Untitled

by smitaagrawal

General metrics

55,904	7,334	299	29 min 20 sec	56 min 24 sec
characters	words	sentences	reading time	speaking time

Score

98

This text scores better than 98% of all texts checked by Grammarly

Writing Issues

86 Issues left

Critical

86 Advanced

Plagiarism

2 %

17 sources

2% of your text matches 17 sources on the web or in archives of academic publications

Writing Issues

82	**Clarity**	
79	Passive voice misuse	
1	Hard-to-read text	
1	Wordy sentences	
1	Intricate text	
4	**Delivery**	
3	Inappropriate colloquialisms	
1	Incomplete sentences	

Unique Words

Measures vocabulary diversity by calculating the percentage of words used only once in your document

15%

unique words

Rare Words

Measures depth of vocabulary by identifying words that are not among the 5,000 most common English words.

42%

rare words

Word Length

Measures average word length

3.7

characters per word

Sentence Length

Measures average sentence length

24.5

words per sentence

Untitled

87 %\documentclass[pdflatex,sn-mathphys]{sn-jnl}% Math and Physical Sciences Reference Style

\documentclass[pdflatex,sn-basic]{sn-jnl}% Math and Physical Sciences Reference Style

\jyear{2021}

\theoremstyle{thmstyleone}%

\newtheorem{theorem}{Theorem}% meant for continuous numbers

\newtheorem{proposition}[theorem]{Proposition}%

\theoremstyle{thmstyletwo}%

\newtheorem{example}{Example}%

\newtheorem{remark}{Remark}%

\theoremstyle{thmstylethree}%

\newtheorem{definition}{Definition}%

\raggedbottom

\usepackage{natbib}

\usepackage{multirow}

\usepackage{graphicx}

\usepackage{caption}

\usepackage{subcaption}

\usepackage[T1]{fontenc}

\usepackage{url}

\usepackage{xcolor}

\usepackage{float}

\usepackage{ulem}

%\usepackage[round]{natbib}

%\usepackage{natbib,stfloats}

%Import the natbib package and sets a bibliography and citation styles

%\usepackage{natbib}

% \usepackage{apacite}

%\bibliographystyle{abbrvnat}

%\setcitestyle{authoryear,open={(},close={)}}

\begin{document}

\title[]{Fuzzy logic aided smart irrigation system using soil moisture sensor, weather prediction, and convoluted neural networks}

% A smart watering system, backed by fuzzy logic, uses convoluted neural networks, weather forecasts, and soil moisture sensors.

%%===%%

%% Prefix -> \pfx{Dr}

%% GivenName -> \fnm{Joergen W.}

%% Particle -> \spfx{van der} -> surname prefix

%% FamilyName -> \sur{Ploeg}

%% Suffix -> \sfx{IV}

%% NatureName -> \tanm{Poet Laureate} -> Title after name

%% Degrees -> \dgr{MSc, PhD}

%% \author*[1,2]{\pfx{Dr} \fnm{Joergen W.} \spfx{van der} \sur{Ploeg} \sfx{IV}

\tanm{Poet Laureate}

%% \dgr{MSc, PhD}}\email{iauthor@gmail.com}

%%===

===========%%

%\author[1]{\fnm{Pranshu} \sur{Patel}}\email{19bce191@nirmauni.ac.in}

88 %\equalcont{These authors contributed equally to this work.}

%\author[2]{\fnm{Yugma} \sur{Patel}}\email{19bce204@nirmauni.ac.in}

89 %\equalcont{These authors contributed equally to this work.}

%\author[3]{\fnm{Utsav} \sur{Patel}}\email{19bce200@nirmauni.ac.in}

90 %\equalcont{These authors contributed equally to this work.}

%\author[4]{\fnm{Vrukshal} \sur{Patel}}\email{19bce203@nirmauni.ac.in}

91 %\equalcont{These authors contributed equally to this work.}

%\author[5]{\fnm{Nitya} \sur{Patel}}\email{19bce188@nirmauni.ac.in}

92 %\equalcont{These authors contributed equally to this work.}

%\author*[6]{\fnm{Parita} \sur{Oza}}\email{parita.prajapati@nirmauni.ac.in}

93 %\equalcont{These authors contributed equally to this work.}

%\author*[7]{\fnm{Usha} \sur{Patel}}\email{ushapatel@nirmauni.ac.in}

%\affil[1]{\orgdiv{Department of Computer Science and Technology}, \orgname{Nirma University}, \orgaddress{ \city{Ahmedabad}, \postcode{382481}, \state{Gujarat}, \country{India}}}

\abstract{Among the developing countries, a large part of their economies is dependent on their agricultural sectors. Besides, countries like India and China also have a high population, which means there will always be a scarcity of resources and difficulties in resource allocation. Due to a global lack of clean water supplies, it is imperative that they be used to their full potential. Increasing global warming led to uncertain rainfall and an-isotropic conditions. This article represents a prototype model for an autonomous automated watering system that aims to minimize both human intervention and water consumption. Traditional methods have been used [1] for watering gardens and farms for many years. These techniques have disadvantages, like less efficiency, wastage of water, and over-irrigation of the plants. This watering process continued for many years, and then some contemporary technologies came into play, like water jets, water sprinklers, drip irrigation, etc. [2] These technologies could prevent water wastage and help farmers and gardeners achieve better [3] results. This technology still requires automation to minimize manual inspection, which is very laborious. This system provides an easy

interface for cultivation in all fields, ranging from domestic to industrial applications. This system will take as input sensed data from soil moisture sensors and images of crops captured by a camera, and weather predictions from the Application Programming Interface(API). The algorithm implemented on the processor will aggregate these details, and as per the threshold value set, it will control the duration for which the irrigation system operates.}

\keywords{Transfer learning, Fuzzy Logic, Image Classification, Irrigation, Raspberry Pi}

\maketitle
{\color{red}}

\section{Introduction}\label{Introduction}

India has always been an agricultural nation. Almost half of the employment in India comes from the agricultural sector. Agriculture is the primary means of
94 living for about 58\% of India's population \citep{percent}. The share of agriculture in GDP was 19.9\% in 2020-21 \citep{percent}. However, optimal water use during irrigation of crops needs to be addressed [4] because the lack of fresh water is a rising concern in many parts of the world.
Fig. \ref{fig:percapita} shows the decrease in per capita water supply over 100 years. Water availability per capita has decreased by 70\% since 1950, and this trend is expected [5] to continue over the next decade.

Before the advent of manual water irrigation methods, farmers primarily relied on rain for watering their crops. However, farmers cannot rely solely on rainwater due to global warming because climate change is unpredictable. We have also observed abrupt drought conditions in many places in recent years. Moreover, only 2.5\% of the total volume of water comprises freshwater, so the usage of freshwater needs to be appropriately managed \citep{freshwater}. Irrigation is scheduled[6] around the world based on farmers' visual assessment of crops, which wastes nearly half of the water used by traditional irrigation systems \citep{8521025}.
Approximately 70\% of the entire freshwater reserves are used[7] for agriculture, so the water wastage should be minimal to preserve the resources for future generations \citep{waterusage}.

Sprinkle irrigation, drip irrigation, and furrow irrigation are examples of controlled irrigation methods that reduce water waste by 30\% to 70\% \citep{natgeo}. But[8], due to the open-loop layout, these techniques fail to maintain actual water content in the soil, resulting in lower crop quality and quantity when soil nutrients are depleted by under or over-irrigation \citep{MORILLO201543}.

Due to India's high cost and lack of advanced methods, most farmers rely on flood irrigation, which releases water into the field for a set period.

One disadvantage to this irrigation technique is the manual observation and decision-making of how much water to release. Farmers cannot measure the exact amount of soil moisture, and due to uncertain rainfall, it is challenging to determine irrigation duration manually.

```
\begin{figure}[!ht]
\centering
\includegraphics[width=0.9\textwidth]{Percapita.png}
\caption{Population and per capita water supply in India}
\label{fig:percapita}
\citep{kpmg2010water}
\end{figure}
```

\subsection{Motivations}

The motivation for this article is mentioned below:

\begin{itemize}

\item Freshwater scarcity has been one of the world's most pressing issues. Thus, one of the most critical factors addressed by the authors is conserving water and minimizing water wastage while irrigating crops.

\item Some of the essential nutrients plants require are present in the upper soil layers. However, these nutrients drain away due to water-logging and dehydration, which erodes the soil. Therefore, providing the right amount of water for the correct duration is crucial for improving crop yields.

\item The manual operation of the water pump causes complications for farmers, leading to either over-irrigation or under-irrigation of the plants. In addition, farmers cannot determine the amount of moisture in the soil or the current weather conditions. So, the authors proposed an automated system for

irrigating crops that effectively addresses soil moisture, meteorological conditions, and crop health.
\end{itemize}

\subsection{Contributions}

Watering crops at an adequate level is a difficult task due to multiple parameters taking part, making it hard to put them together and decide the duration for irrigation. Researchers worldwide have worked on building smart irrigation system that focuses only on soil moisture or weather conditions. However, no system exists that considers the image of the crop(to classify as a droop or healthy) along with the aforementioned parameters(soil moisture and weather predictions) that influence irrigation duration. In a nutshell, the following are the crisp objectives of the paper.

\begin{itemize}
\item We present a comprehensive study of the numerous smart irrigation systems currently in use and their drawbacks.
\item A new fuzzy logic model, consisting of a soil moisture sensor, weather API, and image classification using a Convoluted Neural Network(CNN) model, is proposed to decide the time for adequate watering of the field.
\item The appropriate membership functions are chosen[9], and their boundary values are tuned to refine the fuzzy logic controller.
\item Performance evaluation of the CNN model is conducted[10] using various evaluation metrics like Accuracy, F1-Score, and curve of loss over the epochs.
\end{itemize}

\subsection{Organization}

The organization of the paper is shown [11] in Fig.~\ref{fig:Paperflow}. Section \ref{Related} discusses the related work that has been done [12] in the domain of smart irrigation. Section \ref{Problem} discusses the system model and problem formulation for the autonomous water irrigation system. Section \ref{methodology} describes the workflow of the proposed module. Section \ref{result} discusses the results and evaluates the performance of the proposed model. Finally, Section \ref{conclusion} discusses the future research directions followed by the conclusion of the paper.

\begin{figure}[!ht]

\centering

\includegraphics[width=0.5\linewidth]{Paper flow.png}

\caption{Organisation of paper}

\label{fig:Paperflow}

\end{figure}

\section{Related work} \label{Related}

The advent of IoT devices has led to significant advancements in many fields, and agriculture is no exception. Scientists have recently tried to incorporate machine learning and deep learning models with IoT devices to analyze the input data and predict the required output efficiently.

We can design tools and technologies that are inherently smart and can aid in reducing both environmental degradation and human efforts by properly using science and technology.

A solar-based smart irrigation system is proposed[13] in \citep{uddin2012automated}. The authors have used solar electricity as the sole source of power. Sensors are installed in the paddy field to monitor the water level continuously and notify the user. The user can operate the motor based on the water level by sending an appropriate message from a remote location.

In \citep{pavithra2014gsm}, authors have come up with the idea of a mobile
95 application for the automatic irrigation control system for efficient use of resources and crop planning. This application uses the General Packet Radio Service (GPRS) feature of a mobile phone as a solution for the irrigation control system. The authors additionally employed the Global System for Mobile Communication (GSM) to notify the user of the precise field condition. The user receives the information in the form of a Short Message Service when they request it (SMS).

Authors in \citep{rajendranath2015implementation} built an automated irrigation system using a temperature sensor, humidity sensor, and soil
96 moisture sensor. All these sensors are interfaced[14] with the micro-controller, and the entire unit is placed[15] under the plant's root zone. The authors have tested this irrigation system under different temperatures and humidity levels of different plants under normal and wet conditions.

In \citep{djuzic2017automatic} depicts the system that uses sensor technology with a micro-controller and other electronics that sense the moisture level of soil and irrigate the plant only if needed.

However, these systems irrigate the field primarily based on soil moisture but do not consider real-time data and crop status.

Authors of \citep{kashyap2021towards} proposed DLiSA(deep learning neural network-based IoT-enabled intelligent irrigation system for precision agriculture) that takes into account soil moisture, climate information, rainfall depth, and crop type to predict the required irrigation time.

Authors of \citep{zhang2018construction} built a water-saving irrigation system using agricultural IoT (Internet of Things). This technology would be able to process weather data in real-time. In addition, the system could make irrigation selections based on the amount of water that has been dissipated[16].

An automation system for sprinkler irrigation using a wireless sensor network is proposed[17] in \citep{nagarajan2018wireless}. The proposed system uses ZigBee and GPRS technologies to transmit and store data. The system monitors soil conditions with the help of sensors like humidity sensors, pH sensors, and temperature sensors. The data sensed by the sensors is then sent[18] to the controller for the process of monitoring.

In \citep{barkunan2019smart} proposed an irrigation system to water a plant as per the type of plant, as some crops or plants need a variable amount of water as they grow. The system begins by taking soil images with a smartphone, calculating wetness levels, and transmitting this information to the micro-controller via the GSM module. The controller then decides the irrigation duration and rate and sends the status of the field to the user's mobile phone. According to the authors, the system saves nearly 41.5\% and

13\% of water compared to traditional flood and drip irrigation methods, respectively.

In \citep{hamami2020application}, an automated irrigation system is proposed[1] to save water and improve the performance of the irrigation system. The system uses soil and weather sensors to measure soil parameters and check the weather conditions.

\section{System Model and Problem Formulation}
\label{Problem}
\subsection{System Model}
There is a field of area A which is divided into i parts of equal area $\{ A_1, A_2, ..., A_i\}$ $\in$ A . There are S soil moisture sensors $\{S_1, S_2, ..., S_i\}$ for each area $A_j \in A$. There are cameras (C) such that $\{ C_1,C_2,C_3,...,C_i\}$ $\in$ C and water pumps (W) such that $\{ W_1, W_2, ..., W_i\} \in W$.

In general, each area A_j is equipped[20] with a soil moisture sensor S_j to measure the soil moisture, a camera C_j for capturing an image of the crop, and a water pump W_j which is used[21] for irrigation $(0 \lt j \leq i)$.

For sensors S_j $\in$ S in area A_j, \{ R_1, R_2, ..., R_n\} $\in$ R are the readings taken within a time interval T which are averaged to a value λ_j.

Images captured by camera C_j are indicated by $I_j \in I $. ϕ represents the condition of the crop(droop or healthy) classified by the image I_j where ϕ $\in$ $\{0,1\}$.
For $A_j \in A$, the precipitation and precipitation probability are indicated by ρ and θ, respectively. Since all of the regions are contained [22] within a single field, the authors assume that the values of ρ and θ are the same.

\begin{table}[!ht]

\begin{center}

\caption{List of symbols}

\begin{tabular}{|c|c|}

\hline

\textbf{Symbol} & \textbf{Name} \\ \hline

A & Area \\

S & Set of soil moisture sensors \\

C & Set of cameras \\

W & Water pump \\

I & Image of plant \\

R & Readings from sensors \\

λ & Value of average soil moisture \\

ρ & Value of precipitation(in mm) \\

θ & Value of precipitation probability \\

ϕ & Crop status (0, 1) \\

Ω & Irrigation duration \\ \hline

\end{tabular}

\label{table:symbol}

\end{center}

\end{table}

\subsection{Problem Formulation}

A farmer F growing crop in an area A has to manually decide, based on his experience and the wetness of the soil, to irrigate the area A. Most of the time, manual estimation of irrigation time results in over-irrigation or under-irrigation. Farmers cannot foresee rain in advance, so heavy rain could harm the crop if the field is already irrigated[23]. Because many farmers in India have a limited quantity of water, efficient water management is also a significant concern.

As stated in Eq. \ref{eq1}, farmer F irrigates the area A_j with λ_j,ρ, θ, and ϕ_j as the average soil moisture of that area, precipitation, precipitation probability, and crop status (droop or healthy) of that area and the irrigation duration is T_x minutes.

\begin{equation} \label{eq1}

F\xrightarrow[\textit{}]{\textit{irrigates}} A_j(\lambda_j,\rho,\theta,\phi_j)

\text{ for } T_x \text{ minutes}

\end{equation}

5T_{min} indicates the minimum amount of time that the plants should be watered[24] in order for the plant to grow, and T_{max} indicates the maximum
97 duration that the plants should be watered[25] such that it does not cause damage to the crop or the soil. Waterlogging and dehydration are indicated by the letters α and β, respectively.

As stated in Eq.\ref{eq2}, water-logging (α) is caused by providing water for longer than the threshold T_{max}, whereas dehydration (β) is caused by providing water for less than the minimum time (T_{min}).

\begin{equation}\label{eq2}
(T_x > T_{max} \implies \alpha) \lor (T_x < T_{min} \implies \beta)
\end{equation}

The optimal irrigation time (Ω) must be estimated[26] for better crop conditions. According to Eq.\ref{eq3}, the optimal irrigation duration, Ω, should prevent both α and β circumstances in area A. The right value of Ω must be chosen[27] according to all the parameters ($\lambda_j, \rho, \theta, \phi_j$) for the water to be used[28] as efficiently as possible, and water wastage must be minimum.

\begin{equation}\label{eq3}
\Omega = \{ T_y \text{ } \vert \text{ }(T_{min} < T_y < T_{max}) : (\lnot \alpha \land \lnot \beta) \textbf{ for } A_j(\lambda_j, \rho, \theta, \phi_j) \}
\end{equation}

98 The goal is to compute the value of Ω for which the motor W will run to irrigate the field.

\section{Proposed methodology}
\label{methodology}

The following paragraphs describe the flow of the proposed module. The subsection \ref{device} describes the specifications of all the used devices along with their connection. Subsection

The proposed model comprises multiple soil moisture sensors, a camera, and a motor. These sensors and cameras are connected to an IoT device(Raspberry Pi) P. First, the moisture content of the soil is detected by a capacitive soil sensor S_j. Then, the soil moisture value is fed[29] into P via an I^2C bus. Moreover, precipitation (ρ) (in mm) in the past 24 hours and precipitation probability (θ) on the current day at that particular location are extracted[30] from a weather API\citep{API} as in Algorithm \ref{alg:two}. Additionally, as illustrated in Algorithm \ref{alg:three}, photos of the crops are captured, and a deep learning model (DenseNet201) is used to classify the plants as healthy or droop. Finally, the fuzzy system receives all the collected and processed data and outputs the irrigation duration Ω.

The motor is activated[31] for Ω seconds by the Raspberry Pi P. The relay switch powers off the motor after Ω seconds, and the system is programmed to sleep till the next day. Fig. \ref{fig:Flow} depicts the overall flow of the irrigation system. The aggregation of soil moisture, precipitation probability, precipitation, crop status, and fuzzy logic for the computation of irrigation time and operation of the water pump is described [32] by the algorithm \ref{alg:one}. The values of the parameters computed by Algorithm \ref{alg:one} are displayed [33] on a Raspberry Pi touch display.

```
\begin{figure}[!ht]
\centering
\includegraphics[width=0.6\linewidth, height=0.75\linewidth]{Flowchart.png}
\caption{Model flowchart}
\label{fig:Flow}
\end{figure}

\begin{figure}[!ht]
\includegraphics[width=1.0\linewidth]{setup.drawio.png}
\centering
\caption{Schematic arrangement of proposed model}
\label{fig:arrangement}
\end{figure}

\subsection{Device Specification} \label{device}
\subsubsection{Raspberry Pi Model B}
```

Fig. \ref{Raspberry Pi} shows the Raspberry Pi model B, which is used [34] for controlling all the sensors, processing the images, controlling the relay switch, and running the system. Raspberry Pi can be considered a mini-computer used for all the computation and processing. It consists of a Quad-core 64-bit 1.5GHz processor with 8GB RAM. It is powered [35] through a 5V DC via a USB-C connector (minimum 3A*). It consists of 2.4 GHz and 5.0 GHz IEEE 802.11ac wireless Gigabit Ethernet for internet access. It also includes two USB 2.0 and two USB 3.0 connectors. The Raspberry Pi Touch Display uses a 2-lane Mobile Industry Processor Interface (MIPI), Display Serial Interface (DSI) display connector to display the interface, and a 2-lane MIPI Camera Serial Interface(CSI) camera port is available to link the camera to the Raspberry Pi.[36] The Raspberry Pi has a total of 40 pins, including 26 GPIO (General Purpose Input Output) pins, eight ground pins, four voltage (3.3V and 5V) connectors, and 2 EEPROM (Electronically Erasable Programmable Read-Only Memory) pins \citep{rasp}.

\subsubsection{Capacitive soil moisture sensor}

The ATSAMD10 chip in the capacitive sensor is a built-in capacitive touch measurement device that provides a range of readings from 200 (very dry) to 2000 (extremely wet). The sensor has five pins: 3-5V power, Ground, I^2C SDA, and I^2C SCL. Fig. \ref{Moisture sensor} depicts the capacitive soil moisture sensor used for measuring the moisture content of the soil. Resistive soil moisture sensors are susceptible to corrosion, which causes measurement errors and only provides a binary output. So, the capacitive moisture sensor is used[37] to measure the soil moisture as it is corrosion-resistant and also provides a precise value of soil moisture\citep{unknown}.

\subsubsection{Camera module V2}

The images of the plants are captured[38] using a Raspberry Pi camera module version 2, as shown in Fig. \ref{Raspberry Pi Camera Module}. A Sony IMX219 8-

99

megapixel sensor is used[39] in the camera module. The Raspberry Pi camera is connected[40] to the Raspberry Pi via a ribbon wire.

\subsubsection{Water pump and Relay}

A 200 psi, 2.5A, 12V DC water pump with a flow rate of 8 liters per minute is employed[41] for water irrigation. The motor is powered[42] by an adapter that the Raspberry Pi controls. A 12 V relay module controls the water pump. The relay switch comprises NO(Normally Open), NC(Normally Closed), and Common terminal(COM). On the opposite side, it has a 5V VCC, Ground, and one signal pin \citep{relay}. The Raspberry Pi activates or deactivates the motor by triggering the relay. An AC connection uses a 12V 6A adapter to power the DC motor. The DC motor and relay switch utilized are depicted[43] in Fig. \ref{DC Motor} and \ref{Relay Switch}, respectively.

\subsubsection{Raspberry Pi Touch Display}

An interface with all of the parameters and irrigation duration would be displayed[44] on the Raspberry Pi Touch Display. The display is 7 inches in size (diagonally). 800(RGB) $\times$ 480 pixels is the display format. It has a DSI port that is used[45] to connect to the Raspberry Pi \citep{display}. The touch display shown in Fig. \ref{Raspberry Touch Display} is used[46] to display the values of all the parameters as well as the irrigation duration.

\begin{figure}[!htbp]

\centering

\begin{subfigure}{0.42\textwidth}

```
\centering
\includegraphics[width=\textwidth]{R_pi_3.jpg}
\caption{Raspberry Pi.}
\label{Raspberry Pi}
\end{subfigure}
\hfill
\begin{subfigure}{0.42\textwidth}
\centering
\includegraphics[width=0.9\textwidth]{Motor.jpg}
\caption{DC Motor.}
\label{DC Motor}
\end{subfigure}
\hfill
\begin{subfigure}{0.42\textwidth}
\centering
\includegraphics[width=\textwidth]{Relay.jpg}
\caption{Relay Switch.}
\label{Relay Switch}
\end{subfigure}
\hfill
\begin{subfigure}{0.42\textwidth}
\centering
\includegraphics[width=\textwidth]{Display.jpg}
\caption{Raspberry Touch Display.}
\label{Raspberry Touch Display}
\end{subfigure}
\hfill
```

```
\begin{subfigure}{0.42\textwidth}
\centering
\includegraphics[width=0.68\textwidth]{Sensor.jpg}
\caption{Moisture sensor.}
\label{Moisture sensor}
\end{subfigure}
\hfill
\begin{subfigure}{0.42\textwidth}
\centering
\includegraphics[width=\textwidth]{Camera.jpg}
\caption{Raspberry Pi Camera Module.}
\label{Raspberry Pi Camera Module}
\end{subfigure}
\caption{Various hardware components used for building the water irrigation
system.}
\label{fig:figures}
\end{figure}
```

```
\subsection{Weather forecast}
```

An application programming interface (API) is a service provided to an application on demand. An API key and the website's URL are required to send the request to the API. The API responds by returning a dictionary containing relevant data, from which the necessary data is retrieved.[47]

When it comes to watering plants, one of the elements to consider is the amount of precipitation (rain). If precipitation on a given day is significant enough, the irrigation period must be low. Therefore, we considered two variables: the amount of precipitation that occurred in the previous 24 hours and the probability of precipitation (likelihood of rain on a given day). RapidAPI is a weather API that returns several weather parameters such as temperature, humidity, and many more, \citep{API} from which the aforementioned values are extracted and provided to the proposed module's Fuzzy logic controller. With area, A as the input parameter, Alg. \ref{alg:one} provides the procedure for extracting precipitation and precipitation probability.

```
\begin{algorithm}
\small
\caption{API}
\label{alg:two}
\textbf{Input: $A_x$ }\\
\textbf{Output: $\rho$,$\theta$ }{}\\
\begin{algorithmic}[1]
\Procedure{API}{$I$}
\State Get an API key to use the API service.
\State Generate the request for information using the API key.
\State Receive the response from server.
\State Extract $\rho$ and $\theta$ from the response.
\State \textbf{return $\rho$, $\theta$}

\EndProcedure
\end{algorithmic}
```

\end{algorithm}

\subsection{Dataset description and processing}

The dataset consists of droop and healthy plant images. A plant is termed [48] droop when its leaf shrivels due to water deficiency. The authors developed one manually because a dataset for droop plants was unavailable. It contains 110 droop images and 125 healthy images in .png format.

When it comes to deep learning, data pre-processing is crucial as it aids in the removal of anomalies and null data, resulting in more accurate results. It is essential to remove such data because failing to do so can result in poor model training. Furthermore, the model may not be able to give generalized results due to the small dataset size. Therefore, data augmentation strategies are employed to avoid this problem. We employed data augmentation techniques such as random horizontal flipping and random rotation.
A batch size of 16 images is taken [49] while training a model. Also, the array of images is normalized using the Min-Max approach into a range of -1 to 1.

\subsection{Image classification}
Apart from soil moisture and weather conditions, the plant's appearance also plays a crucial role in optimizing the irrigation duration. For example, plant leaves sometimes turn brown or yellow when deprived of water. In addition, leaves sometimes bend because of dryness since the cells of leaves cannot

remain erect with less water \citep{smart_garden_guide_2019}. As a result, the condition of the plants is taken into account while selecting the watering time.

100 The camera captures an image of the plant, which is saved on the Raspberry Pi and classified into one of the categories. The Densenet201 deep learning model has only two possible outputs: "healthy" or "droop," which is[50] used[51] to calculate irrigation time.

```
\begin{algorithm}
\small
\caption{Image Classify}
\label{alg:three}
\textbf{Input: $I$ }\\
\textbf{Output: $0$ $or$ $1$ }{}\\
\begin{algorithmic}[1]
\Procedure{ImageClassify}{$I$}
\State Resize image to the required size.
\State Load the TensorFlow Lite model and pass the image to it.
\State The model returns the value between zero to one.
\State The received value is passed through sigmoid function which returns the
class of the image.
\State \textbf{return $0$ or $1$}
\EndProcedure
\end{algorithmic}
\end{algorithm}
```

The authors have used neural networks for image classification. Because they can use many characteristics and operate effectively even on small datasets, neural networks are commonly used[52] in machine learning and deep learning. As seen in Fig. \ref{fig:neural}, any model has several convolution layers, Max-Pooling, and activation functions that are utilized[53] for training the model. Some layers must be eliminated as the learning advances to avoid the model over-fitting. Generally, images are enormous and include too many features, and many of them are mostly redundant, so max-pooling is employed to avoid it. The neural network extracts the critical information using max-pooling, which minimizes the input size.

\begin{figure}[!ht]

\centering

\includegraphics[width=0.75\linewidth]{ Neural.drawio.png}

\caption{Neural Network}

\label{fig:neural}

\end{figure}

In recent times, deep learning has been quite effective in image classification. CNN is frequently used[54] in image classification models among several deep learning approaches such as ANN (Artificial Neural Network) and RNN (Recurring Neural Network). Various pretrained CNN models, such as MobileNet, EfficientNet, VGG16, and others, are available. Most of the CNN models are highly accurate in image classification and detection. The authors of \citep{9532697} examined multiple CNN models in plant disease detection and concluded that deep learning is the best solution for enhancing disease detection and classification accuracy. The authors of \citep{9445342} suggested hybrid models for disease detection in sunflower plants, concluding that the combination of Mobilenet and Vgg-16 utilizing ensemble learning outperformed other CNN models such as AlexNet, InceptionV3, DenseNet-121, DenseNet201, Vgg-16, and MobileNet.

101 Due to the lack of a large dataset to train our model, building a CNN model from scratch would be under-fitting. Hence, the authors employed transfer learning to classify images. As the model does not have to learn all of the features from scratch, it helps to reduce data under-fitting.

We tested most of the aforementioned models because they were all demonstrated to produce an accurate classification. We chose DenseNet201 as it delivers the best accuracy, F1-Score, and loss curve in our module. After training the Densenet201 model, we saved the model to \emph{.h5} format file. We then converted the file to \emph{.tflite} using Tensorflow Lite to deploy it on Raspberry Pi.

\subsection{Fuzzy Logic Interface}

An essential part of calculating the duration for which the motor should supply water to the plants is integrating all the factors (soil moisture content, meteorological conditions, and picture classification). The authors have used fuzzy logic to accomplish this, which produces a precise estimate of irrigation duration. The authors of \citep{chen2010study} concluded that fuzzy logic is accurate in predicting irrigation duration. The final result (irrigation duration) is calculated[55] using fuzzy logic, consisting of multiple fuzzy rules.
Soil moisture, precipitation probability, precipitation, and image categorization are provided[56] as inputs into the fuzzy logic. Fuzzification, fuzzy inference engine, and defuzzification are the three steps in fuzzy logic.

\subsubsection{Fuzzification}

This phase involves dividing or grouping the value into one of the fuzzy set's categories. Fuzzy sets are categories such as high, average, low, and many more. A "fuzzified value" is a value that is created [57] using a membership function. Membership functions are simple equations used to categorize the input value into categories from the fuzzy set when it overlaps with the function. Examples are the triangular membership function, trapezoidal membership function, and other membership functions.

\subsubsection{Fuzzy inference engine}

The fuzzy inference engine uses the fuzzified values computed in the previous phase as inputs. The fuzzy inference engine accepts two more inputs: fuzzy sets and fuzzy rules. For the established categories, fuzzy rules are defined [58]. A fuzzy rule should be defined [59] for each permutation of the fuzzy sets. Because there are three categories for soil, precipitation, and precipitation probability, respectively, and two categories for image classification, the authors created 54(3*3*3*2 = 54) fuzzy rules as shown in Table \ref{table:5}. As a result, 54 rules cover all possible category combinations. The irrigation duration is the output of the fuzzy inference engine. The output can have many values, and the third step is used [60] to combine them into a single value.

\subsubsection{Defuzzication}

The multiple values generated in the previous steps must be combined to provide a single output value that can be defuzzified. Then, the lambda cut, centroid method, weighted average approach, and other defuzzification methods are utilized[61]. Finally, a single output value or crisp output is achieved[62] after utilizing one of the approaches.

The irrigation duration is generated as a crisp output once the fuzzy logic is applied, then utilized to trigger the relay switch. Relay switches manage the circuit by allowing it to open and close as needed. Finally, the switch is turned on for the calculated duration, triggering the DC motor to water the plants.

\begin{table}

\begin{center}

\caption{Fuzzy Rules}

102 \begin{tabular}{|c|c|c|c|c|c|}

\hline

\centering

\begin{tabular}[c]{@{}c@{}}Sr.\\ No.\end{tabular} & \begin{tabular}[c]{@{}c@{}}Soil\\ Moisture\end{tabular} & \begin{tabular}[c]{@{}c@{}}Precipitation\\ Probability\end{tabular} & Precipitation & \begin{tabular}[c]{@{}c@{}}Crop\\ Status\end{tabular} & \begin{tabular}[c]{@{}c@{}}Irrigation\\ Duration\end{tabular} \\ \hline

\hline

Rule 1 & Dry & Low & Low & Droop & High\\

Rule 2 & Dry & Low & Low & Healthy & High \\

Rule 3 & Dry & Low & Normal & Droop & Normal \\

Rule 4 & Dry & Low & Normal & Healthy & Normal \\

Rule 5 & Dry & Low & High & Droop & Low \\

Rule 6 & Dry & Low & High & Healthy & Low\\

Rule 7 & Dry & Normal & Low & Droop & Normal \\

Rule 8 & Dry & Normal & Low & Healthy & Normal \\

Rule 9 & Dry & Normal & Normal & Droop & Normal \\

Rule 10 & Dry & Normal & Normal & Healthy & Low \\

Rule 11 & Dry & Normal & High & Droop & Low \\

Rule 12 & Dry & Normal & High & Healthy & Low \\

Rule 13 & Dry & High & Low & Droop & Normal \\

Rule 14 & Dry & High & Low & Healthy & Low \\

Rule 15 & Dry & High & Normal & Droop & Normal \\

Rule 16 & Dry & High & Normal & Healthy & Low \\

Rule 17 & Dry & High & High & Droop & Low \\

Rule 18 & Dry & High & High & Healthy & Low \\

Rule 19 & Medium & Low & Low & Droop & High \\

Rule 20 & Medium & Low & Low & Healthy & Normal \\

Rule 21 & Medium & Low & Normal & Droop & Normal \\

Rule 22 & Medium & Low & Normal & Healthy & Normal \\

Rule 23 & Medium & Low & High & Droop & Low \\

Rule 24 & Medium & Low & High & Healthy & Low \\

Rule 25 & Medium & Normal & Low & Droop & High \\

Rule 26 & Medium & Normal & Low & Healthy & Normal \\

Rule 27 & Medium & Normal & Normal & Droop & Normal \\

Rule 28 & Medium & Normal & Normal & Healthy & Low \\

Rule 29 & Medium & Normal & High & Droop & Low \\

Rule 30 & Medium & Normal & High & Healthy & Low \\

Rule 31 & Medium & High & Low & Droop & Normal \\

Rule 32 & Medium & High & Low & Healthy & Low \\

Rule 33 & Medium & High & Normal & Droop & Normal \\

Rule 34 & Medium & High & Normal & Healthy & Low \\

Rule 35 & Medium & High & High & Droop & Low \\

Rule 36 & Medium & High & High & Healthy & Low \\

Rule 37 & Wet & Low & Low & Droop & Normal \\

Rule 38 & Wet & Low & Low & Healthy & Normal \\

Rule 39 & Wet & Low & Normal & Droop & Normal \\

Rule 40 & Wet & Low & Normal & Healthy & Low \\

Rule 41 & Wet & Low & High & Droop & Low \\

Rule 42 & Wet & Low & High & Healthy & Low \\

Rule 43 & Wet & Normal & Low & Droop & Normal \\

Rule 44 & Wet & Normal & Low & Healthy & Low \\

Rule 45 & Wet & Normal & Normal & Droop & Normal \\

Rule 46 & Wet & Normal & Normal & Healthy & Low \\

Rule 47 & Wet & Normal & High & Droop & Low \\

Rule 48 & Wet & Normal & High & Healthy & Low \\

Rule 49 & Wet & High & Low & Droop & Low\\

Rule 50 & Wet & High & Low & Healthy & Low\\

Rule 51 & Wet & High & Normal & Droop & Low\\

Rule 52 & Wet & High & Normal & Healthy & Low\\

Rule 53 & Wet & High & High & Droop & Low\\

Rule 54 & Wet & High & High & Healthy & Low\\

\hline

\end{tabular}

```
\label{table:6}
\end{center}
\end{table}

\begin{algorithm}[!ht]
\small
\caption{Irrigation Duration System}
\label{alg:one}
\textbf{Input: $\{ R_1, R_2, ...., R_n\}$, $I$, $\rho$, $\theta$}\\
\textbf{Output: Irrigation Duration ($\Omega$) }{}\\
\begin{algorithmic}[1]
\Procedure{IrrigationDuration}{\{$R_1$, $R_2$, ...., $R_n$\}, $I$, $A_j$}
\State $\lambda$ $\gets$ ($R_1$ + $R_2$ + $\dots$ + $R_n$) $\mathbin{/}$
$n$.
\State $\rho$, $\theta$ $\gets$ $API(A_j)$ ( Alg.\ref{alg:two} )
\State $\phi$ $\gets$ \texttt{ImageClassify(I)}( Alg.\ref{alg:three})
\State According to the membership function(MF) fuzzified values of the
variables is calculated.
\State $\mu_1$ = \texttt{MF$(\lambda)$}
\State $\mu_2$ = \texttt{MF$(\rho)$}
\State $\mu_3$ = \texttt{MF$(\theta)$}
\State $\mu_4$ = \texttt{MF$(\phi)$}
\State {$\mu_1$,$\mu_2$,$\mu_3$,$\mu_4$} $\in$ $\mu$
\State Categories are established for each of the parameters :
$\lambda$($\gamma_1$,$\gamma_2$,..),$\rho$($\gamma_1$,$\gamma_2$,..),
$\theta$($\gamma_1$,$\gamma_2$,..),$\phi$($\gamma_1$,$\gamma_2$,..)
where $\gamma_1$,$\gamma_2$,.. $\in$ $\gamma$
```

\State Fuzzy rules($\delta_1,\delta_2,....,\delta_m$) $\in$ δ are established according to the categories.

\State η_1,η_2,...,η $\in$ η $\gets$

\texttt{FuzzyInferenceEngine(γ, δ, μ)}

\State Ω = \texttt{Defuzzy(η_1,η_2,...,η)}

\State Operate motor W for time Ω.

\EndProcedure

\end{algorithmic}

\end{algorithm}

\subsection{Experimental Setup}

The components are wired[63] together as indicated in Fig. \ref{fig:setup}. The Raspberry Pi can be thought[64] of as a controller that controls all other devices. A USB-C adaptor is required to power it. The moisture sensor is attached to a 4 Pin JST PH 2mm Pitch Plug. The other ends of the JST plug are attached to the Raspberry Pi's V_in, Ground, SCL, and SDA pins.

The camera module is linked[65] to the Raspberry Pi's 2-lane MIPI CSI camera interface.

The relay switch's[66] GND, VCC, and signal pins connect to the Raspberry Pi's Ground, 5V, and GPIO pin 8. The motor is connected[67] to the NC and COM pins on the relay's other side. The adaptor is linked[68] in series with the motor for the power supply. When the signal pin of the relay is set[69] to high, the motor turns on. The display is attached to the Raspberry Pi's 2-lane MIPI DSI display interface via a ribbon cable.

\begin{figure}[!ht]

\centering

\includegraphics[width=1.0\linewidth]{setup.jpg}

\caption{Connection of all devices}

\label{fig:setup}

\end{figure}

\subsubsection{Computing Facilities}

We used CoLab's GPU runtime environment to reduce the time required for all computations. Google colab has a 2.20GHz Intel(R) Xeon(R) processor, 66GB of hard drive storage, and 13GB of RAM.

\subsubsection{Hyper-parameters of model}

Hyperparameters are crucial in the training of any model. These parameters are in addition to those obtained from the dataset. Therefore, to increase the model's accuracy, it is critical to get the correct values for these parameters. The values of hyperparameters we utilized to improve the model's learning capability are shown[70] in table \ref{table:5}.

\begin{table}[!ht]

\begin{center}

\caption{Hyperparameter Tuning}

\begin{tabular}{|c|c|}

\hline

\textbf{Hyperparameter} & \textbf{Value}\\

\hline

Batch Size & 16\\

Validation Split & 0.2\\

Epochs & 30\\

Dropout & 0.2\\

Loss Function & Binary Cross Entropy\\

Optimiser & Adam\\

Learning Rate & 0.00001\\

\hline

\end{tabular}

\label{table:5}

\end{center}

\end{table}

% \subsection{Evaluation metrics}

% Accuracy (Eq. \ref{accuracy}), F1-Score (Eq. \ref{f1score}), and a graph of training and testing loss and accuracy are used [71] to evaluate the proposed system.

% \begin{equation}\label{accuracy}

% \text{Accuracy} = \frac{TP+TN}{TP+TN+FP+FN}

% \end{equation}

% \begin{equation}\label{precision}

% \text{Precision} = \frac{TP}{TP+FP}

% \end{equation}

% \begin{equation}\label{recall}

% \text{Recall} = \frac{TP}{TP+FN}

% \end{equation}

% \begin{equation}\label{f1score}

% \text{F1-Score} = \frac{2*Precision*Recall}{Precision+Recall}

% \end{equation}

% Where, \emph{TP} = True Positive, \emph{TN} = True Negative, \emph{FP} = False Positive, \emph{FN} = False Negative

% \newline

% Training and validation accuracy can sometimes differ by a wide margin, indicating that the model has been over-fitted. However, the discrepancy between training and validation accuracy should be low so that the model can correctly predict droop and healthy plants. Furthermore, as the number of epochs grows, the loss should decrease.

In this section, we evaluate the outputs of the proposed model. We observe the accuracy and loss curves of the CNN model and decide which model performs best for the given dataset. Moreover, we compare our work with other works in this domain.

\section{Results}

\label{result}

The main objective of our system is to optimize water usage along with automation which is achieved [72] by using fuzzy logic. The fuzzy logic in the proposed system requires four parameters: soil moisture, precipitation, precipitation probability, and crop status. The API directly feeds the values of precipitation and precipitation probability to fuzzy logic. The soil moisture sensor takes five readings within a time interval of one second. The average value of the readings is sent [73] to fuzzy logic. The crop image is passed [74] through a CNN model (Densenet201 in our case) whose output (droop or healthy) is passed to the fuzzy module as an input. Fuzzy logic now calculates the irrigation duration according to the provided rules\ref{table:1}.

We explored models such as Densenet201, ResNet152V2, and others for image classification. We observe the accuracy and loss curves of the CNN model and decide which model performs best for the given dataset. A comparative analysis of pretrained CNN models in terms of overall accuracy and F1 - score is presented in Table \ref{table:1} which depicts that Densenet201, InceptionV3, Resnet152V2, and Resnet50V2 obtained better training and validation accuracy over 30 epochs than other models. As accuracy is not adequate to measure a model's performance in deep learning due to the risk of over-fitting, we employed the F1- score as an evaluation metric.

The loss and accuracy curves for the four models are shown [75] in Fig.\ref{fig:graph}. There was a gradual decrease in loss and an increase in accuracy in each of the four graphs. Minimal spikes were found in the DenseNet201 model, as shown in Fig.\ref{fig:graph}, along with maximum F1-

Score, accuracy, and minimum loss. Therefore we have incorporated it into our module for image classification.

DenseNet201 uses a condensed network to create models that are simple to train and very parametrically efficient. This [76] adds variation to the next layer's input and improves performance. We chose DenseNet201 above the other three models for these reasons. The basic learning rate for the model was set [77] at 0.0001 at the outset, and adaptive moment estimation (Adam) was employed for optimization since it is simple to implement and modifies the value of the learning rate in real-time as each epoch passes. The loss function is calculated [7] using the binary cross-entropy approach, virtually perfect for binary classification issues.

Table \ref{table:2} indicates some of the soil moisture values, precipitation, precipitation probability, and image classification and corresponding output for the irrigation.

\begin{figure*}[!ht]

\centering

\begin{subfigure}[t]{0.48\linewidth}

\centering

\includegraphics[width=\linewidth]{DenseNet201.png}

\caption{DenseNet201}

\label{fig:graph1}

\end{subfigure}

\begin{subfigure}[t]{0.48\linewidth}

\centering\includegraphics[width=\linewidth]{InceptionV3.png}

```
\caption{InceptionV3}
\end{subfigure}
\begin{subfigure}[t]{0.48\linewidth}
\centering\includegraphics[width=\linewidth]{ResNet152V2.png}
\caption{ResNet152V2}
\end{subfigure}
\begin{subfigure}[t]{0.48\linewidth}
\centering\includegraphics[width=\linewidth]{ResNet50V2.png}
\caption{ResNet50V2}
\end{subfigure}
\caption{}
\label{fig:graph}
\end{figure*}

\begin{table}[!ht]
\centering
\setlength{\tabcolsep}{17pt}
\caption{Comparison of different models}
\label{table:1}
\renewcommand{\arraystretch}{1.5}
\begin{tabular}{|c|c|c|}
\hline
\textbf{Name of model} & \textbf{Accuracy} & \textbf{F1-Score} \\
\hline
DenseNet201 & 100.00 & 1.00 \\
ResNet152V2 & 100.00 & 1.00 \\
ResNet101V2 & 93.75 & 0.94 \\
```

```
DenseNet169 & 93.75 & 0.94 \\
ResNet50V2 & 93.75 & 0.92 \\
NasNetLarge & 93.75 & 0.94 \\
MobileNet & 87.50 & 0.86 \\
InceptionV3 & 87.50 & 0.83 \\
DenseNet121 & 87.50 & 0.83 \\
Xception & 87.50 & 0.83 \\
NasNetMobile & 81.25 & 0.77 \\
\hline
\end{tabular}
\end{table}

\begin{table}[!ht]
\caption{Fuzzy logic output for different conditions}
\label{table:2}
\centering
\begin{tabular}{|c|c|c|c|c|}
\hline
\begin{tabular}[c]{@{}c@{}}Soil\\ Moisture\\ $\lambda$\end{tabular} &
\begin{tabular}[c]{@{}c@{}}Precipitation\\ Probability\\ $\rho$\end{tabular} &
\begin{tabular}[c]{@{}c@{}}Precipitation\\ $\theta$\end{tabular} &
\begin{tabular}[c]{@{}c@{}}Crop\\ Status\\ $\phi$\end{tabular} &
\begin{tabular}[c]{@{}c@{}}Irrigation\\ Duration\\ $\Omega$\end{tabular} \\
\hline
550 & 20 & 20.0 & 1 & 2.87 \\
1100 & 60 & 15.0 & 1 & 0.75 \\
300 & 30 & 4.0 & 0 & 4.00 \\
```

800 & 45 & 2.0 & 0 & 5.55 \\

450 & 27 & 6.0 & 1 & 2.73 \\

900 & 52 & 4.7 & 1 & 1.4 \\ \hline

\end{tabular}

\end{table}

\section{Comparison with other work in the domain}

\label{comparison}

Some of the earlier research in this field relied entirely on soil moisture to determine how long to irrigate \citep{hadi2020iot, rivai2019drip, sayanthan2018arduino}. However, they did not take into account current meteorological conditions or forecasts for the future. After that, systems incorporating a soil moisture sensor as well as weather prediction were presented[79] \citep{velmurugan2020iot}. Furthermore, temperature and humidity were also considered by several researchers when estimating irrigation duration \citep{nagarajan2018wireless}.

In addition to considering the aforementioned characteristics, the paper's authors have also considered the crop images for computing the irrigation duration.

Table \ref{Hardware} shows the comparison of various hardware components used by authors of \citep{9696484,krishnan2020fuzzy} and the components used by the authors of this paper. The authors in

\citep{9696484,krishnan2020fuzzy} employed a resistive soil moisture sensor, which corroded with time, resulting in measurement mistakes. As a result, the system had to be re-calibrated more frequently, and the sensors had to be replaced[80]. Our research uses a capacitive soil moisture sensor to address these issues.

Table \ref{Methodology} demonstrates a comparison of the authors' methodologies of \citep{9696484,krishnan2020fuzzy} and the proposed methodology by the authors of this paper. The authors in \citep{9696484} have taken a single image, and \citep{krishnan2020fuzzy} have not captured any photographs of the plants for classification. In contrast, we have captured images of the crops every 24 hours, giving the most up-to-date crop condition and optimal irrigation duration. Furthermore, the authors in \citep{9696484} did not utilize any particular logic to combine all of the parameters generated. However, we used fuzzy logic to combine all the parameters and compute the optimum duration for watering the crops as given in Table \ref{Methodology}. The authors of \citep{krishnan2020fuzzy} use the rain sensor to detect rainfall. Rain sensors are switches that activate on rainfall and thus give binary information about whether it is raining or not. In contrast, when we use weather API, we get the precipitation of the current day(in mm) and the weather prediction for the next day.

\begin{table}[!ht]

\caption{Hardware comparison}

\label{Hardware}

\centering

```
\begin{tabular}{|c|cccccc|}

\hline

Source & Sensor & \begin{tabular}[c]{@{}c@{}}pH\\ Sensor\end{tabular} &

DHT11 & \begin{tabular}[c]{@{}c@{}}Pi Camera\\ Module\end{tabular} &

\begin{tabular}[c]{@{}c@{}}Relay\\ Switch\end{tabular} & \begin{tabular}[c]

{@{}c@{}}Rain\\ Sensor\end{tabular} \\ \hline

\citep{krishnan2020fuzzy} & Resisitive & \textbf{-} & \checkmark & \textbf{-} &

\checkmark & \checkmark \\

\citep{9696484} & Resistive & \checkmark & \checkmark & \textbf{-} &

\checkmark & \textbf{-} \\

This paper & Capacitive & \textbf{-} & \textbf{-} & \checkmark & \checkmark &

\textbf{-} \\ \hline

\end{tabular}

\end{table}


\begin{table}[!ht]

\caption{Methodology comparison}

\label{Methodology}

\centering

\begin{tabular}{|c|ccc|}

\hline

Reference & \begin{tabular}[c]{@{}c@{}}Weather\\ API\end{tabular} &

\begin{tabular}[c]{@{}c@{}}Fuzzy\\ Logic\end{tabular} & \begin{tabular}[c]

{@{}c@{}}Deep\\ Learning\end{tabular} \\ \hline

\citep{krishnan2020fuzzy} & \textbf{-} & \checkmark & \textbf{-} \\

\citep{9696484} & \textbf{-} & \textbf{-} & \textbf{-} \\

This paper & \checkmark & \checkmark & \checkmark \\ \hline
```

\end{tabular}

\end{table}

\section{Conclusion and Future research directions}

\label{conclusion}

In this paper, the authors proposed a fuzzy logic controller-based model that
103 can be used[81] to reduce water wastage and optimize the irrigation duration. Soil moisture sensors are used[82] to measure the soil moisture along with precipitation and precipitation probability extracted from a weather API and the crop status(droop or healthy) determined by using a deep learning CNN model(DenseNet201) for image classification. Various pretrained CNN models were examined[83] for accuracy and F1 score, and DenseNet201 was the most accurate, with an F1 score of 1. Farmers can use the proposed approach to automate the irrigation operation.

The future research directive will include scaling the model in a defined geographical area. The system can be configured[84] to control it remotely. Many more parameters, such as plant disease detection and notification, can be added. The amount of available water for irrigation can also be taken[85] into consideration. Furthermore, the soil's nutritional characteristics can be identified[86] to identify the type and quantity of fertilizer to apply.

In this paper, the authors proposed a fuzzy logic controller-based model that can be used to irrigate the field without any monitoring and saves human labor.

Furthermore, it focuses on the crucial requirements of the crop while saving water.

\section* {Conflict of Interest}

None declare

\section*{Data Availability}

My manuscript has no associated data

%\bibliographystyle{agsm}

%\bibliographystyle{apalike}

% \bibliographystyle{harvard}

%\bibliography{bibliography}{}

%\bibliographystyle{agsm}

\bibliography{mybib}

%\include{bibliography}

%\bibliographystyle{agsm}

\end{document}

1.	*been used*	Passive voice misuse	Clarity
2.	*etc*	Inappropriate colloquialisms	Delivery
3.	*better*	Incomplete sentences	Delivery
4.	*be addressed*	Passive voice misuse	Clarity
5.	*is expected*	Passive voice misuse	Clarity
6.	*is scheduled*	Passive voice misuse	Clarity
7.	*are used*	Passive voice misuse	Clarity
8.	~~But~~ → However, Nevertheless	Inappropriate colloquialisms	Delivery
9.	*are chosen*	Passive voice misuse	Clarity
10.	*is conducted*	Passive voice misuse	Clarity
11.	*is shown*	Passive voice misuse	Clarity
12.	*been done*	Passive voice misuse	Clarity
13.	*is proposed*	Passive voice misuse	Clarity
14.	*are interfaced*	Passive voice misuse	Clarity
15.	*is placed*	Passive voice misuse	Clarity
16.	*been dissipated*	Passive voice misuse	Clarity
17.	*is proposed*	Passive voice misuse	Clarity
18.	*is then sent*	Passive voice misuse	Clarity
19.	*is proposed*	Passive voice misuse	Clarity
20.	*is equipped*	Passive voice misuse	Clarity
21.	*is used*	Passive voice misuse	Clarity

22.	*are contained*	Passive voice misuse	Clarity
23.	*is already irrigated*	Passive voice misuse	Clarity
24.	*be watered*	Passive voice misuse	Clarity
25.	*be watered*	Passive voice misuse	Clarity
26.	*be estimated*	Passive voice misuse	Clarity
27.	*be chosen*	Passive voice misuse	Clarity
28.	*be used*	Passive voice misuse	Clarity
29.	*is fed*	Passive voice misuse	Clarity
30.	*are extracted*	Passive voice misuse	Clarity
31.	*is activated*	Passive voice misuse	Clarity
32.	*is described*	Passive voice misuse	Clarity
33.	*are displayed*	Passive voice misuse	Clarity
34.	*is used*	Passive voice misuse	Clarity
35.	*is powered*	Passive voice misuse	Clarity
36.	*The Raspberry Pi Touch Display uses a 2-lane Mobile Industry Processor Interface (MIPI), Display Serial Interface (DSI) display connector to display the interface, and a 2-lane MIPI Camera Serial Interface(CSI) camera port is available to link the camera to the Raspberry Pi.*	Hard-to-read text	Clarity
37.	*is used*	Passive voice misuse	Clarity
38.	*are captured*	Passive voice misuse	Clarity
39.	*is used*	Passive voice misuse	Clarity

40.	*is connected*	Passive voice misuse	Clarity
41.	*is employed*	Passive voice misuse	Clarity
42.	*is powered*	Passive voice misuse	Clarity
43.	*are depicted*	Passive voice misuse	Clarity
44.	*be displayed*	Passive voice misuse	Clarity
45.	*is used*	Passive voice misuse	Clarity
46.	*is used*	Passive voice misuse	Clarity
47.	*is retrieved*	Passive voice misuse	Clarity
48.	*is termed*	Passive voice misuse	Clarity
49.	*is taken*	Passive voice misuse	Clarity
50.	~~which is~~	Wordy sentences	Clarity
51.	*is used*	Passive voice misuse	Clarity
52.	*are commonly used*	Passive voice misuse	Clarity
53.	*are utilized*	Passive voice misuse	Clarity
54.	*is frequently used*	Passive voice misuse	Clarity
55.	*is calculated*	Passive voice misuse	Clarity
56.	*are provided*	Passive voice misuse	Clarity
57.	*is created*	Passive voice misuse	Clarity
58.	*are defined*	Passive voice misuse	Clarity
59.	*be defined*	Passive voice misuse	Clarity
60.	*is used*	Passive voice misuse	Clarity

61.	*are utilized*	Passive voice misuse	Clarity
62.	*is achieved*	Passive voice misuse	Clarity
63.	*are wired*	Passive voice misuse	Clarity
64.	*be thought*	Passive voice misuse	Clarity
65.	*is linked*	Passive voice misuse	Clarity
66.	~~switch's~~ → switch is	Inappropriate colloquialisms	Delivery
67.	*is connected*	Passive voice misuse	Clarity
68.	*is linked*	Passive voice misuse	Clarity
69.	*is set*	Passive voice misuse	Clarity
70.	*are shown*	Passive voice misuse	Clarity
71.	*are used*	Passive voice misuse	Clarity
72.	*is achieved*	Passive voice misuse	Clarity
73.	*is sent*	Passive voice misuse	Clarity
74.	*is passed*	Passive voice misuse	Clarity
75.	*are shown*	Passive voice misuse	Clarity
76.	*This*	Intricate text	Clarity
77.	*was set*	Passive voice misuse	Clarity
78.	*is calculated*	Passive voice misuse	Clarity
79.	*were presented*	Passive voice misuse	Clarity
80.	*be replaced*	Passive voice misuse	Clarity
81.	*be used*	Passive voice misuse	Clarity

82.	*are used*	Passive voice misuse	Clarity
83.	*were examined*	Passive voice misuse	Clarity
84.	*be configured*	Passive voice misuse	Clarity
85.	*be taken*	Passive voice misuse	Clarity
86.	*be identified*	Passive voice misuse	Clarity
87.	*documentclass[pdflatex,sn-mathphys]{sn-jnl}% Math and Physical Sciences Reference Style \documentclass[pdflatex,*	Unable to generate bibliography with Springer Latex template https://tex.stackexchange.com/questions/615138/unable-to-generate-bibliography-with-springer-latex-template	Originality
88.	*equalcont{These authors contributed equally to this work.} %\author*	xetex - Springer Nature LaTeX Template Running head is ... - Stack Exchange https://tex.stackexchange.com/questions/627375/springer-nature-latex-template-running-head-is-running-out-of-margin-in-two-colu	Originality
89.	*equalcont{These authors contributed equally to this work.} %\author*	xetex - Springer Nature LaTeX Template Running head is ... - Stack Exchange https://tex.stackexchange.com/questions/627375/springer-nature-latex-template-running-head-is-running-out-of-margin-in-two-colu	Originality
90.	*equalcont{These authors contributed equally to this work.} %\author*	xetex - Springer Nature LaTeX Template Running head is ... - Stack Exchange https://tex.stackexchange.com/questions/627375/springer-nature-latex-template-running-head-is-running-out-of-margin-in-two-colu	Originality

91.	*equalcont{These authors contributed equally to this work.} %\author*	xetex - Springer Nature LaTeX Template Running head is ... - Stack Exchange https://tex.stackexchange.com/questions/627375/springer-nature-latex-template-running-head-is-running-out-of-margin-in-two-colu	Originality
92.	*equalcont{These authors contributed equally to this work.} %\author*	xetex - Springer Nature LaTeX Template Running head is ... - Stack Exchange https://tex.stackexchange.com/questions/627375/springer-nature-latex-template-running-head-is-running-out-of-margin-in-two-colu	Originality
93.	*equalcont{These authors contributed equally to this work.} %\author*	xetex - Springer Nature LaTeX Template Running head is ... - Stack Exchange https://tex.stackexchange.com/questions/627375/springer-nature-latex-template-running-head-is-running-out-of-margin-in-two-colu	Originality
94.	*The share of agriculture in GDP was 19.*	Export curbs as a tool for price control | The Express Tribune https://tribune.com.pk/story/2324184/export-curbs-as-a-tool-for-price-control	Originality
95.	*automatic irrigation control system for efficient use of resources and crop planning.*	GSM based Automatic Irrigation Control System for Efficient Use of ... https://www.academia.edu/7575963/GSM_based_Automatic_Irrigation_Control_System_for_Efficient_Use_of_Resources_and_Crop_Planning_by_Using_an_Android_Mobile	Originality
96.	*sensor. All these sensors are interfaced with the micro-controller,*	Performance improvisation using IOT based sensor network in ... https://biomedicineonline.org/index.php/home/article/download/11	Originality

		0/77/152	
97.	*should be watered such that it does not*	How To Grow A Bonsai Orange Tree https://www.growabonsaitree.com/species/bonsai-orange-tree/	Originality
98.	*The goal is to compute the value of*	Security analysis and secure channel-free certificateless searchable public key authenticated encryption for a cloud-based Internet of things	Originality
99.	*The Raspberry Pi camera is connected to the*	Support for official raspberry pi camera https://social.msdn.microsoft.com/Forums/en-US/46570c72-9c89-477f-b220-74bc02638618/support-for-official-raspberry-pi-camera?forum=WindowsIoT	Originality
100.	*The camera captures an image of the plant,*	News Bureau | ILLINOIS https://news.illinois.edu/view/6367/801583	Originality
101.	*Due to the lack of a large dataset to*	Functional Map of the World https://arxiv.org/pdf/1711.07846.pdf	Originality
102.	*begin{tabular}{|c|c|c|c|c|c|} \hline*	32. For the following data, calculate gross reproduction rate. \begin ... https://www.sarthaks.com/2839689/following-calculate-reproduction-tabular-hline-female-population-female-births-tabular	Originality
103.	*Soil moisture sensors are used to measure the*	Soil Moisture Sensor Library For Proteus - The Engineering Projects https://www.theengineeringprojects.com/2020/07/soil-moisture-sensor-library-for-proteus.html	Originality