Detection and Recognition of Soccer Ball and Players

Thesis submitted in partial fulfilment of the requirements for the degree of

Master of Technology

in

Electronics and Communication Engineering

(Specialization: *Electronics and Instrumentation*)

by

Upendra Rao Moyyila
Roll No: 213EC3223

Department of Electronics & Communication Engineering
National Institute of Technology Rourkela
Rourkela, Odisha-769008
May2015

Detection and Recognition of Soccer Ball and Players

Thesis submitted in partial fulfilment of the requirements for the degree of

Master of Technology

in

Electronics and Communication Engineering

(Specialization: *Electronics and Instrumentation*)

by

Upendra Rao Moyyila

Roll No: 213EC3223

Under the Supervision of

Prof. Umesh Chandra Pati

Department of Electronics & Communication Engineering
National Institute of Technology Rourkela
Rourkela, Odisha-769008
May2015

DEDICATED TO MY TEACHERS,

FAMILY AND FRIENDS

Department of Electronics & Communication Engineering
National Institute of Technology, Rourkela

CERTIFICATE

This is to certify that the Thesis Report entitled — **"Detection and Recognition of Soccer Ball and Players"** submitted by **Mr. Upendra Rao Moyyila** bearing roll no. **213EC3223** in partial fulfilment of the requirements for the award of Master of Technology in Electronics and Communication Engineering with specialization in **"Electronics and Instrumentation Engineering"** during session 2013-2015 at National Institute of Technology, Rourkela is authentic work carried out by him under my supervision and guidance.

To the best of my knowledge, the matter embodied in the thesis has not been submitted to any other University / Institute for the award of any Degree or Diploma.

Prof. Umesh Chandra Pati

Place: Associate Professor

Date: Dept. of Electronics and Comm. Engineering

National Institute of Technology

Rourkela-769008

ACKNOWLEDGEMENT

I am greatly indebted to my supervisor **Dr. Umesh Chandra Pati** for his well regard guidance in selection and completion of the project. It is a great privilege to work under his guidance. I express my sincere gratitude for providing pleasant working environment with necessary facilities.

I would also like to thank to the Department of Science and Technology, Ministry of Science and Technology, Government of India for their financial support to setup **Virtual and Intelligent Instrumentation Laboratory** in Department of Electronics and Communication Engineering, National Institute of Technology, Rourkela under Fund for Improvement of S&T Infrastructure in Universities and Higher Education Institutions (FIST) program in which the experimentation has been carried out.

My lab mates **Satish Bhati and Pratish Kumar Sahoo** made this task more interesting by discussing various topics and suggesting new ideas. Also, I would thank all my mates of the specialization of Electronics and Communication Engineering for their constant support. I am grateful to **Mr. Sourabh Paul** for his encouragement.

Most importantly, I would thank my mother, relatives and friends for always being on my side and supporting me with their blessings.

DATE:
PLACE:

ROLL NO: 213EC3223
Dept. of ECE
NIT, Rourkela

ABSTRACT

In this work, an efficient, mere algorithm for detection and recognition of a soccer ball and players has been proposed. It actually involves two stages, detection followed by recognition. In long shot frames, existing algorithms will most likely be unable to identify the ball when it converged with the lines in the field. In this way, we present a strategy that can separate lines from the ball and section the ball capably. First, the ground (background) is disposed of from the scene and the edges of the original frame are distinguished. At that point, Hough line transform is used identify lines. After line identification, the lines are wiped out from the original frame so that just detected players and the ball remain in the scene. Now moving recognition of player’s.

In this work, approach to recognition of players is based on player jersey number. Jersey numbers are stored in database. Jersey number is the base image and the original frame will be the test image. SURF (Speeded-Up Robust Feature) Features in both the base image and test images are found. Then, Features (Feature descriptors) are extracted and match features accordingly. After features getting matched, geometric transform is estimated using Affine Transform. After estimating, the original base image has been recovered, and structural similarity has been checked to avoid false interpretation. If the structural similarity index value is above the defined limit, then the corresponding player statistics will be displayed. Higher the similarity index value, lesser will the probability for false interpretation.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF ACRONYMS

SIFT	: Scale Invariant Feature Transform
SURF	: Speeded-Up Robust Features
LoG	: Laplacian of Gradients
MSER	: Maximally Stable Extremal Regions
SUSAN	: Smallest Univalue Segment Assimilating Nucleus
FAST	: Features from Accelerated Segment Test
HT	: Hough Transform
SE	: Structuring Element
SSIM	: Structural Similarity
PCA	: Principal Component Analysis
ICA	: Independent Component Analysis
LDA	: Linear Discriminant Analysis

Chapter 1

Introduction

INTRODUCTION

In the field of machine vision, sports feature investigation from the view of computer vision has pulled in much consideration. Moving object detection is connected in diverse application ranges, for example, therapeutic image analysis, human tracking, activity framework checking and games feature dissecting. These days, sports feature is one exceptionally prominent exploration zone, which includes players' development investigation, strategies, arbitrator's choices, etc... Through detection and recognition of the moving articles (players, ball), a few abnormal high level observation should be possible, e.g. highlight extraction, occasion identification and strategy investigation.

The challenges associated with ball detection can be attributed to the following factors:

- when the ball moves fast, it may appear as long blur strip;
- The ball's candidates like size, color, velocity, shape, etc... changes over frames;
- Sometimes, the ball may merged with lines, sometimes occluded by players
- Sometimes ball appears like a little white blob; Playfield appearance fluctuates from spot to place and time to time, and can't be demonstrated correctly;
- many similar objects such as the ground lines and socks of the player are seen in the scene

Automatic player recognition would be of awesome significance in soccer feature recovery or in giving enlarged data about player. Jersey number in soccer feature is exploited to perform player acknowledgment. By watching Jersey samples, it is identified that large pixels in number and Jersey number significantly vary in their saturation, i.e. number of Jersey pixels lie in Achromatic scope of HSV color space. So jersey features will be detected using SURF, jersey features will be compared with in a real-time frames for detection and recognition of players.

1.1. Motivation

Segmentation of background from a scene is achievable. But, it is challenging issue to separate players and ball from the remaining objects in the scene. Sometimes the ball may not be able to get identified if it is merged with the line. Also, the players in occluded situation may not get identified in colloidal situations. While recognizing players, there is a chance of misinterpretation if the player is slightly sheared. Presently algorithms requires much computation and execution time.

Edges can be recognized effortlessly in the images. Consequently, utilizing these edges a few highlights can be acquired by which we can fulfill identification of players and ball. These highlights has an advantage of less time consuming, But has few limits like low accuracy.

In the process of feature based matching for player recognition, SIFT (Scale Invariant Feature Transform) or SURF (Speeded-Up Robust Features) have much importance. In real time applications SURF has better utility than SIFT because of execution time is 3-5 times faster when compared.

1.2. Objective

To develop a fast and efficient algorithm that separate lines from the ball, separate the ball and detect the players in the occluded situations.

The next objective is to develop an algorithm which uses SURF features for recognition of players and display player strategies.

1.3. Thesis organization

There are five chapters included in the thesis including first chapter which is about introduction.

Chapter 2: Literature Review

In second chapter, all the literature survey carried out in my project has been discussed.

Chapter 3: Detection of Soccer ball and players

In this chapter background of the work is discussed. Background includes some theory about segmentation, edge detection, line detection, morphological processing. The proposed algorithm is presented and explained briefly with results.

Chapter 4: Recognition of Players

In this chapter some essential features, feature detection, feature matching techniques are discussed. SURF Features are briefly explained. The proposed algorithm is presented and reviewed briefly with results.

Chapter 5: Conclusion:

In this chapter summary of project is described and concluded with future scope and improvements needed.

CHAPTER 2

LITERATURE REVIEW

LITERATURE REVIEW

The literature survey that has been carried out in this work is described as explained briefly as follows:

Xiao-Feng Tong, Han-Qing Lu, and Qing-Shan Liu used the color, shape, and size the ball for segmentation [1]. X. Yu, C. Xu, H. W. Leong, and Changsheng Xu proposed a trajectory based algorithm for estimating the ball candidates in each frame based on size [2]. T. D'Orazio and C. Guaragnella proposed a method that uses circle Hough transform for detecting ball [3]. Jong-Yun Kim, Tae-Yong Kim proposed Density-based approach [4] to separate the players and the lines. O. Utsumi, K Miura, I. Ide, S. Sakai, H. Tanaka proposed Color-based elimination of ground approach is used for identification of objects in a video [5].M. M. Naushad Ali, M. Abdullah-AlWadud and Seok-Lyong Lee proposed an efficient ball detection algorithm [6]. L. A. F. Fernandes and M. M. Oliveira explained Hough transform voting scheme [7].

K. Jung, K. I. Kim and A. K. Jain presented a survey of image and video text information extraction [8]. Q. Ye, Q. Huang and S. Jang proposed an algorithm where learning vector quantization algorithm is used for player jersey number segmentation that segments jersey number from background [9]. Size and pipe-like attributes of digital characters are used to filter the candidates.

K-NN classifier and Zernike moment features are employed for number recognition. M. Bertini, A. D. Bimbo and W. Nunziati proposed an algorithm [10] where player numbers are recognized using algorithm proposed by P. Viola and M. Jones [11]. The algorithm is basically designed for face recognition. M. Bertini, A. D. Bimbo and W. Nunziati proposed an algorithm where Harris detector and maximally stable regions are used for number detection [12]. E. L. Andrade Neto, E. Khan, J. C. Woods and M. Ghanbari proposed an algorithm that uses concept of picture trees and region adjacency for performing object searching. Region analysis is further utilized to isolate player number from candidates region [13].A player jersey number is treated as texture information. So, K. Jung, K. I. Kim and A. K. Jain proposed an algorithm for player number extraction [14]. Andrade, E.L., Khan, E., Woods, J.C., Ghanbari proposed easy way of extracting player jersey number [15]. Frejlichowski, D., Wierzba, P proposed automatic player face identification algorithm [16].

Chapter 3

Detection of Soccer Ball and Players

DETECTION OF SOCCER BALL AND PLAYERS

Background theory

3.1. Region

A region is defined as gathering of pixels connected with homogenous properties. While interpreting an image, region is principal key as it corresponds to an object or in the scene. Images get partitioned into regions for accurate interpretation. Region growing has an advantage that it can be applied to noisy images. A region of an image is defined as a connected homogenous subset of the image with respect to some criterion such as gray level or texture.

3.2. Segmentation

Segmentation is the way of extracting and representing information from an image to group pixels together into regions of similarity. Segmentation has much importance in interpreting an image if the image is preprocessed for noise reduction. In image segmentation, Regions having similar characteristics are identified and assembled together. Image segmentation may use region detection, statistical classification, thresholding, edge detection or combination of any two or more. But in most of the cases edge-based or region-based segmentation is preferred.

3.2.1. Edge-based segmentation

Edge-based Segmentation depends on edges between distinct regions, and the segmentation algorithm will find boundary separating these regions. It involves two steps. Edge detection and edge linking. Operators like Laplacian, Gradient, LoG (Laplacian of Gradients), Sobel, Canny filtering can be used for finding discontinuities in gray level or color or texture etc. Edges detected cannot be used directly for edge linking. To represent the region border, post-processing steps are included so that edges are combined into edge chains. Edge linking has local processing and global processing .In local processing magnitude and direction of gradient vector are estimated. Global Processing uses Hough Transform (HT) for edge linking.

3.2.2. Region-based segmentation

Region-based segmentation depends on common patterns in grey values within a region (cluster) of neighbor pixels. A cluster is a region, and according to anatomical roles Region-based segmentation will group regions considering value similarity (Gray value differences or gray value variance) and spatial proximity (Euclidean distance, closeness of a region).

There are three approaches in Region-based segmentation:

(i) Region merging

Recursively merges regions of similarity. Similarity includes (Gray values, Size, Color, Shape, Texture, Spatial proximity). Determining similarity between two regions is most important step. One or few pixels are will be assigned with labels and those pixels are referred as “Seed(s)”. In view of these seed regions neighboring pixels are included as long as some predicate evaluates to genuine (True). But choice of seed is difficult. One way is to start at a single point and grow region. Another way is pick random number of start points so that labelling of all the pixels can be avoided.

(ii)Region splitting

It divides unrelated regions recursively. The approach is opposite to that of region merging. First, take an assumption that image is treated as homogenous. If it is not true, split the image into four sub images and repeat the steps until the entire image is spitted into homogenous regions. The disadvantage is that adjacent and homogeneous regions may be created, but not merged.

(iii)Split and Merge

It is also known as “Quad-tree”. Separate the image into regions based on a given similarity measure. Then merge regions based on the same or a different similarity measure. It gives best results of segmentation. If a region R is heterogeneous then split it into four sub regions. If two adjacent regions are homogeneous, they are merged. When no further merging or splitting is possible, the algorithm stops.

3.2.3. Segmentation by thresholding

Thresholding is the simplest algorithm for image segmentation. In case of a gray scale image, a binary image can be created using threshold. For a given pixel, if the intensity value is less than the defined constant (Threshold) then the pixel is replaced with a black pixel. Else, the pixel is replaced with white pixel. For a given image the selection of these threshold can be obtained from histogram.

Multi-level thresholding

Multi-level thresholding is used if there are three or more dominant modes in the image histogram. The histogram will be partitioned by multiple threshold.

Adaptive thresholding

Adaptive thresholding is used if the images have uneven illumination. It approach is dividing that original image into sub images, and for each sub image thresholding is applied.

3.2.4. Color-based segmentation

In most of the sports, the playground will be in green color and appears dominant over others in a video sequence frames. Eliminating green field can segment players and other objects of interests from the scene. The Green field is eliminated according to equation 1.

$$Ground_{(x,y)} = \begin{cases} 0, & g(x,y) > r(x,y) > b(x,y); \\ 1, & otherwise \end{cases} \quad (1)$$

Where g(x,y),r(x,y),b(x,y) are green,red,blue components at a pixel (x,y).

3.3. Edge detection

In Image processing, detection of an edge is a critical topic. There are various edge location strategies, for example, Prewitt, Robert, Sobel and wary. Edge recognition is an essential stride in recognizing the object. It is a procedure of discovering sharp discontinuities in an image. The discontinuities are sudden changes in pixel values that describe object boundaries in a scene. So, the objective of edge detection is to deliver a line drawing of the input image. The extracted features are then utilized in object recognition and object tracking.

A traditional technique for edge recognition includes the utilization of operators, a two-dimensional channel (filter). In an image, an edge is identified where the gradient is greatest. The operator lives up to expectations by recognizing these vast gradients to discover the edges. There are many operators intended to identify certain sorts of edges. The operators can be arranged to look for vertical, diagonal, horizontal edges. One major issue with edge detection is the presence of noise in an image. It's hard to reduce noise because the image will be either misshaped or obscured. Instead, the operator can reduce localized noisy pixels by taking the average of enough data. However, always step change in intensity may not be possible in every case. Sometimes the intensity change can be gradual where the operator then must be altered for proper edge detection. In this way, there are issues of missing genuine edges, false edge recognition, and high computational time. Thus, canny and Sobel edge detectors are acquainted with defeat these issues.

3.3.1. Sobel operator

Let us take image neighborhood defined as shown:

Z1	Z2	Z3
Z4	Z5	Z6
Z7	Z8	Z9

The Sobel edge detector is a discrete differential operator. The operator utilizes two 3x3 kernels for spotting gradients along x-direction and y-direction.

Let us take image neighborhood defined as shown:

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

G_x

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

G_y

The image is convolved with both kernels to estimate the derivatives in horizontal and vertical change. At every given point, magnitude of the gradient can be approximated with:

$$G = \sqrt{G_x^{2} + G_y^{2}} \qquad (2)$$

G_x = (z7+2z8+z9)-(z1+2z2+z3); G_y = (z3+2z6+z9)-(z1+2z4+z7).

However, the gradient magnitude can be computed fast with:

$$G = |G_x| + |G_y| \qquad (3)$$

And the Orientation,

$$\theta = \tan^{-1}(G_y / G_x) \qquad (4)$$

Because of Sobel operator's smoothing impact (Gaussian smoothing), it is less sensitive to noise in images. Then again, smoothing influences the exactness of edge detection. As it were, the Sobel technique does not create an image with high precision for edge detection. However, its quality is sufficiently satisfactory to be utilized as a part of various applications.

3.3.2. Canny operator

The Canny administrator is also known as the optimal detector. Initial, a Gaussian channels (filters) is utilized to smooth the image for noise removal. Second, like Sobel operator find the gradient magnitudes. Third, non-maximum suppression is connected in which the algorithm uproots pixels that are not a piece of an edge. The last step includes the utilization of hysteresis thresholding along edges. Hysteresis utilizes two limits, upper and lower. In the event that a pixel gradient is higher than the upper edge, then the pixel will be considered as an edge. On the off chance that a pixel gradient is underneath the lower limit, then the pixel will be disposed of. At last, if the pixel gradient is between the two limits, then just the pixel that is associated with the upper edge is considered as an edge.

Fig 3.3(a) Input test image

Fig 3.3(b) Sobel operator result

Fig 3.3(b) Canny operator result

3.3.3. Sobel operator Vs Canny operator

Sobel operator is simple, and it detects edges and their orientation. On the other hand, Sobel is Inaccurate and sensitive to noise. In Canny operator, smoothing effect is done to remove noise. Good localization is provided, and the response is smooth. It enhances signal to noise ratio using non-maximal suppression and Provide noise immunity. The problem with the canny operator is time-consuming. So, in real-time applications it's hard to implement canny compared to Sobel operator. So, in this work Sobel operator is used to find any gradients to reach real-time requirements.

3.4. Line detection

Basic line detection masks available for line detection at various orientations is as follows:

-1	-1	-1
2	2	2
-1	-1	-1

Horizontal line

-1	2	-1
-1	2	-1
-1	2	-1

vertical line

-1	-1	2
-1	2	-1
2	-1	-1

+45° inclination line

2	-1	-1
-1	2	-1
-1	-1	2

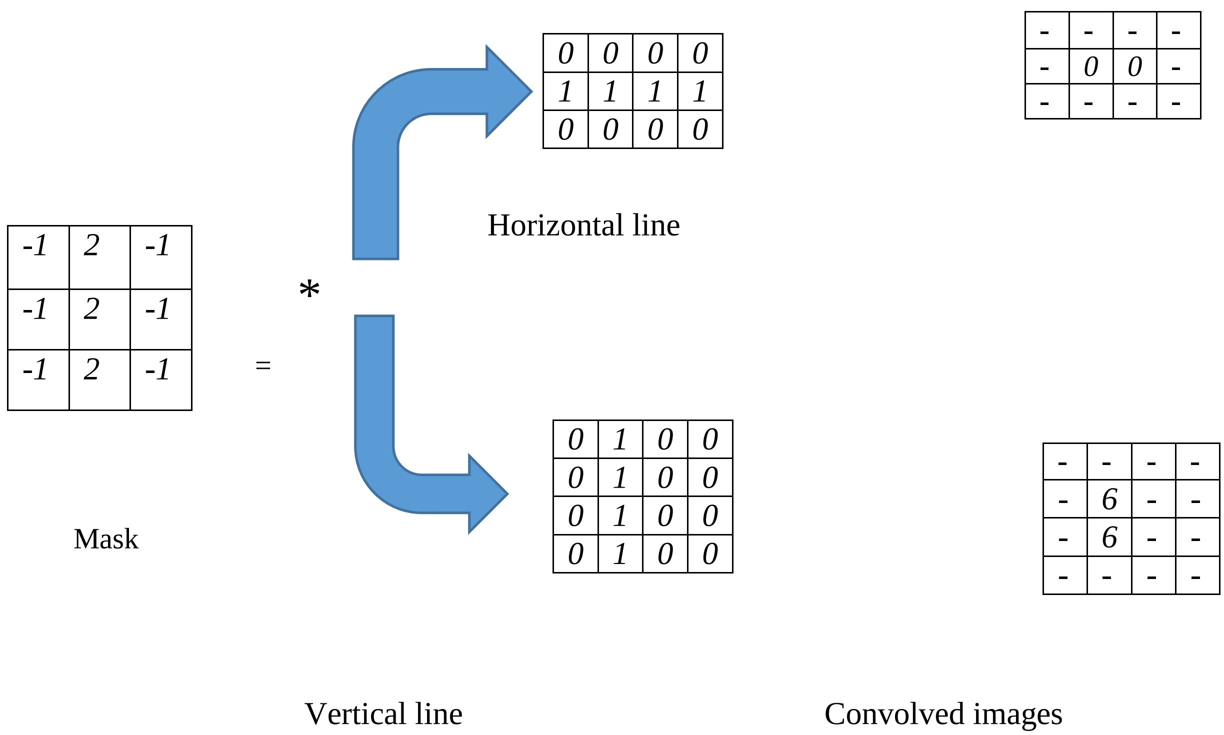
-45° inclination line

All the masks are applied over the entire image, responses are taken and maximum response is considered.

At a given point(x,y) Let R1(x,y),R2(x,y),R3(x,y),R4(x,y) be the responses of horizontal, vertical, +45° inclination line, -45° inclination line masks respectively. Then, the overall response R is defined as

$$R(x, y) = \max(|R1(x, y)|, |R2(x, y)|, |R3(x, y)|, |R4(x, y)|),$$

If R(x, y) > T, then discontinuity. Where T is the discontinuity.



In the above example, the given input mask is convolved with the image. It is observed that the convolution is maximum if the image contains vertical edges. It is so because the mask of entry is a vertical line detector. If the mask of entry is horizontal line detector, then the convolution is maximum where the image contains horizontal edges.

But in practice, simply these line detection masks may not solve the purpose completely because there will be lines with different orientations. Simply defining horizontal, vertical, diagonal will not identify all the lines. So, here comes Hough transform to solve this purpose efficiently.

3.4.1. Hough Transform

Let us take any point (x_i, y_i) in xy-plane. Then, the straight line equation passing through point (x_i, y_i) in slope-intercept form is given by

$$y_i = m x_i + c \quad (5)$$

Where ‘m’ is the slope of the line with respect to x-axis and ‘c’ is a constant

There will be infinite number of lines pass through the point (x_i, y_i) for different values of ‘m’ and ‘c’. However, rewriting the equation as:

$$b = -a x_i + y_i \quad (6)$$

Now x_i, y_i are constants and ‘a’ is a function of ‘b’. Considering ab-plane (also called ‘Parameter Space’) yields a single line equation for a fixed pair (x_i, y_i). Further, a second point (x_j, y_j) similarly has a line in parameter space joined with it, and unless they are parallel, this line meets the line associated with (x_i, y_i) eventually at (a', b'), where ‘a'’ is the slope and ‘b'’ is the intercept of the line containing both (x_i, y_i) & (x_j, y_j) in the xy-plane. Indeed, all the points on this line have lines in parameter space that meet at (a', b').

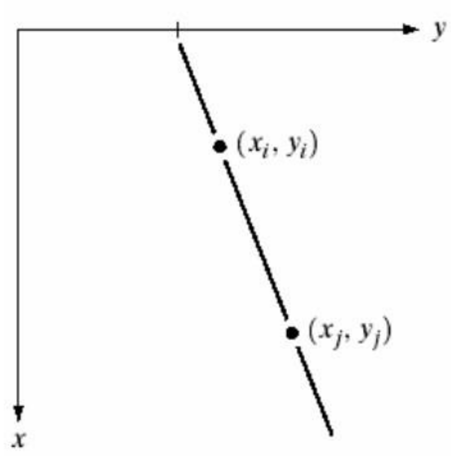


Fig 3.4(a) xy-plane

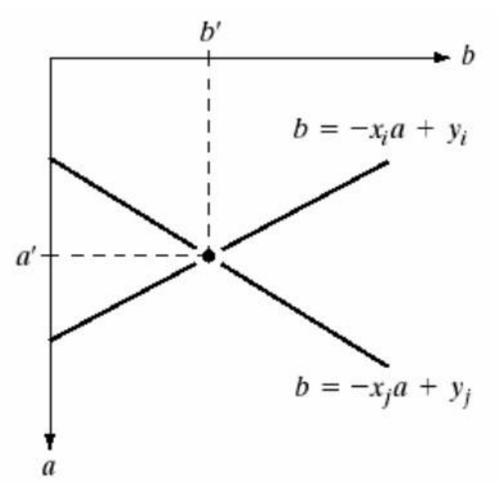


Fig 3.4(b) parameter space

On a basic level, the parameter-space lines comparing to all points ((x_k,y_k) in the xy-plane could be plotted. Using the points that are identified in parameter space, where number of parameter space lines intersect, the principal lines in that plane get identified. But In practice this approach has a difficulty, however is that a (slope of the line) approaches infinity as the line approaches the vertical direction.one way around this difficulty is to use normal representation of a line.

$$X\cos\theta + y\sin\theta = \rho. \quad (7)$$

Where ρ, θ are space parameters.

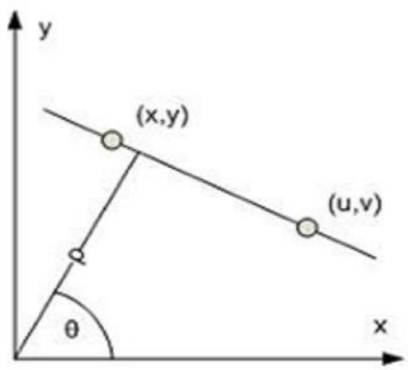


Fig 3.4(c) (ρ, θ) Parameterization of line in the xy-plane

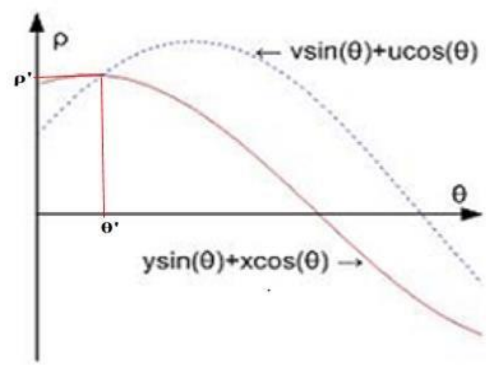


Fig 3.4(d) sinusoidal curves in (ρ, θ) planes

The point line (ρ', θ') corresponds to the line passing through (x_i,y_i) & (x_j,y_j) in the xy-plane

EXAMPLE:

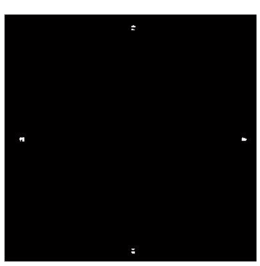


Fig 3.4(e) test points

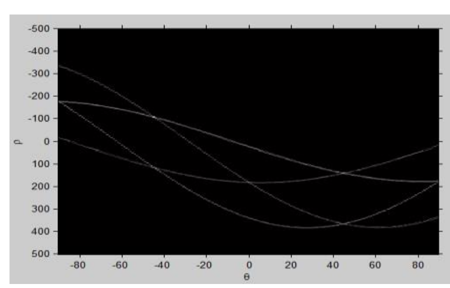


Fig 3.4(f) Hough transform

In the Hough transform intersection of two or more points implies that a line is detected.

3.5. Morphological processing/Filtering

Expansion is a process where some background pixels adjacent to the region are transformed from 0 to 1.It fills gaps and makes the region smooth. Shrinking is a process where some pixels are transformed from 1 to 0. It removes noise, and achieves thinning. For better smoothing both expansion and shrinking are combined.

Morphological processing includes 4 filtering operations (2 basic +2 derived).Basic morphological processing includes Erosion and Dilation. Derived operations include opening and closing. They utilize structuring element (SE) whose size and shape may vary.

EXAMPLE:

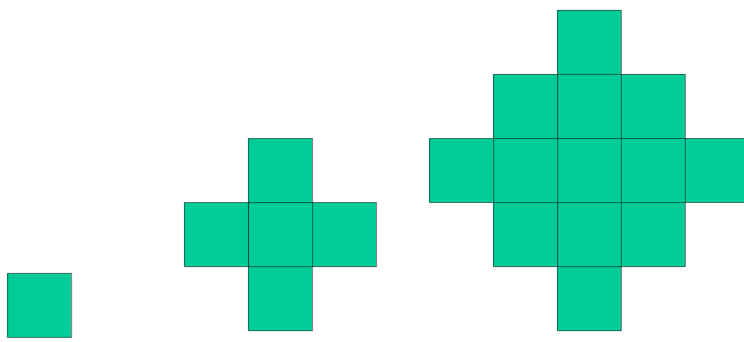


Fig 3.5(a) SE:circle with R=1,2,3

3.5.1. Erosion

Erosion is a concept in which the object gets shrinked. SE is applied on every pixel (x, y) of the image F. Let the center of SE be (x, y). F(x, y) =1 if the whole SE is included in the region. Else, f(x, y) =0**.**

3.5.2. Dilation

Dilation is a concept in which the object gets expanded. SE is applied on every pixel (x, y) of the image F. Let the center of SE be (x, y). F(x, y) =1 if at least one among all pixels of SE is included in the region. Else, f(x, y) =0.

3.5.2. Opening

Opening is a concept of erosion followed by dilation with same SE. It is used to filter out "active" details. Finally it shrinks the region.

3.5.3. Closing

Closing is a combination of dilation followed by erosion with same SE. It is used for smoothening by expansion, Filling gaps and holes that are smaller than the SE.

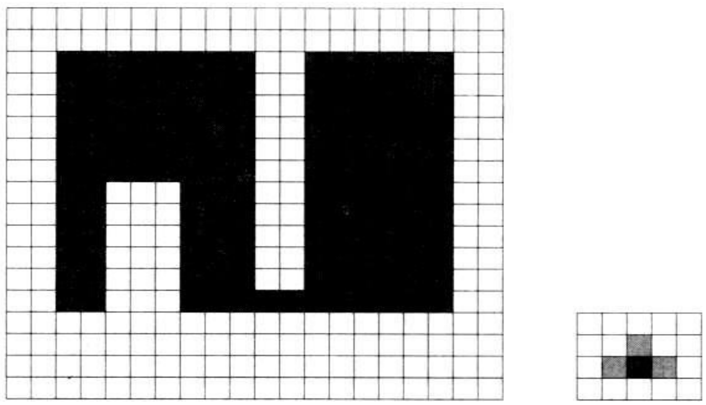


Fig 3.5(b) Original image and Structural Element

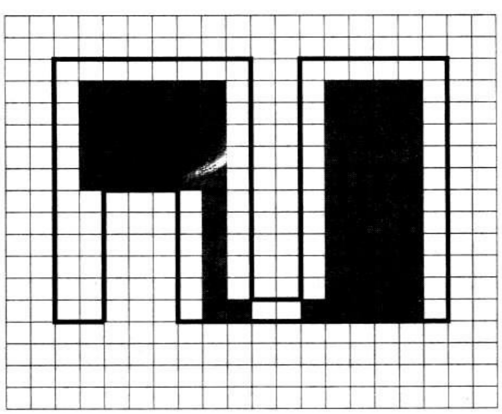


Fig 3.5(c) Erosion

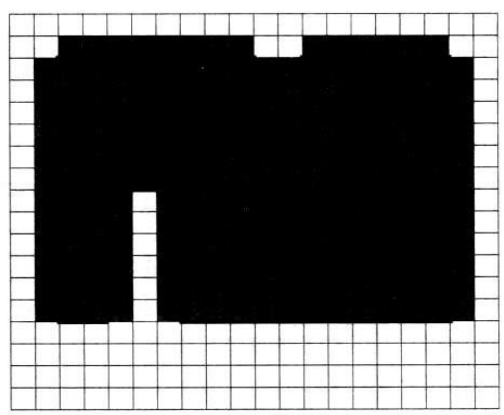


Fig 3.5(d) Dilation

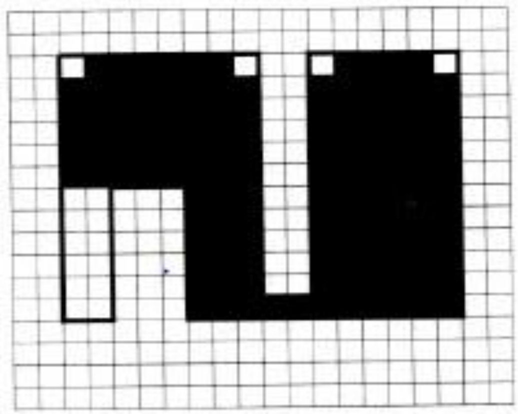


Fig 3.5(e) Opening

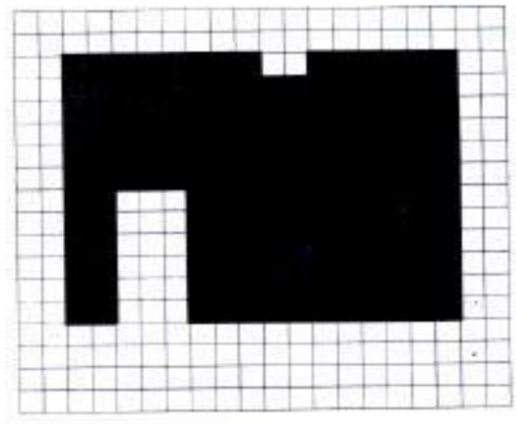


Fig 3.5(f) Closing

3.6. Proposed algorithm:

The flowchart for the proposed algorithm is shown in Fig 3.6 (a). It involves mainly four stages which include background elimination, obtaining player and ball attributes, Line detection and line elimination, unwanted object elimination. The operation of flowchart is explained step by step as follows: First, take an input frame as shown in Fig 3.6 (b).observe that the field is green in color and dominant over other colors. Here, we apply color based elimination of ground as explained in section 3.2.4 (using equation 1) and the result is as shown in Fig 3.6 (c).In Fig 3.6(c),observe that some grey color objects like the field lines, goal posts, and ball are missing. To retain then, we apply Sobel gradient operator to the input frame that detect sharp changes and the result is as shown in Fig 3.6 (d).Fig 3.6(c) & Fig 3.6(d) are combined to give complete ground elimination result as shown in Fig 3.6(e).Dilation and erosion concepts are used here to connect disjoint ends.

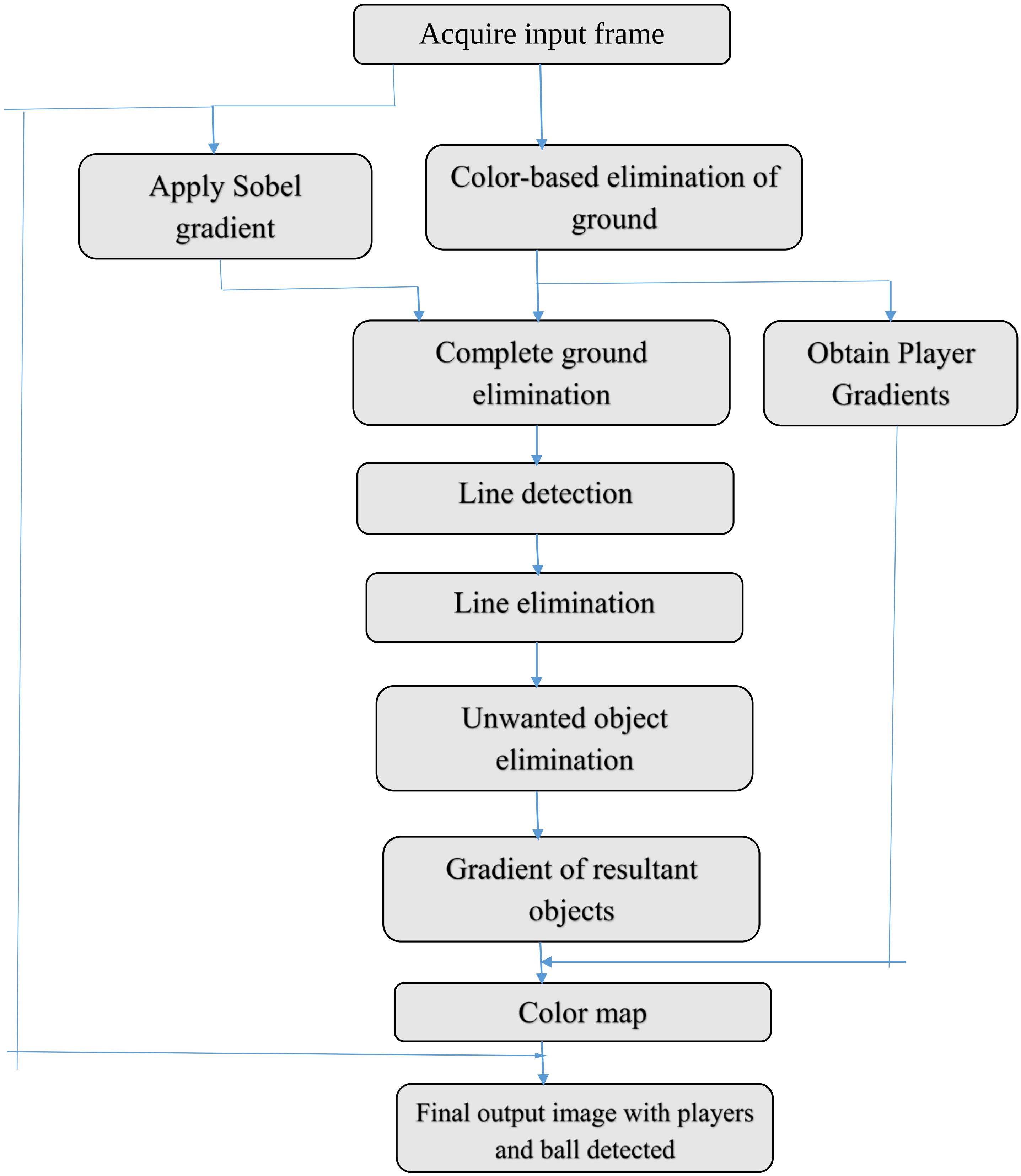


Fig 3.6(a) Flowchart for detection of soccer ball and players

Now aim to keep plays and ball alone in the field. For that lines and other unwanted objects present in the scene has to be eliminated. To do so, lines has to be detected. Here, we use Hough transform concept for detecting lines. After detecting the number of lines detected can be known by counting the number of Hough peaks. According to the Hough peaks, voting schemes is implemented, where only high priority peaks are considered. After applying Hough transform voting scheme, the detected lines are as shown in Fig. 3.6(f).Directly we can't extract lines from the scene. But, indirectly we can extract lines using Hough coordinates as shown in 3.6. (g). for that, First a zero matrix of input frame size is taken. A line will have starting and ending coordinate. Using those coordinates the lines are derived. After the line detection, the detected lines are removed from the complete ground elimination result (Fig 3.6(e)) and the Line elimination result is as shown in Fig 3.6(h). The remaining objects incorporate ball, players, some little protests, some dotted lines and some huge objects in the ground.

Our point is to keep the ball and the players in the ground. Thus, characterize some ideal edge by taking ratio of area and perimeter of every object. The articles with extensive range are wiped out. The ball and players have a higher thickness of pixels. As a result, the ratio of area and perimeter is high compared with the little protests and the spotted lines. The ideal limit quality takes out most of the undesirable items. In this work the threshold limit is 1.75. The unwanted object elimination result is shown in Fig 3.6(i) and its gradient (gradient of resultant objects) is as shown in Fig 3.6(j).After unwanted object elimination only players and ball remain in the scene. But observe that in the original frame the players under occlusion and the player merged with the line are not included in the scene after unwanted object elimination. To recover those players, obtain player gradients before thresholding by applying Sobel gradient to Ground elimination result as shown in Fig 3.6(k).Now combine player gradient to gradient of resultant objects which yields Final gradient as shown in Fig 3.6(l), where all the player and ball gradients are obtained. Input image is a color image. But final gradient is an intensity image. So, directly they can't be combined. So color-map is applied to the gradient as shown in Fig 3.6(m).After color-map, the input frame is added to yield the final output image with all players and ball detected as shown in the Fig 3.6(n).

3.7. Results and Discussions

The input image is taken final match played between Brazil and Germany in 2002 FIFA World cup. The algorithm effectively distinguishes the ball in the sequence of video frames when the ball is merged with the line. This algorithm is applied to different soccer match frames, particularly when the ball is appended with the line, and players are in collision.

Fig 3.6(b) Input frame

Fig 3.6(c) Color-based elimination of ground

Fig 3.6(d) Sobel gradient image

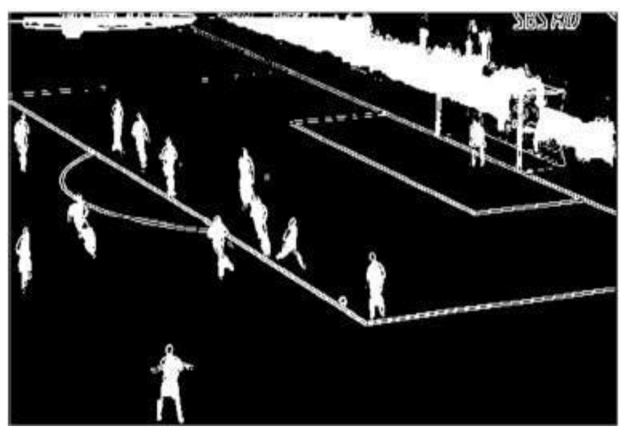
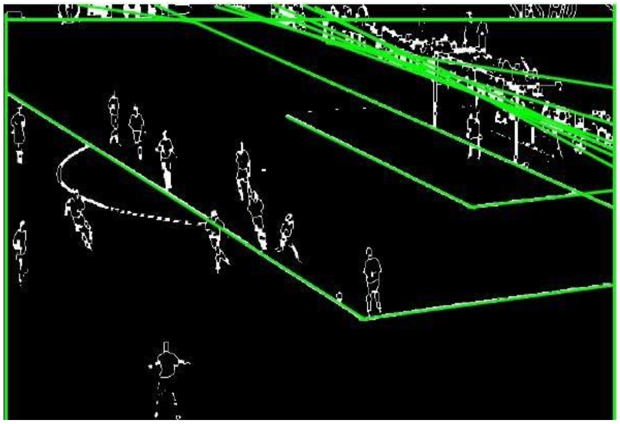


Fig 3.6(e) Ground elimination result



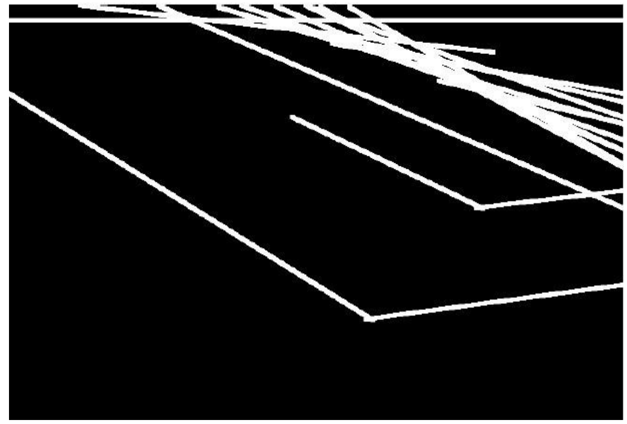


Fig 3.6(f) Line detection

Fig 3.6(g) Extracted lines

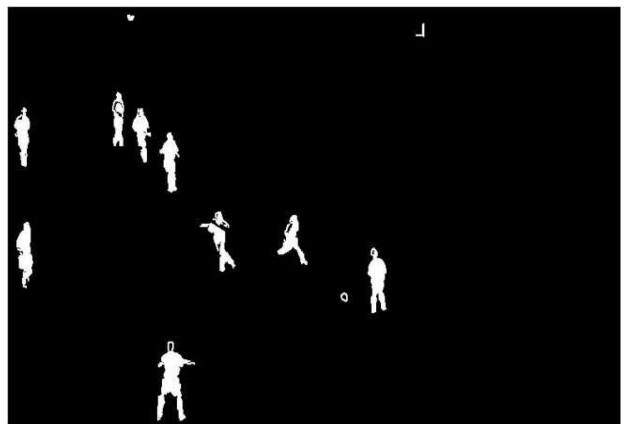
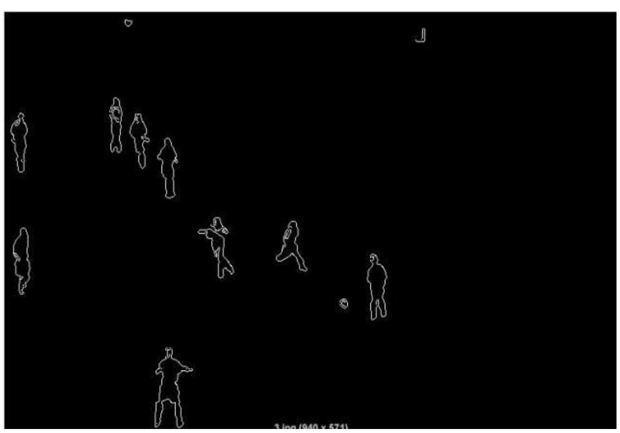


Fig 3.6(h) Line elimination result

Fig 3.6(i) Unwanted object elimination result



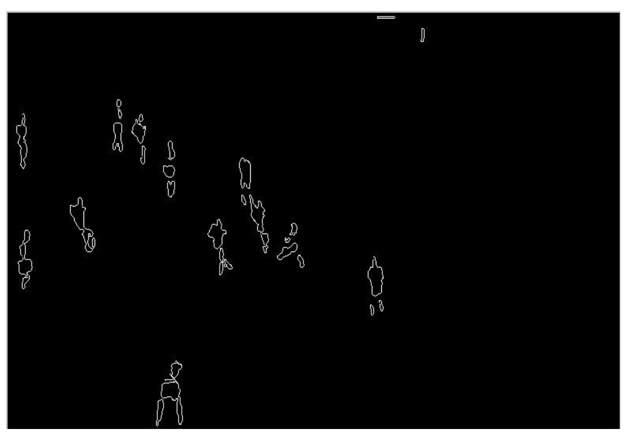
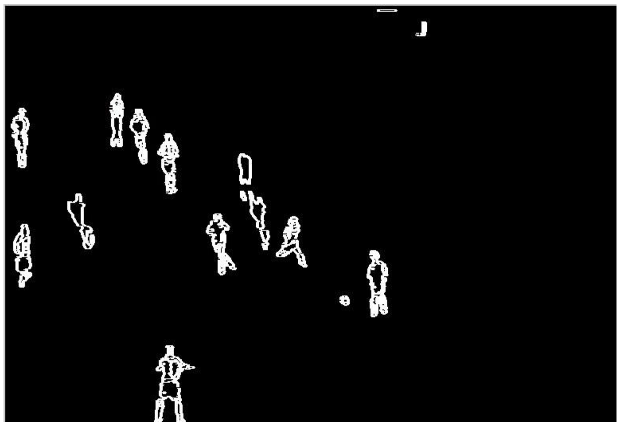


Fig 3.6(j) Gradient of the resultant objects

Fig 3.6(k) Player gradients,



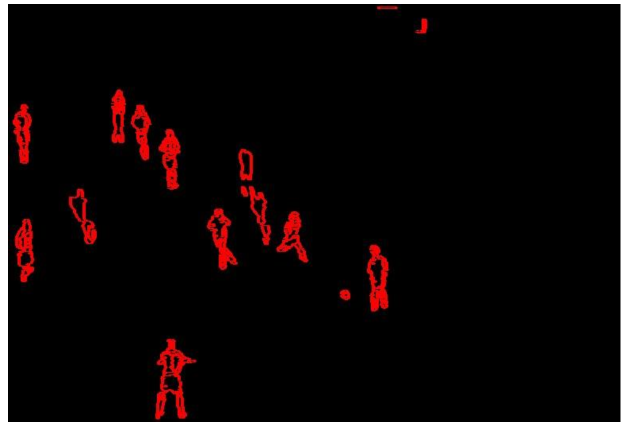


Fig 3.6(l) Final gradient image

Fig 3.6(m) Final color mapped gradient image

Fig 3.6(n) Final output image

Chapter 4

Recognition of Players

RECOGNITION OF PLAYERS

4.1. Feature Detection

The idea of feature detection is a technique that point at figuring reflections of image information and making local decisions at every pixel whether there is an image feature or not. A feature can be defined as basic primitives (or) point of interests in an image also it is a starting point for subsequent algorithms. Feature detection comes under low-level image –analysis.

Features include:

Edges: An edge is defined as set of points that constitute boundary between two regions. In practical an edge is also defined as a set of points in an image that have greatest gradient magnitude. An edge is a one-dimensional structure.

Corners: A point like features in an image that have two-dimensional structure. These algorithms are well established and aligned so that the need of explicit edge detection is avoided. For a given image gradient, these algorithms will search for high level of curvature.

Ridge: A ridge is defined as a one dimensional curve that represents symmetry axis, and also has an attribute of local ridge width associated with each ridge point. But there is a difficulty in this type of approached algorithms. i.e., for general class of grey scale images it is difficulty to extract ridge compared to that of other features like corner, edge, blobs.

Blobs: Blobs give an integral portrayal of image structures as far as areas, rather than corners that are more point like. Blob detectors can detect areas in an image which are too smooth to be detected by a corner detector. In this work, Blobs are used in SURF feature detection in recognizing a player.

Common edge detectors are Canny, Sobel, Harris, SUSAN detectors. Common corner detectors include Harris, SUSAN, FAST, LoG, and Determinant of Hessian. Blob detectors are FAST, LoG, MSER, Grey level blobs.

4.2. Feature Extraction

When features have been distinguished, a neighborhood image patch around the feature can be extracted. This extraction may include truly extensive measures of image processing. The outcome is known as a ***feature descriptor*** or ***feature vector***. Among the methodologies that are utilized to feature description,, one can say N jets and neighborhood histograms . Besides providing attribute information, it also provide complementary attributes. Complimentary attributes include orientation of edges, gradient magnitude and polarity in edge detection, Blob strength in detection of blob.

4.3. Feature matching.

Gray scale-based matching: It inspects images as 2 D flags and make utilization of statistical ways to deal with pursuit out the correlation among signals, thus get their looking like and homonymy focuses. Gray scale-based matching give smart results, however, take longer when contrasted with Feature-based matching.

Feature-based matching: Feature-based matching discovers a few alternatives to the image like focuses, lines, surfaces and planes, thus characterizes properties of these choices thus matches the images following these attributes. The comparative decisions that square measure utilized square measure compositions, shapes, and spatial positions. It includes singularly partial pixels and reduces computation. It offers registration precision because of the local sensitivity of the matching properties. The feature extraction decreases the effects of noise and will expand the compliance with changes in intensity values, transforming, and occlusion. The feature-based systems are not accurate compared to gray-scale matching.

4.4. Speeded Up Robust Features (SURF)

SURF could be a sturdy native feature detector. It's part impressed by SIFT(Scale-Invariant Feature Transform).The standard version of SURF is many times quicker than SIFT.SURF is predicated on the sum of 2 D Haar transform responses associate degreed makes an economical use if integral pictures. Applications embrace visual perception, Face Recognition, creating 3D scenes, object following.

SURF could be a detector and a superior descriptor points of interest in a picture, wherever the image is remodeled into coordinate's victimization the technique referred to as a multi-resolution. Is to create the copy of the first image with pyramidal Gaussian or Laplacian pyramid form and acquire the image with constant size however with reduced information measure. Therefore a special blurring impact on an imaginative image, referred to as scale- is achieved. This system ensures that the purpose of interest or scale invariant

Generating feature vector is as follows:

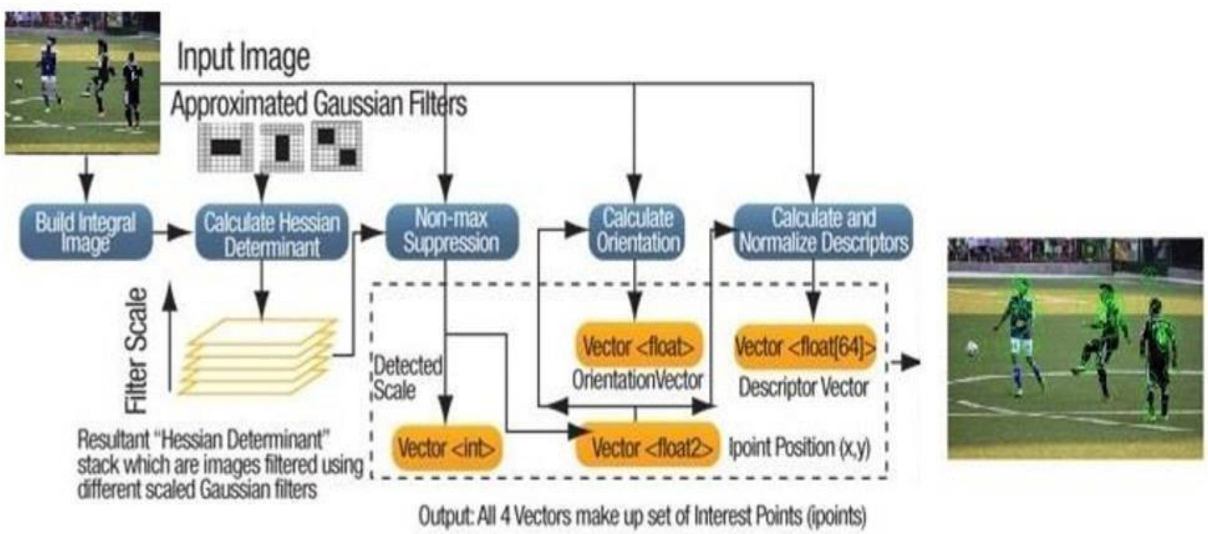


Fig 4.4(a) Generating feature vector using SURF

For rotation invariance, for every feature recognized, Look at pixels around of 6*σ radius. Compute the x and y Haar transform for every point. Utilize the subsequent values as x and y coordinated in a Cartesian map. Weight every point with a Gaussian of 2*σ considering the separation from the interest point. Rotate a wedge of π/3 radians around the circle. Pick direction of greatest aggregate weight.

Feature vector in SURF is a 64 dimensional vector for each 16 sub-regions, each sub region include 4 values.

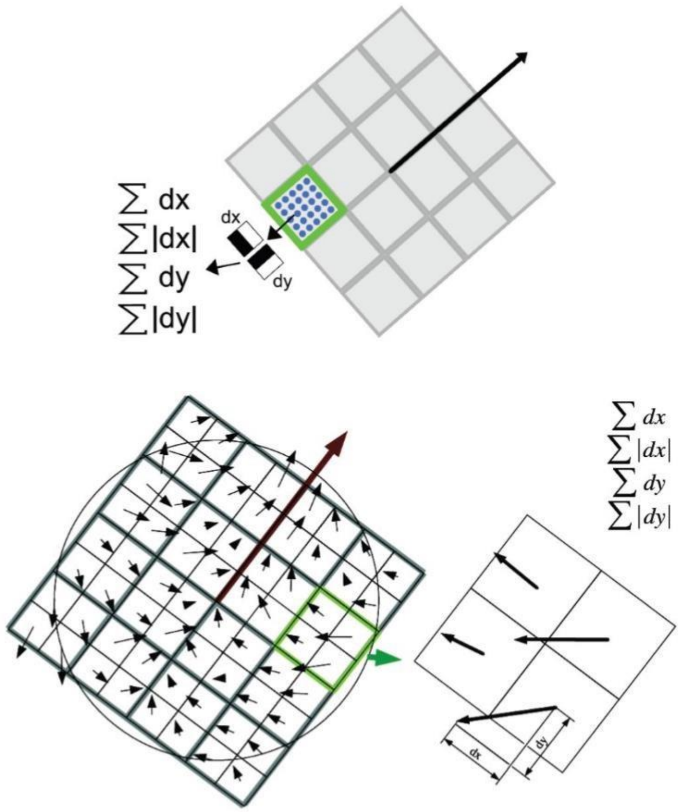


Fig 4.4(b) Feature descriptor/vector

4.5. Proposed algorithm

The flow of proposed algorithm is shown in Fig 4.5(a).Jersey number is the base image and video frame is the test image. Our approach is based on feature detection, extraction, matching and finally identifying player. SURF (Speeded Up Robust Features) approximates or even outperforms previously proposed schemes with respect to repeatability, distinctiveness, and robustness, yet can be computed and compared much faster. This is achieved by relying on integral images for image convolutions; by building on the strengths of the leading existing detectors and descriptors (specifically, using a Hessian matrix-based measure for the detector, and a distribution-based descriptor); and by simplifying these methods to the essential. This leads to a combination of novel detection, description, and matching steps. Affine transformation is used for estimating geometric transformations. After recovering the base image, SSIM algorithm is used for estimating structural similarity index value and finally displaying player identity.

A base image of player jersey number as shown in Fig. 4.5(b) is taken. A reference frame is as shown in Fig. 4.5(c).For feature detection, SURF algorithm is used. SURF features are detected, extracted for both base image and test image as shown in Fig. 4.5(d) and Fig 4.5(e) respectively. Then, feature matching is applied and the result is as shown in Fig. 4.5(f). Finally affine transformation is used to estimate the geometric transformation after which the player jersey number is recovered as shown in Fig 4.5(g).Now, we have base image and recovered image. SSIM is a similarity measure. There is chance of wrong identification, let us say if jersey number 37 has shear along x-axis, the same may be wrongly identified as jersey number 31 which is not true. So we will go for SSIM. After Apply SSIM (Structural Similarity index) algorithm (Fig .4.5(h)), finally displayed player identity in recognized player region as shown in Fig. 4.5(i).

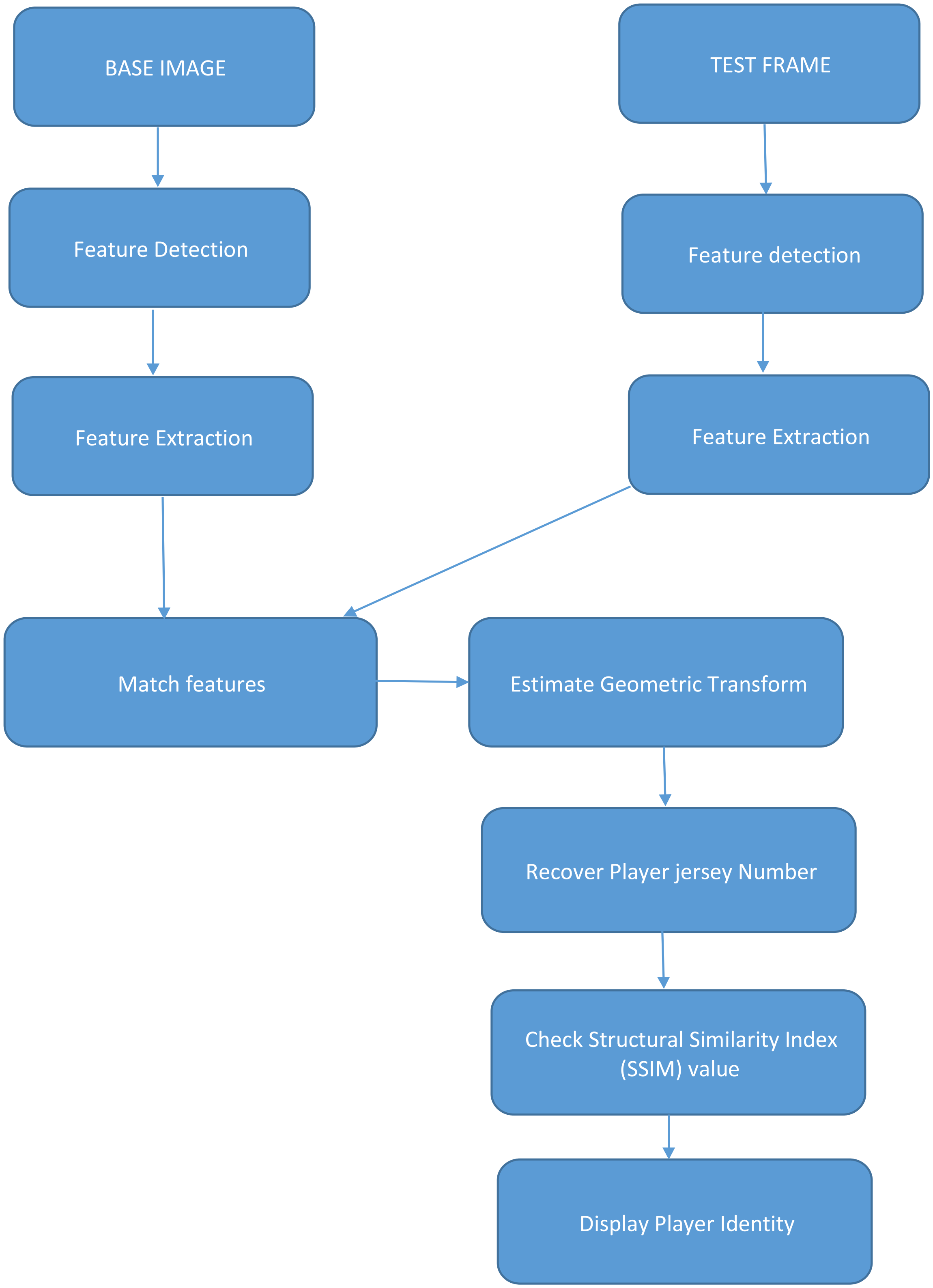


Fig 4.5(a) Flow chart for recognition of players

Fig 4.5(b) Base image

Fig 4.5(c) Test frame

Fig 4.5 (b) SURF features of Base image

Fig 4.5(e) SURF features of Test Frame

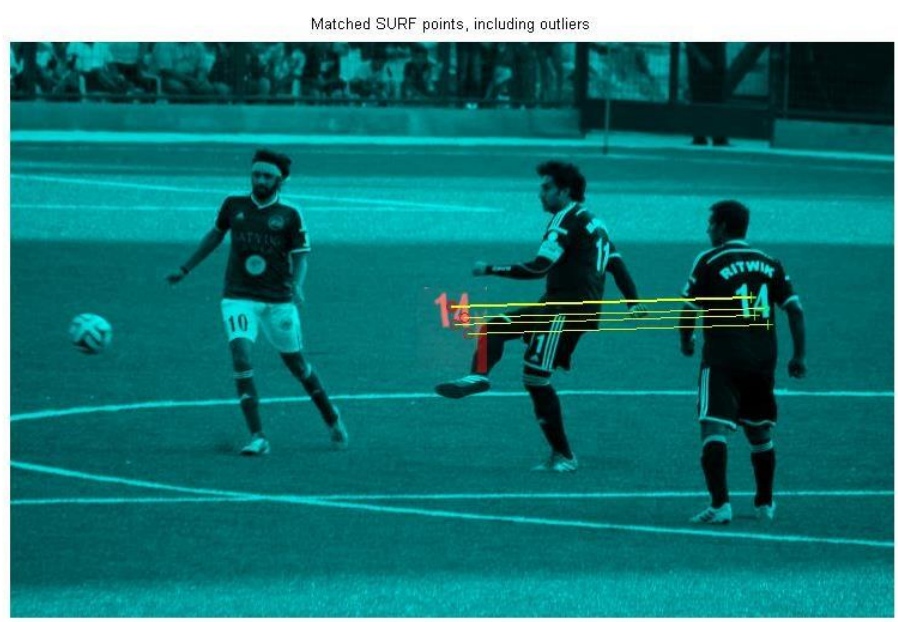


Fig 4.5(f) matched features

Fig 4.5(g) Recovered Player number

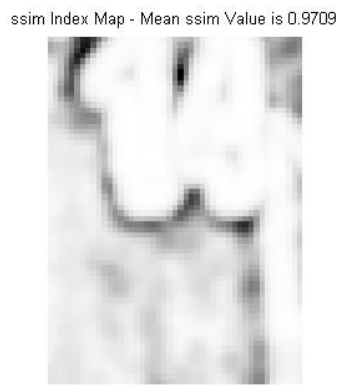


Fig 4.5(h). SSIM value

Fig 4.5(i) Final output showing player Identity

4.6. Results and Discussions

The algorithm effectively recognizes the player using jersey number. the same algorithm is applied for multiple player recognition and the results are as shown.

EXAMPLE 1:

Fig 4.6(a) Input base images1

Fig 4.6(b) output recognized players1

EXAMPLE 2:

Fig 4.6(c) Input base images2

Fig 4.6(d) output recognized players2

EXAMPLE 3:

Fig 4.6(e) Input base images3

Fig 4.6(e) Output recognized players3

CHAPTER 5

CONCLUSION

CONCLUSION

Canny operator is widely used for edge detection. But, it is a time consuming .So, in our work Sobel operator is used to reach real-time applications. Color based elimination of ground is used as it involves less computation and less complexity. HT is used for line detection and extraction. The main aim of this work include detection of the soccer ball and players within a Sequence of video frames, especially when the ball is merged with the lines in the ground and the players are in collision. The experimental results show the capability and robustness of detecting ball and players. But there is a limitation, the algorithm fails to detect goal keeper, because during threshold, he gets eliminated considering as an object with very large area.

SURF is preferred over SIFT. Fast robustness feature detection of SURF algorithm is used for each frame image, to find player quickly and display player information. Affine transform is used to estimate transformation. SSIM is used to avoid false identification. The experimental results show the capability and robustness of recognizing players.

5.1 Future scope

Implement an algorithm that can detect and differentiate player according to team. To, achieve this, the concept of color filters can be used. Rather than recognition the player with jersey number, an alternative approach like face recognition algorithm can be can be implemented. Principal Component Analysis (PCA), Independent Component Analysis (ICA), Linear Discriminant Analysis (LDA) are some of few approaches for face recognition.

REFERENCES

[1]. Xiao-Feng Tong, Han-Qing Lu, Qing-Shan Liu, "An Effective and Fast Soccer Ball Detection and Tracking Method," Proceedings of 17th International conference on Pattern Recognition (ICPR), vol. 4, pp.795-798, 2004.

[2]. X. Yu, C. Xu, H. W. Leong, Changsheng Xu, "Trajectory-based ball detection and tracking in broadcast soccer video," IEEE Transactions on Multimedia, vol. 8, no. 6, pp.1164-1178, 2006.

[3]. T. D'Orazio, C. Guaragnella, "A new algorithm for ball recognition using circle Hough transform and neural classifier," Pattern Recognition, vol. 37, no. 3, pp.393-408, 2004.

[4]. Jong-Yun Kim, Tae-Yong Kim, "Soccer Ball Tracking using Dynamic Kalman Filter with Velocity Control," Proceedings of sixth International Conference on Computer Graphics, Imaging and Visualization (CGIV), Pp.367–374, Aug. 2009.

[5]. O. Utsumi, K Miura, I. Ide, S. Sakai, H. Tanaka, "An object detection method for describing soccer games from video," Proceedings of IEEE International Conference on Multimedia and Expo., vol. 1, pp. 45–48, 2002.

[6]. M. M. Naushad Ali, M. Abdullah-Al-Wadud and Seok-Lyong Lee, "An Efficient Algorithm for Detection of Soccer Ball and Players", Proceedings of Conference on Signal and Image Processing (SIP), pp.1-8, 2012.

[7]. L. A. F. Fernandes and M. M. Oliveira, “Real-time line detection through an improved Hough transform voting scheme,” Pattern Recognition, vol. 41, pp. 299–314, 2008.

[8]. K. Jung, K. I. Kim and A. K. Jain, “Text information extraction in images and video: a survey”, Pattern Recognition, vol. 37, number 5, pp. 977–997, May 2004.

[9]. Q. Ye, Q. Huang and S. Jang. “Jersey number detection in sports video for athlete identification”, Proceedings of Visual Communication and Image Processing (VCIP), vol. 5960, pp.1599-1605, July 2005.

[10]. M. Bertini, A. D. Bimbo and W. Nunziati, "Player identification in soccer videos", Proceedings of the 7th ACM SIGMM international workshop on Multimedia information retrieval, pp. 25-32, 2005.

[11]. P. Viola and M. Jones. “Rapid object detection using a boosted cascade of simple features”, Proceedings of CVPR, pp. 511-518, 2001.

[12]. M. Bertini, A. D. Bimbo and W. Nunziati, “Matching faces with textual cues in soccer videos”, IEEE International Conference on Multimedia and Expo, pp. 537 – 540, July 2006.

[13]. E. L. Andrade Neto, E. Khan, J. C. Woods and M. Ghanbari, “Player classification in interactive sport scenes using prior information region space analysis and number recognition”, IEEE International Conference on Image Processing (ICIP) 2003, Barcelona, Spain, vol. III, pp. 129- 132, September 2003.

[14]. K. Jung, K. I. Kim and A. K. Jain, “Text information extraction in images and video: a survey”, Pattern Recognition, Volume 37, number 5, pp. 977–997, May 2004.

[15]. Andrade, E.L., Khan, E., Woods, J.C., Ghanbari, M. "Player Identification in Interactive Sport Scenes Using Region Space Analysis, Prior Information and Number recognition. Proc. of the International Conference on Visual Information Engineering, pp. 57–60 , July 2003

[16]. Frejlichowski, D., Wierzba, P. "A Face-Based Automatic Identification of Football Players during a Sport Television Broadcast". Polish Journal of Environmental Studies 17 (4C), pp. 406–409, 2008.

Dissemination

Upendra Rao. M, Umesh C.Pati, "A Novel Algorithm for Detection of Soccer Ball and Player", Proceedings of 4th International Conference on Communication and Signal Processing, pp. 344-348, April, 2015.