MDPI

Review

Recent Advances and Emerging Directions in Fire Detection Systems Based on Machine Learning Algorithms

Bogdan Marian Diaconu

Engineering Faculty, University "Constantin Brancusi" of Tg Jiu, 30 Calea Eroilor, 210135 Targu Jiu, Romania; bogdan.diaconu@e-ucb.ro

Abstract: Fire detection is a critical safety issue due to the major and irreversible consequences of fire, from economic prejudices to loss of life. It is therefore of utmost importance to design reliable, automated systems that can issue early alarms. The objective of this review is to present the state of the art in the area of fire detection, prevention and propagation modeling with machine learning algorithms. In order to understand how an artificial intelligence application penetrates an area of fire detection, a quantitative scientometric analysis was first performed. A literature search process was conducted on the SCOPUS database using terms and Boolean expressions related to fire detection techniques and machine learning areas. A number of 2332 documents were returned upon the bibliometric analysis. Fourteen datasets used in the training of deep learning models were examined, discussing critically the quality parameters, such as the dataset volume, class imbalance, and sample diversity. A separate discussion was dedicated to identifying issues that require further research in order to provide further insights, and faster and more accurate models.. The literature survey identified the main issues the current research should address: class imbalance in datasets, misclassification, and datasets currently used in model training. Recent advances in deep learning models such as transfer learning and (vision) transformers were discussed.

Keywords: fire detection; fire prevention; machine learning; deep learning; computer vision; dataset

1. Introduction

Fire represents an event with potential catastrophic consequences—property and environment damage and bodily injury—if not properly contained and the combustion prerequisites are met. This is especially applicable to forest fires (Saha et al. [1], Kala [2], Elogne et al. [3]), the chemical and oil industry (Lowesmith et al. [4], Aydin et al. [5], Solukloei et al. [6]), various process industries (Park et al. [7], Zhou and Reniers [8]), buildings (Østrem and Sommer [9], Ibrahim et al. [10]), transportation (Wang et al. [11]), and ecosystems and wildlife (Davies et al. [12], Lindenmayer et al. [13], Liu et al. [14], Batista et al. [15]).

Fire prevention is the first line of defense against large-scale-disaster-caused fires, while fire detection—coupled with fire containment and warning systems—can be viewed as the last line of defense. Conventional fire detection relies on sensory systems sensitive to fire by-products, such as smoke, heat or gaseous compounds resulting from the combustion process. Smoke and gas detectors are not effective outdoors, and in indoor conditions and they can be activated inadvertently. Video surveillance cameras in the visible spectra can only detect flame when the fire has already fully developed. Smoke and heat detectors require a temperature high enough or a minimum smoke density to activate, limiting their applicability to the early detection of small fires. Indirect fire detection based on sensing the presence of smoke consists of measuring the backscattered light emitted by a light source (emitting on a very narrow spectrum) and reflected on smoke particles. These devices usually have a long response time and cannot detect fires that do not generate smoke. Since most flame types generate some type of gases, systems based

on their detection provide faster and more accurate results. Metal-oxide gas sensors are the most common fire detection systems based on this principle, Curbat et al. [16], Derbel [17]. Smoke detectors are designed and operate based on the standard definition of smoke: "the airborne solid and liquid particulates and gases evolved when a material undergoes pyrolysis or combustion" [18]. Based on this definition, smoke detectors refer only to the detection of fire particulates and not to gas detection. A comprehensive review on chemical sensors for fire detection was conducted by Fonollosa et al. [19]. Gas sensors offer an advantage over smoke sensors as they can operate faster than smoke sensors (gases are usually released faster than smoke). Another advantage of gas sensors is the fact that by triggering a gas alarm, they provide protection to building occupants (fire in buildings produces more casualties due to toxic emissions rather than actual burns) [20].

A critical issue of smoke detectors is their inability to discriminate between smoke particles from genuine fires and particles generated by other sources, such as dust or dense water vapor. This results in a high rate of FPs, which raises serious concerns related to costs. For example, in a briefing paper Chagger and Smith [21] reported that the Fire and Rescue Service Authorities claimed that the associated cost of false alarms was as high as 1 billion pounds per year. The same source reported that 53% of the alarms were FPs during the period 2011–2012.

The correct discrimination of genuine fires from nuisances (especially in early phases) from processing information from a single sensor has been long recognized as very difficult and error-prone (Ishii et al. [22]). Multi-sensor systems for heat, CO/combustion by-products and smoke detectors have been considered to overcome this issue. Such systems, consisting of multiple different sensors, operate in conjunction with algorithmic solutions that improve the overall sensitivity compared to individual sensor sensitivity. For example, adding heat and CO sensors to smoke sensors improves the sensors' performance significantly, (Fonollosa et al. [19]).

This review paper presents recent studies and perspectives on machine learning (ML) algorithms applied for fire detection systems. A special focus was set on discussions of the performance metrics, datasets used for machine learning models training (structure, class imbalance, etc.) and misclassification issues. In order to produce more precise models, a workflow was suggested in the Conclusion section based on a misclassification cost function that accounts for case-specific implications of FPs and FNs.

2. Materials and Methods

A bibliometric analysis was conducted in the Scopus database using the following Boolean expression.

(KEY ("fire" "deep learning") OR KEY ("fire" "artificial intelligence") OR KEY ("fire" "machine learning") OR KEY ("fire detection" "machine learning")) returned 2332 documents, which were exported as Scopus files while retrieving the author keywords and indexing keywords fields. The file was processed by means of the bibliometric analysis tool VOSviewer (a software tool for constructing and visualizing bibliometric networks) conducting a co-occurrence analysis of keywords. A total number of 13,521 keywords were identified and the minimum number of occurrences of a keyword was set to 10. Out of the total number of keywords, 505 keywords met the threshold. The first 10 keywords (sorted by the total link strength) are presented in Table 1.

The co-occurrence graph of the keywords is presented in Figure 1. A number of six clusters containing 148, 115, 107, 48, 46 and 41 items were generated. It is interesting to note that the term "deforestation", although not included in the search expression, occurred 401 times, suggesting a significant research effort in the direction of the relationship between fires and forest degradation.

In order to gain further insight into the terms and the relationships between them occurring in the main flux of publications on the topic of fire detection using artificial intelligence, a co-occurrence analysis was conducted on the abstracts of the documents returned by the search process. A term was included if it occurred more than 20 times.

The list of terms was cleaned, removing the terms with general use (no necessarily related to the topic), resulting a number of 235 items. The analysis tool returned three clusters containing 112, 66 and 57 items, respectively. The co-occurrence graph resulting from the analysis of the abstracts is presented in Figure 2.

Table 1. Top ten keywords and resulting from Scopus query.

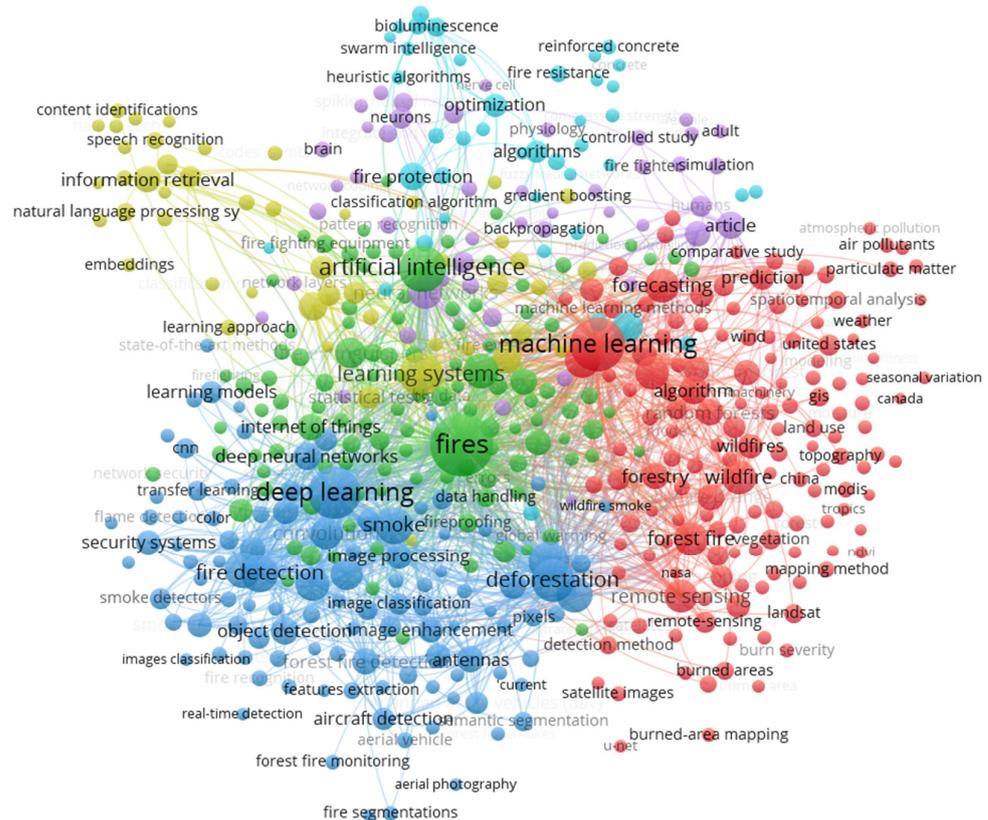| Pos	Keyword	Occurrences	Total Link Strength
1	"fires"	1315	14,293
2	"machine learning"	836	8501
3	"deep learning"	736	7665
4	"deforestation"	401	5339
5	"fire hazards"	422	4919
6	"learning systems"	348	4696
7	"fire detection"	535	4167
8	"fire detectors"	259	3424
9	"smoke"	269	3191
10	"forest fires"	277	2960

Figure 1. Co-occurrence graph for key words from the Scopus query results.

From Figures 1 and 2 and from Table 1, it follows that many items returned by the SCOPUS search engine may discuss topics diverging from the review topic, that is fire detection with machine learning algorithms. Further analysis was performed, as presented in the flowchart in Figure 3. A number of 136 articles were selected for full-text review based on the exclusion criteria. Another 42 more references with relevance to the topic but not directly related were included, resulting a total number of 178 references.
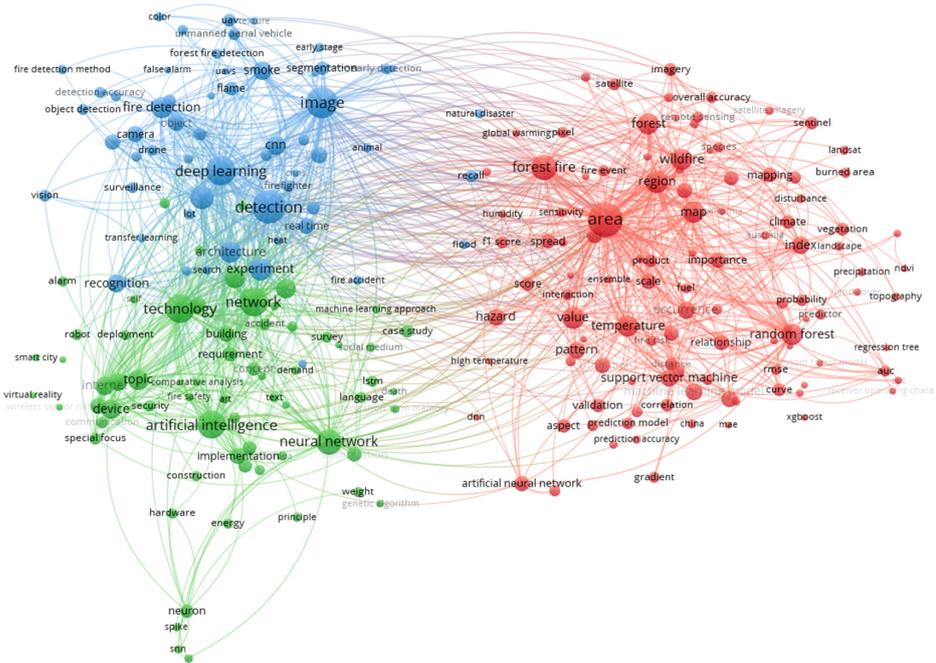
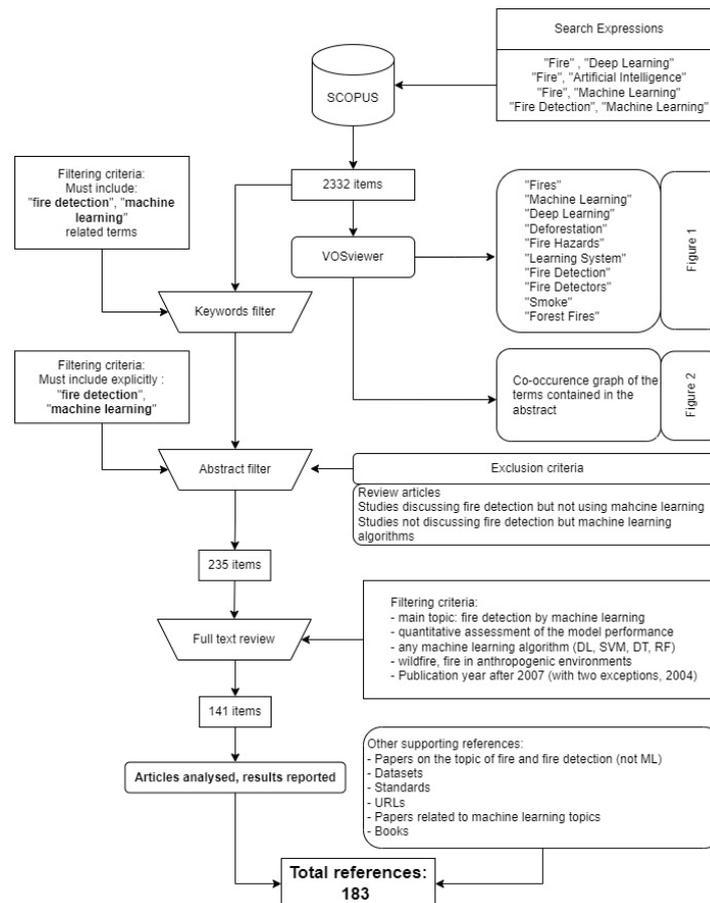**Figure 2.** The co-occurrence graph of terms contained in the abstracts.

Figure 3. The flowchart of the selection of the research articles discussed in this review.

3. Results and Discussion

3.1. Machine Learning and Deep Learning Techniques in Fire Detection Sensory Systems

Machine learning (ML) and deep learning (DL) techniques have developed recently at a very high rate and have been implemented in sensor systems in order to improve their precision and sensitivity. The main purpose of applying ML techniques to fire detection is to improve its sensitivity, detect genuine fires in early stages and not trigger false alarms. In terms of ML terminology, fire detection is a binary (or sometimes a multi-class) classification problem. The standard metrics for the assessment of a classification ML algorithm, precision, recall, sensitivity, specificity, the F1-score and the ROC curve must be assessed in accordance with the costs associated with false positives (FP) and false negatives (FN). Generally, in fire detection problems, an FN incurs costs significantly higher than an FP. Therefore, both precision and recall metrics are important, which up to a point, justifies the selection of the F1-score as a relevant metric. The ROC curve is also important in calibrating the ML model and selecting the optimal hyperparameters in order to minimize the misclassification costs. ML techniques have been extensively used in remote sensing, for example for geological surveys, Han et al. [23]. Typical ML techniques in sensory systems are Decision Trees (DTs) and Random Forests (RFs), Support Vector Machines (SVMs), k-Nearest Neighbor and K-Means Clustering. As suggested by many studies, ensemble methods [24] (bagging, stacking and boosting) offer significantly better performance than individual algorithms. The DL techniques with large applicability are the Neural Networks (NNs), Convolutional Neural Networks (CNNs), various types of recurrent neural networks, Long Short-Term Memory (LSTM), and Gated Recurrent Units (GRU)

DL techniques are different from ML techniques in the following ways:

- DL are more suitable for big data and for data with a large number of features, such as images or video streams. However, the computational cost of DL techniques is significantly higher than that of ML techniques; in some cases, expensive hardware with special capabilities is required, such as GPUs.
- Feature extraction is performed automatically in the case of DL techniques (such as CNNs processing images or video streams). This is not possible in the case of ML techniques, which require some form of manual feature extraction.

In the case of fire detection using machine learning algorithms, the choice of the algorithms depends mainly on the fire settings and layout. The following settings in which occurrence of fire can have serious to catastrophic effects were identified in the literature:

3.1.1. Forest Fire

Forest fires have several particular features, and multidimensional and synergetic effects, distinguishing them from other fire types:

- Due to the large areas on which they can occur, forest fires are extremely difficult to detect in early stages, when suppression could be achieved effortlessly;
- Forest fires are a non-negligible source of GHG emissions (CO, CO_2, CH_4, NO_x and particulate matter) and have a direct effect on the atmospheric chemistry and atmospheric heat budget (Saha et al. [1]);
- Forest fires cause a decrease in the terrestrial ecosystem's productivity and exhaustion of the forest environment carbon stock, Amiro et al. [25];
- Forest fires cause soil fertility, crop productivity and water quality and quantity degradation, Venkatesh et al. [26].

3.1.2. Civil and Industrial Facilities

Fires occurring in industrial facilities have effects that depend mostly on the flammable material. CO, CO_2, and smoke are the fire by-products occurring most frequently, independently of the flammable material's nature. In civil structures, such as tunnels or confined spaces, the rapid oxygen exhaustion is another life-threatening factor that poses addi-

tional risks to the personnel, including fire-fighting personnel. The nature of the industrial environment can cause avalanche effects caused by fires, such as explosions.

3.1.3. Office and Residential Facilities

In such cases, the flammable materials consist mainly of textiles, furniture and plastic material. These types of fires cause the release of volatile organic compounds (approximately two times more than in the case of industrial fires, Alharbi et al. [27]). In the case of such fires, the main factor that contributes to casualties consists of smoke and poisonous gases resulting from combustion.

3.2. Machine Learning Algorithms for Forest Fire Detection

Early efforts to develop accurate and precise early warning systems for forest fires were reported as early as the 2000s. Iliadis et al. [28] developed a fuzzy sets-based expert system to assign a forest fire risk score to the prefectures of Greece. The basic unit of the expert system presented in [28] is the fuzzy expected interval, defined as narrow intervals of values that provide the best description of the forest fire problem in a country or a part of a country for a certain time period. The advent of remote sensing and satellite imagery offered a new development direction for fire detection systems.

Real Time Object Detection (RTOD) algorithms include CNN family (R-CNN, Region-Based Convolutional Neural Networks, Fast R-CNN), and YOLO (You Only Look Once). YOLOv3 is currently considered the RTOD algorithm that offers the best trade-off between accuracy and detection speed, Li et al. [29]. YOLOv3 is based on DarkNet53, which consists of 53 convolutional layers, including the last fully connected layer. The YOLOv3 architecture is presented in Figure 4 [30].
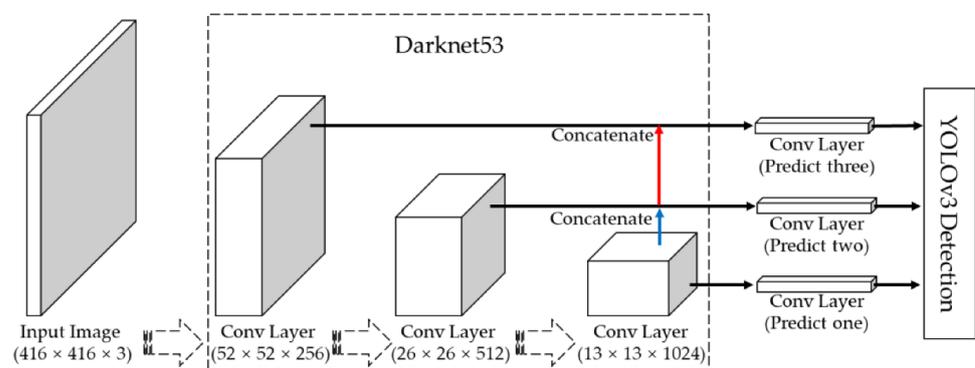
Figure 4. YOLOv3 architecture. Blue and red lines represent two-fold up-sampling. Reprinted/adapted from Ref. [30] under the terms and conditions of the CC BY license. ©2019, MDPI.

Based on the YOLOv3 architecture, Li et al. [29] proposed a novel architecture (ALFR-Net), presented in Figure 5 (details on the implementation of the blocks DLFR, HAG and ADC are given in [29]).

The performance of the ALFRNet architecture was assessed by means of the P-R curve considering two values of the IOU (Intersection Over Union) parameter, as shown in Figure 6.

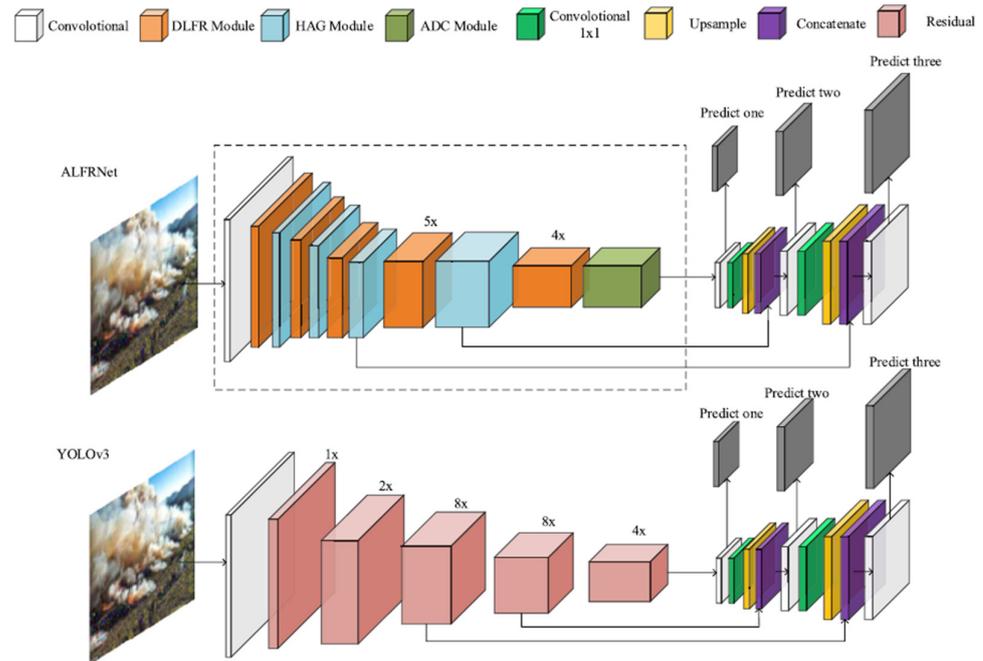**Figure 5.** The architecture structure proposed in [29] compared to YOLOv3. Reprinted/adapted with permission from Ref. [29]. ©2022, Elsevier.
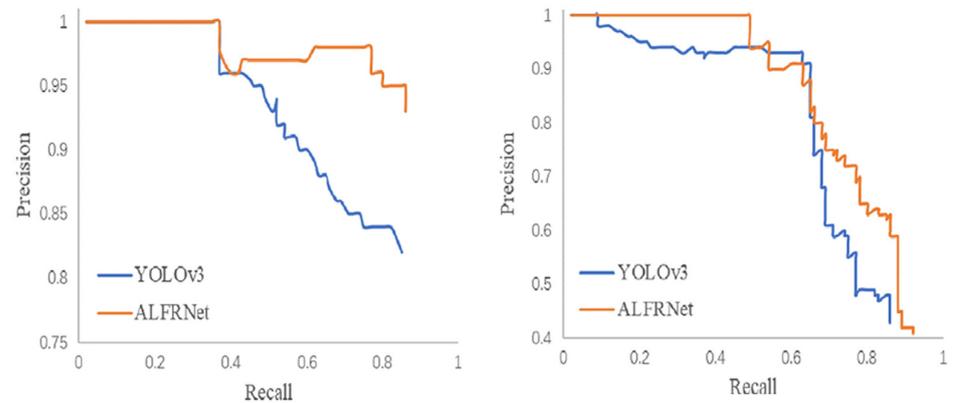

Figure 6. P-R curves for IOU parameter 0.5 (**left**) and 0.75 (**right**). Reprinted/adapted with permission from Ref. [29]. ©2022, Elsevier.

The original dataset consisted of 2431 smoke images obtained by a web crawler. Data augmentation was applied through methods such as High-pass filtering, Gamma transformation, Rotation, Retinex, Laplace Transformation, Salt-and-pepper noise addition and Gaussian Noise addition. The augmented dataset contained 4326 images. A comparison with YOLOv3 from the performance point of view was conducted based on the mean average accuracy metric (mAP), which is a quantitative indicator to assess the detection effect of multi-category targets, defined as

$$mAP = \int_0^1 P(R)dR$$

Another performance indicator was the Frame per Second, which estimates the detection speed. The FPS is defined as the inverse of the time required to process each picture frame. The performance of the ALFRNet compared to YOLOv3 is presented in Table 2.

Table 2. ALFRNet vs. YOLOv3 performance comparison. Reprinted/adapted with permission from Ref. [29]. ©2022, Elsevier.

Architecture	$mAP_{IOU=0.5}$[%]	$mAP_{IOU=0.75}$[%]	FPS	No. of Parameters
YOLOv3	84.94	74.77	33	61,576,342
ALFRNet	87.26	79.95	43	23,336,302

The architecture proposed in [29] reached a better performance (assessed through mAP) and a significantly higher detection speed. The ALFRNet number of parameters was less than half compared to those of YOLOv3, which allows deployment on a lighter hardware.

Shamsoshoara et al. [31] produced a two-class ("fire" and "no fire") labeled dataset containing aerial video recordings and heatmaps of fires, collected by drones during a pre-scribed burning piled detritus in a pine forest (located in Northern Arizona). Two machine learning problems were discussed in [31]: (1) the binary classification of video frames and (2) developing segmentation methods to determine the flame borders. The binary classi-fication model was based on the Xception network (proposed by Chollet [32]). Xception was developed by replacing the standard Inception modules with depth-wise separable convolutions. The training set consisted of 25,018 frames of type "fire" and 14,357 frames of type "non-fire". Data augmentation methods such as flipping and random rotation were used to generate new frames and address the issue of bias for unbalanced classes. A test 8617-frame dataset consisting of frames including 5137 fire-labeled frames and 3480 non-fire frames was used to assess the loss and accuracy, as presented in Table 3:

Table 3. Accuracy and loss for the test, validation and training datasets [31]. Reprinted/adapted with permission from Ref. [31]. ©2021, Elsevier.

Dataset	Accuracy	Loss
Test	0.7623	0.7414
Validation	0.9431	0.1506
Training	0.9679	0.0857

From Table 3, it can be observed that the model has a significant underfitting issue (the accuracy value is much lower for the test set than for the training set). From the analysis of the confusion matrix presented in Table 4, it can be observed that a significant number of false negatives (799) were predicted. The P and R values (not provided in the article) were 0.757 and 0.8295, respectively.

Table 4. The confusion matrix [31].

		Predicted Label	
		Fire	Non-Fire
True	Fire	3888	799
label	Non-Fire	1248	2680

The classification and segmentation models are presented in Figure 7 left and right, respectively.

Six samples of the test set along with the expected ground truth masks and the generated masks from the trained network are presented in Figure 8. The first image row consists of the input frames to the model and the second image row represents the ground truth (the expected mask). The third image row represents the mask generated by the trained network. The image segmentation algorithm performance was assessed in terms of P (0.92), R (0.84), ROC-AUC (0.9985), the F1-Score (0.8775), sensitivity (0.8312), specificity (0.9996) and the IoU (0.7817).

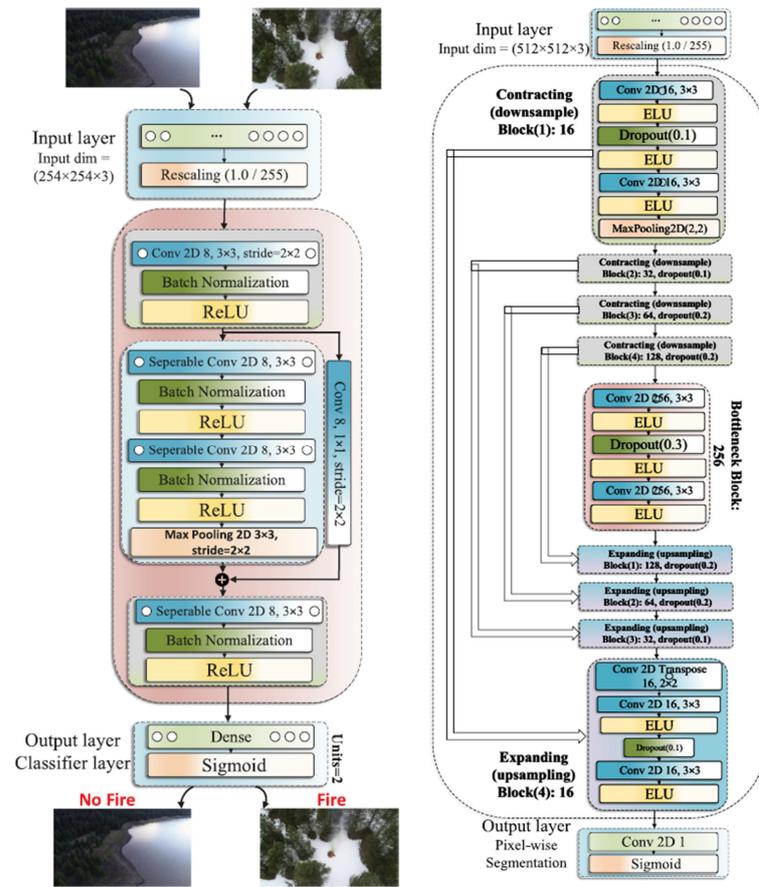**Figure 7.** The classification (**left**) and the segmentation models (**right**). Reprinted/adapted with permission from Ref. [31]. ©2021, Elsevier.

A summary of studies on fire detection based on image processing and deep learning is presented in Table 5.
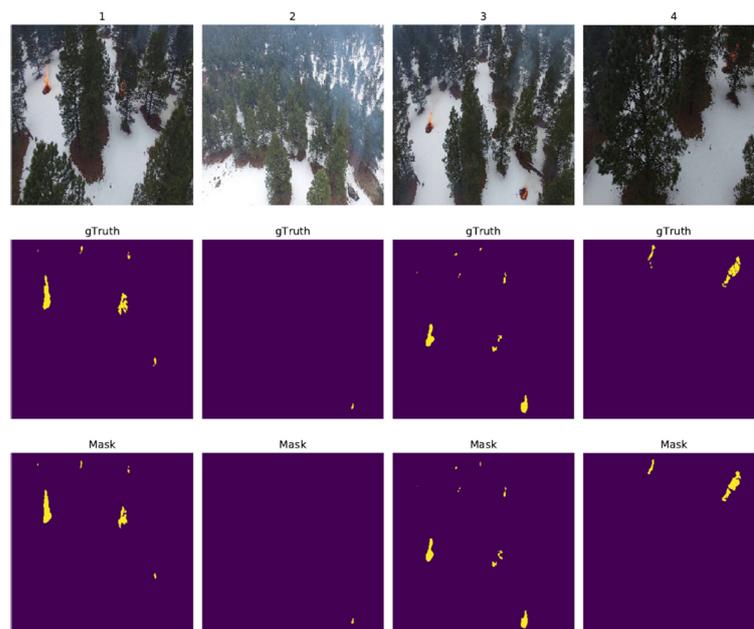

Figure 8. The performance of the segmentation algorithm for six frames from the test set. Reprinted/adapted with permission from Ref. [31]. ©2021, Elsevier.

Table 5. Deep learning custom architectures and models for fire detection through image processing.

Ref	Dataset	Fire Type	Architecture	Metrics
Zhan [33] (2022)	I. Data of 6913 smoke images intercepted from the video dataset by using screenshots. II. A total of 5832 images from the video dataset of the FLAME dataset [31]. III. A total of 1093 images simulated smoke generation when a forest fire occurred.	Forest	ARGNet adjacent layer composite network based on a recursive feature pyramid with deconvolution and dilated convolution and global optimal non-maximum suppression.	mAP50: 90 mAP75: 82 FPS: 122 Jaccard index: 0.84
Zhang [34] (2016)	Images captured from a video stream (every five frames). A total of 21 positive and 4 negative sequences. Images were resized at 240×320. Fire patches were annotated manually with 32×32 boxes. Train set: 178 images, 12,460 patches (1307 positive, 11,153 negative). Test set: 59 images, 4130 patches (539 positive, 3591 negative).	Forest	Three convolutional layers Three pooling layers Two fully connected layers Softmax activation in the output layer	Accuracy: 0.931 Detection rate: 0.845 False positive rate: 0.039 Time per image: 2.1 s
Reis [35] (2023)	FLAME dataset [31]. A total of 14,357 negative and 25,018 positive instances in the training dataset, and 3480 no-fire and 5137 fire instances in the test dataset.	Forest	DenseNet121 (Huang [36]) CNN-based architecture with each layer directly connected to all previous layers. A total of 121 convolution layers, four dense blocks, three transition layers, and a SoftMax layer in the output layer. Image input $224 \times 224 \times 3$.	Accuracy P R F1-Score Cohen's Kappa Confusion matrix
Song [37] (2021)	UIA-CAIR/Fire-Detection-Image—Dataset (620 three-channel JPEG files) consisting of 110 positive samples and 510 negative samples. Image augmentation (translation, flipping and rotation) was applied to the positive samples in order to balance the classes. The total number of images after data augmentation was 950 (190 training images, 760 test images), normalized to $64 \times 64 \times 3$.	Outdoor	C1: 64 5×5 convolution kernels by a step size 2. Pooling layer (MaxPooling). C2: 128 5×5 convolution kernels by a step size 2. Pooling layer (MaxPooling). C3: 256 5×5 convolution kernels by a step size 2. Pooling layer (MaxPooling). 1024-node fully connected layer Activation function: LeakyReLU (C1-C2), ReLU (C3).	Accuracy: For the training set 99.5%; For the test set 94%.

Table 5. *Cont.*

Ref	Dataset	Fire Type	Architecture	Metrics
Fernandes [38] (2004)	Committee of neural networks connected sequentially; each unit eliminates the pattern that it classifies as atmospheric noise and passes on the pattern that is considered smoke. The model was trained with the same smoke signatures and noise patterns that caused false alarms in the previous neural network.	Forest	A total of 1410 patterns containing smoke signatures, 17,174 atmospheric noise patterns with 21 points and 8334 atmospheric noise patterns with 41 points for each value of the added noise amplitude.	False alarm (%): 0.9 Misdetections (%): 15.5
Ahn [39] (2023)	A total of 10,163 images of early and indoor fires from AI HUB [40]. False fire alarm response items for video fire detectors: direct sunlight and solar-related sources, electric welding, black body sources (electric heater), and artificial lighting (incandescent lamp, fluorescent lamp, halogen lamp), candles, rainbow, laundry, flags, red cloths, yellow wires, light reflections, and cigarette smoke.	Indoor	YOLOv5 pre-trained model. The default YOLOv5 hyperparameters values were used. The epoch number was 300 and batch size 32. The training, validation, and test dataset volumes were 70, 20, and 10% of the image dataset.	P (Val/Test): 0.94/0.91 R (Val/Test): 0.93/0.97 mAP: 0.96/0.96
Biswas [41] (2023)	CNN. Pretrained Inception-ResNet-v2 [42] (164 layers, 1000 output classes). Inception-ResNet-v3 [43]. Modified Inception-ResNet-v3 [41].	Indoor/Outdoor	Three-channel JPEG images. Training set: 694 positive class, 286 negative class. Validation set: 171 positive class, 68 negative class	Training and validation accuracy.
Muhammad [44] (2018)	CNN consisting of five convolution layers, three pooling layers, and three fully connected layers. Input layer size 224 × 224 × 3	Indoor/Outdoor	A. Foggia [45]: 31 videos in indoor and outdoor environments. A total of 14 positive class and 17 negative class. B. Chino [46]: 226 images: 119 positive class, 107 negative class consisting of fire-like images: sunsets, fire-like lights, and sunlight coming through windows	FPR: w FT: 0.0907 w/o FT: 0.0922 FNR: w FT: 0.0213 w/o FT: 0.106 Accuracy: w FT: 0.94 w/o FT: 0.90

Table 5. *Cont.*

Ref	Dataset	Fire Type	Architecture	Metrics
Yar [47] (2023)	YOLOv5-based, modified to detect both small fire areas and large fire areas.	Building(B) Indoor electric(I) Vehicle(V)	Building fire: 723 Indoor electric fire: 118 Vehicle fire: 1116	P (B/I/V): 0.8/0.95/0.87 R (B/I/V): 0.64/0.95/0.86 F1-Score (B/I/V): 0.7/0.95/0.86 mAP (B/I/V): 0.7/0.96/0.92
Valikhujaev [48] (2020)	CNN consisting of four convolution layers with dilation rate 1, 2 and 3, each convolution layer followed by a MaxPooling layer; a final Dropout layer; and two FC layers each followed by a Dropout layer.	Indoor/Outdoor	Total of 8430 fire images. Total of 8430 smoke images.	Train set accuracy: 0.996 Test set accuracy: 0.995 R: 0.97 Inference time

3.3. Smoke Detection

Especially during the early stages of the fire, smoke is almost translucent and it has a low contrast, diffuse boundaries and a rapidly changing shape. Standard smoke detection methods employ image processing approaches to extract smoke features, Gong et al. [49]. Other approaches include frameworks for high-definition video streams using fused spatial and frequency domain features, Liu et al. [50], wavelet analysis, Gubbi et al. [51] and statistical properties, Yuan et al. [52]. Deep learning-based algorithms treat the smoke detection task as either an image classification problem (Muhammad et al. [53]) or a video classification problem (Lin et al. [54]). A less conventional approach was reported by Lin et al. [54], where smoke detection was achieved through a 3D convolutional network, extracting both spatial and temporal features. In general, these deep learning models are trained on smoke datasets collected in controlled environments. The problem occurs when weather conditions (fog, rainfall, nebulosity, etc.) considerably modify the conditions under which the algorithm was trained. A standard approach to deal with these occurrences is dehazing the images, Berman et al. [55]. More advanced methods include synthesizing fog on images or developing smoke datasets in foggy environments, Gong et al. [56]. Sathishkumar et al. [57] employed transfer learning models VGG16, InceptionV3 and Xception and a heterogenous dataset collected from several sources to develop models with the purpose of detecting fire and smoke in still images. A four-classe dataset, as presented in detail in Table 6, was used.

Table 6. The dataset considered in [57]. Reproduced/adapted from [57] under a Creative Commons Attribution 4.0 International License.

Dataset	Classes			
	Fire	No Fire	Smoke	Smoke Fire
Train	2161	2150	490	510
Validation	200	200	200	200
Test	200	200	200	200

Due to the unbalanced nature of the dataset (as it can be observed in Table 6), image augmentation techniques (shifting, rotating, scaling, flipping, blurring, padding, translation and affine transformations) were applied. The transfer learning models were trained in two modes, Feature Extractor mode and Fine-Tuning mode. The performance (measured by the confusion matrix) was considerably higher in the Fine-Tuning mode.

The studies summarized in Table 5 aimed mainly at detecting the flame caused by fires. Depending on the fire environment and flammable material, smoke can be released in early phases of the fire, before the flame develops fully and extinguishing becomes more difficult.

For this reason, the accurate and rapid detection of smoke can contribute significantly to the successful extinguishing of fires. It is known that smoke is more difficult to detect than flame (Muhammad et al. [44]). Smoke detection techniques in images based on manually designed features involve identification of the color, shape, texture and other specific features and their variations in different scenes, followed by analysis and discrimination by means of machine learning classifiers, Wu et al. [58]. The principal difficulty in smoke detection in images based on machine learning techniques is the fact that smoke changes rapidly and randomly, and the uncertainty of different types of features is significant, so the manual-based methods cannot express accurately the essential features of smoke. This results in poor accuracy and robustness for a real application. Deep learning techniques have good performance in smoke detection applications, as shown in LeCun et al. [59]; however, there are several issues and limitations, such as a long training time, a necessity for large and diverse datasets, and their deployment on embedded platforms (Zeng et al. [60]). Smoke segmentation is considerably more challenging than smoke detection, requiring the accurate identification and separation of smoke components from the background scene at the pixel level, followed by a transformation from coarse-grained recognition to fine-

grained recognition. The object is thus represented as a set of fine pixels, and then pixels with similar features are clustered together and defined as the same class, Khan et al. [61]. Max-pooling and average-pooling operations are considered suitable for capturing local dominant details and global contexts of objects such as smoke, Yuan et al. [62].

Khan et al. [61] developed a smoke detection and segmentation framework for clear and hazy environments based on a Convolutional Neural Network. The full pipeline proposed in [61] consists of two modules: (i) smoke detection module using a lightweight CNN model and (ii) a module performing semantic smoke segmentation. The architecture proposed in [61] is presented in Figure 9.
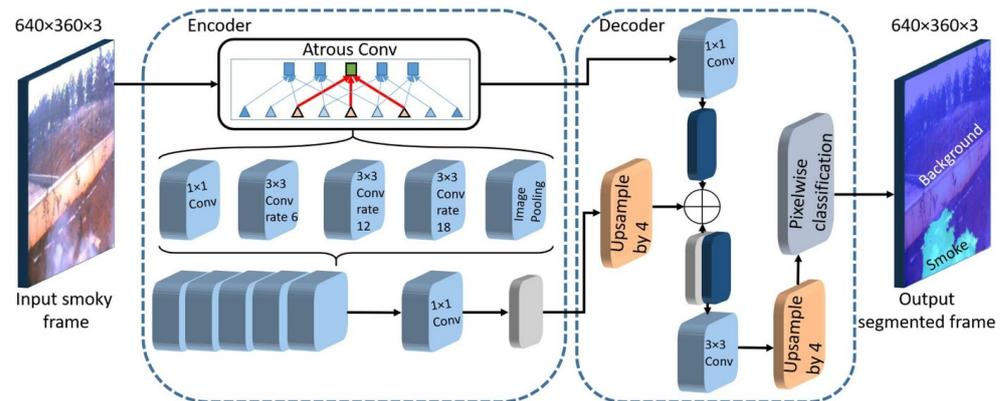

Figure 9. The architecture of the smoke segmentation model presented in [61]. Reproduced/adapted with permission from Ref. [61]. ©2021, Elsevier.

Three datasets were used in [61], as follows:

- DS1, consisting of four different classes: (1) "smoke", (2) "non smoke", (3) "smoke with fog", and (4) "non-smoke with fog". The total number of images was 72,012 with 18,532 for each "smoke" and "smoke with fog" and 17,474 for "non-smoke" and "non-smoke with fog", individually.
- DS2, consisting of seven videos of smoke, captured in different environments: smoke at long distance, smoke in a parking lot with other moving objects, and smoke released by a burning cotton rope.
- DS3, consisting of 252 images annotated from DS1 by considering two classes ("smoke" and "smoke with fog").

The performance of the proposed framework (DeepSmoke) was compared to Seg-Net (Badrinarayanan et al. [63]), DeepLab (Chen et al. [64]), FCN (Long et al. [65]), and DeconvNet (Noh et al. [66]), as shown in Figure 10.
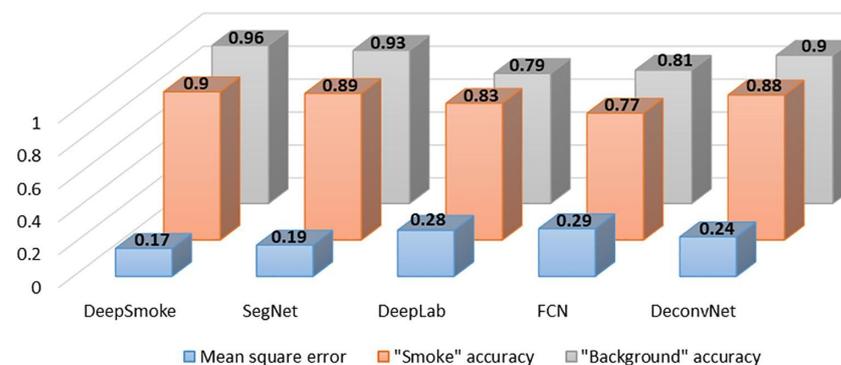

Figure 10. DeepSmoke [61] performance compared to other architectures. Reproduced/adapted with permission from Ref. [61]. ©2021, Elsevier.

Ma et al. [67] proposed a four-stage smoke segmentation algorithm based on improved double Truncation distance Self-adaptive Density Peak Clustering (TSDPC):

- Super pixel segmentation;
- Redefinition of sample point local density;
- Adaptive double truncation distance;
- Automatic selection of cluster centers.

The performance metrics were the mean intersection over union (mIoU) and volume overlap error (VOE). The dataset consisted of 22 smoke images, in which the smoke area was marked manually. The results produced by the proposed model were compared to other algorithms, as shown in Figure 11.
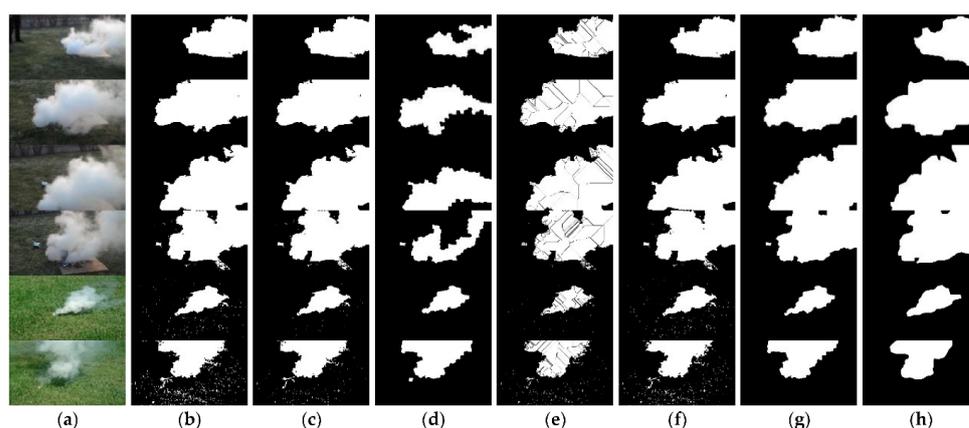

Figure 11. Smoke segmentation results: (**a**) original image; (**b**) Fuzzy C-Means; (**c**) K-means; (**d**) Density Peak Clustering; (**e**) Watershed; (**f**) Mean Iteration Threshold; (**g**) TDSPC; and (**h**) Manually labeled images. Reproduced/adapted under the terms and conditions of the CC BY license from Ref. [67]. ©2022, MDPI.

Further tests were conducted on a public dataset [68] consisting of 400 smoke images and ground truth maps. A comparison between TDSPC, Particle Swarm Optimization and peak-graph-based fast density peak clustering (Guan et al. [69]) in terms of accuracy and F1-Score is presented in Table 7.

Table 7. Comparison of TDSPC with other clustering algorithms. Reprinted/adapted under the terms and conditions of the CC BY license from Ref. [67]. ©2022, MDPI.

Algorithm	Accuracy	F1-Score
PGDPC	45.57%	39.14%
PSO	64.11%	45.81%
TSDPC	64.74%	51.27%

The main difficulty in smoke detection is the complex mixture of the background and itself, with borders not clearly defined; in fact, the term "border" is not even applicable to small smoke, a more appropriate term being "transitions". Smoke images have properties similar to those of liquids (time-varying texture, color, and shape). This makes the task of separating the smoke from the background highly challenging. It is particularly difficult to detect smoke in foggy environments. Image augmentation techniques employed usually in image processing with CNNs (position augmentation and color augmentation) can improve significantly the performances of models based on CNNs. Several studies discussing smoke detection are summarized in Table 8.

Table 8. Smoke detection architectures, datasets and performance metrics.

Ref	Architecture	Dataset	Image Augmentation	Performance Metrics
Kim [70] (2023)	Enhanced bidirectional feature pyramid network (BiFPN) applied to the YOLOv7 head portion integrated with a convolutional block attention module.	Aerial photos of wildfire smoke and forest backgrounds. A total of 3500 forest fire smoke pictures, 3000 non-forest fire smoke pictures, respectively. All images resized to the size of 640 × 640.	Rotation Flipping	AP: 0.786 FPS: 153
Ashiquzzaman [71] (2020)	Deep architecture: six convolutional layers (each followed by one Batch Normalization and one MaxPooling layer), and three fully connected layers (each followed by a Batch Normalization layer). Activation function: ELU.	Foggia et al. [45]: 12 fire videos, 11 smoke videos and 1 normal video. Keyframes manually selected. Classes: Fire; Smoke; Negative.	N/A	<table><tr><td></td><td>P</td><td>R</td><td>F1</td></tr><tr><td>0</td><td>0.84</td><td>0.79</td><td>0.81</td></tr><tr><td>1</td><td>0.91</td><td>0.89</td><td>0.9</td></tr><tr><td>2</td><td>0.87</td><td>0.95</td><td>0.91</td></tr><tr><td colspan="4">Confusion matrix</td></tr></table>
He [56] (2021)	VGG16 (He et al. [72]). A total of 13 convolution layers and 3 full connection layers. Each group of 3 convolution layers followed by MaxPooling. A total of 3 fully connected layers before the output. Activation function SoftMax, 1000 classes.	<table><tr><td>Type</td><td>Train</td><td>Test</td></tr><tr><td>Smoke</td><td>6662</td><td>1680</td></tr><tr><td>Smoke and fog</td><td>6842</td><td>1680</td></tr><tr><td>Non-smoke</td><td>6720</td><td>1681</td></tr><tr><td>Non-smoke and fog</td><td>6720</td><td>1681</td></tr></table>	N/A	Accuracy: 0.999 P: 0.999 R: 0.999 F1-Score: 0.999
Hu [73] (2022)	Architecture built on YOLOv5 by replacing the SPP module with the Soft-SPP and adding the VAM module.	Available online [74]	N/A	AP: 0.79 FPS: 122
Liu [75] (2019)	First channel: AlexNet-based network with a residual block to extract generalizable features. Second channel: a dark-extracted convolution neural network is used to train the dark channel dataset.	<table><tr><td></td><td>Pos Class</td><td>Neg Class</td></tr><tr><td>Train</td><td>3967</td><td>3967</td></tr><tr><td>Val</td><td>430</td><td>430</td></tr><tr><td>Test</td><td>500</td><td>500</td></tr></table>	N/A	Accuracy: 0.986 Binary cross-entropy: 0.077
Yuan [76] (2023)	VGG16 (Simonyan [77]) was used as backbone for obtaining contextual features from the input image. A 3D attention module developed through fusing attention maps along three axes. Multi-scale feature maps obtained through Atrous convolutions and attention mechanisms followed by mixed pooling of average and maximum.	A total of 1000 virtual smoke images but with different background images (Yuan, [78])	N/A	MAE (DUTS dataset [79] was used for performance comparison).

Accurate and fast smoke detection can significantly improve the overall accuracy of fire detection by reducing the number of FNs, which in general is more important than the number of FPs in the case of fire detection with machine learning algorithms.

In some types of fires such as wildfires occurring during daytime, smoke is a critical element for fire detection. During the early stages of fire development and spread, smoke is the only detectable object. It is therefore essential to include smoke detection capabilities in the models designed for wildfires.

Li et al. [80] developed a fire segmentation method based on deep learning with the purpose of assisting the firefighting teams in realistically assessing the fire scale and formulate a response plan accordingly. An encoder–decoder structure was proposed as follows: the encoder consists of a deep CNN and atrous spatial pyramid pooling. The encoder generates feature maps with four different resolutions. The decoder fuses the features and restores the feature size by means of up sampling. The dataset used to assess the performance of the model proposed was the FLAME dataset [31]. The loss in the segmentation accuracy was compensated by introducing two different image features in order to enrich the fire information. Compared to the original deeplabv3+ (Chen et al. [81]) upon which the architecture is based, it was reported that the segmentation speed was considerably higher.

3.4. Machine Learning Algorithms for Fire Detection in Civil and Industrial Facilities

The fire lifecycle in civil and industrial environments, either indoor or outdoor, is different from that of wildfires. The main difference is given by the nature of the flammable material and the availability of the air supply. Wildfires are characterized by a large surface area, where they can occur and develop before being detected; from this point of view, fires in civil and industrial facilities can be detected during their early stages. On the other hand, a wide range of conditions exist for civil and industrial facilities: the flammable material's nature, the air supply, the possibility to cause more devastating effects (explosions, damage to the buildings structure, etc.), and the risk to personnel and assets.

Pincott et al. [82] used the TensorFlow Object Detection API with Faster Region-based CNN (R-CNN) Inception V2 and SSD MobileNet V2 to develop two fire detection models for indoor environments. The R-CNN performed an initial scan of the target image for possible objects and generated a large number of proposed regions. A CNN processed these regions and fed them to an SVM classification layer. The detected and classified region was represented through bounding boxes. A similar architecture was used for the second model. The two models are represented in Figure 12.

The performances of the two models, quantified by means of the confusion matrix, are presented in Figure 13.

The F1-Scores reported for the two models were as follows: R-CNN, 0.85 (Fire class) and 0.91 (Smoke class); SSD: 0.68 (Fire class) and 0.74 (Smoke class). The overall performance was better for the R-CNN model than for the SSD. Both models were trained on the same dataset containing 480 images. The R-CNN model still had a high rate of FNs (10.99% Fire images misclassified as Smoke and 14.29% Fire images misclassified as Normal), which is not suitable for an efficient fire detection system. The main cause for the high rate of misclassification was most likely the low number of images in the training set.

Avazov et al. [83] compiled a two-class dataset consisting of 9200 nighttime and daytime flame images. No details on the scenery and image metadata were mentioned. Image augmentation by rotation (90°, 180° and 270°) was applied to increase the dataset volume. A number of 24,385 fire images were used in the training dataset alongside with 10,000 flame-like images. The test set consisted of 3215 flame images and no flame-like ones. All images were resized to 416×416 and two YOLO versions were considered, YOLOv4 and YOLOv3. The test set accuracy reached the highest value (96.3%) for YOLOv4 for a model size of 245 MB and a training time of 72 h. It can be argued though that the absence of negative samples in the test set can artificially increase the accuracy of the model. Another disputable point is the class imbalance in the training set. The FP number was reported

for all models, with YOLOv4 having the best performance. However, the number of FNs could be more important than FPs for this kind of problem.
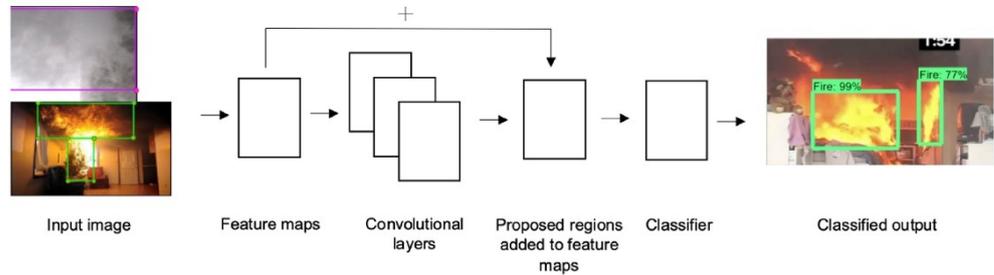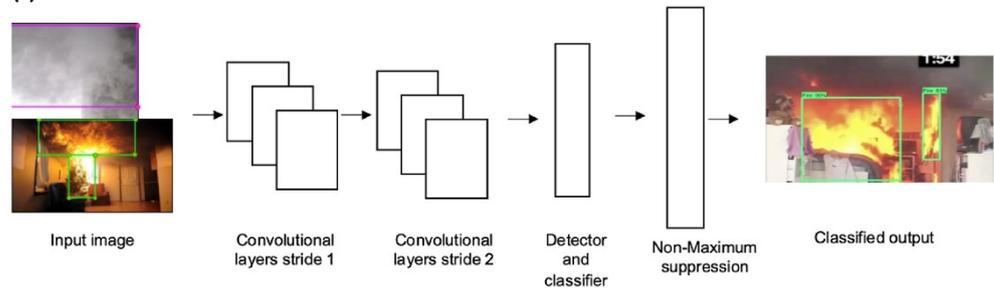

Figure 12. (**a**) The Region-based CNN model based on Inception V2; (**b**) The Single Shot Detector model based on MobileNet V2.. Reprinted/adapted with permission from Ref. [82]. ©2022, Elsevier.

Figure 13. The confusion matrix for the R-CNN with Inception V2 model (**a**) and for the SSD with MobileNet V2 model. (**b**) Reprinted/adapted with permission from Ref. [82]. ©2022, Elsevier.

Kim et al. [84] collected images and videoclips from public sources in order to train a four-stage deep learning fire detection architecture, presented in Figure 14.
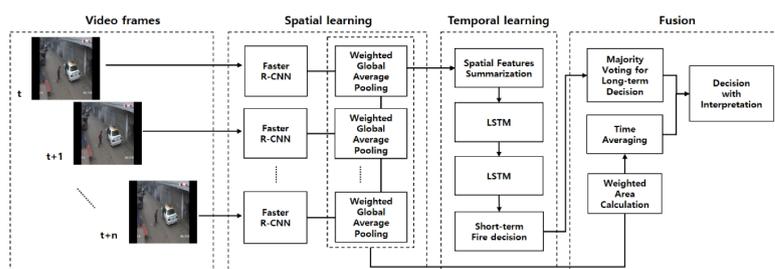

Figure 14. The four-stage deep learning architecture proposed in [85]. Reprinted/adapted under the terms and conditions of the CC BY license from Ref. [84]. ©2019, MDPI.

The sequence of operations performed in the network is presented in Figure 15. Flame/smoke objects were detected for each frame of the video. The CNN/Faster R-CNN features in the detected bounding boxes temporally accumulated for a period, during which the fire decision was taken by the majority voting process.
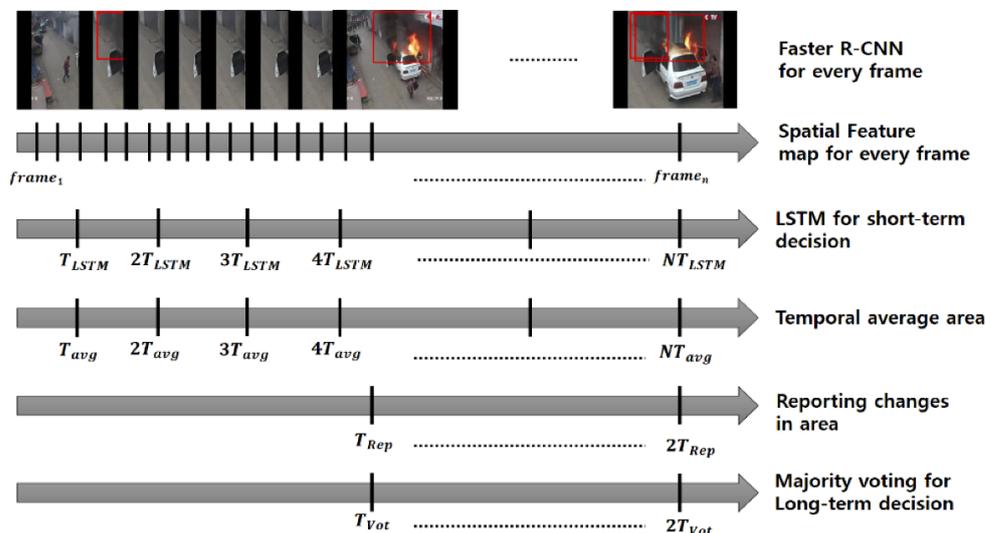

Figure 15. Time sequence diagram of the fire detection process [84]. Reprinted/adapted under the terms and conditions of the CC BY license from Ref. [84]. ©2019, MDPI.

The performance of the architecture quantified through accuracy, FP rate and FN rate was 95%, 3.04% and 1.73%, respectively. This paper does not report important information on the train/test sets.

Jadon et al. [85] proposed a lightweight architecture (FireNet) in the form of a shallow neural network suitable for low-cost hardware such as Raspberry Pi, with the main objective of achieving a tradeoff between accuracy and a high processing speed (high frame rate). A linear network architecture with three cascaded three-layer pairs (Convolutional–Pooling–Dropout) followed by a flattening layer and three FC layers was used. Low resolution images (64×64) were used as input.

A custom dataset was compiled gathering images from [45], Flick and Google. In the attempt to compensate for the low diversity of the Foggia [45] dataset, the authors added fire and non-fire images containing fire-like objects. In the end, a highly diverse training dataset with 1124 positives (fire images) and 1301 negatives (non-fire images) was compiled. In the same way, the test dataset consisted of 46 fire videos (19,094 frames) with 16 non-fire videos (6747 frames) and 160 negatives containing fire-like elements.

The complete detection system consists of a camera connected to a Raspberry Pi microcontroller and two cloud services (Amazon S3 for uploading fire images and Twilio—a cloud-based message service). Several metrics were reported and compared to the metrics for the Foggia [45] dataset. All metric values were slightly higher for the Foggia [45] dataset, which was explained by the limited image diversity in this dataset. Some degree of overfitting can be noticed in the training curves (training and validation accuracy recorded for 100 epochs).. The accuracy curve for the test set would have been more relevant and it would have probably shown a more significant degree of overfitting.

Shees et al. [86] proposed an improved FireNet version (FireNet-v2) keeping the same architecture and reducing the number of filters in the first three convolutional layers, which resulted in a significantly lower number of trainable parameters. The confusion matrix for the architecture FireNet-v2 is presented in Table 9.

Table 9. The confusion matrix for the Foggia [45] and FireNet [85] datasets.. Reprinted/adapted with permission from Ref. [86]. ©2023, Elsevier.

		Foggia Dataset [45]				**FireNet Dataset [85]**	
		Predicted Class				**Predicted Class**	
		Fire	No Fire			Fire	No Fire
True class	Fire	314	49	True class	Fire	553	40
	No Fire	4	3017		No Fire	4	274

From the values reported in Table 9, it can be noticed that the Foggia dataset was heavily unbalanced, which considerably reduced the relevance of the accuracy value. The architecture FireNet-v2 resulted in an accuracy of 98.43% (Foggia [45] dataset) and 94.95% (FireNet [86] dataset). However, it was not mentioned to which dataset, training or test, these values refer.

Saponara et al. [87] developed a deep learning architecture consisting of an input layer (for the input size $64 \times 64 \times 3$), four convolution layers with ReLU activation function, two MaxPooling2D, two AvgPooling2D layers, a Dropout layer (0.6), and a final FC layer with SoftMax activation function. The kernel size for the first convolution layer was 5×5 and 3×3 for the rest. A balanced dataset containing 200 images was compiled and uploaded to the Ground Truth Labeler app in Matlab. Although the SoftMax activation function was used, the architecture performed a binary classification.

The performance of the architecture was evaluated using the dataset from [85] (Jadon et al.). The accuracy, R, P and F1-Score were 93.6%, 100%, 92.4% and 96.04%, respectively. The most important and interesting finding of this study was the fact that the architecture did not report any FNs. In the context of the low volume of the dataset and no information on the class imbalance, this result is rather ambiguous.

Special civil structures exist in which standard computer vision techniques are less efficient in accurately detecting fires compared to outdoor environments. The lighting condition is the main factor that determines the difference as well as the high computational cost for vision-based algorithms. Such civil infrastructure elements are tunnels, in which fire is the disaster with the highest degree of criticality, considering the effects and the probability of occurrence. Moreover, in such environments, the firefighting action and equipment are difficult to deploy and the environment is highly changing and unpredictable. In the case of tunnels, false alarms can be more disruptive and incur higher costs than in the case of other environments. It is therefore important to develop systems with a lower rate of FPs. Sun and Xu [88] developed a standard multi-neural network architecture based on temperature data from tunnel sensors. A dataset consisting of temperature time series recorded at different locations along the tunnel was used. A new dimension was added to the test set consisting of the accumulation values (the sum of all temperature values in the time series for each location). The dataset was labeled accordingly (Fire/No-Fire) and then it was divided randomly into several subsets. A number of neural networks equal to the number of subsets were created and each one was trained using the previously generated subsets. An experimental set up presented in Figure 16 was built to test the performance of the algorithm. The set up consisted of a tunnel experimental platform with the dimensions 90 m $\times$ 3 m $\times$ 3 m and a 20 m section containing 8-layer PVC-sheath cables as a fire source. A two-meter, 2 kW electric heating wire was placed in the middle of the first layer to cause the ignition of the cables.
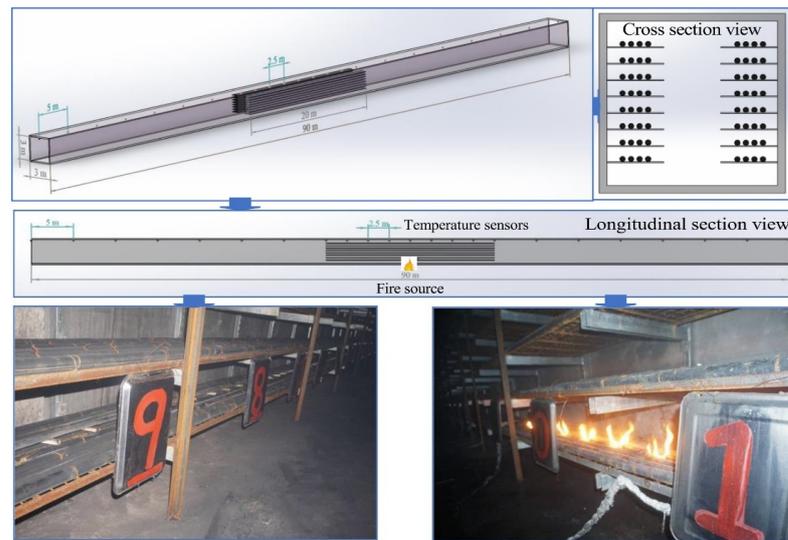
Figure 16. The experimental set up and the sensor distribution. Reprinted/adapted with permission from Ref. [88]. ©2022, Elsevier.

A three-stage experiment was conducted based on the fire observation and experimental operation: (i) the initial stage (from 0 to 360 s), (ii) the fire developing stage (from 361 s to 3750 s) and (iii) the extinguished stage (from 3751 s to 4405 s). It was reported that the algorithm was capable of issuing early warnings (in less than 3 s) in tunnel fires with a high accuracy (96.8%) using only one temperature sensor for 60 m of tunnel length.

Another reason for which systems based on computer vision are not efficient in tunnels is the fact that smoke generated by fires cannot be dissipated and hinders the visibility of the cameras. For this reason, other sensory systems, such as temperature, can increase the system reliability. In such cases though, not only is the momentary value of the temperature important but also its rate of change. Machine learning systems with memory features such as RNN can be employed to address this feature. Wu et al. [89] developed a fire and smoke detection system based on temperature sensors placed along the tunnel. The architecture was based on LSTMs and is presented in Figure 17.
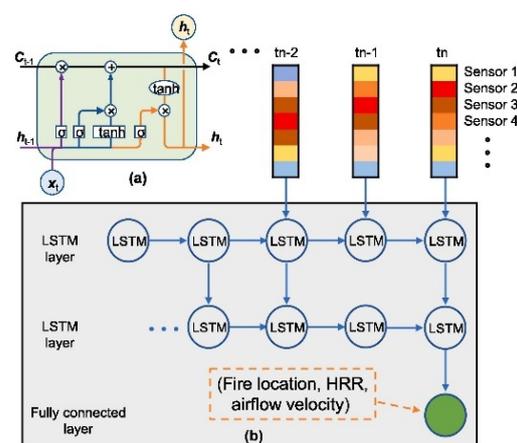
Figure 17. (**a**) Typical LSTM cell; (**b**) the LSTM architecture proposed in [89]. Reprinted/adapted with permission from Ref. [89]. ©2022, Elsevier.

It is a well-known fact that LSTM architecture performance depends significantly on the activation function. It is worthy to mention that this study correctly considered several activation functions and assessed the performance of the architecture. A training dataset was compiled by means of performing tunnel fire simulations using the Fire Dynamics

Simulator v 6.7. A number of 400 fire scenarios (16 fire locations, five HRR values and five ventilation conditions) were considered. The results are presented in Figure 18.

Figure 18. Predictions of (**a**) fire location, (**b**) fire size (dissipated power) and (**c**) ventilation. Reprinted/adapted with permission from Ref. [89]. ©2022 Elsevier.

From Figure 18, it can be noted that the algorithm can determine the location of a fire with high accuracy. However, the accuracy in estimating the power dissipated by the fire is significantly lower.
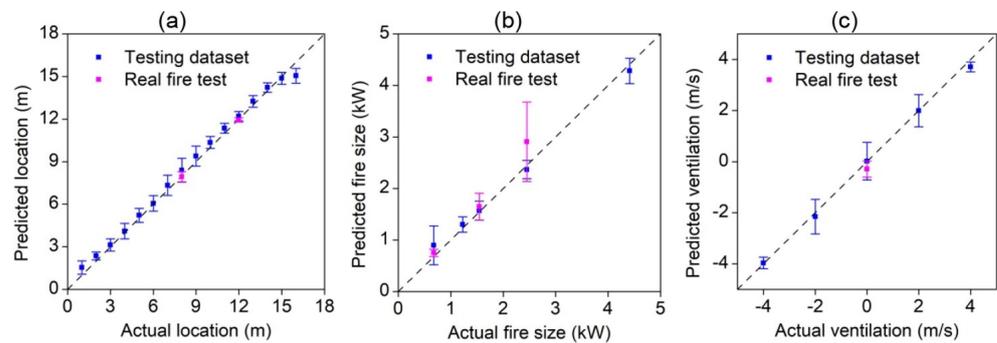
The information on the power dissipated by the fire and its rate of change could be a critical parameter in its early stages, allowing a rough estimation of spread potential and the balanced allocation of fire extinguishing resources. Wang et al. [90] developed an experimental set up based on a stereo vision camera system HBV-1780-2 S2.0 with the purpose of locating the flame in the image and estimating the HRR from scaling the fire image. The architecture of the system is presented in Figure 19.
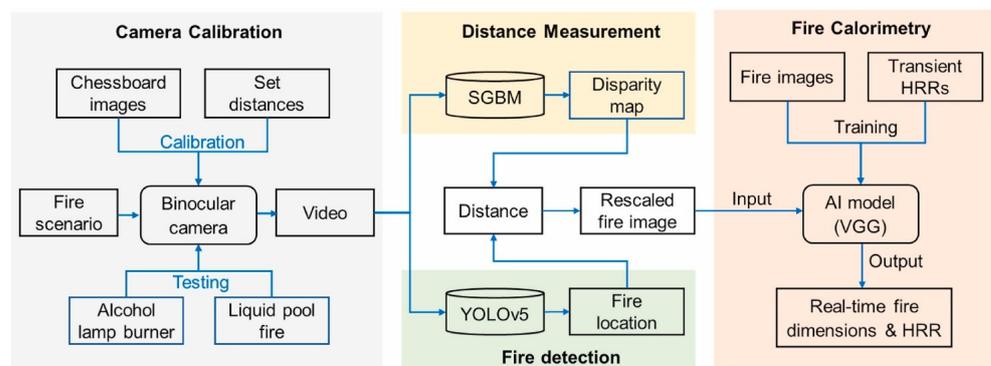
Figure 19. Computer vision-based framework for fire calorimetry and distance measurement. Reprinted/adapted with permission from Ref. [90]. ©2023, Elsevier.

The main element in the distance estimation was the object detection algorithm YOLOv5 while the HRR was determined by means of the transfer learning architecture VGG. The train dataset for VGG consisted of over 200 fire tests in the NIST Fire Calorimetry Database [91]. The distance measurement errors for four cases (0.5 m, 0.6 m, 0.7 and 0.8 m) were 0.009 m, 0.020 m, 0.017 m and 0.004 m, respectively. A fire scenario was set up using a 30 cm diameter pool filled with 200 mL of propanol. In order to vary the distance from the fire source, the camera system was moved randomly. The fuel-burning mass loss rate was recorded by means of a scale throughout the experiment. The fire images captured by the camera system were processed in real time (resizing to match the input size of YOLO, 244 × 244). The results of the experiment are presented in Figure 20.
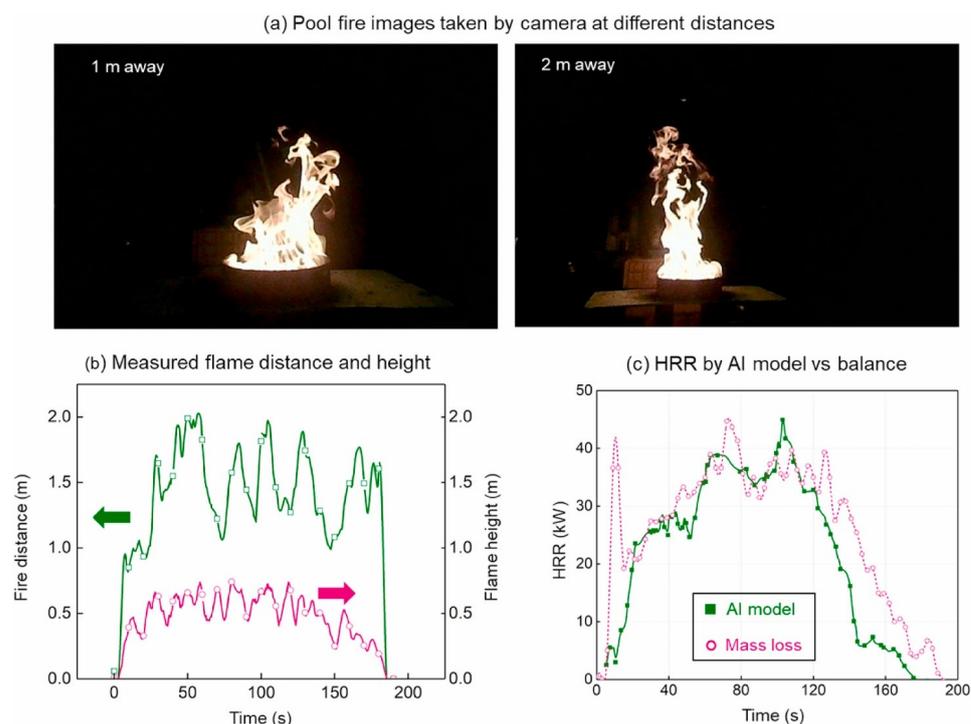
Figure 20. The results of the automatic fire distance and HRR measurement. Reprinted/adapted with permission from Ref. [90]. ©2023, Elsevier.

The fire distance and flame height were estimated correctly (Figure 20b). However, the algorithm fails significantly to correctly estimate the HRR, especially during the early stages of the fire.

Kim and Ruy [92] used three-channel color images and one-channel IR data. An IR image is a 2D matrix with a single channel (gray scale) consisting of temperature values. The advantage of the IR channel is that it increases the discrimination capacity for non-fire situations in images containing flame-like elements. Transfer learning was employed in this study using the Xception API module from Keras [93] (trained with Imagenet [94]). The only modification to the Xception architecture was the replacement of the last FC layer with a Dropout layer and a binary classification FC layer. A custom dataset was generated consisting of images captured by a combined sensor (a camera in the visible wavelength interval and an IR sensor, capturing images in the infrared spectral region synchronously with the camera). The training set consisted of 10,836 images (6342 fire, 4494 fire-resembling objects) and the test set consisted of 14,206 images (7096 fire, 6923 fire-Like). In order to investigate the influence of the channel number on the effectiveness of the fire detection process, the existing RGB dataset was reduced to 1/10 (true: 600, false: 450). Other datasets were created, modifying the original RGB dataset: (i) Red, Red, Blue and Green, Green, Blue, in which red and green are mixed up, and (ii) Zero, Green, Blue and Red, Zero, Blue, in which red and green are not recognized. The performance of the algorithm for the five datasets discussed above is presented in Table 10.

Table 10. The influence of the number of channels on the algorithm performance [92].

Dataset	R [%]	P [%]	F1-Score [%]
Red, Green, Blue	98.9	99.8	99.3
Green, Green, Blue	97.3	80.0	87.8
Red, Red, Blue	90.9	86.5	88.6
Zero, Green, Blue	16.9	91.1	28.5
Red, Zero, Blue	0	0	0

The confusion matrix is presented in Table 11. The number of false negatives suggests that the model could benefit from further improvement, even though the number of false negatives is 0.

Table 11. The confusion matrix for the composite dataset [92].

		Predicted Class	
		Fire	No Fire
True class	Fire	6863	313
	No Fire	0	7030

It was found that the performance of the model decreased significantly when the number of channels was reduced; adding an IR to the RGB channels increased the algorithm performance above that of the RGB dataset. Using the composite dataset (RGB + IR), correct predictions were made for cases where the RGB-only dataset reported FPs (high-luminosity objects), as well as correct predictions for the cases in which using the IR dataset only, FPs were reported (high temperature objects).

While fire detection systems for wildfires rely mostly on computer vision and CNN-based architectures, for indoor fires other models considering parameters such as the temperature and gas concentration are employed, as summarized in Table 12; algorithms with other objectives than fire detection/classification exist.

Table 12. Algorithms, data sources and objectives for different settings.

Ref	Algorithm	Setting	Dataset/Data Source	Objectives
Liu [95] (2021)	PSO	Tunnel	Temperature from sensor arrays	Identify the fire source locations
Fang [96] (2023)	Feed-forward ANN	Building consisting of a large number of rooms	Temperature from sensor arrays	Localization of the fire source
Yar [47] (2023)	YOLOv5 with Stem module in the backbone; larger kernels replaced with smaller kernel sizes in the neck section; a P6 module in the head section.	Building fire (seen from outside) Indoor electric fire Vehicle fire	Fire images: 723 building fire; 118 indoor electric fire; 1116 vehicle fire.	Fire detection and classification
Truong [97] (2012)	Moving region detection by Adaptive Gaussian mixture model; Fuzzy C-means clustering color segmentation; SVM Classification.	Indoor Outdoor	Two-class images dataset: 25,000 fire frames; 15,000 non-fire frames. Test sets: Fire Video [98] VisiFire [99]	Fire detection
Wang [100] (2023)	VGG16	Various indoor fires Vehicle fires	NIST Fire Calorimetry Database [101] consisting of HRR versus time values	Predict the HRR using flame images
Wang [102] (2022)	VGG16	Indoor fires in a ISO 9705 compartment	CFD-generated dataset (FDS 6.7.5)	Predict the HRR
Hu [103] (2023)	Feed-forward ANN	Indoor fire	CFD-generated dataset	Predict the maximum ceiling temperature
Hosseini [104] (2022)	Deep CNN	Outdoor: vegetation, industrial settings Indoor	Eight classes: Flame; White smoke; Black smoke; Flame and white smoke; Flame and black smoke; Black smoke and white smoke; Flame, white smoke and black smoke; Normal.	Fire and smoke detection and classification

Table 12. *Cont.*

Ref	Algorithm	Setting	Dataset/Data Source	Objectives
Chen [105] (2007)	Decision Tree	Aircraft cargo compartment	CO and CO_2 concentration Smoke density	Fire and smoke detection
Qu [106] (2022)	Stacking ensemble learning: LSTM; Decision Tree; Random Forest; Support Vector Machine; K-Nearest Neighbor; XGBoost.	Defined by standard ISO/TR 7240-9-2022 [107]: Cotton rope smoldering fire; Polyurethane plastic open flame; Heptane open flame.	European standard fires: Time history of CO and smoke concentration, and temperature	Fire detection
Kim [108] (2015)	Bayesian classification	Indoor environment with a hallway with two adjacent rooms	Frames extracted from videos	Fire and smoke detection
Favorskaia [109] (2016)	Support Vector Machine Random Forests Boosted Random Forests	Outdoor	Frames extracted from videos	Smoke detection
Smith [110] (2023)	Random Forests	Wildfire	Pixel maps	Predict the wildfire hazard

3.5. Transfer Learning in Fire Detection with Deep Learning Techniques

Transfer learning leverages the features learnt from a type of problem and a large dataset on a new problem. Pre-trained models have proven their superiority over trained-from-scratch models, due, in part, to the large volume of heterogenous datasets used in the development of the model and due to the model complexity as well.

In general, transfer learning problems consist of preserving the backbone of the original network (freezing the model weights), replacing the last FC layer(s) and retraining the network with a specific dataset (a fire dataset in this case). Adapting a pre-trained model to a new dataset requires (optionally) unfreezing one or more layers at the head of the deep network and re-training the model with the new dataset. The output layer size is adapted depending on the number of classes in the new dataset. During the retraining, the weights of the network are kept unchanged and only the parameters of the replaced layers are modified. Alternatively, the whole model can be retrained (unfreezing all layers) with a reduced value of the learning rate, Usmani et al. [111].

Tsalera et al. [112] investigated the performance of several transfer learning models using Forest-Fire (Khan et al. [113]), Fire-Flame [114] and a set of images not used in the training process, consisting of images from 3rd-party sources. The dataset characteristics are presented in Table 13.

Table 13. The datasets used in [112].

Datasets Parameters	Datasets		
	Forest Fire	**Fire-Flame**	**Unknown Images**
Classes	Fire/No Fire	Fire/No Fire	Fire/No Fire
Image acquisition	Terrestrial & Aerial	Terrestrial & Aerial	Terrestrial & Aerial
Fire type	Forest	General	General
Resolution	250×250 pixels	Varies	Varies
View angle	Front & Top	Front	Front & Top
No of images	1900	2000	200
Class balance	Balanced	Balanced	Balanced

The transfer learning models considered in [112] were selected in accordance with the criteria of a high classification accuracy, generalization capabilities on new data and deployment on hardware with limited computational and communication capabilities (Table 14).

Table 14. The CNNs considered in [112] and their performances on the two datasets.

CNN	SqueezeNet		ShuffleNet		MobileNet v2		ResNet50	
Layers	18		50		53		50	
Parameters	1.24×10^6		1.4×10^6		3.5×10^6		25.6×10^6	
CNN performance								
Dataset	Accuracy (%)	Training time (s)	Accuracy (%)	Training time (s)	Accuracy (%)	Training time (s)	Accuracy (%)	Training time (s)
Forest-Fire	97.11	45	97.89	68	98.95	151	97.63	166
Fire-Flame	95.00	77	96.00	90	97.5	164	96.00	175

The selected CNNs were trained with the ImageNet dataset [94] and then re-trained with the specific objective of fire detection. The classification accuracy on the two datasets is presented in Table 13. The deepest network (MobileNet) reached the highest accuracy for both datasets (at the cost of the largest training time). The re-trained CNNs were evaluated on images outside the training domain (the Unknown image dataset), first with the original images and then with noised images (Gaussian noise or Salt & Pepper noise with two values of the PSNR, 10 dB and 15 dB). For images outside the training sets, ShuffleNet

produced the highest classification performance for the cross-dataset with clear images. However, ResNet-50 outperformed all other CNNs for noised images.

Huang et al. [115] considered two classic models as backbone networks, ResNet50 and MobileNet v2, to develop a fire detection technology using a Wavelet-CNN method, applying the 2D Haar transform to extract spectral features of the image, which were further delivered into the CNNs at different layer stages. The two architectures considered in [115] are presented in Figure 21.

The datasets considered in [115] were ImgDS1, containing 1135 fire images from the Corsican Fire Database [116,117], a few fire and non-fire images sampled and augmented from [45], and several random fire and non-fire images from online resources, in which fire was difficult to distinguish due to the stage composition and colors. The second dataset, ImgDS2, consisted of 119 fire images and 107 fire-like images [46]. The third dataset VDS3 consisting of 8 fire videos and 12 non-fire videos from the study of Gong et al. [118]. A number of 2190/2215 fire/no-fire images from ImgDS1 were used for the training process and 633/629 fire/no-fire images from ImgDS1 and ImgDS2 were used for testing. The performance of the models is presented in Figure 22.
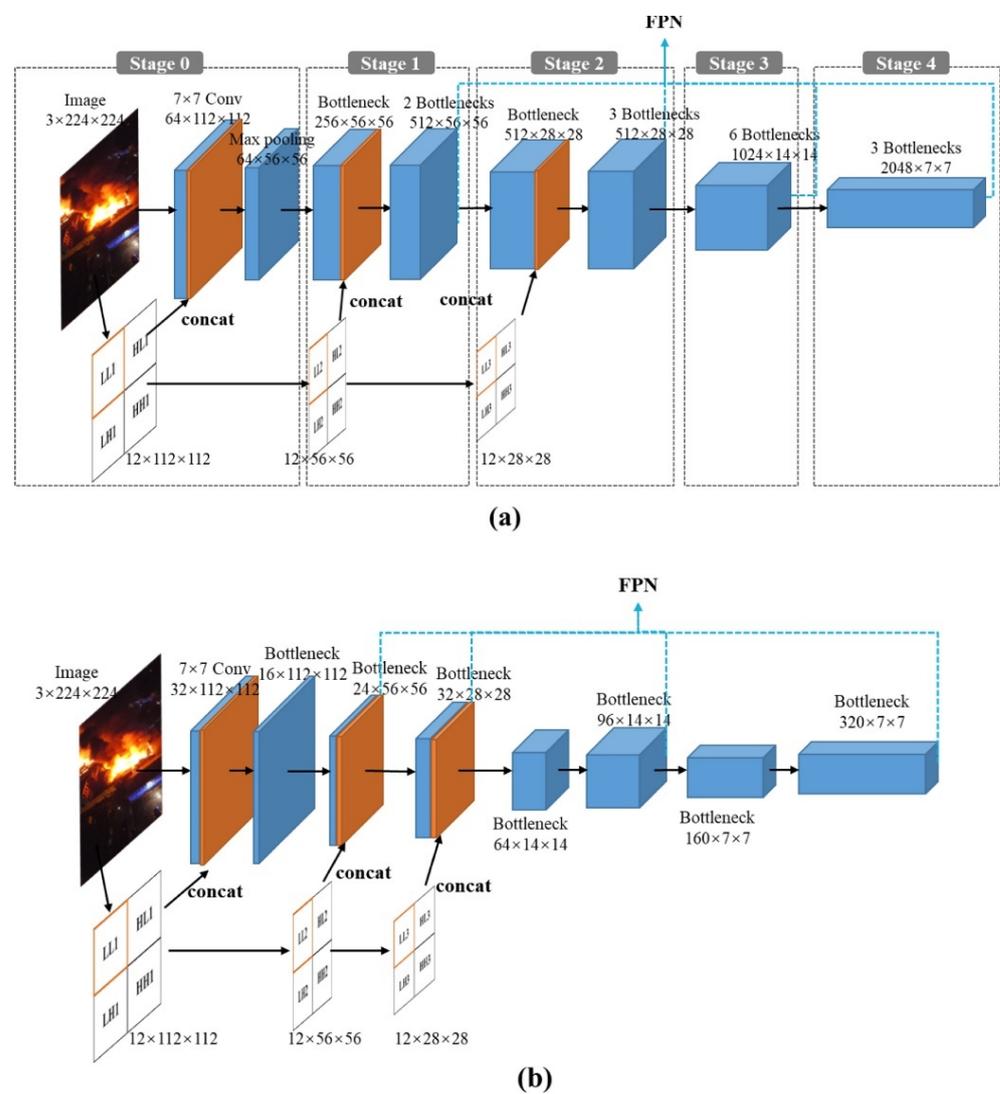

Figure 21. CNNs with Wavelet transform and three-level decomposition of the input image: (**a**) ResNet50; (**b**) MobileNet v2. Reprinted/adapted with permission from Ref. [115]. ©2020, Elsevier.
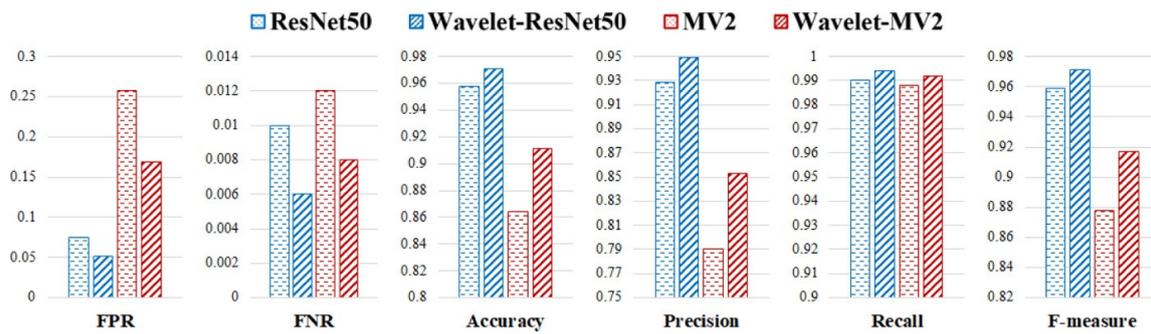
Figure 22. The performance of the architectures investigated in [116]. Reprinted/adapted with permission from Ref. [115]. ©2020, Elsevier.

The performances of different models were quantified through the standard metrics (accuracy, P, R, and F1-Score) as well as the FPR and FNR. The best performance was achieved by Wavelet-ResNet50. However, the performance in terms of processing speed was the highest for Wavelet-MV2 (20.45 FPS) and only 3.03 FPS for Wavelet-ResNet50. This suggests the fact that the wavelet layers, which partially replace the convolution layers, can reduce the processing time.

Reis and Turk [35] investigated the performance of transfer learning models InceptionV3, DenseNet121, ResNet50V2, VGG-19 and NASNet-Mobile on fire prediction performance using the FLAME dataset [31]. Details on the transfer learning architectures considered in [35] are provided in Table 15.

Table 15. Transfer learning architectures considered in [35].

Architecture	No. of Classes, Input Size	Structure	Ref
DenseNet121	Total of 1000 classes, input image size 224 × 224 × 3	Total of 121 convolution layers Four dense blocks Three transition layers One SoftMax layer (output)	[36]
InceptionV3	Total 1000 classes, input shape 229 × 229 × 3 Total of 23,851,784 parameters	154-layer Three modules: (A) 5 × 5 convolution is divided into two 5 × 5 convolutions; (B) 7 × 7 convolution is divided into a 1 × 7 and a 7 × 1 layers; (C) 3 × 3 convolution is divided into two layers, 1 × 3 and a 3 × 1	[43]
ResNet50V2	Total of 1000 classes, input image size	Three network variants, with 50/101/152 layers	[119]
VGG-19	Total of 1000 classes, 143,667, 240 trainable parameters	Total of 16 convolutional, 5 max pooling, 3 FC layers and a SoftMax output layer	[77]
NASNetMobile	Total of 1000 classes, Total of 5,326,716 trainable parameters, input image size 224 × 224 × 3	Scalable CNN architecture consisting of cells using the reinforcement learning method	[120]

The architectures were pre-trained from scratch with the ImageNet dataset (1000 classes) [94]. Then, the models with pre-trained weights and were retrained with the FLAME dataset, which was split into training, validation and test subsets. The process is represented in Figure 23.

(**a**)

(**b**)
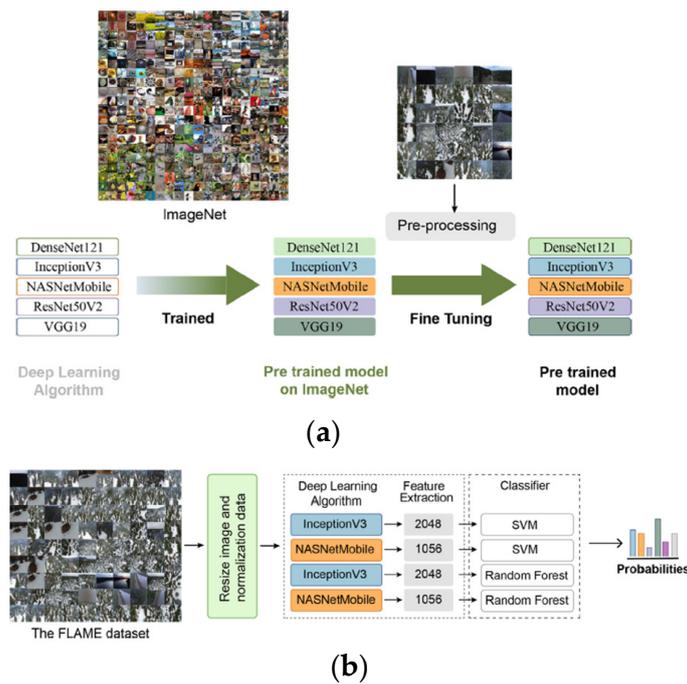
Figure 23. Transfer learning and retraining with the FLAME dataset. (**a**). The transfer learning process (**b**). The retraining and coupling with SVM and RF algorithms. Reprinted/adapted with permission from Ref. [35]. ©2023, Elsevier.

Four typical inference results (two with the ground truth value "No Fire" and two with "Fire") are presented in Figure 24.

Figure 24. Ground truth and inference results for the five transfer learning architectures. Reprinted/adapted with permission from Ref. [35]. ©2023, Elsevier.

From Figure 24, it can be noticed that both FPs and FNs resulted from inference. The only architecture that identifies correctly all situations was VGG19.

The main issue of the transfer learning architectures is the large number of trainable parameters, which results in a large inference time and high hardware requirements. For the smoke/fire classification problems, high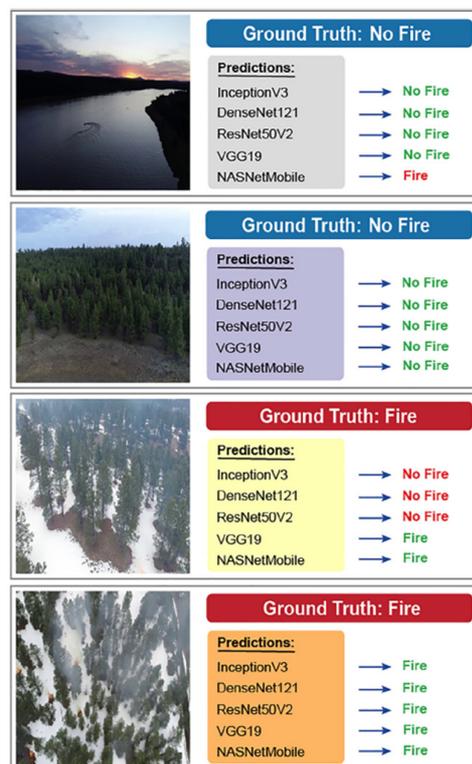 performance architectures, which are usually pre-trained with millions of images and thousands of classes, may not always produce the best performance. Compact architectures such as SqueezeNet [121] can be faster than heavy, high-performance transfer learning models. SqueezeNet is a deep lightweight architecture which uses 1/50 of the AlexaNet parameters but reaches the same accuracy [121]. Peng and Wang [122] developed a series of SqueezeNet architectures with the purpose of detecting smoke from video streams. The basic architecture of SqueezeNet is presented in Figure 25. The macro-architecture (Figure 25a) consists of a series of blocks (Fire blocks, Figure 25b), intercalated with pooling layers, plain convolutional layers and a SoftMax output layer.

A depth-wise separable convolution was used to optimize the SqueezeNet network instead of a standard convolution. Three SqueezeNet architectures were used: (a) the original SqueezeNet architecture (SqueezeNet1); (b) an architecture in which Batch-normalization layers were added after each fire block and convolutional layer (SqueezeNet2); and (c) a depth-wise separation convolution layer was used instead of the standard convolution layer (SqueezeNet3). A dataset containing 50,000 images (smoke/non smoke 0.5/0.5) was compiled from online public resources. The performance of the architectures and a comparison with high-performance, transfer learning architectures are presented in Table 16.
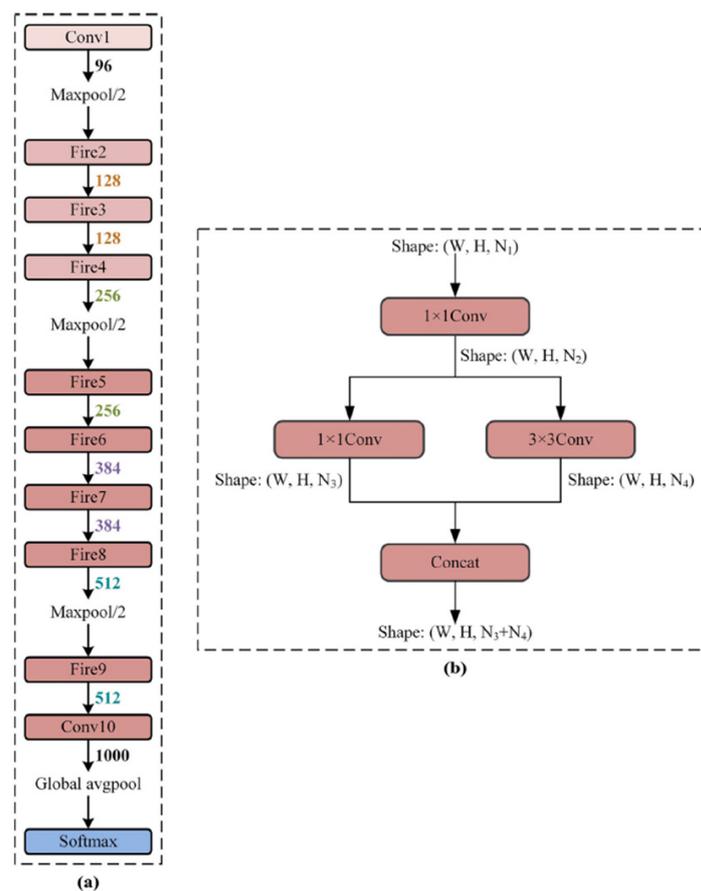

Figure 25. SqueezeNet structure: (**a**) macro-architecture; (**b**) SqueezeNet fire module structure. Reprinted/adapted with permission from Ref. [122]. ©2019, Elsevier.

Table 16. The performance metrics of the SqueezeNet architectures and comparison with high-performance architectures, as reported in [122]. Reprinted/adapted with permission from Ref. [122]. ©2019, Elsevier.

Architecture	P	R	Accuracy	F1-Score
SqueezeNet1	97.92	93.77	95.80	95.71
SqueezeNet2	96.48	96.07	96.27	96.26
SqueezeNet3	97.95	96.31	97.12	97.10
Xception	94.14	96.49	95.30	95.36
MobileNet	94.43	99.53	96.91	96.99
MobileNetV2	99.00	96.58	97.78	97.75
ShuffleNet	94.43	98.44	96.39	96.47
ShuffleNetV2	98.99	97.33	98.15	98.13
AlexNet	94.27	96.06	95.16	95.20
VGG16	96.50	98.55	97.51	97.54

SqueezeNet3, which reached the best classification performance (F1-Score), had an average prediction time of 78.49 ms, while the prediction time of the fastest high-performance transfer learning architecture AlexaNet was 89.19 ms for a model size of 226 MB compared to 1.20 MB (SqueezeNet3).

A summary of transfer learning models, datasets and the performance metrics is presented in Table 17.

Table 17. Transfer learning models based on image datasets.

Ref	Transfer Learning Architecture	Dataset/Classification Type	Dataset Volume/Classes	Image Augmentation	Performance Metrics
Khan [123] (2022)	FFireNet derived from MobileNetV2 by freezing the original weights and replacing the FC output layer.	Forest Fire Dataset [124]/Binary classification	Total of 1900 RGB forest environment images, resolution 250×250. Train/Test: 80/20.	Rotation 1–50° Scaling 0.1–0.2 Shear 0.1–0.2 Translation 0.1–0.2	TP: 189 TN: 185 FP: 5 FN: 1
Zhang [125] (2022)	Backbone: ResNet variants (18, 34, 50 and 101), VGGNet and Inception pretrained on ImageNet dataset. Shallow feature extraction blocks frozen, deep feature extraction blocks fine-tuned.	Selected from FLAME [31]: 31,501, 7874 and 8617 image samples as training, validation and test, respectively.	FLIR camera images: 640×512. Zenmuse camera: 1280×720. Phantom camera: 3480×2160.	Mix-up Zhang [126] (sample expansion algorithm for computer vision, which creates new images by mixing different types of images). Image augmentation (rotate, flip, pan, combine).	F1-Score: (ResNet101): 81.45% FNs: VGGNet: 1148 Inception: 964 ResNet: 925
Ghali [127] (2022)	EfficientNet-B5 and DenseNet-201 fed independently with the unprocessed image. The outputs are concatenated and passed through a Pooling layer.	FLAME [31]/Binary classification	Training set: 20,015 fire/11,500 no-fire images. Validation set: 5003 fire /2875 no-fire images. Test set: 5137 fire/3480 no-fire images.	N/A	Test dataset: TP: 4839 TN: 2496 FP: 984 FN: 298 F1-Score: 84.77% Inference speed: 0.018 s/image
Bahhar [128] (2023)	YOLOv5 Xception MobileNetV2 RestNet-50 DenseNet121	Images captured from FLAME [31] dataset (29 FPS video streams)/Binary classification	25,018 fire class 14,357 no fire class 70% training set 20% validation set 10% test set	Image augmentation (Horizontal flip Rotation)	F1-Score: 0.93
Alexandrov [129] (2019)	Local Binary Patterns [52] Haar YOLOv2 Faster R-CNN Single Shot Multibox Detector (Liu [130])	Image captures from video streams of wildfires recorded from UAVs. Real smoke + Forest background dataset dataset (Favorskaya) [109] Simulative smoke + Forest background dataset (Zhang [131])	6600 Fire 15,600 No-Fire 12,000 images 12,000 images	N/A	FN: 0 (LBP and Haar) FPS: 14.62 {Haar}; 22.4 (LBP)
Xie [132] (2023)	YOLOv5 with a Small Target-Detection Layer	Total of 20,000 flame images collected by a fire brigade/Binary classification.	640×640 RGB images Total of 13,733 images in the test set	Image augmentation (Rotation, Flip) Brightness balance	mAP: 96.6% FPS: 68

Table 17. *Cont.*

Ref	Transfer Learning Architecture	Dataset/Classification Type	Dataset Volume/Classes	Image Augmentation	Performance Metrics
Majid [133] (2022)	VGG-1622 GoogLeNet ResNet50 EfficientNetB0	DeepQuestAI [114] Saied (FIRE Dataset) [134]	Total of 3988 fire images, and 3989 non-fire images. Training/Test: 80/20. 224 × 224 RGB images.	N/A	F1-Score: VGG16: 73.73% GoogLeNet: 87.06% ResNet50: 93.27% EfficientNetB0: 94.00%
Dogan [135] (2022)	Ensemble learning based on ResNet18, ResNet50, ResNet101 and IResNetV2	Compilation from FIRE Dataset [136,137]/Binary classification: Wildfires; Outdoor fires; Building fires; Indoor fires.	PNG files 865 positive class 785 negative class (including fire-resembling objects)	N/A	Specificity Sensitivity P F1-Score Accuracy Confusion matrix

Issues requiring further discussion can be formulated as follows: The positive samples from the Forest Fire Dataset [124] contain (1) both fire and smoke images and (2) fire-only images. Not differentiating between smoke and no-smoke images can be detrimental to the accuracy of the model. In fact, it was reported that some images in which dense fog is represented were misinterpreted as FPs. FPs occur mainly for autumn images, in which leaves' chromatic range is similar to that of fire. A significant FN number resulted, especially in the case of VGGNet. No image analysis was presented in [125] to explain the cause of FNs. The same problem occurred in [127]. The training, validation and test sets considered in [127] were considerably unbalanced. This issue was not approached and there was no discussion on how the class imbalance affects the model performance. Class imbalance issue was spotted for [128] too. There was no discussion on how the training, validation and test set were generated (if class stratification was preserved).

3.6. Image Datasets for Fire Detection Models

Datasets for computer vision are a key element in constructing a highly accurate fire detection model based on deep learning techniques. Most studies do not thoroughly discuss the quality, the content and the class balance of the datasets. Several publicly available fire image/video datasets are presented in Table 18.

Table 18. Image/video streams datasets for CV-based fire detection.

Dataset	File Format	Classes	Environment	Content	Sensor Type/Collection Methodology
FLAME [134]	Raw video Image files Images and masks	Fire No Fire	Wildfires	Total of 39,375 JPEG frames resized to 254 × 254 (training and validation, 1.3 GB). Total of 8617 JPEG frames resized to 254 × 254 (test), 301 MB. A total of 2003 fire frames 3408 × 2160 for fire segmentation problems, 5.3 GB. Total of 2003 ground truth mask frames, 3408 × 2160 for fire segmentation problems, 23.4 MB.	Full HD and 4K camera: Zenmuse X4S camera, Phantom 3 Camera Thermal Camera: FLIR Vue Pro R Drones: DJI Matrice 200, DJI Phantom 3 Professional
BoWFire [138]	Image files (JPG files)	Fire No-Fire	Buildings Industrial Urban	Training dataset: 80/160 images 50 × 50 Fire/No-Fire. Test dataset: 119/107 images of various resolutions Fire/No-Fire.	Sensor type: N/A/ Source: Image scraper (Google) Organized and examined by team members
FIRESENSE [139]	Video files (AVI)	Fire No-Fire Smoke No-Smoke	Buildings Urban Forest	Smoke: 13 positive files, 9 negative files. Fire: 13 positive files, 16 negative files.	Not provided
MIVIA Fire Detection Dataset [140]	Video files (AVI)	Fire No-Fire	Buildings Urban Forest	11 positive video files 16 negative video files Various resolutions from 320 × 240 to 800 × 600.	Not provided
MIVIA Fire Detection Dataset [141]	Video files (AVI)	Smoke Smoke and red sun reflections Mountains Clouds Red reflections Sun	Outdoor, non-urban	Total of 149 files, each consisting of an approximately 15 min recording.	Not provided
KMU [142]	Video files (AVI)	Fire Smoke Normal	Indoor Outdoor Wildfire	Indoor/outdoor flame, 22 files, 216 MB. Indoor/outdoor smoke, 2 files, 17.9 MB. Wildfire smoke, 4 files, 15.6 MB. Smoke- or flame-like moving objects, 10 files, 13.6 MB.	No provided

Table 18. *Cont.*

Dataset	File Format	Classes	Environment	Content	Sensor Type/Collection Methodology
Fire & Smoke [143]	Image files (JPG) Annotations (XML)	Fire No-Fire	Indoor Outdoor Urban	Total of 100 files, 1920 × 1080 or above. Various lighting conditions, varied distance and view point.	Mobile phones Crowdsource contribution
Forest Fire Images [144]	Image files (JPG)	Fire No-Fire	Forest	Train dataset: 2500/2500 Fire/No-Fire images. Test dataset: 25/25 Fire/No-Fire images. Total dataset size: 406.51 MB.	Sensor type: not provided. Combined and merged from other datasets.
Wildfire dataset [145]	Image files (JPG, PNG)	Smoke and Fire Smoke only Fire-like objects Forested areas Smoke-like objects	Forest	Train datasetTest dataset Validation dataset Maximum resolution: 19,699 × 8974 Minimum resolution: 153 × 206 Total dataset size: 2701 files, 10.74 GB	Source: online platforms such as government databases, Flickr, and Unsplash
FlameVision [146]	Image files: JPG for detection tasks, PNG for classification. Total of 4500 annotation files for detection tasks.	Fire No-Fire	Wildfire Outdoor	Classification dataset: 5000 positive, 3600 negative images, split into train, validation and test. Total dataset size: 4.01 GB	Source: Videos of real-life wildfires were collected from the internet.
ForestFireSmoke [147]	Image files (JPG)	Fire Smoke Normal	Forest	Training set: 10,800 files class-balanced from all three classes. Test set: 3500 files class-balanced from all three classes. Total size: 7.12 GB.	Not provided
VisiFire [99]	Video files	Fire Smoke Normal	Outdoor	13 fire videos 21 smoke videos 21 forest smoke videos 2 normal videos	Not provided

Table 18. *Cont.*

Dataset	File Format	Classes	Environment	Content	Sensor Type/Collection Methodology
DSDF [49]	Image files	Smoke (SnF) Non-smoke w fog (nSF) Non-smoke wo fog (nSnF) Smoke and fog (SF)	Outdoor	6528 SnF images 6907 nSnF images 1518 SF images 3460 nSF images Difficult samples were included: local blur and image smear; non-smoke images with smoke-like objects; different color and shapes of the target smoke.	Source: collected from public online resources; photos taken under laboratory conditions.
DFS [148]	Image files	Large flame Medium flame Small flame Other		Total of 9462 images, of which: 3357 Large flame; 4722 Medium flame; 349 Small flame; 1034 Other.	Source: collected from public online resources.

The FLAME dataset [114] is the most widely used, being reported in Shamsoshoaraet al. [31], Reis et al. [35], Tsalera et al. [112], Wang et al. [149], Harkat et al. [150], Bouguettaya et al. [151], Pourbahrami et al. [152], Zhan et al. [33], Sun et al. [153,154]. Other datasets exist that include fire images, such as Aerial Image Database for Emergency Response Applications (AIDER), Saini et al. [155], Kyrkou et al. [156], consisting of images collected using a UAV platform and classified into five classes: "collapsed building", "fire/smoke", "flood", "traffic accidents", and "normal".

Another recent dataset (FASSD) was created by Wang et al. [157]. It is highly heterogenous and consists of typical images and videos in existing open-access flame or smoke datasets, Internet crawlers, computer graphics artwork, Sentinel-2 (10 m resolution) and Landsat-8 (30 m resolution) remote sensing images. It includes fire, smoke, and confusing non-fire/non-smoke images taken at various distances (near and far), different scenes (both indoor and outdoor) and different ambient light intensities (night and day). Images from various sensor types were included (surveillance cameras, unmanned aerial vehicles and multi-source satellites). It consists of 101,087 images, as follows: 59,177 positive samples (labeled), and 41,910 images with more than 60 kinds of fire-like and smoke-like images as negative samples. A number of 82,666 flame objects and 57,742 smoke objects were included. A computer vision dataset consisted of a number of 95,314 images and a remote sensing dataset included 5773 remote sensing images.

3.7. Remote Sensing Based on Image Capture

3.7.1. Optical Cameras

Remote sensing cameras acquiring light in the spectral range 400 nm to 700 nm are considered optical (RGB) cameras. A low cost, high spatial resolution, adaptability to various lighting conditions and size are the main arguments for their use [158]. One of the drawbacks of the optical cameras is their limited field of view, which makes it necessary to use more cameras to cover a larger viewing angle. Recently, cameras with a 360° viewing angle have been developed and employed in forest fire detection applications, Barmpoutis et al. [159]. Novac et al. [160] employed a type of optical camera combining an RGB standard camera with a depth sensor (developed by Microsoft in 2009 to solve the problem of depth information), to identify the forest fire properties, such as size and height. The common issues the RGB cameras face are as follows: the impossibility of detecting smoke during dark, it finds it very hard to detect wildfires in dense forests, and its sensitivity to environmental conditions (shadows, sunlight intensity and angle, and clouds).

3.7.2. Thermal IR Cameras

IR sensors are sensitive to electromagnetic radiation in the spectral ranges MWIR (3000 to 5000 nm) and LWIR (8000 to 14,000 nm). Their main advantage over RGB cameras is the fact they are insensitive to illumination and are able to detect fires behind obstructing objects by means of the thermal radiation emitted. A typical use case for a thermal IR camera is the dataset described and processed in the study of Shamsoshoara [31]. The main issues the thermal IR camera faces are its low spatial resolution and the thermal distance problem, as discussed in the study of Cetin et al. [161].

3.8. Performance Metrics, Class Imbalance and Data Augmentation

Fire detection and classification models report as their performance metrics accuracy, precision, recall and the F1-Score. For fire detection problems, the criticality is to correctly classify the positive samples (minimize the FN rate)—that is, maximize the recall. Minimizing the FP rate is also important, which is a high value of specificity (false fire alarms can have some economic costs) but far less important than FN rate. From these points of view, the most important metric is the recall, followed by sensitivity. The F1-Score can also offer relevant information on the performance of the classifier. In practices of fire detection systems, positive samples occur much less frequently than negative samples. For such

a dataset, very high values of accuracy (Song et al. [37]) can be reached, which makes accuracy an inappropriate metric for assessing a fire detection model's performance (de Almeida Pereira [162]). A more comprehensive picture of a model performance is provided by the confusion matrix.

Most machine learning algorithms for fire detection and classification are of a Blackbox type. Al-Bashiti et al. [163] used two large and complex databases. (i) Short [164]—a spatial description of wildfires occurring in the US from 1992 to 2015, containing 50 parameters: discovery day of wildfire (a numerical value ranging between 1 and 365), year of wildfire (a numerical value ranging between 1992 and 2015), latitude and longitude of the wildfire occurrence, wildfire cause (in thirteen categories), and the state where the wildfire occurred. (ii) A database collected from the burned areas of Montesinho natural park, (Northeast region of Portugal) containing 517 wildfires that occurred from January 2000 to December 2003. Geographic features, temporal variables, and average monthly weather parameters (temperature, relative humidity, wind speed, and rainfall), as well as distinct weather-based indices: the Fine Fuel Moisture Code (which influences ignition and fire spread, Duff Moisture Code, Drought Code, Initial Spread Index, which correlates with fire velocity spread, Buildup Index, and Fire Weather Index, defined in the Canadian system for rating fire danger [165]. The objective of the study was to identify hidden patterns in the two datasets by means of several Blackbox machine learning algorithms: deep-learning, Decision Trees, Extreme Gradient Boosted Trees (ExGBT), Logistic Regression and Genetic Algorithms. Such algorithms do not offer any information, for example, on how misclassification occurred. The explainability method of Shapley Additive exPlanations (SHAP), presented in the study of Lundberg and Lee [166], was applied to ExGBT to analyze its performance. SMOTE was applied to the second database in order to achieve class balance. A feature importance plot was obtained by means of SHAP analysis, revealing what the most important feature for each class was.

Class imbalance in the datasets used for the model training was very scarcely mentioned throughout the works surveyed in this review. Class imbalance is a major issue in the problem of deep learning burn-severity mapping, the other being the scarcity of annotated large-scale datasets, no comprehensive comparison of deep learning algorithms and the uncontrolled influence of input spectral channels. Hu et al. [167] considered a large Landsat-based bi-temporal burn-severity assessment dataset (Landsat-BSA) and restructured it through visual data cleaning based on raw annotated MTBS data [168] (around 1000 large fire events in the United States from 2010 to 2019). The objective was to compare the performance of deep-learning segmentation models and highlight the UNet [169] series on pixel-wise burn severity classification. The class imbalance issue was addressed by converting the loss function to down-weight easy examples. In such a way, the training was focused on hard negative examples. Other loss functions for imbalanced datasets can be mentioned, as follows:

- Cross-entropy loss, commonly used for multi class datasets. Not particularly effective in case of highly imbalanced datasets.
- Dice loss. Used frequently in image segmentation problem. Dice loss consists of minimizing the overlap between prediction and ground truth. Dice loss weighs FPs and FNs equally, which is one of its limitations. For highly unbalanced data such as small burned areas with a large unburned background, the FN detections can be weighted higher than FPs to improve the recall rate.
- Lovász loss (Berman et al. [170]) is used in image segmentation problems (flaw detection in industrial production lines, tumor segmentation, and road line detection). Lovász Softmax loss is appropriate for multiclassification tasks.
- Online Hard Example Mining (OHEM) loss, Shrivastava et al. [171]. OHEM puts more emphasis on misclassified examples, but unlike focal loss, OHEM completely discards easy examples.
- Focal loss, Lin et al. [172]. It was designed to address one-stage object detection situations in which an extreme imbalance (in the order of magnitude 1:1000) exists between

classes. It is defined based on the standard binary cross-entropy loss introducing a modulating factor, which reduces the loss contribution from easy examples and extends the range in which a sample receives low loss.

- For semantic segmentation problems, a list of loss functions were reviewed by Jadon [173]: Tversky loss (a generalization of Dice loss coefficient, adding a weight to FPs and FNs), Focal Tversky loss (focuses on hard examples by down-weighting easy and common samples), Sensitivity Specific Loss, Shape-aware loss, Combo Loss (a weighted sum of Dice loss and a modified cross-entropy), Exponential Logaritmic Loss, Distance map-derived loss penalty term, Hausdorff Distance Loss, Correlation Maximized Structural Similarity Loss, and Log-Cosh Dice Loss.

Other standard techniques in tackling the problem of imbalanced datasets are over-sampling and undersampling, described extensively in textbooks such as that of Fernandez et al. [174]. Data augmentation for time series was reviewed by Iwana and Uchida [175]. Tian et al. [176] developed an AI-based fire detection system designed to assist first aid first responders in the initial emergency response to fire events involving mixed or unknown materials. The dataset consisted of data collected from samples of burning materials using a cone calorimeter and a FTIR gas analyzer. Data were augmented using two algorithms:

- Suboptimal Warping Time Series Generator (SPAWNER), Kamycki et al. [177]. The SPAWNER algorithm creates a new time series by computing the average two random time series with the same class label. Before averaging the two time series, they are aligned using dynamic time warping (a standard method that warps the time dimension of two time series to find an optimal alignment between them, Sakoe and Chiba [178]).
- Discriminative Guided Warping is a dynamic time warping algorithm that aligns two time series using high-level features using a reference time series to guide the warping of the time step of another time series.

For image datasets, data augmentation consists of increasing the diversity of images by adding new samples to the dataset; the new samples are derived from existing images by applying various geometric manipulations (rotation, flip, resizing, shift, and cropping) or modifying the contrast, saturation and luminosity (Biswas et al. [41], Sousa et al. [179]), contrast-constrained histogram equalization, Gauss noise and optical distortion (Sun et al. [180]).

Conventional approaches (e.g., CNNs) relay heavily on a set of standard fire characteristics and are not flexible enough to adapt across a wider range of fire conditions. Advanced algorithms have been proposed to address the limitations of the current machine learning techniques for fire detection. Zhao et al. [181] proposed a hybrid approach that mixes the traditional methods with DL techniques. The algorithm enhances the flame feature detection by extracting the texture and color information from images, using the HSV (Hue, Saturation and Value) model and CLBP (Complete Local Binary Pattern). The algorithm employs YOLOv8, which compared to previous YOLO versions, uses the Distribution Focal Loss in the regression loss to deal with imbalanced class problems. The heterogenous dataset consisted of the FLAME forest fire dataset and a non-fire dataset consisting of normal forest images and sunlight images. The standard performance metrics (accuracy, precision, recall and F1-Score) were used to compare the model performance with 11 other models from the literature. The model performed slightly better than any other considered.

Another issue occurring frequently in fire detection problems is the presence of flashing lights, red objects and a high-brightness background. In urban environments or buildings, artificial lighting devices can interfere with the operation of computer vision-based fire detection systems. Sun et al. [180] used scene prior knowledge and causal inference mechanisms to reduce the influence of disturbance-causing objects. A three-stage procedure was proposed as follows: (i) artificial lighting device images were collected and transfer learning was employed to train a YOLOv3 model; (ii) the artificial lighting regions were masked with a novel segmentation method; and (iii) an Inception v4 transfer learning

model was used to carry out the classification task. The focal cross-entropy loss function, Lin et al. [172], was used to address the class imbalance.

Transformer-based algorithms represent a recent trend in improving the performance of the fire detection models based on machine learning algorithms. Shahid and Hua [182] proposed an approach based on transformers for image recognition, Dosovitskiy et al. [183]. The algorithm proposed in [182] processed a fire image as a collection of patches and captured the dependences among various patches. It was reported that the detection accuracy improved (using two publicly available datasets). Lin et al. [184] developed a small-object forest fire detection model integrating a Swin transformer backbone [185]. Integrating the Swin transformer into the mask R-CNN leveraged its self-attention mechanism to capture the global information and improve local information to obtain a larger sensory field and contextual information. Yang et al. [186] proposed a lightweight fire-detection network combining a CNN and a transformer to model global and local information, with the purpose of reaching a tradeoff between accuracy and detection speed. Wang et al. [187] proposed a decoder-free fully transformer-based detector to achieve early smoke and flame detection, with the purpose of improving the detection performance for fires with varying sizes. In the first stage, data augmentation was performed to enhance the capability of generalizing; a detection-oriented transformer backbone network was treated as a single-layer feature extractor to produce fire-related features, which were then fed into an encoder-only single-layer dense prediction module. Zhang et al. [188] proposed a deep learning model featuring a vision transformer to issue fire warnings. A dataset consisting of 1500 images (1000 images used for training and 500 for test). Two important findings were reported, as follows: (i) after 50 epochs, the accuracy of image recognition reached a flat value of approximately 97.4%, and (ii) the training effect of the vision transformer on large datasets is more significant than that of traditional transfer learning models (DenseNet, VGG). Qian et al. [189] proposed a multi-scale two-component, fully integrated model, with one component for large-target forest fire detection and one for small targets, with the objective of enhancing the system capability of identifying both large-scale and small-target fires. In order to reduce the model sensitivity to noise, occlusion and scale variation, a new edge loss function was proposed.

4. Conclusions and Recommendations for Further Investigation and Limitations

A diverse and significant volume of research has been identified on the topic of fire detection models based on machine learning algorithms. From a machine learning perspective, fire detection is a classification problem (binary classification or multi class classification in some cases). The most widely used algorithms are based on computer vision (streaming video or imaging) and further processing by CNNs. Although most studies describing computer vision and CNN-based algorithms report all standard performance metrics, very few discuss the importance of every metric for this specific type of problem. Studies discussing the problem of minimizing the FNs and FPs, respectively, (in this order of importance) are scarce. Even fewer studies report ROC or P-R curves (for example, [162]). Most studies surveyed in this review report accuracy as the main performance metric, which has limited relevance for this type of problem, especially in the context of significant class imbalance, when a high accuracy value can be misleading. Further research is required in the direction of algorithm optimization, with the purpose of minimizing the FP and FN rates (in this particular order). In general, transfer learning models have higher performance metrics than most models developed from scratch. In such cases, the learning rate must be reduced so that the weights of the first layers do not undergo significant changes and only the last newly added FC layer(s) weights are updated, in such way that the model adapts to the new problem. Further research and experimentation are required in the issue of the fine tuning and learning rate optimization.

Some approaches to improve the algorithm performance include the detection of smoke in the images [119], which can enhance the discrimination performance of the model, improving ultimately the model performance.

Given the importance of the dataset for the performance of the model, a survey of fire-related datasets was included in this review, presenting the most important information on twelve fire datasets used in fire detection problems. Some common issues shared by most datasets analyzed in this review can be summarized as follows:

- Very few references report critical dataset properties, such as class imbalance and content diversity and difficulty (fire resembling objects that could confuse the algorithm);
- The volume of the datasets is insufficient in some cases, such as the Fire & Smoke dataset [143];
- It is not clear whether mixed datasets that contain various types of fires (forest, urban) such as MIVIA [140] and Flame Vision [146] offer any advantage over specialized datasets (i.e., forest fire only, building fire only, etc.).

The inference time is a performance metric of secondary importance. It is expected that a model with low FP and FN rates and a high inference time is more desirable than a fast model with higher FP and FN rates. Another important model feature is the size, which was discussed in several studies (e.g., [85]), and the model suitability to be deployed on edge devices was assessed.

Fire detection systems in civil and industrial facilities have some features that differentiate them from systems designed to detect wildfires. Systems based on computer vision may not be as effective in such environments due to the confined spaces (e.g., tunnels) where smoke accumulation considerably obstructs the visibility. In fact, it can be noticed from the low performance of the model developed in Pincott et al. [83], based on image processing only (high FP and FN rates) that computer vision and CNN is not a suitable choice. In such cases, other parameters such as temperature and fire byproducts' concentrations of gaseous components (CO, CO_2) can improve its effectiveness.

As a general, covering remark, it can be stated that despite a large number of both experimental and theoretical studies existing, employing various model architectures (from vanilla CNN to transfer learning architectures) and datasets, the reliability of the fire detection systems based on machine learning algorithms is the main concern that limits the development of such systems.

Some specific conclusions and recommendations for further research, derived from the general conclusion, can be summarized as follows:

- The implications, and ultimately, the cost of FPs and FNs must be thoroughly assessed. In this respect, a practical, result-oriented workflow, could be the following:
  - (i) Defining a misclassification cost function that quantifies the FPs and FNs implications; this depends on the fire type (e.g., forest fire, civil building fire, ship fire, etc.) and requires a thorough assessment of the FPs and FNs consequences; in the context of the reliability concept mentioned earlier, the lower the misclassification cost function value, the higher the reliability of the model.
  - (ii) Adjusting the model hyperparameters in such way that the model performance, defined in terms of relevant metrics (recall, sensitivity, confusion matrix, ROC-AUC and P-R curve), would minimize the misclassification cost function. Ideally, the misclassification cost function should be used for model training instead of standard cost functions for classification (categorical cross-entropy, binary cross-entropy, etc.).
- Further investigation should be carried out in order to understand the mechanisms that lead to misclassified samples (which are scarcely discussed, e.g., [42]). This could be particularly useful in the case of CNN-based algorithms, where visualizing the processed image after each layer could explain how features are extracted explain how/why misclassification occurred.
- Very few studies discuss the class stratification when training/validation/test sets were prepared. Even fewer discuss cross validation, (e.g., [45]). In a more general context of classification problems, it is important to mention that issues such as overfitting occur frequently and sometimes are not thoroughly investigated. The

general approach consists of splitting randomly a dataset into training and test subsets (and sometimes validation subset) with or without cross-validation. However, it is known that most datasets consist of samples with some degree of similarity. A fairly well-designed and trained model can produce results that could falsely induce the conclusion that overfitting is low or inexistent. Using for test samples from other datasets (that could be considered outliers for the train/test datasets) can sometimes reveal the true degree of overfitting.

- Although several public datasets exist, more effort is required to build new, high-quality datasets, or to improve the existing ones. Amongst the dataset features that are important for the fire classification problem, several were spotted that deserve special attention: dataset volume, class imbalance, positive samples difficult to classify or negative samples resembling positives (discussed in very few studies, e.g., [49]), and content diversity.

- Discussing the implications of class imbalance and addressing this issue has been scarcely spotted throughout the literature collection considered in this review. This is especially important, since class imbalance issues were identified for quite a few datasets examined in this review.

- For computer vision-based techniques, an interesting approach could be adding a new spectral channel to the RGB channel (Kim and Ruy [92]) or another feature to the dataset. A recent trend is the so-called sensor fusion philosophy, which consists of collecting and processing information from two or more different types of sensors, such as an RGB camera and a thermal IR camera. Not only has optical/IR detection been reported but also other sensor combinations. Benzekri et al. [190] used a network of wireless sensors to record parameters such as temperature and carbon monoxide concentration as input data for a wildfire detection DL algorithm.

- Although the literature search process resulted in a large number of relatively recent studies, both theoretical (based on datasets only) and experimental (involving some kind of practical approach), no report on any operational, production scale system was identified.

  Some limitations of this review and debatable points can be summarized as follows:

- The main source of information for this review was the SCOPUS database. The choice of SCOPUS was justified by its comprehensiveness and by the fact that it already includes other databases (such as IEEE Xplore, arXiv, ChemRxiv). However, the search could be extended over other important information sources, such as SCIE.

- Some topics covered in this review (such as transfer learning in fire detection problems) are over represented in the literature set considered in this review. Transfer learning in fire detection problems could be a topic for a standalone review.

Nomenclature

ANN	Artificial Neural Network
AP	Average Precision
AUC	Area under Curve
CFD	Computational Fluid Dynamics
CNN	Convolutional Neural Network
DL	Deep Learning
FC	Fully Connected
FN	False Negative
FNR	False Negative Rate
FP	False Positive
FPR	False Positive Rate
FPS	Frames per Second
FT	Fine Tuning
HRR	Heat Release Rate
LSTM	Long Short-Term Memory
MTBS	Monitoring Trends in Burn Severity
NIR	Near Infrared
P	Precision
PSNR	Peak Signal-to-Noise Ratio
NDVI	Normalized Difference Vegetation Index
R	Recall
RF	Random Forest
ROC	Receiver Operator Characteristic
SHAP	SHapely Additive exPlanations
SMOTE	Synthetic Minority Oversampling Technique
SVM	Support Vector Machine
SVR	Support Vector Regression
TSDPC	Truncation distance Self-adaptive Density Peak Clustering

References

1. Saha, S.; Bera, B.; Shit, P.K.; Bhattacharjee, S.; Sengupta, D.; Sengupta, N.; Adhikary, P.P. Recurrent forest fires, emission of atmospheric pollutants (GHGs) and degradation of tropical dry deciduous forest ecosystem services. *Total Environ. Res. Themes* **2023**, *7*, 100057. [CrossRef]
2. Kala, C.P. Environmental and socioeconomic impacts of forest fires: A call for multilateral cooperation and management interventions. *Nat. Hazards Res.* **2023**, *3*, 286–294. [CrossRef]
3. Elogne, A.G.; Piponiot, C.; Zo-Bi, I.C.; Amani, B.H.; Van der Meersch, V.; Hérault, B. Life after fire—Long-term responses of 20 timber species in semi-deciduous forests of West Africa. *For. Ecol. Manag.* **2023**, *538*, 120977. [CrossRef]
4. Lowesmith, B.; Hankinson, G.; Acton, M.; Chamberlain, G. An Overview of the Nature of Hydrocarbon Jet Fire Hazards in the Oil and Gas Industry and a Simplified Approach to Assessing the Hazards. *Process Saf. Environ. Prot.* **2007**, *85*, 207–220. [CrossRef]
5. Aydin, N.; Seker, S.; Şen, C. A new risk assessment framework for safety in oil and gas industry: Application of FMEA and BWM based picture fuzzy MABAC. *J. Pet. Sci. Eng.* **2022**, *219*, 111059. [CrossRef]
6. Solukloei, H.R.J.; Nematifard, S.; Hesami, A.; Mohammadi, H.; Kamalinia, M. A fuzzy-HAZOP/ant colony system methodology to identify combined fire, explosion, and toxic release risk in the process industries. *Expert Syst. Appl.* **2022**, *192*, 116418. [CrossRef]
7. Park, H.; Nam, K.; Lee, J. Lessons from aluminum and magnesium scraps fires and explosions: Case studies of metal recycling industry. *J. Loss Prev. Process Ind.* **2022**, *80*, 104872. [CrossRef]
8. Zhou, J.; Reniers, G. Dynamic analysis of fire induced domino effects to optimize emergency response policies in the chemical and process industry. *J. Loss Prev. Process Ind.* **2022**, *79*, 104835. [CrossRef]
9. Østrem, L.; Sommer, M. Inherent fire safety engineering in complex road tunnels—Learning between industries in safety management. *Saf. Sci.* **2021**, *134*, 105062. [CrossRef]
10. Ibrahim, M.A.; Lönnermark, A.; Hogland, W. Safety at waste and recycling industry: Detection and mitigation of waste fire accidents. *Waste Manag.* **2022**, *141*, 271–281. [CrossRef]
11. Wang, S.; Zhang, Y.; Hsieh, T.-H.; Liu, W.; Yin, F.; Liu, B. Fire situation detection method for unmanned fire-fighting vessel based on coordinate attention structure-based deep learning network. *Ocean Eng.* **2022**, *266*, 113208. [CrossRef]
12. Davies, H.F.; Visintin, C.; Murphy, B.P.; Ritchie, E.G.; Banks, S.C.; Davies, I.D.; Bowman, D.M. Pyrodiversity trade-offs: A simulation study of the effects of fire size and dispersal ability on native mammal populations in northern Australian savannas. *Biol. Conserv.* **2023**, *282*, 110077. [CrossRef]
13. Lindenmayer, D.; MacGregor, C.; Evans, M.J. Multi-decadal habitat and fire effects on a threatened bird species. *Biol. Conserv.* **2023**, *283*, 110124. [CrossRef]

14. Liu, W.; Zhang, Z.; Li, J.; Wen, Y.; Liu, F.; Zhang, W.; Liu, H.; Ren, C.; Han, X. Effects of fire on the soil microbial metabolic quotient: A global meta-analysis. *CATENA* **2023**, *224*, 106957. [CrossRef]

15. Batista, E.K.L.; Figueira, J.E.C.; Solar, R.R.C.; de Azevedo, C.S.; Beirão, M.V.; Berlinck, C.N.; Brandão, R.A.; de Castro, F.S.; Costa, H.C.; Costa, L.M.; et al. In Case of Fire, Escape or Die: A Trait-Based Approach for Identifying Animal Species Threatened by Fire. *Fire* **2023**, *6*, 242. [CrossRef]

16. Courbat, J.; Pascu, M.; Gutmacher, D.; Briand, D.; Wöllenstein, J.; Hoefer, U.; Severin, K.; de Rooij, N. A colorimetric CO sensor for fire detection. *Procedia Eng.* **2011**, *25*, 1329–1332. [CrossRef]

17. Derbel, F. Performance improvement of fire detectors by means of gas sensors and neural networks. *Fire Saf. J.* **2004**, *39*, 383–398. [CrossRef]

18. ASTM. *ASTM Standard Terminology of Fire Standards*; ASTM: West Conshohocken, PA, USA, 2004.

19. Fonollosa, J.; Solórzano, A.; Marco, S. Chemical Sensor Systems and Associated Algorithms for Fire Detection: A Review. *Sensors* **2018**, *18*, 553. [CrossRef]

20. National Research Council. *Fire and Smoke: Understanding the Hazards*; National Academies Press: Washington, DC, USA, 1986; ISBN 0309568609.

21. Chagger, R.; Smith, D. *The Causes of False Fire Alarms in Buildings*; Briefing Paper; BRE Global Ltd.: Watford, UK, 2014.

22. Ishii, H.; Ono, T.; Yamauchi, Y.; Ohtani, S. An algorithm for improving the reliability of detection with processing of multiple sensors' signal. *Fire Saf. J.* **1991**, *17*, 469–484. [CrossRef]

23. Han, W.; Zhang, X.; Wang, Y.; Wang, L.; Huang, X.; Li, J.; Wang, S.; Chen, W.; Li, X.; Feng, R.; et al. A survey of machine learning and deep learning in remote sensing of geological environment: Challenges, advances, and opportunities. *ISPRS J. Photogramm. Remote. Sens.* **2023**, *202*, 87–113. [CrossRef]

24. Dietterich, T.G. Ensemble Methods in Machine Learning. In *Multiple Classifier Systems. MCS 2000*; Lecture Notes in Computer Science; Springer: Berlin/Heidelberg, Germany, 2000; Volume 1857. [CrossRef]

25. Amiro, B.D.; Chen, J.M.; Liu, J. Net primary productivity following forest fire for Canadian ecoregions. *Can. J. For. Res.* **2000**, *30*, 939–947. [CrossRef]

26. Venkatesh, K.; Preethi, K.; Ramesh, H. Evaluating the effects of forest fire on water balance using fire susceptibility maps. *Ecol. Indic.* **2020**, *110*, 105856. [CrossRef]

27. Alharbi, B.H.; Pasha, M.J.; Al-Shamsi, M.A.S. Firefighter exposures to organic and inorganic gas emissions in emergency residential and industrial fires. *Sci. Total Environ.* **2021**, *770*, 145332. [CrossRef] [PubMed]

28. Iliadis, L.S.; Papastavrou, A.K.; Lefakis, P.D. A computer-system that classifies the prefectures of Greece in forest fire risk zones using fuzzy sets. *For. Policy Econ.* **2002**, *4*, 43–54. [CrossRef]

29. Li, J.; Zhou, G.; Chen, A.; Wang, Y.; Jiang, J.; Hu, Y.; Lu, C. Adaptive linear feature-reuse network for rapid forest fire smoke detection model. *Ecol. Inform.* **2022**, *68*, 101584. [CrossRef]

30. Ma, H.; Liu, Y.; Ren, Y.; Yu, J. Detection of Collapsed Buildings in Post-Earthquake Remote Sensing Images Based on the Improved YOLOv3. *Remote Sens.* **2020**, *12*, 44. [CrossRef]

31. Shamsoshoara, A.; Afghah, F.; Razi, A.; Zheng, L.; Fulé, P.Z.; Blasch, E. Aerial imagery pile burn detection using deep learning: The FLAME dataset. *Comput. Netw.* **2021**, *193*, 108001. [CrossRef]

32. Chollet, F. Xception: Deep learning with depthwise separable convolutions. In Proceedings of the 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Honolulu, HI, USA, 21–26 July 2017; Available online: https://arxiv.org/abs/1610.02357 (accessed on 1 August 2023).

33. Zhan, J.; Hu, Y.; Zhou, G.; Wang, Y.; Cai, W.; Li, L. A high-precision forest fire smoke detection approach based on ARGNet. *Comput. Electron. Agric.* **2022**, *196*, 106874. [CrossRef]

34. Zhang, Q.; Xu, J.; Xu, L.; Guo, H. Deep Convolutional Neural Networks for Forest Fire Detection. In Proceedings of the International Forum on Management, Education and Information Technology Application (IFMEITA 2016), Guangzhou, China, 30–31 January 2016. [CrossRef]

35. Reis, H.C.; Turk, V. Detection of forest fire using deep convolutional neural networks with transfer learning approach. *Appl. Soft Comput.* **2023**, *143*, 110362. [CrossRef]

36. Huang, G.; Liu, Z.; Van Der Maaten, L.; Weinberger, K.Q. Densely connected convolutional networks. In Proceedings of the 2017 IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 4700–4708. [CrossRef]

37. Song, X.; Gao, S.; Liu, X.; Chen, C. An outdoor fire recognition algorithm for small unbalanced samples. *Alex. Eng. J.* **2021**, *60*, 2801–2809. [CrossRef]

38. Fernandes, A.M.; Utkin, A.B.; Lavrov, A.V.; Vilar, R.M. Development of neural network committee machines for automatic forest fire detection using lidar. *Pattern Recognit.* **2004**, *37*, 2039–2047. [CrossRef]

39. Ahn, Y.; Choi, H.; Kim, B.S. Development of early fire detection model for buildings using computer vision-based CCTV. *J. Build. Eng.* **2023**, *65*, 105647. [CrossRef]

40. AIHuB. 2022. Available online: https://aihub.or.kr/ (accessed on 29 July 2023).

41. Biswas, A.; Ghosh, S.K.; Ghosh, A. Early Fire Detection and Alert System using Modified Inception-v3 under Deep Learning Framework. *Procedia Comput. Sci.* **2023**, *218*, 2243–2252. [CrossRef]

42. Szegedy, C.; Ioffe, S.; Vanhouche, V.; Alemi, A.A. Inception-v4, Inception-ResNet and the Impact of Residual Connections on Learning. *arXiv* **2016**, arXiv:1602.07261. [CrossRef]

43. Szegedy, C.; Vanhoucke, V.; Ioffe, S.; Shlens, J.; Wojna, Z. Rethinking the Inception Architecture for Computer Vision. *arXiv* **2015**, arXiv:1512.00567.

44. 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]

45. 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]

46. 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 28th 2015 Conference on Graphics, Patterns and Images, Salvador, Brazil, 26–29 August 2015; pp. 95–102. [CrossRef]

47. Yar, H.; Khan, Z.A.; Ullah, F.U.M.; Ullah, W.; Baik, S.W. A modified YOLOv5 architecture for efficient fire detection in smart cities. *Expert Syst. Appl.* **2023**, *231*, 120465. [CrossRef]

48. Valikhujaev, Y.; Abdusalomov, A.; Cho, Y.I. Automatic Fire and Smoke Detection Method for Surveillance Systems Based on Dilated CNNs. *Atmosphere* **2020**, *11*, 1241. [CrossRef]

49. Gong, X.; Hu, H.; Wu, Z.; He, L.; Yang, L.; Li, F. Dark-channel based attention and classifier retraining for smoke detection in foggy environments. *Digit. Signal Process* **2022**, *123*, 103454. [CrossRef]

50. Liu, Z.; Yang, X.; Liu, Y.; Qian, Z. Smoke-Detection Framework for High-Definition Video Using Fused Spatial- and Frequency-Domain Features. *IEEE Access* **2019**, *7*, 89687–89701. [CrossRef]

51. Gubbi, J.; Marusic, S.; Palaniswami, M. Smoke detection in video using wavelets and support vector machines. *Fire Saf. J.* **2009**, *44*, 1110–1115. [CrossRef]

52. Yuan, F.; Fang, Z.; Wu, S.; Yang, Y.; Fang, Y. Real-time image smoke detection using staircase searching-based dual threshold AdaBoost and dynamic analysis. *IET Image Process* **2015**, *9*, 849–856. [CrossRef]

53. 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]

54. Lin, G.; Zhang, Y.; Xu, G.; Zhang, Q. Smoke Detection on Video Sequences Using 3D Convolutional Neural Networks. *Fire Technol.* **2019**, *55*, 1827–1847. [CrossRef]

55. Berman, D.; Treibitz, T.; Avidan, S. Non-local image dehazing. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 27–30 June 2016. [CrossRef]

56. He, L.; Gong, X.; Zhang, S.; Wang, L.; Li, F. Efficient attention based deep fusion CNN for smoke detection in fog environment. *Neurocomputing* **2021**, *434*, 224–238. [CrossRef]

57. Sathishkumar, V.E.; Cho, J.; Subramanian, M.; Naren, O.S. Forest fire and smoke detection using deep learning-based learning without forgetting. *Fire Ecol.* **2023**, *19*, 9. [CrossRef]

58. Wu, X.; Lu, X.; Leung, H. A Video Based Fire Smoke Detection Using Robust AdaBoost. *Sensors* **2018**, *18*, 3780. [CrossRef]

59. LeCun, Y.; Bengio, Y.; Hinton, G. Deep learning. *Nature* **2015**, *521*, 436–444. [CrossRef]

60. Zeng, J.; Lin, Z.; Qi, C.; Zhao, X.; Wang, F. An Improved Object Detection Method Based On Deep Convolution Neural Network for Smoke Detection. In Proceedings of the International Conference on Machine Learning and Cybernetics (ICMLC), Chengdu, China, 15–18 July 2018; pp. 184–189. [CrossRef]

61. Khan, S.; Muhammad, K.; Hussain, T.; Del Ser, J.; Cuzzolin, F.; Bhattacharyya, S.; Akhtar, Z.; de Albuquerque, V.H.C. DeepSmoke: Deep learning model for smoke detection and segmentation in outdoor environments. *Expert Syst. Appl.* **2021**, *182*, 115125. [CrossRef]

62. Yuan, F.; Li, K.; Wang, C.; Fang, Z. A lightweight network for smoke semantic segmentation. *Pattern Recognit.* **2023**, *137*, 109289. [CrossRef]

63. Badrinarayanan, V.; Kendall, A.; Cipolla, R. Segnet: A deep convolutional encoder-decoder architecture for image segmentation. *IEEE Trans. Pattern Anal. Mach. Intell.* **2017**, *39*, 2481–2495. [CrossRef] [PubMed]

64. Chen, L.-C.; Papandreou, G.; Kokkinos, I.; Murphy, K.; Yuille, A.L. DeepLab: Semantic Image Segmentation with Deep Convolutional Nets, Atrous Convolution, and Fully Connected CRFs. *IEEE Trans. Pattern Anal. Mach. Intell.* **2018**, *40*, 834–848. [CrossRef]

65. Long, J.; Shelhamer, E.; Darrell, T. Fully Convolutional Networks for Semantic Segmentation. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Boston, MA, USA, 7–12 June 2015.

66. Noh, H.; Hong, S.; Han, B. Learning deconvolution network for semantic segmentation. In Proceedings of the IEEE International Conference on Computer Vision, Santiago, Chile, 7–13 December 2015. [CrossRef]

67. Ma, Z.; Cao, Y.; Song, L.; Hao, F.; Zhao, J. A New Smoke Segmentation Method Based on Improved Adaptive Density Peak Clustering. *Appl. Sci.* **2023**, *13*, 1281. [CrossRef]

68. Available online: https://github.com/sonvbhp199/Unet-Smoke/tree/main (accessed on 12 July 2023).

69. Guan, J.; Li, S.; He, X.; Chen, J. Peak-Graph-Based Fast Density Peak Clustering for Image Segmentation. *IEEE Signal Process Lett.* **2021**, *28*, 897–901. [CrossRef]

70. Kim, S.-Y.; Muminov, A. Forest Fire Smoke Detection Based on Deep Learning Approaches and Unmanned Aerial Vehicle Images. *Sensors* **2023**, *23*, 5702. [CrossRef] [PubMed]

71. Ashiquzzaman, A.; Lee, D.S.S.; Oh, S.M.M.; Kim, Y.G.G.; Lee, J.H.H.; Kim, J.S.S. Video Key Frame Extraction & Fire-Smoke Detection with Deep Compact Convolutional Neural Network. In Proceedings of the SMA 2020: The 9th International Conference on Smart Media and Applications, Jeju, Republic of Korea, 17–19 September 2020. [CrossRef]

72. He, K.; Zhang, X.; Ren, S.; Sun, J. Deep residual learning for image recognition. In Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, USA, 27–30 June 2016; pp. 770–778. [CrossRef]

73. Hu, Y.; Zhan, J.; Zhou, G.; Chen, A.; Cai, W.; Guo, K.; Hu, Y.; Li, L. Fast forest fire smoke detection using MVMNet. *Knowl. Based Syst.* **2022**, *241*, 108219. [CrossRef]

74. Available online: https://github.com/guokun666/Forest_Fire_Smoke_DATA (accessed on 18 July 2023).

75. Liu, Y.; Qin, W.; Liu, K.; Zhang, F.; Xiao, Z. A Dual Convolution Network Using Dark Channel Prior for Image Smoke Classification. *IEEE Access* **2019**, *7*, 60697–60706. [CrossRef]

76. Yuan, F.; Shi, Y.; Zhang, L.; Fang, Y. A cross-scale mixed attention network for smoke segmentation. *Digit. Signal Process* **2023**, *134*, 103924. [CrossRef]

77. Simonyan, K.; Zisserman, A. Very deep convolutional networks for large-scale image recognition. In Proceedings of the 2015 IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR, Boston, MA, USA, 7–12 June 2015. [CrossRef]

78. Yuan, F.; Zhang, L.; Xia, X.; Huang, Q.; Li, X. A Wave-Shaped Deep Neural Network for Smoke Density Estimation. *IEEE Trans. Image Process* **2020**, *29*, 2301–2313. [CrossRef]

79. Available online: http://saliencydetection.net/duts/#orga661d4c (accessed on 24 July 2023).

80. Li, M.; Zhang, Y.; Mu, L.; Xin, J.; Yu, Z.; Jiao, S.; Liu, H.; Xie, G.; Yingmin, Y. A Real-time Fire Segmentation Method Based on A Deep Learning Approach. *IFAC-PapersOnLine* **2022**, *55*, 145–150. [CrossRef]

81. Chen, L.-C.; Zhu, Y.; Papandreou, G.; Schroff, F.; Hartwig, A. Encoder-Decoder with Atrous Separable Convolution for Semantic Image Segmentation. *arXiv* **2018**, arXiv:1802.02611v3, 801–818.

82. Pincott, J.; Tien, P.W.; Wei, S.; Calautit, J.K. Indoor fire detection utilizing computer vision-based strategies. *J. Build. Eng.* **2022**, *61*, 105154. [CrossRef]

83. Avazov, K.; Mukhiddinov, M.; Makhmudov, F.; Cho, Y.I. Fire Detection Method in Smart City Environments Using a Deep-Learning-Based Approach. *Electronics* **2022**, *11*, 73. [CrossRef]

84. Kim, B.; Lee, J. A Video-Based Fire Detection Using Deep Learning Models. *Appl. Sci.* **2019**, *9*, 2862. [CrossRef]

85. 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.

86. Shees, A.; Ansari, M.S.; Varshney, A.; Asghar, M.N.; Kanwal, N. FireNet-v2: Improved Lightweight Fire Detection Model for Real-Time IoT Applications. *Procedia Comput. Sci.* **2023**, *218*, 2233–2242. [CrossRef]

87. Saponara, S.; Elhanashi, A.; Gagliardi, A. Exploiting R-CNN for video smoke/fire sensing in antifire surveillance indoor and outdoor systems for smart cities. In Proceedings of the 2020 IEEE International Conference on Smart Computing (SMARTCOMP), Bologna, Italy, 14–17 September 2020. [CrossRef]

88. Sun, B.; Xu, Z.-D. A multi-neural network fusion algorithm for fire warning in tunnels. *Appl. Soft Comput.* **2022**, *131*, 109799. [CrossRef]

89. Wu, X.; Zhang, X.; Jiang, Y.; Huang, X.; Huang, G.G.; Usmani, A. An intelligent tunnel firefighting system and small-scale demonstration. *Tunn. Undergr. Space Technol.* **2022**, *120*, 104301. [CrossRef]

90. Wang, Z.; Ding, Y.; Zhang, T.; Huang, X. Automatic real-time fire distance, size and power measurement driven by stereo camera and deep learning. *Fire Saf. J.* **2023**, *140*, 103891. [CrossRef]

91. Mcgrattan, K. *Heat Release Rates of Multiple Transient Combustibles*; US Department of Commerce, National Institute of Standards and Technology: Washington, DC, USA, 2020. [CrossRef]

92. Kim, D.; Ruy, W. CNN-based fire detection method on autonomous ships using composite channels composed of RGB and IR data. *Int. J. Nav. Arch. Ocean Eng.* **2022**, *14*, 100489. [CrossRef]

93. Available online: https://www.tensorflow.org/api_docs/python/tf/keras/applications/xception (accessed on 5 September 2023).

94. Available online: https://image-net.org (accessed on 2 August 2023).

95. Liu, X.; Sun, B.; Xu, Z.D.; Liu, X. An adaptive Particle Swarm Optimization algorithm for fire source identification of the utility tunnel fire. *Fire Saf. J.* **2021**, *126*, 103486. [CrossRef]

96. Fang, H.; Xu, M.; Zhang, B.; Lo, S. Enabling fire source localization in building fire emergencies with a machine learning-based inverse modeling approach. *J. Build. Eng.* **2023**, *78*, 107605. [CrossRef]

97. Truong, T.X.; Kim, J.-M. Fire flame detection in video sequences using multi-stage pattern recognition techniques. *Eng. Appl. Artif. Intell.* **2012**, *25*, 1365–1372. [CrossRef]

98. Available online: http://www.ultimatechase.com/Fire_Video.htm (accessed on 7 September 2023).

99. Available online: http://signal.ee.bilkent.edu.tr/VisiFire/Demo/FireClips/ (accessed on 7 September 2023).

100. Wang, Z.; Zhang, T.; Huang, X. Predicting real-time fire heat release rate by flame images and deep learning. *Proc. Combust. Inst.* **2023**, *39*, 4115–4123. [CrossRef]

101. Available online: https://www.nist.gov/el/fcd (accessed on 7 September 2023).

102. Wang, Z.; Zhang, T.; Wu, X.; Huang, X. Predicting transient building fire based on external smoke images and deep learning. *J. Build. Eng.* **2022**, *47*, 103823. [CrossRef]

103. Hu, P.; Peng, X.; Tang, F. Prediction of maximum ceiling temperature of rectangular fire against wall in longitudinally ventilation tunnels: Experimental analysis and machine learning modeling. *Tunn. Undergr. Space Technol.* **2023**, *140*, 105275. [CrossRef]

104. Hosseini, A.; Hashemzadeh, M.; Farajzadeh, N. UFS-Net: A unified flame and smoke detection method for early detection of fire in video surveillance applications using CNNs. *J. Comput. Sci.* **2022**, *61*, 101638. [CrossRef]

105. Chen, S.-J.; Hovde, D.C.; Peterson, K.A.; Marshall, A.W. Fire detection using smoke and gas sensors. *Fire Saf. J.* **2007**, *42*, 507–515. [CrossRef]

106. Qu, N.; Li, Z.; Li, X.; Zhang, S.; Zheng, T. Multi-parameter fire detection method based on feature depth extraction and stacking ensemble learning model. *Fire Saf. J.* **2022**, *128*, 103541. [CrossRef]

107. *ISO/TR 7240-9:2022*; Fire Detection and Alarm Systems. ISO: Geneva, Switzerland, 2022.

108. Kim, J.-H.; Lattimer, B.Y. Real-time probabilistic classification of fire and smoke using thermal imagery for intelligent firefighting robot. *Fire Saf. J.* **2015**, *72*, 40–49. [CrossRef]

109. Favorskaya, M.; Pyataeva, A.; Popov, A. Spatio-temporal Smoke Clustering in Outdoor Scenes Based on Boosted Random Forests. *Procedia Comput. Sci.* **2016**, *96*, 762–771. [CrossRef]

110. Smith, J.T.; Allred, B.W.; Boyd, C.S.; Davies, K.W.; Jones, M.O.; Kleinhesselink, A.R.; Maestas, J.D.; Naugle, D.E. Where There's Smoke, There's Fuel: Dynamic Vegetation Data Improve Predictions of Wildfire Hazard in the Great Basin. *Rangel. Ecol. Manag.* **2023**, *89*, 20–32. [CrossRef]

111. Usmani, I.A.; Qadri, M.T.; Zia, R.; Alrayes, F.S.; Saidani, O.; Dashtipour, K. Interactive Effect of Learning Rate and Batch Size to Implement Transfer Learning for Brain Tumor Classification. *Electronics* **2023**, *12*, 964. [CrossRef]

112. Tsalera, E.; Papadakis, A.; Voyiatzis, I.; Samarakou, M. CNN-based, contextualized, real-time fire detection in computational resource-constrained environments. *Energy Rep.* **2023**, *9*, 247–257. [CrossRef]

113. 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]

114. Available online: https://github.com/DeepQuestAI/Fire-Smoke-Dataset (accessed on 2 August 2023).

115. Huang, L.; Liu, G.; Wang, Y.; Yuan, H.; Chen, T. Fire detection in video surveillances using convolutional neural networks and wavelet transform. *Eng. Appl. Artif. Intell.* **2020**, *110*, 104737. [CrossRef]

116. Available online: https://cfdb.univ-corse.fr/ (accessed on 3 August 2023).

117. Toulouse, T.; Rossi, L.; Campana, A.; Celik, T.; Akhloufi, M.A. Computer vision for wildfire research: An evolving image dataset for processing and analysis. *Fire Saf. J.* **2017**, *92*, 188–194. [CrossRef]

118. Gong, F.; Li, C.; Gong, W.; Li, X.; Yuan, X.; Ma, Y.; Song, T. A Real-Time Fire Detection Method from Video with Multifeature Fusion. *Comput. Intell. Neurosci.* **2019**, *2019*, 1939171. [CrossRef]

119. He, K.; Zhang, X.; Ren, S.; Sun, J. Identity mappings in deep residual networks. In *Computer Vision–ECCV 2016. ECCV 2016*; Leibe, B., Matas, J., Sebe, N., Welling, M., Eds.; Lecture Notes in Computer Science; Springer: Cham, Switerland, 2016; Volume 9908. [CrossRef]

120. Zoph, B.; Vasudevan, V.; Shlens, J.; Le, Q.V. Learning Transferable Architectures for Scalable Image Recognition. In Proceedings of the 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–23 June 2018. [CrossRef]

121. 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. [CrossRef]

122. Peng, Y.; Wang, Y. Real-time forest smoke detection using hand-designed features and deep learning. *Comput. Electron. Agric.* **2019**, *167*, 105029. [CrossRef]

123. Khan, S.; Khan, A. FFireNet: Deep Learning Based Forest Fire Classification and Detection in Smart Cities. *Symmetry* **2022**, *14*, 2155. [CrossRef]

124. Forest Fire Dataset. Available online: https://www.kaggle.com/datasets/alik05/forest-fire-dataset (accessed on 28 August 2023).

125. Zhang, L.; Wang, M.; Fu, Y.; Ding, Y. A Forest Fire Recognition Method Using UAV Images Based on Transfer Learning. *Forests* **2022**, *13*, 975. [CrossRef]

126. Zhang, H.; Cisse, M.; Dauphin, Y.N.; Lopez-Paz, D. mixup: Beyond empirical risk minimization. *arXiv* **2017**, arXiv:1710.09412.

127. Ghali, R.; Akhloufi, M.A.; Mseddi, W.S. Deep Learning and Transformer Approaches for UAV-Based Wildfire Detection and Segmentation. *Sensors* **2022**, *22*, 1977. [CrossRef] [PubMed]

128. Bahhar, C.; Ksibi, A.; Ayadi, M.; Jamjoom, M.M.; Ullah, Z.; Soufiene, B.O.; Sakli, H. Wildfire and Smoke Detection Using Staged YOLO Model and Ensemble CNN. *Electronics* **2023**, *12*, 228. [CrossRef]

129. Alexandrov, D.; Pertseva, E.; Berman, I.; Pantiukhin, I.; Kapitonov, A. Analysis of Machine Learning Methods for Wildfire Security Monitoring with an Unmanned Aerial Vehicles. In Proceedings of the 24th Conference of Open Innovations Association FRUCT, Moscow, Russia, 8–12 April 2019.

130. Liu, W.; Anguelov, D.; Erhan, D.; Szegedy, C.; Reed, S. SSD: Single shot multibox detector. In Proceedings of the European Conference on Computer Vision, Amsterdam, The Netherlands, 11–14 October 2016. [CrossRef]

131. Zhang, Q.-X.; Lin, G.-H.; Zhang, Y.-M.; Xu, G.; Wang, J.-J. Wildland Forest Fire Smoke Detection Based on Faster R-CNN using Synthetic Smoke Images. *Procedia Eng.* **2018**, *211*, 441–446. [CrossRef]

132. Xie, X.; Chen, K.; Guo, Y.; Tan, B.; Chen, L.; Huang, M. A Flame-Detection Algorithm Using the Improved YOLOv5. *Fire* **2023**, *6*, 313. [CrossRef]

133. 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]

134. Available online: https://ieee-dataport.org/open-access/flame-dataset-aerial-imagery-pile-burn-detection-using-drones-uavs (accessed on 31 August 2023).

135. Dogan, S.; Barua, P.D.; Kutlu, H.; Baygin, M.; Fujita, H.; Tuncer, T.; Acharya, U. Automated accurate fire detection system using ensemble pretrained residual network. *Expert Syst. Appl.* **2022**, *203*, 117407. [CrossRef]

136. FIRE Dataset. Available online: https://www.kaggle.com/datasets/phylake1337/fire-dataset (accessed on 31 August 2023).

137. Available online: https://www.kaggle.com/datasets/atulyakumar98/test-dataset (accessed on 31 August 2023).

138. Available online: https://bitbucket.org/gbdi/bowfire-dataset/src/master/ (accessed on 31 August 2023).

139. Available online: https://zenodo.org/record/836749 (accessed on 31 August 2023).

140. Available online: https://mivia.unisa.it/datasets/video-analysis-datasets/fire-detection-dataset/ (accessed on 31 August 2023).

141. Available online: https://mivia.unisa.it/datasets/video-analysis-datasets/smoke-detection-dataset/ (accessed on 31 August 2023).

142. Available online: https://cvpr.kmu.ac.kr/ (accessed on 31 August 2023).

143. Available online: https://www.kaggle.com/datasets/dataclusterlabs/fire-and-smoke-dataset (accessed on 31 August 2023).

144. Available online: https://www.kaggle.com/datasets/mohnishsaiprasad/forest-fire-images (accessed on 31 August 2023).

145. Available online: https://www.kaggle.com/datasets/elmadafri/the-wildfire-dataset (accessed on 31 August 2023).

146. Available online: https://www.kaggle.com/datasets/anamibnjafar0/flamevision (accessed on 31 August 2023).

147. Available online: https://www.kaggle.com/datasets/amerzishminha/forest-fire-smoke-and-non-fire-image-dataset (accessed on 31 August 2023).

148. Wu, S.; Zhang, X.; Liu, R.; Li, B. A dataset for fire and smoke object detection. *Multimed. Tools Appl.* **2023**, *82*, 6707–6726. [CrossRef]

149. Wang, M.; Yu, D.; He, W.; Yue, P.; Liang, Z. Domain-incremental learning for fire detection in space-air-ground integrated observation network. *Int. J. Appl. Earth Obs. Geoinf.* **2023**, *118*, 103279. [CrossRef]

150. Harkat, H.; Nascimento, J.M.; Bernardino, A.; Ahmed, H.F.T. Fire images classification based on a handcraft approach. *Expert Syst. Appl.* **2023**, *212*, 118594. [CrossRef]

151. Bouguettaya, A.; Zarzour, H.; Taberkit, A.M.; Kechida, A. A review on early wildfire detection from unmanned aerial vehicles using deep learning-based computer vision algorithms. *Signal Process* **2022**, *190*, 108309. [CrossRef]

152. Pourbahrami, S.; Hashemzadeh, M. A geometric-based clustering method using natural neighbors. *Inf. Sci.* **2022**, *610*, 694–706. [CrossRef]

153. Sun, L.; Qin, X.; Ding, W.; Xu, J. Nearest neighbors-based adaptive density peaks clustering with optimized allocation strategy. *Neurocomputing* **2022**, *473*, 159–181. [CrossRef]

154. Available online: https://github.com/ckyrkou/AIDER (accessed on 1 September 2023).

155. Saini, N.; Chattopadhyay, C.; Das, D. E2AlertNet: An explainable, efficient, and lightweight model for emergency alert from aerial imagery. *Remote Sens. Appl. Soc. Environ.* **2023**, *29*, 100896. [CrossRef]

156. Kyrkou, C.; Theocharides, T. EmergencyNet: Efficient Aerial Image Classification for Drone-Based Emergency Monitoring Using Atrous Convolutional Feature Fusion. *IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens.* **2020**, *13*, 1687–1699. [CrossRef]

157. Wang, M.; Jiang, L.; Yue, P.; Yu, D.; Tuo, T. FASDD: An Open-access 100,000-level Flame and Smoke Detection Dataset for Deep Learning in Fire Detection [DS/OL]. V3. Science Data Bank. 2022. Available online: https://cstr.cn/31253.11.sciencedb.j00104.00103 (accessed on 1 November 2023).

158. Chen, Y.; Zhang, Y.; Xin, J.; Yi, Y.; Liu, D.; Liu, H. A UAV-based forest fire detection algorithm using convolutional neural network. In Proceedings of the 2018 37th Chinese Control Conference (CCC), Wuhan, China, 25–27 July 2018. [CrossRef]

159. Barmpoutis, P.; Stathaki, T.; Dimitropoulos, K.; Grammalidis, N. Early Fire Detection Based on Aerial 360-Degree Sensors, Deep Convolution Neural Networks and Exploitation of Fire Dynamic Textures. *Remote Sens.* **2020**, *12*, 3177. [CrossRef]

160. Novac, I.; Geipel, K.R.; Gil, J.E.d.D.; de Paula, L.G.; Hyttel, K.; Chrysostomou, D. A Framework for Wildfire Inspection Using Deep Convolutional Neural Networks. In Proceedings of the 2020 IEEE/SICE International Symposium on System Integration (SII), Honolulu, HI, USA, 12–15 January 2020. [CrossRef]

161. Çetin, A.E.; Dimitropoulos, K.; Gouverneur, B.; Grammalidis, N.; Günay, O.; Habiboğlu, Y.H.; Töreyin, B.U.; Verstockt, S. Video fire detection—Review. *Digit. Signal Process* **2013**, *23*, 1827–1843. [CrossRef]

162. Pereira, G.H.d.A.; Fusioka, A.M.; Nassu, B.T.; Minetto, R. Active fire detection in Landsat-8 imagery: A large-scale dataset and a deep-learning study. *ISPRS J. Photogramm. Remote Sens.* **2021**, *178*, 171–186. [CrossRef]

163. Al-Bashiti, M.K.; Naser, M. Machine learning for wildfire classification: Exploring blackbox, eXplainable, symbolic, and SMOTE methods. *Nat. Hazards Res.* **2022**, *2*, 154–165. [CrossRef]

164. Short, K.C. *Spatial Wildfire Occurrence Data for the United States, 1992–2015 (FPA_FOD_20170508)*, 4th ed.; Forest Service Research Data Archive: Fort Collins, CO, USA, 2017. [CrossRef]

165. Stocks, B.J.; Lynham, T.J.; Lawson, B.D.; Alexander, M.E.; Van Wagner, C.E.; McAlpine, R.S.; Dubé, D.E. The Canadian Forest Fire Danger Rating System: An Overview. *For. Chron.* **1989**, *65*, 450–457. [CrossRef]

166. Lundberg, S.M.; Lee, S.I. A unified approach to interpreting model predictions. *Adv. Neural Inf. Process. Syst.* **2017**, *30*, 4768–4777. [CrossRef]

167. Hu, X.; Zhang, P.; Ban, Y. Large-scale burn severity mapping in multispectral imagery using deep semantic segmentation models. *ISPRS J. Photogramm. Remote Sens.* **2023**, *196*, 228–240. [CrossRef]

168. Available online: https://www.mtbs.gov/direct-download (accessed on 9 September 2023).

169. Ronneberger, O.; Fischer, P.; Brox, T. U-Net: Convolutional Networks for Biomedical Image Segmentation. In *Medical Image Computing and Computer—Assisted Intervention*; Springer: Berlin/Heidelberg, Germany, 2015. [CrossRef]

170. Berman, M.; Triki, A.R.; Blaschko, M.B. The Lovasz-Softmax Loss: A Tractable Surrogate for the Optimization of the Intersection-Over-Union Measure in Neural Networks. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–22 June 2018; pp. 4413–4421. [CrossRef]

171. Shrivastava, A.; Gupta, A.; Girshick, R. Training Region-Based Object Detectors with Online Hard Example Mining. In Proceedings of the 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, USA, 26 June–1 July 2016.

172. Lin, T.-Y.; Goyal, P.; Girshick, R.; He, K.; Dollár, P. Focal loss for dense object detection. In Proceedings of the IEEE International Conference on Computer Vision, Venice, Italy, 22–29 October 2017.

173. Jadon, S. A survey of loss functions for semantic segmentation. In Proceedings of the 2020 IEEE Conference on Computational Intelligence in Bioinformatics and Computational Biology (CIBCB), Via del Mar, Chile, 27–29 October 2020. [CrossRef]

174. Fernandez, A.; Garcia, S.; Galar, M.; Prati, R.; Krawczyk, B.; Herrera, F. *Learning from Imbalanced Data Sets*; Springer Nature: Cham, Switzerland, 2018; ISBN 978-3-319-98073-7.

175. Iwana, B.K.; Uchida, S. An empirical survey of data augmentation for time series classification with neural networks. *PLoS ONE* **2021**, *16*, e0254841. [CrossRef] [PubMed]

176. Tian, S.; Zhang, Y.; Feng, Y.; Elsagan, N.; Ko, Y.; Mozaffari, M.H.; Xi, D.D.; Lee, C.-G. Time series classification, augmentation and artificial-intelligence-enabled software for emergency response in freight transportation fires. *Expert Syst. Appl.* **2023**, *233*, 120914. [CrossRef]

177. Kamycki, K.; Kapuscinski, T.; Oszust, M. Data Augmentation with Suboptimal Warping for Time-Series Classification. *Sensors* **2020**, *20*, 98. [CrossRef]

178. Sakoe, H.; Chiba, S. Dynamic programming algorithm optimization for spoken word recognition. *IEEE Trans. Acoust. Speech Signal Process.* **1978**, *26*, 43–49. [CrossRef]

179. Sousa, M.J.; Moutinho, A.; Almeida, M. Wildfire detection using transfer learning on augmented datasets. *Expert Syst. Appl.* **2020**, *142*, 112975. [CrossRef]

180. Sun, K.; Zhao, Q.; Wang, X. Using knowledge inference to suppress the lamp disturbance for fire detection. *J. Saf. Sci. Resil.* **2021**, *2*, 124–130. [CrossRef]

181. Zhao, H.; Jin, J.; Liu, Y.; Guo, Y.; Shen, Y. FSDF: A high-performance fire detection framework. *Expert Syst. Appl.* **2024**, *238*, 121665. [CrossRef]

182. Shahid, M.; Hua, K.-L. Fire detection using transformer network. In Proceedings of the 2021 International Conference on Multimedia Retrieval, Taipei, Taiwan, 21–24 August 2021.

183. Dosovitskiy, A.; Beyer, L.; Kolesnikov, A.; Weissenborn, D.; Zhai, X.; Unterthiner, T.; Dehghani, M.; Minderer, M.; Heigold, G.; Gelly, S.; et al. An image is worth 16x16 words: Transformers for image recognition at scale. In Proceedings of the 9th International Conference on Learning Representations, ICLR 2021, Virtual. 3–7 May 2021. [CrossRef]

184. Lin, J.; Lin, H.; Wang, F. STPM_SAHI: A Small-Target Forest Fire Detection Model Based on Swin Transformer and Slicing Aided Hyper Inference. *Forests* **2022**, *13*, 1603. [CrossRef]

185. Ali, A.; Touvron, H.; Caron, M.; Bojanowski, P.; Douze, M.; Joulin, A.; Laptev, I.; Neverova, N.; Synnaeve, G.; Verbeek, J.; et al. XCIT: Cross-covariance image trans-formers. *Adv. Neural Inf. Process Syst.* **2021**, *34*, 20014–20027.

186. Yang, C.; Pan, Y.; Cao, Y.; Lu, X. CNN-Transformer Hybrid Architecture for Early Fire Detection. In *Artificial Neural Networks and Machine Learning—ICANN 2022. ICANN 2022*; Pimenidis, E., Angelov, P., Jayne, C., Papaleonidas, A., Aydin, M., Eds.; Lecture Notes in Computer Science; Springer: Cham, Switerlandm, 2022; Volume 13532. [CrossRef]

187. Wang, X.; Li, M.; Gao, M.; Liu, Q.; Li, Z.; Kou, L. Early smoke and flame detection based on transformer. *J. Saf. Sci. Resil.* **2023**, *4*, 294–304. [CrossRef]

188. Zhang, K.; Wang, B.; Tong, X.; Liu, K. Fire detection using vision transformer on power plant. *Energy Rep.* **2022**, *8*, 657–664. [CrossRef]

189. Qian, J.; Bai, D.; Jiao, W.; Jiang, L.; Xu, R.; Lin, H.; Wang, T. A High-Precision Ensemble Model for Forest Fire Detection in Large and Small Targets. *Forests* **2023**, *14*, 2089. [CrossRef]

190. Benzekri, W.; Moussati, A.E.; Moussaoui, O.; Berrajaa, M. Early Forest Fire Detection System using Wireless Sensor Network and Deep Learning. *Int. J. Adv. Comput. Sci. Appl.* **2020**, *11*, 496–503. [CrossRef]