



Data article

Aerial imagery pile burn detection using deep learning: The FLAME dataset

Alireza Shamsoshoara ^{a,*}, Fatemeh Afghah ^a, Abolfazl Razi ^a, Liming Zheng ^a, Peter Z. Fulé ^b, Erik Blasch ^c^a School of Informatics, Computing, and Cyber Systems, Northern Arizona University, Flagstaff, AZ, United States of America^b School of Forestry, Northern Arizona University, Flagstaff, AZ, United States of America^c Air Force Research Laboratory, Rome, NY, United States of America

ARTICLE INFO

ABSTRACT

Keywords:

Aerial imaging
Fire monitoring dataset
Fire detection and segmentation
Deep learning

Wildfires are one of the costliest and deadliest natural disasters in the US, causing damage to millions of hectares of forest resources and threatening the lives of people and animals. Of particular importance are risks to firefighters and operational forces, which highlights the need for leveraging technology to minimize danger to people and property. *FLAME* (Fire Luminosity Airborne-based Machine learning Evaluation) offers a dataset of aerial images of fires along with methods for fire detection and segmentation which can help firefighters and researchers to develop optimal fire management strategies.

This paper provides a fire image dataset collected by drones during a prescribed burning piled detritus in an Arizona pine forest. The dataset includes video recordings and thermal heatmaps captured by infrared cameras. The captured videos and images are annotated, and labeled frame-wise to help researchers easily apply their fire detection and modeling algorithms. The paper also highlights solutions to two machine learning problems: (1) Binary classification of video frames based on the presence [and absence] of fire flames. An Artificial Neural Network (ANN) method is developed that achieved a 76% classification accuracy. (2) Fire detection using segmentation methods to precisely determine fire borders. A deep learning method is designed based on the U-Net up-sampling and down-sampling approach to extract a fire mask from the video frames. Our FLAME method approached a precision of 92%, and recall of 84%. Future research will expand the technique for free burning broadcast fire using thermal images.

Specifications table

Subject	Artificial Intelligence
Specific subject area	An aerial imagery dataset for pile fire detection based on image classification and fire segmentation using deep learning
Type of data	Table Video Image Figure
How data were acquired	Full HD and 4K camera: Zenmuse X4S camera, Phantom 3 Camera Thermal Camera: FLIR Vue Pro R Drones: DJI Matrice 200, DJI Phantom 3 Professional
Data format	Raw: .MP4, .MOV Analysed: .JPEG, .PNG For more information please visit these two repositories: https://ieee-dataport.org/open-access/flame-dataset-aerial-imagery-pile-burn-detection-using-drones-uavs https://github.com/AlirezaShamsoshoara/Fire-Detection-UAV-Aerial-Image-Classification-Segmentation-UnmannedAerialVehicle

(continued on next page)

* Corresponding author.

E-mail address: Alireza_Shamsoshoara@nau.edu (A. Shamsoshoara).

(continued)

Subject	Artificial Intelligence
Parameters for data collection	The resolution for the FLIR camera is 640 x 512. The resolution for the Zenmuse camera is 1280 x 720. The resolution for the Phantom camera is 3480 x 2160. The resolution for the Fire_NoFire classification is 254 x 254.
Description of data collection	Data was collected using aerial drones (UAVs). The test was conducted with fire managers from the Flagstaff (Arizona) Fire Department who carried out a burn of piled slash on city-owned lands in a ponderosa pine forest on Observatory Mesa. The prescribed fire took place on January 16th, 2020 with the temperature of 43F (6C) and partly cloudy conditions and no wind.
Data source location	Flagstaff, Arizona, USA, Ponderosa pine forest on Observatory Mesa
Data accessibility	Repository name: THE FLAME DATASET: AERIAL IMAGERY PILE BURN DETECTION USING DRONES (UAVS) Data identification number: 10.21227/qad6-r683 Direct URL to data: https://ieeexplore.ieee.org/abstract/document/9270000 Direct URL to GitHub repository: https://github.com/AlirezaShamsoshoara/Fire-Detection-UAV-Aerial-Image-Classification-Segmentation-UnmannedAerialVehicle
Related research article	Shamsoshoara, A., Khaledi, M., Afghah, F., Razi, A. and Ashdown, J., Distributed cooperative spectrum sharing in uav networks using multi-agent reinforcement learning. <i>IEEE Annual Consumer Communications & Networking Conference (CCNC)</i> , IEEE. 10.1109/CCNC50000.2019.8651796 Shamsoshoara, A., Khaledi, M., Afghah, F., Razi, A., Ashdown, J. and Turck, K., A solution for dynamic spectrum management in mission-critical UAV networks, <i>16th annual IEEE international conference on sensing, communication, and networking (SECON)</i> , IEEE. 10.1109/SAHCN50000.2019.8824917 Shamsoshoara, A., Afghah, F., Razi, A., Mousavi, S., Ashdown, J. and Turk, K., An autonomous spectrum management scheme for unmanned aerial vehicle networks in disaster relief operations. <i>IEEE Access</i> , 10.1109/ACCESS.2020.2982932
Related project(s)	Most of the related projects are mentioned in the website below: https://www.cefns.nau.edu/~fa334/

Value of the data

- Based on the reports from National Interagency Fire Center (NIFC) in the USA, total number of 51,296 fires burned more than 6,359,641 acres of lands yearly on average from 2010 to 2019 accounting for more than \$6 billion in damages.
- This dataset motivates researchers to seek novel solutions for early fire detection and management. In particular, recent advances in aerial monitoring systems can provide first responders and operational forces with more accurate data on fire behaviour for enhanced fire management.
- Recent advances in artificial intelligence (AI) and machine learning have made image-based modeling and analysis (e.g., classification, real time prediction, and image segmentation) even more successful in different applications.

1. Introduction: Scope, significance, and problem definition

Wildfires have caused severe damage to forests, wildlife habitats, farms, residential areas, and ecosystems during the past few years. Based on the reports from National Interagency Fire Center (NIFC) in the USA, total number of 51,296 fires burned more than 6,359,641 acres of lands yearly on average from 2010 to 2019 accounting for more than \$6 billion in damages [1,2]. These alarming facts motivate researchers to seek novel solutions for early fire detection and management. In particular, recent advances in aerial monitoring systems can provide first responders and operational forces with more accurate data on fire behavior for enhanced fire management.

Traditional approaches to detecting and monitoring fires include stationing personnel in lookout towers or using helicopters or fixed-wing aircraft to surveil fires with visual and infrared imaging. Recent research has suggested Internet of Things (IoT) innovations based on wireless sensor networks [3–5], but such networks would require further investment and testing before providing practical information. At broader scales, satellite imagery is widely used for assessing fires globally [6,7], but typically at relatively coarse resolution and with the availability of repeat images constrained by satellite orbital patterns.

Considering the challenges and issues of these methods, using Unmanned Aerial Vehicles (UAVs) for fire monitoring is gaining more traction in recent years. UAVs offer new features and convenience including fast deployment, high maneuverability, wider and adjustable viewpoints, and less human intervention [8–11]. Recent studies investigated the use of UAVs in disaster relief scenarios and operations such as wildfires and floods, particularly as a temporary solution when terrestrial networks fail due to damaged infrastructures, communication problems, or spectrum scarcity [12–14].

Recent advances in artificial intelligence (AI) and machine learning have made image-based modeling and analysis (e.g., classification, real time prediction, and image segmentation) even more successful in different applications. Also, with the advent of nanotechnology semiconductors, a new generation of Tensor Processing Units (TPUs) and Graphical Processing Units (GPUs) can provide an extraordinary computation capability for data-driven methods [15]. Moreover, modern drones and UAVs can be equipped with tiny edge TPU/GPU platforms to perform on-board processing on the fly to facilitate early fire detection before a catastrophic event happens [16,17].

Most supervised learning methods rely on large training datasets to train a reasonably accurate model. Studies such as [18] used a fire dataset from public sources to perform fire detection based on pre-trained ANN architectures such as MobileNet and AlexNet. However, that dataset was based on terrestrial images of the fire. To the best of our knowledge, there exists no aerial imaging dataset for fire analysis, something in urgent need to develop fire modeling and analysis tools for aerial monitoring systems. Note that aerial imagery exhibits different properties such as low resolutions, and top-view perspective, substantially different than images taken by ground cameras.

In this paper, we introduce a new dataset as a collection of fire videos and images taken by drones during a prescribed burning slash piles in Northern Arizona. The images were taken by multiple drones with different points of view, different zoom, and camera types including regular and thermal cameras. Pile burns can be very helpful to study spot fires and early-stage fires. Pile burns are typically used by forest management for cleaning up forest residues (“slash”) such as branches and foliage from forest thinning and restoration projects. Forest treatments are a key management strategy for reducing fuels and the burning of slash piles is often the most economically efficient and safe means of removing slash. Piles must be monitored by fire managers for a few days after ignition to avoid spread outside the intended burn area. Using automated aerial monitoring systems can substantially reduce the forest management workload. Moreover, recently Google proposed Federated Deep Learning (FDL) as decentralized and distributed learning approaches for wireless devices that have limited resources such as battery and wireless communication capability [19]. In FDL, wireless devices such as UAVs utilize their local information and data to train their own model and then they share the model parameters such as weights with the federated aggregation center. Later, the FDL aggregation center receives all model parameters from the nodes (e.g., drones) and starts merging them and distributes a unique model to all UAVs again to converge faster. FDL methods make it possible to keep the local data private and secure; also it is suitable for low bandwidth and wireless applications with limited amounts of batteries. Also, it only shares models’ parameters instead of sharing all images and videos with the FDL center which results in a low latency outcome. Current FLAME dataset is collected using different drones. Hence, it is possible to consider the FDL applications in this dataset as well for the future challenges and open problems [20].

We propose two sample problems to evaluate the use of dataset for real-world fire management problems. The contributions of this paper include i) proposing the first of its kind aerial imaging dataset for pile burn monitoring which includes both normal and thermal palettes as well as FLAME (Fire Luminosity Airborne-based Machine learning Evaluation), ii) a DL-based algorithm for frame-based fire classification which can be used for early fire detection, and (iii) a DL-based image segmentation method for pixel-wise fire masking for fire expansion modeling. The rest of the paper is structured as follows. Section 2 presents the FLAME dataset along with the related information regarding the hardware and data. Section 3 discusses the methodology based on the two defined challenges, namely fire classification and fire segmentation. The experiments and results are illustrated in Section 4 over a variety of metrics. Conclusions and discussion points are provided in Section 6.

2. FLAME dataset: Hardware and applicable data

This section details the hardware used to collect information, the data modalities, and types of the captured information.

Prescribed burning of slash piles is a common occurrence primarily during the winter months in high-elevation forests of the Southwest. Prescribed fires provide excellent opportunities for researchers to collect and update imagery data. The current study shows the results of the first test, and from which is available to continually update the dataset by adding more test results. The test was conducted with fire

managers from the Flagstaff (Arizona) Fire Department who carried out a burn of piled slash on city-owned lands in a ponderosa pine forest on Observatory Mesa. The prescribed fire took place on January 16th, 2020 with the temperature of 43°F (~ 6°C) and partly cloudy conditions and no wind.

2.1. Hardware

This study utilizes different drones and cameras to create a dataset of fire aerial images. Table 1 describes the technical specification of the utilized drones and cameras.

2.2. Applicable data for the defined problem

This section presents the details of the captured images, videos. The captured videos are converted to frames based on the recorded Frames Per Second (FPS). Four types of video including the normal spectrum, fusion, white-hot, and green-hot palettes are available in the FLAME dataset [25].

The normal spectrum palette was recorded using both Zenmuse X4S and the phantom 3 camera. Other thermal and IR outputs were collected using the Forward Looking Infrared (FLIR) vue Pro R camera. Several video clips which include both fire and no fire footage are available. The FLIR camera has a 1280 × 720 resolution with 29 frame per seconds (FPS). Another 6 min of video is available for one pile burn from the start of the burning at 1280 × 720 resolution and 29 FPS. The H.264 codec was used for all the recordings. More details about these videos are available in Table 2 along with the dataset link. Fig. 1 demonstrates some representative frames from both fire and no-fire videos. The full videos are available in the FLAME dataset repository.

The FLAME dataset also includes thermal videos such as Fusion, WhiteHot, and GreenHot palettes. All videos were captured with the resolution of 640 × 512 and with 30 FPS. Multiple videos of fire and no-fire types with different lengths are available. Fig. 2 shows some randomly selected frames for these thermal videos. More details about the FLAME dataset are available in Table 2. Also, a sample video of this dataset is available on YouTube [26]. Sections 3.1 and 3.2 demonstrate some of the videos conversions into frames to address research challenges such as fire classification and fire segmentation. Researchers can use applications of their choice to extract the frames from the videos based on the required FPS. The FLAME dataset including all images, videos, and data are available on IEEE-Dataport [25].

3. Goals: Suggested experiments and methodology

This section presents two example applications that can be defined based on the collected FLAME dataset along with Deep Learning solutions for these problems. The first problem is the fire versus no-fire classification using a deep neural network (DNN) approach. The second problem deals with fire segmentation, which can be used for fire detection by masking the identified fire regions on video frames classified as fire-containing in the first problem.

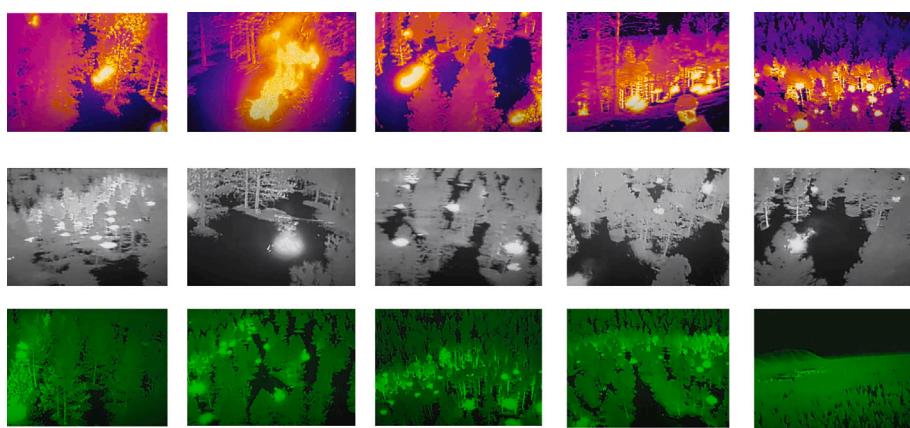
3.1. Fire vs no-fire classification

The image classification problem is one of the challenging tasks in the image processing domain. In the past, traditional image processing techniques utilized RGB channel comparison to detect different objects such as fire in frames or videos [27–29]. These traditional methods are not free of errors and are not fully reliable [30]. For instance, RGB value comparison methods that usually consider a threshold value to detect fire may detect sunset and sunrise as a false positive outcome. However, training a DNN to perform this image classification task helps to learn elements not germane to the fire. Also, some studies such as [31,32] perform pixel-based classification and segmentation based on the HSV (Hue, Saturation, Value) format. In the present study, a supervised

Table 1

Technical specification of hardware and tools.

	Phantom 3 Professional, DJI, 1280 gram, diagonal size=350 mm, Max speed=16 m/s(~57 kph), max flight time is 23 min, Flight time is reduced to 18 mins due to additional weight [21].
	Matrice 200, DJI, 3.80 kg, size:71 6 mm × 220 mm × 236 mm, payload up to 2 kg, 16 m/s (~61 kph), batteries: (TB50) and TB55 Max flight time: 38 min, operation range of 7 km [22].
	Zenmuse X4S, DJI, gimbal: Matrice 200, weight: 253 gram, Field Of View (FOV): 84°, resolution: Full HD to Cinematic 4K sensor: CMOS 20MPixels [23].
	Vue Pro R, FLIR, IR camera, control: Bluetooth and Pulse Width Modulation (PWM) signal, FOV: 45°, resolution: 640 × 512 Lens: 6.8 mm thermal, no gimbal [24].
	Phantom 3 camera, DJI, sensor: 1/2.3" CMOS 12.4MPixels FOV: 94°, (FPSs): 24 to 60, resolution: HD, FHD, UHD [21].

**Fig. 1.** Frame samples of the normal spectrum palette.**Fig. 2.** Frame samples of thermal images including Fusion, WhiteHot, and GreenHot palettes from top row to the bottom row.

machine learning method is used to classify the captured frames from camera. For mixed images when fire and non-fire parts coexist, the frame will be considered as the fire-labeled frame and when there is no fire in the frame, it will be considered as non-fire-labeled. Instead of the green or fusion heat map, the normal range spectrum of images for the classification was selected using the Zenmuse X4S and the camera from DJI Phantom 3. The binary classification model which was used in

this study is the Xception network [33] proposed by Google-Keras¹. The Xception model is a deep Convolutional Neural Network (DCNN). The structure of the DCNN is shown in **Fig. 3**. Replacing the standard *Inception* modules of the Inception architecture with depth-wise separable convolutions resulted in the Xception network [33–35].

¹ https://keras.io/examples/vision/image_classification_from_scratch/

Table 2

Dataset information ([Link](#)) [25].

Type	Camera	Palette	Duration	Resolution	FPS	Size	Application	Usage	Labeled
1	Video	Zenmuse	Normal(.MP4)	966 s	1280 × 720	29	1.2 GB	Classification	–
2	Video	Zenmuse	Normal(.MP4)	399 s	1280 × 720	29	503 MB	–	–
3	Video	FLIR	WhiteHot(.MOV)	89 s	640 × 512	30	45 MB	–	–
4	Video	FLIR	GreenHot(.MOV)	305 s	640 × 512	30	153 MB	–	–
5	Video	FLIR	Fusion(.MOV)	25 mins	640 × 512	30	2.83 GB	–	–
6	Video	Phantom	Normal(.MOV)	17 mins	3840 × 2160	30	32 GB	–	–
7	Frame	Zenmuse	Normal(.JPEG)	39,375 frames	254 × 254	–	1.3 GB	Classification	Train/Val
8	Frame	Phantom	Normal(.JPEG)	8617 frames	254 × 254	–	301 MB	Classification	Test
9	Frame	Phantom	Normal(.JPEG)	2003 frames	3480 × 2160	–	5.3 GB	Segmentation	Train/Val/Test
10	Mask	–	Binary(.PNG)	2003 frames	3480 × 2160	–	23.4 MB	Segmentation	Train/Val/Test

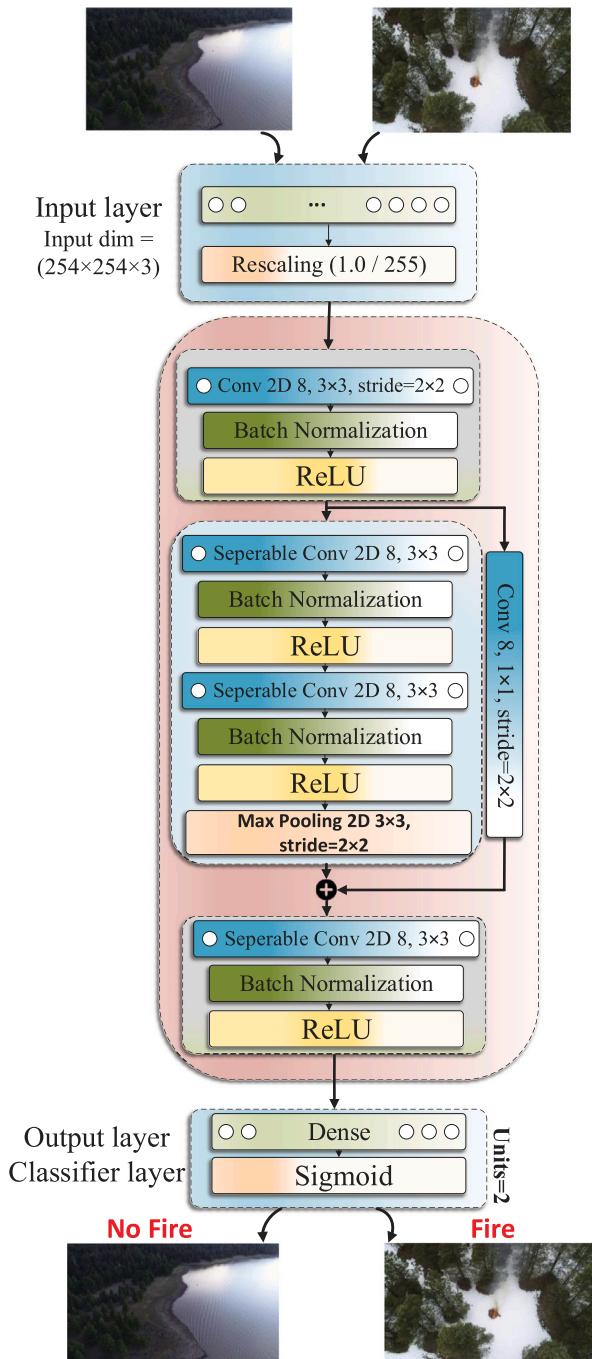


Fig. 3. Small version of the Xception network for the fire classification.

Fig. 3 is the concise version of the Xception model. The Xception model has three main blocks: (1) the input layer, (2) the hidden layers, and (3) the output layer. The size of the input layer depends on the image size and the number of channels which in our case is $(254 \times 254 \times 3)$. Then the value of RGBs in different channels are scaled to a float number between 0 and 1.0. The hidden layers rely on depth-wise separable convolutions and shortcut between the convolution blocks (ResNet [36]). The entry flow of the hidden layers is a pair of 2-Dimensional (2D) convolutional blocks with a size of 8 and a stride of 2×2 . Each block follows a batch normalization and a Rectified Linear Unit (ReLU) activation function [37]. The batch normalization is used to speed up the training process and bring more randomness by decreasing the importance of initial weights and regularize the model. Next, the model follows two separable 2D convolutional blocks. The last block of the hidden layer is a separable 2D convolutional layer with a size of 8 followed by another batch normalization and the ReLU function. Since the fire-detection is a binary classification task (Fire/No Fire), the activation function for the output layer is a Sigmoid function. The equation for the Sigmoid function is shown in (1),

$$P(\text{label}=\text{Fire}) = \sigma(\text{label}=\text{Fire}|\zeta(\theta)) = \frac{1}{1 + e^{-\zeta(\theta)}}, \quad (1)$$

where $\zeta(\theta)$ is the value of the output layer which is extracted based on the input frames and the RGB values of each pixel and all the weights across the hidden network. θ is the weight for the last layer of the network. The output value of the Sigmoid function is the probability of fire detection based on the imported frames into the network. To train the Xception network and find the weights of all neurons, a value loss function is targeted to increase the accuracy of the networks and find the optimal values for the weights. As the problem in this section is a binary classification, the considered loss function is a binary cross-entropy defined as

$$\mathcal{L}(y, \hat{y}) = - \frac{1}{N} \sum_{i=1}^N y_i * \log(p(\hat{y}_i)) + (1 - y_i) * \log(1 - p(\hat{y}_i)), \quad (2)$$

where N is the number of total samples in each batch used to update the loss function for each epoch. y is the ground truth label for the frames of types fire ($y = 1$) and no/fire ($y = 0$) based on the training data. $p(\hat{y})$ is the predicted probability of a frame belonging to the fire class. Next, the Adam optimizer is used to minimize the loss function and find the optimal weights during the learning process. After training the network with the training dataset, the evaluation is performed using a test dataset in Section 4. The implemented code for this learning model is available on GitHub [38]. To work with the code of this section, items 7 and 8 from Table 2 are used for the training and test sets. A detailed explanation about the repository is given on GitHub. The code used in this section works with Tensorflow 2.3.0 and Keras 2.4.0. To train the model, the user can change the Mode of the program to “Training” in the config.py file. The user can change all variables and parameters such as batch size and number of epochs in the config.py file. After the training phase, the user can change the Mode to “Classification” to evaluate the performance on the test set.

3.2. Fire segmentation

This section considers the problem of image segmentation for frames labeled as “fire” by the fire classification algorithm presented in Section 3.1. Studying the fire segmentation problem is useful for scenarios like detecting small fires [39]. Also, fire segmentation helps fire managers localize different discrete places of active burning for the purpose of fire monitoring. The goal is to propose an algorithm to find the pile burn segments in each frame and generate relevant masks. These segmentation problems were handled differently in the past using image processing and RGB threshold values to segment different data batches which exhibits relatively high error rates [30,40,41]. The goal is to develop an image semantic segmentation to perform a pixel-wise classification for each frame at the pixel level to define a fire mask for the generated output. To accomplish this task, a DCNN model is implemented to predict the label of each pixel based on the imported data. This segmentation problem can be recast as a binary pixel-wise classification problem, where each pixel can take two labels: “fire” and “non-fire” (background). To accomplish the image segmentation task, the fire test dataset from Section 3.1 is considered as a training dataset. To train a DCNN model, a Ground Truth Mask dataset is required. Different tools and applications such as Labelbox [42], Django Labeller [43], LabelImg [44], MATLAB Image Labeler [45], GNU Image Manipulation Program (GIMP) [46], etc are available to perform different types of the manual image segmentation such as pixel labeling, annotation (rectangles, lines, and cuboid) on the Regions Of Interest (ROI) to provide training data for the utilized deep learning model. The MATLAB (TM) Image Labeler is used on 2003 frames to generate the Ground Truth Masks. This subcategory of the FLAME dataset of masks and images is presented in Table 2. The implemented image segmentation model is adopted from the U-Net convolutional network developed for biomedical image segmentation [47]. U-Net is an end-to-end technique between the raw images and the segmented masks. A few changes are made to this network to accommodate the FLAME dataset and adapt it to the nature of this problem. The ReLU activation function is changed to Exponential Linear Unit (ELU) of each two-dimensional convolutional layer to obtain more accurate results [48]. The ELU function has a negative outcome smaller than a constant α for the negative input values and it exhibits a smoother behavior than the ReLU function. The structure of the customized U-Net is shown in Fig. 4. The backbone of the U-Net consists of a sequence of up-convolutions and concatenation with high-resolution features from the contracting path.

The size of the input layer is $512 \times 512 \times 3$ designed to match the size of the inputs images and three RGB channels. For computational convenience, the RGB values (between 0 and 255) are scaled down by 255 to yield float values between 0 and 1. Next, it follows the first contracting block including a two-dimensional fully convolutional layers with the ELU activation function, a dropout layer, another same fully convolutional layer, and a two-dimensional max pooling layer. This structure is repeated another three times to shape the left side of the U shape. Next, there are two two-dimensional fully connected layers with a dropout layer in between, the same structure of the left side is repeated for the right side of the U shape to have a symmetric structure for the up-convolution path in each block. Also, there exists a concatenation between the current block and the peer block from the contracting path. Since the pixel-wise segmentation is a binary classification problem, the last layer has the Sigmoid activation function.

The DCNN utilizes a dropout method to avoid the overfitting issue in the FLAME dataset analysis and realize a more efficient regularization noting the small number of ground truth data samples. The utilized loss function is the binary cross entropy similar to (2). The Adam optimizer is used to find the optimal value of weights for the neurons. The evaluation of the FLAME-trained model with the ground truth data is described in Section 4.2. The implemented code for this section is

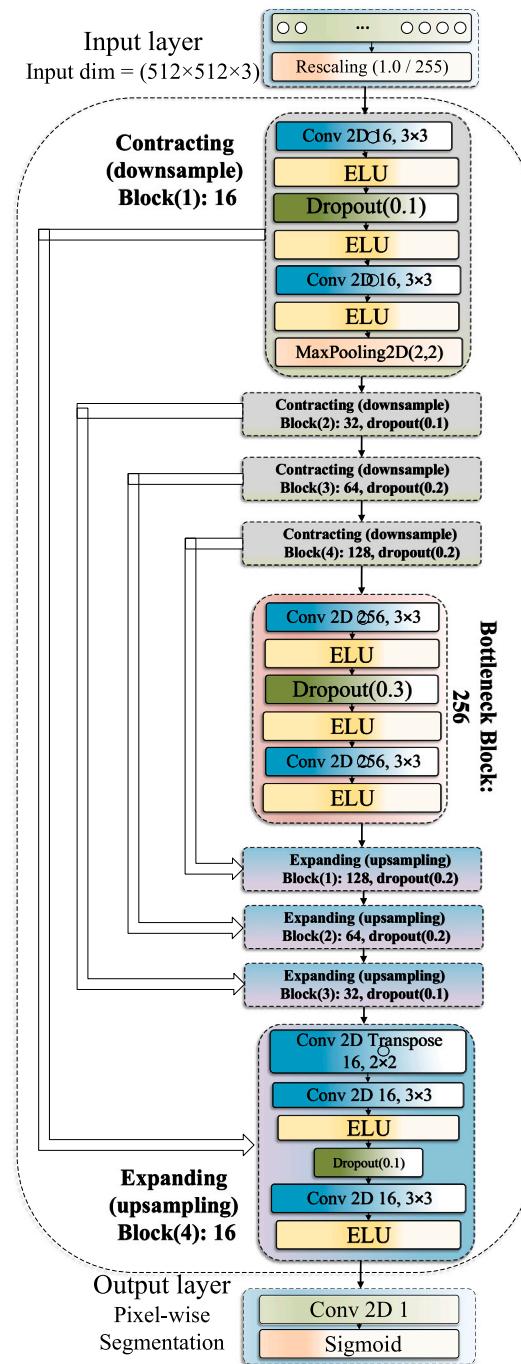


Fig. 4. Customized version of the U-Net for the fire segmentation.

available on GitHub [38]. The detailed explanation for the repository and the code is available on GitHub. This section uses items 9 and 10 from Table 2 to access the fire images and their ground truth data masks. The user can change the Mode to “Segmentation” in the config.py file to run the fire segmentation code. The user can change all variables such as batch size, number of epochs, number of classes and channels, and training and test sets ratio in the config.py file.

4. Results: Metrics and guidance on reporting results

In this section, we present the results of the two different problems of fire classification and fire segmentation. First, we provide the details of the parameters used in our experiments. Next, we discuss the results

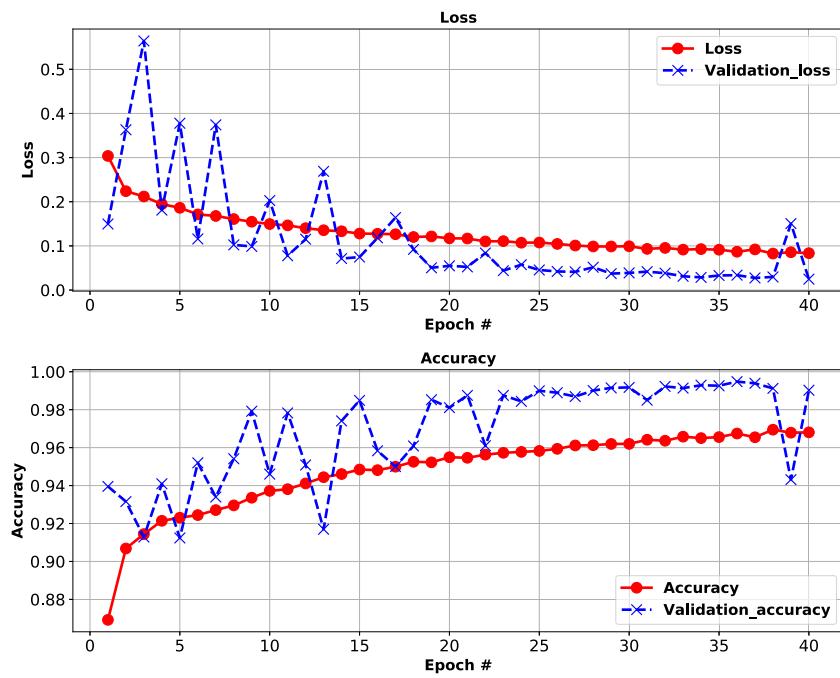


Fig. 5. Accuracy and loss values for the training and validation sets.

of each algorithm. All simulations for the training, validation, and testing phases, are performed using a AMD Ryzen 9 3900X with NVidia RTX 2080 Ti on an Ubuntu system.

4.1. Fire vs no-fire classification

In the training section, the total number of the frames is 39,375 which includes 25,018 frames of type “fire” and 14,357 frames of type “non-fire”. The training dataset is further split to 80% training and 20% validation sets. All frames are shuffled before feeding into the network. Also, augmentation methods such as horizontal flipping and random rotation are used to create new frames and address the issue of bias for unbalanced number of samples in the two “fire” and “non-fire” classes. The training phase ran over 40 epochs and the learning rate for the Adam optimizer is set to 0.001 which remains fixed during the training phase. Also, the batch size of 32 is used to fit the model in the training phase. To evaluate the accuracy and loss on the test dataset, 8617 frames including 5137 fire-labeled frames and 3480 No-fire-labeled frames are fed into the pre-trained networks. Table 3 reports loss and accuracy on training, validation, and test sets. It is noteworthy to mention that all frames for the training phase are collected using the Matrice 200 drone using Zemuse X4S camera and all frames for the test set are collected using the Phantom drone and its default mounted camera. Therefore, no overlap exists between the training and test samples. This fact confirms that our method is not biased to the imaging equipment properties, and the actual accuracy would be even higher when using the same imaging conditions for the training and test phase. The achieved accuracy of the “Fire vs No-Fire” classification is 76.23%.

Fig. 5 demonstrates the loss and accuracy for the training phase for both the training and validation sets. Also, Fig. 6 presents the confusion matrix for this binary fire classification task for all predictions. The vertical axis shows the true label of frames and the horizontal axis expresses the predicted label. The confusion matrix considers two classes which is plotted for the test dataset. Since, the ratio of fire and No-fire frames was imbalanced at the training phase, the rate of the false positive (classifying a true no-fire as fire) is higher than the false negative rate (classifying a true fire as no-fire).

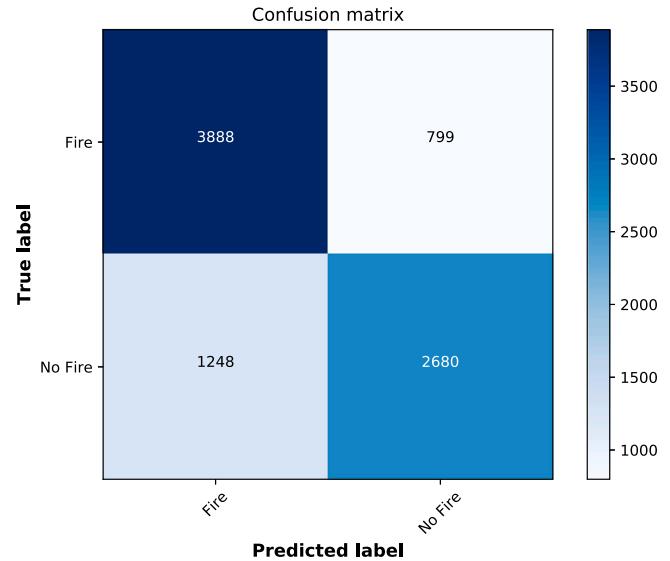


Fig. 6. Confusion matrix for the true and predicted labels.

Table 3

Accuracy and loss for evaluation of the fire classification.

Dataset	Performance	
	Loss	Accuracy(%)
Test set	0.7414	76.23
Validation set	0.1506	94.31
Training set	0.0857	96.79

4.2. Fire segmentation

The purpose of fire segmentation is to accurately localize and extract the fire regions from the background. Therefore, the video frames within the test set which are labeled as “fire” by the fire classification stage in section (Section 3.1) are used here for training, validation, and test. The total number of frames is 5137 and 2003 masks generated

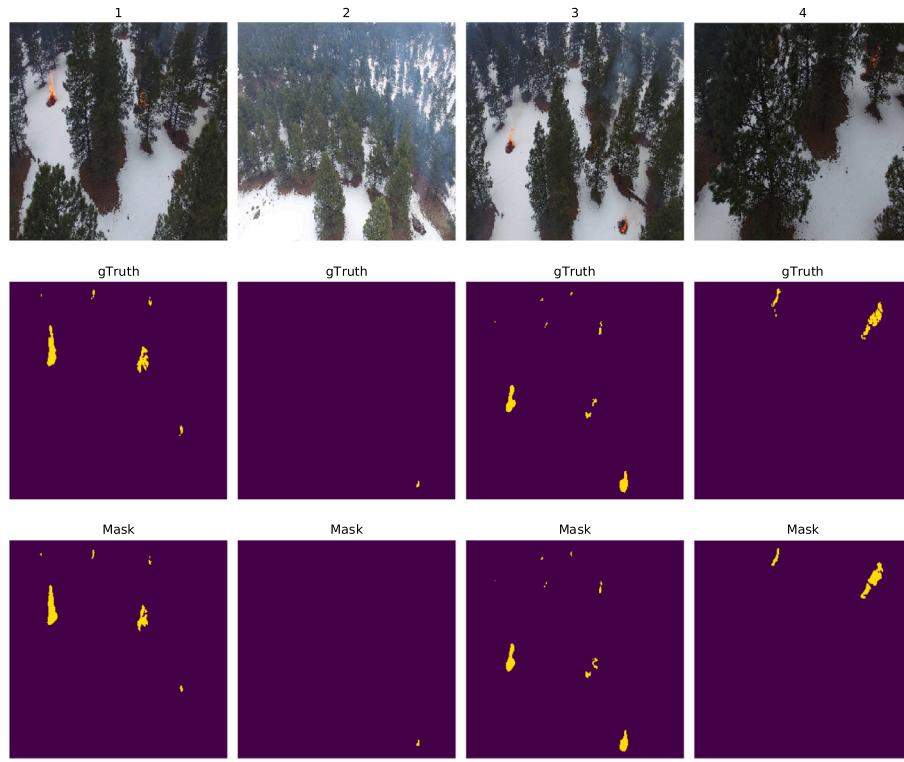


Fig. 7. Performance of the fire segmentation on six frames of the test set.

using the MATLAB (TM) Image Labeler tool for the ground truth data. The ground truth masks and data are generated based on the human subject matter expert (SME) eye efficiency to mark the fire pixels using manual polygon shape in MATLAB (TM) Image Labeler. The split ratio between the training and validation data is 85% and 15%. The frames and ground truth data were shuffled accordingly before importing into the training model. The maximum number of epochs is 30; however, an early stop callback was considered when the performance does not substantially change. The batch size for the training is 16. Fig. 7 demonstrates six samples of the test set along with the expected ground truth masks and the generated masks from the trained network. The first row is the input frame to the model, the second row is the ground truth (gTruth) which is the expected mask, and the last row is the generated mask by the trained model. Also, Table 4 shows the performance evaluation of this model. In this table, precision, recall, and Area Under Curve (AUC), F1-score, sensitivity, specificity, and Mean Intersection-Over-Union (Mean IOU) are reported.

4.3. Federated learning: Fire segmentation on drone's computer

This section uses the structure of Section 3.2 and the U-Net model is utilized for the Fire Segmentation challenge. To test the capability of the drone's computer regarding the machine learning approach, an AI mini-computer, NVidia Jetson Nano [17], is used to train a model for the Fire segmentation approach on the FLAME dataset.

Jetson Nano is available in two versions: (1) 4GB RAM developer kit, and (2) 2GB RAM developer kit. In this Implementation, the 4GB version is used with the technical specifications of a 128-core Maxwell GPU, a Quad-core ARM A57 @ 1.43 GHz CPU, 4GB LPDDR4 RAM, and a 32GB microSD storage. To test Jetson Nano for the federated learning, items (9) and (10) from Table 2 are used for the fire segmentation. We assumed that each drone is equipped with Jetson Nano. Tensorflow Lite (TF-Lite) and the default Tensorflow are the two versions of this framework. TF-Lite is designed and integrated for inference on devices and models with limited computation capability. The full

explanation of how to install Tensorflow on NVidia Jetson Nano is available online [49].

Since Jetson Nano has limited RAM, we assumed that each drone has access to a portion of the FLAME dataset. Only 500 fire images and masks are considered for the training and validation phase on the drone. As we aimed at learning a model on a smaller subset of the FLAME dataset and inferring that model, the default Tensorflow version is used here. For this purpose, only 500 images and masks were used for the training and validation of the fire segmentation approach. Also, the image and mask dimension for each input is reduced to $128 \times 128 \times 3$ rather than $512 \times 512 \times 3$. To save more memory on the RAM, all peripherals were turned off and only WiFi was working at that time for the Secure Shell (SSH) connection. The designed and trained model was sent using the SSH File Transfer Protocol (SFTP) to another desktop computer as an aggregation federated server. An external wireless network interface card was used and the Jetson Nano booted using a cellphone power-bank. The setup of this node is demonstrated in Fig. 8.

Compared to the desktop training and validation phases in Section 4.2, the IOU was reduced because of the smaller number of training images and masks. The reported fire segmentation accuracy is 99% which is similar to the previous section. However, the IOU metric which is important in image segmentation problems is reduced to 49.75%. Also, the training phase was longer due to limitations on GPU cores. For the future, researchers may consider two or more local federated nodes to increase the performance for the global trained model via the aggregated federated server.

5. Open challenges regarding the dataset

This study proposed two different challenges regarding the dataset. We encourage other researchers to consider this available FLAME dataset and improve the accuracy of the fire classification problem which might include providing more ground truth data as a labeled mask for the fire segmentation. Also, three other thermal images such as GreenHot, WhiteHot, and the fusion are also available for further investigation regarding the fire segmentation and classification. Other

Table 4

Performance evaluation of the customized U-Net on the fire dataset for the fire segmentation.

Dataset	Performance evaluation						
	Precision(%)	Recall(%)	AUC(%)	F1-Score(%)	Sensitivity(%)	Specificity(%)	IOU(%)
Image Segmentation	91.99	83.88	99.85	87.75	83.12	99.96	78.17

**Fig. 8.** NVidia Jetson Nano with the wireless card and power-bank battery.

considerations include the type of fire elements including the different structures of the fire (white hot core, exterior, etc.). These elements can be segmented as different parts of the fire to have better understanding of the fire behavior. Another challenge or problem could be investigating different fire detection models on these thermal images to see which type of the data has better accuracy for the model. Perhaps, another important research direction would be developing integrative imagery-based fire spread models by incorporating other environmental factors such as the terrain model, and the vegetation fuel profile of the region. Extracting such factors from the images and videos and comparing with alternative sources can advance the image-based fire modeling algorithms. Other open challenges and future directions regarding the FLAME dataset include but not limited to (1) transfer learning, (2) context-based fire detection using a model and then zero-shot learning, (3) fire content analysis, (4) temporal analysis, (5) surrogate airborne perspective analysis, (6) metric design, (7) performance standards, (8) user displays, (9) federated and distributed learning using multiple drones, (10) edge node efficiency, and (11) occlusion robustness.

6. Conclusion

This paper provided the FLAME (Fire Luminosity Airborne-based Machine learning Evaluation) dataset for pile burns in Northern Arizona forest. Two drones were used to collect aerial frames and videos in four different palettes of normal, Fusion, WhiteHot, and GreenHot using normal and thermal cameras. The frames were used in two different applications, in the first challenge, a convolutional neural network was used as a deep learning binary fire classification to label data. In the second approach, a machine learning approach was proposed to extract fire masks from fire labeled data as an image segmentation technique. These exemplary applications show the utility of the FLAME dataset in developing computer tools for fire management and control. Also, FLAME dataset can be used as a benchmark dataset for

testing generic image processing algorithms. We provide numerical result for the performance of the proposed two algorithms developed for image classification and detection. We believe that developing more advanced models by the research community can further improve the reported results. Another potential use for this dataset is developing fire classification and detection algorithms by a collective analysis of different imaging modalities including regular and thermal images. Also, researchers can utilize fire segmentation methods to define related networking and monitoring problems, such as optimal task scheduling for a fleet of drones to optimally cover the pile burns in a certain region at shortest time possible.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgment

This material is based upon work supported by the Air Force Office of Scientific Research, United States of America under award number FA9550-20-1-0090 and the National Science Foundation, United States of America under Grant Number 2034218. Thanks to Neil Chapman and Paul Summerfelt of the Flagstaff Fire Department for providing access to the prescribed fire.

References

- [1] National interagency fire center, 2020, <https://www.nifc.gov/fireInfo/nfn.htm>, (Accessed on 28 october 2020).
- [2] National interagency fire center, 2020, https://www.nifc.gov/fireInfo/fireInfo_statistics.html, (Accessed on 28 october 2020).
- [3] J. Toledo-Castro, P. Caballero-Gil, N. Rodríguez-Pérez, I. Santos-González, C. Hernández-Goya, R. Aguasca-Colomo, Forest fire prevention, detection, and fighting based on fuzzy logic and wireless sensor networks, Complexity 2018 (2018) 1–17, <http://dx.doi.org/10.1155/2018/1639715>.
- [4] H. Kaur, S.K. Sood, Fog-assisted IoT-enabled scalable network infrastructure for wildfire surveillance, J. Netw. Comput. Appl. 144 (2019) 171–183.
- [5] J.L. Coen, W. Schroeder, S.D. Rudlosky, Transforming wildfire detection and prediction using new and underused sensor and data sources integrated with modeling, in: Handbook of Dynamic Data Driven Applications Systems, Springer, 2018, pp. 215–231.
- [6] Q. Huang, A. Razi, F. Afghah, P. Fule, Wildfire spread modeling with aerial image processing, in: 2020 IEEE 21st International Symposium on "a World of Wireless, Mobile and Multimedia Networks", WoWMoM, IEEE, 2020, pp. 335–340.
- [7] P. Friedlingstein, M. Jones, M. O'sullivan, R. Andrew, J. Hauck, G. Peters, W. Peters, J. Pongratz, S. Sitch, C. Le Quéré, et al., Global carbon budget 2019, Earth Syst. Sci. Data 11 (4) (2019) 1783–1838.
- [8] M. Erdelj, E. Natalizio, K.R. Chowdhury, I.F. Akyildiz, Help from the sky: Leveraging UAVs for disaster management, IEEE Pervasive Comput. 16 (1) (2017) 24–32.
- [9] F. Afghah, M. Zaeri-Amirani, A. Razi, J. Chakareski, E. Bentley, A coalition formation approach to coordinated task allocation in heterogeneous UAV networks, in: 2018 Annual American Control Conference, ACC, 2018, pp. 5968–5975, <http://dx.doi.org/10.23919/ACC.2018.8431278>.
- [10] R. Aggarwal, A. Soderlund, M. Kumar, D. Grymin, Risk aware SUAS path planning in an unstructured wildfire environment, in: 2020 American Control Conference, ACC, IEEE, 2020, pp. 1767–1772.
- [11] F. Afghah, A. Razi, J. Chakareski, J. Ashdown, Wildfire monitoring in remote areas using autonomous unmanned aerial vehicles, in: IEEE INFOCOM 2019 - IEEE Conference on Computer Communications Workshops, INFOCOM WKSHPS, 2019, pp. 835–840, <http://dx.doi.org/10.1109/INFCOMW.2019.8845309>.
- [12] A. Shamsoshoara, M. Khaledi, F. Afghah, A. Razi, J. Ashdown, Distributed cooperative spectrum sharing in uav networks using multi-agent reinforcement learning, in: 2019 16th IEEE Annual Consumer Communications & Networking Conference, CCNC, IEEE, 2019, pp. 1–6.

- [13] A. Shamsoshoara, M. Khaledi, F. Afghah, A. Razi, J. Ashdown, K. Turk, A solution for dynamic spectrum management in mission-critical uav networks, in: 2019 16th Annual IEEE International Conference on Sensing, Communication, and Networking, SECON, IEEE, 2019, pp. 1–6.
- [14] A. Shamsoshoara, F. Afghah, A. Razi, S. Mousavi, J. Ashdown, K. Turk, An autonomous spectrum management scheme for unmanned aerial vehicle networks in disaster relief operations, *IEEE Access* 8 (2020) 58064–58079.
- [15] Y.E. Wang, G.-Y. Wei, D. Brooks, Benchmarking TPU, GPU, and CPU platforms for deep learning, 2019, arXiv preprint arXiv:1907.10701.
- [16] Edge TPU - google, 2020, <https://cloud.google.com/edge-tpu>, (Accessed on 29 october 2020).
- [17] NVIDIA jetson nano, 2020, <https://developer.nvidia.com/embedded/jetson-nano-developer-kit>, (Accessed on 29 october 2020).
- [18] H. Wu, H. Li, A. Shamsoshoara, A. Razi, F. Afghah, Transfer learning for wildfire identification in UAV imagery, in: 2020 54th Annual Conference on Information Sciences and Systems, CISS, IEEE, 2020, pp. 1–6.
- [19] K. Bonawitz, H. Eichner, W. Grieskamp, D. Huba, A. Ingerman, V. Ivanov, C. Kiddon, J. Konečný, S. Mazzocchi, H.B. McMahan, et al., Towards federated learning at scale: System design, 2019, arXiv preprint arXiv:1902.01046.
- [20] B. Brik, A. Ksentini, M. Bouaziz, Federated learning for UAVs-enabled wireless networks: Use cases, challenges, and open problems, *IEEE Access* 8 (2020) 53841–53849.
- [21] DJI - phantom 3 professional, 2020, <https://www.dji.com/phantom-3-pro>, (Accessed on 30 August 2020).
- [22] D. Company, DJI - Matrice 200 V1, 2020, <https://www.dji.com/matrice-200-series/info#specs>, (Accessed on 30 August 2020).
- [23] DJI Zenmuse X4s - specifications, FAQs, videos, tutorials, manuals, DJI GO - DJI, 2020, <https://www.dji.com/zenmuse-x4s/info#specs>, (Accessed on 30 August 2020).
- [24] FLIR Vue pro r radiometric drone thermal camera | FLIR systems, 2020, <https://www.flir.com/products/vue-pro-r/>, (Accessed on 08 December 2020).
- [25] A. Shamsoshoara, F. Afghah, A. Razi, L. Zheng, P. Fulé, E. Blasch, Aerial images for Pile burn detection using drones (UAVs), *IEEE Dataport*, 2020, <http://dx.doi.org/10.21227/qad6-r683>.
- [26] A. Shamsoshoara, Aerial images for pile burn detection using drones (UAVs), WINIP LAB (2020) https://youtu.be/bHK6g37_KyA, Wireless Networking & Information Processing (WINIP) LAB, accessed on 11/19/2020.
- [27] T. Celik, H. Demirel, Fire detection in video sequences using a generic color model, *Fire Saf. J.* 44 (2) (2009) 147–158.
- [28] M.M. Umar, L.C.D. Silva, M.S.A. Bakar, M.I. Petra, State of the art of smoke and fire detection using image processing, *Int. J. Signal Imaging Syst. Eng.* 10 (1–2) (2017) 22–30.
- [29] N.I. binti Zaidi, N.A.A. binti Lokman, M.R. bin Daud, H. Achmad, K.A. Chia, Fire recognition using RGB and YCbCr color space, *ARPEN J. Eng. Appl. Sci.* 10 (21) (2015) 9786–9790.
- [30] C. Yuan, Z. Liu, 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.
- [31] X. Qi, J. Ebert, A computer vision based method for fire detection in color videos, *Int. J. Imaging* 2 (S09) (2009) 22–34.
- [32] S. Kundu, V. Mahor, R. Gupta, A highly accurate fire detection method using discriminative method, in: 2018 International Conference on Advances in Computing, Communications and Informatics, ICACCI, IEEE, 2018, pp. 1184–1189.
- [33] F. Chollet, Xception: Deep learning with depthwise separable convolutions, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2017, pp. 1251–1258.
- [34] C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens, Z. Wojna, Rethinking the inception architecture for computer vision, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2016, pp. 2818–2826.
- [35] C. Szegedy, S. Ioffe, V. Vanhoucke, A. Alemi, Inception-v4, inception-resnet and the impact of residual connections on learning, 2016, arXiv preprint arXiv: 1602.07261.
- [36] K. He, X. Zhang, S. Ren, J. Sun, Deep residual learning for image recognition, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2016, pp. 770–778.
- [37] Y. Li, Y. Yuan, Convergence analysis of two-layer neural networks with relu activation, in: *Advances in Neural Information Processing Systems*, 2017, pp. 597–607.
- [38] A. Shamsoshoara, Fire-detection-uav-aerial-image-classification-segmentation-unmannedaerialvehicle, in: GitHub repository, GitHub, 2020, <https://github.com/AlirezaShamsoshoara/Fire-Detection-UAV-Aerial-Image-Classification-Segmentation-UnmannedAerialVehicle>.
- [39] C. Yuan, Z. Liu, Y. Zhang, Aerial images-based forest fire detection for firefighting using optical remote sensing techniques and unmanned aerial vehicles, *J. Intell. Robot. Syst.* 88 (2–4) (2017) 635–654.
- [40] T. Celik, H. Özkarmanlı, H. Demirel, Fire and smoke detection without sensors: Image processing based approach, in: 2007 15th European Signal Processing Conference, IEEE, 2007, pp. 1794–1798.
- [41] A. Khalil, S.U. Rahman, F. Alam, I. Ahmad, I. Khalil, Fire detection using multi color space and background modeling, *Fire Technol.* (2020) 1–19.
- [42] Image segmentation software from Labelbox, 2020, <https://labelbox.com/product/image-segmentation>, (Accessed on 25 october 2020).
- [43] image labelling tool: Bitbucket, 2020, <https://bitbucket.org/ueacomputervision/image-labelling-tool/src/master/>, (Accessed on 25 october 2020).
- [44] labelImg: LabelImg is a graphical image annotation tool and label object bounding boxes in images, 2020, <https://github.com/tzutalin/labelImg>, (Accessed on 25 october 2020).
- [45] Image labeler, MATLAB & simulink, 2020, <https://www.mathworks.com/help/vision/ug/get-started-with-the-image-labeler.html>, (Accessed on 25 october 2020).
- [46] GIMP - GNU image manipulation program, 2020, <https://www.gimp.org/>, (Accessed on 25 october 2020).
- [47] O. Ronneberger, P. Fischer, 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.
- [48] ELU activation function, ML glossary documentation, 2020, https://ml-cheatsheet.readthedocs.io/en/latest/activation_functions.html#elu, (Accessed on 26 october 2020).
- [49] Installing tensorflow for jetson platform :: NVIDIA deep learning frameworks documentation, 2021, <https://docs.nvidia.com/deeplearning/frameworks/install-tf-jetson-platform/index.html>, (Accessed on 07 january 2021).



Alireza Shamsoshoara received a B.Sc. degree in Electrical Engineering and an M.Sc. degree in Electrical and Communication Engineering in 2012 and 2014 respectively. Also, he received an M.Sc. degree in Informatics from Northern Arizona University in 2018. He worked as a network engineer from 2015 to 2017. Currently, he is a Ph.D. student in the School of Informatics, Computing & Cyber Systems at Northern Arizona University. His main research interests are wireless networks, UAV networks, spectrum sharing, and machine learning.



Fatemeh Afghah is an Associate Professor with the School of Informatics, Computing and Cyber Systems, Northern Arizona University, where she is the Director of Wireless Networking and Information Processing Laboratory. Her research interests include wireless communication networks, decision making in multi-agent systems, radio spectrum management, and AI-assisted healthcare. Her research has been continually supported by NSF, AFRL, AFOSR and ABOR. She is the recipient of several awards including the AFOSR Young Investigator Award (2019), NSF CAREER Award (2020), NAU's Most Promising New Scholar Award (2020), and NSF CRII Award (2017). She is the author/co-author of over 80 peer-reviewed publications.



Abolfazl Razi is an assistant professor of Electrical Engineering at Northern Arizona University (NAU). He received his B.S., M.S. and Ph.D. degrees, all in Electrical Engineering, respectively from Sharif University (1994–1998), Tehran Polytechnic (1999–2001), and University of Maine (2009–2013). Prior to joining NAU, he held two post-doctoral positions in the field of machine learning and predictive modeling at Duke University (2013–2014), and Case Western Reserve University (2014–2015). He is the recipient of several competitive awards including the Best Research of MCI in 2008, Best Graduate Research Assistant of the Year Award from the College of Engineering, University of Maine in 2011, and the Best Paper Award from the IEEE/CANEUS Fly By Wireless Workshop in 2011. His current research interests include AI-enabled wireless networking, nonlinear compressive sensing, topology control, UAV networks, and predictive modeling.



Liming Zheng received a B.Sc. degree in Electrical Engineering at Northern Arizona University in 2019 and a B.Sc. degree in Electronic and Information Engineering at Chongqing University of Posts and Telecommunications in 2019. Currently, he is an Informatics & Computing Ph.D. student in the School of Informatics, Computing & Cyber Systems at Northern Arizona University. His main research interests are UAV networks and machine learning.



Peter Fulé's research is at the intersection of forests, wild-fire, climate and people, using multiple research techniques including tree rings and simulation modeling to assess tree growth and forest fires over many centuries.



Erik Blasch is a program officer at the Air Force Research Laboratory (AFRL) – Air Force Office of Scientific Research (AFOSR) in Arlington, VA. Previously he was a principal scientist at the AFRL Information Directorate in Rome, NY, USA (2012–2017), exchange scientist to the Defence Research and Development Canada (DRDC) in Valcartier, Quebec (2010–2012), and Information Fusion Evaluation Tech Lead for the AFRL Sensors Directorate–COMprehensive Performance Assessment of Sensor Exploitation (COMPASE) center in Dayton, OH (2000–2009). Additional assignments include USAF Reserve Officer Col supporting intelligence, acquisition, and space technology. He was an adjunct associate professor in Electrical and Biomedical Engineering (2000–2010) at Wright State University and the Air Force Institute of Technology (AFIT) teaching classes in signal processing, electronics, and information fusion as well as research adjunct appointments at the Univ. of Dayton (2001–2014), Binghamton University (2012–2017), and Rochester Institute of Technology (2015–2017). Dr. Blasch was a founding member of the International Society of Information Fusion (ISIF), (www.isif.org), 2007 President,

and Board of Governors (2000–2010). He served on the IEEE Aerospace and Electronics Systems Society (AEES) Board of Governors (2011–2016), distinguished lecturer (2012–2018), co-chair of 5 conferences, and associate editor of 3 academic journals. He has focused on information fusion, target tracking, robotics, and pattern recognition research compiling 800+ scientific papers and book chapters. He holds 25 patents, received 33 team-robotics awards, presented 60+ tutorials, and provided 9 plenary talks. His coauthored books include High-Level Information Fusion Management and Systems Design (Artech House, 2012), Context-enhanced Information Fusion (Springer, 2016), Multispectral Image Fusion and Colorization (SPIE, 2018), and Handbook of Dynamic Data Driven Applications Systems (Springer 2018). Dr. Blasch received his B.S. in Mechanical Engineering from the Massachusetts Institute of Technology ('92) and Masters' Degrees in Mechanical ('94), Health Science ('95) and Industrial Engineering (Human Factors) ('95) from Georgia Tech and attended the University of Wisconsin for a MD/PhD Neuroscience/Mechanical Engineering until being call to military service in 1996 to the United States Air Force. He completed an MBA ('98), MS Econ ('99), and Ph.D. ('99) in Electrical Engineering from Wright State University and is a graduate of Air War College ('08). He is the recipient of the IEEE Bioengineering Award (Russ-2008), IEEE AEES magazine best paper Award (Mimmo-2012), Military Sensing Symposium leadership in Data Fusion Award (Mignogna-2014), Fulbright scholar selection (2017), and 15 research/technical and team awards from AFRL. He is an American Institute of Aeronautics and Astronautics (AIAA) Associate Fellow, Society of Photonics and Industrial Engineers (SPIE) Fellow, and Institute of Electrical and Electronics Engineers (IEEE) Fellow.