Intelligent Agriculture System

EE712 - Embedded System Design

Aayush Shrivastava
Nirmal shah
Bhavya Dixit

Under the Guidance of
Prof. Dinesh Sharma and Prof. Laxmeesha Somappa

Indian Institute of Technology Bombay

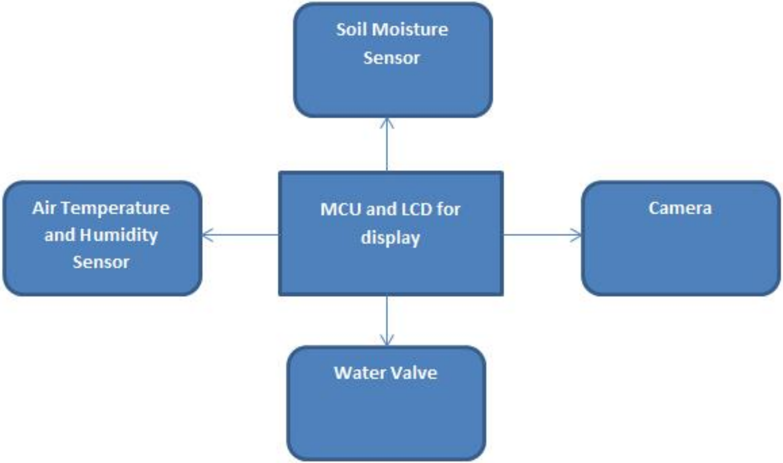
May 02, 2023

Table of Contents

Motivation

- In order to optimize irrigation, increase fertilization, and other farming practices. We developed a **Smart Agriculture System** which will help farmers in making data-driven decisions.
- We can have real time monitoring of various parameters, like **air temperature, air humidity and soil moisture** presence using different sensors.
- By automating the irrigation facility, we would decrease the labor and increase the productivity.
- We can also implement a **vision based crop disease detection method**, which would help farmer to detect any diseases and take timely action increasing the yield of the crops.

Schematic Description



- **Raspberry Pi:** Raspberry Pi 4 Model B
  - Raspberry pi is a 40 GPIO pin development board having integrated CPU and on chip GPU. It has applications in weather monitoring, home security, smart agriculture, robotics etc.

Processor	64 bit Quad Core, ARM Cortex A72 v8
Operating System	Raspberry Pi OS (loaded in Micro SD card)
RAM size	4 GB
IO Interface	USB 3.0 (2), USB 2.0 (2), Micro HDMI port(2), 4 pole stereo audio and video port
Display	2 lane MIPI DSI Display port, 2 lane MIPI CSI Camera port
Wireless Connection	Bluetooth 5.0, BLE, 802.11ac wireless standard at 2.4GHz and 5GHz
LAN	Gigabit ethernet

Table: Technical Details

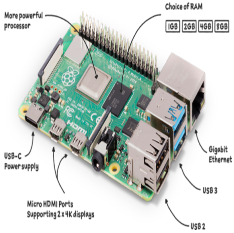


Figure: Raspberry Pi 4B

Components Used

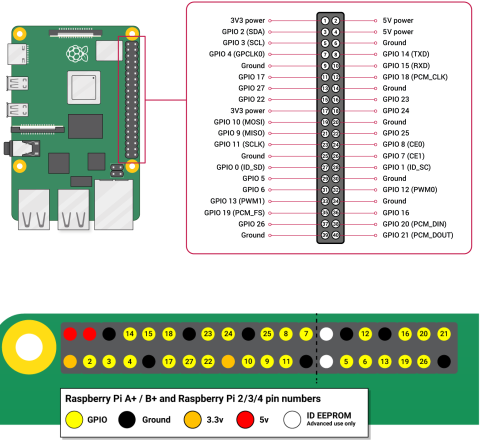


Figure: Raspberry Pi 4B Pin Configuration [1]

[1] Image Source:www.raspberrypi.com

Components Used

- **Air Temperature and Humidity Sensor:** DHT 22
  - DHT22 is a low cost air temperature and humidity sensor which uses a capacitive humidity sensor and thermistor to measure the surroundings air and produces a digital signal for the Data pin.

Parameter	Measuring Range
Input Voltage	3 to 5 Volt
Humidity	0 to 100% Accuracy: 2 to 5%
Temperature	-40 to 80°C Accuracy: 0.5°C
Sampling rate	0.5 Hz

Table: Sensor Specifications

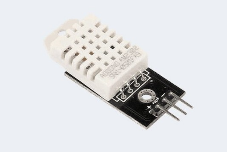


Figure: Humidity/Temp. Sensor

  - This sensor is more precise, more accurate, and works in a bigger range of temperature/humidity as compared to other Humidity sensor.

Components Used

- **Soil Moisture Sensor:** EC - 1258
  - It is a digital soil moisture sensor. When inserted in the soil, it measures moisture or water level content in it.
  - It gives a digital output of 5V when moisture level is high and 0V when the moisture level is low in the soil.
  - The sensor includes a potentiometer to set the desired **moisture threshold**. When the sensor measures more moisture than the set threshold, the digital output goes high and an LED indicates the output. When the moisture in the soil is less than the set threshold, the output remains low.
  - Digital output D0 is connected to Raspberry Pi to sense the presence of soil moisture level.
  - Analog output A0 can be connected to an ADC and Raspberry Pi to get exact moisture level in soil.

Components Used

Parameters	Specifications
Operating voltage	3.3 V to 5V
Indicator	Red (Power) Green (Digital output)
Digital output interface (D0)	Presence of moisture(above threshold) = 1 Absence of moisture(below threshold) = 0
Analog output interface (A0)	Moisture level in soil

Table: Sensor Specifications

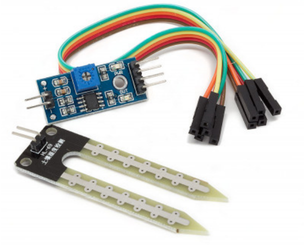


Figure: Soil Moisture Sensor[a]

[a]Image Source: www.electronicscomp.com

- **Camera Module:** EC -1165
  - Raspberry Pi has 2 lane MIPI CSI interface for this 5MP Camera module which connects to Raspberry Pi through a short flexible ribbon cable.
  - The CSI bus is capable of extremely high data rates, and it exclusively carries pixel data.

Parameters	**Specifications**
Camera Pixels	5 MP
Still Picture Resolution	2592 x 1944 p
Video	1080p @30fps 720p @60fps
Video Resolution	640 x 480 p
Port	15 pin MIPI CSI

Table: Camera specifications

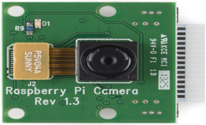


Figure: Camera Module

Team member's individual contributions

- **Aayush**: Implemented the Machine learning model for vision based disease detection.
- **Nirmal**: Integrated the sensors and camera with Raspberry Pi and contributed in python programming.
- **Bhavya**:Tested the sensors individually and contributed in python programming.

Implementation details

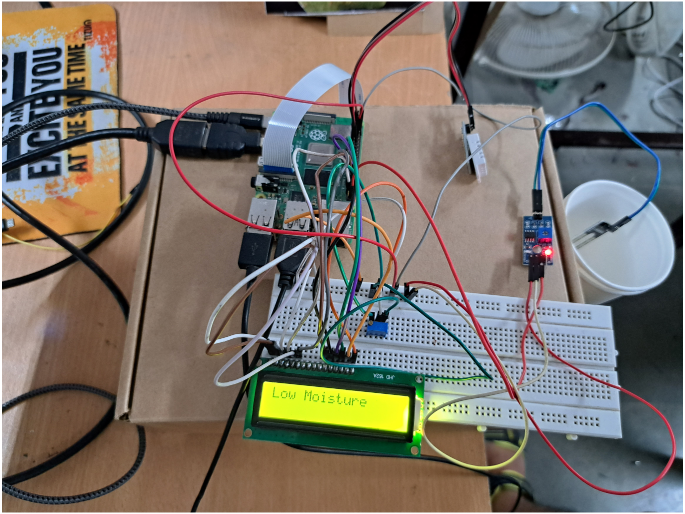


Figure: Circuit setup

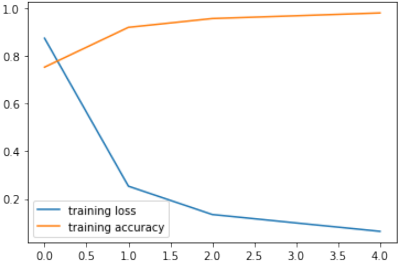
Implementation details

- **Interfacing Raspberry Pi with Soil Moisture sensor:**
  - We connect digital output D0 of sensor to GPIO pin 21 of Raspberry Pi, Vcc connected to 5V and GND connected to 0V.
  - When water level in soil is below threshold value then LCD displays **"No Moisture"** and when it is above threshold value then LCD displays **"Water level is OK"**,
- **Interfacing Raspberry Pi with Temperature and Humidity sensor :**
  - We connect data pin of sensor to GPIO pin 4 of Raspberry Pi, Vcc to 5V and GND to 0V.
  - DHT 22 sensor observes air temperature and humidity of surroundings and display these values in LCD.
- **Interfacing Raspberry Pi with camera:**
  - We connect Raspberry Pi Camera to 2 lane MIPI CSI interface which captures the image of plant leaf and detect disease using machine learning model.

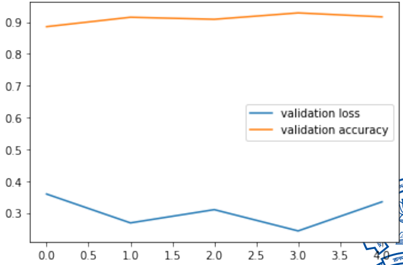
Implementation Details

Machine Learning Model

- A machine learning model was trained to detect diseases based on the images received of the plant's leaves.
- A CNN was implemented, whose layers are shown in the image on next page.
- We get a training accuracy of 97.89% and 91.64% accuracy on the validation set. For training on 5 epochs, the training and validation graphs are shown below.



((a)) Training



((b)) Validation

The CNN model used for disease detection

Model: "sequential"

Layer (type)	Output Shape	Param #
batch_normalization (BatchNormalization)	(None, 250, 250, 3)	12
conv2d (Conv2D)	(None, 248, 248, 32)	896
max_pooling2d (MaxPooling2D)	(None, 124, 124, 32)	0
conv2d_1 (Conv2D)	(None, 122, 122, 64)	18496
max_pooling2d_1 (MaxPooling2D)	(None, 61, 61, 64)	0
conv2d_2 (Conv2D)	(None, 59, 59, 128)	73856
max_pooling2d_2 (MaxPooling2D)	(None, 29, 29, 128)	0
flatten (Flatten)	(None, 107648)	0
dense (Dense)	(None, 256)	27558144
dense_1 (Dense)	(None, 38)	9766

Figure: Summary of the trained CNN model

Implementation Details

The following image shows the working of the model as it correctly predicts the disease.

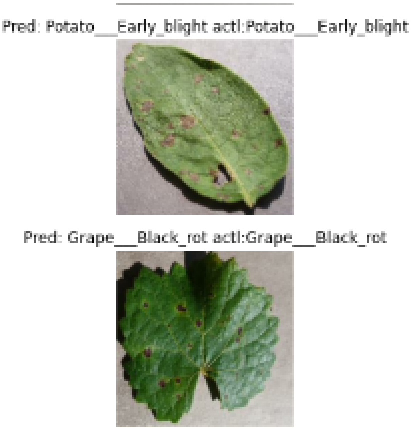


Figure: Model correctly predicting disease on plant leaves (Here Pred: is the predicted disease and actl: is the actual disease

Challenges faced during implementation

- DHT22 sensor has a sampling rate of 0.5 Hz i.e. it senses air temperature and humidity every 2 second which is a bit slow.

For the machine learning part

- The model training on the data set for 1 epoch took around 1hr 40min. So training the model for a larger number of epochs was a challenge as it took around 8 hours for the complete training of the model.
- Since we cannot train the model on R-pi, the training was done on the cloud, and the model was loaded onto R-pi. So the model size was initially an issue due to memory constraints of the R-pi, so we had to keep the model not very complex such that it can be loaded on R-pi.

Demonstration video

Embedded Video

Thank You for Your Attention!