

An Early Forest Fire Detection System Based On M300 Drone and H20T Camera

1st Given Name Surname

dept. name of organization (of Aff.)

name of organization (of Aff.)

City, Country

email address or ORCID

2nd Given Name Surname

dept. name of organization (of Aff.)

name of organization (of Aff.)

City, Country

email address or ORCID

Abstract—This paper presents a drone-based early forest fire detection system. Using multiple aerial sensors, thermal images, RGB images, and distance between forest fire points and drones can be captured and measured from the air. To take advantage of data from different sources for forest fire detecting and confirming, both deep learning-based and traditional computer vision algorithms are employed. The onboard computer and ground station computer are designed to work collaboratively according to the different complexity and computational demands of sub-modules in this system. Integrating different sensor data and a two-phase strategy of potential early forest fire detection and confirmation , the proposed system achieves a relatively low false alarm rate and has good robustness in the outdoor early forest fire detecting experiments.

I. INTRODUCTION

Forest fire has significant effects on the ecosystem, global climate and human beings. Directly, the forest fire reduces the forest coverage, burns the vegetation, and the animal resources. Moreover, potential and side effects generated by the forest fire can produce long-term and continuous damages to the environment and climate: the forest fire will lower organic matters and water in the soils [1], change the ground surface acid heavily for vegetation restoration [2]; on the other hand, burning destroys the forest as the “Lung of Earth” and emits huge amounts of greenhouse gases into the atmosphere [3], which deteriorate climate change [4]. The forest fire now gradually becomes a global challenge. According to [5], the “Black Summer” fire in southeast Australia directly killed 33 lives and destroyed over 3000 houses; indirectly health impacts from this forest fire attributed to 417 lives lost and 3151 hospitalisations; in China and United States, however, 190kha and 830kha of land have been burned so far in 2021 separately [6]. Early forest fire detection and alarm, as a result, can minimize the loss by preventing the forest fire from turning into a disaster during the incipient period.

Temporally, forest fire development can be simply divided into 6 phases: (1) incipient phase, (2) growth phase (pre-flashover), (3) flashover, (4) fully developed phase (post-flashover), (5) decay phase, and (6) extinction [7]. In the incipient period of the forest fire, there usually no strong smoke or considerable flames, and the ignited zone could be very small, which may bring about difficulties for the wide-range but low-resolution detection methods. For example,

VIIRS Satellites [8] have resolution with $375m$ and $12h$ revisit time which is still inadequate to detect the small-range early forest fire.

In the early days of forest fire monitoring, human observers and watchtowers are employed to acquire the Spatio-temporal information of the forest fire. Apparently, watchtowers lack the flexibility and wardens potentially suffer danger [9]. With the development of electronic and embedded systems, a forest fire can be monitored by the pre-placed sensors, including the temperature sensors, smoke detector, etc. These pre-placed sensors can also network with the IoT (Internet of Things) [10].But such solution heavily depends on the communication capability and the power supplement [11]. Implementation of these networks could also be unfeasible when monitoring the primeval forest. Another approach is online monitoring the forest fire using remote sensing. However, remote sensing-based methods also have backward: satellite images and data usually have wider range but relatively longer revisit time [12]; Manned or manually-operated aircraft costs higher human labour [9].

Drones, on the other hand, have multiple advantages in terms of the early forest fire detection: relative low cost [2]; with onboard computer, decision making, mission planning, motion controlling can be executed onboard automatically; multiple types of sensors can be mounted as the payload of the drones, such as RGB camera, infrared camera, laser rangefinder, and multi-spectral camera, etc. to get various information of the forest fire; Due to the high flexibility, drones are less likely to suffer connection failure or the failure of power supplement.

In this work, we present an online early forest fire detection system based on drone platform. Multiple types of sensors are employed to detect multiple features of the fire flame and smoke. The hardware and system architecture are illustrated in Section III. Both deep convolutional neural networks (CNN) and traditional computer vision algorithms are implemented to process the visual images (RGB images) and infrared images (thermal images), which will be presented in Section IV. The outdoor experiment results are given in Section V. Finally, Section VI summarizes this paper and prospects the future work. For the benefit of community, we make our source code

public, which can be found on Github¹.

II. RELATED WORKS

Compared with watchtowers, satellites and pre-placed sensors, the main advantage of drone-based platform for forest fire detection missions are the flexibility of the drone as well as multiple types of payload sensors.

A. RGB Camera

RGB camera, due to its low price and intuitive results, is extensively applied in forest fire detection missions. Methods and algorithms in the computer vision field can be directly implemented in the forest fire images processing tasks: RGB, HSV, and HSI color space, Ostu segmentation method [13] and optical flow [9] [14] are implemented to detect, segment, and track the forest fire flame. Fuzzy logic and extended Kalman filter are combined in [15] to segment the forest fire smoke. In [16], the authors proposed 7 segmentation rules based on the YCbCr and RGB color space to segment the forest fire pixel.

Compared with threshold and rules-based methods, combining the human-defined features (color, texture, motion features, etc.) with traditional machine learning algorithm(random forest, support vector machine, etc.) usually achieves higher accuracy. Flame texture and color feature in YCbCr space are extracted as forest fire features in [17], where an extreme learning machine is trained to classify through these extracted features. Multiple features and learning-based methods cross experiments have been conducted and compared in [18].

The deep learning-based image processing methods represented by convolutional neural network (CNN) extract more types of features. These deep learning-based methods can be categorized into three main applications: classification, semantic and instance segmentation, and object detection [19]. Several classification algorithms based on CNN including VGG16, Resnet50, and MobileNet, etc. have been implemented with DJI M100 and Odroid-XU4 onboard computer [20]. Based on Faster-RCNN, a spatial texture analysis forest fire flame detection algorithm is proposed [21]. In [22], DJI M210 is used as the drone platform, on which deep learning algorithms are implemented to realize pre-mentioned three main tasks. Many general and popular object detection and segmentation framework have been applied in the forest fire detection task: YOLOv3 for fire detection is trained and implemented on the DJI drones and Manifold onboard computer in [23] and [24]. 360-degree sensor and region-based method is proposed to reduce the false alarm generated by the DeepLabV3+ neural network framework in [25]. Using the RGB image patches and CNN, the forest fire flame is classified and roughly located in [26].

B. Infrared Camera

Infrared camera directly generates thermal images through capturing information that reflects the abnormal heat distribution of the forest fire. Each pixel in thermal images stands for the temperature of that corresponding point.

To alignment the RGB and thermal images, homography matrix without the object depth is calculated and used in [27]. In [28], Gabor and fractal feature combined with simple linear iterative clustering method is used to segment the smoke through the thermal images. A DJI M600 drone based forest fire detection, localization, and suppression system is proposed in [29], in which thermal images is used to segment forest fire pixels. Besides the value of each pixel in the thermal images, motion features of the forest fire can also be used for detecting: Optical flow and Otsu segmentation methods are employed to segment and track the fire using thermal images in [30].

III. DRONE BASED EARLY FOREST FIRE DETECTION PLATFORM

A. System Architecture

1) *Hardware:* For the onboard sensors, the ZenMuse H20T camera is chosen. It embedded wide camera, zoom camera, infrared camera, and laser rangefinder in one payload, which significantly simplifies the payload mounting and improves the system reliability. DJI M300 Drone is selected as the platform. The chosen hardware are shown Fig.1.

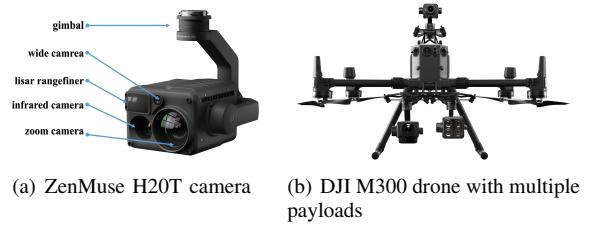


Fig. 1. ZenMuse H20T camera and DJI M300 drone

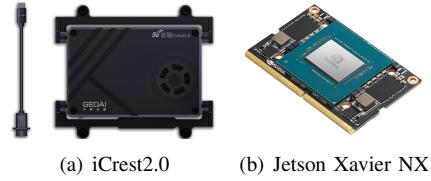


Fig. 2. iCrest2.0 onboard computer and NVIDIA Jetson Xavier NX

The onboard computer is the computational unit for real-time high level mission control, image processing, and motion control. It should equip with GPU(s) intending to accelerate the inference of deep learning-based image processing. Thus, the iCrest 2.0 onboard computer provided by GEOAI and NVIDIA is chosen for this work; this computer is based on the NVIDIA Jetson Xavier NX, which can bring up to 21 TOPs accelerated computing power to run modern neural networks in parallel and process data from multiple high-resolution sensors. With the development of GEOAI, iCrest 2.0 has the ability to communicate with DJI M300 and H20T Camera online. Ground station computer should have the ability to monitor DJI M300 mission states and to accelerate a more

¹https://github.com/ConcordiaNAVlab/forest_fire_detection_system

complex image segmentation algorithm. Morden laptops can do this job. For outdoor experiment, we use Lenovo IdeaPad 15 with Intel i5-6300HQ CPU and NVIDIA GeForce GTX 950M GPU.

The proposed early forest fire detection system does not heavily rely on the hardware platform, which mainly benefits from the hardware abstraction of Robot Operating System (ROS). Theoretically, the proposed algorithm and software can be embedded in any platform, sensors with similar functions as we use.

2) *Software*: Software is developed based on the ROS [31] and DJI Onboard software development kit (SDK) [32], which provides communication between our software applications and the hardware abstraction. Sub-modules are designed to run on onboard computer and ground station separately as the following Fig.3 showing:

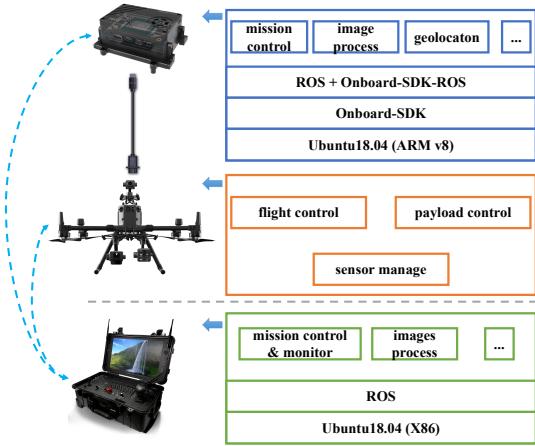


Fig. 3. proposed system software

Through the APIs (application programming interface) provided by the DJI onboard SDK, our applications running on iCrest 2.0 can access the sensor and autopilot data and send control commands to the flight control module and payload control module (such as gimbal driver, etc.) via a wired connection. A task manager and image segmentation applications are running on the ground station computer simultaneously. The former application is designed to monitor and control the whole detection mission; the latter is an image segmentation application as a “completion” of the onboard computer.

For the benefit of the real-time application, TensorRT developed by NVIDIA is used for optimizing and accelerating the deep learning-based algorithms inference [33]. According to the Pytroch and TensorRT implementation project website [34], Resnet18 can achieve around 7 times faster after using the TensorRT on NVIDIA Xavier NX.

B. System Functions

For the early forest fire detection mission, the proposed system performs the following functions:

- regular patrol and surveillance over the given region of the forest.

- potential early forest fire detection.
- online potential early forest fire region confirmation.
- early forest fire point geolocation.
- early forest fire alarming.

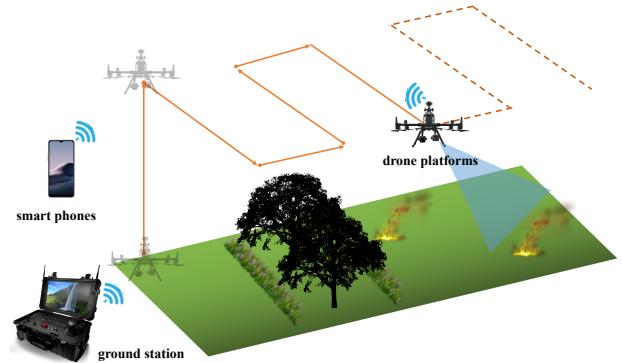


Fig. 4. typical early forest fire detection mission scenario

As shown in Fig.4, the early forest fire surveillance and detection mission started with a pre-defined path. During this stage, the drone will take off and fly along the zigzag path, making sure that each part of the given zone of the forest can be covered. The zigzag path planning should take the drones power, the region shape, and the sensor effectual working distance into consideration. Multiple drones can patrol over different forest regions according to the mission distribution. Aerial images captured from RGB zoom camera is processed by the Resnet18-based image classification algorithm, after which the current image is classified into 3 possible classes: normal, smoke and fire; simultaneously, computer vision-based segmentation algorithm marks the potential fire source with captured thermal images.

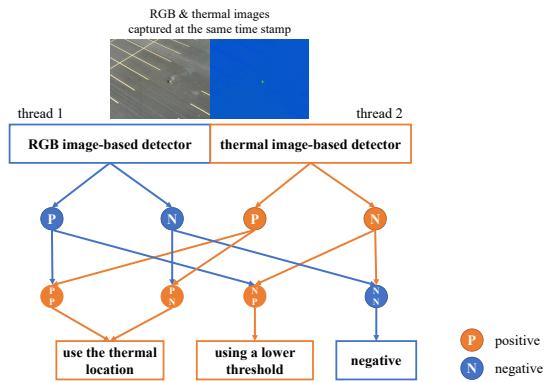


Fig. 5. confirmation request generating process

If the classification algorithm or the thermal image-based algorithm detects the potential forest fire, a confirmation request will be triggered. Otherwise, the drone(s) will keep flying along the pre-defined path until the mission finish or the power alarming triggers returning requests.

As shown in Fig.5, RGB and thermal images captured at the same time stamp will be fed into the 2 types of detectors:

- if they both have negative results, no event will be triggered;
- if both 2 detectors give positive results or only thermal image-based detector has a positive result, the confirmation will be triggered and the gimbal will be rotated;
- if only the RGB image-based detector predicts positive, a lower threshold will be used to segment the thermal image; At this time, if thermal image-based detector gives positive then the confirmation request will be triggered. Otherwise, this frame will be marked as negative.

The segmentation and classification algorithms will be introduced in detail in Section IV.

Once the confirmation is triggered, the drone(s) will interrupt the patrolling mission and hover. Then a gimbal controlling algorithm will rotate the camera facing the potential forest fire point according to the segmentation on the thermal images. The gimbal controlling method will also be presented in Section IV. Meanwhile, zoom RGB camera on H20T will also be set to “ $5\times$ ” in order to capture the potential forest fire point in detail, as shown in Fig.6:

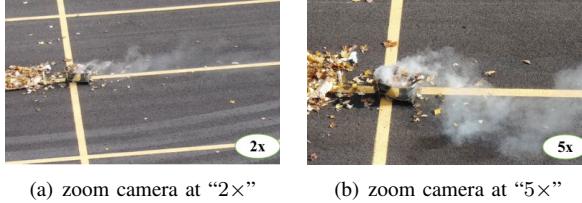


Fig. 6. DJI H20T zoom camera

After that, an Unet-based early forest fire flame and smoke segmentation algorithm running on the ground station will be triggered by drones automatically. Ground station computer, with more advanced hardware, has higher computing resource to run more complex neural networks. The RGB images captured by the H20T will be sent to the ground station by wireless image transmission. If the confirmation is established, the early forest fire region will be located and then the forest fire alarm and fire point geolocation with detection results will be sent to other termination such as smartphones, for the forest rangers or firefighters.

In summary, the proposed early forest fire detection system takes the hardware and sensors limitation, algorithms false alarm, and computation power distribution into consideration, achieving considerable robustness.

IV. VISION AND THERMAL IMAGES BASED EARLY FOREST FIRE DETECTION ALGORITHMS

A. Resnet18-based Early Forest Fire Classification

Resnet18 is used for the onboard early forest fire classification, through which the RGB images captured by the H20T camera are classified into normal, potential fire and potential smoke.

Resnet [35] series use the identity mapping to address the vanishing gradient and exploding gradient problem generated

by the layers becoming deep. The following Fig.7 shows the Resnet18 architecture: green block shows the basic component in the Resnet; layers among the residual block learn residual value instead of the original value, which makes the network more robust while training. Although Resnet has many variations, they can be separated into 3 parts: input, output, and convolutional module.

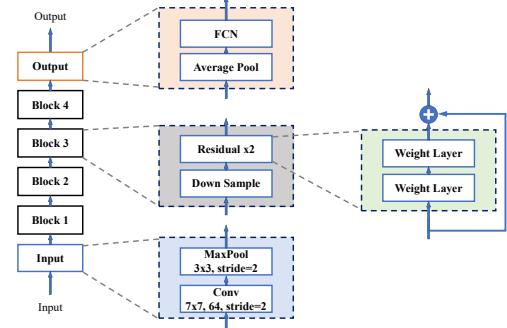


Fig. 7. Brief architecture of ResNet18-based forest fire classifier

Input module consists of 7×7 convolutional layer and a 3×3 max-pooling layer; the middle convolutional layer consists of 4 similar residual blocks. The features map channel accumulates from 64 to 512; finally, after an average pooling and fully connected layer, feature channel changes from 512 to 3, according to 3 possible classes of an image.

B. Unet-based Early Forest Fire Smoke Segmentation

Unet [36] structure is widely used in image segmentation. In the proposed system, Unet is designed to run on the ground station computer for its more complex architecture than Resnet18.

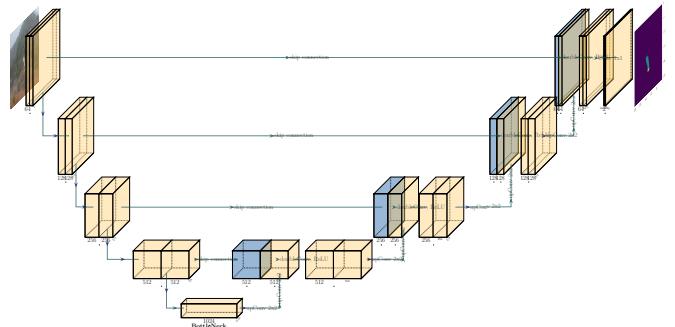


Fig. 8. UNet structure for forest fire segmentation

The network architecture is as shown in above Fig.8 , which can be described as consisting of a contraction path (left) and an expansion path(right). The contraction path is the same as the traditional convolutional network. It consists of an unpadded convolution with a convolution kernel size of 3×3 . After each convolution, the features pass through a ReLU function, and a maximum pool with a size of 2×2 , the stride of 2. This maximum pooling is the process of downsampling. After downsampling, the number of channels is doubled.

The expansion path is up-convolution with the size of 2×2 , the output channels of the upper convolution are half of the original, and then connected in series with the corresponding feature map (after cropping) to obtain channels of the same size as the original, and then pass through two sizes of 3×3 convolution and ReLU function. The corresponding cropping feature map is necessary because there will be a loss of boundary pixels in the process of our convolution. In the last layer, the desired target type is obtained through convolution with a convolution kernel size of 1×1 . In this network, there are 23 convolutional layers. The network adopts the common Encoder-Decoder structure, and adds to the original structure the operation of directly intercepting information from the encoder and placing it in the decoder. This operation can effectively retain the edge detail information in the original image and prevent excessive edges.

Compared with traditional fully convolutional networks, Unet can achieve relative acceptable precision from less training data based on the application of skip connection.

C. Threshold and Sliding Window-based Thermal Image Segmentation

1) *HSV color space*: The thermal images captured by the H20T represent the heat radiation at each pixel. Inspired by the segmentation method proposed in [13], the HSV color space and threshold are employed to segment the potential heat source; after that, morphological operations are used to clear the noise. Finally, the sliding window is used to find the highest potential heat from each patch.

Considering the difficulty of separating colors and brightness in RGB space, HSV color space uses the hue, saturation, and value to represent a color. Only the hue channel is used for describing the color, the other two channels describe the purity and brightness.

In the proposed system, we use the H20T “north pole isothermal” mode to generate the thermal images, as shown in Fig.9. Under this mode, with the temperature increasing, the color of this point will change from blue to red; the brightness of this point will also be higher. It is noteworthy thermal images generated directly by H20T are still in the format of RGB.

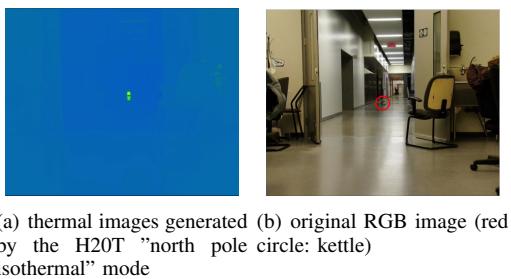


Fig. 9. thermal and RGB images captured by H20T

Using HSV color space makes it possible to find a threshold for three channels, especially for color channel. Conversion between RGB and HSV color space can be defined by Eq(1~4).

$$\begin{aligned} R' &= R/255 & C_{max} &= \max(R', G', B') \\ G' &= G/255 & C_{min} &= \min(R', G', B') \\ B' &= B/255 & \Delta &= C_{max} - C_{min} \end{aligned} \quad (1)$$

$$H = \begin{cases} 0 & \Delta = 0 \\ 60 \times \left(\frac{G' - B'}{\Delta} + 0 \right) & C_{max} = R' \\ 60 \times \left(\frac{B' - R'}{\Delta} + 2 \right) & C_{max} = G' \\ 60 \times \left(\frac{R' - G'}{\Delta} + 4 \right) & C_{max} = B' \end{cases} \quad (2)$$

$$S = \begin{cases} 0 & C_{max} = 0 \\ \frac{\Delta}{C_{max}} & C_{max} \neq 0 \end{cases} \quad (3)$$

$$V = C_{max} \quad (4)$$

Then a threshold is defined based on the experiment to segment the heat point and background:

$$P_{heat} = \begin{cases} 1 & T_l < H(x, y) < T_u \\ 0 & \text{otherwise} \end{cases} \quad (5)$$

In Eq(5), P is the rule to segment the heat point; $H(x, y)$ is the H value at (x, y) and T_l, T_u are the threshold lower and upper boundary.

After the threshold segmentation, there will be discrete pixels around the heat point. The morphological operation can be used as the geometry filter to reduce the noise. With the opening operation (erosion first, then dilation), most of the noise can be suppressed.

2) *Sliding Window*: It could be seen in Fig.10, the sliding window (in green) traversals every patch of the thermal image, and compare the heat pixels inside. The patch with the highest heat pixel number will be marked as the heat point position on the image. For example, patch 1 is chosen as the location of the heat point in the following Fig.10 :

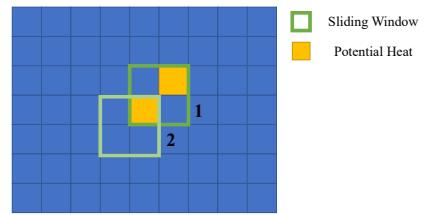


Fig. 10. sliding window generate the patches with heat point pixels

For the benefit of the real-time property, the size of the sliding window in our proposed system is fixed.

D. Gimbal Control and Fire Point Geolocation

1) *Gimbal Control*: Gimbal controller will rotate and zoom the H20T camera to focus on the potential early forest fire point with the location in the thermal image after confirmation request is triggered. Inspired by [37], a PD controller is designed for the rotating task, which is shown in the Fig.11: X_{error} and Y_{error} are defined as the pixel difference between the desired position and the current position of the fire point. Then controller is designed and tuned to generate the control command for the DJI gimbal controller.

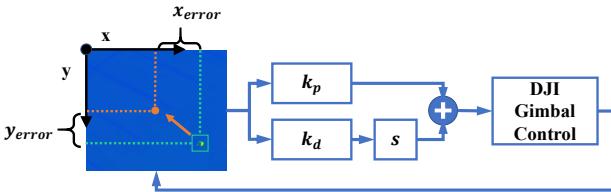


Fig. 11. gimbal control by tuning K_p and K_d of a PD controller

2) *Geolocation*: The geolocation of the early forest fire point can be estimated by the laser rangefinder and gimbal angle as show in Fig.12.

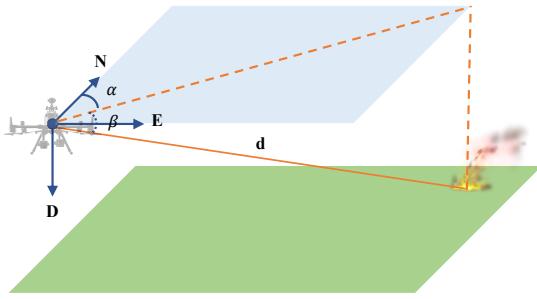


Fig. 12. geolocation of the early forest fire point

In the navigation coordinate system (NED), the relative location can be defined as:

$$\begin{cases} l_N = d \cos \beta \cos \alpha \\ l_E = d \cos \beta \sin \alpha \\ l_D = d \sin \beta \end{cases} \quad (6)$$

where the α and β are the yaw and pitch angle of the gimbal, which can be accessed by the DJI Onboard SDK. Once the local position is calculated, the global position of the fire point can be easily expressed with the drone's Global Positioning System (GPS) information:

$$\begin{cases} lon = lon_{M300} + \frac{l_N}{R_e} \\ lat = lat_{M300} + \frac{l_E}{R_e \cos(lon)} \\ alt = alt_{M300} - l_D \end{cases} \quad (7)$$

where the lon, lat, alt stands for longitude, latitude, and altitude of GPS location, R_e is the average radius of the earth.

V. EXPERIMENTS

A. Drone Flight Trajectory

In the outdoor experiment, a synthetic fire point, which generates heat and smoke through burning leaves and woods, is placed in front of the M300 drone. After taking off, the system performs according to our proposed strategy. After the early forest fire is detected and confirmed, the system gives alarm. Finally, M300 returns home and finishes the mission.

Fig.13 shows the trajectory of the M300 during the experiment.

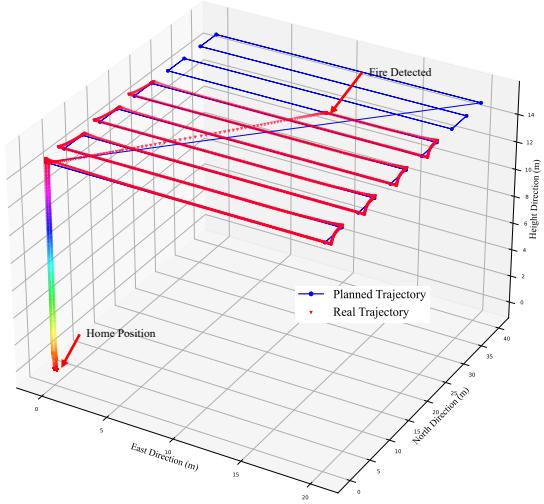


Fig. 13. M300 planned trajectory and real flight test trajectory

It should be noted that currently due to the insufficient APIs of DJI onboard SDK, the laser rangefinder can only be manipulated manually. But except for that, all the other processes can be done automatically by proposed system.

B. Gimbal Control

Through ground testing, the control frequency of DJI Gimbal controller as shown in Fig.11 is 1 ~ 2Hz. The main reason is that the gimbal control API in DJI Onboard SDK is based on the ROS "service" communication method, which has a longer time of delay. Thus the proposed PD controller has frequency of 1Hz.

In the controller software implementation, once the errors are less than the given thresholds, or it takes too long time to control the gimbal, the controller will stop. The following Fig.14 shows the control output and the pixel error along the yaw and pitch direction; After 9 times of control, , the pixel errors along x and y direction both converge into the given boundary (20 pixels in experiment).

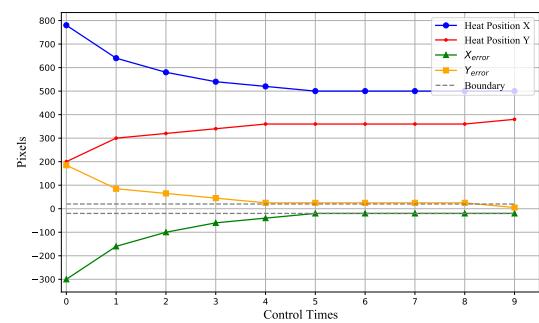


Fig. 14. heat point position and pixel errors: after errors converge into the given boundary, control commanding stops.

C. Thermal Images Segmentation and Localization

RGB color space use all 3 channels to express color features, which means using RGB color space may be more difficult

to segment heat point through rule-based methods than HSV color space which only use H channel to stands for color. This is because under the “north pole isothermal model”, H2OT infrared camera images change from blue to red with the temperature increasing. It results all the 3 channels in RGB color space will change during the temperature changing, but only the H channel for HSV color space. For example, when the temperature is relatively low, red channel can not be used to locate the heat point as shown in Fig.15:

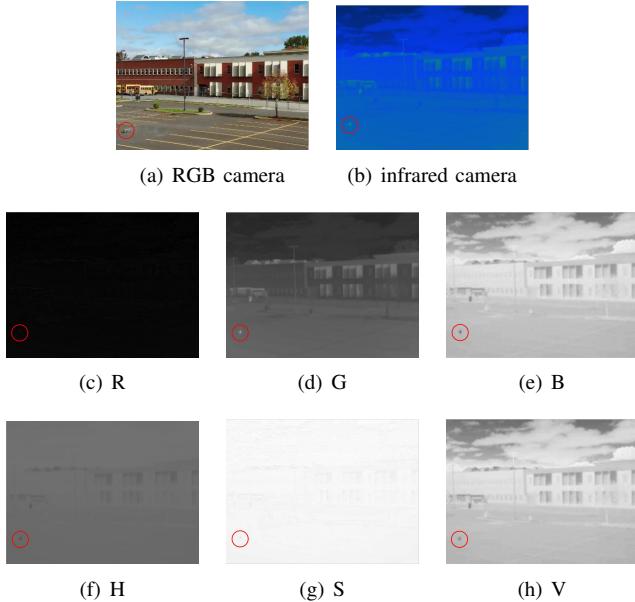


Fig. 15. HSV and RGB color space channels. The red circles indicate heat point. In this example, “R” channel fails to reflect the heat position.

Therefore, the H channel is used for the following processing as shown in Fig.16:

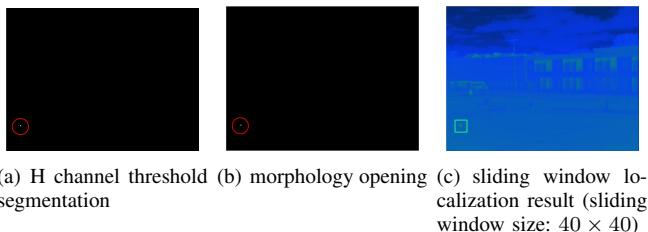


Fig. 16. heat point segmentation and localization process

The location in the thermal image is used to rotate the gimbal.

D. Resnet18-based Classification

The training and validation dataset is collected from Internet as well as synthetic experiments pictures.

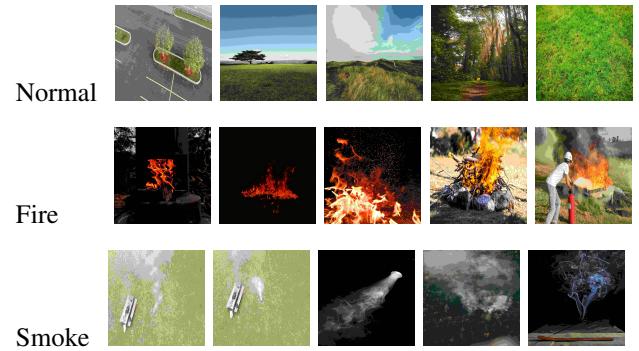


Fig. 17. Examples of three different classes dataset for training Resnet18-based forest fire classifier.

In the training process, the learning rate is set as to 1×10^{-5} . Dataset with 500 images are resized to 255×255 . 400 images are used for training. Both batch normalization and data augmentation including random flipping and rotating and are implemented . Other 100 Images of validation set are randomly picked from the entire 500 images dataset. After 20 epochs, the average validation accuracy of “fire”, “smoke” and “normal” achieves over 82% . The following Fig.18 shows training loss and accuracy during 50 epochs.

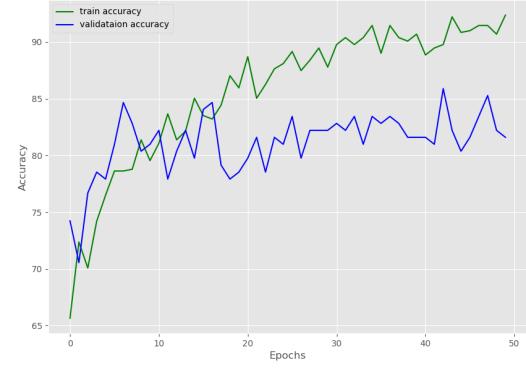


Fig. 18. Resnet18 training and validation accuracy during 50 epochs

In the outdoor experiment, parameters trained from 20 epochs is used. Some example classification results are shown in the following Fig.19.



Fig. 19. Resnet18 classification result examples

1) Unet-based Smoke Segmentation: Similar to the Resnet18, the training validation set for Unet are also collected from Internet as well as our synthetic data which is captured by the DJI Phantom 4 Pro. The mask region are generated by

Labelme [38] manually. Some examples images and masks of the training and validation set are shown in Fig.20.



Fig. 20. Examples of three different classes dataset for training Unet-based forest fire smoke segmentation.

The Unet is trained on ground station computer. During the training process, learning rate and batch size are set to 1×10^{-6} and 8 separately. In the preprocessing procedure, random image flipping, rotating and random brightness contrast adjustment are implemented before batch normalization and training. After 90 epochs, the accuracy achieves over 75% and changes slightly with any epoch increasing as Fig.21 shows:

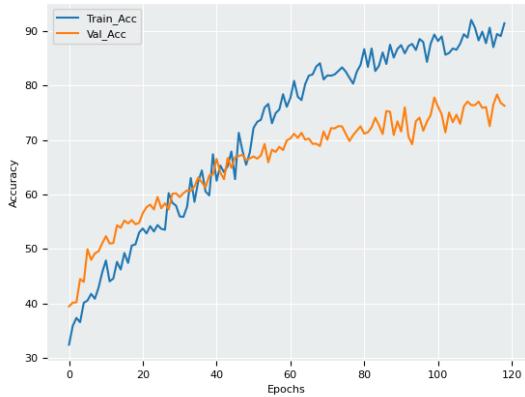


Fig. 21. Unet training loss and accuracy during 120 epochs

In the outdoor experiment, the frame rate of the online segmentation is set to 5 Hz. Some segmentation result video frames are shown in the Fig.22: The first row shows the original images with the mask (in translucent red); the second row shows the segmented smoke in white.

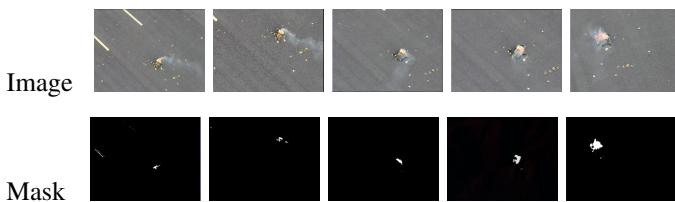


Fig. 22. Unet smoke segmentation result examples

Once the smoke is segmented successfully, forest fire alarm will be sent to different terminals.

VI. CONCLUSION

This paper presents an early forest fire detection system based on aerial sensor data. DJI M300 and H20T camera are used for the data collection. Then a two-phase early forest fire detection and confirmation strategy is proposed: Resnet18 and threshold-based thermal images segmentation algorithm are implemented to process the RGB and thermal images separately at the detection stage. Then an Unet-based segmentation neural network is developed to accomplish the confirmation. Fire point geolocation and gimbal control algorithms are also implemented to locate the confirmed early forest fire point. The proposed strategies for decision-making based on multiple aspects including algorithms, sensor data as well as hardware makes implementation more reliable and robust.

However, there are still several improvements that can be done in future work. Due to the incomprehensible of software implementation, laser rangefinder-based geolocation is not flexible enough to work on an onboard computer automatically. Resnet18 can only output classification results rather than the object location, which may not fully use the information provided by the RGB images even in the "rough" detection stage. The works for dealing with these challenges are on the way.

REFERENCES

- [1] S. Verma and S. Jayakumar, "Impact of forest fire on physical, chemical and biological properties of soil: A review," *proceedings of the International Academy of Ecology and Environmental Sciences*, vol. 2, no. 3, p. 168, 2012.
- [2] K. Sazawa, H. Yoshida, K. Okusu *et al.*, "Effects of forest fire on the properties of soil and humic substances extracted from forest soil in gunma, japan," *Environmental Science and Pollution Research*, vol. 25, no. 30, pp. 30 325–30 338, 2018.
- [3] S. Singh, H. Singh, V. Sharma *et al.*, "Impact of forest fires on air quality in wolgan valley, new south wales, australia—a mapping and monitoring study using google earth engine," *Forests*, vol. 13, no. 1, p. 4, 2022.
- [4] B. J. Stocks, M. Fosberg, T. Lynham *et al.*, "Climate change and forest fire potential in russian and canadian boreal forests," *Climatic change*, vol. 38, no. 1, pp. 1–13, 1998.
- [5] N. J. Abram, B. J. Henley, A. S. Gupta *et al.*, "Connections of climate change and variability to large and extreme forest fires in southeast australia," *Communications Earth & Environment*, vol. 2, no. 1, pp. 1–17, 2021.
- [6] Global Forest Watch, "Forest Monitoring, land use & deforestation trends," Dec. 22, 2021. [Online]. Available: <https://www.globalforestwatch.org/>
- [7] C.-Y. Lee, C.-T. Lin, C.-T. Hong *et al.*, "Smoke detection using spatial and temporal analyses," *International Journal of Innovative Computing, Information and Control*, vol. 8, no. 6, pp. 1–11, 2012.
- [8] W. Schroeder, P. Oliva, L. Giglio *et al.*, "The new VIIRS 375 m active fire detection data product: Algorithm description and initial assessment," *Remote Sensing of Environment*, vol. 143, pp. 85–96, 2014.
- [9] C. Yuan, Z. Liu, and Y. Zhang, "Aerial images-based forest fire detection for firefighting using optical remote sensing techniques and unmanned aerial vehicles," *Journal of Intelligent & Robotic Systems*, vol. 88, no. 2, pp. 635–654, 2017.
- [10] S. Abdullah, S. Bertalan, S. Masar *et al.*, "A wireless sensor network for early forest fire detection and monitoring as a decision factor in the context of a complex integrated emergency response system," in *2017 IEEE Workshop on Environmental, Energy, and Structural Monitoring Systems (EESMS)*. IEEE, 2017, pp. 1–5.

- [11] P. Dasari, G. K. J. Reddy, and A. Gudipalli, "Forest fire detection using wireless sensor networks," *International Journal on Smart Sensing & Intelligent Systems*, vol. 13, no. 1, 2020.
- [12] V. Sherstjuk, M. Zharikova, and I. Sokol, "Forest fire-fighting monitoring system based on UAV team and remote sensing," in *2018 IEEE 38th International Conference on Electronics and Nanotechnology (ELNANO)*. IEEE, 2018, pp. 663–668.
- [13] C. Yuan, Z. Liu, and Y. Zhang, "UAV-based forest fire detection and tracking using image processing techniques," in *2015 International Conference on Unmanned Aircraft Systems (ICUAS)*. IEEE, 2015, pp. 639–643.
- [14] C. Yuan, Z. Liu, and Y. Zhang, "Vision-based forest fire detection in aerial images for firefighting using UAVs," in *2016 International Conference on Unmanned Aircraft Systems (ICUAS)*. IEEE, 2016, pp. 1200–1205.
- [15] C. Yuan, Z. Liu, and Y. Zhang, "Learning-based smoke detection for unmanned aerial vehicles applied to forest fire surveillance," *Journal of Intelligent & Robotic Systems*, vol. 93, no. 1-2, pp. 337–349, 2019.
- [16] J. Vasconcelos Reinolds de Sousa and P. Vieira Gamboa, "Aerial forest fire detection and monitoring using a small UAV," *KnE Engineering*, vol. 5, no. 6, p. 242–256, Jun. 2020.
- [17] C. E. Prema, S. Vinsley, and S. Suresh, "Efficient flame detection based on static and dynamic texture analysis in forest fire detection," *Fire technology*, vol. 54, no. 1, pp. 255–288, 2018.
- [18] F. A. Hossain, Y. M. Zhang, and M. A. Tonima, "Forest fire flame and smoke detection from UAV-captured images using fire-specific color features and multi-color space local binary pattern," *Journal of Unmanned Vehicle Systems*, vol. 8, no. 4, pp. 285–309, 2020.
- [19] L. P. Osco, J. M. Junior, A. P. M. Ramos *et al.*, "A review on deep learning in UAV remote sensing," *arXiv preprint arXiv:2101.10861*, vol. abs/2101.10861, 2021.
- [20] C. Kyrou and T. Theocharides, "Deep-learning-based aerial image classification for emergency response applications using unmanned aerial vehicles," in *CVPR Workshops*, 2019, pp. 517–525.
- [21] P. Barmpoutis, K. Dimitropoulos, K. Kaza *et al.*, "Fire detection from images using faster R-CNN and multidimensional texture analysis," in *ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2019, pp. 8301–8305.
- [22] D. Kinaneva, G. Hristov, J. Raychev *et al.*, "Early forest fire detection using drones and artificial intelligence," in *2019 42nd International Convention on Information and Communication Technology, Electronics and Microelectronics (MIPRO)*. IEEE, 2019, pp. 1060–1065.
- [23] Z. Jiao, Y. Zhang, L. Mu *et al.*, "A YOLOv3-based learning strategy for real-time UAV-based forest fire detection," in *2020 Chinese Control And Decision Conference (CCDC)*. IEEE, 2020, pp. 4963–4967.
- [24] Z. Jiao, Y. Zhang, J. Xin *et al.*, "A deep learning based forest fire detection approach using UAV and YOLOv3," in *2019 1st International Conference on Industrial Artificial Intelligence (IAI)*. IEEE, 2019, pp. 1–5.
- [25] P. Barmpoutis, T. Stathaki, K. Dimitropoulos *et al.*, "Early fire detection based on aerial 360-degree sensors, deep convolution neural networks and exploitation of fire dynamic textures," *Remote Sensing*, vol. 12, no. 19, p. 3177, 2020.
- [26] Q. Zhang, J. Xu, L. Xu *et al.*, "Deep convolutional neural networks for forest fire detection," in *Proceedings of the 2016 international forum on management, education and information technology application*. Atlantis Press, 2016.
- [27] M. Sadi, Y. Zhang, W.-F. Xie *et al.*, "Forest fire detection and localization using thermal and visual cameras," in *2021 International Conference on Unmanned Aircraft Systems (ICUAS)*. IEEE, 2021, pp. 744–749.
- [28] M. Dai, P. Gao, M. Sha *et al.*, "Smoke detection in infrared images based on superpixel segmentation," in *MIPPR 2019: Remote Sensing Image Processing, Geographic Information Systems, and Other Applications*, vol. 11432. International Society for Optics and Photonics, 2020, p. 1143213.
- [29] A. Moffatt, N. Turcios, C. Edwards *et al.*, "Collaboration between multiple UAVs for fire detection and suppression," in *2021 International Conference on Unmanned Aircraft Systems (ICUAS)*. IEEE, 2021, pp. 730–737.
- [30] C. Yuan, Z. Liu, and Y. Zhang, "Fire detection using infrared images for UAV-based forest fire surveillance," in *2017 International Conference on Unmanned Aircraft Systems (ICUAS)*. IEEE, 2017, pp. 567–572.
- [31] ROS, "ROS: Robot Operating System," Dec. 22, 2021. [Online]. Available: <https://www.ros.org/>
- [32] DJI, "Onboard SDK: Expand the capabilities of your aerial platform with onboard compute," Dec. 22, 2021. [Online]. Available: <https://developer.dji.com/onboard-sdk/>
- [33] N. Ayoub and P. Schneider-Kamp, "Real-time on-board deep learning fault detection for autonomous UAV inspections," *Electronics*, vol. 10, no. 9, p. 1091, 2021.
- [34] Nvidia-AI-Iot, "Nvidia-ai-IOT/torch2trt: An easy to use pytorch to tensortr converter," Dec. 22, 2021. [Online]. Available: <https://github.com/NVIDIA-AI-IOT/torch2trt>
- [35] K. He, X. Zhang, S. Ren *et al.*, "Deep residual learning for image recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 770–778.
- [36] O. Ronneberger, P. Fischer, and T. Brox, "U-net: Convolutional networks for biomedical image segmentation," in *International Conference on Medical image computing and computer-assisted intervention*. Springer, 2015, pp. 234–241.
- [37] D. Regner, J. Salazar, P. Buschinelli *et al.*, "Object tracking control using a gimbal mechanism," *The International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences*, vol. 43, pp. 189–196, 2021.
- [38] Wkentaro, "wkentaro/labelme: Image polygonal annotation with python (polygon, rectangle, circle, line, point and image-level flag annotation)." Dec. 22, 2021. [Online]. Available: <https://github.com/wkentaro/labelme>