CHRIST THE KING POLYTECHNIC COLLEGE
COIMBATORE-641032
DEPARTMENT OF ELECTRONICS AND COMMUNICATION ENGINEERING
FULLY AUTOMATIC SMART IRRIGATION SYSTEM

LIST OF STUDENTS &DESIGNATION

1. 18403593- ISMAYIL SHABAH.N.V
2. 18403592- HARIHARAN.S
3. 18403598- MADHUMITHA.M
4. 18403617- UDHAYAKUMAR.M
5. 18124842- JAYAVINOTHKUMAR.R

NAME OF THE GUIDE

Mr GARRY THOMSON.P B.E.,
LECTURER
DEPARTMENT OF ECE.

SYNOPSIS

The agricultural land mass is more than just being a feeding sourcing in today's World. Indian economy is highly dependent on agricultural productivity. Therefore, in the field of agriculture, detection of disease in plants plays an important role. Also, the famers do not have proper facilities or even idea that they can contact to expert.

In this project, an automated irrigation sensor was designed and implemented to use in agricultural crops. An Android App was developed in the Smartphone to operate directly the computing and connectivity components, such as digital camera and the Bluetooth network (HC-05). The mobile App wakes up the Smartphone, activating the device with user-defined parameters. Then, the built-in camera takes a picture of the soil through an anti-reflective glass window and an RGB to grey process is achieved to estimate the ratio between wet and dry areas of the image After the Bluetooth connection is enabled, the ratio is transmitted via a router node to a gateway for control an irrigation water pump. The IR sensor is used to friends monitoring. The irrigation and friends monitoring process is done by the ARDUINO UNO. Finally, the App sets the Smartphone into the sleep mode to preserve its energy. The sensor is powered by rechargeable batteries, charged by a photovoltaic panel. The experimental results show that the use of Smartphone as an irrigation sensor could become a practical tool for agriculture.

5. BLOCK DIAGRAM

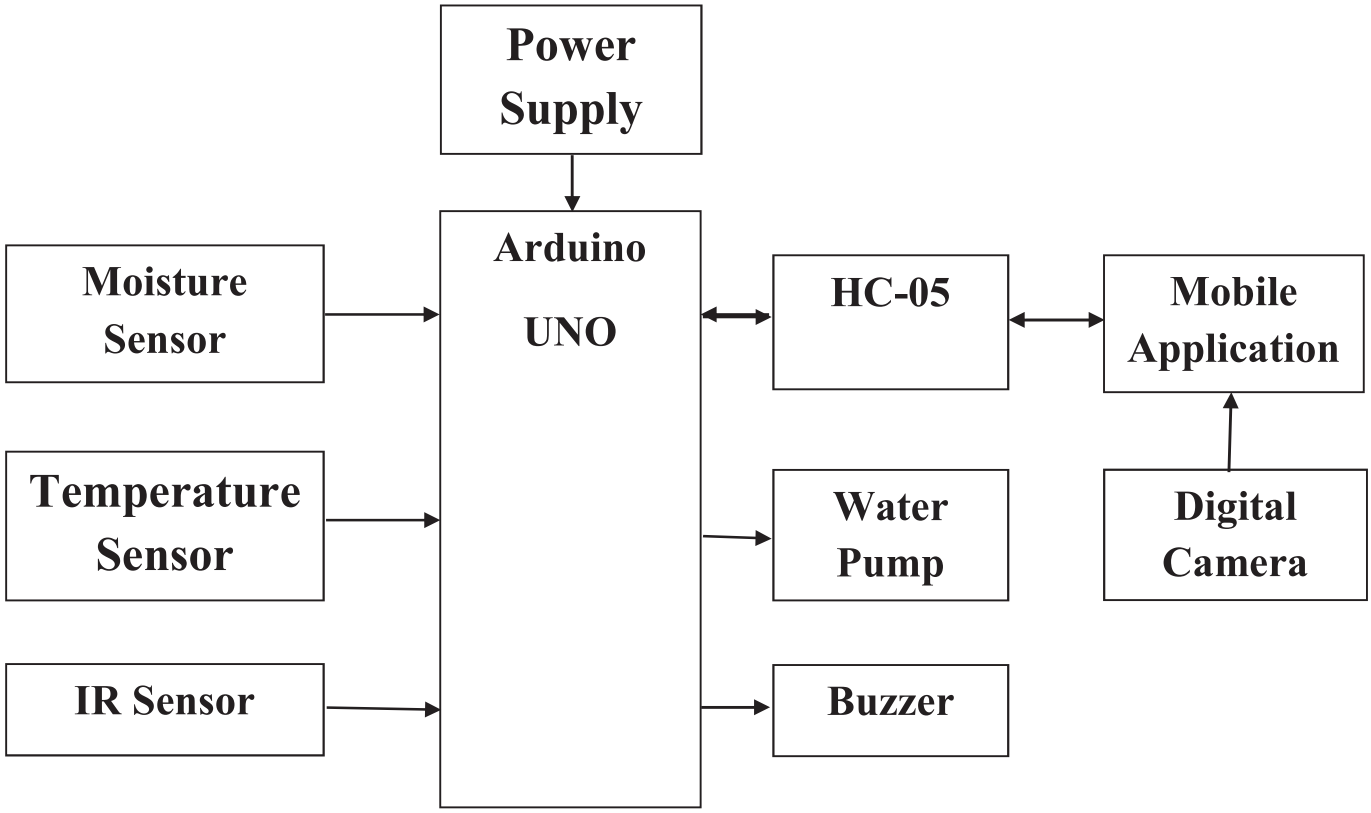


TABLE OF CONTENTS

1. ABSTRACT

The agricultural land mass is more than just being a feeding sourcing in today's World. Indian economy is highly dependent on agricultural productivity. Therefore, in the field of agriculture, detection of disease in plants plays an important role. Also, the famers do not have proper facilities or even idea that they can contact to expert. An automated irrigation sensor was designed and implemented to use in agricultural crops. The sensor uses a smartphone to capture and process digital images of the soil nearby the root zone of the crop, and estimates optically the water contents. The sensor is confined in a chamber under controlled illumination and buried at the root level of the plants. An Android App was developed in the smartphone to operate directly the computing and connectivity components, such as the digital camera and the BLUETOOTH network. The mobile App wakes up the smartphone, activating the device with user-defined parameters. Then, the built-in camera takes a picture of the Leaves through an antireflective glass window and an RGB to gray process is achieved to estimate the ratio of RGB to identifying the diseases on the leaves. After the BLUETOOTH connection is enabled, the ratio is transmitted via a router node to a gateway for control an irrigation water pump. Finally, the App sets the smartphone into the sleep mode to preserve its energy. The sensor is powered by rechargeable batteries, charged by a photovoltaic panel. The smartphone irrigation sensor was evaluated. The experimental results show that the use of smartphones as an irrigation sensor could become a practical tool for agriculture.

2. INTRODUCTION

Our Indian economy is depended on agricultural productivity to a large extend. Thus detection of disease in plant is very important. A plant disease is the occurrence of change in normal functioning of the plant or its part that can affect the productivity. The process is done with the help of image processing. Image processing is used for determine the disease in the affected area and measuring affected area of disease. The agricultural land mass is more than just being a feeding sourcing in today's world. Indian economy is highly dependent of agricultural productivity. Therefore in field of agriculture, detection of disease in plants plays an important role. To detect a plant disease in very initial stage, use of automatic disease detection technique is beneficial. For instance a disease named little leaf disease is a hazardous disease found in pine trees in United States. The affected tree has a stunted growth and dies within 6 years. Its impact is found in Alabama, Georgia parts of Southern US. In such scenarios early detection could have been fruitful.

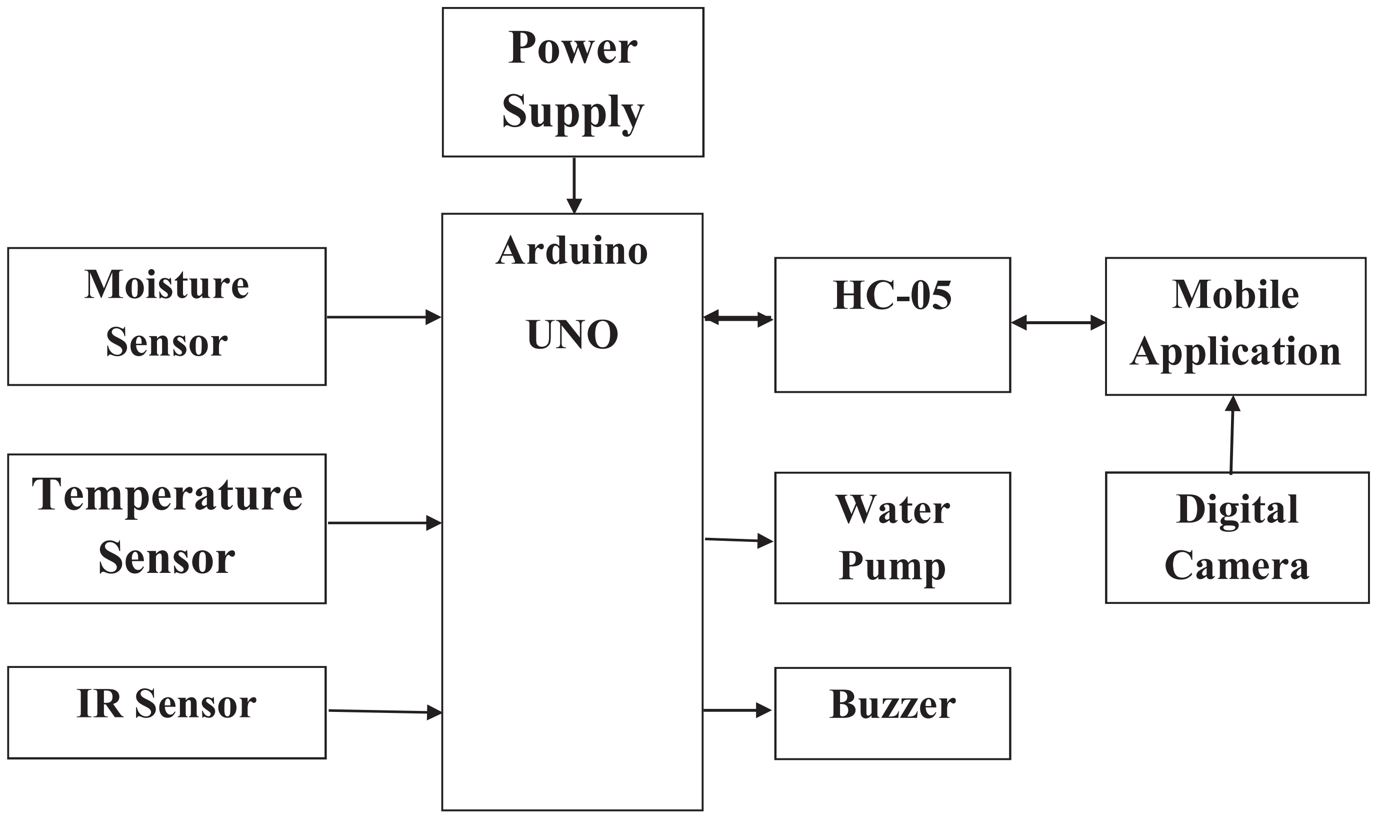
3. EXISTING SYSTEM

The existing system of the project uses SVM classification method. It only detects the disease. The diseases diagnosed with the help of existing system include Blast, Brown spot and Narrow brown spot.

4. PROPOSED SYSTEM

In this proposed system, the diseases are diagnosed using Naïve Bayes algorithm. Along with that, we analyse the growth time period required for the plant with and without the particular disease. The percentage of affected area is shown and according to that, the list of fertilizers as well as organic methods for treatment of the disease is recommended as part of the result. The list of diseases we concentrate in the proposed system are Black horse riding, Bacterial leaf steak and Brown spot.

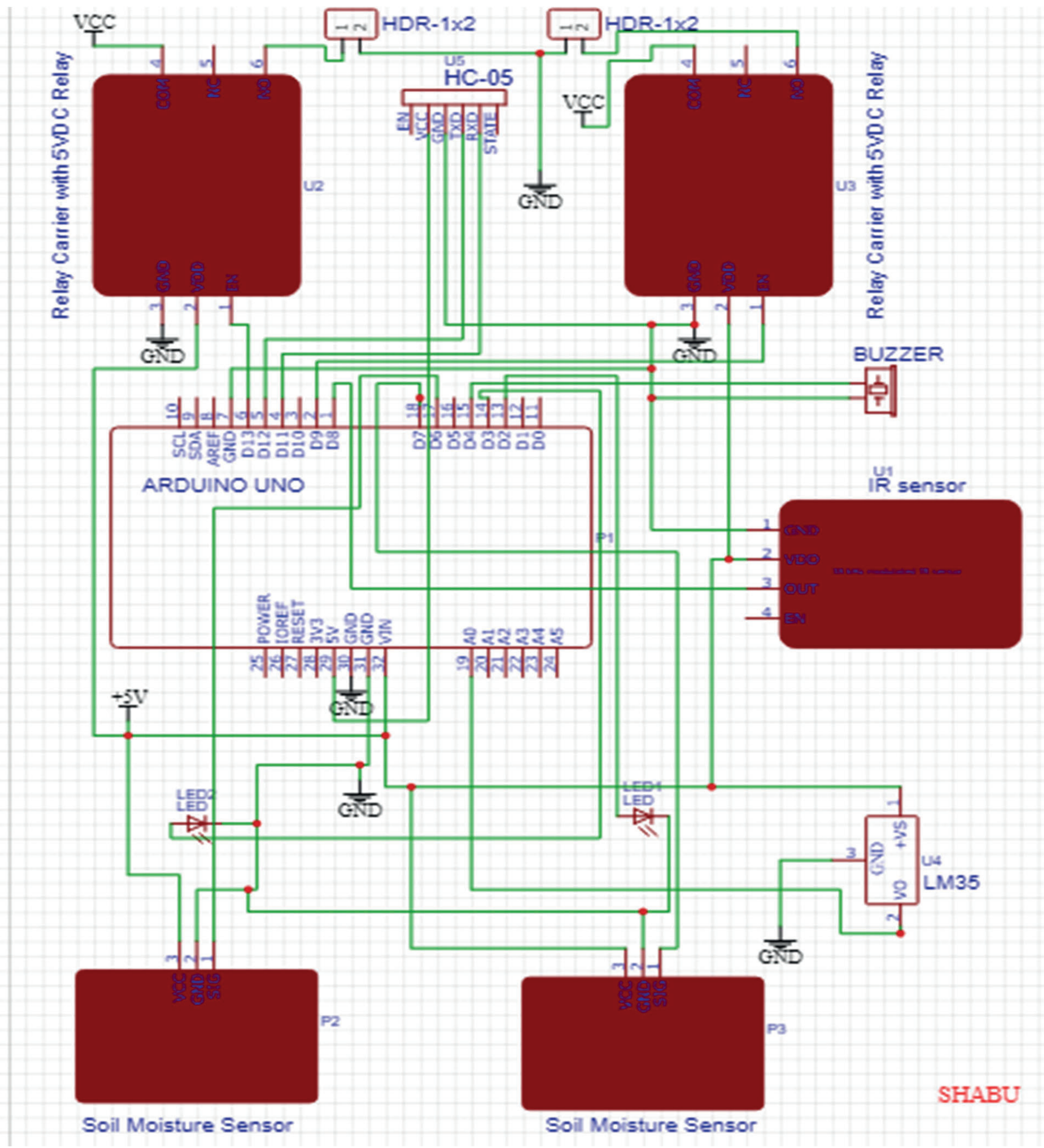
5. BLOCK DIAGRAM



6. BLOCK DIAGRAM DESCRIPTION

The Arduino is used as controller in this project. Soil Moisture Sensor and IR Sensor is connected to the Arduino's digital input pins. And the temperature sensor is connected to the Arduino's analog input pin. Then the motor pump driver is connected to the digital output pin of the Arduino, Also the buzzer is connected to Arduino's digital output pin. And The HC-05 Bluetooth module connected to Arduino's RX and TX pins.

7. CIRCUIT DIAGARAM



8. COMPONENTS

- Arduino UNO
- Soil Moisture Sensor
- Temperature Sensor
- IR Sensor
- HC-05
- Motor pump
- Buzzer
- Power supply

9. COMPONENTS DESCRIPTION

9.1. ARDUINO UNO

The Arduino Uno is a microcontroller board based on the ATmega328 (datasheet). It has 14 digital input/output pins (of which 6 can be used as PWM outputs), 6 analog inputs, a 16 MHz ceramic resonator, a USB connection, a power jack, an ICSP header, and a reset button. It contains everything needed to support the microcontroller; simply connect it to a computer with a USB cable or power it with a AC-to-DC adapter or battery to get started. The Uno differs from all preceding boards in that it does not use the FTDI USB-to-serial driver chip. Instead, it features the Atmega16U2 (Atmega8U2 up to version R2) programmed as a USB-to-serial converter. Revision 2 of the Uno board has a resistor pulling the 8U2 HWB line to ground, making it easier to put into DFU mode. Revision 3 of the board has the following new features:

- 1.0 pinout: added SDA and SCL pins that are near to the AREF pin and two other new pins placed near to the RESET pin, the IOREF that allow the shields to adapt to the voltage provided from the board. In future, shields will be compatible both with the board that use the AVR, which operate with 5V and with the Arduino Due that operate with 3.3V. The second one is a not connected pin, that is reserved for future purposes.
- Stronger RESET circuit.
- Atmega 16U2 replace the 8U2. "Uno" means one in Italian and is named to mark the upcoming release of Arduino 1.0. The Uno and version 1.0 will be the reference versions of Arduino, moving forward. The Uno is the latest in a series of USB Arduino boards, and the reference model for the Arduino platform; for a comparison with previous versions, see the index of Arduino boards.

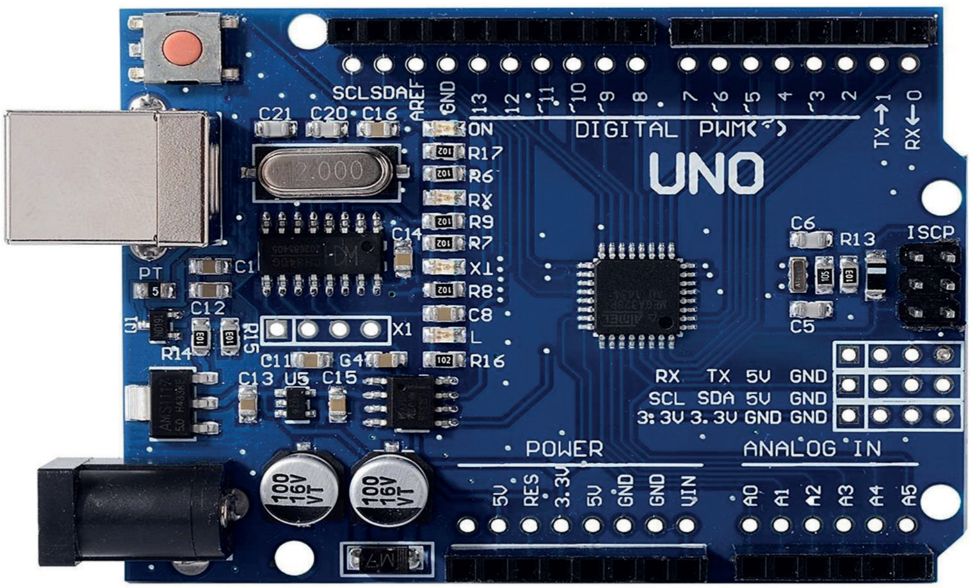


Fig.1 Arduino UNO

Summary

Microcontroller	: ATmega328
Operating Voltage	: 5V
Input Voltage (recommended)	: 7-12V
Input Voltage (limits)	: 6-20V
Digital I/O Pins	: 14 (of which 6 provide PWM output)
Analog Input Pins	: 6
DC Current per I/O Pin	: 40 mA
DC Current for 3.3V Pin	: 50 mA

Flash Memory 32 KB (ATmega328) of which 0.5 KB used by bootloader

SRAM 2 KB (ATmega328)

EEPROM 1 KB (ATmega328)

Clock Speed 16 MHz

Power

The Arduino Uno can be powered via the USB connection or with an external power supply. The power source is selected automatically. External (non-USB) power can come either from an AC-to-DC adapter (wall-wart) or battery. The adapter can be connected by plugging a 2.1mm centre-positive plug into the board's power jack. Leads from a battery can be inserted in the Gnd and Vin pin headers of the POWER connector. The board can operate on an external supply of 6 to 20 volts. If supplied with less than 7V, however, the 5V pin may supply less than five volts and the board may be unstable. If using more than 12V, the voltage regulator may overheat and damage the board. The recommended range is 7 to 12 volts.

The power pins are as follows:-

- **VIN**, The input voltage to the Arduino board when it's using an external power source (as opposed to 5 volts from the USB connection or other regulated power source). You can supply voltage through this pin, or, if supplying voltage via the power jack, access it through this pin.

- **5V,** This pin outputs a regulated 5V from the regulator on the board. The board can be supplied with power either from the DC power jack (7 - 12V), the USB connector (5V), or the VIN pin of the board (7-12V). Supplying voltage via the 5V or 3.3V pins bypasses the regulator, and can damage your board. We don't advise it.

- **3.3V,** A 3.3 volt supply generated by the on-board regulator. Maximum current draw is 50 mA.

- **GND.** Ground pins.

Memory

The ATmega328 has 32 KB (with 0.5 KB used for the bootloader). It also has 2 KB of SRAM and 1 KB of EEPROM.

Input and Output

Each of the 14 digital pins on the Uno can be used as an input or output, using pinMode(), digitalWrite(), and digitalRead() functions. They operate at 5 volts. Each pin can provide or receive a maximum of 40 mA and has an internal pull-up resistor (disconnected by default) of 20-50 kOhms. In addition, some pins have specialized functions:

- **Serial: 0 (RX) and 1 (TX).** Used to receive (RX) and transmit (TX) TTL serial data. These pins are connected to the corresponding pins of the ATmega8U2 USB-to-TTL Serial chip.
- **External Interrupts: 2 and 3**. These pins can be configured to trigger an interrupt on a low value, a rising or falling edge, or a change in value. See the attachInterrupt() function for details.
- **PWM: 3, 5, 6, 9, 10, and 11.** Provide 8-bit PWM output with the analogWrite() function.
- **SPI: 10 (SS), 11 (MOSI), 12 (MISO), 13 (SCK).** These pins support SPI communication using the SPI library.
- **LED: 13.** There is a built-in LED connected to digital pin 13. When the pin is HIGH value, the LED is on, when the pin is LOW, it's off.

The Uno has 6 analog inputs, labeled A0 through A5, each of which provide 10 bits of resolution (i.e. 1024 different values). By default they measure from ground to 5 volts, though is it possible to change the upper end of their range using the AREF pin and the analogReference() function. Additionally, some pins have specialized functionality:

- **TWI**: A4 or SDA pin and A5 or SCL pin. Support TWI communication using the Wire library.

There are a couple of other pins on the board:

- **AREF**: Reference voltage for the analog inputs. Used with analogReference().

- **RESET:** Bring this line LOW to reset the microcontroller. Typically used to add a reset button to shields which block the one on the board.

Communication

The Arduino Uno has a number of facilities for communicating with a computer, another Arduino, or other microcontrollers. The ATmega328 provides UART TTL (5V) serial communication, which is available on digital pins 0 (RX) and 1 (TX). An ATmega16U2 on the board channels this serial communication over USB and appears as a virtual com port to software on the computer. The '16U2 firmware uses the standard USB COM drivers, and no external driver is needed. However, on Windows, a .inf file is required. The Arduino software includes a serial monitor which allows simple textual data to be sent to and from the Arduino board. The RX and TX LEDs on the board will flash when data is being transmitted via the USB-to-serial chip and USB connection to the computer (but not for serial communication on pins 0 and 1).

A SoftwareSerial library allows for serial communication on any of the Uno's digital pins. The ATmega328 also supports I2C (TWI) and SPI communication. The Arduino software includes a Wire library to simplify use of the I2C bus; see the documentation for details. For SPI communication, use the SPI library.

Programming

The Arduino Uno can be programmed with the Arduino software (download). Select "Arduino Uno from the Tools > Board menu (according to the microcontroller on your board).

The ATmega328 on the Arduino Uno comes preburned with a bootloader that allows you to upload new code to it without the use of an external hardware programmer. It communicates using the original STK500 protocol (reference, C header files).

You can also bypass the bootloader and program the microcontroller through the ICSP (In-Circuit Serial Programming) header; see these instructions for details. The ATmega16U2 (or 8U2 in the rev1 and rev2 boards) firmware source code is available. The ATmega16U2/8U2 is loaded with a DFU bootloader, which can be activated by:

- On Rev1 boards: connecting the solder jumper on the back of the board (near the map of Italy) and then resetting the 8U2.
- On Rev2 or later boards: there is a resistor that pulling the 8U2/16U2 HWB line to ground, making it easier to put into DFU mode.

You can then use Atmel's FLIP software (Windows) or the DFU programmer (Mac OS X and Linux) to load a new firmware. Or you can use the ISP header with an external programmer (overwriting the DFU bootloader).

Automatic (Software) Reset

Rather than requiring a physical press of the reset button before an upload, the Arduino Uno is designed in a way that allows it to be reset by software running on a connected computer. One of the hardware flow control lines (DTR) of the ATmega8U2/16U2 is connected to the reset line of the ATmega328 via a 100 Nano farad capacitor. When this line is asserted (taken low), the reset line drops long enough to reset the chip. The Arduino software uses this capability to allow you to upload code by simply pressing the upload button in the Arduino environment. This means that the bootloader can have a shorter timeout, as the lowering of DTR can be well-coordinated with the start of the upload. This setup has other implications. When the Uno is connected to either a computer running Mac OS X or Linux, it resets each time a connection is made to it from software (via USB). For the following halfsecond or so, the bootloader is running on the Uno. While it is programmed to ignore malformed data (i.e. anything besides an upload of new code), it will intercept the first few bytes of data sent to the board after a connection is opened. If a sketch running on the board receives one-time configuration or other data when it first starts, make sure that the software with which it communicates waits a second after opening the connection and before sending this data.

The Uno contains a trace that can be cut to disable the auto-reset. The pads on either side of the trace can be soldered together to re-enable it. It's labeled "RESET-EN". You may also be able to disable the auto-reset by connecting a 110 ohm resistor from 5V to the reset line; see this forum thread for details.

USB Overcurrent Protection

The Arduino Uno has a resettable polyfuse that protects your computer's USB ports from shorts and overcurrent. Although most computers provide their own internal protection, the fuse provides an extra layer of protection. If more than 500 mA is applied to the USB port, the fuse will automatically break the connection until the short or overload is removed.

9.2. SOIL MOISTURE SENSOR

Soil moisture sensors measure the volumetric water content in soil. Since the direct gravimetric measurement of free soil moisture requires removing, drying, and weighting of a sample, soil moisture sensors measure the volumetric water content indirectly by using some other property of the soil, such as electrical resistance, dielectric constant, or interaction with neutrons, as a proxy for the moisture content. The relation between the measured property and soil moisture must be calibrated and may vary depending on environmental factors such as soil type, temperature, or electric conductivity. Reflected microwave radiation is affected by the soil moisture and is used for remote sensing in hydrology and agriculture. Portable probe instruments can be used by farmers or gardeners.

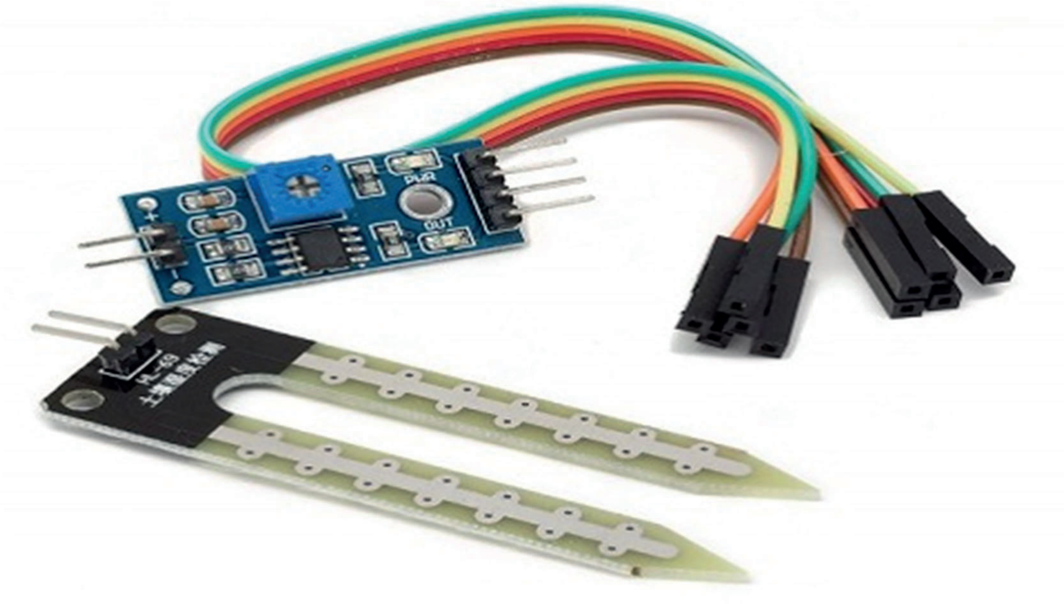


Fig.2 Soil Moisture Sensor

In the context of water management for irrigation, measuring and monitoring soil water status is an essential component of best management practices (BMPs) to improve the sustainability of agriculture.

Water content in the soil can be directly determined using the difference in weight before and after drying a soil sample. This direct technique is usually referred to as the thermo-gravimetric method (or simply gravimetric) when expressing water content as weight of water over weight of dry soil, GWC[$lb^3 lb^{-3}$] (i.e., the ratio of the mass of water present in a sample to the mass of the soil sample after it has been oven-dried (100–110 °C) to a constant weight). On the other hand, the thermo-volumetric method (or simply volumetric*)* gives the water content as volume of water in a volume of undisturbed soil VWC [$ft^3 ft^{-3}$] (i.e., volume of water related to the volume of an oven-dried undisturbed sample (soil core)). Although these direct methods are accurate (±0.01 $ft^3 ft^{-3}$) and inexpensive, they are destructive, slow (2 days minimum), time-consuming and do not allow for making repetitions in the same location. Alternatively, many indirect methods are available for monitoring soil water content. These methods estimate soil moisture by a calibrated relationship with some other measurable variable. The suitability of each method depends on several issues like cost, accuracy, response time, installation, management and durability.

Depending on the quantity measured, indirect techniques are first classified into volumetric and tensiometric methods. While the former gives volumetric soil moisture, the latter yields soil suction or water potential (i.e., tension exerted by capillarity). Both quantities are related through the soil water characteristic curve specific to a given soil.

Most soil moisture sensors are designed to estimate soil volumetric water content based on the dielectric constant (soil bulk permittivity) of the soil. The dielectric constant can be thought of as the soil's ability to transmit electricity. The dielectric constant of soil increases as the water content of the soil increases. This response is due to the fact that the dielectric constant of water is much larger than the other soil components, including air. Thus, measurement of the dielectric constant gives a predictable estimation of water content.

Bypass type soil moisture irrigation controllers use water content information from the sensor to either allow or bypass scheduled irrigation cycles on the irrigation timer (Figures 1 and 2). The SMS controller has an adjustable threshold setting and, if the soil water content exceeds that setting, the event is bypassed. The soil water content threshold is set by the user. Another type of control technique with SMS devices is “on-demand” where the controller initiates irrigation at a low threshold and terminates irrigation at a high threshold.

9.3. TEMPERTURE SENSOR

The LM35 series are precision integrated-circuit temperature devices with an output voltage linearly proportional to the Centigrade temperature. The LM35 device has an advantage over linear temperature sensors calibrated in Kelvin, as the user is not required to subtract a large constant voltage from the output to obtain convenient Centigrade scaling. The LM35 device does not require any external calibration or trimming to provide typical accuracies of ±¼°C at room temperature and ±¾°C, over a full −55°C to 150°C temperature range. Lower cost is assured by trimming and calibration at the wafer level. The low-output impedance, linear output, and precise inherent calibration of the LM35 device makes interfacing to readout or control circuitry especially easy. The device is used with single power supplies, or with plus and minus supplies. As the LM35 device draws only 60 µA from the supply, it has very low self-heating of less than 0.1°C in still air. The LM35 device is rated to operate over a −55°C to 150°C temperature range, while the LM35C device is rated for a −40°C to 110°C range (−10° with improved accuracy). The LM35-series devices are available packaged in hermetic TO transistor packages, while the LM35C, LM35CA, and LM35D devices are available in the plastic TO-92 transistor package. The LM35D device is available in an 8-lead surface-mount small-outline package and a plastic TO-220 package.

Features:-

• Calibrated Directly in Celsius (Centigrade)

• Linear + 10-mV/°C Scale Factor

• 0.5°C Ensured Accuracy (at 25°C)

• Rated for Full −55°C to 150°C Range

• Suitable for Remote Applications

• Low-Cost Due to Wafer-Level Trimming

• Operates From 4 V to 30 V

• Less Than 60-μA Current Drain

• Low Self-Heating, 0.08°C in Still Air

• Non-Linearity Only ±¼°C Typical

• Low-Impedance Output, 0.1 Ω for 1-mA Load

Applications:-

• Power Supplies

• Battery Management

• HVAC

• Appliances

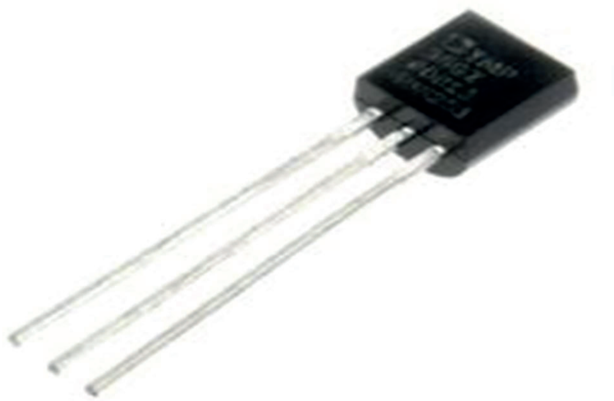


Fig.3 LM35

9.4. IR SENSOR

Infrared technology addresses a wide variety of wireless applications. The main areas are sensing and remote controls. In the electromagnetic spectrum, the infrared portion is divided into three regions: near infrared region, mid infrared region and far infrared region.

The wavelengths of these regions and their applications are shown below.

- Near infrared region —700 nm to 1400 nm — IR sensors, fibre optic
- Mid infrared region — 1400 nm to 3000 nm — Heat sensing
- Far infrared region — 3000 nm to 1 mm — Thermal imaging

The frequency range of infrared is higher than microwave and lesser than visible light.

For optical sensing and optical communication, photo optics technologies are used in the near infrared region as the light is less complex than RF when implemented as a source of signal. Optical wireless communication is done with IR data transmission for short range applications.

The working of any Infrared sensor is governed by three laws: Planck's Radiation law, Stephen – Boltzmann law and Wien's Displacement law.

The basic concept of an Infrared Sensor which is used as Obstacle detector is to transmit an infrared signal, this infrared signal bounces from the surface of an object and the signal is received at the infrared receiver.

There are five basic elements used in a typical infrared detection system: an infrared source, a transmission medium, optical component, infrared detectors or receivers and signal processing. Infrared lasers and Infrared LED's of specific wavelength can be used as infrared sources. The three main types of media used for infrared transmission are vacuum, atmosphere and optical fibres. Optical components are used to focus the infrared radiation or to limit the spectral response.

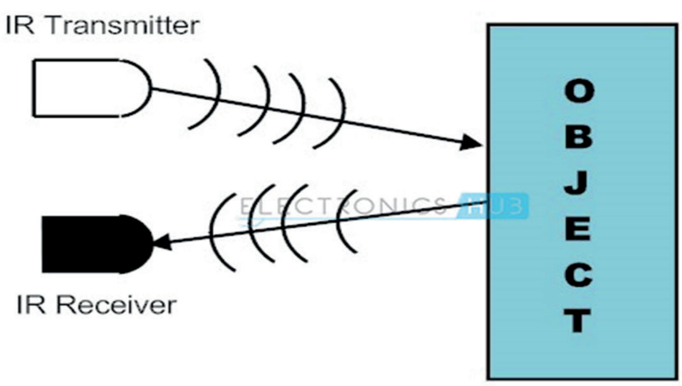
Optical lenses made of Quartz, Germanium and Silicon are used to focus the infrared radiation. Infrared receivers can be photodiodes, phototransistors etc. some important specifications of infrared receivers are photosensitivity, directivity and noise equivalent power. Signal processing is done by amplifiers as the output of infrared detector is very small.

Types of IR Sensors

Infrared sensors can be passive or active. Passive infrared sensors are basically Infrared detectors. Passive infrared sensors do not use any infrared source and detects energy emitted by obstacles in the field of view. They are of two types: quantum and thermal. Thermal infrared sensors use infrared energy as the source of heat and are independent of wavelength. Thermocouples, piezoelectric\ detectors and bolometers are the common types of thermal infrared detectors.

Quantum type infrared detectors offer higher detection performance and are faster than thermal type infrared detectors. The photosensitivity of quantum type detectors is wavelength dependent. Quantum type detectors are further classified into two types: intrinsic and extrinsic types. Intrinsic type quantum detectors are photoconductive cells and photovoltaic cells.

Active infrared sensors consist of two elements: infrared source and infrared detector. Infrared sources include an LED or infrared laser diode. Infrared detectors include photodiodes or phototransistors. The energy emitted by the infrared source is reflected by an object and falls on the infrared detector.



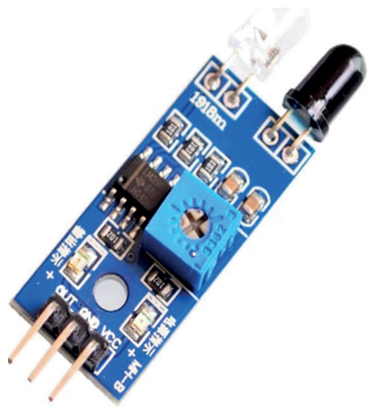


Fig.4 IR Sensor

9.5. HC-05

The **HC-05** is a very cool module which can add two-way (full-duplex) wireless functionality to your projects. You can use this module to communicate between two microcontrollers like Arduino or communicate with any device with Bluetooth functionality like a Phone or Laptop. There are many android applications that are already available which makes this process a lot easier. The module communicates with the help of USART at 9600 baud rate hence it is easy to interface with any microcontroller that supports USART. We can also configure the default values of the module by using the command mode. So if you looking for a Wireless module that could transfer data from your computer or mobile phone to microcontroller or vice versa then this module might be the right choice for you. However do not expect this module to transfer multimedia like photos or songs; you might have to look into the CSR8645 module for that.

The **HC-05** has two operating modes, one is the Data mode in which it can send and receive data from other Bluetooth devices and the other is the AT Command mode where the default device settings can be changed. We can operate the device in either of these two modes by using the key pin as explained in the pin description.

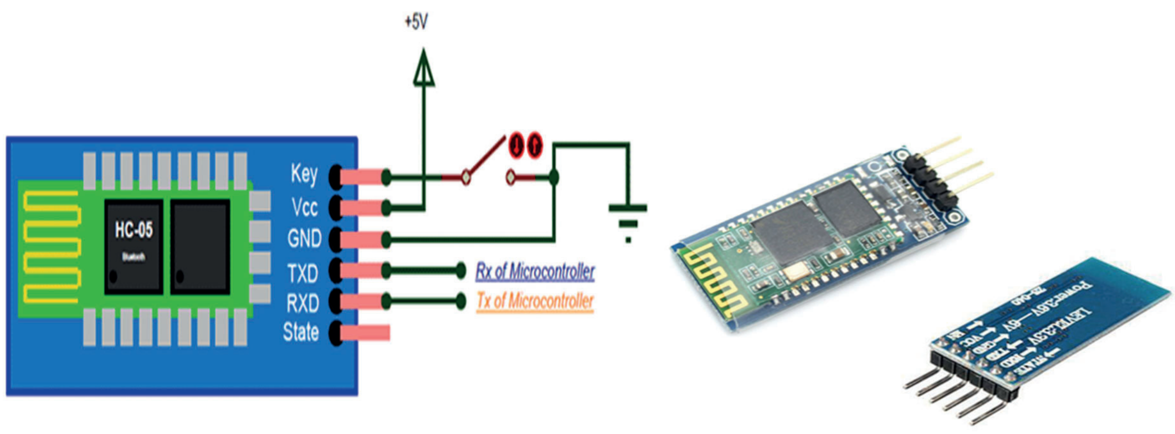


Fig.5 HC-05

It is very easy to pair the HC-05 module with microcontrollers because it operates using the Serial Port Protocol (SPP). Simply power the module with +5V and connect the Rx pin of the module to the Tx of MCU and Tx pin of module to Rx of MCU as shown in the figure below

During power up the key pin can be grounded to enter into Command mode, if left free it will by default enter into the data mode. As soon as the module is powered you should be able to discover the Bluetooth device as “HC-05” then connect with it using the default password 1234 and start communicating with it. The name password and other default parameters can be changed by entering into the

HC-05 Default Settings

- Default Bluetooth Name: “HC-05”
- Default Password: 1234 or 0000
- Default Communication: Slave
- Default Mode: Data Mode
- Data Mode Baud Rate: 9600, 8, N, 1
- Command Mode Baud Rate: 38400, 8, N, 1
- Default firmware: LINVOR

9.6. MOTOR PUMP

High quality made, CE passed, mini DC submersible water pump. It is built by brushless motor which is quieter and last longer. Easy to maintain with fish tank, SOLAR WATER PUMP, accelerating water circulation, miniature garden, indoor fountain, toys, and other water circulation system.

Fig.6 Motor Pump

Specification:-

- Input Voltage: 3.5-5V DC
- Power: 1W
- H.max.: 0.4-2.0M
- Q.max.: 200L/H
- Size: 38 * 34 * 27mm
- Wire Length: 140cm

Applications:-

- Water Vending Machines
- Mini water Fountain
- Coffee Vending Machines
- Juice vending Machines

9.7. BUZZER

A buzzer or beeper is an audio signalling device, which may be mechanical, electromechanical, or piezoelectric. Typical uses of buzzers and beepers include alarm devices, timers and confirmation of user input such as a mouse click or keystroke.

Buzzer is an integrated structure of electronic transducers, DC power supply, widely used in computers, printers, copiers, alarms, electronic toys, automotive electronic equipment, telephones, timers and other electronic products for sound devices. Active buzzer 5V rated power can be directly connected to a continuous sound, this section dedicated sensor expansion module and the board in combination, can complete a simple circuit design, to "plug and play."

Fig.7 Buzzer

Features

- Black in colour
- With internal drive circuit
- Sealed structure
- Wave solder able and washable

Applications

- Computer and peripherals
- Communications equipment
- Portable equipment
- Automobile electronics
- POS system

9.8. POWER SUPPLY

IC 7805 is a 5V fixed three terminal positive voltage regulators IC. The IC has features such as safe operating area protection, thermal shut down, internal current limiting which makes the IC very rugged. Output currents up to 1A can be drawn from the IC provided that there is a proper heat sink.

Capacitor C1 filters it and 7805 regulates it to produce a steady 5Volt DC. The circuit schematic is given below.

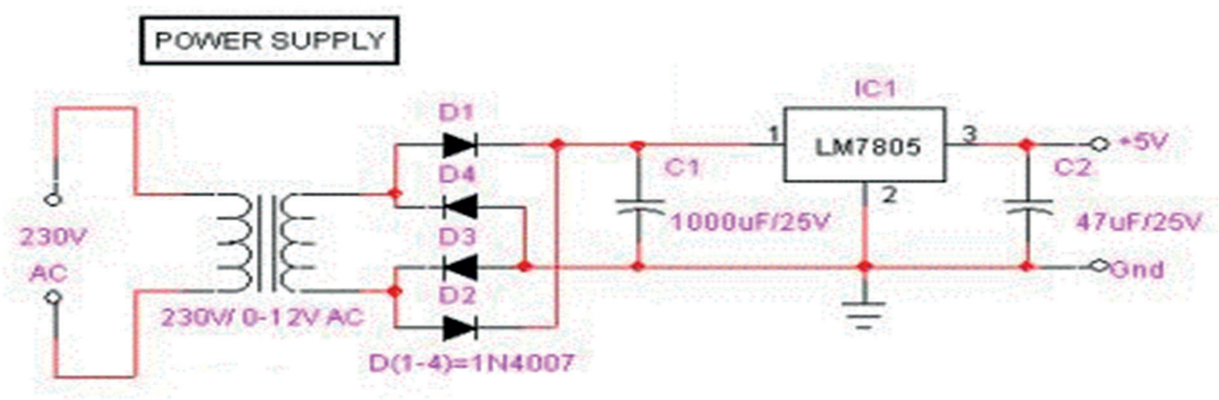


Fig.8 Power Supply

10. MOBILE APPLICATIONS

10.1. INTRODUCTION

Our Indian economy is depended on agricultural productivity to a large extend. Thus detection of disease in plant is very important. A plant disease is the occurrence of change in normal functioning of the plant or its part that can affect the productivity. The process is done with the help of image processing. Image processing is used for determine the disease in the affected area and measuring affected area of disease.

The steps involved in the identification of the paddy diseases are acquisition of image, pre-processing of image, segmentation of image, feature extraction, classification and growth prediction. The allowed types of Image formats are jpeg, png, gif, bmp.

After the initial step of image capturing, various processing techniques can be concerned with concrete problems of the image for performing the task.

Plant disease identification by visual way is more laborious task and at the same time, less accurate and can be done only in limited areas. Whereas if automatic detection technique is used it will take less efforts, less time and become more accurate. In plants, some general diseases seen are brown and yellow spots, early and late scorch, and others are fungal, viral and bacterial diseases. Image processing is used for measuring affected area of disease and to determine the difference in the colour of the affected area.

Image segmentation is the process of separating or grouping an image into different parts. There are currently many different ways of performing image segmentation, ranging from the simple thresholding method to advanced colour image segmentation methods. These parts normally correspond to something that humans can easily separate and view as individual objects. Computers have no means of intelligently recognizing objects, and so many different methods have been developed in order to segment images. The segmentation process is based on various features found in the image. This might be colour information, boundaries or segment of an image.

Image contrast is enhanced using techniques like histogram equalization and contrast adjustment. The classification is first done using the Minimum Distance Criterion with K-Mean Clustering and shows its efficiency with accuracy of 86.54%. Now the detection accuracy is improved to 95.71% by Naïve Bayes with proposed algorithm.

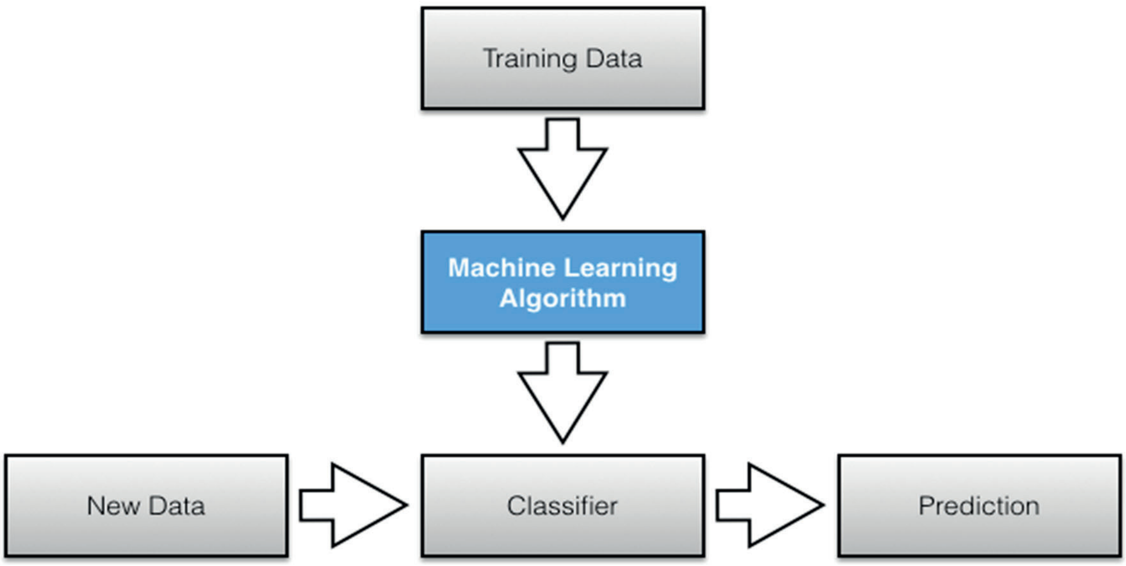


Fig.9 Naïve Bayes Classifier

Data from various censoring devices combined with powerful learning algorithms and domain knowledge led to many great inventions that we now take for granted in our everyday life: Internet queries via search engines like Google, text recognition at the post office, barcode scanners at the supermarket, the diagnosis of diseases, speech recognition by Siri or Google Now on our mobile phone, just to name a few.

One of the sub-fields of predictive modelling is supervised pattern classification; supervised pattern classification is the task of training a model based on labelled training data which then can be used to assign a pre-defined class label to new objects. One example that we will explore throughout this article is spam filtering via naive Bayes classifiers in order to predict whether a new text message can be categorized as spam or not-spam. Naive Bayes classifiers, a family of classifiers that are based on the popular Bayes' probability theorem, are known for creating simple yet well performing models, especially in the fields of document classification and disease prediction.

10.2. LIST OF DISEASES

1. Black Horse Riding

Symptoms:

Most of the patches caused by this disease are a result of infection by pathogenic fungi. Once into the leaf, the fungi continue to grow and leaf tissue is destroyed. Resulting spots vary in size from that of a pinhead to spots that encompass the entire leaf. Dead areas on the leaves are usually brown, black, tan or reddish in colour. Occasionally the necrotic areas have a red or purple border. Partial to complete defoliation may occur under favourable conditions for the causal fungus.

Fig 10.2.1 Black Horse Riding Disease

Fertilizer:

You can use a chemical fungicide or any number of organic options such as Copper, Lime Sulphur, Neem Oil, Potassium or Ammonium Bicarbonate, Sulphur.

2. Brown Spot

Symptoms:

Infected seedlings have small, circular or oval, brown lesions, which may girdle the coleoptile and cause distortion of the primary and secondary leaves Infected seedlings become stunted or die Young or underdeveloped

Fig 10.2.2 Brown Spot Disease

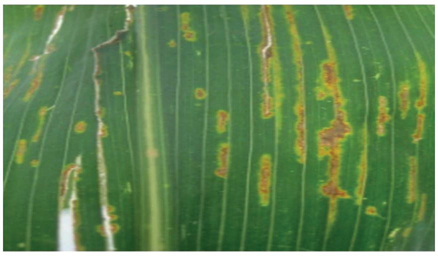
Fertilizer:

Fungicides like triadimefon can help in the reduction of this disease.

3. Bacterial Leaf Streak

Symptoms:

Symptoms initially appear as small, water-soaked, linear lesions between leaf veins. These streaks are initially dark green and later become light brown to yellowish gray. The lesions are translucent when held against the light. Entire leaves may become brown and die when the disease is very severe.



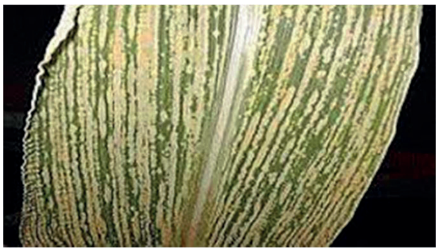


Fig 10.2.3 Bacterial Leaf Streak

Fertilizer:

Apply copper based fungicide on leaves or pesticide named Epsom salt can help in controlling the disease.

10.3. APPLICATION DESCRIPTION

10.3.1. PROBLEM DEFINITION

A system for identifying leaf diseases like Black horse riding, Brown spot and Bacterial leaf steak and its percentage is detected. It is mainly based on the android application using Naïve Bayes algorithm. This paper evaluates the techniques in digital image processing for detecting, diagnosing, recognizing of crop leaf diseases is used for automatically the disease for more accuracy than prediction through MATLAB using SVM clustering algorithm.

10.3.2. SYSTEM ARCHITECTURE

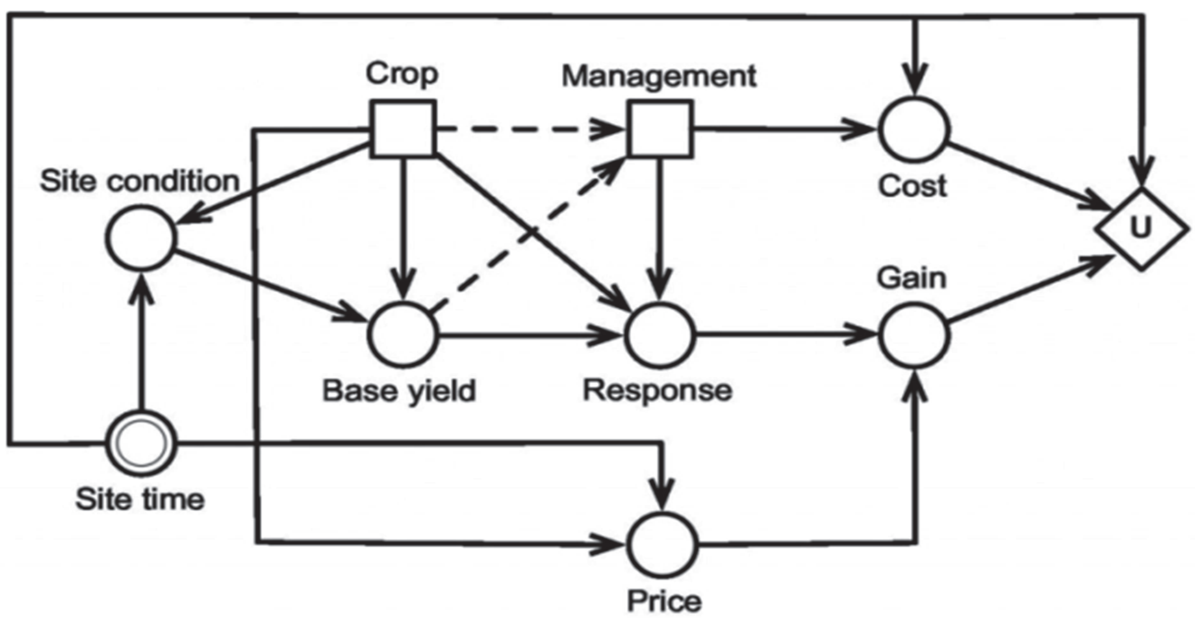


Fig 10.3.1 System Architecture

The users or nodes involved in our projects are Sender, Intermediate and Receiver. In order to send file, the sender has to find out the list of nodes which are connected with the sender. From that available list he can choose receiver. Then the sender has to analyse the performance of each and every node which is connected with the sender. The performance analysis list will return the priority based result so that sender can choose the intermediate to send the file. The

Intermediate will receive the file from sender then it will analyse the performance so that it can send data to another intermediate or receiver. In the receiver side, the receiver has to select the file path to receive the file from sender or intermediate. Then the receiver can view the file received file.

10.3.3. MODULE DESCRIPTION

Module 1: Image Acquisition

In this, the leaf colour images are acquired through the digital camera. The sample images are captured in the fields. Images are obtained at different times of a day. Moreover, leaves with varying canopy size were selected to increase the difficulty of the classification problem.

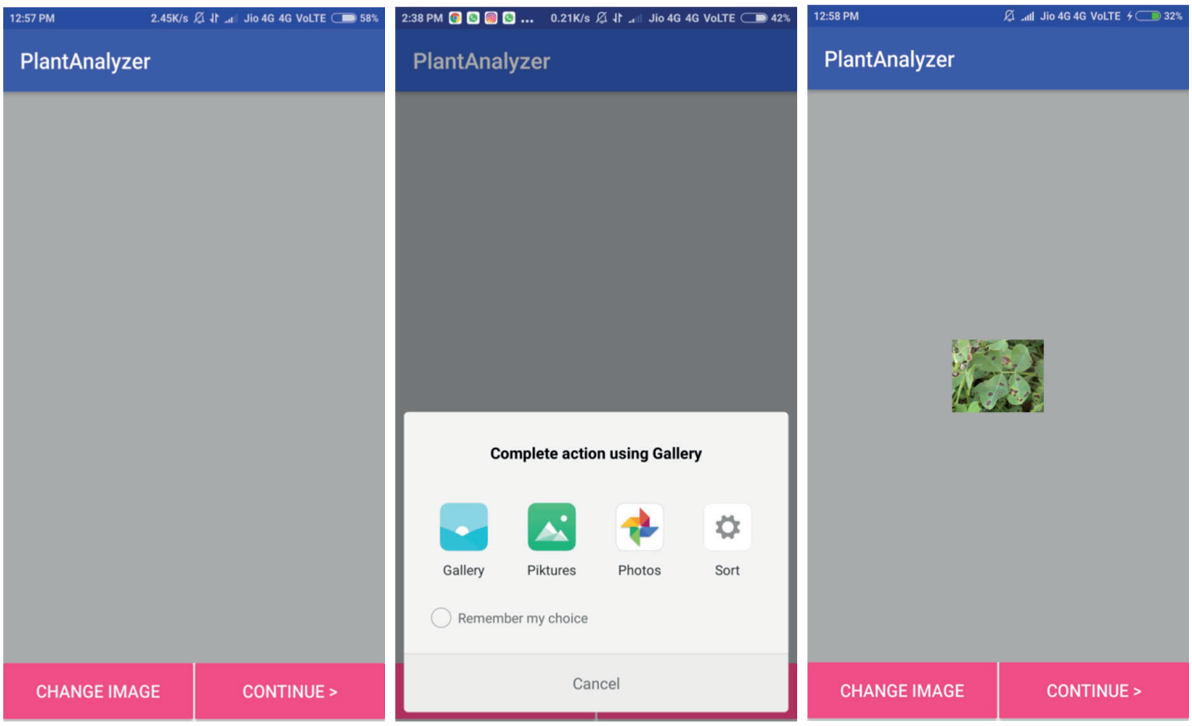


Fig 10.3.2 Image acquisition screenshots

Module 2: Image Pre-processing

Gray Transformation

The gray transformation processing is to turn the colour image to the gray image. The purpose of gray transformation is to reduce the amount of colour data in the image so as to speed up the following processing. The colour difference between plants and background in the colour images should be kept as well as possible in the gray image.

Bitmap Extraction

The bitmap is extracted where the required leaf part is separated from the background.

Histogram Analyses

The image thus obtained is analysed with the pixel intensity in order to form the histogram graph.

Image Segmentation

In this study, Colour based pattern method is used for the segmentation task. Other techniques for segmentation are thresholding - based segmentation, edge-based segmentation, Colour-based segmentation and watershed segmentation etc. All the segmentation steps are done depending on the binary image. Therefore, the grayscale image is converted into binary image from the pre-processing stage. Thresholding based on gray image is used to change the binary image.

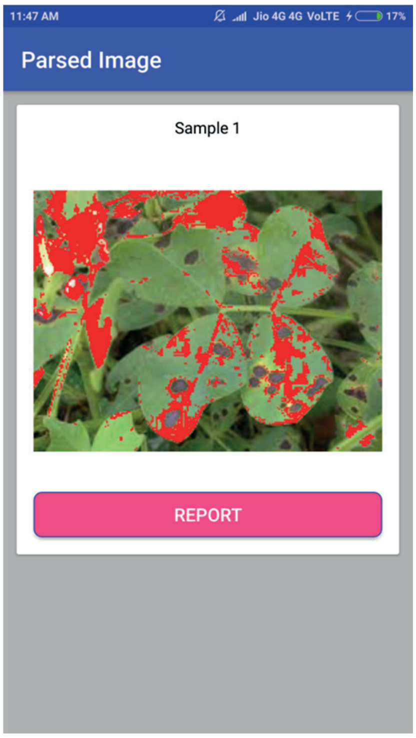


Fig 10.3.3 Image segmentation screenshots

Image Classification

In this, the image is classifying using Navy Bayes algorithm. Navy Bayes classifier is a type of linear classifier. This classifier is based on Bayes algorithm. This algorithm need small dataset to work compare to the other method.

Module 3: Growth Prediction

Growth is predicted with the analyses of size and shape of the leaf. The image of leaf is preceding a white background. This module also includes sun light condition, fertilizer and pesticide for the plant.

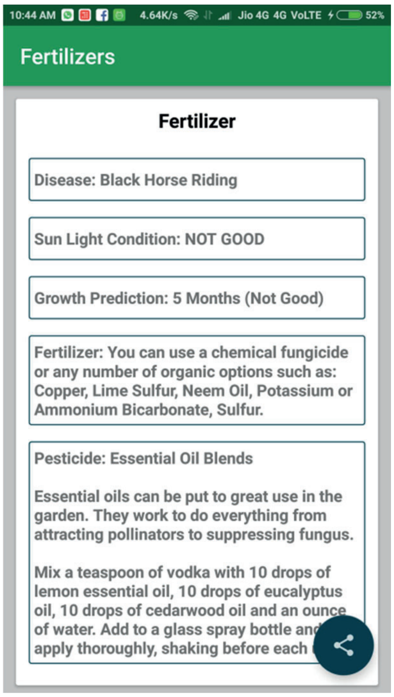


Fig 10.3.4 Growth prediction screenshots

10.4. RESULT

The accurately detection and classification of the plant disease is very important for the successful cultivation of crop and this can be done using image processing. This paper discussed various techniques to segment the disease part of the plant. This paper also discussed some Feature extraction and classification techniques to extract the features of infected leaf and the classification of plant diseases. The use of ANN methods for classification of disease in plants such as self-organizing feature map, back propagation algorithm, SVMs etc. can be efficiently used. From these methods, we can accurately identify and classify various plant diseases using image processing techniques.

The image is processed and the 3 mentioned diseases are checked.

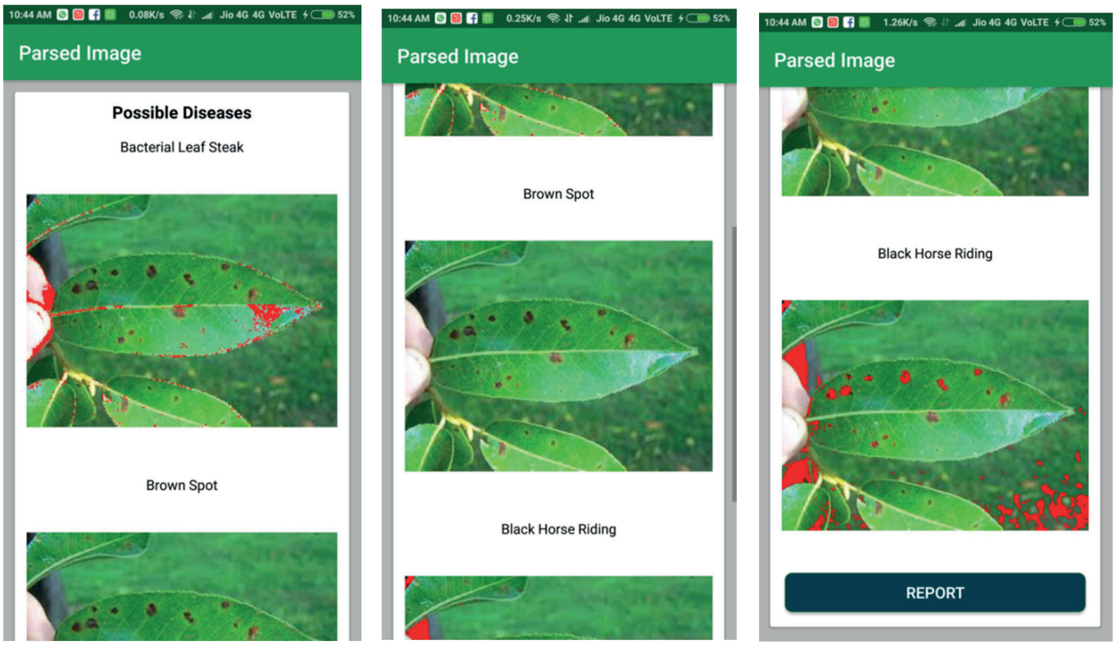


Fig 10.3.5 Disease type's screenshots

If the disease is found, then the percentage of disease is also shown along with the report image.

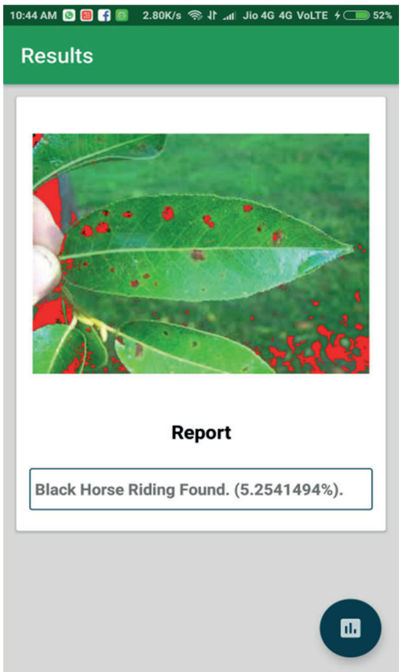


Fig 10.3.6 Result screenshot

The next step is the report generation where the factors like Disease name, Sun Light Condition, Growth Prediction, Fertilizer, and Pesticide are shown according to the percentage of disease. The growth prediction is set by the time taken (in months) for the plant to grow when it is affected by that disease considering the percentage of affected area.

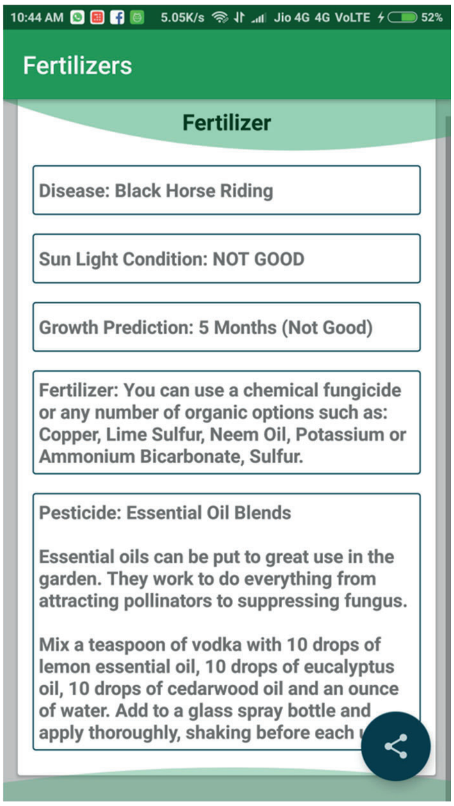


Fig 10.3.7 Report screenshot

The application also support the sharing of the report generated through WhatsApp, Shareit, Mail etc.

11. CODING

11.1. CODING FOR ARDUINO

```
#include <SoftwareSerial.h>
 SoftwareSerial mySerial(11, 12);

 const int sensor=A1; // Assigning analog pin A1 to variable 'sensor'
float tempc;  //variable to store temperature in degree Celsius
float vout;  //temporary variable to hold sensor reading
int moist = 6;
int moistsens = 7;
int irsens = 8;
int pump = 9;
int led= 2;
int led1= 3;
int buz= 4;
int ir,mois,moissen;

void setup() {

mySerial.begin(9600);
  Serial.begin (9600);
  pinMode(sensor,INPUT);
```

```
pinMode(moist, INPUT);
pinMode(moistsens, INPUT);
  pinMode(ir, INPUT);
   pinMode(pump, OUTPUT);
     pinMode(led, OUTPUT);
       pinMode(led1, OUTPUT);
       pinMode(buz, OUTPUT);
     delay(100);
}

void loop()
{
 // digitalWrite(led , LOW);
  digitalWrite(buz , LOW);



vout=analogRead(sensor);
vout=(vout*500)/1023;
tempc=vout;
Serial.print("in DegreeC=");
Serial.print("\t");
Serial.print(tempc);
```

```
Serial.println();
mySerial.print("in DegreeC=");
mySerial.print("\t");
mySerial.print(tempc);
mySerial.println();
delay(500);

ir= digitalRead(irsens);
   mois= digitalRead(moist);
     moissen= digitalRead(moistsens);
   if(ir==LOW)
   {
     digitalWrite(buz , HIGH);
     mySerial.println("!!!INTRUDDED ALERT!!!");
         delay(500);
   }



   if(mois==HIGH)
   {
      digitalWrite(buz , HIGH);delay(200);digitalWrite(buz , LOW);
     digitalWrite(pump , HIGH);     digitalWrite(led1 , HIGH);
```

```
 mySerial.println("Water Level low");
      delay(1500);
       // digitalWrite(pump , LOW);
  }
    else
  {
   digitalWrite(pump , LOW);      digitalWrite(led1 , LOW);
//      Serial.println("Water Level low");
//          delay(1500);
       // digitalWrite(pump , LOW);
  }

  if(moissen==HIGH)
  {
   digitalWrite(led , HIGH);
  // digitalWrite(pump , HIGH);
  // mySerial.println("Water Level-1 low");
      delay(1500);
       // digitalWrite(pump , LOW);
  }
    else
  {
```

```
      digitalWrite(led , LOW);
//      Serial.println("Water Level low");
          delay(1500);
           // digitalWrite(pump , LOW);
    }



    delay(100);
}
```

11.2. CODING FOR MOBILE APPLICATION

BITMAP Extraction

Image preprocessing: (Bitmap extraction and Grayscale conversion)

```
package Util;

import android.graphics.Bitmap;

import java.util.Collections;

import java.util.Comparator;

import java.util.HashMap;

import java.util.LinkedList;

import java.util.List;

import java.util.Map;


public class ImageColour {

    public ImageColour(Bitmap image) throws Exception {

        int height = image.getHeight();

        int width = image.getWidth();

        Map m = new HashMap();

        for (int i = 0; i < width; i++)

            for (int j = 0; j < height; j++) {
```

```
            int rgb = image.getPixel(i, j);
            int[] rgbArr = getRGBArr(rgb);
              if (!isGray(rgbArr)) {
                Integer counter = (Integer) m.get(rgb);
                if (counter == null)
                    counter = 0;
                counter++;
                m.put(rgb, counter);
            }
        }
    String colourHex = getMostCommonColour(m);
}

public static String getMostCommonColour(Map map) {
    List list = new LinkedList(map.entrySet());
    Collections.sort(list, new Comparator() {
        public int compare(Object o1, Object o2) {
            return ((Comparable) ((Map.Entry) (o1)).getValue())
                    .compareTo(((Map.Entry) (o2)).getValue());
```

```java
        }
    });
    Map.Entry me = (Map.Entry) list.get(list.size() - 1);
    int[] rgb = getRGBArr((Integer) me.getKey());
    return Integer.toHexString(rgb[0]) + " " +
Integer.toHexString(rgb[1]) + " " + Integer.toHexString(rgb[2]);
}

public static int[] getRGBArr(int pixel) {
    int red = (pixel >> 16) & 0xff;
    int green = (pixel >> 8) & 0xff;
    int blue = (pixel) & 0xff;
    return new int[]{red, green, blue};
}

public static boolean isGray(int[] rgbArr) {
    int rgDiff = rgbArr[0] - rgbArr[1];
    int rbDiff = rgbArr[0] - rgbArr[2];
    int tolerance = 10;
```

```
        if (rgDiff > tolerance || rgDiff < -tolerance)
            if (rbDiff > tolerance || rbDiff < -tolerance) {
                return false;
            }
        return true;
    }
}
```

<u>Image cropping and view manager module code:</u>

```
package Util;


import android.content.Context;
import android.graphics.Canvas;
import android.graphics.Movie;
import android.net.Uri;
import android.os.SystemClock;
import android.util.AttributeSet;
import android.util.Log;
import android.view.View;
```

```
import java.io.FileNotFoundException;

import java.io.InputStream;

public class GifImageView extends View {

   private InputStream mInputStream;

   private Movie mMovie;

   private int mWidth, mHeight;

   private long mStart;

   private Context mContext;


   public GifImageView(Context context) {

      super(context);

      this.mContext = context;

   }


   public GifImageView(Context context, AttributeSet attrs) {

      this(context, attrs, 0);

   }
```

```
    public GifImageView(Context context, AttributeSet attrs, int
defStyleAttr) {
        super(context, attrs, defStyleAttr);
        this.mContext = context;
        if (attrs.getAttributeName(1).equals("background")) {
            int id =
Integer.parseInt(attrs.getAttributeValue(1).substring(1));
            setGifImageResource(id);
        }
    }

    private void init() {
        setFocusable(true);
        mMovie = Movie.decodeStream(mInputStream);
        mWidth = mMovie.width();
        mHeight = mMovie.height();
        requestLayout();
    }
    @Override
```

```
    protected void onMeasure(int widthMeasureSpec, int
heightMeasureSpec) {
        setMeasuredDimension(mWidth, mHeight);
    }
    @Override
    protected void onDraw(Canvas canvas) {
        long now = SystemClock.uptimeMillis();
        if (mStart == 0) {
            mStart = now;
        }
        if (mMovie != null) {
            int duration = mMovie.duration();
            if (duration == 0) {
                duration = 1000;
            }
            int relTime = (int) ((now - mStart) % duration);
            mMovie.setTime(relTime);
            mMovie.draw(canvas, 0, 0);
            invalidate();
```

```
        }
    }


    public void setGifImageResource(int id) {
        mInputStream =
mContext.getResources().openRawResource(id);
        init();
    }
    public void setGifImageUri(Uri uri) {
        try {
            mInputStream =
mContext.getContentResolver().openInputStream(uri);
            init();
        } catch (FileNotFoundException e) {
            Log.e("GIfImageView", "File not found");
        }
```

12. WORKING PRINCIPLE

In this project, the operation of the motor pump is determined by the output of the soil moisture sensor and the temperature sensor, and the pump is activated if the soil moisture sensor output is low. When the H output is HIGH, the pump is in the off position. The pump operates according to the temperature output. The Temperature sensor measures the temperature of the atmosphere, supposes the temperature sensor detects a rainy atmosphere, goes to the signal controller, and deactivates the pump. The IR sensor is used for fence monitoring. IR sensors will be installed on agricultural land, and IR sensors will detect harmful animals that cross the borders of agricultural land. If the IR sensor detects abnormalities, the IR sensor will notify the user through the controller, and it will turn on the buzzer and create some fear sound for the threating animals. And, the HC-05 is used to communicate with the user using the Android app called Bluetooth Terminal HC-05 on the user's mobile phone for identifying the status of hardware.

The RGB ratio of leaf disease is taken by the built-in camera of plant analyzer app. The application will calculate the RGB using the Gray process through a reflective glass window. Application will also show the solutions, growth predication and best fertilizer for that plant to improve the production. Also it will generate the final report of the plant and there is an option to share the report.

13. ADVANTAGES

- Reduce the man power
- Easy to identifying diseases
- Naive Bayes require small number of data sets compare to SVM method
- It can make probabilistic predictions.
- Protection from harmful animals
- Saving of water and money
- Low cost
- Solar panel and Batteries can use as power supply

14. DISAVANTAGES

- This does not apply to long distance communication because we are using the Bluetooth module. GSM applies to remote communication instead of Bluetooth module.
- If we want to take a photo of the infected leaf, we should go ahead and take a photo of the leaf.
- It requires larger number of test data.
- It tends to generalize the result more.

15. CONCLUSION

A developed smartphone irrigation sensor complied with the conceived concept of an optically triggered automated irrigation using a soil imaging process. Due to rapid growth of smartphone appliances at affordable prices, this App represented a simple and practical implementation. The sensor installation in the field can be done simultaneously with the preparation of the cultivation beds and irrigation tubes, so there is no significant additional labour, nevertheless compared with traditional sensors; the installation in the field requires more effort and time. The irrigation sensor has an inherent advantage over other kind of soil moisture sensors for irrigation purposes. The outcome of others depends of soil characteristics like: density, compaction, gravimetric or mixture of their components among others. The irrigation sensor is of non-contact type, requiring only an in situ calibration to acquire the dynamic range for any soil type. This procedure may represent a disadvantage respect to other kind of sensors. The irrigation sensor is a low power consumption standalone device that can be maintained operative with a small solar panel and rechargeable batteries in order to operate for the whole cultivation period, without the usage of cables or external wired connections.

16. REFERENCES

- https://www.farnell.com/datasheets/1682209.pdf
- https://www.ijitee.org/wp-content/uploads/papers/v8i7/G5432058719.pdf
- http://tinkbox.ph/sites/tinkbox.ph/files/downloads/5V_BUZZER_MODULE.pdf
- http://www.brainzstore.com/admin/productimages/5V%20Mini%20Water%20Pump/water%20pump%20mini%205v.pdf
- http://www.ti.com/lit/ds/symlink/lm35.pdf
- https://www.rhydolabz.com/documents/26/IR_line_obstacle_detection.pdf
- http://www.electronicaestudio.com/docs/istd016A.pdf
- https://www.electronicshub.org/understanding-7805-ic-voltage-regulator/