**Synopsis**

**A**

**Project Synopsis**

**On**

**Decision Review System**

**Submitted for the award of degree of**

**Bachelor of Technology**

**In**

**Computer Science & Engineering**

By

**Saksham** (20020004088)

**Sachi** (20020004087)

**Under the Supervision of**

**Ms. Vaishali**



**Department of Computer Science &Engineering**

**Satyug Darshan Institute of Engineering & Technology**

**FARIDABAD-121002**

**Apr 2023**

**INTRODUCTION**

With technology taking over every aspect of life, it was only a matter of time before it was utilized to the advantage of sports.

Technology in sport is used with an objective to eliminate human errors that may have earlier affected the course of the game drastically.

The Decision Review System was introduced across various sports, to give the players an opportunity to review such errors from the on field umpire.This project identifies and examines the use of officiating technology and it impact across different sports such as badminton, rugby, tennis and cricket. In addition to this, statistics collected within various sports have proved time and again that umpiring decisions have been more accurate with DRS than without. It is silly to argue that we should not have DRS unless it has 100% accuracy—you anyway cannot get 100% accuracy without using technology.

It is enough that DRS improves the accuracy percentage—as it doubtless has, from 90.3% to 95.8%—for there to a strong case for its use. Furthermore, the DRS also has other things going for it.As the Guardian’s Rob Smythe has pointed out, it is changing the game in some fundamental ways, and definitely for the better. It has made the game even more competitive and fair

Judgement .

**SYSTEM REQUIREMENT**

⁕ The Basic System Requirements are as follows :-

**Hardware Configuration**

1. PROCESSOR : Intel I3 Processor
2. RAM REQUIRED : Minimum 4 GB Ram
3. SYSTEM TYPE : 32- Bit OPERATING SYSTEM

**Software Requirements**

1. OpenCV Library - 32- or a 64-bit computer, Windows 7,10or 11, macOS X 10.11 or higher, or Linux RHEL 6/7, Python2.7, 3.4, 3.5 or 3.6
2. Tkinter (8.6. 11)

iii. Imutils - Open CV Library, Python 2.7 or 3.Visual Studio Code IDE (1.61.1)



**LITERATURE SURVEY**

**Decision Review System in Cricket**

This paper suggests that the most common instances of referrals are for caught behind and Leg Before Wicket (LBW).As the final decision rides on chance of agreement of the field umpire along with, the DRS algorithms since this technology predicts the nature of the trajectory of the delivery and such factors are dealt in very small measurements (in millimeters). Various errors included may change the final result for the same delivery if replayed again. The paper put forth the main idea that is to have a continuous real time feed clicked from the six cameras strategically placed in the stadium each operating at the specific frame rate which is enough to capture the pitched delivery to provide data points for mapping the virtual trajectory of the ball. Though, there is always a question of the accuracy with which the technology works as its decision cannot be referred again and is considered final. Hence, the accuracy of the technology used is limited to a certain range, but along with the human intervention in the decision making, this review system is bound to be the perfect solution to such a problem.

**Decision Review system in Tennis**

All Grand Slam tournaments of tennis provide line review system which uses Hawk-Eye ball tracking system. The players can use this system to evaluate debatable line calls.All Hawk-Eye systems are based on the principles of triangulation using visual images and timing data provided by a number of high-speed video cameras located at different locations and angles around the area of play. The tennis hawk eye system uses up to ten cameras. The system rapidly processes the video feeds from the cameras and ball tracker. The system uses something called data store which is used to store a 3D blue-print model of the playing area/ court and also stored in it are the rules of the game.

**Decision Review System in Football**

Decision Review system in Football comprises of various systems such as smart ball system, Goal Ref System, Hawk eye System, etc. The system which has had a very successful trial and is the leading candidate out of all the other system is the Hawk eye system. Primarily used in the English Football league, Premier League the Hawk Eye system uses three cameras targeted on every goal-line, and each of them take video footage at 600 frames per second. Hawk-Eye has the means to grant a conclusive decision on whether the football has fully crossed the goal line or not and transfer this information to the main referee in less than a second.

**PROPOSED SYSTEM**

Traditionally, in various sports the final judgement is given by the referee or the umpire. While usually most of these decisions are correct, often there are discrepancies behind these decisions since human judgement is involved which has the potential to change the final result of the game. Our System is basically an aid to the umpire in case of such discrepancies where in only a good quality camera and a system to operate the software is needed. What this human aid does is, it operates on frame rate to give a clear picture of the exact moment where the decision needs to take place and the final call is given by the referee themselves.

For the module :

Our project will be implemented using Python and Tkinter, OpenCV library, Python Imaging Library, Imutils, Pillow module etc. Decisions like stump out, run out etc. can be decided by altering the source video’s frame rate for the which, we will be optimizing and using frame subtraction using OpenCV library.

1 **GUI of DRS**:

Tkinter is used to make GUI of the Decision Review System and various computer vision algorithms that give decisions based on different criteria. Python offers multiple options for developing GUI (Graphical User Interface). Out of all the GUI methods, Tkinter is the most commonly used method. Being a Python binding to the Tk GUI toolkit, it gives standard Python interface to the Tk GUI toolkit, and is Python&#39;s de facto standard GUI. Tkinter also includes standard GNU/Linux, Microsoft Windows and mac OS installs of Python.

2 **Frame Subtraction**:

Frame subtraction will be done using OpenCV (OpenSource Computer Vision Library) which is a library of programming functions mainly aimed at real-time computer vision. OpenCV is an open-source library which is very useful for computer vision applications such as video analysis, CCTV footage analysis and image analysis. OpenCV supports a wide variety of programming languages like Python, C++, Java.It can process images and videos to identify objects, faces, or even the handwriting of a human. OpenCV is written by C++ and has more than 2,500 optimized algorithms. When we create applications for computer vision that we don’t want to build from scratch we can use this library to start focusing on real world problems. There are many companies using this library today such as Google, Amazon, Microsoft and Toyota. Many researchers and developers contribute.We can easily install it in any OS like Windows, Ubuntu and Mac OS. With reference to our project, OpenCV library will be majorly used to bring changes to the frame i.e.,

increasing or decreasing the frame speed.

3 **Python Imaging Library**:

Python Imaging Library (expansion of PIL) for which Pillow module is to be

installed, is the de facto image processing package for Python language. It

incorporates lightweight image processing tools that aids in editing, creating and

saving images and with respect to our project, it will load images from an array.

4 **Imutils Module**:

Imutils which is basically a series of convenience functions to make basic image

processing functions such as translation, rotation, resizing, skeletonization, and

displaying images easier with OpenCV will also be used for the implementation.

**REFERENCES**

1. <https://www.google.co.in/>
2. <https://stackoverflow.com/>
3. <https://www.codewithharry.com/>
4. <https://www.wikipedia.org/>
5. <https://docs.python.org/3/library/>