

Gesture Recognition to Make Umpire Decisions

Lesha Bhansali

Meera Narvekar

ABSTRACT

With the growing increase of the utilization of technology in sports; our novel project the Umpire gesture Recognition System aims squarely to introduce a more robust technology to show Umpire choices with the assistance of Gesture Recognition and trailing of hand movement of the Umpire. This technology helps to alleviate the burden of the score-keepers. It conjointly minimizes errors in displaying Umpire choices therefore adding to a more robust viewing expertise. This paper presents four easy but economical ways to implement hand gesture recognition specifically Subtraction, Gradient, Principal elements Analysis and Rotation Invariant. The ways used were in to retrieve the right matches. The results supported speed and accuracy was analysed.

Keywords

Gestures, Umpire, Cricket, Euclidean, Eigen Vector, Gradient, GUI,

1. INTRODUCTION

As the game of cricket has evolved over the decades, so has the technology behind the game. Our system "Umpire Hand Gesture Recognition System"(UHGRS) created has been instrumental in increasing the accuracy of the decisions, provide flawless replays to help see what the naked eye misses. The technology does not only benefit the players but is equally instrumental in providing a better viewing experience to the audience. Often while watching a cricket match live at a stadium, it is difficult to discern the umpire decisions due to poor visibility. The score-keepers have to be constantly alert as to when there is a fall of a wicket or a wide ball, dead ball, no ball etc. "To err is human". As we humans are prone to error this technology come to their rescue.. If the scorekeeper accidentally misses to display the correct signals on the flat screen, the audience will definitely disapprove it. Thus, the "UHGRS" is a clever way to use technology and minimize the errors associated with displaying the correct umpire decisions. It also helps to ease the effort of the score-keepers. The system detects the movements of the umpires hands and palms. For example, when an umpire raises his hand to signal OUT, the camera captures and recognizes this gesture. It then matches the image with its database and then displays "OUT" on the screen. The images are labelled using integer numbers starting from 1. The database was created using ten different images for all the six hand gestures.

2. METHODOLOGY

The gestures being shown in the Figure 1 below mainly displays the various gestures the umpire makes during a cricket match. For example the first image where the umpire is waving both his hands represents six runs scored by the batsman.

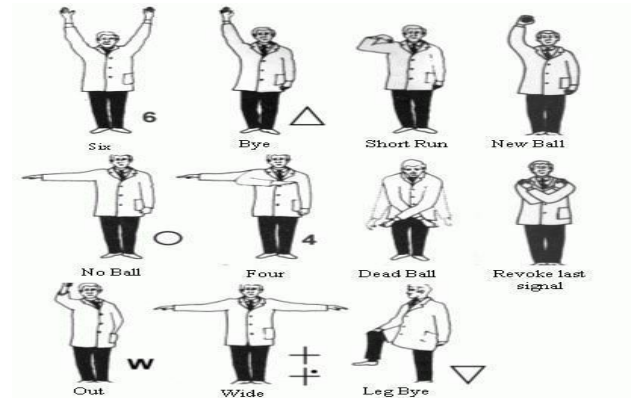


Figure 1

One of the foremost goals of Hand Gesture Recognition is to identify hand gestures and classify them as accurately as potential. The methods used for the development of this technology are:

- Subtraction Method: this method involves a straightforward subtraction between a pair of photos, constituent per constituent to see them.
- Gradient Method: this method involves investigating edges in a very image and work the sensible pixels that comprise them for each row or column thus making a comparison.
- Region of Interest: The goal is to figure out and study the Eigenvectors of the varied photos thus to specific each image with its principal elements (Eigenvectors).

For Hand gesture recognition systems to be successful it is necessary to develop testing and grading procedures that specifically address this technique and would begin with image Subtraction method techniques like noise removal, followed by (low-level) feature extraction to search out lines, regions and areas.

2.1 Subtraction technique

This is an easy technique to implement. This technique involves first changing all the pictures into black and white. It is done by choosing a threshold value for the pixels i.e., if T is that the hand-picked threshold value and p is the component intensity price then replace p=0 if p<T. We then performed direct subtraction of every pixels within the image with the corresponding pixels in each of the pictures and calculated the

Euclidian distance between them. The value of 'd' can imply the highest match. Here, 'd' [1] is outlined in terms of f1 and f2, where they are area units of two pictures being compared. X and Y represents the coordinates of the umpires position.

$$d = \sqrt{\sum \sum (f1(x,y) - f2(x,y))^2}$$

2.2 Gradient technique

The steps concerned during this method are as follows: First of all, we implemented the gradient magnitude calculation. The aim is to outline wherever within the image the most important gradient magnitudes are present. Then, it'll be used to determine a threshold within the gradients so as to filter the concerned area of interest (hands, palms and legs) and to discard all the background.

The magnitude is calculated to determine the area of interest by eliminating the background. This is don't with the help of the coordinates of the area of concern. Hence [1] x,y represent these coordiantes.

$$\text{Magnitude} = \sqrt{dx^2 + dy^2}$$

Sobel filter, sometimes called sobel **operator**, is used in image processing and computer vision, particularly within edge detection algorithms, and creates an image which emphasizes edges and transitions.

This is used to approximate the gradients. The sobel matrix helps in identifying the intensity of the image. The greater the value the higher the intensity in that region. The Sobel operator used for gradient-x is as shown below. Where greater the number means more emphasis on that region.

1	0	0
2	0	2
-1	0	1

Similarly Sobel operator used for gradient-y is as shown below:-

2	1	1
-2	0	-2
-1	-1	1

Then, we used a mathematician filter to blur the image and have a homogeneous image. It allowed to get higher ends up in the gradient magnitude. The goal of this filter is to erase the background defects. It's very vital to obtain a regular background to avoid noise. We created a gradient magnitude threshold that had to erase the lower levels gradients so as to keep the higher ones. This cut all the noise and regularized the background. Then, succeeding step was to calculate the geometric distance between the vectors of the various pictures analysed. This half was formed to check the various photos, by scrutiny of histograms. With this, we calculated area of unit to acknowledge the various gestures.

2.3 Region Of Interest

During this section, we studied the hand gesture recognition through Principal element Analysis, however we needed some mathematical background to know the strategy. This technique is termed Eigenfaces. It's a helpful applied mathematics technique that has found application in numerous fields (such as face recognition and image compression). This is often conjointly is a standard technique for locating patterns in knowledge of high dimension too. Before analyzing an outline of this technique, we first introduce mathematical techniques which will be used.

A. Mathematical Backgrounds:

1) Standard Deviation:

For the notation, we used the image X to discuss with the complete sample and used the image Xi to point a particular knowledge of the sample. The formulae below calculates standard deviation. [2]

$$\frac{\sum_{i=1}^n Xi}{n}$$

2) Variance

Variance is another live information in every set. It calculated the variations in the image to help identify patterns.

3) Covariance

Covariance is expressed as [2]:

$$\frac{\sum (X_i - \bar{X})(X_i - \bar{X})}{(n - 1)}$$

4) Eigenvectors

The eigenvector of an operator area unit which is a nonvector, is converted to a scalar multiple. The scalar is then known as the Eigenvalue related to the eigenvectors.

5) Eigen value

Each eigenvector is associated with an Eigenvalue. The Eigenvalue offers us data concerning the importance of the eigenvector. The Eigenvalues area unit very vital, as they agree upon some threshold to filter the nonsignificant eigenvectors, so we will keep simply the principal ones.

B. Main Steps of the method:

First of all, we had to make an information set. The aim is to decide on variety of images and which have good resolution as these are most effective for recognition with the little information. Then, succeeding step is to cypher the mean from each of the information dimensions in terms of x and y. The mean ablated is typical across every dimension. The step 3 is to calculate the variance matrix of the information. We had to find the simplest way to find out the principal eigenvectors instead of calculating the large variance matrix. The strategy consists in selecting a brand new variance matrix. Variance matrix for A (matrix calculated using Sobel filter) is equal to C and C is calculated as:

$$C = A * A'$$

Then, the eigenvectors and therefore the Eigenvalues of C area states the principal elements of our knowledge set. The remaining eigenvectors can have Eigenvalue around zero. Then, we calculated the eigenvectors and therefore the Eigenvalues of the variance matrix. This gave us the principal orientation of the information

3. PROPOSED SYSTEM

The first side to put into thought is the style of underlying structure design. An important reason for choosing this framework is that it is user-friendly. This section describes intimately the design of the system and roles played by every layer. With the assistance of MATLAB we could do it in a simple manner. The very last step was to check by calculating the geometric distance between the coefficients that area unit before every eigenvector.

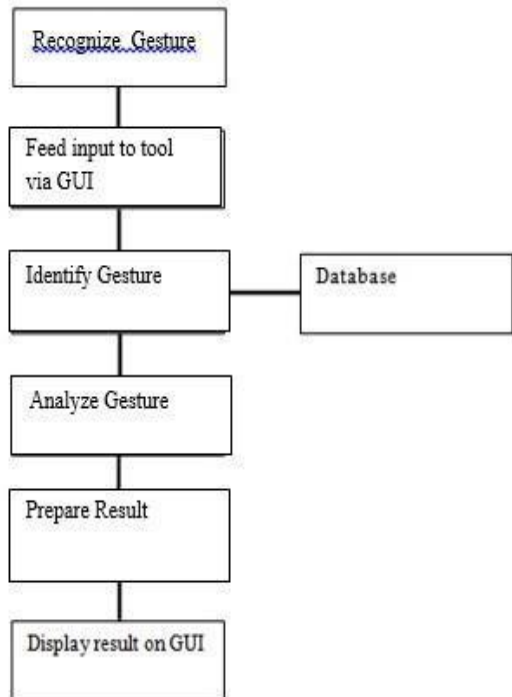


Figure II

3.1 Presentation Layer

This layer contains the various UI elements that are visible on the screen. It represents the interface which is displayed and developed for the desktop. It is the information translator and ensures that data is given within the correct format. It conjointly acts as the layer for taking input from the user.

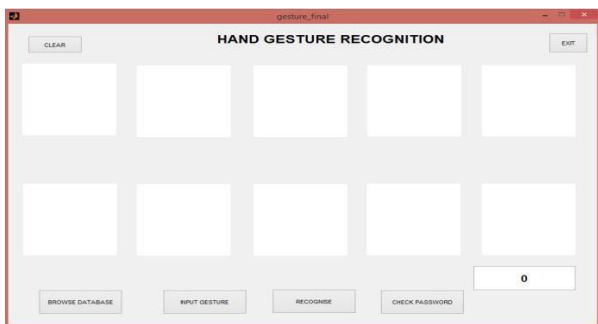


Figure III

3.2 Application Layer

It contains components for presentation and business logic. It is used to communicate between the data layer and the presentation layer. This layer consists of the Matlab code.

3.3 Data Layer

It contains all the images of the defined gestures for the purpose of matching. Where each category of gesture is given a specific number. In our case the numbers used are one to six. Each category further contains twenty images considering different factors like background of the umpire, size of this hands and legs, the amount of noise etc.

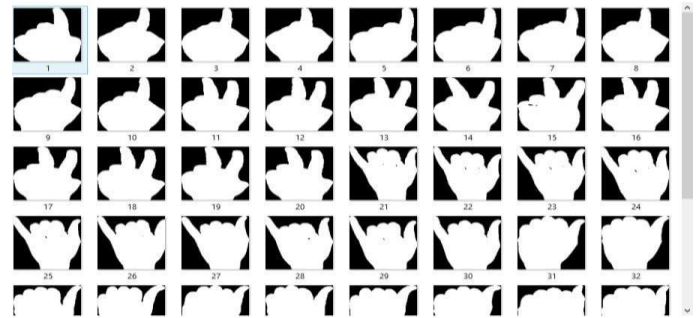


Figure IV

4. TESTING AND RESULTS

This section starts with a walkthrough of the system as a user. The section ends with associate degree analysis study conducted to assess how this approach works.

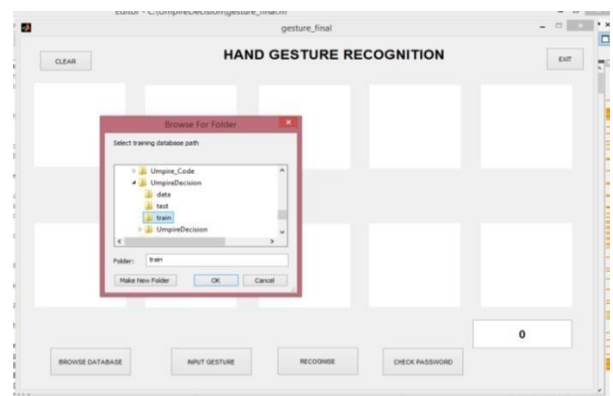


Figure V

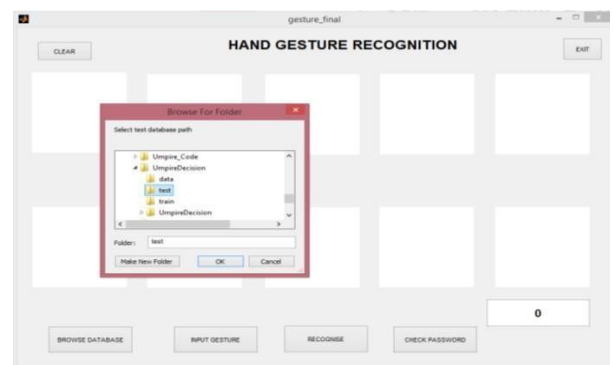


Figure VI

We compared the assorted algorithms supported the results obtained by checking the MATLAB implementations with a test information for six gestures containing twenty pictures for every gesture. Every image was processed with the 3 methods mentioned above. Then the strategy that works best for a selected kind of gesture was inferred supported and the results were obtained. We tested the subsequent six gestures particularly OUT, SIX, NEWBALL, NO-BALL, DEAD-BALL, FOUR. The diagram below explains the flow of our system. This is a demonstration from our project. The GUI looks like Figure4. We input the images in the first five frames where multiple consecutive images are captured. These five images are then compared to our database. A perfect match for the gesture is obtained and the images from our database in the frame below.

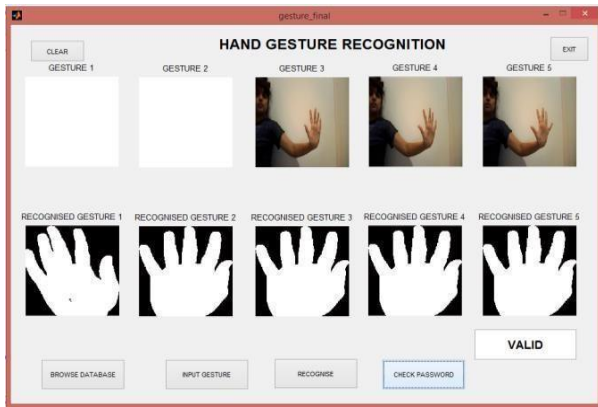


Figure VII

5. CONCLUSION AND FUTURE WORK

We have conferred results on the recognition of Umpire's gestures during a cricket match. The novelty of our work is that we've had an inclination to use the recognition of gesture to display the output directly on screens. Gesture recognition is then performed among the device domain that avoids the problems associated with correct image segmentation. We have an inclination to use our approach to the popularity of various sports. Results show that our recognition system is capable of recognising a group of six umpire gestures from the game of cricket and performs best once using a feature set. We plan of increasing the range of our database to make our system a full-proof one. Also in addition we want to show the performance of segmenting gestures from a stream of continuous gestures by selecting candidate gestures by the existence of movement.

6. ACKNOWLEDGEMENT

The authors wish to thank all the staff and especially Prof.

Prakash Paranjpe of Department of Computer Engineering. We would also like to thank our honourable principal Dr. Hari Vasudevan of D. J. Sanghvi College of Engineering for giving us the facilities and providing us with a propitious environment for working in college. We would also like to thank S.V.K.M. for encouraging us in such cocurricular activities.

7. REFERENCES

- [1] Xia Liu and Kikuo Fujimura, "Hand Gesture Recognition using Depth Data", Proc. of the Sixth IEEE International conference on automatic Face and Gesture Recognition, pp. 529-534, 2004.
- [2] L. Bretzner, I. Laptev, and T. Lindeberg, "Hand Gesture using multi-scale color features, hierarchical models and particle filtering", Proc. of the Fifth International conference on Automatic Face and Gesture Recognition, pp. 423- 428, 2003.
- [3] V. Pavlovic, et al. visual interpretation of hand gesture for human-computer interaction: a review, IEEE Trans. On Pattern anal. Mach. Intel. 19(7), pp 677-695, 1997.
- [4] Attila Licsar and TamasSziranyi, "Supervised training based hand gesture recognition system", Proc. of the 16th International Conference on Pattern Recognition, Vol. 3, pp 30999 – 31003, 2002.
- [5] Chambers, Graeme S., Venkatesh, Svetha, West, GeoffA.W. and Bui, Hung H. 2004, Segmentation of intentional human gestures for sports video annotation, in *MMM 2004 :Proceedings of the 10th International Multimedia Modelling Conference*, IEEE Computer Society, Los Alamitos, Calif., pp. 124-129.