Android Based Robot

By

Muhammad Ashraf
Hamza Khalid
Usman Ahmed

Submitted to the Faculty of Computer Software Engineering, National University of Sciences and Technology, Islamabad in partial fulfillment for the requirements of a B.E Degree in Computer Software Engineering

JULY 2011

Android Based Robot

This project proves the concept that android operating system can be used as a platform in the robotic field to control different functionalities of all types of robot. The android application developed in this project is used to control the movement of the robot (T-copter) and it also provides virtual presence to the end user who is controlling the robot using an interface.

Software requirement specifications, software detail design and architecture have been done in initial phase. In implementation phase, three parts of software has been completed. An interface, used to control the movement of the robot (T-copter) and also display the live video stream coming from the robot (T-copter). An android application which is running on an android device housed on the robot (T-copter), this device is connected to the robot hardware via Bluetooth device and is also connected to the user interface via the network (wifi). Android application consists of two modules, first one is live video streaming module and the second one is used to receive the movement control signals. Arduino has its own software program burn into it which receives the control signal from Bluetooth device and sends the movement signals to the actuators of the robot. All the three parts of the software, including android application, user interface and Arduino's program has been achieved. Unit testing, system testing and integration testing of the system validated the required results. All the

developed functionalities are working efficiently and the results conform to the software requirement specification document.

CERTIFICATE OF CORRECTNESS AND APPROVAL

Certified that work contained in this thesis “Android Based Robot” carried out by Muhammad Ashraf, Hamza Khalid and Usman Ahmed under supervision of Maj. Dr. Abdul Ghafoor for partial fulfillment of Degree of Bachelor of Software Engineering is correct and approved.

Approved By

____

(Supervisor Name)

_____Department

MCS

Dated:_______________________

Dedication

We humbly thank Allah Almighty, the Merciful and the Beneficent, who gave us health, thoughts and co-operative people to enable us to achieve this goal

Acknowledgments

Maj Doc. Abdul Ghafoor has been ideal supervisor for this project who helped us during the entire project. His support and encouragement was greatly needed and deeply appreciated.

Preface

Android based robot (T-copter) proves the concepts that android platform can be used in the Robotic field for developing different types of application to control the different functionalities of robots. The android application developed in this project is used to control the movement of the robot (T-copter) and it also provides the virtual presence to the place where the robot (T-copter) is moving. The Android smart phone used in this project also reduces cost to build a robot because android smart phone comes with built in sensors so that's why there is no need to buy multiple sensor's hardware.

This project provided us chance to get knowledge of android platform by using android SDK for developing the android application which is used in this project. The live video streaming module which is developed in the android application provided us chance to learn about networking concepts because network programming is used for developing the live video streaming module of the android application.

The project work has been divided among all the three members of the project, Hamza khalid and Muhammad ashraf has worked on the development of the android application and Usaman ahmed has worked on the hardware interfacing.

Table of Contents

Chapter 1 Introduction to Android based Robot (T-copter)

1.1 Introduction……………………………………………………..1
1.2 Background……………………………………………….....1
1.3 Statement of problem……………………………………........2
1.4 Statement of goals & scope of the project…………………...2
1.5 Document organization……………………………………….2
1.6 Summary……………………………………………………3

Chapter 2 Related work

2.1 Introduction……………………………………………………..4
1.7 RC quad-copter………………………………………..……4
1.7.1 Project Overview……………………….………………...5
1.7.2 Software overview……………………….………………...6
1.8 Autonomous T-copter……………………………….......7
1.8.1 Project overview…………………………….................8
1.8.2 Software overview……………………. ..………………….8
1.9 Summary……………………………………………………..9

Chapter 3 Software Requirement Specification

3.1 Introduction………………………………….………10
3.2 Purpose……………………………………….. …...........10
3.3 Overview of document.............................. … …...............10
3.4 Project Scope……………………………. . …… ..………...11
3.5 Overall Description……………………. …………...……11
3.5.1 Product Perspective…………. . …. …..…………………......11
3.5.2 Product Features……………………………………......12
3.5.3 User Classes…………………………………….……...13
3.5.4 User Characteristics………………………………………13

3.5.5 Operating Environment……………………….…………..13
3.5.6 Design and Implementation Constraints……………............14
3.6 System Features…………………………………………..…14
3.6.1 System Feature 1: Video Streaming……………………..15
3.6.2 System Feature 2: robot (T-copter) control………………..…17
3.6.3 Use case diagram………………………………..………18
3.7 External Interface Requirements…………………………..19
3.7.1 User Interfaces……………………………………….......19
3.7.2 Hardware Interfaces………………………………………19
3.7.3 Software Interfaces……………………………………....20
3.7.4 Communications Interfaces…………………………..……20
3.8 Nonfunctional Requirements……………………….………21
3.8.1 Performance Requirements……………………….............21
3.8.2 Safety Requirements………………………………..…...22
3.8.3 Security Requirements………………… . ……..……..……22
3.8.4 Software Quality Attributes……… . . …………….……..…..23
3.9 Summary…………………………… . … ………….…. 25

Chapter 4 Software Design and Architecture

4.1 Introduction………………………………………………26
4.2 Architecture Diagram……………………………………....26
4.3 Use Case Diagram………………………………………...29
4.4 Class Diagram…………………………………………….30
4.5 Sequence Diagrams………………………………………32
4.5.1 Start video stream……………………………………..32
4.5.2 Finish video stream…………………………………........33
4.5.3 Move up……………………………………………….34
4.5.4 Move down……………………………………………...35
4.5.5 Move straight……………………………………… …... ..36
4.5.6 Move backward……………………………………………..37
4.5.7 Turn right…………………………………………….……38

4.5.8 Turn left……………………………………………………........39
4.5.9 System Sequence Diagram……………………….... …….40
4.9.10 Deployment diagram…………………………………...……..41
4.10 Summary…………………………………………………......42

Chapter 5 Implementation

5.1 Introduction…………………………………………...……43
5.2 Phases of software implementation……………………...…...44
5.2.1 User Interface………………………………… ………44
5.2.2 Android application with initial
Communication implementation……………………………......…45
5.2.3 Live video streaming implementation
in Android application………………………………… …..........46
5.2.4 Implementation of the movement control functionality of the Robot (T-copter)…………………………………………………..............47
5.3 Hardware Implementation…………………………. …….49
5.4 Summary………………………………………………….…..51

Chapter 6 Testing

6.1 Introduction………………………………………………..….52
6.2 Unit testing………………………………………………….52
6.3 Integration testing……………………………………… . ..53
6.4 System testing……………………………………………..54
6.5 Summary……………………………………………………..54
Chapter 7 Results and analysis

7.1 Introduction……………………………………………….....55
7.2 software results…………………………………………….55
7.2.1 Live video streaming module's results………… ..………….55
7.2.2 Robot's movement control module's results…… ..………..56
7.3 Hardware results…………………………………………...56
7.4 Analysis……………………………………………………..56

7.4 Summary……………………………………………… ..…….57
Chapter 8 Conclusion and Future Work
8.1 Introduction……………………………………………… …...58
8.2 Conclusion…………………………………… ……………58
8.3 Future work…………………………………… …………..58
8.4 Summary…………………………………………………59
9. Appendix……………………………………………………. 60
10. Bibliography…………………………………………………67

LIST OF FIGURES

Figure #	Figure Name	Page Number
2.1	Rc-quadcopter	5
2.2	Rc-quadcopter data flow	6
4.1	Architecture diagram	27
4.2	Architecture diagram	28
4.3	Architecture diagram	29
4.4	use case diagra	30

	m	
4.5	Class diagram	31
4.6	start video stream diagram	32
4.7	finish video stream diagram	33
4.8	move up	34
4.9	move down	35
4.10	move straight	36
4.11	move backward	37
4.12	turn right	38
4.13	turn left	39
4.14	system sequence	40

	diagram	
4.15	deployement diagram	41
5.1	android control system	43
5.2	hardware implementation	50

Chapter 01

Introduction to Android based Robot (T-copter)

1.1 Introduction

The main purpose of this project is to prove that the Android operating system can be used as a platform for the robotic field. This project proves that the Android operating system can support different types of robotic applications efficiently. The software developed in this project controls the robot (T-copter) through android operating system by using an android application which is running on the Google phone. Live video streaming module of the android application is also developed in the project which is being used to see the live video stream so that the movement of the robot (T-copter) can be monitor from other end.

1.2 Background

Android is a powerful Operating System supporting a large number of applications in Smart Phones. Android is a software bunch consists not only operating system but also middleware and key applications. In smart phones Android operating system comes with many sensors which provide multiple functionalities for different type of applications. Due to availability of these sensors in just one smart phone these Android based smart phones can be used in the robotic field because in this case there is no need to buy all the sensors individually and then integrate them for making of some specific robot (T-copter). It also reduces the cost of the hardware as we can achieve functionality of different sensors in just one product. Android operating system has been chosen for making of our robot (T-copter) because it also provides the high processing speed which is needed to complete the required tasks in a specific time frame. Other smart phones can also be used for in the ROBOTIC field but Android SDK is much more powerful as compared to other operating systems SDK. Android smart phones can also be used to connect to lan using on board wifi. These all features available in the android operating system make this operating system the ideal candidate for the robotic field.

1.3 Statement of problem

Nowadays there are many robots(flying and ground vehicles) which are used to achieve the virtual presence to a remote place but these are of highly cost but this project has achieved the virtual presence to remote place using Android platform which reduces the cost of robot by using Android smart phone.

1.4 Statement of goals & scope of the project

This project will enable to prove that the Android operating system can be used as a platform to the robot's functionality in the robotic field. The software applications (Android applications) developed in this project use the android operating system to control the different functionalities (movement of robot (T-copter) and live video streaming) of the robot.

1.5 Document organization

The document is divided into different sections including project related work, project's software requirement specification, software design and architecture, implementation and testing. Documentation is written in the sequence as the work of the project has been done.

1.6 summary

The android smart phones can be used in the robotic field as android operating system is the platforms which can supports different type of robotic functionalities. Android provides its SDK which much more powerful as compared to other platform’s SDK. This project proves that the android operating is the best platform which can be used to control different functionalities of the robot (T-copter) by using its powerful processing speed.

Chapter 02

Related work

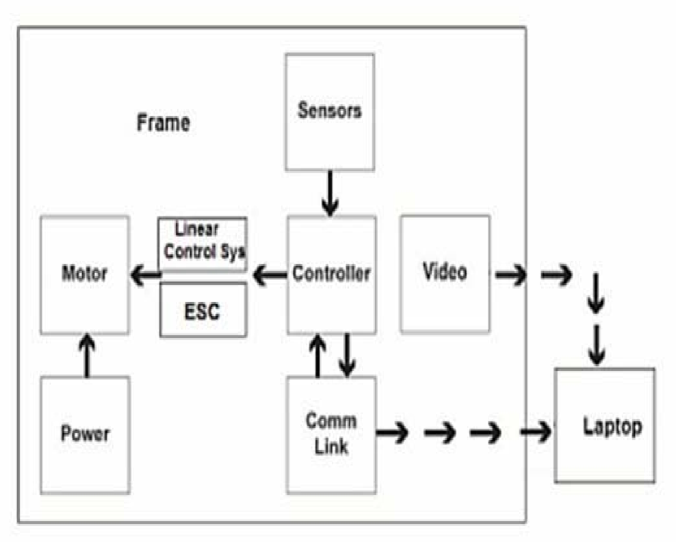
2.1 Introduction

This chapter is about related work to our project which has been done previously. Descriptions of two projects have been mentioned in this chapter. The first project is “controlling a Quad-copter by radio transmission or operate under the guidance of limited autonomous protocols” which was done by the School of Electrical Engineering and Computer Science, University of Central Florida, Orlando, Florida. The second project described in this chapter is “Autonomous quad-copter”.

2.2 RC Quad-Copter

The quad-copter in this project can be controlled by radio transmission and also by using some autonomous protocols. In this project the stability of the quad-copter has been achieved using five degree of freedom (Dof) inertial measurement unit (IMU).

Different type of sensor data has been integrated using proportional-integral-derivation controller (PID). Onboard micro controller will maintain the feedback loop. The design of the quad-copter is set so that the weight of the quad-copter can be minimized.

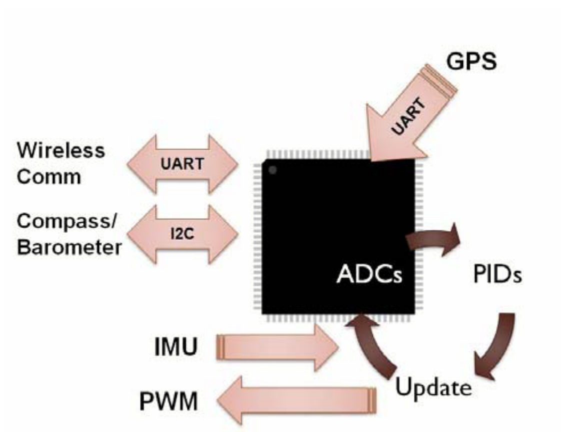


2.1 Rc-quad copter

2.2.1 Project Overview

The quad-copter in this project has robust sensor suit so that this quad-copter can also be controlled in the autonomous mode. In autonomous mode, the quad-copter includes GPS module so that when this quad-copter is given so specific GPS location it can make its flight to the target location using the co-ordinates given by the GPS without any further human control. To make this quad-copter autonomous there is need of additional subsystems including the ultrasonic proximity sensors. Ultrasonic proximity sensors are needed so that the quad-copter may detect and avoid the obstacles.

A digital compass is also used in this quad-copter so that the direction could be detected and corrected as needed during the quad-copter flight. All the mentioned sensors will then send the data to the MCU. MCU is the main processing unit of the quad-copter which must process the incoming information by using its algorithms and then prompt the needed sensors to start working when some control command given by the MCU to the other sensors.



2.2 Rc-quad copter data flow

The major and complex task required by the MCU is to maintain the level flight of the quad-copter by varying the speed of the speed controllers which are connected to the motors of the quad-copter. MCU have to do some calculations which are needed to maintain the level flight of the quad-copter. MCU maintains the level flight by getting data from the IMU(Inertia Measurement Unit) and perform the calculations on this data. The IMU get the data from the triple-axis accelerometer and dual-axis gyroscope and combines this data using sensor fusion algorithm. All the subsystem of the quad-copter is interdependent, centrally linked by the MCU frame and the power system.

2.2.2 Software overview

The software part includes the polling of the sensors, action perception loop and controlling movement. The code is written in the C programming language, Using AVR studio IDE. The action perception loop is the main loop in the software. This loop will act on the sensor perceived data and continuously run. The software of this quad-copter needs to keep track of the time which it takes to run the action perception loop for calculation s of the sensor data. In order to keep the flight, 50 HZ must be maintained. Polling of the sensors is done by using the internal ADC of the MCU. The ADC has the resolution of 10 bits.

The software for movement control includes a sensor fusion algorithm and a PID loop. The sensor algorithm takes the sensor reading from the accelerometer and the gyroscope and after getting these reading it combines these values into better estimated angle. This algorithm takes the force projection vector of the accelerometer and then relates it to the gyroscope's instantaneous velocity. It does this by projecting the X and Y component of the force vector into the Z plan. The gyroscope uses its algorithm for measuring the speed of the angle between the projection and the Z axis. This algorithm produces better estimate of the force vector which can be used to calculate the angle position of the quad-copter. PID loop runs the estimated angle and it produces the adjustment which is needed by the motors to correct the position of the quad-copter.

2.3 Autonomous T-copter

The purpose of this project is to create an aerial T-copter which can autonomously patrol the border and wirelessly inform about the suspicious activities by using the sensors mounted on the body of the T-copter. This project was done in intension to drastically reduce the human endangerment during the protection of border and other security purposes. This T-copter can be used to achieve multiple functionalities depending on the location where the T-copter is being deployed; the user may choose range of sensors mounted on the body of the T-copter to perform some specific task.

The design goal of the project is to build a platform which is capable of providing the smooth flight to survey the UAV's surrounding areas. T-copter system uses a gyroscope and an accelerometer to provide the stability during the flight. The T-copter also has the ability to carry the payload including the IR sensors and night vision camera.

2.3.1 Project overview

The prototype in this project is the aerial vehicle which can fly independently of the operator by receiving the information from the user end to perform its tasks.

Four brushless motors are being used in the design of the T-copter; these motors control the direction, elevation and the tilt by varying the input voltage to that motors. Multiple-axis accelerometer and gyroscope is used to allow for multiple degrees of freedom while reading the information regarding the status of the T-copter. The use of these sensors maintains the stability in constant changing conditions.

2.3.2 Software overview

To control the movement of the T-copter, software uses the complementary filtering so that the best source of angular measurement could be chosen. The accelerometer provides the accurate short term measurement whereas the gyroscope provides the better reading of angle for a long time period. Accelerometer and gyroscope are sampled and scale the output by looking at the speed and the movement of the T-copter. Complementary filtering was chose instead of Karman filtering because complementary filtering is easy to implement and less resource consuming sub routine. PID controller is used in the software to make the microcontroller inputs more accurate. PID is used to reject the errors from the IMU.

2.4 Summary

This chapter describes about the work related to our project which was previously done in this field. There is description of two projects in this chapter which are related to our work but they are both using different technologies as compared to our project. The mentioned projects have used the RC controlled technology in their projects and they have used autonomous mechanisms to control the T-copter without the effort of human.

Chapter 03

Software Requirement Specification

3.1 Introduction

This chapter contains the description of all requirements which are required to develop the software of Android based Robot (T-copter). It contains all the functional requirements which should be there in the end product. Description of design and implementation constraints, operating environment and the entire non functional are also included in this chapter.

3.2 Purpose

The main purpose of this project is to prove that the Android operating system can be used as a platform for the Robotic field. All type of robotic applications can be supported by the Android operating system efficiently. In this project we are controlling the robot (T-copter) through android operating system by using an android application. One other android live video streaming application is also used to see the live video stream so that the movement of the robot (T-copter) can be monitor from other end.

3.3 Overview of document

The Overall Description section, of this document gives an overview of the functionality of the product. It describes the informal requirements and is used to establish a context for the technical requirements specification.

The System feature section of this document describes about all the main system features including the functional requirements which are involved in that feature.

The external interface requirement section describes about different user, hardware, software and communication interfaces.

The non functional requirements section contains all the non functional requirement detail.

3.4 Project Scope

In this project a hardware subsystem will control the electromechanical system. A software layer will interact with hardware subsystem by delivering the control signal from control end handheld device (Google phone) in real time. A control end subsystem with a generic user interface will enable the user to control the movement of the robot (T-copter) and view live video stream coming from the robot (T-copter) end. User can interact with this web interface by sending control signals back to the hardware subsystem over the wireless LAN. Live streamed images being streamed to the web interface to give the user a virtual presence.

3.5 Overall Description

3.5.1 Product Perspective

This is a new self-contained product. This project will provide the end product as a robot (T-copter) which can be controlled with the help Google phone which uses the Android operating system to control the movement of the robot (T-copter). Two main applications will be developed in the software part, one is the android application to control the movement of robot (T-copter) and the other one is the live streaming android application which provides the movement view of the robot (T-copter) to the user end. The main fields covered by this project are robotics, mobile computing and live video streaming.

3.5.2 Product Features

There are three main features of the software which will enable the user to efficiently control the movement of the robot (T-copter). The first one is the android application which will be used to get the movement signals from the user end and then after some manipulation send the movement command to the hardware part so that the robot (T-copter) can be moved in the desired direction.

For monitoring the movement of robot (T-copter) android live video streaming application will be developed so that the user could easily monitor the movement of the robot (T-copter) and then by viewing the live video streaming coming from the Google phone the user can easily move the robot (T-copter) in the desired direction. This android application will also be running on the Android operating system which is installed on the Google phone.

A user interface will be developed so that user can view the live video stream of the robot (T-copter) movement and can control the robot (T-copter) movement by using the robot's movement controls on the interface. There will be six controls button available on the user interface as up, down, forward, backward, right and left. There will also a view panel which will show the live video stream coming from the robot's body.

3.5.3 User Classes

3.5.3.1 Robotic field

This project proves the concept that the Android operating system can be used to control the robot's functionality. The android applications developed in this project can be used in Robotic fields to control different functionality of the robots.

3.5.3.2 Android users

Android users can use the live video streaming module as an app for themselves. This app will help them in monitoring their places from a remote place which are in the wifi signal range.

3.5.4 User Characteristics

User will be available with an interface through which the robot's movement can easily be control and the live video streaming can also be seen from that interface. User can easily interact with the interface by clicking on the control buttons for controlling the movement of the robot (T-copter). A view panel will be available which will display the live video streaming coming from the robot (T-copter) end.

3.5.5 Operating Environment

Android Software applications will be operated on the android operating system. We will be using android operating system version 1.6 for the development and testing of the software. Same version of the operating system will also be used for operating the software. An Arduino chip will be used which has the built in microcontroller on it; this microcontroller will be used for sending the control signals to the mechanical parts of the robot (T-copter).

3.5.6 Design and Implementation Constraints

Application Programming Interface (APIs) of the Google android can cause problems like camera crash, unable to send the data signal to the android mounted on the robot (T-copter). This problem can be solved by using the 1.5 API for the android, as the latest API of the android is a little buggy.

Video Stream which is coming in remotely from the robot (T-copter) via internet can create some issues like data loss or video stream lag. This issue can be overcome by using good coding techniques which will allow the flow of the video stream without any lag via internet.

Communication between android (Google phone) and the hardware (mechanical part of the robot (T-copter)) can cause problems, if the code is not properly implemented. So the code has to be implemented in such a way that the android is compatible with mechanical part of the robot (T-copter).

The mechanical failures can also create issues.

3.6 System Features

The main purpose of this project is to develop a robot (T-copter) which can be controlled with the help of android Google phone by using Android operating system. With the help of this project the concept of controlling a robot (T-copter) via Android operating system can be proved. We will also show the usefulness of communication of android operating system with the mechanical hardware and its connectivity.

3.6.1 System Feature 1: Video Streaming

3.6.1.1 Description and Priority

Main purpose of this feature is to make android Google phone in to a live video streaming device. By using this feature the user can view live video stream of the robot (T-copter) from the camera on the Google phone thus controlling and viewing the movements of the robot (T-copter). This live video stream will be implemented over the internet.

As the robot (T-copter) will be used for surveillance, this feature of live video streaming is a very important feature of this project. As it turns your Google phone in to a live video streaming device, we can use this feature as a separate module for different purposes.

As the robot (T-copter) will be controlled through an android Google phone, the cost of the robot (T-copter) decreases as only a mobile device could be used to control different operations required to move a robot (T-copter). Also no expensive cameras are required to for the video streaming feature as Google phone provides its own good quality camera mounted in the phone.

3.6.1.2 Stimulus/Response Sequences

The user will connect to the android Google phone via internet. After the user will be connected with the Google phone; the app running on the android will start streaming live video by using the camera on the Google phone, to the user end. At the user end the live video stream will be displayed on a particular user interface. The end user will close the connection of live video stream by selecting close video stream option on the user interface which will stop the stream coming from the Google phone on the robot (T-copter).

3.6.1.3 Functional Requirements

There will be a user interface on the user end which will provide different control options in the form of interface. There will be different option on the interface for the movement of the robot (T-copter) e.g. move forward, backward, up down, right and left and stop live video stream. This user interface can be accessed by different authorized users.

There will be an android app which will be running on the Google phone mounted on the robot (T-copter). This app will continuously capture the video and stream this live video to the end user so that the end user can view the video coming from the robot's Google phone. Also this app is going to pass signals for the robot (T-copter) movement to the microcontroller on the Arduino chip.

The microcontroller mounted on the Arduino chip will accept the signal from the android app sent by the user. The software burned in to the micro controller will then transfer the signal to the respective mechanical part of the robot (T-copter); in return the robot (T-copter) will start to move.

3.6.2 System Feature 2: robot (T-copter) control

3.6.2.1 Description and Priority

This feature describes the communication of android operating system with the mechanical part of the robot (T-copter). To communicate android with the hardware part of the robot (T-copter), we are going to use a chip named Arduino. Arduino will act as an interface between android and the mechanical part of the robot

(T-copter). As Arduino has a built in micro-controller, so it will decrease the cost of the project by a large factor.

3.6.2.2 Stimulus/Response Sequences

The microcontroller will connect with the android operating system. Android operating system will connect with the internet using a specific IP address. The movement signal sent from the user will pass through the android application. The android app will transfer the signal to the micro-controller mounted on the Arduino chip. The Arduino chip will transfer the signal to the respective mechanical part of the robot (T-copter), thus moving the robot (T-copter).

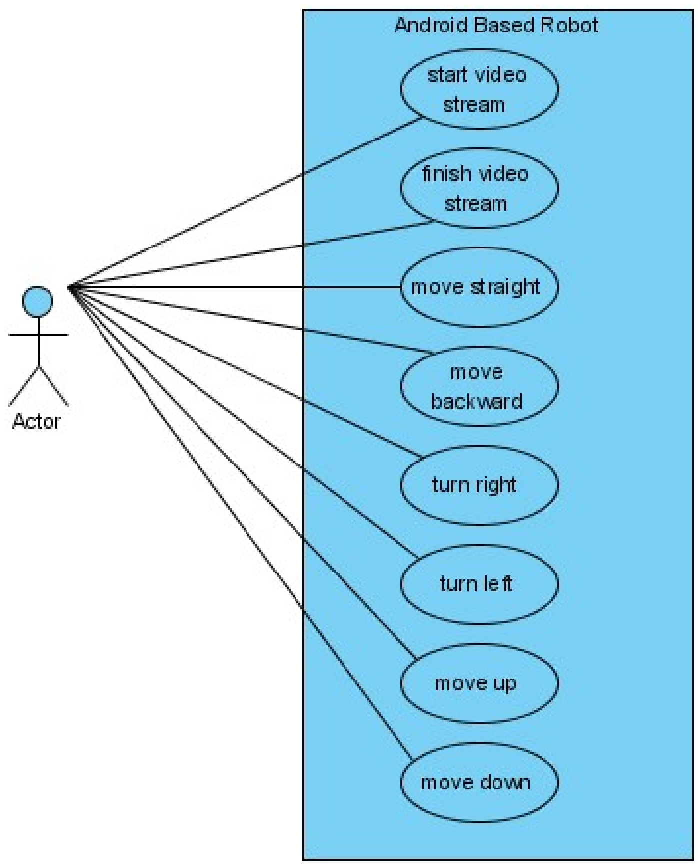
3.6.2.3 Functional Requirements

There will be a user interface on the user end which will provide different control options in the form of interface. There will be different option on the interface for controlling the movement of robot (T-copter) e.g. move straight, move back, turn left and turn right. This user interface (see appendix A) can be accessed by different authorized users by logging with their passwords.

There will be an android app which will be running on the Google phone mounted on the robot (T-copter). This app will be written in python which control the movement of the robot (T-copter) by sending different moves commands to the microcontroller on the Arduino.

3.6.3 Use case diagram

3.1 use case diagram



3.7 External Interface Requirements

3.7.1 User Interfaces

There will be only one user interface having two sections of the interface, one is for viewing the live video stream coming from the robot (T-copter) and other one having control buttons for controlling the robot (T-copter) movements.

3.7.1.1 Live video stream section of interface

In live video stream section of interface a user can view the live video stream coming from the robot (T-copter). There will be two control buttons available in that section, one button is for start getting the live video stream from the Google phone mounted on the robot (T-copter). The other button will be there for stop getting video stream from the Google phone mounted on the robot (T-copter).

3.7.1.2 Robot (T-copter) control menu

There will be a robot (T-copter) controls menu section on the main interface which contains different movement options for moving the robot (T-copter) in different directions. There will be a button for moving straight, button for moving back, button for turning left a button for turning right, button for moving up, button for moving down.

3.7.2 Hardware Interfaces

This will implement the interfacing of android app with the microcontroller mounted in the Arduino chip. Code will be written to transfer the movement signal from the android app to the micro-controller, so the robot (T-copter) can move as indented by the user. To move the mechanical parts of the robot (T-copter) code would have to be burned in to the microcontroller, so that it could accept the signal from the android app. Python and Java code will be used to provide communication protocols between the hardware and the software.

3.7.3 Software Interfaces

The user will first provide its user name and password to gain access to the user interface for controlling the robot (T-copter). Then the user will connect to the android app by providing a specific IP address and port number. For this a verification reply will be sent from the android's app to the user interface that the user is now connected to the robot (T-copter) via internet.

Once the user is connected to the robot (T-copter), the user will prompt the android app for sending the live video stream through the user interface. The live video stream will be sent from the android's app using the Google phone's in built camera. Once the user interface previews the live video stream, the user can now start sending the movement signals by using the web controls built in the user interface. The signals for the movement of the robot (T-copter) will be received by the android's app. The android's app pass these signals to the Arduino chip. The software burned into the micro-controller of the Arduino will pass these signals to the respective mechanical part of the robot (T-copter).

3.7.4 Communications Interfaces

3.7.4.1 Communication between user interface and Google phone (mounted on the robot (T-copter))

For controlling the robot's movement initially a connection will be established between the user interface and the Google phone using HTTP protocol. After establishing the connection with the Google phone on the robot (T-copter), now we will be able to send movement commands(move straight, move back, turn right and turn left) to the robot (T-copter) by using HTTP protocol.

3.7.4.2 Live video streaming

For getting live video stream from the robot (T-copter), we will use the RTP/ RTSP (real time streaming protocol). We will initially establish the RTP connection by selecting the start video stream option on the user interface, after establishing the RTP protocol the Google phone will start sending video stream to the user interface and it continue to stream until the user select the stop video stream option on the user interface.

3.7.4.3 Arduino Communication

For communication between mechanical parts of the robot (T-copter) and Arduino, we are going to use communication protocol used in python named serial communication protocol. By using this protocol, Arduino is going to send the signals to respective mechanical part of the robot (T-copter) to control its movement.

3.8 Nonfunctional Requirements

3.8.1 Performance Requirements

3.8.1.1 Response times

Response time is very important for software‘s live video stream module because if there will be less response time then the end user can view the live video stream without long delay. Response time should be less, so that the end user can view the live video stream and control the movement of the robot (T-copter) to the desired direction. If the response time is large then end user will not be able to move the robot (T-copter) to the desired direction as the video stream would have delay factor in that scenario.

3.8.1.2 Throughput

Throughput is important for the software part which will control the different movements of the robot (T-copter). It is concerned with the number of movement commands which a user will send to the software app running on the android Google phone mounted on the robot (T-copter) body. Throughput should sustain the different movements of the robot (T-copter) without any confliction.

3.8.2 Safety Requirements

Lag/Delay in video stream of from the robot (T-copter) can cause possible loss or harm to the end user. This delay may occur due to slow internet connection. .Due to this delay the user may not be able to see in which direction the robot (T-copter) is being moved. Hence, the user can experience some problems while doing surveillance and important information may get lost. To solve this issue we are going to code an efficient video streaming server, which will reduce this delay issue.

3.8.3 Security Requirements

For satisfying the security requirement of the software we need to have efficient coding so that the commands which are being sent to the robot (T-copter) and the live video stream would not be accessible to any unauthorized person. At the user end we need to maintain access to some specific authorized users who will be allowed to log in to the user interface with their account password so that only authorize user could view and control the robot (T-copter) movements.

3.8.4 Software Quality Attributes

3.8.4.1 Adaptability

As software coding is done using java and python, we can use this software in any environment preferred by the user.

3.8.4.2 Availability

It is very important for the user that the software remains online at all times, so that the user can use the software to control the robot (T-copter).

3.8.4.3 Correctness

In this case, information regarding live video stream and robot's movement controls should reach the user and the robot (T-copter) at a specific point of time.

3.8.4.4 Flexibility

The user interface and hardware interface should be flexible enough to pass their information without any hurdle.

3.8.4.5 Interoperability

The communication between the 3 software modules i.e. user interface, android app, Arduino software module, should be flexible enough that the passing of messages do not cause any errors.

3.8.4.6 Portability

The software that is being developed is portable enough that it can work with any mechanical body used.

3.8.4.7 Reliability

The software modules should be reliable enough that the software provides reliable information to the users and to the robot (T-copter). In this project reliability will deal with live streaming video, passing of robot (T-copter) movement signals.

3.8.4.8 Reusability

One of the main qualities of this software is that it can be reused with any type of robotic mechanical body, whether it may land rover or a hex copter.

3.8.4.9 Robustness

This term will deal with the stress handled by the software in extreme conditions. Extreme conditions with our software can be related with slow internet connectivity and frequency of user input. Live up to date video stream should be provided to the user even if the internet connection is slow. Similarly correct input should reach the robot (T-copter) as intended by the user.

3.8.4.10 Testability

In testability we have to check our system's output by providing it with different inputs. These inputs will check the compatibility and stability of the software running on the robot (T-copter) and on the user interface. Also testing will be done to check the live video stream from the phone to the user interface, to check that the video stream reaches the user at the right time on the user interface.

3.8.4.11 Usability

This software can be used to run any robotic mechanical body using android Google phone via internet. The software module running in android can also be used as a separate product in which your mobile can be turned in to a live video streaming device and hence can be used to provide different functions to the user.

3.9 Summary

Software requirement specification covers all the functional and non-functional requirements which are needed for the project. It covers all the main functionality of the system including live video streaming module and the robot's movement control module. The sequence of functions performed for achieving specific is described in detail which gives idea that how a specific task will be completed.

Chapter 04

Software Design and Architecture

4.1 Introduction

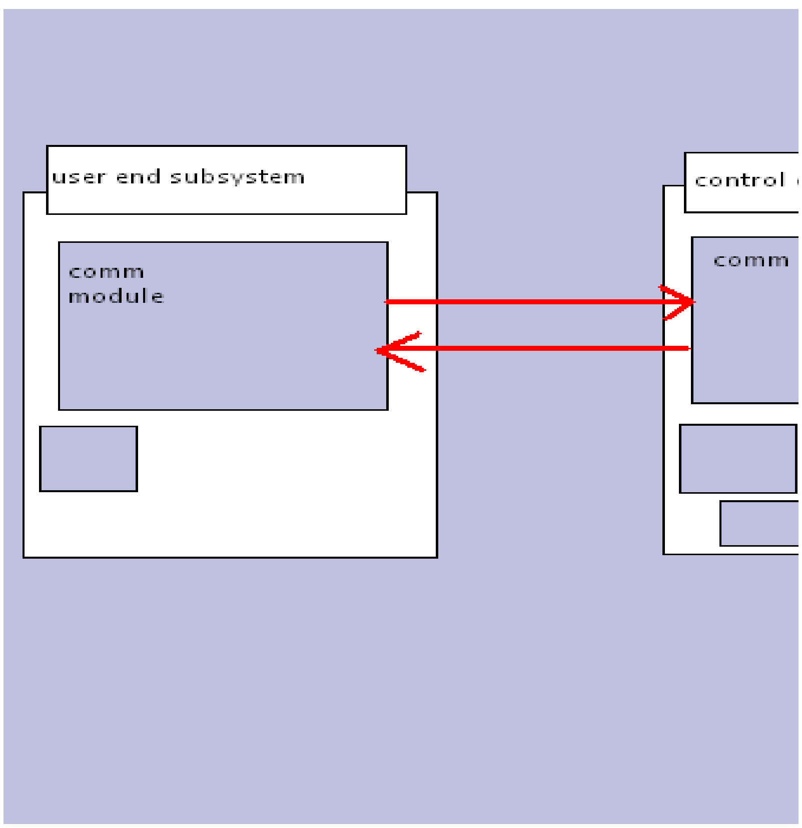
This chapter includes Architecture diagram (level 1, level2, level 3), Use case diagram, Class diagram, sequence diagrams, system sequence diagrams and deployment diagrams. Architecture diagram provides detail view of the Architecture structure of the system and other design diagram provides complete view of the detail design of the system.

4.2 Architecture Diagram

Level 1

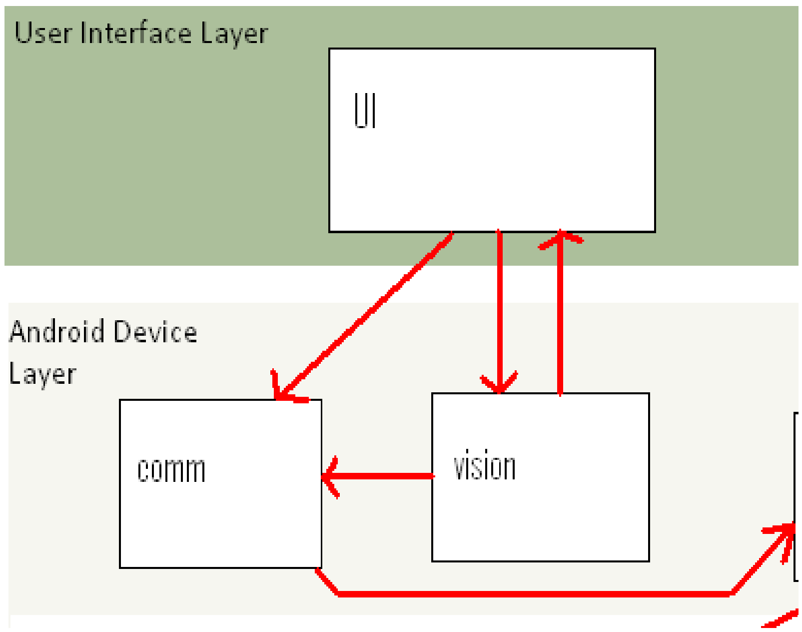
Note: Architecture design pattern is pipes and filters

"Pipes and filters" architecture design pattern is used because multiple Threads are used in the software implementation. Filters are implemented by separate threads.



4.1 Architecture diagram

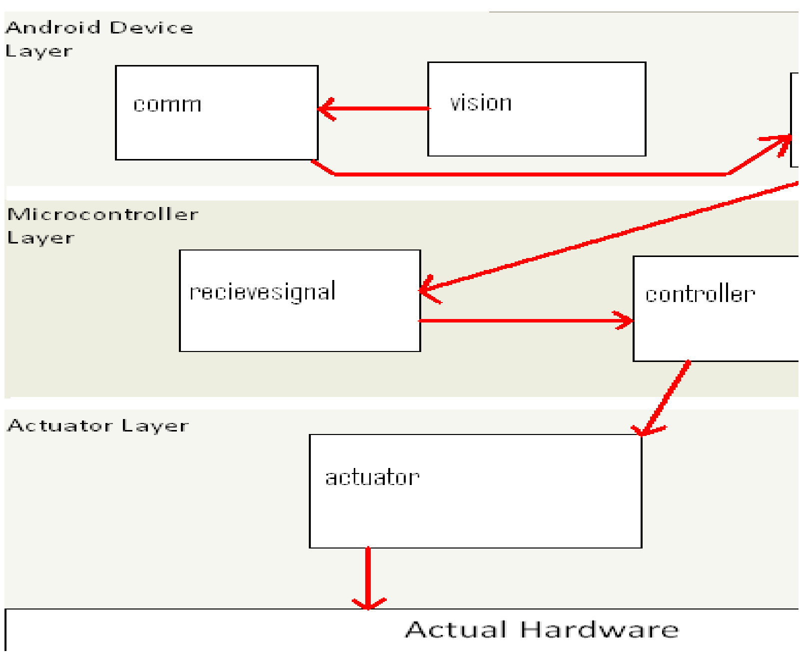
Level 2



4.2 Architecture diagram

At user interface level, UI filter is implemented which passes the data to the communication (comm) filter of the Android device layer by using pipe (shown in red arrow). Comm filter then passes the data to the send signal filter after doing some manipulation on it. Vision filter (live video streaming module) receives the live video stream request from the UI filter and then sends back the data (live video stream).

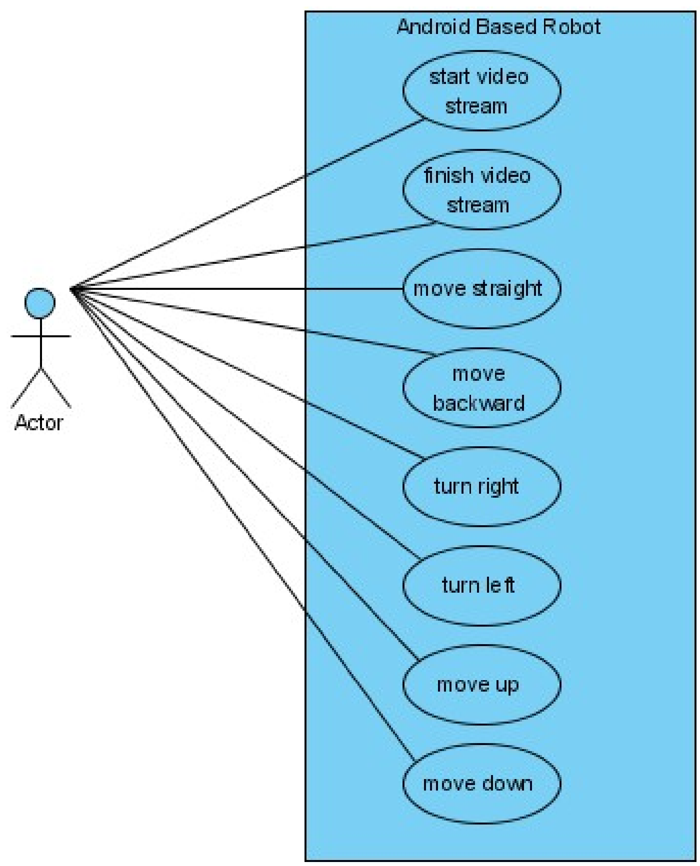
Level 3



4.3 Architecture diagram

At this level, vision filter pass the data through to the comm (communication) module. Comm module sends the data to the send signal. Sendsignal filter will filter the input values and after implementing some logic on it and then pass the data to the receive signal filter. Receive signal filter pass the data to the controller which then pass the data to the actuator layer for getting the movement of the robot (T-copter).

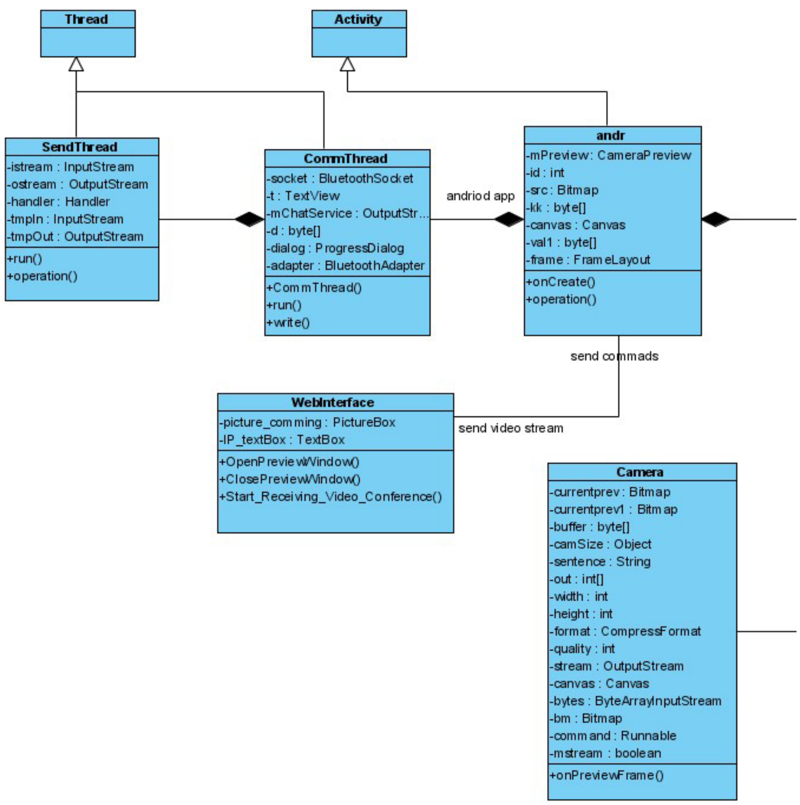
4.3 Use Case Diagram



4.4 use case diagram

There are total eight use cases to which the user will directly interact. Only one primary user will be involved in the system who will directly interact with the system for controlling the robot (T-copter) movement.

4.4 Class Diagram



4.5 Class diagram

In this class diagram, Thread and Activity are the built in classes from which SendThread,CommThread and andr classes are inherited.

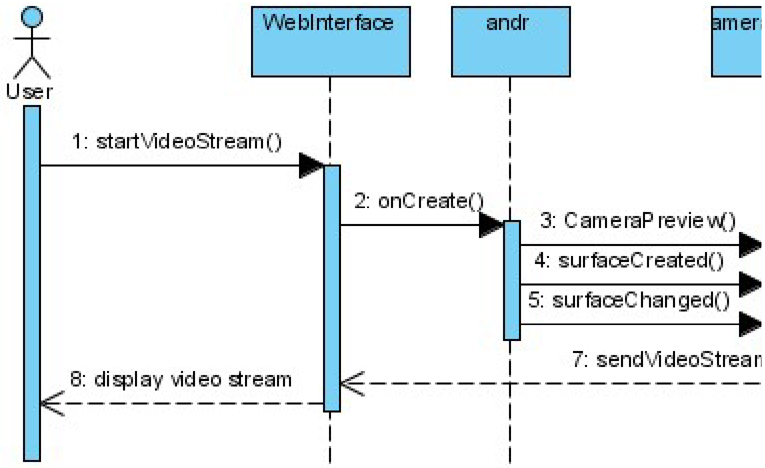
Composition relationship exits between SendThread and CommThread class(object of SendThread is used in CommThread), CommThread and andr class(object of CommThread is used in andr), andr and CameraPreview class(object of andr is used in CameraPreview), Camera and CameraPreview class(object of Camera is used in CameraPreview).

There is also a simple link between use interface class and anrd class because these two class will communicate for sending and receiving the signals to each other.

4.5 Sequence Diagrams

Sequence diagrams shows the complete sequence of actions of a use case that in which sequence the functions of different classes will be called for a specific Use case.

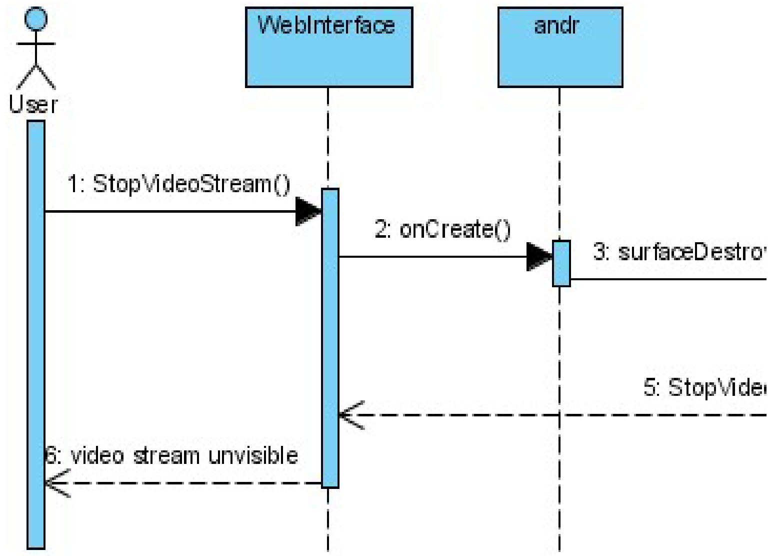
4.5.1 Start video stream



4.6 start video stream diagram

It shows the sequence of functions of different classes involved to start video stream.

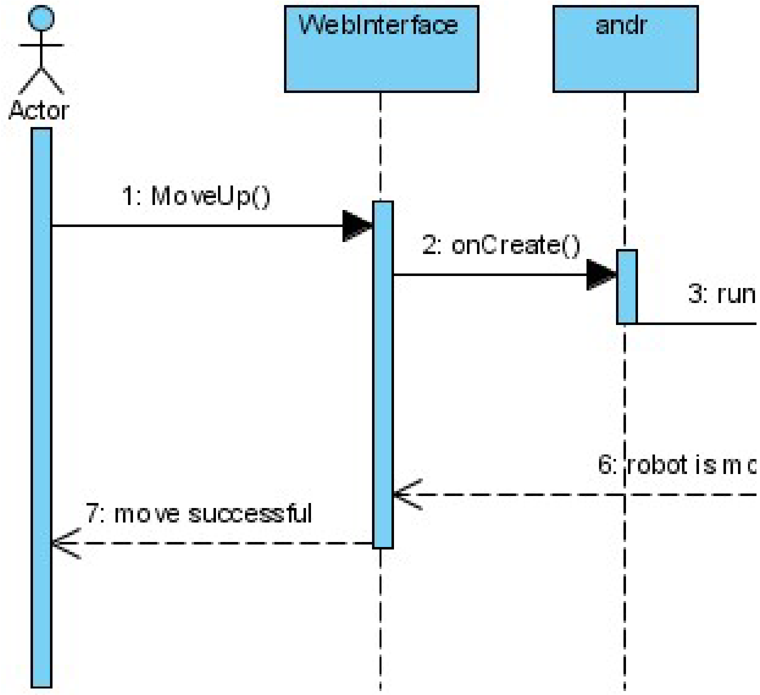
4.5.2 Finish video stream



4.7 finish video stream diagram

It shows the sequence of functions of different classes involved to finish video stream.

4.5.3 Move up

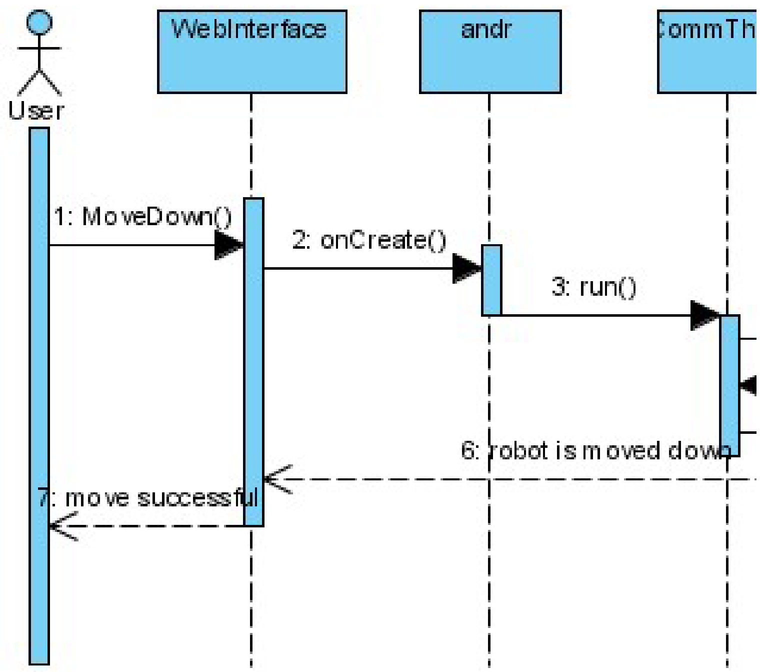


4.8 move up

It shows the sequence of functions of different classes involved to move the robot (T-copter) up.

4.5.4 Move down

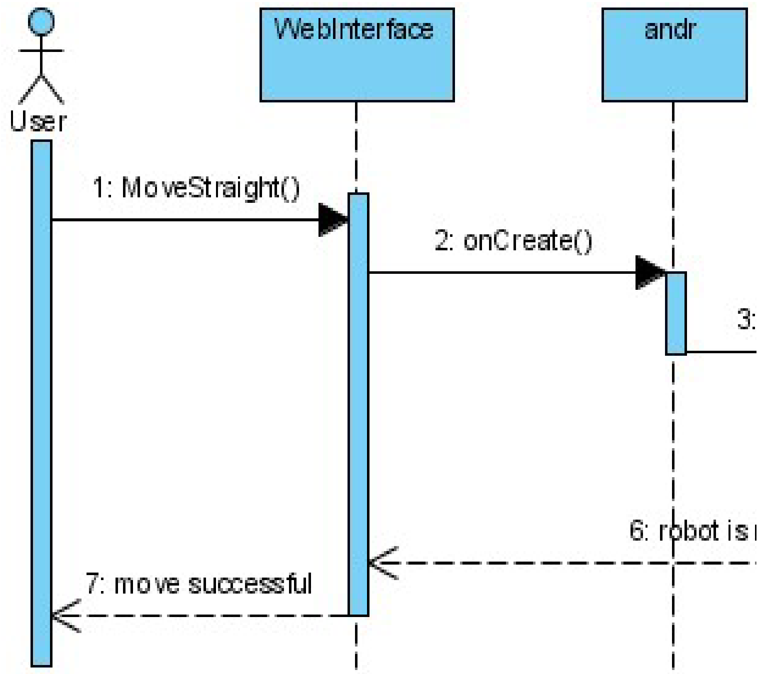
4.5.4 Move down



4.9 move down

It shows the sequence of functions of different classes involved to move the robot (T-copter) down.

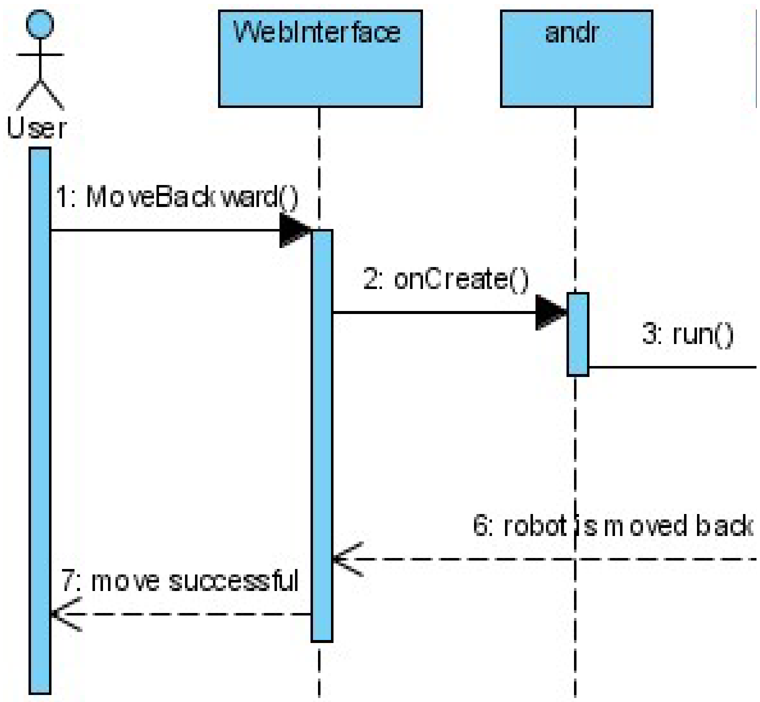
4.5.5 Move straight



4.10 move straight

It shows the sequence of functions of different classes involved to move the robot (T-copter) straight.

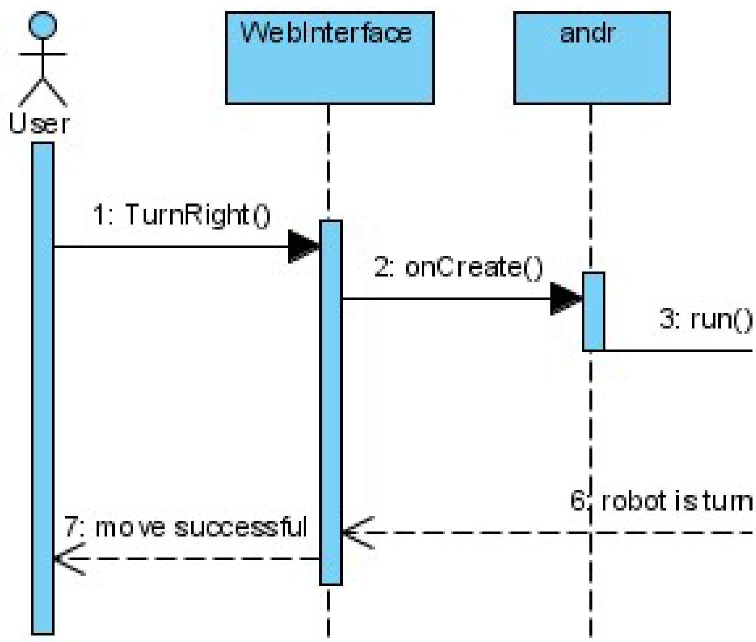
4.5.6 Move backward



4.11 move backward

It shows the sequence of functions of different classes involved to move the robot (T-copter) backward.

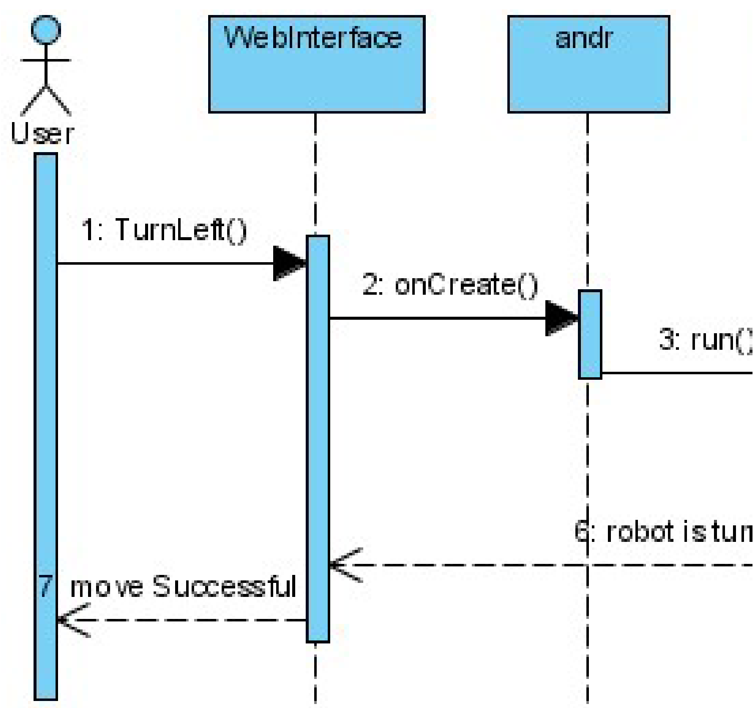
4.5.7 Turn right



4.12 turn right

It shows the sequence of functions of different classes involved to move the robot (T-copter) right.

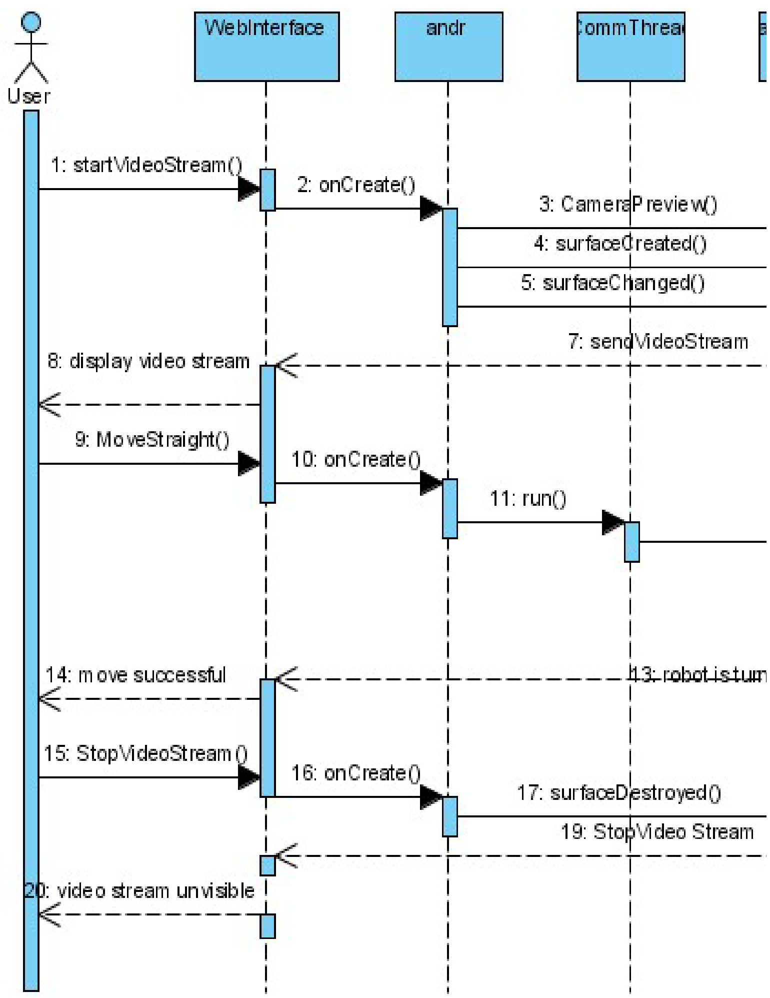
4.5.8 Turn left



4.13 turn left

It shows the sequence of functions of different classes involved to move the robot (T-copter) left.

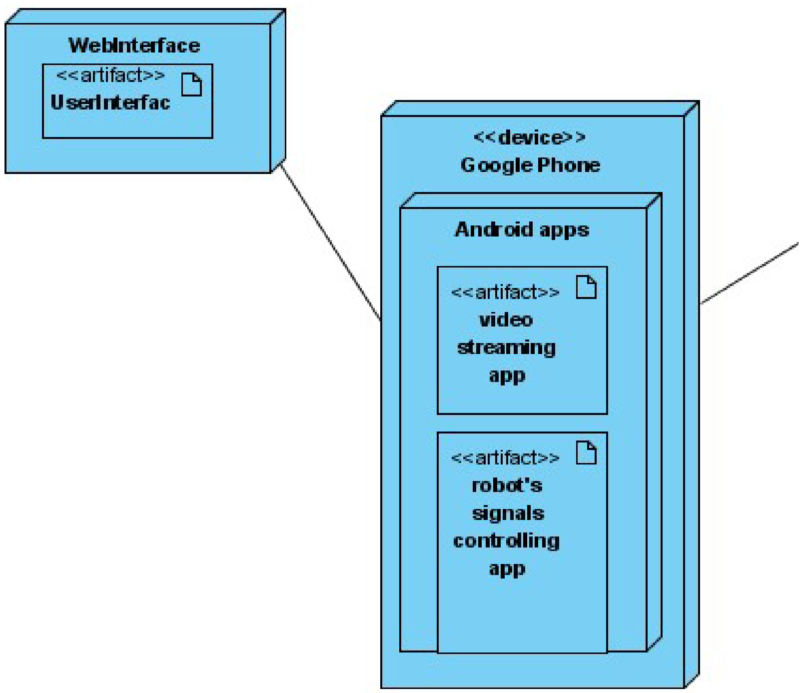
4.5.9 System Sequence Diagram



4.14 system sequence diagram

It shows the sequence of all the functions calls of different classes which are used to carry out all the functionality of the complete system.

4.5.10Deployment diagram



4.15 deployement diagram

There are four nodes in the deployment diagram, including a software node as “user interface” which contain the user interface artifact, hardware node as a device “Google phone” which contains two android applications installed in it. Two other hardware nodes as devices are “Arduino” and “robot (T-copter) body”. The links between the nodes shows the sequence of interaction between the nodes.

4.10 Summary

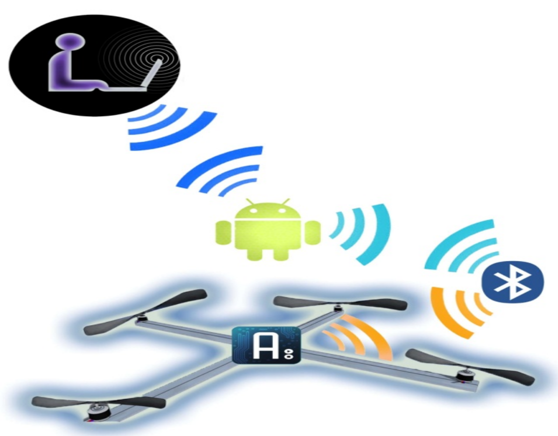
All the description of detail design and architecture is mentioned in this chapter. Detail description of the system functionalities have been showed using use case diagram, sequence diagrams, system sequence diagram and deployment diagram. The architecture design is also explained which gives the overview of the architecture structure of the system.

Chapter 05

Implementation

5.1 Introduction

This chapter describes about the different phases of implementation in detail. Implementation is divided into two sections, software implementation and hardware implementation. Software implementation consists of seven major phases. In the first phase User end interface has been developed. in second phase Android application has been developed with initial



functionality of communication between User end interface and the android application, after establishing the successful communication between the user end interface and the android application then the live video streaming functionality has been implemented in the android application in the third phase. Movement control functionality of the robot (T-copter) has been implemented in the fourth phase of

5.1 android control system

development. In the fifth phase, the communication between the android application and the Bluetooth device has been achieved. The communication between the Bluetooth device and the Arduino has been done in the sixth phase of implementation. In the seventh phase, the communication between the Arduino and the mechanical part of the robot (T-copter) has been done.

5.2 Phases of software implementation

5.2.1 User Interface

This phase of the implementation involves developing the User end interface. At the end of project the user will interact with this interface to control the ROBOT’S movement and for viewing the live video stream coming from the moving robot’s body.

5.2.1.1 Design of interface

User interface consists of total eight buttons as move up, move down, move right, move left, move straight, move backward, start video stream and stop video stream. It also have a video display panel which shows the live video streaming coming from the Google phone mounted on the robot’s body.

5.2.1.2 Logic of interface

Logic of interface has been divided into three parts. In the first part, the connection over wifi between the user end interface and the android application has been made, in the second part live video streaming code has been implemented which request for live video stream from the android application on the robot (T-copter) and then receive the live video streaming coming from the Google phone mounted on the robot’s body and then display the live video stream on the panel on the interface. In the third part, the User interface receive the movement command signals (move up, move down, move right, move left, move straight, move backward) from the user and then it passes that signals to the android application which is running on the Google phone(mounted on the robot's body).

5.2.2 Android application with initial communication implementation

In this phase of implementation, android application has been developed in which the initial communication functionality between the user interface and the android application has been implemented. This logic implemented in this part of android application will enable the android application to listen for the communication request from sent from the user interface using TCP over wifi. This part of the android application was very important because this connection between the user interface and the android application will enable the rest of functionalities (live video streaming and movement control of robot (T-copter)) to be implement in the android application.

A socket is created and then this socket is used to get input stream (getsock.getInputStream()). After creating the socket, an object(outToServer) of "DataInputStream" is created and input stream is then assigned to that object. This "DataInputStream" object (outToServer) then read the data into a buffer so that it could pass to the blutooth device in next phase of development.

5.2.3 Live video streaming implementation in Android application

In this phase of implementation live video streaming functionality has been implemented so that the user at the interface end can view the live video streaming going from the android application which is running on the android operating system (running on Google phone mounted on the robot's body).

The main class of the android application named as "andr" implements the "SensorListener" interface so that all the sensors of the android smart phone can be easily accessed.

In the main class, object of CameraPreview is created so that by using different functionality of this class the video stream can be captured and streamed to the interface at the user end. Object of Bitmap, Canvas and frame are also created so that the live video stream can be captured by using these objects. Object of SensorManager is created so that some specific sensor could be choose as “SensorManager.SENSOR_ORIENTATION” is choosing the gyro sensor of the android smartphone.

“onCreate()” function initialize the services of SensorManager so that the sensors could be accessed. After this Object of CameraPreview is called which starts the thread running in the CameraPreview class. “surfaceCreated()” function is used which initialize and open the camera view and starts its services (Camera.open()). The camera set its display as “ mCamera.setPreviewDisplay(holder)”. In “Camera.PreviewCallback” class “onPreviewFrame()” is used, this function initially create a thread and then start it. In the thread, Object(mJpegData) of “volatile byte[]“ is used which stores the compressed YUV to jpgeg data.

“compressYuvToJpeg()” is the function which converts the YUV data formate to Jpeg formate. In this method object(yuvImage) of “YuvImage” is created which takes the Yuv formate data and then compressed it into the Jpeg formate and then return the compressed data(mJpegCompressionBuffer.toByteArray();).

Socket is created and bind to the ip address and the port number. Object(outToServer) of DataOutputStream is created so that the data could be send back to ther interface at the user end. outToServer object then write the data(mJpegData) back to the user interface. This thread continue its working until the video stream of the camera is available. When the camera stops capturing the video stream the thread will stop its working and "surfaceDestroyed()" will be called in that case which stop the camera preview (mCamera.stopPreview()) and camera will be free of capturing further video stream(mCamera.release()).

5.2.4 Implementation of the movement control functionality of the Robot (T-copter).

For controlling the robot's movement user will click on the specific movement button on the interface and the control signal will be passed to the Android application.

5.2.4.1 The communication between the android application and the Bluetooth device

In android application, “anrd” class of the application will make connection to the Bluetooth device.

An object of the “CommThread” class will be created in the “anrd” class. This object is used to initiate the Thread of the “CommThread” calss. BluetoothAdapter default properties will be passed to the “CommThread” class’s constructor. This application will detect the Bluetooth device by its name or address (adapter.getRemoteDevice("00:06:66:04:E2:57")).

After making connection to the Bluetooth device the Android application will get the user input which is given on the interface by the user. After making a socket, the application will bound the socket with an IP and a port number and then get the input stream. After getting the input stream data, the data is passed to the “write()” function with the object of Bluetooth socket.

In the “write()” function, object of “SendThread” class will be created and thread of that class will be started. In that thread, object of OutputStream will be created and it will send the data to the Bluetooth device.

5.2.4.2 Communication between the Bluetooth device and the Arduino

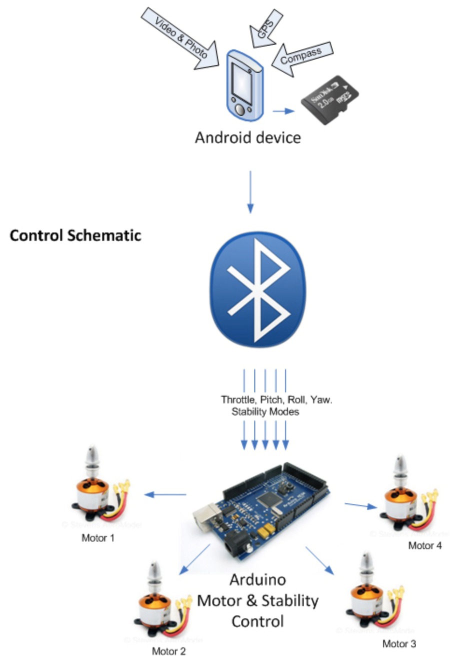
Bluetooth device is connected to the Arduino. A program is written and burn into the Arduino which will receive the data from the Bluetooth device.this program will get the data coming from the blutooth deive and then pass it to microcontroller.

5.2.4.3 Communication between the Arduino and the mechanical part

Arduino program get the data values from the Bluetooth device and then after some manipulation it will use its implemented logic to send control signal to the microcontroller which wil then control the movement of the mechanical part of the robot (T-copter)(by controlling the speed of motors on the robot (T-copter)).

5.3 Hardware Implementation

In the hardware implementation, the body of the robot (T-copter) has been made by integrating different hardware components including brushless motors, speed controllers, battery, props, Arduino, Bluetooth device and Google phone.



5.2 hardware implementation

To control the speed of the brushless motors a program has been written in and burnt into the Arduino. When the Android application gets the contol input from the interface from the user end, the Google phone pass the control values to the Bluetooth device. Bluetooth device then pass that values to the Arduino and Arduino check the input values and based on the input values it passes the control signals to the mechanical part (motor) of the robot (T-copter) and thus move the robot (T-copter) in desired direction.

Using off the shelf hardware prototyping platform called Arduino. It serves all the requirements completely. Following code shows setup function which is always executed when Arduino starts and it shows the serial port open function at speed of 115200 buad.

5.4 Summary

Implementation phase has been divided into two sections, one is the software implementation and the other is the hardware implementation. The software implementation section is further divided into seven major phases. The main two software modules includes live video streaming module and the robot's movement control module are implemented in these seven phases of implementation.

Chapter 06

Testing

6.1 Introduction

Testing has been done at different level in this project to make sure the high quality of the end product. Different testing techniques have been adopted for removing the errors from the system. Different level of testing including unit testing, integration testing and system testing has been done so that the system could be checked in detail and the unwanted results could be removed from the end product. Different types of test cases have been made in every level of testing to make sure that the system provides its required result.

6.2 Unit testing

In unit testing, all the modules have been tested to make sure that they are all working efficiently without producing any kind of error. All the functionality of the software has been tested in this level of testing. Live video streaming module and the robot's movement control module of the android application has been tested in detail in at this level.

For the testing of live video streaming module of the android application, the connection between the user interface and the android application has been tested that they are working efficiently. Live video streaming module is also tested for the time delay issue to make sure that there will not be large time delay in the live video stream.

For testing the robot's movement control module of the android application, initially some dummy inputs were sent to the android application from the user end interface which sends the inputs to the Arduino and Arduino check the input values and use its own program to turn on the specific LED to verify that the control signals are successfully working.

Arduino program is also tested in this level of testing to make sure that it will send the correct command to the mechanical part of the robot (T-copter).

6.3 Integration testing

In integration testing, the testing during the integration of the different software modules and the testing during the integration of software and the hardware part has been done.

In first step of integration testing, the live video streaming module and the robot's movement control module are integrated and tested by running them parallel. The quality of the live video stream has been tested that there must be no long time delay issue after integration of both modules. The robot's movement control module's working is also checked that it's working efficiently as it was before the integration of the both modules.

The testing during the integration of the software and the hardware is also done to make sure that there will be no problems after integrating the software (android application) part with the hardware (Bluetooth device ,Arduino and mechanical parts). Testing has been done after integrating the android application (running on the google phone) with the Bluetooth device and the Arduino. Arduino is then connected to the mechanical part of the robot’s body. While integration, connection between the android application and the Bluetooth device has been tested, the connection between the Bluetooth device and the Arduino has been tested and the conection between the Arduino and the mechanical parts of the robot (T-copter) has been tested.

6.4 System testing

System testing has been done after integrating all the software and the hardware parts. In this testing level, the system as a whole is tested to make sure that it’s giving then required outputs without generating any kind of errors.

For system testing the inputs from the user have been given to the interface and then check that the connection has been made without generating any error and also test the live video streaming to make sure that there is not enough delay in the live video streaming.

robot’s movement is also tested by giving some move command to the interface at the user end so that it could be checked that the robot (T-copter) is moving in the desired direction without generating any kind of unexpected results.

6.5 Summary

Testing has been done at every level to make sure that the end product will not produce some unexpected results. Testing has been done in the software part as well as in the hardware part to make sure that the end system will provide the required quality results without generating errors.

Chapter 07

Results and analysis

7.1 Introduction

This chapter is about the achieved results of the project. The project has been divided into two sections, software and the hardware. The software section includes live video streaming module of the android application and the robot’s movement control module of the same android application. For checking the software results, the output of live video streaming and the robot’s movement control module has been analyzed. For analyzing the hardware results, the flight of the T-copter has been checked and its working as expected at the start of the project.

7.2 software results

7.2.1 Live video streaming module’s results

The software part is divided into two major modules, one is the live video streaming module and the other one is the robot’s movement control module. For checking the results of the live video streaming module of the android application, the live video stream has been analyzed at the user end interface. At the start of the project , it was expected that the live video streaming should have minimum time delay in streaming (less than 1 sec) and when the output of the live video streaming module has been analyzed the live video streaming module was giving accurate results with time delay of less than 1 sec in streaming the live stream from the robot to the User end at the interface side.

7.2.2 ROBOT'S movement control module's results

Second part of the software is the robot's movement control module. This module of the android application has been implemented to control the movement of the robot (T-copter). For analyzing the movement control module's results, different movement inputs are given at the interface at the user end. User gave all the inputs including move up, move down, move right, move left, move forward and move backward. All the movements' commands have been tested and the robot (T-copter) was moving according to the given movement commands.

7.3 Hardware results

Hardware of the system includes all the structure of the T-copter having Arduino, Bluetooth device, Google phone, speed controllers, motors and propeller mounted on it. For analyzing the hardware results, the flight of the T-copter has been checked that it fly properly without facing balancing issues. Different movement commands has been given on the User end interface for checking the flight results of the T-copter that it moves to the right direction as the input is given to it.

7.4 Analysis

The previously made T-copter systems contain much hardware mounted on the body of the T-copter body. The hardware includes separate camera, balancing and other sensors, and RC control system. The previously made T-copter system uses the RC technology for controlling the T-copter.

In Android based robot (T-copter) the hardware cost has been removed as there is only one Google phone which is mounted on the robot (T-copter) which have its own built in camera, wifi and other sensors which can be used for different functionality of the robot (T-copter). The operating system used in the Google phone is the Android which is much powerful. Android provide much processor speed which is needed for performing multiple functionalities of the robot (T-copter) at a time.

7.5 Summary

All the results of the software modules have been analyzed to check that they are working properly as expected in the start of the project(software requirement specification document).Results of the live video streaming and the robot's movement control modules has been checked for validation. The hardware result includes the working of the T-copter that it should not face any balancing problem during its flight. Detailed system validation has been done in this phase.

Chapter 08

Conclusion and Future Work

8.1 Introduction

This chapter includes the conclusion of the project and description about the further extension of the project for enhancing the system functionalities. Further functionalities can be added in this system which can make this robot (T-copter) for use of surveillance purposes in the future.

8.2 Conclusion

This project proves that the android platform can be used in Robotic field. The android smart phone comes with multiple built in sensors which reduces the cost for making a robot (T-copter) because by using these android smart phones there is no need to mount extra hardware (multiple sensors) on the body of the robot (T-copter). There are other smart phone platforms are also available but android SDK is more powerful as compared to others. The high processor speed of the android makes it suitable choice for performing multiple complex functionalities in the Robotic field.

8.3 Future work

This project can be further extended by adding extra functionalities into it. The robot (T-copter) made in this project provides the live video streaming functionality and it can be control by a user manually. In the future, some autonomous mechanism can be implemented in this project so that this robot (T-copter) can controlled without a human effort.

For implementing the autonomous control into this robot (T-copter) we need some extra sensors including IR sensor. IR sensor will be able to detect any obstacle coming in the way of the robot (T-copter) so this way the robot (T-copter) can change its direction by use of some autonomous direction changing algorithms.

Currently the robot (T-copter) is controlled over wifi in this project. This communication can be further upgrade to internet in which the robot (T-copter) will be controlled over internet from a web interface from a remote location. There is need to modify the android application used in this project for controlling it via web interface.

By adding more functionality to this robot (T-copter), it can be used for surveillance purpose of security agencies in the future.

8.4 Summary

The android platform can be used in the Robotic field because it reduces the cost of hardware and all types ofrobotic functionalities can be achieved by it. Android provides the power SDK for developing different applications for robot (T-copter)s. The robot (T-copter) which has been made in this project can be further enhanced as autonomous robot (T-copter). Some further autonomous application and by using some more sensors the robot (T-copter) can be controlled without human effort.

APPENDIX

APPENDIX A

User Manual

1. Overview of Android Based Robot

The aim of this project is to prove that the Android operating system can be used as a platform for the robotics field. This project proves that the Android operating system can support different types of robotic applications efficiently. The software developed in this project controls the robot (quad-copter) through android operating system by using an android application which is running on the Google phone. Live video streaming module of the android application is also developed in the project which is being used to see the live video stream so that the movement of the robot can be monitor from other end.

1.1 Key Features

The main features of Android based Robot are:

- Live video streaming module of android application (Used to send live video from robot's body to the interface at the user end).
- Robot's movement Control module of android application (Used to control the robot's movements).

2. Specification

Specification consists of two sections, the hardware specifications and the software specifications.

2.1 Hardware specification

The following hardware is used in Android based robot.

- Google phone is used on which the android application will be running.
- Arduino is used to receive signal commands from the Bluetooth device and then pass the control signal to the speed contollers after some manipulation.
- Blutooth device is used to receive the control signal

from the Google phone and then pass it to the Arduino.

- 4 Brushless motors are used for controlling the

movement of quad copter.

- 4 Props are used for lifting the quad copter in the

air.

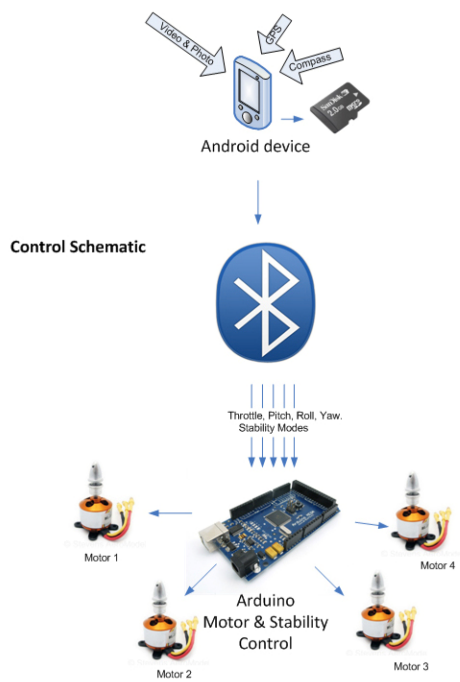
- Speed controllers are used to control the speed of

the brushless motors.

- Quad copter Fram.

- Power battery is used to provide the power to the

whole system.



2.2 software specification

The software of Android based robot consists of two parts as following

- Android application(developed in Java) running on Google phone which consist of two modules, first one is the live video streaming module and the other one id the robot’s movement control module.
- User interface (developed in C#) which provides the interface to the user to control the movement of the robot and to view the live video stream coming from the robot.

3. User interface

User interface consist of two sections, one is for displaying the live video streaming(Main Menu) which is coming from the android application running on the Google phone (mounted on the robot) and the other containing the movement controls of the robot through which the user will control the robot.

3.1 Main Menu

Main Menu interface form consists of three buttons and a display panel.

3.1.1 Start Stream

Press Start stream button for start getting the live video stream from the robot.

3.1.2 Finish Stream

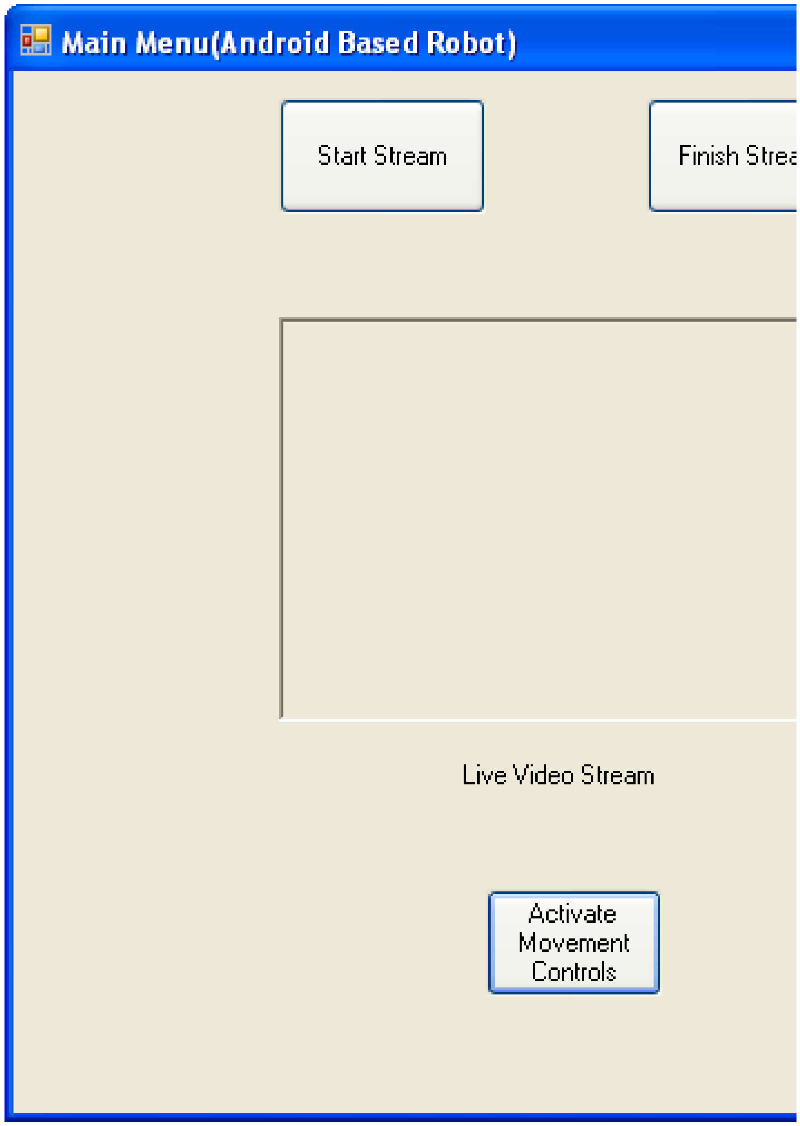
Press Finish Stream for stopping the live video stream coming from the robot

3.1.3 Activate Movement Controls

Press Activate Movement Controls button for controlling the movement of the robot.

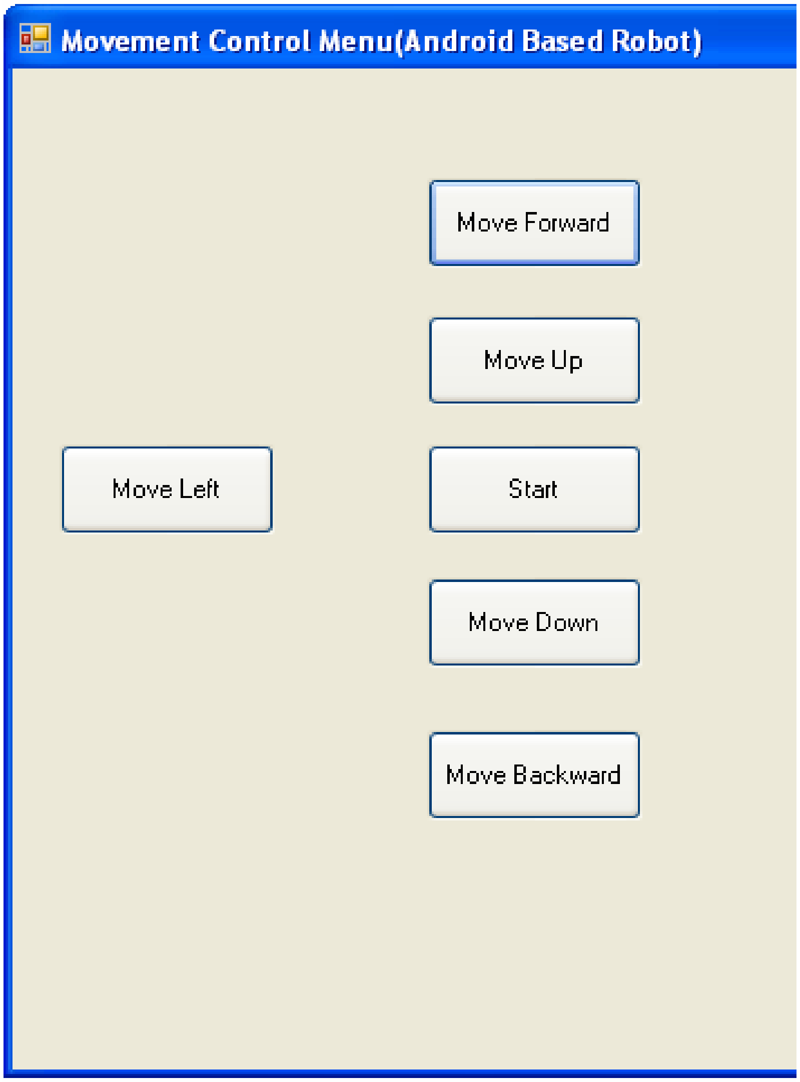
3.1.4 Display Panel

Main Menu form interface also includes a video display panel which displays the live video stream coming from the android application running on the Google phone (Mounted on the robot).



3.2 Movement Control Menu

This interface form is used to control the movement of the robot. Movement Control Menu consists of eight buttons.



3.2.1 Start

Press start button for starting the robots motors.

3.2.2 Move Up

Press move up for moving the robot upward

3.2.3 Move Down

Press move down for moving the robot down

3.2.4 Move Forward

Press move forward for moving the robot forward

3.2.5 Move backward

Press move backward for moving the robot backward

3.2.6 Move right

Press move right for moving the robot right

3.2.7 Move left

Press move left for moving the robot left

3.2.8 Connect

Press Connect for initializing the movement controls.

Bibliograph

y

Bibliography

[1] Gabriel Hoffmann, Compilers, Schematic of reaction torques on each motor of a quadrotor aircraft, due to spinning rotors, Retrieved January 23, 2011, from http://en.wikipedia.org/wiki/File:Quadrotor_yaw_torque.png

[2] Kerman, Judith B. (1991). Retrofitting Blade Runner: Issues in Ridley Scott's Blade Runner and Philip K. Dick's Do Androids Dream of Electric Sheep Bowling Green, Bowling Green State University Popular Press. ISBN 0-87972-509-5.

[3] Bray, Tim (28 April 2010). "Multitasking the Android Way". Android Developers. Retrieved 2011-03-03. from http://android-developers.blogspot.com/2010/04/multitasking-android-way.html

[4] Tarmo Virki and Sinead Carew (2011-01-31)."Google topples Symbian from smartphones top spot".Reuters. Retrieved 1 February 2011 from http://uk.reuters.com/article/2011/01/31/oukin-uk-google-nokia-idUKTRE70U1YT20110131

[5]Android Open Source Project. Open Handset Alliance. Retrieved 2011-03-04, from http://en.wikipedia.org/wiki/Open_Handset_Alliance

[6] Jackson, Rob (10 December 2008). "Sony Ericsson, HTC Androids Set For Summer 2009". Android Phone Fans. Retrieved 2011-04-07 from http://phandroid.com/2008/12/10/sony-ericsson-htc-androids-set-for-summer-2009/

[7] Markoff, John (2007-11-04). "I, Robot: The Man Behind the Google Phone". The New York Times. Retrieved 2011-4-9. from http://www.nytimes.com/2007/11/04/technology/04google.html?_r=2&hp=&pagewanted=all

[8] Kirsner, Scott (2007-09-02). "Introducing the Google Phone". The Boston Globe. Retrieved 2011-4-4. from http://www.boston.com/business/technology/articles/2007/09/02/introducing_the_google_phone/

[9] Pearce, James Quintana (2007-09-20). "Google's Strong Mobile-Related Patent Portfolio".mocoNews.net. Retrieved 2011-03-7 from http://www.moconews.net/entry/419-googles-strong-mobile-related-patent-portfolio/

[10] Mithun Chandrasekhar (2 February 2011). "Google's Android Event Analysis". AnandTech. Retrieved 24 February 2011. "I confirmed this with Google; Honeycomb, at least in the current form, will not be coming to non-tablet devices.from http://www.anandtech.com/show/4150/googles-android-event-analysis/2

[11] "Android Brand Guidelines". Android. 26 March 2011. Retrieved 2011-04-2 from http://www.android.com/branding.html

[12] "Android Compatibility". Android Open Source Project. Retrieved 31 jan 2011 ,from http://source.android.com/compatibility/index.html

[13] Greg Kroah-Hartman (2010-02-02). "Android and the Linux kernel community". Retrieved 2011-05-02,from http://www.kroah.com/log/linux/android-kernel-problems.html

[14] "Microchip ChipKit and Arduino pin compatibility analysis". Rugged Circuits, Retrieved 2011-05-04, from http://ruggedcircuits.com/blog/2011/05/22/microchip-chipkit-and-arduino-pin-compatibility-analysis/

[15] "ArduinoBoardMega". Arduino. Retrieved 2011-05-

02,from http://arduino.cc/en/Main/ArduinoBoardMega.