applied sciences

Article

A Video-Based Fire Detection Using Deep Learning Models

Byoungjun Kim and Joonwhoan Lee *

Division of Computer Science and Engineering, Chonbuk National University, Jeonju 54896, Korea
* Correspondence: chlee@chonbuk.ac.kr

Received: 14 June 2019; Accepted: 15 July 2019; Published: 18 July 2019

Abstract: Fire is an abnormal event which can cause significant damage to lives and property. In this paper, we propose a deep learning-based fire detection method using a video sequence, which imitates the human fire detection process. The proposed method uses Faster Region-based Convolutional Neural Network (R-CNN) to detect the suspected regions of fire (SRoFs) and of non-fire based on their spatial features. Then, the summarized features within the bounding boxes in successive frames are accumulated by Long Short-Term Memory (LSTM) to classify whether there is a fire or not in a short-term period. The decisions for successive short-term periods are then combined in the majority voting for the final decision in a long-term period. In addition, the areas of both flame and smoke are calculated and their temporal changes are reported to interpret the dynamic fire behavior with the final fire decision. Experiments show that the proposed long-term video-based method can successfully improve the fire detection accuracy compared with the still image-based or short-term video-based method by reducing both the false detections and the misdetections.

Keywords: deep learning; fire detection; Faster R-CNN; spatiotemporal feature; LSTM; majority voting; dynamic fire behavior

1. Introduction

Fire is an abnormal event which can quickly cause significant injury and property damage [1]. According to the National Fire Protection Association (NAPA), the United States fire department responded to an estimated 1,319,500 fires during 2017 [2], which resulted in 3,400 civilian fire fatalities, 14,670 civilian fire injuries, and an estimated $23 billion in direct property loss. In order to reduce such disasters, fire detection without a false alarm at an early stage is crucial. Accordingly, various automatic fire detection technologies are being developed, and are widely used in real life.

In general, two broad categories of technologies can be identified: traditional fire alarm and fire detection by computer vision. Traditional fire alarm technology is based on smoke or heat sensors that require proximity for activation. These sensors need human involvement to confirm a fire in case of alarm. Furthermore, such systems require various equipment to provide information on the size, location, and burning degree of the fire. To overcome these limitations, researchers have been investigating computer vision-based methods combined with various types of supplementary sensors [3–6]. This category of technologies gives larger surveillance coverage and offers the advantage of less human intervention with a faster response, as a fire can be confirmed without requiring a visit to the fire location, and provides detailed fire information such as location, size, and degree. Despite these advantages, however, some issues remain concerning the system complexity, and false detection according to diverse reasons. Therefore, researchers have invested significant effort to address these issues in terms of computer vision technology.

Early research on computer vision-based fire detection was focused on the color of a fire within the framework of a rule-based system, which is often sensitive to environmental conditions such as illumination and weather. So, further studies added supplementary features to the color of a fire, including area, surface, boundary, and motion of the suspected region, with other types of decision-making algorithms, such as Bayes classifier and multi-expert systems, in order to make a robust decision. Nevertheless, almost all the research tries to detect the flame and smoke in a single frame of Closed-Circuit Television (CCTV) or a limited number of frames in a short period.

In general, it is not an easy task to explore the static and dynamic characteristics of diverse flame and smoke to be exploited in a vision system, as it requires a large amount of domain knowledge. In the deep learning approach however, these exploration and exploitation processes can be replaced by the training of an appropriate neural network with a sufficient amount of data in order to avoid overfitting. This approach, therefore, becomes convenient once a dataset with many flame and smoke images or video clips has been built.

In this paper, we propose a deep-learning-based fire detection method, which imitates the human detection process and which we call the detection and temporal accumulations (DTA) for the fire decision. Usually, a human can detect a suspected fire object in a scene, continuously monitor it, and accumulate the temporal behaviors to finally decide whether it is a fire or not. We assume that this DTA process can greatly reduce erroneous fire detection.

In the proposed method, the suspected region of fire (SRoF) is detected with its spatial features against non-fire objects by the Faster Region-based Convolutional Neural Network (R-CNN). Then, the features summarized from the object detection model in successive frames are accumulated by Long Short-Term Memory (LSTM) to classify whether there is a fire or not in a short-term period, which can be treated as a person's glance for fire detection.

The decisions for successive short-term periods are then combined in the majority voting for the final decision in a long-term period. In addition, the areas of SRoFs, including both flame and smoke, are calculated and their temporal changes are reported to interpret the dynamic fire behavior with the final fire decision.

Experiments show that the proposed long-term video-based method can successfully improve the fire detection accuracy compared with the still image-based or short-term video-based method by reducing both the false detections and the misdetections. The method discriminates fires from fire-like video sequences especially well. For example, chimney smoke, sunset, and clouds often induce errors in conventional computer vision-based fire detection. Also, the method reflects the temporal behavior of real fire situations well by providing area information of SRoFs.

Therefore, our key contributions can be summarized as follows:

(1) We propose a deep learning-based fire detection method that avoids the time-consuming efforts to explore hand-crafted features. Because it automatically generates a set of useful features after training, it is sufficient to construct the proper deep learning model and to gather a sufficient amount of training data. Therefore, we have constructed a large fire dataset which contains diverse still images and video clips, including the data from well-known public datasets. Not only is the dataset used for the training and testing of our experiment, but it also could be an asset for future computer vision-based fire detection research.

(2) Our deep learning-based method emulates a human process of fire detection called DTA, in that SRoFs are detected in one scene and the temporal behaviors are continuously monitored and accumulated to finally decide whether it is a fire or not. In the method, Faster R-CNN is used to detect SRoFs against non-fire objects with their spatial features, and LSTM temporally accumulates the summarized spatial features by using the weighted Global Average Pooling (GAP), where the weight is given by the confidence score of a bounding box. The initial decision is made in a short period, and the final decision is made by the majority voting of the series of decisions in a long period.

(3) The proposed method has been experimentally proven to provide excellent fire detection accuracy by reducing the false detections and misdetections. Also, it successfully interprets the temporal SRoF behavior, which may reduce false dispatch of firemen.

The remainder of this paper is organized as follows. The related work is introduced in Section 2, the details of our proposed method are given in Section 3, and the experimental results and discussions are presented in Section 4. Finally, several concluding remarks are given in Section 5.

2. Related Work

2.1. Computer Vision-Based Fire Detection

In conventional fire detection, much research has continuously focused on finding out the salient features of fire images. Chen [7] analyzed the changes of fire using an RGB and HSI color model based on the difference between consecutive frames and proposed a rule-based approach for fire decision. Celik and Demirel [5] proposed a generic rule-based flame pixel classification using the YCbCr color model to separate chrominance components from luminance ones. In addition, Wang [8] extracted the candidate fire area in an image using an HSI color model and calculated the dispersion of the flame color to determine the fire area. However, color-based fire detection methods are generally vulnerable to a variety of environmental factors such as lighting and shadow.

Borges and Izquierdo [9] adopted the Bayes classifier to detect fires based on additional features such as the area, surface, and boundary of the fire area to color. Mueller [10] proposed the neural network-based fire detection method using optical flow for the fire area. In the method, two optical flow models are combined to distinguish between fire and dynamically moving objects. In addition, Foggia [11] proposed a multi-expert system which combines the analysis results of a fire's color, shape, and motion characteristics. Although insufficient, the supplementary features to color, including texture, shape, and optical flow, can reduce the false detections.

Nevertheless, these approaches require domain knowledge of fires in captured images essential to explore hand-crafted features and cannot reflect the information spatially and temporally involved in fire environments well. In addition, almost all methods using the conventional approach only use a still image or consecutive pairs of frames to detect fire. Therefore, they only consider the short-term dynamic behavior of fire, whereas a fire has a longer-term dynamic behavior.

2.2. Deep Learning-Based Approach

Recently, deep learning has been successfully applied to diverse areas such as object detection/classification in images, speech recognition, and natural language processing. Researchers have conducted various studies on fire detection based on deep learning to improve performance.

The deep learning approach has several differences from the conventional computer vision-based fire detection. The first is that the features are not explored by an expert, but rather are automatically captured in the network after training with a large amount of diverse training data. Therefore, the effort to find the proper handcrafted features is shifted to designing a proper network and preparing the training data.

Another difference is that the detector/classifier can be obtained by training simultaneously with the features in the same neural network. Therefore, the appropriate network structure becomes more important with an efficient training algorithm.

Sebastien [12] proposed a fire detection network based on CNN where the features are simultaneously learned with a Multilayer Perceptron (MLP)-type neural net classifier by training. Zhang et al. [13] also proposed a CNN-based fire detection method which is operated in a cascaded fashion. In their method, the full image is first tested by the global image-level classifier, and if a fire is detected, then a fine-grained patch classifier is used for precisely localizing the fire patches. Muhammad et al. [14] proposed a fire surveillance system based on a fine-tuned CNN fire detector. This architecture is an efficient CNN architecture for fire detection, localization, and semantic understanding of the scene of the fire inspired by the Squeeze Net [15] architecture.

In the deep layer of CNN, a unit has a wide receptive field so that its activation can be treated as a feature that contains a large area of context information. This is another advantage of the learned features with CNN for fire detection.

Even though CNN showed overwhelmingly superior classification performance against traditional computer vision methods, locating objects has been another problem. In the proposed method, we adopt the object detection model to localize the SRoFs and non-fire objects, which includes the flame, smoke for the SRoFs, and other objects irrelevant to the fire for the non-fire objects. The objects irrelevant to the fire increase false alarms due to variations in shadows and brightness, and will often detect objects such as red clothes, red vehicles, or sunset. We detect the fire objects by using the Faster R-CNN model, even though it does not have to be confined to the object detection model. The deep object detector, either single- or multi-stage, is usually composed of CNN-type feature extractors, followed by a localizer with a classifier. Therefore, our object detection model includes the feature extractor with a relatively wider area of receptive field than the detected SRoF area and can gather more context information.

Although the CNN-based approaches provide excellent performance, it is hard to capture the dynamic behavior of fire, which can be obtained by recursive-type neural networks (RNN). LSTM proposed by Hochreiter and Schmidhuber [16] is an RNN model that solves the vanishing gradient problem of RNN. LSTM can accumulate the temporal features for decision making through the memory cells which preserve the internal states and the recurrent behavior. However, the number of recursions is usually limited, which makes it difficult to capture the long-term dynamic behavior necessary to make a decision. Therefore, special care must be taken to consider the decision based on long-term behavior with LSTM.

Recently, Hu et al. [17] used LSTM for fire detection, where the CNN features are extracted from optical flows of consecutive frames, and temporally accumulated in an LSTM network. The final decision is made based on the fusion of successive temporal features. Their approach, however, computes the optical flow to prepare the input of CNN rather than directly using RGB frames.

3. Proposed Method

3.1. Network Architecture

Traditional computer vision-based fire detection methods have widely used the static characteristics or the short-term temporal behaviors such as colors and motions of flame and smoke. However, as fires show variable temporal appearance, the detection accuracy of such methods that depend on the static and short-term temporal behaviors is limited.

We propose a deep learning-based fire detection method, which imitates the human process, called DTA for fire decision. We assume that this DTA process can greatly reduce erroneous fire decisions. The proposed network architecture is divided into three sections.

In the first section, we detect the SRoFs or non-fire objects in the video frames using a deep object detection model, Faster R-CNN, which consists of CNN feature extractors and a bounding box localizer with a classifier. Here, the bounding boxes locate three different classes: flame, smoke, and non-fire. Usually, flame and smoke cannot be well-separated in a fire so that the smoke-only object in a bounding box is classified into smoke. Also, the whole fire region is treated as one bounding box. A non-fire object implies a still image that has no objects related with a fire or a class of objects that are difficult to differentiate from a fire, such as a chimney evening glow, smoke, and cloud. The non-fire objects have their own bounding boxes. Then, the bounding boxes, including SRoFs and non-fire objects, are projected on the learned feature maps in the last layer of CNN of Faster R-CNN in order to extract the corresponding spatial features.

In the second section, the summarized and concatenated CNN features are temporally accumulated to capture the dynamic behaviors of fire, and the short-term fire decision is made in the two-stage LSTM network. Here, we do not differentiate flame from smoke in the section, so that the LSTM consecutively accumulates both flame and smoke features to decide between fire or non-fire.

Then, in the third section, the short-term decisions are combined in the last majority voting stage for long-term fire decision. The last block also integrates the information to interpret the dynamic fire behaviors in order to determine whether the area of SRoFs, including flame and smoke detected by bounding boxes at Faster R-CNN stages, is increasing or not, for the long-term period. Figure 1 shows the proposed network architecture.

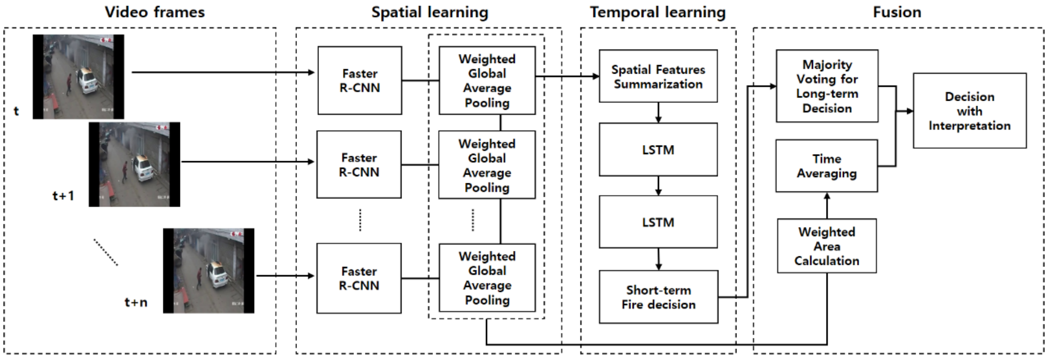

Video frames
Spatial learning
Temporal learning
Fusion
t
t+1
t+n
Faster R-CNN
Weighted Global Average Pooling
Faster R-CNN
Weighted Global Average Pooling
Faster R-CNN
Weighted Global Average Pooling
Spatial Features Summarization
LSTM
LSTM
Short-term Fire decision
Majority Voting for Long-term Decision
Time Averaging
Weighted Area Calculation
Decision with Interpretation

Figure 1. The proposed network architecture.

Figure 2 presents a timing diagram that shows the decision period for each block. The fire objects of flame or smoke are detected for each frame of video, and the CNN features of Faster R-CNN in the detected bounding boxes are temporally accumulated for a period T_{LSTM}. The fire decision for every T_{LSTM} is involved in the majority voting process for every time period T_{vot}, which implies that the final fire decision is repeated for every T_{vot}. The areas of flame and smoke objects are calculated for every frame and smoothed by taking the average over T_{ave}. The changes of average flame and smoke areas in video frames are reported for the time interval T_{rep}. For convenience, we set $T_{rep} = T_{vot}$.

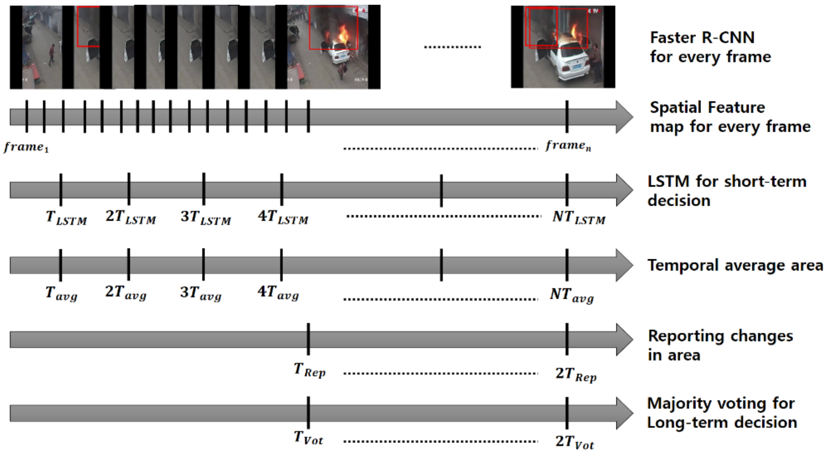


Figure 2. Timing diagram of fire detection.

3.2. *Fire Object Detection Based on Faster Region-Based Convolutional Neural Network (R-CNN)*

Faster R-CNN is a CNN-based object detection method that combines both the Fast R-CNN and the Region Proposal Network (RPN) to share a convolutional network after excluding the fully connected layer. So Faster R-CNN shares a similar structure for object detection with Fast R-CNN, except that Faster R-CNN includes RPN [18] to generate region proposals for objects. Based on the proposals, Faster R-CNN extracts the spatial features through the ROI pooling operation, and then calculates the object positions with class scores by fully connected layers. Usually, Faster R-CNN provides a higher mean Average Precision (mAP) than the single-stage object detection models such as SSD(Single Shot Multibox Detector) [19] and YOLO(You Only Look Once) [20]. Reportedly [21], Faster R-CNN showed mAP as high as 34.9% for the MS-COCO dataset, when the shared CNN feature extractor is equipped with ResNet 101 [22], which we have adopted in our work.

In our method, Faster R-CNN provides the bounding boxes of flame, smoke, and non-fire regions in an image, as shown in Figure 3. Figure 4 represents the sample images of flame, smoke, and non-fire objects. The non-fire objects resemble real fire objects, such as chimney smoke, sunset, and cloud. In addition, the image containing objects that are not related to a fire is itself treated as a non-fire object.

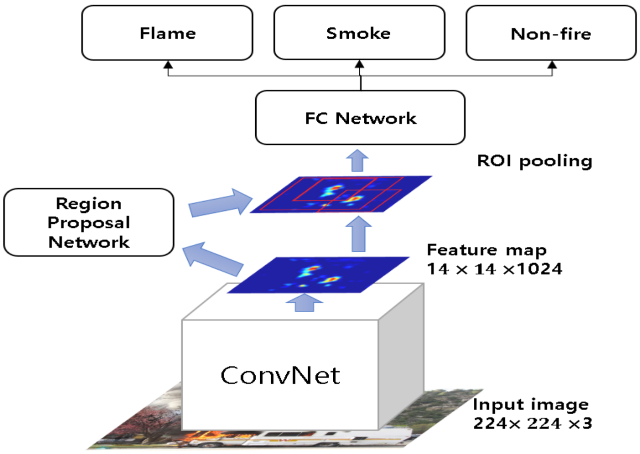
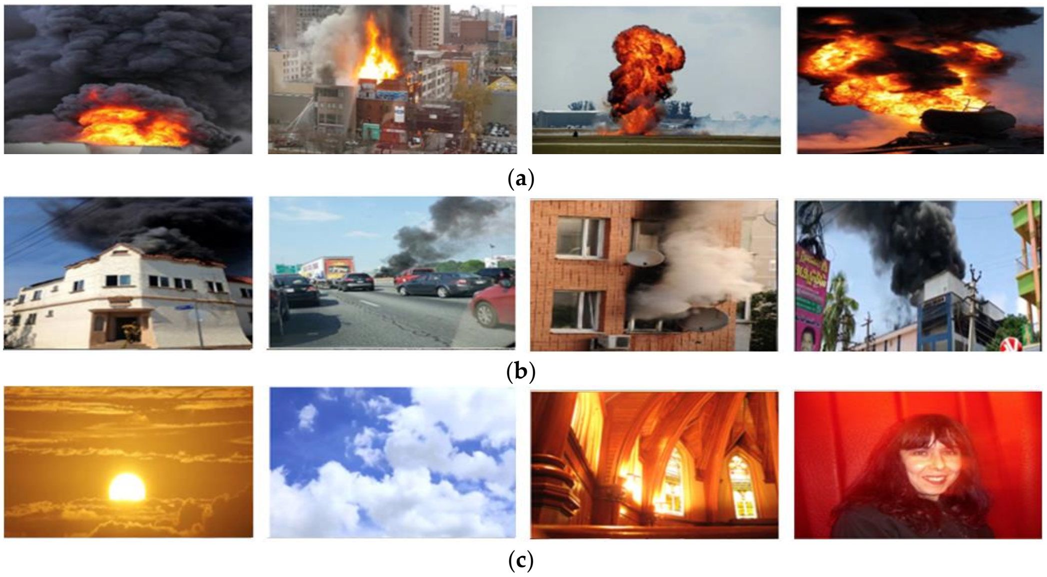


Figure 3. Faster Region-Based Convolutional Neural Network (R-CNN) structure for fire detection.

When the Faster R-CNN detects the classes of flame, smoke, and non-fire objects, false detections may arise due to the possible presence of various types of non-fire objects in a single frame that is similar to a fire. The non-fire objects shown in Figure 4 that resemble fires include sunset, chimney smoke, cloud, etc.

As aforementioned, however, applying this deep object detection model to find SRoFs and non-fire objects offer an advantage. Because the consecutive convolution enlarges the effective area of the operation, a bounding box resulting from the deep object detection model can include the larger area of a receptive field than it encloses. So, the more context information around objects can be captured in the boxes. This implies that SRoF detection becomes robust because it better reflects the context information around the bounding box.


(**a**)

(**b**)

(**c**)

Figure 4. Sample images for training Faster R-CNN; (**a**), (**b**) and (**c**) are flames, smoke, and non-fire images, respectively.

3.3. The Spatial Features Extration

The coordinates of the bounding boxes are projected on the $n \times n \times d$ activation map to extract the spatial features. Here, we extract them in the last layer of CNN, and $n = 14$, and $d = 1024$ when ResNet 101 is used as a base net. For the projected region, we take a scalar feature by taking a weighted average over each feature map. Note that the feature is extracted from the bounding boxes SRoFs, including flames and smoke, and non-fire objects, so that it may give the pure spatial features of diverse types of fire and non-fire objects. Figure 5 shows the part of this feature extraction in our proposed method, including the Faster R-CNN object detection model.

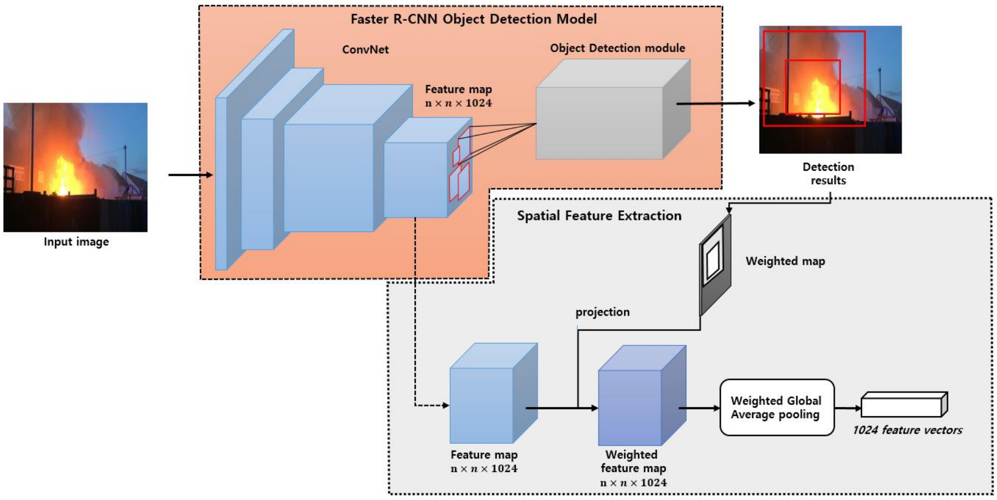


Figure 5. The spatial feature extraction from Faster R-CNN.

The Faster R-CNN can provide more than one SRoF or non-fire object because an image can contain several bounding boxes. Faster R-CNN can detect multiple objects in a frame, where each object is enclosed by a bounding box with its class score and can intersect with each other. Therefore, we should carefully investigate the multiple areas to further consider the temporal behavior of a fire or non-fire object. Figure 6 shows the case where there are several SRoFs.

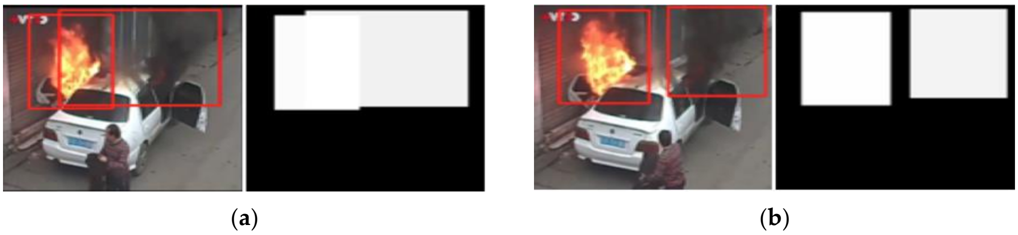

(**a**) (**b**)

Figure 6. Frames with multiple suspected regions of fire (SRoFs). (**a**) Shows the case that the predicted flame and smoke area intersect, and (**b**) shows the case that the predicted flame and smoke area do not intersect.

In our proposed method, a weighted Global Average Pooling (GAP) scheme is adapted to extract the spatial features. After insignificant bounding boxes are filtered out by thresholding their own confidence score, the significant SRoFs and non-fire objects are selected. The image which does not contain any specific small bounding box is treated as a non-fire object whose bounding box covers the whole image with confidence score 1. Note that the significant SRoF or the non-fire object has its own confidence score which can be used to take the weighted GAP. Figure 5 shows the process to extract the spatial features from the last layer of CNN of the Faster R-CNN object detector with d feature maps, where $d = 1024$. From each feature map f_i, the scalar feature value is determined as follows:

$$v_i = \frac{1}{Z}\sum_{\substack{for\ each\ bounding\ box\ b\ of\ SRoFs\\ or\ non-fire\ objects}} w_b \cdot \sum\nolimits_{(x,y)\in b} f_i(x,y), \tag{1}$$

where,

$$Z = \sum_{\substack{for\ each\ bounding\ box\ b\ of\ SRoFs\\ or\ non-fire\ objects}} w_b \cdot \sum\nolimits_{(x,y)\in b} 1. \tag{2}$$

The vector $\mathbf{v} = (v_1, v_2, \ldots, v_d)$ represents the aggregated spatial feature for SRoFs or non-fire objects detected by Faster R-CNN in an image or a frame of a video. In general, the prominent features among d can be found projecting the bounding box of SRoFs or non-fire objects on the feature map similar to the class activation map [23]. Because they are merely spatial features which do not contain temporal information, the feature selection in our proposed method is transferred to the following LSTM stage of the temporal aggregation in a short-term.

3.4. Long Short-Term Memory (LSTM) Network for Fire Features in a Short-Term

In general, it is not appropriate to detect and judge the fire without considering the temporal behavior. In the proposed method we aggregate the changes in the extracted spatial features using LSTM in a short period T_{LSTM}, and try to determine whether it is a fire or a non-fire object. Here, we do not differentiate between flame and smoke. Because the SRoFs or non-fire objects in consecutive frames have been determined by Faster R-CNN, we merely temporally accumulate the spatial features within the corresponding bounding boxes in LSTM and determine whether the consecutive box is fire or not for the short-term. This important step in DTA in the proposed method is similar to a person's quick glance to detect a fire. We assume that the fire decision based on a glance depends on the short-term dynamic characteristics of the fire.

In the proposed method, the LSTM network consists of two stages in which the number of memory cells in LSTM is determined experimentally. The short temporal features pooled through the LSTM network are used to make a short-term fire decision by two soft-max units, one for fire or the other for non-fire. Figure 7 shows the part of LSTM used to accumulate and decide a fire in a short time period.

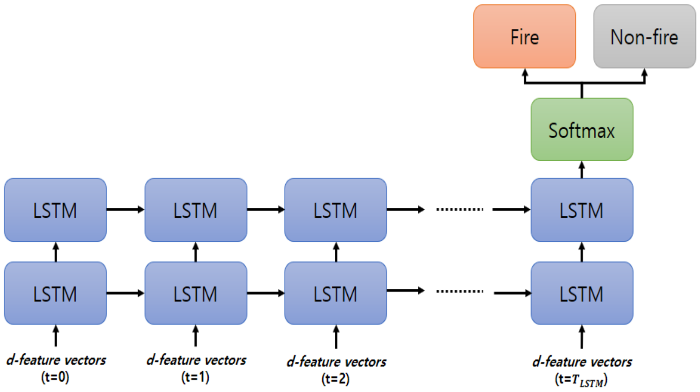


Figure 7. The long short-term memory (LSTM) network for fire detection.

In our method, the LSTM network is separately trained using the weighted GAP spatial features of CNN in bounding boxes. That means the d-dimensional features for fire or non-fire objects in consecutive frames of video clips should be calculated and prepared sequentially for the training of LSTM. We construct the new video dataset which consists of the same video training data for Faster R-CNN and additional video data to supply sufficient training data. The consecutive d-dimensional spatial features calculated from the trained Faster R-CNN for a video from the video dataset are prepared as input streams for the LSTM training. The output label for the LSTM short-term decision is determined according to the annotation of a video clip.

3.5. Majority Voting for Fire Decision

The decision from LSTM reflects a temporal behavior in a short period like a person's quick glance. As the resulting decision is not stable, we make an ensemble of short-term decisions for a period T_{vot} to make the final fire decision. Again, this is similar to human behavior in deciding a fire. The decisions based on short glances are accumulated and combined to make a firm decision on whether it is a fire or not.

The proposed method combines the majority voting in a time window which contains all the decisions from LSTM. The final fire decision by majority voting is given by:

$$\begin{gathered} Final\ Decision = fire, \\ if\ N_{fire} > N_{non-fire}\ for\ T_{vot} \end{gathered} \tag{3}$$

where, N_{fire} and $N_{non-fire}$ represent the number of fire and non-fire decisions respectively, in the LSTM stage during the time window T_{vot}. One can use either weighted voting where the more recent decision has a larger weight in the temporal window or simply take the sum of each soft-max output during T_{vot}.

3.6. The Time Average over Weighted Areas of Suspected Regions of Fire (SRoFs)

The fire judgment from LSTM is based on the temporally aggregated spatial features in the SRoFs and the non-fire objects. Here, we can consider additional temporal features related to the area of SRoFs. The multiple regions allow us to take the weighted sum of SRoFs, where the weights are given by the confidence score corresponding to the SRoF. In Equation (2), Z can be treated as the weighted area of objects in a frame. However, we separately calculate the weighted areas for flame and smoke objects to give a more precise interpretation. Therefore, the weighted areas are calculated as:

$$Z_{flame} = \sum_{\substack{for\ each\ bounding\ box\ b \\ of\ flame\ SRoFs}} w_b \cdot \sum_{(x,y)\,\in\, b} 1 \tag{4}$$

$$Z_{smoke} = \sum_{\substack{for\ each\ bounding\ box\ b \\ of\ smoke\ SRoFs}} w_b \cdot \sum_{(x,y)\in b} 1 \tag{5}$$

After calculating the weighted areas in consecutive frames, we take averages over a period T_{ave}, then for every T_{rep}, the consecutive average areas are given separately from or with the final fire decision to obtain a better understanding of the current dynamic behavior of the fire. Figure 8 shows the process to generate information on the fire's dynamic behavior.

Another fire decision can be made independently from the majority voting of the LSTM decisions. For example, the dynamically increasing or decreasing areas of flame and smoke are detected and accumulated by another type of decision-making algorithm. This other fire decision could be merged with the previous fire decision from the majority voting of the LSTM decisions for a more refined decision, even though it is not implemented. While the time for majority voting can vary, the accuracy of the model improves with longer durations for the majority voting.

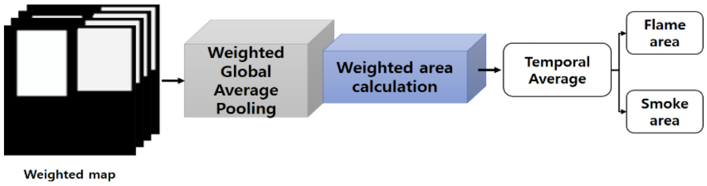


Figure 8. Calculation of weighted areas of SRoFs.

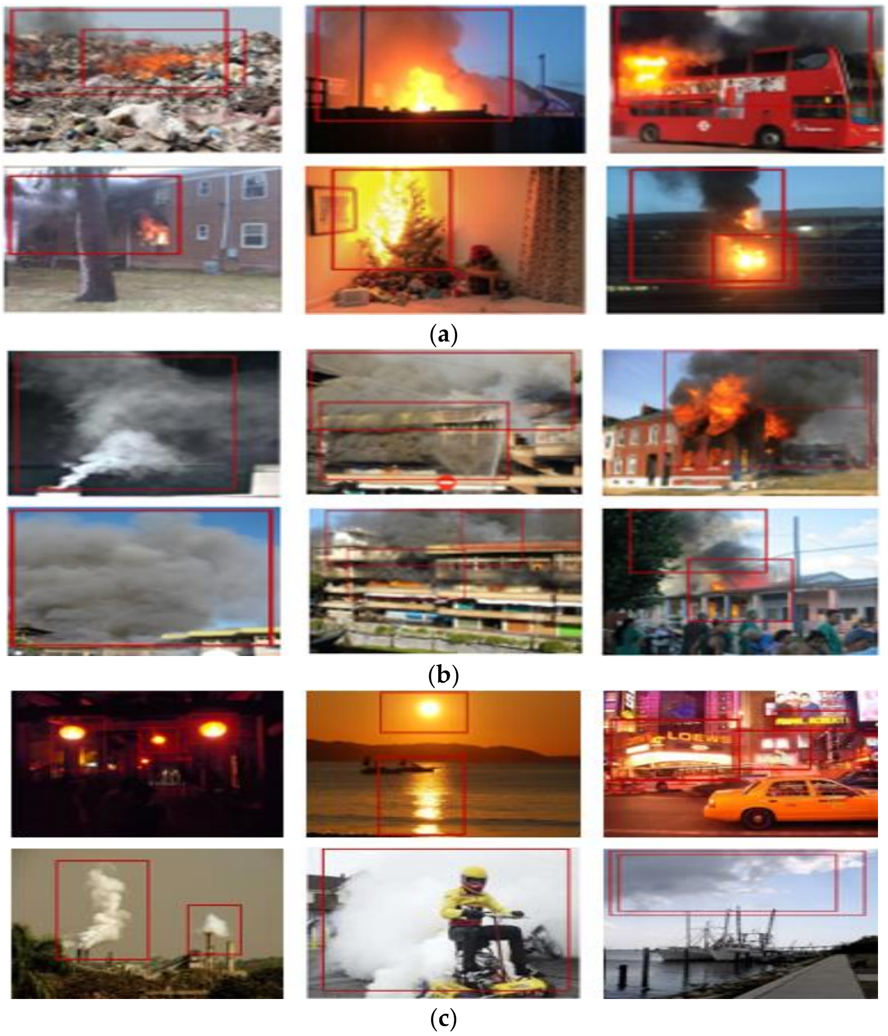
4. Experiments and Results

Our method does not use end-to-end training because the weighted GAP within bounding boxes and the majority voting processes are included. Because both of them are non-differentiable operations, Faster R-CNN and LSTM stages should be separately trained in the proposed method.

4.1. Training Faster R-CNN and Its Accuracy

Faster R-CNN requires still images for training data so we collected them from several data sources. Some fire and smoke images were taken from Youtube video clips. Also, the same data as the previous works were added [6,24–26], which include still images and frames taken from video clips. In addition, the Flickr-fire dataset was included in our dataset. Finally, we constructed a dataset of 73,887 still images, consisting of 22,729 flame, 23,914 smoke, and 27,244 non-fire images. The images are divided into 75% for training, 15% for validation, and 10% for test data. For training, the data are augmented by a horizontal flip. Table 1 shows the training parameters for Faster R-CNN.

The performance of the Faster R-CNN is measured by mAP and is shown in Table 2. The sample results of the flame and smoke detection are shown in Figure 9. There are several false positive detections for clouds, chimney smoke, lighting lamp, steam, etc., which are almost undetectable without considering the temporal characteristics.


(**a**)

(**b**)

(**c**)

Figure 9. Results of Faster R-CNN fire detection, (**a**) flames detection results, (**b**) smoke detection results, and (**c**) false positive detection results.

Table 1. Training parameters of Faster R-CNN.

Parameter	Method
Iteration	150,000
Step size	100,000
Weight decay	0.00004
Learning rate	0.01
Learning rate decay	0.00001(iteration equal step size)
Batch size	256
Pre-train weight	ResNet-101

Table 2. Mean Average Precision (mAP) of Faster R-CNN.

mAP	Flame	Smoke	Non-fire
88.3%	89.4%	87.5%	88.1%

4.2. Training LSTM and Its Performance

The LSTM in the proposed method is trained with video clips. We collected 1,309 video clips from Youtube, comprising 672 clips of fire, and 637 of non-fire. As in the Faster R-CNN, the video clips of non-fire objects can include hard negative examples like clouds, chimney smoke, lighting lamp, and steam, or simply objects that are irrelevant to fire. Figure 10 shows samples of flame, smoke, and non-fire objects from the LSTM training video dataset.

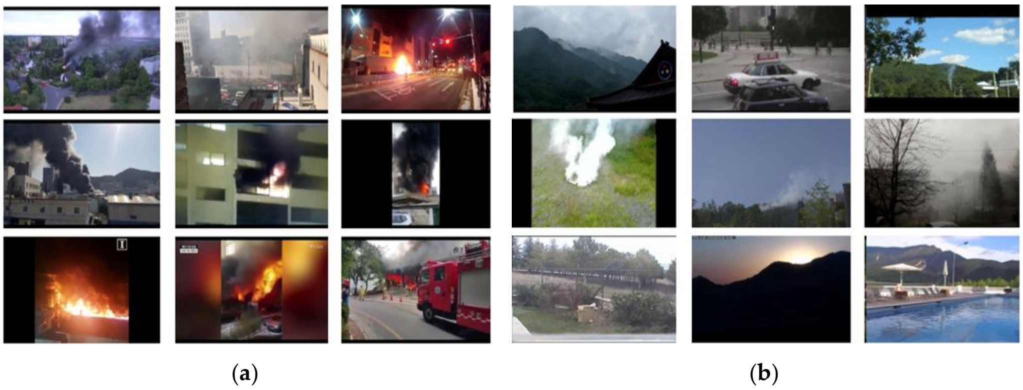


Figure 10. Sample still shots including flame, smoke, and non-fire objects taken from video clips for LSTM training. The (**a**) images are taken from videos of fire, while the (**b**) images are from non-fire video clips.

Here, we do not discriminate between flame and smoke as mentioned before. The video clips are divided into 60 consecutive frames with 15 frames of overlap that last for about 2 s if 30 frames per second are assumed. This implies the LSTM network captures short-term dynamic behaviors of a fire or non-fire and decides whether it is a fire or not for every 1.5 s. Here we assume a person's quick glance for fire decision happens every 1.5 s. The time duration can be adjusted according to the situation when our method is implemented.

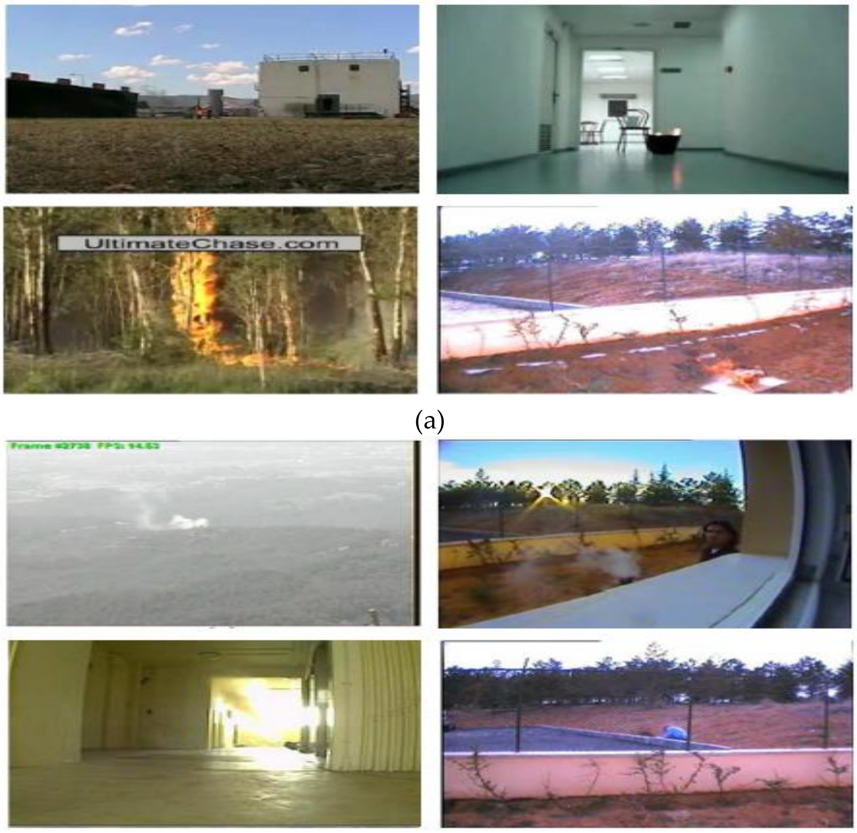
For the LSTM training, we prepared 8,527 positive and 7,547 negative examples of 60 frame video clips from Youtube videos. From the examples, 75% of the data were selected for training, 15% for validation, and 10% for testing. From each video clip, we obtained bounding boxes and their corresponding 1024-dimensional feature for every consecutive frame, which gave a set of sequential inputs to LSTM. Table 3 shows the parameters of LSTM training and the performance of the test data shown in Table 4, according to the number of memory cells in LSTM. To compare the performance with another method, we evaluated the results using the dataset in reference [11]. Information of the public dataset which consists of 31 video clips under different conditions, and the still shots taken from several samples, are shown in Table 5 and Figure 11, respectively.

Table 3. Training parameters of LSTM.

Parameter	Method
Input size	1024
Time step	60
LSTM cell unit	128/256/512/1024
Learning rate	0.001
Learning rate decay	0.0001 (epoch equal 120)
Weight decay	0.0004
Dropout	0.5
Batch size	256
Weight initialization	Xavier initialization
epoch	200

Table 4. Performance of test video clips for LSTM hidden cell unit.

Method	Accuracy (%)
SRoF-LSTM, Hidden cell unit = 128	92.12
SRoF-LSTM, Hidden cell unit = 256	93.87
SRoF-LSTM, Hidden cell unit = 512	95.00
SRoF-LSTM, Hidden cell unit = 1024	93.50



(a)

(**b**)

Figure 11. Set of the representative images from Foggia et al.'s [11] dataset. The (**a**) images are taken from videos of fires, while the (**b**) images are from non-fire videos.

Table 5. The dataset information [11].

Video Name	Resolution	Fames	Frame Rate	Fire	Description
Fire1	320 × 240	705	15	Yes	A fire generated into a bucket and a person walking near it.
aFire2	320 × 240	116	29	Yes	A fire very far from the camera generated into a bucket.
Fire3	400 × 256	255	15	Yes	A big fire in a forest.
Fire4	400 × 256	240	15	Yes	See the notes of the video Fire3.
Fire5	400 × 256	195	15	Yes	See the notes of the video Fire3.
Fire6	320 × 240	1200	10	Yes	A fire generated in a red ground.
Fire7	400 × 256	195	15	Yes	See the notes of the video Fire3.
Fire8	400 × 256	240	15	Yes	See the notes of the video Fire3.
Fire9	400 × 256	240	15	Yes	See the notes of the video Fire3.
Fire10	400 × 256	210	15	Yes	See the notes of the video Fire3.
Fire11	400 × 256	210	15	Yes	See the notes of the video Fire3.
Fire12	400 × 256	210	15	Yes	See the notes of the video Fire3.
Fire13	320 × 240	1,650	25	Yes	A fire in a bucket in indoor environm ent.
Fire14	320 × 240	5,535	15	Yes	Fire generated by a paper box. The video has been acquired by the authors near a street.
Fire15	320 × 240	240	15	No	Some smoke seen from a closed window. A red reflection of the sun appears on the glass.
Fire16	320 × 240	900	10	No	Some smoke pot near a red dust bin.
Fire17	320 × 240	1725	25	No	Some smoke on the ground near a moving vehicle and moving trees.
Fire18	352 × 288	600	10	No	Some far smoke on a hill.
Fire19	320 × 240	630	10	No	Some smoke on a red ground.
Fire20	320 × 240	5,958	9	No	Some smoke on a hill with red buildings.
Fire21	720 × 480	80	10	No	Some smoke far from the camera behind some moving trees.
Fire22	480 × 272	22,500	25	No	Some smoke behind a mountain in front of the university of salerno.
Fire23	720 × 576	6,097	7	No	Some smoke above a mountain.
Fire24	320 × 240	372	10	No	Some smoke in a room.
Fire25	352 × 288	140	10	No	Some smoke far from the camera in a city.
Fire26	720 × 576	847	7	No	See the notes of the video Fire24.
Fire27	320 × 240	1,400	10	No	See the notes of the video Fire19.
Fire28	352 × 288	6,025	25	No	See the notes of the video Fire18.
Fire29	720 × 576	600	10	No	Some smoke in a city covering red buildings.
Fire30	800 × 600	1,920	15	No	A person moving in a lab holding a red ball.
Fire31	800 × 600	1,485	15	No	A person moving in a lab with a red notebook.

4.3. Majority Voting and Interpretation of Fire Behavior

The LSTM short-term fire decisions during T_{vot} are involved in the majority voting for the final fire decision. Because only short video clips are included in the dataset for comparison in Table 6, we take $T_{vot} = 10$ s for majority voting. Even with such a short-term ensemble, the accuracy increases by up to 97.92%.

Table 6. Performance comparison with other methods.

Methods	False Positive (%)	False Negative (%)	Accuracy (%)
Proposed method (hidden unit cell = 512)	3.04	1.73	95.00
Proposed method (Majority Voting = 10 s)	2.47	1.38	**97.92**
Khan Muhammad et al. [14]	8.87	2.12	94.50
Foggia et al. [11]	11.67	0.00	93.55
De Lascio et al. [27]	13.33	0.00	92.86
Habibugle et al. [28]	5.88	14.29	90.32
Rafiee et al. (YUV color) [29]	17.65	7.14	74.20
Celik et al. [5]	29.41	0.00	83.87
Chen et al. [7]	11.76	14.29	87.1
Arpit Jadon et al. [30]	1.23	2.25	96.53
Khan Muhammad et al. [31]	**0**	**0.14**	95.86

Also, we collected an additional 38 video clips from the internet, including Youtube, which have a relatively long playing-time. Table 7 represents the categorized fire/non-fire video clips with their time-varying behaviors. Figure 12 shows samples of the video dataset. We performed the majority voting for the final fire decision and evaluated the accuracy according to the time period of T_{vot} and the results are summarized in Table 8. Note that the longer time period provides better accuracy in general because more LSTM decisions are combined in the majority voting to make a robust ensemble. However, the dispatch of firemen should be done as early as possible so that T_{vot} can be adjusted by the trade-off between the accuracy and the critical time for dispatch.

We have compared the performances of our scheme in terms of three metrics including false positive, false negative, and accuracy. While the method of Khan Muhammad et al. [31] performs the best in terms of false positive and false negative, ours with the delayed decision of the majority voting in 10 s outperforms in accuracy. Note that our proposed method can produce this better by introducing the delayed decision in DTA.

Table 7. Fire detection accuracy of Faster R-CNN.

Fire State Change	Interpretation	Number of Video Clips
Decreasing	Decreasing flame/Increasing smoke or steam	9
Increasing	Increasing flame	9
Maintaining	Sustain flame/smoke	11
Non-fire	False object	11

Table 8. Accuracy of final fire decision after majority voting with respect to T_{vot}.

30 s	1 min	1 min 30 s	2 min	2 min 30 s	3 min
96.73%	99.28%	99.64%	99.94%	100%	100%

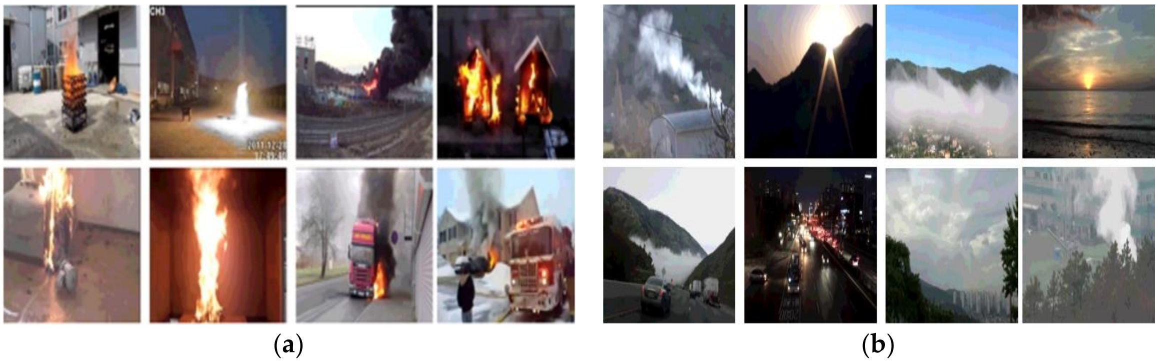


Figure 12. Sample still shots taken from video clips for the experiment of majority voting and interpretation of dynamic fire behavior: (**a**) real fires, and (**b**) non-fires such as chimney smoke, sunset, cloud, fog, and light.

Also, we monitored the changes in the area of smoke (or steam) and flame. In the experiment we set $T_{rep} = T_{vot} = 1$ min. Because Faster R-CNN frequently confuses between true smoke and steam, the results include both areas without distinction. In the video clip shown at the first row and the first column, a fire starts with a flame, but a man pours a bottle of water to extinguish the flame as it grows, then steam (or smoke) begins to increase. Figure 13 shows the sample frames of the video sequence and Figure 14 represent the changes in areas (pixels in a frame) of flame and smoke and the final decisions over time. The decision of majority voting starts with fire but then changes into non-fire.

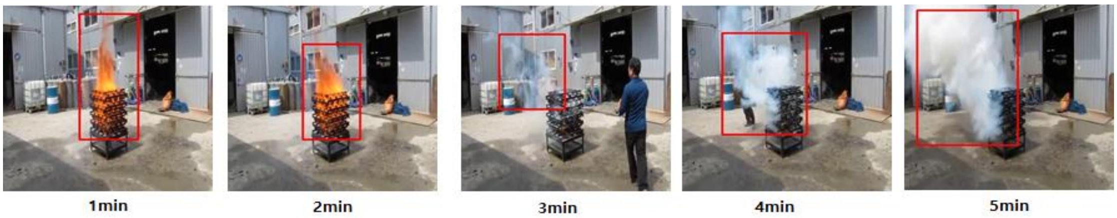


Figure 13. Sample still shots of the video clip shown in the first row and first column.

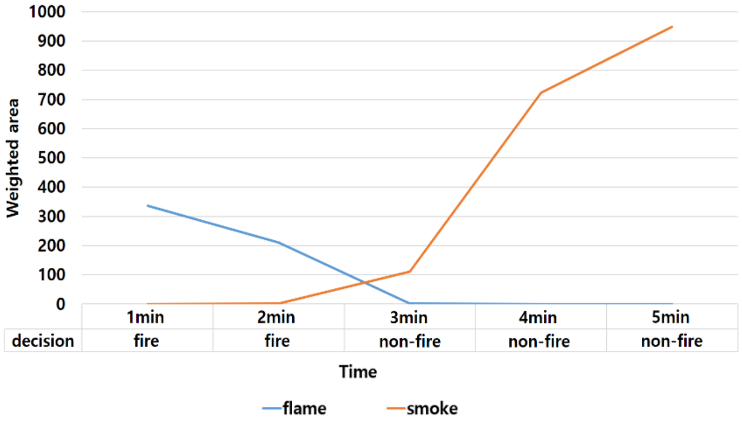


Figure 14. Changes in areas of flame and smoke with final decisions by majority voting for the video clip of Figure 13.

Figure 15 represents a sunrise video clip. Even though Faster R-CNN wrongly detects the flame objects and the area of flame increases, the final decisions of majority voting are consistently non-fire, as shown in Figure 16. We obtained similar experimental results for the other 36 video clips in Table 7, which represent the correct final decisions even though Faster R-CNN provides wrong object detection for clouds, steam from man-holes, and sunset video clips. In the video clip of Figure 17, the fire is increasing after decreasing for about 3 min, and Figure 18 shows the successfully interpreted changes in the fire. Figure 19 shows some still shots of a continuously decreasing fire, and the corresponding interpretation in Figure 20 captures the behavior of the fire

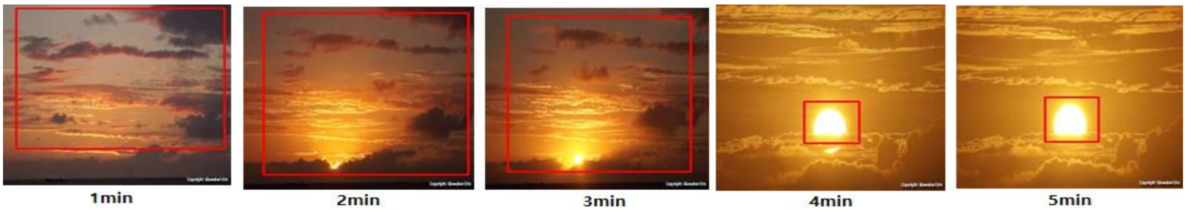


Figure 15. Sample still shots of the sunrise video clip.

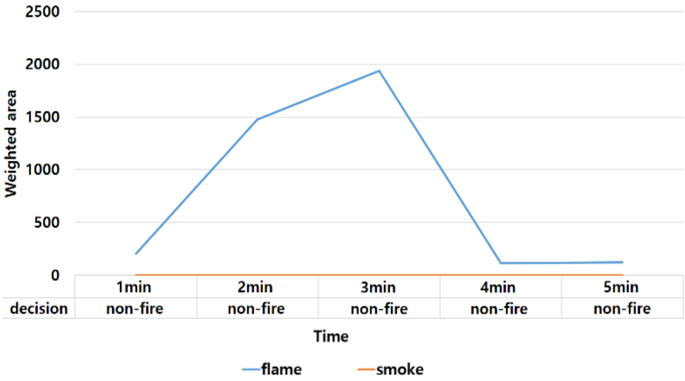


Figure 16. Changes in areas of flame and smoke with final decisions by majority voting for the video clip of Figure 15.

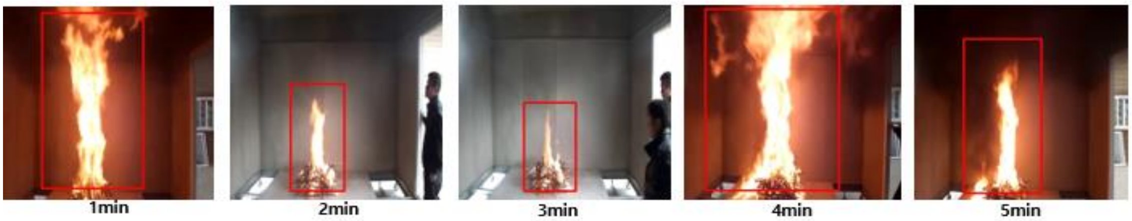


Figure 17. Sample still shots of decreasing followed by increasing flame video clip.

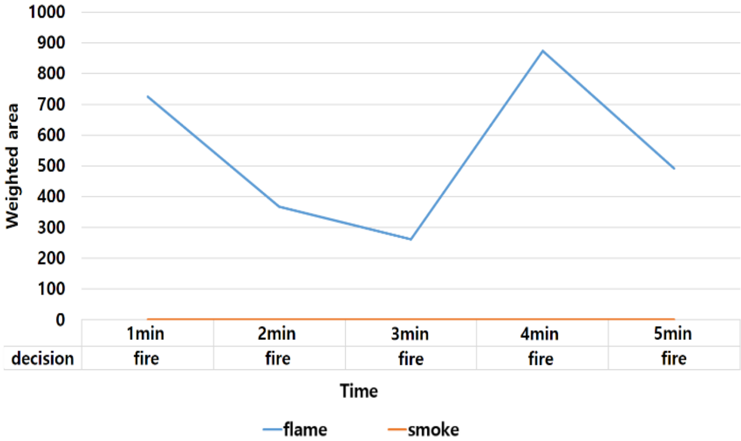


Figure 18. Changes in areas of flame and smoke with final decisions by majority voting for the video clip of Figure 17.

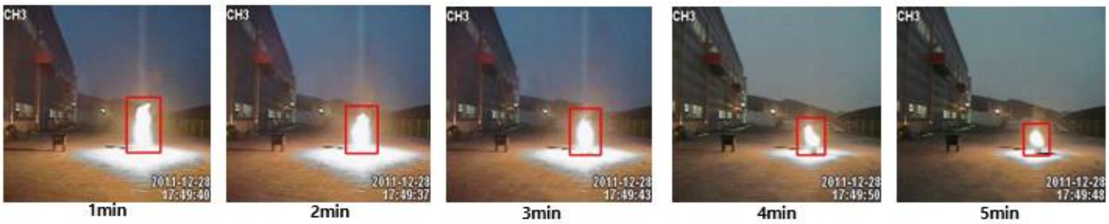


Figure 19. Sample still shots of decreasing flame video clip.

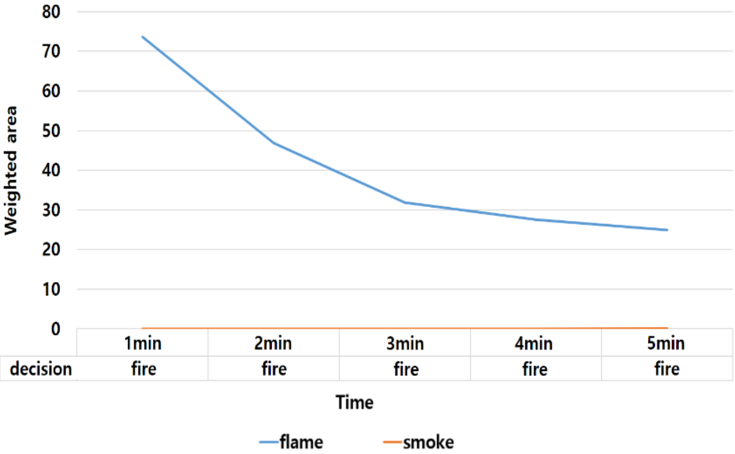


Figure 20. Changes in areas of flame and smoke with final decisions by majority voting for the video clip of Figure 19.

5. Conclusions

We have proposed a deep learning-based fire detection method, called DTA, which imitates the human detection process. We assumed that the DTA process can greatly reduce erroneous fire detection. The proposed method uses the Faster R-CNN fire detection model to detect the SRoF based on its spatial features. Then, the features summarized from the SRoFs and non-fire regions in successive frames are accumulated by LSTM to classify whether there is a fire or not in a short-term period. The successive short-term decisions are then combined in the majority voting for the final decision in a long-term period. In addition, the areas of both flame and fire are calculated and their temporal changes are reported to interpret the dynamic behavior of the fire with the final fire decision.

The proposed method has been experimentally proven to provide excellent fire detection accuracy, by reducing the false detections and misdetections, and to successfully interpret the temporal behavior of flame and smoke, which possibly reduces the false dispatch of firemen. In addition, we have constructed a large fire dataset which contains diverse still images and video clips that enhance the data from well-known public datasets. Not only is the dataset used for the training and testing of our experiment, but it also could be an asset for future fire research.

Author Contributions: Conceptualization, B.K. and J.L.; Data curation, B.K.; methodology, B.K. and J.L.; Project administration, B.K.; Resources, B.K.; Writing—original draft, B.K.; Writing—review & editing, J.L.

Funding: For the research leading to these results, the authors would like to thank the Korean Ministry of Education and Brain Korea 21 PLUS Project for their funding.

Acknowledgments: This work was supported in part by Basic Science Research Program through the National Research Foundation of Korea(NRF) funded by the Ministry of Education (No. 2019R1A6A1A09031717) and Brain Korea 21 PLUS Project, National Research Foundation of Korea.

Conflicts of Interest: The authors declare no conflict of interest. The funding sponsors had no role in the design of the study, in the collection, analyses, or interpretation of data, in the writing of the manuscript, and in the decision to publish the results.

References

1. Chi, R.; Lu, Z.M.; Ji, Q.G. Real-time multi-feature based fire flame detection in video. *IET Image Process.* **2016**, *11*, 31–37. [CrossRef]
2. Evarts, B. *Fire loss in the United States during 2017*; National Fire Protection Association, Fire Analysis and Research Division: Quincy, MA, USA, 2018.
3. Qiu, T.; Yan, Y.; Lu, G. An autoadaptive edge-detection algorithm for flame and fire image processing. *IEEE Trans. Instrum. Meas.* **2012**, *61*, 1486–1493. [CrossRef]
4. Liu, C.B.; Ahuja, N. Vision based fire detection. In Proceedings of the 17th International Conference on Pattern Recognition (ICPR), Cambridge, UK, 26 August 2004; pp. 134–137.
5. Celik, T.; Demirel, H.; Ozkaramanli, H.; Uyguroglu, M. Fire detection using statistical color model in video sequences. *J. Vis. Commun. Image Represent.* **2007**, *18*, 176–185. [CrossRef]
6. Ko, B.C.; Ham, S.J.; Nam, J.Y. Modeling and formalization of fuzzy finite automata for detection of irregular fire flames. *IEEE Trans. Circuits Syst. Video Technol.* **2011**, *21*, 1903–1912. [CrossRef]
7. Chen, T.H.; Wu, P.H.; Chiou, Y.C. An early fire-detection method based on image processing. In Proceedings of the International Conference on Image Processing (ICIP), Singapore, 24–27 October 2004; pp. 1707–1710.
8. Wang, T.; Shi, L.; Hou, X.; Yuan, P.; Bu, L. A new fire detection method based on flame color dispersion and similarity in consecutive frames. In Proceedings of the 2017 Chinese Automation Congress (CAC), Jinan, China, 20–22 October 2017; pp. 151–156.
9. Borges, P.V.K.; Izquierdo, E. A probabilistic approach for vision-based fire detection in videos. *IEEE Trans. Circuits Syst. Video Technol.* **2010**, *20*, 721–731. [CrossRef]
10. Mueller, M.; Karasev, P.; Kolesov, I.; Tannenbaum, A. Optical flow estimation for flame detection in videos. *IEEE Trans. Image Process.* **2013**, *22*, 2786–2797. [CrossRef] [PubMed]
11. Foggia, P.; Saggese, A.; Vento, M. Real-time fire detection for video surveillance applications using a combination of experts based on color, shape, and motion. *IEEE Trans. Circuits Syst. Video Technol.* **2015**, *25*, 1545–1556. [CrossRef]
12. Frizzi, S.; Kaabi, R.; Bouchouicha, M.; Ginoux, J.M.; Moreau, E.; Fnaiech, F. Convolutional neural network for video fire and smoke detection. In Proceedings of the IECON 2016—42nd Annual Conference of the IEEE Industrial Electronics Society, Florence, Italy, 23–26 October 2016; IEEE: Piscataway, NJ, USA, 2016; pp. 877–882.
13. Zhang, Q.; Xu, J.; Guo, H. Deep convolutional neural networks for forest fire detection. In Proceedings of the 2016 International Forum on Management, Education and Information Technology Application, Guangzhou, China, 30–31 January 2016; Atlantis Press: Paris, France, 2016.
14. Muhammad, K.; Ahmad, J.; Lv, Z.; Bellavista, P.; Yang, P.; Baik, S.W. Efficient Deep CNN-Based Fire Detection and Localization in Video Surveillance Applications. *IEEE Trans. Syst. Man Cybern. Syst.* **2018**, *99*, 1–16. [CrossRef]
15. Iandola, F.N.; Han, S.; Moskewicz, M.W.; Ashraf, K.; Dally, W.J.; Keutzer, K. SqueezeNet: AlexNet-level accuracy with 50x fewer parameters and<0.5 MB model size. *arXiv* **2016**, arXiv:1602.07360.
16. Hochreiter, S.; Schmidhuber, J. LSTM can solve hard long time lag problems. In *Advances in Neural Information Processing Systems*; NIPS: San Diego, CA, USA, 1997; pp. 473–479.
17. Hu, C.; Tang, P.; Jin, W.; He, Z.; Li, W. Real-Time Fire Detection Based on Deep Convolutional Long-Recurrent Networks and Optical Flow Method. In Proceedings of the 2018 37th Chinese Control Conference (CCC), Wuhan, China, 25–27 July 2018; IEEE: Piscataway, NJ, USA, 2018; pp. 9061–9066.
18. Ren, S.; He, K.; Girshick, R.; Sun, J. Faster R-CNN: Towards real-time object detection with region proposal networks. In Proceedings of the Neural Information Processing Systems (NIPS), Montreal, QC, Canada, 7–12 December 2015.
19. Liu, W.; Anguelov, D.; Erhan, D.; Szegedy, C.; Reed, S.; Fu, C.Y.; Berg, A.C. SSD: Single shot multibox detector. In Proceedings of the European Conference on Computer Vision (ECCV), Amsterdam, The Netherlands, 8–16 October 2016.
20. Redmon, J.; Divvala, S.; Girshick, R.; Farhadi, A. You only look once: Unified, real-time object detection. In Proceedings of the Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, USA, 26 June–1 July 2016.

21. Object Detection: Speed and Accuracy Comparison (Faster R-CNN, R-FCN, SSD, FPN, RetinaNet and YOLO). Available online: https://medium.com/@jonathan_hui/object-detection-speed-and-accuracy-comparison-faster-r-cnn-r-fcn-ssd-and-yolo-5425656ae359 (accessed on 28 March 2018).
22. He, K.; Zhang, X.; Ren, S.; Sun, J. Deep residual learning for image recognition. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 26 June–1 July 2016.
23. Zhou, B.; Khosla, A.; Lapedriza, A.; Torralba, A.; Oliva, A. Learning deep features for discriminative localization. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 26 June–1 July 2016.
24. Chino, D.Y.T.; Avalhais, L.P.S.; Rodrigues, J.F.; Traina, A.J.M. Bowfire: detection of fire in still images by integrating pixel color and texture analysis. In Proceedings of the 2015 28th SIBGRAPI Conference on Graphics, Patterns and Images, Salvador, Brazil, 26–29 August 2015; IEEE: Piscataway, NJ, USA.
25. Verstockt, S.; Beji, T.; De Potter, P.; Van Hoecke, S.; Sette, B.; Merci, B.; Van De Walle, R. Video driven fire spread forecasting (f) using multi-modal LWIR and visual flame and smoke data. *Pattern Recognit. Lett.* **2013**, *34*, 62–69. [CrossRef]
26. Bedo, M.; Blanco, G.; Oliveira, W.; Cazzolato, M.; Costa, A.; Rodrigues, J.; Traina, A.; Traina, C., Jr. Techniques for effective and efficient fire detection from social media images. *arXiv preprint* **2015**, arXiv:1506.03844.
27. Di Lascio, R.; Greco, A.; Saggese, A.; Vento, M. Improving fire detection reliability by a combination of video analytics. In Proceedings of the International Conference Image Analysis and Recognition, Vilamoura, Portugal, 22–24 October 2014; Springer: Cham, Switzerland, 2014.
28. Habiboğlu, Y.H.; Günay, O.; Çetin, A.E. Covariance matrix-based fire and flame detection method in video. *Mach. Vis. Appl.* **2012**, *23*, 1103–1113. [CrossRef]
29. Rafiee, A.; Dianat, R.; Jamshidi, M.; Tavakoli, R.; Abbaspour, S. Fide and smoke detection using wavelet analysis and disorder characteristics. In Proceedings of the 2011 3rd International Conference on Computer Research and Development (ICCRD), Shanghai, China, 11–13 March 2011; pp. 262–265.
30. Jadon, A.; Omama, M.; Varshney, A.; Ansari, M.S.; Sharma, R. FireNet: A Specialized Lightweight Fire & Smoke Detection Model for Real-Time IoT Applications. *arXiv* **2019**, arXiv:1905.11922.
31. Muhammad, K.; Khan, S.; Elhoseny, M.; Ahmed, S.H.; Baik, S.W. Efficient Fire Detection for Uncertain Surveillance Environment. *IEEE Trans. Ind. Inform.* **2019**, *15*, 3113–3122. [CrossRef]