Fire and Smoke recognition in crowdsourced images with YOLO networks

Ana Madeira
abreumadeira.ana@gmail.com
Catarina Silva
catarina@dei.uc.pt
Alberto Cardoso
alberto@dei.uc.pt
Bernardete Ribeiro
bribeiro@dei.uc.pt

University of Coimbra
CISUC - Centro de Informática e Sistemas
FCTUC-DEI - Departamento de Engenharia Informática
Coimbra, Portugal

Abstract

The early detection of a fire can largely mitigate harmful consequences. With the improvement in image quality, it is now possible to develop intelligent systems for visually detecting forest fires. An intelligent system for fire detection was implemented based on deep learning techniques for image object detection. As part of the fire detection approach development, different datasets are proposed to train and evaluate the YOLO models, specific to the fire and smoke recognition problem. The proposed Fire/Smoke annotated datasets can be used in future smoke, and fire detection research. Results show that a YOLOv4 one-stage detector can be used for image fire and smoke detection tasks, trained using manually annotated datasets and applied to a real application using crowdsourced data.

1 Introduction

As a way for people to report fires detected using their smartphones, the FireLoc project [1] of the Foundation of Science and Technology (FCT) [2] is set up as an alternative way of reporting fires. This project is based on voluntary contributions and aims to develop a system in which, through a smartphone application, users can send photos of fire taken with their smartphone camera. If present, smoke and fire are recognized in the images submitted, and the forest fire can be located on a map. This information is then sent to a server. The developed system will correspond to the submission validation module and validate whether there is fire or smoke in each contribution.

In this paper, the main focus is the development of an intelligent system for fire and smoke detection. integrated in the FireLoc application, using the proposed post-processing steps to obtain the image classification results, identifying whether user submissions are valid, i.e., whether they contain smoke or fire.

2 Related work

Recent studies show advantages in considering the localization as well as the classification of existing objects in an image as part of object detection problems [7]. The models used for object detection can be divided into two categories: Two-stage detection frameworks and One-stage (Unified) detection frameworks. In the first, the process is divided into two phases. First, there is a proposal for candidate regions of the image that may contain objects to be detected [6]. The classification is then made based on the first result, fine-tuning the regions, discarding false positives (for example, Faster R-CNN). One-stage detection frameworks perform the process at once, without the initial region proposal step and therefore allow a single model to be used, predicting the bounding boxes that contain the objects present, as well as the probabilities of these belonging to the classes considered [5] (for example the YOLO models). The YOLOv4 models analyze the image's features using different resolutions, maintaining the original image's height/width ratio. These models manage, by adapting the size of the initial anchors to the specific dataset, to detect objects of various scales in the images [6] and allow the correct detection of overlapping objects of different classes [3]. For this reason, they present advantages when used for this problem, since it is common to have the presence of smoke in the images where there is fire. However, these models have some disadvantages, such as the need to have a considerable amount of annotated images to obtain good results. As such, to solve the lack of annotated data, two datasets are proposed with the annotation of Fire and Smoke class objects for training and model evaluation. These datasets were used to optimize the results in the context of forest fires. In addition, the transfer learning technique was also used, with a pre-trained model with the Imagenet dataset. The initial weights resulting from the pre-training were kept, responsible for the extraction of more low-level features. The last layers of the model, responsible for the extraction of features specific to the problem, were retrained [6]. The use of transfer learning makes the training process less time-consuming and improves the model's learning capacity.

[1] Project PCIF / MPG / 0128/2017, FireLoc - Where's the Fire? - Identification, positioning, and monitoring forest fires with crowdsourced data

3 Proposed approach

For the development of the fire and smoke detection system, an object detection approach was adopted, using YOLOv4 [3] models. These models detect the specific location of fire and smoke in the images. Therefore, they require an indication of where the objects are present within the image, using bounding boxes. To perform the manual annotation of the training and testing datasets proposed, according to the YOLO annotation format, the tool LabelImg was used.

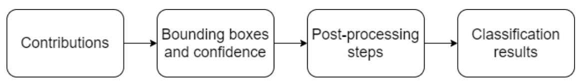

Figure 1: Detection system steps

Figure 1 shows the sequence of steps necessary to detect fire and smoke in the images submitted by the application users. The model first identifies the parts of the images that contain fire or smoke with bounding boxes, and the corresponding confidence score associated with each detected object. The classification results are then obtained with the post-processing step, which allows integration with the FireLoc system. For the classification results, the **Fire**, **Smoke**, and **Neutral** classes are considered. The images in which fire is detected belong to class **Fire**, and the images in which smoke is detected, and no fire is detected belong to class **Smoke**. The remaining images, in which no object is detected, belong to the Neutral class.

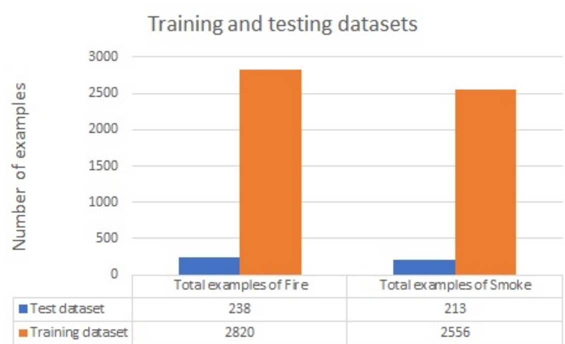

Figure 2: Number of object examples in each dataset

For the training of these models, an open-source dataset from [1] was used. This dataset contains an equal amount of images from each class: **Smoke**, **Fire**, and **Neutral**). It contains 1000 images of each category and is named *Fire-Smoke-Dataset*. To complement this training dataset approximating their characteristics and test the models' performance in the context of forest fires, an image dataset taken in a simulacrum[2] performed by firefighters was also used. The pictures were taken using different smartphones and tablets and correspond to a context similar to that of using the FireLoc application. These datasets were annotated according to the YOLO annotation format, resulting in the distribution of examples shown in the graph of figure 2.

4 Results and Discussion

For the evaluation of YOLOv4 models' performance, the IoU threshold was set to 0.3, lower than the typical 0.5 value used int the COCO dataset detection competition. The lowering of the IoU threshold used was intended to increase the tolerance for object location errors, obtaining a greater number of valid detections. The results of object detection in the images obtained by the models were evaluated by calculating the mean average precision (mAP) value obtained and the classification results of the images as a whole, using confusion matrices. One of the reasons for the analysis of mAP is that, by considering the level of confidence in the detections made, it is possible to evaluate the relationship between false positives and false negatives [4]. The analysis of mAP results allows the evaluation of the models' detection performance. The datasets proposed for training and testing were used for training the models, using the default input size for the images, 416x416. Two classes of objects present in the images were considered: Fire and Smoke, and the anchors were adjusted to the training dataset, using k-means, to approximate the dimensions to those of the objects present in the test images.

Additional tests were performed to adjust the confidence threshold. The model with the best performance obtained in the previous test was used, varying the confidence threshold between 5%, 10%, 15%, 20%, and 25%. All detections whose confidence indicated by the model is lower than the value of the defined threshold are discarded and, as such, the adjustment of the defined threshold may have an influence on the results of both fire and smoke detection in the images (that is, the ability to detect the location in the space of the images correctly) as well as the classification of the images as a whole. Before this test, the confidence threshold value used was the same as in the YOLOv4 [3] work. The graph in table 1 shows that the lower the defined confidence threshold, the best mAP result.

Confidence threshold	**5%**	**10%**	**15%**	**20%**	**25%**
mAP results	**62.2**	60.3	**57.5**	54.8	52.22

Table 1: Test mAP results

These results indicate that the lower the confidence threshold, the greater the number of detections performed, resulting in better mAP results. Therefore, although the best result is obtained with 5% confidence, detection performance was compared considering 5% and an intermediate point 15% confidence.

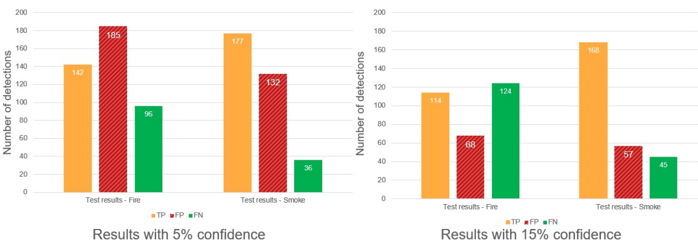

Figure 3: Test results with different confidence threshold

	Predictions – confidence 5 %		
Ground-truth	**Fire**	**Neutral**	**Smoke**
Fire	78	0	6
Neutral	1	22	14
Smoke	14	0	80

Table 2: Classification results with 5% confidence

The results presented in the confusion matrices in tables 2 and 3, correspond to the classification with three classes, after the application of the first post-processing step. The results obtained with 5% confidence,

[2]Images collected on May 15, 2019, in tests carried out in Serra da Lousã by *Associação para o Desenvolvimento e Aerodinâmica Industrial (ADAI)*

	Predictions – confidence 15 %		
Ground-truth	**Fire**	**Neutral**	**Smoke**
Fire	67	0	17
Neutral	0	31	6
Smoke	6	5	83

Table 3: Classification results with 15% confidence

in graph 2, show that the increase of false positive detections does not have a significantly negative impact on the classification results, with the model correctly identifying all the dataset's images where fire or smoke is present. However, as a consequence of the increase of the number of detections that are considered valid, the number of false positives also increases, reaching a total of 15 Neutral images in which smoke or fire is mistakenly identified.

Therefore, two points of operation for the system are proposed, alternating the threshold of confidence in the detections between 5% and 15%, depending on the number of submissions made by the application users. The possibility of switching between two operating points allows for a larger or smaller filter to be applied to the submitted images, adapting the system's operation to times of greater or lesser occurrence of forest fires.

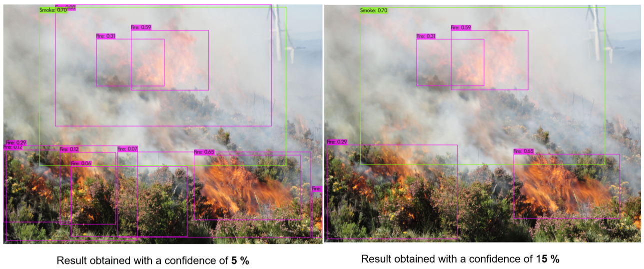

Figure 4: Example of detection results

When comparing the detection results obtained in the same image considering the two proposed operating points, in figure 4, it is possible to observe repeated detections of the same object in both results. When using a 5% confidence threshold, the number of repetitions is noticeably higher, resulting in a greater number of false negatives in the detection results (in the graphs of figure 3). In this case, independently of the defined operation point, the image would be correctly classified with class **Fire**.

5 Conclusions and Future Work

In conclusion, the results obtained for the fire and smoke detection problem in static images are promising. With the developed system, it is possible to get the fire or smoke location detected in the image's space. The YOLO models also allow fire and smoke detections to be made on video, which can be useful in integrating with the FireLoc system if video submissions, in addition to still images, are accepted. The system can be dynamic, absorbing new information by applying new cycles of additional training, using images submitted by users, adapting it better to the application's operating context.

References

[1] Fire-Smoke-Dataset. https://github.com/DeepQuestAI/Fire-Smoke-Dataset/releases/download/v1/FIRE-SMOKE-DATASET.zip, note = accessed 2020-08-22, .

[2] FireLoc - Localize o Fogo. https://fireloc.org/, note = accessed 2020-09-06, .

[3] Alexey Bochkovskiy, Chien-Yao Wang, and Hong-Yuan Mark Liao. YOLOv4: Optimal Speed and Accuracy of Object Detection. (May), 2020.

[4] Mark Everingham, Luc Van Gool, Christopher K.I. Williams, John Winn, and Andrew Zisserman. The pascal visual object classes (VOC) challenge. *International Journal of Computer Vision*, 88(2):303–338, 2010. ISSN 09205691.

[5] Licheng Jiao, Fan Zhang, Fang Liu, Shuyuan Yang, Lingling Li, Zhixi Feng, and Rong Qu. A survey of deep learning-based object detection. *IEEE Access*, 7(3):128837–128868, 2019. ISSN 21693536.

[6] Li Liu, Wanli Ouyang, Xiaogang Wang, Paul Fieguth, Jie Chen, Xinwang Liu, and Matti Pietikäinen. Deep Learning for Generic Object Detection: A Survey. *International Journal of Computer Vision*, 128(2):261–318, 2020. ISSN 15731405.

[7] Zhong Qiu Zhao, Peng Zheng, Shou Tao Xu, and Xindong Wu. Object Detection with Deep Learning: A Review. *IEEE Transactions on Neural Networks and Learning Systems*, 30(11):3212–3232, 2019. ISSN 21622388.