IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received December 22, 2021, accepted January 15, 2022, date of publication January 25, 2022, date of current version February 4, 2022.

Digital Object Identifier 10.1109/ACCESS.2022.3145911

Automatic Early Detection of Wildfire Smoke With Visible Light Cameras Using Deep Learning and Visual Explanation

ARMANDO M. FERNANDES[1,2], ANDREI B. UTKIN[1], (Senior Member, IEEE), AND PAULO CHAVES[1]

[1]INOV—Instituto de Engenharia de Sistemas e Computadores Inovação, 1000-029 Lisboa, Portugal
[2]INESC-ID—Instituto de Engenharia de Sistemas e Computadores—Investigação e Desenvolvimento, 1000-029 Lisboa, Portugal

Corresponding author: Armando M. Fernandes (arm.fernandes@gmail.com)

This work was supported by the ResNetDetect from "Fundação para a Ciência e a Tecnologia" from Portugal under Project PCIF/MPG/0051/2018.

ABSTRACT The present work focus was developing a system for early automatic detection of smoke plumes in visible-light images. The system used a realistic dataset gathered in 274 different days from a total of nine real surveillance cameras, with most smoke plumes being viewed from afar and 85% of them occupying less than 5% of the image area. We employed the innovative strategy of using the whole image for classification but "asking" the neural networks to indicate, in a multidimensional output, which image regions contained a smoke plume. The multidimensional output helped to focus the detector on the smoke regions. At the same time, the use of the whole image prevented wrong image classification caused by a constrained view of the landscape under analysis. Another strategy used was to rectify the detection results using a visual explanation algorithm, Gradient-weighted Class Activation Mapping (Grad-CAM), to ensure that detections corresponded to the smoke regions in an image. The detection algorithms tested were residual neural networks (ResNet) and EfficientNet of various sizes because these two types have given good results in the past in multiple domains. The training was done using transfer learning. Our dataset contained a total of 14125 and 21203 images with and without smoke, respectively, making it, to the best of the author's knowledge, one of the largest or even the largest reported dataset in the scientific literature in terms of the number of images with smoke collected from large distances of various kilometers. This dataset was fundamental to achieve realistic results concerning smoke detection efficiency. Our best result in the test set was an Area Under Receiver Operating Characteristic curve (AUROC) of 0.949 obtained with an EfficientNet-B0.

INDEX TERMS Wildfire, smoke detection, deep learning, ResNet, EfficienNet, Grad-CAM.

I. INTRODUCTION

Despite all scientific advances, wildfires continue to cause extensive destruction throughout the world, frequently causing human fatalities. It is common to see gigantic wildfires in Australia, the United States of America, Spain, or Portugal during summer. Various methods have been developed to detect wildfires as soon as they start. It is fundamental to attack the flames before they spread too much to extinguish them with relative ease. These methods include LIDARs [1], [2], spectrometers [3], wireless sensor networks [4], [5], satellite and drone-based surveillance [6], citizen oversight supported by specific applications [7], and fire lookout towers with human observers [8] or towers with surveillance cameras operating with visible or infrared light [6]. A thorough review can be found in Barmpoutis *et al.* [6]. They all present advantages and disadvantages. The LIDARs have high accuracy and sensitivity but have reduced recognition efficiency at more than 10 km due to laser beam degradation in the atmosphere. The spectrometers, as the LIDARs, have the additional cost of requiring visible-light cameras for detection confirmation. Even though cheap, wireless sensor networks still have to be made environmentally friendly to remove the need to recover after the utilization period.

The associate editor coordinating the review of this manuscript and approving it for publication was Mostafa Rahimi Azghadi.

Satellite surveillance has the disadvantage of the high cost of the satellite deployment and — due to presenting low image acquisition rate and/or spatial resolution — is inappropriate for early detection within a few minutes after ignition. Drone technology has evolved tremendously over the past few years. However, it is still challenging to maintain a fleet of drones to cover large areas all day long under various weather conditions. Citizen oversight is very appealing but does not constitute a reliable, professional infrastructure. The lookout towers with human observers are hard to maintain due to difficulties hiring people for a seasonal job with demanding working conditions. The infrared cameras offer the advantage of detecting hot spots, which may be relevant when reducing false positive alarms but typically present relatively small ranges (less than 10 km) and high prices.

In contrast, visible-light cameras are significantly cheaper than their infrared counterparts and can provide reliable detection at 10 km or even further. These cameras have been used reliably and cost-effectively for years in all possible weather conditions, apparently providing the most well-established method for wildfire detection. It is possible to point out various commercial systems of this type, such as FireWatch [9], ForestWatch [4], and SmokeD [10]. Alkhatib [4] reports FireWatch and ForestWatch systems, using surveillance cameras that operate in the visible range, installed in 186 and 138 towers. These systems also included automatic detection, but the number of false alarms tended to be too large when trying to improve the system detection ability, making it capable of consistently detecting small events. In fact, there is a trade-off between the true positive and false positive percentages. A true positive percentage corresponds to the number of smoke images generating alarms divided by the total number of smoke images. The false positive percentage is the number of alarm-generating smoke-free images divided by the total number of smoke-free images. The trade-off means that an improvement in true positives percentage usually leads to worsening the false positive percentage and vice-versa. One of the objectives of developing a detection system is defeating this trade-off and improving one of these quantities while keeping the other unaltered. A work published in 2012 [11] that analyzed field tests from FireWatch and ForestWatch systems concluded that ''the low rate of detection for small research fires was of particular concern, given the need to detect fires quickly if they are to be suppressed by the initial attack.'' In experiments with bonfires producing flames predominantly smaller than one meter, the work showed that FireWatch detected only one out of six and ForestWatch none out of five. In addition, the systems took a median value of 24 to 31 minutes more time to detect fires than a person in a lookout tower. In this case, even though the false positives were not a problem, the detection ability was far from perfect, leaving a large room for improvements in automatic detection systems. Unfortunately, we are not aware of other more recent evaluations of well-established systems, such as [11]; we can only analyze the advances in automatic wildfire detection with visible-light cameras from various published scientific articles. Even though one can frequently find reported true positive percentages close to 100% and false positive percentages close to zero, these values' reliability is questionable due to the relatively small number of images used to train and test the algorithms. These images may not represent all conditions that the system might need to analyze after being trained. The way to overcome this drawback is to continue gathering large amounts of images, with and without fire, that are as realistic as possible and to look for algorithms capable of providing good results for these images. With this in mind, the present research employs, to the best of the author's knowledge, one of the largest datasets (if not the largest) reported in the scientific literature regarding the number of images with wildfires observed from long distances. The dataset contains mostly images of bonfires and small to medium-scale fires and some fully developed, large-scale fires. Moreover, in the study, we test the most recent deep learning algorithms for smoke detection — residual neural networks (ResNet) [12] and EfficientNets [13]. A thorough comparison of both types of networks is performed to achieve the best possible results. ResNets were chosen because they are a landmark in image classification with convolutional neural networks since they boost the number of hidden layers relevant to solving highly complex problems. Concurrently, they exhibit a relatively small size which allows them to be trained in a relatively small amount of time using standard GPU cards. EfficientNets were an obvious choice due to their ability to beat other networks of equivalent size and complexity in terms of classification efficiency. The present work does not consider networks such as MobileNets [14] because we are not so limited in the availability of resources to need such small architectures that usually cannot reach classification efficiencies as good as those of larger architectures. Hyperparameter optimization is done with Hyperopt [15]. It suggests new hyperparameter values based on the results obtained with previously tested values. The Gradient-weighted Class Activation Mapping (GradCAM) [16], a visual explanation algorithm, is used to test images originating alarms, which allows confirming that the neural network's ''attention'' (region of interest) is on the fire location and not somewhere else. All our images are collected using real surveillance towers with visible-light cameras installed in the Leiria region, Portugal, and correspond to long-distance observation from locations with good visibility; none of the images was from the internet. A peculiarity of our images is that only a tiny part of them exhibit visible flames. This is typical when gathering images from a long distance due to the small size of the flame and obstructions in visibility such as uneven terrain, smoke or fog, trees or other objects. In face of this, our automatic system aims at detecting smoke plumes that rapidly rise above any obstacles. The smoke plumes are harder to distinguish from the background than the high contrast red flames visible in many published works [17], [18]. To lead the detection algorithm to focus on the smoke region while considering the whole image and increase the true positives percentage while not affecting the false

positives percentage, we divide the image into sub-regions. Simultaneously, we transform the neural networks' output from the scalar binary form (0/1 for absence/presence of fire) to a binary vector, with ones in the image sub-regions where the smoke exists. Per example, the array had a dimensionality of nine when subdividing the image into 3×3 sub-regions. We do not know of any other work employing this tactic, that will be shown to help in obtaining the best results. In the present work, even small bonfires were to be detected since, in many cases, large fires start with small burns that grow uncontrolled and, consequently, the authorities should monitor every burn, mainly in the seasons with large fire risk. The conjugation of the small dimensions of a starting fire with the height at which the cameras were positioned and the large distance between the cameras and the fires results in the smoke plume occupying only small portions of the images. In fact, 85% of the smoke plumes in our dataset occupied less than 5% of the image area, with the median of the occupation being 0.96%.

The present work deals with real automatic smoke detection using visible-light images collected at large distances. The authors believe that combining the dataset's realistic nature and large dimension with cutting-edge deep learning algorithms and GradCAM to check the true positives allowed achieving one of the most, or even the most, reliable estimate to date of the expectable detection efficiencies in this situation. This is relevant since it is frequent to find excellent results in the literature that cannot be trusted due to being obtained with small datasets that do not cover significant parts of the possible input space. In addition, their true positive percentage results usually were not checked to assure that they were due to wildfire-related features. In summary, the contribution of the present article to the body of knowledge, to the best of the author's knowledge, is threefold: 1) We present a new method that helps detection algorithms focus on regions with smoke while considering the whole landscape; 2) A visual explanation algorithm rectifies, in a novel way, the true detection percentage by checking all detections to know if they originate from regions with smoke plumes; 3) A reliable estimate of smoke plume detection efficiency is provided due to using the most extensive dataset, in terms of the number of images with smoke plumes, gathered up to date.

II. LITERATURE REVIEW

This literature review will mainly focus on works closely related to the present work due to employing deep learning to detect smoke plumes in landscapes. This leads to the choice of two works, Govil *et al.* [19] and Hohberg [20]. Since dataset size is an essential feature in the present work, the studies of Frizzi *et al.* [18] and Yin *et al.* [21] are also analyzed due to standing out regarding this feature. Finally, the literature review will examine the use of ResNet, EfficientNet, and GradCAM in fire and smoke detection. Please see Gaur *et al.* [22] for a thorough review of works using other types of algorithms for smoke and fire detection that operate with indoor and outdoor images. These other algorithms include Support Vector Machines (SVM), Hidden Markov Models (HMM), Kalman filters, etc., frequently used to analyze features such as color, wavelets, texture, or motion.

Searching the literature on smoke detection in images using deep learning shows that Govil *et al.* [19] is the work whose data is the most similar to that presented here. However, their test set contained only 100 and 150 images with and without smoke, respectively. The reported true positive percentage is 86% for this test set, and the false positive percentage is 5%. The training set [19] contains 8500 images with smoke and 85000 images without smoke. The work also made tests by scanning 184160 images collected during nine days, and the false positive rate was only 0.66 false detection per day per camera. However, one cannot judge the system's overall quality because the true positive percentage was not reported, which means that the images in this large test set were not attributed to smoke or fire. Looking at the reported detection delay, one may see that the 50^{th} and 75^{th} percentiles were 13.5 and 17.75 min, which suggests that some images might have contained undetected fire features.

The master thesis of Hohberg [20] also used images similar to ours since the plume can occupy a small portion of the image, but their images seem to be monochromatic while ours are RGB. The most interesting feature of this work is the large amount of different deep neural networks tested, namely CaffeNet, GoogLeNet, C3D GoogLenet, and an ensemble of the created algorithms. Unfortunately, this work used only a total of 5980 images for training and validation sets, obtaining, at best, for true positive and false positive percentages in the validation set 87.9% and 0.64%, respectively, when using the ensemble. Hohberg did not use a test set.

When analyzing other works that also used deep learning, Frizzi *et al.* [17] and Yin *et al.* [20] stand out due to the unusually large number of images employed and the good detection and false positive percentages. The issue with these works is that, in Frizzi *et al.*, one of the two images shown in the article as an example does not seem to have been taken from a high and distant camera. This is incompatible with a remote detection scenario such as ours. Additionally, the two images exhibit significant amounts of highly distinguishable flames due to their intense red color, which may help to locate smoke. In the present case, only a few images contained visible flames. In the case of Yin *et al.* [21], the non-smoke images seemed to be frequently taken from a close distance and had objects with colors that facilitated separation from smoke. Further, the smoke images were mainly composed of smoke, which helped the separation from other scenes with color. However, Yin *et al.* do not seem to separate smoke from clouds, which should be challenging when all the images contain only smoke. Frizzi *et al.* presented the false positive percentage of 1.2%, testing 1758 smoke images and 2399 images without smoke in totality, including training validation and test sets, of 8915 images with smoke and 11752 images without smoke. Yin *et al.* reported a false positive percentage of 0.6%, testing 1240 and 1648 images with smoke and

without smoke, respectively, in a total of 5695 images with smoke and 18522 images without smoke. Regarding correct smoke detection, the true positive percentage, Yin *et al.*, and Frizzi *et al.* reported 96.37% and 96.6%, respectively.

When focusing on the algorithms used for fire and smoke detection, Yin *et al.* used a modified ZF-Net, Frizzi *et al.* a classical deep convolutional neural network, and Govil *et al.* InceptionV3. Further looking into published research, one may find cases where EfficientNet (Renjie *et al.* [17]) and residual nets [23] are used. Nevertheless, from the samples shown, the images in both works were significantly different from ours as they frequently contain only flames and are mainly collected from a close distance, to the point that we do not believe that comparison is adequate. However, they present good results and datasets with appropriate amounts of images. In Renjie *et al.*, the objective seems to be the detection of fire instead of smoke.

Regarding the use of GradCAM, a scientific article in wildfire detection from drone images crawled from the internet [24] used a Class Activation Mapping (CAM) algorithm [25] to check the image features used to detect the wildfire. Notably, they do not seem to rectify their true positive percentages using this information as we do.

III. DATASET

The images for smoke detection are gathered from cameras mounted in nine surveillance towers installed at distances from the sea between approximately 10 and 60 km in the region of Leiria, Portugal. The cameras operate in the visible spectrum and form a large-area wildfire surveillance system. The images are collected from 2019-05-06 to 2020-05-31 in a total of 274 days. The images containing smoke or not are gathered in 186 and 188 of these days, respectively. There are images with smoke originating from all the surveillance tower cameras, but each does not have images with smoke for all 186 days. FIGURE 1 shows the number of images collected per day. During winter 2019/2020, there were no images collected. The gathered image dataset contains 14125 images with smoke plumes and 21203 images without smoke. The whole dataset is divided into training, validation, and test sets. The training set is used to train the neural networks; the validation set allows choosing the best group of hyperparameter values to train the neural networks, and the test set allows to assess the neural networks' ability to generalize, i.e., to provide correct results for previously unseen images. The training, validation, and test sets are composed of 8638, 2824, and 2663 images with smoke and 8495, 5724, and 6984 images without smoke, respectively, containing, in total, 17133, 8548, and 9647 images. In all images containing smoke, its position is annotated by a rectangular sub-region. Even though the annotations do not represent a perfect outline of the smoke plumes, they are a good proxy of their size. FIGURE 2 contains the histogram of the percentage of each image covered by the smoke plume annotation area. There exist 22255 such annotations because some images have more than one smoke plume.

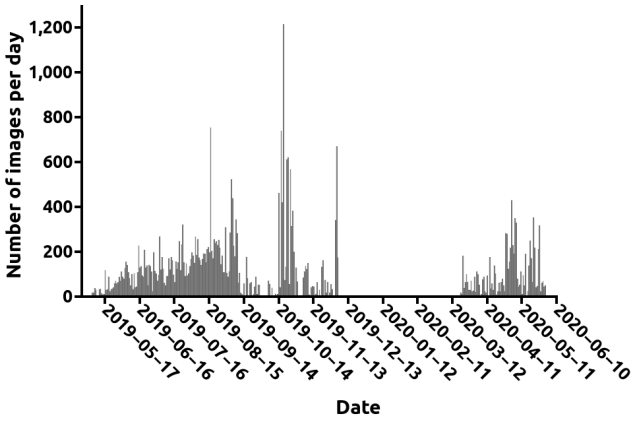

FIGURE 1. The number of images collected per day between 2019-05-06 and 2020-05-31.

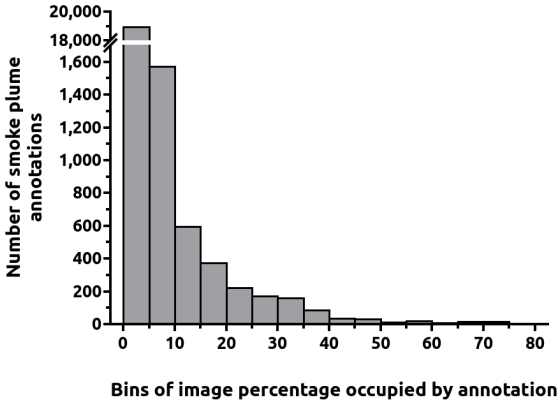

FIGURE 2. Histogram of the percentage of each image covered by a smoke plume annotation. The first bin contains 85% of all annotations.

Statistically, 85% and 92% of all smoke plume annotations occupied, respectively, less than 5% and 10% of the image area. This shows that plumes under detection are small compared to the image size. FIGURE 3 depicts various typical distant landscape images from our dataset. The images can show flat ground or hills or a mix of both. The sky can be cloudy or clean, which originates variations in the available sunlight and the ground shadows. The clouds can be high in the sky or relatively low. The images can contain houses or be almost devoid of human-made features. No significant differences between images with and without smoke were found regarding these characteristics. The top left image has a smoke plume annotation occupying an image area equal to the median area of the annotations, 0.96%. The training, validation, and test sets contain images of three different groups of surveillance towers; this way, it is possible to assess the created neural networks' potential for classifying images from towers that are not yet installed. The training, validation, and test sets contained images from four, two, and three towers, respectively. Since the cameras in the surveillance towers are rotating, the training, validation, and test sets included, respectively, the images from 103, 66, and 63 view directions for images with smoke and 108, 56, and 81 view directions for images without smoke.

FIGURE 3. Various landscapes with smoke plumes. The top left image includes an annotation occupying an image area equal to the median area of the annotations, 0.96%. The annotation was added only for illustrative purposes.

IV. METHODS

A. RESIDUAL NEURAL NETWORKS (RESNET)

Residual neural networks [12] solve the problem that deeper neural networks performed worse than shallower networks. The way to build deeper networks was the introduction of shortcut connections allowing for residual learning. The shortcut connections are direct connections from the input to the output of the various (residual) modules that compose a residual network. Residual learning means that instead of learning an input-to-output mapping called H, one can use each module to approximate a residual function F = H-x where x is the module input, propagated by the shortcut connections. While the modules can learn H or F, it is easier to learn F-functions. The shortcut connections also allow for better propagation of the training gradients that tend to vanish in deeper networks. The residual modules are formed by a 1×1 convolution, followed by a 3×3 convolution and another 1×1 convolution. This module is said to have a bottleneck design. The 1×1 convolutions reduce and subsequently increase the dimensions of the information flowing through the network. The dimensions are reduced before and increased after the computationally expensive 3×3 convolutions.

B. EFFICIENTNET

EfficientNets are one of the most advanced neural network structures created up to now. They were first presented in 2019 [13]. They were developed to be smaller and more efficient neural networks than their predecessors. Their leap forward was realizing that scaling in terms of width, depth, and resolution was not independent. Therefore, a compound scaling method was proposed. The depth width and resolution depended on three parameters optimized by grid search and another parameter used to scale up the neural network complexity. The grid search was done for EfficientNet-B0, whose architecture was determined by a multi-objective neural

architecture search designed to optimize both accuracy and floating-point operations per second (FLOPS). EfficientNet is constructed mainly of mobile inverted bottleneck, MBConv blocks [24] with squeeze-and-excitation added. These blocks contain a 1×1 convolution to increase the number of channels, followed by 3×3 and 1×1 convolutions that reduce the number of channels. This allows adding, to the output, the block input passing through a residual connection. The squeeze-and-excitation gives different weights to the various channels instead of considering them all equal. The learning process defines the values of these weights.

C. TRANSFER LEARNING

When training large neural networks such as a ResNet or an EfficientNet, we do not always have available an amount of data that allows us to obtain the best possible results. In this situation, it is now standard practice to use a pre-trained [26] neural network, usually with the Imagenet dataset [27]. This way, the neural network section dedicated to feature extraction is well adjusted to real images. It can extract characteristics such as edges with different orientations, even before training with the data from the new problem occurs. However, it is necessary to remove the top of the pre-trained neural network in order to create a fully connected layer with a number of output neurons that is adequate for the problem in hands. Next, a part or the whole neural network is retrained with the data relevant for the new problem. In the present work, we used the latter possibility. All the neural networks reported in the present work were trained employing transfer learning of neural networks pre-trained with the Imagenet dataset.

D. AREA UNDER RECEIVER OPERATING CHARACTERISTIC CURVE (AUROC)

The Receiver Operating Characteristic (ROC) curve plots the percentage of true positive versus the percentage of false positives when the decision threshold of the classifier varies between its minimum and maximum value. ROC curves are the best way to choose the models capable of achieving the highest true positive percentage for a certain false positive percentage and demonstrate the trade-off between these two percentages. When trying to optimize hyperparameter values with software such as Hyperopt, it is necessary to have one parameter that is minimized or maximized. The whole ROC curve cannot be used for this purpose because it is a two-dimensional, visual approach; since the area under the ROC curve (AUROC) can be, it is the optimization parameter employed in the present article. The AUROC is also used to choose between model types once their hyperparameter values are set.

E. GRADIENT-WEIGHTED CLASS ACTIVATION MAPPING (GRADCAM)

Neural networks are known as black boxes because they produce results that are not easily explainable. Consequently, when they provide strange results, there might not be a way of understanding its cause. The objective of Gradient-weighted Class Activation Mapping (GradCAM) [16] is to provide, in the words of the inventors, ''visual explanations'' of the neural network decisions. To give these explanations, a coefficient of the importance of a convolutional feature map before the fully connected layers is determined. This coefficient corresponds to calculating the score gradient for a specific class with respect to the activation of a feature map from a convolutional layer and, afterward, average pooling these gradients that come from backpropagation. Next, the coefficient of each feature map and the feature map activations are multiplied and added over all the feature maps. The GradCAM outcome is the result of passing this sum by a ReLU activation function. This ReLU is used to show only the pixels that have a positive influence on the desired outcome.

In the present work, GradCAM, when employed, serves to rectify the true positive percentages. The reason is that, sometimes, the algorithms pointed out an image as having smoke even though they were not focusing on the smoke but in some other image feature such as a cloud. The rectified AUROC results presented below correspond to having an overlap between the neural network attention area, calculated with the GradCAM, and the annotation area, larger than 10% of the area of the annotation. The 10% value guarantees proximity and at least some intersection between the annotation and the attention area when a smoke plume is considered to be correctly detected. The neural network attention area is the region of the GradCAM output image where values that vary between 0 and 255 surpass the value 200. This value allows delimitating image regions whose size is comparable to that of the smoke plume. FIGURE 4 shows an example. The middle image depicts the attention area of the neural network in yellow, and the bottom image is the segmentation of the attention area. The AUROC rectification derived from the correction of the true positive percentages while the false positives remained unchanged.

F. MOSAIC OUTPUT

In the present case, we have decided to test the possibility of using images where the smoke to be detected can occupy a small area of the image. The purpose is to provide the neural network with information about the surroundings of the smoke plumes that must be detected. A narrower field-of-view can easily lead to misinterpretation of what is visible in the image. This is relevant, per example, when trying to distinguish smoke from clouds, with the former being usually more localized and generally having a different shape from the latter. However, these characteristics were only visible if the image had a broad view that included the whole smoke plume and/or at least large parts of the cloud structure.

When creating a neural network for wildfire smoke detection, its output may have a dimensionality of one, representing a value that indicates a detection when passing a certain threshold. In this work, the decision to analyze the entire image led us to train the neural network to show the sub-regions of the input image where smoke is present.

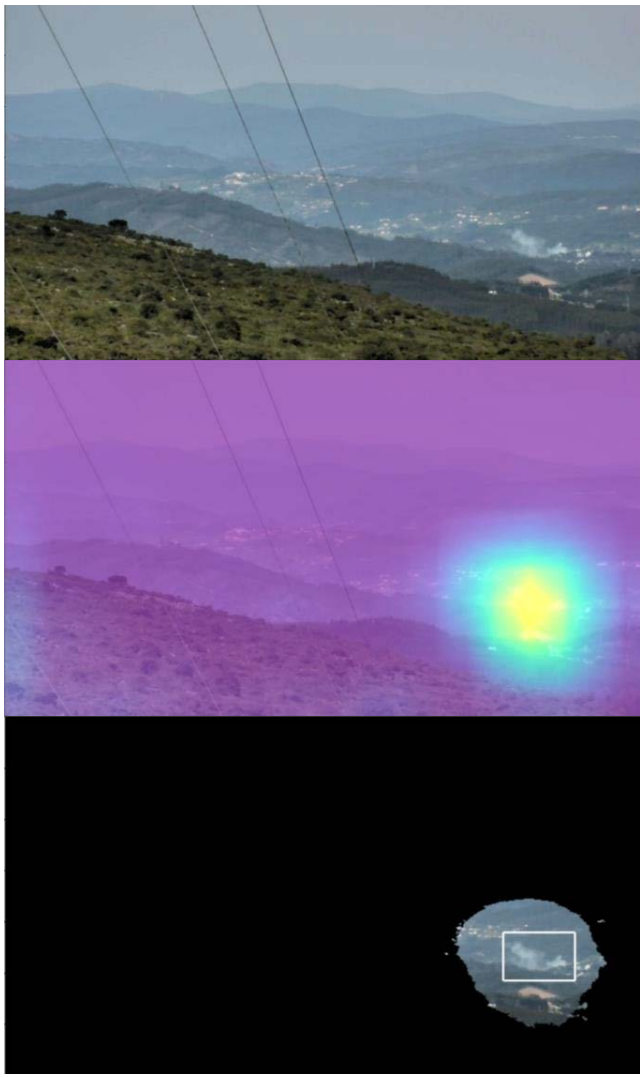

FIGURE 4. Example of GradCAM application. Top: original image with smoke plume; Middle: neural network attention area in yellow and blue; Bottom: Segmented region of attention with annotation in white.

Consequently, the neural network output has a dimensionality equal to the number of the input image sub-regions, with zeros for the outputs corresponding to those having no smoke and ones for those with smoke. The input image region with smoke is that delimited by the rectangular annotation. FIGURE 5 shows an example of a 3 × 3 mosaic. The advantage of this approach is that of helping the neural network to focus on the sub-regions with smoke, and therefore alarms for an image containing smoke plumes are actually triggered by these plumes. This strategy also has the advantage of providing a rough location of the smoke plume position in the image. The mosaic approach does not require changing the input image before applying the neural network; only the output differs. The annotation and mosaic region boundary lines, shown in FIGURE 5, are just for explanation purposes; they are not drawn on the input images given to the neural network.

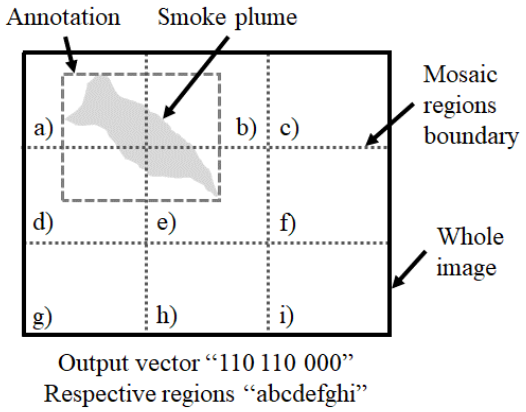

FIGURE 5. Smoke image with 3 × 3 mosaic regions and output vector for the smoke annotation.

G. AUGMENTATION

Augmentation [28] artificially increases training data by transforming the images in the original training dataset. The neural network training uses augmentation since that is known to improve the classification results. Image augmentation is done in five different ways, creating an undulation of the image pixels in the horizontal and the vertical directions separately, blurring the image using a Gaussian, adding noise from a discrete uniform, and selecting image regions. Blurring consists of convolving the images with a Gaussian kernel.

H. HYPEROPT

Hyperopt [15] did the hyperparameter optimization of the ResNets and the EfficientNets. It implements a Bayesian approach called Sequential Model-based global Optimisation (SMBO) [29] that is suitable when there is a high cost to evaluate a fitness function. This cost comes from the fact that the fitness function is, in the present case, the classification efficiency of the neural networks trained with a specific set of hyperparameter values and, it may take many hours to calculate and sometimes days. The strategy behind SMBO is to minimize the number of evaluations of the fitness function by finding the most promising hyperparameter values based on previous evaluations of the fitness. This is different from grid search, which evaluates all combinations of hyperparameter values, or from a random search where the chosen values do not consider information from previous runs. SMBO works with a surrogate function which is a probabilistic representation of the score of the fitness function, taking into account all the evaluations of this fitness. From this surrogate function, it is possible to find the next set of hyperparameter values to test by maximizing a selected function, Expected Improvement for Hyperopt. In Hyperopt's SMBO, the surrogate function is determined using the Tree-structured Parzen Estimator (TPE) [29]. It calculates the probability of having a specific score given a set of hyperparameters. This requires using the Bayes rule and manipulating the probability of having a set of hyperparameters given a certain score. Hyperopt

employs a range of values for each hyperparameter to optimize, and the algorithm continuously tries to find the best combination of values inside the range.

V. NUMERICAL RESULTS

In the present work, wildfire smoke detection uses both ResNets and EfficientNets. For ResNet, the architectures chosen have 18, 34, and 50 layers, and for EfficientNet, the trained types are of the B0 up to B5 kind. The tendency is to use the smaller architectures to get the best possible generalization for the same amount of training patterns. The used loss function is binary cross-entropy, and the number of training epochs is forty at maximum, but early stopping is employed. Training ends when the AUROC does not improve after five epochs. Training also ends when AUROC does not reach 0.8, 0.9, or 0.95 after 3, 5, or 10 epochs to prevent wasting time with ineffective models. Hyperopt optimizes learning rate, batch size, kernel regularisation, dropout value, and the number of neurons in the fully connected network on the neural networks' top. We use the training algorithm Nadam [30], for which Hyperopt also optimizes the two beta values. Each Hyperopt run for hyperparameter optimization consists of ten trials from which the one with the best AUROC is chosen. The obtained results, shown in TABLE 1, correspond to calculations with the image size of 349 by 620 pixels, a mosaic, when used, of 3 × 3, and augmentation of twice the number of real training images. These values originated in running some preliminary optimization tests. Further attempts to improve the results for the best neural network from TABLE 1 by changing these parameters did not return significant improvements. This suggests that the extensive optimization done concerning the neural network type, namely ResNets of three different sizes and six EfficientNets, has already provided results that are hard to improve due to the proximity to an optimal combination of parameter values. The calculations employed graphical processing units (GPU) NVIDIA Tesla K20 and NVIDIA GeForce RTX 2080 Ti.

A. RESULTS FOR THE VALIDATION SET

The results for the validation set are shown in TABLE 1, FIGURE 6, and FIGURE 7.

TABLE 1 contains the results obtained for the three different types of ResNet tested and the six types of EfficientNet. Each row contains the results of two distinct neural networks, one trained without any mosaic (columns 1-3) and the other with a 3 × 3 mosaic (columns 4-6). For each neural network, three columns are presented: 1) one for the AUROC without rectification, i.e., considering all network detections as being correct; 2) another with AUROC with rectification, i.e., excluding the detections where the neural network "attention" is not close to a smoke plume annotation, and 3) a third column showing the number (quantity) of trainable weights that connect the various neurons and form the convolutional filters within a neural network. This last parameter is included for analysis since it is a visible consequence of the optimization procedure due to the variation of the number of neurons in the fully connected layer before the output neurons. In addition, as we will see later, it may help to unveil at least some part of what is happening with the results. The table's column 7 also contains the percentage of variation in the number of trainable weights between neural networks of the same type, without and with mosaic. FIGURE 6 and FIGURE 7 depict the values from TABLE 1, the first one for the AUROC values and the second one for the number of trainable weights.

TABLE 1. Unrectified and rectified AUROC for various ResNet (RN) and EfficientNets (B) trained without mosaic and with 3 × 3 mosaic output. The number of trainable weights and its percentage of variation between the cases without and with mosaic.

	No Mosaic			With Mosaic			
	AUROC Unrectified	AUROC Rectified	Number of trainable weights	AUROC Unrectified	AUROC Rectified	Number of trainable weights	% Variation in trainable weights
Column number	1	2	3	4	5	6	7
RN18	0.979	0.905	12206948	0.963	0.836	12222956	0.131
RN34	0.983	0.920	21801108	0.971	0.943	21339316	-2.12
RN50	0.975	0.908	26609593	0.963	0.867	23740401	-10.8
B0	0.984	0.831	6571549	0.971	0.958	4136557	-37.1
B1	0.982	0.863	9077185	0.974	0.950	8448193	-6.93
B2	0.981	0.890	9815995	0.967	0.905	9828003	0.122
B3	0.982	0.895	11465233	0.973	0.841	13015241	13.5
B4	0.981	0.898	18445617	0.977	0.679	21152625	14.7
B5	0.975	0.897	32440785	0.968	0.748	31427793	-3.12

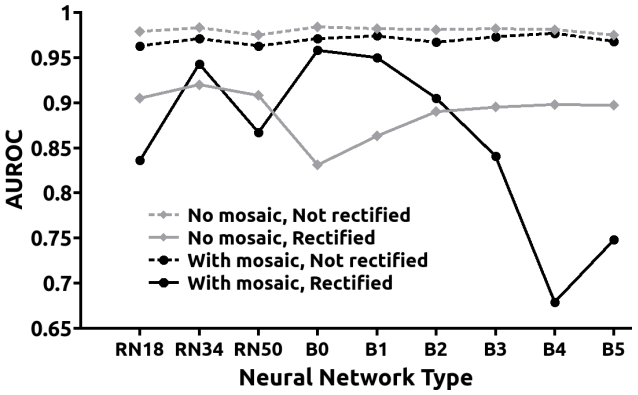

FIGURE 6. Depiction of the AUROC values from TABLE 1. RN stands for ResNet, and B0 to B5 are EfficientNets.

1) ANALYSIS OF VALIDATION RESULTS WITHOUT MOSAIC

When looking at the results from TABLE 1 and FIGURE 6 for neural networks without mosaic and without rectification (column 1 in TABLE 1 and dash grey curve in FIGURE 6), i.e., without ensuring that the focus of the network coincides with the region containing smoke, one may see that all AUROC values had a slight variation between 0.975 and

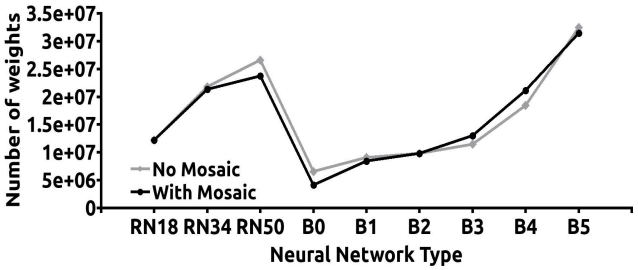

FIGURE 7. Depiction of the number of trainable weights from TABLE 1. RN stands for ResNet, and B0 to B5 are EfficientNets.

0.984. Reporting results without mosaic and without rectification is the standard way in previously published articles. However, for results after rectification (column 2 in TABLE 1 and solid grey curve in FIGURE 6), the values drop, in some cases quite dramatically, which is the case for EfficientNet-B0 where the AUROC of 0.984 plunged to 0.831. All neural network types demonstrate a significant decrease in their AUROC after rectification. This is a strong indication that the neural networks often classify some spurious features as smoke plumes, so choosing a classifier that does not focus on smoke may lead to many future misclassifications. The large gap between unrectified and rectified AUROC values provides support to the need for rectification.

With no mosaic but with rectification (column 2 in TABLE 1), the maximum AUROC attained with ResNet is 0.920, which is better than the maximum value of 0.898 for the EfficientNets. This suggests that the former might be less susceptible to focusing on features that are not smoke. The best ResNet had 34 layers; increasing or decreasing the number of layers led to worse results. With EfficientNets, the rectified AUROC values improved as the networks became more complex, between B0 and B5 (see column 2 in TABLE 1 and solid grey curve in FIGURE 6). This seems to be vaguely similar to what happens when training with Imagenet data, with saturation occurring for B3, B4, and B5.

2) ANALYSIS OF VALIDATION RESULTS WITH MOSAIC

When the mosaic output is used without rectification (column 4 in TABLE 1 and a dashed black curve in FIGURE 6), the AUROC values range between 0.963 and 0.977. This is a slight decrease, in general, relative to the situation without mosaic and without rectification (column 1 in TABLE 1 and dashed grey curve in FIGURE 6). The most significant decrease in AUROC between not using and using mosaic with no rectification, i.e., comparing columns 1 and 4 in TABLE 1, occurs for ResNet-18 and is 0.016. However, this decrease happens for all network types, see dotted grey and black curves in FIGURE 6, which is expectable since the problem with the mosaic output is more complex to solve than just classifying the image as containing smoke or not. The reason is that the neural network output dimensionality is larger when using mosaic, and the problem nature is different since we no longer detect only the presence of smoke but, at the same time, indicate the smoke position.

When analyzing the EfficientNets with mosaic and rectified results (column 5 in TABLE 1 and solid black curve in FIGURE 6), a considerable variation in the AUROC values between B0 and B5 is visible. The B0 network type reached an AUROC of 0.958, the best rectified result of the various neural network types tested, including ResNet; B4 obtained the smallest AUROC value of 0.679. While for B0, the drop in AUROC between unrectified and rectified results (columns 5 and 6 in TABLE 1) was only 0.013, for B4, the fall was 0.298, giving us an indication that the more complex EfficientNet did not respond well to the mosaic output.

When rectifying the results for the use of mosaic, the outcome was somewhat surprising in the EfficientNets case (see column 5 and solid black curve). The reason is that the curves in FIGURE 6, with and without mosaic, both with rectified results (the solid grey and black curves), followed different trends. While in the case without mosaic (the solid grey curve), the AUROC improved and saturated when the EfficientNets became more complex, as already mentioned, in the case with mosaic (black solid curve), the AUROC became worse when the complexity increased, at least from B0 to B4. This might be due to differences in the problem nature, namely the change from without mosaic to with mosaic.

Regarding the ResNet case, the AUROC values follow similar trends, regardless of whether the mosaic is used or not. Both situations have rectified results illustrated by the solid grey and black curves in FIGURE 6. The AUROC values for ResNet-18 and ResNet-50 are worse than for the two ResNet-34, but there is a difference between the two curves: the solid grey curve is flatter than the black solid curve. This means that, for ResNet-34, the AUROC diminishing due to rectification was smaller when the mosaic was employed. The opposite occurs for ResNet-18 and ResNet-50, i.e., the smaller drop happens when the mosaic is not used. Consequently, the best ResNet is the architecture with 34 layers and with mosaic employed.

3) ANALYSIS OF THE NUMBER OF WEIGHTS

When no mosaic is used, ResNet-34, which presents the best rectified AUROC for both ResNets and EfficientNets, had the third-largest number of weights for both ResNet and EfficientNet (column 2 in TABLE 1). ResNet-34 has better results than ResNet-50 in all situations, suggesting that the latter might be too large for the number of images available for training. For EfficientNets, the increase in the number of weights does not lead to worse AUROC (solid grey curves in FIGURE 6 and FIGURE 7); one concludes that, for these neural networks, without mosaic, having a large number of weights is not harmful. On the other hand, it is interesting to note that the best neural network with mosaic and with rectified values, for both ResNet and EfficienNet, is EfficientNet B0, which is the one with the smallest number of weights. Consequently, with mosaic, the large number of weights does

not seem to be helpful. The reduced number of weights brings in the advantage of faster training and validation relative to larger EfficientNets. It also facilitates neural network deployment by requiring GPU with less memory to run.

For EfficientNets, there seems to exist some correlation between the number of weights in the neural networks and the rectified AUROC values. For B0 and B1, the number of weights when the mosaic is used (black curve in FIGURE 7) is smaller than for no mosaic (grey curve in FIGURE 7), and the AUROC values are better, see solid black and grey curves in FIGURE 6. For B3 and B4, the same trend occurs; the smaller number of weights, now for the no mosaic case, also means better AUROC. With B2, the almost equal number of weights corresponds to a similar AUROC value. Concerning B5, following the previous logic, the smaller number of weights with mosaic than in no-mosaic case was expected to result in better AUROC for the former. However, this is not observed, even though B5 AUROC (with mosaic, rectified) undergoes an improvement with respect to B4. This change in the trend suggests that the influence of the number of weights may not be straightforward and/or that there may be some other factors influencing the results. In addition, the possible correlation between AUROC and the number of weights does not imply that the latter's variation is driving the former's variation.

4) RECEIVER OPERATING CHARACTERISTIC (ROC) CURVES IN VALIDATION

To further analyze the obtained results, the receiver operating characteristic (ROC) curves for the best models with and without mosaic are plotted in FIGURE 8 to reveal the impact of using the mosaic output on the balance between true positive and false positive percentages. This is relevant because a usual complaint about automatic detection systems is that they exhibit a large false positive percentage. Subsequently, one must check if employing the mosaic-like output can improve the situation, lowering this percentage.

A detailed view of FIGURE 8-a) is shown in FIGURE 8-b) to help understand what is happening in the small false positive percentages regime. As seen from FIGURE 8-a), the most obvious difference introduced by the mosaic application is that the true positive percentages are larger throughout most of the ROC curves. Surprisingly, for small false positive percentages, less than ∼2%, the models without mosaic reached better true positive percentages than those with mosaic. This raises the problem that if we are willing to accept a model with a lower true positive percentage to have a better false positive percentage, we should choose the models without mosaic with smaller AUROC. The following section will address this issue.

B. RESULTS FOR THE TEST SET

Before measuring a neural network generalization ability using the test set, one must first choose the best neural network using the validation set results. Considering the results of the previous section, selecting as the best neural network

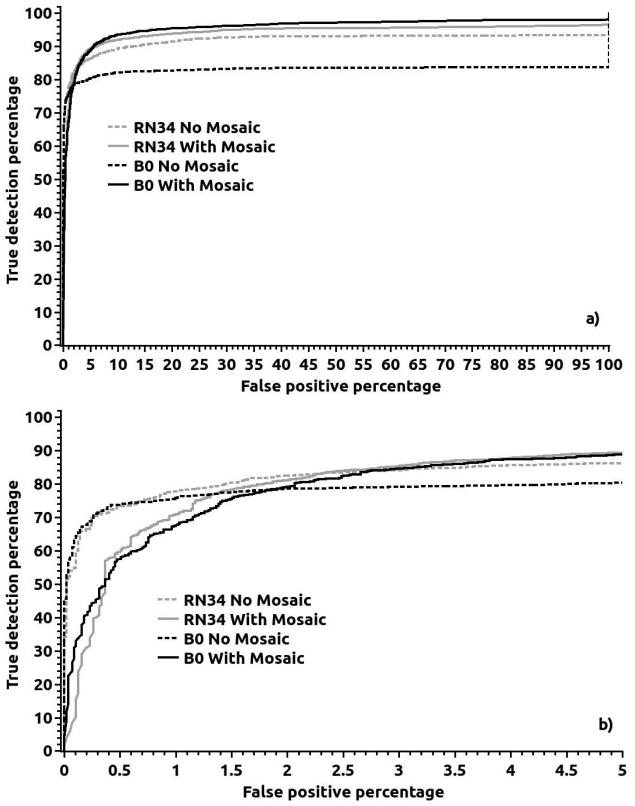

FIGURE 8. ROC curves for the best models from TABLE 1 with and without mosaic output. In b) is a detail of the ROC from a) for the small false positive regime. All values are rectified.

the one with the best AUROC would lead to choosing the EfficienteNet-B0 with mosaic output. However, if the choice aimed at finding the model with the best true positive percentages for the smallest false positive percentages, one would choose the ResNet-34 without mosaic. Another possibility is to select EfficientNet-B0 for situations where the false positive percentage is not so critical and ResNet-34 otherwise. The last option is somewhat interesting but requires knowing if both models generalized well so that we could use them. The generalization analysis consists of checking the results in the test set. It would have been better to test only one model, to avoid any risk of overfitting to the test set, but the circumstances did not seem to offer us better options.

In addition, at this point, this risk is minimal since all values of the model parameters have already been set. The validation and test ROC curves are plotted in FIGURE 9, only for rectified results, to help analyze the two models' generalization ability. FIGURE 9-b) shows the small false positive percentage regime from FIGURE 9-a). Both figures reveal a good coincidence between the validation and test curves for the EfficientNet-B0 with mosaic, suggesting a good generalization. With the ResNet-34 without mosaic, there is a separation between the validation and test curves, indicating that generalization is worse than in the EfficientNet case. FIGURE 9-b) also shows that in the test set, for the small false positive percentage regime, ResNet-34 without

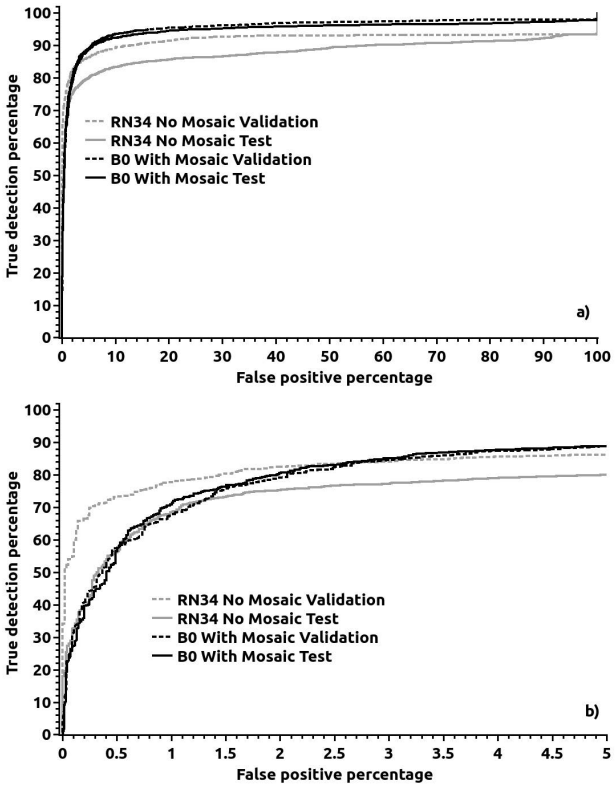

FIGURE 9. ROC curves in validation and test sets for the best models from TABLE 1. In b) is a detail of the ROC from a) for the small false positive regime. All values are rectified.

mosaic is not better than EfficientNet-B0 with mosaic. After a false positive percentage of approximately 1.2%, it becomes worse. This result indicates that ResNet-34 without mosaic is not useful in the small false positive percentage regime, as the validation results imply. It also suggests that employing the mosaic output is not detrimental to the false positive percentage because both models with and without mosaic show the same behavior in this regime for the test set.

TABLE 2 contains the comparison of the AUROC values for the validation and test sets of only EfficientNet-B0 with mosaic since the previous results show that using this model is good enough in all the false positive percentage ranges. The table contains the percentages of true and false positives for a decision threshold of 0.5. This is a typical value for the threshold when the classifier output is between zero and one, as in the present case. The F1-score and the Matthews Correlation Coefficient (MCC) are also provided for the same decision threshold.

The validation set's rectified AUROC, F1-Score, and MCC are 0.958, 0.894, and 0.842, respectively. With the decision threshold of 0.5, these evaluation metrics correspond to the true positive percentage of 89.2% and the false positive percentage of 5.1%. For the test set, the rectified AUROC, F1-Score, and MCC are 0.949, 0.882, and 0.840, with the true positive percentage of 85.3% and the false positive percentage of 3.1% for the same decision threshold. The variation in rectified AUROC, F1-Score, and MCC between validation and test are only 0.009, 0.012, and 0.002. A mismatch between validation and test is visible in the rectified true positive percentages. The latter is smaller/worse by 3.9 percentage points, but the false positive percentage is two percentage points smaller/better. The results difference may be due to collecting the validation and test images in different surveillance towers with various surrounding geographical characteristics.

TABLE 2. Comparison of the AUROC values for the validation and test sets of EfficientNet-B0 with mosaic from TABLE 1. F1-score and Matthews Correlation Coefficient (MCC) are also given.

	Validation	Test
Unrectified AUROC	0.971	0.965
Unrectified true positive percentage	89.6%	86.0%
Rectified AUROC	0.958	0.949
Rectified F1-score	0.894	0.882
Rectified MCC	0.842	0.840
Rectified true positive percentage	89.2%	85.3%
False positive percentage	5.1%	3.1%

The gap between unrectified and rectified AUROC values is 0.013 in the validation set, being almost the same, 0.016, in the test set. This indicates that in test, as in validation, the neural network concentrates attention on the smoke plumes when issuing an alarm as desired in this kind of application. For validation and test, the drop in true positive percentages between the unrectified and rectified cases is 0.4 and 0.7 percentage points, respectively.

FIGURE 10. Examples of images with smoke plumes that are not detected by the best neural network, EfficientNet-B0 with mosaic.

FIGURE 10 presents two images with smoke plumes that were not detected by EfficientNet-B0 with mosaic, from

FIGURE 11. Examples of images without smoke plumes that originate false positives with the best neural network, EfficientNet-B0 with mosaic.

TABLE 2. When analyzing all the misdetections, apparently it is not possible to find obvious reasons for their occurrence. One possible reason considered was the plume size and contrast, but one may find various examples of distinct, large plumes that are not detected and of small, less perceptible ones that are detected. Another possible reason analyzed was the smoke plume shape but again one may find plumes of various shapes that are detected or not. FIGURE 11 contains two images without smoke plumes that originate false positives with the same neural network. The top image contains low clouds which are a major cause of false positives due to their resemblance to smoke. The bottom image should not be problematic, since the natural illumination seems normal and the clouds are inexistent, but still, something triggers a detection. From the analysis of FIGURE 10 and FIGURE 11, one may conclude that the non-linearity typical of neural networks makes it hard to understand, excluding the clouds' situation, the reasons behind the neural network decisions, even with GradCAM.

Finally, the parameter values for the best neural network were number of training epochs, 3, number of hidden neurons in the neural networks' top, 100, learning rate, 9.97e-05, batch Size, 16, beta one, 0.865, beta two, 0.999, kernel_size, 3, dropout, 0.145, kernel regularization L1, 5.54e-07, and L2, 8.88e-05, activity regularization L1, 6.96e-07, and L2, 1.31e-4.

C. EXPERIMENTS ON ANOTHER DATASET AND OTHER NEURAL NETWORK TYPES

To analyze whether the mosaic approach brings any advantage with new images, we created smoke detectors using the HPWREN dataset [31]. We employed 671 and 608 smoke and non-smoke images, respectively. With this small dataset, the mosaic effect was not visible for ResNets and EfficientNets. Consequently, it was decided to test MobileNets and InceptionV3 [32] for the existence of the mosaic effect. MobileNets were chosen due to their small number of trainable weights, less than 1 million, because the larger size of ResNets and EfficientNets was a hypothetical cause of the mosaic effect not being visible on the small HPWREN dataset. InceptionV3, with a significantly larger number of weights, more than 20 million, was chosen due to having already been used by Govil *et al.* [19] with good results on an HPWREN dataset, different from that used in the current section and which, up to the authors' knowledge, is not publicly available. The SuperDuper-v1, SuperDuper-v2, and SuperDuper-edge models [33] created by "AI For Mankind" [34], which collaborates with HPWREN, were not included in the present analysis because they are "object detectors" that operate differently from ResNet or EfficientNet that are "image classifiers." Judging from the site images and information [33], the SuperDuper models were trained to place a rectangle around the detected smoke plumes, which implies providing the smoke plume location/annotation to the neural network training algorithm; therefore they do not require a mosaic to help focus on the smoke plume. On the contrary, works before ours, do not employ the location/annotation in the ResNets, EfficientNets, MobileNets, and InceptionV3 training. Consequently, the mosaic brings in this relevant information for guiding the neural network training. The HPWREN dataset, the MobileNets, and InceptionV3, as well as the related results, appear only in the present section. They are not mentioned anywhere else in the present article to avoid confusion.

For a MobileNetV2 with parameter alpha equal to 0.35 and a 3×3 mosaic, the rectified AUROC is 0.923; without mosaic, it is only 0.748. For a larger MobileNetV2 with parameter alpha equal to 0.5 and a 2×2 mosaic, the rectified AUROC is also 0.923, and without mosaic is 0.87. For InceptionV3 with a 3×3 mosaic, the rectified AUROC is 0.985 while without mosaic it is 0.957. These values show that the better results obtained with the mosaic approach are not exclusive of the Leiria dataset described in section III. They also show that the mosaic approach can be beneficial for networks besides ResNet and EfficientNet, specifically MobileNets and InceptionV3, despite the large differences in their number of weights.

VI. DISCUSSION

A. COMPARISON TO PREVIOUS WORKS

1) HOHBERG AND GOVIL *et al.* WORKS

When comparing the present work to others published in the literature, Hohberg [20] and Govil *et al.* [19] seem to have images comparable to ours — despite those of Hohberg are black and white, and ours are RGB. Hohberg used, in total, 5980 images for training and validation with unclear proportions assigned to each set, which is only 85% of the number

of images that we used only for testing, namely 9647 images. In addition, he did not use a test set, obtaining true positive and false positive percentages in the validation set of 87.9% and 0.64%, respectively. The use of smaller datasets is problematic because outdoor images, either containing smoke or not, present a large variability in the characteristics of the observed (chaotic) phenomena, being almost impossible to guarantee that a representative sample of those phenomena has been collected. Consequently, less data usually means a worse coverage of all the possible phenomena. As a result, Hohberg's significantly smaller dataset, together with not providing classification efficiencies for an independent test set, most probably make his results less reliable than ours.

Concerning Govil *et al.*, a very small test set of only 100 and 150 images with and without smoke, respectively, is used, achieving a true positive percentage of 86% and a false positive percentage of 5%. A test set with more than 180 thousand images is also used. Still, only the false positive percentage is reported, and results cannot be duly evaluated without the true positive percentage value. The reason is that a good false positive percentage may cause a less good true positive percentage. In summary, one of Govil *et al.* test sets is too small, and the results provided for the other are insufficient for comparison.

2) FRIZZI *et al.* AND YIN *et al.* WORKS

When comparing our work to those of Frizzi *et al.* [18] and Yin *et al.* [21], we start with analyzing the number of samples used. For testing, we used 2663 and 6984 images with and without smoke, respectively. This compares favorably to the 1758 and 1240 images with smoke, for Frizzi *et al.* and Yin *et al.*, respectively, and 2399 and 1648 images without smoke, also respectively. The present work used 51% (1.51×) and 115% (2.15×) more images with smoke than Frizzi *et al.* and Yin *et al.*, respectively. These values are even more dilated for the number of images without smoke; the present work employed 191% (2.91×) and 324% (4.24×) more images than Frizzi *et al.* and Yin *et al.*, respectively.

Regarding the content of the images, those of Yin *et al.* are quite different from ours, with smoke occupying the image area wholly. This facilitates the separation from colorful objects but might bring problems in smoke-clouds separation, which is essential when filming landscapes throughout the year. An additional difference from our images is filming at short range. Considering the essential differences shown, a comparison between the present work and Yin *et al.* is only mildly relevant. The images from Frizzi *et al.*, judging from the two samples shown, were less different from ours than those of Yin *et al.*. Nevertheless, if the amount of images taken at close range, which is unknown to the authors, is large, it can significantly change the problem solved by Frizzi *et al.* when compared to that solved in the present work. The contrary may also happen. These doubts about image content create a considerable uncertainty regarding the relevance of comparing Frizzi *et al.* work with ours; nevertheless, the significant difference in the number of images used again suggests that our results may be more reliable than those of Frizzi *et al.*. In terms of true smoke detection, our work with 85.3% (86% unrectified) in the test set compared to the 96.37% and 96.6% in Yin *et al.* and Frizzi *et al.*, respectively. For the false positives, our work presents 3.1% against 0.6% and 1.2% in Yin *et al.* and Frizzi *et al.*, respectively. In summary, Frizzi *et al.* and Yin *et al.* present better results than this work. Still, their results may be less reliable than ours due to the smaller amount of data used, and, in addition, their images may be so different from ours that the comparison may not be conclusive.

B. USING MOSAIC-LIKE OUTPUT

1) MOSAIC AND FALSE POSITIVES

For the EfficientNet-B0 with the mosaic output that achieved the best AUROC in the validation set, the results in the test set, compared to the validation set, did not suffer any relevant difference, suggesting that the mosaic output can lead to good generalization. In the test set, in the small false positive regime, the EfficientNet-B0 with mosaic shows performance equivalent to that of the best model without mosaic, ResNet-34. In other words, the introduction of mosaic output does not improve or degrade the false positive percentage. This suggests that the better AUROC values for mosaic output are not due to a false positive percentage reduction but appear because of the improvement of the true positive percentage.

2) MOSAIC LIMITATIONS

Although the introduction of the mosaic-like output allows obtaining good results in some situations and even obtain the best rectified validation AUROC, unfortunately, it is not always beneficial for reasons that are not yet very clear. In addition, the use of mosaic output brought in the additional cost of having to annotate all the images containing smoke. When it worked, as in the case of the EfficientNet-B0, it induced the neural network to ''focus its attention'' on the smoke regions, which is the behavior one wanted to obtain with the mosaic output. This conclusion is supported by a difference between rectified and unrectified AUROC results in validation, with mosaic, of only 0.013 for B0. This is the best case. Contrarily, in the no mosaic case, the smallest difference in AUROC between the rectified and unrectified cases is significantly larger, namely, 0.063. For EfficientNets B3, B4, and B5, the use of mosaic was detrimental; the rectified AUROC values when the mosaic is used are worse than when it is not. The worse rectified AUROC values for some EfficientNets with mosaic might have something to do with the increase in the total number of weights, but this is probably not the only factor since it does not explain the B5 case. Unfortunately, understanding exactly why the mosaic output may lead to worse results is outside the scope of the present article, but this issue will be addressed in future publications.

C. IS RESULT RECTIFICATION NECESSARY?

Looking at the results in TABLE 1, one might be tempted to use the classifier with best-unrectified AUROC, arguing that images with smoke would still generate an alarm. Then a human observer would check the image and see the plume. Under this approach, one would have selected the EfficientNet-B0 without mosaic output. Its unrectified AUROC in validation is 0.984, which is apparently much better than the 0.958 of the EfficientNet-B0 that has the best validation result with mosaic. However, when rectifying the AUROC of the former, it drops to 0.831, indicating that the neural network issues alarms due to something that, most likely, is not smoke. This has the potential of significantly increasing the amount of false positive alarms. Even if the amount of false positives is not too large with the current dataset, there is no assurance that smoke-free images will never have the characteristics that trigger a false positive alarm. It is also possible to argue that the neural network might be ''seeing something'' caused by the smoke plume that we cannot understand, which would not induce any false positive in the future. This might be the case, but to follow this assumption means falling into the largely criticized ''black box'' approach in which we cannot verify in any way what the neural network is doing.

D. RESULT RECTIFICATION IN THE FUTURE

The present article indicates that when no mosaic output is used but rectification is employed to assure that neural networks are focusing on the desired features, the true positive percentages and AUROC are smaller than in the case without rectification. This reduction opens the possibility that many scientific works reporting results for network classifiers without mosaic and rectification, which is the standard procedure up to now, may have their true positives induced by the wrong features. When this is the case, it creates the issue of knowing if these true positives can be considered as being correctly based or not. Finally, if the research community realizes that detection based on wrong features is common, then a discussion about the standard use of result rectification should begin.

VII. CONCLUSION

The present work has shown that neural networks may present a significant mismatch between the number of images where a plume is said to exist and the number of images where the neural network is genuinely ''focusing its attention'' on a smoke region. It has also shown that, in some cases, it is possible to help correct this mismatch by introducing a mosaic-like output of the neural network. This output is a multidimensional vector whose entries receive the value of one when the corresponding region of the image for the entry contains a smoke plume and the value zero otherwise. Unfortunately, the mosaic output does not always lead to better results. We suspect that this has to do with the total number of trainable weights composing the neural network, but further studies are required to confirm this.

The best overall neural network created is the EfficientNet-B0 configuration, which possesses the smallest number of trainable weights among all the neural networks tested. The AUROC value of 0.949, obtained with the test set, corresponds to true and false positive percentages of 85.3% and 3.1%, respectively. The EfficientNets provide better results than the ResNet architecture but only when the mosaic output is employed.

Finally, we would like to point out that we believe that the presented results are some of the most reliable in this subject obtained up to date. The reason for this belief is the use of a highly realistic dataset due to: 1) containing a large number of images; 2) that were collected under various weather conditions using surveillance towers and; 3) in 274 different days.

REFERENCES

[1] A. M. Fernandes, A. B. Utkin, A. V. Lavrov, and R. M. Vilar, ''Development of neural network committee machines for automatic forest fire detection using lidar,'' *Pattern Recognit.*, vol. 37, no. 10, pp. 2039–2047, Oct. 2004.

[2] A. M. Fernandes, A. B. Utkin, A. V. Lavrov, and R. M. Vilar, ''Design of committee machines for classification of single-wavelength lidar signals applied to early forest fire detection,'' *Pattern Recognit. Lett.*, vol. 26, no. 5, pp. 625–632, Apr. 2005.

[3] R. V. de Almeida and P. Vieira, ''Forest fire finder–DOAS application to long-range forest fire detection,'' *Atmos Meas. Tech.*, vol. 10, no. 6, pp. 2299–2311, 2017.

[4] A. A. A. Alkhatib, ''A review on forest fire detection techniques,'' *Int. J. Distrib. Sensor Netw.*, vol. 10, no. 3, Mar. 2014, Art. no. 597368, doi: 10.1155/2014/597368.

[5] M. A. Alwadi, ''Energy efficient wireless sensor networks based on machine learning,'' Ph.D. dissertation, Univ. Canberra, Bruce, ACT, Australia, 2015, doi: 10.26191/EENC-J011.

[6] P. Barmpoutis, P. Papaioannou, K. Dimitropoulos, and N. Grammalidis, ''A review on early forest fire detection systems using optical remote sensing,'' *Sensors*, vol. 20, no. 22, p. 6442, Nov. 2020, doi: 10.3390/s20226442.

[7] (Apr. 2021). *Signalert*. [Online]. Available: http://www.signalert.net/en/

[8] (Apr. 2021). *Fire Lookout*. [Online]. Available: https://firelookout.org/index.html

[9] (Apr. 2021). *IQ Firewatch*. [Online]. Available: https://www.iq-firewatch.com/

[10] (Apr. 2021). *SmokeD*. [Online]. Available: https://smokedsystem.com/

[11] S. Matthews, A. Sullivan, J. Gould, R. Hurley, P. Ellis, and J. Larmour, ''Field evaluation of two image-based wildland fire detection systems,'' *Fire Saf. J.*, vol. 47, pp. 54–61, Jan. 2012.

[12] K. He, X. Zhang, S. Ren, and J. Sun, ''Deep residual learning for image recognition,'' in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Las Vegas, NV, USA, Jun. 2016, pp. 770–778, doi: 10.1109/CVPR.2016.90.

[13] M. Tan and Q. V. Le, ''EfficientNet: Rethinking model scaling for convolutional neural networks,'' 2019, *arXiv:1905.11946*.

[14] M. Sandler, A. Howard, M. Zhu, A. Zhmoginov, and L.-C. Chen, ''MobileNetV2: Inverted residuals and linear bottlenecks,'' in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Salt Lake City, UT, USA, Jun. 2018, pp. 4510–4520, doi: 10.1109/CVPR.2018.00474.

[15] J. Bergstra, B. Komer, C. Eliasmith, D. Yamins, and D. D. Cox, ''Hyperopt: A Python library for model selection and hyperparameter optimization,'' *Comput. Sci. Discovery*, vol. 8, no. 1, 2015, Art. no. 014008, doi: 10.1088/1749-4699/8/1/014008.

[16] 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,'' *Int. J. Comput. Vis.*, vol. 128, no. 2, pp. 336–359, 2020, doi: 10.1007/s11263-019-01228-7.

[17] R. Xu, H. Lin, K. Lu, L. Cao, and Y. Liu, "A forest fire detection system based on ensemble learning," *Forests*, vol. 12, no. 2, p. 217, Feb. 2021, doi: 10.3390/f12020217.
[18] S. Frizzi, R. Kaabi, M. Bouchouicha, J.-M. Ginoux, E. Moreau, and F. Fnaiech, "Convolutional neural network for video fire and smoke detection," in *Proc. 42nd Annu. Conf. Ind. Electron. Soc.*, Florence, Italy, Oct. 2016, pp. 877–882, doi: 10.1109/IECON.2016.7793196.
[19] K. Govil, M. L. Welch, J. T. Ball, and C. R. Pennypacker, "Preliminary results from a wildfire detection system using deep learning on remote camera images," *Remote Sens.*, vol. 12, no. 1, p. 166, Jan. 2020, doi: 10.3390/rs12010166.
[20] S. P. Hohberg, "Wildfire smoke detection using convolutional neural networks," M.S. thesis, Dept. Math. Inform., Freie Univ., Berlin, Germany, Tech. Rep., 2015.
[21] Z. Yin, B. Wan, F. Yuan, X. Xia, and J. Shi, "A deep normalization and convolutional neural network for image smoke detection," *IEEE Access*, vol. 5, pp. 18429–18438, 2017, doi: 10.1109/ACCESS.2017.2747399.
[22] A. Gaur, A. Singh, A. Kumar, A. Kumar, and K. Kapoor, "Video flame and smoke based fire detection algorithms: A literature review," *Fire Technol.*, vol. 56, no. 5, pp. 1943–1980, Sep. 2020, doi: 10.1007/s10694-020-00986-y.
[23] Y. Valikhujaev, A. Abdusalomov, and Y. I. Cho, "Automatic fire and smoke detection method for surveillance systems based on dilated CNNs," *Atmosphere*, vol. 11, no. 11, p. 1241, Nov. 2020, doi: 10.3390/atmos11111241.
[24] M. Park, D. Q. Tran, D. Jung, and S. Park, "Wildfire-detection method using DenseNet and CycleGAN data augmentation-based remote camera imagery," *Remote Sens.*, vol. 12, no. 22, p. 3715, Nov. 2020, doi: 10.3390/rs12223715.
[25] B. Zhou, A. Khosla, A. Lapedriza, A. Oliva, and A. Torralba, "Learning deep features for discriminative localization," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Las Vegas, NV, USA, Jun. 2016, pp. 2921–2929, doi: 10.1109/CVPR.2016.319.
[26] S. Bozinovski, "Reminder of the first paper on transfer learning in neural networks, 1976," *Informatica*, vol. 44, no. 3, pp. 291–302, Sep. 2020, doi: 10.31449/inf.v44i3.2828.
[27] O. Russakovsky, "ImageNet large scale visual recognition challenge," *Int. J. Comput. Vis.*, vol. 115, no. 3, pp. 211–252, Dec. 2015, doi: 10.1007/s11263-015-0816-y.
[28] C. Shorten and T. M. Khoshgoftaar, "A survey on image data augmentation for deep learning," *J. Big Data*, vol. 6, p. 60, Jul. 2019, doi: 10.1186/s40537-019-0197-0.
[29] J. Bergstra, R. Bardenet, Y. Bengio, and B. Kegl, "Algorithms for hyper-parameter optimization," in *Proc. 25th Int. Conf. Neural Inf. Process. Syst.*, 2011, pp. 2546–2554.
[30] T. Dozat, "Incorporating nesterov momentum into adam," Stanford Univ., Stanford, CA, USA, Tech. Rep., 2016.
[31] AI For Mankind and HPWREN. (Jan. 2021). *Open Wildfire Smoke Datasets*. Accessed: Oct. 28, 2021. [Online]. Available: https://github.com/aiformankind/wildfire-smoke-dataset
[32] C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens, and Z. Wojna, "Rethinking the inception architecture for computer vision," 2015, *arXiv:1512.00567*.
[33] AI For Mankind. (Aug. 28, 2020). *Wildfire Smoke Detection Research*. Accessed: Dec. 16, 2021. [Online]. Available: https://github.com/aiformankind/wildfire-smoke-detection-research
[34] *AI For Mankind*. Accessed: Dec. 16, 2021. [Online]. Available: https://aiformankind.org/

ARMANDO M. FERNANDES was born in Barreiro, Portugal, in 1974. He received the degree in physics engineering from the Instituto Superior Técnico, Lisbon, Portugal, in 1997, and the Ph.D. degree in physics engineering entitled "Forests Fire Detection by Analysis of Backscattered Laser Radiation," from the Instituto Superior Técnico, in 2005.

He participated in 12 scientific projects with national and international funding and has been working in machine learning, since 2001, and hyperspectral spectroscopy, since 2009. He is currently with INOV, Lisbon, as the Principal Researcher in the project ResNetDetect that aims at creating an automatic wildfire detection system. He has one patent and published 31 articles in scientific journals indexed in Web of Science from Clarivate, nine related to wildfire detection.

Dr. Fernandes was part of a team that won the Portuguese BES Award for Innovation 2005 with work on wildfire detection.

ANDREI B. UTKIN (Senior Member, IEEE) received the M.Sc. (Hons.) and Ph.D. degrees in physics and mathematics from Leningrad (now St. Petersburg) State University, Russia, in 1982 and 1986, respectively, and the Ph.D. degree in physics from the Universidade Técnica de Lisboa, Portugal, in 2007. He is currently a Wave Physics and Engineering Group Coordinator with INOV—INESC Inovação, Lisbon, Portugal. His research interests include LIDAR and imaging systems, laser-induced fluorescence, spectroscopy, nonstationary electromagnetics, signal processing, and artificial intelligence. He is also a guest editor and a reviewer for different international journals and conferences.

PAULO CHAVES was born in Luanda, Angola, in 1969. He received the B.Sc. degree in electrical engineering from the Instituto Superior de Engenharia de Coimbra, Coimbra, Portugal, in 1992, and the degree in electrical engineering from the Faculdade de Ciências e Tecnologia, Universidade de Coimbra, Coimbra, in 1996. He is currently the Head of the Remote Monitoring and Electronics Department, INOV. His research interests include image processing, sensor instrumentation processing algorithms, EDGE, and FOG processing and positioning systems for intelligent transportation systems. Since 2010, he has been working on safety- and security-related projects. He has participated in more than 15 research projects for ESA, national, and European funding. He was a member of the CEN/TC-278 Standardization Committee at the national level.