Automatic Total Cricket Score Generation

Mala M[1], Sanjay B E[2],Sathvik K V[3], Shaik Mohammed Usman[4] ,Shamanth C T[5]

[1] *Assistant Professor ,Computer Science Engineering , AMC Institute , karnataka, India*
[2] *Student, Computer Science Engineering , AMC Institute , karnataka, India*
[3] *Student, Computer Science Engineering , AMC Institute , karnataka, India*
[4] *Student, Computer Science Engineering , AMC Institute , karnataka, India*
[5] *Student, Computer Science Engineering , AMC Institute , karnataka, India*

ABSTRACT

Cricket is beyond the shadow of a doubt one of the most popular forms of sports in thesouthern region of Asia. This form of sports is widely played in more than 125 countries recognized by the International Cricket Council. With the flourishing of cricket, various aspects of the game are being automated with the advent of technology. Use of computer vision in assisting third umpire decision is indubitably the well-liked one. One of the most challenging issues that first initiates the discussion on its prosperity is the duration of the game. The onfield umpire has to authorize decisions almost after each delivery following which modifications arecarried out in the scorecard which is a very tedious process. The traditional approaches that have been considered so far involve wearing a specialized hand glove, which collides with thebeauty of the originality in the field. In this paper, an approach is proposed and a prototype is implemented to automate the umpire's decision by interpreting his hand gesture. The region ofinterest is selected using a Haarcascade-classifier and then the particular gesture is recognized using logistic regression. This process would eliminate the manual updating of scorecards and thereby reduce the game duration notably. In addition, it excludes the prerequisite of wearing special gloves involving sensors. The efficiency of the algorithm is then cross-checked with the training and test data.

Keyword :-*Tensor flow ,AI ML*

1. Introduction:

The authors built a constrained, lab-based Sign Language recognition system with the goal of making it a mobile assistive technology. They examine using multiple sensors for disambiguation of noisy data to improve recognition accuracy. The experiment compares the results of training a small gesture vocabulary using noisy vision data, accelerometer data and both data sets combined. The authors chose to use a rule–based grammar for sentence structure in the training and testing process. Speech recognition often uses statistical grammars for increased accuracy. These grammars are built by tying together phonemes (the simplest unit of speech) and training on the transition between the phonemes. The sets are usually done with bigrams (two phonemes tied together) or trigrams (three phonemes). Training using bigrams or trigrams requires considerably more data because representations of each transition of each word are now needed. In our case, the bigrams and trigrams would be built by tying together gestures. The current data set is too small to effectively train using bigrams or trigrams, but we intend to continue collecting data with the goal of implementing these techniques.

1.1 Problem Statement :

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 accidently 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 umpire's hands and palms.

1.2 Motivation:

The main motivation of our project is to reduce the time and human effort and improve the speed in the cricket scoreboard calculation. 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 as we humans are prone to error this technology come to their rescue. The manual processing is taking more time. It takes lots of time to record the process and transaction into a paper so this can help in this place. a lot of space is required to maintain the record physically. To solve the problem they are going for computerization.

2. LITERATURE SURVEY

The authors built a constrained, lab-based Sign Language recognition system with the goal of making it a mobile assistive technology. They examine using multiple sensors for disambiguation of noisy data to improve recognition accuracy. The experiment compares the results of training a small gesture vocabulary using noisy vision data, accelerometer data and both data sets combined. The authors chose to use a rule–based grammar for sentence structure in the training and testing process. Speech recognition often uses statistical grammars for increased accuracy. These grammars are built by tying together phonemes (the simplest unit of speech) and training on the transition between the phonemes. The sets are usually done with bigrams (two phonemes tied together) or trigrams (three phonemes). Training using bigrams or trigrams requires considerably more data because representations of each transition of each word are now needed. In our case, the bigrams and trigrams would be built by tying together gestures. The current data set is too small to effectively train using bigrams or trigrams, but we intend to continue collecting data with the goal of implementing these techniques.

Advantage:

- Benefit of the proposed design is that the user can monitor the camera's view via the head mounted display.
- Provides accuracy.

Disadvantage:

- Data set is too small to effectively train using bigrams or trigrams.
- The current system has only been trained on a very small vocabulary.

2.1 An approach to automate the scorecard in cricket with computer vision and machine learning:

The proposed method (**MD. Asif Shahjalal**) for recognizing cricket umpire gesture from real time video using logistic regression algorithm has basic two parts. These are: 1) Learning Phase, 2) Recognition phase. In the learning phase we have to give a lot of gesture images as training data for input. Each of the training image is resized to 20x20 pixel. This is done for fast and efficient computation. Then the resized images are converted to grayscale which is also for efficiency. Then all the images put into a matrix where each row of the matrix refers the pixel intensity of an image. The last column of the matrix represents the output (which gesture picture is in the corresponding image) of that image. Then using the matrix cost function is calculated and minimized by using gradient descent algorithm. For the lowest value of the cost function the corresponding parameter values are consider as weight. Now on the recognition phase we capture video from a static camera or any other source, then we separate the frames of the video. After separating to select the region of interest from the input images we use Haar-Cascade-Classifier. After selecting the region of interest, we convert it to a 20x20 pixel image, vectorize it and send the image feature value X and previously obtain weight to logistic regression hypothesis. The hypothesis gives the probability of a gesture to be true or false. The recognition step works in real time environment and it can provide output at a very decent amount of time.

3 Overall system design illustrating the steps for video summarization

Automatic video(Aravind Ravi and etl) summarization has gained increased attention in the recent past. Sports highlights generation, movie trailer generation, automatic headlines generation for news are some examples of video summarization. The focus of the present work is sports video summarization in the form of highlights. The

highlights of a game provide the summary of important events of that game such as a goal in soccer or a wicket in cricket. It is a challenging task to summarize the highlights from sports videos as these videos are unscripted in nature. An efficient approach can be based on identifying key events from the sports video and use them to automatically generate the highlights. Among sports, cricket is the most popular game in the world after soccer and has the highest viewership rating. In the game of cricket, the umpire is the person with the authority to make important decisions about events on the field. The umpire signals these events using hand signals, poses and gestures. This innate characteristic of the cricket video can be leveraged as one approach for solving the problem of cricket highlight generation. Therefore, a system can be developed to detect the unique signals and poses shown by the umpire to automatically generate cricket highlights. A method for umpire pose detection for generating cricket highlights based on transfer learning is proposed in this work.

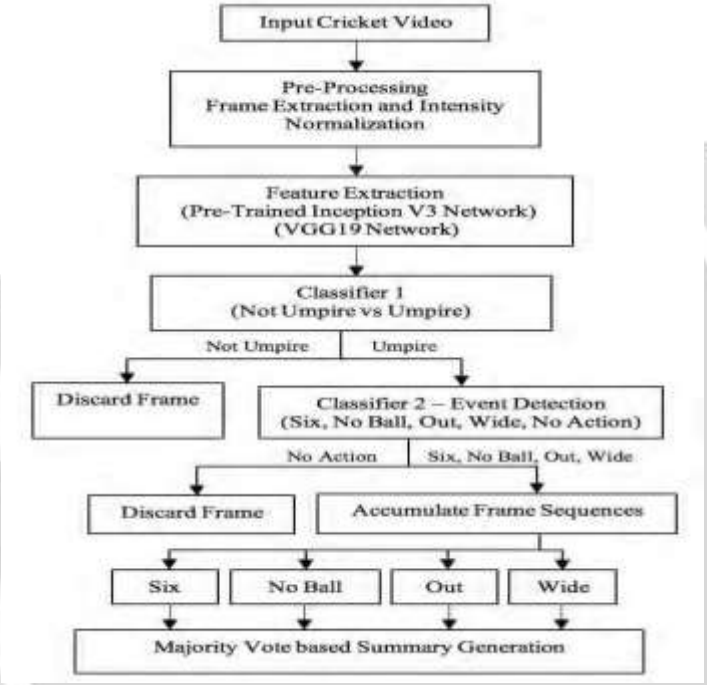

Fig. 3.1. Overall system design illustrating the steps for video summarization

4. Data flow

The DFD is straightforward graphical formalism that can be utilized to speak to a framework as far as the info information to the framework, different preparing did on this information and the yield information created by the framework. A DFD model uses an exceptionally predetermined number of primitive images to speak to the capacities performed by a framework and the information stream among the capacities.

The principle motivation behind why the DFD method is so famous is most likely in light of the way that DFD is an exceptionally basic formalism- It is easy to comprehend and utilization. Beginning with the arrangement of abnormal state works that a framework performs, a DFD display progressively speaks to different sub capacities. Actually, any various leveled model is easy to get it.

The human personality is such that it can without much of a stretch see any progressive model of a framework in light of the fact that in a various leveled model, beginning with an extremely straightforward and unique model of framework, distinctive points of interest of a framework are gradually presented through the diverse orders. A data-flow diagram (DFD) is a graphical representation of the "stream" of information through a data framework. DFDs can likewise be utilized for the perception of information handling

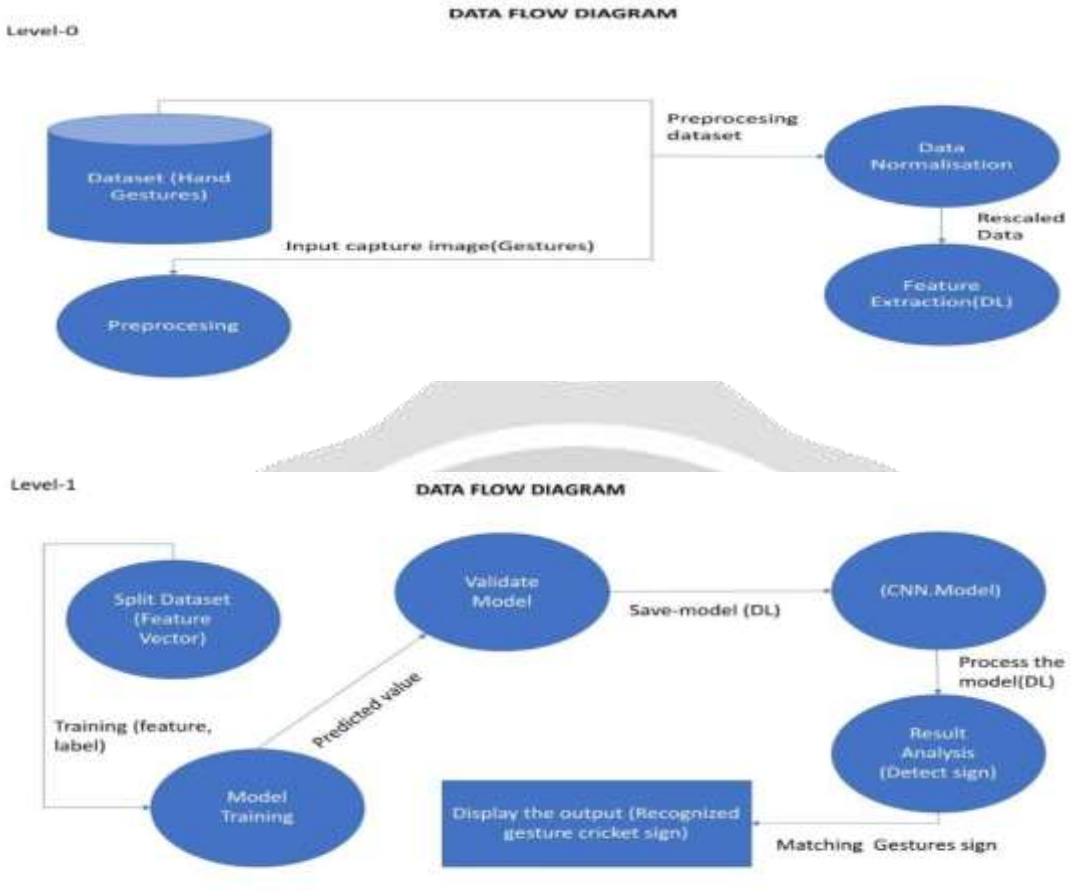

Fig.3.2 Data Flow Diagram

5 Software Requirements:

5.1 PYTHON

Python is an object-oriented programming language created by Guido Rossum in 1989.

It is ideally designed for rapid prototyping of complex applications. It has interfaces to many OS system calls and libraries and is extensible to C or C++. Many large companies use the Python programming language include NASA, Google, YouTube, BitTorrent, etc.

Python is widely used in Artificial Intelligence, Natural Language Generation, Neural Networks and other advanced fields of Computer Science. Python had deep focus on code readability & this class will teach you python from basics.

Characteristics of Python

- It provides rich data types and easier to read syntax than any other programming languages
- It is a platform independent scripted language with full access to operating system API's
- Compared to other programming languages, it allows more run-time flexibility
- It includes the basic text manipulation facilities of Perl and Awk

- A module in Python may have one or more classes and free functions
- Libraries in Pythons are cross-platform compatible with Linux, MacIntosh, and Windows

5.2 Jupyter Notebook:

The Jupyter Notebook is an incredibly powerful tool for interactively developing and presenting data science projects. This article will walk you through how to set up Jupyter Notebooks on your local machine and how to start using it to do data science projects.

First, though: what is a "notebook"? A notebook integrates code and its output into a single document that combines visualizations, narrative text, mathematical equations, and other rich media. This intuitive workflow promotes iterative and rapid development, making notebooks an increasingly popular choice at the heart of contemporary data science, analysis, and increasingly science at large.

Best of all, as part of the open source Project Jupyter, they are completely free. The Jupyter project is the successor to the earlier IPython Notebook, which was first published as a prototype in 2010. Although it is possible to use many different programming languages within Jupyter Notebooks, this article will focus on Python as it is the most common use case. (Among R users, R Studio tends to be a more popular choice).

To get the most out of this tutorial you should be familiar with programming, specifically

Python and pandas specifically. That said, if you have experience with another language, the Python in this article shouldn't be too cryptic, and will still help you get Jupyter Notebooks set up locally. Jupyter Notebooks can also act as a flexible platform for getting to grips with pandas and even Python, as will become apparent in this article.

We will:

- Cover the basics of installing Jupyter and creating your first notebook
- Delve deeper and learn all the important terminology
- Explore how easily notebooks can be shared and published online. Indeed, this article *is* a Jupyter Notebook! Everything here was written in the Jupyter Notebook environment, though you are viewing it in a read-only form.

6 Result

There are two types of results to the operation — one in which the convoluted feature is reduced in dimensionality as compared to the input, and the other in which the dimensionality is either increased or remains the same. This is done by applying Valid Padding or Same Padding in the case of the latter. In above example our padding is 1.

In our example when we augment the 5x5x1 image into a 7x7x1 image and then apply the 3x3x1 kernel over it, we find that the convoluted matrix turns out to be of dimensions 5x5x1. It means our output image is with same dimensions as our output image (Same Padding).

On the other hand, if we perform the same operation without padding, in the output we'll receive an image with reduced dimensions. So our (5x5x1) image will become (3x3x1).

 Feature Extraction: example:

Lets say we have a handwritten digit image like the one below. We want to extract out only the horizontal edges or lines from the image. We will use a filter or kernel which when convoluted with the original image dims out all those areas which do not have horizontal edges:

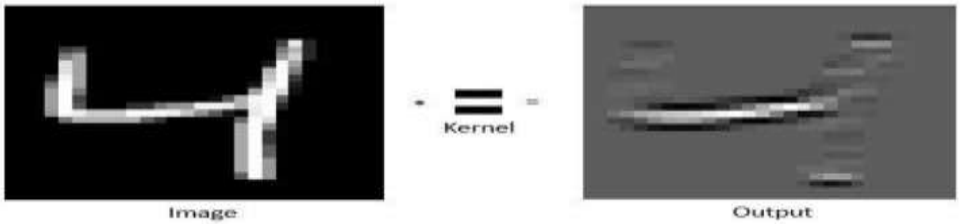

Fig.5.7: Horizontal filter example

Notice how the output image only has the horizontal white line and rest of the image is dimmed in Fidg 5.7 The kernel here is like a peephole which is a horizontal slit. Similarly for a vertical edge extractor the filter is like a vertical slit shown in Fig 5.8 peephole and the output would look like:

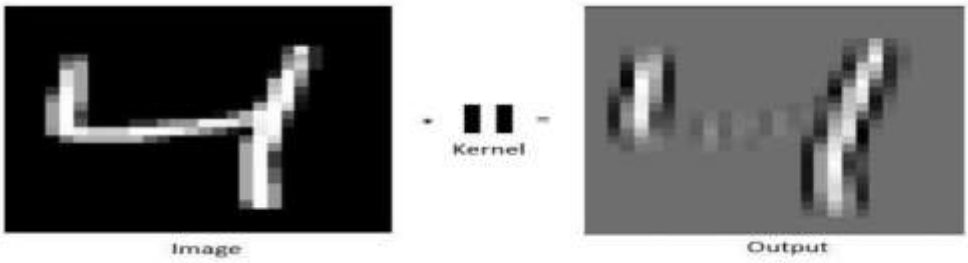

Fig.5.8: Vertical filter example

7. CONCLUSIONS:

We presented an approach to real-time alphabet sign language image classification using deep learning. Our proposed deep learning model based on CNN was able to perform with 98.3 % accuracy on our test dataset. Our work also included data gathering using a web camera to increase the dataset size to more than 500,0 RGB images making our prediction more robust. Our process was also capable of real-time prediction using image frames from a web camera with rates of 50 to 100 Hz.

8. REFERENCES:

1. Cricket Umpire Association srilankahttps://cricketumpiresassociation.blogsp ot.com/2009/10/cricket-umpires-association-ofsri.html.
2. Umpire(Cricket)https://en.wikipedia.org/wiki/U mpire_(cricket) .
3. "Match officials". International Cricket Council. Retrieved 8 December 2017.
4. Automation of Cricket Scoreboard by Recognizing Umpire Gestures . Vaishnavi K. Nair
5. An Approach to Automate the Scorecard in Cricket with Computer Vision and Machine Learning ,Md. Asif Shahjalal.
6. Block Diagram and Explanation of RF Transceivers.https://www.watelectronics.com/ rf-transceiver-module-with-block-diagramexplanation/
7. Smart Traffic Signals."https://www.ijsr.net/conf/RISE2017/I JSR31.pdf"

8. AutoSilence: Using Information and Communication Technology for Silence Zone. SumitBusa.
9. ECB ACO Archived 16 March 2015 at the Wayback Machine Education – find a course.
10. Hot Spot Technique in Cricket. Sandeep Sangwan.
11. Hari R and Wilscy M 2014 Event detection in cricket videos using intensity projection profile of Umpire gestures IEEE India Conference (INDICON) pp 1-6
12. Bhansali L and Narvekar M 2016 Gesture recognition to make umpire decisions International Journal of Computer Applications, vol. 148, no. 14