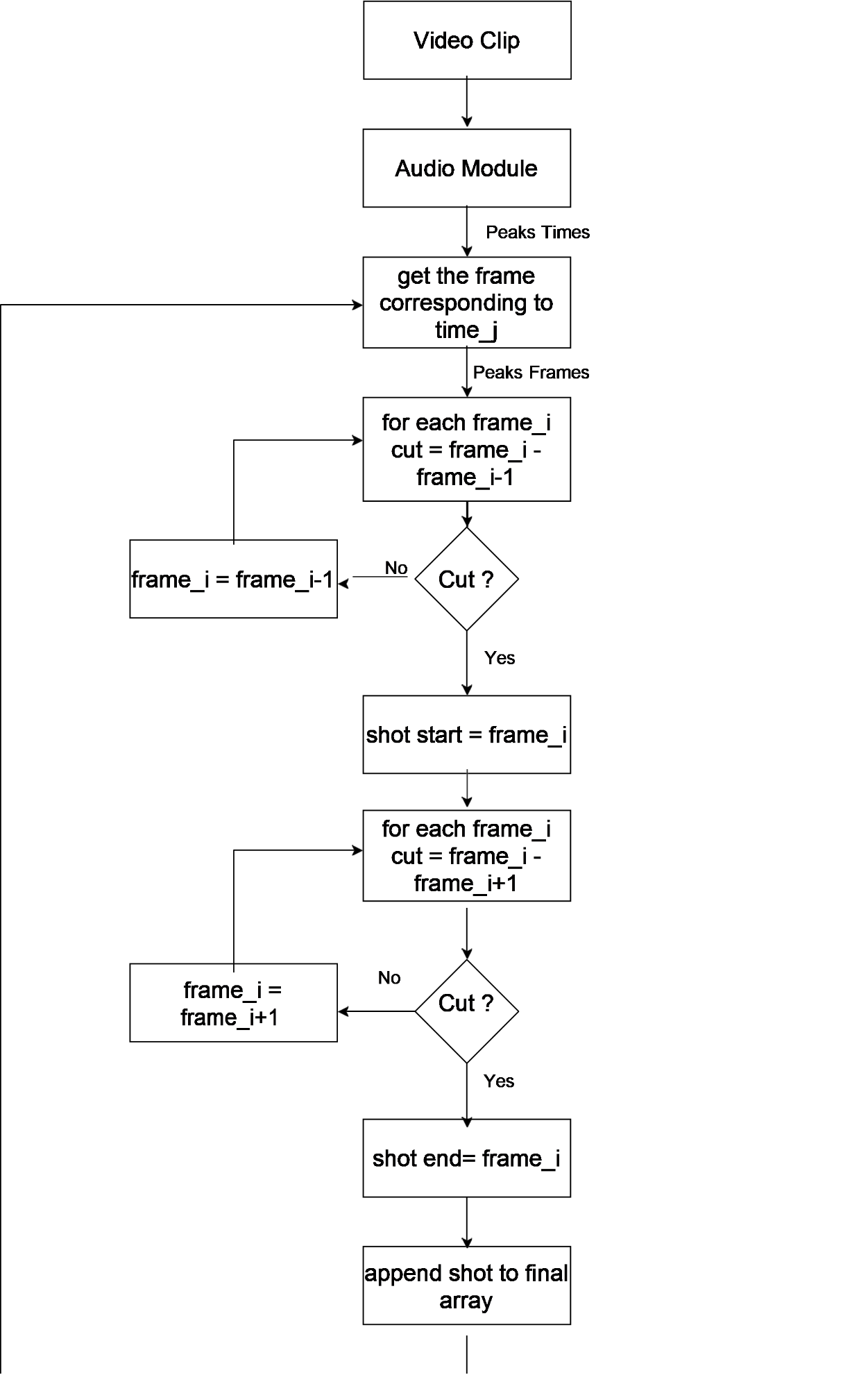
2- Starting from this frame explore frames ascendingly tell finding a shot cut

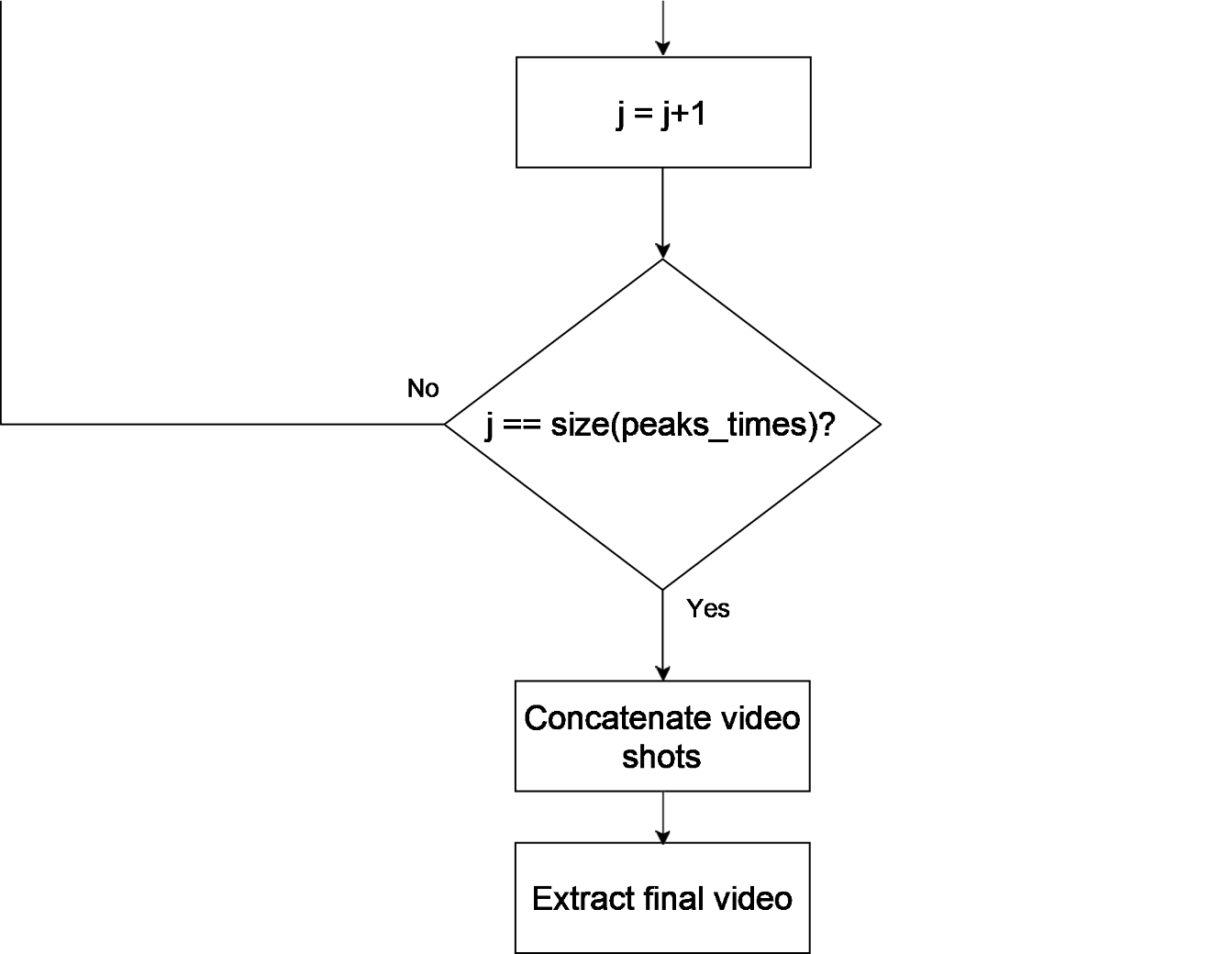
= shot end

3- Having shot start and shot end, append this shot to the final array

5- Concatenate shots into one array

6- Extract the summarized video

****

****

**Results**

To evaluate each audio threshold level in different leagues we calculate a score based on the goals detected, the interesting events detected and also the length of the output summarized video, goals is the most interesting event so we multiply the ratio of the detected goals and total number of goals in the match by 0.6, for other interesting events we multiply it’s ratio by 0.3 and we take the difference of 1 and ratio of length of output video and total length of video and multiply it by 0.1 to give a higher score for smaller length output video, then we add results to obtain the score.

**Equation:**

**Score** = 0.6\***E1** + 0.3\***E2** + 0.1\*(1 - **E3**)

E1: no. Goals detected / total goals

E2: no. Of other interesting events detected / total no. Of events

E3: Length of output video / total length of video (=90)

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| League | Match | Audio Level Threshold | Output video Length in minutes | Goals Detected accuracy | Other events detection accuracy | Score |
| BUNDESLIGA | Fc Augsburg  VS  Dortmund | 85 | 28:06 | 4/8 | 7/13 | 0.53 |
| 90 | 10:03 | 3/8 | 3/13 | 0.38 |
| 95 | 8:51 | 2/8 | 0/13 | 0.24 |
| Bayern Munich  VS   1. Bremen | 85 | 16:06 | 6/7 | 16/16 | 0.9 |
| 90 | 12:07 | 5/7 | 14/16 | 0.78 |
| 95 | 8:28 | 4/7 | 11/16 | 0.64 |
| Bayer Leverkusen  VS   1. Dusseldorf | 85 | 20:35 | 3/3 | 16/16 | 0.98 |
| 90 | 15:50 | 3/3 | 10/16 | 0.87 |
| 95 | 6:47 | 2/3 | 6/16 | 0.6 |
| LIGUE 1 | PSG  VS  Monaco | 85 | 31:14 | 6/6 | 9/12 | 0.89 |
| 90 | 23:46 | 6/6 | 10/12 | 0.92 |
| 95 | 14:48 | 5/6 | 7/12 | 0.76 |
| Rennes  VS  Nantes | 85 | 17:28 | 5/5 | 11/14 | 0.92 |
| 90 | 12:38 | 4/5 | 8/14 | 0.74 |
| 95 | 5:01 | 4/5 | 4/14 | 0.66 |
| PSG  VS  Marseille | 85 | 26:27 | 4/4 | 8/10 | 0.91 |
| 90 | 19:44 | 4/4 | 7/10 | 0.89 |
| 95 | 8:35 | 4/4 | 6/10 | 0.87 |
| PREMIER LEAGUE | Man. United  VS  Man. City | 85 | 29:38 | 2/2 | 5/8 | 0.85 |
| 90 | 21:42 | 2/2 | 4/8 | 0.83 |
| 95 | 10:33 | 2/2 | 2/8 | 0.76 |
| Aston Villa  VS  Man. City | 85 | 25:16 | 7/7 | 8/8 | 0.97 |
| 90 | 17:01 | 7/7 | 7/8 | 0.94 |
| 95 | 13:07 | 7/7 | 4/8 | 0.83 |
| Chelsea  VS  Arsenal | 85 | 24:29 | 4/4 | 7/10 | 0.88 |
| 90 | 18:04 | 4/4 | 6/10 | 0.86 |
| 95 | 6:51 | 3/4 | 3/10 | 0.63 |

In the above chart, for each league an average score is calculated for each audio threshold level by adding their scores and divide it by total number of matches (=3).

**Conclusion:**

For most leagues we have tested level 85 threshold has the best results.

In premier league level 90 threshold has an average score very close to level 85 threshold so we can use it instead to reduce the length of the output summarized video.

**Chapter 6: System Testing and Verification**

In this chapter, you have to explain all the steps you carried out to ensure that project outcomes are realized correctly. Your testing setup, strategy and environment should therefore be described. Your efforts for unit testing as well as integrated system testing should be given. Finally, the results from different testing scenarios should be highlighted and discussed.

In this space, before the first section, write an introductory paragraph on how you test and verify the correct operation of your system

**6.1. Testing Setup**

Explain the setup you are using in testing your project

**6.2. Testing Plan and Strategy**

Explain the methodology you follow while testing your project in details

**6.2.1. Module Testing**

Explain the steps you carried out to test different modules within the project. Give and discuss the results obtained from the testing of these modules

**6.2.2. Integration Testing**

Explain the steps you carried out to test the integrated system of your project. Give and discuss the results obtained from this whole project testing

**6.3. Testing Schedule**

Mention your testing schedule

**6.4. Comparative Results to Previous Work**

Give a summary of comparative results to previous work in Tabulated and or Graphical form along with a short commentary.

**Chapter 7: Conclusions and Future Work**

This chapter should summarize the whole project, it features and limitation. Moreover, you should give directions for future work

In this space, before the first section, write an introductory paragraph for the chapter

**6.1. Faced Challenges**

Mention all the problems/challenges that you faced while working with the project and how you overcome them

**6.2. Gained Experience**

Mentioned the experience/skills that you gained from working with the project

**6.3. Conclusions**

Write your conclusions regarding the project. Mention its features and limitations

**6.4. Future Work**

Give possible extensions, enhancements and future work of you project, such that subsequent students could build on your work and develop larger systems/platforms.

**References**

The references should be ordered according to their appearance in the text. Ensure that all references are cited throughout your report text. The following are examples of how to write different types of references “[1] Book, [2] Journal/magazine articles, [3] conference paper, [4] website, [5] thesis”. Replace the fields with those of your used references. Question marks “??” should be replaced by the corresponding number

1. Author1, Author 2,…, “Book title,” name of publishing firm, edition, year
2. Author1, Author2,…., “Title of journal article,” name of the journal, vol. ??, no. ??, pp. ??, year of publication
3. Author1, Author2,…, “Title of conference paper,” in proceedings of conference name, city, country, date, year, pp. ??
4. Author or Corporation name, “Title,” year, link for the website, last accessed: date of last access
5. Author, “Thesis title,” M.Sc./Ph.D. thesis, Department, University, year

**Appendix A: Development Platforms and Tools**

This appendix explains used tools, platforms, and hardware kits. Any ready-made module should be mentioned and discussed in this appendix. The appendix is divided into two main sections; one for the hardware and the other is for software. Within each section, you could add as much subsections as needed, according to the number of tools and platforms that you use in your project.

In this space, before the first section, write an introductory paragraph to the appendix

**A.1. Hardware Platforms**

A description of any used hardware platforms/kit should be written in this section. Each platform/kit is better described in a separate subsection. (A1.1..)

**A.2. Software Tools**

A description of any used software tool/package should be written in this section. Each tool/package is better described in a separate subsection (A2.1,..)

**Appendix B: Use Cases**

Include all your use cases

**Appendix C: User Guide**

Prepare a user guide for your project. Ensure that the guide is clear, detailed and easy for an ordinary customer to use your project. Employ figures and charts as needed to facilitate the use of your guide

**Appendix D: Code Documentation**

Your code or parts of the code you feel necessary could be included here (optional) however for one copy of this report an attached CD with all of the code is a must.

Remember you will deliver three copies of this report.

**Appendix D: Feasibility Study**

Give a detailed feasibility study of your project