International Journal of Applied Engineering Research
ISSN 0973-4562 Volume 10, Number 10 (2015) pp. 25493-25500

http://www.ripublication.com

Automating Runout Decisions In Cricket Using Image Processing

Sabarish B A [1], Sheethal Mohan[2], D P MamthaShri[3] ,Rishikesh C Ajit[4], A VR Arun[5]

[1]Assistant Professor
Department of Computer Science and Engineering,
Amrita School of Engineering,
Amrita Vishwa Vidyapeetham (University),
Coimbatore - 641 112.
{2,3,4,5}Final year B.Tech,
Computer Science and Engineering Student
Amrita School of Engineering,
Amrita Vishwa Vidyapeetham (University),
Coimbatore - 641 112.
[1]sabarishpm@gmail.com, [2]sheethalmohan93@gmail.com, [3]shri.mamtha@gmail.com
[4]rishikesh.ajit@yahoo.in, [5]avrarun2511@gmail.com

Abstract

Decision making is crucial in any part of the game to ensure fairness among the teams. An unfair decision due to human error may change the result of the game. Cricket is the game which is most worshiped after the game of soccer. This paper introduces the concept of automating run out decisions given by the third umpire, whichis achieved through various image processing techniques and algorithms like Speeded Up Robust Features, absolute difference algorithm, edge detection etc. We work on the frame in which the bails have first deflected. Popping crease position and leg/bat position of the batsman in the selected frame is compared to make a decision on run out. Implementation of the system shows better accuracy than human decision and also improves the quality of decision making process.

Keywords: SURF, descriptor, detector, interest points.

Introduction

Cricket is the second most viewed sports and has commercial importance. In the game of cricket, the need to have a third umpire was brought into action when the reliability of the decisions made by the on field umpires was questioned. The game usually calls for the third umpire when the on field umpire is unsure of an incident. The third umpire holds the advantage of replaying the scene several times, examining it and then arriving at a decision. This in turn consumes time and the players in the field have to stay on hold and not resume the match until the third umpire arrives at a conclusion. In this paper, we propose a methodology to automate the decisions which would save time and would accurately predict the result. This is achieved through various image processing algorithms. This paper deals with methods to validate that of only the run out results. In future, if this proposed method seems efficient and effective, this can be further extended to predicting the rest of the game results.

Preleminaries:

Image processing is a method which takes image as input and performs some operations on it to extract useful information from the image. Image processing is mainly used for identifying objects in an image, measuring objects in an image, identifying image of interest, visualization and image restoration.In this paper, we have used image processing to identify the frame in which the bails have first deflected as well as to find the crease and leg/bat position of the batsman to check if the input video footage is a run out or not. Interest points are those which have a clear mathematical definition or a well-defined position in the image space. Features can be edges, interest points, corners or ridges. Image matching algorithms can be gray based or feature based. Gray based matching is vulnerable to illumination variance and noise. Two promising algorithms for feature point detection and matching are SURF and SIFT. Comparing SIFT against SURF, SURF outperforms SIFT. SURF is faster and more robust against several image transformations than SIFT. SURF algorithm is used to identify the frame of interest. Frame of interest in this system is the frame in which bails have first deflected. The interest points in the base wicket image are compared with the interest points in each frame. The first frame in which variance is above a threshold is selected as the frame of interest. Further processing is done only on the frame of interest. The frame of interest is converted to gray scale. In gray scale conversion, we extract the red, green, blue values of each pixel and apply an equation to convert the RGB values to a single gray value. The RGB value is then replaced by the gray value. The colored image is converted to gray scale to increase the edge and feature detection capability.

A. *Image matching and comparison using SURF algorithm:*

1. *Video capture:*The input can be a video from a camera or an already stored video footage. First search for camera. If a connected camera is available, capture the video sequence and store it on your computer. Else, consider the stored video footage as input.
2. *Split into frames:* Video is split into frames for applying SURF on each frame
3. *Identification of Region of Interest:* Hessian matrix is used to find regions of interest. Hessian detector chooses interest points based on the Hessian matrix at each location. Hessian matrix at location X(x,y) in an image I is

$$H(X) = \begin{bmatrix} L_{xx}(x) & L_{xy}(X) \\ L_{xy}(x) & L_{yy}(X) \end{bmatrix}$$

$L_{xx}$is the second order partial derivative in x direction, $L_{xy}$is the second order partial derivative in x andy direction and $L_{yy}$is the second order partial derivative in y direction.

Hessian detector identifies more reliable interest points than most of the detector

1) Formation of SURF descriptor:

Descriptors are used to describe the previously found image points. Descriptor calculation consists of two steps: Orientation assignment and descriptor based on sum of Haar wavelet responses. For orientation assignment, a circular region of radius 6s is chosen around the interest point(s is the scale). Haar wavelet response in x and y direction are calculated for this region. The responses are then weighed by Gaussian function centered at the point of interest. The responses within the sliding orientation window of 60 degrees are summed to estimate the dominant orientation. Next step is to extract the descriptor.A square region of size 20s centered at the point of interest and oriented along the previously calculated orientation is chosen. The window is subdivided into 4x4 sub-regions and horizontal and vertical Haar wavelet responses are calculated for each sub-region. The descriptor vector for each sub-region is given by

$$\mathbf{V = (\Sigma\, dx\, ,\, \Sigma\, dy\, ,\, \Sigma\, l\, dx\, l\, ,\, \Sigma\, I\, dy\, l\,)}$$

Where dx is the horizontal Haar wavelet response and dy is the vertical Haar wavelet response.

2) SURF matching:

We try to match the interest points in the test image and reference image. This is done by comparing the difference is Euclidean distance between the corresponding descriptors. A matching is found if the distance is less than a particular threshold.

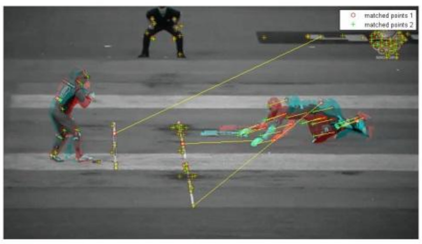


Figure: Screenshot of SURF points

B. Edge detection using Canny edge detector:

Edge detection maintains the structural information in an image and removes the irrelevant information. Edges are areas with strong intensity contrast. Canny edge detection is a reliable technique to find the edges in an image. The steps in the development canny edge detection algorithm are

1. *Smoothing using Gaussian filter:*

Smoothing is done to eliminate noise in an image. Gaussian filter is used to smooth the image. The kernel of the Gaussian filter is shown below

$$B = 1/159 \begin{bmatrix} 2 & 4 & 5 & 4 & 2 \\ 4 & 9 & 12 & 9 & 4 \\ 5 & 12 & 15 & 12 & 5 \\ 4 & 9 & 12 & 9 & 4 \\ 2 & 4 & 5 & 4 & 2 \end{bmatrix}$$

Here, standard deviation is 1.4.

As the size of the Gaussian mask increases, the detector's noise sensitivity decreases.

2. *Find edge strength and edge direction:*

Gradients at each pixel are determined by using Sobel's operator. It uses two 3x3 convolution masks-one to find the gradient in the x-axis (columns) and the other to find the gradient in the y-axis (rows).

$$\begin{array}{ll} G_x = & \\ -20 & \end{array} \begin{bmatrix} -1 & 0 & 1 \\ & 2 & \\ -1 & 0 & 1 \end{bmatrix}$$

$$G_y = \begin{bmatrix} 1 & 2 & 1 \\ 0 & 0 & 0 \\ -1 & -2 & -1 \end{bmatrix}$$

Edge strength |G| is given by,

$$|G| = |G_x| + |G_y|$$

Edge direction is given by the formula,

$$\Theta = \tan^{-1} (|G_y| / |G_x|)$$

Care should be taken if the value of $G_x$is zero. If G_X is zero, the direction depends on G_Y. If G_y is zero, then direction is taken as zero degrees. Else, the direction is ninety degrees.

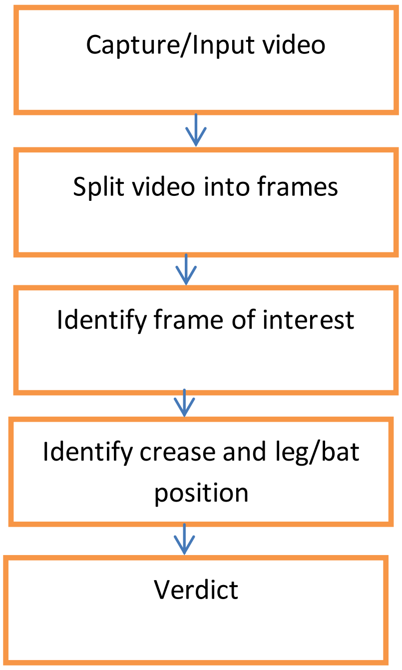
3. *Non-maximum suppression:*

Non-maxima suppression is used to thin the edges. Blurred edges are converted to sharp edges. In this algorithm, the edge strength of the current pixel is compared with the edge strength of positive and negative gradient direction. If the current pixel's edge strength has largest value than all the pixels in the same direction, the value will be preserved. Else, it will be suppressed.

4. Double thresholding:
We should remove the edges caused by noise. For this we fix two thresholds. The edge pixels above the high threshold are considered strong. Edges below the low threshold are suppressed and the edge pixels between high and low thresholds are considered as weak.

5. Edge tracking:
All the strong edges are considered as true edges. Weak pixels which are connected to the strong pixels are considered true edges. The other weak edges are removed. To check if the weak edge is connected to a strong edge, we use Binary Large OBject analysis. Here, we consider the 8-connected neighborhood. If there is atleast one strong pixel in the BLOB, then the weak edge is considered as a true edge.

Basic Architechture of The Proposed System:



1. Capture/Input video and split the video into frames:
Input is a video footage. Frames are fundamental entity of a video file. Split the video into frames and save each frame as an image file. Individual frames are considered and worked on.

2. Identify the frame of interest

The frames obtained after the initial processing are tested one by one for a change in the position of the wicket/bails, which indicates that the wicket has been hit by the ball. Each frame is taken into consideration and checked for change in position of the wicket/bails by matching the interest points of the base wicket image and the frame under consideration. If the variance is within an acceptable range the next frame is selected and tested for the same condition and the process is repeated until there is a significant difference in the interest point locations and their descriptors. The frame chosen is then tested with different conditions to give the final verdict

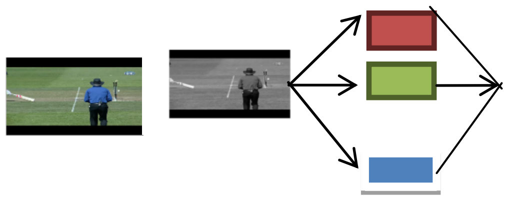
Figure: Screenshot of the frame of interest

3. Convert to gray scale:

In a colored image, every pixel has RGB components. That is, every pixel is represented by three matrices-red matrix, green matrix and blue matrix, each representing the corresponding color intensities. Color affects the processing speed and it is recommended to convert it to gray scale. One of the simplest method to convert an RGB image to gray scale is averaging.

gray = (red + green + blue)/3

This method is fast and simple



4. Identify crease and leg/bat position:

Hough transformation can be used to find the crease. It is one of the most widely used feature extraction technique.Use Canny edge detector to reduce the image to a set of edges. The resulting image will be the input to Hough transformation. Edges are represented in polar form with ϴ being the angle between the x-axis and the closest point on the line from origin and r is the distance from origin to the closest point.Hough Transform uses the concept of histogram. Pixels that map to the same ϴ and r can be assumed to be of the same line. Longest and strongest line of specified width and length is chosen as the crease.We can easily identify the batsman by identifying the moving object. This can be done by finding the absolute difference of consecutive frames.

5. Verdict:
There are two cases to be considered.

Case 1: Batsman moving towards the right popping crease
Right most co-ordinate of the bat/leg (bat point) is considered and a straight horizontal line is drawn passing through that point. The point where the line meets the crease is saved as the corresponding crease point. The x co-ordinates of crease point and theleg/bat point are compared. If x co-ordinate of crease is greater than that of bat/leg point, then it is not a run out.

Case 2: Batsman moving towards the left popping crease
Left most co-ordinate of the bat/leg position of the batsman is saved as the bat/leg point. A straight horizontal line passing through the bat/leg point is drawn to meet the crease. The point of contact of the line with the crease is chosen as the crease point. x co-ordinates of both the selected point are compared. If x co-ordinate of bat/leg point is lesser than or equal to crease point, it is a runout.

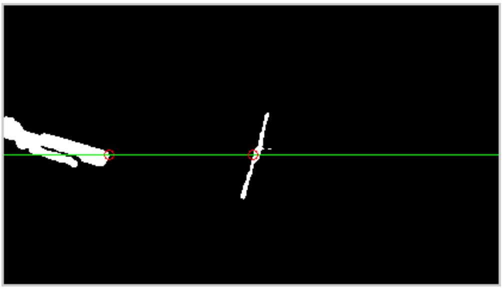


Figure: Screenshot of bat point and crease point

Conclusion

In this paper, a way to automate run out using image processing techniques has been proposed. Sometimes, deciding whether a batsman has been run out is a difficult and tricky task. The on field umpire has to replay the video and analyze it. This might halt the game for a long time which spoils the spirit of the game. This system uses image processing techniques like SURF to find the frame in which deflection of bails have first occurred and then compare the crease and bat/leg positions to come up with the verdict in less than a second.

This system is extremely efficient and addresses the problem of reducing the time taken by the third umpire on taking decisions concerning run out conditions.

References

[1] H. Kandil and A. Atwan ," A Comparative Study between SIFT- Particle and SURF-Particle Video Tracking Algorithms", International Journal of Signal Processing, Image Processing and Pattern Recognition, Vol. 5, No. 3, September, 2012.

[2] Johannes Bauer, NikoS¨underhauf and Peter Protzel," COMPARING SEVERAL IMPLEMENTATIONS OF TWO RECENTLY PUBLISHED FEATURE DETECTORS", In Proc. of the International Conference on Intelligent and Autonomous Systems, IAV, Toulouse, France.

[3] ShashankYeole, Nisheet Sharma, YogeshShinde and SufiyanShaikh,**"** Use of Image Processing Techniques for Making Run out Decision in Cricket**"** , International Journal Of Engineering And Computer Science ISSN:2319-7242 Volume 3, Issue 5 may, 2014 Page No. 6212-6215.

[4] Kauleshwar Prasad, Richa Sharma and DeepikaWadhwani," A Review on Object Detection in Video Processing", International Journal of u- and e- Service, Science and Technology Vol. 5, No. 4, December, 2012.

[5] WazirZada Khan, Mohammed Y Aalsalem and QuratulAinArshad**,"** The Aware Cricket Ground**",** IJCSI International Journal of Computer Science Issues, Vol. 8, Issue 4, No 2, July 2011.

[6] P. Ashok Kumar," Automated Decisions in Aware Cricket Ground: A Proposal", International Journal of Information and Education Technology, Vol. 2, No. 3, June 2012.

[7] Adrian G. Bors¸ and Ioannis Pitas," Prediction and Tracking of Moving Objects in Image Sequences", IEEE TRANSACTIONS ON IMAGE PROCESSING, vol. 9, NO. 8, August 2000.

[8] N. Paragios and G. Tziritas," Adaptive detection and localization of moving objects in image sequences", Signal Processing: *Image Communication* 14 (1999) 277Ð296.

[9] Isaac Cohen and G´erard Medioni," Detecting and Tracking Moving Objects for Video Surveillance", IEEE Proc. Computer Vision and Pattern Recognition Jun. 23-25, 1999.

[10] J.JasmineAnitha and S.M.Deepa, "Tracking and Recognition of Objects using SURF Descriptor and Harris Corner Detection", International Journal of Current Engineering and Technology, Vol.4, No.2 (April 2014).

[11] LiLi," Image Matching Algorithm based on Feature-point and DAISY Descriptor", JOURNAL OF MULTIMEDIA, vol. 9, no. 6, June 2014.

[12] FarhanaWani, Adeel Ahmed khan and Dr. S Basavaraj Patil," A Survey of Image Processing Techniques for Identification and Tracking of Objects from a Video Sequence", International Journal of Computer & Organization Trends,vol 6,no 1 , Mar 2014.

[13] G.T. Shrivakshan and Dr.C. Chandrasekar ,"A Comparison of various Edge Detection Techniques used in Image Processing", IJCSI International Journal of Computer Science Issues, Vol. 9, Issue 5, No 1, September 2012.