



Article

# Visual Intelligence in Smart Cities: A Lightweight Deep Learning Model for Fire Detection in an IoT Environment

Muhammad Nadeem <sup>1</sup>, Naqqash Dilshad <sup>2</sup>, Norah Saleh Alghamdi <sup>3</sup>, L. Minh Dang <sup>4</sup>, Hyoung-Kyu Song <sup>4</sup>, Junyoung Nam <sup>1</sup> and Hyeonjoon Moon <sup>1,\*</sup>

<sup>1</sup> Department of Computer Science and Engineering, Sejong University, Seoul 05006, Republic of Korea; 22110104@sju.ac.kr (M.N.); nampang@sju.ac.kr (J.N.)

<sup>2</sup> Department of Convergence Engineering for Intelligent Drone, Sejong University, Seoul 05006, Republic of Korea; dilshadnaqqash@sju.ac.kr

<sup>3</sup> Department of Computer Sciences, College of Computer and Information Sciences, Princess Nourah bint Abdulrahman University, P.O. Box 84428, Riyadh 11671, Saudi Arabia; nosalghamdi@pnu.edu.sa

<sup>4</sup> Department of Information and Communication Engineering and Convergence Engineering for Intelligent Drone, Sejong University, Seoul 05006, Republic of Korea; minhdl@sejong.ac.kr (L.M.D.); songhk@sejong.ac.kr (H.-K.S.)

\* Correspondence: hmoon@sejong.ac.kr

**Abstract:** The recognition of fire at its early stages and stopping it from causing socioeconomic and environmental disasters remains a demanding task. Despite the availability of convincing networks, there is a need to develop a lightweight network for resource-constraint devices rather than real-time fire detection in smart city contexts. To overcome this shortcoming, we presented a novel efficient lightweight network called FlameNet for fire detection in a smart city environment. Our proposed network works via two main steps: first, it detects the fire using the FlameNet; then, an alert is initiated and directed to the fire, medical, and rescue departments. Furthermore, we incorporate the MSA module to efficiently prioritize and enhance relevant fire-related prominent features for effective fire detection. The newly developed Ignited-Flames dataset is utilized to undertake a thorough analysis of several convolutional neural network (CNN) models. Additionally, the proposed FlameNet achieves 99.40% accuracy for fire detection. The empirical findings and analysis of multiple factors such as model accuracy, size, and processing time prove that the suggested model is suitable for fire detection.

**Keywords:** disaster management; fire monitoring; fire classification; deep learning; MobileNet; lightweight model; internet of things; smart cities



**Citation:** Nadeem, M.; Dilshad, N.; Alghamdi, N.S.; Dang, L.M.; Song, H.-K.; Nam, J.; Moon, H. Visual Intelligence in Smart Cities: A Lightweight Deep Learning Model for Fire Detection in an IoT Environment. *Smart Cities* **2023**, *6*, 2245–2259. <https://doi.org/10.3390/smartcities6050103>

Academic Editors: Faisal Jamil and Zeinab Shahbazi

Received: 25 July 2023

Revised: 19 August 2023

Accepted: 21 August 2023

Published: 28 August 2023



**Copyright:** © 2023 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

## 1. Introduction

Smart cities experience the far-reaching impacts of unaddressed fires, extending beyond immediate destruction to encompass socioeconomic and environmental consequences [1,2]. Fires, whether they are wildfires, building fires, or car fires, pose substantial threats to lives, property, and ecosystems in densely populated and technologically advanced urban areas. The aftermath of fires in smart cities presents intricate challenges, affecting human safety, straining municipal resources, and causing economic losses, property damage, and environmental degradation. According to the Global Fire Report of 2018, fires impacted a significant number of structures, ranging from 2.5 to 4.5 million, and caused nearly 62,000 fatalities across 57 countries during the period from 1993 to 2016 [3]. The National Fire Data System (NFDS) stated that from September 2020 to 2021, there were 24,539 buildings destroyed by fires in Republic of Korea. The fires resulted in 250 fatalities, 1646 incidents of injury, and 705,960 USD in immediate destruction to property [4]. Similarly, from September 2020 to 2021, there were 78,219 car fires in Republic of Korea. These fires caused 461 deaths, 1875 injuries, and 357,609 USD in property destruction [5]. The damages caused by wildfires have increased in the United States and other

nations during the last twenty years. From the 1990s onwards, an average of 72,200 forestry burns resulted in the burning of approximately 7 million acres each year. This number has continued to rise until the year 2000.

In contrast to structure and wildfires, vehicle fires are the most destructive natural catastrophes in the natural life cycle. There are various reasons for wildfire, including an increase in temperature, climate variability, lightning from clouds, sparking from falling boulders, and summertime friction of dry branches [6]. In 2016, 1161 persons in Southern Europe were affected by wildfire, resulting in a loss of 5.5 billion USD [7]. In 2016, burning forests affected a total of 158,290 individuals, marking the third highest figure observed since 2006; however, it is still below the one million individuals who experienced the dangerous forest fires in 2007 in Macedonia. The Forestry and Fire Prevention Department in California estimated that 2018 was among the most lethal years in the history of California, including 7500 fire incidents that demolished over 1,670,000 acres and more than 100 lives suffered from this [8]. These alarming figures inspired the researchers to build an efficient system for the early identification and control of fires. To ensure the resilience and functionality of smart cities, effective fire detection systems are crucial. Integrating advanced technologies such as visual sensors and Deep Learning (DL) models can prevent or minimize the extensive consequences of fires, safeguarding lives, property, and the delicate urban-environmental balance amid increasing urbanization and climate change challenges. These systems play a pivotal role in ensuring the sustainable growth and safety of smart cities, aligning with the imperative of environmental sustainability and urban planning.

Numerous researchers have explored the use of soft computing techniques in combination with conservative fire alert systems (CFAS) and optical sensors to mitigate the propagation of flames [9]. In CFAS, researchers employed sensing devices such as flame and smoke sensors that involve direct contact with the fire to anticipate fire occurrences. However, scalar sensor-based systems fail when they need more information, such as how much area is on fire, where it is, and the intensity of the fire. Moreover, these sensors need human interaction, which means that if an alarm sounds off, a person needs to visit the place for confirmation. To navigate these problems, researchers came up with various methods by utilizing visual sensors [9,10]. Vision-based approaches are significant for fire detection. Conservative fire detection (CFD) and DL-based techniques are used in surveillance systems to automatically monitor fire incidents [11–14].

These automated systems are good because they respond quickly, require less human intervention, are cheap, and cover a larger area. However, fire detection with TFD-based techniques is hard and takes a lot of time because TFD-based strategies involve hand-crafted feature extraction, which is a lengthy process and requires domain specialists [15]. Mainly with TFD-based techniques, it is difficult to detect fires early and set the alarms because of changes in the lighting, reflections, and the low detection performance [11]. Considering the application of DL models in diverse fields [16,17], including fire detection in surveillance technology, we incorporated them into our study. While DL offers an end-to-end feature extraction technique, it is resource-intensive and needs a significant amount of training data [18]. So, in this paper, we proposed an efficient lightweight FlameNet model that achieves exceptional detection accuracy and has low false alarm rates, as well as the ability to be implemented to resource-constrained tools (RCT):

- Considering the problems of IoT devices in the real world concerning limited computing power, we present a lightweight deep model that works effectively when compared to the well-known lightweight models such as NASNetMobile and EfficientNet; the proposed FlameNet model achieves higher performance in terms of accuracy, frames per second(FPS), and small footprint on the disk, while having fewer trainable parameters.
- To assist the intermediate features, we progressively modified spatial attention (MSA), which refined the backbone extracted features leading to superior performance. The empirical findings show that our suggested system gave superior performance compared

to the state-of-the-art (SOTA) models with respect to accuracy, has 24.34% fewer parameters than NASNetMobile, and, in terms of time complexity, when tested on Raspberry Pi (RPi) and a central processing unit (CPU), it obtained 8.96 and 10.64 FPS, respectively, in a real-time environment.

- Different benchmark datasets for fire detection in specific environments can be found in the literature, but they are not adaptable to a wide range of situations. To address this issue, we developed a new composite dataset that includes challenging images of various fire and non-fire categories. This dataset is collected from popular public datasets to ultimately train our model on diverse data. Furthermore, as part of our evaluation of our proposed dataset, we re-implemented SOTA studies to test its performance and diversity. As a result, we were able to compare different approaches and evaluate how well they performed in addressing the challenges we faced in our dataset.

The rest of this paper is organized in the following manner: in Section 2, we have discussed a brief description of the literary work as well as its benefits and drawbacks; Section 3 explains the internal, in-depth information regarding the proposed dataset as well as about the architecture of our proposed method; and experimental findings are given in Section 4; lastly, Section 5 concludes the paper with findings and suggestions for future directions.

## 2. Related Work

Fire is an atypical occurrence that has the potential to result in significant loss of life and physical harm, as well as swift and extensive destruction of valuable assets. In order to avoid the dangers of fire, numerous methods were used to monitor and control fires in cities to save lives and property. CFAS and vision sensors-based systems are two things that researchers have decided to make to the field of detection systems in recent times. Different types of sensors, including smoke, temperature, and photosensitive sensors, are employed by CFAS to detect Fires [19–22]. However, CFAS methodologies are required to be close to the fire, like in an enclosed area, and they do not work if the fire is burning from a long distance, like in an outdoor area. Moreover, the CFAS cannot provide any further details about the status of the fire or how fast it is burning. The CFAS methods need human intervention, such as visiting a fire site to validate the presence of fire in the occurrence of an alert. Numerous visual sensor-based approaches for fire detection have been introduced in the literature to address these limitations [23,24]. There are two main types of vision-based systems for fire detection: those that rely on traditional fire detection (TFD) and those that use DL-based algorithms. Digital image processing and pattern recognition techniques are frequently employed in methods based on TFD. For example, the authors used temporal, spatial, and spectral analysis as well as other methodologies to find the fire areas in an image [25]. However, the approach they used is based on the presumption that fires possess an atypical shape, which is not always accurate since objects in motion can also undergo structural transformations. TFD techniques include wavelet analysis and the quick Fourier transform [26].

Moreover, in another study, authors used mobility assessment, shape diversity, color characteristics, and bag-of-word for classifying fires [27]. Antecedent methods also used a gray-level co-occurrence matrix and an oriented gradient histogram in combination with SVM [28]. In TFD-based approaches, manually crafted feature extraction is a complicated and time-intensive task, and these approaches are unable to accomplish a high level of precision. DL-based approaches that use Closed-Circuit Television (CCTV) surveillance systems are very important for fire detection. The inclusion of automated end-to-end acquisition of features enhances the intuitiveness and efficiency of such models. Particularly in comparison to TFD, the DL methods performed better because they were more accurate and had fewer erroneous alarms. For example, authors employed a custom-built CNN framework that could be used to identify fire and smoke [29]. They used a small sample of images to evaluate the performance, but they failed to compare those results to any SOTA

approach. In another follow-up study, the authors employed two pre-trained SOTA CNN models, namely, VGG16 and ResNet50, for the detection of the fire. A CNN-based approach is employed to detect flames across surveillance networks for disaster risk monitoring and prevention [13], in which the author uses a pre-trained AlexNet model.

In addition to this, they exhibit an intelligent means of selecting a camera according to its priority. For this research study, the main concern with their work is that their suggested approach takes a lot of time and is hard to set up on RCT. Scholarly researchers expanded their work and utilized GoogLeNet-like neural architecture to find fires quickly in surveillance videos. This assisted them in navigating the time complexity of the model and improve its performance [30]. They did experiments on two different benchmark datasets and obtained more accurate results than SOTA techniques. In the subsequent procedure, researchers implied an efficient lightweight SqueezeNet framework for detecting and locating fires quickly and efficiently in surveillance systems [31]. In this work, they also figured out how intense the fire was and what components were being noticed. In another study, authors managed to show a deep CNN-based technique that uses less energy and can find early signs of smoke in both regular and foggy situations [11]. Furthermore, authors also came up with lightweight deep models [32,33] based on MobileNetV2 for monitoring fires in uncertain situations [34], where a light DCNN with a few intense convolution layers is used, making it costly to run on computational devices. They shrink the dimension of the created model to 3 MB without sacrificing its competence and achieving SOTA precision on two baseline datasets [32]. Additionally, the authors presented advanced convolutional generative adversarial neural networks for the detection of fire that were trained on actual images, incorporating the random vectors. In this case, the discriminator was trained on its own by utilizing smoky images without the generator [34].

The authors of [35] introduced a technique that uses a strong color model to identify suitable burn areas. In their proposed study, they apply a motion-intensity-aware approach for the analysis of motion to distinguish between fire and non-fire zones based on spatiotemporal properties. Researchers in [36] proposed a deep silence network that can find the areas of an image where there are forest fires. By using the concept based on CNN, they combined the salient areas at the pixel and object levels to generate a hazy saliency map. In another study, the authors introduced a vision transformer-based approach for fire detection, in which a picture is segmented into patches of uniform size to establish a spatial correlation. In another follow-up study, the authors employed channel attention with other backbone feature extractors [37]. They used the same assessment procedures as [30,33] and tested their approach on two baseline datasets. The authors presented a forest fire detection algorithm built on top of a fuzzy-based optimized thresholding and spatial transformer network (STN)-based CNN [38], in which the softmax layer is employed for categorizing fire scenes using a spatial transformer network and then an adaptive threshold operation relying on an entropy function. A summary of the included literature is presented in Table 1. Based on previous research, numerous DL-based methods for the detection of fires have been designed and proven to yield convincing results. However the reliability of detection must be enhanced and the number of false alarms needs to go down in order to save lives and property. Moreover, these models are hard to compute and need effective GPUs and TPUs in order to do so. To address these issues, we proposed an efficient lightweight CNN-based model, FlameNet, for the detection of fire that has lower false alarm rates, high detection accuracy and is deployable via RCD.

**Table 1.** Summary of the included literature. The ✓ mark indicates that dataset is publicly available while × represents datasets with restricted access.

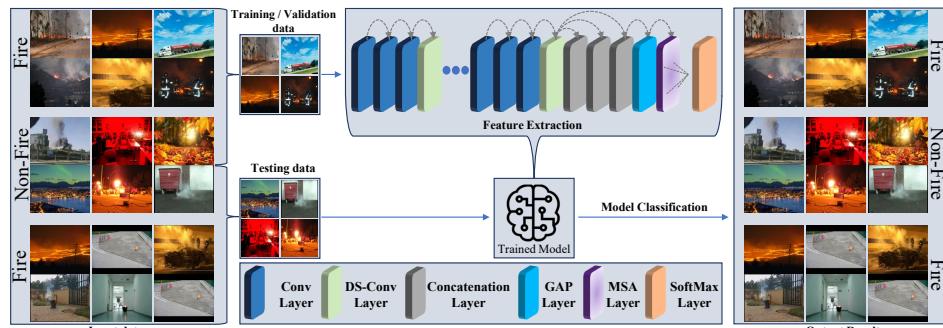
Ref.	Overview	Dataset Used	Dataset Availability	Method	Condition
[27]	Author propose a method that is able to detect fires by analyzing videos acquired by surveillance cameras	videos dataset	×	ML base method	Indoor, outdoor
[28]	Author proposed system to detect fire under surveillance area	image dataset	×	SVM	Indoor
[29]	Author proposed CNN for video fire and smoke detection	videos dataset	×	CNN	Outdoor
[31]	Author proposed CNN architecture, inspired by the SqueezeNet architecture for fire detection, localization, and semantic understanding	Foggia, Chino dataset	✓	Custom-CNN	Indoor, outdoor
[33]	Author proposed an efficient CNN-based system for fire detection in videos captured in uncertain surveillance scenarios	image dataset	✓	CNN	Indoor
[34]	Author propose a vision-based method to detect smoke using Deep Convolutional Generative Adversarial Neural Networks (DC-GANs)	video clips dataset	✓	DCGAN	Outdoor
[35]	Author proposed a robust ICA K-medoids-based color model is developed to reliably detect all candidate fire regions in a scene	VisiFire, AzarFire dataset	✓	ICA-K-medoids based	Indoor, outdoor
[36]	Author proposed video smoke detection method based on deep saliency network	Video dataset	×	deep saliency network	Outdoor
[37]	Author presents a custom framework for detecting fire using transfer learning with SOTA CNNs trained over real-world fire breakout images	combine dataset	✓	Attention base CNN	Indoor
[38]	Author proposed an STN-based CNN and fuzzy entropy optimized thresholding for forest fire detection	images dataset	×	STN-based CNN	Outdoor
[39]	Authors employed neural networks to swiftly identify instances of fire and smoke in both indoor and outdoor settings, utilizing video footage	fire, smoke images	✓	RCNN, LSTM	Outdoor
[40]	Authors combined manual features and DL features to develop a rapid and precise forest smoke detection system	smoke images	×	DL	Outdoor

### 3. Proposed Methodology

This section represents the details about the dataset collection and the proposed model to address the problem of accurate fire detection. The dataset presented in Figure 1 is curated by combining various well-known, publicly available datasets to represent diverse, complex, and confusing samples, ensuring the model's robustness and generalizability. The proposed FlameNet framework is presented in Figure 2. In the training phase, the FlameNet is trained on the newly curated dataset, i.e., Ignited-Flames, and the most prominent deep features are extracted, while in the testing phase, the model predicts the label for the input image.



**Figure 1.** Sample images from our Ignited-Flames dataset. In the first row, we included the sample of different fire samples, and in the second row, we presented non-fire images.



**Figure 2.** The proposed FlameNet framework for efficient fire detection.

### 3.1. Dataset Collection

Acquiring appropriate data for the purpose of evaluation poses a significant challenge, necessitating a substantial investment of time. The current datasets are kind of small and only look at specific situations, such as indoor or outdoor situations. However, we want our models to be good at understanding things and not getting things wrong too often. That is why we created Ignited-Flames, a composited dataset, by combining different challenging images from the benchmark dataset. We collected different categories of fire and non-fire images from BoW [41], which has 121 fire and 107 non-fire images, SV-Fire [1], which has 1000 fire and 500 non-fire images, Foggia [13], which has 7018 fire images and 7018 non-fire images, Saied Fire [42], containing 755 fire and 244 non-fire images, and Sharma datasets [13], with 110 fire and 541 non-fire images to make a new and challenging composite dataset. The overall statistics of the Ignited-Flames dataset are listed in Table 2.

**Table 2.** Overall statistics of the newly created composite data with a total of 9004 fire images and 8402 non-fire images.

Ignited-Flames Dataset	Training	Testing	Validation	Total
Fire	6456	1811	737	9004
Non-Fire	6076	1671	655	8402

This article demonstrates a few examples of images and gives some general statistics about the new dataset. There are 17,406 images in the Ignited-Flames dataset in total, of which 9004 are fire images and 8402 are non-fire. The proposed Ignited-Flames dataset is split into three distinct categories: testing, validation, and training. The training set encompasses 70% of the entire dataset, with the validation set comprising 20%, and the remaining 10% designated for the testing set (Figure 1) shows a few examples from the recently assembled dataset.

### 3.2. Deep Features Extraction

In the field of sophisticated video surveillance, CNNs are employed for a wide variety of tasks, including plant disease detection [43–45], video summarizing [46], and crowd counting [47], as well as object detection [48] and vehicle re-identification [49]. The CNN structure consists of three major components: the convolution layer (CL), the pooling layer (PL), and the fully linked Layer (FL). A deep CNN has only one input and many hidden, fully connected, and Softmax layers [50]. The extracted feature maps are down-sampled using mean, minimal, and maximal pooling for dimension reduction.

It can be hard to choose the right architecture for a particular scenario to achieve satisfactory outcomes while maintaining computational complexity [51]. Following the proposed architecture, every CNN comes with its own pros and cons. For example, VGG16 and AlexNet architectures are simple to design and build. The AlexNet architecture was represented in the ImageNet competition, and since then it has become standard for DL architecture. Adding more CLs to a network is supposed to improve its efficiency, and the VGG model supports this claim. As a strong feature extractor capable of handling huge datasets and challenging background identification tasks, the authors recommended VGG16, a 16-layer design that uses the same filter size and has a significant classification improvement.

Despite their various benefits, VGG19 and VGG16 are not resource-efficient in terms of parameters. CNN architectures such as EfficientNetB0, MobileNetV1, and NASNetMobile exhibit enhanced robustness and cost-effectiveness. MobileNetV1 and NASNetMobile are specifically engineered to ensure prompt and predictable response times, making them well-suited for applications requiring rapid processing [52]. These architectures offer significant advantages in terms of computational efficiency, making them favorable choices in various scientific and professional contexts [53]. Taking into account real-world implementation,

resource computing cost, and suppression of restrictions in existing lightweight models, this paper offers FlameNet, an effective lightweight fire classification and detection model. The proposed FlameNet is based on the MobileNetV1 and is built by using depthwise separable convolutions, with the exception of the first layer, which employs a full convolution. Every layer in the model is accompanied by batch normalization and the Rectified Linear Unit (ReLU) nonlinearity, except for the final fully connected layer. This last layer lacks nonlinearity and directly connects to a softmax layer for classification. Considering both depthwise and pointwise convolutions as separate layers, the MobileNetV1 model consists of 28 layers.

The MobileNet neural layers utilize  $3 \times 3$  and  $1 \times 1$  kernel sizes. The input size provided to the model is  $224 \times 224$  with 3 channels for RGB image format. Global average pooling (GAP) is utilized to reduce the dimensionality and obtain the average values of different features. Additionally, the model incorporates a GAP layer to obtain average feature values and a concatenation layer for combining features. The convolutional strides used have sizes of 1 and 2. ReLU serves as the activation function across the model's levels. The dropout rate is scaled to 0.2 to prevent over-fitting. The Softmax is added to the final layer and corresponds to the two classes, namely, fire and non-fire. The dense layers of MobileNetV1 are eliminated, resulting in the extraction of a feature map with dimensions of  $7 \times 7$  and 1024 channels. These extracted features are represented by  $\Omega$  which is mathematically shown in Equation (1):

$$\Omega = (\Phi, (\alpha))(x), \quad (1)$$

where  $\Phi$  represents feature vectors ( $7 \times 7$ ),  $\alpha$  represents channels, and  $x$  is input. The feature vector  $\Omega$  obtained from Equation (1) involves a comprehensive range of data, including the object's configuration, border details, hues, shapes, and other relevant information. Nevertheless, these are less representative features, and utilizing them directly leads to inaccurate results, especially in complex scenarios. The  $\Omega$  feature map is further improved through the utilization of MSA. This module effectively captures the most essential spatial patterns.

### 3.3. Modified Spatial Attention

We introduced MSA to further refine the intermediate features extracted from the backbone network. A spatial attention map is generated by exploiting the inter-spatial relationship of features. In contrast to channel attention, spatial attention directs its focus toward the spatial regions containing informative components. In order to compute the spatial attention, we first employ average and max pooling operations. The outputs of the operations are then fused effectively to generate a refined feature descriptor. The utilization of pooling operations has demonstrated effectiveness in highlighting informative regions. The spatial module exploits the inter-spatial connections among features. In contrast to the channel attention mechanism, the MSA is designed to prioritize the identification of the most critical region, thereby enhancing the capabilities of the intermediate features. The inclusion of pooling operations along the axis is an effective strategy for emphasizing regions of high information content. The application of these two pooling operations results in the generation of enhanced features. The MSA is depicted in Figure 3.

$$\Omega M_a \Sigma = Avg - P(f) \quad (2)$$

$$\Omega M_a \lambda = max - P(f), \quad (3)$$

where  $\Sigma$  represents Avg and another notation,  $\lambda$ , represents max. Afterward, the feature maps that have been generated are merged through an addition operation and then subjected to convolution by a convolutional layer, resulting in the creation of a two-dimensional SA feature map. In the MSA module, we incorporated two convolutional layers, which were subsequently followed by the ReLU activation function. The initial layer employs a  $1 \times 1$  convolution, while the second layer implements a  $3 \times 3$  convolution. Instead of

employing dilated convolution as suggested in the previous research [54], we chose to utilize standard convolutions. The modification is validated empirically.

$$M_a P(f) = (\otimes(f^{1 \times 1}(\otimes(f^{3 \times 3}\Omega M_a \Sigma + \Omega M_a \lambda))) \quad (4)$$

Here, the symbol  $f$  denotes the size of the filter employed in the convolutional layer of the MSA module. The MSA map, denoted as  $M_a P(f)_{GAP}$ , can be derived by applying the GAP operation on the feature maps of  $M_a P(f)$ . Subsequently, the output of the GAP operation is concatenated with the output of the function  $f$ . This process is illustrated below.

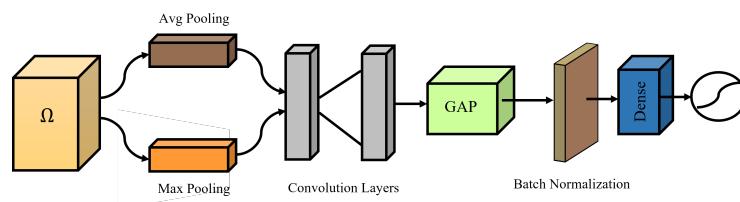
$$M_a P(f)_{GAP} = GAP(f) \quad (5)$$

$$f_s pa = \Theta[M_a P(f)_{GAP}, f]. \quad (6)$$

Following the concatenation operation, the resulting feature maps, denoted as  $f_s pa$ , undergo batch normalization. Afterward, we combine the feature maps normalization with  $\Omega$  to yield  $f_s paf$ :

$$f_s paf = \otimes[Batchnormal, \Omega]. \quad (7)$$

Subsequently, the  $f_s paf$  features were propagated to a dense layer containing 100 neurons. Ultimately, a softmax is implemented to categorize the input images based on their respective classes.



**Figure 3.** The architecture of the proposed modified spatial attention.

FlameNet incorporates two main primary parts. In the initial part, fire and non-fire images from the input dataset are fed into the proposed network, which detects and classifies fires accurately. During the subsequent stage, the model proceeds to execute a course of action in accordance with the anticipated classification of the input image. In the event that the anticipated classification denotes a fire occurring within a building edifice or a fire transpiring within a vehicle, a notification is transmitted to the emergency response agency in closest proximity, thereby facilitating expeditious intervention. Figure 2 presents the suggested framework of our proposed model. Before designing the new FlameNet framework, we first look at how well well-known ImageNet and pre-trained CNN architectures such as VGG16, ResNet50, MobileNetV1, and NASNetMobile work.

#### 4. Results and Discussions

This section focuses on assessment measures and evaluation metrics in detail, as well as discussing the newly created dataset along with the quantitative and qualitative results. Initially, the experimental setup, as well as the performance measurements, are discussed; then, a discussion on the Ignited-Flames dataset results is presented, and finally, the findings are evaluated. All of the models, including our proposed network, were trained with a low learning rate over a total of 10 epochs to make sure they recalled most of what they had learned. In Section 4.3 of the article, SOTA models are used to provide a comparison with the suggested network, and the key hyper-parameters utilized in these tests are outlined. Based on the findings, every model was retrained with its own default input size and a batch size of 32, and the adaptive moment estimation (Adam) optimizer was set to  $1 \times 10^{-5}$ . The tests were performed on the Windows 10 operating system with an NVIDIA RTX 2070

Super GPU with 8 GB of onboard memory, a Keras DL framework, and TensorFlow for the backend using the 3.9.12 Python version. As shown in the following Equations (8)–(11), different numbers of metrics, including accuracy, recall rates, and F1-measure values, are used to assess how well the proposed model performs.

#### 4.1. Evaluation Metrics

In the context of problems with classification, accuracy is commonly defined as the proportion of correct predictions made by the model across all categories of predictions.

$$\text{Accuracy} = \left( \frac{TP + FN}{TP + TN + FP + FN} \right), \quad (8)$$

where the terms  $TP$ ,  $TN$ ,  $FP$ , and  $FN$  represent True Positive, True Negative, False Positive, and False Negative, respectively. *Precision* is a metric that shows the proportion of the dataset that is marked as “Fire” is actually fire. The predicted positives and negatives ( $TP$  and  $FP$ ) are the images that are predicted to be fire, and the images that show fire are  $TP$ .

$$\text{Precision} = \left( \frac{TP}{TP + FP} \right) \quad (9)$$

The *recall* is a metric that indicates the proportion of observations in a dataset that the model anticipated to have a fire. The expected true positives and fire pictures are denoted by  $TP$ .

$$\text{Recall} = \left( \frac{TP}{TP + FP} \right) \quad (10)$$

The *F1-score* is the calculation of the precision and recall harmonically.

$$\text{F1-score} = 2 \times \left( \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}} \right) \quad (11)$$

#### 4.2. Performance Analysis with State of the Art Networks

This section compared the proposed network to various CNN-based architectures that had already been trained for the purpose of fire recognition and detection. These models were analyzed in terms of the FPR (False Positive Rate), FNR (False Negative Rate) as presented in Table 3. Adding more, in terms of number of parameters, precision, recall, F1-score, and accuracy, as presented in Table 4. Additionally, the proposed Ignited-Flames dataset was evaluated by re-implementing SOTA studies as listed in Table 5. Xception demonstrates FPR and FNR scores of 0.0994 and 0.0195, respectively, achieving an accuracy of 93.69%. ResNet50 exhibits impressive metrics with FPR, FNR, and accuracy values of 0.0733, 0.0464, and 93.98%, respectively. EfficientNetB0 achieves a notable FPR of 0.0199, FNR of 0.0188, and accuracy of 95.98%. Similarly, NASNetMobile attains FPR, FNR, and accuracy rates of 0.0122, 0.01688, and 96.04%, respectively. With VGG16, an accuracy of 98.63% is achieved, accompanied by FPR and FNR of 0.0017 and 0.0251, respectively. Notably, our proposed model surpasses SOTA techniques, attaining the most favorable outcomes with FPR, FNR, and accuracy rates of 0.0022, 0.0168, and 99.40%, respectively. This shows our model’s superior performance in terms of minimized false alarm rates and highest accuracy. Xception and ResNet50 have low accuracy, which is 93.69% and 93.98%, EfficientNetB0 and NASNetMobile obtained an accuracy of 95.98% and 96.04%, but NASNetMobile is lighter than EfficientNetB0 in terms of parameters. Similarly, VGG16 and our proposed network have the highest accuracy, which is 98.63% and 99.40%, as compared to the previously discussed models, but our proposed method is the most accurate and lightweight. A comparison between the proposed approach and VGG16 indicates that VGG16 findings are comparable to those of the proposed network. However, the key difference is the highest number of parameters; VGG16 contains 14.72 million parameters, while our proposed network has 3.23 million. Table 4 represents the finding acquired

by using pre-trained models. These pre-trained models show better efficiency with a comparatively low false alarm rate. However, there is still a high prevalence of incorrect predictions that require improvement.

**Table 3.** FPR and FNR of FlameNet against SOTA. The downward arrow ( $\downarrow$ ) shows lower value is better while the upward arrow ( $\uparrow$ ) indicates that higher is better.

Model	FPR $\downarrow$	FNR $\downarrow$	Accuracy (%) $\uparrow$
Xception	0.0994	0.0195	93.69
ResNet50	0.0733	0.0464	93.98
EfficientNetB0	0.0199	0.0188	95.98
NASNetMobile	0.0122	0.0168	96.04
VGG16	0.0017	0.0251	98.63
FlameNet	0.0022	0.0168	99.40

**Table 4.** Evaluation of our proposed model by using the same batch size of 32 and input size of  $224 \times 224$  against the SOTA models using the Ignited-Flames dataset. The downward arrow ( $\downarrow$ ) shows lower value is better while the upward arrow ( $\uparrow$ ) indicates that higher is better.

Model	Classss	Classification Report				Parameters (M) $\downarrow$
		Precision	Recall	F1-Score	Accuracy (%) $\uparrow$	
Xception [55]	Fire	0.99	0.89	0.93	93.69	20.87 M
	NonFire	0.86	0.98	0.92		
ResNet50 [56]	Fire	0.96	0.92	0.94	93.98	23.59 M
	NonFire	0.92	0.96	0.93		
EfficientNetB0 [57]	Fire	0.94	0.98	0.96	95.98	40.52 M
	NonFire	0.98	0.94	0.96		
NASNetMobile [58]	Fire	0.99	0.93	0.96	96.04	4.27 M
	NonFire	0.92	0.99	0.95		
VGG16 [59]	Fire	0.98	1.00	0.99	98.63	14.27 M
	NonFire	1.00	0.97	0.99		
FlameNet	Fire	0.98	1.00	0.99	99.40	3.23 M
	NonFire	1.00	0.98	0.99		

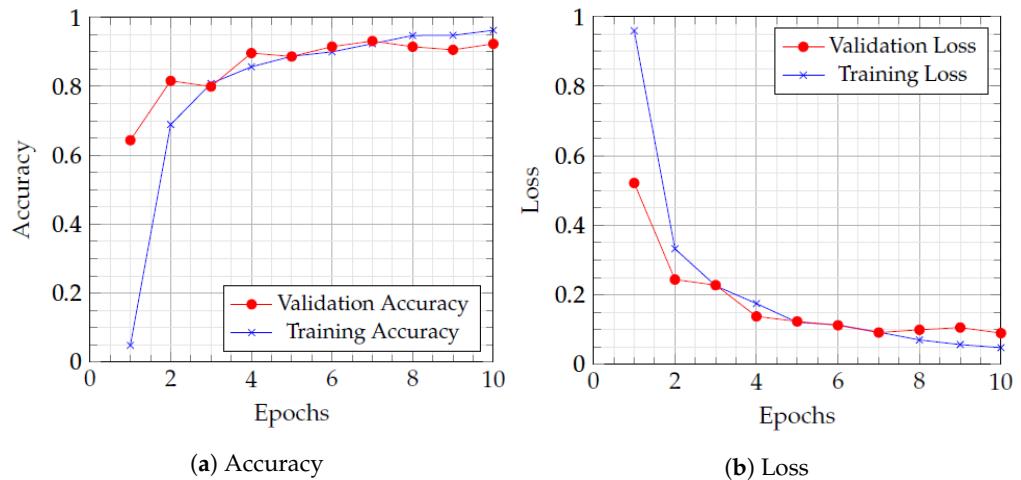
**Table 5.** Results of different SOTA studies on the proposed Ignited-Flames dataset. The downward arrow ( $\downarrow$ ) shows lower value is better while the upward arrow ( $\uparrow$ ) indicates that higher is better.

Method	Classss	Classification Report				Parameters (M) $\downarrow$
		Precision	Recall	F1-Score	Accuracy (%) $\uparrow$	
Dilshad et al. [1]	Fire	0.95	0.77	0.85	87.38	9.99 M
	NonFire	0.83	0.96	0.89		
Yar et al. [60]	Fire	0.94	0.93	0.93	93.11	11.17 M
	NonFire	0.93	0.93	0.93		
Sharma et al. [13]	Fire	0.93	1.00	0.96	96.18	23.59 M
	NonFire	1.00	0.93	0.96		
Khan et al. [61]	Fire	0.98	0.98	0.98	98.28	20.02 M
	NonFire	0.98	0.98	0.98		

As a result, this study investigated the accuracy and erroneous prediction of a fine-tuned and pre-trained CNN architecture (MobileNetV1). Figure 4a shows the training accuracy and validation accuracy while Figure 4b shows the training loss and validation loss of our proposed network. Accuracy and loss are represented on the vertical axis, while the horizontal axis indicates the number of epochs completed. The results represented

in Figure 4 showcase the effectiveness of our proposed network in the domain of fire detection and classification. The training and validation accuracy line graph of the model changes as the number of training and validation iterations varies, as represented in Figure 4a,b. Similarly, the values of training and the validation loss decrease from 0.9 to 0.04, as presented in Figure 4b. Additionally, Figure 5, we can see the confusion matrices for all of the SOTA models that were trained using the Ignited-Flames dataset. The red diagonal relates to TP, while the saturation indicates the correct identification. The proposed network has a higher overall classification accuracy than the SOTA models, despite the incorrect prediction of certain images in both fire and non-fire categories.

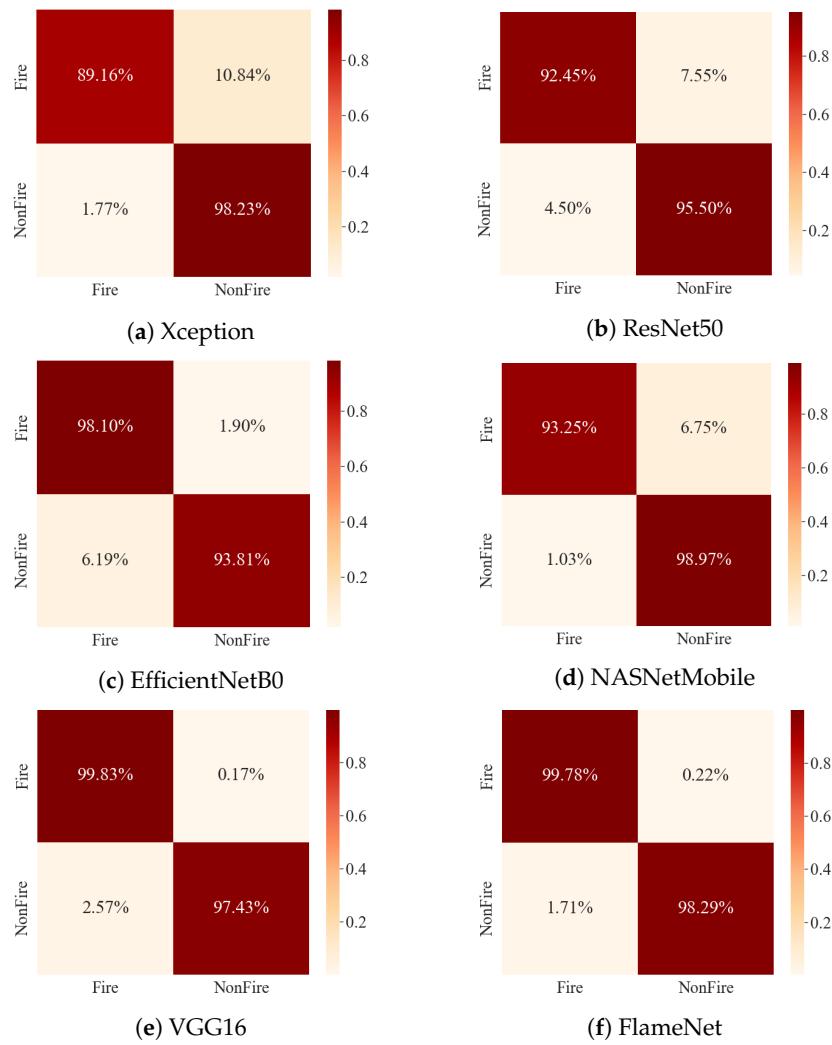
Additionally, we conducted an empirical evaluation of several DL models for the classification of fire and non-fire images on the Ignited-Flames dataset, as given in Table 5. The models examined are ResNetFire by Sharma et al. [13], LW-CNN by Yar et al. [60], DeepFire by Khan et al. [61], and E-FireNet by Dilshad et al. [1]. The results revealed that E-FireNet achieved an accuracy of 87.38% with a precision of 0.95, a recall of 0.77, and an F1-score of 0.85 for the “Fire” class. For the “NonFire” class, E-FireNet achieved a 0.83 precision, 0.96 recall, and 0.89 F1 score. On the other hand, RestNetFire achieved impressive precision and recall scores of 0.93 and 1.00, respectively, with an F1-Score of 0.96 for the “Fire” class. Similarly, for the “NonFire” class, RestNetFire demonstrated a precision of 1.00, a recall of 0.93, and an F1-Score of 0.96. LW-CNN and DeepFire depict high precision, recall, and F1-Score values for both fire and non-fire classification.



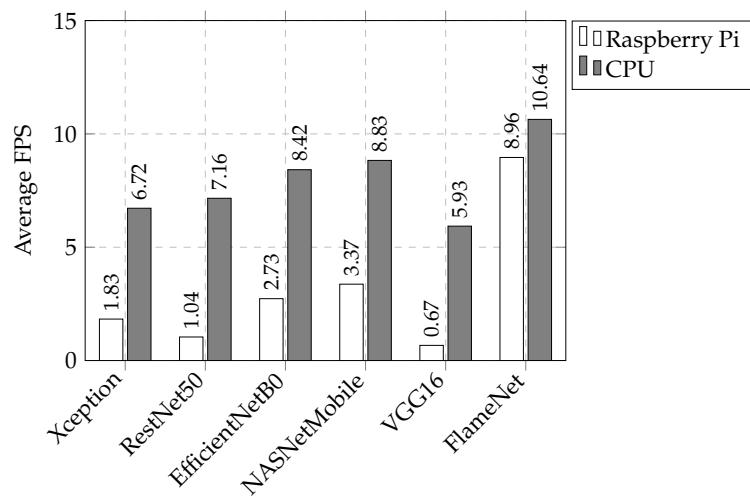
**Figure 4.** Line graphs illustrating accuracy and loss during training and validation of the proposed FlameNet method.

#### 4.3. Time Complexity Analysis

To evaluate the efficacy of a deep model, its performance and deployment capability must be examined in real time across several systems, such as Raspberry Pi (RPi) and CPU. The parameters of the RPi and CPU used to analyze the FPS of our proposed network are specified in Section 4. The FPS value of our presented model, by using RPi, is 8.96; while in the case of the CPU, this value increased to 10.64. In Figure 6, we evaluated our presented model by comparing its performance in terms of FPS with several baseline models. By using the RPi and the CPU, the experimental results show that the FPS for the Xception model is 1.83 and 6.72, respectively. However, for the ResNet50 model, these values are 1.04 and 7.16, respectively. Similarly, the values for the EfficientNetB0 model are 2.73 and 8.42. On the other hand, the NASNetMobile model achieved 3.37 in the case of the RPi and 8.83 for the CPU, and for the VGG16 model, these values are 0.67 and 5.93, respectively. Lastly, for our proposed network, these values are 8.96 and 10.64. Our presented network outperforms other baseline methods in terms of time complexity, demonstrating its superior effectiveness. Therefore, in terms of time, our approach proves to be highly efficient in real-world operations and processes.



**Figure 5.** Confusion Matrices of the different CNN models against our proposed method.



**Figure 6.** Comparison of our proposed method against different DL models by FPS.

## 5. Conclusions

Fire scenario classification using CNN-based smart monitoring systems has been crucial in preventing sociological, ecological, and economic harm. However, existing studies have primarily focused on accuracy improvement, while giving less attention to model computation and generalization. This research introduces FlameNet, an efficient network

for accurately classifying fire and non-fire imagery without neglecting computational efficiency and generalization capabilities. While conducting the comparison with the SOTA method, our proposed network achieved the highest testing accuracy of 99.40% with fewer parameters. Moreover, FlameNet achieved a precision of 0.99 with respect to fire and 1.00 with respect to the non-fire class, with a recall of 1.00 in fire and 0.98 in the non-fire class, and an F1-score of 0.99 in both classes. Additionally, the new Ignited-Flames dataset was created by combining the challenging fire and non-fire images. Nine CNN models and the suggested network were used in a series of experiments, and their results were evaluated with regard to the accuracy, parameters, and FPS on two local systems (RPi and CPU) using the testing data. FlameNet does face certain limitations and challenges in real-world implementation. One notable example is its current focus on binary fire detection (fire vs. non-fire), rather than precisely localizing the type of fire source, such as fires on cars, buildings, ships, or trains, among others. In the future, we aim to enhance FlameNet and address its limitations, enhancing the training data by encompassing a wider range of fire types and scenarios such as car fire, bike fire, train fire, etc. Another approach involves annotating the dataset and employing more efficient algorithms such as Faster CNN or Detectron2 to bolster FlameNet's fire detection accuracy.

**Author Contributions:** Methodology, Investigation, Writing—original draft, M.N.; Conceptualization, Validation, Investigation, N.D.; Data curation, N.S.A.; Project administration, L.M.D.; Formal analysis, J.N.; Funding acquisition, H.-K.S.; Resources, Writing—review & editing, Supervision, Project administration, Funding acquisition, H.M. All authors have read and agreed to the published version of the manuscript

**Funding:** This work was supported by Basic Science Research Program through the National Research Foundation of Korea (NRF), funded by the Ministry of Education (2020R1A6A1A03038540), and the National Research Foundation of Korea (NRF) grant funded by the Korean government and the Ministry of Science and ICT (MSIT) (2021R1F1A1046339), and the Institute of Information and communications Technology Planning and Evaluation (IITP) under the metaverse support program to nurture the best talents (IITP-2023-RS-2023-00254529) grant funded by the Korean government (MSIT).

**Data Availability Statement:** Data will be made available on request.

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

## References

1. Dilshad, N.; Khan, T.; Song, J. Efficient Deep Learning Framework for Fire Detection in Complex Surveillance Environment. *Comput. Syst. Eng.* **2023**, *46*, 749–764. [[CrossRef](#)]
2. Shah, S.A.; Seker, D.Z.; Rathore, M.M.; Hameed, S.; Yahia, S.B.; Draheim, D. Towards disaster resilient smart cities: Can internet of things and big data analytics be the game changers? *IEEE Access* **2019**, *7*, 91885–91903. [[CrossRef](#)]
3. Rathnayake, R.; Sridarran, P.; Abeynayake, M. Fire risk of apparel manufacturing buildings in Sri Lanka. *J. Facil. Manag.* **2021**, *20*, 59–78. [[CrossRef](#)]
4. Nordenfjeldske Development Services (NFDS), Fire Statistics. 2021. Available online: <https://www.nfds.go.kr/stat/general.do> (accessed on 20 June 2023).
5. Insurance Information Institute. 2021. Available online: <https://www.iii.org/fact-statistic/facts-statistics-wildfires> (accessed on 20 June 2023)
6. Dubey, V.; Kumar, P.; Chauhan, N. Forest fire detection system using IoT and artificial neural network. In *Proceedings of the International Conference on Innovative Computing and Communications*; Springer: Berlin/Heidelberg, Germany, 2019; pp. 323–337.
7. Wolters, C. *California Fires Are Raging: Get the Facts on Wildfires*; National Geographic: Washington, DC, USA, 2019.
8. Guha-Sapir, D.; Hoyois, P.; Wallemacq, P.; Below, R. *Annual Disaster Statistical Review 2016: The Numbers and Trends*; Centre for Research on the Epidemiology of Disasters: Brussels, Belgium, 2018.
9. Muhammad, K.; Ahmad, J.; Baik, S.W. Early fire detection using convolutional neural networks during surveillance for effective disaster management. *Neurocomputing* **2018**, *288*, 30–42. [[CrossRef](#)]
10. El-Hosseini, M.; ZainEldin, H.; Arafat, H.; Badawy, M. A fire detection model based on power-aware scheduling for IoT-sensors in smart cities with partial coverage. *J. Ambient Intell. Humaniz. Comput.* **2021**, *12*, 2629–2648. [[CrossRef](#)]
11. Khan, S.; Muhammad, K.; Mumtaz, S.; Baik, S.W.; de Albuquerque, V.H.C. Energy-efficient deep CNN for smoke detection in foggy IoT environment. *IEEE Internet Things J.* **2019**, *6*, 9237–9245. [[CrossRef](#)]

12. Yin, Z.; Wan, B.; Yuan, F.; Xia, X.; Shi, J. A deep normalization and convolutional neural network for image smoke detection. *IEEE Access* **2017**, *5*, 18429–18438. [[CrossRef](#)]
13. Sharma, J.; Granmo, O.C.; Goodwin, M.; Fidje, J.T. Deep convolutional neural networks for fire detection in images. In *Engineering Applications of Neural Networks, Proceedings of the 18th International Conference, EANN 2017, Athens, Greece, 25–27 August 2017*; Springer: Berlin/Heidelberg, Germany, 2017; pp. 183–193.
14. Shah, S.A.; Seker, D.Z.; Hameed, S.; Draheim, D. The rising role of big data analytics and IoT in disaster management: Recent advances, taxonomy and prospects. *IEEE Access* **2019**, *7*, 54595–54614. [[CrossRef](#)]
15. Minh, D.; Wang, H.X.; Li, Y.F.; Nguyen, T.N. Explainable artificial intelligence: A comprehensive review. *Artif. Intell. Rev.* **2022**, *55*, 3503–3568 [[CrossRef](#)]
16. Nguyen, T.N.; Lee, S.; Nguyen, P.C.; Nguyen-Xuan, H.; Lee, J. Geometrically nonlinear postbuckling behavior of imperfect FG-CNTRC shells under axial compression using isogeometric analysis. *Eur. J. Mech.—A/Solids* **2020**, *84*, 104066. [[CrossRef](#)]
17. Nguyen, T.N.; Nguyen-Xuan, H.; Lee, J. A novel data-driven nonlinear solver for solid mechanics using time series forecasting. *Finite Elem. Anal. Des.* **2020**, *171*, 103377. [[CrossRef](#)]
18. Dang, M.; Nguyen, T.N. Digital Face Manipulation Creation and Detection: A Systematic Review. *Electronics* **2023**, *12*, 3407. [[CrossRef](#)]
19. Yu, L.; Wang, N.; Meng, X. Real-time forest fire detection with wireless sensor networks. In Proceedings of the 2005 International Conference on Wireless Communications, Networking and Mobile Computing, Wuhan, China, 26 September 2005; Volume 2, pp. 1214–1217.
20. Podržaj, P.; Hashimoto, H. Intelligent space as a framework for fire detection and evacuation. *Fire Technol.* **2008**, *44*, 65–76. [[CrossRef](#)]
21. Jan, H.; Yar, H.; Iqbal, J.; Farman, H.; Khan, Z.; Koubaa, A. Raspberry pi assisted safety system for elderly people: An application of smart home. In Proceedings of the 2020 First International Conference of Smart Systems and Emerging Technologies (SMARTTECH), Riyadh, Saudi Arabia, 3–5 November 2020; pp. 155–160.
22. Roque, G.; Padilla, V.S. LPWAN based IoT surveillance system for outdoor fire detection. *IEEE Access* **2020**, *8*, 114900–114909. [[CrossRef](#)]
23. Malbog, M.A.F.; Lacatan, L.L.; Dellosa, R.M.; Austria, Y.D.; Cunanan, C.F. Edge detection comparison of hybrid feature extraction for combustible fire segmentation: A Canny vs Sobel performance analysis. In Proceedings of the 2020 11th IEEE Control and System Graduate Research Colloquium (ICSGRC), Shah Alam, Malaysia, 8 August 2020; pp. 318–322.
24. Khan, R.A.; Uddin, J.; Corraya, S.; Kim, J. Machine vision based indoor fire detection using static and dynamic features. *Int. J. Control. Autom.* **2018**, *11*, 87–98.
25. Liu, C.B.; Ahuja, N. Vision based fire detection. In Proceedings of the 17th International Conference on Pattern Recognition, ICPR 2004, Cambridge, UK, 23–26 August 2004; Volume 4, pp. 134–137.
26. Zhang, Z.; Zhao, J.; Zhang, D.; Qu, C.; Ke, Y.; Cai, B. Contour based forest fire detection using FFT and wavelet. In Proceedings of the 2008 International Conference on Computer Science and Software Engineering, Wuhan, China, 12–14 December 2008; Volume 1, pp. 760–763.
27. 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](#)]
28. Jayashree, D.; Pavithra, S.; Vaishali, G.; Vidhya, J. System to detect fire under surveilled area. In Proceedings of the 2017 Third International Conference on Science Technology Engineering & Management (ICONSTEM), Chennai, India, 23–24 March 2017; pp. 214–219.
29. 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, 24–27 October 2016; pp. 877–882.
30. Muhammad, K.; Ahmad, J.; Mehmood, I.; Rho, S.; Baik, S.W. Convolutional neural networks based fire detection in surveillance videos. *IEEE Access* **2018**, *6*, 18174–18183. [[CrossRef](#)]
31. 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](#)]
32. Howard, A.G.; Zhu, M.; Chen, B.; Kalenichenko, D.; Wang, W.; Weyand, T.; Andreetto, M.; Adam, H. Mobilenets: Efficient convolutional neural networks for mobile vision applications. *arXiv* **2017**, arXiv:1704.04861.
33. 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](#)]
34. Aslan, S.; Güdükbay, U.; Töreyin, B.U.; Çetin, A.E. Early wildfire smoke detection based on motion-based geometric image transformation and deep convolutional generative adversarial networks. In Proceedings of the ICASSP 2019—2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Brighton, UK, 12–17 May 2019; pp. 8315–8319.
35. Hashemzadeh, M.; Zademedhi, A. Fire detection for video surveillance applications using ICA K-medoids-based color model and efficient spatio-temporal visual features. *Expert Syst. Appl.* **2019**, *130*, 60–78. [[CrossRef](#)]
36. Xu, G.; Zhang, Y.; Zhang, Q.; Lin, G.; Wang, Z.; Jia, Y.; Wang, J. Video smoke detection based on deep saliency network. *Fire Saf. J.* **2019**, *105*, 277–285. [[CrossRef](#)]

37. Majid, S.; Alenezi, F.; Masood, S.; Ahmad, M.; Gündüz, E.S.; Polat, K. Attention based CNN model for fire detection and localization in real-world images. *Expert Syst. Appl.* **2022**, *189*, 116114. [[CrossRef](#)]
38. Reddy, G.; Avula, S.; Badri, S. A novel forest fire detection system using fuzzy entropy optimized thresholding and STN-based CNN. In Proceedings of the 2020 International Conference on COMmunication Systems & NETworkS (COMSNETS), Bengaluru, India, 7–11 January 2020.
39. Kim, B.; Lee, J. A video-based fire detection using deep learning models. *Appl. Sci.* **2019**, *9*, 2862. [[CrossRef](#)]
40. Peng, Y.; Wang, Y. Real-time forest smoke detection using hand-designed features and deep learning. *Comput. Electron. Agric.* **2019**, *167*, 105029. [[CrossRef](#)]
41. Chino, D.Y.; Avalhais, L.P.; Rodrigues, J.F.; Traina, A.J. 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, 26–19 August 2015; pp. 95–102.
42. Saied al Saied Fire Dataset. 2021. Available online: [https://www.kaggle.com/datasets/phylake1337/fire-dataset?select=fire\\_dataset](https://www.kaggle.com/datasets/phylake1337/fire-dataset?select=fire_dataset) (accessed on 20 June 2023).
43. Parez, S.; Dilshad, N.; Alanazi, T.M.; Lee, J.W. Towards Sustainable Agricultural Systems: A Lightweight Deep Learning Model for Plant Disease Detection. *Comput. Syst. Eng.* **2023**, *47*, 515–536. [[CrossRef](#)]
44. Parez, S.; Dilshad, N.; Alghamdi, N.S.; Alanazi, T.M.; Lee, J.W. Visual Intelligence in Precision Agriculture: Exploring Plant Disease Detection via Efficient Vision Transformers. *Sensors* **2023**, *23*, 6949. [[CrossRef](#)]
45. Khan, H.; Haq, I.U.; Munsif, M.; Mustaqeem; Khan, S.U.; Lee, M.Y. Automated wheat diseases classification framework using advanced machine learning technique. *Agriculture* **2022**, *12*, 1226. [[CrossRef](#)]
46. Dilshad, N.; Hwang, J.; Song, J.; Sung, N. Applications and challenges in video surveillance via drone: A brief survey. In Proceedings of the 2020 International Conference on Information and Communication Technology Convergence (ICTC), Jeju, Republic of Korea, 21–23 October 2020; pp. 728–732.
47. Zahir, S.; Khan, R.U.; Ullah, M.; Ishaq, M.; Dilshad, N.; Ullah, A.; Lee, M.Y. Robust Counting in Overcrowded Scenes Using Batch-Free Normalized Deep ConvNet. *Comput. Syst. Eng.* **2023**, *46*, 2741–2754. [[CrossRef](#)]
48. Dilshad, N.; Ullah, A.; Kim, J.; Seo, J. Locateuav: Unmanned aerial vehicle location estimation via contextual analysis in an iot environment. *IEEE Internet Things J.* **2022**, *10*, 4021–4033. [[CrossRef](#)]
49. Dilshad, N.; Song, J. Dual-Stream Siamese Network for Vehicle Re-Identification via Dilated Convolutional layers. In Proceedings of the 2021 IEEE International Conference on Smart Internet of Things (SmartIoT), Jeju, Republic of Korea, 13–15 August 2021; pp. 350–352.
50. Ullah, M.; Amin, S.U.; Munsif, M.; Safaev, U.; Khan, H.; Khan, S.; Ullah, H. Serious games in science education. A systematic literature review. *Virtual Real. Intell. Hardw.* **2022**, *4*, 189–209. [[CrossRef](#)]
51. Khan, H.; Ullah, M.; Al-Machot, F.; Cheikh, F.A.; Sajjad, M. Deep learning based speech emotion recognition for Parkinson patient. *Image* **2023**, *298*, 2. [[CrossRef](#)]
52. Munsif, M.; Ullah, M.; Ahmad, B.; Sajjad, M.; Cheikh, F.A. Monitoring neurological disorder patients via deep learning based facial expressions analysis. In *IFIP International Conference on Artificial Intelligence Applications and Innovations*; Springer: Berlin/Heidelberg, Germany, 2022; pp. 412–423.
53. Munsif, M.; Afridi, H.; Ullah, M.; Khan, S.D.; Cheikh, F.A.; Sajjad, M. A lightweight convolution neural network for automatic disasters recognition. In Proceedings of the 2022 10th European Workshop on Visual Information Processing (EUVIP), Lisbon, Portugal, 11–14 September 2022; pp. 1–6.
54. Woo, S.; Park, J.; Lee, J.Y.; Kweon, I.S. Cbam: Convolutional block attention module. In Proceedings of the European Conference on Computer Vision (ECCV), Munich, Germany, 8–14 September 2018; pp. 3–19.
55. Chollet, F. Xception: Deep learning with depthwise separable convolutions. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 1251–1258.
56. Wightman, R.; Touvron, H.; Jégou, H. Resnet strikes back: An improved training procedure in timm. *arXiv* **2021**, arXiv:2110.00476.
57. Tan, M.; Le, Q. Efficientnet: Rethinking model scaling for convolutional neural networks. In Proceedings of the International Conference on Machine Learning, Long Beach, CA, USA, 10–15 June 2019; pp. 6105–6114.
58. Zoph, B.; Vasudevan, V.; Shlens, J.; Le, Q.V. Learning transferable architectures for scalable image recognition. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–23 June 2018; pp. 8697–8710.
59. Simonyan, K.; Zisserman, A. Very deep convolutional networks for large-scale image recognition. *arXiv* **2014**, arXiv:1409.1556.
60. Yar, H.; Hussain, T.; Khan, Z.A.; Koundal, D.; Lee, M.Y.; Baik, S.W. Vision sensor-based real-time fire detection in resource-constrained IoT environments. *Comput. Intell. Neurosci.* **2021**, *2021*, 5195508. [[CrossRef](#)] [[PubMed](#)]
61. Khan, A.; Hassan, B.; Khan, S.; Ahmed, R.; Abuassba, A. DeepFire: A Novel Dataset and Deep Transfer Learning Benchmark for Forest Fire Detection. *Mob. Inf. Syst.* **2022**, *2022*, 5358359. [[CrossRef](#)]

**Disclaimer/Publisher’s Note:** The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.