

# Hawk-Eye Technology

Prashant Kumar<sup>1</sup>, Gaurav Saxena<sup>2</sup>

<sup>1,2</sup>Dept of Electronics & Communication Engineering

<sup>1,2</sup>Poornima College of Engineering, Rajasthan Technical University, Jaipur, India

**Abstract-** *Many sports have become very reliant on monitoring systems such as Hawk-Eye in Tennis; this paper has looked into just how accurate the technology is, and assessed the effectiveness of the technological approaches used to implement the system. The level of accuracy is vital for sport monitoring systems, in a number of sports such as Line Calling Decisions in Tennis. Even a small margin of error can affect the decision of whether the ball is called in or out. As many high profile sporting industries have placed a great deal of dependence upon this technology, if it were to prove inaccurate, the sporting world would incur devastating consequences. As governing bodies would then be under a large amount scrutiny, from all over the world. Players, coaches and even spectators would all start to question the decisions made using this technology made in past. And as these were designed to rule out human error in such cases as line calling, any major failings found in the technology would render them useless.*

## I. INTRODUCTION

The game of cricket has attained great commercial importance and popularity over the past few years. As a result, there has been felt a need to make the game more interesting for the spectators and also to try and make it as fair as possible. The component of human error in making judgments of crucial decisions often turns out to be decisive. It is not uncommon to see matches turning from being interesting to being one sided due to a couple of bad umpiring decisions. There is thus a need to bring in technology to try and minimize the chances of human error in such decision making. Teams across the world are becoming more and more professional with the way they play the game. Teams now have official strategists and technical support staff which help players to study their past games and improve. Devising strategies against opponent teams or specific players is also very common in modern day cricket. All this has become possible due to the advent of technology. Technological developments have been harnessed to collect various data very precisely and use it for various purposes.

The Hawk-Eye is one such technology which is considered to be really top notch in cricket. The basic idea is to monitor the trajectory of the cricket ball during the entire duration of play. This data is then processed to produce

lifelike visualizations showing the paths which the ball took. Such data has been used for various purposes, popular uses including the LBW decision making software and colorful wagon wheels showing various statistics. This paper attempts to explain the intricate details of the technology which goes behind the Hawk-Eye. We first start off with a general overview of the system and an outline of the challenges that we might face, then move on to the details of the technology and end with various applications where one sees this technology being put to use.

## II. HAWK-EYE: A GENERAL OVERVIEW

Cricket is a ball game played within a predetermined area. A system comprising of video cameras mounted at specific angles can be used to take pictures. These pictures are then used to locate the position of the ball. The images are then put together and superimposed on a predetermined model to form a complete visualization of the trajectory of the ball. The model includes, in this case, the pitch, the field, the batsmen and fielders etc. For this to be possible, we need to sample images at a very high rate and thus need efficient algorithms which can process data in real time. Such technologies are widely used today in various sports such as Tennis, Billiards which also fall in the category of ball games played within a restricted area. Our discussion will mostly contain applications which specific to the game of cricket, however in some cases, we will mention how similar techniques are applied in other games.

There are various issues which crop up when one tries to design and implement such a system. In the game of cricket, the general issues are:

- a) The distance at which the cameras see the pitch and the ball are dependent on the dimensions of each ground and can vary greatly.
- b) Just the individual images don't help too much; for the system to be of practical use, one must ensure that it can track the 3D trajectory of the ball with high precision. In order to get this accuracy, the field of view of each camera should be restricted to a small region – this means one needs more cameras to get the coverage of the entire field.

- c) Fielders and spectators might obstruct the camera's view of the ball and the ball might get lost in its flight in one or more of the cameras. The system should be robust enough to handle this, possibly by providing some redundancy.
- d) The ball might get confused with other similar objects – for instance, with flying birds or the shadow of the ball itself. The image processing techniques used need to take care of these issues. Luckily, there are techniques which are easy to implement and are well known to the Image Processing community on the whole, to take care of these.
- e) To help in judging LBW calls, the system needs to be made aware of the style of the batsman – whether he is right or left handed. This is because the rules of LBW are dependent on the position of the stumps and are not symmetrical about the middle stump. Thus, the system needs to detect whether a particular ball has pitched outside the leg stump of a batsman or not.
- f) To determine the points at which the ball makes contact with the pitch, the batsmen or other objects is very hard. This is because we don't really know these spots beforehand and the model and the real pictures taken by cameras need to be merged to give such a view.
- g) We will see how the Hawk-Eye technology successfully treats each of these issues and provides a robust system to be used in practice. The top-level schematic picture of the system and its various parts is as shown below (each color represents a block of steps which are related):



Fig. 1:- Hawk-Eye flow diagram

The figure above shows precisely the steps that are involved in the computation. The process is started with some calibration of the cameras. This is required to deal with the problem raised in 1 above, about the non-uniform distance of the cameras from the playing area. After this basic calibration

is done and the system is up and running, we can start processing the video input which we get from the cameras. In each of the images obtained, the first aim is to find the ball initially. Once this is done, a geometric algorithm is used to look at multiple images (which are 2D) and then combine them cleverly to get the co-ordinates of the ball in 3D space. This process is now repeated for multiple times every second (typically at the rate of 100 times per second). Thus, we have the position of the ball in 3D space at many moments in every second. The final step is to process these multiple positions and find a suitable fitting curve which best describes the flight of the ball. As we have sampled the positions of the ball at very short time intervals, the flight of the ball can be very accurately determined. A description of the exact algorithms involved in the entire process will be skipped here. We instead try to give an intuitive description of each step in great detail, so as to give the reader a feel of what goes into the system, without plunging into the gory details.

### III. STEP-BY-STEP DETAILS OF THE HAWK-EYE SYSTEM

In this section, we go into the technical details of the steps involved in the HAWKEYE system. The process, as done before, can be broken down into the following steps (we will divide the process into these seemingly disjoint steps so that it is easy to explain the details, however many of the steps are overlapping).

#### 3.1 THE CAMERAS

Typically, for a cricket field, 6 cameras are used. These cameras are placed around the field at roughly the places as indicated in the diagram below:

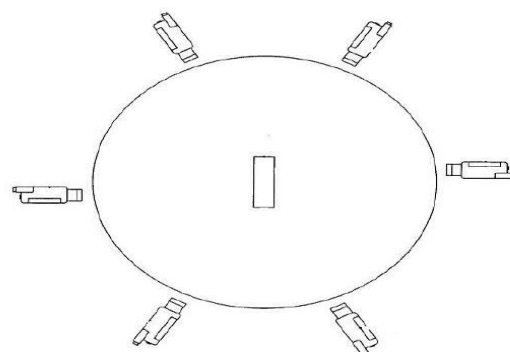


Fig. 2:- Camera setup

As one can see, the 6 cameras in use are positioned at roughly 60 degree from each other. They are placed high in the stands, so that there is lesser chance of their view being blocked by the fielders. There are two cameras, one each looking at the wickets directly in sideways fashion. These 6

cameras are calibrated according to the distance they are at from the pitch. In order to get good accuracy, one needs to restrict the view of each camera to a smaller region. This means each camera image would show a more prominent picture of the ball and hence the ball will be located more accurately. However, we also need to keep in mind that the whole field of play has to be covered by just the 6 cameras which are available. This puts some limitation on how restricted the view of a camera can be. Nevertheless, the accuracy obtained by using 6 cameras is acceptable to the standards prevalent today. Some further setting up is essential for the system to work correctly. The cameras need to be fixed to some frame of reference, which is defined very conveniently in terms of the wickets on the pitch, and the line joining them. This is useful when we want to use an automated program to merge images from different cameras to form one 3D image.

Also, to avoid unnecessary computation and make the system more efficient, the cameras can be operated in active or passive mode. In the passive mode, no imaging is done and hence the system is more or less completely inactive. The cameras can be triggered into active mode either by detecting some motion in the vicinity of the pitch, or manually by some external trigger. In either case, all the cameras are synchronized and go into active mode simultaneously. The cameras are then designed to stay in the active mode for a fixed time before going off into passive mode. This action of going into passive mode can be manually overridden in exceptional cases. The different modes for the cameras are especially effective for a game like cricket as the game involves significant pauses between phases of actual play.

As described in 5 in the list of issues, the system needs to know if the batsman is right or left handed. The front view cameras are used to do this. This information, as previously said is useful in making LBW decisions and formulating other statistics. For instance, we commonly see the analysis of a bowler's pitching areas done separately for a left and aright handed batsman. While this is not a very difficult task to do manually every time the batsman on strike changes, the system does provide some way of automating it. Once this setting is done, the cameras are ready to take pictures in their field of view and have them sent to a computer which processes them.

### **3.2 PREPARATION BEFORE STARTING TO PROCESS**

Additional features might be loaded into the system to enable it to process the data in a more reliable and useful manner. These might include a statistical generator, which is

used to produce statistics based on the data collected. These are the statistics which we see on television during and after the match for analysis. Such statistics can also be used by teams and players to study their game and devise strategies against their opponents. Indeed, the raw data about the paths of the ball might be too much for any human to digest and such statistics turn out to be easier to handle and understand. The statistics generator might also aid in storing data such as the average velocity of the ball. This data is crucial as it can help the ball detection algorithm to predict the rough location of the ball in an image given the position in the previous image. Such considerations are useful to reduce the computations involved in the processing of the data collected from the video cameras. Once such additional machinery is setup correctly, we are all set to start collecting data and start processing it to churn out tangible statistics and visualizations. It might be noted at this stage that there is some more information which might be required to process the data correctly. We will point out such things at later points in the paper, where it fits in more appropriately.

### **3.3 COLOR IMAGE PROCESSING JOB**

This part of the system can be further divided into 3 major parts:

- a) Identifying pixels representing the ball in each image.
- b) Applying some geometric algorithm on the set of images at each instant.
- c) Coming up with the 3D position of the ball in space.

We now explain each of these operations in detail:

#### **3.3.1 IDENTIFYING PIXELS REPRESENTING THE BALL IN EACH IMAGE.**

To identify the pixels representing the cricket ball in every image taken by each of the video cameras: An algorithm is used to find the pixels corresponding to the ball in the image obtained. The information which is used in order to achieve this is the size and shape of the ball. It should be noted that the system does not use the colour of the ball as that is not really same throughout the course of a game, nor is it same across all forms of cricket. A blob detection scheme can be used to detect around object in the image. Knowing the approximate size of the ball, we can eliminate other round objects, such as helmets worn by players. The shadow of the ball also will resemble the ball in shape and size and thus presents itself as a very viable candidate for a blob representing a group of pixels corresponding to the ball itself. The position of the sun at the given instant of time and also information about the position of the ball in previous images is used to make sure this confusion is avoided. Thus, by taking due care, we can be sure

that the round object which has been located is indeed the cricket ball, which is the object of interest. After this stage, we have as output the x and y coordinates of the ball in each image. In some cases, it might be the case that the system is unable to determine the exact position in some images. At such times, “Not Found” is returned by that particular camera. One must note at this point that 6 cameras are used to take images. Actually, in the ideal case one can do the job with just 4 cameras. Thus, we have some redundancy and hence, can afford to have a bad result from one of the cameras at some points and still produce a complete picture.

**3.3.2 GEOMETRIC ALGORITHM**

The data x and y co-ordinates from each camera (or a “Not Found” in some cases, which is ignored) is obtained by the Geometric Algorithm which is at work inside the Haw-Eye system. The image taken from each camera is just a 2D image and lacks depth. Now, knowing the exact positions of the cameras in space (with respect to the pitch) and the x and y co-ordinates of the ball in more than one of the images taken by these cameras, one can determine accurately the position of the ball in 3D.

**Triangulation:**

Triangulation is a process of determining the location of a point by measuring angles to it from either end fixed at baseline.

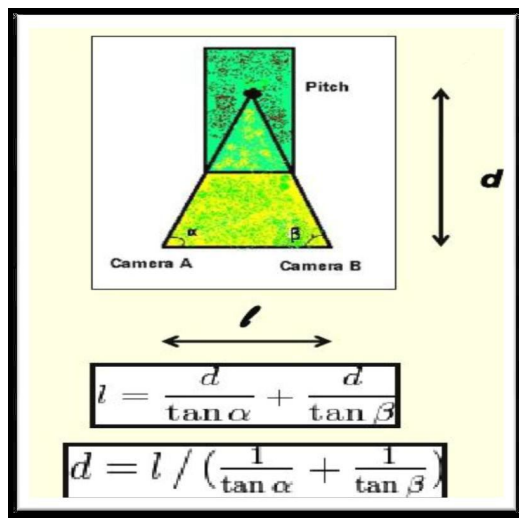


Fig. 3:- Triangulation

Let us consider the simple case in which we assume the cameras to be mounted at ground level, positioned with their vision parallel to the ground. We wish to get information about the 3D position of the ball from the positions (x1,y1) and (x2,y2) obtained by resolving the ball from 2D images

from Cameras 1 and 2 shown in the image below. The ball is actually at the position shown by the red circle, at some height above ground.

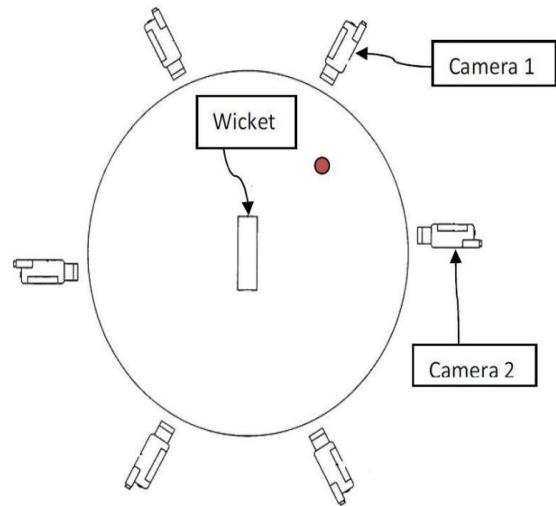


Fig. 4:- Ball Position

The view in the cameras will look something like the one shown below. The view below shows the picture as seen by Camera 2 in the figure above.

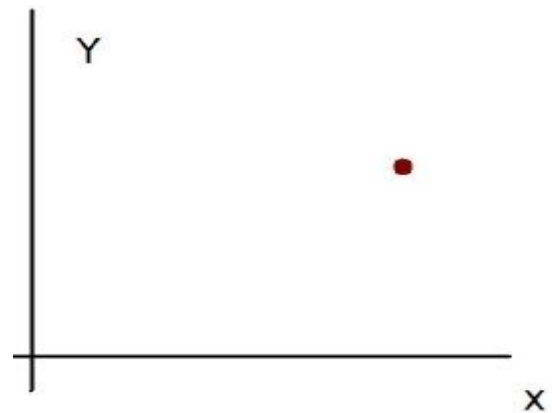


Fig. 5:- Camera 2 result

In this simplistic scenario, the height of the ball above the ground is given directly by the y co-ordinate in the images, y1 and y2. Both these values should ideally be equal, but we might want to take the average in case they are not exactly equal. Now, the one parameter we need to determine is the depth of the ball as measured by Camera 2. Once we have that information, we will have all the data to infer the position of the ball in 3D space with respect to the pitch. Note that we know the positions of the cameras with respect to the pitch in advance. Let us assume that the radial angle, as seen from the wickets marked in the figure, between Camera 1 and Camera 2 is  $\theta$  and the radius of the field is. Then, the depth of the ball as seen from Camera 2 is as follows:

$$\text{Depth} = r - (r \cos(\theta) + x_1 |\sin(\theta)|)$$

Thus, we see that knowing the co-ordinates of the ball in two cameras, we can get the position of the ball in 3Dspace with respect to one of the cameras and thus, with respect to the wickets. In the realistic case, the cameras are mounted high above the ground and thus, finding the height of the ball above the ground is not as trivial as it was here. One needs to rotate the axis correctly in order to do the calculations that were simple here as it concerned only planar geometry. In real life, cricket grounds are not perfectly circular and hence even that has to be taken into consideration. We do not go into those details here, but just note that it is standard mathematics to get the 3D co-ordinates of the ball given the information in two images.

### 3.3.3 3D POSITION OF THE BALL IN SPACE

The Geometric Algorithm described with the help of an example above provides us with a ready recipe to find the 3D position of the ball in space. We just use this method and as a result, now have the position of the ball as captured at that instant, in 3D space, with respect to any of the reference points we had considered while setting up the system.

## 3.4. PUTTING FRAMES AT VARIOUS TIMES TOGETHER

Now we have the exact position of the ball in 3D space at a given instant of time. Next, what needs to be done is putting together this data, collected at various time instants into a single picture which shows us the trajectory of the ball. We can split this part of the process into two parts. Again, the reader should understand that these parts are very much related and we split them here in our explanation just to make it easily understandable. The two parts to this computation are:

- a) Tracking the ball at various instants.
- b) Predicting the flight or trajectory of the ball.

### 3.4.1 TRACKING THE BALL AT VARIOUS INSTANTS

Suppose the images are taken by cameras at times  $0, t_1, \dots, t_n$  during the play of a single ball.

Doing the computation as described above at each time instant  $t_i, 0 \leq i \leq n$ , we will get  $n$  points, say  $(x_i, y_i, z_i)$  for  $0 \leq i \leq n$ . Now, on the model that we have built previously consisting of a picture of the pitch, ground and wickets etc., we plot these  $n$  points. When looked at in their proper sequence, these points tell us about the path followed by the ball when it travelled during the last ball that was played. With these points plotted

in the 3D space, we can move on to the next and final stage in the processing of a single delivery, namely, predicting the flight of the ball.

### 3.4.2 PREDICTING THE FLIGHT OR TRAJECTORY OF THE BALL

We have  $n$  points in space which we know represent the position of the ball at some particular time instants, which are also known. Now, there is a standard technique, used commonly in the field of Computer Aided Geometric Design which can be invoked here. This allows us to draw as good an approximation as required to the original curve, passing through the given points<sup>2</sup>. This technique gives us a curve which is continuous and differentiable, meaning it is smooth all along, starting at the first point and ending at the last point among our  $n$  points. This smooth curve is an approximation to the original curve which the ball would have followed. The more points we can get on the curve and the higher degree of polynomial basis we choose to use, we will end up with better approximations to the original curve. The better approximations obviously come at some additional cost – the added cost of computation of the approximation. Hence, the system uses some degree such that the computation time is small enough, at the same time the accuracy is acceptable. More can be done with the information about the points. We can also extend the curve to points which we have not been recorded at all – indeed, it might be the case that the ball struck the batsman and deflected away, but we want to see where the ball was headed, particularly to help adjudge LBW cases. This extension uses some basic mathematics and ensures that the extended curve is also smooth at all points, particularly at the point from where the extended part starts that is the last point which we have recorded among the points. During the flight of the ball, it might go through some points which are of special interest. These include the ball hitting the pitch, the stumps, and the batsman among others. These points are predicted by superimposing the trajectory which we built, onto the model that we have fed into the system. It should be noted that there is a possibility that such critical points may not be recorded in any of the images taken by the system and in such cases; the reliance is completely on the predicted flight of the ball. Also, for the particular key-event of the ball striking the batsman, the sideways cameras, which look directly at the wickets at either end of the pitch, are the most reliable sources.

## IV. APPLICATIONS OF HAWK-EYE

Hawk-Eye has had far-reaching consequences in many sports. Primarily in cricket, Hawk-Eye is a process that makes the current judgmental call on a LBW decision, very

predictive. While no technology is flawless and Hawk-Eye has its own share of these, it is up to 99.9% accurate. This has made the LBW decision, a predictive one. More importantly, such technology can be used to evaluate the skills of the umpire as well.

#### 4.1 CRICKET

The England Cricket Board (ECB) has already set-up the Hawk-Eye system not only at about ten cricket venues around the country but also in the training academy to aid umpires, as well. While the Hawk-Eye has made its mark and derives its appeal from the ability to predict the flight of the delivery, it is a very useful tool for collecting statistics. The information associated with each delivery bowled is routinely processed, even when the outcome of the delivery is not doubted. As a result, the strategy used by a bowler as a function of bowling spells, delivery no. in the over, batsman facing the delivery and so on can be gauged. Similarly, the scoring patterns of a batsman around the ground using wagon-wheels are routine in match day telecasts. These are so cleverly generated that they give a real-life feel to it. Commentators also are able to move them about to make a finer point, about a batsman. However appealing and nice that it may seem, a keen cricketing eye will notice that the wagon wheel is less accurate than the other data. This is because the wagon-wheel is generated from data collected from outside the pre-determined pitch area. The location, depth, trajectory of the ball in-flight at an arbitrary point on the ground is more difficult to determine, than when it is on the pitch.

##### 4.1.1 LBW DECISIONS

As mentioned previously, the Hawk-Eye can accurately capture the trajectory of the ball and also predict the future direction of the ball using mathematical calculations. This is put to use in deciding whether a batsman was OUT LBW on a particular ball. Thus, the system determines the exact point at which the ball struck the batsman. Using the trajectory of the ball up to that point, the system predicts the path the ball would have taken had the batsman not been present in the way. Thus one can know the lateral position of the ball with respect to the stumps as well as the height of the ball at the point when it reaches the line of the stumps. The figure below gives an example of the trajectory of the ball being predicted. Note that in this picture, the system has got rid of the batsman from the picture so as to give us a complete view of the path of the ball since it left the bowler's hand. This is exactly what one needs to decide if the ball would have hit the stumps and if that is the case, the batsman has a chance of being given

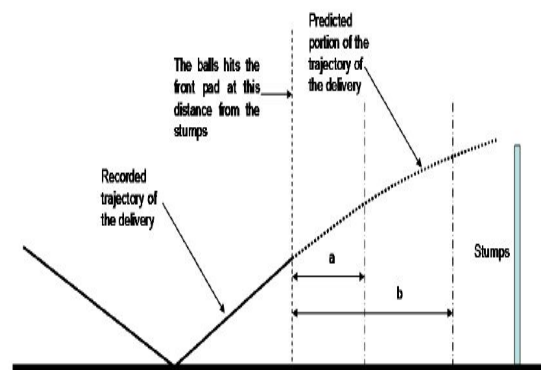


Fig. 6:- LBW Trajectory

The system is well equipped to handle the various complex clauses which the LBW rule has. For instance, it can check if the ball had pitched outside the leg stump of the batsman. If this is the case, the batsman is NOT OUT even if the ball is going on to the stumps. Recall that the front view cameras are used to determine whether a batsman is right or left handed. That information is useful here. Another clause states that the batsman should not be given OUT if he is hit outside the line of off-stump and is attempting to play a shot. Now, the part of whether the batsman is playing a shot has to be decided manually and the system is not capable of doing it. However, the point of impact is accurately known and one can see exactly where the batsman was hit.

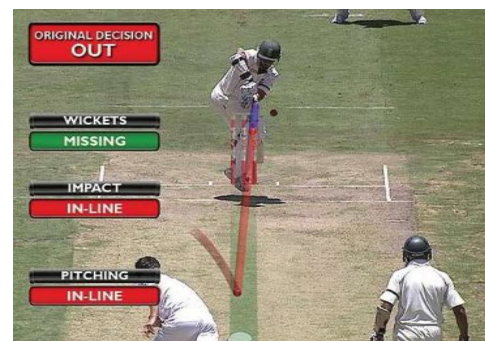


Fig. 7:- LBW decision using Hawk-Eye

The kind of accuracy which HAWKEYE offers is difficult to get for any human umpire. The system also includes a way to do probabilistic analysis and hence bring in the factor of “benefit of doubt” which goes to batsman currently. The main idea behind this is to have a region which the human umpire would believe the ball would have been in this region is just taken to be a circle centred at the accurate position of the ball and radius. The value of this radius is calculated taking into consideration the distance between the point of impact of the ball with the batsman and the stumps – that is, the distance which the ball is yet to cover. This models quite accurately the uncertainty which the umpires feel while making the decision manually. Thus, if the batsman is playing

forward, the radius will have a higher value, than when the batsman is struck, playing back. To keep the “benefit of doubt” still with the batsman, the decision goes in favour of the bowler only if a significant portion of the probable region (circle or radius described above) lies in line with the stumps. The system thus is very robust and seems to be better than human umpires as it stands – and it can only improve.

#### 4.1.2 DE-SPIN

The De-Spin graphics help us in understanding how the ball has deviated after pitching. The graphic produced shows the predicted path of the ball, had it held its line even after pitching. This is particularly interesting to look at, in the case of spinners, where one can see both the flight being given by the bowler and the spin that he manages to extract from the pitch. Looking at the action and the De-Spin graphics for a particular bowler is useful for batsman to notice any changes in action when the spinner is bowling a “trick” ball –which might be a googly or flipper in the case of a leg spinner, or a “doosra” in the case of an off spinner.

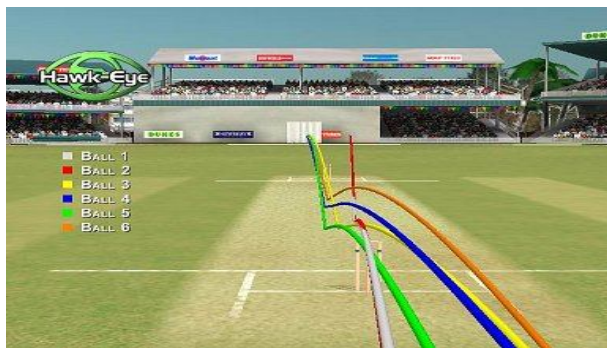


Fig. 8:- De-spin using Hawk-Eye

#### 4.1.3 WAGON WHEELS

The trajectories which the ball has taken after being hit by the batsman are recorded in the system. This is used to generate a graphic showing 1s, 2s, 3s, 4s, and 6s all in different colors for a batsman. These details allow the commentators, spectators and players to analyze the scoring areas of the batsman and also judge if he has played more shots along the turf or in the air. Such information is vital for a fielding captain, who might alter his field placement in subsequent matches to adapt to the hitting pattern of a particular batsman.



Fig. 9:- Wagon Wheel

#### 4.2 TENNIS

Hawk-eye was first used in tennis in the year 2004(US open tennis).In tennis Hawk-eye generates the impact of the ball whether the ball is “IN” or “OUT” the line of tennis court.In Tennis, lines calls made by Hawk-Eye are completely accurate. Tennis has been quick to adapt to this technology and Hawk-Eye arbitrations are legal since the NASDAQ-100 tennis tournament. Players can challenge line calls, following which Hawk-Eye determines whether the ball was pitched in or out.

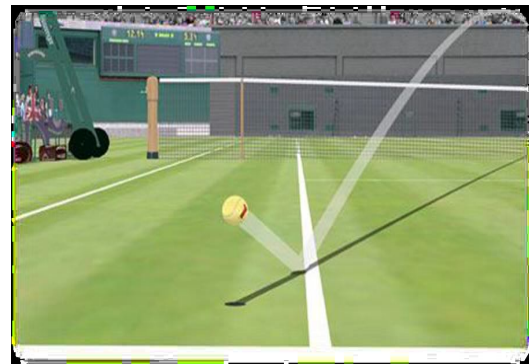


Fig. 10:- Tennis Hawk-Eye

#### V. CONCLUSION

We have looked at various aspects of the Hawk-Eye technology. Initially, we outlined the main problems which one could encounter while trying to implement such a system for a sport like cricket. Then, we looked into the details of each step of the process which finally gives us the wonderful looking graphics that we see on TV during cricket analysis shows. We got a fair understanding of the algorithms and mathematics which goes into the system. With the help of examples, we looked at the applications which the technology finds in modern day sport, with cricket being our main focus. We got an understanding of how the graphics can be produced, using the setup, which also was described in detail. As the technology getting better we can see the role of this technology in various sports like rugby, baseball, football,

rugby, ice-hockey, horse racing, badminton and many other sports events. It enables the match officials to take fair decisions during the game. Overall the technology which is used in Hawk-Eye is so accurate that there is a belief in everyone about its accuracy in terms of decision making. So, whichever sports it is applied it gives a better dimension to the game. So, in future we will see in every sports played around the world, the Hawk-Eye will play a major role in decision making process. Also, the technology is getting so advanced that new innovations will also be done to improve the quality of the service which is given by this technology.

### REFERENCES

- [1] Video Processor Systems for Ball Tracking in Ball Games. International Patent, publishing number WO 01/41884 A1
- [2] “Hawk Eye: A Logical Innovative Technology Use in Sports for Effective Decision Making” By: Bal Singh Baljinder, Dureja Gaurav DOI: 10.2478/v10237-012-0006-6
- [3] <http://singularityhub.com/2011/03/09/hawk-eye-augments-referees-with-computers-professional-sports-are-evolving/>