Palestine Polytechnic University

College of Administrative Science & Informatics

Department of Information Technology

Wide Vision Based Touch Screen

Submitted by:

Amr Al Tamimi

Fahed Al Qawasmeh

Murad Al Salaymeih

Supervisor:

Eng. Manal Tamimi

This project is presented as partial fulfillment for requirements of BSc degree in information technology.

June - 2009

ABSTRACT

Touch screens are input devices that allow users to interact easily with the PC, through finger touches on the screen surface.

Touch screen was used in many applications of real life, because of its flexibility, efficiency, ease of use and high response; it became one of the most widely used input devices today.

Our project is wide vision based touch screen that aims to help the user to interacts with the PC using her/his finger, and this project is a development to the traditional data show display to making it sensitive to the user touch, the user must put colored marker on her/his finger.

The system contain two cameras fixed on the sides of the screen in addition to a datashow, two images are taken from the cameras. The finger position is extracted from the images through image processing techniques, the coordination of the finger position are converted from the image coordinate system to the real world coordinate system, then tangible events are send to the operating system.

The systems worked correctly, and succeed in achieving the actions of mouse click, double click, and drag and drop.

DEDICATION

To our mothers

To our fathers

To our brothers and sisters

To our teachers

To our friends

To everyone who helped us

To whom we love

We dedicate our humble effort

Project team

ACKNOWLEDGMENT

We would like to thank everyone who helped us to complete this project, and to appreciate all the encouragement of our families and friends…

Then we would like thank the college of Administrative Science and Informatics. Special thanks to the Dean of Information Technology Department Dr.Mohammad Hssuna, and to the Information Technology Department's Head Mr. Akram Ihshayesh …

We afford deep thanks to our supervisor Eng. Manal Al-Tamimi who have granted us support, orientation, guidance, help, encouragement and advice…
And special thanks to Dr.Hashem Al-Tamimi for his help and advice in improving the project…
Finally we would like to thank Mr. Hamed AL-Qawasmeh for his advices…

TABLE OF CONTENTS

LIST OF FIGURES

List of Tables

CHAPTER 1

Introduction

- Overview
- Definition of Touch Screen
- Advantages of Touch Screen
- Types of Touch Screens
- General Idea
- Problem Statement
- Project Importance
- Related Works
- Project Domain
- System Objectives
- Methodology
- Expected Results
- Team

1.1. Overview

In the last few years we were in a huge technology evolution in both software and hardware that focus on improvement of speed, efficiency, flexibility, and time. The new developed applications came with a problem that lie in the inflexibility of most used input devices (mouse and keyboard) to support such applications for unskilled users.

Hardware developers solved such problem by creating an interactive technology (Wide Screen) that require the user only to touch with their finger or using the so-called "Stylus" which looks like a pen to use the screen and click on when they want.

1.2. Definition of Touch screen

A Touch Screen is a display which can detect the position of a user touch within the display area. Enables the user to interact with the computer by touching display area. [1]

1.3. Advantage of touch screen

There are a number of advantages for the touch screens. They include the following:[2]

1. Easy to use (as the user simply touches what he or she sees on the display).
2. Touch monitors can even be mounted on the wall.

1.4. Types of touch screens

There are many types of touch screens including the following:

1.4.1. Resistive touch screen

A resistive touch screen panel is composed of several layers. The most important are two thin metallic electrically conductive and resistive layers separated by thin space. When some object touches this kind of touch panel, the layers are connected at a certain point, the panel then electrically acts similar to two voltage dividers with connected outputs. This causes a change in the electrical current which is registered as a touch event and sent to the controller for processing. [3]

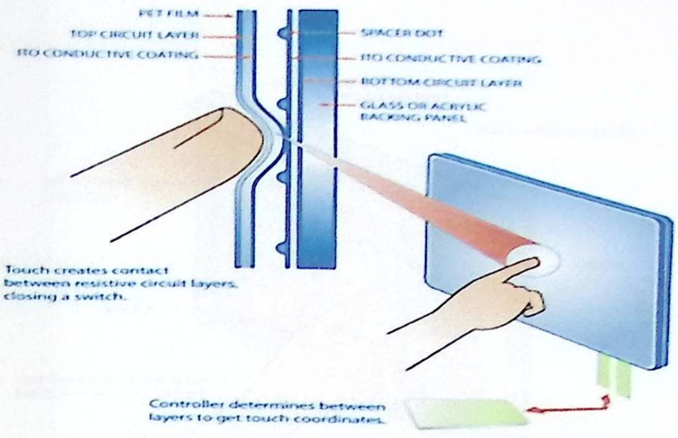


Figure 1.1: Resistive touch screen. [4]

- **Advantages of Resistive touch screen: [5]**
  - Activated by any stylus.
  - High touch point resolution.
  - Low power requirements.

- **Disadvantages of Resistive touch screen:**
  - Polyester surface can be damaged.

1.4.2. Surface Acoustic Wave: [2]

Surface Acoustic Wave (SAW) technology uses ultrasonic waves that pass over the touch screen panel. When the panel is touched, a portion of the wave is also touched. This change in the ultrasonic waves registers the position of the touch event and sends this information to the controller for processing.

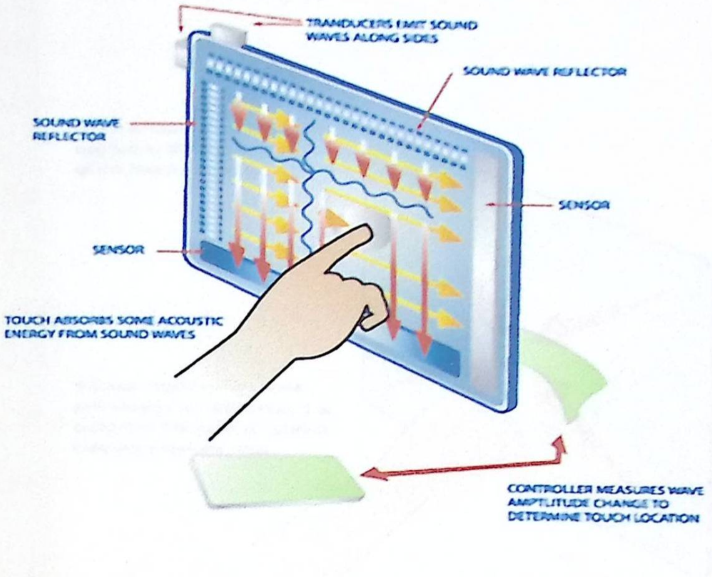


Figure 1.2: Surface Acoustic Wave. [5]

- **Advantage of SAW touch screen:** [5]
  - Good optical clarity.
  - Z-axis capability.
  - Durable glass front.

- **Disadvantages of SAW touch screen:** [2,5]
  - Surface wave touch screen panels can be damaged by outside effects.
  - Signal affected by surface liquids or other contaminants.
  - Difficult to industrialize

1.4.3. Capacitive touch screen:[2]

A glass panel is coated with a conductive coating material that it is used fused into the class .the coating is connected to four electrodes at the edges of the screen. Each electrode is connected to an oscillator circuit. When operator touches the screen, the body capacitance of the operator causes a change in the impedance of the screen.

The impedance change causes the oscillator frequencies to vary, and frequency differentials are converted into X, Y.

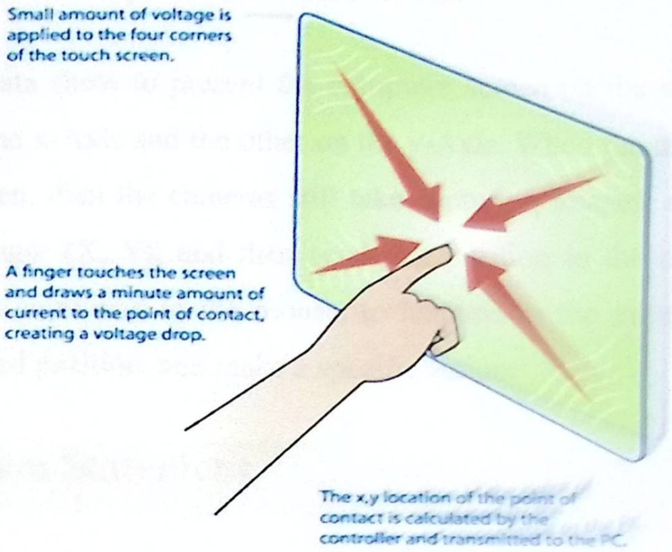


Figure 1.3: Capacitive Toch Screen.[4]

- **Advantages of Capacitive touch screen: [5]**
  - Extremely durable.
  - Very accurate.
  - Good optical clarity.
  - Good resolution.

- **Disadvantages of Capacitive touch screen:**
  - Severe scratch can affect operation within the damaged area

1.5. General Idea

The main idea in our project is to design a special kind of touch screen whose principles are based on computer vision and cameras.

We will use the data show to present the computer screen on the wall, and put two cameras; one on the x-Axis and the other on the y-Axis. When the user places his\her finger on the screen, then the cameras still take capturing images, and detecting the position of the finger (X, Y), and then send this position to the operating system, which will send a command to the mouse, to respond to the event by moving the mouse to the desired position, and make a specific action.

1.6. Problem Statement

Since the evolution of the computer age, interaction between the users and their computer was carried out through two input devices, the keyboard and the mouse. The two devices were considered adequate to perform the tasks required of them. Yet, it seems that there has been some dissatisfaction. For example the lecturer faced difficultly to movement between the Data Show and the computer.

Computer companies tried to resolve this "dissatisfaction" through the invention of touch screens that led to the increase of interaction between the user and the hard drives. Yet, this "solution" came at the expense of the customer as it came at a hefty price.

In our project we are trying to develop a system that allows the user to interact directly with computer applications through simply touching an image on the wall at a reduced cost. The project presumes that the organization using the project has Data Show present more frequently.

1.7. Project Importance

The importance of this project is to make use of the vast evolution in the field of the image processing, and depending on equipment which is already in our organization (such as data show, laptop) in producing an inexpensive effective and efficient presentation tools, this idea can support our universities with modern presentation tools with low costs.

Touch screen technology provides a fast and intuitive interface for the user. It also simplify customer interactions as users do not have to know how to use a computer, but simply touch the monitor to make their choices.

1.8. Related Works

Vision-based Interaction with Fingers and Papers. Zhengyou Zhang, 2003, *International Symposium on the CREST Digital Archiving Project* pages 83–106.

This paper presents Visual Screen, uses technique to transform an ordinary screen into a touch screen using an ordinary camera. The setup method position a camera so it can see the whole screen. The system calibration involves the detection of the screen region in the image, which determines the projective mapping between the image plane and the screen. In runtime, the user can use his/her finger to interact with the computer and the system locates the finger position in the image and converts the image position to the cursor position on the screen·

- **Accurate image-screen calibration**

The plane perspective from the screen plane and its 2D projection on the image plane is described by a homography, a 3 ×3 matrix defined up to a scale factor. This matrix can be easily determined from 4 pairs of image-screen point correspondences. The correspondences are not difficult to obtain because we know the screen coordinates of four screen corners, and their corresponding image points can either be detected automatically or specified by the user. [10]

- **Segmentation**

It's difficult to separate the indicator from the background screen because its contents Change frequently, this observation forms the base of the segmentation algorithm described as Follows. They firstly compute a color model for the screen without the indicator. A number of pictures with rich color are displayed on the screen in order to make the model as general as possible. To compute this background model all of the pixels in the image are histogram med namely, for each pixel its color intensity is placed in the proper bin of a preferred possible 256 intensity levels. [10]

- **Finger Tip Locating**

The algorithm can be elaborated as the follows. A cumulative total of the numbers of pixels that belong to the foreground are calculated on a scan line by scan line basis starting at the top of the image containing the indicator. The resultant histogram will be referred as horizontal histogram the horizontal pixels first appear and increase in cumulative total thereafter. The identified scan line roughly corresponds to where the indicator tip location may be found. Next, a number of lines above and below the identified line are selected and each is scanned to find the start and end of the foreground pixels in the horizontal direction. In addition, the center point of each series of foreground pixels along each of the scan lines is determined and a line is robustly fit through these points. The pixel corresponding to the indicator tip location is then determined by scanning all pixels within the previously identified indicator window to find the boundary pixels. The pixel corresponding with the tip of the indicator is the boundary pixel where the previously determined centerline intersects the boundary of the indicator. Histogram is next analyzed to determine the scan line where the foreground. [10]

1.9. Project Domain

The idea behind this project is to develop a wide area and cheap display system that enables the user to interactive with the system by touching the screen with his/her finger.

- **In Academia Field**

  Most of universities and schools use this technology in the education process as it is facilitate the interaction between the educators and educated.

- **In Organizations Field**

  Many organizations need data show displays to make presentations much more interesting and easier. This technology can help them to make the presentation in easy way.

1.10. System Objective

This project aims to achieve the following objectives:

- Build a vision-based wide touch screen that allows the user to be interactive with the computer in an easy manner.
- Build touch screen system in low cost.
- To facilitate lectures in educational settings.
- Touch screen for users with special needs.

1.11. Methodology

In our project we depended on the waterfall methodology. This model is a sequential development process that divided the project into sequential phases with some overlap. The phases of the project include analysis, design, implementation, testing integration, and maintenance.

During the process of building our project, we carried out the following steps:

- Reviewing of other projects related to our project.
- Analyzing our project including objectives and requirement.
- Setting up the cameras in the working environment. We tested several cameras and choose the best one.
- Collecting the software and libraries that we used it in the implementation of the project.

1.12. Expected Results

The results we are expecting at the end of our project:

- System accurately determine the position on the display area.
- The user can make all mouse actions such as open folder, drawing, and click on icons, etc.

1.13. Team

Project team:

Amr Al Tamimi Fahed Al Qawasmi. Murad Al Salaymeih

Supervisor:

Eng. Manal al Tamimi

CHAPTER 2

System Planning

- Introduction
- Risks and Solutions
- Project Limitations and Constraints
- System Resources
- System Alternatives
- Time Feasibility

2.1. Introduction

In this chapter we will discuss the feasibility of our project. This phase is considered to be one of the most important stages of the project. We will also look into the resources such as hardware, software and cost as well as an evaluation of the risks that can arise during the project.

2.2. Risks and Solution

This section discusses the system risks that project team have faced during the development process. It also discusses the steps taken by the team to solve the ensuing problem and reduce the risks to minimum, thus increasing the possibility of success.

2.2.1. Risks Management

Some risks may occur because of the software or hardware used in the system.

1. **Hardware risks:**
   - Computer / Microprocessor malfunctions.
   - The malfunctioning of the used webcams.
2. **Software risks:**
   - Caused by viruses infections.
   - Problems that might occur in the software development environment.
3. **People / Staff Risks:**
   - Illness or absence of any member of the team during the developing of the project
4. **Organizational Risks:**
   - Facing financial problems.
   - Facing project resources problems.

- Shortage of wide angle camera.

5. **Requirements Risks:**
   - Risks that might occur if new changes become necessary in system requirements that need major changes in the system design.

6. **Estimation Risks:**
   - Risks that may drive from the wrong estimation in the system design, implementation, resources and management.

2.2.2. Risk Avoidance

1. Fix and adjust cameras correctly in suitable location.
2. Try to solve any problems that might take place while building the system.
3. Estimate costs truly.
4. Continuous backup of the system as to avoid any possible computer failure.
5. Try to estimate accurately the time needed for each task in project schedule.
6. Taking care when using hardware components and using them according to their specifications.
7. Using only the compatible software development environment to implement the project.
8. Good estimation of system requirements.

2.3. Project Limitation and Constraints

1. The position of camera must be constant.
2. Using constant frame with specific dimensions.
3. We must use finger glove with a specific color.
4. The display area of data show screen must be constant.

5. Any very poor/huge light on the room may affect the functionality of the cameras.

2.4. System Resources

System resources can be classified into hardware, software, and human resources. Alternative resources were also identified in case of the failure of any of the principal resources.

2.4.1. Required Development System Resources

1. **Hardware Resources**: The following table lists the hardware resources required to develop the wide vision-based touch screen system:

Table 2.1: Hardware development resource.[6]

No.	Item	Quantity	Cost Per Unit	Total Cost
1	computer Centrino due 1.66GHz (2 CPU) RAM 512MB Hard Disk drive 80 GHz Monitor 17 inch	1	$700	$700
4	Data show	1	$649	$649
5	FireWire-Camera(Webcams)	2	$150	$300
6	FireWire-Connection cables	1	$6	$6
7	Physical frame	1	$10	$10
8	AC Power	1	$4	$4
	Total			**$1669**

2. **Software Resources:** The following table lists the Software resources necessary for the development of wide vision based touch screen system.

Table 2.2: Software development resources.[6]

No.	Software	Cost
1	Windows XP professional	$185
2	Windows Microsoft Office	$298
3	Microsoft Visual Studio 2005	$299
4	OpenCV	Free
	Total	**$782**

3. **Human Resource:** The human resources include the members of the project team, working on analysis, design, and implementation

Table 2.3: Development human resources.

Member	Hours/week	Cost/hours	Total cost/week
Programmers(3)	30	$12	$360
Total		**$360**	

The cost of human recourses during 15 weeks:

$360 * 15 week = $5400.

The cost of human recourses during one month:

$360 * 4 week = $ 1440.

4. **Other working costs:** Other costs such as books, papers, pens, and transportations are estimated to be $20/ week. (Total per 15 weeks is $300).

2.4.2. Required Operation of system resources

Assuming that data shows and computers are available in lecture rooms in our universities, then the additional cost required to operate and execution this project will be as the following:

1. **Hardware operational resources:** The following table lists the hardware resources required to operate the wide vision-based touch screen system.

Table 2.4: Hardware Operational Recourses.[6]

No.	Item	Quantity	Cost Per Unit	Total Cost
1	FireWire-Camera (Webcams)	2	$150	$300
2	FireWire-Connection cables	1	$6	$6
3	Physical frame	1	$10	$10
4	AC Power	1	$4	$4
Total				**$320**

2. **Software operational recourses:**

The following table lists the Software resources necessary for operating wide vision-based touch screen system.

Table 2.5: Software operational recourses.[6]

Software	Cost
Microsoft Windows XP Professional	$185
Total	**$185**

3. **Total System Development Costs:**

The following table lists the total costs for the resources needed to develop this project.

Table 2.6: development cost

Resources	Cost
Hardware Resources Development Costs.	$1669
Software Resources Development Costs.	$782
Human Resources Development Costs.	$5400.
Other Working cost.	$300
Total Costs:	**$8151**

4. Total System Operational Costs:

The following table lists the total costs for resources needed to operate this project

Table 2.7: Total operational cost

Resources	Cost
Hardware Resources Operational Costs.	$320
Software Resources Operational Costs.	$185
Total Costs.	**$505**

2.5. System Alternatives

2.5.1. Infrared

Infrared is an invisible electromagnetic radiation waves that is longer than the visible light and shorter than the radio waves. The name infrared means below red and red here means the portion of visible light.

Infrared technology has many applications in life today including target acquisition and tracking by military applications. [7]

- **Infrared touch screen Advantages:**
  1. Low cost.
  2. Infrared signal are not visible to user.
  3. High reliability, accuracy, and stability.
  4. Accurate touch position performance.
- **Infrared touch screen Disadvantages:** [14]
  1. Didn't work in wide range touch screen.
  2. It needs direct line of sight between transmitter and receiver.

2.6. Time Feasibility

In this section, we attempt to show how we have spread the given period over the development stages. The total time interval that it took to develop the system was 15 weeks.

2.6.1. Time Schedule

While this project is a graduation project, it must be developed during one semester, so our time schedule distribution of the different tasks over sixteen weeks.

The project team has managed the allocated time as follows:

Table 2.8: Time Schedule

Task	Work	Time in weeks
T1	Information collection.	2
T2	System Planning.	2
T3	System Analysis.	2
T4	System Requirements.	3
T5	System Design.	3
T6	Coding and Implementation.	6
T7	System Testing.	2
T8	Documentation.	15

2.6.2. Gantt chart for time schedule

Table 2.9: Gantt chart for time schedule

Task \ week	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
1	■	■													
2			■	■											
3				■	■										
4						■	■	■							
5									■	■	■				
6										■	■	■	■	■	■
7														■	■
8	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■

2.6.3. Technical Feasibility

Our project requires a programming experience in C++, C, and Visual Studio in filed of image processing, our team already has the required experience any shortage, any shortage in experience may be gained from experts in the university or from any open recourses in the world.

2.6.4. Legal Feasibility

While looking to our situation, there are no legal limitations or policies in building such a project. When implementing it in the real business environment, surely commercial rules, governmental regulation, and other policies should be considered.

CHAPTER 3

System Requirements

- Introduction
- Functional Requirements
- Nonfunctional Requirements

3.1. Introduction

In this chapter we shall describe in details system requirements after categorizing them as functional and nonfunctional requirements.

3.2. Functional Requirements

The functional requirements can be classified into categories: System requirements and user requirements.

3.2.1. System Functional Requirements:

1. The system must take the dimensions of the datashow displaying area. The installed cameras must be placed in a manner that identifies the displaying area.
2. The system must use two cameras, one on X-Axis and other on Y-Axis. When the user puts his/her finger on the wall, the cameras must be able to capture an image and send it to the system. The system, in turn, must be able to determine the position of finger.
3. The system must be able to determine the finger position and block any other random clicks.
4. The system should be able to capture the image, process it through converting the RGB image to HSV image.
5. After image processing, the system runs a scanning for image pixels to isolate the finger from the whole image. The system would then determine the location of the touch on display area.
6. The system will take an action that is similar to using the mouse as input device, depending on the position of the cursor on the screen when the touch occurred.

3.2.2. User Functional Requirement:

1. The user can determine the position of the cursor by moving his /her finger on the data show displaying area.
2. The user can make many events such as click, double click, drag and drop
3. Datashow displaying area will ignore multiple touches and accept only one touch.
4. The user will touch anywhere in boundaries of data show displaying area.

3.3. Nonfunctional Requirements

Non-functional requirements (as its name suggest) are requirements that are not directly concerned with specific function delivered by the system. It has to be stated, though, that non-functional requirement may be more critical than functional requirements.

Nonfunctional requirements are divided into two categories:

3.3.1. System Non functional Requirements:

1. The system must be functioning on real time.
2. The view of the two cameras must cover all boundaries of data show displaying area.
3. Reliability:

   The system must be able to determine the position of the finger correctly.
4. Efficiency:

   The touch screen must be accurate, able to respond and maintainable.
5. Performance:

   The data show displaying area will provides the user with a fast reaction.

3.3.2. User Non functional Requirement:

1. **Ease of use:**

The touch screen will be easy to use and through which the user can control the position on data show displaying area.

2. **Accuracy :**

The system must be able to provide a high level of accuracy specially when determining the position on data show screen.

3. **Flexibility :**

The touch screen will provide the user with a very flexible presentation tool than any other traditional tool.

4. **Speed:**

The speed of the response in the system will be high to the degree that it makes the system real time.

5. **Efficiency:**

The touch screen system that is more efficient than traditional input device, and increases interaction with user.

CHAPTER 4

System Analysis

4.1. Introduction

In this chapter we want to clarify how the system works in more details and classify all steps and stages of processing. We also attempt to illustrate how users can interact with the System.

4.2. How the System Works

Before the user can start using the system and to interact with it, she\he must be sure that the cameras are on the it's position and that they are fixed in the right physical position in the frame. The user must also make sure that the four corners of the display area are propositional to the computer screen using the projection equation. When all the previous steps were taken, the user operates the system and interacts with it as follows:

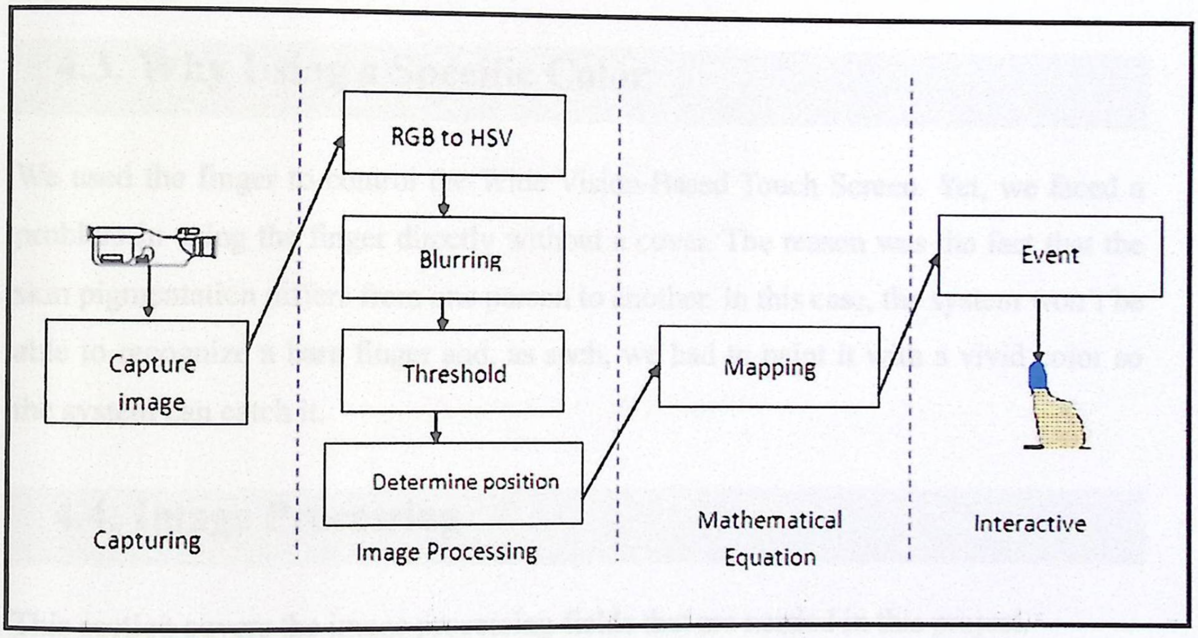


Figure 4.1: : How system work

- The user would place a glove on his finger. The glove should have a distinguished color. The user can choose any color as he/she wants but the color should be distinguishable.
- The user would touch the wall inside the boundaries of the datashow. The cameras would take the image of the finger and then send it to the system for processing.
- In reality, the system would transform the image from an RGB type to an HSV type. The Blurring Algorithm would then be applied in order to reduce the noise in the image before the Threshold Algorithm is applied. The second algorithm specifies the shape of the finger and distinguishes it from other parts of the image and assigns it an X and Y value.
- The system sends the X and Y values to the Operating System which provides the appropriate order to the courser to execute specific events in the specified position.

4.3. Why Using a Specific Color

We used the finger to control the Wide Vision-Based Touch Screen. Yet, we faced a problem in using the finger directly without a cover. The reason was the fact that the skin pigmentation differs from one person to another. In this case, the system won't be able to recognize a bare finger and, as such, we had to paint it with a vivid color so the system can catch it.

4.4. Image Processing

This section covers the image processing fields that are needed in this project.

4.4.1. Image

As a matter of definition, a computer image is a matrix (a two-dimensional array) of pixels. The value of each pixel is proportional to the brightness of the corresponding point in the scene.

4.4.2. Color Images

Instead of using just one image plane, color images are represented by three intensity components. These components generally correspond to red, green, and blue (the RGB model) although there are other color schemes including the HSV color model (which is defined by the components hue, saturation and value).

4.4.3. RGB

The RGB color model is an additive color model in which red, green, and blue light are added together in various ways to reproduce a broad array of colors. The name of the model comes from the initials of the three additive primary colors, red, green, and blue. [16]

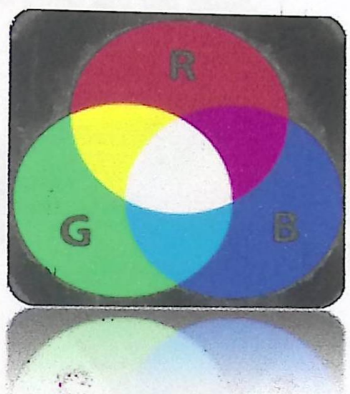


Figure 4.2: RGB Color[17]

4.4.4. HSV[18]

The HSV (Hue, Saturation, and Value) model: It is also called HSB (Hue, Saturation, Brightness) as it defines a color space in terms of three components:

Hue (H): the color type (such as red, green). It ranges from 0 to 360 degree, with red at 0 degree, green at 120 degree, blue at 240 degree and so on.

Saturation (S) of the color ranges from 0 to 100%. Also sometimes it called the purity. The lower the saturation of a color, the more "grayness" is present and the more faded the color will appear.

Value (V) of the Brightness (B) of the Color: The brightness ranges from 0 to 100%. It is a nonlinear transformation of the RGB color space or how bright or dark it is. Please note here that HSV and HSB are the same.

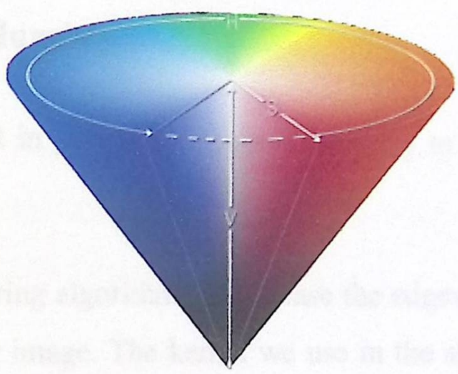


Figure 4.3: HSV Color [19]

4.5. Algorithms Used in the System

We use the following algorithms in the System.

4.5.1. Algorithm Converting RGB to HSV

The camera sends the RGB image to the system which, in turn, processes the image to convert it to HSV image. The operation is carried out through using algorithms that convert RGB to HSV. We, then, take Hand S value and ignore the value of V.

- **Why ignore the value of V:**

Because the light varies from one environment to another and the fact that this system may not run correctly if light is changing. As such, we have ignored value of "V" because "V" is the percentage of the light in the HSV color and the light effect negatively the system execution and functioning.

4.5.2. Image Blurring

It is a widely used effect in graphics software, typically to reduce image noise and reduce detail.

In this stage we use blurring algorithm to decrease the edges in the image in order to eliminate the noise in the image. The kernel we use in the algorithm is (5*5). It was used after a number of trial-and-error attempts.

4.5.3. Image Threshold [11]

Threshold is the simplest method of image segmentation. Individual pixels in a binary image are marked as object pixels if their value is greater than some threshold value (assuming an object to be brighter than the background) and as background pixels otherwise. Typically, an object pixel is given a value of "1" while a background pixel is given a value of "0."

If the original image is f (i , j) the output (threshold) image, b(i , j), is calculated:

$$b(i,j) = \begin{cases} 0, & f(i,j) < T \\ 1, & f(i,j) \geq T \end{cases}$$

- **How to make threshold:**

After applying a blurring on the HSV image, the threshold algorithm was applied on the H image and on the S image. Two threshold images were produced as a result. We then made a comparison between each pixel in the two images; in case the two pixels were "white", then we would put white pixel in the new image. If, however, the two pixels were "black", we would then put black pixel in the new image. In case one pixel was white and the other was black, then we would put black pixel in the new image. After comparing all pixels, one image will be produced as the white pixel describes the finger.

4.5.4. Algorithm to Determine Finger Position

In this stage we can find the position of the finger by scanning all pixels in the image from down to top. When the algorithm finds its first white pixel, it would consider this pixel as the position of the finger.

4.6. Context Diagram

The following drawing illustrates the contents of the System and its relationship with its immediate environment. Essentially, the user places a finger on the wall, the cameras take the picture, the system processes it using a number of steps to get the specified event.

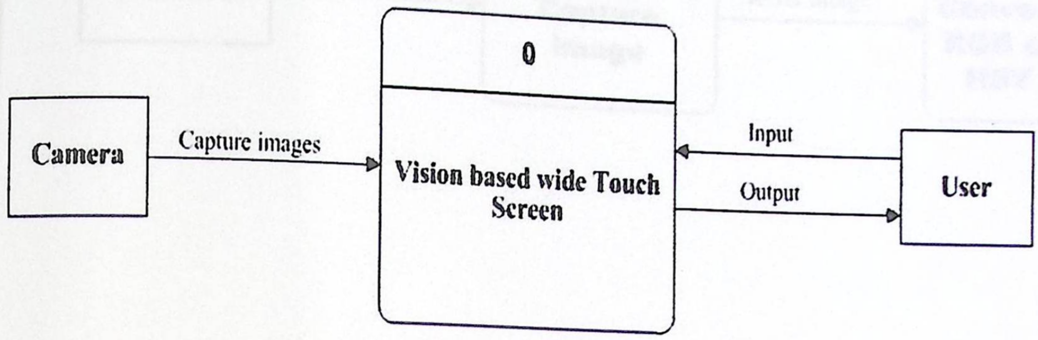


Figure 4.4: Context diagram

4.7. Data Flow Diagrams

The following diagram illustrates Level Zero of the data flow. It shows the data processes starting from the point of taking the image until the "event" take place.

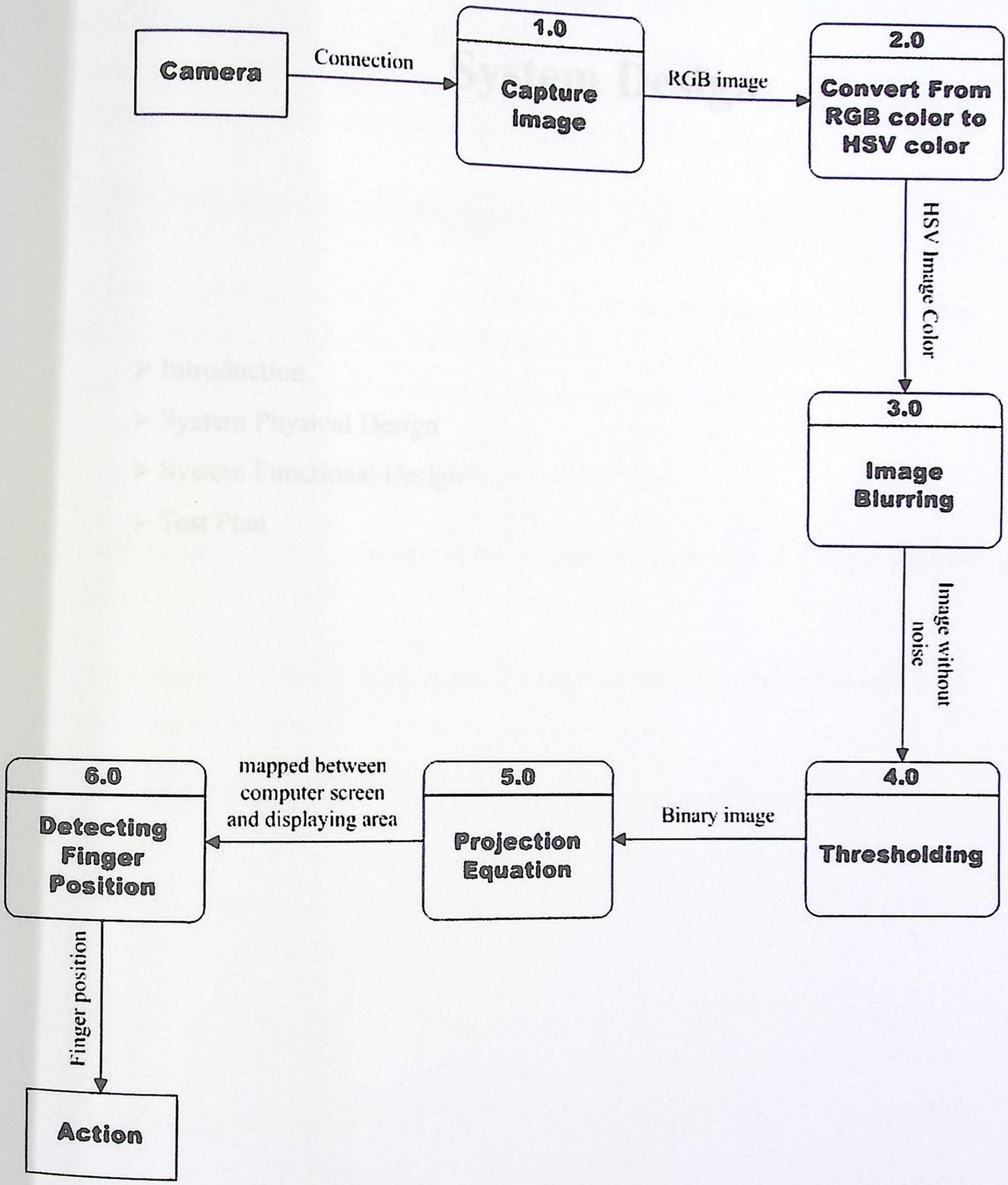


Figure 4.5: Dataflow Diagram

CHAPTER 5

System Design

- Introduction
- System Physical Design
- System Functional Design
- Test Plan

CHAPTER 5

System Design

- Introduction
- System Physical Design
- System Functional Design
- Test Plan

5.1. Introduction

This chapter focuses on the main design concepts in the Wide Vision Based Touch Screen. It also describes the main component of the system using block diagram to show how these components are connected with each other. Moreover, this chapter will discuss the main architecture of the system, how it works, and how it is modeled.

5.2. System Physical Design

We built a physical frame of the system as to setup the cameras on the frame, one on the X-Axis, and the other camera on the Y-Axis. This step was carried out in fear that any unintended movement would cause the system to malfunction.

5.2.1. Components of the physical frame

Aluminum frame: It consisted of two aluminum bars formed at a shape with 90-degree angle.

Two cameras: Placed on the frame, Camera-1 on the Y-Axis while Camera- 2 was placed on the X-Axis.

The following figure illustrates the physical equipments of the system:

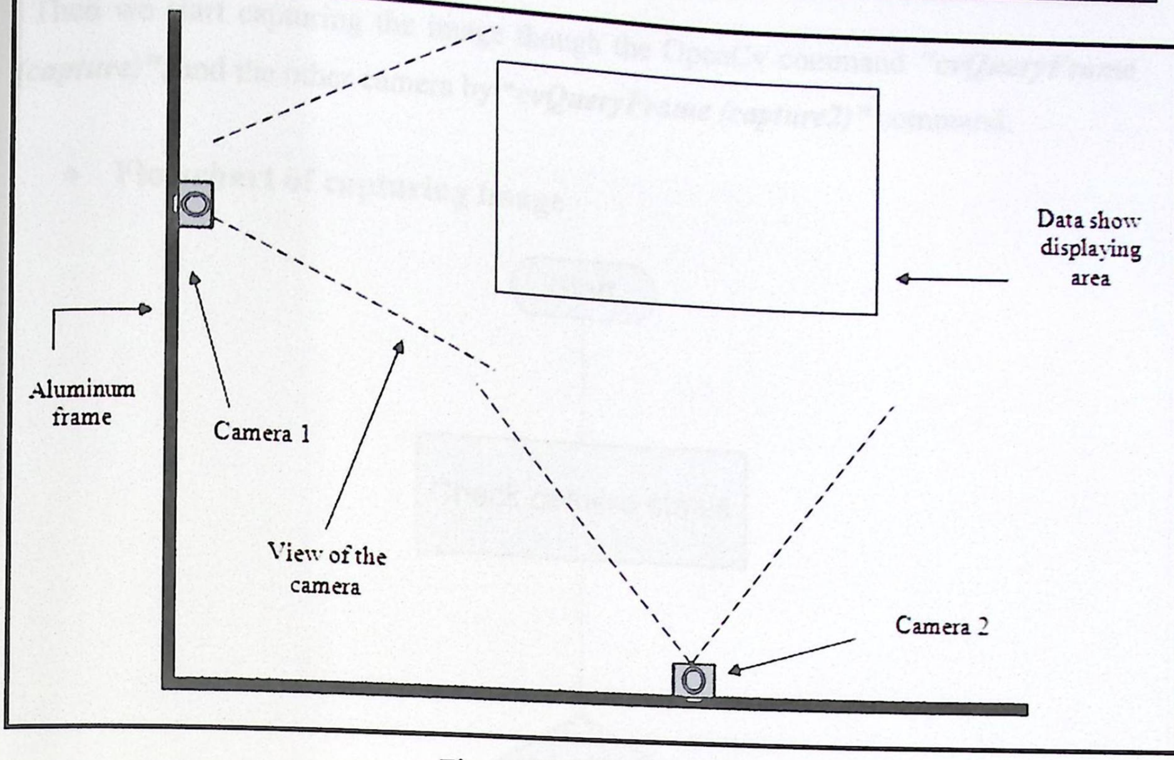


Figure 5.1: Physical design

5.3. System Functional Design

In this section we will clarify the functional design for our system through describing the algorithm and Pseudo code for each algorithm and explaining the mathematical equation used.

5.3.1. Capturing the Image

In this stage in our project we can capture the image by using two cameras fixed on the Frame, one on the x-axis (camera on the right side) and the other on the y-axis (camera on the left side).

The process of capturing the image is done through using OpenCv library. This process will determine the camera that will capture the image; the camera on y-coordinate can be determined by this command ***"cvCaptureFromCAM(0)",*** and x-Axis can be determined by ***"cvCaptureFromCAM(1)".***

Then we start capturing the image though the OpenCv command *"cvQueryFrame (capture)"*, and the other camera by *"cvQueryFrame (capture2)"* command.

- **Flowchart of capturing image**

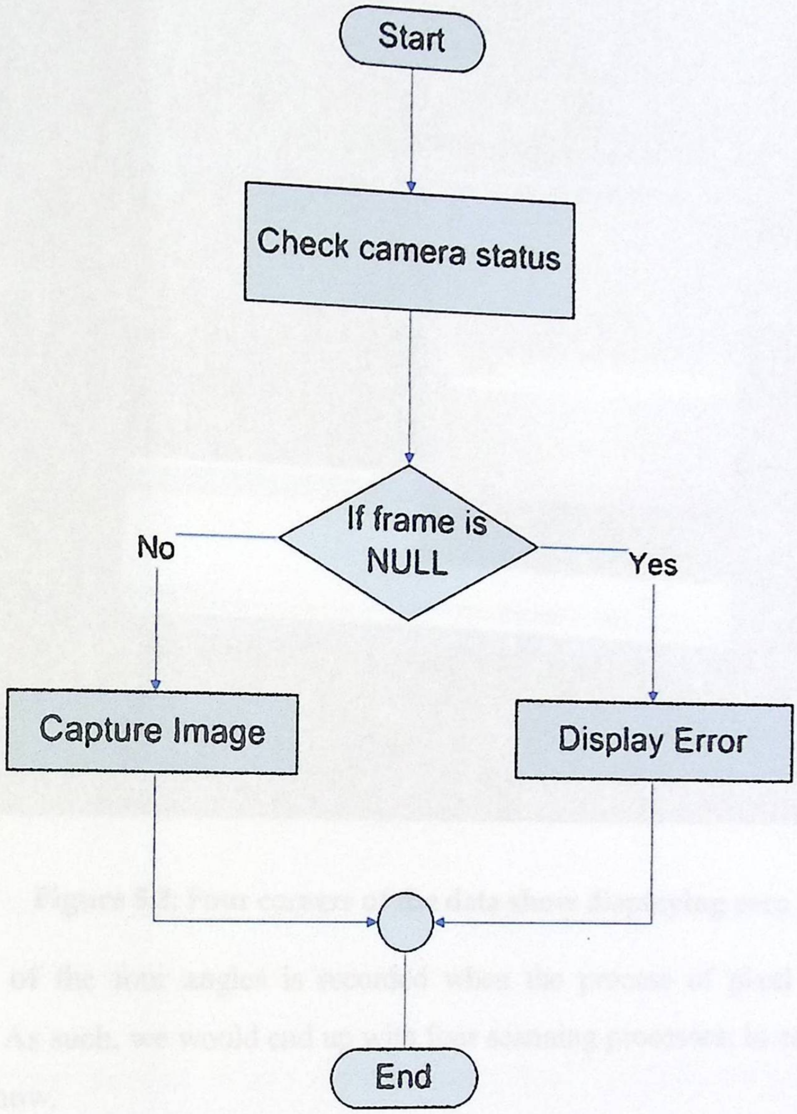


Figure 5.2: Capturing image

5.3.2. Detecting the four corners of the data show displaying area

When the system starts to function, the cameras record the readings concerning the locations of the four angles. It would, then, make a comparison between the computer screen and the data show displaying area using a specific mathematical equation (which will be listed later).

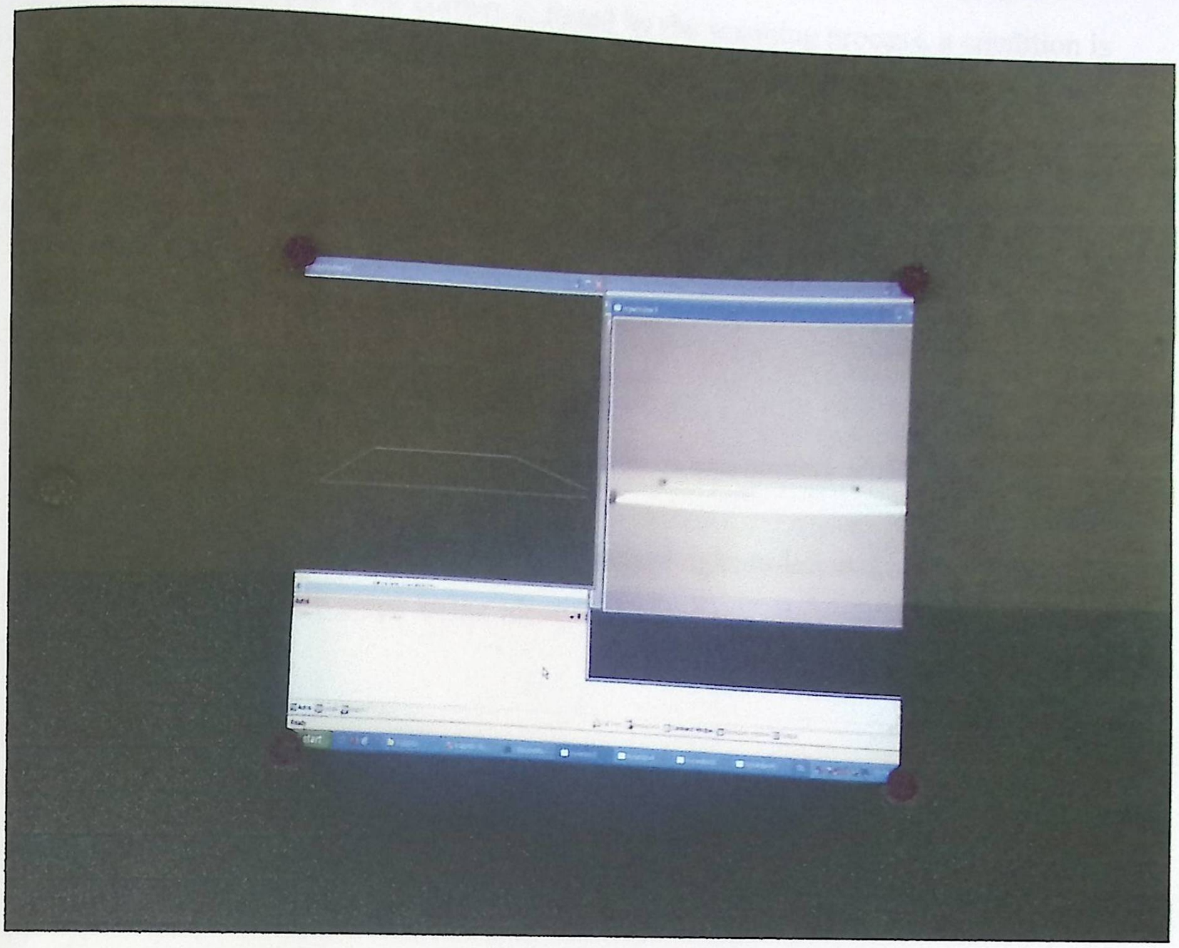


Figure 5.3: Four corners of the data show displaying area

Each value of the four angles is recorded when the process of pixel scanning is undertaken. As such, we would end up with four scanning processes; in case the angle of the datashow.

In order for the system to identify the corners of the display area, the team has placed 4 objects at each projected corner. The identified measurements will be used later to recognize the position of the finger.

In order to detect the corners of the display area, the system carries out the following steps:

1. At the first stage a simultaneous scanning process that goes from left-to-right and top-to-bottom takes place starting from the left side of the display area. When one of the four corners is found by the scanning process, a condition is being checked to see whether or not the point is located in the first half of the display. If so, the system saves a coordination point for it. If not, no coordination point is taken.
2. The second stage includes, again, a simultaneous scanning process in the opposite direction; right-to-left and bottom-to-up with the condition checking as listed in the first stage.

The result of these two steps is the identification of two corners of the display area.

Two additional similar steps but with the direction right-to-left and bottom-to-top. The end result of these two steps is the identification of the other two corners of display area.

After the implementation of the before-mentioned process, the four corners of the display area will be identified.

5.3.3. Process of take value of the colored process of determining alternate colors:

When the system starts to operate, the user uses a "tool" specially designed to record the values of the color of his/her finger. As such, if the user wanted to place a new color on his hand, he can place his finger at a specific location on the display area where the "tool" would take a range color for the finger and then assigns that color to be the finger color. On the other hand, if the user does not want to have a new color, the application would, then, get the old values of the color.

Figure 5.4: Taking the value of the colored finger

5.3.4. Convert from RGB image to HSV image.

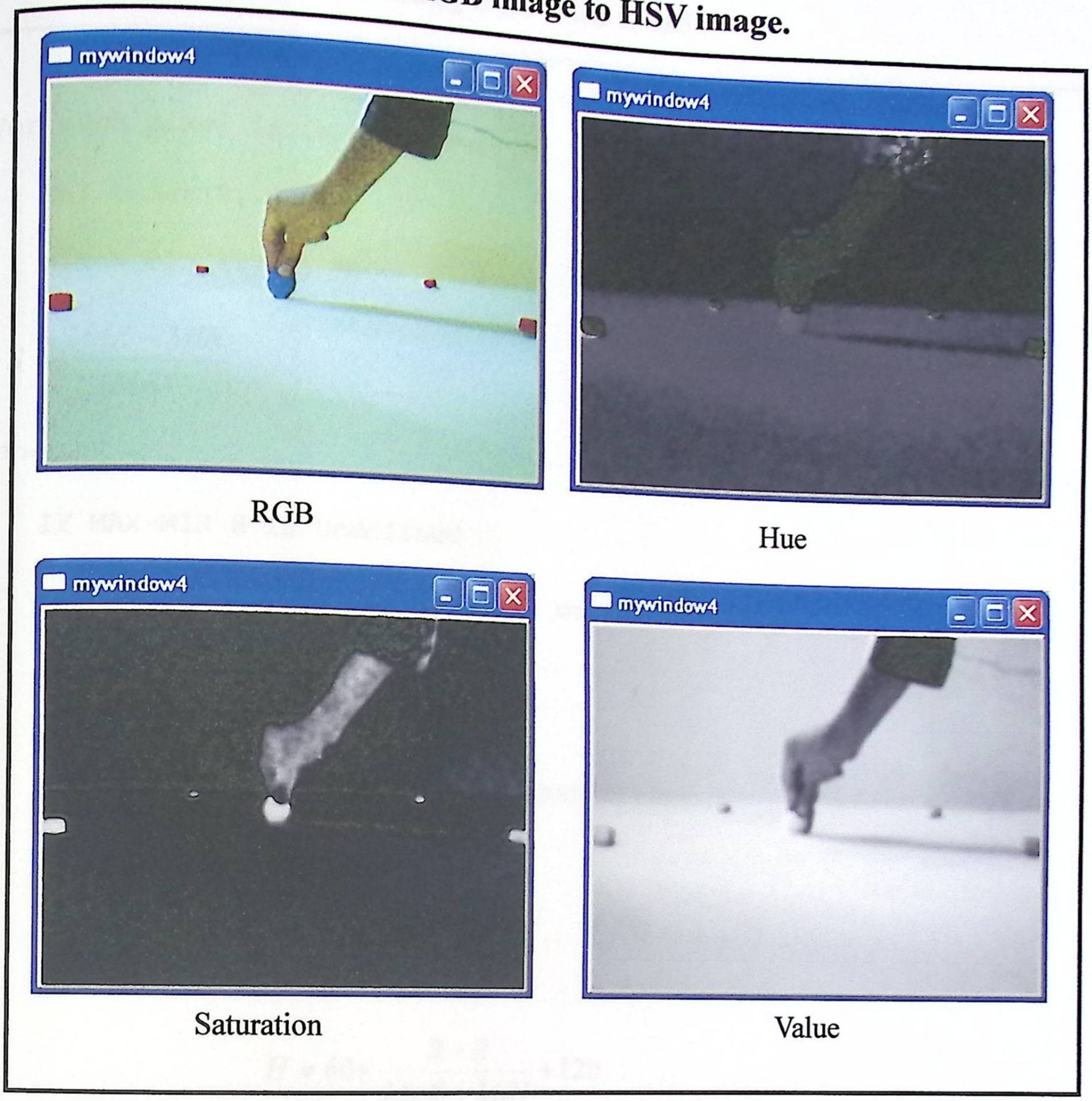


Figure 5.5: RGB to HSV

- **Pseudo code of converting from RGB to HSV**

```
Convert from RGB to HSV
For each pixel in image
   MAX = max(R, G, B)
   MIN = min (R, G, B)
```

$$S = \frac{MAX - MIN}{MAX}$$

```
   V=MAX
   IF MAX=MIN H is Undefined
     Else if Max =R and G greater or equal B then
```

$$H = 60 \times \frac{G - B}{MAX - MIN} + 0$$

```
       Else if MAX=R and G less than B then
```

$$H = 60 \times \frac{G - B}{MAX - MIN} + 360$$

```
         Else if Max=G then
```

$$H = 60 \times \frac{B - R}{MAX - MIN} + 120$$

```
           Else
```

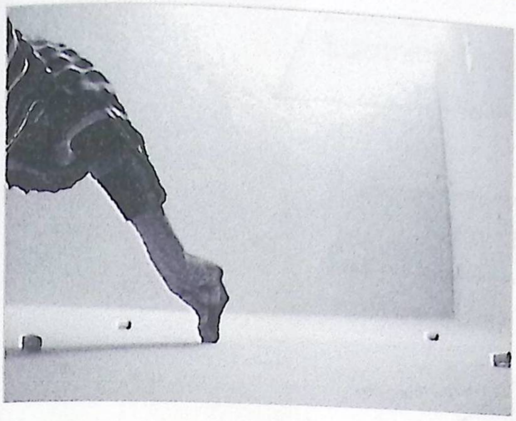
$$60 \times \frac{R - G}{MAX - MIN} + 240$$

```
   End if
End for
```

Pseudo code 1. Convert form RGB to HSV

5.3.5. Image Blurring

We use the following algorithm to enhance the image by eliminating the noise from it.



Before blurring

After blurring

Figure 5.6: image blurring

5.3.6. Process of Threshold

After applying the image blurring algorithm on the HSV image, the threshold algorithm is then applied on the H image and on the S image. A comparison is then carried between each pixel in the two resulting images.

- **Pseudo code of threshold**

```
Threshold

    For Each Pixel in Frame
        If Pixel (X, Y) is in Hue Range
            And in saturation Range Then
                    Set Pixel (X, Y) = White
                Else
                    Set Pixel (X, Y) = Black
            End If
    End For
   Return Black and white image
```

Pseudo code 2. Process of thresholding

- **Flowchart of threshold**

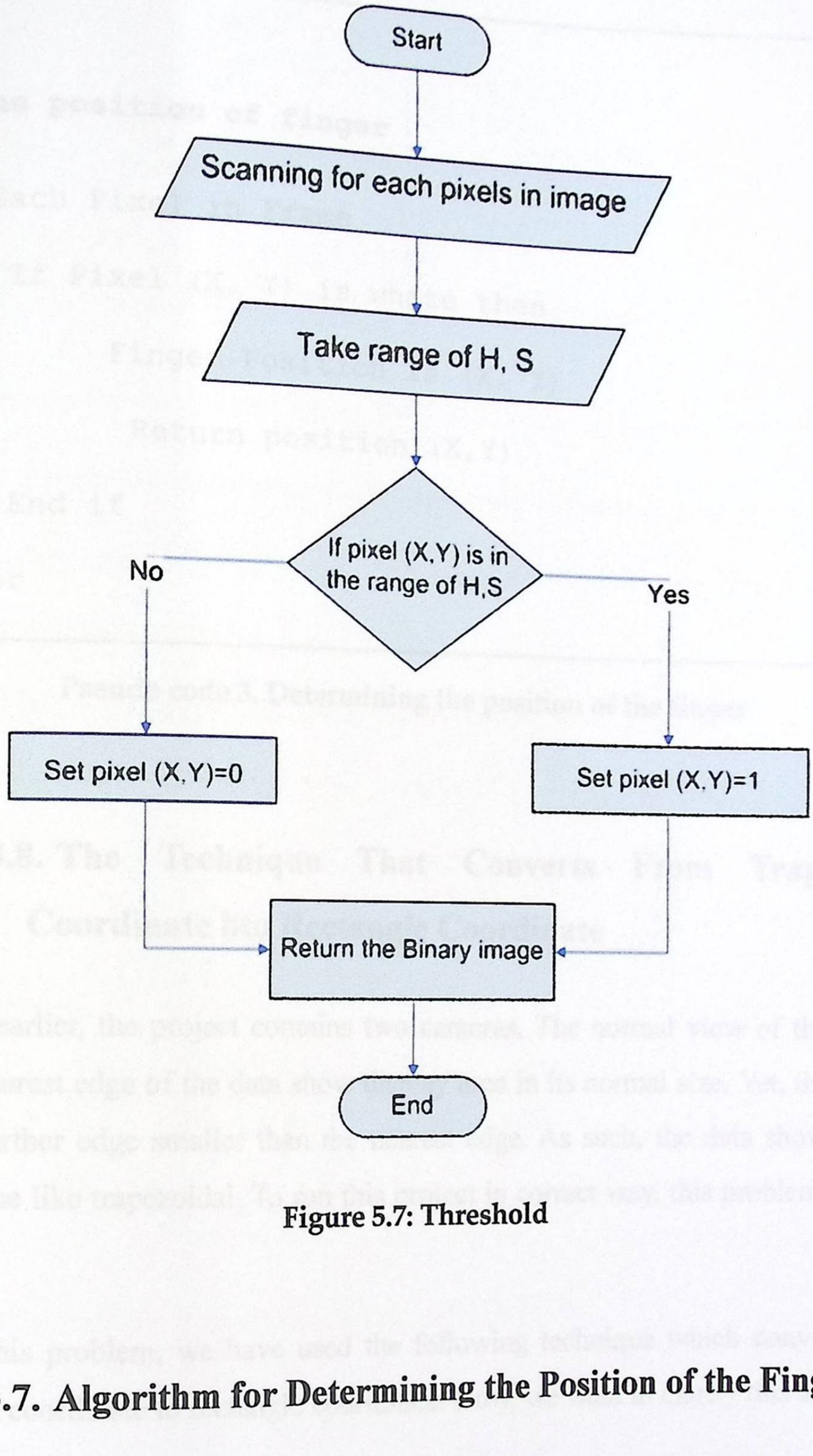


Figure 5.7: Threshold

5.3.7. Algorithm for Determining the Position of the Finger:

In this stage we can find the position of the finger by scanning all pixels in the image from down to top. When it finds the first white pixel, the algorithm will consider this pixel as the position of the finger.

- **Pseudo code for Determine position of the finger**

```
Determine position of finger

 For Each Pixel in Frame

     If Pixel (X, Y) is white then

          Finger Position is (X, Y)

           Return position (X,Y)

     End if

 End For
```

Pseudo code 3. Determining the position of the finger

5.3.8. The Technique That Converts From Trapezoidal Coordinate bto Rectangle Coordinate

As stated earlier, the project contains two cameras. The normal view of the camera sees the nearest edge of the data show display area in its normal size. Yet, the camera sees the farther edge smaller than the nearest edge. As such, the data show display area become like trapezoidal. To run this project in correct way, this problem must be resolved.

To solve this problem, we have used the following technique which converts from trapezoidal coordinate to rectangle coordinate. Now, we want to clarify this equation:

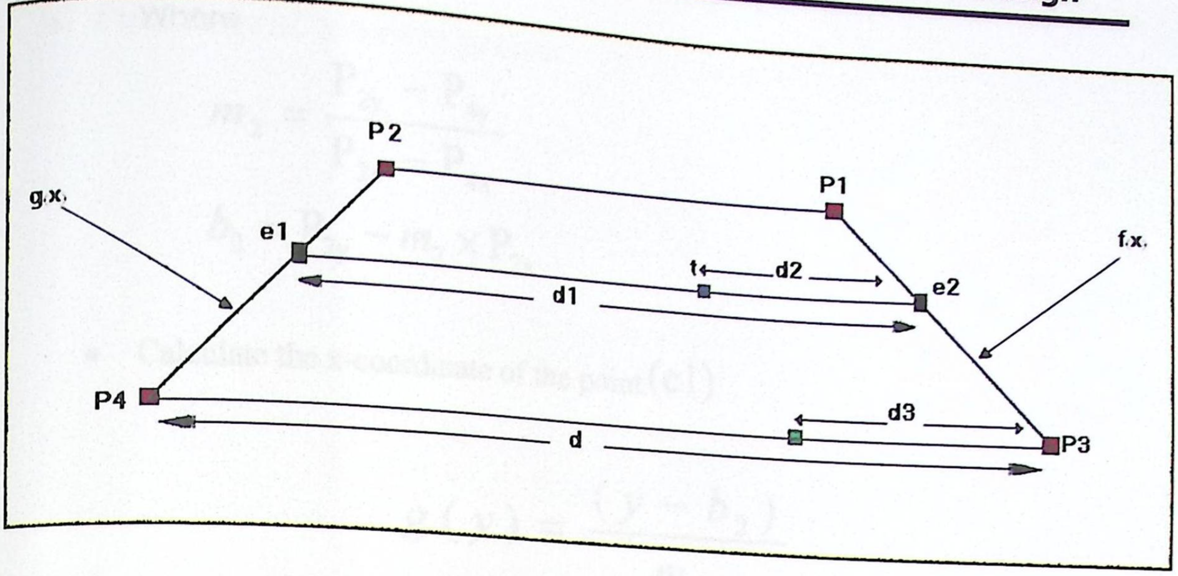


Figure 5.8: Converting from trapezoidal to rectangle variables

- When the program runs, it determines the four corners of the data show displaying area (P_1, P_2, P_3, P_4)
- Then it determines the finger position (t) on the display area.
- The line equation between the points (P_1, P_3) is calculated:

$$f(x) = m_1 x + b_1$$

Where

$$m_1 = \frac{P_{1y} - P_{3y}}{P_{1x} - P_{3x}}$$

$$b_1 = P_{1y} - m_1 \times P_{1x}$$

- Calculate the x-coordinate of the point (e2)

$$f(y) = \frac{(y - b_1)}{m_1}$$

- The line equation between the points (P_2, P_4) is calculated

$$g(x) = m_2 x + b_2$$

Where

$$m_2 = \frac{P_{2y} - P_{4y}}{P_{2x} - P_{4x}}$$

$$b_2 = P_{2y} - m_2 \times P_{2x}$$

- Calculate the x-coordinate of the point (e1)

$$g(y) = \frac{(y - b_2)}{m_2}$$

- Then calculate the ratio (r) between distances (d) and (d1).

$$r = \frac{d}{d_1}$$

Where the distance between any two points (x_1,y_1) and (x_2,y_2) is calculated as follows:

$$distance = \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2}$$

- And then calculate the distance (d2) between (e2, t).

- Then calculate the distance (d2) in the normal size (d3).

$$d_3 = r * d_2$$

- This process is applied on the two cameras (left camera and bottom camera). d3 is the final coordination that we need. For left camera, it represents the y-coordinate, and for bottom camera, it represents the x-coordination.

- **Flowchart of previous process**

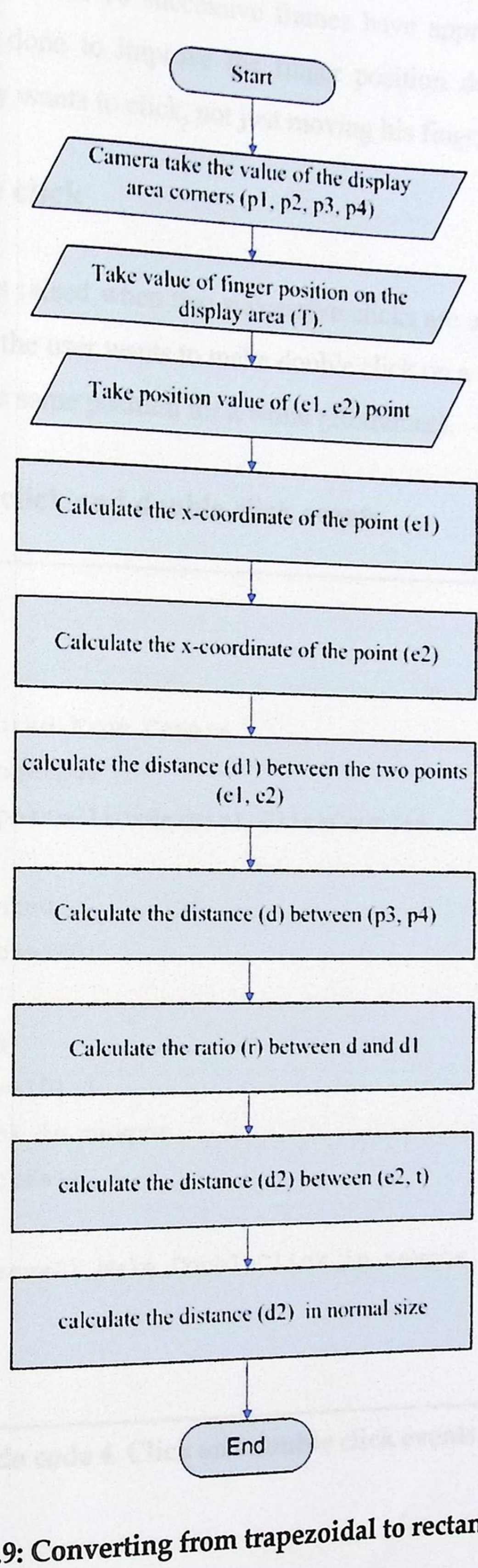


Figure 5.9: Converting from trapezoidal to rectangle

5.3.9. Click Events

The click event is raised when 10 successive frames have approximately the same finger position. This is done to improve the finger position determination and to ensure that the user really wants to click, not just moving his finger.

5.3.10. Double click

The double click event is raised when two successive clicks are approximately in the same position. So, when the user wants to make double click on a specific position, he must hold pressing on the same position for a while (20 frames).

- **Pseudo code of click and double click events**

```
ClickCount=0
ClickMargin=6
NumOfClicks=0
For Each Frame Captured From Camera
    Newpos = Getfingerpos
    If (newpos-oldpos<=ClickMargin) ClickCount++
  Else {
          ClickCount=0
          NumOfClickes=0;
    }
    Oldpos = newpos
    If (ClickCount==10) {
          Make Click in newpos
          NumOfClickes++
    }
    If (NumOfClickes==2) Make DoubleClick in newpos
End For
```

Pseudo code 4. Click and double click events

5.3.11. Drag and Drop

Drag and Drop is a composite event, it is composed of three events with this order: mouse down, mouse move and mouse up.

When the user wants to do a drag and drop using his ordinary mouse, first he will click and hold the click button (mouse down), and then moves the mouse to the destination position (mouse move), and then releases the button (mouse up).

In this project, the drag and drop is done with same way, but the different thing is how to determine when the user clicks (mouse down) and when he releases (mouse up). To do that, the user puts his finger in the desired position on the datashow display area, and remains his finger in the same position to make a mouse down event, and then the user can move his finger to the destination position which corresponds to mouse move event, finally, he must remains his finger in that position to make a mouse up event.

- **Drag and Drop Algorithm:**

```
1. Wait for 10 successive frames with the same finger position
2. Send mouse down event to the mouse with the finger position
3. Start mouse move event
4. Wait for 10 successive frames with the same finger position
5. Send mouse up event to the mouse with the finger position
```

Pseudo code 5. Drag and Drop algorithm

5.4. Test Plan

The process of testing the system is the most important stage throughout the development process. The importance of testing the system is to verify the reliability of each unit

Each part of the system is designed to make sure it serves the specifications and requirements of the experiment; here the project team describes briefly the methodology that the team has adapted to test the system steps that will be followed in the system testing as described below:

1. Unit testing:

Unit testing is one of the testing types that depend on separating or dividing the system into components that each one of them will be tested separately to ensure that they are operating correctly and that they meet the specification.

2. Integration testing:

The integration testing which depends on testing all components together as a whole testing to ensure that the system works properly and meet the specification .

3. System testing:

It is one of the most important stages at all software projects as it aims to ensure that the system meets its specifications and is working as properly as expected.

6.1. Introduction

In this chapter we will describe, in detail, the process of implementing the system in its environment. During the evolution and the development of the modern technology, it becomes important to develop the appropriate environment in order to implement and operate projects. Such environment consists of hardware and software components that are required for the operation process. This chapter describes software and hardware used in the system for the operation phase, and how they are installed.

6.2. Hardware Implementations

The Wide Vision Based Touch Screen project consists of many hardware components including:

- Computer.
- Data show.
- Cameras (FireWire digital camera).
- FireWire Connection Cables.
- Physical Frame.

6.2.1. Laptop computer with the following specifications

- Centrino Due.
- CPU with 1.66 GHz (2 CPU).
- Memory size 512 MB, at least to deal with data that need to be a large. memory capacity.
- HD 80 GB.
- Monitor 15.

6.1. Introduction

In this chapter we will describe, in detail, the process of implementing the system in its environment. During the evolution and the development of the modern technology, it becomes important to develop the appropriate environment in order to implement and operate projects. Such environment consists of hardware and software components that are required for the operation process. This chapter describes software and hardware used in the system for the operation phase, and how they are installed.

6.2. Hardware Implementations

The Wide Vision Based Touch Screen project consists of many hardware components including:

- Computer.
- Data show.
- Cameras (FireWire digital camera).
- FireWire Connection Cables.
- Physical Frame.

6.2.1. Laptop computer with the following specifications

- Centrino Due.
- CPU with 1.66 GHz (2 CPU).
- Memory size 512 MB, at least to deal with data that need to be a large. memory capacity.
- HD 80 GB.
- Monitor 15.

6.2.2. Data Show

Any data show can be used, only if it is flexible in order to control dimensions.

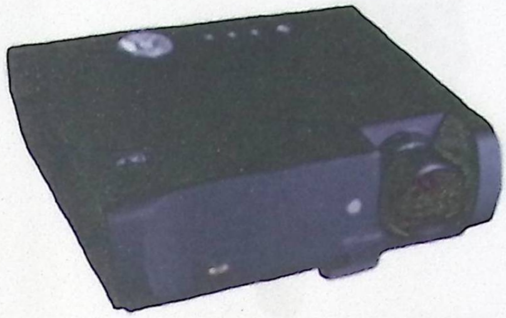


Figure 6.1: Data show [6]

6.2.3. Cameras (Fire-i)

- These cameras have the following features:
  - VGA Camera (640×480) resolution.
  - Full motion video at up 30 frames per second.
  - High quality bright.
  - Focus range: 1cm to infinity.
  - White balance: automatic or manual control.

- **Camera Requirements:**
  - Windows XP/Vista/Linux
  - 1 GHz or faster processor.
  - 512 MB RAM.

Figure 6.2: FireWire Camera [13]

6.2.4. FireWire connection cables

Type of cables that is used to connection between computer and camera is Fire-wire.

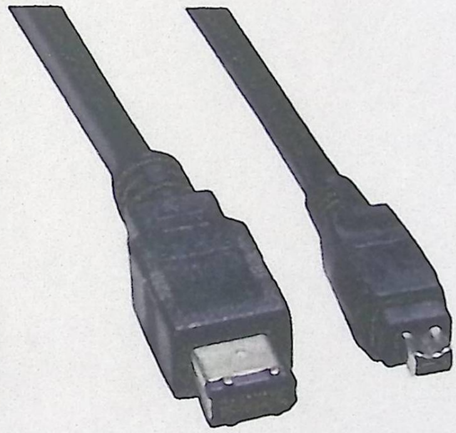


Figure 6.3: FireWire connection cables[12]

6.2.5. Physical Frame

Frame consists from two part of aluminum with constant size, used to fixing cameras on it.

Figure 6.4: Physical frame

6.3. Software Implementations

The Software was used in the process of building and development of the system. It includes:

- Windows XP Professional.
- Microsoft Visual Studio.Net 2005.
- C\C++ Language
- OpenCv library.

6.3.1. Windows XP Professional

Windows XP is an Operating System created by Microsoft. This operating system is characterized by high performance when it used and it is compatible with files as NTFS5, NTFS4, FAT16 and FAT32. It also supports all the languages of the world and supports a large number of application and software including the internet.

6.3.2. C programming language

In our project we use C programming language.

- **Why Using C**

C is one of the most widely used third generation programming languages. Its power and flexibility ensures that it is still the leading choice for almost all areas of applications, especially in the software development environment. Many applications are written in C or C++, including the compilers for other programming languages. It is the language of many operating systems that written in including UNIX, DOS and Windows. It continues to adapt to new users, the latest being Java, which is used for programming Internet applications. C is a robust language, it is flexible and portable and it can produce fast, compact code. It, also, provides the programmer with objects to create and manipulate complex structures (e.g. classes in C++) and low level routines to control hardware. It is also one of the few languages that have an international standard, ANSI. [7]

- **Disadvantage of C**

"Programming languages like C have philosophy like "Programmers know what to do". It does not help or assist or warn you about anything, unless there is some syntactical error. Either there is some type mismatch, some macro redefinition, or some array's index goes out of bound". [9]

6.3.3. Microsoft Visual Studio.Net 2005

It can be used to develop, console and use graphical interface applications along with Windows Forms applications, web sites, web applications, and web services in both native code together with managed code for all platforms supported by Microsoft Windows, Windows Mobile, .NET Framework,

Visual Studio supports languages by means of language services, which allows any programming language to be supported (to varying degrees) by the code editor and debugger (provided a language-specific service has been authored).

- **The Reasons of Using Visual Studio.Net 2005**
  - Ease of use.
  - Integrated development environment.
  - Supports multiple languages within the project.
  - Compatibility with windows environments.
  - Supports debugging, tracing and error handling.

- **The following is the starting of a new project in Visual Studio.Net**

- **Start the program from Start menu.**

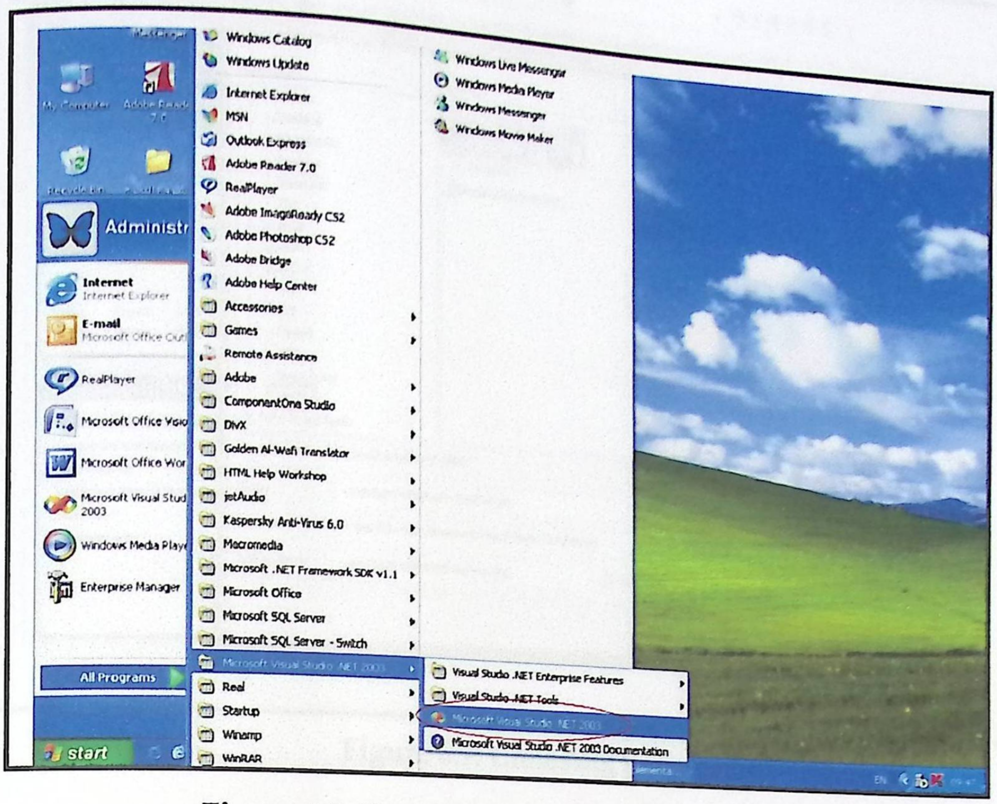


Figure 6.5: Start Visual Studio 2005 from Start menu.

- **Create a new project in Visual Studio 2005**

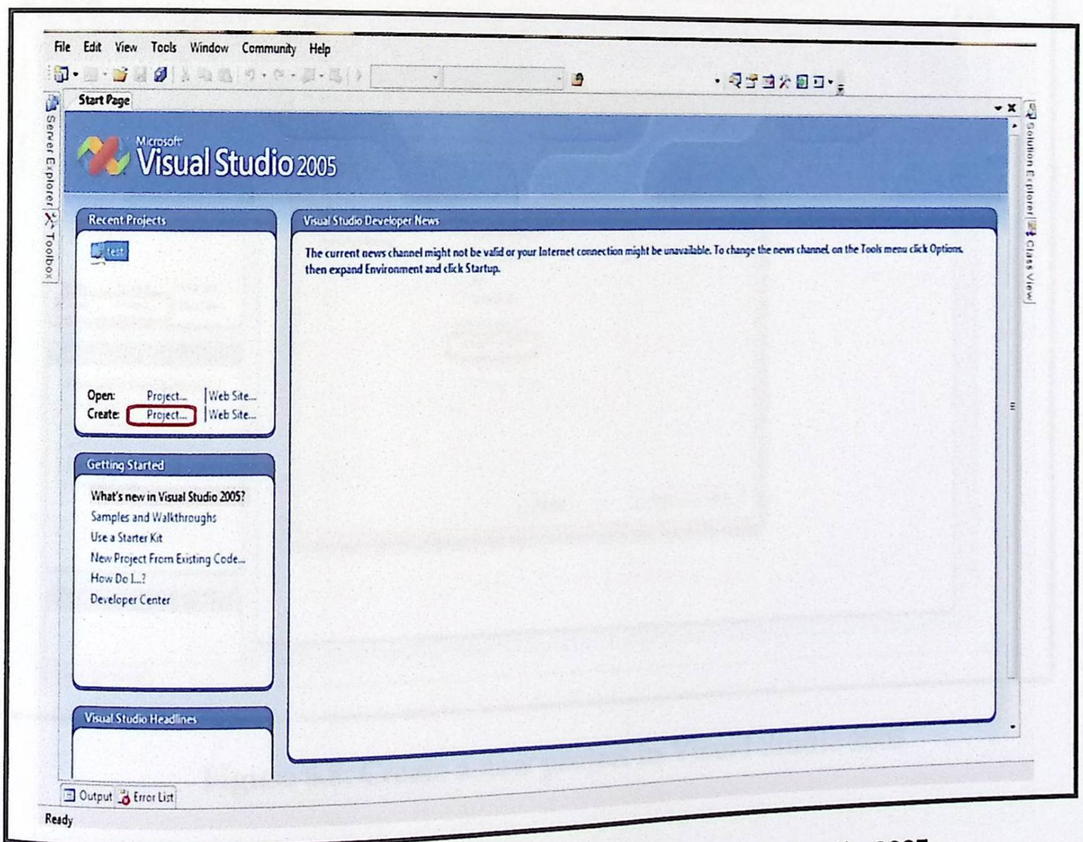


Figure 6.6: Create a new project in Visual Studio 2005

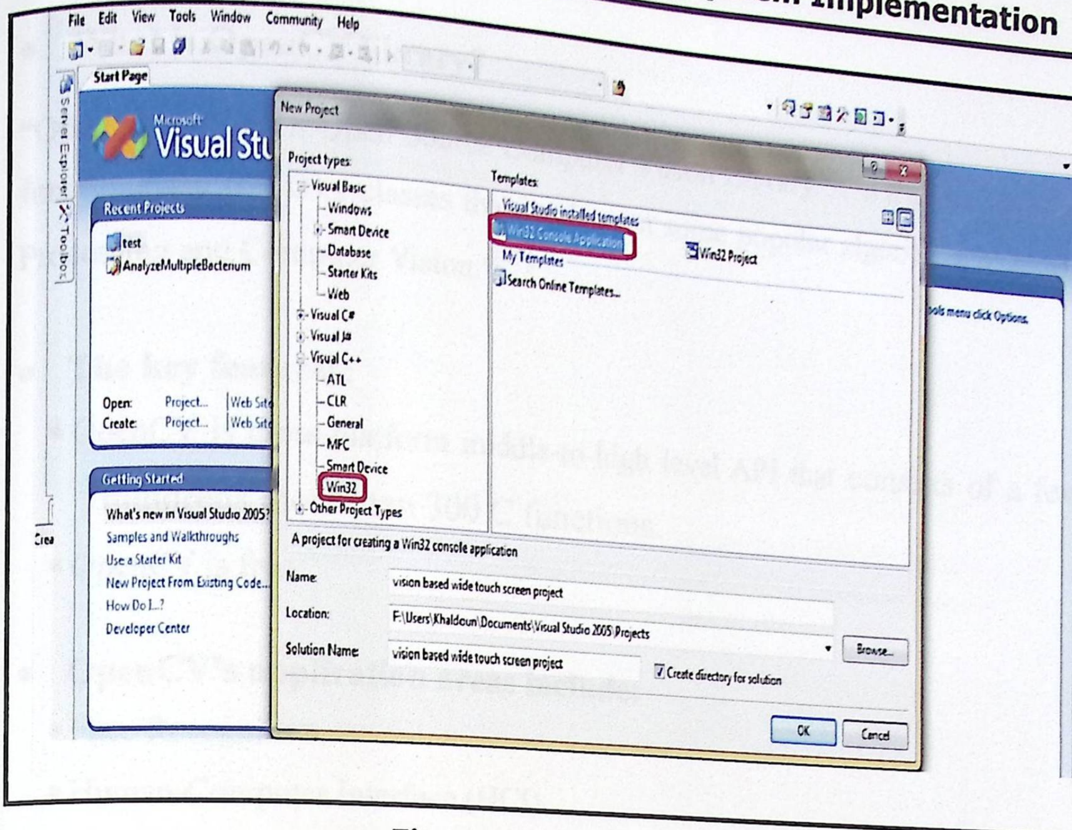


Figure 6.7: Choosing project type

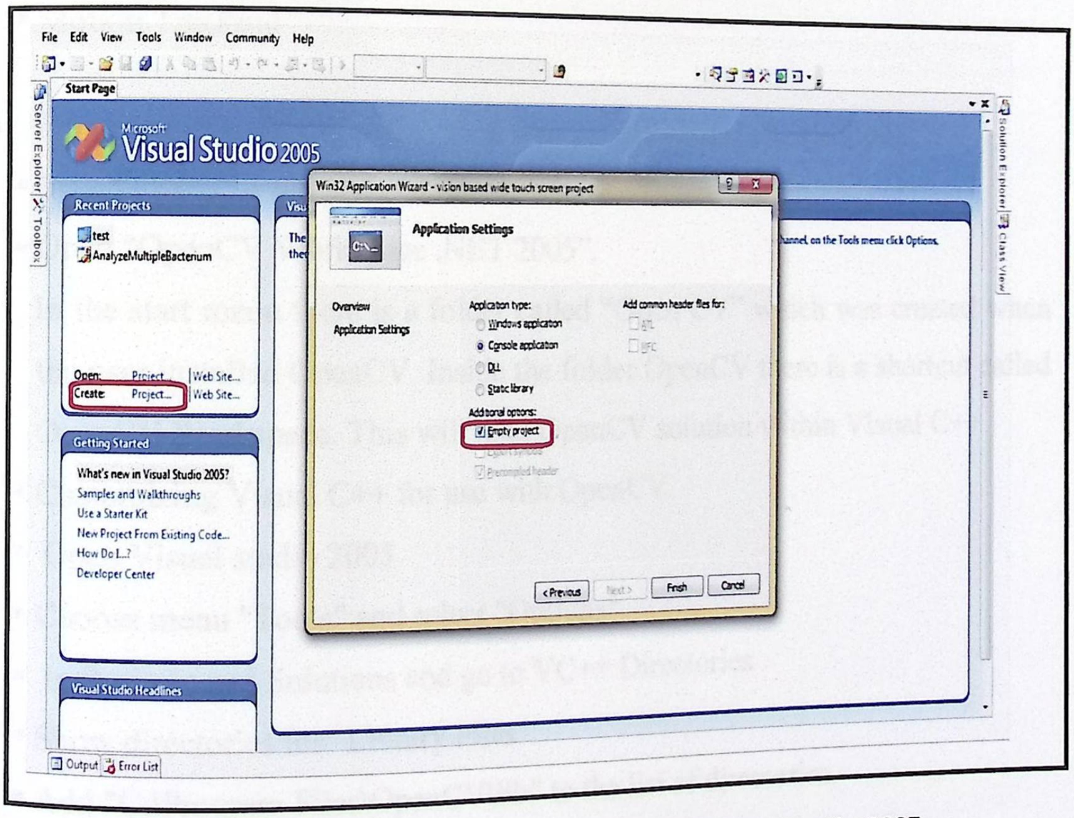


Figure 6.8: Create a new project in Visual Studio 2005

- **What is OpenCV library?**

"OpenCV stands for Open Source Computer Vision Library. It is a collection of C functions and few C++ classes that implement some popular algorithms of Image Processing and Computer Vision."[20]

- **The key features:**
  - OpenCV is cross-platform middle-to-high level API that consists of a few hundreds more than 300 C functions.
  - OpenCV is free

- **OpenCV's application areas include:**
  - Face Recognition.
  - Human-Computer Interface (HCI).
  - Mobile robotics.
  - Segmentation and Recognition.
  - Motion Tracking

- **How to connect OpenCv with Microsoft visual studio 2005**
  - Install OpenCV library.
  - Open "OpenCV Workspace .NET 2005".

  In the start menu there is a folder called "Open CV" which was created when the user installed OpenCV. Inside the folder OpenCV there is a shortcut called OpenCV Workspace. This will load OpenCV solution within Visual C++.
  - Customizing Visual C++ for use with OpenCV.
  - Open Visual studio 2005
  - Choose menu "Tools" and select "Options"
  - In Projects and Solutions and go to VC++ Directories
  - Show directories for "Library Files".
  - Add "C:\Program Files\OpenCV\lib" to the list of directories.

- **Select OpenCv workspace 2005.**

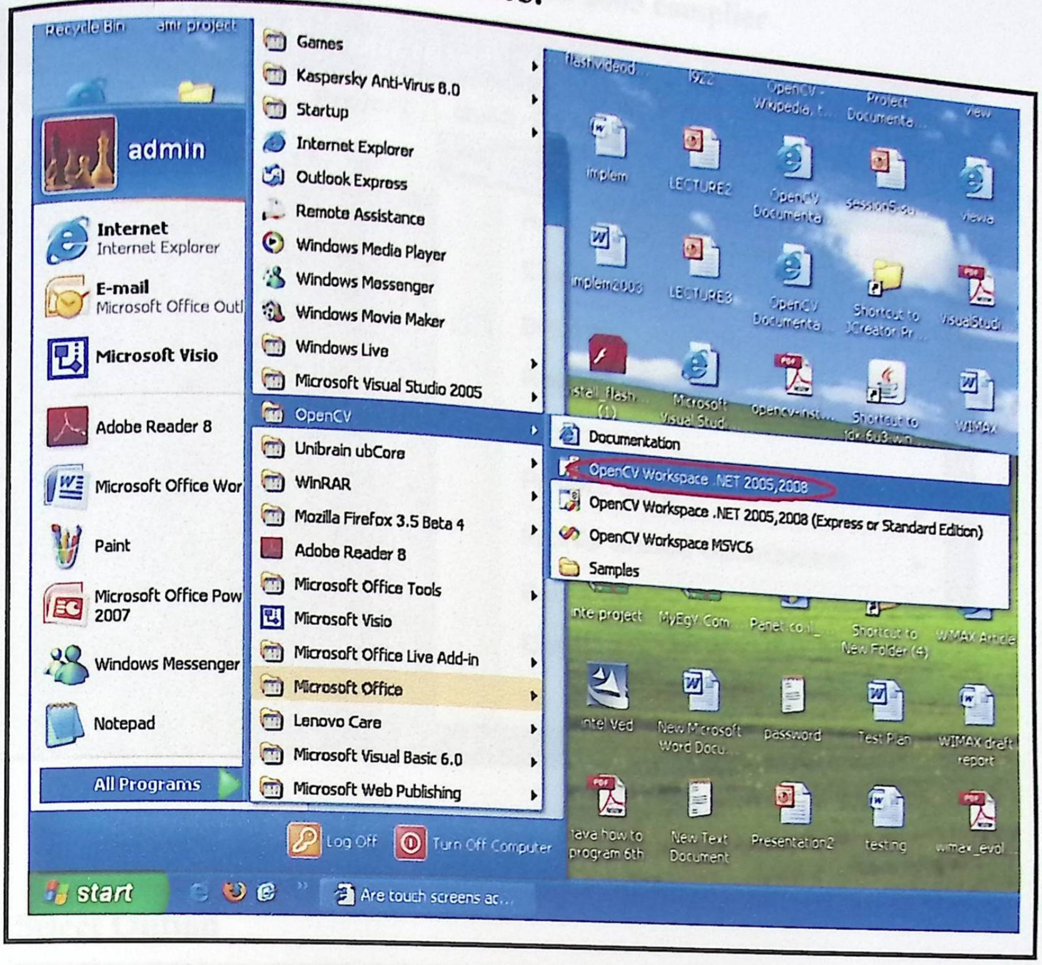


Figure 6.9: Start OpenCv

- **Build OpenCv library on visual studio 2005 complier**

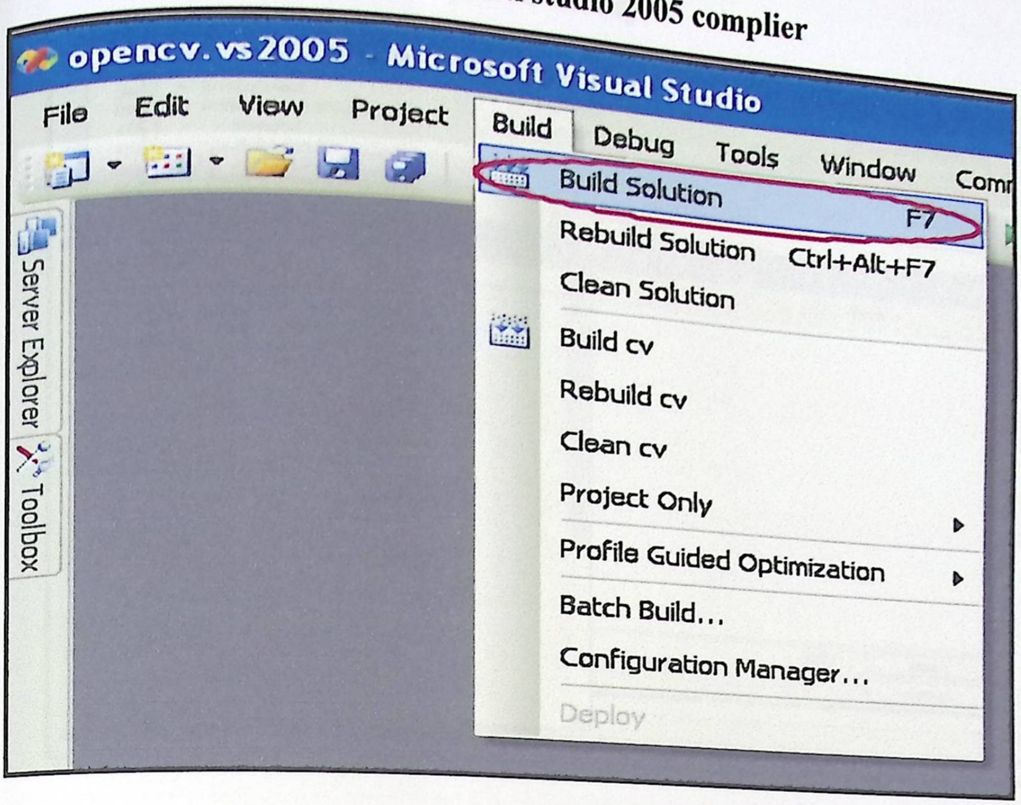


Figure 6.10 Build OpenCv library

- **Select Option**

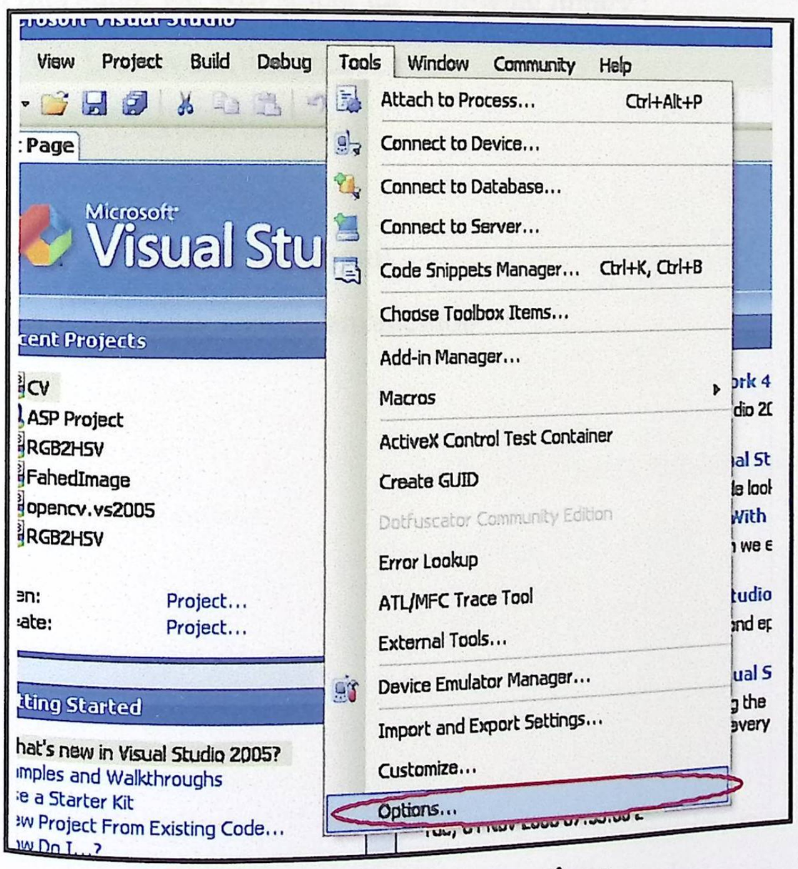


Figure 6.11: Select options

- **Select Executable Files and add library.**

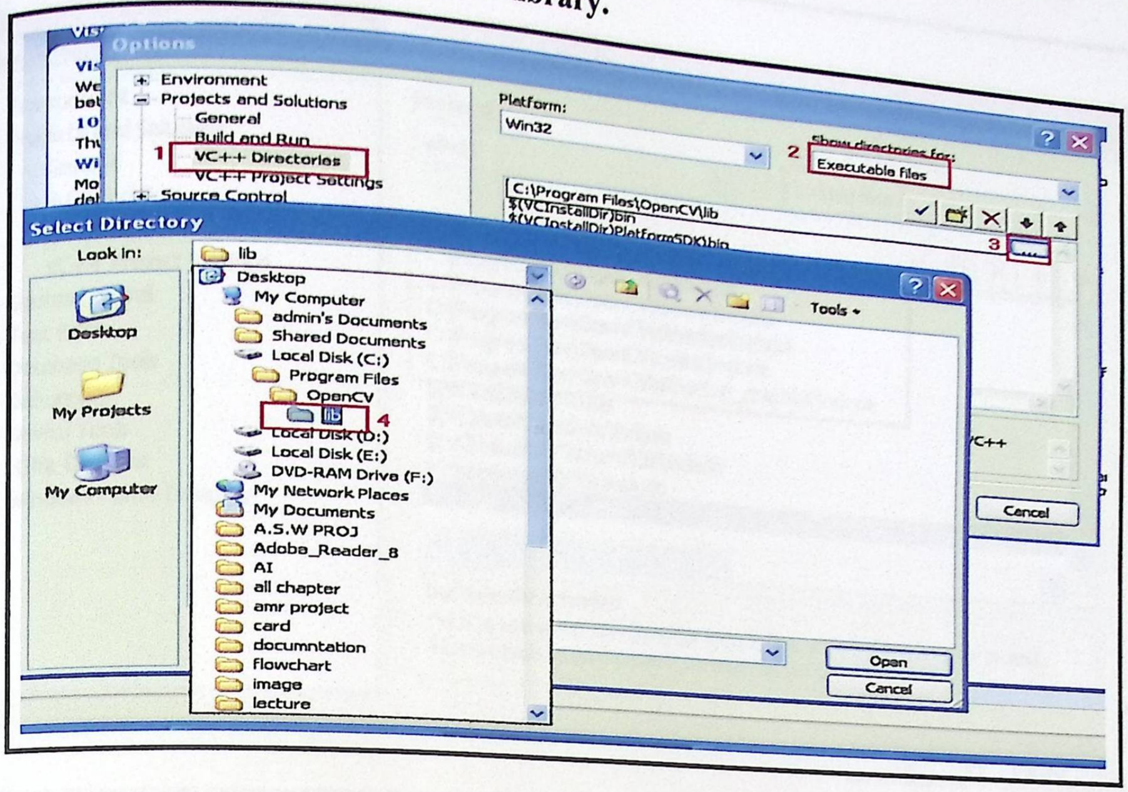


Figure 6.12: Select Executable files and add library

- In this stage we will select the following library :
  - \..\..\cvaux\include\
  - ..\..\cxcore\include\
  - ..\..\cv\include\
  - ..\..\otherlibs\highgui\
  - ..\..\otherlibs\cvcam\include

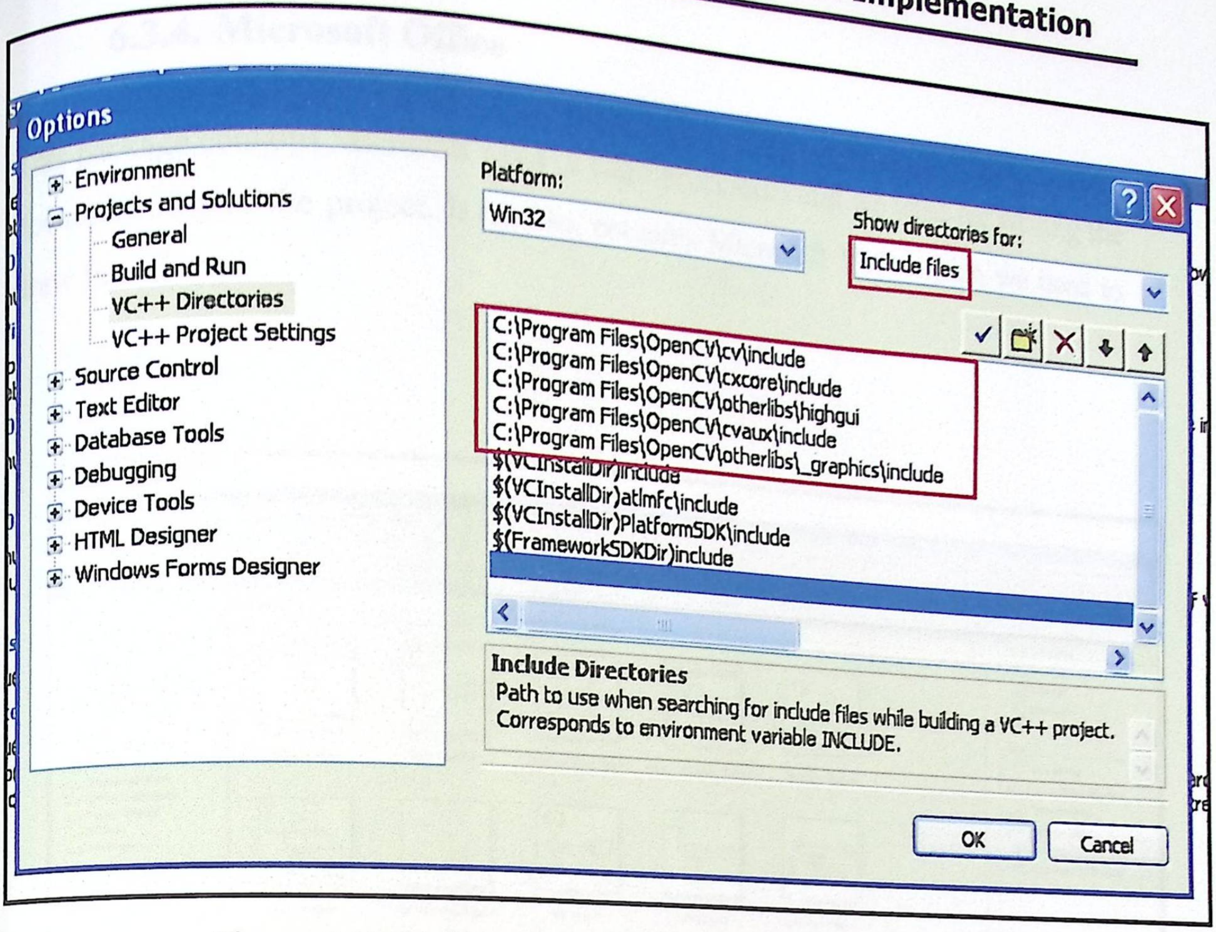


Figure 6.13: Linking OpenCv Library with Visual Studio 2005

- **Select Library files and adding library.**

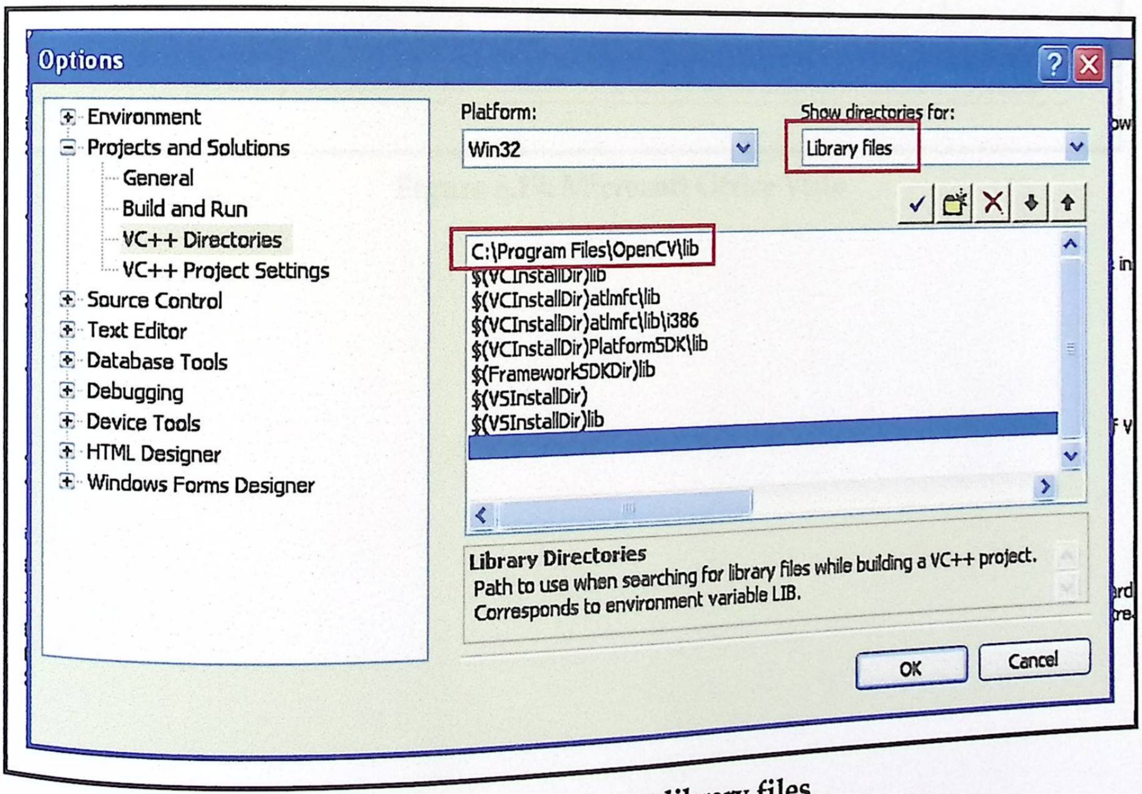


Figure 6.14: Select library files

6.3.4. Microsoft Office

This package contains Microsoft word (a word processor) that we used for writing the documentation of the project. It is, also, contains Microsoft Visio which we used to draw flowcharts.

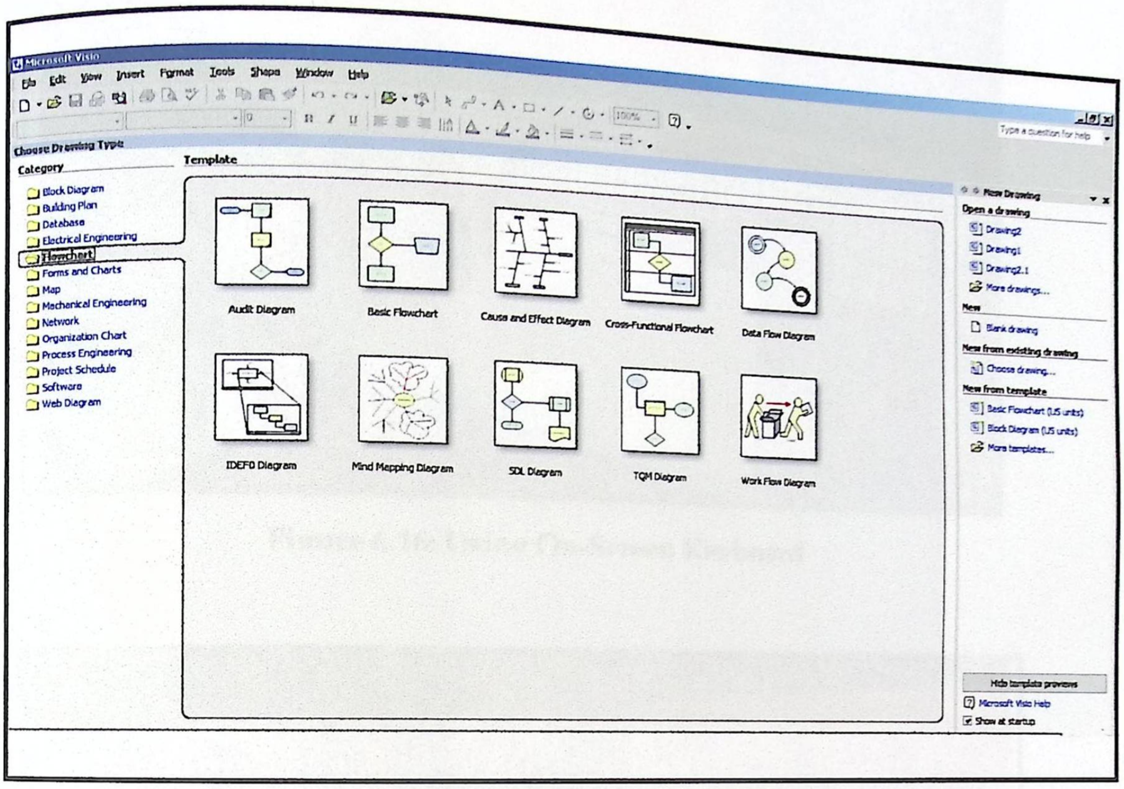


Figure 6.15: Microsoft Office Visio

- **Snapshots of the project**

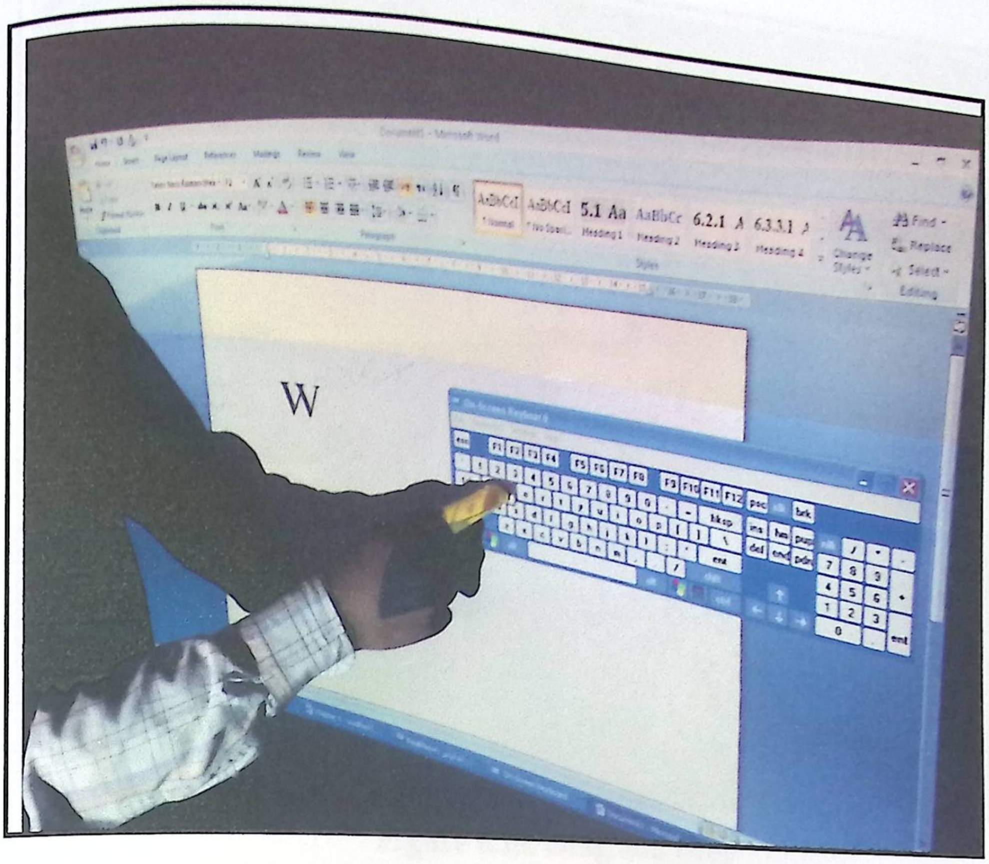


Figure 6.16: Using On-Screen Keyboard

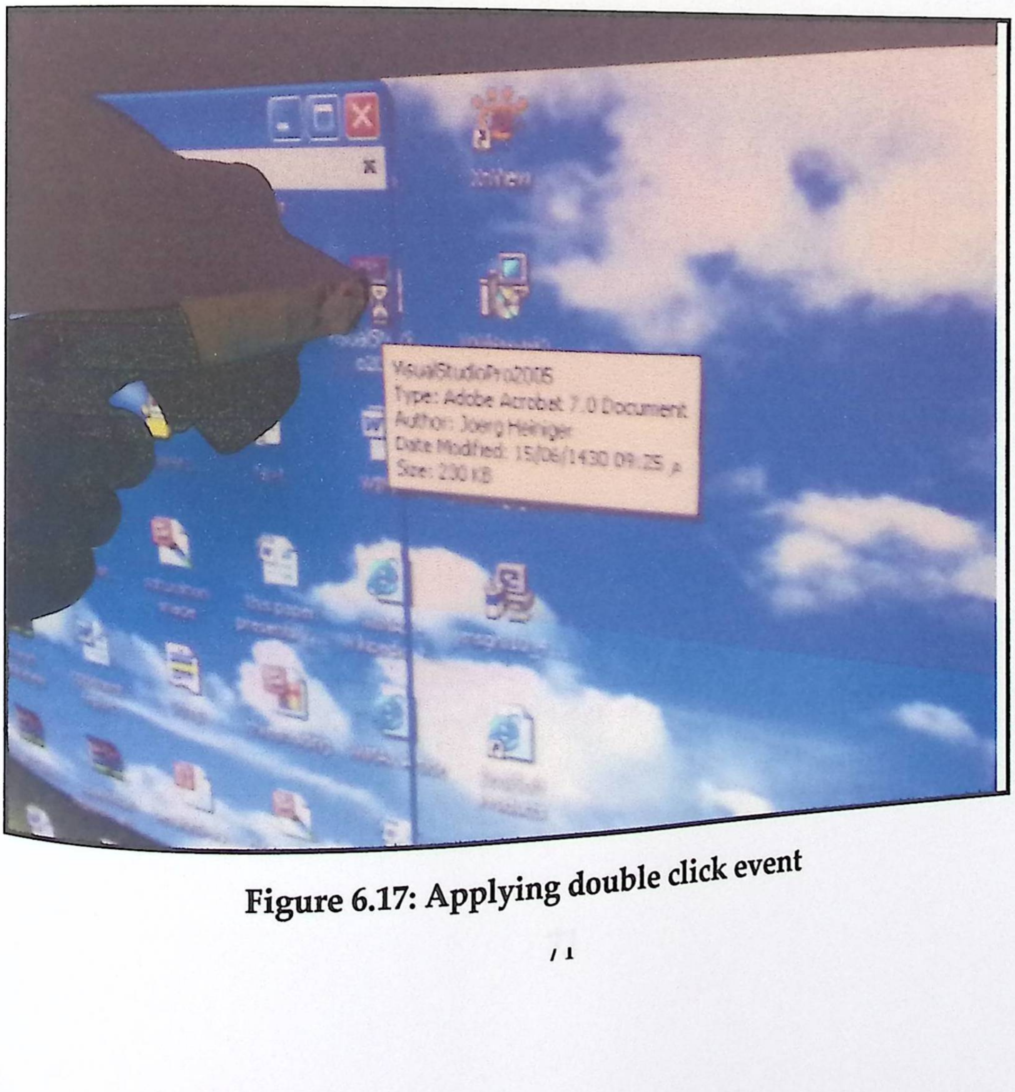


Figure 6.17: Applying double click event

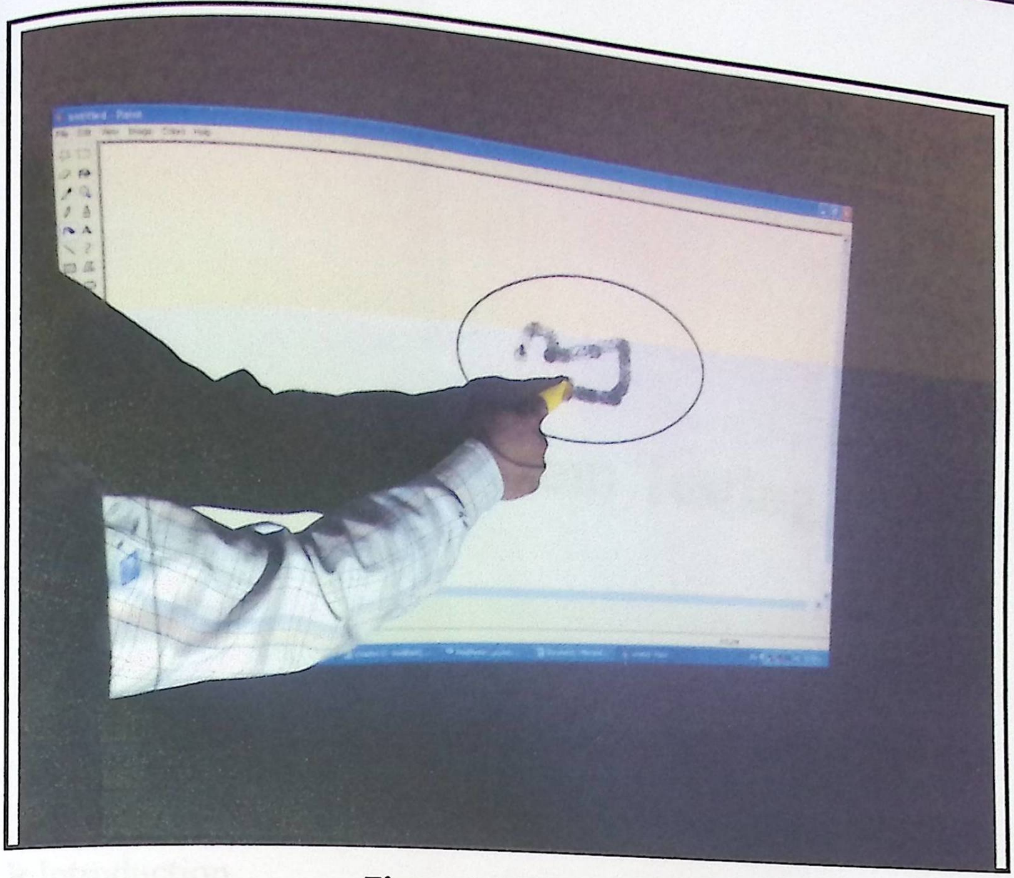


Figure 6.18: Drag and Drop

CHAPTER 7

System Testing

- ➢ Introduction
- ➢ Unit Testing
- ➢ Integration Testing
- ➢ System Testing

7.1. Introduction

System testing is the most important step we must do before deliver the system ,to ensure it worked as we exactly expected ,and to ensure that it meet all requirements that we specified before.

In Wide Vision Based Touch Screen, system testing is considered to be very important in checking the efficiency and the effectiveness of the system. It should be taken into consideration that any error may affect the whole system and consequently any other related errors may appear.

The testing process includes three levels:

- Unit testing.
- Integration testing.
- System testing

7.2. Unit Testing

Unit testing is one of the testing types that depend on separating or dividing the system into components that each one of them will be tested separately to ensure that each component function in a correct manner.

We started testing each unit of the system separately in order to ensure that it meets the specifications.

7.2.1. Testing the ability of the cameras to take image

Figure7.1 shows that the team has tested the ability of each camera separately to capture images. The testing was positive for all cameras.

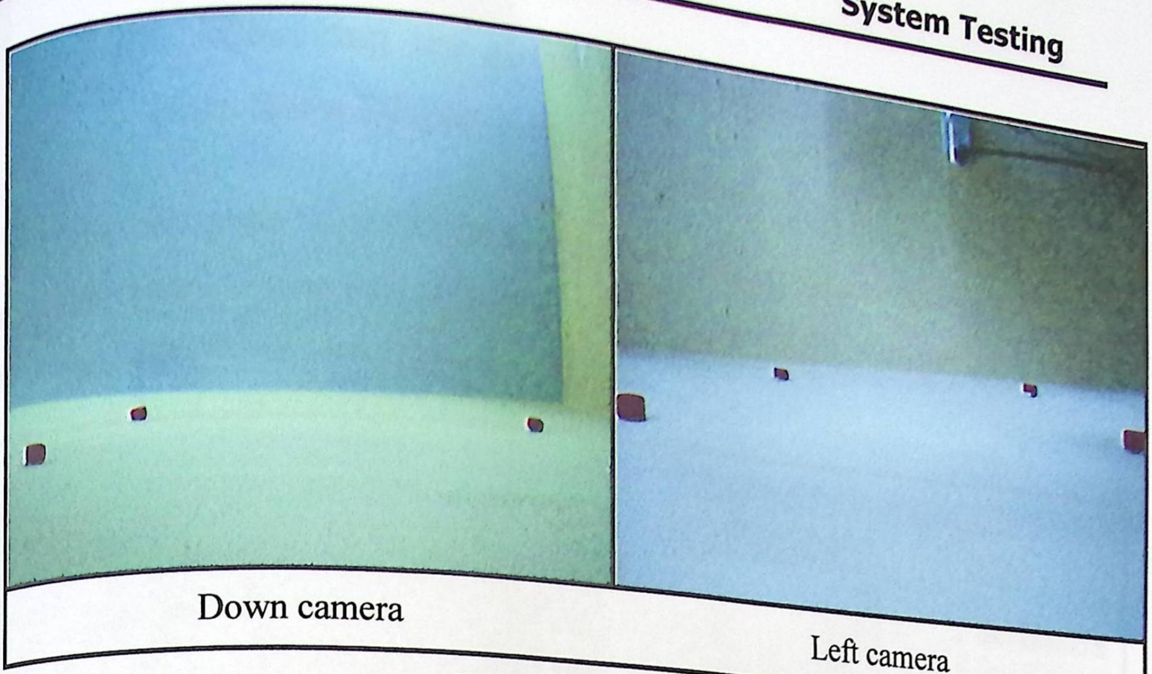


Figure 7.1: Correct capture

Figure 7.2 illustrates a problem in connecting the camera which led to its inability to capture an image.

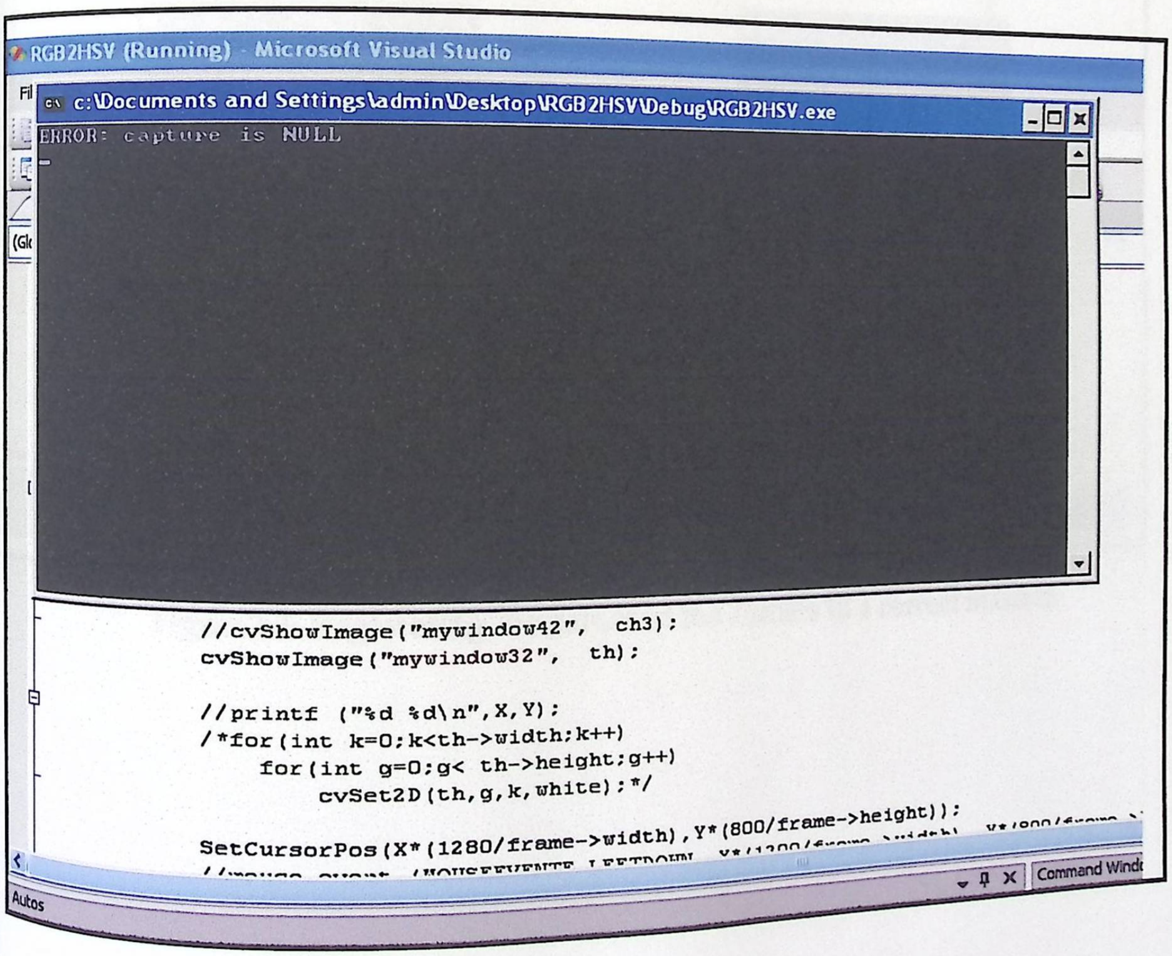


Figure7.2: Error in capturing

7.2.2. Checking the ability of capturing the corners of the Data Show Display Area

The following graph illustrates the operation of capturing the corners of the display area. The operation went smoothly as the system registered the values of the corners and gave the results of such values on the screen.

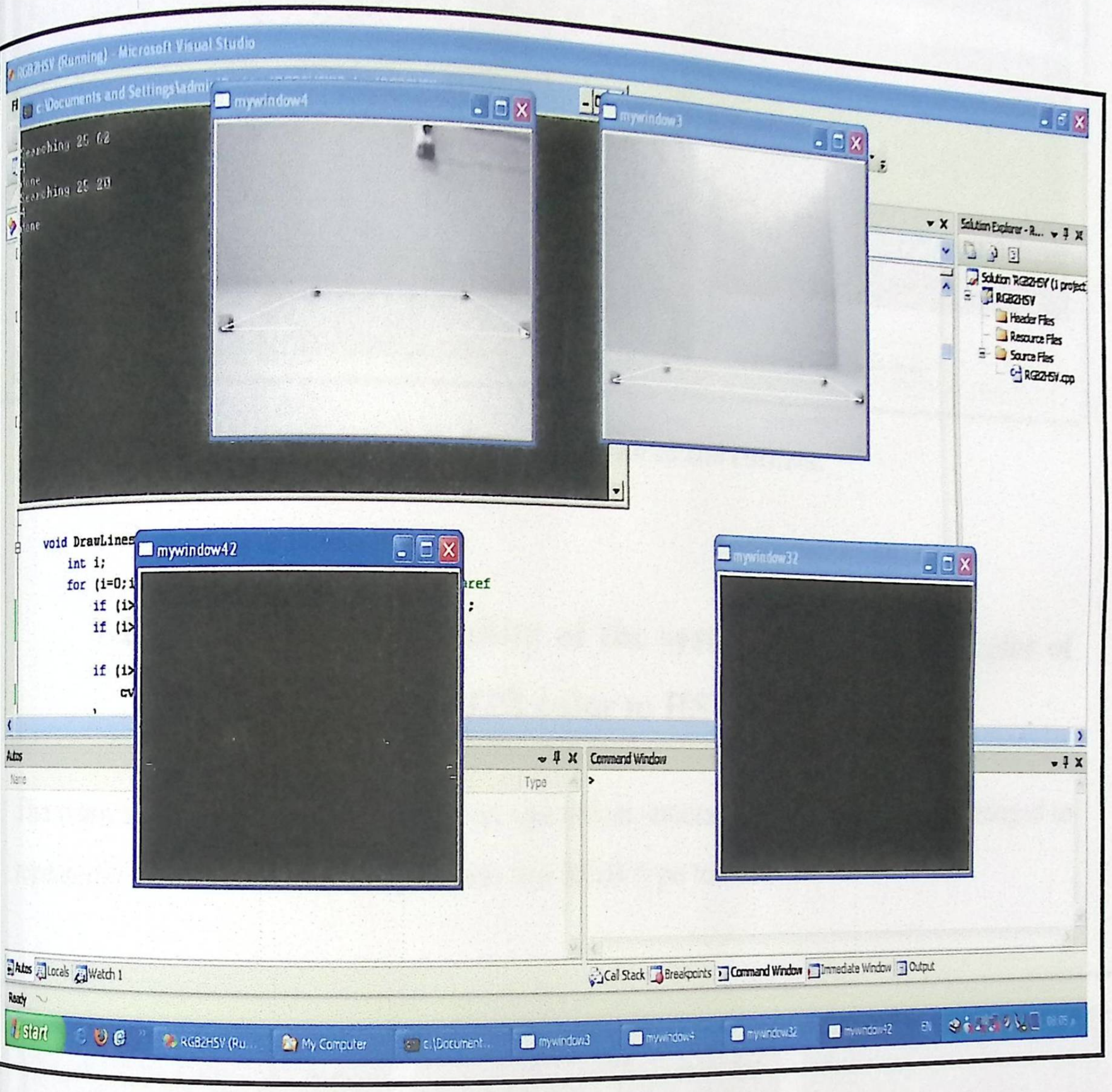


Figure7.3: Registering the values of the corners in a correct manner.

Figure 7.4 illustrates a mishap that has taken place due to the fact that the corners of the display area were "unseen" by the cameras.

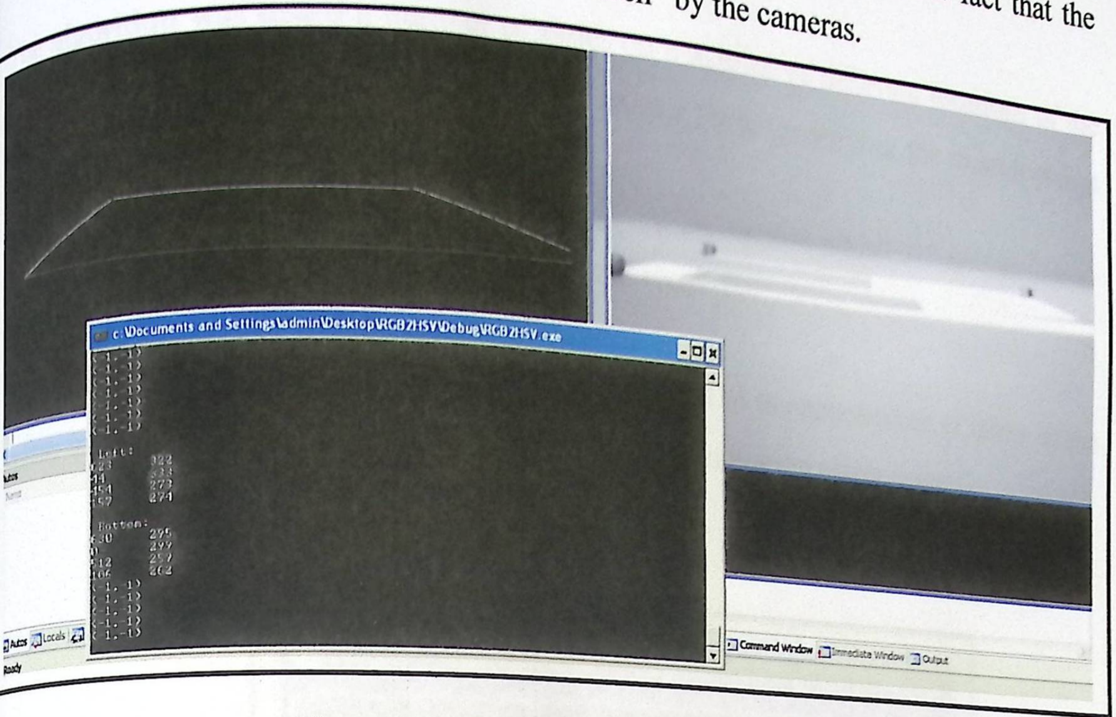


Figure7.4: Incorrect take of the corners.

7.2.3. Testing the ability of the system to change the color of the image from RGB color to HSV color

The team tested the afore-mentioned operation successfully as the system managed to repeatedly to convert the image from the RGB type to HSV

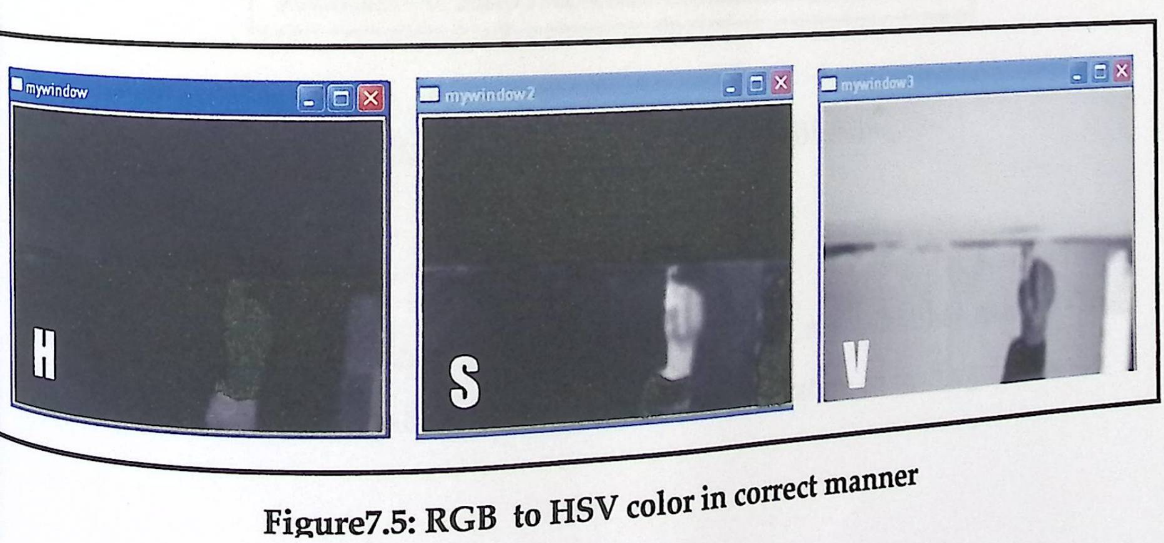


Figure7.5: RGB to HSV color in correct manner

7.2.4. Testing of Thresholding algorithm

The algorithm was tested a number of times on a number of different locations of the finger. In all of the tests, the algorithm was successful in identifying the exact location of the finger.

7.2.5. Testing of Blurring algorithm

The non-presence of the blurring algorithm resulted in the appearance of noise in the taken image which affected negatively the functioning of the entire system.

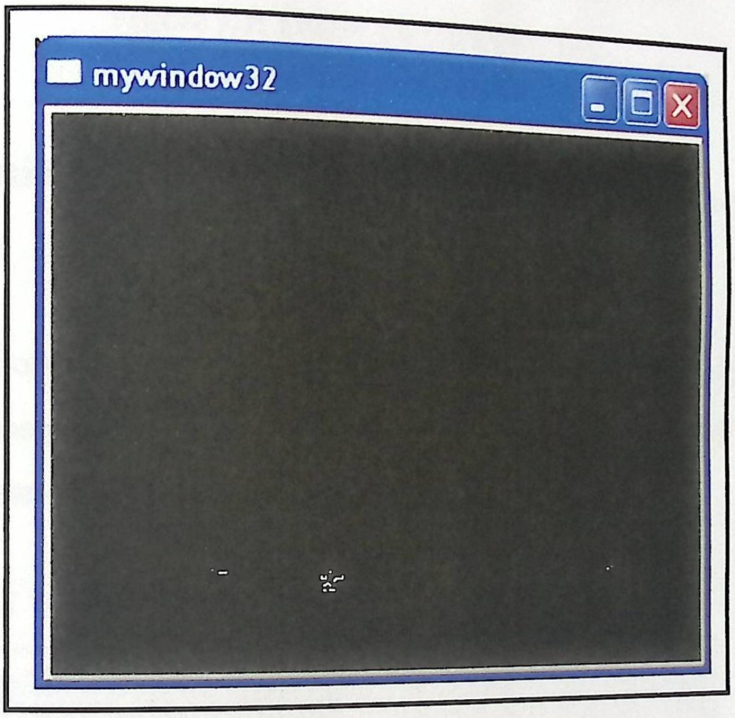


Figure7.6: Threshold without Blurring.

As illustrated in figure 7.7, the application of the blurring algorithm has reduced the noise in the image leading to better identification of the location of the finger.

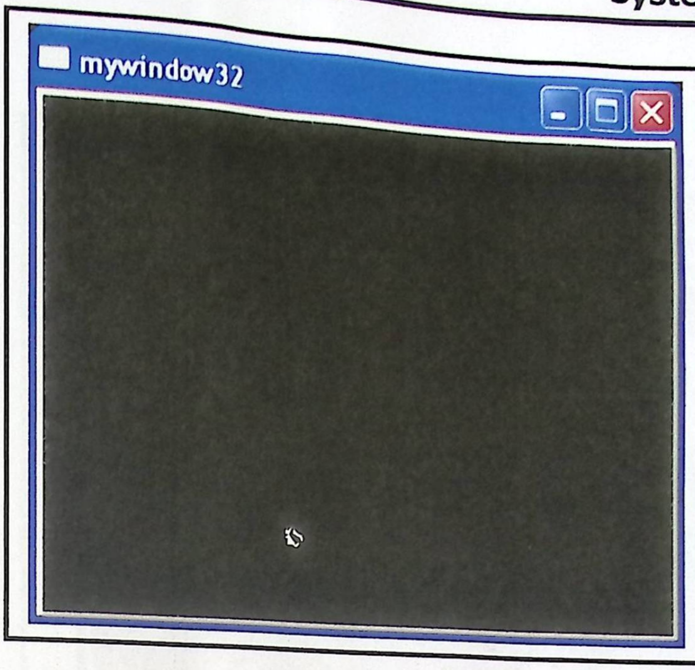


Figure 7.7: Threshold with Blurring.

7.2.6. Testing the color of the finger is within the range of the color

During this operation the team has tested more than one color by placing it on the finger of the proposed user. Some of the vivid colors (such as red and blue) were accepted by the system illustrated in figure 7.8

Here found correct values of color and the result of the threshold

Figure7.8: Take Value of Color.

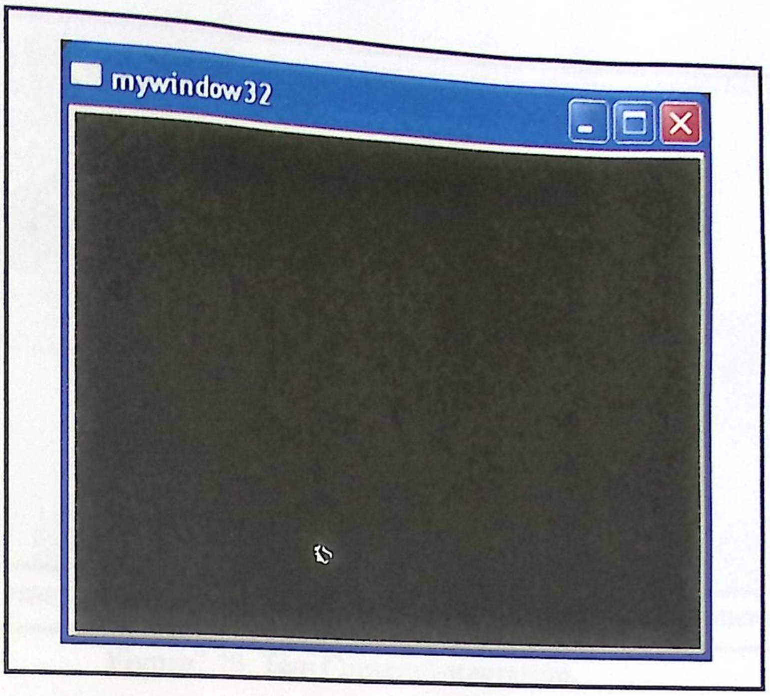


Figure7.9: Correct Color.

7.3. Integration Testing

In this test, the team has integrated more than one segment of the system with each other as to make sure these parts are working collectively.

7.3.1. Testing the synchronization of the work of both cameras

While testing the functionality of each camera separately during the Unit Testing phase, each one proved to work in a successful manner. In order for the entire system to work, the team needed to integrate the function of both cameras as to check their compatibility and their ability to work in synchronization. The testing of the integrated function of both cameras the result they work together in harmony under a number of variable circumstances.

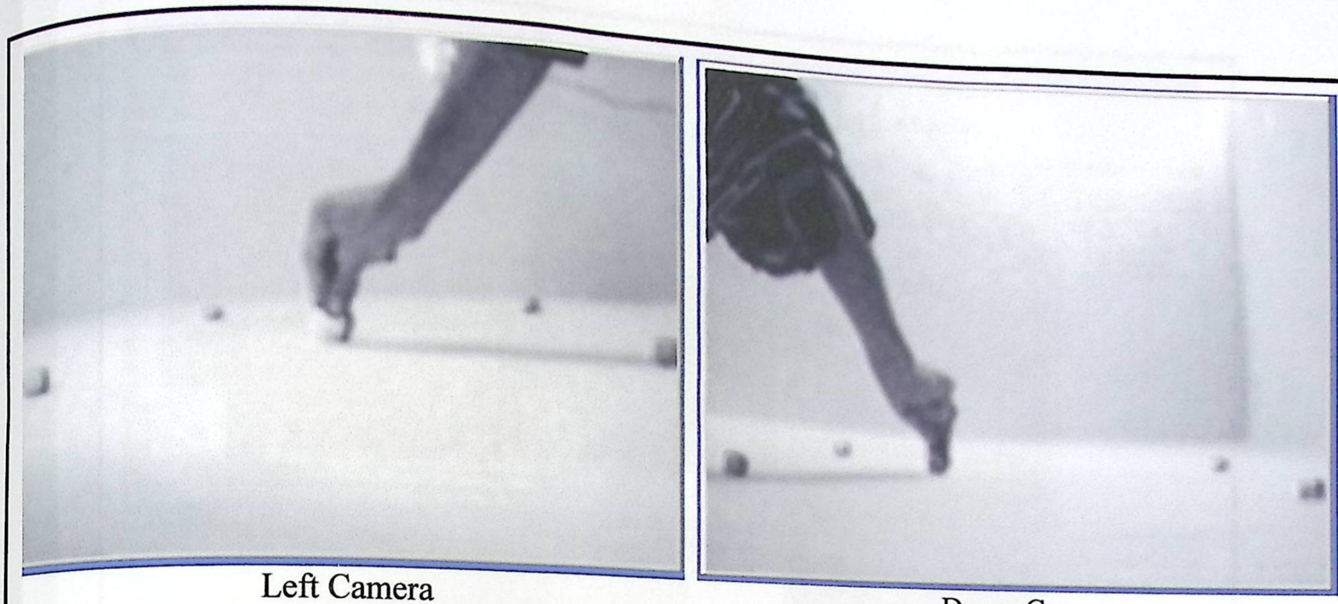


Figure7.10: Two Camera Integration.

7.4. System Testing

In this type of testing the team examined the consistency and cohesiveness of all parts of the system combined. The goal was to make sure that the different parts of the system function correctly in relation to each other.

One main category that was tested in this type was the impact of changing illumination on the system's ability to function properly. The testing was carried out on three stages:

7.4.1. Testing the system with the degree of illumination was not present

In this case the system did not respond to the input as indicated by the following figure

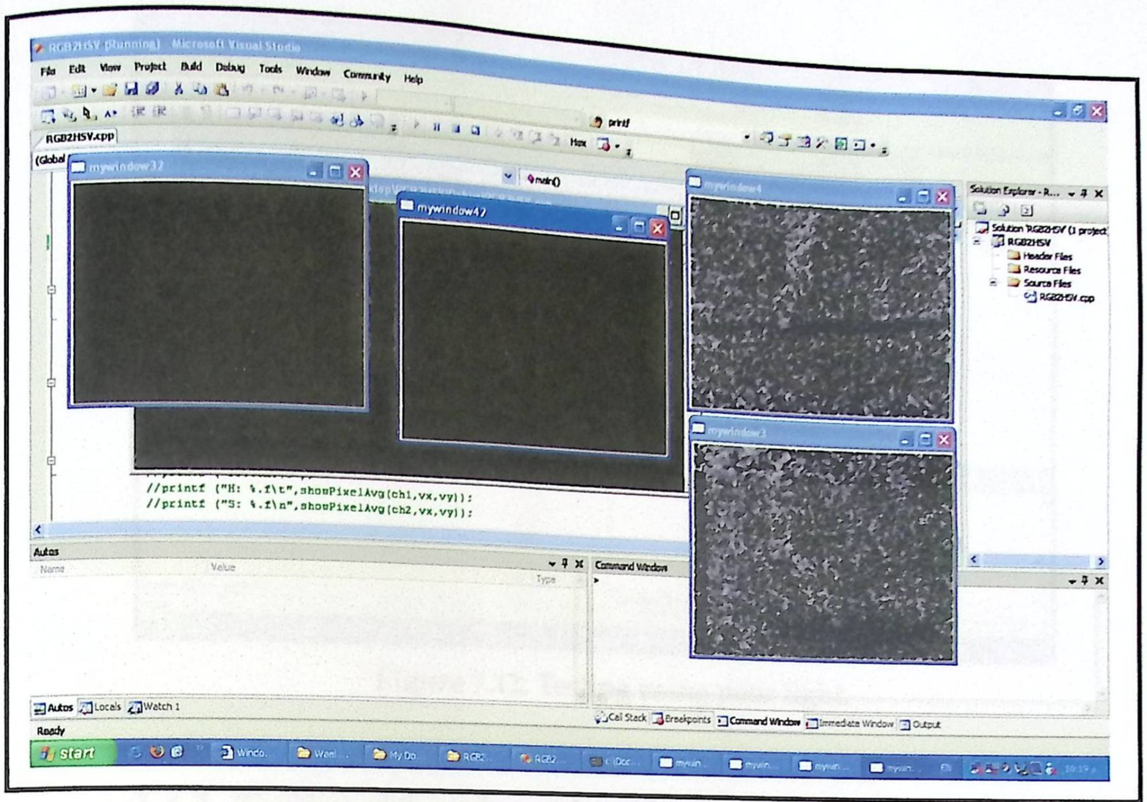


Figure 7.11:Testing without Light.

7.4.2. Testing the system when illumination is weak

The testing took place through turn off one of the lights present at the working environment while keeping one on. In this case, the system functioned smoothly with positive results. However, when we turned both sources of light off and used only the illumination stemming from the laptop, the system did not respond as reflected in the following figure.

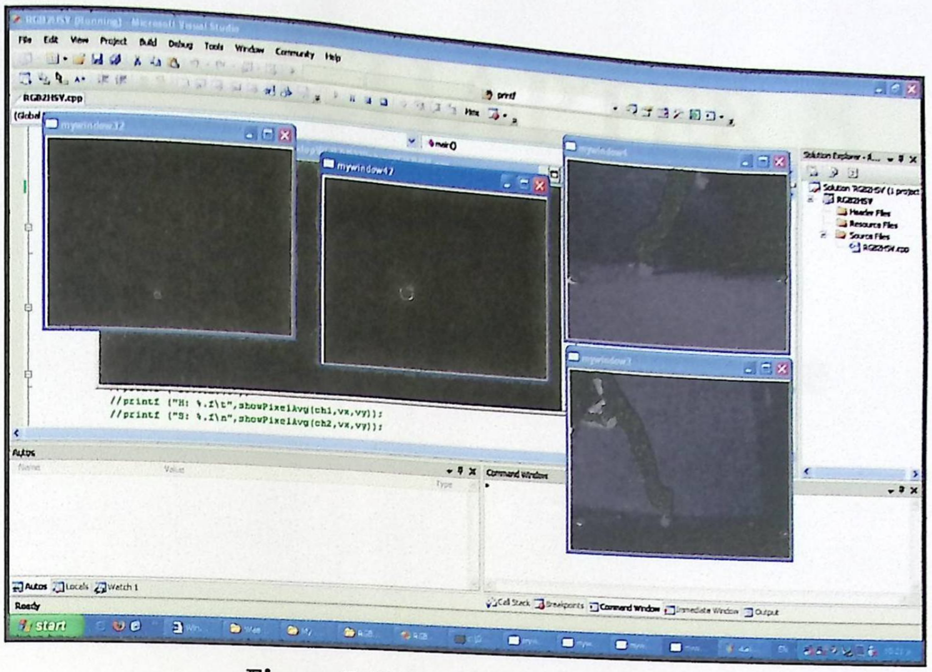


Figure 7.12: Testing using poor light.

7.4.3. Testing the system when the illumination is "normal"

The team tested the system during the day light and in a room with ordinary lighting. The results in that type of environment were excellent as the system functioned extremely well in terms of both Real Time and accuracy as reflected in the following figure:

Figure 7.13: Testing using normal light.

CHAPTER 8

System Maintenance

- Introduction
- Installation
- Maintenance Plan

8.1. Introduction

Maintenance of the system represents the final stage of the project life cycle. It is the process of returning to the beginning of SDLC and repeating development steps focusing on system change until the change is implemented. Also, this chapter focuses on the implementation of the system in the real environment.

8.2. Installation

As to assure that the new application\program runs in a satisfactory manner with the needed efficiency, a suitable system environment must be provided. The minimum operational requirements must be attended to as to assure that the program functions effectively and it does what it is supposed and programmed to do. In order to reach the desired outcome the system must go through the following three stages:

1. **Establishment of the production environment:**
   The system must have the minimal requirement to run the application as described in chapter two.

2. **Deployment of the new system:**
   The system must describe the steps that should be followed to operate the system. Afterwards, the company or the university should be asked whether or not the new system covers all the functions desired by it.

3. **Running the system :**
   At the end of the second step and after the establishment of the production environment, the company\university can run the system in its own environment.

8.3. Maintenance plan

When the system is being operated in the "real environment", there would be a possibility of failure or a series of errors to take place. That happens either due to a faulty system or to the fact that a new user might not be well trained to run the program smoothly. As such, an alternative plan must be presented to main the system and to prevent, as much as possible, the occurrence of any mistakes.

8.3.1. Camera Maintenance

The company\university that intends to use the application must have a number of spare cameras to be used in case of the cameras malfunctions. Moreover, a frequent maintenance of the cameras must be undertaken to make sure that the lenses of the cameras are problem-free and that no dust accumulates on them.

CHAPTER 9

Conclusions and Recommendations

- Introduction
- Conclusions
- Recommendations
- Challenges

9.1. Introduction

This chapter reviews the results of the project as a whole, including the conclusion that we came out with, the results of our project, recommendation and the problems that we faced.

9.2. Conclusion

In the aftermath of carrying out all of the needed steps as required by project, the results were as follows:

- We built a Wide Vision-Based Touch screen that is interactive and sensitive to a finger touch.
- We managed to carryout all of the "events" such a, *click* and *double click, drag and drop;* all such events are normally carried out by a regular computer mouse.
- The team used all of the "events" on other computer applications such as *Paint* program. The results were satisfactory.
- The system was tested in a regular light and illumination setting without any negative effects as it worked as expected.
- The system began automatically acquiring the data show display area.

9.3. Recommendation

- To improve the system to use one camera.
- Fit the cameras on the show display associated with DataShow as to enable an easier future usage of the system. The physical frame can then be removed for easier movement which, in turn, makes the display area much more dynamic (vs. static as in the case of the physical frame).
- An attempt can be made to add an option at the start of the functioning of the system without the need to use a glove on the finger.

9.4. Challenges

The following are the main challenges and problems that we faced while the project building the system:

- The cameras used in the experiment had a small resolution. We are sure that if we had had cameras with higher resolution, the results of the project would have been improved.
- The cameras used in implementing the project had a narrow view which made the size of the display frame relatively large.
- We faced a problem in indicating the color of the glove placed on the finger. We found out later that the problem lay in the cameras. Using other distinctive colors, however, solved the problem.
- We faced a problem in providing the ideal tools and equipment used in the project.
- We face a problem in providing data show to implement and maintenance the project.
- Changing the position of the camera for any reasons that might induce problems in the project. Such problems are sure to take place as the dimensions of the frame we designed for the project are fixed.

REFERENCES

1. WordNet,http://wordnetweb.princeton.edu/perl/webwn?s=touch%20screen
2. WiKIAnswer, http://wiki.answers.com/Q/Advantages_of_touch_screen.
3. WIKIPEDIA, http://en.wikipedia.org/wiki/Touchscreen
4. PLANER,http://www.planarembedded.com/technology/touch
5. Technology comparison ,http://www.touchscreen-me.com/dir/about-touch-screens/technologies-comparison-resistive.html
6. Amazon,Amazon. http://www.amazon.com
7. WIKIPEDIA ,http://en.wikipedia.org/wiki/Infrared
8. Rationale for American National Standard for Information Systems – Programming Language
9. WiKIAnswer,http://wiki.answers.com/Q/What_is_Disadvantage_of_c_programming
10. Vision-based Interaction with Fingers and Papers. Zhengyou Zhang, 2003, *International Symposium on the CREST Digital Archiving Project* pages 83–106.
11. WIKIPEDIA, http://en.wikipedia.org/wiki/Adaptive_thresholding
12. PC-Connection,http://pc.pcconnection.com/1/1/2249-pc-connection-cables-cat5e-350mhz-ethernet-patch-cable-blue-snagless-10ft-351010.html
13. TechFuels,http://www.techfuels.com/applications/7516-unibrain-fire-i-digital-camera-web-cam.html
14. WiKIAnswer,http://wiki.answers.com/Q/What_are_the_disadvantages_of_infrared
15. Ezine articles,http://ezinearticles.com/?What-Are-The-Advantages-Of-Infrared-Sauna-Kits?&id=905012
16. WIKPEDIA,http://en.wikipedia.org/wiki/RGB

17. Color Presentation of Astronomical Images,
http://www.allthesky.com/articles/imagecolor.html
18. Tech FAQ,http://www.tech-faq.com/hsv.shtml.
19. http://ilab.usc.edu/wiki/index.php/HSV_And_H2SV_Color_Space
20. OpenCV Library Docunintation.