**SMART HELMET FOR AGRICULTURE**

**A PROJECT REPORT**

*Submitted in partial fulfillment of the*

*requirement for the award of the*

*Degree of*

**BACHELOR OF TECHNOLOGY**

**in**

**ELECTRONICS AND COMMUNICATION ENGINEERING**

*By*

**CHUNDURI ASRITHA-13BEC1042**

**CHAVA SAI TEJA -13BCE1032**

**VALETI LALITHA-13BEC1160**

*Under the Guidance of*

**Dr.S.R.S.PRABAHARAN**

****

SCHOOL OF ELECTRONICS ENGINEERING

VIT University

CHENNAI. (TN) 600127

*(MAY 2017)*

**HELMET FOR AGRICULTURE**

**A PROJECT REPORT**

*Submitted in partial fulfillment of the  
 requirement for the award of the*

*Degree of*

**BACHELOR OF TECHNOLOGY  
in**

**ELECTRONICS AND COMMUNICATION ENGINEERING**

*By*

**CHUNDURI ASRITHA-13BEC1042**

**CHAVA SAI TEJA – 13BCE1032**

**VALETI LALITHA-13BEC1160**

*Under the Guidance of*

**Dr.S.R.S.PRABAHARAN**

****

SCHOOL OF ELECTRONICS ENGINEERING

VIT University

CHENNAI. (TN) 600127

*(MAY 2017)*

***CERTIFICATE***

This is to certify that the Project work titled “ ***Smart helmet for agriculture***” that is being submitted by “***Chunduri Asritha***” and “***Valeti Lalitha”*** and “***Chava Sai Teja***” is in partial fulfillment of the requirements for the award of **Bachelor of Technology**, is a record of bonafide work done under my guidance. The contents of this Project work, in full or in parts, have neither been taken from any other source nor have been submitted to any other Institute or University for award of any degree or diploma and the same is certified.

**Guide**

**The thesis is satisfactory / unsatisfactory**

**Internal Examiner External Examiner**

|  |  |
| --- | --- |
| Program Chair | Dean (SENSE) |

**ACKNOWLEDGEMENTS**

Working on the project under proper guidance and help was a great chance for learning and professional development. During this tenure of working on the project we came across a lot of learning new things. We are extremely grateful to **Dr.S.R.S.Prabaharan,** Dean of the School of Electronics Engineering, VIT University, and our Program Chair **Dr. R. Manojkumar**for their support.

We use this opportunity to express our deepest gratitude and special thanks to our guide **Dr.S.R.S.Prabaharan** in spite of being extraordinarily busy with his duties, took time out to hear, guide and keep us on the correct path.

We express our deepest thanks to **Mr. Gasper**, Analog Circuits Lab Assistant for helping in setting up the hardware and for arranging the required facilities for the project. We choose this moment to acknowledge his contribution gratefully.

We will strive to use gained skills and knowledge in the best possible way, and will continue to work on their improvement.

**TABLE OF CONTENTS**

|  |  |
| --- | --- |
| **LIST OF TABLES** | **7** |
| **LIST OF FIGURES** | **8** |
| **LIST OF ABBREVIATIONS** | **9** |
| **Abstract** | **10** |
| **1 Introduction** | **11** |

1. Objective
2. Background
   1. *Main problem* **11**
   2. *Existing System* **11**
   3. *Different Solutions*  **12**

**2** **Project Description**

|  |  |
| --- | --- |
| 2.1 Methodology | **13** |

1. Design Approach

|  |  |  |
| --- | --- | --- |
|  | *2.2.1 Weed detection part( using Image processing)* | **16** |
|  | *2.2.2 Power supply for raspberry pi* | **17** |
|  | *2.2.3 Power supply in spraying part* | **17** |
|  | *2.2.4 Data transmission* | **19** |
| 2.3 | Standards | **21** |
| 2.4 Constraints, Alternatives and Tradeoffs | |  |
|  | *2.4.1 Constraint-1* | **22** |
|  | *2.4.2 Constraint-2* | **22** |
|  | *2.2.3 Constarint-3* |  |
|  | *2.2.4 Constraint-4* |  |
| 2.5 | Technical Specification |  |
|  | *2.5.1 Raspberry Pi 3b* | **24** |
|  | *2.5.2 MAX 232* | **24** |
|  | *2.5.3 RF transmitter and receiver* | **25** |
|  | *2.5.4 PIC 16F877A* | **26** |
|  | *2.5.5 Relay driver and circuit* | **27** |
|  |  |

|  |  |  |
| --- | --- | --- |
| **3** | **Project Demonstration** | **29** |
| **4** | **Cost Analysis** | **33** |
| **5** | **Results and discussions** | **34** |
| **6** | **Conclusion** | **38** |
| **7** | **Appendix** | **39** |
| **8** | **References** | **47** |

**LIST OF TABLES**

**Table 1 -** Cost analysis

**LIST OF FIGURES**

Figure 1: Block diagram of the entire project

Figure 2: Block diagram for weed detection algorithm

Figure 3: Flow diagram of entire project

Figure 4: Power supply for raspberry pi

Figure 5: Power supply for Microcontroller

Figure 6: Raspberry pi 3model B

Figure 7: Pin diagram of MAX 232

Figure 8: RF transmitter pin diagram

Figure 9: RF receiver pin diagram

Figure 10: pin diagram of PIC16f877A

Figure 11: Relay driver

Figure 12: Selecting a microcontroller in MPLAB

Figure 13: Raspberry Pi and camera

Figure 14: Raspberry pi interfacing with RF transmitter

Figure 15: Sprayer prototype implementation

Figure 16: Motor for spraying pesticide

Figure 17: Overall circuit of raspberry pi and power supply through solar panel and super capacitor

Figure 18: Raspberry Pi’s screen On Laptop

Figure 19: Output of raspberry pi after image processing

Figure 20: Histogram of taken image

Figure 21: Edge detection of taken image

**LIST OF ABBREVIATIONS**

**RFID** - Radio FrequencyIdentification **MCU** –Microcontroller

**TX** –Transmit

**RX** –Receive

**IEEE** –Institute of Electrical and Electronics Engineers **ISO** –International Organization for Standardization **IEC** - International Electro technical Commission

**USB** –Universal Serial Bus **OS** –Operating System

**SPI** –Serial PeripheralInterface **RF** –Radio Frequency

**DIP** –Dual Inline Package

**RISC** –Reduced Instruction Set Computer **RAM** –Random Access Memory

**EEPROM** –Electrically Erasable Programmable Read Only Memory

**LED** –Light Emitting Diode

**AC** –Alternating Current

**DC** –Direct Current **GND** –Ground

**PCB** –Printed Circuit Board

**IC** –Integrated Circuit

**i/o** –input/output

***Units:***

**GHz** –Giga Hertz **V** - Volt

**MHz** –Mega Hertz **mA** –milli Ampere

**KHz** –Kilo Hertz **cm** - centimeter

**Kbps** –Kilo bits per second **s** –seconds

**Mbps** –Mega bits per second

**ABSTRACT**

Agriculture is the backbone of human sustenance on this world. Now a days with growing population weneed the productivity of the agriculture to be increased a lot to meet the demands.

In olden days they used natural methods to increase the productivity, such as using the cow dung as a fertilizer in the fields. That resulted increase in the productivity enough to meet the requirements of the population.

But later people started thinking of earning more profits by getting more outcomes. So, there came a revolution called “Green Revolution”. After this period usage of deadly poisons as herbicides has increased to a drastic level. By doing so we succeeded in increasing the productivity but we overlooked damage done to the environment, which will arise a doubt in our sustenance on this beautiful earth.

So, in this project we have implemented some methods to reduce the usage of herbicides by spraying them in appropriate amounts only in the weed affected areas. In this project we implemented image processing technique to detect the weed areas in an image we took from the fields and sent trigger to automatic sprayer to spray adequate quantity of pesticide.

**CHAPTER 1**

**INTRODUCTION**

* 1. **OBJECTIVE**
* Detecting the weeds in the crop through the webcam mounted on the helmet.
* Processing the taken images using image processing technique
* Depending on the area of weed we calculate the amount of pesticide to be sprayed to the effected crop.
* This calculated amount of pesticide to be sprayed is send to automatic sprayer which is present in the fieldthrough RF module in the form of a trigger.
  1. **BACKGROUND**
     1. *Main problem*

As India is mostly an agriculture based country people concentrate more on productivity of crop, so unknowingly people are using chemical based pesticide to meet their expectations.Excess use of pesticides to the crop leads to pollution such as soil pollution, air pollution. This in turn leads to loss in fertility of soil and may also lead to decrease in productivity of crop for future generations. Thus there should be change in process of using pesticides to the crop.

* + 1. *Existing system*

With the advancement in the technology people started using the herbicides to remove the weeds.But to detect the weeds they are still using manual power in many parts of the world.Later there came few methods to detect the weeds automatically but due to lack of their accuracy they are unable to reach to the people.

* + 1. *Different solutions*

Manual Detection:

* In olden days weed detection was done by employing some men especially for that purpose.
* They will detect and pluck them out manually using their hands.
* This takes lot of time and also involves lot of physical work.
* When it comes to large farms this method is not efficient due to requirement of more labor, thus leads to increase in cost also.

Spraying of pesticides (manually)

* In this method manually pesticides are sprayed in a random quantity.
* Sometimes this leads to excess use of pesticides which may leads to different pollutions.
* Due to excess pesticide usage gradually fertility of land also decreases, thus reduction in efficiency.
* Mostly now-a-days people are using it. Cost is also less when compared to manual detection.

Automated Spraying:

* In this method image processing is used for detecting the weed from the taken images of the crop.
* Later this data is used for automated spraying of pesticide.
* This is time efficient.
* Reduces damage caused to crop by excess pesticide use.

**CHAPTER 2**

**PROJECT DESCRIPTION**

**2.1 METHODOLOGY**

The proposed system is to detect the weed in the crop and then trigger the data i.e. regarding amount of weed, to the automated sprayer. Here webcam is used for capturing images of the crop. This webcam is connected through USB cable to one of the USB port of raspberry pi. Taken images of the crop are processed using image processing technique in the raspberry pi. First open CV need to configured in raspberry pi.

BLOCK DIAGRAM

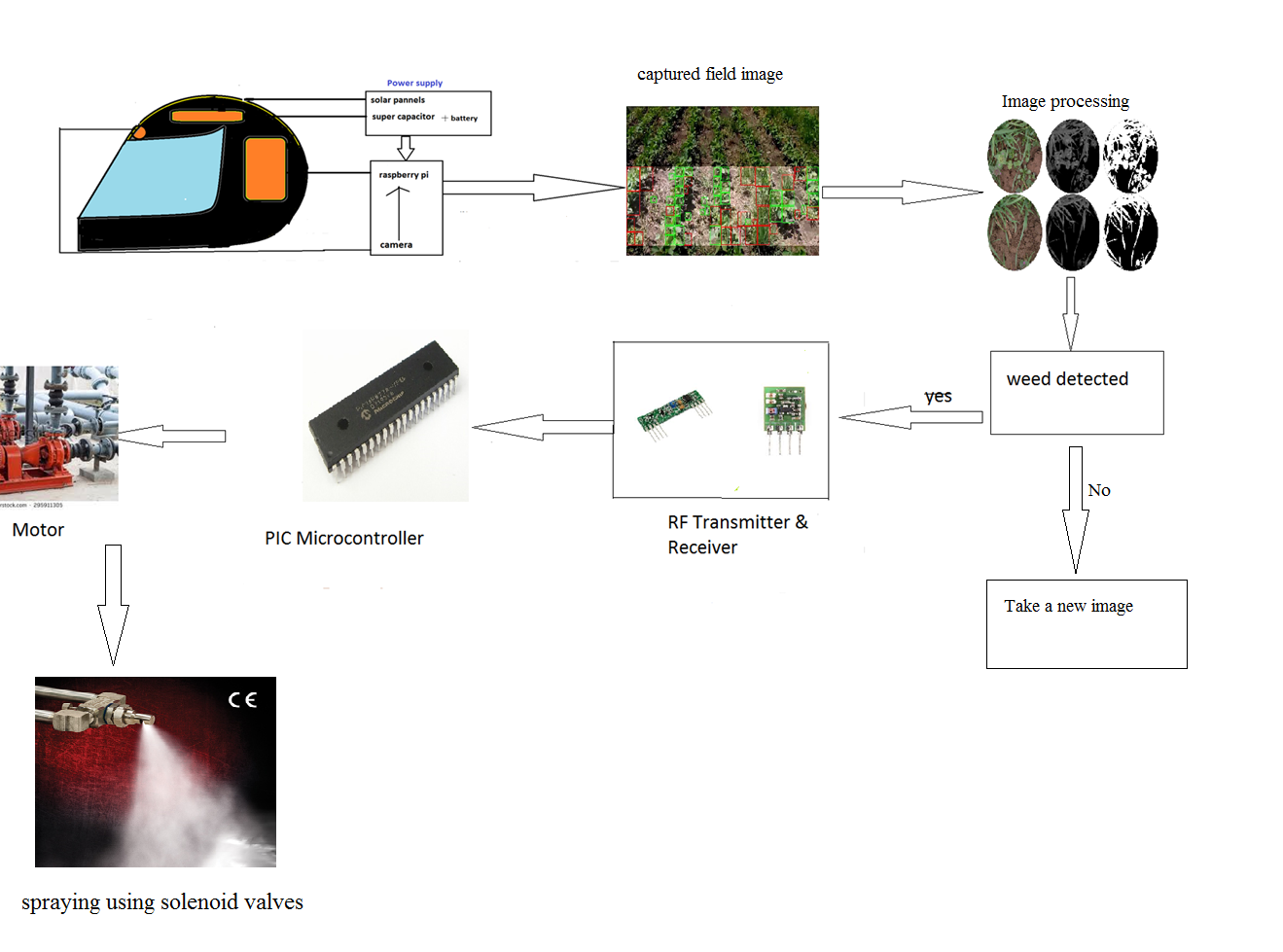


Figure 1: Block diagram of the entire project

Code is to be configured in python language is used for detecting weed using image processing in raspberry pi. Power supply for raspberry pi is through solar panels, super capacitor and battery which is the most efficient way to supply power and there would be no need of charging the battery separately or change it frequently. So all the 3 modules i.e. camera, raspberry pi, power supply (solar panels, super capacitor and battery) for pi are placed in helmet. For transmitting the output of image processing i.e. amount of weed present in the crop we will be using RF transmitter .It is connected to raspberry pi.

Now in the sprayer part which is present in the field itself we will be using PIC microcontroller to process the data received from raspberry pi. In PIC microcontroller , code for automatic spraying of accurate quantity of pesticide to effected crop is configured. The data from raspberry to PIC microcontroller is received using RF receiver.

Using the PIC microcontroller code the motor will switch on for a particular amount of time depending on weed quantity.

WEED DETECTION ALGORITHM:

The algorithm prepares an image for further advanced processing and is consists of: Loading the imagefrom source, colour segmentation and edge detection.

Edge detection is also a method of image segmentation which uses the fact that the edge frequencies and veins of both the crop and the weed have different density properties (strong and weak edges), to separate the crop from the weed. The image after both color segmentation and edge detection is left with the edges and veins of both the crop and the weed in white and the remaining part completely black.

In color segmentation we draw histogram for our taken images. Then calculate the mean of color of leaf and background. Thus if mean value is greater than 1800 we will consider there is disease in the crop.

So by applying AND operation on edge detector and color segmentation we will get to know the amount of weed in the crop. Thus required amount of pesticide is sprayed to crop.

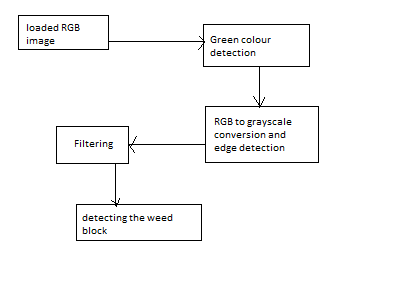


Figure 2: Block diagram for weed detection algorithm

Here we used “**canny edge detector algorithm**” for edge detection:

The Process of Canny edge detection algorithm can be broken down to 5 different steps:

1. Apply Gaussian filter to smooth the image in order to remove the noise
2. Find the intensity gradients of the image
3. Apply non-maximum suppression to get rid of spurious response to edge detection
4. Apply double threshold to determine potential edges
5. Track edge by [hysteresis](https://en.wikipedia.org/wiki/Hysteresis): Finalize the detection of edges by suppressing all the other edges that are weak and not connected to strong edges.

Color segmentation using histogram:

* You can consider histogram as a graph or plot, which gives you an overall idea about the intensity distribution of an image. It is a plot with pixel values (ranging from 0 to 255, not always) in X-axis and corresponding number of pixels in the image on Y-axis.
* It is just another way of understanding the image. By looking at the histogram of an image, you get intuition about contrast, brightness, intensity distribution etc of that image. Almost all image processing tools today, provides features on histogram.

Find Histogram

BINS: The histogram shows the number of pixels for every pixel value, i.e. from 0 to 255 which requires 256 values to show the above histogram. The number of pixels for all pixel values separately are not required, and number of pixels in a interval of pixel values are needed, such as the number of pixels lying between 0 to 15, then 16 to 31, ..., 240 to 255.Then only 16 values are needed to represent the histogram.For that whole histogram need to be split into 16 sub-parts and value of each sub-part is the sum of all pixel count in it. This each sub-part is called "BIN.BINS is represented by the term histSize in OpenCV docs.

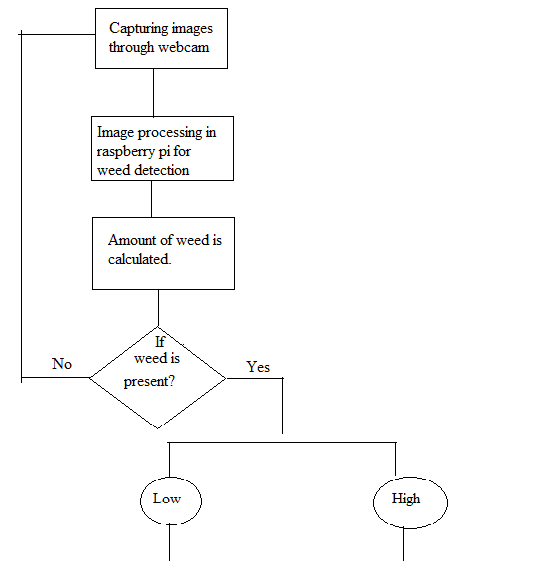
RANGE: It is the range of intensity values need to measure. Normally, it is [0,256], i.e. all intensity values.

Plotting Histograms

There are two ways for this,

1. Short Way : use Matplotlib plotting functions
2. Long Way : use OpenCV drawing functions

FLOW CHART



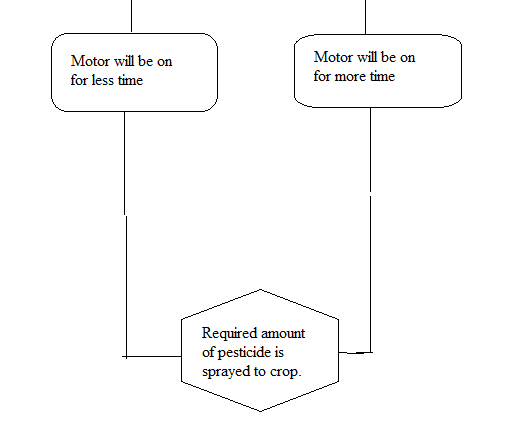


Figure 3: Flow diagram of Smart Helmet for Agriculture

* 1. **DESIGN APPROACH**
     1. ***Weed detection part(using Image processing)***

Here the camera through which crop images are taken is connected to raspberry pi in which image processing is done. Webcam is connected through USB cable to one of the USB port of raspberry pi.

* + 1. ***Power supply for raspberry pi***

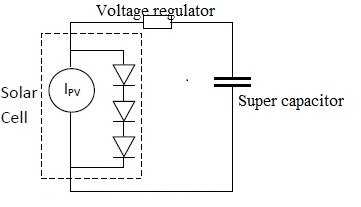


Figure 4: Power supply for raspberry pi

The energy harvester is like a hose with an endless supply of water filling a barrel, which is analogous to a super capacitor. If the hose is still running once the barrel is full, the water may overflow. This situation differs from that of a battery, which has a limited energy supply and thus would require a series regulator.

* + 1. ***Power supply in spraying part***

It consists of a transformer which has its primary winding connected to the mains supply. A secondary winding, electro-magnetically coupled but electrically isolated from the primary is used to obtain an AC voltage of suitable magnitude and it drives the circuit to the rectifier which converts the AC power supply to the DC power supply. Here the rectifier circuit used is the bridge rectifier. Here two diodes in series conducts at any one time .The main advantage is that the diodes requires only half the reverse break down voltage. The rectifier is followed by a filter to convert to dc and finally the 12v dc supply is sent in to a voltage regulator .Here the 7805 voltage regulator is used to provide a fixed voltage of 5V to the microcontroller.

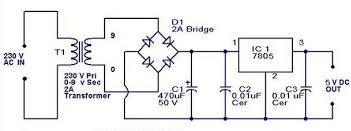


Figure 5: Power supply for Microcontroller

* + 1. ***Data transmission***

For transmitting data from raspberry pi to PIC microcontroller we will be using RF transmitter and receiver.

RF TX-434**:**  The transmitter output is up to 8mW at 433.92MHz with a range of approximately 100 foot (open area) outdoors.  Indoors, the range is approximately 50 foot, and will go through most walls. It accepts both linear and digital inputs, can operate from 1.5 to 12 Volts-DC

RF RX-434:  The receiver also operates at 433.92MHz, and has a sensitivity of 3uV.  The RWS-434 receiver operates from 4.5 to 5.5 volts-DC, and has both linear and digital outputs.

* 1. **STANDARDS**
* Wi-Fi

The Raspberry Pi 3 is equipped with 2.4 GHz Wi-Fi.

IEEE standard is 802.11n (150 Mbit/s).

* RF transmitter

The transmitter output is up to 8mW at 433.92MHz with a range of approximately 100 foot (open area) outdoors.  Indoors, the range is approximately 50 foot, and will go through most walls.

* RF receiver

The receiver also operates at 433.92MHz, and has a sensitivity of 3uV.  The RWS-434 receiver operates from 4.5 to 5.5 volts-DC

* USB

IEEE 1394 is an interface standard for a serial bus for high-speed communications and isochronous real-time data transfer.

* 1. **CONSRTAINTS ALTERNATIVES AND TRADEOFF**

***2.4.1 Constraint-1***

For showing the design as a prototype the webcam and the RF module that were chosen are meant only for a smaller range and smaller distances. Hence this system cannot be implemented directly in the fields. But the same idea can be used for the real time implementation.

***Alternative***

As an alternative high resolution camera, more efficient power supply can be used.

***Tradeoff***

Cost will be more for implementing the above mentioned alternative.

***2.4.2 Constraint-2***

As of know we are using Canny edge detection algorithm combined with color segmentation for detecting the weed based on the criterion of part with color other than green as weed. Hence, if weed is of green color it will not be identified.

***Alternative***

Depending on edge frequency of different kinds of weed we need to differentiate between weed and normal plant. Thus above mentioned problem is solved.

***Tradeoff***

For each kind of crop there will be different edge frequencies for normal crop and weed. So it will be difficult to design a overall algorithm for weed detection in different crops.

***2.4.3 Constraint-3***

Here, in our weed detection code color segmentation only with white background and green leaf and respective threshold values is implemented. So if the background coloris changed then the output will not be as desired.

***Alternative***

The threshold values need to accordingly changed based on the background color whenever there is change in the implementation environment.

***Tradeoff***

It will be time consuming process to change the threshold values for every change in the implementation surrounding.

***2.4.4 Constraint-4***

Solar panels and super capacitor are used as power supply which will not meet the power requirement for raspberry pi.

***Alternative***

So we need to use a 12v battery paralleled with solar panels as power supply for raspberry pi.

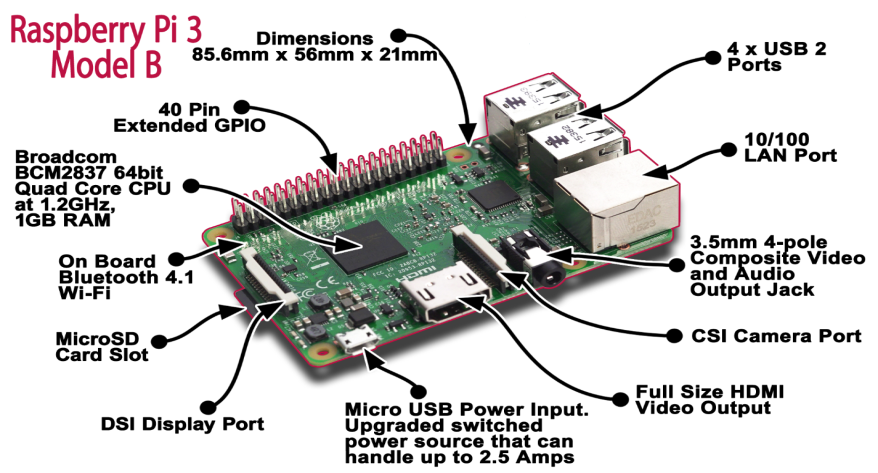
***Tradeoff***

12v battery is over weighing to be inserted in the helmet as the farmer can carry it only when he is stationary. And also it gives power supply for just 30mins to 1hr that we use.

**2.5 TECHNICAL SPECIFICATIONS**

***2.5.1 Raspberry pi 3 b***

The raspberry pi used in the project is of version3 and model b with specifications



Available [online]:<https://www.element14.com/community/docs/DOC-80899/l/raspberry-pi-3-model-b-technical-specifications>

Figure 6: Raspberry pi 3model B

* Next Generation QUAD Core Broadcom BCM2837 64bit processor
* Processor speed of 1.2Ghz
* BCM43438 Wi-Fi on board
* Upgraded switched power source up to 2.5 Amps (can now power even more powerful devices over USB ports)
* 40pin extended GPIO

* 4 x USB 2 ports
* 4 pole Stereo output and Composite video port
* Full size HDMI output
* CSI camera port for connecting the Raspberry Pi camera
* DSI display port for connecting the Raspberry Pi touch screen display
* Micro SD port for loading your operating system and storing data
* Upgraded switched Micro USB power source (now supports up to 2.5 Amps)

***2.5.2MAX232***

Here we are using RS232 as a serial communication interface between pi and rf transmitter

## C1+ Vcc

1. 16
2. 15
3. 14
4. 13
5. 12
6. 11
7. 10
8. 9

V+ GND

MAX

2

3

2

C1- T1 out

**M**

**A**

**X**

**2**

**3**

**2**

C2+ R1 in

C2- R1 out

### V-T1 in

### T2 out T2 in

R2 in R2 out

#### Figure 7 : Pin diagram of MAX 232

* Power Supply: +3.0V to +5.5V
* No. of Ext. Capacitors: 4
* Operates up to 120 kbit/s
* Two Drivers and Two Receivers

***2.5.3RF Transmitter & Receiver***

The RF TX-434 and RF RX-434 are extremely small, and are excellent for applications requiring short-range RF remote controls.

***TX-434:***

* The transmitter output is up to 8mW at 433.92MHz
* with a range of approximately 100 foot (open area) outdoors and indoors, the range is approximately 50 foot
* operate from 1.5 to 12 Volts-DC

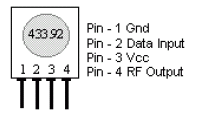


Figure 8: RF transmitter pin diagram

***RF RX-434:***

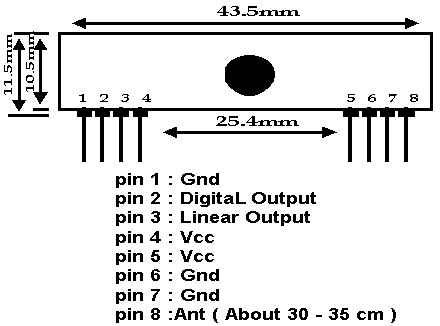
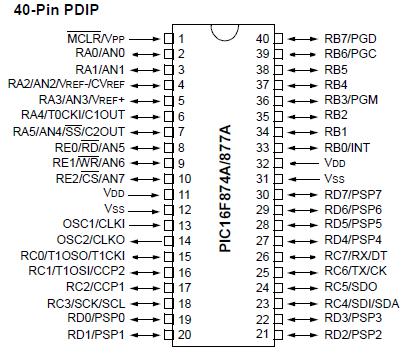
* operates at 433.92MHz
* operates from 4.5 to 5.5 volts-DC, and has both linear and digital outputs.
* 

Figure9: RF receiver pin diagram

***2.5.4 PIC16F877A***

The microcontroller used in this project is pic16f877a 40 pin DIP. It is a 8 bit microcontroller with RISC architecture with 35 set of instructions. The instructions are 14bit long.[6]

* Max operating speed-20MHz
* Time for 1 instruction cycle = Osc/4
* 8kBytes of program memory
* 368 Bytes RAM,256B EEPROM
* 13bit program counter
* Maximum output current by i/o pin =25mA
* Maximum operating voltage =5V



Available[online]:http://microcontrollerslab.com/pic16f877a-introduction-features/

Figure 10: pin diagram of PIC16f877A

***2.5.5 Relay Driver Circuit:***

We are using the following relay driver circuit to control the motor.

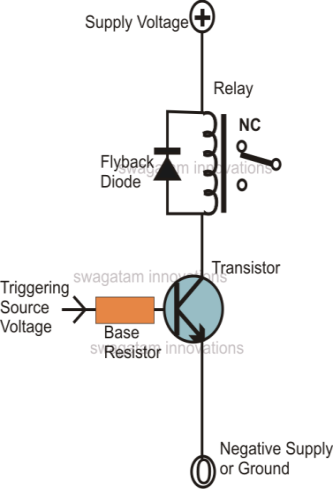


Figure11 : Relay driver

* CL100B npn transistor
* Electromagnetic relay

***2.5.6 Software specifications***

MPLAB X IDE-v3.5

MPLAB X IDE is a software program that runs on a PC (Windows, Mac OS, Linux to develop applications for Microchip microcontrollers and digital signal controllers. It is called an Integrated Development Environment (IDE), because it provides a single integrated environment to develop code for embedded microcontrollers. MPLAB X IDE is based on the open source NetBeans IDE from Oracle. Taking this path has allowed us to add many frequently requested features very quickly and easily, while also providing us with a much more extensible architecture to bring you even more new features in the future.

XC8 COMPILER

The MPLAB XC8 is a full-featured, highly-optimized ANSI C compiler for the PIC10/12/16/18 microcontroller families.The compiler is used along with the IDE -

MPLAB X IDE. This supports all of the microchip’s 8-bit microcontrollers. It provides wide range of libraries that can be used for the easy of programming the microcontroller.

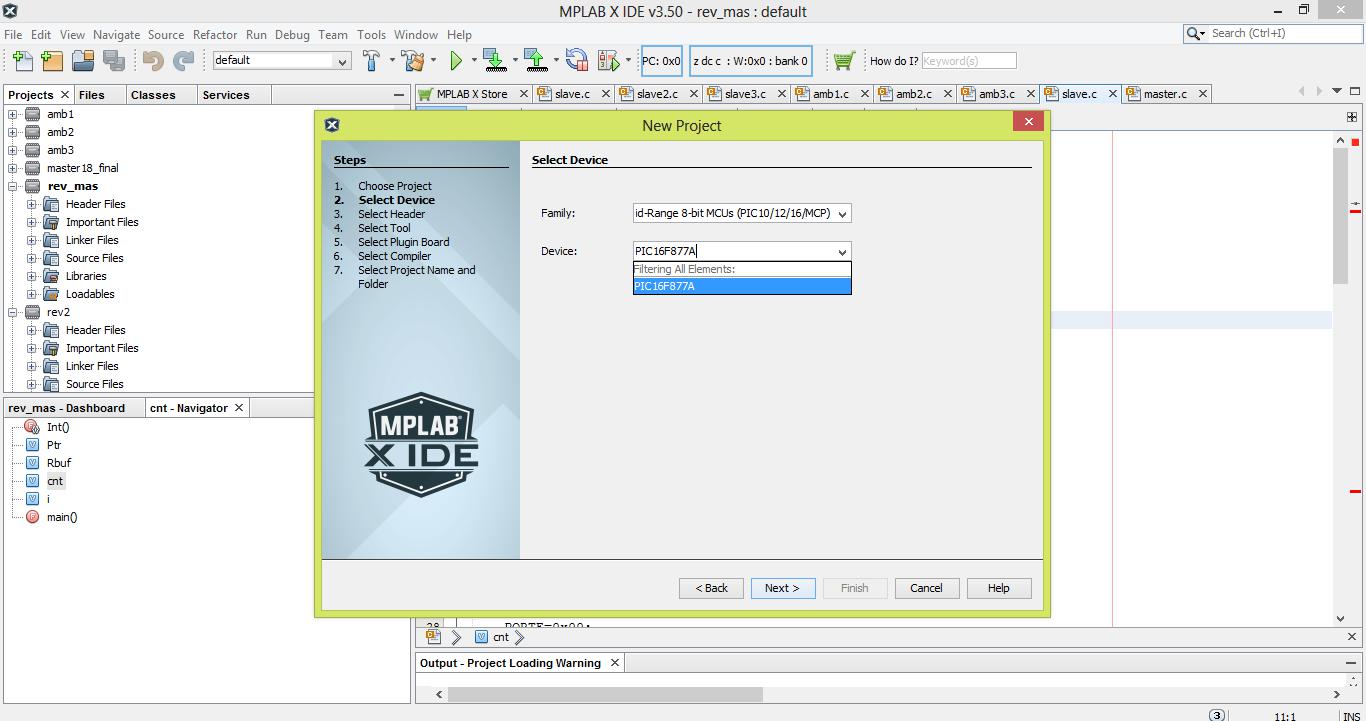


Figure 12: Selecting a microcontroller in MPLAB

**CHAPTER 3**

**DEMONSTRATION OF PROJECT**



Figure13 :Raspberry Pi and camera



Figure 14: Raspberry pi interfacing with RF transmitter

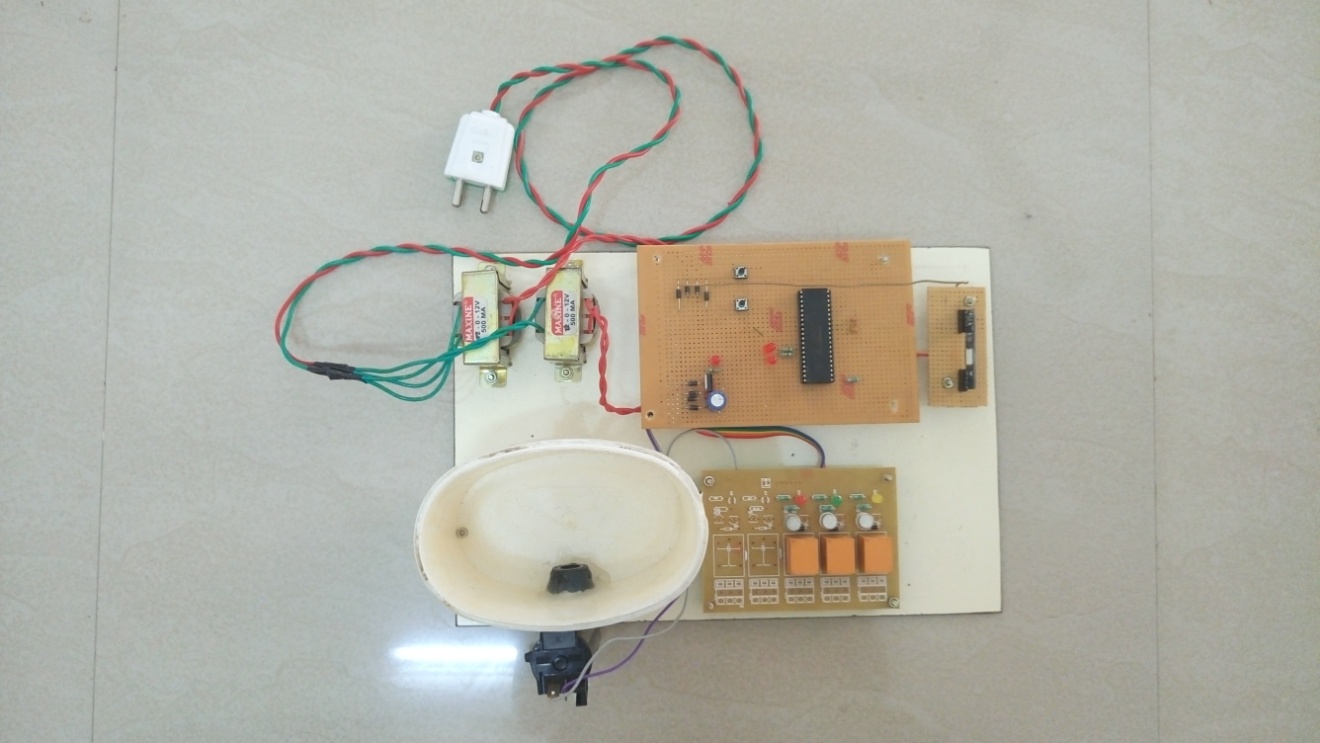


Figure 15: Sprayer prototype implementation



Figure 16: Motor for spraying pesticide

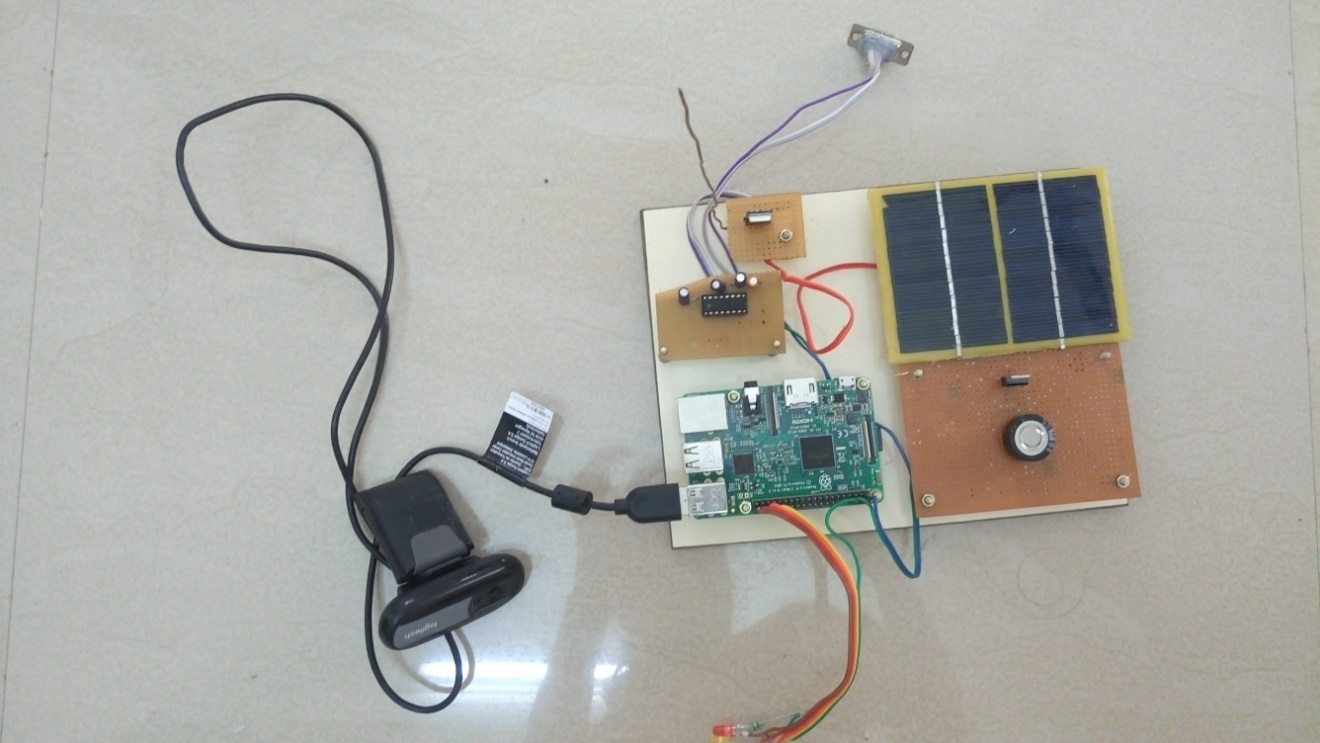


Figure17 : Overall circuit of raspberry pi and power supply through solar panel and super capacitor

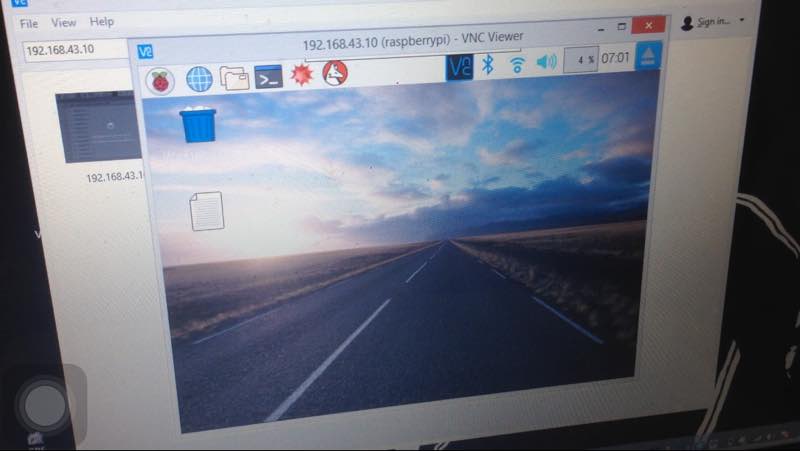


Figure18 :Raspberry Pi’s screen On Laptop

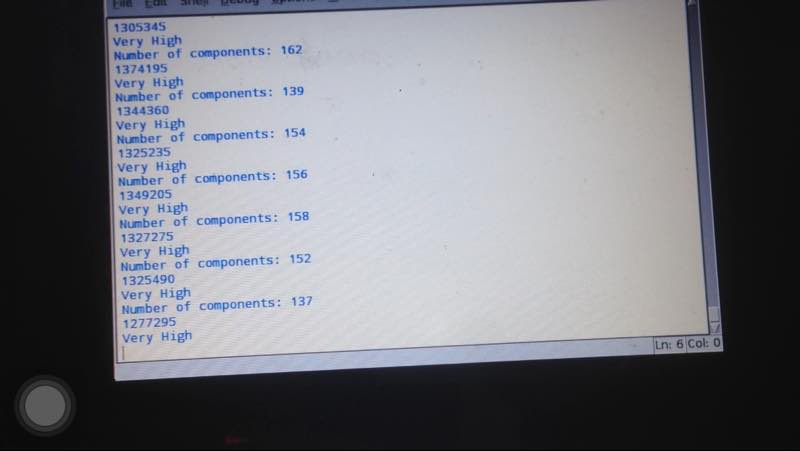


Figure19 :Output of raspberry pi after image processing

**CHAPTER 4**

**COST ANALYSYS**

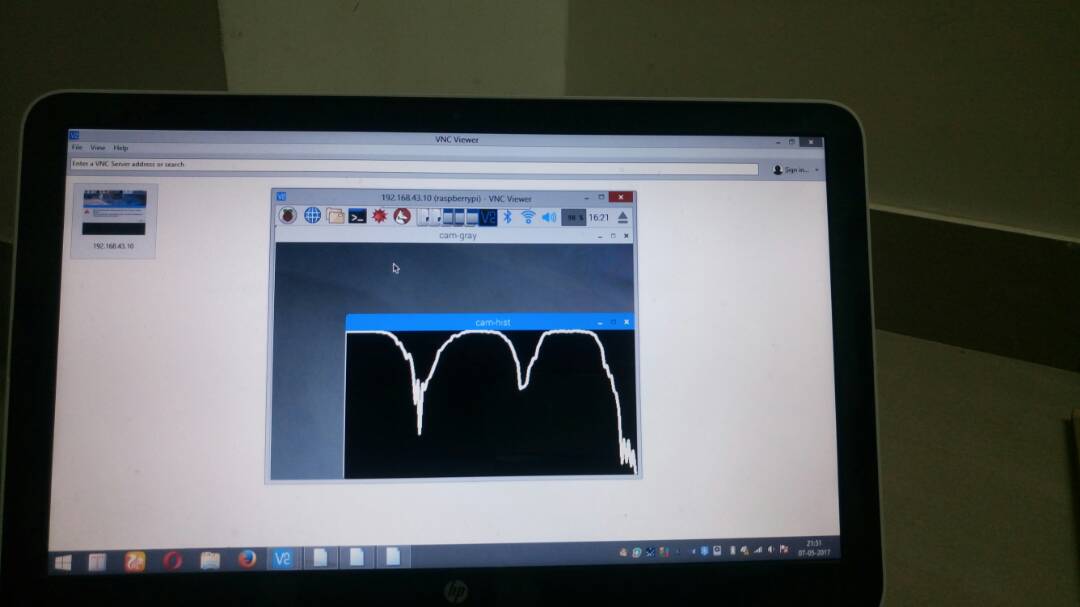
|  |  |  |  |
| --- | --- | --- | --- |
| SR.NO | COMPONENTS | COST |  |
|  |  |  |  |
| 1 | PIC 16F877A+IC base | 1X170 | 170 |
|  |  |  |  |
| 2 | raspberry pi 3 B | 1X 3000 | 3000 |
|  |  |  |  |
| 3 | Webcam | 1 X 1000 | 1000 |
|  |  |  |  |
| 4 | Crystal Oscillator | 1X 10 | 10 |
|  |  |  |  |
| 5 | Capacitors |  | 20 |
|  |  |  |  |
| 6 | Resistor |  | 20 |
|  |  |  |  |
| 7 | IC7805 | 1 X 15 | 15 |
|  |  |  |  |
| 8 | AC Transformer | 1X 120 | 120 |
|  |  |  |  |
| 9 | RF434 MODULE | 1X 160 | 160 |
|  |  |  |  |
| 10 | D1N4001 Diodes |  | 10 |
|  |  |  |  |
| 11 | PCB | 1X25 | 25 |
|  |  |  |  |
| 12 | Battery | 1X60 | 60 |
|  |  |  |  |
| 13 | Solar panels |  | 300 |
|  |  |  |  |
| 14 | LED |  | 15 |
|  |  |  |  |
| 15 | Super capacitor |  | 350 |
|  |  |  |  |
| 16 | Helmet |  | 700 |
|  |  |  |  |
|  |  | TOTAL | 5970 |
|  |  |  |  |

Table 1: Cost analysis

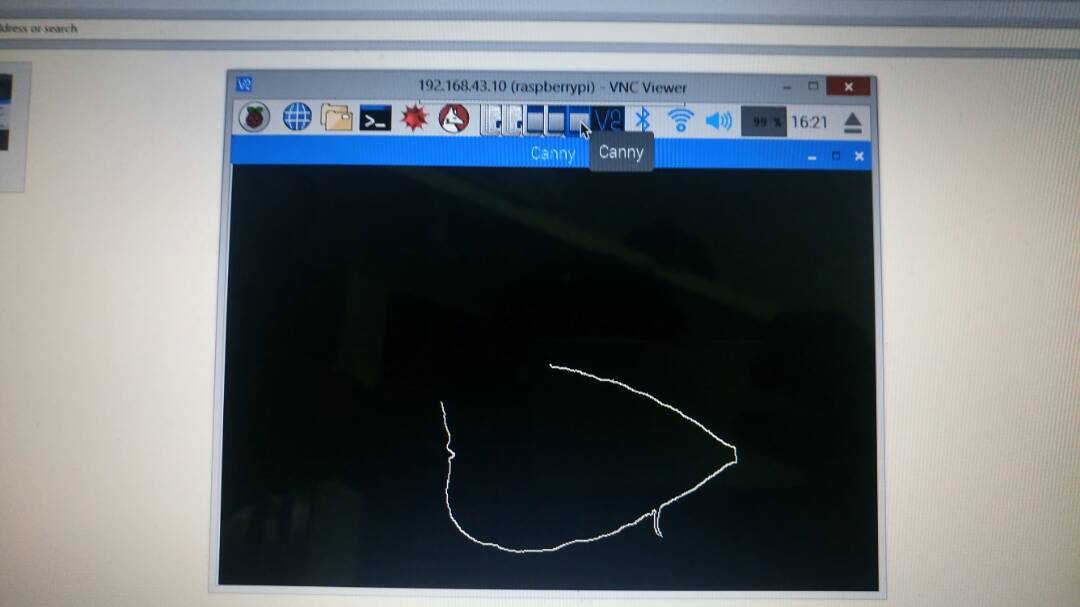
**CHAPTER 5**

**RESULTS AND DISCUSSIONS**

* Images of crop are taken using webcam and are further processed in raspberry pi.
* Implementation of weed detection algorithm on the crop images in pi.
* Amount of weed in the crop is calculated.
* From calculated weed amount, time for which motor should be on to spray the pesticide on the weed is set.
* Thus a particular amount of pesticide is sprayed for respective weed quantities.
* Apart from image processing, the important aspect of system is power supply. It is through solar panels, super capacitor and battery, the most efficient way of power supply.

****

**Figure 20: Histogram of taken image**

****

**Figure 21: Edge detection for taken image**

**CHAPTER 6**

**CONCLUSION**

In this project we have developed a method by which we can detect the weed using image processing. The input of the detected weed area is given to the automatic sprayer which sprays only in the weed area. By doing so we can reduce the usage of pesticides, thus saving the environment. And also we are using helmet as a protection to the farmer from sun and also as a platform for power supply using solar panels mounted on it.

**CHAPTER 7**

**APPENDIX**

import cv2

importnumpy as np

import serial

import string

importRPi.GPIO as GPIO

import time

fromskimage import measure

frommatplotlib import pyplot as plt

GPIO.setmode(GPIO.BOARD)

GPIO.setup(32,GPIO.OUT)

GPIO.setup(36,GPIO.OUT)

GPIO.setup(38,GPIO.OUT)

GPIO.setup(40,GPIO.OUT)

port=serial.Serial(

"/dev/ttyUSB0",

baudrate=2400,

parity=serial.PARITY\_NONE,

stopbits=serial.STOPBITS\_ONE,

bytesize=serial.EIGHTBITS,

writeTimeout = 0,

timeout = 10)

print(port.isOpen())

print("Port Opened...")

defdraw\_hist(name, gray):

hist = cv2.calcHist([gray], [0], None, [256], [0,256])

globalc\_val

c\_val = np.mean([hist[np.nonzero(hist)]])

MAX = max(hist)

plot = np.zeros((256,512))

fori in range(255):

x1 = 4\*i

x2 = 4\*(i+1)

y1 = hist[i]\*512/MAX

y2 = hist[i+1]\*512/MAX

cv2.line(plot, (x1,y1), (x2,y2), 1, 3)

cv2.imshow(name + "-gray", gray)

cv2.imshow(name + "-hist", plot)

def main():

cam = cv2.VideoCapture(0)

while cv2.waitKey(10) == -1:

ret, frame = cam.read()

gray = cv2.cvtColor(frame,cv2.COLOR\_BGR2GRAY)

#draw\_hist("cam",gray)

draw\_hist("cam",frame)

print(c\_val)

edges = cv2.Canny(gray,100,200)

cv2.imshow('Canny',edges)

val\_out = np.sum(edges)

L = measure.label(edges)

print "Number of components:",np.max(L)

LL = np.max(L)

if LL <= 10:

print('Normal')

port.write('[0]')

elif LL > 10:

print(val\_out)

if ((val\_out< 100000) & (c\_val> 1800)):

print ('Low')

port.write('[1]')

GPIO.output(36,True)

time.sleep(1)

GPIO.output(36,False)

#time.sleep(1)

if ((val\_out> 100000) & (val\_out< 300000) & (c\_val> 1800)): # ET

print ('High')

port.write('[2]')

GPIO.output(38,True)

time.sleep(1)

GPIO.output(38,False)

#time.sleep(1)

if ((val\_out> 300000) & (c\_val> 1800)):

print ('Very High')

port.write('[3]')

GPIO.output(40,True)

time.sleep(1)

GPIO.output(40,False)

#time.sleep(1)

cv2.waitKey(10)

if \_\_name\_\_=="\_\_main\_\_":

main()

PIC CODE FOR SPRAYING:

#include <pic.h>

#include <stdio.h>

#include "delay.c"

unsigned char Rbuf[8];

shortintPtr=0,U1p=0;

shortintRrdy\_Flag=0,Sync\_Flag=0;

static void interrupt Int();

void main()

{

ADCON0 = 0x81;

ADCON1 = 0x80;

TRISA = 0x00;

TRISB = 0x00;

TRISE = 0x00;

TRISC = 0x80;

TRISD = 0x00;

PORTA = 0x00;

PORTB = 0x00;

PORTE = 0x00;

PORTC = 0x80;

PORTD = 0x00;

SPBRG = 103;

TXEN = 1; // Enable transmit

BRGH = 1; // ; Select high baud rate

SPEN = 1; // Enable Serial Port

CREN = 1; // Enable continuous reception

RCIF = 0; // Clear RCIF Interrupt Flag

RCIE = 1; // Set RCIE Interrupt Enable

SYNC = 0;

PEIE=1;

GIE=1;

for(;;)

{

if(Rrdy\_Flag == 1)

{

Rrdy\_Flag = 0;

if(Rbuf[0] == '1')

{

PORTC = 0x01;

DelayMs(1000);

DelayMs(1000);

DelayMs(1000);

PORTC = 0x00;

}

else if(Rbuf[0] == '2')

{

PORTC = 0x02;

DelayMs(1000);

DelayMs(1000);

DelayMs(1000);

DelayMs(1000);

DelayMs(1000);

DelayMs(1000);

PORTC = 0x00;

}

else if(Rbuf[0] == '3')

{

PORTC = 0x04;

DelayMs(1000);

DelayMs(1000);

DelayMs(1000);

DelayMs(1000);

DelayMs(1000);

DelayMs(1000);

DelayMs(1000);

DelayMs(1000);

DelayMs(1000);

DelayMs(1000);

DelayMs(1000);

DelayMs(1000);

PORTC = 0x00;

}

else if(Rbuf[0] == '0')

{

PORTC = 0x00;

}

}

}

}

static void interrupt Int()

{

if(RCIF)

{

RCIF = 0;

if(Sync\_Flag != 0)

{

if(RCREG == ']')

{

Rrdy\_Flag = 1;

Sync\_Flag = 0;

}

else

{

if((RCREG >= 0x00) && (RCREG <= 0xFF))

Rbuf[Ptr++] = RCREG;

}

}

else

{

if(RCREG == '[')

{

Sync\_Flag = 1;

Ptr = 0;

}

}

}

}

**CHAPTER 8**

**REFERENCES**

IEEE paper: Image Processing in Agriculture

Mrs. Latha1, A Poojith2, B V Amaranth Reddy3, G Vittal Kumar4 Assistant Professor, ECE Department, Amrita School of Engineering, Bangalore, India1 Student, ECE Department, Amrita School of Engineering, Bangalore, India2,3,4

**International journal of innovation research in electrical, electronics, instrumentation and control engineering Vol 2, Issue 6, June 2014**