



TRIBHUVAN UNIVERSITY  
INSTITUTE OF ENGINEERING  
CENTRAL CAMPUS, PULCHOWK

**PROJECT PROPOSAL**  
**ON**  
**Coordinated Gameplaying Robots**

**Submitted By:**

Anuj Gautam (069-BEX-404)

Dilip Paneru (069-BEX-413)

Ramesh Neupane(069-BEX-433)

Yesh Kumar Adhikari (069-BEX-447)

**Submitted To:**

Department of Electronics & Computer Engineering  
Central Campus, Pulchowk

(Poush 24, 2072)

## **ACKNOWLEDGEMENT**

We would like to thank the department of Electronics and Computers of IOE, Pulchowk Campus for providing us with the opportunity of doing something that hugely strengthens the society as well as us students as a electronics engineer.

We would also like to acknowledge the authors of various papers that we have utilized in making this report. We respect all the person hours gone into writing such articles. Finally, we would like to thank all the people who are directly or indirectly related during our study and preparation of this report.

## **ABSTRACT**

Problem solving in complex domains require multiple agents, dynamic environment and the need for adaptive learning from the environment. Cooperative centralized multirobot learning is an active research area involving multiple learning entities controlled by a single central governmenting body to learn optimal solutions for an overall system. Robotic football is such a complex task. The paper proposes the task of minor football tactics for development of such a cooperative system. The project will aim to analyze the cooperative learning in various practical conditions so as to come up with an optimal collaborative tactics.

## TABLE OF CONTENTS

ACKNOWLEDGEMENT.....	I
ABSTRACT.....	II
TABLE OF CONTENTS.....	III
LIST OF FIGURES.....	IV
1. Introduction .....	1
1.1. Background .....	1
1.2. Problem Statement.....	2
1.3. Objectives.....	3
1.4. Scope.....	4
2. Methodology .....	5
2.1. System Overview.....	5
2.2. Hardware Aspects.....	7
2.3. Software Aspects.....	10
2.3.1. Learning System.....	11
2.3.2. Image Processing.....	12
3. Project Requirements .....	14
3.1. Software Requirements.....	14
3.2. Hardware Requirements.....	14
4. Expected Outcome.....	15
5. Time Schedule.....	16
6. Cost Estimation.....	17
7. References.....	18

## **List of Figures**

Figure 1: 1-v-1 tactic in robot football

Figure 2: System Block Diagram

Figure 3: Working Hierarchy of Central System

Figure 4: Structure of individual Robot

Figure 5: Software parts of the project

Figure 6: Reinforcement Learning

Figure 7: Robot shape with colored patches

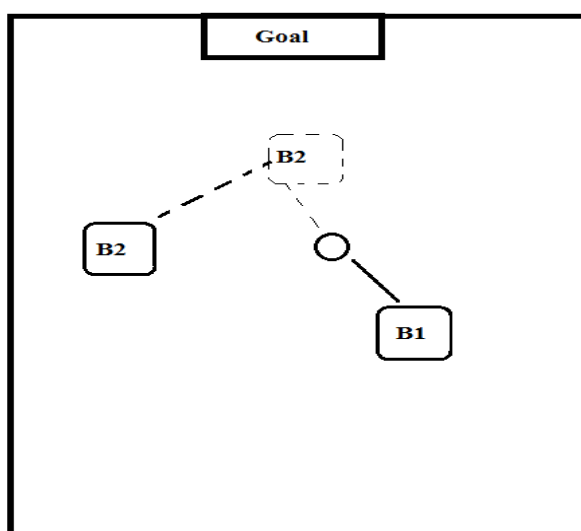
Figure 8: Gantt chart for the project

# 1. INTRODUCTION

## 1.1 Background

The field of distributed robotics has its origins in the late 1980s, when several researchers began investigating issues in multiple mobile robot systems. The problem of efficient multirobot coordination has risen to the forefront of robotics research in recent years. Interest in this problem is motivated by the wide range of application domains demanding multirobot solutions. In general, multirobot coordination strategies assume either a centralized approach, where a single robot/agent plans for the group, or a distributed approach, where each robot is responsible for its own planning. The key advantage of centralized approaches is that they can produce globally optimal plans.

Robot football has been taken as a benchmark for collectively intelligent systems and is an active part of research. The AI and control system of a Robot Football team is highly stratified, each layer having specific functionalities to come together to work as a whole system. The strata are broadly threefold – Skills, Tactics and Plays. Skills are the lower level functionalities such as velocity tracking or actuator enabling; Tactics are sequence of skills such as intercepting the ball or retaining possession while Plays are overall strategies assigning tactics to one or more players.



**figure 1:** 1-vs-1 tactics in robot football

## 1.2 Problem Statement

Robots have been a major part of different solutions to well-known problems. Innumerable intelligent approaches have been applied to solve a vast majority of problems in the world.

What has been recently realized through the inspiration of various biological flocks is that a larger number of robots can solve more distributed problems more efficiently.

Football, being such a cooperative sector, is a curious sector to be solved and optimized using robotics. Therefore, the project will intend to apply and use various algorithms to implement the real-life football tactics in the robotics domain.

The paper primarily focuses on 2 vital football tactics – cooperatively moving the ball to the goal(1-to-1 passing) and an aggressive defense behaviour in a 1 vs 1 scenario. The action space for all these scenarios are chosen according to the logic presented in standard learning scenarios. The work also addresses pressing challenges such as large state space discretization through effective tile coding methods, discussing the possible applications of the tactics in real game situations as well as analyzing the nature of trajectory taken in the state space with varying policies of the opponent.

The problem with tactics designed by handcoded rules is that it makes a very strong assumption about the opponent leaving very little flexibility. Thus, the paper focuses on environment based learning systems.

This abstract problem solution is hoped to be used in arbitrary problems requiring coordinated behaviour.

### 1.3 Objectives

The main objective of the project will be to construct a squad of robots which will be able to cooperatively solve dynamic problems of football like cooperatively getting the ball into the net and strikers versus defender problem. The overall objectives of the project can be enumerated as:

- To meet the course requirements
- Implement Robotic kinematic systems
- Implement communication system between multiple participating agents.
- Solve well known football tactics by mutual cooperation
- Track the robots, terrain and the obstacles using centralized image processing systems
- Detect and avoid obstacles
- Implement collective intelligence algorithms.



## 1.4 Scope

Multibot systems is an emerging field of robotics and has huge applications. The project is scalable to various applications. It can be used for most of the works done by a single robot but is able to solve the problem faster. Since the project uses a centralized system, it will be able to control an arbitrary number of cheap robots. Learning systems enable us to be able to design rules that are very difficult to implement when hardcoded.

The major scopes of coordinatively learning systems are:

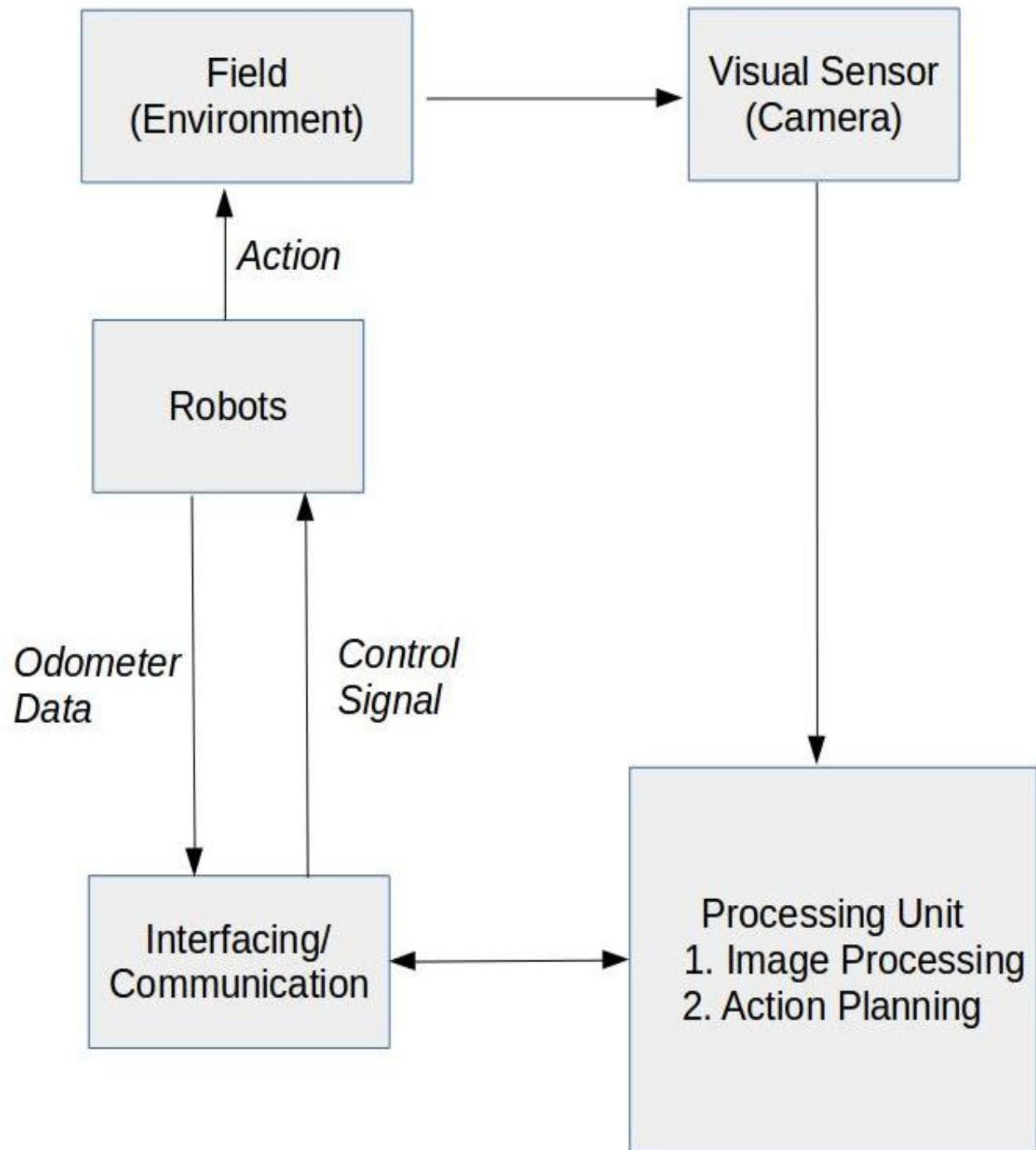
- Coordinated exploration systems
- Post-disaster rescue systems
- Military applications
- Nanorobotics for medical applications
- Warehouse assistance system

## **2. METHODOLOGY**

### **2.1 System Operation Overview**

The system consists of multiple bare-boned mobile robots equipped with just the minimal sensors required for their local actions. The minimal system include basic hardware manipulation module(motor driver), communication module and the pose feedback module.

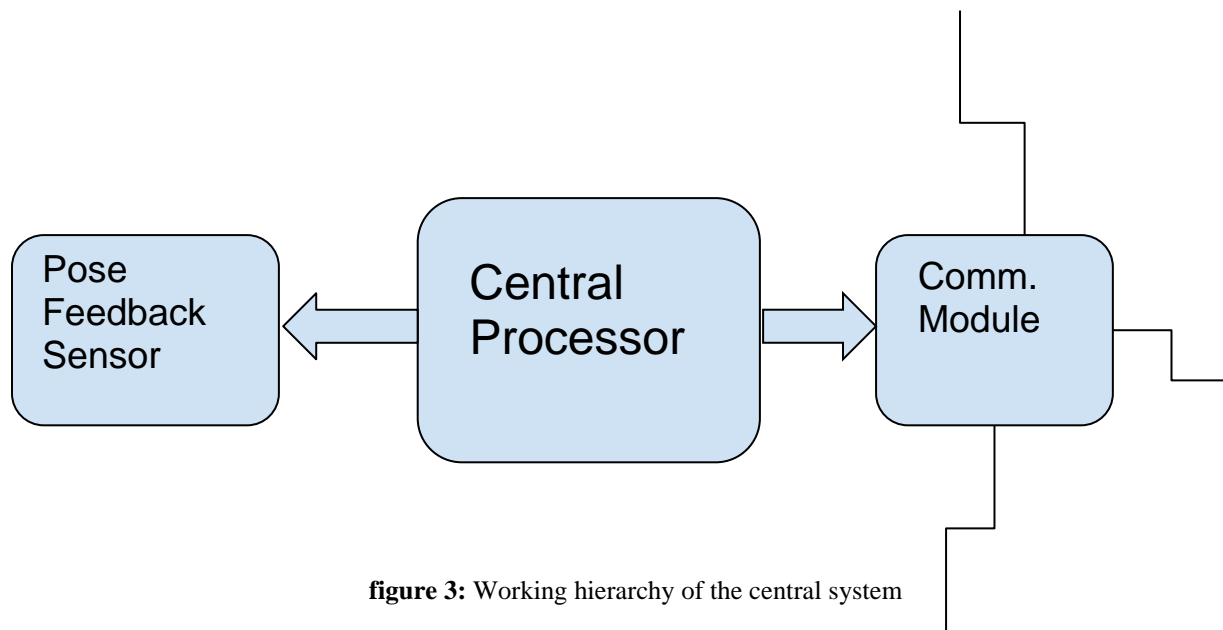
The system requires huge computation power. So the system is centralized and a powerful central computer is proposed to be used for the computationally heavy tasks. The tasks include to inspect, collect and the correct the global pose of the whole environment(the robots, the ball and the objective locations). It will also require to be able to run the learning algorithms used for the solution of the task.



*Figure 2 : Overall System Block Diagram*

## 2.2 Hardware Aspects

In the hardware portion, the system consists of multiple bare-boned mobile robots equipped with just the minimal amount of necessary sensors to keep the cost low. Besides that, the system will also consist of a powerful central computer for the inspection of the system. The centralized computer will also act to correction of various sensor errors from the robot and provide a means to use the powerful algorithm requiring high amount of processing power to solve the collaborative problem.



- **Pose Feedback Sensor:** The pose feedback sensor is a simple sensor which is able to inspect the pose of the objects in the environment. Some candidates for the pose feedback sensor are camera and Kinect Sensor.
- **Communication Module:** The communication module is a module used for the central system to communicate with each individual bots. This is required for various applications in the project. Some of the work done through the communication module include sending the locomotion instruction to each robot and pose correction instruction for each bot. Candidates for the communication module include the WiFi module and the RF module/
- **Central Processor:** The central processor is the main block of the system which will be doing most of the computational works such as pose detection through the data from pose

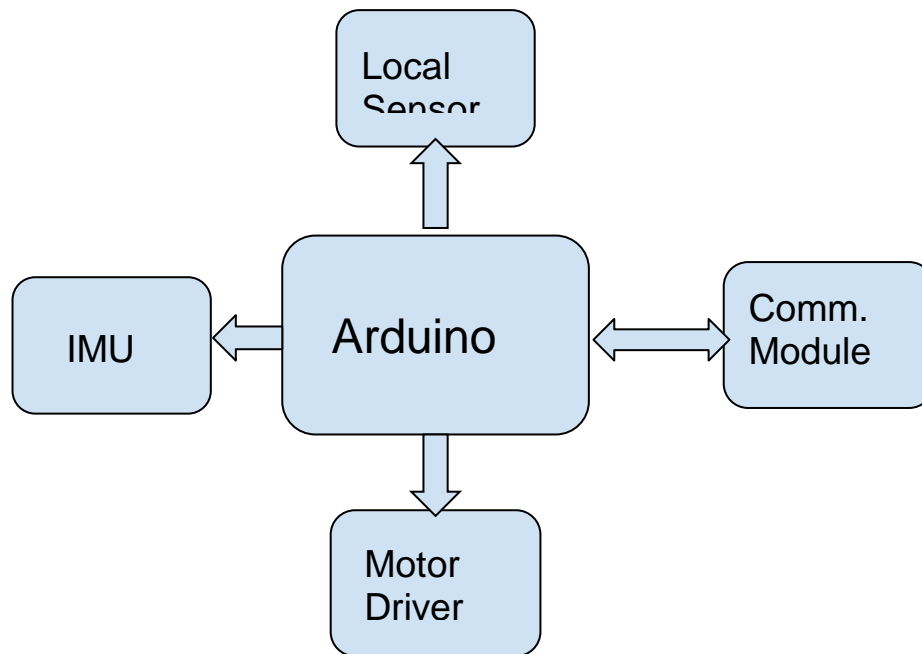
feedback sensor, pose correction and prediction, algorithmic learning and execution of learned instruction.

Arduino has been proposed to be used as the controller for the robot due to its superior hardware compatibility and ease of use. Besides that, a Raspberry Pi might also be used if the computational need is so high that the Arduino might be unable to solve it.

Sensors in the robots consist of IMU for pose tracking. This module will act as the feedback system to contribute a closed loop system in kinematics of each robots.

The remote transmission module will be required for communication. Several types have been proposed for this. The WiFi module and the RF module are the most probable candidates. The selection of the module will be done by weighing the practical demand by the system.

The central computer is proposed to be equipped with a low quality camera to track the global pose of the environment.



**figure 4:** Structure of each individual robots

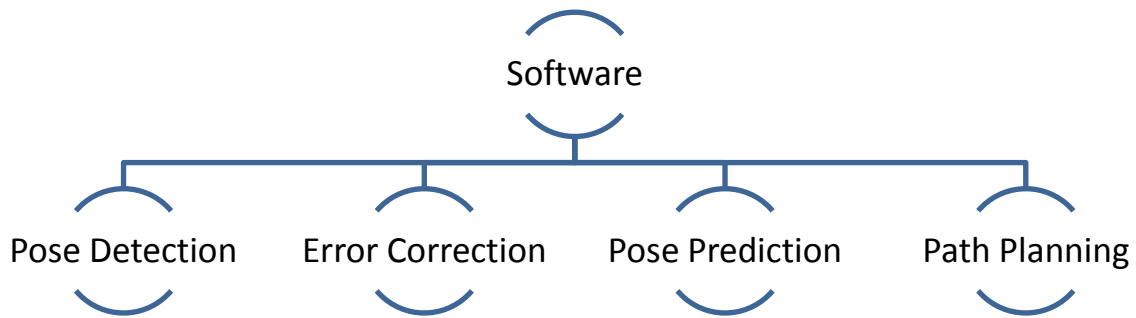
- **Arduino:** The Arduino is the gateway to hardware control. It is used to control each of the individual robots based on the instruction received from the central system. Besides that, it will govern the basic movement of the bot.

- **Communication Module:** The communication module is required for each individual agent to be able to communicate and correspond to the central system.
- **Motor Driver:** The motor driver is for the locomotion part of the system which will drive the desired DC motors for attaining the pose as instructed by the arduino.
- **IMU:** The IMU is for forming a closed loop kinematic system which is very important for efficient and accurate systems.

## 2.3 Software Aspects

The Software Aspects of the system can be divided into the Image Processing and Algorithmic Learning.

The Software Aspects of the project will include various modules of software codes written in the Python Programming Language for the central computer and Arduino-C Programming Language for the mobile robot. The modules include nodes for Pose Detection(Image Processing), Inverse Kinematic Path Planning and Pose Prediction.



**figure 5.** Software parts of the project

### 2.3.1 Learning System

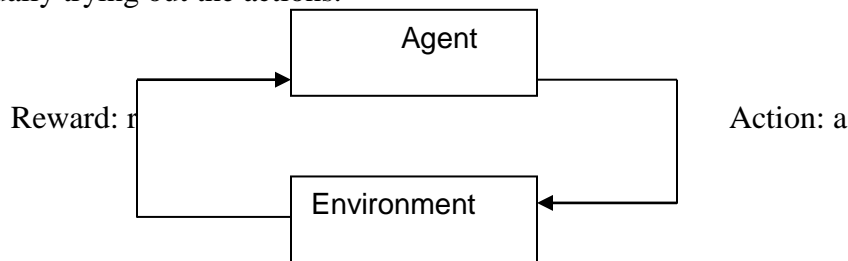
#### Markov Decision Processes(MDP):

A Markov decision process is a tuple  $(S, A, \{P(s,a)\}, \gamma, R)$  where

- $S$  is the set of states.
- $A$  is the set of actions.
- $P(s,a)$  are the state transition probabilities upon taking action  $a$  on states
- $\gamma$  is the discount factor.
- $R(s,a)$  is the reward function.

#### Reinforcement learning:

In reinforcement learning the system is modeled using MDPs and the state transition probabilities and the rewards are not known beforehand. The agent receives feedback in the form of reward for its action. The learning process tries to find a way for choosing actions over time so that the expected reward is maximized. The distribution  $T$  and reward function  $R$  are found out by actually trying out the actions.



*Figure 6: reinforcement learning.*

There are different types of reinforcement learning such as passive reinforcement learning where a fixed policy is input, and , and active reinforcement learning where the optimal policies is learned by actually taking actions. Passive reinforcement learning can be further categorized into model-based learning where the agents learns the approximate model and uses it, and model-free learning where the approximate model is not learned.



### 2.3.2 Image Processing

#### **Position Detection :**

To find the position of the robots as well as ball we can apply some image processing technique.

Since the robots have specific pattern we can apply pattern recognition algorithm. The possible approach which we can use are :

#### **Image segmentation :**

Image segmentation is the process of partitioning a digital image into multiple segments (sets of pixels, also known as super pixels). The goal of segmentation is to simplify and/or change the representation of an image into something that is more meaningful and easier to analyze. Image segmentation is typically used to locate objects and boundaries (lines, curves, etc.) in images. More precisely, image segmentation is the process of assigning a label to every pixel in an image such that pixels with the same label share certain characteristics.

The result of image segmentation is a set of segments that collectively cover the entire image, or a set of contours extracted from the image. Each of the pixels in a region are similar with respect to some characteristic or computed property, such as color, intensity, or texture. Adjacent regions are significantly different with respect to the same characteristics.

#### **Template Matching :**

*Template matching* is a technique in digital image processing for finding small parts of an image which match a template image. It can be used in manufacturing as a part of quality control, a way to navigate a mobile robot, or as a way to detect edges in images.

#### ***Feature based approach***

If the template image has strong features, a feature-based approach may be considered; the approach may prove further useful if the match in the search image might be transformed in some fashion.

#### ***Template based approach***

For templates without strong features, or for when the bulk of the template image constitutes the matching image, a template-based approach may be effective. As aforementioned, since template-based template matching may potentially require sampling of a large number of points,

it is possible to reduce the number of sampling points by reducing the resolution of the search and template images by the same factor and performing the operation on the resultant downsized images (multi resolution, or Pyramid (image processing)), providing a search window of data points within the search image so that the template does not have to search every viable data point, or a combination of both.

### **Patch Comparison and Matching :**

It is not different image processing approach but it helps us to find out the orientation and position of the objects very efficiently and more accurately. We will use simple colored patches embedded on robot whose position can be calculated accurately. To detect orientation, multiple patches can be employed. For detection, the simplest approach to take for a single patch is to use a simple regular geometric shape of a single color. Detection in the vision system can be carried out on a binary or multi-class threshold image from which connected regions of common color class can be extracted. This approach is common, and is known to be quite efficient, so this is the approach we will use. The next variable to determine for a patch shape. We chose circles, because they guarantee rotational invariance, and analytical corrections for the projective distortions of their image centroids are known. In addition, they are compact, minimizing the length of the border with other regions, where thresholding is most difficult. In experiments, other regular shapes such as squares, hexagons, and octagons, perform roughly on par with circles, but they do not offer any benefits in light of the analytical guarantees outlined above.

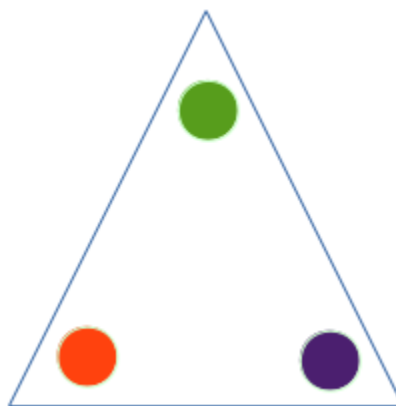


Figure 7 : Robot Shape with different colored patches

### **3. PROJECT REQUIREMENTS**

#### **3.1 Software Requirements:**

- Text Editor
- Python Programming Language Interpreter
- Compiler Toolchain
- git Version Control Software
- Github
- Proteus Professional
- Fritzing
- Arduino IDE
- Various Algorithm toolkits
- Open-Source Libraries

#### **3.2. Hardware Requirements:**

(For 2 robots)

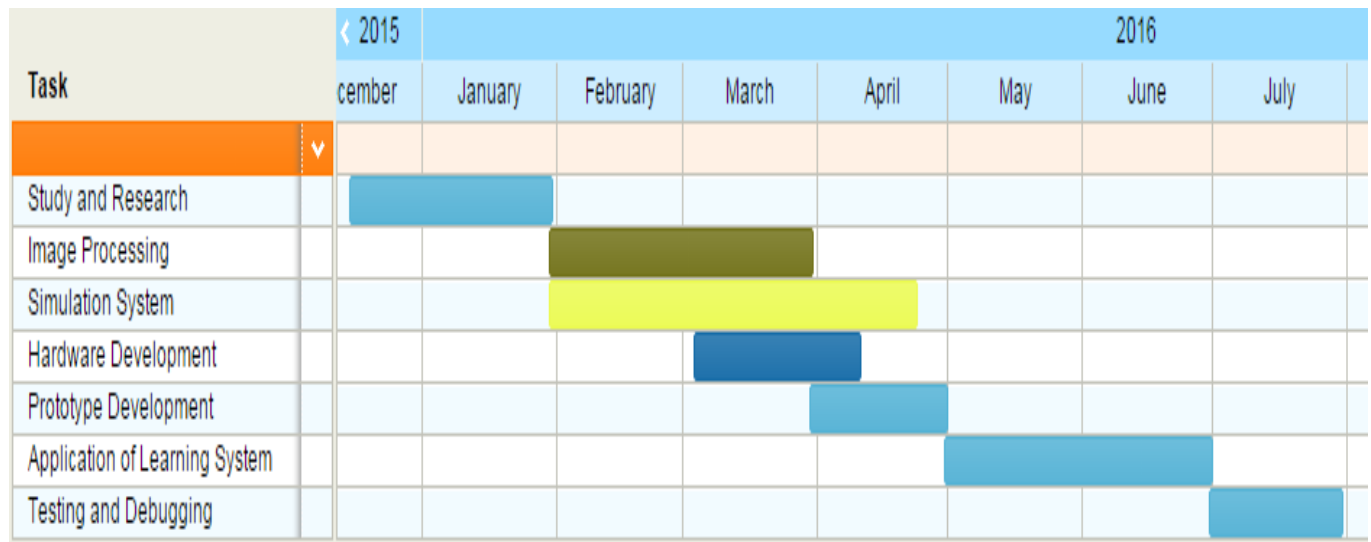
- DC Motors x4
- Dual-Bridge Motor Driver Modules x4
- Arduino x2
- IMU x2
- Central PC x1
- Camera x1

#### **4. EXPECTED OUTCOME**

The project expects to be able to design and develop a system of multiple robots which is able to collectively optimize the solution to the game of football. The system primarily will be expected to be able to track a ball and collectively guide it to the specified area of the field(1-to-1 passing).

## 5. TIME SCHEDULE

The following is the proposed time schedule for the completion of the project.



**figure 8.** Project Schedule Gantt Chart

## 6. COST ESTIMATION

S.No.	Component	Qty.	Rate(NRs.)
1.	Arduino	2	2000
2.	IMU	2	2000
3.	Communication Module	2	2000
4.	Motor Driver	4	3000
5.	DC Motor	6	4800
6.	Robot body and wheel	2	8000
7.	Camera	1	2000
8.	Battery(Li-Po)	2	4000
9.	Misc.		4000
Total			31800

## 7. REFERENCES

- [1] P. Tangamchit, J. Dolan, P. Khosla "Crucial Factors Affecting Cooperative Multirobot Learning".
- [2] S. Choudhury "Application of Reinforcement Learning in Robot Soccer", 2011
- [3] M. Riedmiller, T. Gabel, R. Hafner "Reinforcement learning for robot soccer", 2009
- [4] L. Parker "Current research in multirobot systems", 2003
- [5] R. Brunelli, Template Matching Techniques in Computer Vision: Theory and Practice, Wiley, ISBN 978-0-470-51706-2, 2009 ([1] TM book)
- [6] Aksoy, M. S., O. Torkul, and I. H. Cedimoglu. "An industrial visual inspection system that uses inductive learning." *Journal of Intelligent Manufacturing* 15.4 (August 2004): 569(6). Expanded Academic ASAP. Thomson Gale.
- [7] E. H. Adelson, C. H. Anderson, J. R. Bergen, P. J. Burt and J. M. Ogden, Pyramid methods in image processing [http://web.mit.edu/persci/people/adelson/pub\\_pdfs/RCA84.pdf](http://web.mit.edu/persci/people/adelson/pub_pdfs/RCA84.pdf)