applied sciences

MDPI

Article

A Bayesian Network-Based Information Fusion Combined with DNNs for Robust Video Fire Detection

Byoungjun Kim and Joonwhoan Lee *

Division of Computer Science and Engineering, Jeonbuk National University, Jeonju 54896, Korea; breed213@jbnu.ac.kr
* Correspondence: chlee@jbnu.ac.kr; Tel.: +82-63-270-2406

Abstract: Fire is an abnormal event that can cause significant damage to lives and property. Deep learning approach has made large progress in vision-based fire detection. However, there is still the problem of false detections due to the objects which have similar fire-like visual properties such as colors or textures. In the previous video-based approach, Faster Region-based Convolutional Neural Network (R-CNN) is used to detect the suspected regions of fire (SRoFs), and long short-term memory (LSTM) accumulates the local features within the bounding boxes to decide a fire in a short-term period. Then, majority voting of the short-term decisions is taken to make the decision reliable in a long-term period. To ensure that the final fire decision is more robust, however, this paper proposes to use a Bayesian network to fuse various types of information. Because there are so many types of Bayesian network according to the situations or domains where the fire detection is needed, we construct a simple Bayesian network as an example which combines environmental information (e.g., humidity) with visual information including the results of location recognition and smoke detection, and long-term video-based majority voting. Our experiments show that the Bayesian network successfully improves the fire detection accuracy when compared against the previous video-based method and the state of art performance has been achieved with a public dataset. The proposed method also reduces the latency for perfect fire decisions, as compared with the previous video-based method.

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

Citation: Kim, B.; Lee, J. A Bayesian Network-Based Information Fusion Combined with DNNs for Robust Video Fire Detection. *Appl. Sci.* **2021**, *11*, 7624. https://doi.org/10.3390/app11167624

Academic Editor: José Carlos Bregieiro Ribeiro

Received: 3 July 2021
Accepted: 16 August 2021
Published: 19 August 2021

1. Introduction

Fire is an atypical event that can cause significant injury and property damage over a very short time [1]. According to the National Fire Protection Association (NAPA), in 2017, the United States fire departments responded to an estimated 1,319,500 fires [2], which were responsible for 3400 civilian fatalities, 14,670 civilian injuries, and an estimated $23 billion in direct property loss. To reduce the number and degree of these disasters, fire detection that reliably avoids false alarms and misdetection at an early stage is crucial. To meet this need, a variety of automatic fire detection technologies are widely deployed, with new technologies under development.

The fire detection technology falls into two broad categories: the traditional fire alarm and fire detection using computer vision. Traditional fire alarm technology relies on proximity-activated smoke or heat sensors. The sensors require human involvement to confirm a fire once the alarm sounds. These systems also require additional equipment to determine the size, location, and severity of the fire. To overcome these limitations, researchers have been investigating the possibilities of computer vision-based methods in combination with various supplementary sensors [3–7]. This category of technologies provides more comprehensive surveillance, and allows for less human intervention and faster responses (as a fire can be confirmed without requiring a visit to the fire location), and provides detailed fire information (including location, size, and severity). Despite

these advantages, issues with system complexity and false detection have stymied the development of these systems. As a result, researchers have devoted a great deal of effort to address these issues to computer vision technology. Early research on computer vision-based fire detection was focused on fire color, which varies depending on environmental conditions such as ambient light and weather. Additional studies looked for supplemental features in addition to color, including area, surface, boundary, and motion of and within the suspected region, applying various decision-making algorithms (including Bayes classifier and multi-expert system) to reach a trustworthy decision. With few exceptions, however, researchers have focused on flame and smoke detection from either a single frame or a small number of frames of closed-circuit television (CCTV).

Aside from the static and dynamic characteristics of fire that manifest over a short period, flame and smoke display long-term dynamic behaviors. These dynamic behaviors, when identified by their distinct motion characteristics (such as the optical flow), extracted from consecutive frames of a video sequence, and combined with analysis of static characteristics, can be exploited to improve fire detection. However, it is not an easy task to explore the static and dynamic characteristics of flame and smoke, and exploit them in a vision-based system, as doing so requires a large amount of domain knowledge. By using a deep learning approach, however, exploration and exploitation can be performed by a properly trained neural network. This approach becomes feasible once a sufficient dataset of flame and smoke images and video clips has been built.

The paper proposes a deep learning-based fire detection method, known as detection and temporal accumulations (DTA), which imitates the human fire detection process [3]. In DTA, the suspected region of fire (SRoF) is detected by the Faster Region-based Convolutional Neural Network (R-CNN) using the fire's spatial features as they appear against non-fire objects. Next, features summarized by the object detection model across successive frames are accumulated by long short-term memory (LSTM), which classifies images based on whether there is a fire or not across a short-term period. The decisions for successive short-term periods are then combined in the majority voting for the decision in a long-term period. SRoF areas, including both flame and smoke, are then calculated, and their temporal changes are reported to incorporate the dynamic fire behavior into the final fire decision.

However, the DTA assigned no role to environmental information (such as humidity, weather conditions, fire location) in improving the performance of fire detection. Some information, such as fireplaces location and the existence of smoke, can be obtained from the same visual sources as used by Faster R-CNN, but other environmental information, such as humidity or weather conditions, must be obtained from other sensors. This paper proposes to use Bayesian network to combine all of this information to improve the accuracy and the latency of perfect fire detection. This extended version of DTA is called Detection and Temporal Accumulation with Information Fusion (DTAIF).

Experiments show that a simple Bayesian network can improve the fire detection accuracy when compared with the prior exclusively video-based method, and results in the state of art performance when applied to the public dataset. The proposed method with the simple Bayesian network also reduces the time to reach (latency for) a perfect fire decision as compared to the prior video-based method. Although a simple Bayesian network is designed here only to show the potential, we believe such systems that combine all sensor information by Bayesian net are useful for improving fire detection performance. In addition to DTA-based fire decision approach, our contributions can be summarized as follows.

The Bayesian network-based information fusion combined with deep neural net (DNNs) goes one step further than previous DTA, whereby SRoFs were detected in a scene and the temporal behaviors were continuously monitored and accumulated to ultimately decide whether a fire existed. Humans, however, do not determine the existence of fire, based solely on the localized visual behavior within a fire-like region. In addition to location of the scene, people account for environmental information, such as humidity and weather

conditions, to make robust decision. In order to further emulate this human approach, this paper proposes to use a Bayesian network that fuses various kinds of information (DTAIF). To verify the proposition, a simple Bayesian network is designed as an example and its effectiveness is shown in terms of accuracy and latency for fire decision in the experiments.

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

2. Related Works

As the conventional computer vision techniques, the traditional vision-based fire detection is decomposed into two parts: exploration of appropriate features and exploitation of them for fire decision. Colors of flame and smoke are important for identifying fire region in a still image even though they generally vulnerable to a variety of environmental factors [6–9]. Borges and Izquierdo [10] adopted the Bayes classifier with additional features, including area, surface, and boundary of the fire area. Mueller proposed to use optical flow, a simple type of short-term dynamic feature, for the fire area and the neural network-based decision method [11]. Foggia [12] proposed a multi-expert system for reliable decision that combined the results of an analysis of a fire's color, shape, and motion characteristics. Although they were insufficient to detect a fire on their own, supplementary features, such as texture, shape, and optical flow, were helpful to reduce false detections.

Dimitropoulos et al. [13] modeled the fire behavior by employing various spatial-temporal features, such as color probability, flickering, spatial, and spatial-temporal energy. Lin et al. [14] used several general volume local binary patterns to extract dynamic texture, including LBPTOP, VLBP, CLBPTOP, and CVLBP. Furthermore, dynamic texture descriptors were obtained using Weber Local Descriptor in Three Orthogonal Planes (WLD-TOP) [15]. All these conventional dynamic texture analyses require domain knowledge of fires on captured images. Moreover, almost all methods account for the short-term dynamic behavior of fire, while a fire has longer-term dynamic behavior.

In recent years, deep learning has been successfully applied to diverse areas such as object detection/classification in images, and researchers have conducted several studies on fire detection to explore whether and how deep learning can improve performance [16–19].

The DNN-based approach has several differences from conventional vision-based fire detection. Because the visual features of fire are automatically explored in the multi-layered convolutional neural network (CNN) by training, the effort to identify proper handcrafted features is unnecessary. In addition, the detector/classifier can be obtained by the training simultaneously in the same DNN. Therefore, the appropriate network structure becomes more important with an efficient training algorithm. In addition, DNN training requires a large amount of data for exploring features and determining classifier/detector that exploits them.

Sebastien [17] proposed a fire detection method based on CNN feature extraction, and followed by a multilayer perceptron (MLP)-type classifier. Zhang et al. [18] also proposed a CNN-based fire detection method operated in a cascaded fashion; at first, the global image-level fire classifier was used, then a fine-grained patch classifier for precisely localizing the fire. Muhammad et al. also proposed a fine-tuned CNN fire detector [19], and recently developed an efficient CNN architecture for fire detection, localization, and semantic understanding of the scene of the fire [20].

Xie et al. [21] exploited a simple region of interest (ROI) detection method using motion-flicker-based dynamic features to try to finely decide the ROI whether it is a fire or not by a CNN-based classifier. Yang et al. [22] proposed a lightweight fire detection inspired by MobileNet. Li et al. [23] also adopted object detection models including Faster R-CNN, R-FCN, SSD, and YOLO v.3 for fire detection. Furthermore, Jadon et al. [24] proposed another lightweight real-time fire and smoke detection model, which they named Firenet.

Even though DNN showed superior image-level fire classification performance against traditional computer vision approach, locating objects has been another problem as, for example, in Zhang's cascade approach. The deep object detection model is the right way to simultaneously solve both problems of fire classification and localization, as evidenced by Li's method.

In DTA [3], a Faster R-CNN object detection model was also adopted to localize SRoFs. Then, the dynamic behavior of fire extracted by long short-term memory (LSTM) [25] in the localized regions was exploited in the short-term fire decision. Thereafter, to make the decision reliable, the short-term decisions were then combined for a final fire decision over a long-term period by majority voting. Note that the DTA method exploits both spatial and short-term temporal features, and uses them for fire decision in an ensemble fashion. However, it was not enough because only a limited amount of visual information included in video clips was exploited, and the environmental information from other sensors for fire decision was not included. In addition, the specific domain knowledge where the fire detection system operates was hard to incorporate with the previous DTA method.

3. Proposed Method

3.1. Network Architecture and Timing Diagram

We proposed a deep learning-based fire detection method, which imitates the human fire detection process, called DTA [3]. Using this method, we could discriminate fire or smoke from an evening glow, clouds, and chimney smoke, which are difficult to differentiate from still images. In this paper, the new extended architecture (DTAIF) can be divided into five components as shown in Figure 1.

The first three sections are identical to DTA, i.e., Faster R-CNN for fire object detection in the video frames in the first section (Refer Section 3.2), the learned spatial features accumulation by LSTM and short-term fire decision in the second section, and the majority voting stage for long-term fire decision in the third section (Refer Section 3.4).

In the extended approach of DTAIF, we added a location classifier (Refer Section 3.3) and smoke detector as the fourth component. The location classifier was based on the extracted features from the backbone ResNet152 structure of Faster R-CNN, which also provides smoke detection results. The categories of the locations examined are "building," "traffic (car, train, ship)," "street," "forest or mountain," "other places (bonfire, garbage, and so on)," and "irrelevant place for fire (sky, sea, and so on)." Once it was established that the location was one that has a probability of fire (i.e., all locations except "irrelevant place for fire"), the input was treated as a dangerous place in the corresponding node of the Bayesian net.

Location identified in every frame but the averaged probability of each location was taken throughout the video. The smoke detection state was "*yes*" if more than half the frames include smoke objects during the period. In the fifth section of Bayesian net (Refer Section 3.5), all available information was combined to make the final decision regarding the presence of fire.

There were two groups of information sources in this section of the Bayesian net: visual analysis of a video clip and other sensors. The results of the majority voting for a long-term decision, the location classifier, and the result of the smoke detector from Faster R-CNN were based on the information obtained from video analysis. Many different sensors may have aided in fire detection, measuring temperature, wind speed, and humidity. However, the temperature is weakly related with the fire occurrence, and the wind speed affects the spreading of a fire after breaking. Therefore, we only considered humidity in our simple Bayesian network. Note that the humidity is not only a critical factor of a fire, but also gives an important clue to prevent false fire-decision from cloud, haze, mist, or fog. Because we did not gather enough concurrent data to learn the Bayesian net, the structure and its associated conditional probability tables were reasonably presumed. Additionally, the specific domain knowledge where the fire detection system operates could be successfully considered in the Bayesian net.

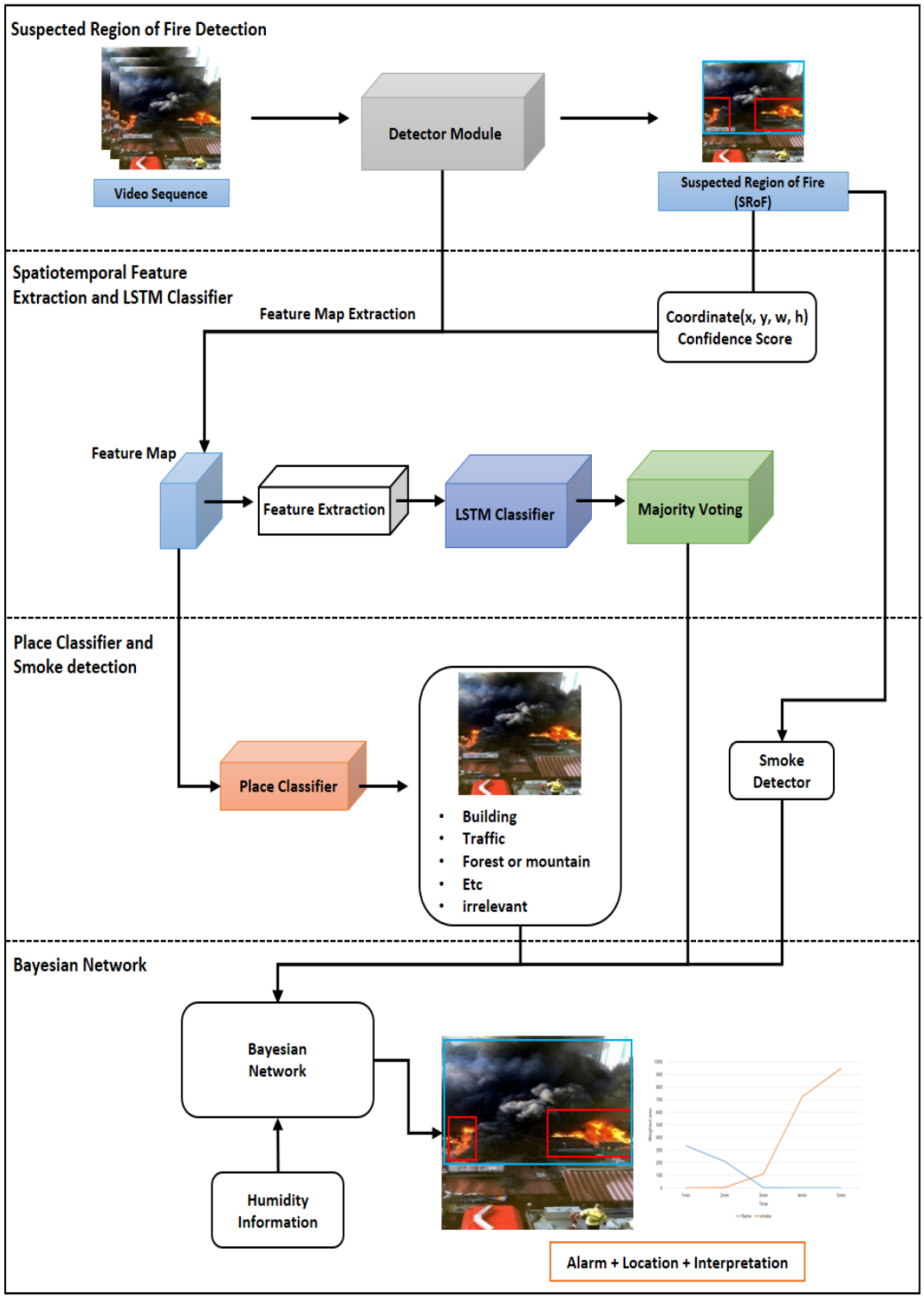

Figure 1. The proposed network architecture [26].

The timing of the proposed method was important to install in a real system and could be adjusted depending on the situation where it operates. Figure 2 presents a timing diagram that shows the decision period for each block. All timing periods were the same as in the previous paper [3], except the timing for Bayesian inference. The fire objects with smoke were detected for each frame of video, and the CNN spatial features of Faster R-CNN were temporally accumulated for a period to make a short-term decision. (Refer

Section 3.2) The short-term fire decision for every T_{LSTM} was involved in the majority voting process for every time-period T_{vot}., which implies that the fire decision based only on visual analysis of a video clip was repeated for every T_{vot}. Because all results of the averaged location classifier and the long-term smoke detector were available at every T_{vot}, the final decision in our Bayesian net was made at the same time interval as T_{vot}. In other words, $T_{vot} = T_{Bayes.}$

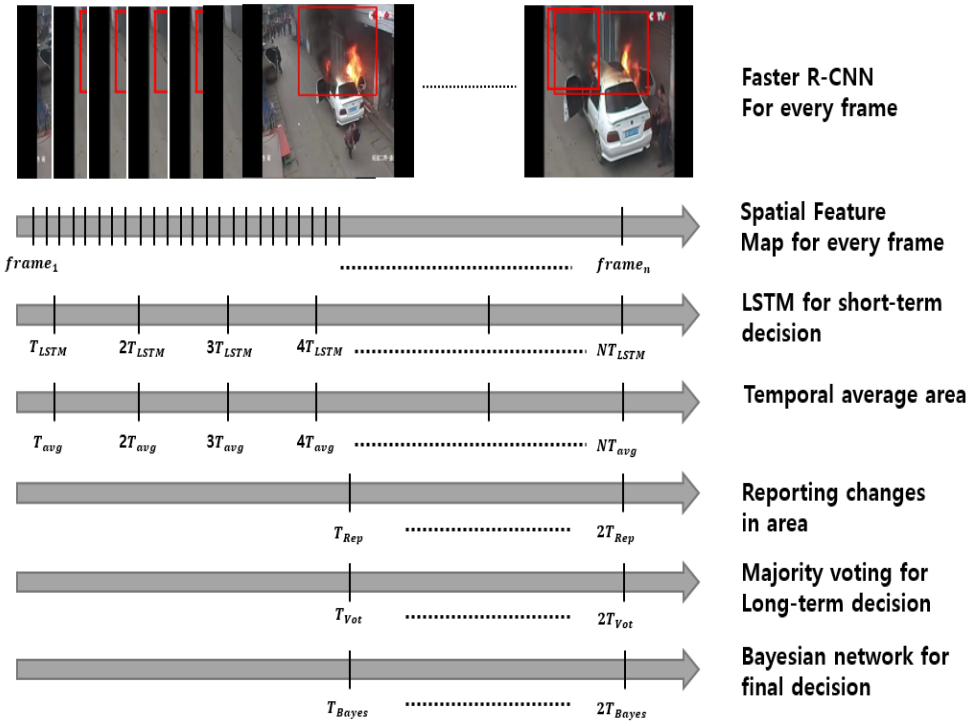

Figure 2. Timing diagram of fire detection [26].

In addition, the areas of flame and smoke were calculated for each frame and smoothed by taking the average over T_{ave}. Changes in the average area in video frames were reported with other additional information at every time instant T_{rep}. For convenience, we assumed that $T_{rep} = T_{vot} = T_{Bayes.}$ The details of this interpreter of long-term dynamic behavior were identical to our previous method [3], except that here the identified location was accounted for [26]. The exact timing for each block depended on the situation where the fire detection system with our proposed method operates, and could be adjusted accordingly.

3.2. Faster R-CNN for Fire Object Detection

As in DTA, Faster R-CNN results in the bounding boxes of flame, smoke, and non-fire regions in an image, as shown in Figure 3. The ConvNet backbone ResNet152 extracts visual features for location classification of the fire scene. The samples of training images are shown in Figure 3, which includes flame, smoke, and non-fire objects. As in Figure 4, there are some hard-negative examples included in non-fire objects that resemble real fire objects, such as sunset, misty street, and cloud. The bounding boxes enclosing flame and smoke objects were treated individually as SRoFs. A non-fire object may have been enclosed by a bounding box if it was a hard-negative example, otherwise, it included the whole frame as a bounding box.

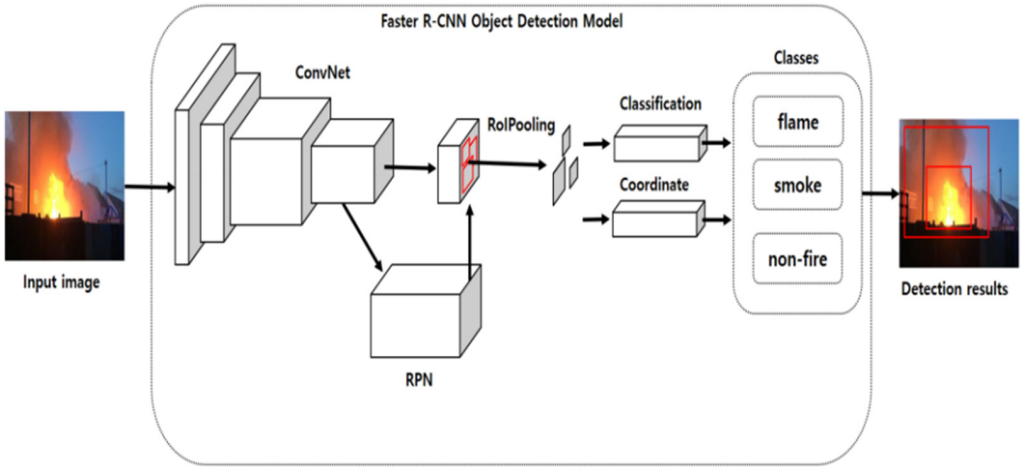

Figure 3. Faster R-CNN structure for fire detection [26].

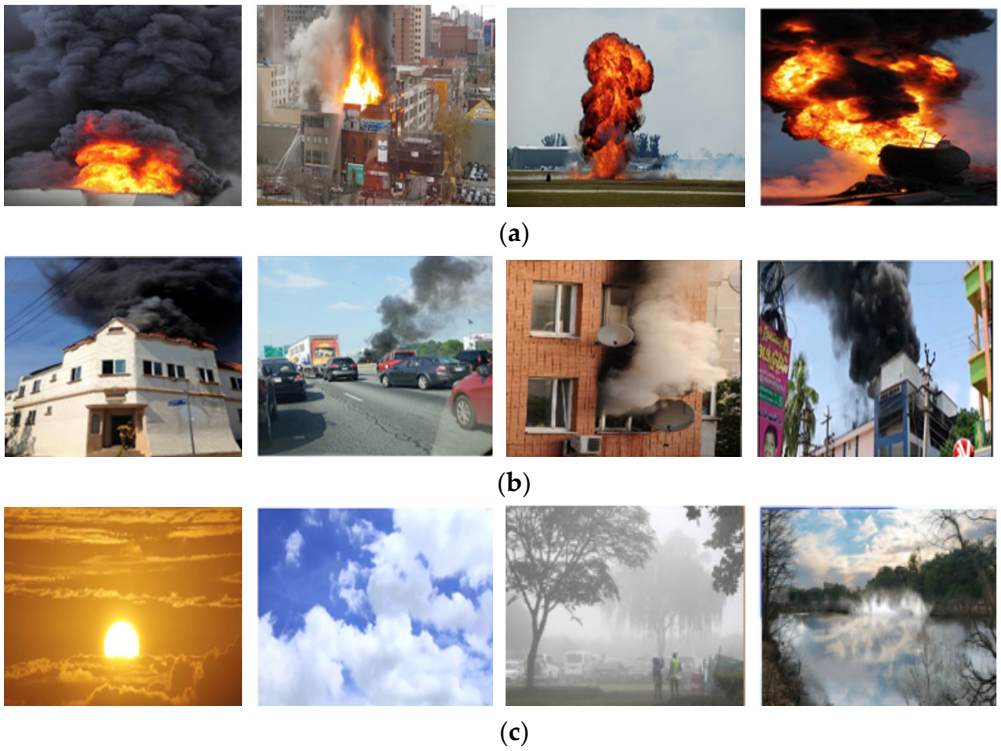

Figure 4. Sample images for training Faster R-CNN; (**a**), (**b**), and (**c**) are fire, smoke, and non-fire images, respectively [3].

Smoke objects were detected at every frame and merged to decide the state of smoke during $T_{Bayes.}$. The smoke detection result was "$S = yes$" if the number of frames which detect the smoke object with a threshold of confidence of 0.5 was more than half of the total frames during $T_{Bayes.}$.

3.3. Spatiotemporal Feature Extraction and Place Classifier

The coordinates of the bounding boxes that enclose flame, smoke, and non-fire objects were projected on the $n \times n \times d$ activation map to extract the spatial features. Here, we extracted them in the last layer of ResNet152 which is a backbone of object detector (Here, $d = 1024$) [3]. For detected object regions of SRoF the weighted global average pooling (GAP) was taken on the bounding boxes, but just GAP was taken on the whole image when

there is no SRoF. The d-dimensional feature for successive frames were fed into LSTM to explore temporal behavior of video clip.

As previously mentioned, the place classification for location identification shares the same feature maps of the last layer of ResNet152. Because the location could be captured in a whole image, however, we took GAP for each feature map and constructed a d-dimensional feature to classify the location of the fire, as shown in Figure 5. The SoftMax classifier was used for every frame and the average probability of a particular location among six categories was calculated for a given set of video frames to make a stable scene-level decision. In our Bayesian net, we did not use all six classes of locations, but classified them into fire-prone or fire-irrelevant locations. Note that the place classifier was constructed efficiently by sharing the feature maps of Faster R-CNN and could provide stable classification by taking ensemble averages over consecutive frames in a scene.

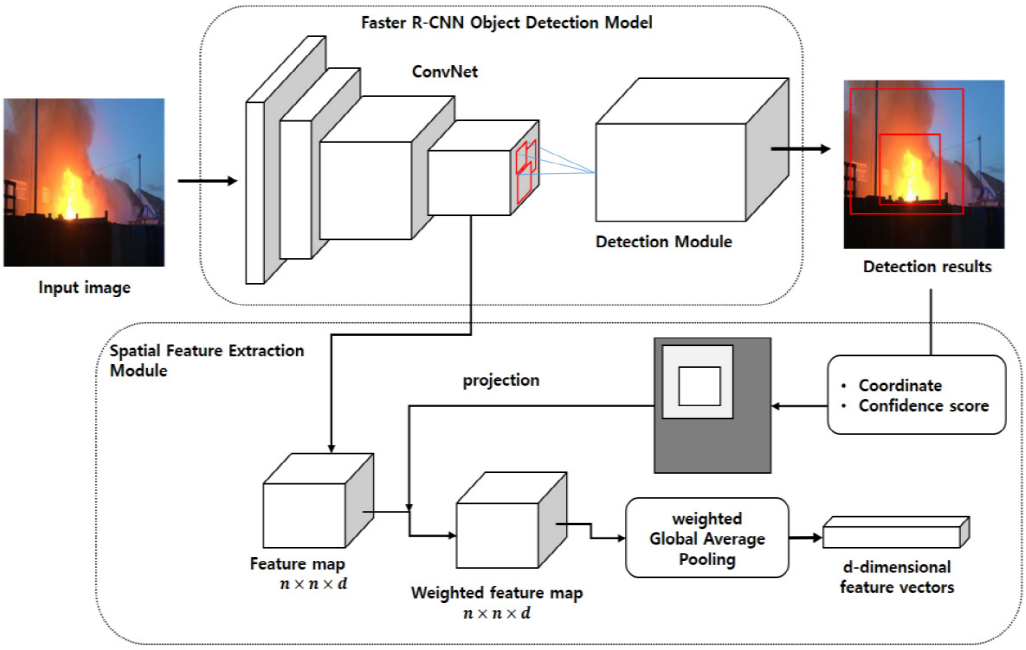

Figure 5. The spatiotemporal feature extraction from Faster R-CNN [26].

3.4. LSTM for Short-Term and Majority Voting for Long-Term Fire Decision

We aggregated the changes in the extracted spatial features using two-stage LSTM network in a short period, and determined whether it was a fire or a non-fire object at every T_{LSTM} [3]. Figure 6 shows the part of LSTM network used to accumulate and decide a fire in a short-term period.

Because the data for training the LSTM network should be videos clips differently from those for Faster R-CNN. We collected video clips of fire and non-fire, and constructed a dataset, from which many frames were extracted and added for training Faster R-CNN. The consecutive d-dimensional spatial features calculated from the trained Faster R-CNN for a video clips were prepared as input streams for the LSTM training.

As the same way as in DTA [3], the LSTM network reflects a temporal behavior in a short-term period, such as a person's quick glance, and its decisions are integrated by taking majority voting to make a reliable fire decision in a long-term period T_{vot}. The fire decision in this stage was made by the ratio of fire to non-fire decisions during the time window T_{vot}, and fed into Bayesian net as a prior probability of fire $\mathrm{P}(F)$.

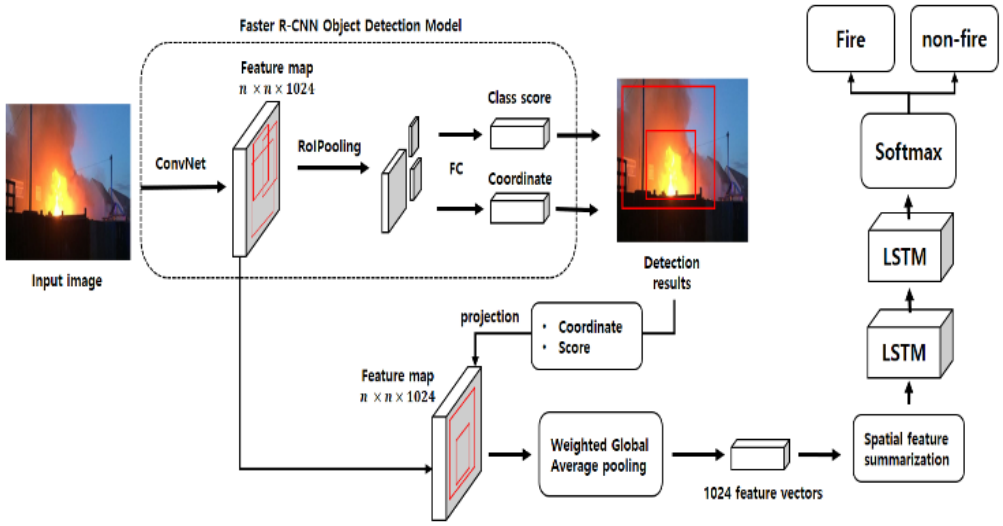

Figure 6. The LSTM network for fire decision [26].

3.5. Bayesian Network for Incorporation of all Information into Final Decision

In general, people determine the existence of a fire not only based on its spatial and temporal behavior, but also about the place where it happens and environmental factors such as humidity and other weather conditions. For robust fire decisions, various types of data measured from diverse sensors can be combined with video data. We propose to use a Bayesian network to combine all this information to improve fire decisions.

According to the types of sensors and the situations where they operate, however, there could be many domain-specific Bayesian networks. Therefore, we confined to the outdoor fires where only crude humidity information was assumed to be known. Due to the lack of domain-specific data, we tried to extract all visual information from video frames as much as possible, including the long-term fire decision, the place category, and the existence of smoke. If there was a sufficient number of concurrent data, the Bayesian network could also be trained by the machine learning algorithm. Unfortunately, we did not have such data, so a simple Bayesian net was constructed as an example to validate our suggestion.

In general, the Bayesian network can also reflect the domain-specific knowledge that is essential to an accurate determination of fire. In addition, location can be finely identified at the point of interest (POI) level within the specific scene where a fire-like event always occurs. For example, a factory chimney that produces smoke can be visually identified and properly reflected in the construction of a domain-specific Bayesian network.

In this paper, however, we should have exploited generic domain knowledge to design a simple Bayesian net with four nodes, as an example shown in Figure 7. The nodes were "Fire (F)," "Cloud/Smoke (S)," "Dangerous Place (D)," and "Humidity (H)." Here, to simplify the conditional probability table, we assumed that nodes F, S, and D, has binary states, such that "$F = yes\ or\ no$," "$S = yes\ or\ no$," and "$D = yes\ or\ no$". The exception is H, which has ternary states, including "*Wet*, *Normal*, or *Dry*".

A priori probability of fire, P(F) is decided from the ratio of majority voting. Note that the ratio of the summed probability of SoftMax at the end of the LSTM stage could be alternatively taken for P(F).

If the maximum averaged probability of location is for "building," "traffic (car, train, ship)," "street," "forest or mountain," "etc. (bonfire, garbage, and so on)," then $D = yes$. In general, almost all location categories are prone to fire except "irrelevant place for fire (sky, sea, and so on)." If the Faster R-CNN detects smoke in more than half the frames during t_{vot}, then $S = yes$. All these state values are obtained from video analysis. One may obtain the state value from a humidity sensor, but there was no concurrent sensor data to the

video in our experiment. So, the state of *H* was set to "Normal" whenever no additional description is given in the experiment.

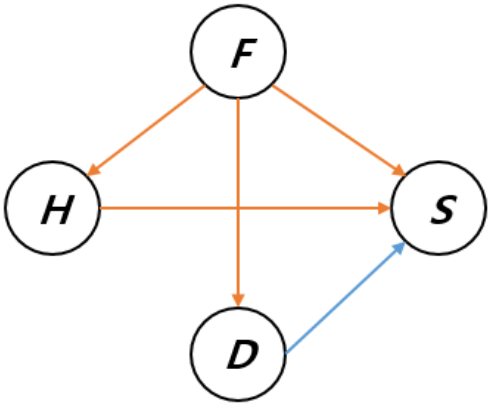

Figure 7. Example of Bayesian Network [26].

Examples of conditional probability tables are assumed as in Tables 1–3. We attempted to set reasonable conditional probabilities according to the general situations where it happens. Tables 1 and 2 show the probabilities of humidity and location conditioned on the fire decision based on the analysis of visual input from a video clip. Table 3 shows the conditional probability of smoke. In the table, the fog on the sea or lake is imagined, and the fire in wet conditions well produces smoke in rows (1) and (2), respectively, as examples. If the situation is ambiguous, then the equal probabilities are assumed in the table. Note that if the concurrent data were sufficiently collected, the structure and corresponding conditional probability table could be determined by training.

Table 1. Conditional Probability P(*H*/*F*) [26].

H/F	Dry	Normal	Wet
No	0.2	0.3	0.5
Yes	0.5	0.3	0.2

Table 2. Conditional Probability P(D/*F*) [26].

D/F	No	Yes
No	0.7	0.3
Yes	0.3	0.7

Table 3. Conditional Probability P(*S* | *F*, *D*, *H*) [26].

F	H	D	S = No	S = Yes
No	*Dry*	*No*	0.9	0.1
	Dry	*Yes*	0.5	0.5
	Normal	*No*	0.7	0.3
	Normal	*Yes*	0.5	0.5
	Wet	*No*	0.4	0.6
	Wet	*Yes*	0.1	0.9
Yes	*Dry*	*No*	0.8	0.2
	Dry	*Yes*	0.5	0.5
	Normal	*No*	0.6	0.4
	Normal	*Yes*	0.5	0.5
	Wet	*No*	0.4	0.6
	Wet	*Yes*	0.3	0.7

Once the state of each node was determined, the posterior probability of root node "fire" could be calculated by

$$P(F|H,D,S) = \frac{P(F,H,D,S)}{P(H,D,S)} = \frac{P(F,H,D,S)}{\sum_{F=yes/no} P(F,H,D,S)'} \quad (1)$$

where

$$P(F,H,D,S) = P(H|F)P(D|F)P(S|F,H,D)P(F) \quad (2)$$

where represents the probability of "*Fire*" determined from the ratio of majority voting. Our simple Bayesian network is designed merely to show the effectiveness of information fusion for better fire decision.

4. Experimental Results and Discussions

Our method cannot use end-to-end training, because there are non-differentiable operations such as majority voting in the composite network. The Faster R-CNN, place classifier, and LSTM stages should be separately trained in the method. We used the libraries of 'Python 3.5', 'OpenCV 3.0', and 'TensorFlow 1.5' for designing deep learning models, and 'pgmpy' for constructing Bayesian net.

4.1. Training Faster R-CNN and Its Performance

The dataset for training and test the Faster R-CNN was constructed by 81,810 still images, including 25,400 flame images and 25,410 smoke images collected from several data sources including YouTube video clips, the previous works [7,27–29], and the Flickr-fire dataset. There were 31,000 non-fire images included in the dataset. The images were divided into 70% for training, 10% for validation, and 20% for test data. For training, the positive data were augmented by a horizontal flip of bounding a bounding box. Note that a fire should have the consistent shape after taking augmentation. Table 4 shows the training parameters for Faster R-CNN.

Table 4. Training Parameters of Faster R-CNN [26].

Parameter	Method
Iteration	250,000
Step size	70,000/150,000
Weight decay	0.0004
Learning rate	0.001
Learning rate decay	0.1
Batch size	1
Pre-train weight	ResNet152(ImageNet)
RPN mini-batch size	256
OHEM sample size	256

The performance of the Faster R-CNN was measured by mean average precision (MAP) and is shown in Table 5, which is a better result than the previous paper due to the elaborate training method such as hard negative example mining. These include several false positive detections for clouds, chimney smoke, lighting lamp, steam, etc., which are almost undetectable without considering the temporal characteristics as shown in Figure 8.

Table 5. mAP of Faster R-CNN [26].

mAP	Flame	Smoke	Non-fire
88.76	88.92	88.75	88.61

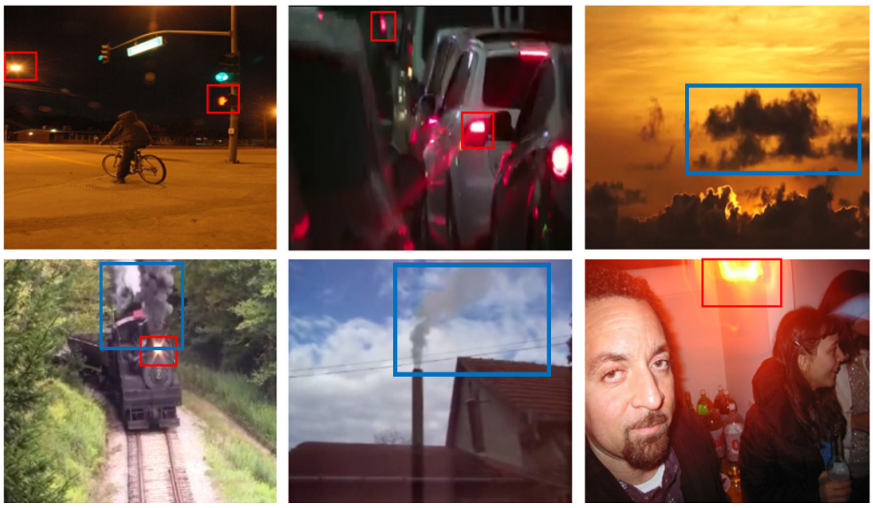

Figure 8. False positive results of Faster R-CNN fire detection. The red color box is flame prediction blue color box is smoke prediction [26].

4.2. Training LSTM and Its Performance before and after Majority Voting

For the training of LSTM following Faster R-CNN, we collected 1709 video clips from YouTube, comprising 872 clips of fire, and 837 of non-fire, where a huge number of positive and negative frames were included. Figure 9 shows sample still shots of the video dataset.

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

Figure 9. Sample still shots including flame, smoke, and non-fire objects take from video clips for LSTM training [26]. The (**a**) images are taken from videos of fire accident, while the (**b**) images are from non-fire video clips.

For training the LSTM network the video clips were divided into 60 consecutive frames with 30 frames of overlap, which implies that the LSTMs captured every 2 s of dynamic behavior of successive frames, but the decision in LSTM network was made for every second, i.e., $T_{LSTM} = 1$ s, if 30 frames per a second were assumed. As the same way as the training of Faster R-CNN, 70% of the data were selected for training, 10% for validation, and 20% for testing. From each video clip, we obtained d-dimensional feature for every consecutive frame as stated in Section 3.3, and fed into LSTM. Note that Faster R-CNN may not properly detect fire objects in some frames even though the frames are from a fire video sequence. In such cases, instead of WGAP, the GAP was taken for the undetected frames and used to make the d-dimensional feature for LSTM input, just as it would have been in a "non-fire" scenario.

Table 6 shows the parameters of LSTM training, and the performance of the test data is shown in Table 7 depending on the number of memory cells in LSTM.

Table 6. Training Parameters of LSTM [26].

Parameter	Method
Input size	1024
Time Step	60
LSTM cell unit	128/256/512/1024
Learning rate	0.0001
Weight decay	0.0004
Batch size	256
Epoch	200
Weight initialization	Xavier
Dropout	0.5

Table 7. Performance of test video clips for LSTM hidden cell unit [26].

Method	Accuracy (%)
Hidden cell unit = 128	95.36
Hidden cell unit = 256	96.52
Hidden cell unit = 512	97.43
Hidden cell unit = 1024	97.07

To compare the performance with other methods, we evaluated the results on the public dataset in [12]. Unfortunately, however, the Foggia dataset includes a video clip that lasts only 4 s, which made it hard to completely validate the proposed method that works well for long video clips. So, there were only three times of fire decision in LSTM to take the majority voting due to the short clip, i.e., $T_{vot} = 3T_{LSTM}$.

Table 8 compares our model's performance against other methods. Our method provided the most accurate fire detection when there were 512 memory cells in the LSTM network.

Table 8. Performance comparison with other methods [26].

Methods	False Positive (%)	False Negative (%)	Accuracy (%)
Proposed method (hidden cell unit = 512)	3.04	1.72	95.00
Proposed method ($T_{vot} = 3T_{LSTM}$)	2.46	1.45	97.68
Khan Muhammad et al. [19]	0.00	0.14	95.86
Foggia et al. [12]	11.67	0.00	93.55
De Lascio et al. [30]	13.33	0.00	92.86
Habibugle et al. [31]	5.88	14.29	90.32
Rafiee et al. (YUV) [32]	17.65	7.14	74.20
Celik et al. [6]	29.41	0.00	83.87
Chen et al. [8]	11.76	14.29	87.1
Yakun xie et al. [22]	2.33	0.84	97.94
Arpit Jadon et al. [24]	1.23	2.25	96.53

In Table 8, the proposed method after majority voting results in 97.68% of accuracy, 2.64% of the false-positive rate, and 1.45% of false-negative rate, is inferior to other CNN-based methods proposed by Khan Muhammad et al. [19], Yakun xie et al. [21], and Arpit Jadon et al. [24]. The result implies the majority voting after LSTM fire decisions in this Foggia dataset is not proper to improve the performance due to the short duration of video clip.

4.3. Training Place Classifier and Its Performance

For training and testing of the location classifier, we selected data from well-known datasets, including the Intel scene classification challenge dataset [33], the scene under-

standing (SUN) dataset [34], the ImageNet dataset [35], among others [36–38]. We also included frames from 70 video clips from YouTube that contained a sunset, a bonfire, and a dumping ground. The total number of images was 330,000, of which about 55,000 images were included for each place category. For the place classifier, 70% of the data were selected for training, 10% for validation, and 20% for testing. Figure 10 shows typical images for location categorization. As explained in Section 3.3, 1024 dimensional features were extracted to feed into the softmax classifier for place classifier. Table 9 shows the hyper parameters for training the place classifier.

(**a**)

(**b**)

Figure 10. Typical images for six place classifiers. (**a**) are building, forest or mountain, and street. (**b**) are traffic, etc., and irrelevant [26].

Table 9. Training Parameters of place classifier [26].

Parameter	Method
Input size	1024
Learning rate	0.01
Learning rate decay	0.01 (step:50, 150)
Weight decay	0.0004
Batch size	128
Epoch	250
Weight initialization	Xavier

The accuracy was 96.12% (of Top-1) and 98.24% (of Top-2) for six categories. The Top-1 accuracy of two classes—"fire-prone" and "irrelevant to fire"—was 98.93%, which is meaningful for deciding the binary states of D in our Bayesian net.

4.4. Experiemts including Bayesian Network

In order to evaluate the performance of Bayesian network, we took the Foggia dataset again and obtained all the visual information, including the results of majority voting, smoke detection, and place classification. The dataset does only contained video clips, and there was little information to infer the state of humidity in the visual contents. So, we set the state of humidity as "*Normal*" and performed the inference by Bayesian network. Table 10 summarizes the results.

Table 10. Performance comparison with other methods.

Methods	False Positive (%)	False Negative (%)	Accuracy (%)
Proposed method ($T_{vot} = 3T_{LSTM}$) [26]	2.46	1.45	97.68
Proposed method ($T_{Bayes} = T_{vot} = 3T_{LSTM}$) [26]	1.56	1.45	98.45
Khan Muhammad et al. [19]	0.00	0.14	95.86
Yakun xie et al. [21]	2.33	0.84	97.94
Arpit Jadon et al. [24]	1.23	2.25	96.53

The table represents that our simple Bayesian net increases 0.77% of accuracy, and reduces 0.90% of the false-positive rate. This implies the additional visual information combined with the Bayesian network, such as the place category and the existence of smoke, can help to improve the fire-detection performance, even though the humidity is set to "Normal".

To sufficiently validate the voting process and Bayesian network, we have collected 40 additional video clips from YouTube which have relatively long-playing times between 4.5 and 6.3 min. Table 11 shows an overview of the fire/non-fire video clips with summarized interpretations of the dynamic behaviors. The set of 11 non-fire video clips consists of fire-like sunset, cloud, and chimney smoke scenes that are easily false detected. Figure 11 shows samples of still shots from the video dataset, which may be useful to longer-term experiments.

Table 11. Collected video clips from YouTube [26].

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

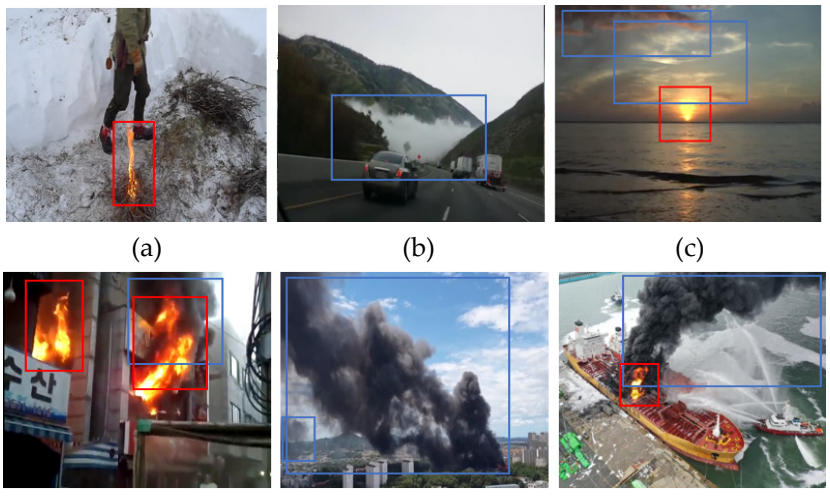

Figure 11. Bayesian network results for 6 cases [26]. The (**a**,**d**–**f**) images are taken from videos of fire, the (**b**,**c**) images are from non-fire video clips.

We picked arbitrarily six video clips as in Figure 11, and observed the changes in the

probability of fire before and after inference to check the effectiveness of our Bayesian net. In the experiment, the humidity state was presumed and set to "*Dry*"," Wet", or "*Normal*", depending on the video content after watching, because there was no information on humidity attached to each video clip.

Table 12 shows the change in the probabilities. All the results in the table show that a posteriori probability is improved for accurate decision except case 6, in which wrong place decision reduces the probability of fire after Bayesian inference. Note that the place was classified as "sea", which is irrelevant to fire. The finely categorized places, according to the domain where the fire detection works, for example, "ship", "harbor" or "open sea" in the case, can remedy the problem. Note that a posteriori probability after Bayesian inference can help to improve the fire decision in this example.

Table 12. Changes in the probability of fire before and after Bayesian inference [26].

Case	Fire(GT)	Set *H*	Visual Analysis			P(*F*|*D*,*H*,*S*)	Result
			P(F)	D	S		
(a)	*Yes*	*Dry*	*0.6*	*Yes*	*No*	0.897	*Yes*
(b)	*No*	*Wet*	*0.3*	*Yes*	*Yes*	0.055	*No*
(c)	*No*	*Normal*	*0.5*	*No*	*Yes*	0.364	*No*
(d)	*Yes*	*Normal*	*0.8*	*Yes*	*Yes*	0.903	*Yes*
(e)	*Yes*	*Normal*	*0.6*	*Yes*	*Yes*	0.777	*Yes*
(f)	*Yes*	*Normal*	*0.8*	*No*	*Yes*	0.595	*Yes*

For the collected dataset, we extended the period $T_{vot} = T_{Bayes}$, because the video clips were long enough. The accuracy depended on the length of the decision period, as shown in Table 13. In the experiment, we set humidity as *Normal* because there was no information about the humidity.

Table 13. Accuracy comparison depending on the decision interval and Bayesian network [26].

Decision Interval ($T_{vot} = T_{Bayes}$)	30 s	1 min	1 min 30 s	2 min	2 min 30 s	3 min
Majority voting	96.3%	97.8%	98.5%	99.4%	100%	100%
Bayesian net	97.6%	98.5%	99.3%	100%	100%	100%

Note that we obtained perfect accuracy using the majority voting method independent of Bayesian inference when the time interval is long enough. This implies that the ensemble of majority voting is as reliable as a human fire detecting, based on its long-time dynamic behavior. Furthermore, the additional Bayesian network is helpful to reduce the time interval for making perfect decisions. Not that the latency is an important factor, because early fire detection is critical not to spread a fire.

We want to reiterate that our process is similar to a human's fire decision. If a person were only to rely on the dynamic behavior of local visual information (such as the behavior in SRoF), although they would ultimately reach the correct conclusion, it would take significant time. On the other hand, however, a person accounted for the overall location circumstances where fires happen, along with other environmental information, meaning the conclusion could be reached faster.

Table 14 shows the overall analysis results, which includes the changes in the area of smoke (or steam) and flame measured in the number of pixels with temporal changes in the ratio of voting, the Bayesian inference result, the classified place category, and the result of smoke detection for video clips in Figure 12. In our experiment, we set $T_{vot} = T_{Bayes} = 1$ min and assume the presumable humidity states depending on the content of the video after watching.

Table 14. Experimental results for Figure 11 [26].

Figure	Humidity	Time	Ratio of Voting	Top-1 Place Classification	Smoke Detection	Bayesian Network Result	Area of Flame	Area of Smoke
(a)	Normal	1 min	fire: 0.93, non-fire: 0.07	building: 0.84	yes	fire: 0.968, non-fire: 0.032	43.2	834.8
		2 min	fire: 0.95, non-fire: 0.05	building: 0.89	yes	fire: 0.977, non-fire: 0.023	13.9	841.7
		3 min	fire: 0.92, non-fire: 0.08	building: 0.87	yes	fire: 0.964, non-fire: 0.036	48.7	947.5
		4 min	fire: 0.94, non-fire: 0.06	building: 0.76	yes	fire: 0.973, non-fire: 0.027	237.8	1238.5
(b)	Dry	1 min	fire: 0.87, non-fire: 0.13	building: 0.71	yes	fire: 0.975, non-fire: 0.025	53.4	449.2
		2 min	fire: 0.92, non-fire: 0.08	building: 0.75	yes	fire: 0.985, non-fire: 0.015	67.9	372.8
		3 min	fire: 0.95, non-fire: 0.05	building: 0.79	yes	fire: 0.991, non-fire: 0.009	294.7	691.3
		4 min	fire: 0.94, non-fire: 0.06	building: 0.78	yes	fire: 0.989, non-fire: 0.011	319.6	538.1
(c)	Wet	1 min	fire: 0.1 non-fire: 0.9	forest or mountain: 0.84	yes	fire: 0.075, non-fire: 0.925	0	682.9
		2 min	fire: 0.08, non-fire: 0.92	forest or mountain: 0.87	yes	fire: 0.059, non-fire: 0.941	0	685.7
		3 min	fire: 0.08, non-fire: 0.92	forest or mountain: 0.82	yes	fire: 0.059, non-fire: 0.941	0	694.6
		4 min	fire: 0.07, non-fire: 0.93	forest or mountain: 0.86	yes	fire: 0.051, non-fire: 0.949	0	688.4
(d)	Normal	1 min	fire: 0.13, non-fire: 0.87	irrelevant place: 0.83	yes	fire: 0.079, non-fire: 0.921	78.4	218.9
		2 min	fire: 0.08, non-fire: 0.92	irrelevant place: 0.85	no	fire: 0.047, non-fire: 0.953	23.7	0
		3 min	fire: 0.1, non-fire: 0.9	irrelevant place: 0.87	no	fire: 0.039, non-fire: 0.961	68.6	0
		4 min	fire: 0.1, non-fire: 0.9	irrelevant place: 0.94	no	fire: 0.039, non-fire: 0.961	87.4	0

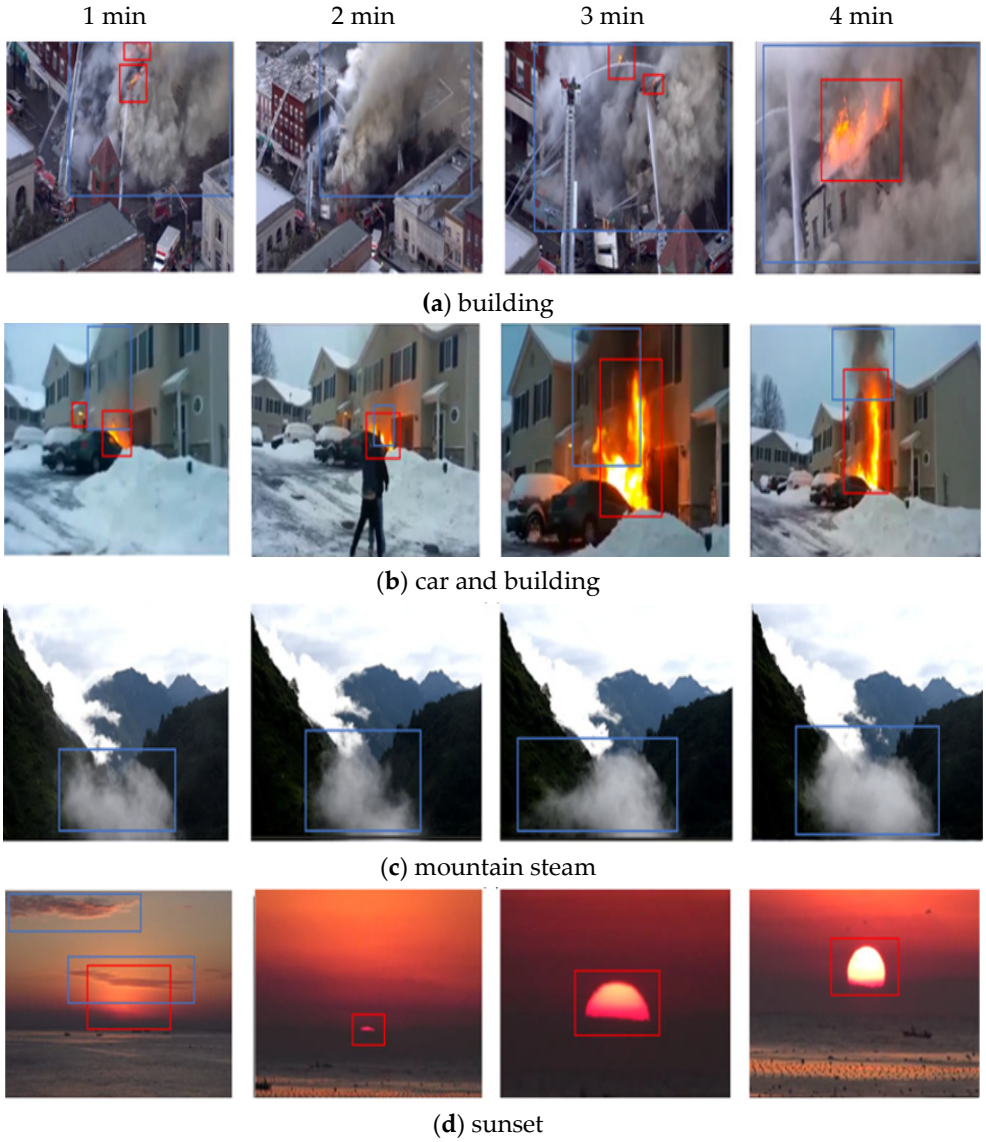

Figure 12. Sample still shots taken from video clips for the experiment [26]. The (**a**,**b**) images are fire videos, while the (**c**,**d**) images non-fire video clips.

5. Conclusions

We proposed a deep learning-based fire detection method, called DTAIF, which imitates the human detection process, one step further from DTA in our previous paper. We assumed that DTAIF can reduce erroneous visual fire detection by combining all the information from different sensors. As the same way as the method in the previous paper, Faster R-CNN is used to detect SRoFs, and LSTM accumulates the local features within the bounding boxes to decide a fire in a short-term period. Then, majority voting of the short-term decisions is taken to make the decision reliable in a long-term period. In addition, to make the final fire decision more robust, this paper proposed to use a Bayesian network that combines environmental information with the visual information. As an example, we have constructed a simple Bayesian network to combine the humidity state with all extractable visual information from video sequences including the place category, and the existence of smoke, and the long-term majority voting-based fire decision.

Our experiments showed that even a simple Bayesian network can improve the fire detection accuracy compared to our previous video-based method and can achieve state-of-art performance for the public dataset. The proposed scheme with the Bayesian network also reduced the latency for perfect fire decisions when compared against our previous method. We believe that the proposed fire detection approach is so general that it can be applied for both indoors and outdoors, even though our example Bayesian network was confined to outdoor fire in the experiment.

We did not consider time complexity and practical implementation in the work. For example, a recent object detection method, such as YOLOv2 [39], instead of Faster R-CNN, can be chosen for further work to make more fast and effective realization of DTAIF. Furthermore, there are several issues to improve our work including learning Bayesian network based on dataset, and graph convolutional network (GCN) to combine domain knowledge instead of Bayesian net.

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. and J.L.; Writing review and editing, J.L. All authors have read and agreed to the published version of the manuscript.

Funding: This research was supported by Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Education (No. 2019R1A6A1A09031717).

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The original contributions presented in the study are included in the article [3,26], further inquiries can be directed to the corresponding author/s.

Acknowledgments: Byoungjun Kim and Joonwhoan Lee designed the study, and performed the experiments and data analysis, and wrote the paper.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Chi, R.; Lu, Z.-M.; Ji, Q.-G. Real-time multi-feature based fire flame detection in video. *IET Image Process.* **2017**, *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. Kim, B.; Lee, J. A Video-Based Fire Detection Using Deep Learning Models. *Appl. Sci.* **2019**, *9*, 2862. [CrossRef]
4. Qiu, T.; Yan, Y.; Lu, G. An auto adaptive edge-detection algorithm for flame and fire image processing. *IEEE Trans. Instrum. Meas.* **2011**, *61*, 1486–1493. [CrossRef]
5. Liu, C.-B.; Ahuja, N. Vision based fire detection. In Proceedings of the 17th International Conference on Pattern Recognition, Cambridge, UK, 26 August 2004; Volume 4, pp. 134–137.
6. 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]

7. 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]
8. Chen, T.-H.; Wu, P.-H.; Chiou, Y.-C. An early fire-detection method based on image processing. In Proceedings of the 2004 International Conference on Image Processing, Singapore, 24–27 October 2004; Volume 3, pp. 1707–1710.
9. Wang, T.; Shi, L.; Yuan, P.; Bu, L.; Hou, X. A new fire detection method based on flame color dispersion and similarity in consecutive frames. In Proceedings of the Chinese Automation Congress (CAC), Jinan, China, 20–22 October 2017; pp. 151–156.
10. Borges, P.; Izquierdo, E. A probabilistic approach for vision-based fire detection in videos. *IEEE Trans. Circuits Syst. Video Technol.* **2010**, *20*, 721–731. [CrossRef]
11. 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]
12. 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]
13. Dimitropoulos, K.; Barmpoutis, P.; Grammalidis, N. Spatio-temporal flame modeling and dynamic texture analysis for automatic video-based fire detection. *IEEE Trans. Circuits Syst. Video Technol.* **2014**, *25*, 339–351. [CrossRef]
14. Lin, G.; Zhang, Y.; Jia, Y.; Xu, G.; Wang, J. Smoke detection in video sequences based on dynamic texture using volume local binary patterns. *KSII Trans. Internet Inf. Syst. (TIIS)* **2017**, *11*, 5522–5536.
15. Favorskaya, M.; Pyataeva, A.; Popov, A. Verification of smoke detection in video sequences based on spatio-temporal local binary patterns. *Procedia Comput. Sci.* **2015**, *60*, 671–680. [CrossRef]
16. Ojo, J.A.; Oladosu, J.A. Effective Smoke Detection Using Spatial-Temporal Energy and Weber Local Descriptors in Three Orthogonal Planes (WLD-TOP). *J. Comput. Sci. Technol.* **2018**, *18*, 35–47. [CrossRef]
17. 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; pp. 877–882.
18. Zhang, Q.; Xu, J.; Xu, L.; 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.
19. 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**, *49*, 1419–1434. [CrossRef]
20. 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.
21. Xie, Y.; Zhu, J.; Cao, Y.; Zhang, Y.; Feng, D.; Zhang, Y.; Chen, M. Efficient Video Fire Detection Exploting Motion-Flicker-Based Dynamic Features and Deep Static Features. *IEEE Access* **2020**, *8*, 81904–81917. [CrossRef]
22. Yang, H.; Jang, H.; Kim, T.; Lee, B. Non-Temporal Lightweight Fire Detection Network for Intelligent Surveillance Systems. *IEEE Access* **2019**, *7*, 169257–169266. [CrossRef]
23. Li, P.; Zhao, W. Image fire detection algorithms based on convolutional neural networks. *Case Stud. Therm. Eng.* **2020**, *19*, 100625. [CrossRef]
24. 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.
25. Hochreiter, S.; Schmidhuber, J. LSTM can solve hard long-time lag problems. *Adv. Neural Inf. Proc. Syst.* **1997**, 473–479. [CrossRef]
26. Kim, B. Video Fire Detection Using Deep Learning Model and Bayesian Network. Ph.D. Thesis, Jeonbuk National University, Jeonju, Korea, 2021.
27. 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, Washington, DC, USA, 26–29 August 2015; pp. 95–102.
28. 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]
29. The Flickr-Fire Dataset. Available online: http://conteudo.icmc.usp.br/pessoas/junio/DatasetFlicker/DatasetFlickr.htm (accessed on 1 November 2020).
30. 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, 22–24 October 2014; Springer: Vilamoura, Portugal, 2014.
31. 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]
32. 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, Shanghai, China, 11–13 March 2011; Volume 3, pp. 262–265.
33. Intel Scene Classification Challenge Dataset. Available online: https://www.kaggle.com/spsayakpaul/intel-scene-classification-challenge (accessed on 1 November 2020).

34. Xiao, J.; Hays, J.; Ehinger, K.A.; Oliva, A.; Torralba, A. Sun database: Large-scale scene recognition from abbey to zoo. In Proceedings of the CVPR, San Francisco, CA, USA, 15–17 June 2010; pp. 3485–3492.
35. Deng, J.; Dong, W.; Socher, R.; Li, L.J.; Li, K.; Fei-Fei, L. Imagenet: A large-scale hierarchical image database. In Proceedings of the 2009 IEEE Conference on Computer Vision and Pattern Recognition, Miami, FL, USA, 20–25 June 2009; pp. 248–255.
36. Zhou, B.; Lapedriza, A.; Xiao, J.; Torralba, A.; Oliva, A. Learning deep features for scene recognition using places database. In Proceedings of the Advances in Neural Information Processing Systems, Palais des Congrès de Montréal, Montréal, QC, Canada, 8–13 December 2014; pp. 487–495.
37. Mittal, G.; Yagnik, K.B.; Garg, M.; Krishnan, N.C. Spotgarbage: Smartphone app to detect garbage using deep learning. In Proceedings of the 2016 ACM International Joint Conference on Pervasive and Ubiquitous Computing, Heidelberg, Germany, 12–16 September 2016; pp. 940–945.
38. Cene—An Image Organization Application. Available online: https://github.com/lecritch/Cene-Image-Classification (accessed on 1 November 2020).
39. Saponara, S.; Elhanashi, A.; Gagliardi, A. Real-time video fire/smoke detection based on CNN in antifire surveillance systems. *J. Real-Time Image Proc.* **2020**, *18*, 889–900. [CrossRef]