Fire and Smoke Detection using Fully Supervised Training Methods and Search by Quad-Tree

Gonçalo de Carvalho Rodrigues Callet Perrolas  
Instituto Superior Técnico  
Lisbon, Portugal, February 2021  
goncalo.perrolas@tecnico.ulisboa.pt

Abstract—Wildfires are one of the most destructive and uncontrollable natural disasters faced by man-kind. The fire-fighting combat teams can greatly benefit from a reliable source of information about the different aspects within a fire scene. By locating, in real time, the current position of the fire fronts, a better fighting strategy can be developed to help the efficiency of the operations. It is also critical, the detection of the early fires ignitions to take action as soon as possible. This work proposes an original deep learning method for fire and smoke detection using aerial images of wildfires. We trained both a full image classifier and a segmentation network in order to detect the presence of fire/smoke and to localize the regions of the images containing said phenomenon. Along side the detection component, we used an innovative Quad-Tree algorithm to increase the precision of the detections, processing the images into smaller patches. The proposed system was able to produce segmentations with a high level of precision and detail, achieving an Avg. IoU of 0.88 for fire and 0.83 for the smoke class on a test aerial image dataset. The system proved to be highly capable of being used on a real fire scenario.

Index Terms—Fire Detection, Smoke Detection, Aerial Images, Wildfire, Convolutional Neural Networks.

I. INTRODUCTION

This work is related to the Firefront project. Its main objective is to create a support system to the fire combat teams. The system will transmit valuable information about the wildfires in real-time to the ground teams using aerial vehicles to capture images of the scene. Next, the images are processed, using a convolutional neural network able to segment the respective areas.

It comes as a challenge the detection of variable size portions of both fire and smoke, and due to the small input size of the first layer of most common CNNs, an extra algorithm was used for doing the dynamic detection on smaller image sections, using a Quad-Tree search method.

Firstly, the images are taken from drones or other aerial vehicles equipped with RGB cameras and a transmission system able to send those images. Then, image processing techniques involving deep neural networks are used in order to detect fire and smoke on those images. It is important to choose an adequate network for the proposed objective, leading to the best results possible and capable of dealing with real life fire images taken from drones. We aim at semantic segmentation of the fire and smoke classes within the images taken.

II. STATE OF THE ART

Over the last two decades, automatic methods of fire fighting technologies have shown an increasing popularity as a research topic. The different approaches and methods used in fire and smoke detection can be divided into two different groups: methods that make use of classic image processing techniques where images are analyzed regarding textures, contrasts or RGB components, and the second group involves all the techniques that resort to neural networks.

Several efforts have been made in the development on the topic of using UAV with vision-based systems to monitor, detect, and fight forest fires. Both classic methods of image processing or deep learning have been used to extract the relevant features that define a fire or a smoke area.

One of the challenges in fire and smoke detection from airborne image is the multi-scale problem that occurs while using neural networks to detect objects with a wide size variance. This is an interesting topic that is mostly absent from related literature and will be discussed more in depth throughout this work. This will also act as a motivation pushing towards the creation of a method that tries to solve this challenge.

A. Classic Methods

The vast majority of classic methods are based on algorithms that make use of the three RGB components of each pixel because the color histogram for fire regions is distinguishable from the background scenario. However, the same cannot be said about the smoke due to high similarities with other objects like clouds.

Different indexes can be calculated according to the RGB values of the pixels in order to enhance the areas of fire and classify the images into the respective classes. There are a lot of different algorithms for fire segmentation using different color spaces. However, a benchmark of their performance shows that they tend to be very biased in regard of the camera characteristics, the environment conditions and other external factors. The that highly fluctuate depending on the datasets used for testing.

We find that the large majority of fire/smoke detection systems that are operating rely on image captured by fixed cameras on high altitude in order to cover large areas of forest terrain. One of the most used techniques in static camera systems is Background Subtraction, where the elements that do
not belong to the background are segmented from the images. One example operating in Portugal is the project CICLOPE [6], that monitors 1,300,000 hectares of forest and detects the occurrence of fire through smoke analysis on the images captured. Similarly, smoke detection can be done through its color, its motion and time domain changes. [7]. These techniques can be applied whenever the cameras are fixed, which is not our case.

In summary, classical methods tend to be very efficient. However, the problem with classical methods is that they rely on hand-crafted features that requires a higher level of expertise and a lot of fine-tuning for specific problems. Changing the environment and the cameras parameters will induce a lot of false detections on the system.

Very recently, the appearance of deep learning has completely revolutionized many areas related with image processing and remote sensing, by using neural networks more specifically, CNN [8].

B. Deep Learning Methods

Techniques involving the use of neural networks have many advantages, the main one being that it allows the collection of characteristics of an image in a more abstract and complex way, being able to evaluate which are the characteristics that define a flame or smoke. This strategy requires a mandatory learning stage, where the network is trained for the task in question. It is also important to mention that there are different types of training, depending on the level of the dataset used. Labels can have a pixel wise identification for each different class (fully supervised) or a more general image level label with the image specific type of class (weakly supervised).

As stated earlier, the most commonly used network for this type of tasks is the CNN. We can find examples of CNN in fire and smoke detection with weakly supervised training [9]. The network created in that work was over-fitted to the data used in training. This is an issue caused by one of the most relevant obstacles in this research topic, the lack of datasets containing fire and smoke images with the respective labelling.

Other strategies can be used in collaboration with CNNs. One example is trying to use the moving dynamic of fire in time domain to extract more features from the fire [10]. The author compared the results using the AlexNet [11], VGG16 [12] and SqueezeNet [13], and they all showed to be able to correctly detect fire. Using the temporal information did not improve the performance significantly and the heat-maps produced while trying to localize the fire regions seem to be very vague and not well defined. The classification networks were trained with a large dataset of image level labels (160K imgs) and they achieved a high level of detection (classification) performance.

There are several CNN extensions, e.g. the R-CNN [14] that can be used in the detection task as it brings computational advantages on searching for fire or smoke instances [15]. This technique may only detect one instance of fire/smoke per image, making it not suitable for detecting multiple overlapping areas on the image. The dataset used by this author was quit complete, accurately labeled and it shown potential to be used in this thesis work.

These machine learning techniques are able to surpass the limitations of the traditional methods, described earlier. Therefore we consider the use of deep learning to be appropriate for the detection task in hand.

III. METHODOLOGY

The proposed system will process images taken from an aerial point of view, either taken from conventional aerial vehicle like an airplane or helicopter or an unmanned vehicle like a drone. Those vehicles must be equipped with a RGB camera with enough resolution to be able to detect small fire/smoke areas and a transmission system to send the images to a processing unit on land. The aerial pictures will then go through our system in order to detect the areas of fire and smoke within them.

The overall structure of the system is represented in figure 1. The first section of the system is the QuadTree block that dynamically analyse the image, dividing it into patches depending on the results of the final segmentation stage. The main purpose of this algorithm is to solve the multi-scale problem of the detection and, depending on the size of the fire/smoke area on the image, the algorithm will tend to either do a more precise detection, involving smaller size patches, or a more global search using the all image. The QuadTree block will output the complete input image or a portion of it.

The output of the QuadTree algorithm will then move along to the detection stage that includes the sequence of a classification network followed by a segmentation network. It is important to state that the detection system for fire and smoke are independent. This means there will be two different pairs of classification and segmentation networks, each one more suitable for the respective class to be detected.

The portions of the images are initially fed as input to the classification network (SqueezeNet) in order to detect the presence of the phenomenon (either fire or smoke) and, if the output of the classification is positive, that image patch goes through to the segmentation network (U-Net) in order to detect the regions of the image containing said class, represented by a binary image.

The binary result will then be used in the QuadTree to evaluate if it is necessary to do a more precise detection on the current analysed patch. If this is not necessary then the process is ended and the final result is reached.

A. Search by Quad-Tree

Quad-Tree methods [1] are often used to partition a two-dimensional space by recursively subdividing it into four quadrants or regions. This is useful for doing a dynamic search for fire or smoke occurrences on the images, starting first with larger scale patches and, if nothing is detected, then moving on to smaller patches by slicing the previous ones into four segments. By using this method we solve the issues related to the multi-scale nature of fire/smoke areas. Small fire/smoke regions in large patches are difficult to be detected by the
neural networks and patches that are almost filled completely with fire or smoke can have dubious detection results due to the lack of more external features that define the phenomenon.

All the logic of the system flow is controlled within this functional block explained in Algorithm 1.

**Algorithm 1: Quad_Tree Algorithm**

```plaintext
Function Quad_Tree (Input_Patch):
    positive = Classify(input_patch);
    if positive then
        input_patch_seg = Segment (input_patch);
        if Ratio(input_patch_seg) < max_ratio AND Size(input_patch) > min_size then
            patches = Split_in_4(input_patch);
            for patch in patches do
                result_i = Quad_Tree(patch_i);
                GluePatches (results, final_result);
        else
            if positive then
                final_result = input_patch_seg;
            else
                final_result = blank_patch;
    return final_result
```

Inside the algorithm there are calls to the same function (recurrence) to be able to process every patch desired and to produce an end result as an aggregate of said patches. The function has an RGB image as an input and will output the gray-scale result of the detections.

There are some relevant parameters that control the behavior of the QuadTree algorithm:

- **Ratio**: Represents the ratio between the number of pixels belonging to positive and negative cases. If a large area of fire is segmented then, the value of that ratio will be bigger than for a smaller fire. The Max_Ratio parameter is used to control how big a fire or smoke must be, in a certain patch, to stop the ”zooming” process.

- **Min_Size**: Expresses the minimum size of patch the algorithm can reach. This parameter can be helpful to adjust the system to the camera equipped on the vehicle, if the camera has a big resolution sensor even smaller patches can have a lot of information.

**B. Patch Processing (Classification + Segmentation)**

The detection system is composed of two different neural networks, one for classification and other for segmentation. The classification network chosen was a SqueezeNet model [13] due to its level of accuracy compared to state-of-art models like Alex-Net [11] while having a lot fewer parameters making the model more suitable for smaller datasets. For segmentation we used U-Net [16], one of the most used state-of-the-art models for semantic segmentation.

By using a classification stage before doing segmentation the overall performance is increased due to the fact that it is easier to create a more complete dataset with image level labels then with pixel level. This also helps reducing the number of false detections on the segmentation.

In Figure 2 are exposed some examples of outputs from the SqueezeNet from a different set of positive and negative images. The output consists of a probabilistic array containing each class probability. ([P(fire), P(negative)]).

The overall logic of the system can be explained as following: each patch of the image produced by the Quad-Tree stage is given as input to the SqueezetNet [13] to determine if the patch contains fire or smoke. If a phenomenon is detected then the patch moves along to the segmentation stage where the areas of fire/smoke are segmented by the U-Net [16]. If nothing is detected in the classification stage the patch is not segmented and the process moves to the next patch in the sequence.

Next, in Figure 3, are exposed some examples of outputs from the U-Net for both classes.

**C. Datasets**

In order to train and evaluate the chosen networks there is a need to gather a set of images containing the adequate labelling for the respective network.

For the classification networks the labelling needed is on image level, indicating if it contains the phenomenon or not. For the segmentation networks, the labels have to specify...
each pixel class. In total, four different types of datasets were gathered for the task of fire/smoke classification/segmentation.

The images gathered for the fire segmentation dataset mainly came from three different sources: Corsican Dataset [17] (RGB images with pixel wise labelling), smaller datasets found online and a batch of images gathered online that were manually labelled to extend as much as possible the size of the dataset. The smoke segmentation dataset consists also of datasets with pixel wise labels found online [18] [19] and some more images segmented manually.

The nomenclature negative identifies that the image does not include the respective phenomenon. The hardest negatives for the network to distinguish in case of fire are images containing sunsets/sunrises, reddish color sky, red foliage and red objects that can appear in fire fighting situations (like fire trucks, airplanes and helicopters). For smoke, the main negative object that is extremely hard to distinguish are clouds. So, in order to properly train the network against these cases it is important to include in the datasets images containing difficult negative cases.

The datasets for classification include the ones used for the segmentation training together with some additional images. In Table 4 we present an overview of the datasets used for the training phase where is identified the corresponding number of images.

Table 4: Overivew of the Dataset

| Classification | Fire | Positive | 800 images |
|                | Negative | 520 images |
| Smoke          | Positive | 500 images |
|                | Negative | 300 images |

The number of images gathered in the datasets goes to show how hard it was to find complete and adequate datasets containing pixel segmentation. Doing manual segmentation is a very demanding task but it helps to improve the detection performance of the networks. The labelling was done using the Image Labeler App included in the Computer Vision System Toolbox 8.0 from Matlab [20]. We focused our efforts in gathering images that add diversity of situations that were not covered in the Corsican Dataset, mainly cases where the images are taken from long distances, from an aerial perspective and with small areas of fire/smoke.

Each dataset was randomly split into three separated sub-sets, one for training, one for validation and the last one for testing purposes (divided in the following proportions: 70%, 20% and 10% ).

IV. TOOLS USED

The code was written and compiled inside the Google Colab virtualization environment using Python 3.6.9, running on a single core hyper threaded Xeon Processor at @2.3Ghz and
a Tesla K80 (GPU). The libraries used for neural network development were Tensorflow 2.3.0 [21] and Keras 2.4.0. [22].

V. NETWORK TRAINING

As explained earlier, the system has in total four different network instances, each one having a different task. We must train two instances of the SqueezeNet and two instances of the U-Net. On the training stage it was used the training and validation sub-sets of each dataset.

A. SqueezeNet (Classification)

We first train the networks for classifying each patch for fire or smoke using the classification dataset. We set the training parameters as follow:

**Fire Train Parameters:**
- **Optimizer:** Adam()
- **Learn. Rate:** 0.001
- **β₁:** 0.9;
- **β₂:** 0.999;
- **є:** 1e-7;
- **Loss:** Binary Cross-entropy;
- **Batch Size:** 32;
- **Patience:** 20;
- **Epochs:** 150;
- **Monitor:** Val. Loss;

**Smoke Train Parameters:**
- **Optimizer:** Adam()
- **Learn. Rate:** 0.001
- **β₁:** 0.9;
- **β₂:** 0.999;
- **є:** 1e-7;
- **Loss:** Binary Cross-entropy;
- **Batch Size:** 32;
- **Patience:** 20;
- **Epochs:** 150;
- **Monitor:** Val. Loss;

The parameters chosen were the ones that lead to the best validation losses while not adding too much processing time. The final training for fire took approximately two hours and a half and for smoke took two hours.

The results are exposed on Table 5, where the accuracy and loss values are shown, corresponding to the lowest validation loss point.

<table>
<thead>
<tr>
<th>Training Set</th>
<th>Accuracy</th>
<th>Loss</th>
</tr>
</thead>
<tbody>
<tr>
<td>Fire</td>
<td>0.9856</td>
<td>0.9610</td>
</tr>
<tr>
<td>Smoke</td>
<td>0.9511</td>
<td>0.1004</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Validation Set</th>
<th>Accuracy</th>
<th>Loss</th>
</tr>
</thead>
<tbody>
<tr>
<td>Fire</td>
<td>0.9598</td>
<td>0.9101</td>
</tr>
<tr>
<td>Smoke</td>
<td>0.1436</td>
<td>0.3047</td>
</tr>
</tbody>
</table>

The models have an overall good performance, although the smoke detection performance is a bit worse than the fire one. As expected, it is harder for the network to correctly train with a smaller dataset and the smoke phenomenon has more intricacies that make its detection quite more difficult than fire.

B. U-Net (Segmentation)

Similarly to the training stage of the two instances of the SqueezeNet, we now expose the training of the two U-Net’s. The parameters used in the training phase are as follow:

**Fire Train Parameters:**
- **Optimizer:** Adam()
- **Learn. Rate:** 0.001
- **β₁:** 0.9;
- **β₂:** 0.999;
- **є:** 1e-7;
- **Loss:** Binary Cross-entropy;
- **Batch Size:** 32;
- **Patience:** 30;
- **Epochs:** 200;
- **Monitor:** Val. Loss;

**Smoke Train Parameters:**
- **Optimizer:** Adam()
- **Learn. Rate:** 0.001
- **β₁:** 0.9;
- **β₂:** 0.999;
- **є:** 1e-7;
- **Loss:** Binary Cross-entropy;
- **Batch Size:** 32;
- **Patience:** 50;
- **Epochs:** 200;
- **Monitor:** Val. Loss;

The parameters used followed the same logic as before, trying to achieve the lowest value of loss for the validation dataset. The Adam optimizer proved again to be very capable of producing good results.

In Table 6 are shown the accuracy and loss values for the training point of lowest validation loss. Similarly as the classification stage, we again conclude that the smoke class has worst detection performance from the network due to same causes stated earlier.

<table>
<thead>
<tr>
<th>Training Set</th>
<th>U-Net Fire</th>
<th>U-Net Smoke</th>
</tr>
</thead>
<tbody>
<tr>
<td>Accuracy</td>
<td>0.9241</td>
<td>0.8415</td>
</tr>
<tr>
<td>Loss</td>
<td>0.0267</td>
<td>0.1829</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Validation Set</th>
<th>Accuracy</th>
<th>Loss</th>
</tr>
</thead>
<tbody>
<tr>
<td>Fire</td>
<td>0.9091</td>
<td>0.7428</td>
</tr>
<tr>
<td>Smoke</td>
<td>0.0682</td>
<td>0.3667</td>
</tr>
</tbody>
</table>

Overall, the training results for the smoke class are considerably worse both in terms of accuracy and loss. On the U-Net dedicated to smoke there is a significant drop on pixel accuracy comparing the training set and the validation set results, contrary to the fire class, that proved to be more consistent.

VI. RESULTS

A. Overall System Performance Test

This test was used to evaluate the overall performance of the proposed system, and to see how it behaves with some of its components disabled. We make use of the datasets reserved for testing proposes. The evaluation of each class is made separately and, for each, we tested four different models:

1. **Q+C+S** : This model corresponds to the complete system created in this thesis work. It uses all the components explained earlier, the QuadTree algorithm, the classification stage and finally the segmentation stage;
2. **Q+S** : This model is very similar to the first one, except that the component of classification is removed, the QuadTree assumes the classification always positive;
3. **C+S** : On the next two models, the QuadTree methodology is removed and the input of the networks consists simply of the original images, there is no patch
processing. This model includes the classification and segmentation stage:

- **4, S**: This final model consists only of the segmentation stage.

To evaluate the performance of the four different system models we use the Average IoU metric together with its SD. We also analysed the pixel accuracy and the average processing time per image.

Starting with the fire class, the results are shown on Table 8.

The complete systems showed a better performance while compared to the other models. Using the QuadTree methods, we achieved a considerable improvement from the model without it (improvement from 0.849 to 0.883 Avg. IoU). The Avg. IoU set of values were also less disperse while using the complete systems as the SD(Avg. IoU) values are smaller. The results also show that, while using the QuadTree method, it really is beneficial to have the classification stage. If the QuadTree method is not used, the classification stage seems to not improve the performance, leading to belief that the segmentation is well capable by itself while using this dataset.

Regarding the processing time of each model, we noticed that the complete system does not increase significantly the processing time of each image. While compared with the simplest model 4, it took approximately only ten times more.

From what we have seen so far, for the fire class, the complete system shows evidence of being a suitable and efficient solution for the multi-scale problem.

Next, the same evaluations are made, but this time for the smoke class, starting with the comparison of the four different models performance in terms of Avg. IoU, pixel accuracy and processing time. This comparison is shown in Table 9.

The complete systems showed a better performance while compared to the other models. Using the QuadTree methods, we achieved a considerable improvement from the model without it (improvement from 0.849 to 0.883 Avg. IoU). The Avg. IoU set of values were also less disperse while using the complete systems as the SD(Avg. IoU) values are smaller. The results also show that, while using the QuadTree method, it really is beneficial to have the classification stage. If the QuadTree method is not used, the classification stage seems to not improve the performance, leading to belief that the segmentation is well capable by itself while using this dataset.

In general, the four model performances behaved very similarly to the fire class. Again, the complete system has the best detection performance. Although the standard deviation is quite similar to the other models, this time there was a more significant improvement on pixel accuracy. The processing time increase was very similar to the fire class and, once again, the worst model was the Q + S model, reinforcing the assumption that the QuadTree methodology does in fact require the classification of each patch in order to become an
advantageous model.

After this experiment, once again, we proved that the smoke detection task is a lot more challenging. The results for the fire class are generally more consistent and the resulting segmentations are more in line with the ground-truth masks of the dataset.

B. Tests On Aerial Images

This experiment has the purpose of testing the system with images that better represent the real life scenarios in which the system will be used. We gathered a set of images taken from a aerial point of view from a considerable distance, simulating the perspective of the vehicle that would take the wildfire images. Some of the images belong to a dataset specially collected for works related with the Firefront Project and others were collected from the web.

On Figure 7, we shown two results from the Firefront dataset in which the fire areas are very small (to better visualise the results, we added a zoom crop in both cases).

These types of images, where the aerial vehicle is quite far from the fire, represent a challenge for networks with a small input layer. For that reason, we developed the QuadTree method. The method was shown to be very capable and the resulting segmentations display a high level of detail and precision.

The same level of performance can be seen in Figure 10, where we test more images taken from a long distance to the wildfire.

Next, we make similar tests for the smoke class. In Figure 11 we tested one of the images from the Firefront dataset to see how the smoke detection behaves.

Looking at the figure, we can see that the resulting segmentation is not as precise as we have seen for the fire class. The boundaries that limit the smoke area are not as well defined and regions that appear to have similarities with smoke are sometimes incorrectly detected as being smoke (in the figure, the smoke extends along one of the dirt roads).

Similarly, on Figure 12, we can see the same type of behavior.

The smoke region is correctly detected and the background clouds are avoided although there are some false positives near the lake and the left region of the image. In spite of occasional poor detections, in general, the system dedicated for smoke detection is able to segment with a relatively good level of accuracy. At more extent, we showed on Figure 13, examples of well performed detections.

The results achieved in this experiment, support the concluding statements made throughout this work. The system dedicated for the fire class has better performance overall and the resulting detections are more in-line with the original image labels. The boundaries created seem to be correct and very precise.

The system developed for smoke detection shows a performance decrease while compared to the fire one but, in general, the system is well capable of detecting even small smoke regions. The detections produced tend to be less precise and the limits are more vague and ambiguous. As stated earlier, smoke detection is a lot more challenging and, consequently, this is the component that could be more significantly improved in future works.

VII. CONCLUSION

As we reach the end of this thesis work, some conclusions can be drawn about the overall experience, results and possible future work.

The system developed shows some innovative characteristics that distinguish him from other state-of-the-art methods
Subjective. Two different subjects are going to label the images the performance of the networks used. This task is hugely time consuming but they usually pay off, as we increased to our dataset. This type of tasks are really demanding and stages, we decide to manually create labeled images to add very challenging factor. Specially on the smoke detection task, as this shown to be a have really helped to improve the results achieved in this work, research area. Working with bigger and better datasets could and fully labelled database of fire and smoke images that challenging. If there was a centralized, open-source, complete both the fire and smoke classes which made this work more extensive as possible. From the research it was made, there is a lack of datasets with fully segmented labelled images for both the fire and smoke classes which made this work more challenging. If there was a centralized, open-source, complete and fully labelled database of fire and smoke images that would have been a really significant improvement step in this research area. Working with bigger and better datasets could have really helped to improve the results achieved in this work, specially on the smoke detection task, as this shown to be a very challenging factor.

Due to the small size of the dataset gathered at early stages, we decide to manually create labeled images to add to our dataset. This type of tasks are really demanding and time consuming but they usually pay off, as we increased the performance of the networks used. This task is hugely subjective. Two different subjects are going to label the images differently, specially the smoke class. This fact can be an interesting topic of study, to find how much a subjective labelled dataset can influence a detection problem and to quantify the differences of the labelled images among different people.

In this work, only RGB images are used to do the fire detection. In future studies it would be interesting to use other different types of cameras (for example thermal and infra-red cameras) while trying to fuse the different spectral image information to see if it improves the performance of the detection. Using other spectres can be a demanding task, due to the complete lack of datasets of non-RGB images of wildfires. By only using RGB images, our system is limited by the atmospheric conditions. If smoke or clouds are covering the wildfire areas beneath the vehicle, no fire will be detected. Solving these challenges could represent a future progress step of this work.

Also as progress to this work, after doing the fire detection, it would be interesting to have a component that would outline the fire front in shape of elongated lines. This would represent a connection link between the fire detection task and other project components, for example the geo-referencing of the fire or components related with the fire behaviour prediction.

Finally, looking at the results produced by the developed system, we can conclude with high level of confidence that the system is capable of achieving the proposed objectives in which this thesis work was based on. From the tests done on real images taken from an aerial point of view, the systems shown to be capable of detecting both fire and smoke with a relatively low time and computational cost, making it a truly viable option of an implementation that could help the brave fire fighting teams.

The project developed in this thesis was also published on a Portuguese scientific journal [23] related with pattern recognition, proving the merit and the quality of the work produced. The publication of the article and the achieved results help to confirm the true potential of the proposed methods.

We truly hope that the work produced in this thesis project represent a powerful addition in the Firefront Project and that it could help to motivate other individuals to study this interesting topic.

**References**


