

Received March 26, 2020, accepted April 11, 2020, date of publication April 15, 2020, date of current version April 30, 2020.

Digital Object Identifier 10.1109/ACCESS.2020.2987991

# Optimal Placement and Intelligent Smoke Detection Algorithm for Wildfire-Monitoring Cameras

JIE SHI<sup>ID</sup>, (Graduate Student Member, IEEE), WEI WANG<sup>ID</sup>, (Graduate Student Member, IEEE),  
YUANQI GAO<sup>ID</sup>, (Graduate Student Member, IEEE),  
AND NANPENG YU<sup>ID</sup>, (Senior Member, IEEE)

Department of Electrical and Computer Engineering, University of California at Riverside, Riverside, CA 92501, USA

Corresponding author: Nanpeng Yu (nyu@ece.ucr.edu)

**ABSTRACT** Smoke produced by wildfires is usually visible much earlier than flames. Hence, early detection of wildfire smoke is essential to prevent severe property losses and heavy casualties from catastrophic wildfires. Camera networks are being built and expanded to achieve timely wildfire smoke detection. To achieve the best camera coverage and detection accuracy with limited budget, an intelligent video smoke detection algorithm and an optimal wildfire camera placement strategy are in a critical need. In this paper, we propose an efficient video smoke detection framework designed for embedded applications on local cameras. It consists of two modules. In the first module, the original video frames are processed by local binary patterns and a dense optical flow estimator. In the second module, the produced features are then fed into a lightweight deep convolutional neural network, which serves as a binary classifier to detect the presence of smoke. We also formulate the wildfire camera placement problem as a binary integer programming problem to minimize the overall fire risk of a given area. Case studies on real-world videos are carried out to validate the accuracy as well as the computational and memory efficiency of the proposed smoke detection framework. We also validate our proposed camera placement strategy by simulating the deployment of wildfire cameras across a test region in Southern California.

**INDEX TERMS** Smoke detection, camera placement, deep neural network, local binary patterns, optical flow.

## I. INTRODUCTION

Uncontrolled wildfires can cause billions of dollars in losses and heavy casualties. For example, a single wildfire in 2018 killed 86 people and burned more than 153 thousand acres across Butte County of California. Early detection of wildfires could make a huge difference when it comes to stabilizing a rapidly spreading fire and preventing catastrophic losses. Regions with high wildfire hazards, particularly California, have been expanding their camera networks to improve their wildfire detection systems [1]. Most wildfires start with visible smoke. The flames are usually hidden at the initial stage, making smoke the only observable feature of early wildfires. Thus, the goal of the wildfire camera

networks is to detect smoke and send timely warnings of wildfires to the fire department.

There are two technical questions associated with the expansion of camera networks. First, how to automatically and efficiently detect smoke from videos captured by the camera networks. Second, how to achieve the maximum fire risk reduction with limited camera network expansion/deployment budget. To answer the first technical question, we need to address two technical challenges. First, visually monitoring would require 24 hours of oversight by two or more people working in shifts, which demands tremendous man power for a large scale camera network. Second, sending high definition videos of a large camera network to a central processor requires high communication bandwidth, which can be extremely costly. To tackle these two technical challenges, we need to develop an automatic and computationally efficient smoke detection framework, which

The associate editor coordinating the review of this manuscript and approving it for publication was Tony Thomas.

can be executed on local camera platforms. Before answering the second technical question, we need to recognize that the installation of a large number of remote cameras with remote communication and independent power system can cause considerable costs. To address this technical question, we aim to develop an optimal camera placement algorithm, which minimizes the risk of delayed wildfire detection with a limited budget for camera networks deployment. Note that the optimal camera placement decision is highly related to the performance of the smoke detection algorithm. The deterioration of smoke detection accuracy as a function of the distance between smoke and cameras should be taken into consideration when designing the optimal camera placement strategy. The objective of this paper is to provide viable answers to these two critical technical questions.

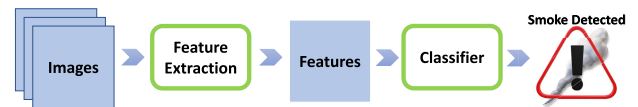
With the prevalent installation of digital cameras, automatic smoke detection in images and videos has received widespread interest in the past two decades. The related literature will be discussed comprehensively in Section II. Recently inspired by the remarkable success of deep learning, researchers have incorporated deep neural networks into smoke detection frameworks and produced promising results. However, these deep learning based approaches are usually computationally intensive and require relatively large memory space, limiting their applicability in local embedded applications such as wildfire camera platforms.

Installing remote wildfire camera on the top of mountains or other vantage points is usually very expensive due to the cost of satellite communication equipment and independent power system module. It is reported [2] that the total infrastructure cost of a single wildfire camera is around \$75,000. There are additional operating and maintenance costs associated with each remote camera, which vary by location. Meanwhile, the distance between wildfire smoke and the camera can affect the smoke detection accuracy as distant smoke gets blurred in the video. Facing limited budget, the network of cameras need to be strategically placed to maximize the overall benefit.

Numerous image/video based smoke detection algorithms have been developed and tested on real-world data. However, there is a lack of a comprehensive review of existing research. To the best of our knowledge, [3] published in 2013 is the only survey paper that systematically reviews the existing smoke detection algorithms in the literature. Many innovative smoke detection frameworks, such as those based on deep neural networks, have been proposed since then. To fill the gap, we provide a comprehensive review of existing image/video smoke detection algorithms in this paper.

The contributions of this paper are summarized as follows:

- We develop a lightweight physics-based video smoke detection algorithm for remote cameras with limited computational and storage resources. By extracting useful physical properties of the videos and leveraging the MobileNetV2 framework, the proposed algorithm is not



**FIGURE 1.** Two-stage framework of most image/video smoke detection approaches.

only fast and efficient, but also as accurate as the state-of-the-art algorithms.

- We develop an innovative optimal wildfire camera placement strategy to maximize fire risk reduction given a limited budget.
- We present a comprehensive review of the existing image/video smoke detection methodologies in the literature. In particular, we cover the most recent approaches based on deep learning.

The rest of this paper is organized as follows: Section II provides a comprehensive review of the existing image/video smoke detection literature. Section III presents the framework and technical methods used in our proposed video smoke detection algorithm. Section IV discusses the problem formulation of optimal camera placement. Section V carries out case studies of the proposed smoke detection framework with real-world wildfire smoke videos. Section V also provides a demonstration of our proposed optimal camera placement strategy for a small test region. The conclusions are stated in Section VI.

## II. LITERATURE REVIEW OF IMAGE/VIDEO SMOKE DETECTION

In this section, we present a comprehensive review of the existing image/video smoke detection algorithms in literature. We divide the literature into two groups: computer vision based approaches and deep learning based approaches. Throughout this paper, we consider a method that does not involve any deep neural network as a computer vision based approach. Otherwise, it is a deep learning based approach. At the end of the literature review section, we discuss the potential improvements that can be made beyond the existing work.

### A. COMPUTER VISION BASED APPROACHES

To the best of our knowledge, the earliest smoke detection framework based on digital images dates back to the early '90s, when some seminal works [4], [5] were carried out by a European research group. This research topic experienced exponential growth beginning in 2000. Although a large number of papers have been published since then, the majority of these works detect smoke in images or videos through a common two-stage framework as shown in Fig. 1. The first stage is designed to extract features,<sup>1</sup> which often include color, motion, texture, shape, and energy. These features are then

<sup>1</sup>In this paper, a feature refers to any piece of information that is relevant for detecting smoke in an image or video.

sent to a classifier in the second stage which detects smoke presence. There are many ways to extract features and build classifiers. The techniques involved are often shared or combined in different papers over a long period of time. Moreover, a typical smoke detection framework usually incorporates multiple feature extraction processes. Therefore, it is more informative to review the literature in a taxonomical rather than chronological manner. In the following two parts, we present a systematic review of the methods used for smoke feature extraction and smoke detection classifier.

### 1) SMOKE FEATURE EXTRACTION

The features often extracted for smoke detection include color, motion, texture, shape, and energy. The motivation and techniques for extracting each type of feature are summarized separately below.

#### *a: COLOR FEATURE*

It is observed that many types of smoke exhibit a grayish or a light blue color although the specific color bandwidth can vary greatly with background environment and burning materials. Meanwhile, the sudden occurrence of smoke will generally decrease the chrominance channel values of the image pixels. Based on these two observations, the following techniques are often used to evaluate the color properties in given images and extract corresponding features for classification:

- **Color value rules:** This type of technique defines smoke color value ranges with rules. If a pixel's color value falls into these ranges, then it is considered as a potential smoke pixel. The related references include [6]–[17].
- **Decrease in U and Y channel of YUV color space:** This type of technique checks the trends of chrominance values of image pixels. As the smoke gets thicker, the chrominance values of the smoke region are expected to decrease. Refer to [18], [19] for examples.
- **Reference color model:** This type of technique builds a reference smoke color model. The distances between the given image pixel colors and that of the reference model are quantified and serve as features. The related references include [20]–[22].
- **Color histogram analysis:** This type of technique creates a histogram of pixels' color values from a certain image region (e.g., a region with moving object). The histogram itself can serve as features [23], [24]. Alternatively, the distance between this histogram and a predefined template can be used as a feature [25].
- **Probabilistic model:** This type of technique builds a probabilistic model to represent the density distribution of smoke color. A given image region is regarded as a potential smoke area if the corresponding probability exceeds a predefined threshold [26], [27].
- **Others:** [28] applies fuzzy c-means clustering to segment possible smoke regions based on color. Reference [29] employs a 1-D discrete wavelet transform (DWT) to

monitor the temporal variation of high frequency components of average color values for each channel in a given image block. The measurement of this variation is used as a color feature.

#### *b: MOTION FEATURE*

Although smoke is moving objects, their motions are not sharply contrasted. The moving smoke is essentially propagation of newly generated smoke particles on an already smoke-filled background [30]. In general, the smoke has an upward moving trend near the fire sources before they disperse in the sky. These properties are usually exploited in the video smoke detection frameworks. The commonly used motion feature extraction techniques are summarized as follows:

- **Background subtraction:** Background subtraction is a large family of approaches broadly applied in various computer vision tasks to detect moving objects. The fundamental idea is to initialize and maintain a background model. The moving object mask is derived by performing a subtraction between the current frame and the background model. Many video smoke detection frameworks use background subtraction to extract motion features [8], [10]–[15], [17]–[20], [22], [23], [26]–[29], [31]–[40].
- **Optical flow:** Estimating optical flow is another popular approach to extract motion features in computer vision applications. The optical flow based methods attempt to estimate the motion velocities and directions of individual points, assuming the brightness of the moving objects is constant through sequential frames [41]. The related video smoke detection references include [11], [16], [26], [37], [42], [43].
- **Others:** Reference [8] proposes a fast motion evaluation algorithm that only calculates the orientation of motion. The motion orientations are then accumulated temporally across a time interval. The most frequent motion orientation is then selected as a feature to reduce overall errors. Reference [25] applies motion history image (MHI) to detect and segment moving objects from smoke videos. Reference [44] proposes a Choquet fuzzy integral based method to detect moving objects.

#### *c: TEXTURE FEATURE*

Smoke has a unique texture, which can be used to distinguish it from most surrounding objects. The following two types of texture extraction methods are often employed in the image/video smoke detection algorithms:

- **Gray level co-occurrence matrices (GLMC):** GLMC describes the spatial relationships of pairs of gray values of pixels in images [45]. The fundamental idea is to count the frequency of gray value pairs of pixels that have certain spatial offset. The related smoke detection references include [33], [34], [38], [44], [46].
- **Local binary patterns (LBP):** LBP is widely used to measure the spatial structure of local image texture [47].

The simplest pixel-wise LBP operator compares the gray value of a given pixel with its 8 neighboring pixels, which yields an 8-digit binary number. This number represents the texture information of the corresponding local image block. The related smoke detection references include [23], [24], [34], [36], [40], [48]–[51]

- Others: Reference [37] develops a high order linear dynamical system to extract texture features from video frames. Reference [17] applies center symmetric local ternary patterns (CS-LTP) to generate texture features. Reference [40] employs local phase quantization (LPQ) to obtain texture features.

#### d: SHAPE FEATURE

The shapes of many types of smoke can be volatile and disordered. They also change frequently due to the propagation of smoke particles and airflow variation. The following two types of approaches are often applied to evaluate shape information of suspicious smoke regions:

- Quantity change of potential smoke pixels: Smoke shape variation corresponds to the quantity change of potential smoke pixels in video frames. This is particularly evident in the early stage of smoke, when the size of smoke is growing fast. The related smoke detection literature includes [6], [13], [15], [31].
- Perimeter/area (P/A) ratio: P/A ratio is simply the perimeter of an object divided by its area. It reflects the magnitude of shape disorder. The related smoke detection references include [7], [13], [15], [25]

#### e: ENERGY FEATURE

Smoke gradually softens the edges in video frames when it is not sufficiently thick to cover the entire background. This results in gradual energy loss of the video frames, especially in the high frequency domain. Therefore, 2D discrete wavelet transform (DWT) are often adopted to monitor the energy change in high frequency domain.

2D DWT captures both frequency and location information in an image. It usually serves as a filter bank for multi-resolution analysis (MRA) in the computer vision applications. For smoke detection tasks, a single level DWT is often used for extracting high frequency signals. The related references include [18]–[23], [29], [31], [37]. [52] adopted both 2D DWT and 2D discrete cosine transform (DCT) to generate energy features based on frequency domain signals.

In addition to the commonly used features mentioned above, some other features have also been extracted and tested in smoke detection applications. For example, [18] and [19] extract flicker information using temporal wavelet transform. Reference [12] proposes to use correlation descriptors to describe spatio-temporal relationship between video blocks. References [24] and [26] extract gradient based histograms as spatial features. References [24] and [53] extract Haar-like features to represent certain characteristics in the

image such as edges. Reference [39] builds a smoke saliency map based on motion and lightness.

## 2) CLASSIFIERS APPLIED IN SMOKE DETECTION

The classifiers take the extracted features as inputs and output whether or not there is smoke in a given image or video. The commonly used classifiers in smoke detection literature are listed as follows:

- Rule based classifier: This type of classifier assesses the features with one or multiple rules. If all or certain combinations of rules are satisfied, then smoke is detected. The related references include [6]–[10], [16]–[18], [27], [30], [32], [42], [44], [54].
- Support vector machine (SVM): SVMs are simple and effective classifiers that are widely used in machine learning applications. For binary classification, SVM attempts to find a hyperplane that separates two classes with maximal margin. Many video smoke detection frameworks employ SVMs as classifiers [12], [23], [28], [29], [34], [37], [38], [40], [49], [50], [52].
- Shallow Neural network: Shallow neural networks were often used as classifiers in image/video smoke detection frameworks. Different from the deep learning based approaches that will be discussed later, the neural networks adopted in the computer vision based approaches are usually feed-forward neural networks with one or two hidden layers [11], [13], [15], [31], [33], [43], [46], [48].
- Mixture of Gaussians (MoG): MoG is a probabilistic model to represent the probability density of feature vectors. It can serve as a binary classifier to assess the probability of the presence of smoke in a given image/video. The related references include [20]–[22].
- K-nearest neighbor (KNN): KNN is a non-parametric classifier that simply assigns an object to the class most common among its k nearest neighbors. See [13], [15], [51], [52] for KNN's usage in smoke detection.
- AdaBoost classifier: AdaBoost is an ensemble classifier composed of a bunch of weak classifiers. Each weak classifier is trained with weighted data points. The final decision is a weighted combination of all weak classifiers. The related video smoke detection references include [24], [36], [53].
- Others: Reference [25] proposes to use a fuzzy logic classifier. Reference [26] builds a random forest as the classifier. Reference [14] develops an ensemble classifier called entropy-functional-based online adaptive decision fusion.

## B. DEEP LEARNING BASED APPROACHES

Deep neural networks have been remarkably successful in various computer vision applications. In particular, deep convolutional neural networks (CNNs) based frameworks are currently the best solutions to a variety of image classification tasks such as the ImageNet Challenges [55]. Inspired by

the great success of convolutional neural networks, many researchers have adopted deep neural networks to solve the image/video smoke detection problem. In this paper, we consider a smoke detection algorithm as a deep learning based approach if it utilizes a deep neural network with at least one convolutional neural network layer.

References [56]–[58] are among the earliest works that use deep CNNs to detect smoke in either images or videos. Reference [56] employs AlexNet to classify a given image as a smoke or non-smoke image. Reference [57] detects and locates the fire/smoke regions using a CNN adapted from AlexNet. Reference [58] proposes a CNN with 6 convolutional layers and 2 fully connected layers to classify each sliding window of a given video frame into three classes: “Negative”, “Fire”, and “Smoke”. The location of the detected fire or smoke is indicated by the corresponding sliding window. Reference [59] builds an image smoke detection framework based on a 14-layer CNN that includes batch normalization and data augmentation. Reference [60] adopts Faster R-CNN [61] to detect and locate smoke regions with synthetic smoke images. References [62] and [63] develop domain adaptation based frameworks to better exploit synthetic smoke image data. Reference [64] proposes to use DeepLabV3+ [65] and a generative adversarial network to detect and predict the trend of smoke. Reference [66] builds a recurrent convolutional network for video smoke detection. Reference [67] uses MobileNetV2 to detect smoke in foggy surveillance environments. Reference [68] develops a joint video smoke detection framework based on faster R-CNN and 3D CNN. It is reported that the detection accuracy has been significantly improved in comparison to the 2D CNN frameworks.

Recently, researchers started to combine traditional feature extraction techniques with deep CNNs. For example, [69] applies motion-based geometrical image transformation on the video frames before sending them to a deep convolutional generative adversarial network for video smoke detection. Reference [70] proposes a composite smoke detection scheme that combines motion and color detection with YOLOv2 [71]. Specifically, the motion and color detection is carried out at the same time with YOLOv2. The final detected smoke regions are the intersections between the bounding boxes given by YOLOv2 and the potential smoke blocks suggested by the motion and color detection.

### C. CONNECTION BETWEEN THE PROPOSED APPROACH AND LITERATURE

The deep CNN based smoke detection frameworks generally outperform the traditional computer vision based approaches. However, most deep CNNs are computationally expensive and requires a large storage space for millions of network parameters. This is not the ideal solution for smoke detection using remote wildfire-monitoring cameras systems, which have limited computation power and storage capability. Although lightweight and computationally efficient CNNs such as MobilNetV2 have been adopted to detect smoke [67],

its accuracy is limited by the fact that video frames are only processed individually, thereby losing the key motion information in the video. In this paper, we develop a lightweight and computationally efficient smoke detection algorithm, which achieves a similar level of detection accuracy as that of deep CNNs. Our proposed algorithm synergistically combines the powerful feature extraction capability of the traditional computer vision based approach with a lightweight CNN. Specially, we propose to extract valuable motion information from videos by using local binary patterns (LBP) and optical flow analysis. These extracted features are then fed into the lightweight CNN to perform smoke detection. As will be shown in the case study, our proposed lightweight smoke detection algorithm can achieve a similar level of performance as that of deep CNNs by using only a fraction of the computational resource.

We have discussed two groups of smoke detection algorithms as well as the connection between our proposed framework and the existing approaches in the above paragraphs. In the next two sections, we will elaborate on the specific procedures of the proposed methodologies, which include a video-based smoke detection algorithm and an optimal wildfire camera placement strategy.

## III. PHYSICALLY INSPIRED LIGHTWEIGHT VIDEO SMOKE DETECTION ALGORITHM

In this section, we present the technical details of our proposed physically inspired lightweight video smoke detection algorithm. We first provide the overall framework and design philosophy of the proposed algorithm. Then we discuss the key modules of our proposed algorithm, which include the local binary patterns, dense optical flow, and MobileNetV2.

### A. OVERALL FRAMEWORK

The overall framework of the proposed smoke detection algorithm is illustrated in Fig. 2. The overall framework can be divided into two stages as shown at the top of the figure. In the first stage, we extract the texture, shape and motion information from two sequential frames of a given video and generate the corresponding LBP-motion image. In the second stage, we leverage the lightweight MobileNetV2 to classify if smoke is present in the LBP-motion image and the corresponding video frames.

The lower part of Fig. 2 shows the flowchart of feature extraction and LBP-motion image generation. First, two sequential frames from a given video are transformed into grayscale images. Note that the input two frames might not be adjacent. For slow moving object like smoke, the time interval between these two frames needs to be sufficiently large to capture the motion. The first grayscale image is then processed by the local binary patterns, yielding its LBP representation that retains the texture and shape information. Meanwhile, both of the grayscale images are smoothed by a Gaussian filter and then sent to the dense optical flow estimator for motion feature extraction. The estimated dense optical flow consists of an angle matrix and a magnitude

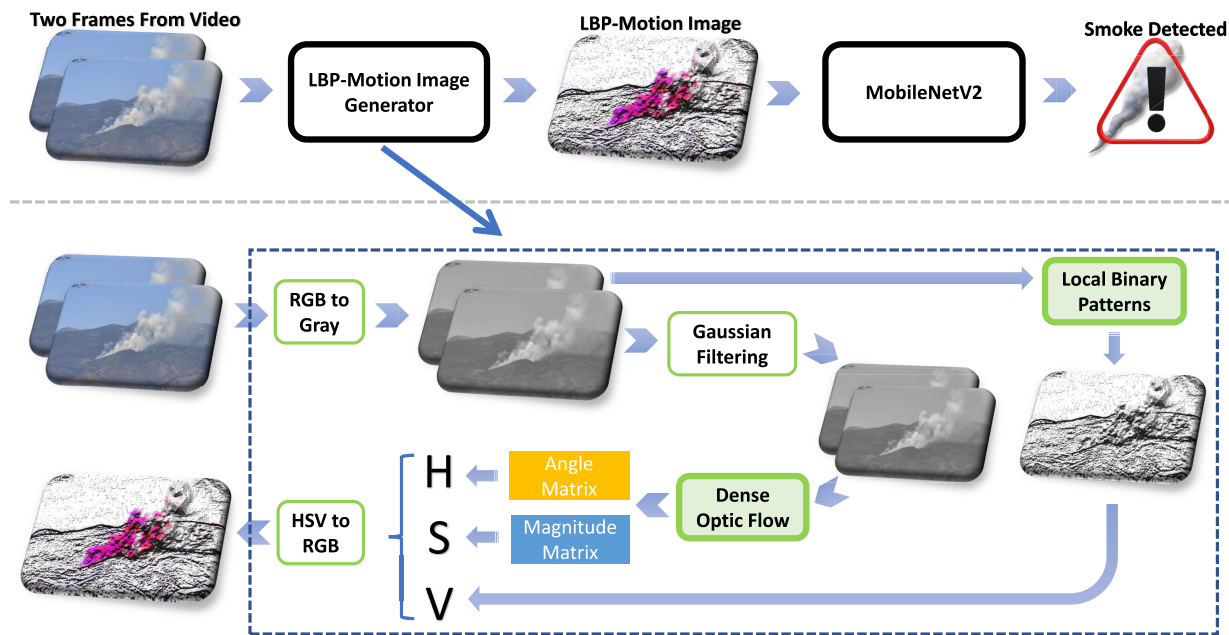


FIGURE 2. The overall framework of the proposed smoke detection algorithm.

matrix, describing the displacement of each pixel between the two frames. These two matrices along with the LBP representation are then scaled and combined to form hue, saturation, and value channels in the HSV color space, respectively. At last, we obtain the final LBP-motion image by converting this HSV representation into the RGB space.

Here we briefly discuss the design philosophy of the above mentioned feature selection algorithm. It is observed that humans rely on specific (diagnostic) object regions for accurate image recognition, which remain relatively consistent (invariant) across variations [72]. This observation motivates us to focus on diagnostic features that capture the inherent properties of wildfire smoke. As discussed in Section II-A, five features (color, motion, texture, shape, and energy) are often extracted for smoke detection. In this work, we only focus on the features that we deemed as diagnostic: texture, shape, and motion. These three features can be extracted through LBP-motion images that we construct. We ignore the energy feature since the wildfire cameras (usually PTZ cameras) might miss the initial stages of smoke. We also discard the color feature since it can not be treated as a diagnostic feature of wildfire smoke. This is because the colors of smoke in videos can vary significantly with the background environment, burning materials, and camera settings (see Fig. 3). It is also supported by the fact that we humans do not need much color information in finding wildfire smoke. For example, humans can easily tell if a wildfire smoke exists in a grayscale video even though there is no other color than black and white.

Note that the LBP-motion images generated by our proposed algorithm replaces color information of the original video frames with motion information. The color of each

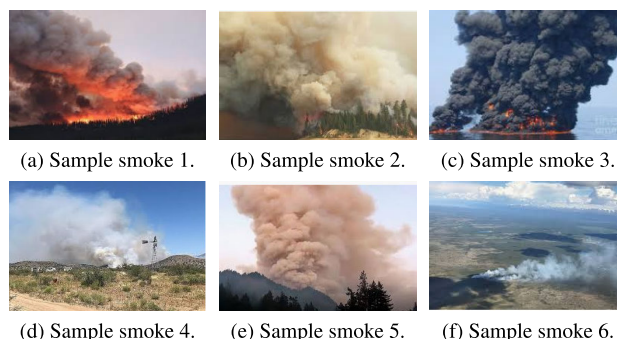


FIGURE 3. Sample smoke with different colors.

pixel in a LBP-motion image describes its moving direction and speed. It is worth noting that the LBP-motion image has the same dimension as the original video frames, thereby adding no additional complexity to image/video classification task in the second stage.

### B. KEY MODULES

There are three key modules in the proposed smoke detection framework: local binary patterns, dense optical flow, and MobileNetV2. We discuss them below individually.

#### 1) LOCAL BINARY PATTERNS

Local binary patterns were first proposed in [73] to measure the spatial structure of local image texture. The fundamental idea is to compare the gray value of a given pixel with that of its neighbors. The signs of differences are used to represent the local patterns. Despite the simple mechanism, LBP and

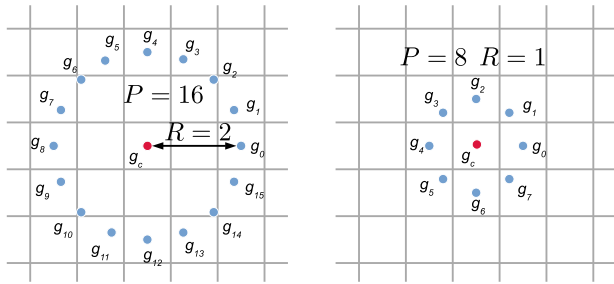


FIGURE 4. Two examples of a given central pixel and its neighbors with  $(P = 16, R = 2)$  and  $(P = 8, R = 1)$ .

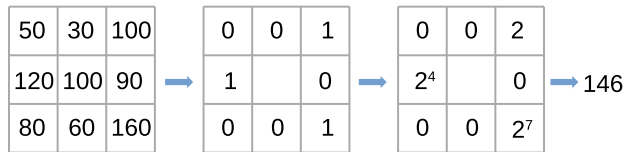


FIGURE 5. Example of LBP code calculation with  $P = 8$  and  $R = 1$ .

its variants are among the most successful texture descriptors in computer vision and pattern recognition applications.

Let  $LBP_{P,R}$  denote the LBP operator, where  $P$  represents the number of neighbors and  $R$  is the distance between the center point and each neighbor. See Fig. 4 for two examples of a given central pixel and its neighbors with  $(P = 16, R = 2)$  and  $(P = 8, R = 1)$ , respectively. The LBP code of a given pixel in an image is derived by [47]:

$$LBP_{P,R} = \sum_{p=0}^{P-1} s(g_p - g_c)2^p \quad (1)$$

where  $s(x) = \begin{cases} 1, & x \geq 0 \\ 0, & x < 0. \end{cases}$   $g_c$  is the gray value of the given pixel.  $g_p$  is the gray value of the pixel where neighbor  $p$  falls into. For each pixel in an image, we can apply the above operator to obtain its corresponding LBP code. This code describes the local texture information surrounding that pixel.

In this work, we use the conventional LBP operator where  $P = 8$  and  $R = 1$ . The LBP code of a pixel thereby ranges between 0 and 255. Fig. 5 shows an example of how the LBP operator calculates an LBP code with  $P = 8$  and  $R = 1$ . All the LBP codes together naturally form a new grayscale image that extracts both the texture and the shape information in the original image.

## 2) DENSE OPTICAL FLOW

Optical flow is the distribution of apparent velocities of movement of brightness patterns in an image [41]. It describes the motion of pixels in sequential video frames caused by the relative movement between camera and corresponding objects. The dense optical flow is defined as the optical flow for all the pixels in a given video frame. In this work, we estimate the dense optical flow by adopting the two-frame approach proposed in [74] due to its high computational efficiency and accuracy.

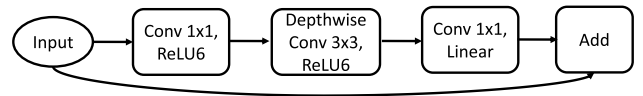


FIGURE 6. Bottleneck depth-separable convolution with residuals.

The fundamental idea of the adopted dense optical flow estimation method is briefly discussed below. We refer the readers to [74] and [75] for a detailed description of the technical method and algorithm. The gray values of a pixel and a certain neighborhood can be approximated by a quadratic polynomial function with respect to the coordinates  $\mathbf{x}$ :

$$f(\mathbf{x}) \sim \mathbf{x}^T \mathbf{A} \mathbf{x} + \mathbf{b}^T \mathbf{x} + c \quad (2)$$

where  $A$  is a symmetric matrix.  $\mathbf{b}$  is a vector of coefficients.  $c$  is a scalar. The above approximation process is called polynomial expansion. For a pixel with coordinates  $\mathbf{x}$ , we carry out polynomial expansion on two sequential video frames, giving us coefficients  $(A_1, \mathbf{b}_1, c_1)$  and  $(A_2, \mathbf{b}_2, c_2)$ . Let  $\mathbf{d}$  denote the displacement of this pixel between two frames. Then we have:

$$\begin{aligned} f_2(\mathbf{x}) &= f_1(\mathbf{x} - \mathbf{d}) \\ &= \mathbf{x}^T A_1 \mathbf{x} + (\mathbf{b}_1 - 2A_1 \mathbf{d})^T \mathbf{x} + \mathbf{d}^T A_1 \mathbf{d} - \mathbf{b}_1^T \mathbf{d} + c_1 \\ &= \mathbf{x}^T A_2 \mathbf{x} + \mathbf{b}_2^T \mathbf{x} + c_2 \end{aligned} \quad (3)$$

By matching the coefficients in the last two lines of the above equation, we obtain the following relationship:

$$2A_1 \mathbf{d} = -(\mathbf{b}_2 - \mathbf{b}_1) \quad (4)$$

Ideally,  $A_1$  and  $A_2$  should be the same. However, this is not true in practice. Therefore, Eq. (4) is reformulated as:

$$(A_1 + A_2) \mathbf{d} = -(\mathbf{b}_2 - \mathbf{b}_1) \quad (5)$$

The displacement can be estimated by solving Eq. (5). It is worth noting that the above discussion only covers the fundamental idea and basic procedures of the adopted dense optical flow estimation method. The full algorithm includes additional filtering processes, iterative computation, and multi-scale implementation.

## 3) MobileNetV2

MobileNetV2 is a deep convolutional neural network specifically designed for mobile and resource constrained environments [76]. MobileNetV2 has achieved remarkable success in embedded applications due to its lightweight and high efficiency in computation. In this paper, we adopt MobileNetV2 to detect smoke in the videos captured by wildfire cameras.

The basic building block of MobileNetV2 is a bottleneck depth-separable convolution with residuals as shown in Fig. 6. This bottleneck structure takes a low-dimensional compressed representation as an input, which is then transformed to high dimension and filtered by a depthwise separable convolution. The expanded features are then compressed back to low dimension through a linear convolution. This structure is motivated by the assumption that manifolds of interest can be embedded in low-dimensional subspaces.

TABLE 1. Structure of MobileNetV2.

Operator	Output channel #	Repeated times	Stride
conv2d	32	1	2
bottleneck	16	1	1
bottleneck	24	2	2
bottleneck	32	3	2
bottleneck	64	4	2
bottleneck	96	3	1
bottleneck	160	3	2
bottleneck	320	1	1
conv2d	1280	1	1
average pool	1280	1	-
fully connected	1	1	-

Compared to standard convolutional layers, the depthwise separable convolutions significantly reduces both computation and model size of MobileNetV2. For example, the computation of a  $3 \times 3$  depthwise separable convolution is usually 8 to 9 times less than that of its standard counterpart at only a small cost of accuracy [77].

Table 1 shows the full structure of MobileNetV2 employed in this work. The first layer is a 2D convolutional layer. The main structure of MobileNetV2 is constructed by stacking sequences of bottleneck blocks. A 2D convolutional layer, a global average pooling layer, and a fully connected output layer make up the rest of our MobileNetV2. Given smoke detection is a binary classification task, the output layer is thereby a fully connected layer with one neuron activated by sigmoid function, which is different from the original structure in [76]. We train this neural network through stochastic gradient decent based on Adam optimization algorithm [78]. The loss function to be minimized is the binary cross entropy as:

$$L(y, \hat{y}) = -y \log(\hat{y}) - (1 - y) \log(1 - \hat{y}) \quad (6)$$

where  $y \in \{0, 1\}$  is the true label of a given input.  $\hat{y}$  is the output of MobileNetV2.

We have discussed the structure of the proposed video-based smoke detection framework as well as the technical details of its modules in the above paragraphs. In the next section, we provide a detailed description of the specific procedures of our proposed optimal wildfire camera placement strategy, the goal of which is to maximize the fire risk reduction for a target area through wildfire camera deployment given a limited budget.

#### IV. OPTIMAL PLACEMENT OF WILDFIRE CAMERAS

In this section, we first provide a brief discussion about the background and the goal of wildfire camera placement planning. We then formulate the optimal placement of wildfire cameras as a binary integer program. This particular formulation falls into the category of set cover problem. The optimal or approximate solutions can be obtained through commercial optimization solvers.

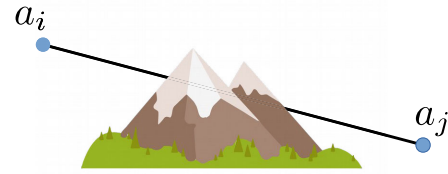


FIGURE 7. Example of occlusion.

We aim to find the optimal placement of wildfire cameras, which achieves the maximum fire risk reduction of the target area given limited budget. Suppose we are given a test region with a fire risk map. The fire risk of a sub-region can be reduced by a certain percentage if it can be closely monitored by one or more of the wildfire cameras. The magnitude of risk reduction depends on the effective monitoring range of the camera and the distance between the area being monitored and the location of the camera. Note that the costs of installing and maintaining wildfire cameras can vary significantly by location. The area covered by a wildfire camera depends on the elevation of its surrounding terrains.

Let  $A$  denote the given test region where wildfire cameras will be deployed. First, we discretize  $A$  into a square grid. Define  $\mathcal{S}^A = \{a_1, a_2, \dots, a_N\}$  as the set of all cells in the grid, where  $a_i$  represents the  $i$ th cell.  $N$  is the total number of cells. Note that the wildfire cameras can only be placed at suitable locations. Let  $\mathcal{S}^C = \{a_i | i \in \mathcal{P}^C\}$  be a subset of  $\mathcal{S}^A$ , where  $\mathcal{P}^C$  collects the indices of cells feasible for wildfire camera installation. Define  $x_i$  as the decision variable which equals 1 if a camera is placed at  $a_i$  and 0 otherwise. Define  $r_i$  as the fire risk of cell  $a_i$ . Then, we can minimize the overall fire risk of the region by solving the following optimization problem:

$$\min_{\{x_i | i \in \mathcal{P}^C\}} \sum_{i=1}^N r_i \cdot \max \left( r_{min}, 1 - \sum_{j \in \mathcal{P}^C} p_{ij} s_{ij} x_j \right) \quad (7)$$

subject to

$$\sum_{i \in \mathcal{P}^C} c_i x_i \leq B \quad (8)$$

$$x_i \in \{0, 1\}, \quad \forall i \in \mathcal{P}^C \quad (9)$$

where  $r_{min}$  denotes the minimum fire risk proportion.  $c_i$  represents the net present value of installation and maintenance cost if a camera is placed at  $a_i$ .  $B$  is the overall budget.  $s_{ij}$  is a variable that equals 1 if  $a_i$  can be effectively observed by a given camera from  $a_j$  and 0 otherwise. We treat  $a_i$  as invisible from a camera at  $a_j$  (i.e.,  $s_{ij} = 0$ ) if a straight line between them is intercepted by some other cells. See Fig. 7 for an example where occlusion exists.  $p_{ij}$  denotes the proportion of fire risk reduction for  $a_i$  if it can be monitored by the wildfire camera at  $a_j$ . Note that  $p_{ij}$  is highly related with the distance between the location of the camera  $a_i$  and the cell  $a_j$  being monitored and the camera's effective range. The objective function evaluates the overall fire risk of the region after the



wildfire camera deployment. Eq. (8) represents the budget constraint. Eq. (9) restrict  $x_i$  to be a binary variable.

The above optimization problem is a binary integer program that is NP-complete. This particular problem formulation can be categorized into the family of set cover problems. Global optimal solutions for small-scale problems and approximate solutions for large-scale problems can be obtained by commercial integer program solvers in reasonable time.

## V. CASE STUDIES

We conduct two case studies to validate our proposed video smoke detection framework and the wildfire camera placement strategy. The first case study uses real-world wildfire smoke videos to evaluate the performance of the proposed smoke detection algorithm. In the second case study, we demonstrate our proposed optimal wildfire camera placement strategy for a test area in the Riverside County of California.

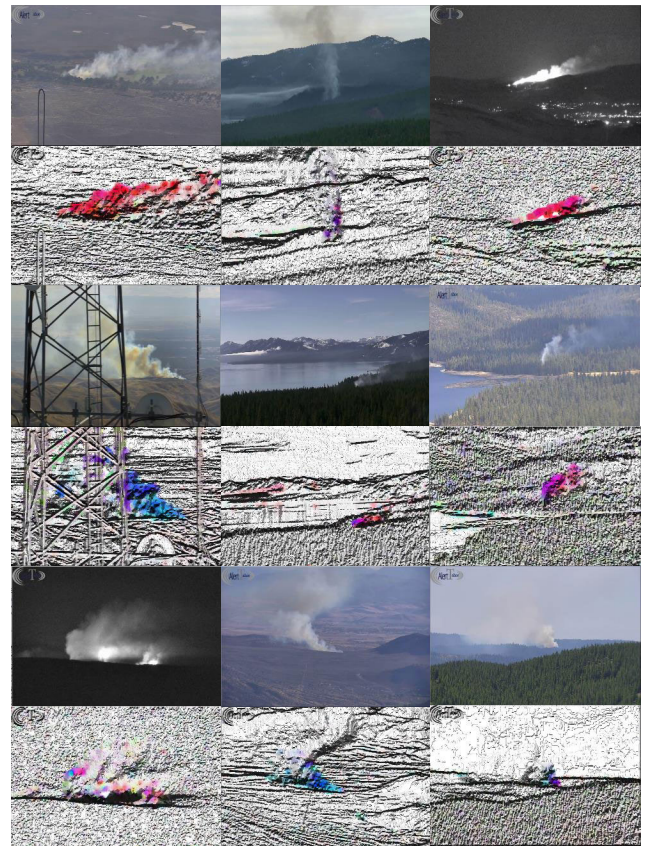
### A. SMOKE DETECTION WITH REAL-WORLD VIDEOS

In this subsection, we evaluate the performance of our proposed video smoke detection algorithm with real-world wildfire smoke videos. First, we briefly describe the dataset used in the case study. Then, we compare the performance of the proposed physics-based algorithm with that of a benchmark algorithm [67], which directly feeds the original video frames individually as inputs into a CNN such as MobileNetV2. This comparison will quantify the benefits of explicitly extracting diagnostic features and embedding them into the LBP-motion images. At last, we demonstrate the computational efficiency of the proposed framework in terms of computation time and memory usage.

#### 1) DATASET DESCRIPTION

The dataset includes 120 wildfire smoke videos downloaded from ALERTWildfire [79] and 120 non-smoke videos downloaded from YouTube. ALERTWildfire is a program, which places PTZ cameras to detect and monitor wildfires across California and its four bordering states. This camera network provides a wide coverage of high fire hazard regions and grew rapidly. The videos provided by ALERTWildfire are time-lapse videos with 60 times speed increase. The motion of smoke is thereby very discernible. We directly use adjacent frames to generate LBP-motion images from these videos. Fig. 8 shows several smoke video frames from ALERTWildfire and their corresponding LBP-motion images.

We divide the videos randomly into two groups, the training video group and the testing video group. The training video group consists of 96 smoke videos and 96 non-smoke videos. The testing video group collects the rest. All the videos are resized with a resolution of  $240 \times 180$ . For each video, we generate 75 LBP-motion images from the first 76 frames. Therefore, we have 14,400 and 3,600 LBP-motion images produced, from the training and testing video groups respectively, which make up the training and testing datasets for MobileNetV2.



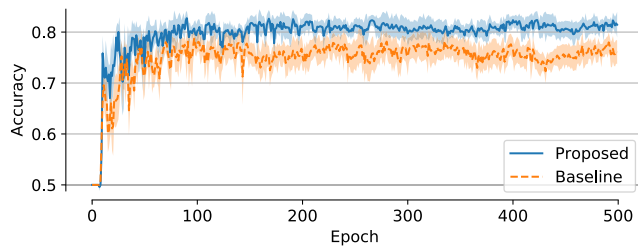
**FIGURE 8.** Sample smoke video frames from ALERTWildfire and their corresponding LBP-motion images.

#### 2) PERFORMANCE OF PHYSICS-BASED SMOKE DETECTION ALGORITHM

To verify the advantage of explicitly extracting physics-based features by generating LBP-motion images, we compare the performance of the proposed smoke detection framework with that of [67]. The major difference between the two approaches is that [67] directly takes individual frames from the original video as the inputs of MobileNetV2 while we proposed to generate LBP-motion images and use them as inputs.

We refer to the method proposed in [67] as the baseline approach hereafter. The training and testing datasets for the baseline approach are made up by the first 75 frames from the training and testing video groups, respectively. For both the proposed and the baseline approaches, we train the corresponding MobileNetV2s for 500 epochs. Each batch in an epoch consists of 32 images. The learning rate of Adam optimizer is 0.001. The weights of MobileNetV2 are randomly initialized. We evaluate the performance of intermediate trained models with the testing dataset after each training epoch.

We conduct training and testing on the same training and testing datasets for 10 times with different initial weights. The average out-of-sample performance with respect to the training epoch is shown in Fig. 9. The solid line and dashed line



**FIGURE 9.** Average test accuracy of the proposed and baseline approaches as the training sessions proceed.

**TABLE 2.** Performance comparison between the proposed approach and the baseline approach.

Approach	TPR	PPV	FPR
Proposed	85.31%	79.54%	22.42%
Baseline	78.80%	75.05%	26.74%

indicate the average test accuracy of the proposed approach and the baseline approach, respectively. The shaded areas describe their corresponding 95% confidence intervals. The proposed approach clearly outperforms the baseline approach on the testing dataset in terms of classification accuracy. At the end of our training session, the average test accuracies of the proposed approach and the baseline approach are 81.44% and 76.03%, respectively. Therefore, by explicitly extracting physics-based features by generating LBP-motion images, our proposed algorithm increases the classification accuracy by an average of 5.41%.

In addition to classification accuracy, we also use the following metrics to evaluate the performance of smoke detection algorithms. These metrics are true positive rate (TPR), positive predictive value (PPV), and false positive rate (FPR), which are defined as below:

$$TPR = \frac{TP}{TP + FN} \times 100\% \quad (10)$$

$$PPV = \frac{TP}{TP + FP} \times 100\% \quad (11)$$

$$FPR = \frac{FP}{FP + TN} \times 100\% \quad (12)$$

where  $TP$ ,  $TN$ ,  $FP$ , and  $FN$  are the numbers of true positives, true negatives, false positives, and false negatives, respectively.  $TPR$  reflects a smoke detector's sensitivity to smoke.  $PPV$  is also called precision that measures the credibility of a smoke detector.  $FPR$  describes the false alarm rate of a smoke detector. Similarly to accuracy, we calculate the average values of these metrics from tests on 10 trained models with different initial weights. The results are presented in Table 2. As shown in the table, on average, the proposed approach outperforms the baseline approach in terms of all these metrics.

*Remark:* The case study is based on videos gathered by ALERTWildfire project team. The camera itself costs about \$2,600 and is installed on top of an aluminum tower. The tower is accompanied with a solar panel and microwave

**TABLE 3.** Performance comparison between MobileNetV2 and benchmark CNNs.

Network	Accuracy	Computation	Memory
MobileNetV2	81.44%	0.117 ms	8.9 MiB
ResNet50	79.41%	0.237 ms	90.2 MiB
DenseNet169	82.44%	0.367 ms	49.1 MiB
InceptionV3	83.59%	0.216 ms	83.6 MiB
InceptionResNetV2	83.13%	0.492 ms	208.4 MiB

antennas to provide power supply and data transmission services. The average cost of the entire infrastructure is around \$75,000 and varies depending on the location. To implement our proposed smoke detection algorithm, we plan to first conduct periodic off-line training of the model using a GPU server with newly collected videos. Then the cameras need to be updated with small computing platforms such as Raspberry Pi. The trained models can then be distributed to each camera through wireless communication.

### 3) COMPARISON OF COMPUTATION TIME AND MEMORY USAGE

Here, we compare the computation time and memory usage of MobileNetV2 with that of other state-of-the-art deep CNNs. Specifically, four widely used deep CNNs are used in this comparison. They are ResNet50 [80], DenseNet169 [81], InceptionV3 [82], and InceptionResNetV2 [83]. Note that these four benchmark networks are incorporated into our wildfire smoke detection framework and replaces MobileNetV2.

We use the same training and testing datasets to evaluate the four benchmark networks. Similar to the evaluation process discussed above, we carry out 10 repetitions of training with random initial weights for each benchmark network. As before, one training session consists of 500 epochs. The performance of the benchmark networks are measured by the average test accuracy at the end of the training process. The mean computation time and memory usage for storing network weights are also reported. The mean computation time is calculated by averaging the computing time of 100 individual detection trials. Each detection trial consists of a LBP-motion image generation process and a forward propagation of the corresponding neural network. All the tests are executed through Python code and performed on a laptop with Intel i7-6600U CPU@2.60GHz. The weights of the CNNs are stored in 32-bit float format. The performance of the MobileNetV2 and the benchmark networks are shown in Table 3. It is evident that MobileNetV2 needs much less computational power and memory storage space than the other four benchmark networks. This significant saving in computation and memory resource only costs a slight reduction in accuracy compared with DenseNet169, InceptionV3, and InceptionResNetV2. MobileNetV2 even achieves a slightly higher average accuracy than ResNet50 on our datasets.

The case studies above demonstrate the effectiveness and the advantage of the proposed smoke detection framework.

Next, we show how to optimally place the cameras considering the fire risk and installation and maintenance cost of cameras. The goal is to minimize the risk of wildfire with a limited budget for camera network construction.

### B. OPTIMAL WILDFIRE CAMERA PLACEMENT

In this subsection, we conduct a case study for our proposed optimal wildfire camera placement strategy on a test region in Riverside County, California. First, we briefly describe the sources and backgrounds of three datasets used in the case study. Then, we detail the data preprocessing and the parameters settings. We close the case study by showing testing results from solving the optimization problem formulated in Section IV.

#### 1) DATASETS

We collect three datasets to set up the case study for the optimal wildfire camera placement problem. These datasets are the fire risk map, digital elevation model (DEM), and population density map, respectively.

The fire risk map used in this case study is cropped from the Wildfire Hazard Potential (WHP) map developed by USDA Forest Service [84]. WHP map is, by definition, a raster geospatial product that helps inform wildfire risk or prioritization of fuels management needs across very large spatial scales. Its specific objective is to quantify the relative potential for wildfire that would be difficult for suppression resources to contain. The original WHP map covers the whole conterminous United States at a 270-meter (around 1/6 mile) resolution. The WHP values are represented with integer numbers between 0 and 98,762.

The digital elevation model employed in this case study is derived from [85]. It records the elevation information across most parts of California at a 90-meter resolution. This DEM is generated from the Shuttle Radar Topography Mission (SRTM) datasets which are developed by NASA and other institutions using radar interferometry.

The population density map adopted in this case study is developed by the United States Geological Survey (USGS) [86]. This map is a raster dataset with resolution of 60-meters, which covers the whole conterminous United States. It was generated based on the census geography data collected from the US 2010 census. The record unit is number of people per square kilometer.

#### 2) DATA PREPROCESSING AND PARAMETER SETTINGS

The original fire risk map, DEM, and population density map described above are single band raster images with different resolutions and coverage. We first clip out the California part of each map and align them to the same coordinate system. We then downsample the DEM and population density map to make their resolutions the same as that of the fire risk map. The data for the test region are cropped from these preprocessed maps as shown in Fig. 10. Finally, we downsample the fire risk map, the DEM, and the population density map of the test region to a dimension of  $35 \times 40$  by averaging,

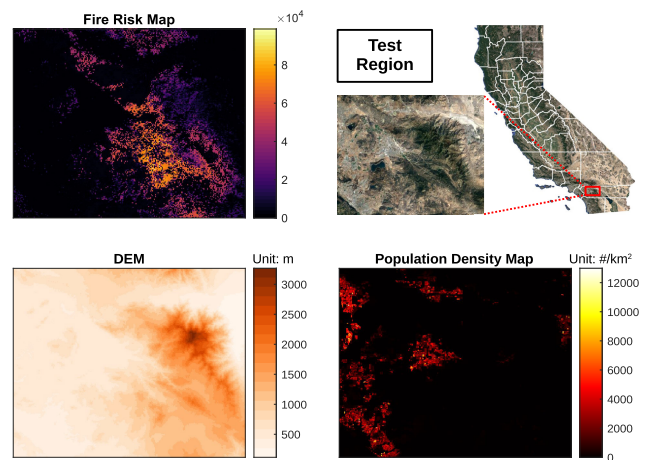


FIGURE 10. Test region and the corresponding fire risk map, DEM, and population density map.

which corresponds to a real-world area of  $35 \times 40$  mi<sup>2</sup>. We consider any cell that has an average elevation greater than 500 meters as a feasible cell for wildfire camera installation.

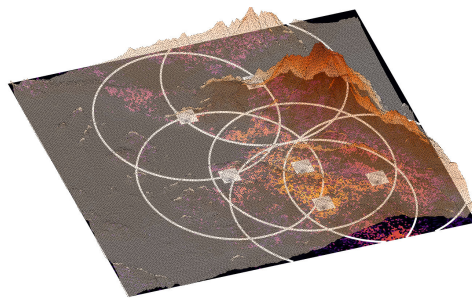
For the case study, we assume the total budget of deploying a wildfire camera network is  $B = \$500,000$ . The net present value of installation and maintenance cost of a wildfire camera placed at cell  $a_i$  is assumed to be:

$$c_i = 2500 + 75000 \times \frac{1}{\rho_i} + 20 \times \max(h_i - 1000, 0) \quad (13)$$

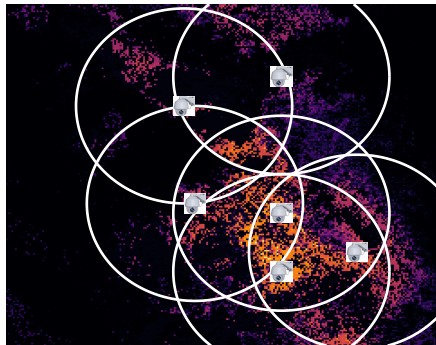
where  $h_i$  is the elevation of cell  $a_i$ .  $\rho_i$  is the population density value of cell  $a_i$  and it is min-max scaled between 1 and 2. \$2,500 represents the cost of a wildfire camera. \$75,000 accounts for the base cost of installation and maintenance for one wildfire camera. Note that the installation and maintenance of cameras at places with sparse population and high elevations are generally more expensive than that of regions with dense population and low altitudes. In the case study, the minimum fire risk proportion  $r_{min} = 0.1$ . We assume all the wildfire cameras are PTZ cameras with effective monitoring range of  $R = 10$  miles. The fire risk reduction proportion is calculated by  $p_{ij} = \frac{0.9}{1+d_{ij}/R}$ , where  $d_{ij}$  is the distance between cells  $a_i$  and  $a_j$ . Note that according to this fire risk reduction formulation, wildfire cameras are more effective in monitoring and reducing fire risks for the regions closer to them. This is because the classification accuracy of deep neural networks is inversely proportional to image blurriness [87]. Objects further away from the cameras correspond to blurrier representations on the frames.

#### 3) TESTING RESULTS

By solving the corresponding binary integer programming problem with Gurobi [88], we obtain the optimal wildfire camera deployment on the test region as shown in Fig. 11b. The background image is the fire risk map of the test region. Six wildfire cameras will be installed in the test region based on the optimal camera deployment plan. Each white circle



(a) Camera placement result with DEM.



(b) Camera placement result with fire risk map.

**FIGURE 11.** The optimal wildfire camera deployment on the test region.

indicates the effective monitoring range of the corresponding wildfire camera. Note that the actual coverage of a camera can be less than the area of the circle due to occlusion. To illustrate the terrains around the camera installation spots, we also present a 3D image displaying the DEM of the test region in Fig. 11a.

As shown in the figures, most areas with high fire risk are covered by the camera network. The resulting camera network deployment and maintenance cost is \$499,841 which is close to the budget limit. The overall fire risk of the test region is reduced to just 36.28% of its original value. In sum, this case study shows that our proposed optimal wildfire camera placement algorithm is able to significantly reduce the fire risk by strategically placing the wildfire cameras in a test region given a limited budget.

## VI. CONCLUSION

This paper proposes a lightweight physics-based video smoke detection framework and an optimal placement strategy for wildfire camera applications. The proposed smoke detection algorithm extracts useful features of videos and embeds them in LBP-motion images. This approach enables us to leverage lightweight deep convolutional neural network to accurately perform video smoke detection with limited computational and storage resources. The case studies with real-world wildfire smoke and non-smoke videos show that our proposed algorithm achieves similar accuracy to the state-of-the-art benchmarks while taking significantly less computational time and memory space.

We also propose an optimal wildfire camera placement strategy, which aims at minimizing the risk of wildfire of a target area with limited budget. The problem is formulated as a binary integer programming problem, which takes monitoring range of cameras, budget constraint, wildfire hazard potential, and object occlusion into consideration. We simulated our proposed wildfire camera placement strategy on a test region in Riverside Country. The case study results show that our proposed strategy helps find an optimal wildfire camera deployment plan, which achieves significant fire risk reduction.

## REFERENCES

- [1] The Sacramento Bee. *Gavin Newsom Wants Fire-Spotting Cameras in California Forests and A Lot More*. Accessed: Jan. 2019. [Online]. Available: <https://www.sacbee.com/news/california/fires/article224080720.html>
- [2] Government Technology. (Nov. 2018). *Wildfire Camera System Provides a Small Measure of Good News in California*. [Online]. Available: <https://www.govtech.com/em/preparedness/Wildfire-Camera-System-Provides-a-Small-Measure-of-Good-News-in-California-.html>
- [3] A. E. Çetin, K. Dimitropoulos, B. Gouverneur, N. Grammalidis, O. Günay, Y. H. Habiboglu, B. U. Töreyn, and S. Verstockt, "Video fire detection-review," *Digital Signal Processing*, vol. 23, no. 6, pp. 1827–1843, Dec. 2013.
- [4] J. S. de Vries and R. A. W. Kemp, "Surveillance sensor for autonomous wildfire detection," in *Proc. Surveill. Technol. Imag. Compon.*, vol. 1952, Nov. 1993, pp. 227–233.
- [5] J. S. de Vries and R. A. W. Kemp, "Results with a multispectral autonomous wildfire detection system," in *Proc. 20th Infr. Technol.*, Oct. 1994, pp. 18–28.
- [6] T.-H. Chen, P.-H. Wu, and Y.-C. Chiou, "An early fire-detection method based on image processing," in *Proc. Int. Conf. Image Process. (ICIP)*, vol. 3, Oct. 2004, pp. 1707–1710.
- [7] T.-H. Chen, Y.-H. Yin, S.-F. Huang, and Y.-T. Ye, "The smoke detection for early fire-alarming system base on video processing," in *Proc. Int. Conf. Intell. Inf. Hiding Multimedia*, Dec. 2006, pp. 427–430.
- [8] F. Yuan, "A fast accumulative motion orientation model based on integral image for video smoke detection," *Pattern Recognit. Lett.*, vol. 29, no. 7, pp. 925–932, May 2008.
- [9] T. Çelik, H. Özkaramanli, and H. Demirel, "Fire and smoke detection without sensors: Image processing based approach," in *Proc. 15th Eur. Signal Process. Conf.*, Sep. 2007, pp. 1794–1798.
- [10] B. U. Töreyn and A. E. Çetin, "Wildfire detection using LMS based active learning," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process.*, Apr. 2009, pp. 1461–1464.
- [11] Y. Chunyu, F. Jun, W. Jinjun, and Z. Yongming, "Video fire smoke detection using motion and color features," *Fire Technol.*, vol. 46, no. 3, pp. 651–663, Jul. 2010.
- [12] Y. H. Habiboglu, O. Gunay, and A. E. Çetin, "Real-time wildfire detection using correlation descriptors," in *Proc. 19th Eur. Signal Process. Conf.*, Aug. 2011, pp. 894–898.
- [13] A. Genovese, R. D. Labati, V. Piuri, and F. Scotti, "Wildfire smoke detection using computational intelligence techniques," in *Proc. IEEE Int. Conf. Comput. Intell. Meas. Syst. Appl. (CIMS) Proc.*, Sep. 2011, pp. 1–6.
- [14] O. Gunay, B. U. Töreyn, K. Kose, and A. E. Çetin, "Entropy-functional-based online adaptive decision fusion framework with application to wildfire detection in video," *IEEE Trans. Image Process.*, vol. 21, no. 5, pp. 2853–2865, May 2012.
- [15] R. Donida Labati, A. Genovese, V. Piuri, and F. Scotti, "Wildfire smoke detection using computational intelligence techniques enhanced with synthetic smoke plume generation," *IEEE Trans. Syst., Man, Cybern., Syst.*, vol. 43, no. 4, pp. 1003–1012, Jul. 2013.
- [16] Y. Jia, J. Yuan, J. Wang, J. Fang, Q. Zhang, and Y. Zhang, "A saliency-based method for early smoke detection in video sequences," *Fire Technol.*, vol. 52, no. 5, pp. 1271–1292, Sep. 2016.
- [17] Y. Wang, T. W. Chua, R. Chang, and N. T. Pham, "Real-time smoke detection using texture and color features," in *Proc. 21st Int. Conf. Pattern Recognition (ICPR)*, Nov. 2012, pp. 1727–1730.
- [18] B. U. Töreyn, Y. Dedeoğlu, and A. E. Çetin, "Wavelet based real-time smoke detection in video," in *Proc. 13th EUSIPCO*, Sep. 2005, pp. 1–4.

- [19] B. U. Töreyn, Y. Dedeoglu, and A. E. Çetin, "Contour based smoke detection in video using wavelets," in *Proc. 14th Eur. Signal Process. Conf.*, Sep. 2006, pp. 1–5.
- [20] P. Piccinini, S. Calderara, and R. Cucchiara, "Reliable smoke detection in the domains of image energy and color," in *Proc. 15th IEEE Int. Conf. Image Process.*, Oct. 2008, pp. 1376–1379.
- [21] S. Calderara, P. Piccinini, and R. Cucchiara, "Smoke detection in video surveillance: A MoG model in the wavelet domain," in *Proc. Int. Conf. Comput. Vis. Syst.*, May 2008, pp. 119–128.
- [22] S. Calderara, P. Piccinini, and R. Cucchiara, "Vision based smoke detection system using image energy and color information," *Mach. Vis. Appl.*, vol. 22, no. 4, pp. 705–719, Jul. 2011.
- [23] M. Cai, X. Lu, X. Wu, and Y. Feng, "Intelligent video analysis-based forest fires smoke detection algorithms," in *Proc. 12th Int. Conf. Natural Comput., Fuzzy Syst. Knowl. Discovery (ICNC-FSKD)*, Aug. 2016, pp. 1504–1508.
- [24] F. Yuan, "A double mapping framework for extraction of shape-invariant features based on multi-scale partitions with AdaBoost for video smoke detection," *Pattern Recognit.*, vol. 45, no. 12, pp. 4326–4336, Dec. 2012.
- [25] C.-C. Ho, "Machine vision-based real-time early flame and smoke detection," *Meas. Sci. Technol.*, vol. 20, no. 4, Apr. 2009, Art. no. 045502.
- [26] B. Ko, J. Park, and J.-Y. Nam, "Spatiotemporal bag-of-features for early wildfire smoke detection," *Image Vis. Comput.*, vol. 31, no. 10, pp. 786–795, Oct. 2013.
- [27] A. Filonenko, D. C. Hernandez, and K.-H. Jo, "Fast smoke detection for video surveillance using CUDA," *IEEE Trans. Ind. Informat.*, vol. 14, no. 2, pp. 725–733, Feb. 2018.
- [28] T. X. Tung and J.-M. Kim, "An effective four-stage smoke-detection algorithm using video images for early fire-alarm systems," *Fire Saf. J.*, vol. 46, no. 5, pp. 276–282, Jul. 2011.
- [29] C.-Y. Lee, C.-T. Lin, C.-T. Hong, and M.-T. Su, "Smoke detection using spatial and temporal analyses," *Int. J. Innov. Comput., Inf. Control*, vol. 8, no. 6, pp. 1–11, Jun. 2012.
- [30] P. Guillemant, "Real-time identification of smoke images by clustering motions on a fractal curve with a temporal embedding method," *Opt. Eng.*, vol. 40, no. 4, pp. 554–564, Apr. 2001.
- [31] Z. Xu and J. Xu, "Automatic fire smoke detection based on image visual features," in *Proc. Int. Conf. Comput. Intell. Secur. Workshops (CISW)*, Dec. 2007, pp. 316–319.
- [32] D. Kim and Y.-F. Wang, "Smoke detection in video," in *Proc. WRI World Congr. Comput. Sci. Inf. Eng.*, vol. 5, Mar. 2009, pp. 759–763.
- [33] Y. Chunyu, Z. Yongming, F. Jun, and W. Jinjun, "Texture analysis of smoke for real-time fire detection," in *Proc. 2nd Int. Workshop Comput. Sci. Eng.*, vol. 2, Oct. 2009, pp. 511–515.
- [34] N. Alamgir, K. Nguyen, V. Chandran, and W. Boles, "Combining multi-channel color space with local binary co-occurrence feature descriptors for accurate smoke detection from surveillance videos," *Fire Saf. J.*, vol. 102, pp. 1–10, Dec. 2018.
- [35] X. Wu, X. Lu, and H. Leung, "Video smoke separation and detection via sparse representation," *Neurocomputing*, vol. 360, pp. 61–74, Sep. 2019.
- [36] Y. Zhao, Z. Zhou, and M. Xu, "Forest fire smoke video detection using spatiotemporal and dynamic texture features," *J. Electr. Comput. Eng.*, vol. 2015, pp. 1–7, Jan. 2015.
- [37] K. Dimitropoulos, P. Barmoutis, and N. Grammalidis, "Higher order linear dynamical systems for smoke detection in video surveillance applications," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 27, no. 5, pp. 1143–1154, May 2017.
- [38] H. Maruta, A. Nakamura, and F. Kurokawa, "A new approach for smoke detection with texture analysis and support vector machine," in *Proc. IEEE Int. Symp. Ind. Electron.*, Jul. 2010, pp. 1550–1555.
- [39] X. Wu, X. Lu, and H. Leung, "A motion and lightness saliency approach for forest smoke segmentation and detection," *Multimedia Tools Appl.*, vol. 79, nos. 1–2, pp. 69–88, Jan. 2020.
- [40] Z. Liu, X. Yang, Y. Liu, and Z. Qian, "Smoke-detection framework for high-definition video using fused Spatial- and frequency-domain features," *IEEE Access*, vol. 7, pp. 89687–89701, 2019.
- [41] B. K. P. Horn and B. G. Schunck, "Determining optical flow," *Artif. Intell.*, vol. 17, nos. 1–3, pp. 185–203, Aug. 1981.
- [42] F. Gomez-Rodriguez, B. C. Arrue, and A. Ollero, "Smoke monitoring and measurement using image processing: Application to forest fires," in *Proc. 13th Autom. Target Recognit.*, vol. 5094, Sep. 2003, pp. 404–412.
- [43] I. Kolesov, P. Karasev, A. Tannenbaum, and E. Haber, "Fire and smoke detection in video with optimal mass transport based optical flow and neural networks," in *Proc. IEEE Int. Conf. Image Process.*, Sep. 2010, pp. 761–764.
- [44] S. Wang, Y. He, J. J. Zou, D. Zhou, and J. Wang, "Early smoke detection in video using swaying and diffusion feature," *J. Intell. Fuzzy Syst.*, vol. 26, no. 1, pp. 267–275, Jan. 2014.
- [45] C. C. Gotlieb and H. E. Kreyszig, "Texture descriptors based on co-occurrence matrices," *Comput. Vis., Graph., Image Process.*, vol. 51, no. 1, pp. 70–86, Jul. 1990.
- [46] Y. Cui, H. Dong, and E. Zhou, "An early fire detection method based on smoke texture analysis and discrimination," in *Proc. Congr. Image Signal Process.*, vol. 3, May 2008, pp. 95–99.
- [47] T. Ojala, M. Pietikainen, and T. Maenpaa, "Multiresolution gray-scale and rotation invariant texture classification with local binary patterns," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 24, no. 7, pp. 971–987, Jul. 2002.
- [48] F. Yuan, "Video-based smoke detection with histogram sequence of LBP and LBPV pyramids," *Fire Saf. J.*, vol. 46, no. 3, pp. 132–139, Apr. 2011.
- [49] F. Yuan, J. Shi, X. Xia, Y. Fang, Z. Fang, and T. Mei, "High-order local ternary patterns with locality preserving projection for smoke detection and image classification," *Inf. Sci.*, vol. 372, pp. 225–240, Dec. 2016.
- [50] F. Yuan, J. Shi, X. Xia, L. Zhang, and S. Li, "Encoding pairwise Hamming distances of local binary patterns for visual smoke recognition," *Comput. Vis. Image Understand.*, vol. 178, pp. 43–53, Jan. 2019.
- [51] M. Favorskaya, A. Pyataeva, and A. Popov, "Verification of smoke detection in video sequences based on spatio-temporal local binary patterns," *Procedia Comput. Sci.*, vol. 60, pp. 671–680, Jan. 2015.
- [52] J. Gubbi, S. Marusic, and M. Palaniswami, "Smoke detection in video using wavelets and support vector machines," *Fire Saf. J.*, vol. 44, no. 8, pp. 1110–1115, Nov. 2009.
- [53] F. Yuan, S. Wu, Y. Yang, Z. Fang, and Y. Fang, "Real-time image smoke detection using staircase searching-based dual threshold AdaBoost and dynamic analysis," *IET Image Process.*, vol. 9, no. 10, pp. 849–856, Oct. 2015.
- [54] J. Vicente and P. Guillemant, "An image processing technique for automatically detecting forest fire," *Int. J. Thermal Sci.*, vol. 41, no. 12, pp. 1113–1120, Dec. 2002.
- [55] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, "ImageNet: A large-scale hierarchical image database," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2009, pp. 248–255.
- [56] C. Tao, J. Zhang, and P. Wang, "Smoke detection based on deep convolutional neural networks," in *Proc. Int. Conf. Ind. Informat. Comput. Technol., Intell. Technol., Ind. Inf. Integr. (ICI3I)*, Dec. 2016, pp. 150–153.
- [57] Q. Zhang, J. Xu, L. Xu, and H. Guo, "Deep convolutional neural networks for forest fire detection," in *Proc. Int. Forum Manage., Edu. Inf. Technol. Appl.*, Jan. 2016, pp. 568–575.
- [58] S. Frizzi, R. Kaabi, M. Bouchouicha, J.-M. Ginoux, E. Moreau, and F. Fnaiech, "Convolutional neural network for video fire and smoke detection," in *Proc. 42nd Annu. Conf. IEEE Ind. Electron. Soc. (IECON)*, Oct. 2016, pp. 877–882.
- [59] Z. Yin, B. Wan, F. Yuan, X. Xia, and J. Shi, "A deep normalization and convolutional neural network for image smoke detection," *IEEE Access*, vol. 5, pp. 18429–18438, 2017.
- [60] Q.-X. Zhang, G.-H. Lin, Y.-M. Zhang, G. Xu, and J.-J. Wang, "Wildland forest fire smoke detection based on faster R-CNN using synthetic smoke images," *Procedia Eng.*, vol. 211, pp. 441–446, Jan. 2018.
- [61] S. Ren, K. He, R. Girshick, and J. Sun, "Faster R-CNN: Towards real-time object detection with region proposal networks," in *Proc. Adv. Neural Inf. Process. Syst.*, Dec. 2015, pp. 91–99.
- [62] G. Xu, Y. Zhang, Q. Zhang, G. Lin, and J. Wang, "Deep domain adaptation based video smoke detection using synthetic smoke images," *Fire Saf. J.*, vol. 93, pp. 53–59, Oct. 2017.
- [63] G. Xu, Q. Zhang, D. Liu, G. Lin, J. Wang, and Y. Zhang, "Adversarial adaptation from synthesis to reality in fast detector for smoke detection," *IEEE Access*, vol. 7, pp. 29471–29483, 2019.
- [64] S. Cheng and J. Ma, "Smoke detection and trend prediction method based on Deeplabv3+ and generative adversarial network," *J. Electron. Imag.*, vol. 28, no. 3, May 2019, Art. no. 033006.
- [65] L.-C. Chen, Y. Zhu, G. Papandreou, F. Schroff, and H. Adam, "Encoder-decoder with atrous separable convolution for semantic image segmentation," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, Sep. 2018, pp. 801–818.
- [66] M. Yin, C. Lang, Z. Li, S. Feng, and T. Wang, "Recurrent convolutional network for video-based smoke detection," *Multimedia Tools Appl.*, vol. 78, no. 1, pp. 237–256, Jan. 2019.

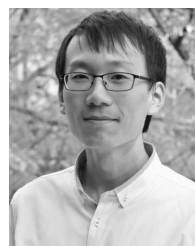
- [67] K. Muhammad, S. Khan, V. Palade, I. Mehmood, and V. H. C. de Albuquerque, "Edge intelligence-assisted smoke detection in foggy surveillance environments," *IEEE Trans. Ind. Informat.*, vol. 16, no. 2, pp. 1067–1075, Feb. 2020.
- [68] G. Lin, Y. Zhang, G. Xu, and Q. Zhang, "Smoke detection on video sequences using 3D convolutional neural networks," *Fire Technol.*, vol. 55, no. 5, pp. 1827–1847, Sep. 2019.
- [69] S. Aslan, U. Gudukbay, B. U. Toreyin, and A. E. Cetin, "Early wildfire smoke detection based on motion-based geometric image transformation and deep convolutional generative adversarial networks," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process. (ICASSP)*, May 2019, pp. 8315–8319.
- [70] Z. Yang, W. Shi, Z. Huang, Z. Yin, F. Yang, and M. Wang, "Combining Gaussian mixture model and HSV model with deep convolution neural network for detecting smoke in videos," in *Proc. IEEE 18th Int. Conf. Commun. Technol. (ICCT)*, Oct. 2018, pp. 1266–1270.
- [71] J. Redmon and A. Farhadi, "YOLO9000: Better, faster, stronger," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 7263–7271.
- [72] H. Karimi-Rouzbahani, N. Bagheri, and R. Ebrahimpour, "Invariant object recognition is a personalized selection of invariant features in humans, not simply explained by hierarchical feed-forward vision models," *Sci. Rep.*, vol. 7, no. 1, p. 14402, Dec. 2017.
- [73] T. Ojala, M. Pietikainen, and D. Harwood, "Performance evaluation of texture measures with classification based on kullback discrimination of distributions," in *Proc. 12th Int. Conf. Pattern Recognit.*, vol. 1, Oct. 1994, pp. 582–585.
- [74] G. Farneback, "Two-frame motion estimation based on polynomial expansion," in *Proc. Scand. Conf. Image Anal.* Berlin, Germany: Springer, Jun. 2003, pp. 363–370.
- [75] G. Farneback, "Polynomial expansion for orientation and motion estimation," Ph.D. dissertation, Linköping Univ. Electron. Press, Linköping, Sweden, 2002.
- [76] M. Sandler, A. Howard, M. Zhu, A. Zhmoginov, and L.-C. Chen, "MobileNetV2: Inverted residuals and linear bottlenecks," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 4510–4520.
- [77] A. G. Howard, M. Zhu, B. Chen, D. Kalenichenko, W. Wang, T. Weyand, M. Andreetto, and H. Adam, "MobileNets: Efficient convolutional neural networks for mobile vision applications," Apr. 2017, *arXiv:1704.04861*. [Online]. Available: <http://arxiv.org/abs/1704.04861>
- [78] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," Dec. 2014, *arXiv:1412.6980*. [Online]. Available: <http://arxiv.org/abs/1412.6980>
- [79] (2019). *ALERTWildfire*. [Online]. Available: <http://www.alertwildfire.org/>
- [80] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 770–778.
- [81] G. Huang, Z. Liu, L. Van Der Maaten, and K. Q. Weinberger, "Densely connected convolutional networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 4700–4708.
- [82] C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens, and Z. Wojna, "Rethinking the inception architecture for computer vision," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 2818–2826.
- [83] C. Szegedy, S. Ioffe, V. Vanhoucke, and A. A. Alemi, "Inception-v4, inception-ResNet and the impact of residual connections on learning," in *Proc. 31st AAAI Conf. Artif. Intell.*, Feb. 2017, pp. 4278–4284.
- [84] G. K. Dillon. (Dec. 2018). *Wildfire Hazard Potential (WHP) for the Conterminous United States (270-m GRID), Version 2018 Classified (2nd Edition)*. USDA Forest Service. [Online]. Available: <https://www.fs.usda.gov/rds/archive/catalog/RDS-2015-0046-2>
- [85] (2000). *90 m DEM Data Basin*. [Online]. Available: <https://databasin.org/datasets/78ac54fabd594db5a39f6629514752c0>
- [86] (2016). *U.S. Block-Level Population Density Rasters for 1990, 2000, and 2010*. U.S. Geological Survey. [Online]. Available: <https://www.sciencebase.gov/catalog/item/57753ebee4b07dd077c70868>
- [87] S. Dodge and L. Karam, "Understanding how image quality affects deep neural networks," in *Proc. 8th Int. Conf. Qual. Multimedia Exper. (QoMEX)*, Jun. 2016, pp. 1–6.
- [88] (2019). *Gurobi Optimizer Reference Manual*. Gurobi Optimization. [Online]. Available: <http://www.gurobi.com>



**JIE SHI** (Graduate Student Member, IEEE) received the B.S. degree in automation from the Shenyang University of Technology, Shenyang, China, in 2012, and the M.S. degree in control theory and engineering from Southeast University, Nanjing, China, in 2015. He is currently pursuing the Ph.D. degree in electrical and computer engineering with the University of California at Riverside, Riverside, CA, USA. His research interests include intelligent systems, machine learning, and smart city.



**WEI WANG** (Graduate Student Member, IEEE) received the B.S. degree in electrical engineering from the Huazhong University of Science and Technology, Wuhan, China, and the M.S.E. degree in electrical and computer engineering from the University of Michigan, Ann Arbor, MI, USA, in 2012 and 2014, respectively. He is currently pursuing the Ph.D. degree in electrical and computer engineering with the University of California at Riverside, Riverside, CA, USA. His research interests include power system optimization and big data analytics in smart grids.



**YUANQI GAO** (Graduate Student Member, IEEE) received the B.E. degree in electrical engineering from Donghua University, Shanghai, China, in 2015. He is currently pursuing the Ph.D. degree in electrical and computer engineering with the University of California at Riverside, Riverside, CA, USA. His research interests include big data analytics in smart grids and machine learning with application to power distribution systems.



**NANPENG YU** (Senior Member, IEEE) received the B.S. degree in electrical engineering from Tsinghua University, Beijing, China, in 2006, and the M.S. degree and the Ph.D. degree in electrical engineering from Iowa State University, Ames, IA, USA, in 2007 and 2010, respectively. He is currently an Associate Professor with the Department of Electrical and Computer Engineering, University of California at Riverside, Riverside, CA, USA. His current research interests include electricity market design and optimization, machine learning in smart grids, and smart energy communities. He is an Editor of the *IEEE TRANSACTIONS ON SMART GRID* and *International Transactions on Electrical Energy Systems*.

• • •