International Journal of Innovation and Applied Studies
ISSN 2028-9324 Vol. 39 No. 2 Apr. 2023, pp. 742-750

http://www.ijias.issr-journals.org/

Smoke and fire detection by a convolutional neural network based on a combinatorial model

Tidiane Fofana[1-2-3], Sié Ouattara[1-2-3-4], and Alain Clement[5]

[1]Laboratory of Signals and Electrical Systems (L2SE)), Institut National Polytechnique Houphouët Boigny, Yamoussoukro, Côte d'Ivoire

[2]Institut National Polytechnique Houphouët Boigny (INPHB), Yamoussoukro, Côte d'Ivoire

[3]Laboratory in Signals and Geographic Information Systems (LARSSIG) of EGT, Abidjan, Côte d'Ivoire

[4]Ecole de Géomatique et du Territoire (EGT), Abidjan, Côte d'Ivoire

[5]LARIS, SFR MATHSTIC, Université d'Angers, Angers, France

ABSTRACT: Work in the field of fire and smoke detection is becoming an increasingly covered subject. Conventional algorithms use exclusively models based on feature vectors. These vectors are difficult to define and depend largely on the type of fire being treated. These traditional methods give results with low detection rates and high false classification rates. The current trend is to take an innovative approach to solving this problem by using an algorithm to automatically determine useful features to classify fire and smoke. In this paper, we propose a convolutional neural network to identify fire and smoke from real-time images. Convolutional neural networks have shown their great performance in the field of object classification. Tested on real image sequences, the proposed approach achieves better classification performance than conventional methods. These results clearly indicate that the use of convolutional neural networks for fire detection is very encouraging.

KEYWORDS: Fire, Smoke, classification, dropout, convolutional neural network.

1 INTRODUCTION

Forest fires like any other sites (house, business, etc.) are one of the disasters that affect humans, animals and vegetation [1]. To avoid large-scale damage, it is essential to detect fires early and accurately. The majority of fire detection systems used today are based on particle sampling, temperature and air transparency tests. Unfortunately, these systems are generally limited indoors, require close proximity to the fire, and cannot provide additional information about the circumstances of the fire, such as size, location and spread. They cannot be used in open spaces and large green areas [2]. This explains the use of video surveillance systems to detect fire and smoke in open spaces [3-8]. Deep learning algorithms have the ability to learn features useful for fire and smoke classification. The main objective of this work is to implement a classification method to detect the presence of fire using cameras with the aim of avoiding large-scale fire damage with accuracy [9 We use a convolutional neural network as a powerful detector of fire and smoke in video images. We have classified our image learning base according to four classes, namely the fire class, the smoke class, the fire and smoke class and the negative class (which does not take into account the first three classes). To this end, different parameters of convolutional neural networks have been dynamically explored in order to optimize the classification or detection of fire and smoke. These include dropout, image augmentation, training image base proportions, number of recursion (epochs). We used the classification model evaluation criteria to evaluate our method, which resulted in a better classification rate of 97%.

2 METHODOLOGY AND PROPOSED METHOD

2.1 CONVOLUTIONAL NEURAL NETWORK

Convolutional networks were first introduced by Fukushima [10], he derived a hier-archical neural network architecture inspired by the research work of Hubel [11]. Lecun [12] generalized them to successfully classify numbers and to recognize handwritten control numbers by LeNet-5. Ciresan [13] used convolutional networks and achieved the best performance in the literature for multiple object recognition for multiple image databases. A convolutional neural network consists of several layers. Figure 1 shows the different layers.

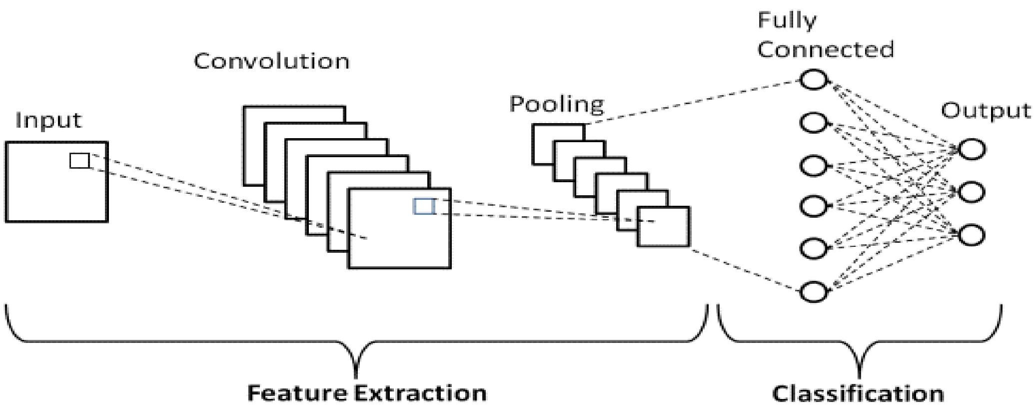

Fig. 1. CNN layers

2.1.1 Convolutional Layers

The convolutional layers form the core of the convolutional network. These layers consist of a rectangular grid of neurons that have a small receptive field extended through the entire depth of the input volume. Thus, the convolutional layer is just an image convolution of the previous layer, where the weights specify the convolution filter.

2.1.2 Pooling Layers

After each convolutional layer, there may be a pooling layer. The pooling layer subsamples their input. There are several ways to do this pooling, such as taking the average or the maximum. For example, Figure 2 shows max pooling on a 2 × 2 Windows.

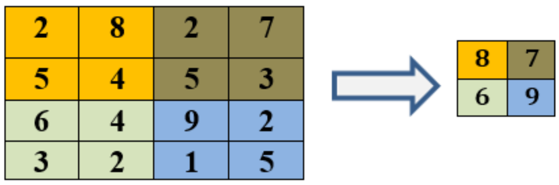

Fig. 2. Max pooling

2.1.3 Fully Connected Layers

Finally, after several layers of convolution and pooling, the high-level reasoning in the neural network is done via fully connected layers. In convolutional neural networks, each layer acts as a detection filter for the presence of specific features or patterns present in the original data. The first layers of a convolutional network detect features that can be easily recognized and interpreted. Later layers increasingly detect more abstract features. The last layer of the convolutional network is able to make an ultra-specific classification by combining all the specific features detected by the previous layers in the input data.

3 Experimental Work

The work of this paper being based on fire detection we have set up an image database divided into two parts (train, validation). The image database used for our study consists of 25,000 images; the image database contains 6250 fire images, 6250 smoke images, 6250 smoke and flames images and 6250 negative images (containing no fire, smoke, and smoke and flames). Our images have dimensions of 150 x 150 pixels for the convolution network. The normalization of the data is done by dividing all the pixel values by 255 to make them compatible with the initial values of the network. The percentage used in this work is 80% and 20%. The choice of the split ratio is based on previous work: 80% and 20% [15].

Fig. 3. Example of images available in the dataset

3.1 Overcoming Overfitting

Overfitting usually involves storing the training data set and usually leads to poor performance on the test data set. This means that performance on the training set may be excellent, but performance on the test set is quite poor. The loss of network generalization capability can be due to many factors, such as the capacity of the network or the nature of the training data set itself. Many measures have been introduced in the literature to overcome overfitting. The following are some of the techniques that have been used in this research to overcome overfitting.

3.2 Dropout

A regularization layer introduced by [16] can be applied to any layer of the network. During the training of the network, some neurons are disabled with a predefined dropout rate probability P. This can be understood as a kind of bagging for neural networks.

3.3 Image Augmentation

Increasing the size of the training set improves network performance. For image datasets, many duplicates can be created by simple modifications to the original dataset, including rotating, flipping, zooming, and cropping. These transformations make the network more robust by defending it against overfitting, and also improve network performance. In our case, the original images were flipped, rotated, zoomed, and shifted. The rotation range used was 40°; and the images were randomly flipped horizontally and vertically; the shift range used was 20%; and the zoom range used was 20%.

3.4 Early Stopping

Early termination is a precautionary measure used to prevent the network from over-adapting. It can be defined as stopping the training phase of the network when the performance on the validation set stops improving for a predefined number of epochs. In our case, the number of epochs is 100.

3.5 MODEL EVALUATION CRITERIA

The evaluation of the fire and smoke detection model can be assessed from the effect and reliability. These two indicators are usually precision and time to test, while the former includes three indicators: precision, recall rate and error rate. Among them, precision is defined as (1), and recall is defined as (2). The error rate is defined as (3).

$$P=\frac{TP}{TP+FP} \tag{1}$$

$$recall=\frac{TP}{TP+FN} \tag{2}$$

$$Error\ rate=\frac{FP+FN}{TP+TN+FP+FN}=1\text{-}recall \tag{3}$$

Among them, TP is the number of positive samples determined by the model to be positive, TN is the number of negative samples determined by the model to be truly negative, and FP is the number of positive samples determined by the model. However, the actual number of negative samples is FN, which is the number of negative samples determined by the model but actually positive, as shown in Table 1.

Table 1. Definition of TP, TN, FP and FN parameters

	Positive	Negative
True	TP	TN
False	FP	FN

4 FIRE DETECTION USING A NEURAL NETWORK MODEL

In this section, the proposed convolutional network architecture for fire detection is presented in Figure 4. Our classification architecture is classical [14], combining convolution and Max pooling. However, in order to achieve a fast classification for real-time detection and localization, we chose a lightweight network. Fig. 4 shows the four convolution layers with a 3 x 3 core. An RGB color image in the visible spectrum passes successively through convolutional operations with kernel of size 3x3 followed by a Max pooling of size 2x2 with a step of 2. The values of the filters used range from 32, 64, 128 and 256. To make our model more stable and efficient, we used a Dropout value of 0.5 before the flatten layer. At the fully connected layers, between the two dense layers we used a Dropout value of 0.1. The output of the last fully connected layer feeds a 4-way Softmax producing a 4-class distribution. For the fully connected layers, we used a ReLu activation function.

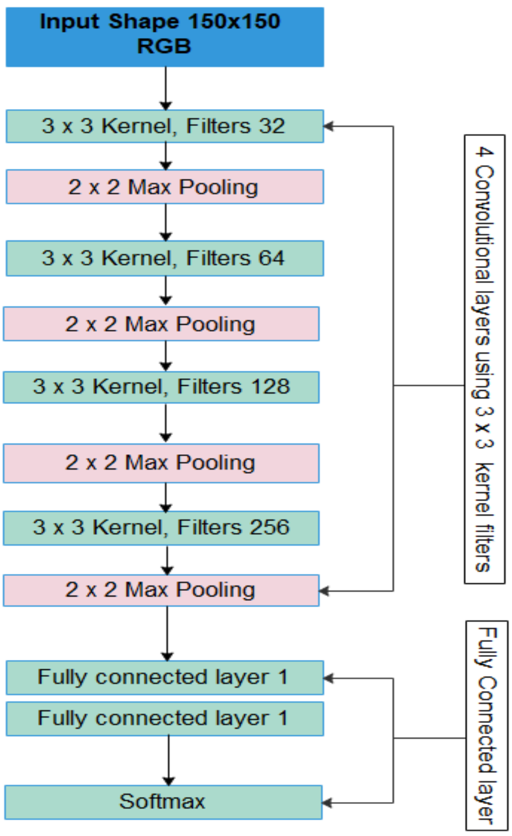

Fig. 4. Convolutional network architecture for fire detection

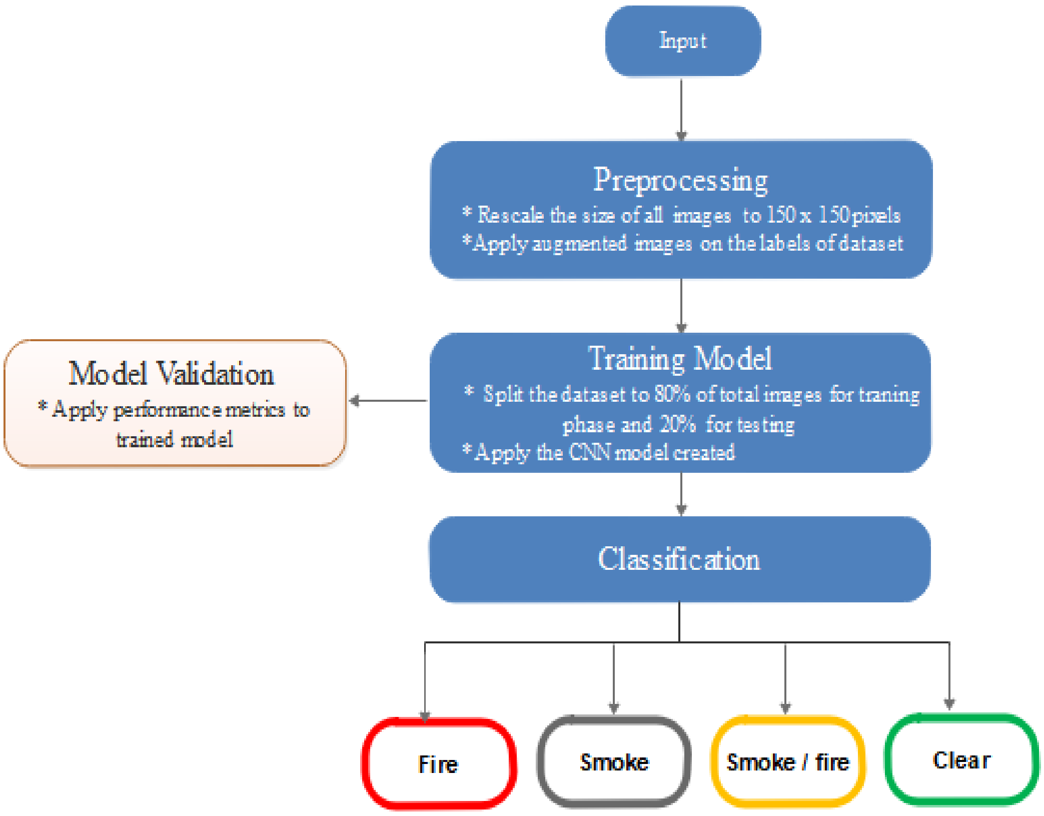

Fig. 5. The proposed method

The proposed method includes three main steps to accomplish the fire detection procedure, as follows:

- **Step 1: Pretreatment**

All images were collected in a dataset and loaded to be scaled to a fixed size of 150 X 150 pixels to be suitable for further processing in the created model. The idea of image augmentation is to add small variations without damaging the core object so that the neural network is more robust to these types of real-world variations. Thus, using image augmentation, we introduce these new images (refined by using various parameters such as zoom, rotation, width and height modification) into our training dataset and train the model.

- **Step 2: Training model and validation**

The pre-processed data set is divided into 80%; 20% according to the Pareto principle. This means that 20% of the image data will be used for the validation phase and 80% of the data will be used to build the training set. We use this breakdown in our created model, and then apply evaluation metrics to show the performance recorded on the validation set.

- **Step 3: Classification**

In the final step of the proposed model, the test data is passed to the classifier set to classify all image patches in one of four cases: fire, smoke, smoke and fire, and clear (no fire, no smoke, smokeless), as shown in Figure 2.

5 Results And Discussion

Our new model was trained and tested on Google Colab, the platform was equipped with 3 GPUs, 25 GB of RAM, python 3.7 and TensorFlow 2.6.0. This model can be used for fire detection in open spaces using cameras as a disaster prevention measure. In this work, we used a substantial number of images for training, preprocessing operations such as rotation, scaling, horizontal tilting on the image axis, random zoom increase and shearing are performed on the set of images used. The particularity of our model is its ability to classify images of simple fires, fires with smoke, smoke and images not containing the three cases mentioned above.

The number of training samples is 25000 images. The data set was divided into four samples; fire, smoke, fire and smoke and the last negative sample (which corresponds to none of the three mentioned). The four samples have 6250 images each, 5000 images for the training phase and 1250 images for the validation phase. This means that we use 20,000 images for the training and 5000 images for the validation.

5.1 Training Experience

All training and test data are divided into two folders with a ratio of 80% for training and 20% for testing. Our entire image base is used as a training set for the model in order to have good grip and generalization ability to avoid overfitting. The target detection threshold in deep learning refers to a threshold value that allows the algorithm to distinguish between positive and negative samples. The model finally provides the score of the target area through the Softmax layer to perform the classification [17].

During the threshold setting process, the algorithm tends to accept the area of the object frame closest to the actual label for setting the high threshold value, discarding the frame area from the object with lower scores, causing the model to generate high precision and low recall. Instead, the low threshold tends to accept low score object frame areas, resulting in high recall and low accuracy. It is therefore necessary to consider the state of equilibrium "between the precision rate and the recall rate. During the training phase, in order to avoid the loss of data diversification caused by too rapid a descent of the gradient, the number of iterations of the network model is set to 100, and the number of batches per iteration is 32, that is, the number of iterations is 3200, and the learning rate is 0.0001. It was important that we could determine the value of the loss and plot the iterative curve of this loss from the pattern of network formation. Fig. 6 is a trend graph of the loss value and the number of iterations caused according to the above steps.

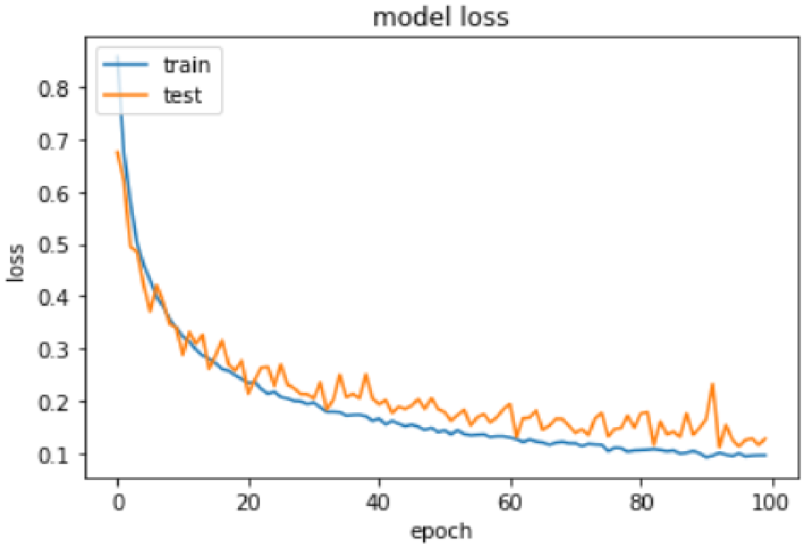

Fig. 6. Relationship between the two loss curves and the number of iterations

The abscissa is the number of iterations, and the ordinate is the loss value. It can be seen that the value of the loss decreases with the number of iterations within a certain range. The learning time of the images is 6.8 hours, and the loss value eventually tends to a stable minimum value of 0.0580 while avoiding other overfitting conditions. During this training phase the accuracy obtained for the test set is 96.68% with our entire image base.

5.2 Tests And Evaluation

The evaluation of our model created for the detection of fire images consists of 5000 images that we had divided as follows: 650 fire images, 970 smoke images, 1136 fire and smoke images and 2244 negative images (not including the other three). All images identified from human vision, in order to realize that they are in the corresponding test directories. All 5,000 test images are evaluated to see the relevance of our model for recognizing fires in open spaces. At the same time, the parameters TP, FP, TN, and FN of the model are identified and counted by human observation according to the results provided by our model. The evaluation of our model is calculated according to the formulas for calculating precision, recall rate and error rate above. The results of the classification of our model are presented in Table 2.

Table 2. Recognition effects of our model

	TP	FP	TN	FN	Precision	Recall	Error rate
Fire	650	20	4350	33	97,01 %	96,01%	3,99%
Smoke	970	28	4030	48	97,19 %	95,28%	7,27%
Smoke /Fire	1136	30	3864	46	97,43 %	96,10%	4,94%
Clear	2244	65	2756	83	97,18%	96,43%	3,15%

Most fire detection methods use two class classification techniques [18] fire or no fire and smoke or no smoke to detect and classify the object on original or resized images.

Each image passes through the convolutional neural network and the fully connected layers to the classifier. Our approach is quite different, it is based on the use of 4 classes and a resizing of the images of size 150 x 150 pixels in the original RGB image to locate the fire images or not. This work takes into account the method of image augmentation which allows us to generalize the classification method much more and to obtain a good accuracy. Our neural network consists of two parts.

The first part is composed of 4 convolutional layers and 4 Max pooling layers, the second part is composed only of fully connected layers. Using the first part of the network, we manage to extract the features of the input image. The second part is used for the classification of the output images with a 4-class Softmax. Our experimental approach is in the same vein as the one proposed [19], in which the authors studied some alternatives of fire detection with sensors.

In this work, the results obtained are about 97% overall for fire detection. Our model makes a good classification and manages to determine the four classes studied. One of the strong points of the model is that it can distinguish between images containing fire and smoke and images of fire only and can distinguish between smoke-like objects.

We can conclude that the parameters of our convolutional network model allow a good classification distinction between fire, smoke, and fire and other images without fire images.

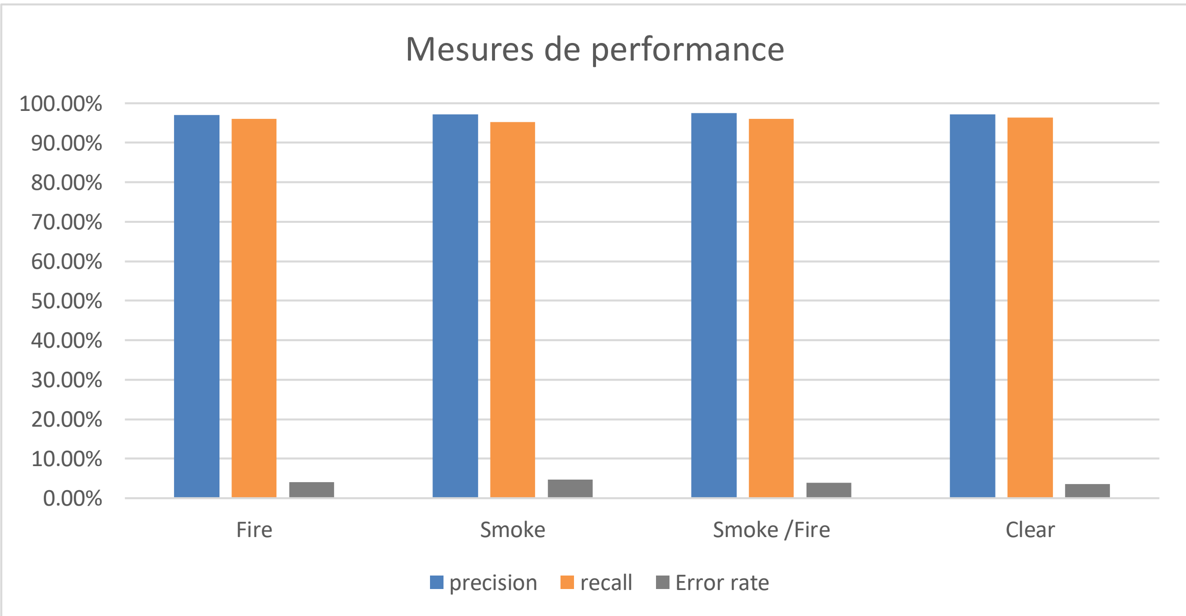

Fig. 7. Performance measures for the 4 classes: fire, smoke, smoke and fire and net area

The classification results obtained from our model are shown in Figure 8.

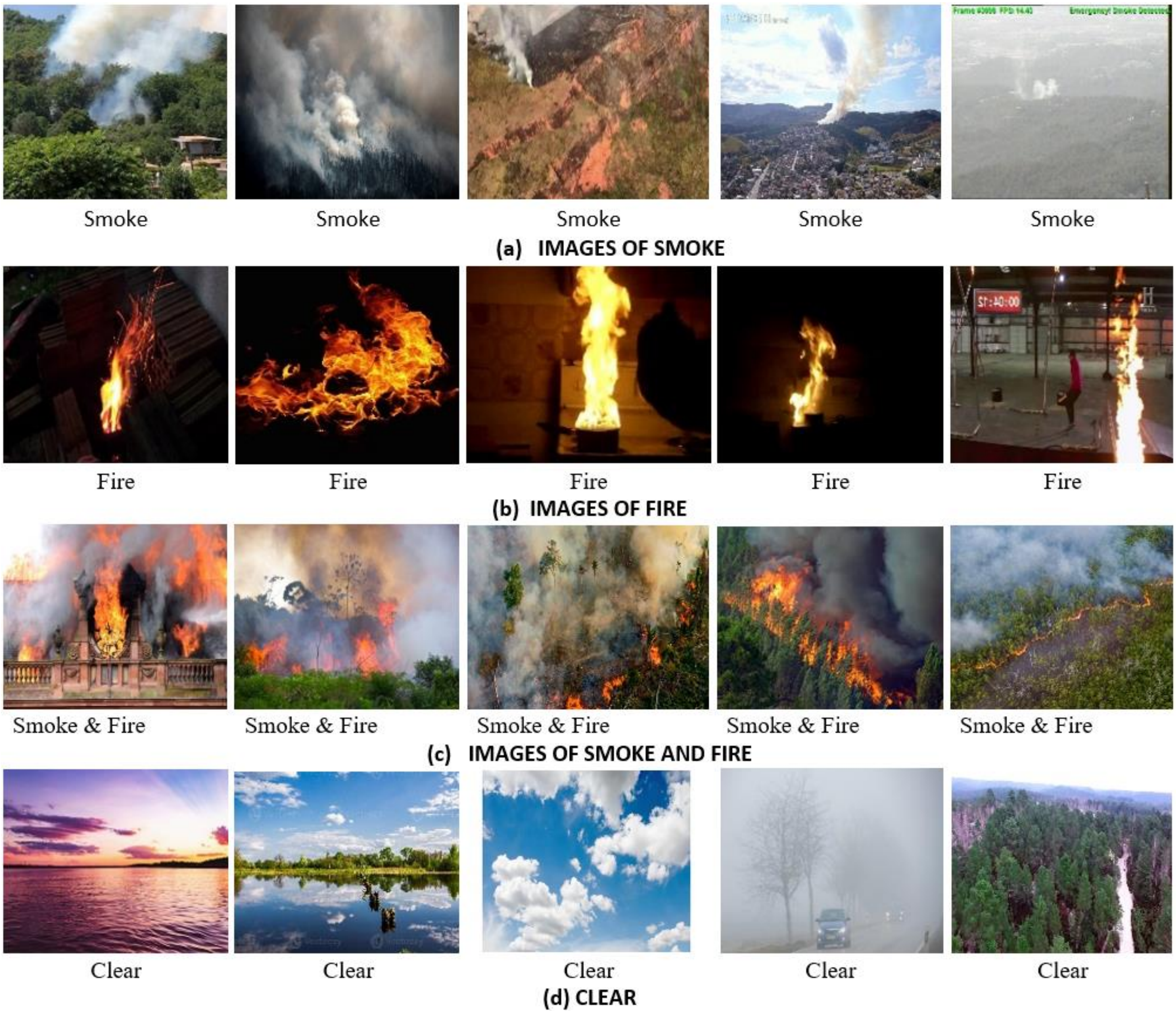

Fig. 8. Raw images of the four cases: (a) positive image for smoke detection; (b) positive image for fire detection; (c) positive image for smoke and fire detection; (d) negative image for fire detection

6 CONCLUSION

This paper describes the image-based fire detection method. One of the findings of our work is that all the essential features of smoke and fire, namely texture, color, sharp edge detection, perimeter clutter analysis, direction of movement and growth of smoke and fire areas were extracted using CNN. Our method allows to determine efficiently; fire, smoke, fire and smoke simultaneously and images not containing the three cases mentioned. Our method makes it possible to determine efficiently; fire, smoke, fire and smoke simultaneously and the images not containing the three cases cited.The accuracies obtained for the detection of fire, smoke, smoke and fire and the net appearance without all three cases are respectively; 97.01%, 97.19%, 97.43% and 97.18%. The proposed work also works well in difficult shooting conditions such as clouds, fog, and cloud reflection in water, which are very similar to smoke. The results show that the current method can learn effective flame and smoke characteristics with high accuracy. Our next challenge is to detect in real time the start of a fire or to characterize a fire from a video in a complex environment.

REFERENCES

[1] H. N. Le Houérou, «Vegetation wildfires in the mediterranean basin: evolution and trends, » *Ecologia Mediterranea,* vol. 13, pp. 13-24, 1987.

[2] S. Verstockt, B. Merci, B. Sette, P. Lambert and R. Van de Walle, «State of the art in vision based fire and smoke detection,» *Internat. Conf. on Automatic Fire Detection,* vol. 2, pp. 285-292, 2009.

[3] R. Sahal, S. H. Alsamhi, J. G. Breslin, and M. I. Ali, «Industry 4.0 towards forestry 4.0: fire detection use case,» Sensors, vol. 21, no. 3, p. 694, 2021.

[4] C. Yuan, Y. Zhang, and Z. Liu, «A survey on technologies for automatic forest fire monitoring, detection, and fighting using unmanned aerial vehicles and remote sensing techniques,» Canadian Journal of Forest Research, vol. 45, no. 7, pp. 783– 792, 2015.

[5] A. Gaur, A. Singh, A. Kumar, A. Kumar, and K. Kapoor, «Video flame and smoke based fire detection algorithms: a literature review,» Fire Technology, vol. 56, no. 5, pp. 1943–1980, 2020.

[6] Y. Luo, L. Zhao, P. Liu, and D. Huang, «Fire smoke detection algorithm based on motion characteristic and convolutional neural networks,» Multimedia Tools and Applications, vol. 77, no. 12, pp. 15075–15092, 2018.

[7] H. Yin, Y. Wei, H. Liu, S. Liu, C. Liu, and Y. Gao, «Deep convolutional generative adversarial network and convolutional neural network for smoke detection,» Complexity, vol. 2020, Article ID 6843869, 12 pages, 2020.

[8] R. Wang, Y. Li, H. Sun, and K. Yang, «Multisensor-weighted fusion algorithm based on improved AHP for aircraft fire detection,» Complexity, vol. 2021, Article ID 8704924, 10 pages, 2021.

[9] A. Gaur, A. Singh, A. Kumar et al., «Fire sensing technologies: a review,» IEEE Sensors Journal, vol. 19, no. 9, pp. 3191–3202,2019.

[10] K. Fukushima, Neocognitron: A self-organizing neural network model for a mechanism of pattern recognition unaffected by shift in position. Biological Cybernetics, 1980, 36 (4), pp.193–202.

[11] D. H. Hubel and T. N. Wiesel, Ferrier lecture: Functional architecture of macaque monkey visual cortex, Proceedings of the Royal Society of London, Series B, Biological Sciences, 1977, 198 (1130): pp.1–59.

[12] Y. Lecun, L. Bottou, Y. Bengio and P. Haffner, Gradient-based learning applied to document recognition, in *Proceedings of the IEEE,* vol. 86, no.11, pp. 2278-2324, Nov 1998.

[13] S. Frizzi, R. Kaabi, M. Bouchouicha, J. M. Ginoux, E. Moreau and F. Fnaiech, «Convolutional neural network for video fire and smoke detection,» *IECON 2016 - 42nd Annual Conference of the IEEE Industrial Electronics Society,* Florence, 2016, pp.877-882.

[14] Ibrahem K., Mauro C., Aleš P.: Comparative Study of First Order Optimizers for Image Classification Using Convolutional Neural Networks on Histopathology Images, Journal of Imaging 6 September 2020, p1-17.

[15] Srivastava, N.; Hinton, G.; Krizhevsky, A.; Sutskever, I.; Salakhutdinov, R. Dropout: A Simple Way to Prevent Neural Networks from Overfitting. J. Mach. Learn. Res. 2014, 15, 1929–1958.

[16] Zolzaya Dashdorj, Min Song. An application of convolutional neural networks with salient features for relation classification [J]. BMC Bioinformatics, 2019, 20 (10suppl).

[17] A. Gaur, A. Singh, A. Kumar et al., «Fire sensing technologies: a review,» IEEE Sensors Journal, vol. 19, no. 9, pp. 3191–3202, 2019.

[18] S. Ye, Z. Bai, H. Chen, R. Bohush, and S. Ablameyko, «An effective algorithm to detect both smoke and flame using color and wavelet analysis,» Pattern Recognition and Image Analysis, vol. 27, no. 1, pp. 131–138, 2017.

[19] D. Sheng, J. Deng, W. Zhang, J. Cai, W. Zhao, and J. Xiang, «A Statistical Image Feature-Based Deep Belief Network for Fire Detection» Research Article 2021, 12 pages.