Fire and Smoke detection using weakly supervised methods.

Bernardo Amaral
Insituto Superior Técnico
Lisbon, Portugal
bernardo.m.amaral@tecnico.ulisboa.pt

***Abstract*—Forest fires have become a recurring disaster worldwide and, every year, thousands of hectares of forests are devastated. The impacts on nature and society are disastrous. To develop robust deep learning methods for fire and smoke detection a large number of data and the related annotations are required. However, the number of publicly available forest fires datasets is very scarce. For this reason, this work proposes an alternative system capable of detecting and localizing fire and smoke in aerial images using only weakly-supervised deep learning methods. A classification model was trained using only image-level labels and, from there, the information from the convolutional layers was extracted to create the first iteration of a segmentation mask. Afterwards, by combining it with the colour and spatial information of the original image, one can create a segmentation mask that can correctly detect the fire/smoke zones. The proposed method was tested and proven to be able to accurately detect fire/smoke at the pixel-level despite never being trained with any supervision at that level. Compared with other fully-supervised methods, the results show that when considering their heavy needs, the proposed weakly-supervised system can strongly compete with them.**

***Index Terms*—fire detection, smoke detection, convolutional neural networks, weakly-supervised methods, deep learning**

I. Introduction

Forest fires are a scourge that every year destroy thousands of hectares of forest around the world. They have a series of effects on both the burned area and the underlying areas. To mitigate the dangers and minimize the impacts on people and nature, there is the need to have systems capable of doing effective prevention, early warning, and a clever firefight.

This work is included in the Firefront project[1] which intends to develop a system to help and support the firefighting teams. This solution consists in creating a much efficient and fast firefight by detecting and tracking fire fronts and their possible reburns. The process would be carried out through the use of aerial vehicles that are equipped with RGB and infrared cameras and other sensors and communication systems.

A. Challenges

The detection of fire and smoke through an image-based system poses a considerable challenge since neither fire nor smoke has a well-defined shape and a constant colour. The usual methods of locating/identifying objects using deep learning are trained on a large amount of fully annotated data, which means that in each image of the dataset there must be an annotation of where each class is present in the image. Though, the creation of such annotations is very expensive.

A possible alternative to this complex and time-consuming process is the use of image-level annotations where instead of having information about the class in each pixel, there is only information about the presence of the class in the entire image. Figure 1 illustrate the difference between the two labelling methods. So, weakly supervised segmentation aims to create segmentation masks at the pixel-level using only image-level labels to train the models. This way, it is possible to create a greater number of annotations in a short period of time. However, this type of annotation has the cost of losing detail in the annotation. It is then necessary to understand whether the advantage of getting a greater number of annotations outweighs the cost of losing detail in the annotations.

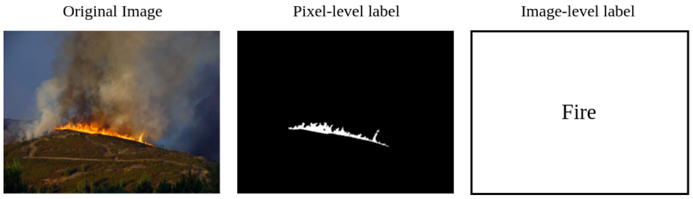


Fig. 1. Labelling methods

II. State Of The Art

A. Weakly-Supervised Segmentation

Most of the weakly supervised segmentation approaches are based on the Class Activation Mapping (CAM) method [1], which will be detailed in Section III, since it will also be an important component of this work. From this starting point, various techniques can be used either to improve this approach by making it more robust and accurate or integrating it with other methods.

When doing a weakly-supervised segmentation the output mask is obtained using the features in the image that the classification model used to make the prediction [2]. Sometimes, these features will only contain the most relevant and distinguishable parts of the object. On [3] they proposed to use a Hide-and-Seek approach where they randomly occlude patches in the training images so that each image can be used for training multiple times but with none or different patches occluded.

[1] http://www.firefront.pt/

A more complex alternative to the CAM is the Grad-CAM proposed in [4]. The authors propose an approach to obtain the object localization using the gradients of the predicted class in the final convolutional layer and not only the layer information. The method can be applied on a wide variety of CNN without the need of performing any architectural changes or re-training. For a sake of simplicity, in this work it will only be used the simpler CAM method.

B. Fire And Smoke Detection

The need to create new and wiser methods of monitoring, detecting and fighting fire has led to the development of recent research. The detection part is the one that has powered the largest amount of approaches, through the use of vision-based systems [5] either terrestrial [6] or aerial [7] . Regarding the type of techniques used, the approaches can be separated into classic, that rely on typical computer vision techniques [8], and deep learning methods [9] where the prediction is done using the features that Artificial Neural Networks extract from several examples of similar images.

1) Classic Methods: The big majority of works in fire and smoke detection based on computer vision are based on colour, spatial and temporal features. These characteristics are very specific for fire compared to other objects [10]. However, the same does not happen with smoke since it can get high similarities with other objects like clouds. Most of the approaches follow a common pipeline, first find moving pixels using background subtraction and then apply a colour model to find fire regions [11]. The base approach is to create a mathematical based model, defining a sub-space on a colour space that represents all the fire-coloured pixels in the image [12]. On this line, Wang et al. [13] proposed a method based on a Gaussian model learned in the YCbCr colour space. The major drawback of only colour based fire detection models is the high false alarm rates since single-colour information is insufficient in most cases so, on [14], the authors added texture analysis to create *Bowfire*.

In a forest fire, the fixed cameras are placed on high altitude spots from which they can cover large areas of forest terrain. One example operating in Portugal is *CICLOPE* proposed in [15]. The authors present a system of a surveillance system that can perform remote monitoring and automatic fire and smoke detection using background subtraction, feature matching and colour analysis.

2) Deep Learning: All of the previous methods depend heavily on the features delimited by the authors, which may make them too specific for a certain situation as concluded in [16]. On the other hand, methods using deep learning methods, automatically learn which features are best for the given problem. On [17] the authors do a comparative analysis between colour-model based methods versus deep learning methods. With a very simple deep learning method they can obtain the best overall performance compared to all colour-based models.

Still using surveillance cameras but now with deep learning methods, the authors in [18] present a solution for real-world surveillance scenarios using a computationally efficient CNN based fire detection system. Q Zhang et al. [19] due to the lack of forest fire smoke images, created a dataset where they added two kinds of smoke, real smoke and simulative smoke, into forest background. It was then used to train a CNN that was later tested on real forest smoke images.

The lack of a good public accessible dataset for fire and smoke makes it hard to develop a good deep learning technique. This problem is transversal and strongly highlighted by the vast majority of authors [20].

With this into account, the number of weakly supervised segmentation approaches has been growing lately.

In [21] the authors create a CNN in which they combine the feature maps of the last convolutional into a single feature-map. Then, using a sliding window on that layer they can predict the presence of fire and smoke. Similarly, in [22], the authors use a classification model to extract the information from three selected feature maps of a convolutional layer and create a mean activation map to later convert into segmentation mask.

The scarcity of datasets in this area of work makes deep learning methods quite limiting. Therefore, in this work, it is proposed to overcome this problem using weakly supervised supervised methods for fire and smoke segmentation. This way, one can use the few existing datasets and complete them with a large number of examples only annotated at the image level. However, all the works in this area using weakly supervised methods present a final detection with little precision and lack of detail. This work overcomes such limitation by combining weakly-supervised method with a post-processing method, creating a segmentation mask with great detail that closely resembles the shape of fire and smoke.

III. Methodology

The proposed approach intends to develop a system capable of detecting and localizing areas of fire and smoke in images using weakly supervised methods. The system is divided into two similar systems, one for fire and one for smoke. Both systems follow the same pipeline described in Figure 2 with the only difference being the parameters used in the different components.

The images will be taken from aerial vehicles equipped with visual-based equipment and will then be transmitted to the proposed system. Then, the image starts by being fed to a classification model the presence of fire/smoke in the whole image will be analyzed. If fire/smoke is detected then it goes on to the next phase, otherwise, the system ends here. The next phase consist in using the CAM method to extract the information that the classification model used to make the classification prediction and create a probabilistic heatmap. In this heatmap, areas with higher probabilities correspond regions where fire/smoke are likely to be present. Then, a binarization is applied to the heatmap to create a binary mask that can locate the fire/smoke location but cannot correctly represent it in terms of its shape. Therefore, a post-processing phase is then required. For that, the Conditional

Random Fields (CRF) method is used which takes as input the coarse and blob-like binary mask and the original image. By combining both inputs and analyzing the spatial and colour correlations, the method can create a segmentation mask at the pixel level that represents the location of the fire/smoke as well as its shape in a detailed way. In the following subsections the methods used in each of the main blocks of the proposed system will be describe in detail .

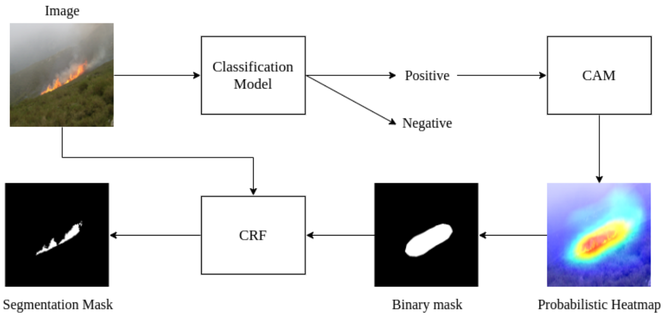


Fig. 2. Overall proposed approach

A. *Classification Model*

For the classification model, the choice was to use the VGG network architecture proposed in [23] as a starting point. The network is composed of 5 main blocks of convolutional layers followed by a max-pooling layer and then has 3 fully-connected layers. However, to apply the CAM algorithm, which will be explained in the next subsection, it was necessary to do some changes in the network. First, it is necessary to remove all fully-connected layers since they disrupt the spatial integrity maintained in the convolution layers. Next, a global average pooling layer is added to calculate the spatial average of each feature map in the last convolutional layer. In the end, one final fully connected layer and *Sigmoid* activation function is added. For the fire model, the 19 weight layer version - VGG19 - was used while for the smoke model the 16-layer version - VGG16 - was used.

B. *Class Activation Mapping*

The CAM algorithm proposed by [1] was used in order to get the information from the classification model. It is based on the idea that a classification CNN develops localization capabilities, despite being trained only with image-level labels. It is then necessary to understand which are the features of the image that the model uses to classify it into the predicted class.

The idea of adding a global average pooling layer after the convolutional layers is to summarize each feature map in the last layer into a node. So, each node represents a feature map that represents a region in the image. To perform the classification, each node will be weighted according to the relevance of the feature map it represents for the prediction. A feature map can be weighted positively if the visual pattern that it represents is relevant for the output or negatively if not. So, one can make a weighted sum of each feature from the last convolutional layer to produce a heatmap as:

$$H_c(x,y) = \sum_{i=1}^{N} w_i^c f_i(x,y). \tag{1}$$

Where H represents the CAM heatmap with the predicted location, w_i^c is the weight of the activation of the i^{th} feature map $f(x,y)$ for the predicted class c and N being the number of feature maps. Figure 3 illustrates the summing process.

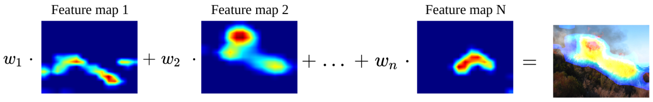


Fig. 3. Weighted feature map sum.

The heatmap will highlight the zones in the image that the model has used for the prediction and thereby those are the zones where the class is more probable to be present.

C. *Conditional Random Fields*

CRF can be used for segmentation tasks [24] either by itself or in combination with other segmentation techniques for example with deep learning neural networks [25]. When by itself the CRF uses traditional hand-crafted features as a prior and when conjugated with other techniques it relies on them to provide the features and then act as post-processing. In a post-processing situation, the image can be seen as a graph where each pixel is perceived as a node and the nodes are connected with edges ξ. Each node can have a finite set of states corresponding to the possible classes and each state has a unary cost $\psi_u(x_i)$ for each pixel. The pairwise cost $\psi_p(x_i, x_j)$ between nodes is determined by the spatial and color distance within the two pixels i and j. The graph may be built as a grid where only adjacent pixels are connected to each other, or fully-connected, as in this case, where each pixel is connected to all other pixels in the image. Finally, the assignment of each pixel to label is treated as an energy minimization problem where the energy corresponds to the sum of the total unary and pairwise costs as in Equation (2). It is an iterative process wherein at each inference step the energy is progressively minimized.

The fully connected CRF uses neighbouring context to predict the class of a pixel. In a fully connected situation every pixel in an image can be used to determine the class of one pixel and so, the energy function can be described as:

$$E(x) = \sum_{i} \psi_u(x_i) + \sum_{i,j} \psi_p(x_i, x_j), \tag{2}$$

where x represents the set of labels corresponding to each pixel, i and j range from 1 to N, being N the size of the image, $\psi_u(x_i)$ is the unary potential and $\psi_p(x_i, x_j)$ is the pairwise potential.

The unary potential sets a cost of assigning a label x_i to pixel i with a probability $P(x_i)$. The probability decides how much weight the unary mask should have in the energy function $E(x)$. The potential is then described as:

$$\psi_u(x_i) = -log(P(x_i)), \tag{3}$$

where $P(x_i)$ is the pixel probability at pixel i.

The pairwise potential $\psi_p(x_i, x_j)$ sets a cost to assign the label to pixel i in pairs, i.e. the cost of pixel i will be according to pixel j. It will analyze the neighbouring pixels to predict the class for pixel i. This potential has the form of a linear combination of Gaussian kernels as:

$$\psi_p(x_i, x_j) = \mu(x_i, x_j) \underbrace{\sum_{m=1}^{K} w^{(m)} k^{(m)}(f_i, f_j)}_{k(f_i, f_j)}, \tag{4}$$

where the term $\mu(x_i, x_j)$ is a label compatibility function which is responsible for introducing a penalty for nearby pixels that are assigned different labels: $\mu(x_i, x_j) = 1$ if $x_i \neq x_j$. Each $k^{(m)}(f_i, f_j)$ is a Gaussian kernel between the feature vectors f_i, for pixel i, and f_j for pixel j. The $w^{(m)}$ acts as a weight factor defining the importance of each Gaussian in the linear combination.

For image segmentation the $k(f_i, f_j)$ is a contrast-sensitive two-kernel potentials as:

$$k(f_i, f_j) = w^{(1)} \underbrace{exp\left(-\frac{|p_i - p_j|^2}{2\theta_\alpha^2} - \frac{|I_i - I_j|^2}{2\theta_\beta^2}\right)}_{appearance\ kernel} + w^{(2)} \underbrace{exp\left(-\frac{|p_i - p_j|^2}{2\theta_\gamma^2}\right)}_{smoothness\ kernel}. \tag{5}$$

It consists of a weighted sum of a position and colour sensitive double Gaussian with weight $w^{(1)}$ with a position-sensitive single Gaussian with weight $w^{(2)}$. The first Gaussian is the appearance kernel and it controls the degrees of nearness and similarity with the idea that nearby pixels with similar colours are likely to belong to the same class. The first term depends on both pixel positions p_i and p_j and the scale of the Gaussian is controlled by the spatial standard deviation θ_α. The larger the standard deviation, the flatter the Gaussian and furthest pixels will be taken into account. The second term depends on both pixel colour intensities I_i and I_j and the scale of the Gaussian is controlled by the colour standard deviation θ_β. As before, the larger the standard deviation, the flatter the Gaussian and wider is the range of colours that will be taken into account. The second Gaussian is the smoothness kernel and it removes small isolated areas giving a sharper boundary delimitation. It only depends on both pixel positions and is controlled by the smoothness standard deviation θ_γ. This standard deviation is a similar behaviour as the first one.

IV. Experiments

To develop the proposed system it was necessary to do some setup procedures to develop and optimize the methods that compose the proposed system. Each of the following subsections will describe them in detail.

A. Dataset

The first and most important step for computer vision and deep learning approaches is the creation of a complex and diverse dataset of images. Given that the number of publicly available forest fires datasets is quite scarce, it was necessary to create them. Therefore, two datasets were created, one with annotations at the image-level to train, validate and test the classification model and another with annotations at the pixel-level to validate and test the segmentation approach.

1) Image-level dataset: As a starting point, for the fire examples, the dataset of [26] was used considering that it contains good examples of forest fires as well as controlled fires. Additionally, it was augmented with negative examples gathered manually from the web. For the smoke examples, the data from [27] and from [28] was used as starting point and it was then augment it with individual images gathered from the web. In total, only 40% of dataset images were from free available datasets and the remaining 60% were gathered by hand which shows the scarcity of good and freely available datasets The dataset composition is described in Table I.

TABLE I
Dataset composition

# Images	Class		Percentage [%]
1807	Fire	Positive	70
		Negative	30
	Smoke	Positive	70
		Negative	30

2) Pixel-level dataset: This set is composed of images from both classes and their ground truth at the pixel-level. For the fire examples, the starting point was once again the dataset of [26] since it also contains the ground truth masks of the images. Regarding the smoke examples, the starting point was [27], to which there had to be some post-processing to get simple binary masks. Both datasets were augmented with an internal dataset created by the team of the Firefront project. In the end, the dataset is composed by 600 images of fire, 260 of smoke and the respective ground-truth masks.

B. Classification Stage

The classification stage is a crucial part of the proposed approach, since it is the entry point of the proposed approach and it will decide which image should be analyzed for segmentation. So, the image will only move forward to the weakly supervised segmentation and then to the post-processing stage if it is classified as positive. Any image that is classified as negative will be discarded. Therefore, it is necessary to have a good classification phase to prevent any false positive or false negative.

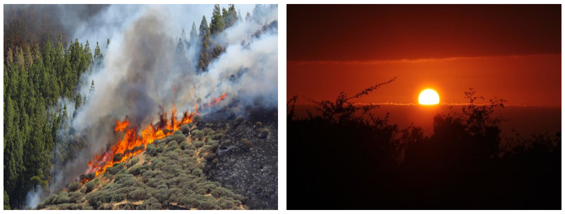
When training the network with this dataset annotated at the image-level, each image is telling the network that the class for which it is labelled is present in the entire image. Thus, the network will learn features of the entire image itself, and not just the specific class. Considering an aerial image setting, there will always be much more information in a single image than just the class itself. Especially, in aerial images of fire and smoke, there will be common co-occurring objects such as vegetation, clouds, etc. So, it is harder for the network to collect class-specific features.

Table II shows the model parameters that lead to the best results, taking into account the classification results and visual perception of the localization results.

TABLE II
MODELS PARAMETERS

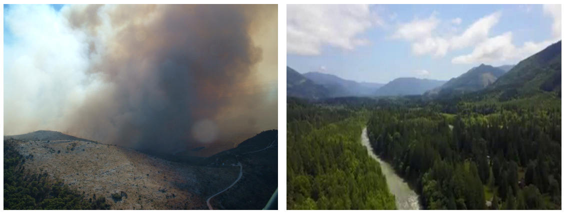
Parameter	Fire Model	Smoke Model
Base Model	VGG19	VGG16
Optimizer	Adam	Adam
Learning Rate	1e-05	1e-06
Loss	Binary Crossentropy	Binary Crossentropy
Batch size	32	32
Early Stopping	patience = 10	patience = 10
Monitor	Validation Loss	Validation Loss

Figure 4 and Figure 5 show examples of correct classification by the fire and smoke model respectively.



(a) Prob. = 1.0 **(b)** Prob. = 0.0

Fig. 4. Examples of images classification and their respective probability of containing fire according to the Fire Model: (a) is a True Positive example and (b) is a True Negative



(a) Prob. = 1.0 **(b)** Prob. = 0.0

Fig. 5. Examples of images classification and their respective probability of containing smoke according to the Smoke Model: (a) is a True Positive example and (b) is a True Negative

C. Weakly-Supervised Stage

So, to extract the location of fire and smoke from the classification model the CAM algorithm was used. Therefore, by summing representative features and subtracting unrepresentative ones, it is possible to highlight the image regions that the network used to predict the class. Consequently, these are the image regions where fire/smoke is more probable to be located. Figure 6 illustrates the overall CAM process for the fire model. In the end, there is illustrated the final heatmap for the input image.

For the fire case, and following what was proposed in [29], all feature maps with negative associated weight were discarded since these were being associated with fire zones rather than background. That said, there will be no subtraction of feature maps, only those with positive weights were used for Equation (1), resulting:

$$H_c(x,y) = \sum_{i=1}^{n} w_i^c f_i(x,y), \quad if\ w_i^c > 0 \tag{6}$$

For the smoke case, this is not verified, and the subtraction of feature maps with negative weights is beneficial.

The final heatmap behaves as an object detector despite no supervision on the location was provided. Figure 7 shows some examples of the CAM heatmap for the fire and smoke model. It can be seen that the heatmaps assign a high probability in the correct location of fire and smoke and their respective extent despite the model has never been trained for that task.

The following step consists on transforming the heatmaps into a binary mask. So, the probabilistic heatmap is thresholded according to its maximum values as:

$$\theta = \alpha \max(H), \tag{7}$$

where θ is the thresholding value, α is a real between 0 and 1 and H is the probabilistic heatmap. Every pixel in the heatmap that has a probability superior to the threshold was set to 1 and below set to 0. For the fire model, the α was set to 0.5 while for the smoke one it was set to 0.2.

D. Post-Processing Stage

In order to address the lack of detail in the masks produced by CAM, there was the need to do some post-processing. The CRF with fully connected nodes [24] was used to transform the binary masks created by CAM (with little detail), into masks that could actually resemble the shapes with well-defined boundaries much detail. To achieve this, the CRF minimizes an energy function as in Equation (2). It is divided in two potentials, the unary described by Equation (3) and the pairwise described by Equation (4).

The unary potential (3) is responsible for assigning a cost to each pixel, according to its probability in the CAM mask. As the CRF takes information from all the pixels in the image, it is needed a unary potential for the foreground as well as the background and for that reason, every pixel in the CAM mask that is not considered to be positive for fire/smoke is considered to be background. So, the two possible states for each pixel are fire/smoke and background. The background mask $M_b(x,y)$ is the opposite of the foreground mask $M_f(x,y)$ as $M_b(x,y) = 1 - M_f(x,y)$. In the foreground mask, every

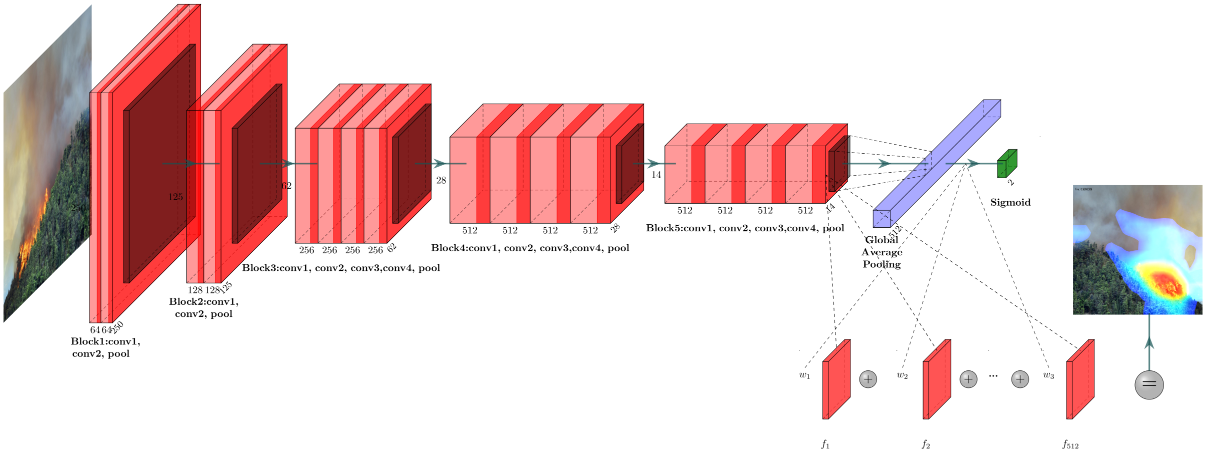
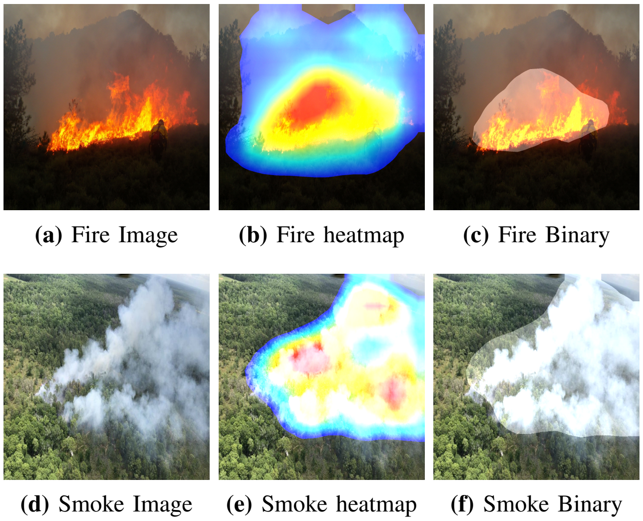

250
125
62
28
14
64 64 250
Block1:conv1, conv2, pool
128 128 125
Block2:conv1, conv2, pool
256 256 256 256 62
Block3:conv1, conv2, conv3,conv4, pool
512 512 512 512 28
Block4:conv1, conv2, conv3,conv4, pool
512 512 512 512 14
Block5:conv1, conv2, conv3,conv4, pool
512
Global Average Pooling
2
Sigmoid
w_1
w_2
w_3
...
f_1
f_2
f_{512}

Fig. 6. Overall CAM approach for the fire model



(a) Fire Image **(b)** Fire heatmap **(c)** Fire Binary

(d) Smoke Image **(e)** Smoke heatmap **(f)** Smoke Binary

Fig. 7. CAM resulting heatmap before and after being converted into binary for both fire and smoke.

positive pixel is set to 0.8 since CAM masks are not extremely precise.

On the pairwise potential (5), the appearance kernel uses the information of the pixel colour (RGB values) and the distance to their neighbours to assign the cost of the pixel. It is divided into two parts, one that is responsible to analyse the degree of nearness controlled by the spatial standard deviation θ_α and another that is responsible to analyse the degree of similarity controlled by the colour standard deviation θ_β. For both fire and smoke, since CAM cannot correctly delimit their boundaries, the θ_α should be set to a high value. Therefore, each pixel is compared with a wide range of pixels around it, allowing to create a segmentation of the whole area of the fire or smoke. The second part is responsible to analyse the degree of similarity controlled by the colour standard deviation θ_β. For fire and taking into account that fire has a very specific and limited colour range, the θ_β should be set to a low value. Therefore, only pixels with very identical colour ranges are considered to be in the same class. Regarding smoke, the θ_β cannot be as low as fire since smoke colours can range a little more and are not so characteristic. The smoothness kernel, uses pixel proximity to remove small isolated regions and give the mask a much sharper boundary, controlled by the smoothness standard deviation θ_γ. For both situations, the value should be chosen to remove some miss-detected areas and give the fire and smoke the correct boundary limits.

For both situations, a total of 5 inference steps are performed to get the final mask. Table III list the best parameters achieved to produce the final masks. Figure 8 illustrates an example of the overall approach of the CRF using the best parameters for smoke.

TABLE III
CRF BEST PARAMETERS

Parameter	Fire	Smoke
$w^{(1)}$	10	8
θ_α	250	100
θ_β	10	5
$w^{(2)}$	5	5
θ_γ	20	10
Iterations	5	5

V. Experimental Results

To validate and test the proposed system several experiments were performed.

A. Comparison with fully-supervised methods.

In this experiment, are compared two fully supervised segmentation methods (Method 1 and Method 2) and the proposed one. For Method 1 is done an extensive comparison on both fire and smoke segmentation while for Method 2 only the metrics for the fire case. Both models were developed in-house by members of the Firefront team using similar datasets.

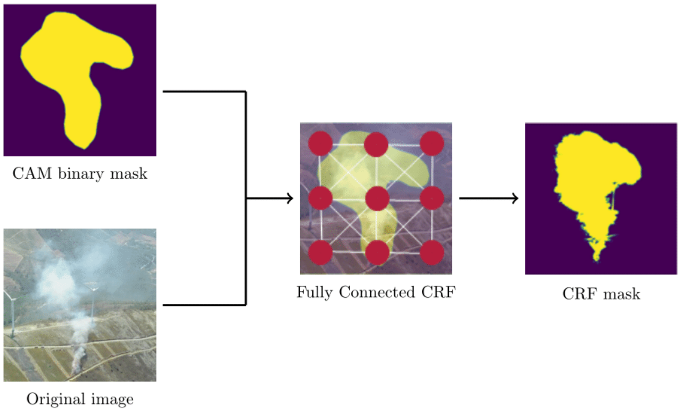


Fig. 8. Example of CRF approach for smoke.

- **Method 1** [30] - Originally the system was composed by three components. The first block analyzes the image and scaled down to the size of the network input, for detecting fire/smoke using a classification model (SqueezeNet [31]). If the classification is positive the scaled image is fed to a segmentation network, otherwise the unscaled image divided in 4 patches and the same process is repeated for each patch. When an image/patch is classified as fire/smoke, it goes to the segmentation network (U-Net [32]) to create a segmentation mask with the regions of the image that contains fire/smoke. The obtained masks are then stitched in the right places to obtain the overall segmented image. The objective of Method 1 is to be able to detect fire/smoke in high resolution images even when the fire/smoke regions are just a few pixels. However, because the proposed method only uses images scaled to the network input size, the comparison will be done using a simpler version of the method without the recursive patch subdivision. This network was trained with a dataset fully annotated at the pixel level.
- **Method 2** - The second method consists in a Deeplab v3+ network [33] applied to the fire detection and was also trained with a dataset fully annotated at the pixel level. For this second method we only had access to the metrics values, we did not have access to the segmentation masks.

Both methods were tested using the Pixel-level dataset. In order to compare them, the average mean intersection over union (mIoU) was computed.

For fire, the performance is shown in Table IV. It can be seen that both fully-supervised methods outperform the proposed one. However, it must be taken into account the effort that the authors had to put to create the strong supervision on their training examples versus the effort of creating weakly supervised supervision. So, it is natural that the performance of fully-supervised methods is better than the weakly-supervised ones. Despite the difference in mIoU, the proposed method can also achieve considerably good results as it can be seen on Figure 9. The fully-supervised masks were obtained using Method 1 and the proposal masks are the output of the proposal method. In Figure 9 is noticeable that the proposed method achieves very good final segmentation mask. The proposal masks can even better represent the small details in the fire shape when compared with the fully-supervised segmentation performed in the entire image. However, from the standard deviation results in Table IV it is concluded that the proposed method has a higher oscillation in the results compared to Method 1. The proposed methods sometimes has some discrepancies resulting in a degradation of the segmentation mask, while Method 1 is more coherent. In summary, the proposed weakly-supervised method can highly compete with the fully-supervised ones with good fire segmentation results despite some small discrepancies.

TABLE IV
MODELS COMPARISON FIRE

Method	Approach	mIoU	Standard Dev.
Method 1	fully-supervised	0.856	0.073
Method 2	fully-supervised	0.902	-
Proposed	weakly-supervised	0.735	0.142

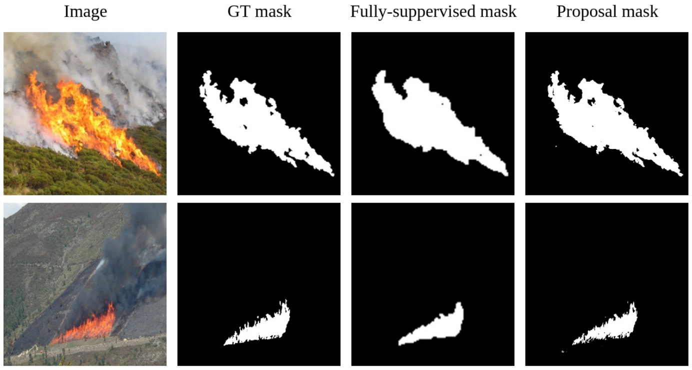


Fig. 9. Fire masks comparison between Method 1 and the proposed approach.

For smoke, the proposed approach will be only compared with Method 1 as Method 2 was not trained to detect smoke. The average mIoU results are shown in Table V. The results show that the proposed method performs on par with the fully-supervised one, achieving a similar values of mIoU and standard deviation. As concluded before, the proposed method can easily resemble the round and soft margins of the smoke zones. Figure 10 represent some results where the proposed method outperforms the fully-supervised one. Not only the proposed mask can represent the outer shape of the smoke but can also outline objects that are inside the smoke area. The fully-supervised masks are more conservative without much detailed margins. However, as expected, there are also some not fully successful examples. These examples occur in images with areas with very similar colours to smoke, like clouds, or when it is not clear the separation with common co-occurring objects and zones. In summary, the proposed weakly-supervised method can perform almost as good as a fully-supervised segmentation method, with the advantage that no pixel-level masks are needed which in the case of smoke can be very ambiguous since smoke does not have sharp boundaries and can sometimes be very dim.

TABLE V
MODELS COMPARISON SMOKE

Method	Approach	mIoU	Stantard Dev.
Method 1	fully-supervised	0.771	0.157
Proposed	weakly-supervised	0.760	0.149

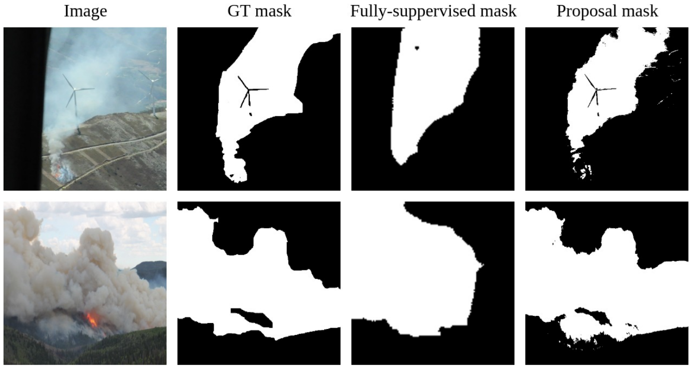


Fig. 10. Smoke masks comparison between Method 1 and the proposed approach.

The results have shown that even by only using annotations at the image-level to train the proposed method, it is possible to compete with methods that uses annotations at the pixel level. The tedious and expensive process of creating pixel-level labels is not completely reflected in the segmentation results, especially for smoke, where the proposed method performs as well. When considering the trade-off between segmentation performance and the expensiveness of the creation of pixel-level labels the proposed smoke segmentation method is the clear winner, while the fire segmentation model also wins if the task at hand tolerates some distortions in the segmentation masks.

B. Assessing the Classification Model

This experiment will evaluate the performance of both models regarding classification.

TABLE VI
CLASSIFICATION PERFORMANCE

Metric	Fire Model	Smoke model
Accuracy	0.854	0.911
Precision	0.834	0.937
Recall	0.901	0.907

From table VI, it can be observed that the fire classifier does not have a very high accuracy nor precision value, but it presents a good recall value. This means that the number of FN is small, i.e., there are few predictions in which there was indeed fire but the model did not predict it. This is a good indicator in a real situation when one does not want to neglect the presence of fire. On the other hand, the precision value is lower, which means that there was a considerable number of FP, i.e., there are images that effectively do not have fire but are being classified as fire. The vast majority of these images are smoke images without fire, which demonstrates once again the high correlation between fire and smoke and the challenge that is to separate both by using labels at the image level. For the smoke situation, all the metrics are slightly higher. Nevertheless, it is possible to perceive the same correlation situation as in fire. In this case, the recall value is a little lower than the precision value since the number of FN is higher than the number of FP. In other words, there are more cases where there was no smoke but the model predicted that there was smoke than cases where there was smoke and the model predicted that there was not.

The models' overall classification performance is good even though the smoke model is slightly superior to the fire one. As expected, it is not easy for the network to correctly extract the class-specific features which make fire and smoke classification at the image-level quite challenging.

C. Evaluating the CRF influence

This experiment evaluates how the CRF affects the system performance in the two phases of the pipeline: weakly supervised segmentation (WSSegm.) and post-processing (Post.Process.). The first stage is evaluated through the masks obtained by CAM while the second stage is evaluated through the masks obtained after the application of the CRF. It will also be evaluated the trade-off between the use of the post-processing stage and processing time. The results that follow were obtained using the Pixel-level dataset.

Table VII illustrates the performance of the two stages for both models in terms of the mIoU, the corresponding standard deviation and the processing time.

TABLE VII
SEGMENTATION PERFORMANCE IN BOTH STAGES

Model	Stage	mIoU	St.Dev.	Proc.Time (s)
Fire	WSSegm.	0.607	0.115	0.068
	Post.Process.	**0.735**	0.142	0.228
Smoke	WSSegm.	0.703	0.121	0.059
	Post.Process.	**0.760**	0.149	0.229

Firstly, one must highlight the good performance of weakly supervised segmentation (WSSegm) taking into account that the model was only trained for image classification at the image-level. It can be observed that the mIoU values for the smoke case are higher than the fire values. This can be explained by the fact that while the fire shape can be quite detailed, smoke usually has a non-detailed shape, sometimes resembling blobs. The processing time in this phase is considerably low since CAM works with small mapping resolutions.

Secondly, after the application of the CRF, a great improvement in the mIoU is noticed. This improvement is much more significant for the fire case since it is necessary to add all the detail and sharpness of the fire shape. For fire, the improvement is about 20% while for smoke it is almost 10%. There is a slight increase in standard deviations but it is

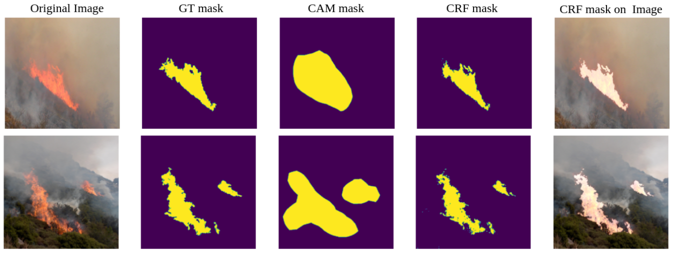


Fig. 11. Comparison of the masks in the different stages for fire

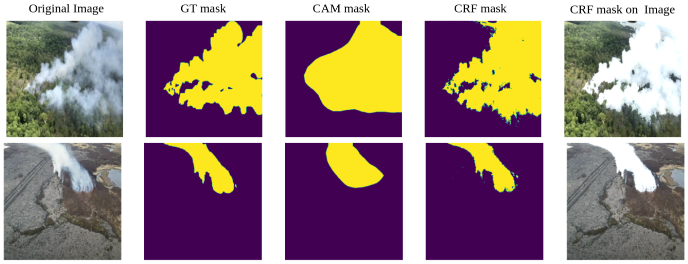


Fig. 12. Comparison of the masks in the different stages for smoke.

not comparable to the improvements in mIoU. Regarding the processing time, applying the 5 iterations of the CRF to each image took an average computation time of $230ms$ for both cases.

Figure 11 illustrates, for fire, some examples of the resulting masks in both stages of the approach as well as the ground-truth (GT) mask. At first, one can conclude that CAM can correctly predict the location of fire, although the CAM masks are quite coarse, in the form of blobs. Thus, when comparing the CAM masks with the GT masks, the mIoU difference is mostly due to lack of detail, rather than poor location. Then, after applying the CRF, the improvements in terms of detail and sharpness are highly notorious. The CRF can transform a coarse and blob-like mask roughly indicating the location of fire, into a mask very similar to the GT. The CRF takes great advantage of the fact that the fire has a very representative and limited colour space. The resulting masks are sometimes even more detailed than masks created on GT.

Figure 12 illustrates, for the smoke case, some examples of the resulting masks in both stages of the approach as well as the GT masks. Considering that smoke does not have shape as precise as fire, it is easier to get a better segmentation mask using only CAM. This is also reflected in the mIoU values in Table VII using only CAM, which are already considerably good. However, it is still necessary to use CRF to correctly delineate the smoke edges. This includes removing areas where CAM masks give overlay between smoke and other non-smoke areas, for example, fire. After applying the CRF it turns out that the masks are much more detailed and much more accurate. Similarly to what happened with the fire, the masks after the application of the CRF can sometimes be more detailed than the ones created by hand.

Despite all the aforementioned advantages, the CRF is still totally dependent on the input CAM mask. So, when this mask gives an unreasonable segmentation, it can sometimes happen that the CRF converges to non-fire/non-smoke zones.

VI. Conclusions

Reaching the end of this work, it is time to draw some conclusions on the study and work developed, and state some proposals as future work to prevent failure cases and improve the system.

The lack of data in this field, especially the scarce amount of freely available datasets with annotations at the pixel-level and the expensive and very subjective process of manually creating those labels, has led to the creation of a system that only relies on weakly-supervised methods. On the other hand, the creation of labels at the image-level for these methods can be almost effortless and makes the process of gathering new images very easy. The system developed has shown to be able to detect and segment fire and smoke zones in an accurate and precise way only using weakly-supervised methods.

In particular, using the CAM method, it was proven that it is possible to train a classification model only with image-level labels and by extracting the features that the model uses for the classification prediction, one can construct a slightly rough heatmap highlighting the fire/smoke zones. Subsequently, by using an energy minimization algorithm, the CRF, it was possible to transform the rough heatmaps previously obtained into a segmentation mask was much detail. The CRF merges the binary mask obtained from the heatmap with the pixel information on the original image to create a considerably accurate segmentation mask of fire/smoke.

The proposed method makes use of the powerful capabilities of CNN on image classification and their ability to model patterns by finding representative class features. Ailed to this, the method also uses the base of classic methods for fire/smoke detection which is the very characteristic colour pattern.

The process of classifying and segmenting fire and smoke zones is already a great challenge using fully-supervised methods based on deep learning since the shape that fire and smoke zones take can be very irregular and sometimes very dim. When using weakly-supervised this challenge is even greater because it is then up to the network to understand which are the class-specific features of fire and smoke in the entire image. So, it was necessary to create a good and complex dataset with several different examples. It was important to have several examples of fire and smoke individually in order to not correlate both of them. Also, it was important to have various examples of negative images where none of them appears to distinguish common co-occurring zones, for example, forests.

The overall results show that when taking into account the heavy needs of a fully-supervised method, the proposed weakly-supervised system can strongly compete with them in terms of segmentation performance. For smoke, the proposed methods even achieve identical performance.

Some limitations were noted using these methods. First, it was noticed that as the model is performing classification in the whole image and the input image size must be small for computational reasons, images with very small zones of fire/smoke could not be detected. As a suggestion, the use of methods with a sliding window could be beneficial. Second, the several parameters in the post-processing stage are static and were tuned in a more generalized way resulting in some undesired situations. In future works, the tuning process may be done using a learning process, similar to the classification model, resulting in dynamic parameters that can adapt to each image. Third, the system presents some small oscillations in the performance results. In future studies, we suggest the use of semi-supervised methods where it can be combined both fully and weakly supervised methods. This way one could use the few datasets available annotated at the pixel level with the ease of gathering images to annotate at the image-level. By combining both approaches it could be possible to develop a more robust and very accurate method for the fire and smoke segmentation.

With this work, we hope to represent a great contribution to the Firefront project and help it to support the brave firefight teams. We also hope that it will serve as motivation for future works in this area since the problem of wildfires is still a catastrophe that seriously affects human beings and our planet.

Acknowledgment

This work was made in the context of the FCT project FIREFRONT (PCIF/SSI/0096/2017).

References

[1] B. Zhou, A. Khosla, A. Lapedriza, A. Oliva, and A. Torralba, "Learning Deep Features for Discriminative Localization," *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, vol. 2016-Decem, pp. 2921–2929, 2016.

[2] X. Zhang, Y. Wei, J. Feng, Y. Yang, and T. Huang, "Adversarial Complementary Learning for Weakly Supervised Object Localization," *CoRR*, vol. abs/1804.06962, pp. 1325–1334, 2018.

[3] K. K. Singh and Y. J. Lee, "Hide-and-seek: Forcing a network to be meticulous for weakly-supervised object and action localization.," *2017 IEEE international conference on computer vision (ICCV)*, pp. 3544–3553, 2017.

[4] R. R. Selvaraju, M. Cogswell, A. Das, R. Vedantam, D. Parikh, and D. Batra, "Grad-CAM: Visual Explanations from Deep Networks via Gradient-Based Localization," *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, vol. 128, no. 2, pp. 618–626, 2017.

[5] P. Barmpoutis, P. Papaioannou, K. Dimitropoulos, and N. Grammalidis, "A review on early forest fire detection systems using optical remote sensing," *Sensors (Switzerland)*, vol. 20, no. 22, pp. 1–26, 2020.

[6] K. Muhammad, J. Ahmad, I. Mehmood, S. Rho, and S. W. Baik, "Convolutional Neural Networks Based Fire Detection in Surveillance Videos," *IEEE Access*, vol. 6, no. c, pp. 18174–18183, 2018.

[7] 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.

[8] T. H. Chen, P. H. Wu, and Y. C. Chiou, "An early fire-detection method based on image processing," *Proceedings - International Conference on Image Processing, ICIP*, vol. 3, pp. 1707–1710, 2004.

[9] Y. Zhao, J. Ma, X. Li, and J. Zhang, "Saliency detection and deep learning-based wildfire identification in uav imagery," *Sensors (Switzerland)*, vol. 18, no. 3, 2018.

[10] B. U. To¨reyin, D. Yigˇithan, G. Ugˇur, and A. E. C. Etin, "Computer vision based method for real-time fire and flame detection," *Pattern recognition letters*, vol. 27, pp. 49–58, 2006.

[11] T. Celik, "Fast and efficient method for fire detection using image processing," *ETRI Journal*, vol. 32, no. 6, pp. 881–890, 2010.

[12] H. Demirel and T. C-elik, "Fire detection in video sequences using a generic color model," *Fire safety journal*, vol. 44, pp. 147–158, 2009.

[13] D.-c. Wang, X. Cui, E. Park, C. Jin, and H. Kim, "Adaptive flame detection using randomness testing and robust features," *Fire Safety Journal*, vol. 55, pp. 116–125, 2013.

[14] D. Y. T. Chino, L. P. S. Avalhais, J. F. R. Jr, A. J. M. Traina, and S. Carlos, "BoWFire : Detection of Fire in Still Images by Integrating Pixel Color and Texture Analysis," *2015 28th SIBGRAPI conference on graphics, patterns and images*, 2015.

[15] M. Batista, B. Oliveira, P. Chaves, J. C. Ferreira, and T. Brandão, "Improved Real-time Wildfire Detection using a Surveillance System," *Proceedings of the World Congress on Engineering 2019 WCE 2019, Lecture Notes in Engineering and Computer Science*, vol. 0958, no. July, 2019.

[16] T. Toulouse, L. Rossi, M. Akhloufi, T. Celik, and X. Maldague, "Benchmarking of wildland fire colour segmentation algorithms," *IET Image Processing*, vol. 9, no. 12, pp. 1064–1072, 2015.

[17] T. Toulouse, L. Rossi, T. Celik, and M. Akhloufi, "Automatic fire pixel detection using image processing : a comparative analysis of rule-based and machine learning-based methods," *Signal, Image and Video Processing*, 2015.

[18] K. Muhammad, J. Ahmad, S. Member, Z. Lv, and P. Bellavista, "Efficient Deep CNN-Based Fire Detection and Localization in Video Surveillance Applications," *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, 2019.

[19] Q.-x. Zhang, G. Xu, J.-j. Wang, Q.-x. Zhang, G. Xu, J.-j. Wang, Q.-x. Zhang, G.-h. Lin, G. Xu, J.-j. Wang, and J.-j. Wang, "Wildland Smoke Detection Based on Protection Faster R-CNN using Synthetic Smoke Images," *Procedia Engineering*, vol. 211, pp. 441–446, 2017.

[20] Q. Zhang, J. Xu, L. Xu, and H. Guo, "Deep Convolutional Neural Networks for Forest Fire Detection," *Proceedings of the 2016 international forum on management, education and information technology application*, no. Ifmeita, pp. 568–575, 2016.

[21] S. Frizzi and R. Kaabi, "Convolutional Neural Network for Video Fire and Smoke Detection," *IECON 2016-42nd Annual Conference of the IEEE Industrial Electronics Society*, pp. 877–882, 2016.

[22] K. Muhammad, S. Khan, M. Elhoseny, S. H. Ahmed, and S. W. Baik, "Efficient Fire Detection for Uncertain Surveillance Environment," *IEEE Transactions on Industrial Informatics*, vol. PP, no. c, p. 1, 2019.

[23] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," *arXiv preprint arXiv:1409.1556*, 2014.

[24] P. Krähenbühl and V. Koltun, "Efficient inference in fully connected crfs with Gaussian edge potentials," *Advances in Neural Information Processing Systems 24: 25th Annual Conference on Neural Information Processing Systems 2011, NIPS 2011*, pp. 1–9, 2011.

[25] L. C. Chen, G. Papandreou, I. Kokkinos, K. Murphy, and A. L. Yuille, "DeepLab: Semantic Image Segmentation with Deep Convolutional Nets, Atrous Convolution, and Fully Connected CRFs," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 40, no. 4, pp. 834–848, 2018.

[26] T. Toulouse, L. Rossi, A. Campana, T. Celik, and M. A. Akhlou, "Computer vision for wild fire research : An evolving image dataset for processing and analysis," *Fire Safety Journal*, vol. 92, pp. 188–194, 2017.

[27] D. Krstinic and T. Jakovcevic, "Image database," Feb 2010.

[28] Q. Zhang, "Research webpage about smoke detection for fire alarm: Datasets."

[29] W. Bae, J. Noh, and G. Kim, "Rethinking class activation mapping for weakly supervised object localization," in *European Conference on Computer Vision*, pp. 618–634, 2020.

[30] G. Perrolas, A. Bernardino, and R. Ribeiro, "Fire and Smoke Detection using CNNs trained with Fully Supervised methods and Search by Quad-Tree," *Proceedings of RECPAD 2020*, pp. 59–60, 2020.

[31] F. N. Iandola, S. Han, M. W. Moskewicz, K. Ashraf, W. J. Dally, and K. Keutzer, "SqueezeNet: AlexNet-level accuracy with 50x fewer parameters and <0.5MB model size," pp. 1–13, 2016.

[32] W. Weng and X. Zhu, "U-Net: Convolutional Networks for Biomedical Image Segmentation," *IEEE Access*, vol. 9, pp. 16591–16603, 2021.

[33] L. C. Chen, Y. Zhu, G. Papandreou, F. Schroff, and H. Adam, "Encoder-decoder with atrous separable convolution for semantic image segmentation," *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, vol. 11211 LNCS, pp. 833–851, 2018.