Leaf Disease Detection System and Automatic Pesticide Spraying Control Robot Using Raspberry Pi

[1]K.Veenanand, [2]Seema Afreen Khan , [3]M.Venkata Vijay, [4]T. Raja, [5]V. Raja Rajeswari

dept of ECE, DVR Dr. HS MIC College of Technology kanchikacherla,AP,India

Abstract: For the detection and prevention of diseases of plants from getting spread, this project proposed a system using image processing. For the image analysis, Conventional Neural Networks were used. It has many advantages for use in big farms of crops and thus it automatically detects signs of disease whenever they appear on the leaves of the plant. In pharmaceutical research leaf disease detection is a necessary and important topic for research because it has advantages in monitoring crops in the field at the form and thus it automatically detects symptoms of the disease by image processing by the CNN algorithm. The term disease means the type of damage to the plants. This project provides the best method for the detection of plant diseases using image processing and alerting about the disease caused by sending to an IOT Server and displaying the name of the disease and precautions on the mobile application of the owner of the system. And by using robots it automatically sprays pesticides to the plants. It will reduce the cost required   for pesticides and other products. This will lead to an increase in productivity of the farming.

Index Terms: Image Processing, Raspberry Pi, Python.

I. INTRODUCTION

The economy of our nation depends heavily on agriculture. Agriculture is the primary source of income for about 70 per of rural households and accounts for 17 per of the GDP overall. A little over 60 per of the population works in agriculture. Therefore, accurate crop disease identification is necessary to support our country's agriculture industry and economy. For their farms, farmers can pick from a wide range of crops. In any event, crop production for standard manufacture and optimum profit is frequently scientific. This could be created with the use of technological knowledge. In order to monitor continuously recurring crops and prevent diseases, which might have a significant impact on many production components, the ultimate authority is required. The picture procedure is the best way to acquire a paid job performing agricultural application duties. Plant images can be used to identify diseases. Assistance for agricultural development can lessen this impact. Human vision is limited in its ability to identify the disease because the majority of the fundamental symptoms are small. This method is time-consuming and tedious. It is necessary to develop a style system that can automatically identify, categorize, and quantitatively diagnose disorders.
Identify applicable funding agency here. If none, delete this.

II. LITERATURE SURVEY

A.   "A smart and automated plant disease diagnosis and control system using Raspberry Pi" by T.S. Sindhuja and
V. Padmavathy (2018) - This paper presents a smart and automated plant disease diagnosis and control system that uses Raspberry Pi and a camera module to detect plant diseases and control them by spraying pesticides. The system uses image processing techniques to identify the symptoms of diseases on the plant leaves and sends the control signals to the pesticide spraying robot.
B.   "Design of an automated plant disease detection and control system using Raspberry Pi" by A. H. A. Rahman et al. (2018) - This paper proposes an automated plant disease detection and control system using Raspberry Pi, OpenCV, and Python. The system uses a camera module to capture images of plant leaves and then applies image processing algorithms to identify the disease. The system is also equipped with a pesticide spraying mechanism to control the disease.
C.   "Smart farming using IoT with automatic plant disease detection and pesticide spraying" by R. Balasubramanian et al. (2019) - This paper describes an IoT-based smart farming system that uses Raspberry Pi, Arduino, and camera modules to detect plant diseases and spray pesticides automatically. The system uses deep learning algorithms to classify plant diseases and sends control signals to the pesticide spraying mechanism.

III. EXISTING SYSTEM

There are several existing systems for leaf disease detection and automatic pesticide spraying control using Raspberry Pi. Here are a few examples:

AgroBot: This is an agricultural robot developed by a team at the University of Warwick, UK. It uses a Raspberry Pi to detect leaf diseases and pests using computer vision and machine learning algorithms. Once a disease or pest is detected, the robot automatically sprays the affected area with pesticides.

AgriBot: This is another agricultural robot developed by a team at the University of Lincoln, UK. It uses a Raspberry Pi and machine learning algorithms to detect and classify different types of crop diseases. The robot can then apply tar-geted treatments, such as spraying with pesticides or applying fertilizer, to the affected areas.

SmartSprayer: This is a commercial product developed by Intelligent Ag Solutions, LLC. It uses a Raspberry Pi and computer vision algorithms to detect and classify weeds in agricultural fields. The system then controls a sprayer to apply herbicides only to the areas where weeds are present, reducing the amount of herbicide used and minimizing environmental impact.

CropVision: This is a software system developed by a team at the University of Manchester, UK. It uses a Raspberry Pi and machine learning algorithms to analyze images of crops and detect diseases and pests. The system can also provide recommendations for pesticide application and other treatments.

These systems are all designed to help farmers reduce the amount of pesticides they use while still protecting their crops from pests and diseases. They rely on advanced computer vision and machine learning techniques to accurately identify and diagnose crop issues, and then use automated sprayers to apply targeted treatments only where they are needed.

IV. RELATED WORK

There have been several research studies and projects on leaf disease detection systems and automatic p "Leaf Disease Detection using Raspberry Pi", by P. M. Jadhav and A. D. Kharate, International Journal of Science and Research, 2018: This research project developed a leaf disease detection system using Raspberry Pi and image processing techniques. The system used a camera to capture images of plant leaves and then analyzed the images to detect and classify different types of diseases.

"Automatic Detection and Classification of Plant Leaf Dis- eases using Raspberry Pi", by S. M. Patil and S. S. Thakur, International Journal of Computer Applications, 2017: This research project developed an automatic plant leaf disease detection and classification system using Raspberry Pi and machine learning algorithms. The system used image pro- cessing techniques to identify and classify different types of diseases in plant leaves, and then provided recommendations for appropriate treatments. pesticide spraying control robots using Raspberry Pi. Here are a few examples of related work: "Automated Pesticide Spraying System for Crop Fields using Raspberry Pi", by C. P. Subathra and S. Sankar, Interna- tional Journal of Innovative Technology and Exploring Engi- neering, 2019: This research project developed an automated pesticide spraying system for crop fields using Raspberry Pi and computer vision algorithms. The system used image processing techniques to detect and classify different types of pests and diseases in crops and then sprayed pesticides only
in the affected areas.

"Smart Irrigation and Fertilization System using Raspberry Pi and IoT", by H. V. Sonawane et al., International Journal of Engineering and Advanced Technology, 2018: This research project developed a smart irrigation and fertilization system for crops using Raspberry Pi and IoT technologies. The system used sensors to collect data on soil moisture, temperature, and nutrient levels and then used machine learning algorithms to determine the optimal amount of water and fertilizer to apply to the crops.

"Leaf Disease Detection using Raspberry Pi", by P. M. Jadhav and A. D. Kharate, International Journal of Science and Research, 2018: This research project developed a leaf disease detection system using Raspberry Pi and image processing techniques. The system used a camera to capture images of plant leaves and then analyzed the images to detect and classify different types of diseases.

"Automatic Detection and Classification of Plant Leaf Dis- eases using Raspberry Pi", by S. M. Patil and S. S. Thakur, International Journal of Computer Applications, 2017: This research project developed an automatic plant leaf disease detection and classification system using Raspberry Pi and machine learning algorithms. The system used image pro- cessing techniques to identify and classify different types of diseases in plant leaves and then provided recommendations for appropriate treatments.

V. METHODOLOGY

Plants are susceptible to several disorders and attacks caused by diseases. There are several reasons that can be characterizable to the effects on the plants, disorders due to the environmental conditions, such as temperature, humidity, nutritional excess or losses, light, and the most common dis- eases that include bacterial, virus, and fungal diseases. Those diseases along with the plants may show different physical characteristics on the leaves, such as changes in shapes, colors, etc. Due to similar patterns, those above changes are difficult to be distinguished, which makes their recognition a challenge and earlier detection and treatment can avoid several losses in the whole plant. In this paper, we are discussed to use of recent detectors such as Faster Region-Based Convolutional Neural Network (Faster R-CNN), Region-based Fully Convolutional Networks (R-FCN), and Single Shot Multi-box Detectors to detect and classification of plant leaf diseases that affect various plants.

Hardware Components:
1) DC Motor
2) L293, L293D
3) Submersable water pump
4) Raspberry pi
5) Registers

A. DC Motor

A DC motor is made to function with DC electric current. Michael Faraday's homopolar motor, which is unusual, and the ball bearing motor, which is (so far) unique, are two examples of pure DC designs. Brushed and brushless motors, which use internal and external commutation to produce oscillating AC current from a DC source, respectively, are by far the two most popular types of DC motors. As a result, they are not strictly speaking DC machines.
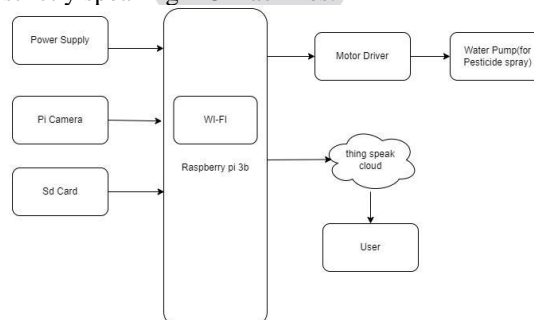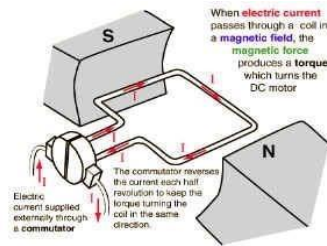

Fig. 1. Block Diagram

Fig. 2. DC Motor

B. L293, L293D

These quadruple high-current half-H drivers are the L293 and L293D. At voltages ranging from 4.5 V to 36 V, the L293 is intended to deliver bidirectional drive currents of up to 1 A. At voltages ranging from 4.5 V to 36 V, the L293D is intended to deliver bidirectional drive currents of up to 600 mA. Both devices are made to drive inductive loads in positive-supply applications, including relays, solenoids, dc, and bipolar stepping motors, among others. All inputs are compatible with TTL. With a Darlington transistor, each output is a complete totem-pole drive circuit.
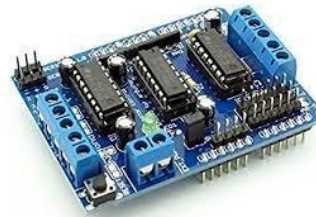

Fig. 3. L293D

C. Submersable water pump

This inexpensive tiny submersible water pump operates at 3-6V DC. It is quite straightforward to use and understand. To begin pumping water, just submerge the pump in water, attach a suitable pipe to the output, and supply the motor with 3-6V. Excellent for creating science projects, fire extinguishers, fire fighting robots, fountains, waterfalls, plant irrigation systems, etc. This motor is nimble, lightweight, and compact. With the help of one of our DC Motor Drivers or Relay Boards, it may be controlled from a microcontroller or Arduino. To power this pump, utilize our 5V SMPS Power Supply Adapter. You may also use our 6V solar panel in conjunction with the proper 6V voltage regulator to power the pump.

Fig. 4. Submersable water pump

D. Raspberry pi

The 64-bit ARMv7 Quad Core Processor used in the Rasp- berry Pi is based on the Broadcom BCM2837. It contains a Micro SD card slot, a LAN port (Ethernet port), 40 general- purpose input/output pins, USB ports, a DSI display connector, a micro-USB power input, a composite video and audio output jack, a CSI camera port, and an HDMI video output. Everything required to support the microprocessor is included. Connect it to a computer running Raspbian OS, and use an adapter to provide electricity. In contrast to all previous boards, the Raspberry Pi 3 model B lacks on-board WiFi, Bluetooth, and USB boot capabilities. The Pi 3 is approximately 50 per quicker than all earlier boards. The term "raspberry" alludes to a custom of fruit naming in the early days of microcomputers. There are numerous fruit-inspired names for computer firms. The reason "Pi" was chosen was because we were planning to build a computer that could only really run Python. The 'Pi' stands for Python, then.
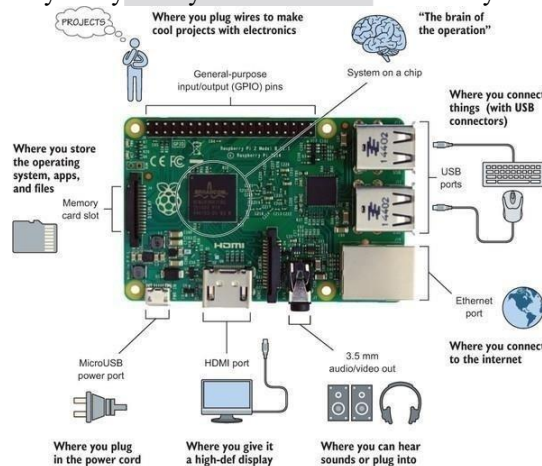

Fig. 5. Raspberry pi

E. *Registers*

The user has access to 16 general-purpose registers thanks to the ARM ISA. The program counter is in register 15, although it can also be used for other purposes. The branch-and-link instruction uses the general-purpose register number 14 as a link register. Although it is not required by the architecture, register 13 is generally utilized as the stack pointer. The Cur- rent Program Status Register (CPSR) has four 1-bit condition flags ('Negative', 'Zero', 'Carry', and 'overflow') and four fields that represent the processor's current state of execution. To transition between the ARM and Thumb instruction sets, utilize the 'T' field. Normal and quick interruptions are enabled by the 'I' and 'F' options, respectively. The mode parameter allows you to choose from seven different execution modes.

VI. SOFTWARE SPECIFICATIONS

Operating systems based on the Linux kernel are gener- ally used by the Raspberry Pi. Version 6 of the ARM, on which the ARM11 is based, is no longer supported by a number of widely used Linux distributions, including Ubuntu. NOOBS is the install manager for the Raspberry Pi. The following operating systems are bundled with NOOBS: RISC OS, which was the operating system of the first ARM-based computer; Archlinux ARM; Open ELEC; Pidora (Fedora Remix); Raspbmc and the XBMC open-source digital media center. • Raspbian (recommended) - Maintained independently of the Foundation; based on ARM hard- float (armhf)-Debian 7 'Wheezy' architecture port; designed for an older ARMv7 processor whose binaries would not work on the Raspberry Pi; however, Raspbian is compiled for the ARMv6 instruction set of the Raspberry Pi making it work but with slower performance.

A. *Boot Process*

1)   Step 1: The Raspberry Pi does not boot as a traditional computer. The Video Core i.e., the Graphics processor actu- ally boots before the ARM CPU. The boot process of the Raspberry Pi can be explained as follows:

•   When the power is turned on, the first bits of code to run are stored in a ROM chip in the SoC and is built into the Pi during manufacture. This is called the first-stage bootloader.

•   A tiny RISC Core (Reduced Instruction Set Computer) is hardwired into the SoC to run this code when it boots up. In order to access the second-stage bootloader, it is utilized to mount the FAT32 boot partition on the SD Card. What exactly is this "second-stage bootloader" that is kept on the SD Card? 'bootcode.bin' is what it is. When mounting an operating system on the SD Card in Windows, this file can be displayed.

B. *The NOOBS installer*

1)   Step 2: On the official website, you may download the NOOBS installer. To install NOOBS on an SD card, a user merely needs to attach the SD card to a computer and run the setup program. Put the card into the Raspberry Pi next. When the computer boots up for the first time, the NOOBS interface loads, allowing the user to choose which operating system to install. This method of operating system installation is significantly more practical. Additionally, once the operating system has been set up on the card using the NOOBS installer, the recovery mode offered by the NOOBS can be accessed by holding down the shift key while the Pi boots. Additionally, it enables the alteration of the operating system's config.txt file.

C. *Open CV with Python*

1)   Step 3: We will learn how to use the Python OpenCV library in this tutorial. OpenCV is an open-source library that may be used in a variety of languages and on a variety of operating systems, including Windows, Linux, and MacOS. However, it is most frequently used in Python for Machine Learning applications, notably in the Computer Vision field. The installation of OpenCV on Mac, Windows, and Linux, as well as image operations, image arithmetic, picture smoothing, and geometric transformations using OpenCV are all covered in this lesson. So let's get started right away.

D. *Image Smoothing*

1)   Step 4: The characteristic of picture smoothing, which is typically used before the photos are sent to a machine-learning model, is quite helpful. By running an image through a low-pass filter, noise and high-frequency features are pri- marily removed from photos. Although there are other filters, including the box filter (averaging filter), the median filter, the mode filter, the Gaussian filter, and many more, we will only discuss the box filter in this article in order to better understand image smoothing and how to implement it using Open CV. Let's say you want to apply a 3x3 box/averaging filter to a 10x10 image. How would you go about doing that? Place your 3x3 filter in the top left corner of the image to begin, and then swap out the central element for the average of the other nine elements. This was the first phase; shift your filter one step to the right and then continue the process until the entire image is covered. Below is an illustration using 10x10 photos and a 3x3 averaging filter for your reference.
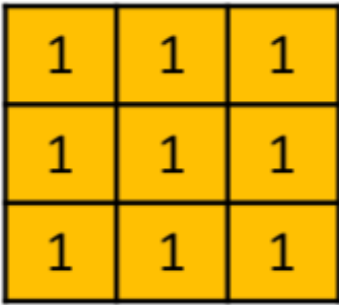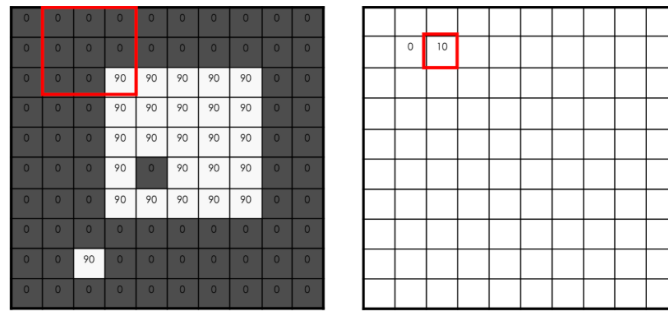

Fig. 6.   Filter/Mask

Fig. 7. Filter Applied 10x10 Image

VII. CONVOLUTION NEURAL NETWORK

A. Convolutional kernels

A group of filters known as convolutional kernels is present in each convolutional layer. The filter, which has the same size as the kernel, is an integer matrix applied to a portion of the input pixel values. In order to make things simpler, each pixel is multiplied by the kernel value that corresponds to it, and the result is then summed to create a single value that represents a grid cell—just like a pixel—in the output channel/feature map.
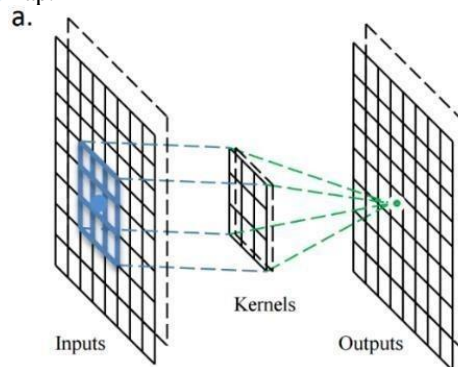

Fig. 8. Convolution Kernels

Fig. 9. Kernel Scanning over the value of input matrix

B. learning algorithms used to detect diseases on plant leaves. This could involve training the system on larger datasets, incorporating more features into the model, or using more advanced algorithms such as deep learning. 2. Expand the range of plants and diseases detected: Another potential area of improvement is to expand the range of plants and diseases that the system can detect. This could involve adding new plant species to the system's database or incorporating new types of diseases that are not currently being detected.

3. Develop a more sophisticated spraying control system: The current system relies on a set of pre-programmed spraying patterns based on the location of the detected disease. A more advanced system could incorporate real-time data on weather conditions, soil moisture, and other factors to adjust the spraying pattern in real-time to maximize efficiency and minimize environmental impact.

IX. CONCLUSION

There are essentially three different types of leaf disease: bacterial, fungal, and viral. It is crucial to have precision in plant disease diagnosis while also ensuring that the procedure moves along quickly. Using a quadcopter to take field-level pictures of the leaves of the various plants on the farm can lengthen the work. For additional processing, this system can be connected to the server. The goal of this endeavor is to identify and categorize leaf diseases and to send all relevant information to the farmer's cell phone through the Internet. Using the Raspberry Pi 3 model B module, we can identify and classify leaf diseases more quickly and accurately. This system's ability to identify the pesticide that must be used to stop the disease from spreading is yet another crucial advan- tage. By eliminating the need for labor-intensive routine plant observation to determine if a plant is afflicted by a disease or not, it provides the name of the pesticide according to the disease, saving labor costs. This technology will significantly increase the farms' yields.

X. REFERENCE

1. B. S. Saini, S. K. Singh, and R. Singh, "A review on detection of plant diseases using image processing techniques," Journal of Agricultural Engineering, vol. 54, no. 2, pp. 51- 66, 2017. This paper provides a comprehensive review of the different image-processing techniques used for plant disease detection.
2. M. R. Islam, M. M. Hossain, M. F. Hassan, and M. S. Islam, "Design and development of a leaf disease detection and

classification system using Raspberry Pi," Computers and Electronics in Agriculture, vol. 161, pp. 272-282, 2019. This paper describes the design and development of a leaf disease detection and classification system using Raspberry Pi.

3. FUTURE WORK

4. There are several potential areas of future work on a leaf disease detection system and automatic pesticide spraying control robot using Raspberry Pi. Here are a few ideas:

5. Improve the accuracy of the disease detection system: One potential area of improvement is to refine the machine IEEE conference templates contain guidance text for com- posing and formatting conference papers. Please ensure that all template text is removed from your conference paper prior to submission to the conference. Failure to remove the template text from your paper may result in your paper not being published.