International Journal of Control and Automation
Vol. 13, No.2, (2020), pp. 1535 - 1542

UAV Based Forest Fires Detection Using Edge Computing and IoT

Sidarth Wadhwa[1], Isha Yadav[2], Krishnamoorthy A[3*], Vijayarajan V[4]

[1]*Undergraduate Student*

[2]*Undergraduate Student*

[3]*Assistant Professor (Sr.)*

[4]*Associate Professor*

School of Computer Science Engineering, VIT University

Vellore, India

[1]*sidarth.wadhwa@gmail.com*, [2]*ishayadav131999@gmail.com*, [3]*krishnamoorthy.arasu@vit.ac.in*, [4]*vijayarajan.v@vit.ac.in*

Abstract

Forest fires are ravaging many ecosystems across the world. Most of the fires start from a single point and grow to immense proportions, making them extremely difficult to deal with. This paper defines a rapid response system that can be deployed in forested areas in order to provide quick and appropriate response to forest fires, before they spread over large areas. The proposed system uses a UAV that is equipped with a camera and a microprocessor, like a Raspberry Pi. The microprocessor processes the camera feed to identify the presence of fire using a convolutional neural network, either by utilizing on-board resources or using supplementary resources made available to it in the form of a Vision Processing Unit (VPU) plugged into its USB port. The benchmarks associated with one such neural network running on the Intel Neural Compute Stick 2 is also supplied. In the event of a fire being detected, the microprocessor issues a command to the actuators, like speakers and sprinklers that are deployed on the ground at appropriate intervals, thus activating them in an effort to ward off animals and reduce the loss of wildlife.

Keywords: *forest fires, rapid response, Internet of Things, edge computing*

1. Introduction

Forest fires are a major cause of concern in today's world. Not only do they cause extensive damage to flora and fauna, but also have many other negative effects including release of large amounts of carbon dioxide into the atmosphere, and soil erosion which, in turn, leads to flooding and landslides. Large forest fires are often controlled by using large amounts of chemicals in firefighting. These chemicals accumulate in the soil for years, affecting its fertility and biodiversity.

There were 46,706 forest fires from January 1 to November 22 in 2019 in the US. During the same period in the year before that, there were 52,080 [3]. In 2018, the total losses due to wildfires in the US were estimated to be to the tune of 24,000 million USD, with the costliest wildfire occurring from November 8-25, 2018 in California, alone costing 8,500- 10,500 million USD [4]. Climate change is only worsening the forest fire situation, as is evident from the Australian bushfires ravaging the continent in 2020.

ISSN: 2005-4297 IJCA

While there have been many advancements in the field of preventing and responding to wildfires, the loss of life, area affected and losses caused remain extremely high. This high- lights the need for innovation in the field of fire response and control. It is necessary to create systems that can further facilitate the firefighters responding to such events. An independent system that can monitor forests for any signs of fires, and take remedial action as soon as possible, to any respectable extent of efficacy should prove helpful in our modern day scenario. This paper presents an integrated system that detects fires using apparatus mounted on a UAV that has been assigned specific monitoring routes, and then activates actuators that can provide a stopgap first response, while warding off animals inhabiting the affected area before they are harmed, and alerting the relevant authorities.

The UAV will be equipped with a microprocessor such as the Raspberry Pi 3 B+, which has a camera connected to it. The Pi will utilize this camera to get a live feed of the forest along its designated route. Simultaneously, it will run inference on the camera feed using a trained convolutional neural network. Since most microprocessors do not have the requisite processing power on board, we utilize an external VPU in the form of the Intel Neural Compute Stick 2 (NCS 2), which works with the Intel OpenVINO toolkit. The NCS 2 is an edge computing device which runs over the Intel Myriad X processor, and can be used for AI development, prototyping and deployment. It runs over USB 3.0 technology and allows for models trained on PCs to be deployed on various platforms, including the aforementioned Raspberry Pi 3, without the need for internet connectivity. Furthermore, it will prevent excessive heat generation on the board itself, which could lead to thermal throttling, thereby reducing performance further. In the event a fire is detected, the microprocessor communicates with the ground unites, i.e. the actuators. These include tank-based sprinklers deployed on the ground on the forest, speakers that can emit sounds unpleasant to the native wildlife, and smart fence posts that can lure animals like deer away from the affected area by using scents like that of acorns. All of these will be activated by the drone mounted microprocessor as and when required. The aim is to reduce the loss of wildlife to fires.

The communications will take place over a long range network that can work through dense forestation. In this case, we con- sider LPWAN (Low Power Wide Area Network), which allows for long range machine to machine (M2M) communication, to be suitable. More specifically, we aim to utilize the capabilities of the LoRaWAN standard, since it allows for very long range communication and features high receiver sensitivity. The low power consumption will ensure low maintenance requirements of the nodes. The nodes will be activated by a message sent to the gateway by the microprocessor mounted to the drone.

2. Background

In the present scenario, airborne systems are used to provide observational capabilities over the large forested areas but the actual monitoring is still carried out by people, highlighting an avenue for automation of the process. Watchtowers are present in some areas for the monitoring of high value assets and the like, but these provide limited visibility and are inflexible. At the same time, satellites often lack the requisite resolution for proper identification of forest fires. Ground based systems can prove expensive and difficult to deploy, and may, at the same time interfere with ecosystems. Since manned aircraft can only be deployed in certain weather conditions and require aerodromes [1], this brings us to the scope for UAV (Unmanned Aerial Vehicle) deployment for the purpose.

UAVs can be applied to the task of monitoring forested areas in various configurations.

ISSN: 2005-4297 IJCA

Kinaneva et al [2] suggest a model involving two distinct UAVs operating at different altitudes, with the one working at the lower altitude (10m - 350m) being called in to verify alerts raised by the one at the higher altitude (350m - 5500m). A rotary wing drone is considered appropriate for the job of monitoring at the lower altitude, while a fixed wing drone is applied to the task of high altitude surveillance (patrol drone). Both the UAVs are said to be equipped with cameras featuring optical or thermal (or both) capabilities. In the even the patrol drone detects a fire, it raises an alarm which activates the rotary wing drone to inspect the area at the GPS coordinates sent to it by the patrol drone. If the drone can verify the occurrence of a fire, it raises another alarm which alerts the authorities and departments so that they can provide appropriate response to the condition. The drones utilize a neural network algorithm to detect fires based on the camera feed. The drone suggested by them to take on the role of the patrol drone is the ALTi Transition-F Vertical Take-Off and Landing (VTOL) fixed wing UAV. For the role of the rotary wing drone, they consider the DJI Matrice 210 RTK drone.

Elsewhere, a bottleneck is encountered when trying to process a large amount of image data. Sherstjuk et al [3] solve this by shifting part of the computation to a ground command center. The sensors mounted to the UAVs in their design include an infrared camera (non-thermal) and a 12 megapixel optical camera. The UAVs follow a patrol-confirmation paradigm. They utilize a GIS based Common Terrain Model along with Image Processing techniques to handle the radiation readings captured by the infrared sensor. Furthermore, they utilize the optical camera sensor for the recognition of smoke and flame, since this can be conducted based on colour. This study indicates to the need to offset computation to a ground station since the remote apparatus deployed on the drones cannot handle the processing load by itself.

Esfahlani [5] utilizes Simultaneous Localisation and Mapping (SLAM) for the navigation of the drones, which capture video streams using monocular lenses. The collected data is sent to a ground station which then generates a map of the area.

Since the deployment area will be large and the actuators far apart, there is a need for us to use a communication protocol that ensures long range, albeit low power operation. Furthermore, the data transmitted will only contain infor- mation regarding whether the actuator should be activated or not. Queralta et al [6] present a study into the LPWAN (Low Power Wide Range Networks) technologies that uti- lize the unlicensed communication band. They conclude that LoRaWAN and Sigfox prove to be useful in deployments that require transmissions of modest volumes of data, while Symphony Link and Ingenu prove better in the case of large volume data transfer. The study shows that the various types of LPWAN technologies available in the unlicensed frequency band provide solutions for a varied collection of deployment scenarios, with each having its own benefits and drawbacks. In order to alleviate the need for a ground station, we utilize a resource local to the system, in the form of the Intel Neural Compute Stick 2. Alternatively, Google’s coral platforms such as the development board and the USB based accelerator, or Nvidia’s Jetson Nano, among others, could also be used.

The Intel Neural Compute Stick 2 [7] is a development kit for AI inferencing on the edge. As such, it does not require a connection to the internet to function. This means that we can run machine learning algorithms (convolutional neural networks in this case) locally, even on platforms such as the Raspberry Pi 3. We decided to go with this in order to boost the inferencing performance, and run inference at rates higher than what would be possible natively on the Pi.

3. Methodology

Our proposed setup consists mainly of three components: the neural network, the deployment system, and the communication protocol. The benchmarks for the neural network running on the NCS 2 are provided, for both synchronous and asynchronous modes, in the Neural Network subsection.

3.1. The Neural Network

The neural network [8] used for testing the performance on the NCS 2 is a pre trained convolutional neural network, based on a reduced InceptionV1 architecture [9]. as shown in Fig. 2, it utilizes three consecutive inception modules, which work together to yield an accuracy of 0.93. The NCS 2 is used in order to boost the inference capabilities of the Pi, and as a substitute to cloud based solutions. This is done to make the system more robust and independent of external dependencies.

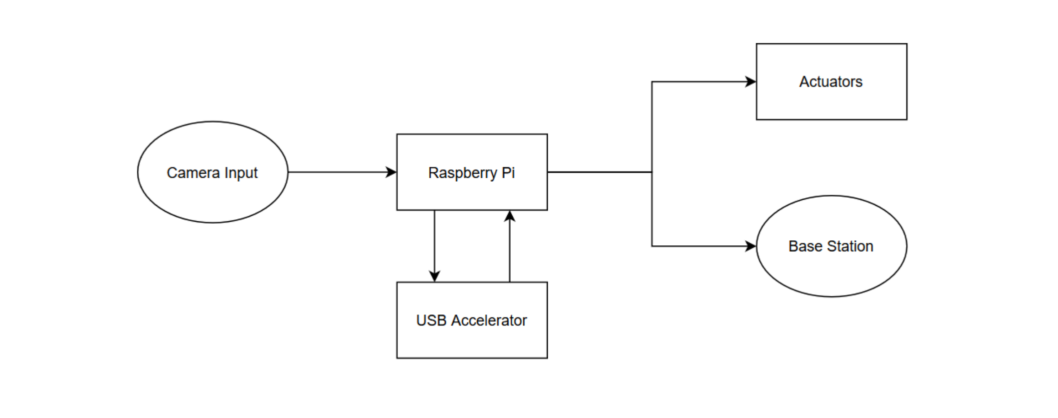

Figure 1. Block diagram of proposed system

The higher the throughput of the device, the higher the number of frames that can be analyzed per second. Using the cloud would require a persistent internet connections and certain quality of service standards to ensure the uninterrupted processing of the input. Edge computing relieves the system of any such necessities by handling the processing locally. It also ensures that deployment can be done on various types of systems, independent of the microprocessor in some cases. To run this network on the target device (Intel NCS 2), we convert it to its respective XML and BIN representations using the Model Optimizer tool which is part of OpenVINO. Once this Intermediate Representation (IR) format is obtained, it can be deployed on the NCS 2. OpenVINO’s Benchmark App is utilized to obtain the benchmarks for the network running on the NCS 2. The model runs with a latency of 33.67 ms and a throughput of 118.724 FPS in asynchronous processing mode on the NCS 2 when it is connected to the Pi, over 7128 iterations of random values as input.

ISSN: 2005-4297 IJCA

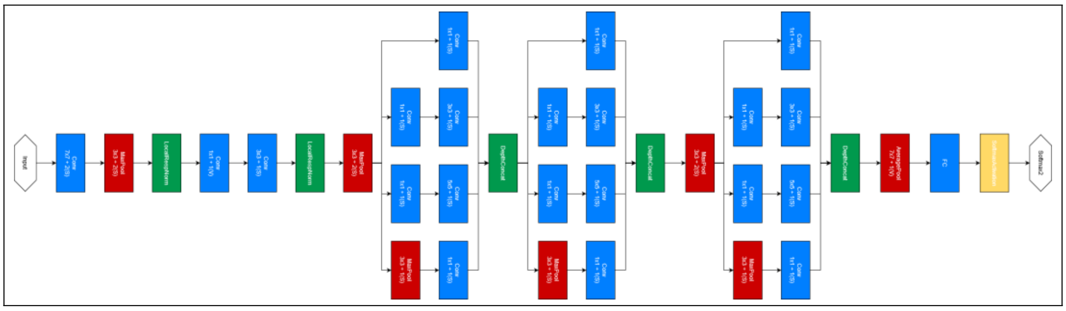

Figure 2. Neural network architecture

Due to a lack of primitives in OpenVINO for running the model directly on ARM processors, it is difficult to obtain a head to head comparison of the model's performance on the same. However, an external study [10], that runs MobileNet v2 SSD trained using the Common Objects in Context (COCO) dataset on various edge computing platforms and the Raspberry Pi 3 B+, can be taken into account as a yardstick measure. While the Pi posts a time of 654.0 ms for the benchmarking test, the Intel NCS 2 accelerator is able to get it done in 118.6 ms, thereby highlighting the performance benefit of using the same instead of using the on board resources of the Pi.

Table 1. Performance Benchmarks

Processing Device	**Latency**	**Throughput**	**Iterations**
Intel Neural Compute Stick 2	33.67 ms	118.724 FPS	7128
CPU (i5-7200U)	107.083 ms	37.00 FPS	2232

3.2. The Deployment System

Since the system is designed to be a modular one, with a microprocessor and related devices mounted to it, the choice of the UAV will depend on the deployment environment. For general purposes, it is assumed that a quadcopter UAV with a camera that can be fed into our microprocessor, is used. The camera should be a consumer grade color camera, while the microprocessor of choice for us is the Raspberry Pi 3B+, due to the fact that it is a supported platform for our accelerator, namely the NCS 2. This system can then be mounted to the UAV in a secure housing that prevents dust and other particles from affecting its functioning. The block diagram in Fig. 1 shows the discrete blocks of the proposed system.

The drone deployment can be undertaken in various configurations. The patrol-confirmation configuration can be used, or a singular drone can be assigned a fixed flight path for surveillance. In the event a fire is detected, the drone can communicate the necessary command to the actuators on the ground, using the LPWAN protocol, as shown in Fig. 3.

The actuators could be chosen from a variety of available options, depending upon the local wildlife and the deployment area. For instance, in areas with lots of elephants, speakers that emit the sound of a swarm of bees can be used to drive them away. Or, connected fence posts with acorn scents that can be used to attract deer to the edge of forests away from area where the fire might spread. Ultrasonic sound waves can also be used to drive wild animals away. The choice of actuators will ultimately depend on the deployment environment and can be made accordingly. At the same time, care should be taken that the actuators used should not make the animals panic, instead dissuading them from entering certain areas, or drawing them away from them will be sufficient.

Algorithm 1 Pseudocode for process

$fireDetect \leftarrow$ inference output
if $fireDetect$ **then**
  $actuator \leftarrow$ activate
  alert base
else
  continue
end if

The UAV will cover a predesignated area of the assigned forest, while constantly running inference on the video stream captured by the connected camera. It will be equipped with a self-flying feature, as is found in many modern drones. In the event a fire is detected, the Raspberry Pi will establish a connection with the actuators on the ground and activate them so that they can perform the first response functions they are designed for, in order to reduce the extent of the damage. At the same time, it will alert the base station so that the associated fire department and other first responders can attend to the situation as soon as possible. This system will aim to facilitate detection of fires in the early stages, to prevent them from reaching a point where they become difficult to control and pose a threat to life.

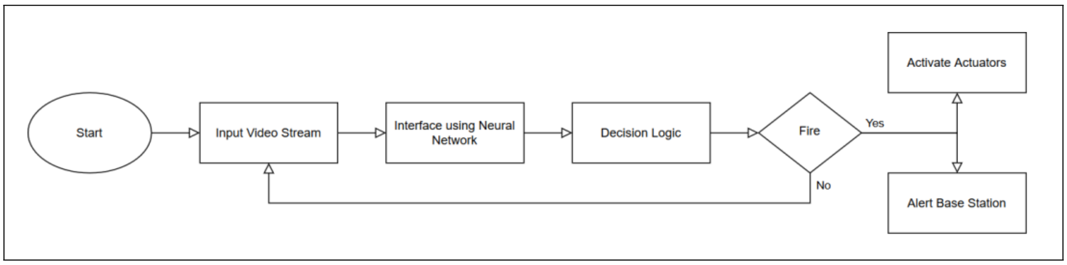

Figure 3. System Design

3.3. The Communication Protocol

The communication protocol to be used should support long range communication, owing to the fact that forests often cover large swathes of land. In such a layout, there may be a large distance between the drone and the actuator that needs to be activated. In addition, the fact that the drone will be up in the air while the actuators will be largely ground based, the distance may be larger still. As a result, we decided to go with an LPWAN (Low Power Wide Area Networks) technology. Specifically, SigFox and LoRa prove to be better for large area coverage. SigFox networks can cover entire cities with one base station, with a range of more than 40 km [11]. The exact protocol to be used will be a function of the geography of the deployment region and the distribution of the actuators across the same.

4. Related Work

Two major methods for detecting forest fires are covered in this section: wireless sensor networks and UAV-based systems.

4.1. Wireless Sensor Networks

Wireless Sensor Networks (WSN) rely on a widespread deployment of sensors in the form of constrained nodes in a network. Groups of nodes are regarded as clusters, with all nodes from that cluster reporting to the respective cluster head. The sensor nodes themselves can be used to measure a wide range of values from temperature to smoke levels, and are aware of their location. These features of WSNs make them an attractive option for deployments which require the monitoring of large scale areas [12] [13]. WSNs continue to evolve and get more energy efficient with time as new paradigms are introduced. Their throughput, efficiency and lifetime have gone from being highly dependent on battery size to being a function of the routing protocols and will continue to develop in the future with more advancements and newer models[14].

The need of the hour is to use an approach that has proved to be more sustainable and viable than traditional WSNs. While dealing with node failures and reliability of information in a WSN pose challenges, multiple paradigms and algorithms exist that can assuage these issues to acceptable extents. However, the issue of deploying and maintaining a WSN across wide, thickly forested areas poses a problem in and of itself.

Our proposed system utilizes a radically different approach to monitoring and sensing fires. The UAV and edge computing based paradigm concentrates most of the system into one deployment unit, thereby reducing the possible points of failure drastically, while also ensuring maintainability.

4.2. UAV-Based Systems

Drones can be deployed to cater to a wide range of needs, from wildlife conservation and search-and-rescue to surveillance. However, dealing with the processing of input video streams often proves to be a challenge in the current context. There do, however, exist certain solutions that can help optimize processing and deal with the restraints concerned with bandwidth, processing capacity, result accuracy and the timely delivery of results. Certain solutions [15] exist which can be used to efficiently distribute computing across drones and edge computing devices to carry out analysis on live video without clogging the bandwidth available trying to offload computation to remote servers. However, such designs depend on novel architectures that could prove to be difficult to deploy and maintain. On the other hand, our system utilizes off-the- shelf components which are readily available and have easy troubleshooting support owing to the vendors' supplementary material and forums.

Our system also works as a potential proof-of-concept for edge computing deployments to cater to fire detection purposes. Different neural networks that incorporate different features can be utilized to provide a more target specific detection scheme. This feature can be used to develop, prototype and test different neural networks to ascertain which ones work best for a specific target region, alleviating the need to develop special hardware for the purposes of prototyping.

4. Conclusion and Future Work

This system has been designed to be an independent, self sufficient system to find and quell fires in the early stages, before they have the chance to grow to dangerous proportions. The proposed system aims to provide continuous coverage and monitoring of forests. The UAV mounted Raspberry Pi monitors the video feed provided to it by the camera, and, assisted by the edge based VPU, runs inference on the video stream, taking necessary action as and when required. The actuators are selected such that they can help evacuate animals from the area affected by the fire. The Pi also sends an alert to the base station in order to alert the relevant authorities and get the fire under control.

The proposed system needs to be put together and deployed in a test region in real world scenarios to ascertain the efficiency and reliability of the system. Once the necessary purchases are made, the system can be put together with relative ease and deployed in the target environment. There is also scope for testing other accelerators for the purpose, besides the Neural Compute Stick 2 used here, in order to gauge relative performance.

References

[1] V. Sherstjuk, M. Zharikova, and I. Sokol, "Forest fire monitoring system based on uav team, remote sensing, and image processing," in *2018 IEEE Second International Conference on Data Stream Mining & Processing (DSMP)*, pp. 590–594, IEEE, 2018.

[2] D. Kinaneva, G. Hristov, J. Raychev, and P. Zahariev, "Early forest fire detection using drones and artificial intelligence," in *2019 42nd Inter- national Convention on Information and Communication Technology, Electronics and Microelectronics (MIPRO)*, pp. 1060–1065, IEEE, 2019.

[3] *https://www.nifc.gov/fireInfo/nfn.htm*.

[4] *https://www.iii.org/fact-statistic/facts-statistics-wildfires*.

[5] S. S. Esfahlani, "Mixed reality and remote sensing application of un- manned aerial vehicle in fire and smoke detection," *Journal of Industrial Information Integration*, vol. 15, pp. 42–49, 2019.

[6] J. P. Queralta, T. Gia, Z. Zou, H. Tenhunen, and T. Westerlund, "Comparative study of lpwan technologies on unlicensed bands for m2m communication in the iot: beyond lora and lorawan," *Procedia Computer Science*, vol. 155, pp. 343–350, 2019.

[7] *https://software.intel.com/en-us/neural-compute-stick*.

[8] A. J. Dunnings and T. P. Breckon, "Experimentally defined convolutional neural network architecture variants for non-temporal real-time fire detection," in *2018 25th IEEE International Conference on Image Processing (ICIP)*, pp. 1558–1562, IEEE, 2018.

[9] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich, "Going deeper with convolutions," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 1–9, 2015.

[10] *https://medium.com/@aallan/benchmarking-edge-computing- ce3f13942245*.

[11] K. Mekki, E. Bajic, F. Chaxel, and F. Meyer, "A comparative study of lpwan technologies for large-scale iot deployment," *ICT express*, vol. 5, no. 1, pp. 1–7, 2019.

[12] L. Yu, N. Wang, and X. Meng, "Real-time forest fire detection with wireless sensor networks," in *Proceedings. 2005 International Conference on Wireless Communications, Networking and Mobile Computing, 2005.*, vol. 2, pp. 1214–1217, IEEE, 2005.

[13] J. Lloret, M. Garcia, D. Bri, and S. Sendra, "A wireless sensor network deployment for rural and forest fire detection and verification," *sensors*, vol. 9, no. 11, pp. 8722–8747, 2009.

[14] A. Krishnamoorthy and V. Vijayarajan, "Energy aware routing technique based on markov model in wireless sensor network," *International Journal of Computers and Applications*, vol. 42, no. 1, pp. 23–29, 2020.

[15] J. Wang, Z. Feng, Z. Chen, S. George, M. Bala, P. Pillai, S.-W. Yang, and M. Satyanarayanan, "Bandwidth-efficient live video analytics for drones via edge computing," in 2018 *IEEE/ACM Symposium on Edge Computing (SEC)*, pp. 159–173, IEEE, 2018.

ISSN: 2005-4297 IJCA