2-DIMENSIONAL MULTI-ROBOTS OBJECT DETECTION SIMULATION

By

MUHAMMAD NAJMI BIN MOHAMAD ZAHIDI

10850

BUSINESS INFORMATION SYSTEM

Dissertation submitted in partial fulfillment of

The requirement of the

Bachelor of Technology (Hons)

(Business Information System)

JANUARY 2011

Universiti Teknologi Petronas

Bandar Seri Iskandar

31750 Tronoh

Perak Darul Ridzuan

CERTIFICATION OF APPROVAL

By

MUHAMMAD NAJMI BIN MOHAMAD ZAHIDI

A project dissertation submitted to the

Business Information System Programme

Universiti Teknologi Petronas

In partial fulfillment of the requirement of the

Bachelor of Technology (Hons)

(Business Information System)

Approved by,

____

(Dr. Jafreezal b. Jaafar)

CERTIFICATION OF ORIGINALITY

This is to certify that I am responsible for the work submitted in this project, that the original work is my own except as specified in the references and acknowledgements, and that the original work contained herein have not been undertaken or done by unspecified sources or persons

MUHAMMAD NAJMI BIN MOHAMAD ZAHIDI

ABSTRACT

The study and research regarding robot playing soccer which is famously known as RoboSoccer has increasingly gain attention among researchers all around the world. The goal of RoboSoccer is to create and build a team of robots (multi-robots) to play with real human being. There are a lot of tasks involve in order to prepare and to create RoboSoccer team such path finding, physical movement and object detection. In this particular project, it will focus only on developing simulation of object detection in RoboSoccer.

In chapter one, it will briefly explain the background of the study and state the problem statement for this project as well as setting up the aim, objectives and scope of study. In chapter two, literature review relating to this project will be discussed and review which based on Aibo Robot Object Detection and Fuzzy Logic. The methodology used in this project will be explained in detail in chapter three. Besides that, in this chapter also will display the timeline of the project, project activities as well as tools required in order to develop and complete the project.

In chapter four, the project will be discussed in detail from the data gathering and analysis until the result and discussion of the simulation testing. Object and players in simulation, experimentation, implementation and prototype will be also discussed in this chapter. This project will later be conclude in chapter five which will link the relevancy of this project to the objectives and suggest some of the future works and recommendations for this project in order to improve the simulation for future purposes

ACKNOWLEDGMENT

By grace of Allah S.W.T., The Almighty, this research was completed within the stipulated time.

I would hereby wish to express my sincere appreciation and gratitude to Dr. Jafreezal b. Jaafar for the on-going dedication, commitment and valuable guidance in making this research a success. An ocean of thanks again to my devoted lecturers for their encouragement and having belief in my potential in conducting this research.

To my fellow colleagues, despite being highly occupied in completing their personal final year projects, the thought to being munificently openhearted and openhanded are greatly appreciated. My special thanks goes to my colleagues, Syed Akmal b. Syed Othman and Wan Muhammad Nasrullah b. Wan Mansor for their support and sharings in times of hardships and obstacles throughout the journey until the finishing line in completing this research.

I would also would like to convey my deep-hearted appreciation to my beloved parents and my family members, especially my dearest mother and father for their incomparable love, continuous moral support, counseling and guidance in ensuring that as learning individual to always do my best and to reap success in spite of all the shortcomings and challenges.

Finally, a huge vat of thanks to those who had directly or indirectly helped me throughout this challenging yet precious journey

TABLE OF CONTENTS

LIST OF FIGURES

CHAPTER 1

INTRODUCTION

1.1 Background of Study

Technology relating to robotics recently has been widely used due to its benefits such as low budget research to help humans in doing specific or general tasks. Human tends to avoid or delegate hard tasks to other people instead of doing it by themselves. This is where robotics technology comes in whereby robots are used to replace human in doing tasks. All tasks involve decision making and robots are programmed to make decisions accordingly and correctly. One of the studies or research relating to robotics technology is RoboSoccer. RoboSoccer is basically robots playing soccer. There are groups of people trying to form a team consisting multi-robots to play soccer with a long term target which is to play with real human. In RoboSoccer, there are many tasks and actions to perform such as physical movement, path findings and object detections.

Object detection is one type of decision making relating to robotics technology. Until now, it still remains as mystery how human perceive objects so accurately and with so little apparent effort. Humans have the capability to detects or differentiate objects easily but not robots. They are not aware of the surrounding or environment which makes it hard for them to identify objects. Human brains somehow can differentiate object by knowing the object, used to the object (indirectly trained) and can differentiate the object with the environment and other objects. All these decision making can be process by human brains. Whereas for robots, all these processes need to be handle first and separately before they can detect object. Therefore, objects detection and recognition then is introduced in robotics technology.

In this particular project, it will focus on object detection. It will demonstrate how multi-robots interact and make decision in an environment to detect objects.

1.2 Problem Statement

The problems of multi robots in object detections are:

(1) Conflict in detection between robots and object/environment

(2) Action selection and reaction selection in detecting object

(3) Integration of the multi robots in detecting objects

1.2.1 Significant and Relevancy of the Project

As mention in the problem statement, this proposed project is based on the problems addressed in detecting and recognizing objects. In this particular project, a simple simulation will be developed and build to show how object can be identify or detect. This simulation is developed by using a form of sports, which is soccer in order to perform and prove the object detection process. The reason soccer is chosen is because in this sport, there are a lot of object detection activities which can be used in this project which can deliver and prove object detection in an understandably and interesting manner. In this project the player is referred as 'robot' and the ball will be referred as 'object'.

1.2.2 Problem Identification

There are a lot of research currently being done in developing and creating soccer robots or famously known as RoboSoccer. The goal of the researchers is to development a robot which can make decision as close as human brains does. Once they succeed, they want to put their robots and humans in a robot versus human match. In order to achieve this, there are a lot of researches and studies need to be done which one of them is object detection. Therefore, object detection is important and vital in reaching that goal and on this particular project, it will focus and discuss on object detection.

1.3 AIM, OBJECTIVES & SCOPE OF STUDY

1.3.1 Aim

To develop a simulation using vector concept object detection algorithm for multi-robots object detection.

1.3.2 Objectives

(1) To implement the vector concept with object detection algorithm in detecting object

(2) To use multi-robots in the same environment in detecting objects

(3) To test the detection rate of the multi-robots with the objects

1.3.3 Scope of Study

This project will be developed in simulation and focus on three main forms of object detection:-

a) Robot and Object

b) Object and Environment

c) Robot and Environment

CHAPTER 2

LITERATURE REVIEW

2.1 "GENERIC OBJECT DETECTION FOR AUTONOMOUS ROBOTS (AIBO)"

Figure 1: Aibo Robot

2.1.1 Introduction

This thesis paper was presented by University Autonoma of Barcelona in 2007 which discusses the generic object detection system for autonomous robots simulated in the Aibo Robot of Sony. It focus on Aibo detection process in different kind of algorithm to detect Aibo reaction and interaction as well as behaviors

2.1.2 Aibo Robot

Aibo is derived from **A**rtificial **I**ntelligence Ro**BO**t. It was designed and manufactured by a Japanese company, SONY. Aibo is unique because it can walk, "see" its surroundings via camera and be able to recognize and detects spoken commands in different languages. Aibo also can learn and mature based on external stimuli from owner or environment or from other Aibos. The reason why Aibo robot is the perfect subject or tool to implement and test artificial intelligent techniques in robotics is because Aibo Robot combines body (hardware) and mind (Aibo Mind 3 software) that enables it to move, think and learn reaction pattern and grows (improve its future judgments). There is no other developed technology in one simple but intelligent robot.

2.1.3 Aibo Robot Object Detection

Based on the thesis it mentioned that, object recognition process is basically composed of three steps which are [1] region of interest (ROI), [2] model matching and [3] classification. All these steps can be done by different number of algorithms depending on the object we want to recognize because, different techniques offers different performance. In this thesis it uses three algorithms which are mainly focusing on fast object detection (robust):-

(a) Adaboost Detection

Adaboost is a machine learning algorithm which derived from Adaptive boosting which is normally being used parallel with other learning algorithms to improve performance. This thesis uses Gentle Adaboost, a modified version of the Real Adaboost algorithm as it combines the algorithm with Newton stepping. Gentle Adaboost algorithm is shown as below:

Gentle AdaBoost

1. Start with weights $w_i = 1/N$, $i = 1, 2, \ldots, N$, $F(x) = 0$.
2. Repeat for $m = 1, 2, \ldots, M$:
   (a) Fit the regression function $f_m(x)$ by weighted least-squares of y_i to x_i with weights w_i.
   (b) Update $F(x) \leftarrow F(x) + f_m(x)$
   (c) Update $w_i \leftarrow w_i e^{-y_i f_m(x_i)}$ and renormalize.
3. Output the classifier $\text{sign}[F(x)] = \text{sign}[\sum_{m=1}^{M} f_m(x)]$

Figure 2: Gentle Adaboost

The challenge is Adaboost algorithm is to associate a large weigh with each good classification and a smaller weigh with poor functions. Adaboost is an aggressive mechanism for selecting a small set of good classification functions and features which at least have a significant variety. It also provides an effective learning algorithm and strong bounds on generalization performance which increase the speed of the detector by focusing attention on promising regions of the image by combining successively more complex classifiers in a cascade structure.

(b) Detectors Cascade

Detectors Cascade is an extension from the Viola and Jones Framework in object detection. It is a degenerated decision tree where at each stage a detector is trained to detect almost all objects of interest while rejecting a certain fraction of the non-objects patterns. Figure below display the detectors cascade algorithm.

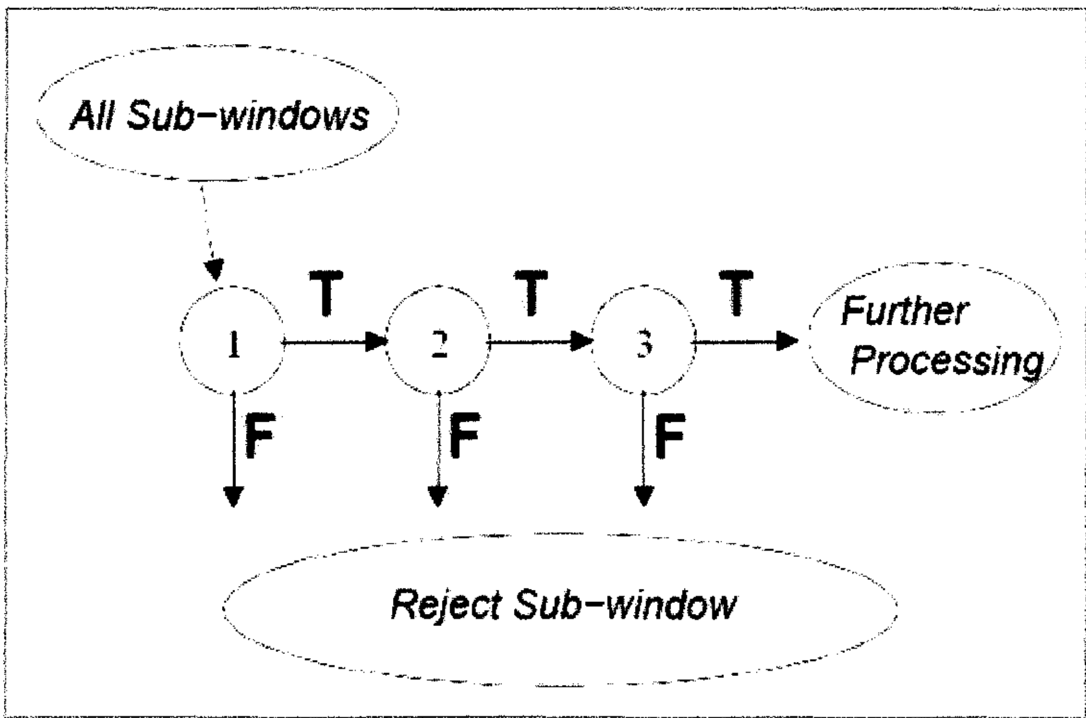

Figure 3: Detectors Cascade

Detection system must certify solid exception or restrictions both on true and false alarm ration. By using formula such that false alarm, f, is fixed for each stage of the cascade and n, is number of stages, it can be applied in determining false alarm ratio which can be applied to true alarm ration too. At each stage, it analyzes only the objects accepted by the previous stages until they are rejected by a detector.

(c) Haar-Like Features

In robust detector, Haar-Like can used instead of image pixels by finding the differences between the sum of the pixels in some contiguous rectangular regions of the image. Lienhart and Maydt extended the Haar-Like Features set used by Viola and Jones Framework by adding the rotated versions of each feature.

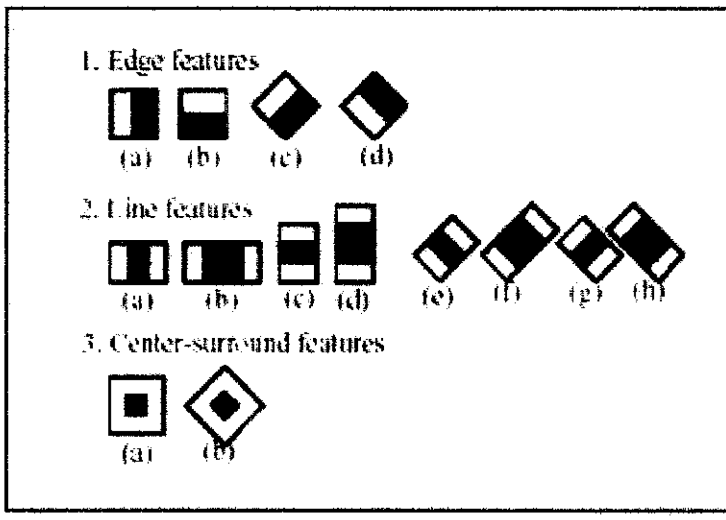

Figure 4: Haar-Like Features

In figure 4, for example in Line Features, (a)=(c)=(e)=(g) and (b)=(d)=(f)=(h), all these are the same object and it just being rotate. Same applied in edge feature and center-surround features. All these features are then being calculated using

(1) Integral Image or Summed Are Table (SAT1)

In SAT1 image, each pixel SAT(x, y) contains the sum of all pixels of the upright rectangle ranging from top-left corner to the bottom-right corner at (x, y).

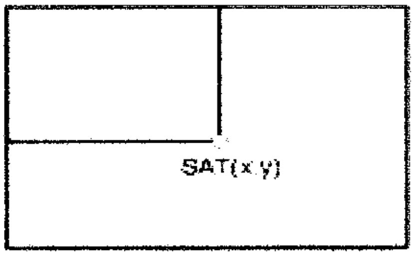

Figure 5: Summed Are Table (SAT1)

(2) 45° Rotated Integral Image or Rotated Summed Area Table (RSAT2)

In RSAT2 image, the image is defined as the sum of the pixels of a 45° rotated rectangle with the bottom most corner at (x, y) and extending upwards till the boundaries of the image.

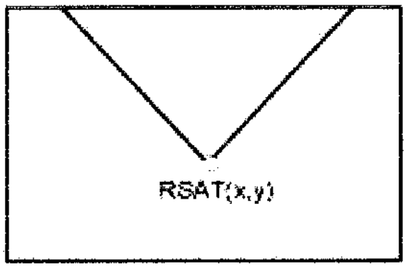

Figure 6: Rotated Summed Are Table (RSAT2)

By using these two methods of calculations of feature based in Haar-Like Features, the object detection rate is much faster than a pixel-based system.

2.2 FUZZY LOGIC

2.2.1 Introduction

This concept of fuzzy logic is taken from 'Fuzzy Implications' and 'Type-2 Fuzzy Logic and Applications' books written by Springer-Verlag Berlin Heidelberg. Based on these books, Fuzzy logic is a form of multi-valued logic derived from fuzzy set theory to deal with reasoning that is approximate rather than accurate which is in contrast with 'crisp logic'. Fuzzy logic variables may have truth value that ranges between 0 and 1 and not 0 or 1 because it is based on approximation. Usually, fuzzy logic is being used in system control and analysis design due to its advantage which can shortens the time of development. Fuzzy logic also can be said, is the way the human brain works and can mimic this in machines so they will perform actions like humans. Thus, to achieve good performance in motion control of robots, one of the ways is to adopt fuzzy logic or fuzzy algorithm.

2.2.2 Fuzzy Logic Method

Fuzzy logic method comprises four main steps which are:

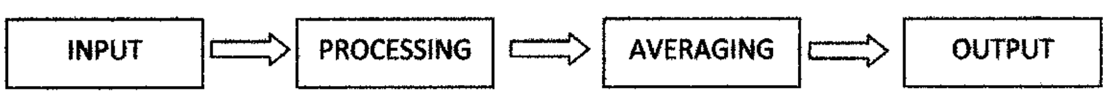

Figure 7: Fuzzy Logic Method

(1) INPUT

Measurement or assessment of system conditions (i.e temperature, height)

(2) PROCESSING

Determine action to be taken based on human determined fuzzy "If-Then" rules with non fuzzy rules

(3) AVERAGING

Determine center of mass for all system conditions

(4) OUTPUT

Decision

2.2.3 Fuzzy IF-THEN Rules

As mentioned before, fuzzy logic is based on approximation. Approximate reasoning is used in addressing the fuzzy logic approximation. It refers to the methods and methodologies that enable reasoning with imprecise inputs to obtain meaningful outputs. Crucial to these conditions, one way to realize approximate reasoning is to apply Fuzzy IF-THEN rules. A fuzzy IF-THEN rules is of the form:

IF x is A THEN $\hat{y}$ is B

Where x, y are linguistic variables and A, B are linguistic expression or values assumed by the linguistic variable. For example,

IF *volume(x)* is *high (A)*,

Then *pressure(y)* is *low (B)*

As for this particular project, some behaviors in soccer robot have been identified in terms of object detection which can apply the fuzzy logic.

(1) **Tracking**(robot move to target position)

IF obstacle IS none between goal and ball AND between target and robot

THEN robot approach target position

(2) **Shooting** (robot move to ball towards goal)

IF ball IS at opponent side AND no obstacle between robot and ball

THEN robot approach ball

(3) **Defending** (robot move to ball outside from goal)

IF ball IS at home side AND no obstacle between robot and ball

THEN robot approach ball

2.3 Review & Findings of the Literature Review

From these literature reviews, the author thinks that this thesis paper really helps in giving the basic idea and a helicopter view on object detection. The research paper of *"GENERIC OBJECT DETECTION FOR AUTONOMOUS ROBOTS (AIBO)"*, in object detection, it involve a lot of ways or algorithm in performing object detection which also depends on the detection rate whether a normal detection or robust detection. In other words, different algorithm produced different result but still in the object detection context. Even though, this literature review regarding the Aibo Robot is a little advanced due to its method of using camera which installed in the robot to detect object, it still helps the author in terms of what to look for and how to perform an object detection process. Therefore, this literature review has given a lot of insights in doing this project.

The Fuzzy Logic from *'Fuzzy Implications'* and *'Type-2 Fuzzy Logic and Applications'* books written by Springer-Verlag Berlin Heidelberg explain the fuzzy logic theory. In this project, approximation will be used a lot thus fuzzy logic is suitable. It will based on approximate reasoning in addressing the fuzzy logic approximation by using the Fuzzy IF-THEN rules which will help in detecting objects.

CHAPTER 3

METHODOLOGY

3.1 Research Methodology

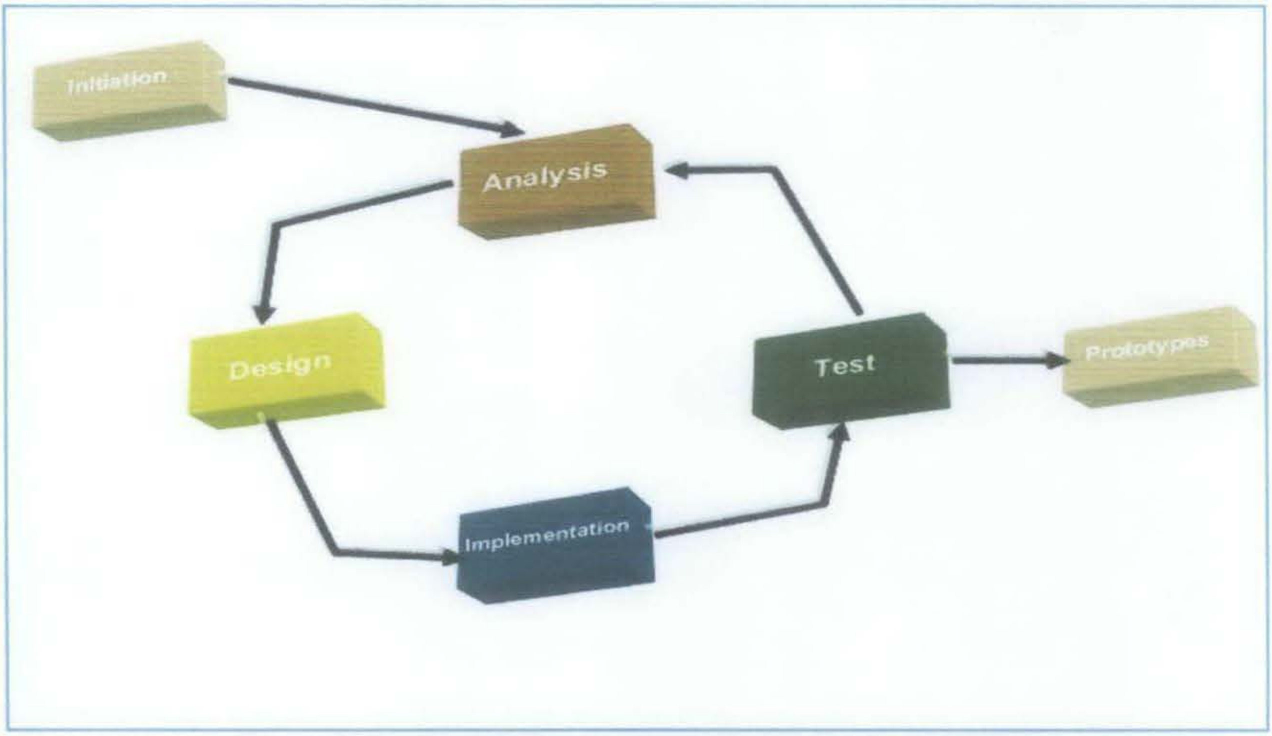

Figure 8: Project Methodology: Evolutionary Prototyping

Since there is no way in certain to know this project flow or process versus the desired outcome, one of the best method to use to perform this project is by using prototyping. There are many types of prototyping and the type of prototyping plan to be used is evolutionary prototyping. Evolutionary prototyping is a method to build a fast or robust prototype in a proper manner and constantly improved it. This method allows modifications such as adding, removing new features during the development. The analysis, design, implementation and testing phases performed concurrently and on each cycle resulting in the required prototype. Evolutionary prototyping is been choose due to incomplete analysis of the project, unexpected problems that might occur during the development and time as well as cost constraints.

3.1.1 Initiation

Initiation phase is the fundamental process of understanding on why the project needs to be built and determining on how the management process of the project would go. Apart from that, in this phase, the project title is determined and how the project will give value. In this case, problem of object detection is being addressed. Based on the problem identified, then the project title is proposed.

3.1.2 Analysis

Analysis phase is where the research is being done on current developed system. Current system design and concepts used is being identified and researches are made. In this case, research on object detection is been carried out. Focus are given on the scope of the project, concept of object detection, algorithm being used (different algorithm, different purpose), module function, parameters and programming language to be used. From the data gathered, analysis on the data is conducted and tries to relate to this particular project.

3.1.3 Design

The design phase is how the system will operate, in terms software, interface and logic manipulation. For this project, at the moment, this project design is just at the simulation stage. In the simulation, there will be a lot of interfaces need to be develop which have different functions based on different environment and situation of the object detection process

3.1.4 Implementation

This is the stage before the prototype being tested and then delivered. The prototype is build based on the deliverables in the analysis phase such as algorithm and logic that is decide to be used and in the design phase such as interface design. All these deliverables are then being implemented to come out with a prototype.

3.1.5 Testing

At this phase, the prototype created at the implementation phase will be tested. This testing phase will test whether the desired output is achieved or not. If the desired output does not achieve, the prototype will be discard and modification will be done through the analysis until the new prototype is created correctly and produce all the desired output.

3.2 Project Activities

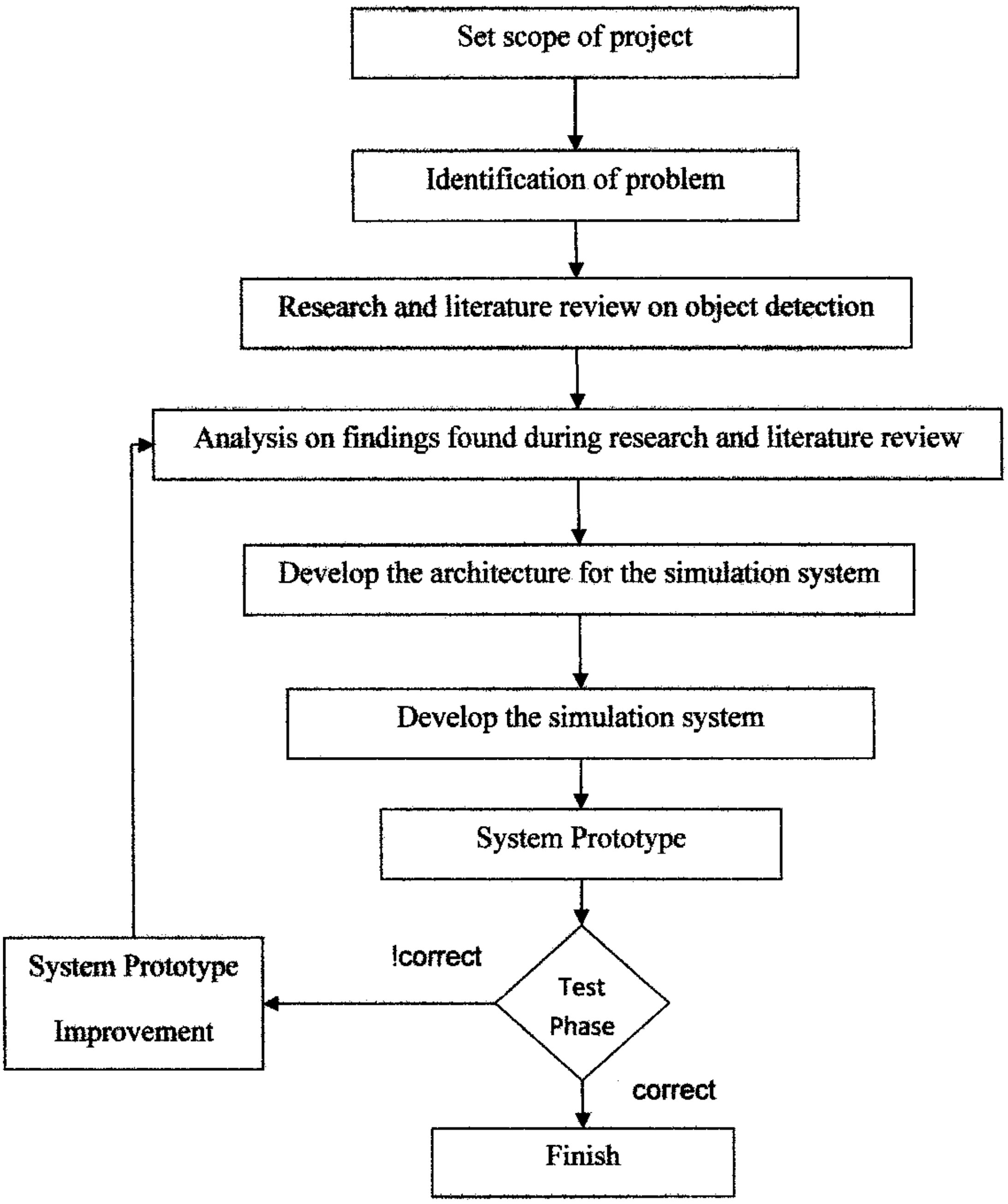

Figure 9: Project Activities

3.3 Project Gantt Chart

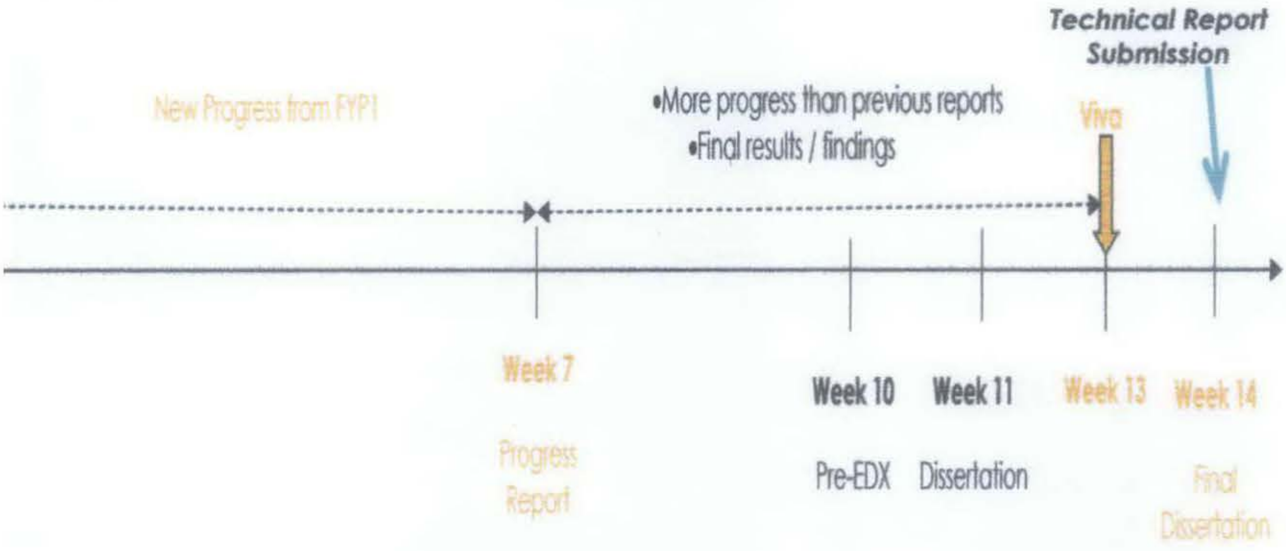

Figure 10: Project Gantt Chart

3.4 Tools Required

As for this project, it needs both software and hardware to assist and complete the project. The tools are divided into software and hardware.

3.4.1 Software

A) Microsoft Visual Studio

Microsoft Visual Studio will be used in developing the simulation part of this project due to its ability to cater all the needs in developing the simulation.

CHAPTER 4

RESULT AND DISCUSSIONS

4.1 Data Gathering & Analysis

4.1.1 Coordinate Concept

Coordinate concept is a concept used to pin point an exact location or position, to manipulate frames and movement of objects. In a window frame there is an invisible coordinate that defines each n every position in that frame. As this project will only use 2 Dimensional, thus it will consist only X-axis and Y-axis.

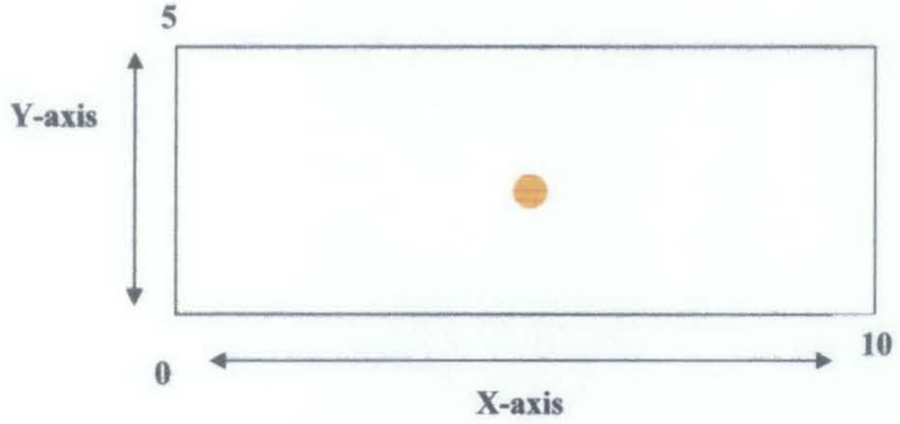

Figure 11: X-axis & Y-axis

The red dot position in this frame is define as (5,3) which 5 correspond to X-axis n 3 correspond to Y-axis. This concept is useful in defining position for any objects in a frame. The coordinate concept will help in deciding and controlling object position in a frame.

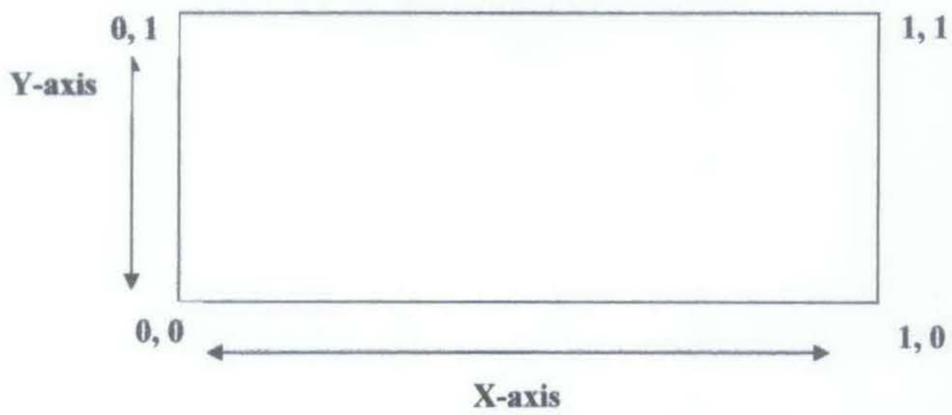

Figure 12: Coordinate in one frame

Figure 12 shows how coordinate is defined in one particular frame. Each frame will have 4 points and to manipulate the frame, these points need to be altered.

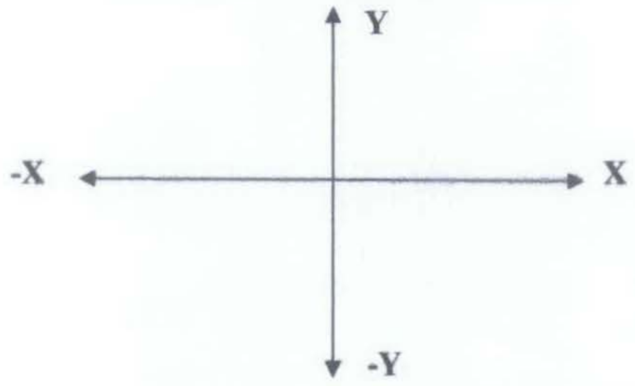

Figure 13: Four Quadrants of X-axis & Y-axis

Figure 13 shows the 4 quadrants of the X-axis and Y-axis. There is a positive and negative value in each axis which correspond the movement of objects along the axis. When an object moves to the right and/or up, it has a positive value on every axis while if it moves to the left and down, it has negative value. Thus, by manipulating the positive and negative value of the axis on the objects, the movement of the object can be determine.

4.1.2 Vector & Scalars Concept

Vector is a value or quantity that has two components which are magnitude and direction at the same time while scalar is a quantity the only has magnitude with no specific direction. Magnitude is the size and strength of the object. This concept is widely used in developing the simulation as it involves different magnitude and directions. Vector can be used graphically in many dimensions which in this case, it is two dimensions (2D).

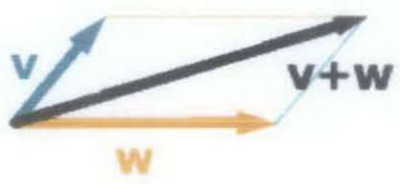

Figure 14: Components of Vectors

Vector is generally oriented on a coordinate system with different dimensions or with Cartesian plan. The Cartesian plane has X-axis and Y-axis. In 2-Dimensional, a vector is broken into their component vectors. Based on figure 13, **(V+W)** is broken into its components, **V** and **W.** When breaking into its components, the vector direction is calculated by:

$$(\mathbf{V}+\mathbf{W}) = \mathbf{V} + \mathbf{W}$$

The magnitude of the vector is calculated by:

$$|\mathbf{V}+\mathbf{W}| = \sqrt{(\mathbf{V}^2 + \mathbf{W}^2)}$$

The coordinate concept and vectors and scalar concept is the two concept which will be widely used in developing the simulation as it will contribute to the size and shape of the frame and objects, movement of the object, collision between objects and the speed of the objects.

4.1.3 Object Detection Algorithm

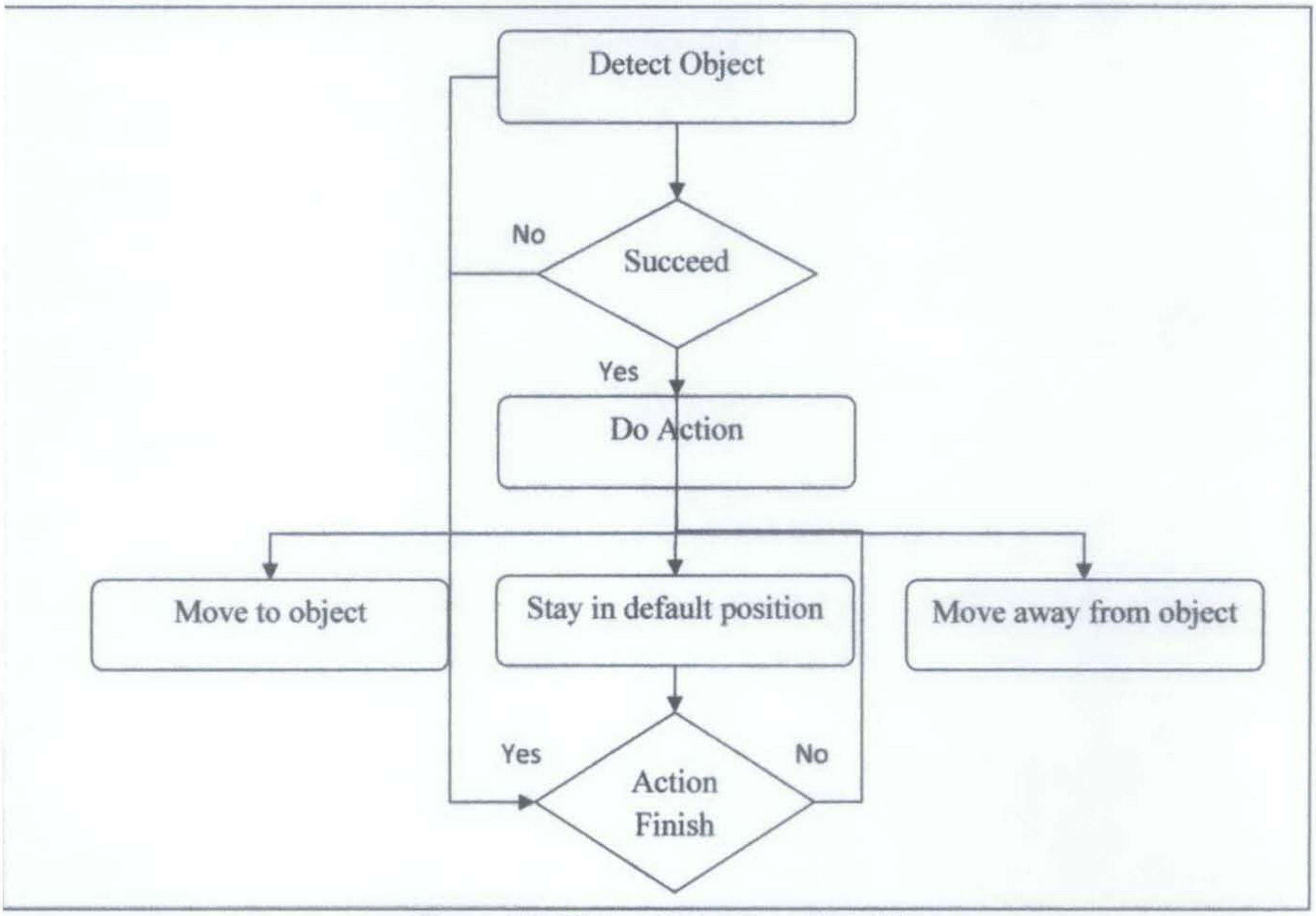

Figure 15: Object Detection Algorithm

4.2 Objects & Players in Simulation

4.2.1 Object Classification & Regions

The object is classified into two groups which are positive and negative object. The reason to have two different objects is to make sure that the subject can detect the object correctly.

(a) Positive Object

Positive object is the object that the subject needs to detect. For this project, the subject needs to detect ball which is round in shape.

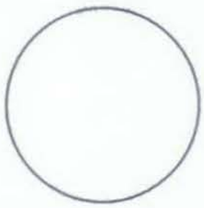

Figure 16: Positive Object (White)

(b) Negative Object

Negative object is the object that the subject needs no detection at all. In this case, any other object that has different shape besides round shape can be used.

Figure 17: Negative Object (Purple)

The region used is a mix region whereby positive and negative object a put together and the robot should only detects positive object.

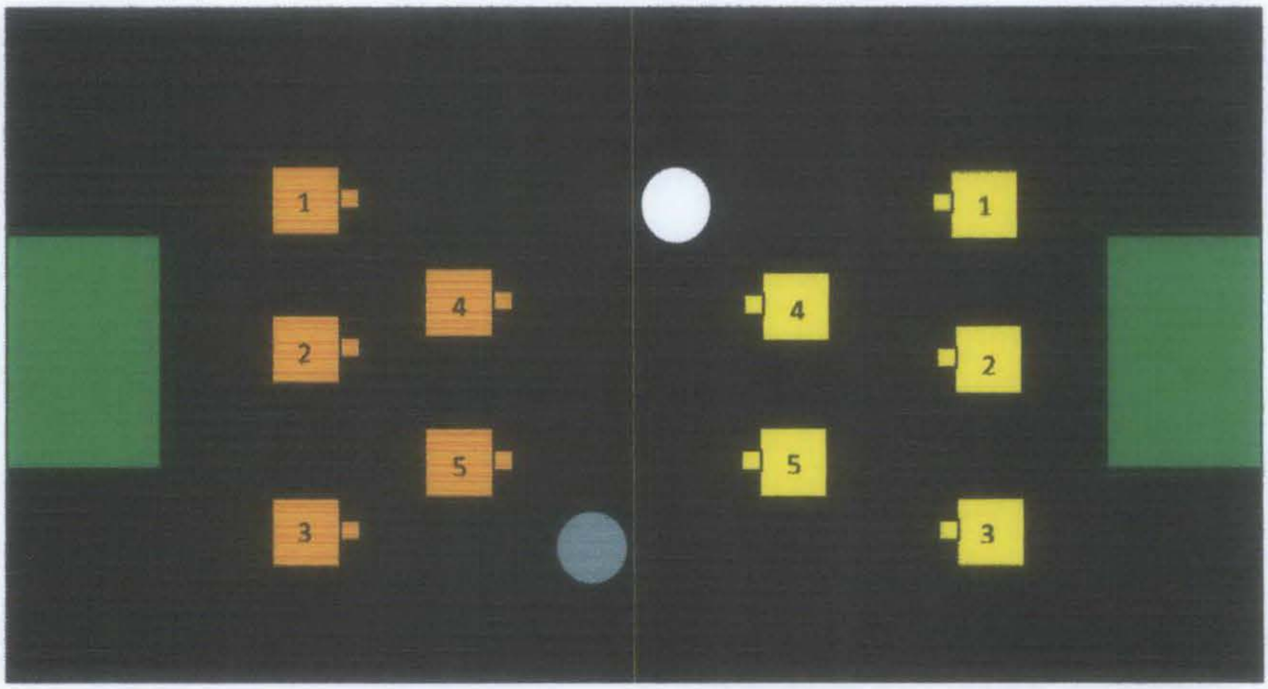

Figure 18: Static View of the simulation

Team A: RED

Team B: YELLOW

Red Line: Half line

Goal Post: Green

Window Size: 800x600(blue)

Field Size: 750x550(black)

4.2.2 Summary Roles of the Robots & Objects

(1) Robots should only detects positive object (white ball)

(2) Robots should ignore negative object (purple ball)

(3) Robot 1, 2, 3 from each team will only react when positive objects enter their own half field. When positive object is out of their own half, all robot back to default position.

(4) Robot 4 from each team will continuously detect object regardless of object position.

(5) Robot 5 from each team will do nothing. Stay in default position

(6) Positive object will return back to default position when enter goal.

4.2.3 Object Movement

They movement of the object is key in this project, not the robots. The object moves first only then the robots will follow. It is done by declaring the object as a vector which will give it 2 values which is X and Y. Then, two values are assigned (value X, value Y). Result of this will make the object moves. The current position of the object will always change when it moves.

```
typedef struct Pemain{
Vect direction;
}Pemain;

Vect direction;

direction.x = 1.0; //ball movement
direction.y = 1.0;

currentPos.x=395.0;//current ball position
currentPos.y=325.0
```

Figure 19: Object Vector Declaration

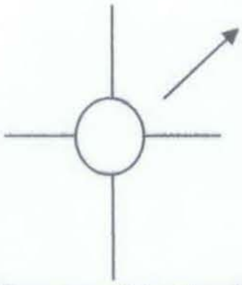

Figure 20: Object Movement in the four quadrants concept

Based on the four quadrants mentioned before, the arrow in figure 20 shows the direction of the object will start to move. One problem encountered during this action is the object will go through the specified frame and then lost. Thus, a force value has been assigned to force it back inside the frame.

```
if(currentPos.x>750 ){

      currentPos.x=730;

}

if(currentPos.y>550 ){

      currentPos.y=530;

}
```

Figure 21: Force Value Constraints

4.2.4 Collision Detection

In this simulation, the collision detection concept also is applied. There are two types of collisions which are:

(a) Collision between object and robots

When object and robots are in contact or touch each other, the object will collide or change direction. The coding snippet is as below:

```
double distanceX = currentPos.x - teamA[i].position.x;
double distanceY = currentPos.y - teamA[i].position.y;
double dist = sqrt(distanceX*distanceX+distanceY*distanceY);
//distance between ball & player

if (dist<20+15){//20=inner ball 15 radius player(to calculate collision)

      direction.x = direction.x + teamA[i].direction.x;
      direction.y = direction.y + teamA[i].direction.y;
double normaldir = sqrt(direction.x*direction.x+direction.y*direction.y);

direction.x/=normaldir;
direction.y/=normaldir;

direction.x*=2.25 ;
direction.y*=2.25 ;
```

Figure 22: Collision between robot and object

(b) Collision between object and environment

This type of collision is to prevent the object to go outside the field (black frame). The object will hit four walls and will change the direction or in the other words bounce back inside the field which will result the ball will always be inside the field. Below is the coding snippet for the upper wall being hit.

```
if(currentPos.y>530){//("Upper wall being hit!!\n");
   double dotp = direction.x*wallNUpper.x + direction.y*wallNUpper.y;
   Vect temp;
   temp.x = 2*wallNUpper.x*dotp;
   temp.y = 2*wallNUpper.y*dotp;
   direction.x = direction.x - temp.x;
   direction.y = direction.y - temp.y; //change_direction(direction);

   }
```

Figure 23: Collision between object and wall

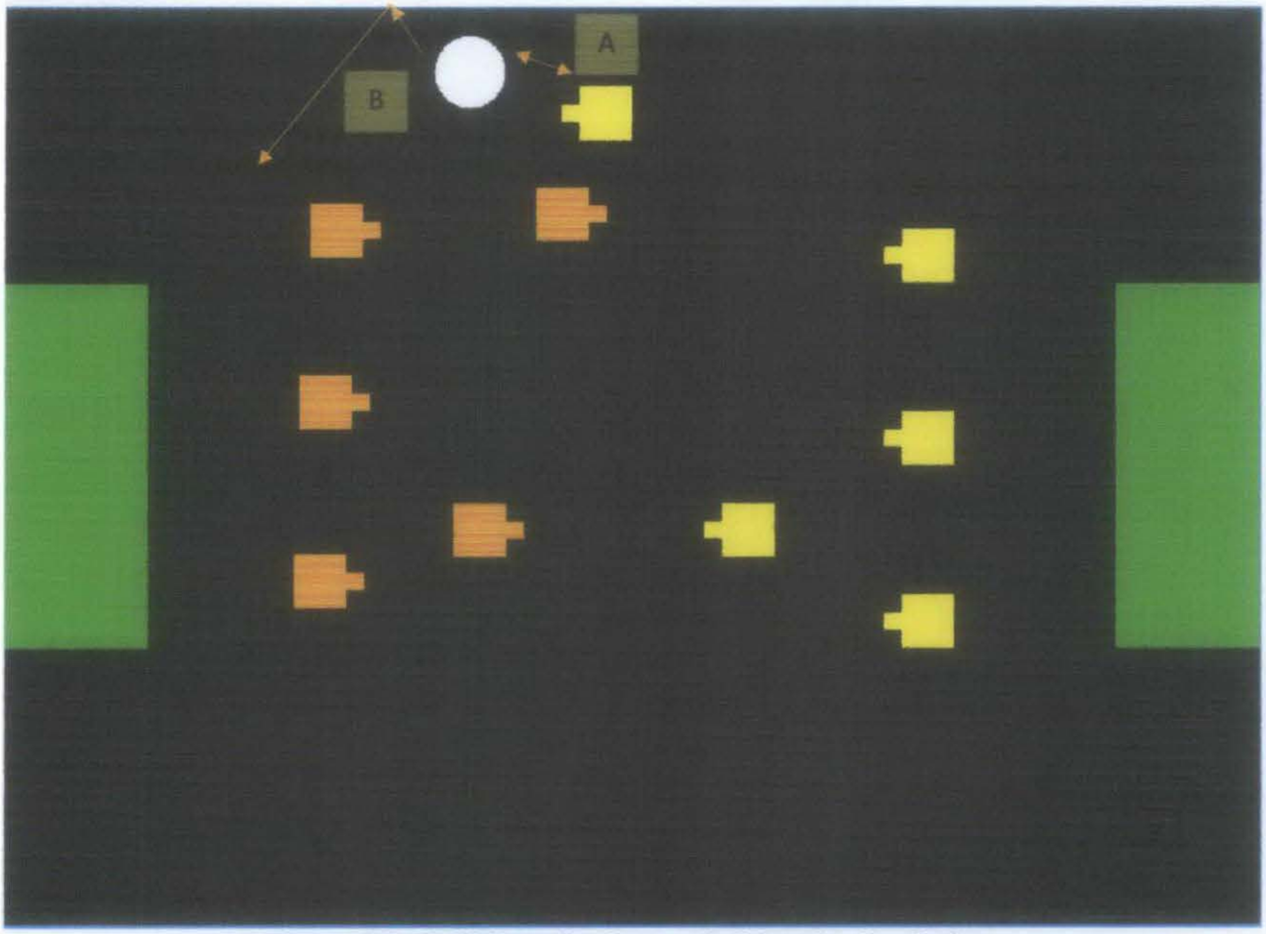

Figure 24: Collision of robot and object in simulation

4.3 Experimentation/Implementation & Prototype

The experiment, modeling, implementation and prototype are done simultaneously in developing this simulation. It is implement based on the scope of project with has been mentioned earlier which are:

(a) Robot and Object

(b) Robot and Environment

(c) Object and Environment

4.3.1 Object Detection - Robot & Object

As mentioned before, object's current position will always change when it moves. To make the robot follow or detects the object, it requires certain formula to be used.

```
Vect dir;
dir.x = currentPos.x - teamA[i].position.x;//distance player n ball
dir.y = currentPos.y - teamA[i].position.y;


double normaldir = sqrt(dir.x*dir.x+dir.y*dir.y);
 dir.x/=normaldir;
 dir.y/=normaldir;
//maintain constant speed regardless distance of the ball n player

dir.x *= 0.8;//increase speed player contact with ball
dir.y *= 0.8;
```

Figure 25: Formula for robot to detect object

(a) Find the direction and distance between robot and object

First, robots need to find the direction to move towards the object. This can be done by subtracting object's current position and robot's current position. It will show the direction to move towards the object. For example:

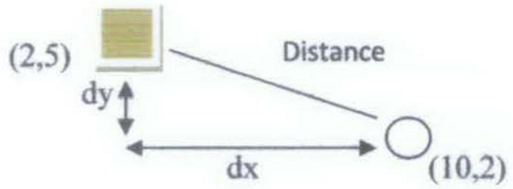

Figure 26: Example of calculating direction between robot & object

Assuming 'red box' is robot and 'white circle' is object. Both have its own coordinate. In order to find the direction to move towards the object, object's current coordinate minus with robot's current coordinate which result in **dy = -3** and **dx = 8.** Thus, the robot will move to the right and below according to the four quadrants concept. The value of dy and dx can be used to determine the distance of between robot and object.

(b) Normalize the speed

Once the distance is discovered, the speed of the object need to be normalizes to achieve a consistent speed. The distance and normalization are calculated by using the formula below:

$$\text{DISTANCE} \rightarrow = \sqrt{(X_2 - X_1)^2 + (X_2 - X_1)^2}$$

$$\text{NORMALIZE} \rightarrow = \sqrt{(X)^2 + (Y)^2}$$

(c) Control the speed of robots towards objects

Speed of the robot towards the object can be manipulated by doubling its current speed. After a collision, if it is too slow or too fast to move towards the object, by increasing or decreasing the speed of the robots, the robots will be able to catch up with the object.

The desired outcome between robot and object detection are:

- Robot detects only positive objects

- Robot ignore negative objects

- Robot that does not programmed to detect, will do nothing

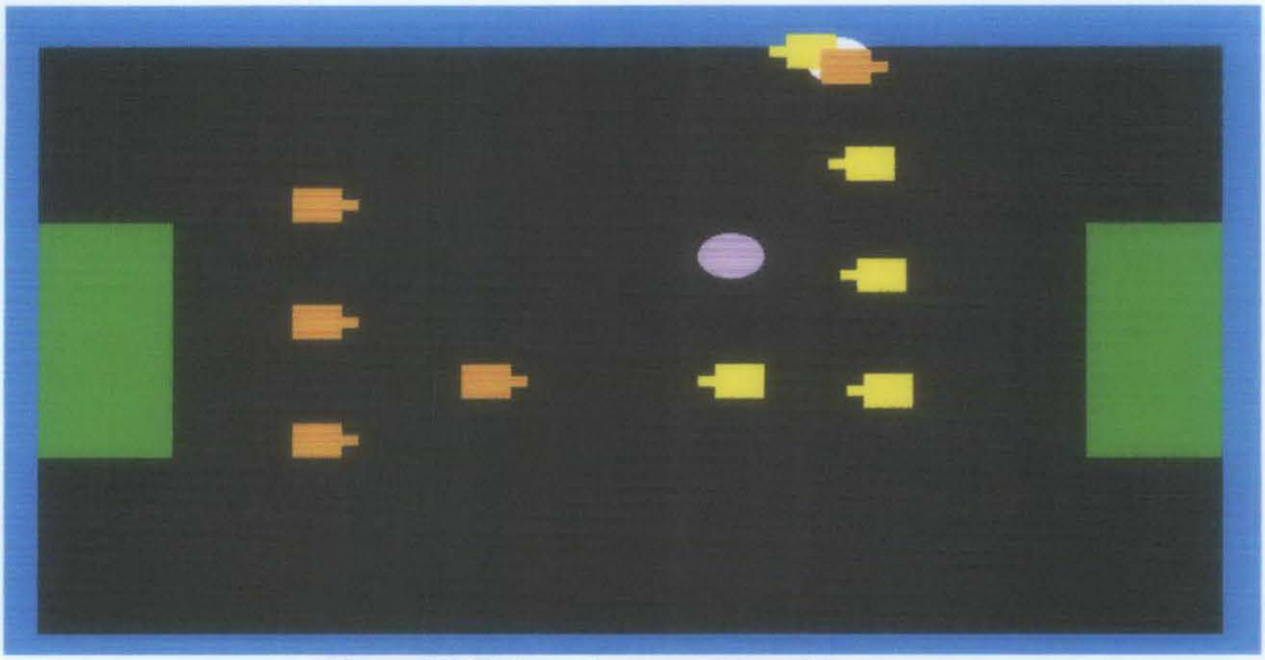

Figure 27: Robots detect positive object.

4.3.2 Object Detection - Robot & Environment

Robot and environment detection is to see how the robots react according to its surroundings. In his case, the robots will react with the object within certain distance that has already been set. The concept of the robot move towards object is still the same but it has a certain constraint to be satisfied. Some robots will only react when the ball enter their own half.

Robot from Team A(red) will only react or detect positive objects when it enters <375x550 while robot from Team B(yellow) will only detects positive object when it enters >375x550.

```
f(currentPos.x<=375 && currentPos.y<=550){
ect dir;
ir.x = currentPos.x - teamA[i].position.x;
ir.y = currentPos.y - teamA[i].position.y;
ouble normaldir = sqrt(dir.x*dir.x+dir.y*dir.y);
ir.x/=normaldir;
ir.y/=normaldir;
/magnitute
ir.x *= 0.7;
ir.y *= 0.7;
eamA[i].direction.x = dir.x;
eamA[i].direction.y = dir.y;
eamA[i].position.x += dir.x;
eamA[i].position.y += dir.y;
else {
eamA[0].position.x = 200.0;
eamA[0].position.y = 400.0;
eamA[1].position.x = 200.0;
eamA[1].position.y = 300.0;
eamA[2].position.x = 200.0;
eamA[2].position.y = 200.0;
```

The desired outcome between robot and environment detection are:

Robots will only detect the object within certain distance only. (When entering their own half-ield)

Robots will only detect object inside the field only.

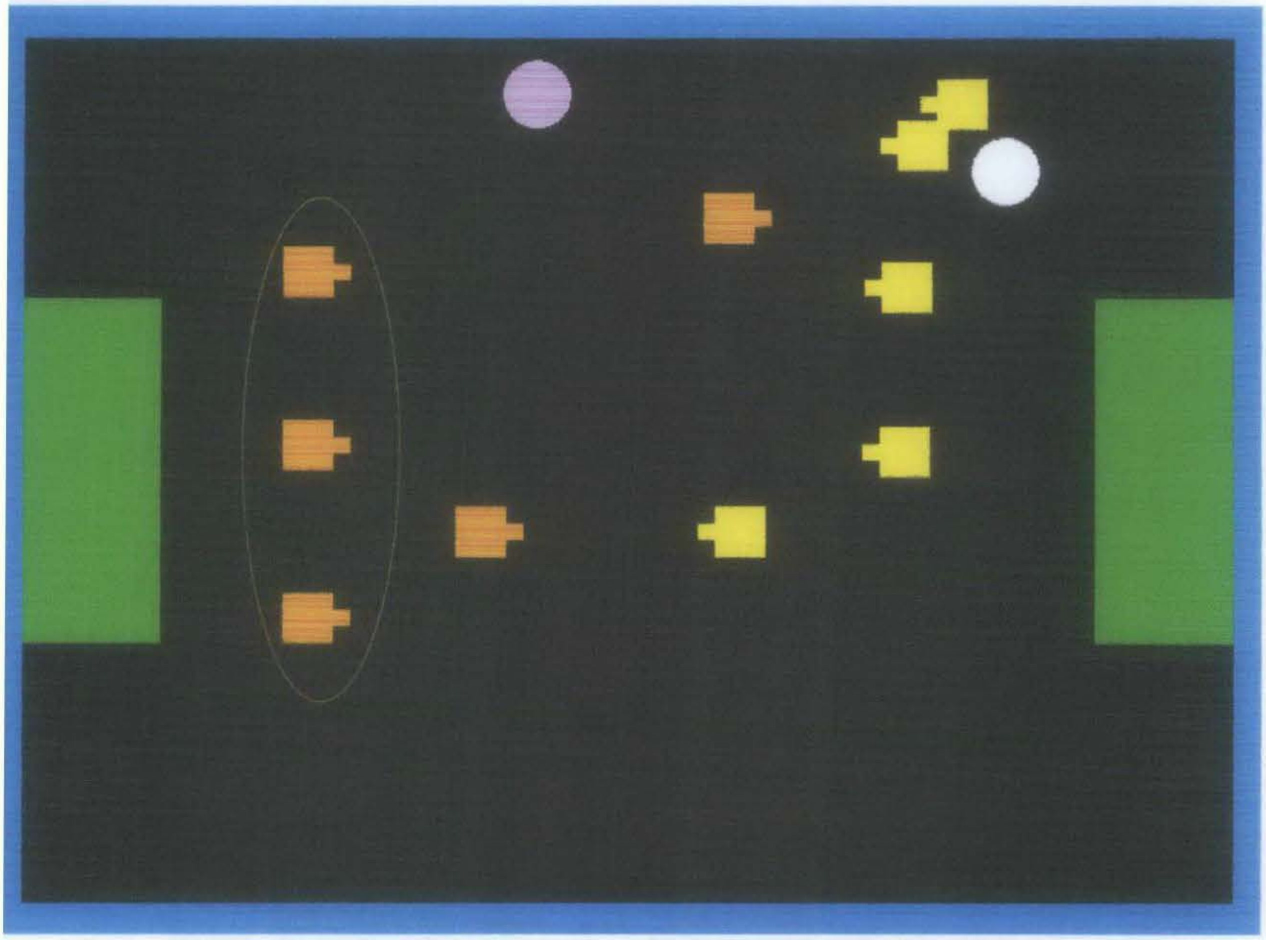

Figure 28: Robot detects positive object in a specific area/distance

4.3.3 Object Detection - Object & Environment

This last type of detection is execute when positive object enters the goal which then will reset back to its initial default position before starting to move again.

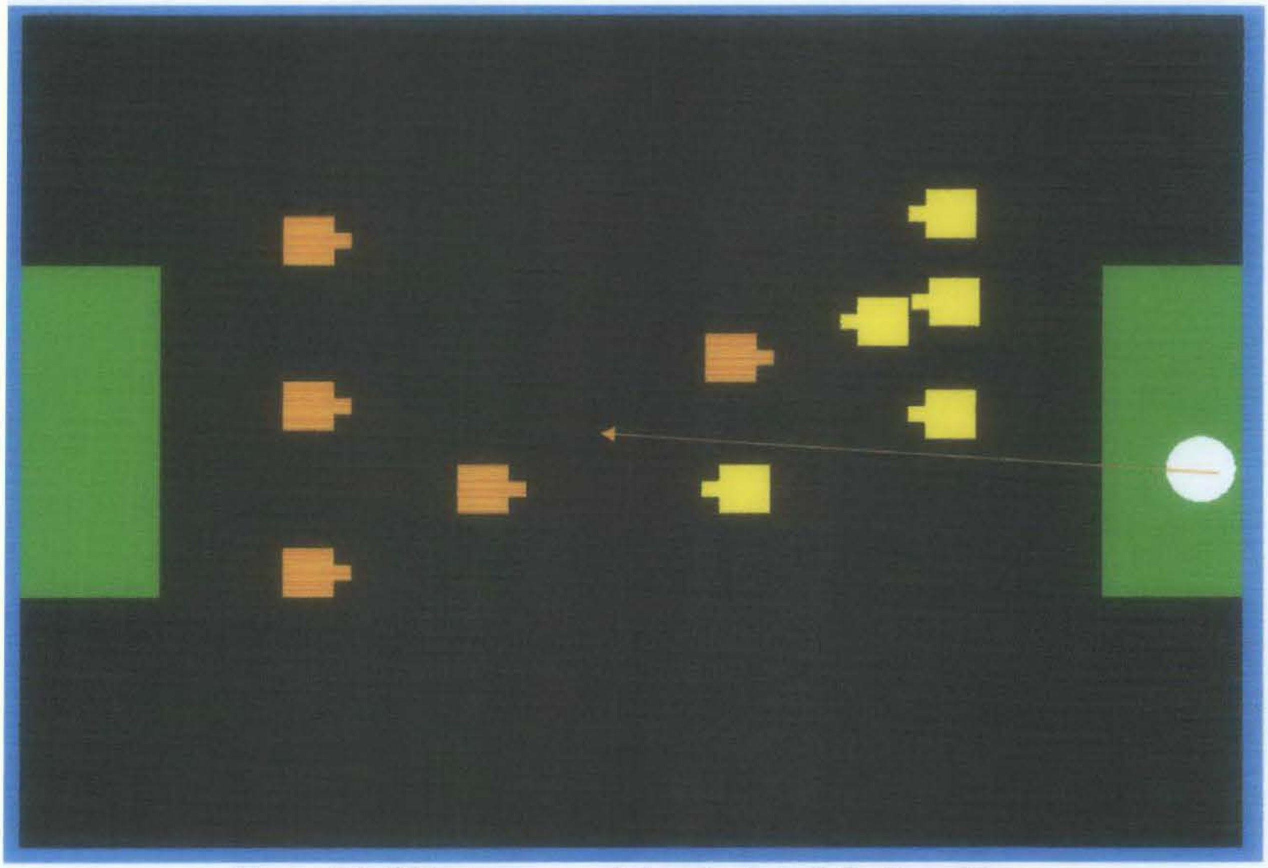

Figure 29: Object enter goal then back to default position

ł.4 Result & Discussion

4.4.1 Result- Detection Rate

[he testing of the simulation is done by using the ten different object starting direction in three lifferent forms of detections which are the robot & object, robot & environment and object & :nvironment. The detection rate between the robot and the object in different object starting lirection as well as in different form of detections are being recorded. Below is the result:

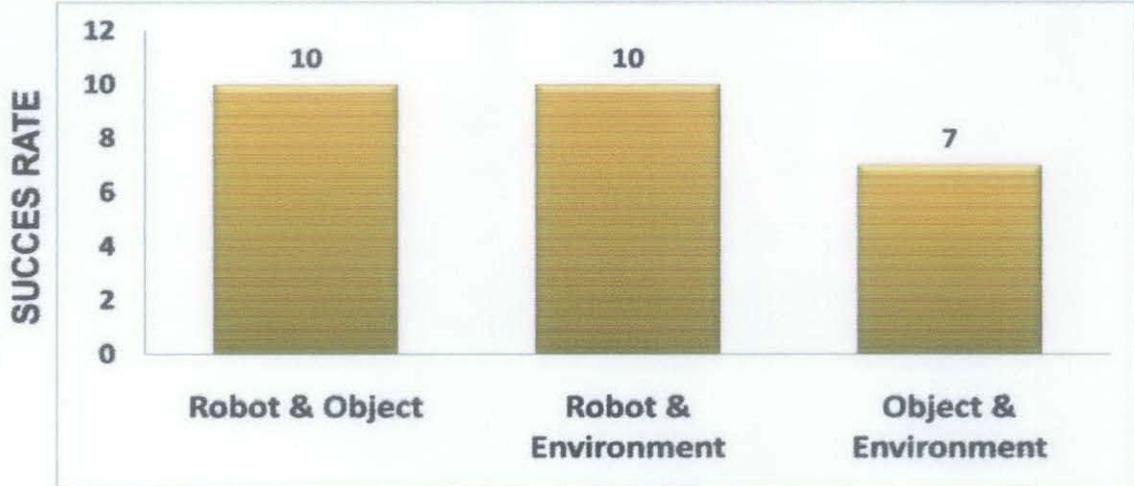

Figure 30: Success Rate

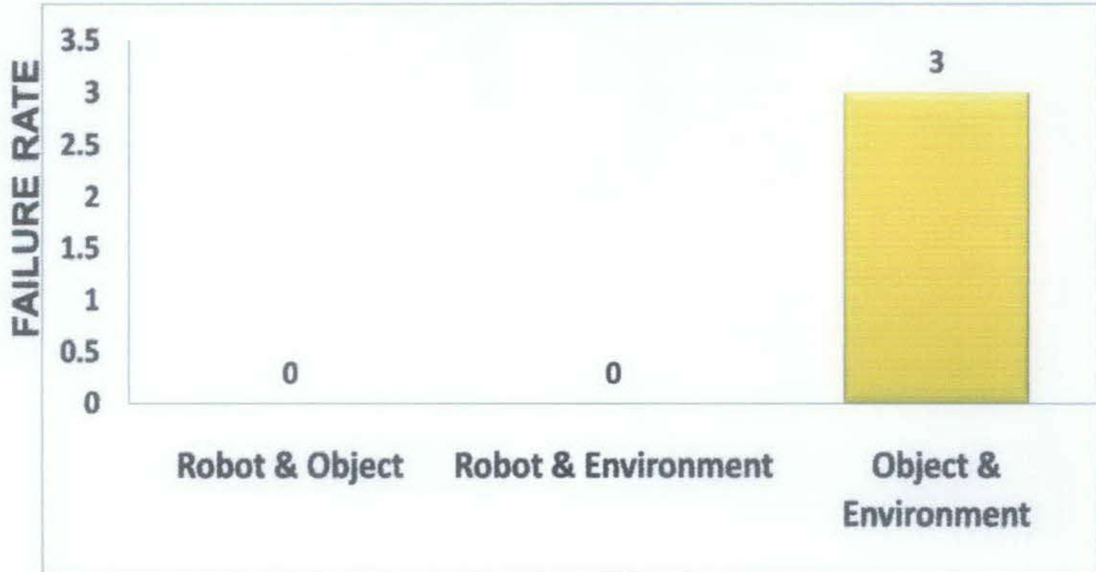

Figure 31: Failure Rate

4.4.2 Discussion

Based on the figures above, it shows success and failure rate after the simulation is tested using 10 different starting directions in 3 different forms of detection. In the (1) Robot & Object and (2) Robot & Environment detections, they have 100% success rate and 0% failure rate. During the testing, the robots perfectly react to the object regardless of their constraints and starting directions of the object.

In the (3) object & environment, the success rate is 70% and 30% failure rate. After manipulating the starting direction of the object, only this form of detection produce a failure. This happened due to the inaccuracy of the logic itself. The object needs to be inside the goal and hit the back of the goal in order to count it as a goal before being reset at its default position and continuing moving. If the object enters the goal but did not hit the back of the goal, it counts as a failure.

Out of 10 different directions, only 3 of the directions that produce failure rate and it only happen in one of the form of detection. Based on the result, the author thinks that this simulation still a success despite the failure rate produced. The success rate is much more higher than the failure rate and the most important thing is the core focus of this simulation is the object detection can be prove in different form of detection.

CHAPTER 5

CONCLUSION

.1 Relevancy to the objectives

he objectives are to implement the vector concept with object detection algorithm in detecting bject, to use multi-robots in the same environment in detecting objects and to test the detection ate of the multi-robots with the object. The objectives mentioned previously are relevant and it an be strongly said it is 100% achieved up to this period even though based on the result there is 0% failure rate in object and environment detection due to accuracy reason. As a whole, it also an be said that this project is successfully executed because the robots have detect and react ccordingly in the simulation and addressed all the problems mentioned by using the vector oncept and object detection algorithm

.2 Suggested Future Work & Recommendations

ven though the simulation is successfully executed, there are still rooms of improvement that an be done such as:

5.2.1 Robots and Robots Interaction

urrently, there is no collision between robot and robot. They tend to overlap on each other luring the object detecting process. By having collision detection between robot and robot in the uture, it will make the simulation much more logic and create a smooth detection process etween the robots and object

5.2.2 Randomness of Ball Movement

As for now, the ball movement is pre-set and it is done manually in the coding. As mentioned efore, the first thing that moves is the ball only then the robots will detect and the testing is lone by using ten different directions which will be change ten times in the coding. It is not fficient and will make the movement of the robots as well as the objects is predictable and will tay the same.

5.2.3 Graphics/Audio

:he simulation is done in two-dimensional (2D) and there is no audio available due to not nough time and less resources as well as to focus more on the core things of the project which is he object detection itself. This simulation will be much more interesting and user friendly if it an be done in three dimensional (3D) and add some audio when robots detects an object or vhen the object enters the goal.

5.2.4 Approached/Logic/Concept Used

\s mentioned before, this simulation is using the vector concept and the object detection lgorithm. It is workable but not effective. There are other logics and algorithm that can be used o determine which are the most efficient way in detecting object such as the bee colony ptimization and ant colony optimization. By manipulating this logic, only then the most fficient way to detect object can be found.

3y maintaining the current condition and keep on finding ways to improve this simulation, a nore intelligent and interesting simulation will be develop which in the long term can help to ransfer this simulation into functional hardware and contribute to the reach the ultimate goal vhich is to create a team of robots to play soccer with real humans.

REFERENCE

1] Minoru Asada, Hiroaki Kitano. March 1999, "*RoboCup-98: Robot Soccer World Cup II*", Springer-Verlag Berlin Heidelberg.

2] Gal A. Kaminka, Pedro U.Lima, Raul Rojas (Eds.). May 2003, "*RoboCup 2002: Robot Soccer World Cup VI*", Springer-Verlag Berlin Heidelberg.

3] Michal Baczynskit, Balasubramaniam Jayaram, June 2008, "*Fuzzy Implications*", Springer-Verlag Berlin Heidelberg.

4] Oscar Castillo, Patricia Melin, July 2007, "*Type-2 Fuzzy Logic: Theory and Applications*", Springer-Verlag Berlin Heidelberg

5] Carlos Gallardo Garcia, Sergio escalera I Petia Radeva, 2007, "*Generic Object Detection For Autonomous Robot*" Journal, University Of Barcelona

6] Surachai Panich, 2010, "*Method of Object Detection for Mobile Robot*", Journal of Computer Science 6 (10): 1122-1124,2010.

7] Wikipedia – The free Encyclopedia, *Aibo.* October 16, 2010

Retrieve from< http://en.wikipedia.org/wiki/AIBO>

8] Wikipedia – The free Encyclopedia, *Adaboost.* October 16, 2010

Retrieve from< http://en.wikipedia.org/wiki/Adaboost>

9] OpenGL – Tutorial

Retrieve from< http://www.videotutorialsrock.com/opengl_tutorial/draw_text/text.php>