AN INTELLIGENT DEVICE TO ASSIST IN PLANT GROWTH USING ARTIFICIAL INTELLIGENCE AND MOISTURE SENSING

Rena Cao [1], Tyler Boulom [2]

[1] University High School, 4771 Campus Drive, Irvine, CA 92612
[2] California State Polytechnic University, Pomona, CA, 91768

ABSTRACT

Houseplants have long held cultural and personal significance, first gaining popularity among the Victorian middle class as symbols of morality and status. Today, over half of American households own at least one houseplant, yet many struggle to keep them alive—on average, households have killed seven plants, with nearly half of plant owners expressing concern about their plant's survival. Despite this, houseplants offer numerous benefits, from improving air quality and acoustics to enhancing mental health, productivity, and social interaction. EcoMonitor addresses the common challenges of plant care, particularly the issue of overwatering, by using AI and soil moisture sensors to assess plant health and alert users through a color-coded LED system. This system provides timely, intuitive reminders that adapt to environmental conditions, unlike generic care guides. By keeping the user actively engaged in care tasks, EcoMonitor promotes both plant health and personal satisfaction. Experimental testing validated the accuracy of the AI classification for different watering conditions and evaluated the effectiveness of LED feedback, showing strong potential for reducing plant mortality while preserving the rewarding experience of plant care.

KEYWORDS

Plant Care, AI Monitoring, Overwatering Prevention, Smart Gardening

1. INTRODUCTION

Houseplants initially became popularized by the middle class in the Victorian Era, where it was seen as a symbol of moral value and status [1]. As society has advanced, such resources are more available to the public and now, over half of American households have at least one houseplant. However, the average household has also killed 7 plants and 48% of citizens are worried about their plant's survival. Houseplants have shown to remove airborne toxins in homes (up to 87% per 24 hours), raising relative humidity to healthier and more comfortable levels and show many positive impacts to the mental health of its caretakers, including relieving stress and increasing productivity at work [2][4]. Raising plants has also been shown to increase social interactions and promote inclusion within the plant community [5]. By forming connections and bonding over a shared interest or hobby, people feel less isolated within their homes and are more connected to others. Those with plants have also been shown to take fewer sick days off of work, as it promotes restoration [6]. Raising plants reinvigorates people by making them feel accomplished, as they succeeded in a difficult task that requires a lot of time and determination. As with any arduous task, raising plants successfully is something to be celebrated and can bring triumph and increase in self-worth. Furthermore, plants can also affect the acoustics of a room, reducing

David C. Wyld et al. (Eds): CSIT, AMLA, IPDCA, NLPA, AIS, IPPR, SPTM – 2025
pp. 173-182, 2025. CS & IT - CSCP 2025 DOI: 10.5121/csit.2025.151413

mostly high frequency noises in rooms with hard materials [3]. Essentially, the leaves can interfere with the sound waves that come from loud noises and act as barriers that soften harsh noises. Overall, the enormous benefits of raising plants, such as reduced levels of stress, better air-quality and increased productivity make raising a plant very worth the time and effort. However, still many people struggle with the upkeep and time it takes to take care of yet another living thing in their household.

EcoMonitor aims to solve this problem by providing reminders to water the plant depending on the current health of the plant. It uses AI to determine the health of the plant while also using a plant monitor to determine the moisture in the soil, to determine if it's overwatered, healthy, or underwatered. This notifies an LED that shines different colors depending on the health of the plant, a bright reminder to those who wish to keep their plant alive. This is an effective solution because overwatering the plant is often even more common than underwatering the plant as the leading cause of deaths, due to the overcautious actions of an owner that wants to keep their plant alive. Therefore, by alerting the owner specifically when the plant must be watered, EcoMonitor completely solves the issue of overwatering, compared to other generic solutions such as just feeling the soil, as the top may not directly affect the whole soil and many people often do not want to get their hands very dirty inside. By preventing both overwatering and underwatering, it's far more difficult for the houseplant to die, as diseases aren't nearly as large of a factor in the death of houseplants, and even a child knows when it's time to water a plant. This is also more reliable than following any generic plant directions, as the environment around plants can largely affect how they must be raised. A plant in a very humid home needs much less water than a plant in a very dry home, so accounting for these differences allows for much healthier plants. Additionally, by encouraging users to do all the "hard work", they still leave the experience with a feeling of achievement that cannot be created through a device that waters plants independently of the user. Those who invest in EcoMonitor also gain the experience of actually taking care of the plant, which is a feeling like no other.

Through the experiments, EcoMonitor aimed to determine the accuracy of the AI algorithm, along with the effectiveness of the colors and patterns of the LED lights. The first experiment dives into the AI accuracy by testing EcoMonitor over a period of time with an overwatered plant, a healthy plant, and an under-watered plant, running the algorithm through each plant regularly to see if the result changed, along with the confidence rating.

2. CHALLENGES

In order to build the project, a few challenges have been identified as follows.

2.1. Training AI Without Existing Data

One major challenge while developing EcoMonitor was the data set used to create and train the AI algorithm. To develop the model, we initially attempted to find an existing database of assorted houseplant images, sorted into healthy, over-watered, and under-watered. Unfortunately, no dataset seemed to appear for the hydration of houseplants (eg. aloe, spider plants, etc). Ultimately, we manually found images of healthy, overwatered, and underwatered plants, which we formatted into a uniform file type and pixel size, which we then input into the teachable as training images for our AI engine, which is TensorFlow. We found over 100 images for each category, however, in the future we would like to expand on the training data found so we can create a more accurate and consistent model for EcoMonitor. Furthermore, we acknowledge that pulling data from the internet is bound to have inconsistencies and errors, such as mislabeled pictures, however each picture was manually compared to ensure similarity between each category and correct labeling of each dataset.

2.2. Enabling Real-Time Data with Firebase

Another challenge we faced was having to go through a third-party database to store the information gathered by the AI and input it into the app. The data is gathered through the raspberry pi, which holds the AI engine, and had to be taken and moved to the app so users could see the information from their mobile device. We decided to use the firebase database to store the information and transfer it to our app. Initially, our app was being programmed using Flutterflow, which is a version of Flutter (an app making program) that was less code intensive. However, because of the lack of coding ability, it has less capabilities and could not connect to the firebase database that we needed to use to transfer data to the app. Therefore, we changed programs to Flutter, which even though it was more code intensive, allowed for information to be directed from firebases to appear in the app for real-time changes. This is because Flutter is far easier to troubleshoot if there are any issues with source code and can also easily implement new features, along with better performance in comparison to Flufferflow.

2.3. Optimizing Camera Placement for Plant Imaging

The last challenge we faced was positioning the camera at an ideal height to allow for a clear picture of the plant. This way, images could be sent to the AI engine to determine the overall health of the plant based on external images. Overwatered and underwatered plants often look very similar, yet require vastly different methods to fix the problem, so it's necessary to be able to differentiate between the two. To solve this issue, a camera stand was modeled using CAD (computer aided design) and then 3D printed to fit our exact plant. The measurements of the camera were taken and input into the design so that a cubby would perfectly fit the shape. Future development may include making a stand that can vary sizes depending on the heights of various plants, as this stand can only account for a standard height. Other stands may also be more flexible for adjustment purposes or easier to hide from the naked eye as well.

3. SOLUTION

The EcoMonitor system utilizes components like the raspberry pi, AI algorithm, and app. These subunits work together in order to transfer information from the raspberry pi, which utilizes the AI algorithm, to the app, where the user can interact with the real-time data that is provided. The raspberry pi connects to the camera, LEDs, and plant monitor sensor. The camera sends information to the AI engine, which categorizes the plant shown into underwatered, healthy, or overwatered. It’s mounted onto a 3D printed stand that adjusts depending on the height of the plant to get a clear view for the AI. The LEDs change color based on the health of the plant, detected by both the plant monitor and the AI engine. Red light stands for underwatered, green for healthy and blue for overwatered. This can alert the plant owner to adjust how to care for the plant based on the current health. The plant monitor detects the moisture of the plant’s soil, which it sends as a percentage to the raspberry pi program to sort and decide its health. After gathering the data, the raspberry pi sends the information into a firebase server that can store and upload the data to the app [15]. The app is created using flutter and coded with dart, a mix of python, java, and C#. The app has a splash screen with EcoMonitor’s logo, then a plant selection screen loads a few seconds later. The screen allows the user to choose which plant they have so that EcoMonitor’s data has a comparison of what moisture levels the plant should have. Specifying specific plants will allow the user to accurately measure their plant’s hydration level. After selecting a plant, a screen with the plant’s current health, the date of last watering, a place to set a reminder for the next time to water, and plant tips specifically for the plant chosen. This allows the user to see the moisture level of the plant along with what the AI classifies the plant as, along with alarms to water. The firebase uploads the data here so the user can see the current statistics of the plant, including the soil's moisture and humidity in real time.

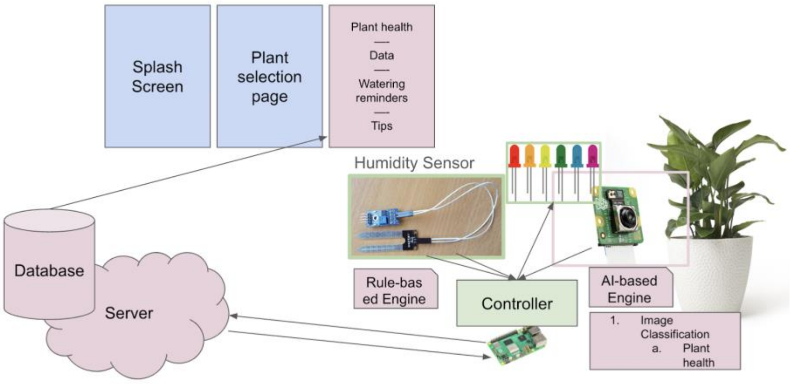

Figure 1. Overview of the solution

One of the most integral parts of EcoMonitor is the raspberry pi, which controls all of the hardware in the system. It controls the camera, LEDs, and plant monitor through Python. On the software aspect, it also utilizes the AI engine, the TensorFlow, which is mainly focused on creating models and has a comprehensive selection of resources and libraries that assist in model creation. In comparison to popular engines like ChatGPT, Tensorflow is better suited to create new models and has a wider range of different models that can be created and used, whereas ChatGPT is mainly focused on human-sounding generated text [14]. OpenAI gives responses on what it thinks about a subject matter whereas TensorFlow where the user tells it what to think. The AI engine uses a camera to look at the plant and come up with a conclusion regarding the health status of the plant based off of a large database of images, which it then transfers to the firebase database and then to the app.

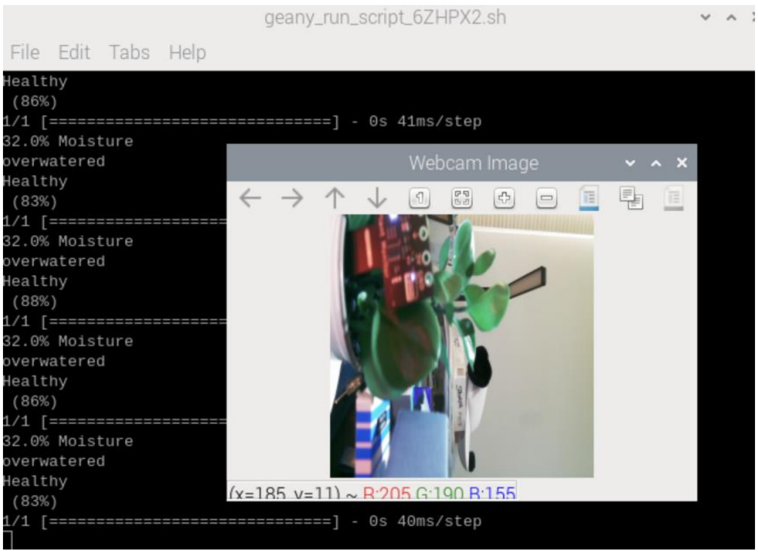

Figure 2. Screenshot of webcam image

In order to control the LED strip, the raspberry pi must first read and display the health of the plant, which is found from the plant monitor by reading the moisture level of the plant [13]. It determines the health of the plant based on the percentage of moisture within the soil, where any plant's soil less than 15 percent moisture is underwatered, anything between 15 percent and 30 percent is healthy, and anything with over 30 percent moisture is overwatered, where the percentage is the amount of water within the soil. After determining the health status of the plant through this, it can control the LEDs using the exact same method, changing the colors of the LED strip depending on the moisture level depicted by the plant monitor. This data also gets sent to the firebase database and ends up in the app, along with the data from the AI engine that also works through the raspberry pi. Using the camera that is connected to the raspberry pi and our tensorflow AI engine, we can detect what the plant looks like externally and determine a very good guess on the health of the plant, based on defining features like the colors of the leaves and the shape of them. After determining the health of the plant, the data is also sent off to the firebase.

The AI model we designed determines the health of houseplants based on its external appearance. Trained from a curated data set specifically made for EcoMonitor due to a lack of existing databases, the AI software analyzes the plant it is being shown from a camera using a convolutional neural network, identifying whether the plant looks closer to being overwatered, underwatered, or healthy [12].

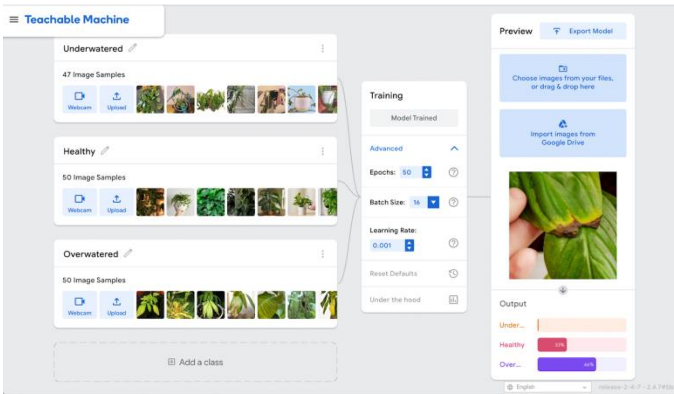

Figure 3. Screenshot of the website 1

The code formats the images being utilized from the raspberry pi camera and resized so that the system can read all the images. It changes them all to be 224 pixels by 224 pixels. This is important because large images can now be processed efficiently, meaning it's easier to process and has better consistency. Afterwards, the image is turned into a numpy array, meaning that it is turned into a bunch of numbers so that the computer can read it, as the computer cannot understand images. For example, humans can understand images based on their own thoughts and experiences, to come up with a conclusion of what our eyes see. However, the computer doesn't have its own thoughts and cannot come out with a conclusion out of an image, as it cannot determine any identifying features of the plant. It can only learn off of what it already knows, and it is very difficult for the AI to recognize every image ever created, especially when feeding the engine new images. Afterwards, it uses the model to make predictions about the plant's classification, while also printing a confidence score for the plant's class. The confidence score is the level of reliability of the data, or how certain the AI is that its answer is accurate. A lower confidence score means the AI isn't completely confident in its answer, that it doesn't believe the image is based on the prebuilt classes. A higher confidence score is better because it means the AI is very confident in its answer. If the AI inaccurately identifies an image with a very high confidence score (above 50%), then we know something needs to be fixed because the AI is very wrong and cannot recognize its mistake. This type of model is a convolutional neural network, or CNN, as it is looking at images and classifying them based on knowledge and training from prior images and data.

The app allows users to track their plant health and determine when to water the plant based on the information on the app. The app allows for easier access to the data provided on the plant monitor, as it can be seen just by opening the app on either a mobile device or computer. Using the plant monitor requires opening up multiple apps and can often be really slow depending on the device used, not to mention it can only be seen on a computer which isn't convenient for most people. Physically checking on the plant is much more inconvenient because many people want to be able to have access to their plants while they are away from home, whether they are at work or on vacation. EcoMonitor allows for seamless monitoring of plants no matter the distance, as the digital platform allows for convenience. By signposting when the plant is dehydrated or drowning, the user can better adapt to the status of the plant and make sure the issue doesn't cause long-term problems or death.

Figure 4. Screenshot of the website 2

The code changes the colors of the text depending on the moisture level of the plants. If the percentage of hydration is less than or equal to 30%, the words are depicted in red. Similarly, if the percentage of hydration is less than or equal to 70%, the words are depicted in yellow. If the plant is completely hydrated, then the words are depicted in green to show that it has very recently been watered and doesn't need any more water. Below, it shows the status of the plant. If the system believes the plant is underwatered, it will label underwater in red. If the system believes the plant to be overwatered, it will be yellow. Lastly, if the plant is deemed healthy, it will return a green color [11]. The use of colors makes the app more vibrant while making the words easier to understand and more simplistic for the user to utilize.

4. Experiment

4.1. Experiment 1

A possible blind spot in EcoMonitor is the AI model's accuracy. It's important that the AI model is accurate because it determines the amount of water that the plant requires to stay alive. If the AI incorrectly assumes the health status of the plant, then the user may accidentally over water the plant, leading to it drowning, or underwater the plant, resulting in it drying out.

The experiment will occur with 3 identical plants. For the course of a few weeks, one will be watered twice a day, one will be watered once every 3 days, and one will be watered once every 2 weeks. All three plants will receive the same amount of water per watering. This way, the variables of the test are a healthy plant, an overwatered plant, and an under-watered plant. All three plants will sit at the same place with an equal amount of sunlight and the exact same type of soil. Afterwards, the AI will test each individual plant to determine whether it classifies them as the accurate category or a different one, determining the accuracy of the model along with the accuracy of the confidence rating given by the model. This experiment was then repeated again for various other types of plants to validate the accuracy of the model.

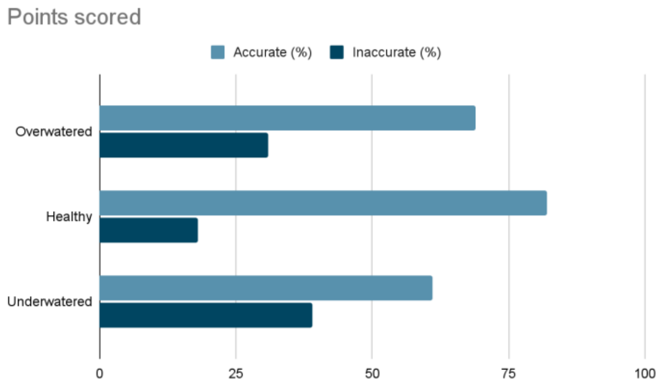

Figure 8. Figure of experiment 1

The double bar graph depicts the accuracy rate of the AI for identifying each type of plant when presented. The light blue represents how often the engine got the categorizing correct, while the dark blue represents how often the AI guessed one of the other two categories. The average accuracy for overwatered plants is 69%, the average accuracy for healthy plants is 82%, and the average accuracy for underwatered plants is 61%. By far, the highest accuracy is for healthy plants, as they were the easiest to identify. Healthy plants are much less likely to have any deformities or notable features that make it stand out from overwatered or underwatered plants, which can often look similar and have similar features, such as yellowing leaves. The lowest accuracy was for underwatered plants, with overwatered plants following closely behind. This is because they look pretty similar in their features oftentimes, making it difficult for the AI engine to determine the difference between the two. If this experiment had a larger dataset, then the gap between accuracies for underwatered and overwatered plants would most likely decrease, as the variation within the data should also decrease.

4.2. Experiment 2

The second experiment we will conduct is determining how effective LEDs are to convince users to water their plants. Through this, we can determine if various bright colors and different patterns are persuasive enough for users to water their plants.

We will provide EcoMonitors to a group of people for the duration of 3 weeks. During this time, they will take care of the plants based on EcoMonitor's reminders, and the system will keep track of how long of a duration between when the light turns a specific color or pattern and the user's response of watering the plant. Three trials will be held, one with solid red lights, as it is the most alarming color to express response, one with blinking red lights to gather attention, and one with alternating red and blue lights to signify urgency, like emergency sirens. Users will also be surveyed to determine which colors were the most and least bothersome to look at, to take into consideration user opinion on varying colors.

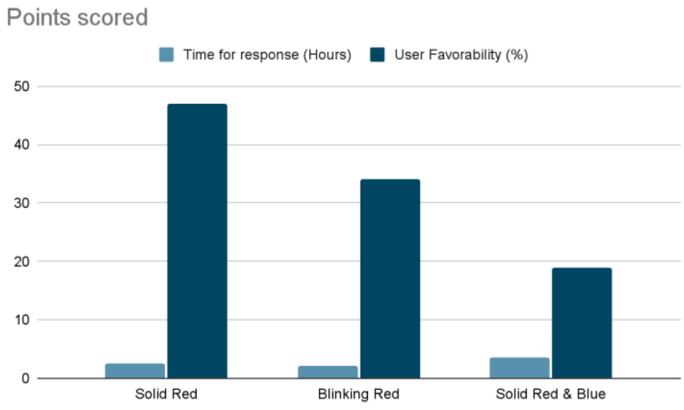

Figure 9. Figure of experiment 2

The double bar graph illustrates the effectiveness and favorability of various combinations of lights and motions in order to determine which one is best to alert the user. Effectiveness wise, blinking red lights had the best effectiveness, as they had the shortest response time amongst many trials. Solid red lights followed in second with solid red and blue being the least effective, due to the calming effect of blue. On the other hand, the combination with the best favorability over all the testers was the solid red lights, followed by blinking red, and then solid red and blue. The solid colors were most pleasing on one's eyes without being irritating or annoying if the user is busy, compared to the repeated flashing of the blinking lights. The complete red is also better looking than a mix of red and blue, which looks more purple from a distance and doesn't effectively notify the user of something urgent.

5. Related Work

A study explores the implications of various "environment parameters like temperature, humidity and light intensity" through monitoring the light, temperature and moisture absorbed by the plant using an IoT based system [8]. While this solution takes into account more variables and measures more data, the system's maximal communication range is merely 15 meters, leading to the device only being useful in close proximity. Once someone is too far away from the system, the freedom of using the cloud is essentially useless. On the other hand, EcoMonitor utilizes VNC viewer's connection with a raspberry pi. which only requires a local network and has no physical distance limitation, allowing for more freedom. So, if the user wants to check on the plant from somewhere farther away (such as another room in their house), they are able to, instead of being right next to the plant, in which the user can clearly see the physical plant.
A different study discusses the use of a smart plant monitoring system where "according to the moisture of the soil the water pump supply to the plant will be controlled" [7]. The system uses information gathered through moisture and light sensors to water the plant independent of an actual human, allowing for less responsibility for people to water the plants themselves. While this device can be more useful to some, the design doesn't provide the actual experience of raising a plant that EcoMonitor provides, as the accomplishment of watering one and taking care of one personally provides numerous mental health benefits to an individual.

One investigation delves into various diseases by using a device that "sends images of plants to classify diseases and updates environmental parameters like air temperature, humidity, soil moisture and pH" [9]. The quick and accurate detection of various plant diseases is helpful for large-scale agricultural use; however, it isn't convenient for small-scale plants. Furthermore, the testing of diseased plants is very tricky because one must first acquire the disease, and it isn't very likely to happen to houseplants. The database of diseases would have to repeatedly be run on all the plants to accurately detect diseases to avoid scanning the wrong plants, and the accuracy would have to be very accurate, in case of using the wrong pesticides or treatments. Therefore, it would barely be used in the device. Furthermore, their software only utilizes 50 images per category, with 25 being used for training and the other 25 being used for testing, which is an extremely small amount for an AI algorithm and will definitely lead to inaccurate and inconsistent results.

6. Conclusions

Some limitations to EcoMonitor include the small database that it draws upon to determine the plant's health. Due to the reason that there were no previous databases of different plant healths, a new one had to be created for the AI engine to draw upon, leading it to be smaller than preferable and consisted of images found off of the internet, which could end up being inaccurate [10]. Each category of underwatered, healthy, and overwatered consisted of slightly more than 100 images, whereas the ideal algorithm would consist of 100+ images. Additionally, a future addition to the project could include a live video feed of the plant or at least a few images being shown each day to keep track of the plant and how it develops over time by tracking its growth, even when the user is away or unable to take care of the plant. This way, if there is any issue with the plant or the system, the user can easily tell and find a solution rather than not knowing that there's a problem. Improvements to the AI could include having custom tips to take care of each plant created by AI.

Ultimately, EcoMonitor provides one-of-a-kind feedback and assistance towards raising plants and allows for convenient plant growth by doing the heavy lifting while avoiding doing all the work, saving the enjoyable process for the user. The AI algorithm paired with the app and LED lights work together to create a cumulative end product that assures a healthy plant.

References

[1] K. S. Nemali and M. W. van Iersel, "Light intensity and fertilizer concentration: I. Estimating optimal fertilizer concentrations from water-use efficiency of wax begonia," HortScience, vol. 39, no. 6, pp. 1287–1292, 2004.

[2] L. B. Yeo, "Psychological and physiological benefits of plants in the indoor environment: A mini and in-depth review," Int. J. Built Environ. Sustain., vol. 8, no. 1, pp. 57–67, 2020, doi: 10.11113/ijbes.v8.n1.597.

[3] V. I. Lohr, "What are the benefits of plants indoors and why do we respond positively to them?," in Proc. II Int. Conf. Landscape Urban Horticulture, vol. 881, 2009.

[4] J. Ma, "Interaction with nature indoor: Psychological impacts of houseplants care behaviour on mental well-being and mindfulness in Chinese adults," Int. J. Environ. Res. Public Health, vol. 19, no. 23, p. 15810, 2022.

[5] M. de Seixas et al., "Horticultural therapy in a psychiatric in-patient setting," BJPsych Int., vol. 14, no. 4, pp. 87–89, 2017.

[6] T. Bringslimark, T. Hartig, and G. G. Patil, "Psychological benefits of indoor plants in workplaces: Putting experimental results into context," HortScience, vol. 42, no. 3, pp. 581–587, 2007.

[7] A. Rane et al., "Design of an IoT based smart plant monitoring system," in Proc. 2022 10th Int. Conf. Emerg. Trends Eng. Technol.–Signal Inf. Process. (ICETET-SIP-22), IEEE, 2022.

[8] S. Siddagangaiah, "A novel approach to IoT based plant health monitoring system," Int. Res. J. Eng. Technol., vol. 3, no. 11, pp. 880–886, 2016.

[9] M. I. Pavel et al., "An IoT based plant health monitoring system implementing image processing," in Proc. 2019 IEEE 4th Int. Conf. Comput. Commun. Syst. (ICCCS), IEEE, 2019.
[10] M. van Lent and J. Laird, "Developing an artificial intelligence engine," in Proc. Game Developers Conf., 1999.
[11] S. R. Brazel et al., "Overwatering may be as detrimental as underwatering in container-grown kale (Brassica oleracea L. acephala)," Sci. Hortic., vol. 316, p. 111961, 2023.
[12] F. K. Sufi, "AI-GlobalEvents: A software for analyzing, identifying and explaining global events with Artificial Intelligence," Softw. Impacts, vol. 11, p. 100218, 2022.
[13] A. Meschtscherjakov et al., "ChaseLight: Ambient LED stripes to control driving speed," in Proc. 7th Int. Conf. Automot. User Interfaces Interact. Veh. Appl., 2015.
[14] A. Kashefi and T. Mukerji, "ChatGPT for programming numerical methods," J. Mach. Learn. Model. Comput., vol. 4, no. 2, 2023.
[15] C. Khawas and P. Shah, "Application of Firebase in Android app development—a study," Int. J. Comput. Appl., vol. 179, no. 46, pp. 49–53, 2018.