

Deployment of a UAV-Based Fire Detection System

Rushiv Arora
College of Information
and Computer Sciences
UMass, Amherst
rrarora@cs.umass.edu

Mohammadjavad Khosravi
Electrical and
Computer Engineering
UMass, Amherst
mkhosravi@umass.edu

Saeede Enayati
Electrical and
Computer Engineering
UMass, Amherst
senayati@umass.edu

Hossein Pishro-Nik
Electrical and
Computer Engineering
UMass, Amherst
pishro@ecs.umass.edu

Abstract—In this paper, we design and implement a fire detection algorithm using an unmanned aerial vehicle (UAV). In particular, we consider a scenario where a UAV is employed to find a fire spot where the fire is in its early stages. To this end, we first propose a path planning algorithm where the goal is to make sure that the UAV finds the fire in the shortest amount of time. During this stage, the UAV visits different parts of the area, takes an image from each part, and sends the image to the control center for further image processing. Then, we deploy a machine learning (ML) model, which is a residual neural network (ResNet), to process the image and determine whether a fire is detected or not. The ML algorithm has been trained using images taken by the drone and images from the Internet. Through experimental results, we show that the proposed path-planning algorithm along with the ML model can detect fire efficiently. The problem and the proposed solution in this paper can be also applied to a search and rescue scenario, where for example, a hiker is missing in a remote area.

Index Terms—Unmanned aerial vehicles, Machine learning, Computer vision, Fire detection, Uniform network coverage.

I. INTRODUCTION

The global community, particularly regions like Australia and California, has observed the severe environmental, economic, and life-disrupting consequences of devastating wildfires [1]–[4]. According to the USDA research data archive, nearly 85% of wildfires are caused by human activities such as camping, arson, and smoking [5].

Several technologies have been used for fire detection and monitoring including ground sensors or satellite imaging. However, these methods are not yet able to offer a fast and reliable solution for wildfire detection and monitoring. Some drawbacks of the current technologies include: i) delayed fire detection due to missing small fires at early stages, ii) relatively long time lag for satellites to overpass the field, and iii) infeasibility of deploying sensors with limited sensing distance ranges (e.g., chemical-based smoke detectors). For instance, smoke detectors are efficient in detecting fires at early stages, but they suffer from short distance ranges [6].

Unmanned aerial vehicles (UAVs) can be used in wildfire detection missions as they can fly in low altitudes to detect the fire in its early stages. They are also agile and can exhibit autonomous behaviors at low operating costs and time-efficient way.

This work was supported by NSF under grants CNS-1932326 and CNS-2150832.

In this paper, we propose a fire detection mission using a UAV where a fire is in its early stages. In this regard, a path planning algorithm along with a machine learning (ML) algorithm has been developed for the drone to find the shortest path to the target while the target's location is not known. Specifically, a residual neural network (ResNet) is utilized in which the dataset used for the training and validation is a combination of the existing datasets as well as pictures taken by the drone where in case of fire surveillance, mostly include small fire spots or smoke in a rural area with other objects around (spare). This is because the goal is to achieve an early detection where the fire is in its initial stages. This problem and the proposed solution can be applied to a rescue mission as well where a hiker is missing. This scenario has been considered in the longer version of this paper [7].

The two algorithms work in series where first the path planning is implemented and then at each time that the UAV visits a new section of the area, it takes an image and sends it back to the ML algorithm at the control center. We show that the proposed algorithms can detect a fire with a high accuracy in the shortest amount of time.

This paper is organized as follows: In Section II, we provide a literature review of the related work. In Section III, we explain the system model including the Camera model, the ML model and the combined algorithms. Section IV provides the results and Section V concludes the paper.

II. RELATED WORK

As far as target detection is concerned, using UAVs in wildfire detection has attracted a lot of attention as well in recent years [8]–[18]. In [8], a detection method utilizing both color and motion features is investigated for UAV-based fire detection. The same authors in [9] developed a detection algorithm using Infrared images in order to detect fire pixels. Infrared images were also used in [10] to infer occurrence of forest fires. Using smoke sensors, a smoke detection algorithm was proposed in [11]. [12] uses the satellite images and infrared sensors to detect smoke and forest fires. In [13], in order to use a smaller dataset and reduce the computational complexity, the authors propose to use a pretrained mobileNetV2 architecture to implement transfer learning. An early wildfire detection mechanism was proposed in [14] using air quality and LiDAR sensors. Thermal infrared imaging, and oblique photogrammetry technologies were exploited in [15] to create a 3-D land surface temperature

model of a Coal Fire. In [16], a multi-modal UAV-collected dataset of dual-feed side-by-side videos of a prescribed fire was developed along with a deep learning-based methodology where the authors reached to an accuracy much higher than the usual single-channel video feeds. Other algorithms for fire detection such as YOLOv3 and RepVGG-YOLOv5In were applied in [17] and [18], respectively.

In wildfire management, a coordination problem was considered in [19] where UAVs were used along with unmanned ground vehicles (UGVs) to fight the fire front. In [20] and [21], the authors aim to provide a surveillance system for monitoring high risk areas for smoke. They use a fixed-wing drone that monitors fires at higher altitudes using an optical and thermal camera, and a rotary-wing drone that will then verify if the positives reported by the fixed wing drone are positive or negative. Deep Q-learning was exploited in [22] to monitor a wildfire front with multiple UAVs. A reinforcement learning-based fire warning and suppression system Using surveillance and firefighting UAVs was proposed in [23]. Furthermore, investigating optimal UAV coalition to fully cover the area, spectrum sharing plus cell assignment, optimal number of UAV and IoT devices for a maximum detection probability, are of the other problems that have been considered in wildfire detections [24]–[27].

III. SYSTEM MODEL

A. Drone and Area Models

We consider a scenario where a UAV is employed to fly at a fixed height H over a relatively large area to look for a potential fire spot where the fire is in its early stages. The UAV considered in this work is a DJI Mavic 2 Zoom [28] with a 4k camera shown in Figure 1.



FIGURE 1: DJI Mavic 2 Zoom with a 4k camera.

In order to apply the path planning algorithm, we need to partition the desired area in same-size segments. In other words, we divide the area into cells of equal size, and at the center of each cell, we place a hypothetical point. The UAV is assumed to have fully covered the cell once it reaches this point. The size of a cell is calculated from the field of view (FOV) of the camera which is the area covered by UAV's camera when it is flying at an altitude H . Considering a rectangular cell with the

width w and height l , the projected area's size can be obtained by the following equations

$$w = 2H \tan\left(\frac{\alpha}{2}\right);$$

$$l = 2H \tan\left(\frac{\beta}{2}\right);$$

where α and β are vertical and horizontal FOV of the camera, respectively.

The complete path is then a list of coordinates, termed as waypoints, from which the drone moves to another until it visits all waypoints or finds the object. To make sure that the drone covers the area completely, projected areas must overlap. Note that the amount of overlap can vary on each side. The horizontal and vertical overlaps are denoted as r_x and r_y , respectively.

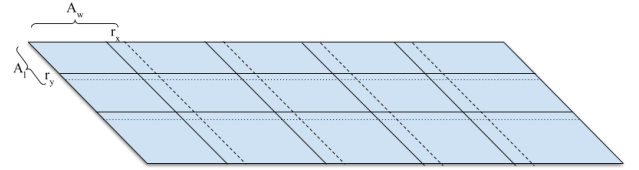


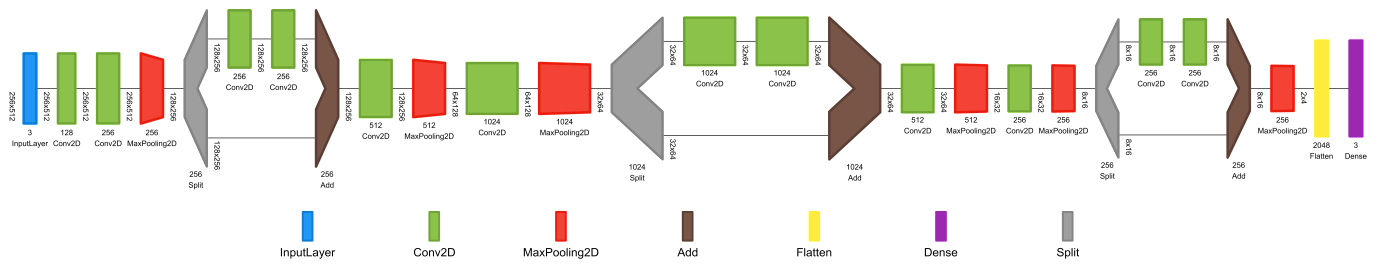
FIGURE 2: Projected areas with overlaps. Centers of rectangles are the path waypoints.

B. Machine Learning

The deep learning model considered in this paper is a Residual Neural Network (ResNet) [29], which is a class of deep neural networks with less complexity than VGG and plain networks. ResNets are known for dealing with the vanishing gradient problem [30]. ResNets approach the vanishing gradient problem by adding shortcut connections to skip one or more layers such that the input of the skipped layers is added to the output of the skipped layers. These identity shortcut connections add neither extra parameter nor computational complexity [29].

Data: The dataset consists of 2646 image divided into 3 classes: Fire, Smoke, and Spare which is neither Fire nor Smoke. Out of the 2646, 987 are Fire, 781 are Smoke, and 818 are Spare. The training data consists of 289 live images captured by the drone at heights of 15-30m as well as images carefully and manually selected from the Internet. The drone images are captured at different locations for the training and test processes. All images are then resized to 256 512 3 for height, width, and channels, respectively.

Model: The details of the ResNet model are highlighted in Figure 3. The designed ResNet has 6 convolution and 3 residual blocks. In other words, it consists of 12 convolutional layers, 12 activation layers, and 6 pooling layers. The convolutional layers consist of 33 kernels. Also, each convolutional block consists of a sequential and ordered arrangement of a convolutional layer, a batch normalization layer, a rectified linearity activation (ReLU) layer, and a maximum pooling layer. Finally, a residual block consists of two sequential convolutional blocks that are added to the input of the first convolutional blocks.



Training: The model is trained for 8 epochs using an Adam Optimizer with the Cross Entropy Loss Function. Training is further executed with maximum learning rate of 0:0005, a weight decay of 10^{-4} , and a gradient clipping of 0:1. A lower learning rate is compatible with the nature of the data as it can be seen that a larger learning rate will cause jumps around the global minima. The data is also preprocessed as follows:

with a fixed probability. Since this is done with every epoch, the model observes new images with each epoch of training and hence the size of the dataset is increased artificially. An example of the data augmentation is shown in Figure 5.

C. Integrated Model

A special action is created by inheriting the `DJIAction` class. The purpose of this action is to receive an image as an input

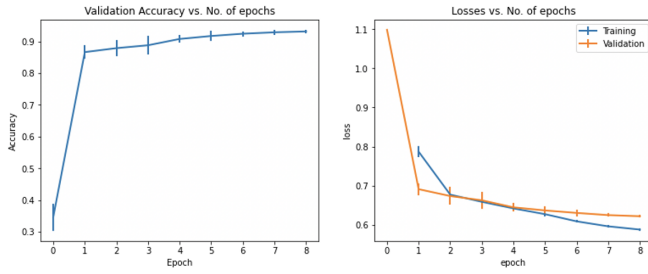


FIGURE 6: Training and Validation plots for the model over 30 seeds. The average values are plotted with the first standard deviation as confidence intervals.

Algorithm 1 Custom Machine Learning Action.

```

Input: Signal from Timeline to start execution. Call to the run() function
Result: Execution of the action with call to Mission Control
class result of ML Recognition. Global variable
image Null if Fetch Camera then
  if Set Media Download Mode then
    if Download Media List then
      image required image from list
      if Run ML Recognition then
        result 0—1—2 Fire—Smoke—Spare
        return Tell Mission Control that Execution was successful
      end
    else return ML Action Error to Mission Control ;
  end
end
else return Media Download Error to Mission Control ;
end
else return Set Media Mode Error to Mission Control ;
end
else return Camera Fetch Error to Mission Control ;

```

and generate a probability distribution across the 3 classes. DJI specific details of this custom machine learning action are highlighted in Algorithm 1.

Once these machine learning actions are created, the timeline mission needs to be scheduled in order to create the custom trajectory, as detailed in Algorithm 2.

IV. EXPERIMENT AND SIMULATION RESULTS

A. Simulation Results

To ensure entire area is surveyed while keeping the distance traveled by the drone within the expected range, certain assumptions are made when designing the path planning:

- 1) The Earth at the location of testing (Amherst, Massachusetts) is assumed to be a sphere, and the radius of the earth for calculations is selected according to the latitude. The selected radius is 6378137m [33] which is obtained using the following equation

$$R(\phi) = \frac{s}{\frac{(a^2 \cos^2 \phi) + (b^2 \sin^2 \phi)}{(a \cos \phi)^2 + (b \sin \phi)^2}} \quad (1)$$

where $R(\phi)$ is the geocentric radius of geocentric latitude, ϕ , and a and b are the equatorial and polar radii, respectively.

- 2) An accuracy of 90% is assumed using the Haversine formula [34] in order to estimate the calculations. While the goal is to remain slightly under the expected distance,

Algorithm 2 Generating the Timeline Mission.

```

Input: Path P = fp1; p2; ::; png the path to be followed. Each point is a
Latitude, Longitude pair
Input: Altitude h of the mission
Result: Timeline Mission T = fm1; m2; ::; mng a Timeline Mission that
will be executed by the Mission Control

T ;;
for point 2 Path P do
  Waypoint Mission w new WaypointMission
  w.add(point)&w.point1.height h
  T T [ w
  MAction ML new MAction
  T T [ ML
end
return T
// Set up action listeners to the Timeline
while Timeline T do
  if action = ML Action then
    if action:result == Fire then
      Clear Timeline T in Mission Control
      Hotpoint Mission hp new HotpointMission
      hp.add(currLocation)&hp.point.height h
      // Optional: Circle the Fire with Hotpoint
      T T [ hp
    end
  end
end
end

```

the overlap between grids is minimized to still ensure maximum area coverage.

The accuracy of these assumptions has been tested in the DJI simulator which is available as part of the DJI Assistant for Mavic 2 software. The software provides the drone's location correct up to 10 decimal points in the simulator. The euclidean distances traveled by the drone were compared to the expected distances at 30 different heights, ranging between 10m to 70m at intervals of 2. At each height, 15 waypoints were recorded and the experiment was repeated 3 times to get a total of 90 tests and 1350 waypoints.

The average accuracy of the distance moved by the drone can be summarized in table I.

TABLE I: Accuracy of the distances moved.

	Longitudinal	Latitudinal
Accuracy	99.48%	99.86%
Standard Deviation	3.7%	1.7%
Variance	0.14%	0.01%

B. Experiment Results

We have used 360 images taken by the drone including 120 images for each class. The images were tested at heights between 10m to 20m. The test images were taken at two separate locations each of which were different from the training and validation images locations. Depending on the availability and nature of the area surrounding the fire spot, a disturbance level was added to the testing images. As an example, during the Fire or Smoke tests, the drone camera settings were modified to include either half or the entire

surrounding tree in the frame. This was done to intentionally perplex the ML algorithm and make it difficult to discern whether the image captured represented Smoke or Spare. This testing was conducted for varying amounts of environmental disturbance. Figure 7 shows this operation.

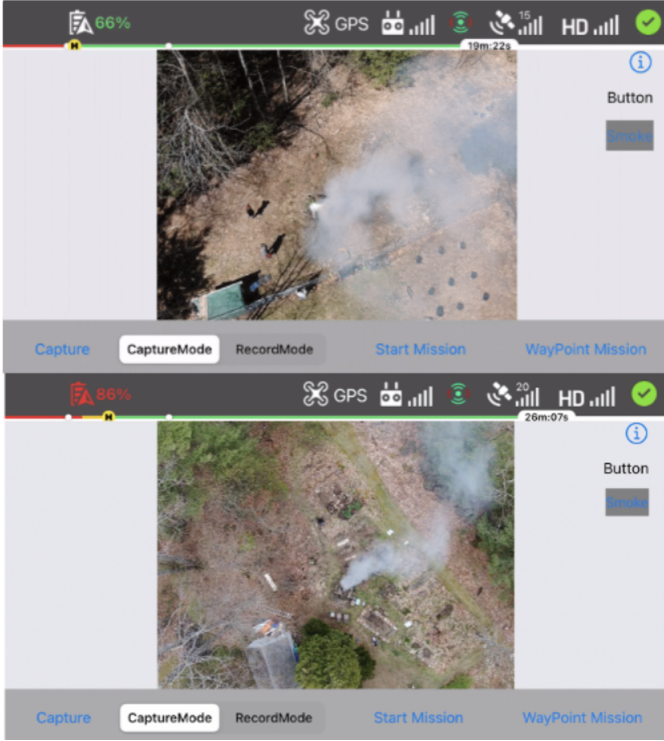


FIGURE 7: Smoke detection at a height of 35m at two different test sites. The green color from the trees in the bottom image is intentional to distract the ML model from detecting smoke.

It has been observed that the largest distance for detecting Fire is 20.2m for the fire pit and 28m for the controlled fire, while the highest distance for detecting Smoke is 35m.

TABLE II: Classification metrics on the test data.

	Precision	Recall	F1-Score	Support
Fire	0.99	0.74	0.85	120
Smoke	0.91	0.95	0.93	120
Spare	0.82	0.99	0.90	120
Accuracy			0.89	360

We present the classification performance in Table II. For this table, the accuracy of each Fire, Smoke, and Spare class is obtained as 74.3%, 94.2%, and 98.3%, respectively, which shows that Fire has the lowest accuracy. This is due to the nature of the fire that the tests were conducted on. In fact, for the sake of safety and in accordance with the Massachusetts state laws, small size of fire pits were used. This made the fire detection from heights larger than 15m and above difficult. Another reason is that when viewing the fire vertically from the above, it is difficult to see the flames as it is covered by sticks and logs.

From Table II, it can be seen that the Fire class has the highest precision. This is due to the fact that extremely few false positives have been observed. The Smoke dataset had fewer false positives as well, and hence had a precision of 0.91%. The Spare class, on the other hand, has the lowest precision since many misclassifications in the Fire testing were classified as Spare when the flames were not visible, giving rise to a larger number of false positives. A highest precision for Fire and Smoke is advantageous as it is indicative of low number of false positives. The Fire class has a lower Recall, while the Smoke and Spare classes have a significantly higher recall which implies that the Fire class has a larger number of False Negatives while the Smoke and Spare classes do not.

Note that it is more feasible to test Smoke rather than Fire due to the possibility of generating larger amounts of smoke with less restrictions. Therefore, since more feasible tests were conducted on Smoke rather than fire, the testing represented an early forest fire, and this led to the results of getting a 94.2% accuracy with a maximum detection height of 35m.

The spare dataset reported the highest accuracy with a value of 98.3%, which only misclassified two images in the 120 images. The Spare class gives a better performance due to the presence of spare data everywhere. Some of the objects on which the test was conducted are: trees, grass, leafless trees, mulch, cars, houses, chairs and tables, and porches amongst others.

Finally, to deal with the assumption that the lower accuracy of the Fire was caused due to less visible flames, a significantly larger controlled burn was conducted in open burning season. Therefore, to boost the flame slightly at the moment of testing, a low-power leaf blower and combustible pine needles were used. The new results are shown in Table III.

TABLE III: Classification metrics of the modified testing for the Fire class.

	Precision	Recall	F1-Score	Support
Fire	0.99	0.93	0.96	100
Smoke	0.97	0.95	0.96	120
Spare	0.92	0.99	0.96	120
Accuracy			0.96	360

According to Table III it can be seen that the metrics are improved. With a higher accuracy reported with more visible flames, the prediction is that on an early forest fire, which is significantly larger than a campfire, the results will be more accurate. Figure 8 shows a boosted campfire.

V. CONCLUSION

This paper suggests a UAV-driven system to address the early fire detection. In particular, a scenario was considered in which the objective is to detect a potential fire in a given area. The proposed solution involves devising a path planning algorithm and an ML model. The ML model which is a residual neural network, has been trained, validated and tested using new image dataset captured by the drone for the early fire images, as well as the existing ones from the Internet. The results are

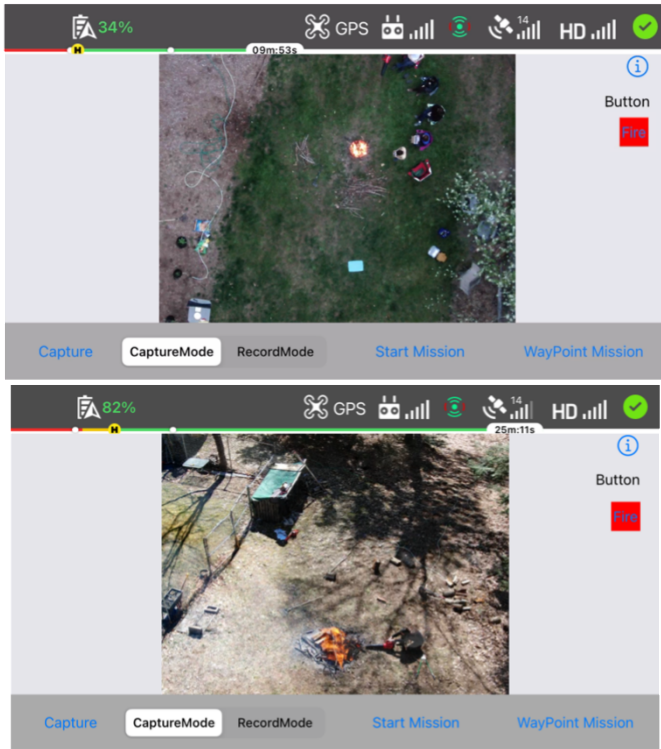


FIGURE 8: Fire detection from a height of 22m and 28m in the top and bottom pictures, respectively. The top picture is the smaller campfire with flames of 1.5ft, and the bottom picture is the controlled fire boosted by a leaf blower to heights > 3ft.

verified through simulation and implementation where it has been observed that the proposed method can efficiently detect the fire at the early stages with high accuracy and precision.

REFERENCES

- [1] P. Gupta and et al., "Impact of California Fires on Local and Regional Air Quality: The Role of a Low-Cost Sensor Network and Satellite Observations," *GeoHealth*, vol. 2, no. 6, pp. 172–181.
- [2] A. Filkov and et al., "Impact of Australia's catastrophic 2019/20 bushfire season on communities and environment. Retrospective analysis and current trends," *Journal of Safety Science and Resilience*, vol. 1, 2020.
- [3] N. Querolo and B. K. Sullivan, "California Fire Damage Estimated at \$25.4 Billion," *Bloomberg*, October 2019.
- [4] C. Government, "2018 Incident Archive," *Cal Fire*, 2019.
- [5] U. D. of the Interior, "Wildfire Causes and Evaluations (U.S. National Park Service)," *National Parks Service*.
- [6] T. M. Cabreira, L. B. Brisolara, and P. R. Ferreira Jr, "Survey on coverage path planning with unmanned aerial vehicles," *Drones*, vol. 3-1, 2019.
- [7] M. Khosravi, R. Arora, S. Enayati, and H. Pishro-Nik, "A search and detection autonomous drone system: From design to implementation," *arXiv preprint arXiv:2211.15866*, 2022.
- [8] C. Yuan, Z. Liu, and Y. Zhang, "Vision-based forest fire detection in aerial images for firefighting using UAVs," in *International Conference on Unmanned Aircraft Systems (ICUAS)*, Arlington, VA, USA, July 2016.
- [9] —, "Fire detection using infrared images for UAV-based forest fire surveillance," in *International Conference on Unmanned Aircraft Systems (ICUAS)*, Miami, FL, USA, 2017, pp. 567–572.
- [10] I. Bosch and et al., "Multisensor network system for wildfire detection using infrared image processing," *The Scientific World Journal*, 2013.
- [11] P. Ramanatha and et al., "UAV based smoke plume detection system controlled via the short message service through the GSM network," in *2016 International Conference on Inventive Computation Technologies (ICICT)*, vol. 2, Coimbatore, India, Aug. 2016, pp. 1–4.
- [12] C. R. Pennypacker and et. al, "FUEGO — Fire Urgency Estimator in Geosynchronous Orbit — A Proposed Early-Warning Fire Detection System," *Remote Sensing*, vol. 5, no. 10, pp. 5173–5192, 2013.
- [13] H. Wu and et. al, "Transfer learning for wildfire identification in UAV imagery," in *54th Annual Conference on Information Sciences and Systems (CISS)*, Princeton, NJ, USA, March 2020, pp. 1–6.
- [14] D. Rjoub, A. Alsharoa, and A. Masadeh, "Early wildfire detection using UAVs integrated with air quality and LiDAR sensors," in *2022 IEEE 96th Vehicular Technology Conference (VTC2022-Fall)*, Londo, UK, 2022, pp. 1–5.
- [15] Z. Shao, Y. Liang, F. Tian, S. Song, and R. Deng, "Constructing 3-D land surface temperature model of local coal fires using UAV thermal images," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 60, pp. 1–9, 2022.
- [16] X. Chen, B. Hopkins, H. Wang, L. O'Neill, F. Afghah, A. Razi, P. Fule, J. Coen, E. Rowell, and A. Watts, "Wildland fire detection and monitoring using a drone-collected RGB/IR image dataset," *IEEE Access*, vol. 10, pp. 121301–121317, 2022.
- [17] C. Xiong, A. Yu, L. Rong, J. Huang, B. Wang, and H. Liu, "Fire detection system based on unmanned aerial vehicle," in *2021 IEEE International Conference on Emergency Science and Information Technology (ICESIT)*, Chongqing, China, 2021, pp. 302–306.
- [18] K. Song, Y. Zhang, B. Lu, W. Chi, and L. Sun, "UAV forest fire detection based on RepVGG-YOLOv5," in *2022 IEEE International Conference on Robotics and Biomimetics (ROBIO)*, Jinghong, China, 2022, pp. 1277–1282.
- [19] C. Phan and H. H. Liu, "A cooperative UAV/UGV platform for wildfire detection and fighting," in *Asia Simulation Conference - 7th International Conference on System Simulation and Scientific Computing*, Nov. 2008.
- [20] D. Kinaneva, G. Hristov, J. Raychev, and P. Zahariev, "Early forest fire detection using drones and artificial intelligence," in *2019 42nd International Convention on Information and Communication Technology, Electronics and Microelectronics (MIPRO)*, 2019, pp. 1060–1065.
- [21] G. D. Georgiev, G. Hristov, P. Zahariev, and D. Kinaneva, "Forest monitoring system for early fire detection based on convolutional neural network and uav imagery," in *2020 28th National Conference with International Participation (TELECOM)*, 2020, pp. 57–60.
- [22] A. Viseras, M. Meissner, and J. Marchal, "Wildfire front monitoring with multiple UAVs using deep Q-learning," *IEEE Access*, pp. 1–1, Jan. 2021.
- [23] F. H. Panahi, F. H. Panahi, and T. Ohtsuki, "A reinforcement learning-based fire warning and suppression system using unmanned aerial vehicles," *IEEE Transactions on Instrumentation and Measurement*, vol. 72, pp. 1–16, 2023.
- [24] F. Afghah, A. Razi, J. Chakareski, and J. Ashdown, "Wildfire monitoring in remote areas using autonomous unmanned aerial vehicles," in *IEEE INFOCOM 2019-IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS)*. IEEE, 2019, pp. 835–840.
- [25] A. Shamsoshoara and et. al, "An autonomous spectrum management scheme for unmanned aerial vehicle networks in disaster relief operations," *IEEE Access*, vol. 8, pp. 58064–58079, March 2020.
- [26] O. Bushnaq and A. Chaaban, "The role of UAV-IoT networks in future wildfire detection," *IEEE Internet-of-Things Journal*, 2021.
- [27] R. B. Zadeh, A. Zaslavsky, S. W. Loke, and S. MahmoudZadeh, "A multiagent mission coordination system for continuous situational awareness of bushfires," *IEEE Transactions on Automation Science and Engineering*, pp. 1–0, 2022.
- [28] S. D. T. C. Ltd., "Mavic 2 - Product Information - DJI," *DJI Official*.
- [29] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," *CoRR*, 2015.
- [30] S. Hochreiter, "The vanishing gradient problem during learning recurrent neural nets and problem solutions," *International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems*, vol. 6-02, pp. 107–116, 1998.
- [31] A. Bauerle and T. Ropinski, "Net2Vis: Transforming Deep Convolutional Networks into Publication-Ready Visualizations," *CoRR*, vol. abs/1902.04394, 2019.
- [32] S. Ioffe and C. Szegedy, "Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift," *CoRR*, vol. abs/1502.03167, 2015.
- [33] L. McNish, *RASC Calgary Centre - Latitude and Longitude*.
- [34] C. C. Robusto, "The Cosine-Haversine Formula," *The American Mathematical Monthly*, vol. 64, no. 1, pp. 38–40, 1957.