Integrating Sensors, Machine Learning, and LoRa Communication for Forest Fire Detection.

Jonathan Kessluk, Arisa Kitagishi, Noora Dawood, Nicholas Hainline

Dept. of Electrical Engineering and Computer Science, University of Central Florida, Orlando, Florida, 32816-2450

***Abstract* — Fires can be destructive and dangerous to property and life. Early detection and response are key to mitigating the possible damage and danger that fires present. This project is a new prototype system to detect fires using sensors targeted for flame, smoke, and gas to detect the presence of a forest fire quickly and efficiently. Machine learning based image processing is done on an image taken at the area of interest to validate and qualify the sensor data. The fire condition is then transmitted via LoRa to a base station that can process the data. The system will draw low current and is designed to derive power from solar panels to function autonomously.**

Keywords* — *Forest Fire Detection, Wireless Communication, Machine Learning, Chirp Modulation, Solar Power Generation, Sensors.

I. Introduction

According to the Federation of American Scientists, in the last 10 years an average of 64,100 wildfires occurred per year and an average of 6.8million acres burned annually [1]. As of July 1, 2020, nearly 24,350 wildfires have burned over 1.4 million acres this year [1] . In the past, forest fires were considered a natural cycle and were ignored [2, 3]. However, with rising awareness emphasizing the preservation of natural resources, as well as data showing increased intensity of fires have resulted in forest fires becoming a global environmental concern. Thus, forest fire detection and monitoring systems have sparked the interests of scientists and researchers worldwide.

The purpose of this project is to design a prototype for a solar powered forest fire detection and monitoring system that will serve as a preventive measure for forest fires. This device would be used in areas where human activity is present especially parts of the forest that are highly susceptible to forest fires. This device can also be used to monitor and detect forest fires to help researchers and firefighters determine potential fires or the severity of the existing fires. In this research paper, a prototype system is promoted as a solution to combat this issue using sensors, computer vision and machine learning, and LoRa communication with the entire device powered by a PV (photovoltaic) system.

II. Background Research

Sensor technology, machine learning, battery charging, solar panels, and RF design are some of the topics the team challenged for this project. In order to create suitable subsystems, it was imperative to understand what other researches have already established in order to make design decisions.

A. Sensors

Current forest fire detection and monitoring systems use video cameras to recognize smoke spectrum, thermal cameras to detect heat glow, IR spectrometers, and LIDAR (detection of light and range) to detect smoke particles using reflected laser [4]. These systems are costly due to the nature of the technology. Our objective was to design a system that can accomplish its goal while driving cost down significantly through careful electronic design and component selection.

Sensors were chosen to target three main characteristics of fire: flame flicker, increase in gas concentrations, and presence of smoke.

1) Gas Sensor

In the event of a fire, the air quality changes; the severity depends on the severity of the fire and the environmental conditions. Forest fires tend to release high levels of N2, O2, CO, CO2, H2 gasses [5]. Changes in oxygen levels can provide indication of the type of fire. A low change in concentration suggests a smoldering fire while large changes suggest liquid fuel fires that rapidly burns [5]. Other methods of gas detection use a combination of sensors to detect temperature and humidity and an algorithm to detect gases such as CO and CO2 [5]. These gas sensors use metal oxide or n-LTPS MOS Schottky diode on a glass substrate [5].

The gas sensor chosen for this project is the BME680 manufactured by BOSCH. This 4 in 1 sensor is able to measure total volatile organic compound measurements by measuring the resistance in gas sensitive-layer. The sensor can also measure humidity with ±3% accuracy, barometric pressure with ±1 hPa absolute accuracy, and temperature with ±1.0°C [6]. When the hot plate on the sensor is heated, the resistance value is measured and is used to determine concentration of total volatile organic compounds Ethanol, Alcohol and Carbon Monoxide [6]. According to German Federal Environmental Agency higher concentrations of VOC indicate lower air quality [7]. The sensor also features an IIR filter which is used for temperature compensation, that is then used to calculate other measurements and provide better accuracy.

2) *Smoke Senosor*

Smoke detection methods that use the photoelectric principle are primarily used for smoldering conditions and are effective in doing so since response times are quick [5]. In this method, the ionization smoke sensor measures smoke relative to the ionization levels in the air [5]. A potential difference is applied through a chamber and the output current is measured as a result [5]. Moreover, photoelectric method dictates that the concentration of smoke in the air will proportionally increase the light scattering capacity [5]. Thus, this method measures the variation in light scattered using optical science and technology to detect the smoke levels in each area. It is also common to combine this method with gas sensing technology for better results.

Smoke detection uses two techniques to detect its presence: non-visual and visual [5]. In a non-visual method, the detection technique looks smoke combustion conditions such as pyrolysis, smoldering, and flaming; these conditions are contingent on the type of fire and the environmental surrounding [5]. Visual techniques mostly use cameras which can detect both flame and smoke [5]. The nature of smoke is that it exists at the beginning of the fire which is crucial to understand when designing fire-detection strategies.

Computer vision will cover the visual aspect of smoke detection and the smoke sensor will use a non-visual approach. The MAX30105 module from Maxim Integrated was chosen to for smoke detection since it combines the uses of three LEDs (Red, Infra-Red (IR), Green), a photo diode and a high performance analog front end (AFE) to differentiate between sub-micron (below 1 micron) particles such as smoke, and super micron (above 1 micron) particles such as dust and steam. The device also works in high ambient light, complete darkness, or artificial light [8]. Moreover, the method of smoke detection is through External Sampling Photoelectric (ESP) smoke detection technology. Rather than using an internal chamber that many traditional smoke detector devices use, an optical smoke detector IC is used to detect smoke outside the device [8] .

3) *Flame sensor*

There are two methods of flame detection: non-visual and visual flame techniques [5]. Non-visual flame sensors use ultra-violet, visible, and infrared rays [5]. Flames emit a radiation whose intensity is determined by the flame temperature and the type of fuel burning [5]. An ultra-violet sensor is used to measure the brightness since UV sensors, infrared and visible light sensors are used to measure flame. However, IR and visible light sensors are more effective than ultra-violet sensors [5]. UV sensors tend give out more false positive alerts due to the UV sensors emitting sparks of UV spectra that essentially interferes with the signal [5].

The sensor will take a non-visual approach by using a pyroelectric infrared detector by PYREOS (ePY12241). There are five key characteristics to consider when using pyroelectric infrared detector: output sensitivity that depends on narrow infrared band, signal to noise ratio, noise equivalent power (NEP), specific detectivity (D*), and response time [9]. The output sensitivity, D*, SNR, and NEP are dictated by the manufacturer in the module design. The user, however, has the ability to choose the electrical response time by adjusting the impendence and capacitance as well as setting the high pass and low pas filter values and the sample rate of the filter. [10] This sensor will be useful in identifying flames in low light, such as at the nighttime. To use in the daytime, it must be paired with PYREOS' sunlight-rejection sensor.

The nature of fires comes with various characteristics such as shape, size, color, location, growth, degree of burning, and dynamic texture. Typical sensors are not capable of measuring each of these characteristics and their parameters accurately [5]. Thus, flame sensors that depend on these techniques give false alarms whose validity can only be evaluated by an experienced individual. A device to solve this issue is using a camera that can capture images of fire and analyze them accordingly to establish fire detection [5]. The visual approach of detecting flame is covered in the Computer Vision section.

B. *Computer Vision*

Machine learning has advanced significantly in the past decade. From a single image, objects can be detected and classified. One of the well-known neural network is YOLO [11] which has an object detection and classification for real-time application.

There were also researches done specifically to detect smoke and fire using computer vision. A method by the Ministry of Public Security of Shenyang Fire Research Institute [12] shows how it can detect fires and smoke at a distance by utilizing optical flow and foreground image accumulation. Another research by Nicholas True [13] utilizes motion detection using frame differencing and color classification to isolate color of fire.

Our project makes use of this technology in order to detect fire using a regular camera. By taking an image using a regular camera, more information is acquired than by using a thermal camera while alleviating the cost significantly. The details can be analyzed using a neural network from the taken image to determine whether the fire exists in the image.

The design of our neural network is inspired by the Fire Detection Net [14] which has an architecture illustrated below.

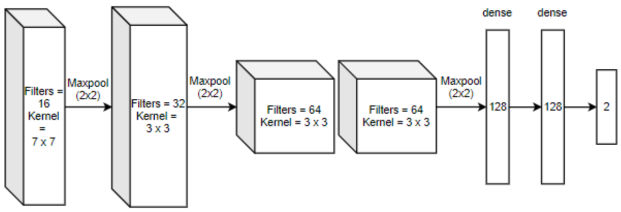

Figure 1: Architecture of neural network

This is designed for devices with limited CPU, memory, and power draw which is desirable for our setup. It uses TensorFlow 2.0 and Keras's Sequential API to build the convolutional neural network.

There will be color classification and optical flow to improve the accuracy of fire detection. Color classification can be performed by detecting pixels that contain a color of fire and masking the others. OpenCV also has optical flow methods such as Lucas-Kanade method and Dense optical flow which are ideal for detecting motions using two frames. [15] Lucas-Kanade method checks the motion of feature points and creates lines representing the motion of those feature points. While the Dense optical flow is based on Gunner Farneback's algorithm which computes motion for all the points in the frame creating a blob of motion rather than lines. For our use, we determined that the Dense optical flow would be better to identify the flickering of the fire rather than the Lucas-Kanade method.

C. Battery Charging

The batteries chosen for this prototype are two 18650 lithium ion (Li-Ion) cells in series to power the system. The Li-Ion integrated charging chip chosen was a LT3652 produced by Linear Technology. It is capable of up to 2A max charging output and comes with a MMPT (Maximum Power Point Tracking) integrated into the chip itself. Its wide input range of 4.95V to 32V with an Absolute max of 40V was more than adequate for charging the two cells. The two cells are each 2700mah to 3200mah depending on which set is used and provide plenty of room for running the system without powering the solar array for minimum two hours.

D. PV Solar Array

The solar array consists of two 1.5-watt panels in parallel that have 16.6 volt and 90 mA max under load with a MMPT (Maximum Power Point Tracking) making sure the input is always optimal. The 3-watt array handles powering the circuit during peak solar radiation conditions and charging the array in the daytime. The lower current output of the panels means that this chip is running off panels that are 100W/m2 which is enough to slowly charge the panels throughout the day.

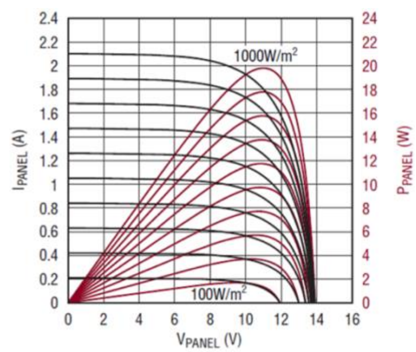

Figure 2:Current vs Voltage vs Power

To find out what panels were needed the following equations were done.

$$V_{OC} = \left(V_{BAT(FLOAT)} + V_{FORWARD(D1)} + 3.3V\right) * 1.15 \quad (1)$$

$$I_{P(MAX)} = I_{CHARGE} * \frac{V_{BAT(FLOAT)}}{\eta * V_{P(MAX)}}$$

$$V_{P(MAX)} = \left(V_{BAT(FLOAT)} + V_{FORWARD(D1)} + 0.75V\right) * 1.15 \quad (2)$$

Solving for these equations gets the following.

$$V_{OC} = 13.8V$$
$$I_{P(MAX)} = 1.8A$$
$$V_{P(MAX)} = 10.9V$$

The two panels chosen are the following.

$$V_{OC} = 20.5V$$
$$I_{P(MAX)} = 0.18A$$
$$V_{P(MAX)} = 16.6V$$

Comparing the graph for the control range and minimum input, this system meets the requirements for the Li-Ion IC to charge batteries.

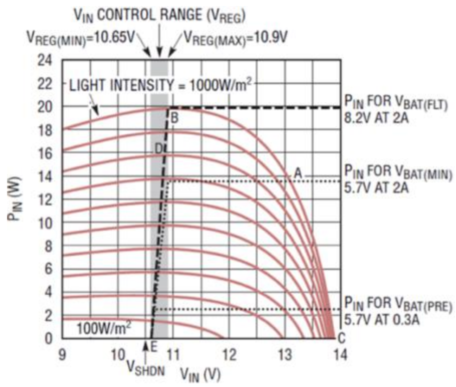

Figure 3: Control range

For our purpose, the lower amperage means slower charge times but with the 15% safety factor on the panels they will provide enough power to keep the system running all day and keep a set of charged batteries maintained.

E. RF Design

1) Frequency Selection: The main discussion of Radio Frequency technologies comes down to range. The following numbers were compared to get the maximum range out of our device. Actual distances depend on a variety of variables and the ones in the table below are just averages.

Technology	Frequency	Range
Bluetooth	2.45 GHz	30 Feet
Wifi	2.45 GHz (or 5GHz)	100 Feet
Zigbee	2.4 GHz	1000 Feet
FSK Modulation (900MHz)	900 MHz	2+ Miles
LoRa	400 MHz / 900 MHz	10+ Miles

Table 1: Comparison Between Technology Ranges and Frequencies

Furthermore, the Free-Space Path Loss equation is used to determine the attenuation of radio energy between two antennas.

$$FSPL = \left(\frac{4\pi d f}{c}\right)^2 \qquad (3)$$

where d is the distance between antennas, f is the frequency, and c is the speed of light. Trying with different ranges, a desired attenuation can be achieved with Table 1. Finally, due to regulations of how much power can be dissipated in the 400MHz bands, 900MHz is a good solution for global use and higher power dissipation.

2) LoRa: LoRa, literally "Long Range", is a proprietary spread spectrum modulation scheme that is derivative of Chirp Spread Spectrum modulation (CSS) which trades data rate for sensitivity within a fixed channel bandwidth [16]. The idea is creating a physical layer protocol that is separate from higher layer implementations which allow the protocol to be generically used with new and existing devices. LoRa is a bandwidth scalable, low power, and long-range modulation technique. It allows a very large link budget that exceeds conventional FSK [16]. This technology is still relatively new, however, and resources describing its usage and modulation is hard to come by. LoRa is ideal compared to other common protocols, e.g. Zigbee, for this project as it boasts (in ideal conditions) a range of up to 30 miles [17].

3) Chirp Spread Spectrum (CSS): To send signals wirelessly it is essential to encode the signal on a physical medium. Typical techniques of doing this are modulating characteristics of a carrier wave like the amplitude, frequency, phase, or any combination thereof. LoRa, on the other hand, uses a proprietary version of Chirp Spread Spectrum Modulation (CSS). Chirp Spread Spectrum was developed for radar applications in the 1940's [16]. It has become more popular recently as it is low power and has great sensitivity. Unlike other modulation techniques, it has the inherent ability to resist multipath fading, Doppler effects, and interference in the same bands. The idea is that a "chirp" has a constant amplitude but the frequency passes through the entire bandwidth in a certain time frame. If the frequency increases it's called an "up-chirp" and if the frequency changes from highest to lowest it is considered a "down-chirp" [18]. The alteration between up-chirps and down-chirps create the symbols for LoRa.

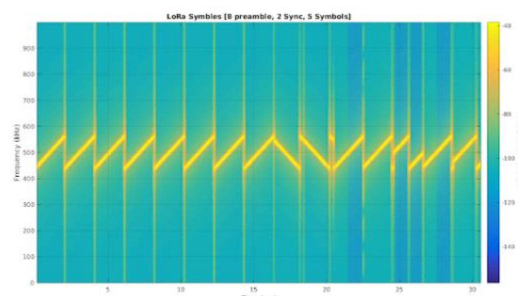
Figure 4: Spectrogram of LoRa physical layer [17]

III. Project F.I.R.E Apllication

Project F.I.R.E. is to be mounted in a pre-determined location. The suggested place to mount the device is moderately high on a tree trunk where the sensors and camera can see a relatively wide area. Once the device is placed in its location, the batteries should be put into the device and the device will attempt to join the mesh network automatically. It will continue to try and join a network until the root node has responded. Then it will attempt to detect fires. The device is now installed! There is no more user interaction with the device needed.

Prior to installing the network, it is imperative that a base station is set up. This "Root Node" of the network is where all the data is sent to. This device is just responsible for passing the data on to a database, saving as a file, or other application, and is up to the user to determine its usage and function. There are a few requirements on how the root node must respond so that a node can join the network correctly (e.g. sending the current time when joining the network to update the clocks).

IV. Design

The following sections cover some of the design choices made for Project F.I.R.E.

A. Hardware

The project contains two controllers, a Raspberry Pi for sensor data and a SAMR35 for the network data. The SAMR35 is very low power and will allow the device to last longer on its batteries while the Raspberry Pi has more power to run machine learning and computer vision algorithms. The sensors will be discussed in a section below. The device uses two batteries as mentioned in a previous section with a Li-Ion charging IC. This made the design easier to implement.

B. Software

The software of the system is simple to understand but became a little complex to implement. There are two portions: Network and Sensor. The Sensor software reads all the sensors, takes two pictures with the camera, and processes that data with some computer vision and machine learning algorithms to determine if there is a fire. The network software handles incoming and outgoing network packets. The network software is implemented as a state machine that is always running whereas the sensor software is sequential and only run once periodically. In the original design, the Raspberry Pi shares its data with the SAMR35 through UART. In the final design due to some constraints, the Sensor Software saves its data to a file and the Network software can read that data.

C. Sensors

Each sensor outputs raw data that is then analyzed to create a data processing algorithm in python to determine probabilities of gas, smoke, and flame.

1) Gas:

The data collection for the gas sensor is very straightforward. Since this sensor was calibrated for a long time, the measurements are stable and relatively accurate. The raw measurements for temperature, humidity, and gas are obtained and then compensated to improve accuracy. The final values are calculated in Celsius, percent, and ohms to interpret the data. Temperature and humidity readings from the can be used to monitor the environment and a warning can be sent when high temperatures exist in a dry environment. In fire conditions, the gas measurement rises significantly higher compared to non-fire conditions. When a gas measurement reaches a maximum value, a warning is sent to indicate high levels of volatile organic compounds present in the atmosphere.

2) Smoke:

The PIM438 module which includes the MAX30105 module receives a raw data from the IR and Visible light detector. The algorithm is designed to take the mean of incoming data and look for changes between the means by taking the difference between the recent mean and the mean taken X readings ago (delta). A change is detected when the delta value is greater than the threshold value. The mean size, delta size, and threshold can be tuned to increase data smoothing and sensitivity. A function was created to count the maximum number of consecutive True changes. If this value is greater than a set value, then flame has been detected.

3) Flame

The sensing element in the ePY12241 provides an output current that is proportional to the rate of change of temperature of the material. The chip uses an analog front end to receive an analog signal, which is then filtered by a high pass filter. The signal then goes into a sigma delta ADC convertor. Then, the low pass filter removes large frequencies, and the data is then read by the MCU.

A data window is specified, and the RMS of the raw signal is taken to determine the signal strength of the combination of the frequencies in the bandpass of the filters used. The reading is then divided by the signal multiplier, which is the sample rate. From here the flame algorithm is similar to the smoke algorithm where the delta is calculated from the current value and the value X readings ago. A change is detected when the delta value is greater than the threshold value. The RMS data window, mean size, delta size, and threshold can be tuned to increase data smoothing and sensitivity. A function was created to count the maximum number of consecutive True changes. If this value is greater than a set value, then flame has been detected.

$$RMS = \frac{\sqrt{(d_1)^2 + (d_2)^2 + \cdots (d_n)^2}}{n} \quad (4)$$

d_n is the raw data collected from the pyroelectric infrared detector.

n is the number of data collected in a data window (data window size).

D. Computer Vision

Our computer vision comprises of two methods: color classification plus optical flow and machine learning. The color classification is done by finding the pixels with certain range of values similar to a fire. The image is first filtered by Bilateral filter to blur some of the textures while maintaining the edges. Sample image from our Raspberry Pi Zero W is shown below.

Figure 5: Sample images for machine learning

To detect motion, OpenCV Dense optical flow [15] is used. Sample output is shown below.

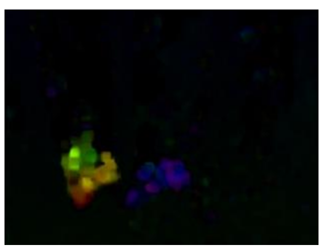

Figure 6: Sample output from OpenCV

Then, we apply filters and thresholding to find contours around the detected fires in both images from color classification and optical flow. The centers of these contours are compared to analyze the overlap between the two methods. Contours with low area size is eliminated to avoid excessive detections of contours and to obtain lower false positives.

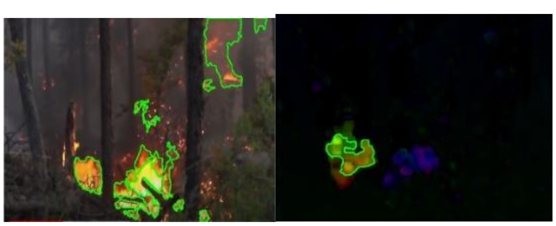

Figure 7: Filters used to find contours

By verifying if the centers of the contours are close to each other with certain area size, we can confidently determine that there is a possible fire.

For our machine learning, we trained a model using the architecture below (with TensorFlow 1.8.0 and Keras 2.1.5) which is inspired by the Fire Detection Net. [19] We tried to simplify the architecture to obtain less parameters while keeping its validation accuracy higher than 90%. But simplifying the architecture too much resulted in lower accuracy. Our final architecture was able to achieve over 1% more accuracy compared to the Fire Detection Net mentioned earlier for our dataset. ReLu was used for our activation function, and binary cross entropy was used as our loss function.

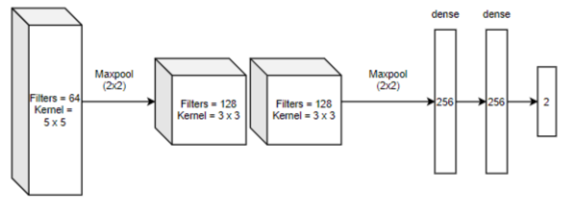

Figure 8: Final architecture

Our dataset is obtained by gathering forest fire and forest images using the Microsoft API. [20] It is composed of about 600 images we obtained online. We dedicated 75% of it to the testing and 25% for evaluation. Sample images are shown below.

Figure 9: Images from Microsoft API

V. RESULTS AND DISCUSSION

This section focuses on testing and implementation. The team's experience with each subsystem will be discussed briefly below.

A. Power Subsystem

The subsystem that runs the Raspberry Pi Zero and all the sensors is made by running two buck converters to drop the voltage of the batteries to 3.3 volts and 5 volts. These two voltage rails are responsible for running all systems as well as the PCB.

B. Sensor Subsystem

After exposing the three sensors to a fire burning in a grill, each sensor was able to successfully detect a fire. The gas sensor showed that when its exposed to a fire, the temperature increased greater than 60°C and the relative humidity was greater than 60%. However, this does not entirely indicate there is a fire, rather it can be used to determine potential forest fire condition. In other climates, the humidity can be lower. The gas measurement in ohms was significantly higher than in non-fire conditions.

The smoke sensor was able to detect more than 20 consecutive True changes which satisfied the requirement for smoke detection.

With a nearby fire, the most significant bit of the sensor is set which resulted in a flame detected. During a test using a lighter placed approximately 1m away in low light, the flame algorithm was able to detect flame with 10 consecutive true statements. This is expected since the lighter produces a smaller flame. The thresholds, channel and analog settings can be adjusted to increase or decrease the algorithm's sensitivity.

C. Network Subsytem

The network subsystem works in theory. Due to some unforeseen complications with implementing the LoRa physical layer separately from a LoRaWAN MAC layer has made the process complex. To save time and effort, the mesh network, and thus the network subsystem, was implemented on the raspberry pi for our demonstration; not the SAMR35. The system uses 16-bit ID numbers meaning that the network, theoretically, can support 65535 nodes. In reality, this is too many nodes. Light arbitration was implemented in the form of delays and timeouts on the receive and send portions of the network to avoid collision.

D. Processing Subsystem

The best model's training result is shown below. It uses binary cross entropy for the loss, batch size of 32, learning rate of 1e-5, and 200 epochs. The highest validation accuracy it was able to achieve was 93.36%.

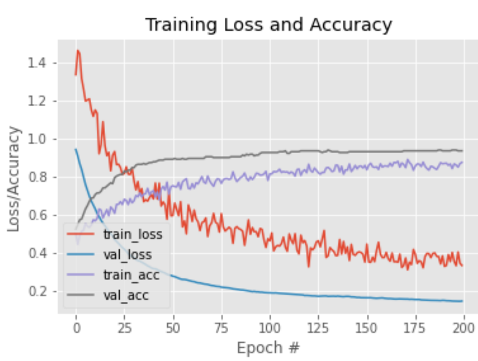

Figure 10: Training loss accuracy

Using our sample test set (16 images) which contains images not present in the dataset we used to train the model, we were able to achieve 87.5% accuracy. Some of the results are shown below.

Figure 11: Positive results from machine learning

The PiCamera works properly in our system and takes two consecutive images with around a second interval. Then, we applied color classification and optical flow as designed. The contours from color classification totaled 10 while there were no contours detected using the optical flow. This calculated to 35% confidence score that there is a fire using color classification and optical flow. The resulted images are shown below.

Figure 12: Detecting contours

The confidence score we received from our model for this image was 40%. From our two methods, we resulted in final confidence score of 37.5%. The low score is most likely due to our environment setup. We trained our models specifically with forest fire and forest environment. Thus, it does not work as well in non-forest environment. For our color classification plus optical flow method, we would need to further fine tune the values for the colors, filters, and thresholding to improve our results.

VI. Conclusion

In conclusion, the team put together a single fire detecting node that is solar powered. Throughout testing and integration, the team noticed that sometimes the fire detection algorithms resulted with some false positives or negatives. This is fixable by enhancing our training data set for the machine learning algorithms and adjusting the sensors configuration settings.

Moreover, the team learned to be cautious with using new technologies due to some issues implementing LoRa on the final product. All in all, the project finished in a working state. It can detect fire and if connected to a working LoRa transceiver, can send data through a custom mesh network software.

Acknowledgement

The authors wish to acknowledge and thank the assistance and support of Dr. Samuel Richie from the University of Central Florida.

We also extend our thanks and gratitude to Cameron Lemme and the members of the STEM@SIEMENS committee at Siemens Energy Inc. in Orlando, FL for funding this project.

We offer our thanks and gratitude to the professors of the Evaluation Committee who have agreed to review and critique our project.

References

[1] Federation of American Scientists, "Wildfire Statistics," Congressional Research Service, 2020.

[2] S. Ouni, Z. T. Ayoub and F. Kamoun, "Auto-organization approach with adaptive frame periods for IEEE 802.15.4/zigbee forest fire detection system.," 16 Jan. 2019. [Online]. Available: https://doi.org/10.1007/s11276-018-01936-x. [Accessed 30 Jan. 2020].

[3] M. Jurvélius, "FOREST FIRES AND INTERNATIONAL ACTION," 2003. [Online]. Available: http://www.fao.org/3/XII/0820-B3.htm. [Accessed 30 Jan. 2020].

[4] K. Nörthemann, J.-E. Bienge, J. Müller and W. Moritz, "Early forest fire detection using low-energy hydrogen sensors," 1 Nov. 2013. [Online]. Available: https://pdfs.semanticscholar.org/b807/d5144095c15f7805fd272cb71a8a023a9516.pdf. [Accessed 30 Jan. 2020].

[5] A. Gaur, A. Singh, A. Kumar, K. S. Kulkarni, S. Lala, K. Kapoor, V. Srivastava, A. Kumar and S. C. Mukhopadhyay, "Fire Sensing Technologies: A Review," 1 May 2019. [Online]. Available: https://ieeexplore.ieee.org/document/8625538. [Accessed 19 Mar. 2020].

[6] BOSCH, "BME680 Datasheet," 2019. [Online]. Available: https://www.bosch-sensortec.com/media/boschsensortec/downloads/datasheets/bst-bme680-ds001.pdf. [Accessed July 2020].

[7] Sensirion, "Indoor Air Qaulity and Total Volatile Organic Compounds," 2019. [Online]. Available: https://www.sensirion.com/fileadmin/user_upload/customers/sensirion/Dokumente/9_Gas_Sensors/Sensirion_Gas_Sensors_IAQ-VOC_Product_Brochure_EN.pdf. [Accessed 3 July 2020].

[8] Maxim Integrated, "MAX30105: High-Sensitivity Optical Sensor for Smoke Detection Applications," 2016. [Online]. Available: https://datasheets.maximintegrated.com/en/ds/MAX30105.pdf. [Accessed 1 July 2020].

[9] PYREOS, "AN136 Application Note: Understanding pyroelectric infrared detectors," 2020. [Online]. Available: https://pyreos.com/wp-content/uploads/2020/01/AN136-Understanding-pyroelectric-infrared-detectors.pdf.

[10] PYREOS, "ezPyroTM SMD I2C Pyroelectric Infrared Sensor," 22 June 2018. [Online]. Available: https://forum.pycom.io/assets/uploads/files/1585523830287-sensor_datasheet-min-3.pdf. [Accessed 20 May 2020].

[11] [Online]. Available: https://arxiv.org/abs/1804.02767.

[12] [Online]. Available: https://www.sciencedirect.com/science/article/pii/S1877705813013222.

[13] [Online]. Available: https://cseweb.ucsd.edu/classes/wi09/cse190-a/reports/ntrue.pdf.

[14] A. Rosebrock, "Fire and Smoke detection with Keras and Deep Learning," [Online]. Available: https://www.pyimagesearch.com/2019/11/18/fire-and-smoke-detection-with-keras-and-deep-learning/.

[15] "Optical Flow," [Online]. Available: https://opencv-python-tutroals.readthedocs.io/en/latest/py_tutorials/py_video/py_lucas_kanade/py_lucas_kanade.html.

[16] Semtech, "AN1200.22 LoRa Modulation Basics," Semtech, Camarillo, CA, 2015.

[17] Semtech, "What is LoRa," Semtech, [Online]. Available: https://www.semtech.com/lora/what-is-lora. [Accessed 13 7 2020].

[18] S. Ghoslya, "LoRa: Symbol Generation," [Online]. Available: https://www.sghoslya.com/p/lora-is-chirp-spread-spectrum.html.

[19] [Online]. Available: https://www.pyimagesearch.com/2019/11/18/fire-and-smoke-detection-with-keras-and-deep-learning/.

[20] [Online]. Available: https://www.pyimagesearch.com/2018/04/09/how-to-quickly-build-a-deep-learning-image-dataset/.

A. A. Alkhatib, "Forest Fire Monitoring," 20 Dec. 2017. [Online]. Available: https://www.intechopen.com/books/forest-fire/forest-fire-monitoring. [Accessed 30 Jan. 2020].

[22] United Nations Enviroment Programme, "Governments, smart data and wildfires: where are we at?," 3 Jan. 2020. [Online]. Available: https://www.unenvironment.org/news-and-stories/story/governments-smart-data-and-wildfires-where-are-we. [Accessed 30 Jan. 2020].

Biography

Jonathan Kessluk will be graduating with a bachelor's in Computer Engineering at UCF. He is currently pursuing opportunities in the aerospace industry with a plan to continue his education to a graduate degree later on.

Arisa Kitagishi will be graduating with a bachelor's in Computer Engineering at UCF. She is planning to pursue graduate school and find a career as a researcher. Her main interests are robotics, biotechnology, A.I. and machine learning.

Noora Dawood will be graduating with a bachelor's in electrical engineering at UCF in the Power and Renewable Energy track. She is currently working in Controls and R&D at Siemens Energy Inc for gas turbines. She plans to move to Vancouver, Canada to pursue an EIT position. Her main interests include renewable energy integration, power transmission, and control and automation systems.

Nicholas Hainline will be graduating with a bachelor's in electrical engineering in the power track from UCF. He is currently pursuing employment for the US government as a civilian contractor and/or a government employee. His interests are computer hardware, server management, and big data storage.