Exploiting R-CNN for video smoke/fire sensing in antifire surveillance indoor and outdoor systems for smart cities

Sergio Saponara, *IEEE IMS Senior Member*, Abdussalam Elhanashi, Alessio Gagliardi, Dip. Ingegneria dell'Informazione
University of Pisa, Via G. Caruso 16, 56122, Pisa, Italy

***Abstract* —This work presents a video-camera-based fire/ smoke sensing technique for early warning in antifire surveillance systems. By exploiting R-CNN (Region Convolutional Neural Network), a detection technique is developed for the measurement of the smoke and fire characteristics in restricted video surveillance environments, both indoor (e.g. a railway carriage, container, bus wagon, homes, offices), or outdoor (e.g. storage or parking areas). The considered application scenario, to reduce costs, is composed of a single, fixed camera per scene, working in the visible spectral range already installed in a closed-circuit television system for surveillance purposes. The training phase is done with indoor and outdoor image sets, with both smoke and non-smoke scenarios to assess the capability of true-positive/true-negative detection and false-positive/false-negative rejection. To generate the training set, a Ground Truth Labeler app is used and applied to the open-access Firesense dataset, including tens of indoor and outdoor fire/ smoke scenes developed as the output of an FP7 project, plus other videos not publicly available, provided by Trenitalia during specific fire/smoke tests on railway wagons performed at their testing facility in Osmannoro, Italy. The achieved results show that the proposed R-CNN technique is suitable for the creation of a smart video-surveillance system for fire/smoke detection.**

Keywords – Video smoke/fire sensing, Ground Truth Labeler, R-CNN (Region Convolutional Neural Network), Smart surveillance

I. INTRODUCTION

Nowadays, fire accidents are the most occurring disasters that lead to both economic and social damage. The recent report by the Brazil National Institute for Space Research (INPE) recorded about 80000 fires in Amazonia in 2019 [1]. Recent reports of the NFPA (National Fire Protection Association) show that the average number of fires per year is about 1.3 millions in US, with high cost in terms of lives (more than 3 thousand of civilian fire fatalities) and economic losses. Many approaches have been proposed for early fire detection in order to mitigate the number of fire accidents and minimize the damage caused by such accidents. The recent development of Computer Vision and Image Processing has led to video-based smoke detection that represents Nature technology with remarkable advantages over traditional methods, such as wide detection areas and a fast response. Typical smoke/fire sensors based on chemical, photometry or thermal detection can react within several minutes, requiring an enough level of fire for detection. Such long delays in detection cause irreparable damages that have led us to the development of novel solutions to enhance the robustness and reliability of smoke/fire detection by using a visual fire detection approach. Closed-circuit television (CCTV) systems and cameras are already installed in most human environments, such as industry, and smart cities (city streets and public transportation), for surveillance purposes. Existing video infrastructure can indeed be exploited to reduce the purchase/installation cost as there is no need for additional products, the software and video processing algorithm for detecting fire and smoke can simply be added. An alternative, low-cost solution could be the realization of ad-hoc architecture based on distributed video camera nodes that can provide a user web interface for a Livestream from the cameras and the autonomous decision making of fire alarms and low-cost implementation on IoT embedded computing nodes [2].

With the availability of massive amounts of computation due to the dissemination of hardware accelerators, graphics processing units (GPU) and high-performance processors, we have witnessed a gradual increase in artificial intelligence techniques. Different deep learning models have been able to outperform human level performance in a custom computer vision application, such as image classification [3]. Consequently, as time moves on, the hand-crafted visual detection approaches have been progressively replaced by deep learning approaches due to their ability to extract features from raw images automatically. Before, such tasks had to be carried out by the developer, whose job it was to carefully, manually extract the features on the input image, combining several different techniques for the classification. Hence, in this work, our target is to present and advance fire and smoke detection by using a region-based convolutional neural network (R-CNN), which achieves significant results for visual pattern recognition.

Hereafter, the paper is organized as follows: Section I and Section II deal with a detailed review of video-based smoke/fire measurements. Section III presents the new R-CNN technique, discussing the global architecture and then each of the layers used in the video processing steps. Section IV discusses the training and validation of the R-CNN detector. Section V shows the R-CNN experiment results and discussion. Conclusions and other experimental targets are drawn in Section VI.

II. STATE OF ART VIDEO-BASED SMOKE/FIRE DETECTORS

Traditionally, researchers attempted to develop hand-crafted methods for fire and smoke detection by calculating manual features, which may be the color, texture, shapes, irregularity, flutter or frequency from the input smoke image [4]. Celik *et al.* [5] proposed a fire detection method based on color models obtained by statistical analysis fuzzy-logic to achieve discrimination between fires and fire-like colored objects. Rafiee *et al.* [6] used static characteristics (two-dimensional wavelet analysis) for detecting the color and the motion, and

978-1-7281-6997-2/20/$31.00 ©2020 IEEE
DOI 10.1109/SMARTCOMP50058.2020.00083

dynamic characteristics like the smoke disorder which implements background subtraction using frame differentiation and exploits the smoke features to discover the smoke areas. In the same direction is the work in [7] that generates the background subtraction using Visual Background Estimation (ViBe). The ViBe technique is used also in [8] in combination with the dissipation function to estimate the background. Although these hand-engineered methods are often not computationally expensive and can be easily deployed on a single-board computer like a Raspberry Pi with a good frame rate, there is the drawback of manually extracting features from the video streams. On the other hand, the DL techniques have the advantage to extract the features automatically [9], making this process more effective and dramatically improving the state-of-the-art in visual object recognition [10, 11].

Various deep learning methods have been proposed for fire detection. Zhang *et al*. used fire patches detection with a pre-trained CCN AlexNet model for forest fire detection [12]. Sharma *et al*. [13] instead, propose a CNN-based fire detection based on a pre-trained VGG16 and Resnet50 as baseline architecture. In both cases, although they achieve good results in terms of accuracy, the large disk size and the total number of parameters make these models not suitable for a low-cost implementation on an embedded platform.

Our contribution in this work is to create a light-weight R-CNN model for embedded application, able to fit into low-cost, low-performance hardware like a Raspberry Pi, and also be able to achieve reasonable performance for real-time fire and smoke detection [14]. A Ground Truth Labeler application has been used for labeling and creating the training set of collected images for our benchmark. We used such dataset in order to identify and label the features of fire and smoke in which to be trained for the R-CNN detector.

III. R-CNN FOR SMOKE/FIRE DETECTION

A. The proposed R-CNN Architerture

The development in neural networks has shown significant advancement in deep learning tasks for object classification and region extraction. In 2014 Grishick et al. explored the Neural Network and proposed the Regional Convolutional Neural Network [15]. R-CNN is a deep learning technique based on object detection methods. Instead of working with a massive number of regions, the R-CNN algorithm generates a number of boxes within the image and analyses whether the desired objects are contained in any of these boxes. It uses an Edge Boxes algorithm to extract these boxes, that are referred to as regions from the image. The proposed R-CNN smoke/fire detector consists of three modules: the first module generates a set of independent region proposals, which define the candidate detection categories that are available to the detector. The second module, a convolutional neural network, extracts a fixed-length feature vector from each region. The third module is a set of classifiers that classify desired regions. The module outputs them as object detected in bounding boxes in the image.

B. Region Proposal Extraction

The Edge Boxes algorithm represents the first module of such proposed detector, and it is used to extract the region of interest (ROI) from the image. It generates proposals for object bounding boxes which are defined as region proposals. The Edge Boxes algorithm is necessary to simplify the image into the information extraction of the object for the detector. Furthermore, these edges can be used to create maps and reduce the set of positions that needs to be further analyzed in order to locate the object within the image [16]. The region proposals will be given as input to the second module: the CNN.

C. Feature Extraction

We extract the features vector from each region proposal using a CNN implementation. The proposed architecture for this CNN is shown in Fig. 1.

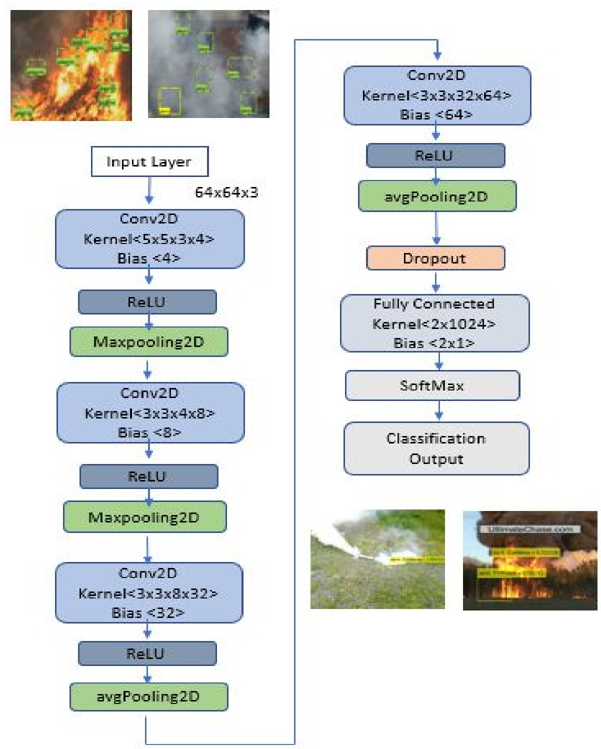

Fig. 1: The proposed architecture of R-CNN detector (fire/smoke detection). It consists of 17 Neural Network layers: Input Image Layer, 4 Convolution layers, 4 ReLU layers, two max pooling layers, two average pooling layers, dropout layer, fully connected layer, SoftMax layer and classification output layer

Deep Network Designer tool from MATLAB is used to build the R-CNN's network layers [17]. The R-CNN object detector consists of 17 layers. The first layer is the Input Image Layer, which introduces the images of the size 64x64x3. Four Convolutional Layers are used as a map feature for the input images, which use sliding filters on the image from both the vertical and horizontal axis. The first convolutional layer is size 5x5 and the rest are with size of 3x3. Four Rectified Linear Units (ReLU) Layers are constructed in this R-CNN, which introduce the non-linearity to the system and convert all values in images smaller than zero to zero. Max pooling and average pooling layers are used to perform downsampling of images into rectangular pooling regions. We applied 2 x 2 for size of pooling with a stride of 2 x 2 for all pooling layers in our network. The Dropout layer with value 0.6 is used to prevent overfitting in the archticture, and the Fully-Connected Layer is used to perform the final classification decision. The Softmax Layer is constructed as a function which computes an input vector of K real numbers. K is the number of neurons in softmax layers. It maps the output value of neuron into new values in interavals between 0 and 1, which can be thought as probability of the predicted class. The softmax function is defined by the formula Eq. (1) :

$$SoftMax\ (z_i) = \frac{exp(z_i)}{\sum_j exp(z_j)} \qquad (1)$$

where $Z_i = \log\overline{P}$ (y = i | x).

Finally, the Classification Output Layer is used to compute cross-entropy loss for multi-class classification.

D. Runtime Detection

At runtime, the detector runs Edge Boxes algorithm on test image and extract around 1000 region proposals. Each proposal is then resized and forwarded to the CNN in order to compute the features. Finally, the region proposal bounding boxes are rifined by a support vector machine (SVM) classifier that is trained using CNN features. For all given scored regions in an image, it is applied a greedy non-maximum suppresion (NMS) (for each classes) rejecting a region if it has an intersection-over-union (IoU) overlap with a higher scoring selected region larger tha a learned threshold. A system overview is showed in Fig. 2.

We used a dataset of 194 images to train the R-CNN detector in our experiment. The following steps discussed in Section IV were carried to process the training for the classifier.

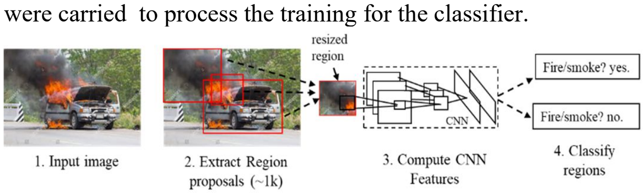

Fig 2 : R-CNN fire / smoke detection system overview.

IV. TRAINING & VALIDATION THE R-CNN DETECTOR

A. Pre-processing the Training Data

- Collected images of fire and smoke were uploaded into the Ground Truth Labeler application in Matlab
- The Regions of Interest (ROI) of fire and smoke data were created to train the detector using a specific Ground Truth algorithm tool to label the selected features with different sizes[18].
- The Ground Truth data was exported into Matlab workspace to produce the arrays and coordinates of labeled features.

B. Training The R-CNN Detector

The created code runs on Matlab script. The detector is saved at the end of the training process. Fine-tunning is applied by using stochastic gradient descent with momentum (sgdm) [19]. We set the mini-batch size of 32. This value was selected based on the size of the training dataset.The learning rate was set to le-4 in order to control the model change in response to the estimated error and mini-batch training loss during the training stage (see Fig 3). Typical momentum 0.9 was used to accelarate the speed of the training. The network trained for 50 Epochs and training accuracy resulted in 93.8 %. All training hyper-parameters were set as given in Table 1. The training initialization weights for the neural network are very important [20], since the better the tuning for the hyper-parameters, the better the model can learn. The final size of the R-CNN fire/smoke detector is about 220KB, very lightweight to be fit into a small embedded device.

TABLE 1: Training Hyper-Parameters For R-CNN

Parameter	Method
Training options	sdgm
L2 Regularization	0.09
Learning rate drop factor	0.2
Momentum	0.9
Number of Epochs	50
Dropout value	0.6
Mini-batch size	32
Learning rate	0.0001

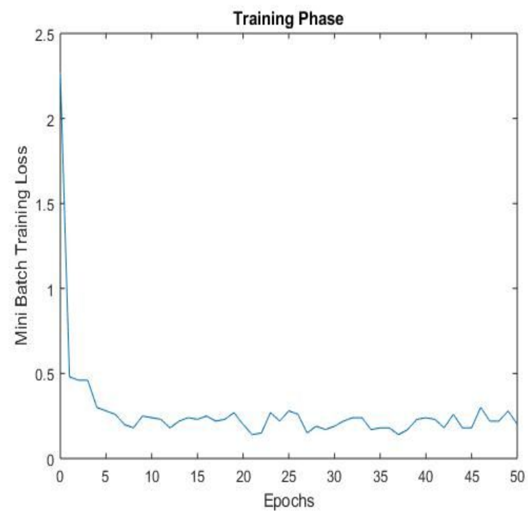

Fig. 3 : Mini-Bartch Loss Curve for the R-CNN

C. Validtion the R-CNN Detector

We prepared an independent dataset of 200 images (100 images with fire/smoke and 100 images with NO fire /smoke) to validate the R-CNN. As per the results from Receiver Operating Characteristic (ROC) analysis, the accuracy for this validation was 91% see Table 2 and Fig. 4.

TABLE 2: Summary for Validation Results by ROC tool analysis

Matrices	Validation values
Number of Images	200
Accuracy	91%
Sensitivity	99%
Specificity	82%

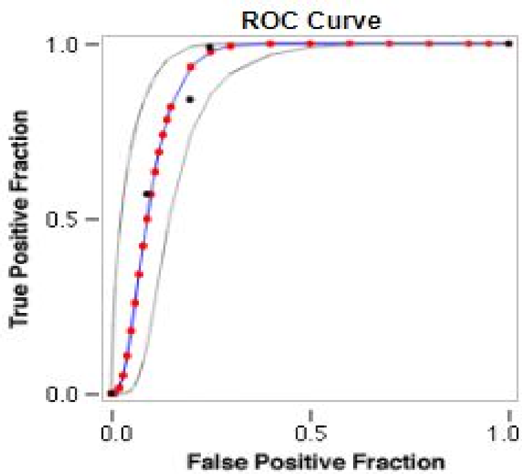

Fig 4: ROC Curve for validation Dataset

V. EXPERIMENT RESULTS AND DISCUSSION

The target of this research is to exploit the trained deep learning model for Video-based camera of fire and smoke detection. In literature, methods used for testing data are still images instead of videos. In methods [21], [22] and [5], there is a lack of diversity for using videos to encounter various realistic situations for fire/smoke and normal conditions. In our proposed approach, we tested 287 videos from different environments (indoor, outdoor, forest, railways, parking and public area).

(a)


(b)

Fig. 5: a) Sample Images from Videos with Fire & Smoke b) Sample Images from Videos with no Fire & Smoke

In the considered data set, 117 videos (65224 frames) contained a non-smoke/fire condition, and 170 videos (107189 frames) contained smoke and fire. Different matrices (false positive rate, false negative rate and accuracy) were analyzed as per confusion matrix criteria in Eq. (2). The R-CNN achieved good classification results for fire and smoke detection see Fig. 5 and overcomes all other methodologies [21], [23], [22], [5] with the best accuracy of 96.5% in respect to 92.86%, 91%, 87.1% and 83.7% and minimum value for false positive rate (see Table 3).

$$\begin{aligned} Accuracy &= (TP + TN) / (TP + FN + TN + FP) \\ False\ positive\ rate &= FP / (FP + TN) \\ False\ negative\ rate &= FN / (FN + TP) \end{aligned} \tag{2}$$

TABLE 3: Performance of the proposed approach vs. state-of-art

Method	False positive (%)	False negative(%)	Accuracy (%)
Proposed R-CNN	**8.5**	**0**	**96.5**
De Lascio et al [21]	13.33	0	92.86
Fu T J, Zheng et al [23]	14	8	91
Chen et al. [22]	11.76	14.29	87.1
Celik et al. [5]	29.41	0.00	83.7

A. Performance of R-CNN with Dataset2

In order to confirm the efficiency of the proposed approach, we evaluated the R-CNN with Dataset2 from Method [24]. The dataset consisted of 46 fire videos and 16 non-fire videos, 160 non-fire images and 639 fire images. The number of videos is limited but it is challenging, e.g. it contains sunset light in non-fire videos (see Fig 5). The results are compared with 4 methodologies reported in [24], [25], [26] and [5]. We used metrics (Accuracy, Recall , Precision and F-Measure) to evaluate the effectiveness and present a complete analysis of our work. Based on the results, the R-CNN showed the best result for Recall value that there was no any false negative value recorded, which means no fire/smoke detection was missed as shown in Table 4. MATLAB 2019a, and a built-in application Neural Network Designer, Ground Truth Labeler and Intel® Core TM I3-6006U CPU @ 2 GHz were utilized as support tools for our experiments.

TABLE 4: The Performance of the proposed approach vs other methodologies

Method	Accuracy %	Recall %	Precision %	F -Measure %
Proposed R-CNN	**93.6**	**100**	**92.4**	**96.04**
Method [24]	93.9	94	97	95
Filonenko et al [25]	85	96	85	90
Yuan et al [26]	86	53	86	65.5
Celik et al. [5]	83	60	60	60

B. R-CNN with Dropout

Dropout is a method that minimizes the problems of overfitting in the neural network. It is a regularization tool that eliminates the complex co-adaption amongst the neurons to achieve the best performance for the model in testing stage [27]. We used a dropout layer in the R-CNN model after the hidden layers. As per the results in the table 5, the accuracy increases by up to 96.5 %. The dropout layer enhanced the performance for the R-CNN and it reduced false detection errors in comparison to other methodologies as can be seen in table 3 and 4. We saw that the overall results of the proposed architecture improved when dropout was used with neural network layers.

TABLE 5: The Performance of R-CNN using the Dropout Layer

R-CNN Architecture	Accuracy (%)
R-CNN	78.3
R-CNN + Dropout	96.5

C. Size of Detected Regions

The regional convolutional neural network technique becomes the mainstream algorithm in object detection. Early detection plays an important role in minimizing the damage and loss from fire and smoke accidents. In our proposed method, the R-CNN can detect multiple objects of fire and smoke in the frame where each object is enclosed by Bounding Box. The proposed approach can detect small fire and smoke regions see Fig. 6. We sampled the minimum values of detected Bounding Boxes from our testing dataset see Table 6. The smallest region recorded was 39 x 17 (width, height in pixels) in image 5. As per these results, the R-CNN has an advantage based on its techniques to detect small regions in videos as an early alerting sign for fire and smoke detection. To be noted that differently from the fast R-CNN adopted in [28] this work aims analyzing video captured through surveillance cameras already installed in smart cities or smart transportation systems, looking to areas distant from few meters to hundreds of meters. Instead in [28] the goal is the analysis of large areas, such as wildland forests, observed from a distant camera.

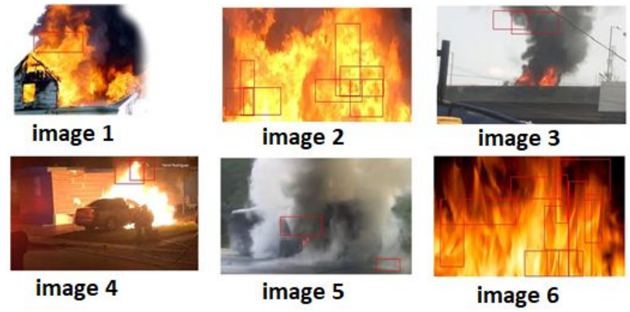

Fig. 6: Bounding Boxes of detected regions of Fire and Smoke from sampled images

TABLE 6: Size of the minimum Detected Bounding Boxes of smoke and fire (width & height in pixels)

Images	Minimum size of Detected Bounding Boxes (*width, height in pixels*)
1	79 x 31
2	44 x 40
3	56 x 22
4	40 x 36
5	39 x 17
6	18 x 74

D. Testing the R-CNN with Embedded Device

The R-CNN detector tested with Raspberry Pi 3 model B and a video camera with running the R-CNN detector code on MATLAB on the PC. We simulated multiple videos as situations of real smoke/fire and non-smoke/fire condition from another computer and exposed them to the video camera (see Fig 7). As per the results, the R-CNN model achieved the detection of fire and smoke and it indicated normal condition as the non-fire/smoke videos simulated to the video camera. This represents only a preliminary investigation for the deployments of the algorithm on such presented embedded system. In fact, we intend in the next future, making a fully implementation on a low-cost Raspberry Pi testing its performance and power consumption while running the R-CNN.

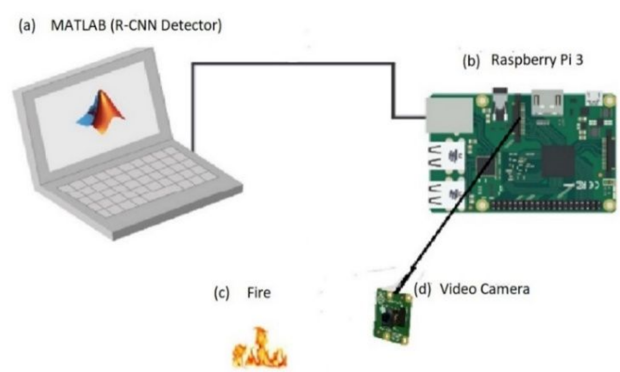

Fig. 7: Testing the R-CNN Detector with real state of fire & smoke . The Hardware setup consists of : a) Running the R-CNN Detector with MATLAB on PC b) Raspberry Pi 3 c) Fire / smoke video simulated from other computer d) Video Camera

VI. CONCLUSIONS & OTHER EXPERIMENTAL TARGETS

In this paper, we proposed fire and smoke deep learning architecture using a Regional Neural Network for surveillance videos. As per the experimental results, the R-CNN detector achieved good classification performance in different environmental conditions, including both indoor and outdoor scenarios. In addition, we demonstrated that the R-CNN can detect small regions as an early alerting sign for the occurrence of fire and smoke accidents. Final network designed with light-weight architecture, which makes it suitable to work on small embedded devices. By using as input the videos obtained by already installed surveillance cameras the proposed technique is very suitable for adoption in smart cities or smart transportation system. In the future, we intend to further our exploration of this work to deploy a network of regional neural network detectors using low-cost IoT devices, such as Raspberry Pi nodes, interconnected through wireless links such as WI-FI and or 4G LTE for which a coverage in smart cities is typically ensured.

ACKNOWLEDGMENT

Work partially supported by the Crosslab Project "Dipartimenti di Eccellenza" by MIUR. We thank the Islamic Development Bank for their support to the PhD work of A. Elhanashi.

REFERENCE

[1] Gibbens, Sarah. "The Amazon Is Burning at Record Rates-and Deforestation Is to Blame." Brazil's Amazon Is Burning in Historic Wildfires-and Deforestation Is to Blame, 23 Aug. 2019,

[2] A. Gagliardi, S. Saponara, "Distributed Video Antifire Surveillance System based on IoT Embedded Computing Nodes", Springer LNEE, vol. 627, 2020

[3] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," Nature, vol. 521, no. 7553, p. 436, 2015

[4] S. Saponara, L. Pilato, L. Fanucci, "Early Video Smoke Detection System to Improve Fire Protection in Rolling Stocks", Proc. of SPIE, vol. 9139, 2014

[5] T. Celik, H. ¨Ozkaramanlı, and H. Demirel, "Fire and smoke detection without sensors: Image processing based approach," 15th European Signal Processing Conference, pp. 1794–1798, IEEE, 2007.

[6] A. Rafiee, R. Dianat, M. Jamshidi, R. Tavakoli, and S. Abbaspour, "Fire and smoke detection using wavelet analysis and disorder characteristics," 3rd IEEE International Conference on Computer Research and Development, vol. 3, pp. 262–265, 2011

[7] S. Vijayalakshmi, S. Muruganand, "Smoke detection in video images using background subtraction method for early fire alarm system," 2nd Int. Conf. on Comm. and Electronics Systems (ICCES), Coimbatore, 2017, pp. 167-171.

[8] "ANTINCENDIO." SEMA Safety S.r.l., semasafety.it/antincendio/. Accessed on 02/10/2019

[9] Y. Bengio, "Learning Deep Architectures for AI," Foundations & Trends in Machine Learning, vol. 2, pp. 1-127, 2009.

[10] A Krizhevsky, I. Sutskever, and G.E. Hinton "ImageNet Classification with Deep Convolutional Neural Networks," Advances in Neural Information Processing Systems, vol. 25, pp. 1097-1105, 2012.

[11] O. Barnich, M. Van Droogenbroeck, "ViBE: A powerful random technique to estimate the background in video sequences" IEEE ICASSP 2009

[12] Q. Zhang, F. Liu, X. Li and B. Li, "Dissipation Function and ViBe Based Smoke Detection in Video," 2nd International Conference on Multimedia and Image Processing (ICMIP), Wuhan, Hubei, China, 2018, pp. 325-329.

[13] J. Sharma, O.-C. Granmo, M. Goodwin, J. T. Fidje, "Deep convolutional neural networks for fire detection in images," International Conference on Engineering Applications of Neural Networks, pp. 183– 193, Springer, 2017.

[14] P. Barmpouti, K. Dimitropoulos, N. Grammalidis, "Real time Video Fire Detection using SpatioTemporal Consistency Energy", 10th IEEE International Conference on Advanced Video and Signal Based Surveillance (AVSS,. Krakow, 2013, pp. 365 – 370.

[15] Girshick, R., J. Donahue, T. Darrell, and J. Malik. "Rich Feature Hierarchies for Accurate Object Detection and Semantic Segmentation", IEEE Conference on Computer Vision and Pattern Recognition. 2014, pp. 580–587.

[16] C. Zitnick and P. Dollár, "Edge boxes, locating object proposals from edges", 13th European Conference on Computer Vision, pp. 391–405, 2014.

[17] Designer, D. (2020). Build Networks with Deep Network Designer- MATLAB & Simulink. [online] Mathworks.com. Available at: https://www.mathworks.com/help/deeplearning/ug/build-networks-with-deep-network-designer.html Retrieved January 10, 2020.

[18] MathWorks Team (2019). Using Ground Truth for Object detection (https://www.mathworks.com/matlabcentral/fileexchange/69180-using-ground-truth-for-object-detection), MATLAB Central File Exchange. Retrieved October 24, 2019.

[19] Glorot, Xavier, and Yoshua Bengio. "Understanding the difficulty of training deep feedforward neural networks." Proceedings of the thirteenth international conference on artificial intelligence and CCD camera", IEEE Trans. on Instrumentation and Measurement, vol. 54, n. 4, 2005

[20] A. Namozov, Y. I. Cho, "An Efficient Deep Learning Algorithm for Fire and Smoke Detection with Limited Data," Advances in Electrical and Computer Engineering, vol.18, no.4, pp.121-128, 2018

[21] Di Lascio, R.; Greco, A.; Saggese, A.; Vento, M. "Improving fire detection reliability by a combination of video analytics". International Conference Image Analysis and Recognition, Vilamoura, Portugal, 22–24 October 2014; Springer: Cham, Switzerland, 2014.

[22] Chen, T.H.; Wu, P.H.; Chiou, Y.C. "An early fire-detection method based on image processing". IEEE International Conference on Image Processing (ICIP), Singapore, 24–27 October 2004; pp. 1707–1710

[23] Fu T J, Zheng C E, Tian Y, Qiu Q M, Lin S J. "Forest fire recognition based on deep convolutional neural network under complex background, Computer and Modernization, 2016, vol. 3, pp. 52-57.

[24] Jadon, A.; Omama, M., Varshney, A., Ansari, S., Sharma, R., "FireNet: A Specialized Lightweight Fire & Smoke Detection Model for Real-Time IoT Applications", arXiv:1905.11922v2, 2019

[25] A. Filonenko et al., "Fast Smoke Detection for Video Surveillance Using CUDA", IEEE Trans. Industrial Informatics, vol. 14, n. 2, 2018, pp. 725-733

[26] F. Yuan, Z. Fang, S. Wu, Y. Yang, and Y. Fang, "Real-time image smoke detection using staircase searching-based dual threshold AdaBoost and dynamic analysis," IET Image Process., vol. 9, n. 10, pp. 849–856, 2015.

[27] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, R. Salakhutdinov, "Dropout: A Simple Way to Prevent Neural Networks from Overfitting", Journal of Machine Learning Research 15 (2014) 1929-1958 (2014)

[28] Qi-xing Zhang, Gao-hua Lin, Yong-ming Zhang, Gao Xu, Jin-jun Wang, "Wildland Forest Fire Smoke Detection Based on Faster R-CNN using Synthetic Smoke Images", Procedia Engineering, vol. 211, 2018