MATH-546 Sec 001 - Pattern Recognition

Project Report

An Image Recognition And Classifier Used For Sports Equipment(Specially used for Hitting)

Author-Niranjan Pawar Date- 12/20/2019

Abstract:- This mini projects uses images of sports equipment (to be specific- cricket bat, fencing sword, ice hockey stick, lawn tennis racket and a table tennis paddle.) for implementing a classifier using Matlab and Jmp. It uses image segmentation to first extract the object from the image and then extracting various features from those objects. These features will be used to train a classifier model and later used for testing other images of similar kind. Finally we use all the obtained data to find how efficient the classifier is.

Introduction:- Being a huge sports fan, motivated me to use select this data set for the project. This kind of model could be used to maintain an inventory of a sports shop, where photos of the shop would be enough to calculate the quantity of equipment present. These two were my primary motivation for selecting this data set of sports equipment. The Classifier is made to differentiate an object(image) into one of the five classes of sport equipment. These classes are cricket bat, fencing sword, ice hockey stick, lawn tennis racket and table tennis paddle.

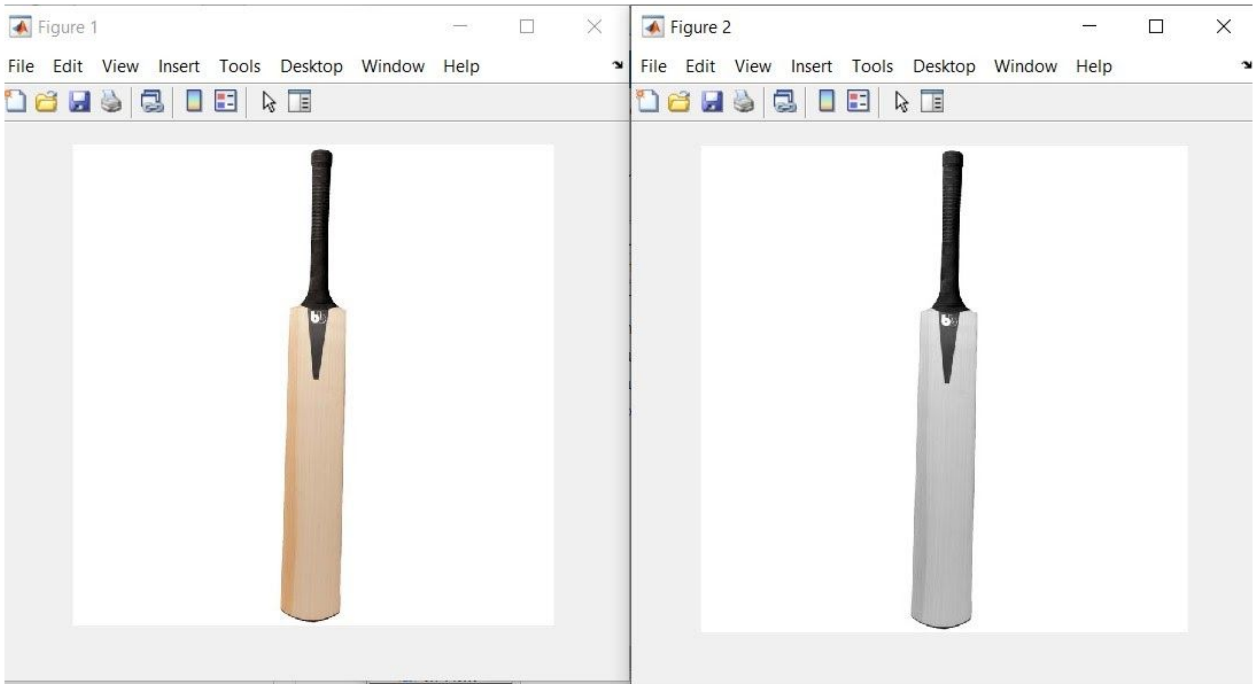
Firstly, the initial step was collection of data, which in this means collecting images of these five classes. After which all the images are to be processed such that we could extract valuable information from it. For doing that, we first segment the images into background and the front ground/object. Later, various properties of the object are chosen and extracted such that this properties can differentiate between the objects as good as possible. The process done until now is done by using Matlab. After that we used JMP for statistical manipulation of extracted features/data. All the obtained data is fetched to the partition model which efficiently uses the data to obtain a decision tree. This decision tree has only children nodes at any level. So the questions asked at each level are either yes or no. We use this cut points to evaluate the images for testing data. After testing, comes the evaluation part, where we calculate the accuracy, precision, sensitivity etc. for the classifier.

Theory:- The order of topics in theory will be based on when they were used in the process of this project.

First we start the project with downloading and installing Matlab which will be used for around 60% of this project.

MATLAB is a multi-paradigm numerical computing environment and proprietary programming language developed by MathWorks. MATLAB allows matrix manipulations, plotting of functions and data, implementation of algorithms, creation of user interfaces, and interfacing with programs written in other languages.

I = rgb2gray(RGB) converts the truecolor image RGB to grayscale image I. The rgb2gray function converts RGB images to grayscale by eliminating the hue and saturation information while retaining the luminance. If you have Parallel Computing Toolbox installed, rgb2gray can perform this conversion on a GPU.



Fig(1). The first image is the original image whereas the second one is result of using rgb2gray.

<u>Thresholding Image</u>:- Thresholding is a non-linear operation that converts a gray-scale image into a binary image where the two levels are assigned to pixels that are below or above the specified threshold value. A grayscale image has value from 0 to 255. After thresholding the image is converted into a binary image with only two values of either 0 or 255.

In this project, we used the following method to threshold.

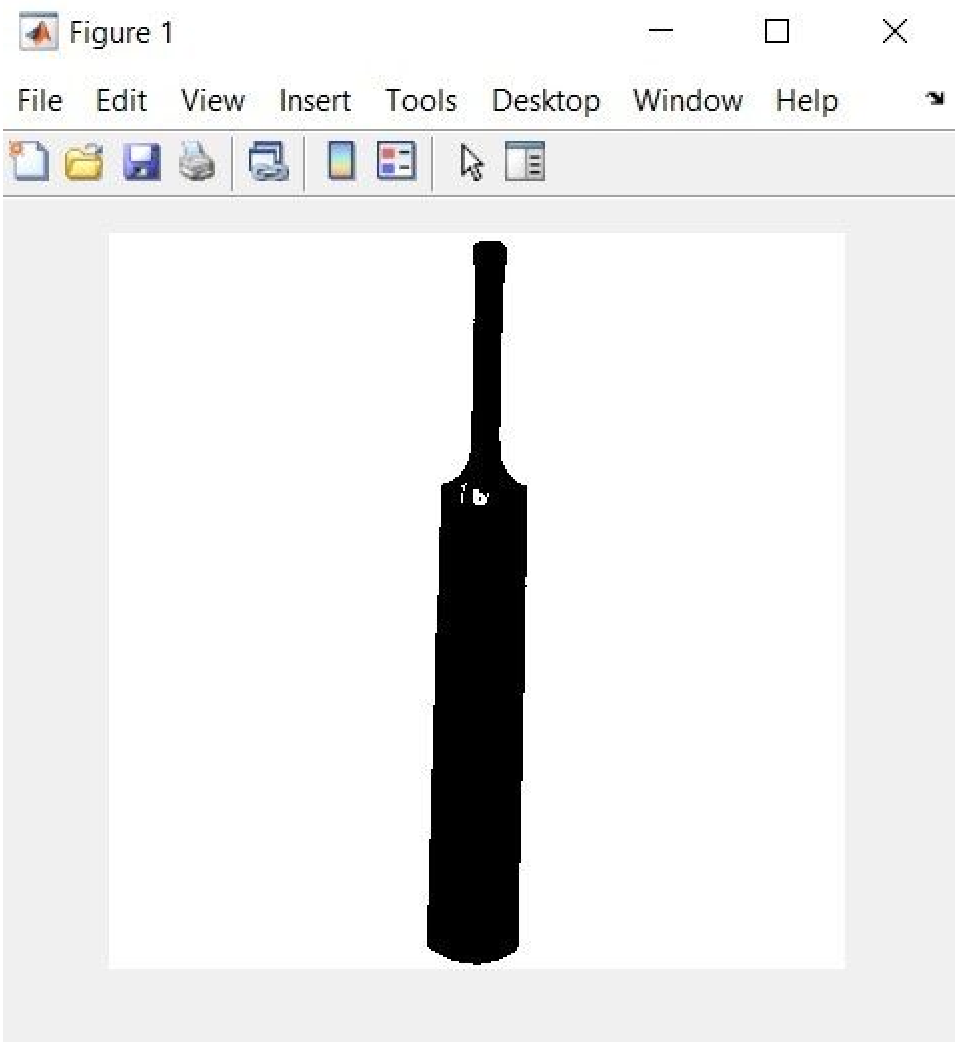
upper = 230;

lower = 230;

i2(i2 >= upper) = 255;

i2(i2 <= lower) = 0;

Here we have initialized values for upper and lower and using that as a reference we equate anything larger to 255 and anything smaller to 0.

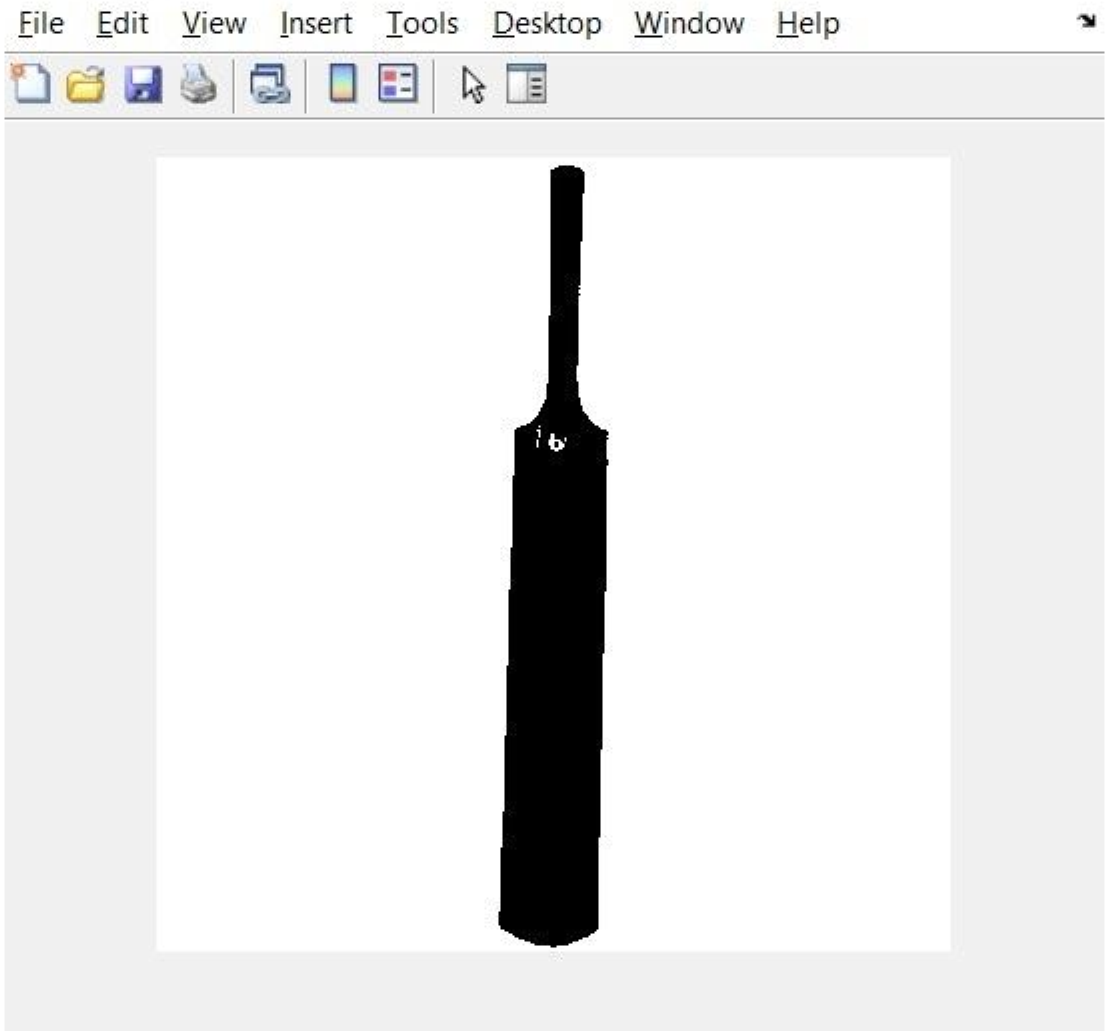


Fig(2).Image after thresholding.

bwmorph

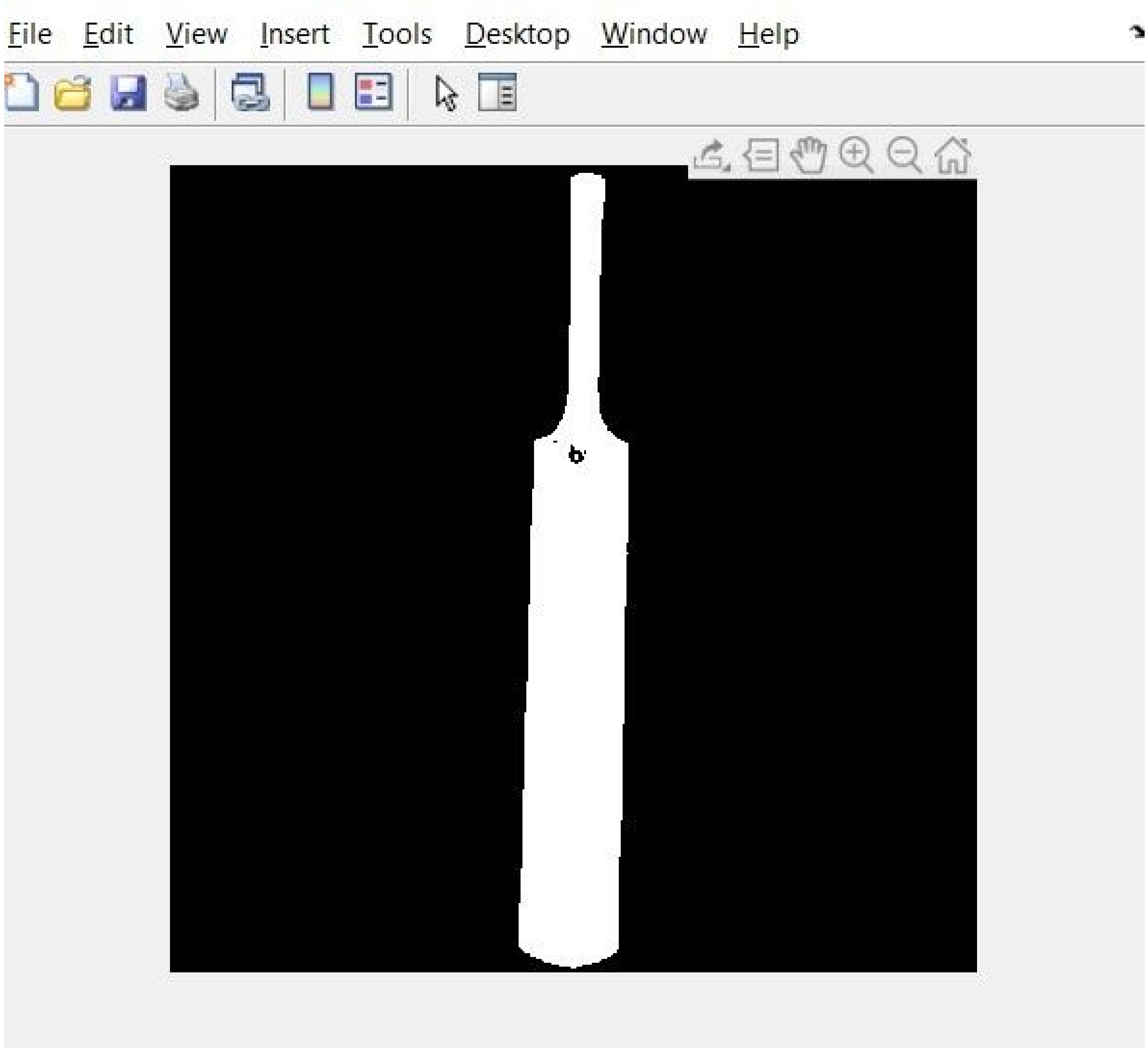
Removes isolated pixels (individual 1s that are surrounded by 0s), such as the center pixel in this pattern.

0 0 0
0 1 0
0 0 0



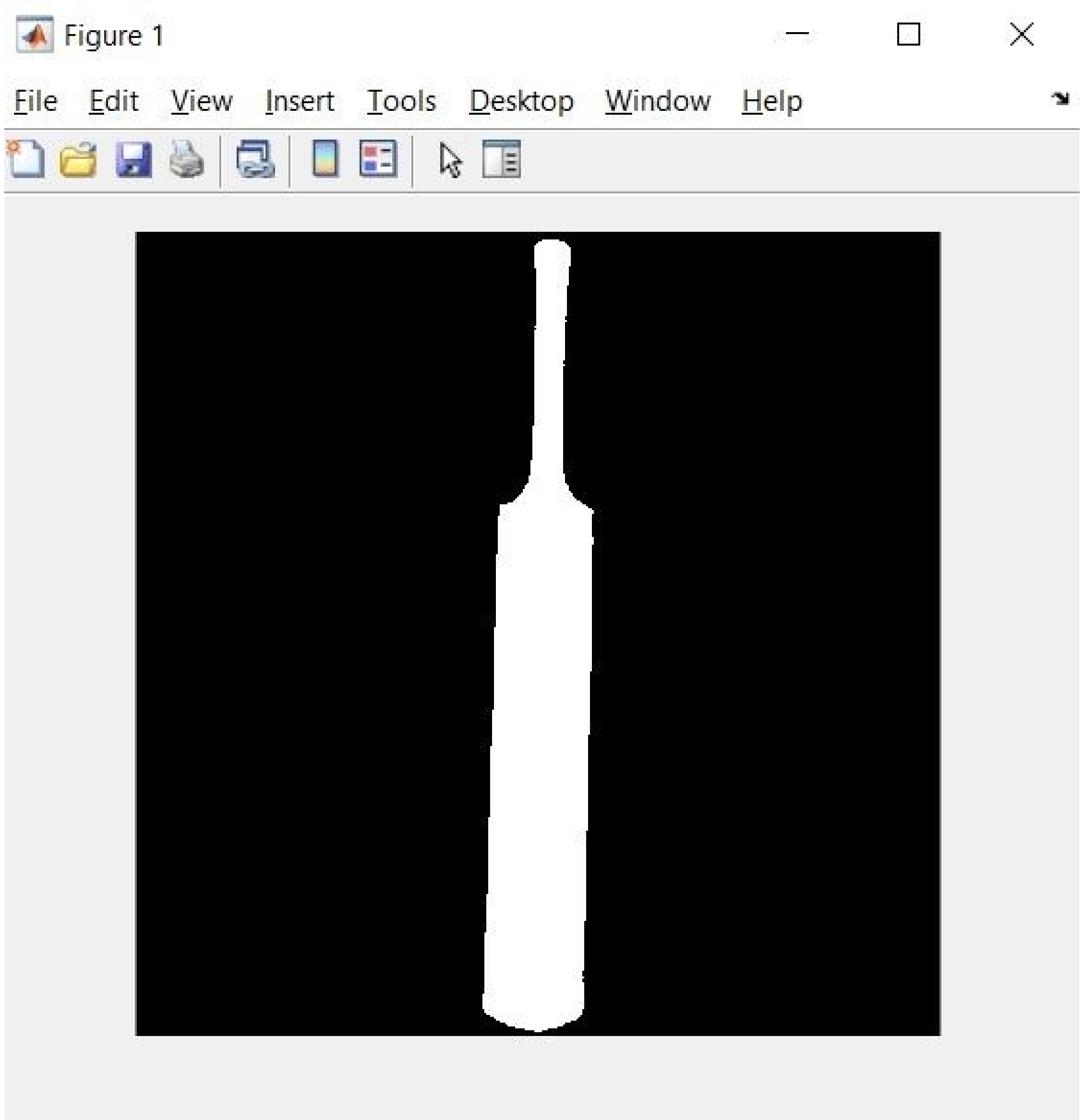
Fig(3). It cleaned small minute impurities, even though it is not visible in this image.

J = imcomplement(I) computes the complement of the image I and returns the result in J.



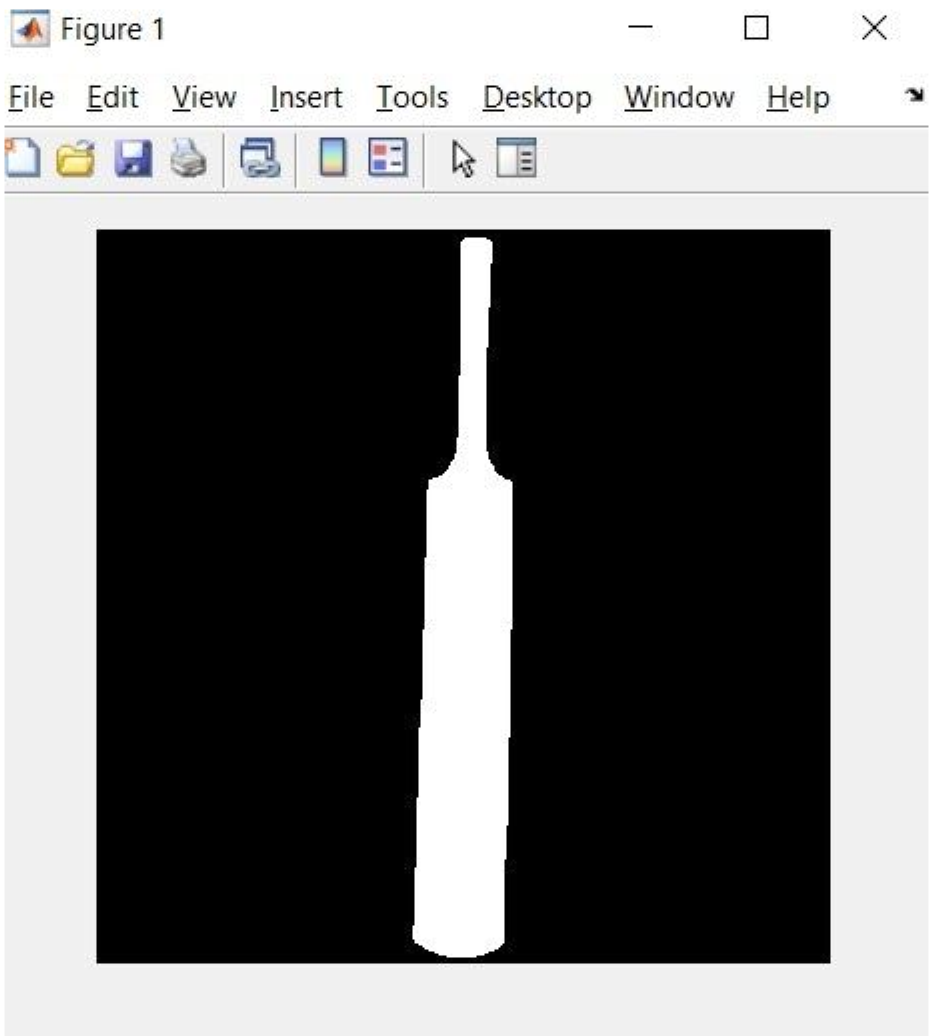
Fig(4). Imcomplement of fig3.

BW2 = imfill(BW,locations) performs a flood-fill operation on background pixels of the input binary image BW, starting from the points specified in locations, whereas BW2 = imfill(BW,'holes') fills holes in the input binary image BW. In this syntax, a hole is a set of background pixels that cannot be reached by filling in the background from the edge of the image.



Fig(5). Result after using imfill.

J = imopen(I,SE) performs morphological opening on the grayscale or binary image I, returning the opened image, J. SE is a single structuring element object returned by the strel or offsetstrel functions. The morphological open operation is an erosion followed by a dilation, using the same structuring element for both operations.



Fig(6). Use of imopen, we can not see tangible changes from fig5. But it is really in some other cases.

stats = regionprops(BW,properties) returns measurements for the set of properties specified by properties for each 8-connected component (object) in the binary image, BW. stats is struct array containing a struct for each object in the image.
Following are the properties used in this project.

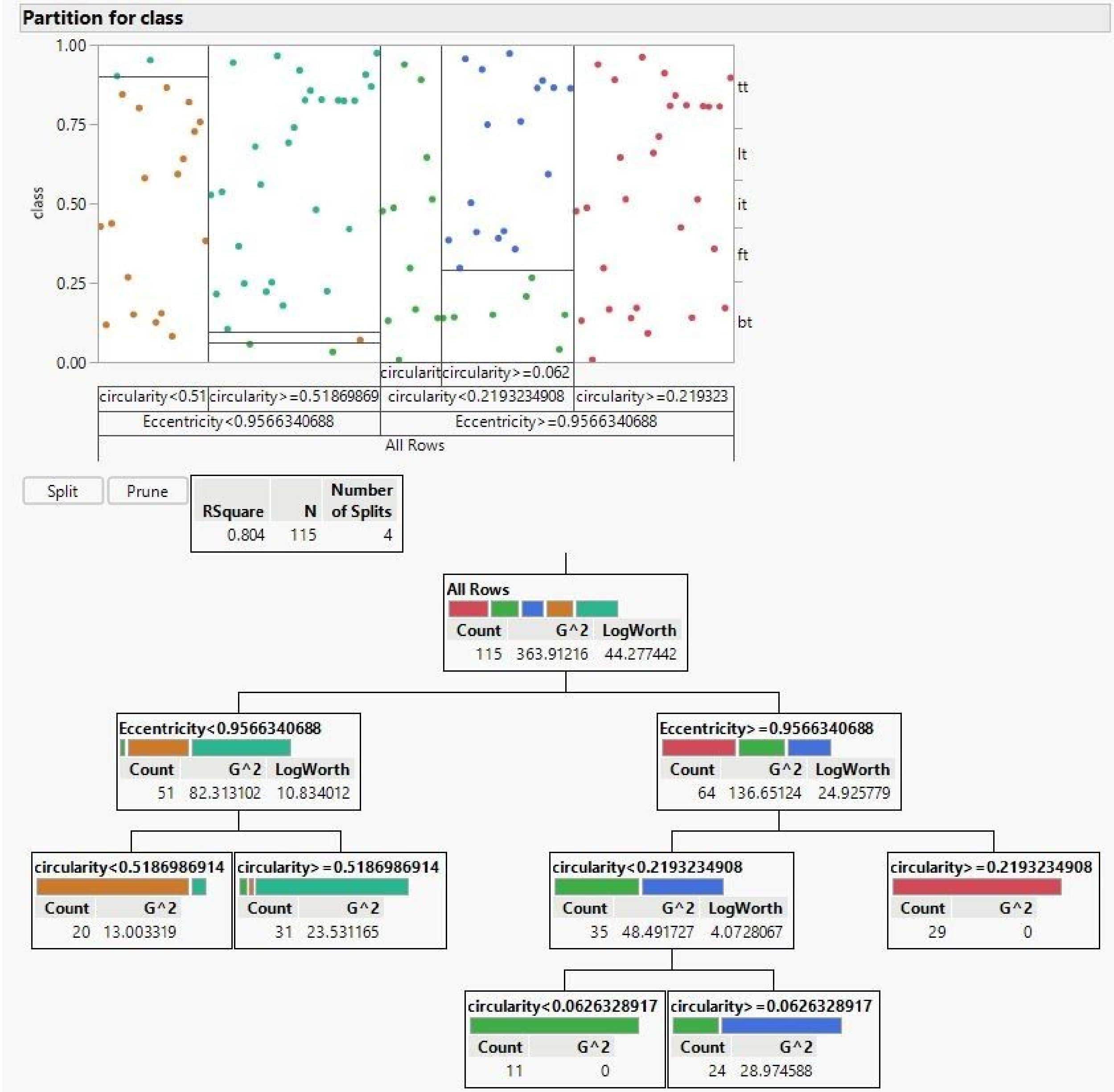
- Area-Actual number of pixels in the region, returned as a scalar. (This value might differ slightly from the value returned by bwarea, which weights different patterns of pixels differently.)
- Circularity-Circularity that specifies the roundness of objects, returned as a struct with field Circularity. The struct contains the circularity value for each

object in the input image. The circularity value is computed as (4*Area*pi)/(Perimeter2). For a perfect circle, the circularity value is 1. The input must be a label matrix or binary image with contiguous regions. If the image contains discontiguous regions, regionprops returns unexpected results. Circularity is not recommended for very small objects such as a 3*3 square. For such cases the results might exceed the circularity value for a perfect circle which is 1.

- Eccentricity-Eccentricity of the ellipse that has the same second-moments as the region, returned as a scalar. The eccentricity is the ratio of the distance between the foci of the ellipse and its major axis length. The value is between 0 and 1. (0 and 1 are degenerate cases. An ellipse whose eccentricity is 0 is actually a circle, while an ellipse whose eccentricity is 1 is a line segment.)
- Extent-Ratio of pixels in the region to pixels in the total bounding box, returned as a scalar. Computed as the Area divided by the area of the bounding box.

JMP (pronounced "jump") is a suite of computer programs for statistical analysis developed by the JMP business unit of SAS Institute.JMP is used in applications such as Six Sigma, quality control, and engineering, design of experiments, as well as for research in science, engineering, and social sciences.The software is focused on exploratory visual analytics, where users investigate and explore data. These explorations can also be verified by hypothesis testing, data mining, or other analytic methods. In addition, discoveries made through graphical exploration can lead to a designed experiment that can be both designed and analyzed with JMP.

The Partition platform recursively partitions data according to a relationship between the predictors and response values, creating a decision tree. The partition algorithm searches all possible splits of predictors to best predict the response. These splits (or partitions) of the data are done recursively to form a tree of decision rules. The splits continue until the desired fit is reached. The partition algorithm chooses optimum splits from a large number of possible splits, making it a powerful modeling, and data discovery tool.



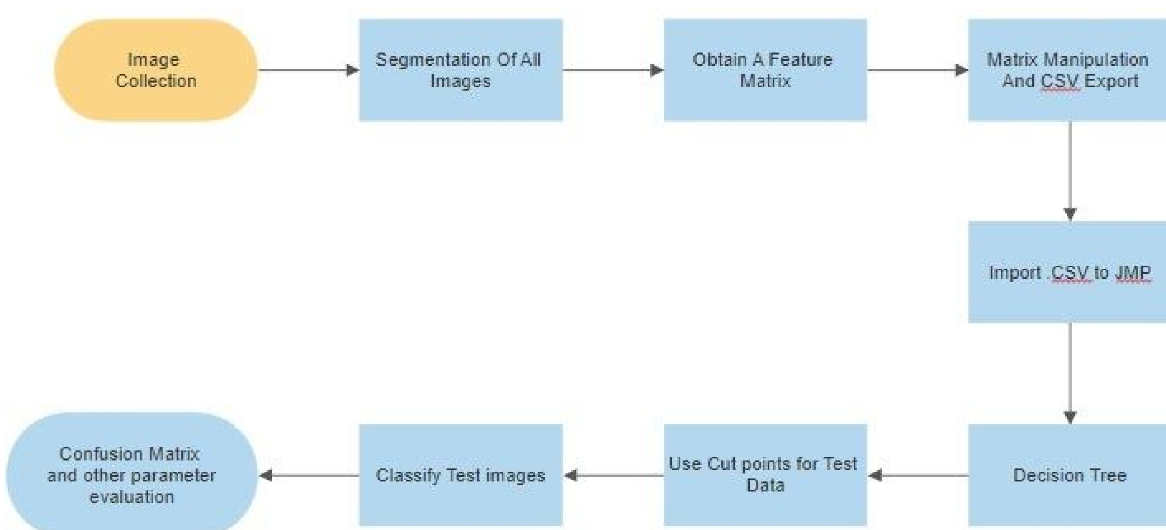
Fig(7). Example of decision tree obtained from Partition model.

A <u>confusion matrix</u> is a table that is often used to describe the performance of a classification model (or "classifier") on a set of test data for which the true values are known. The confusion matrix itself is relatively simple to understand, but the related terminology can be confusing.

The best accuracy is 1.0, whereas the worst is 0.0. It can also be calculated by 1 – ERR. Accuracy is calculated as the total number of two correct predictions (TP + TN) divided by the total number of a dataset (P + N).

Precision refers to the closeness of two or more measurements to each other. Precision (also called positive predictive value) is the fraction of relevant instances among the retrieved instances.
Sensitivity (Recall or True positive rate)
It is also called recall (REC) or true positive rate (TPR). The best sensitivity is 1.0, whereas the worst is 0.0. Sensitivity is calculated as the number of correct positive predictions (TP) divided by the total number of positives (P).
Specificity (True negative rate)
The best specificity is 1.0, whereas the worst is 0.0. Specificity is calculated as the number of correct negative predictions (TN) divided by the total number of negatives (N).

Methodology:-



1. Collection Of Images:- All the images for testing and training are downloaded from the internet. A total of 115(training data) and 50(testing data) images have been used. Mostly the images used have similar backgrounds.

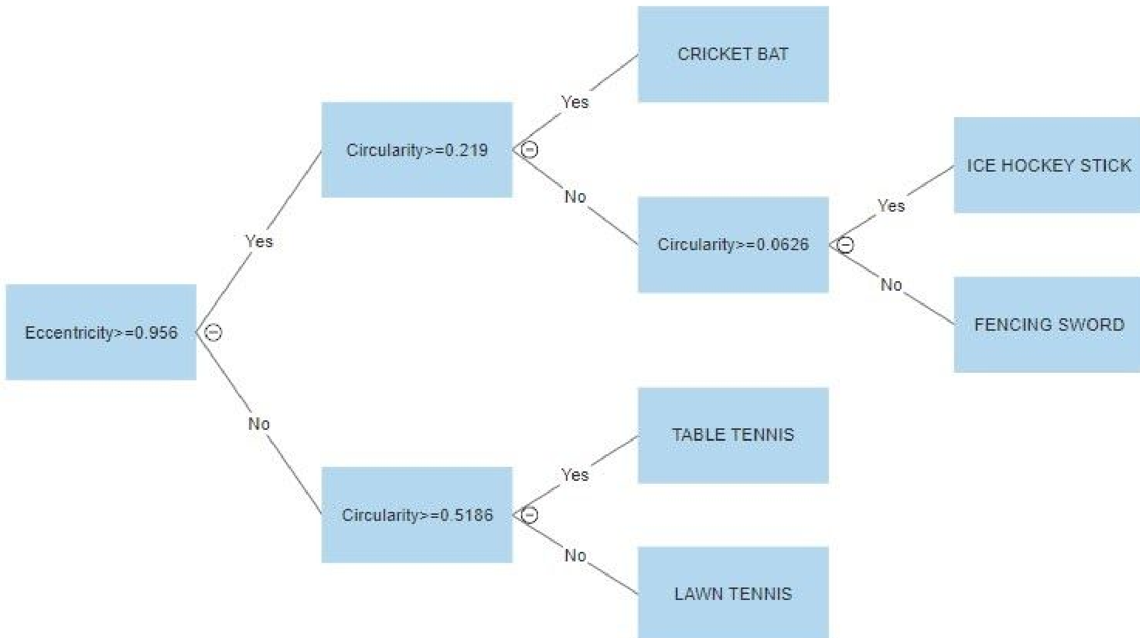
2. Segmentation:- All the images are segmented at once using a for loop and one segmentation method is used for all the images. As the number of data increases, it gets impossible to segment each and every image individually. Segmentation is done using the methods described in the theory part of this report.

3. Feature Extraction:- Using Regionprops, all the wanted properties are placed in a matrix.

4. CSV export and manipulation:- using " writematrix(data,'data.csv')" command, we download the feature matrix as .csv file. But it needs a bit of manipulation before it can be imported to JMP. All the columns have to be named as per the property so that is becomes easier to manipulate in JMP. Also as the part of training, there has to added a new column which tells which class that data entry belongs to.

5. CSV import to JMP:- Open JMP and under File, there is Import Multiple Files.

6. Decision Tree:- We use partition model to obtain a decision tree from the data. Following is the decision tree obtained:-



Fig(9). Decision tree obtained using Partition Model on extracted features.

7. <u>Testing Data</u>:- Using the cut points obtained from the above decision tree, we test data on the test data images. We process the images through the same segmentation methods and feature extraction after which they are measured with the cutpoints data to classify the object.

8. <u>Classify Images</u>:- Now the model is ready to recognize the images and classify to its classes accordingly.

9. <u>Confusion Matrix and Other parameters</u>:-

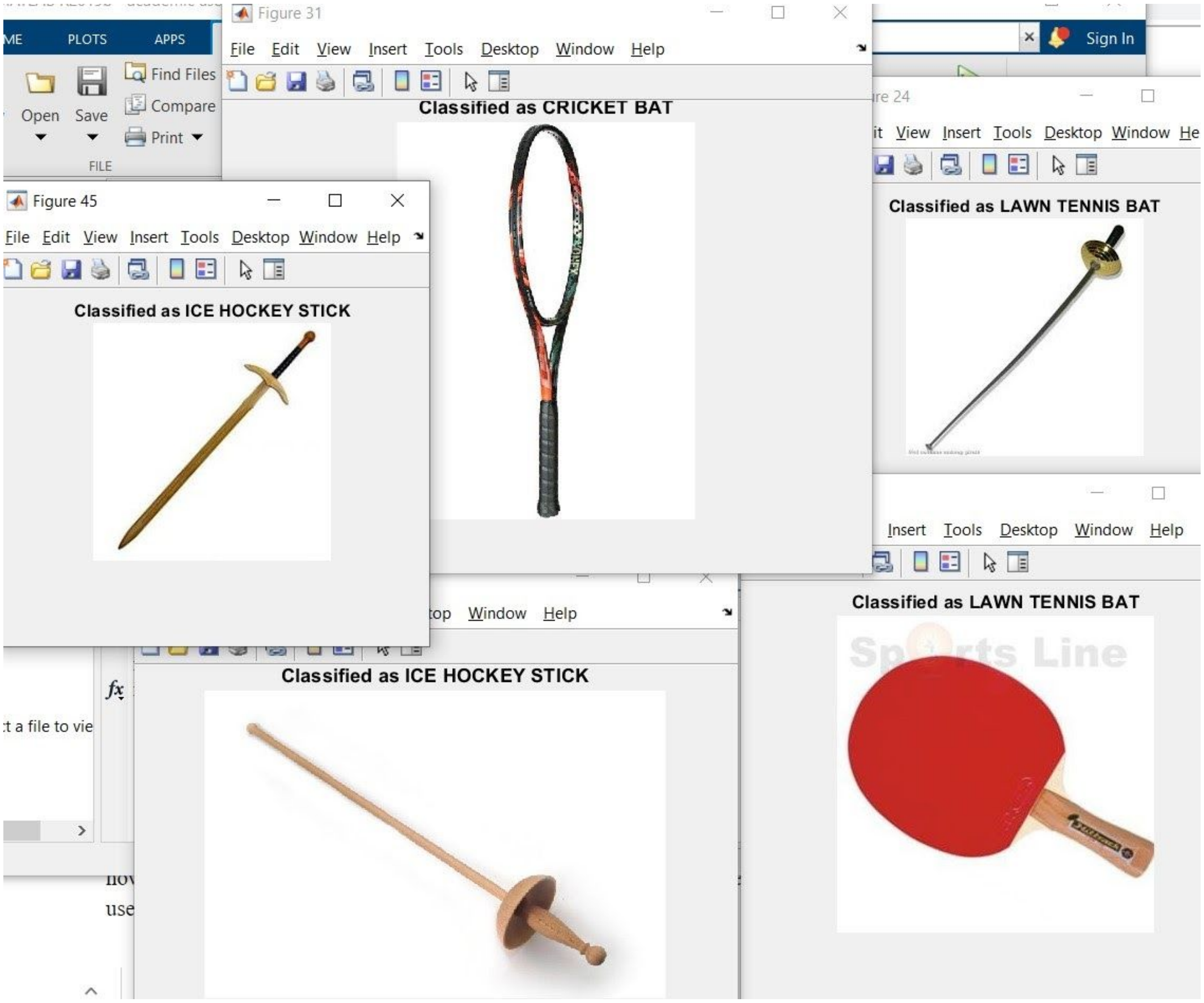
	CRICKET BAT	FENCING SWORD	ICE HOCKEY STICK	LAWN TENNIS	TABLE TENNIS
CRICKET BAT	12	0	0	0	0
FENCING SWORD	0	5	2	1	0
ICE HOCKEY STICK	0	0	10	0	0
LAWN TENNIS	1	0	0	10	0
TABLE TENNIS	0	0	0	2	8

Accuracy= 0.9

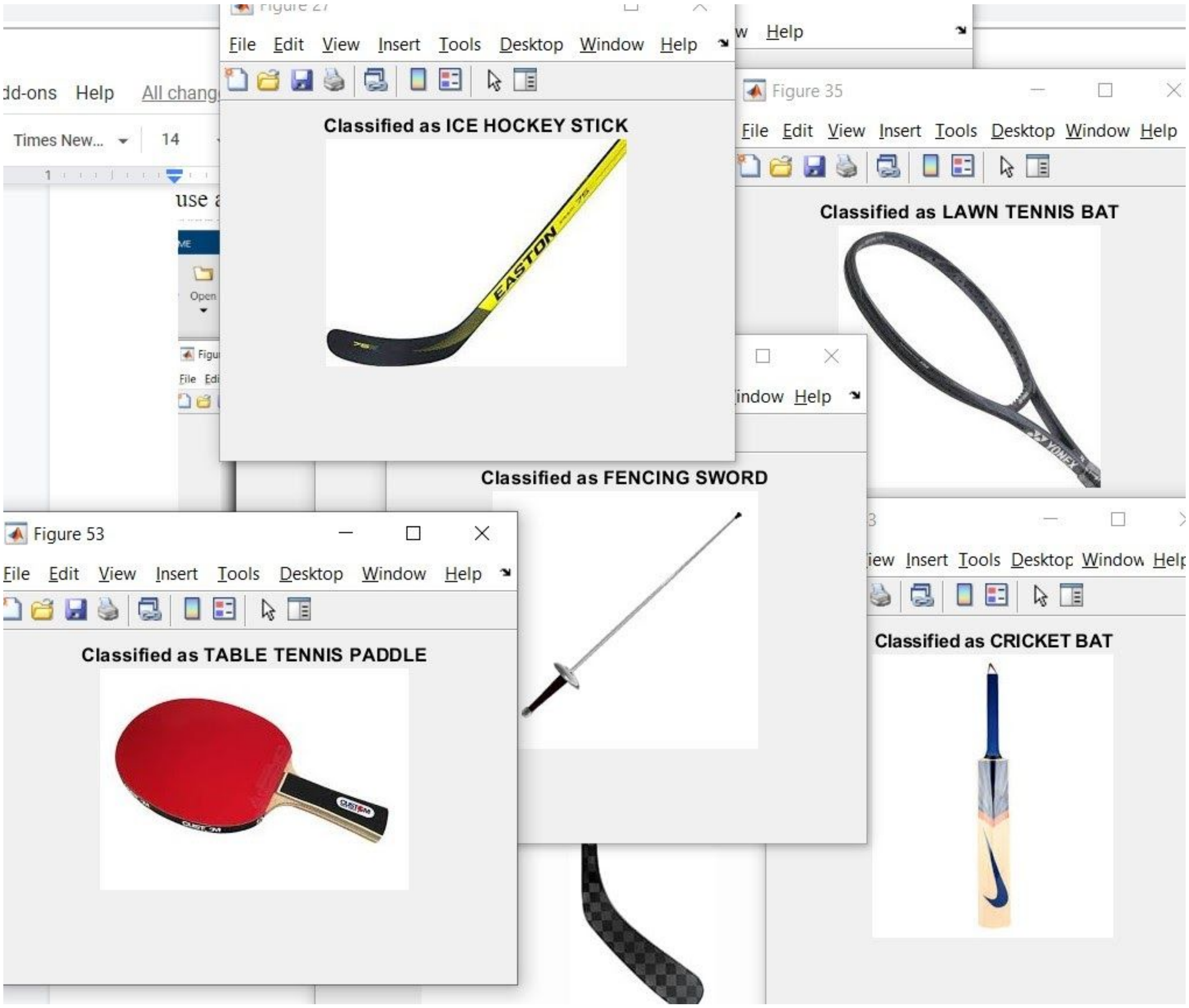
Precision CricketBat=0.923
Precision Fencing =1
Precision IceHockey =0.833
Precision LawnTennis=0.769
Precision TableTennis=1

Sensitivity CricketBat=1
Sensitivity Fencing =0.625
Sensitivity IceHockey =1
Sensitivity LawnTennis=0.909
Sensitivity TableTennis=0.8

Conclusion:- We made a project that classify images of five types quite efficiently with an accuracy of 0.9, in the process of which we learnt image segmentation, how and which features to choose based on the selected classes, we learnt how to use jmp.



Fig(10). Images that were classified in the wrong class.



Fig(11). Images of correctly classified objects.

Difficulties Faced:-

1. Hard to find a segmentation technique that could suit all the classes of images and results could have been improved if images could have been segmented individually.
2. Due to one common segmentation method used, we were unable to use euler's number as a property to differentiate between lawn tennis racket and table tennis paddle.
3. Decision Tree was different than what I expected and also it uses only the eccentricity and circularity as it properties rather than what I thought would use area and extent as well.

4. Wooden swords seems to be wrongly classified as ice hockey sticks.

Potential Improvements:-

1. Even though I added more images (if compared during presentation), the addition of extra images would always help.
2. The difficulty of this project could be improved by choosing images different backgrounds.
3. Addition of Euler's Number as feature of this images.
4. Classifying multiple objects in the same image.

References:-

1. Book:-Pattern Recognition and classification-Geoff Dougherty.
2. Notes:- Maths546 notes.
3. Youtube
4. Wikipedia
5. https://www.mathworks.com/solutions.html?s_tid=gn_sol