Development and Power Characterization of an IoT Network for Agricultural Imaging Applications

Jacob D. Wahl and Jane X. Zhang
Electrical Engineering Department, California Polytechnic State University, San Luis Obispo, USA
Email: jakedwahl@gmail.com, jzhang@calpoly.edu

***Abstract*—This paper describes the development and characterization of a prototype IoT network for use with agricultural imaging and monitoring applications. The sensor and gateway nodes are designed using the ESP32 SoC with integrated Bluetooth Low Energy 4.2 and Wi-Fi. A development board, the Arducam IoTai ESP32, is used for prototyping, testing, and power measurements. Google's Firebase is used as the cloud storage site for image data collected by the sensor. The sensor node captures images using the OV2640 2MP camera module and transmits the image data to the gateway via Bluetooth Low Energy. The gateway then uploads the collected images to Firebase via a known nearby Wi-Fi network connection. This image data can then be processed and analyzed by computer vision and machine learning pipelines to assess crop growth or other needs. The sensor node achieves a wireless transmission data throughput of 220kbps while consuming 150mA of current; the sensor sleeps at 162µA. The sensor node device lifetime is estimated to be 682 days on a 6600mAh LiPo battery while acquiring five images per day based on the development board power measurements. This network can be utilized by any application that requires high data rates, low power consumption, short range communication, and large amounts of data to be transmitted at low frequency intervals.**

***Index Terms*—IoT, smart farming, ESP32, Bluetooth low energy, firebase cloud**

I. Introduction

The agricultural industry as a whole is facing significant challenges, from the depletion of natural resources, labor shortages, climate change, a shift in consumer preferences for sustainability, and the rising world population. These challenges have led to more innovation and investment capital put into modern agriculture than ever before [1]. As a result, smart farming through the use of different complex and connected technologies have been employed to assist the industry enhance operational efficiency, improve productivity, profitability, and maintain sustainability [2]. Primarily, the use of Internet of Things (IoT), Computer Vision (CV), and Machine Learning (ML) are being used together to develop intricate pipelines to keep farmers knowledgeable about their farms, crops, and yields [3], [4].

Manuscript received December 15, 2020; revised May 24, 2021.

One specific application of using the combination of IoT, CV, and ML is to build a remote crop monitoring and yield prediction system. The crop monitoring system would provide farmers with easy access to valuable information regarding their plants' growth conditions, and thus support farmers' decisions about crop production. Additionally, yield predictions prior to harvest dates is essential for improving product management and marketing plans. It would help farmers monitor their fields throughout the year and allow flexibility in hiring an appropriate number of laborers when they will be needed. Misestimating harvest times and labor requirements result in large fiscal losses as well as significant crop waste in the field due to untimely harvests.

This paper proposes the design and development of a custom low-cost and low-power IoT system to be used as the sole data collection step in a crop detection and yield prediction pipeline. Primary data about growth and health of the crops are captured by visual sensors placed in the field. Additional data from sensors that measure ambient temperature, humidity, soil health, and sun exposure can also be included. This research aimed to build a standalone prototype system with scalability in mind.

The rest of the paper is organized as follows: Section II discusses various related work that motivated this research. Section III provides details of each component of the system developed at a high level. Section IV shows the results of power characterization testing for significant system functionalities. Section V discusses the node characteristics and battery life estimates for specific scenarios. Finally, Section VI and VII conclude the paper and offer future research directions.

II. Related Work and Motivation

Significant work has already been done in the computer vision and machine learning sections of the previously mentioned pipeline, leaving IoT open to be explored. This section addresses issues brought up in two previous works focused on CV and ML that became the motivation behind this research to develop an IoT system. Two other related systems are described briefly.

The first work, *Towards a Strawberry Harvest Prediction System Using Computer Vision and Pattern Recognition* by Apitz attempted to develop a data acquisition, computer vision, and prediction pipeline to

doi: 10.12720/jait.12.3.214-219

reliably predict the yield of the field under test. The research also explored three different methods for data collection: manual handheld camera, mounted camera, and quadcopter drones. Manual and drone methods were used for his testing, though it was noted that the mounted camera method could be the best option (but failed to test it as it required extensive design). From these methods, Apitz recorded many shortcomings in image collection. The manual handheld camera method was time consuming and incredibly inconsistent. Changes in viewing angles, illumination levels, and acquisition frequencies (ranging from once a day to once a week) all contributed to more difficult computer vision development. The drone method was quick but was unable to capture images closer than about 10 meters over the ground using autopilot, and this altitude made it impossible to detect strawberries accurately and reliably [3].

The second work, *Strawberry Detection Under Various Harvestation Stages*, by Fitter [4] developed various computer vision techniques to detect strawberries at various stages in their growth cycle. The work used a dataset consisting of 600 manually collected images taken twice a week during the entire season. It has been noted that while the detection accuracy was decent, it could be improved with a larger dataset. More advanced machine learning algorithms using neural networks were not attempted due to lack of data.

A related work, called Vinduino [5], was created by a California farmer to better manage the irrigation system of his vineyard. The project uses multiple soil moisture sensors, located at different depths to prevent overwatering and control irrigation to not exceed the active root zone. The solar powered remote sensor nodes have three gypsum soil moisture sensors and several options for temperature or humidity sensors. The designed board includes a Globalsat LM-210 LoRa module for long-range wireless communication (up to 6 miles), a built-in solar battery charger, and a built-in real time clock (RTC) for precise irrigation timing. This board also has the option to plug in an ESP8266 SoC to provide Wi-Fi connectivity. The functionality of the sensor node is controlled by an Arduino Pro Mini. The data collected is routed to a ThingSpeak cloud server for basic analytics and visualizations.

The second related work, *Long-Range & Self-powered IoT Devices for Agriculture & Aquaponics Based on Multi-hop Topology*, presented a prototype design of a long-range, self-powered IoT device for use in precision agriculture [2]. The system is designed to collect temperature, humidity, light, air pressure, soil acidity, and soil moisture data and transmit it back to an IoT server. The sensor nodes are designed with BMD-340 modules based on the nRF52480 microcontroller with an on-board antenna. The controller has Bluetooth 5 long-range support integrated into the chip. They use the bq25570 integrated circuit from Texas Instruments for energy harvesting, battery charging, and voltage conditioning. The nodes are powered from a Li-Ion battery with a 120mAh capacity. The sensor nodes are programmed to transmit at 125 kilobits per second in long-range mode using Forward Error Correction (FEC) scheme to perform error detection and correction on received data. To obtain sensor coverage of almost any size, a custom designed multi-hop network is used. Once data arrives at the gateway node, the messages are published via MQTT over Wi-Fi to a local network server.

The prototype system described in this work is developed to remedy specific shortcomings from these previous methods, namely: inconsistencies of acquisition frequency, varying illumination levels, varying viewing angles and heights, as well as a time-consuming collection process [3], [4]. An IoT system as proposed can be programmed to acquire crop images at any set frequency, even at specific times of day (morning, midday, and evening for example). This eliminates time inconsistencies from manual image collection caused by human error or forgetfulness and creates a uniform dataset for complex processing models to analyze. By fixing exact times to capture crop images, the problem of varying illumination levels caused by time of day is mitigated. When the sensor node is physically deployed, it will stay in a single stationary location to monitor a specific area of crops. This alleviates irregularity produced by varying viewing angles and heights caused by human image collection. The sensor node is designed to be as low power as possible to extend battery lifetime. This extension makes this system almost fully autonomous and only requires human input every year or more to change batteries. Compared to manual human image capture, this system could easily save an estimated few hours in time per day for a farmer.

III. System Level Design

A visual representation of the data flow in the prototype IoT system is presented in Fig. 1. Each block is described in more detail in this section. Typically, an IoT ecosystem consists of four major components: IoT devices, communication technology, the internet, and data storage [1]. The key advantage to an IoT system is communication in remote and inaccessible areas, making it an emerging technology for farms across the world. According to a 2016 report, the number of IoT connected agricultural devices is expected to grow from 13 million at the end of 2014 to over 225 million in 2024, an increase of 1795% [6].

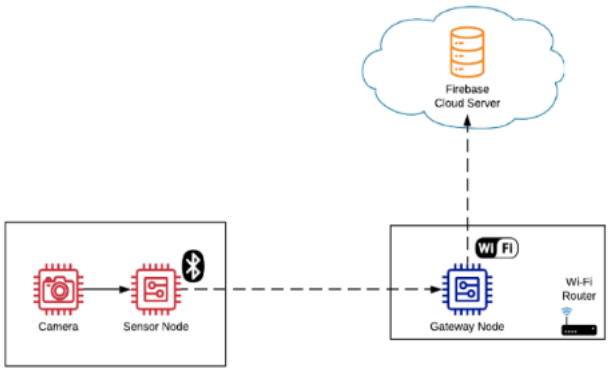

Figure 1. High level IoT system overview.

A. Sensor Node

The sensor nodes are designed with the ESP32 SoC, a module that integrates a dual-core Tensilica Xtensa LX6 microprocessor, Wi-Fi, and Bluetooth 4.2 capabilities on the same chip. The ESP32 also holds an ultra-low power co-processor and an RTC for low current consumption with five different sleep modes [7].

The development board chosen to develop the prototype system is the Arducam IoTai ESP32, seen in Fig. 2. This development board adds a 2-pin battery connector, a USB programming port, a microSD card reader, a PCB antenna, and a parallel camera interface for only $20.

The sensor node is designed with the OV2640 2-megapixel camera module to capture images [8]. The images are JPEG compressed and stored to on-board memory (flash or microSD) while waiting for transmission. Sensor nodes communicate with gateway nodes using the Bluetooth Low Energy protocol. The sensors are programmed to transmit image data using a 1Mbps bitrate according to the BLE v4.2 standard. Throughput is maximized to spend as little time transmitting as possible to reduce energy consumption. The "deep sleep" functionality of the ESP32 is used in between its active states to extend battery life.

While the development board was used for testing and measurements of the prototype device, its low dropout voltage regulator (LDO) has a relatively high quiescent current. If the development board were to be used in the field, an ultra-low quiescent current LDO with similar current sourcing abilities could be swapped to drastically improve battery life.

B. Gateway Node

The gateway node is also based on the same chip, the ESP32. To aggregate images from sensor nodes, the gateway uses BLE to receive data, then uses Wi-Fi to connect to the Firebase cloud and upload images. Before uploading data to the cloud, the gateway converts the image data into base-64 to ensure data integrity in the database.

C. Cloud Storage

Once the images make their way to the gateway, a connection is made to the Firebase real-time database via the internet. A base-64 encoded image file, shown in Fig. 2, is uploaded and stored in Firebase.

Figure 2. Firebase base-64 image string.

There is an open-source Firebase-ESP32 library available to download on GitHub to make the firmware development of this functionality simpler. The base-64 encoding ensures binary image files are reliably stored in databases.

IV. Power Measurements and Results

All testing described in this section was obtained by measuring the voltage drop across a 1Ω series resistor to the development board. Current measurements were calculated in Excel based on oscilloscope voltage readings. As a note, the development board system idle current (empty test script) sits at 52mA. This is the current requirement to power the ESP32 chip, the OV2640 camera, the microSD card reader, and regulators on the board. All following measurements include this baseline current reading.

A. Image Capture

The image capturing functionality of the OV2640 module was tested to obtain system current draw measurements. This data can be used to extrapolate energy consumption for any specific application of the sensor device. Fig. 3 shows the processor boot-up, setup, and image capture system current draw. The red box in the figure highlights five images being captured consecutively.

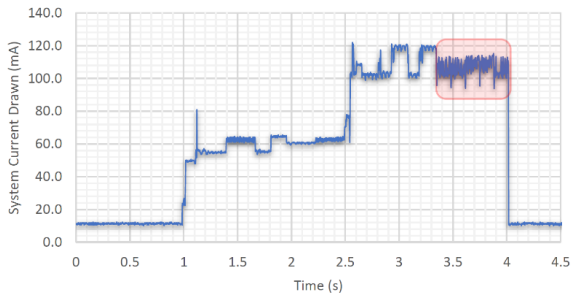

Figure 3. Boot-up and image capture system current.

The camera is programmed to automatically adjust several image exposure effects, so a series of five images are captured in order to adjust to its environment and produce the best image. This can be seen in more detail in Fig. 4.

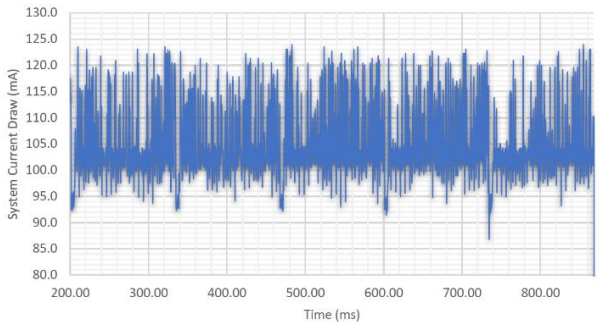

Figure 4. Series image capture system current.

The series image capture for five images takes 660ms to compete while consuming an average current of 105mA. This translates to a time of 132ms per image capture.

B. BLE Advertising

The BLE protocol uses advertising to broadcast device information to nearby recipients. In this system, the advertising and connection process lasted for a maximum of five seconds but typically took less than one second. Fig.

5 shows the current consumption of the sensor node while advertising.

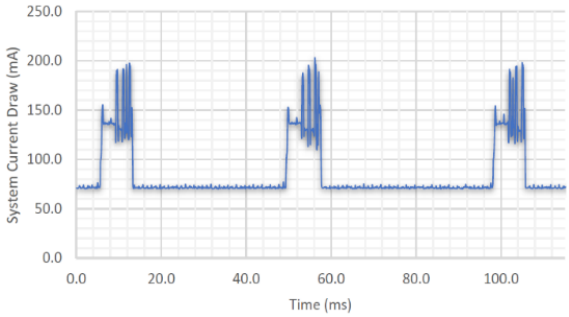

Figure 5. BLE advertising system current.

The advertising current consumption over this 115ms timespan is 83mA. The BLE protocol saves energy by periodically advertising itself in bursts while turning off RF hardware in between. This advertising scheme in Fig. 5 is the default on the ESP32 and has a period of 45ms with an advertising window of 5ms.

C. BLE Transmission

The RF transmission of the sensor node continuously reads image data from memory (flash or microSD) to fill the BLE characteristic value buffer. Fig. 5 and Fig. 6 show data being continually read into the optimal 23-byte buffer then transmitted as BLE notifications. Fig. 6 uses flash storage and Fig. 7 uses microSD storage.

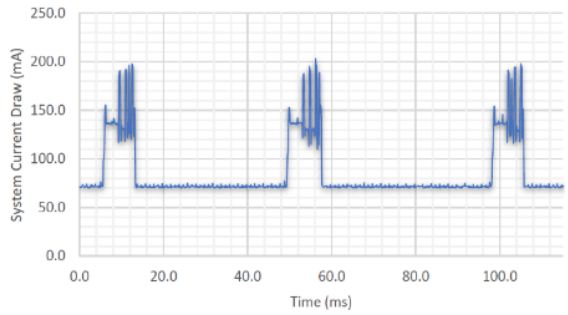

Figure 6. BLE transmission system current – Flash.

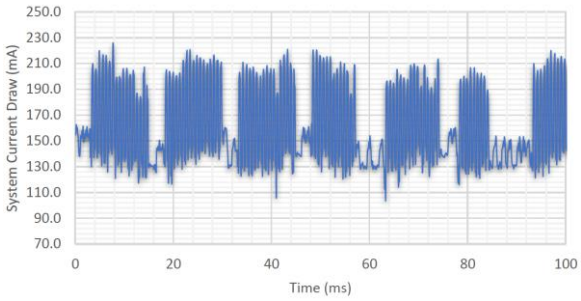

Figure 7. BLE transmission system current – MicroSD.

On average, using flash storage results in a lower current consumption. Flash uses 150mA while transmitting while microSD uses 162mA. In a test of transmitting a 110kB image file over BLE, both types of storage were able to complete the transfer in about 4 seconds. This results in a 220kbps image data throughput figure.

D. Deep Sleep

The low-power deep sleep standby mode was tested with the AlphaLab LNA10 oscilloscope preamplifier. This enables sub-microvolt signals to be displayed on scopes that typically only go down to 1mV/division on the vertical axis. The LNA10 was set for 1000 times differential signal gain.

The ESP32 is put into deep sleep mode when in standby. In this mode, the CPU, most of the RAM, and all of the digital peripherals are powered off [7]. The only parts of the chip that remain powered are the RTC, the RTC peripherals, the ultra-low power processor, and the RTC memory. Fig. 8 shows the current consumption of the development board while in this mode, the units are in microamps. The waveform starts with no microSD card inserted, then a card is inserted.

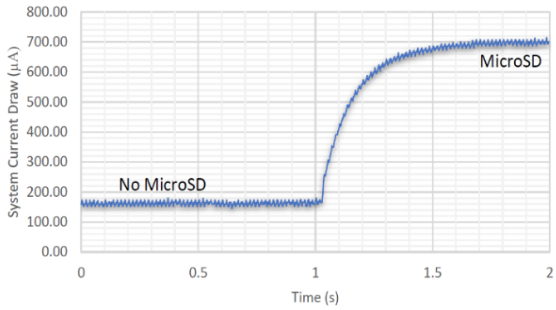

Figure 8. Deep sleep system current.

The microSD card has an incredibly high idle current of 538µA. When using flash, the development board draws a total of 162µA in deep sleep. This figure is very respectable for a development board and could easily be improved by replacing its high quiescent current LDO with an ultra-low quiescent LDO. The on-board LDO this board comes with consumes a large percentage of the 162µA current based on its datasheet [9].

V. Discussion

A. Sensor Node Power Breakdown

Two node states are defined: active and standby. The active state is when the CPU is running; the standby state is when the CPU is put to sleep. A summary of major actions in each state is depicted in Table I. The BLE transmission time will vary based on image file size, but the largest files captured (~200kB) were successfully transmitted in about 10 seconds.

TABLE I. Sensor Node Active & Standby Summary

State	Duration	Average Current
Boot Initialization	2.5s	60mA
Image Capture	1.0s	90.13mA
BLE Advertising	5.0s	83.0mA
BLE Transmission	10.0s	150mA
Deep Sleep	N/A	162µA

The conservative estimate of the total duration of the active state is 18.5 seconds. This active state will be turned on each time a new image acquisition take place.

In between active states, the sensor node goes to sleep. The duration of time the node spends in its standby state will vary but will always be well above 99% of the total standby plus active time. This is because agricultural monitoring and imaging applications require infrequent data. Over a one-hour period, the percentages of time each action would be on is shown in Fig. 9.

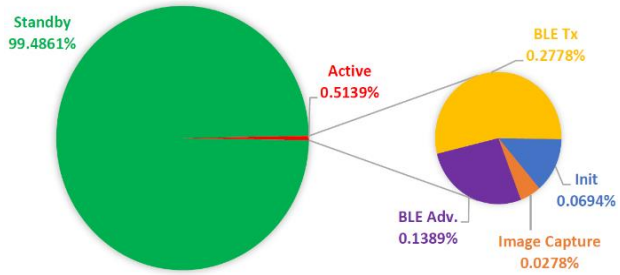

Figure 9. State percentages over one hour.

The active state is shown as a total of 0.5% of the one-hour period while standby is at 99.5%. If the image capture frequency stays at anything less than 24 captures and uploads per day, the sensor node will stay in standby mode for over 99% of the time. Using this data, the next section will estimate battery life for various image acquisition frequencies and battery capacities.

B. *Battery Life Estimates*

The battery options chosen are all Lithium Ion Polymer battery packs with sizeable capacities and manageable dimensions for this application. All are 3.7-volt batteries with 2-pin JST-PH connectors and built-in protection circuitry. The estimates will be very conservative and will use a current of 150mA for the entirety of its active state and a current of 162µA during its standby state.

The conservative estimate shown in Table II is meant to be an absolute worst-case scenario. The calculations are also done using a 20% battery discharge. Three different acquisition scenarios were compared as realistic tests.

TABLE II. SENSOR NODE LIFETIME ESTIMATES WITH VARIOUS LIPO BATTERIES

LiPo Capacity	Scenario 1 1 Image/Day	Scenario 2 5 Images/Day	Scenario 3 1 Image Every Other Day
2500mAh	429 days	258 days	468 days
4400mAh	755 days	454 days	824 days
6600mAh	1133 days	682 days	1236 days

This shows the minimum amount of time the sensor node will last on a specific battery capacity using three example scenarios. In 8 out of 9 scenarios, the sensor node will last well over a year on a single battery charge.

VI. CONCLUSION

Important specifications measured with the prototype system on a development board are now summarized. The developed IoT network achieves preliminary goals initially set for this work: gateway nodes cost less than $30 [10] and sensor nodes will last at least a year on battery power. The sensor nodes will cost closer to $45 with the inclusion of LiPo batteries chosen. With the use of a fully integrated low-cost hardware development board, a free software development environment, and a free cloud storage, each node's cost is minimized. During its active state, the sensor node consumes a maximum average current of 150mA while transmitting over BLE. The node spends over 99% of its time in standby mode consuming 162µA. In a scenario of capturing and transmitting five images per day, a sensor node could be powered from a single 6600mAh LiPo battery for an impressive 682 days. Over this lifetime the sensor node could capture and upload 3,410 images to Firebase. The system also achieves impressive WSN data throughput. Node-to-node BLE communication reaches 220kbps and node-to-cloud Wi-Fi communication reaches 709.5kbps. The system underwent multiple end-to-end system tests to verify complete process operation. The system functions performed perfectly in tests up to 20 meters between sensor and gateway; no tests exceeding this distance were done.

The IoT system characterized meets the goals and requirements for the agriculture-specific network but is not limited to this industry. This network can be utilized by any application that requires a wireless sensor network (WSN), high data rates, low power consumption, short range communication, and large amounts of data to be transmitted at low frequency intervals. As IoT trends towards the inclusion of edge computing and computer vision or machine learning and away from being their own standalone systems, these characteristics will become increasingly desirable. Crop monitoring and yield prediction systems are a prime example of this class of system, drifting further away from the typical low data rate, low bandwidth IoT systems of the past.

VII. FUTURE RECOMMENDATIONS

The developed system contains many attractive elements and serves as a well-performing standalone IoT network to monitor a single crop area. This work was developed with scalability in mind and serves as an excellent starting point to create a many-node IoT mesh network to have full coverage of a large area of interest. The following ideas are presented in chronological order of future development.

One idea to provide a wider reach for the two-node, point-to-point system developed in this work is to make the single sensor node mobile. If a single sensor node were mounted on a rover, for example, the rover could be programmed to drive to designated spots in a crop field where the sensor node would carry out its image collection process. Instead of immediately transmitting data off-board, the device would store it to memory. Once the rover had driven to all designated locations, it would make its way back to within range of the gateway node, enabling the sensor node to dump its collected data. The gateway would proceed with its functionality and upload the image files to Firebase. This is a less intensive method than a small-area mesh network and could produce similar results. The gateway node's firmware would be unaltered, and the sensor node's firmware would only need slight modification. This same idea could be applied to a sort of long railway with a moving part holding the sensor node.

There are adapter boards available that can hold up to four equivalent cameras at once, extending the utility of this idea.

Next, it would be beneficial to design and manufacture a custom sensor node PCB with an emphasis on the inclusion of power efficient supporting electronics. A board that integrated a barebones SoC with a low quiescent current LDO and clever software (GPIO) controlled peripheral power chains for a camera and memory card could result in an ultra-low power node with the same WSN capabilities described in this thesis. If the design of an entire PCB is too complicated or time-intensive, modifying the existing development board would be simple and very effective, too. Removing any LED's and replacing the old LDO with one that has the same pin-out, package size, and current sourcing ability but much lower quiescent current could drastically improve the system sleeping current.

To ensure this system could cover a large area, the number of sensor nodes must increase. This could enable the network to reach further distances because all sensor nodes could act both as collectors and repeaters. The data could hop from one node to the next, eventually reaching the gateway. This idea is known as a mesh network. In this topology, each node has at least two ways to send and receive information. This ensures the whole system does not rely on one node only. The network scheme could use Shortest Path Bridging (SPB) that allows information to be transferred by the shortest available path of nodes. SPB is specified in the IEEE 802.1aq standard. The BLE communication range depends on many things including transmit power, physical environment, and receiver sensitivity. Typically, the realizable range for BLE is between 10 and 100 meters for outdoor applications. A mesh network with sensor nodes that are within this range of each other could span the area of an entire crop field. The ESP32 SoC contains a BLE Mesh core optimized for creating large-scale device networks to get started with this application. A correctly implemented mesh network using the developed sensor and gateway nodes would efficiently enable constant monitoring and imaging of entire farms.

Conflict of Interest

The authors declare no conflict of interest.

Author Contributions

Jacob Wahl conducted the research, collected data, and wrote the paper. Jane Zhang supervised the work and approved the final version.

References

[1] O. Elijah, T. A. Rahman, I. Orikumhi, C. Y. Leow, and M. H. D. N. Hindia, "An overview of Internet of Things (IoT) and data analytics in agriculture: Benefits and challenges," *IEEE Internet Things J.*, vol. 5, no. 5, pp. 3758-3773, Oct. 2018.

[2] R. A. Kjellby, L. R. Cenkeramaddi, A. Froytlog, B. B. Lozano, J. Soumya, and M. Bhange, "Longrange & self-powered IoT devices for agriculture & aquaponics based on multi-hop topology," in *Proc. IEEE 5th World Forum on Internet of Things*, Limerick, Ireland, Apr. 2019, pp. 545-549.

[3] A. M. Apitz, "Towards a strawberry harvest prediction system using computer vision and pattern recognition," master's thesis, California Polytechnic State University, San Luis Obispo, California, 2018.

[4] Y. Fitter, "Strawberry detection under various harvest stages," master's thesis, California Polytechnic State University, San Luis Obispo, California, 2019.

[5] R. V. D. Lee. Vinduino, a wine grower's water saving project. [Online]. Available: https://hackaday.io/project/6444-vinduino-a-wine-growers-water-saving-project

[6] Agricultural IoT will see a very rapid growth over the next 10 years. [Online]. Available: https://machinaresearch.com/news/agricultural-iot-will-see-a-very-rapid-growth-over-the-next-10- years/

[7] ESP32 Series datasheet. [Online]. Available: https://www.espressif.com/sites/default/files/documentation/esp32_datasheet_en.pdf

[8] Advanced information preliminary datasheet. [Online]. Available: https://www.uctronics.com/download/cam_module/OV2640DS.pdf

[9] Voltage Regulator - CMOS, Low Noise, Low-Dropout. [Online]. Available: https://www.onsemi.com/pub/Collateral/NCP500-D.PDF

[10] L. Jackson. (Aug. 9, 2019). Introducing Arducam IoTai - The ultimate IoT (Internet of Things) board with camera support, based on ESP32 and in the shape of Arduino UNO. Arducam. [Online]. Available: https://www.arducam.com/arducam-iotai-esp32-camera-module-arduino-uno-r3-board/

Jacob D. Wahl was born in San Diego, California, USA. He received his Bachelor of Science degree in electrical engineering in 2019 from the California Polytechnic State University in San Luis Obispo, California, USA. He went on to earn his Master of Science degree in electrical engineering in 2020, also from the California Polytechnic State University in San Luis Obispo, California, USA. His academic research interests include embedded systems, internet of things (IoT), digital signal processing, and space hardware. He is now currently working as a Hardware Design Engineer at Viasat, Inc. in Carlsbad, California, USA with the PTE team.

Dr. Jane Zhang was born in Shanghai, China. She received her Diploma in Electrical Engineering from the University of Erlangen-Nuremberg, Germany, in 1997, and her Ph.D. degree in Electrical and Computer Engineering from the Georgia Institute of Technology, Atlanta, in 2002. In 1997 she was involved in the development of motion compensation related video compression techniques at the Telecommunications Institute at the University of Erlangen-Nuremberg. She also worked at Motorola Human-Interface Lab in Palo Alto in 2000 conducting research in biometric speaker verification. In 2003 she joined California Polytechnic State University in San Luis Obispo where she is currently a Professor and Graduate Coordinator. In summer of 2007, she participated in international research collaboration with the Institute of Pattern Recognition and Intelligent System at Shanghai Jiao Tong University under the Wang Family Fellowship. In 2010, she is a visiting scholar at the Institute for Media Innovation from Nanyang Technological University in Singapore. In 2019, she is a visiting professor at the Munich University of Applied Sciences in Germany under the MUAS Fellowship Program. Her research interests are in the areas of digital signal and image processing, computer vision, statistical modeling, machine learning, and multimodal interaction.