Deep Learning Solution for Early Wildland Fire Detection and Suppression

Azarm Nowzad, Kurt Winter
IQ wireless GmbH, Berlin, Germany

Azarm Nowzad, Ralf Reulke
Department of computer science, Humboldt-Universität zu Berlin, Berlin, Germany

Abstract

The application of terrestrial-based video surveillance for early detection of wildfires has been a promising solution and research topic in recent years. A variety of algorithms for early fire detection have been researched, using video data of the surveillance areas. The majority of these Image-based techniques search the images for smoke traces and allow the discovery of a fire as soon as its smoke signature is visible, enabling a prompt intervention and an easy extinction.

Due to the scene complexity and the non-rigid characteristic of smoke, the reliability of the rule-based functions is not always satisfactory. Sufficient flexibility for processing uncertainties in different levels of image processing is essential in the image processing unit.

The process of extracting features automatically has brought us to the application of AI-based solutions. In this paper, faster Region-Based Convolutional Neural Networks (Faster R-CNN) is applied on the real scene smoke labelled dataset. Using a challenging test dataset, the application of the machine learning for the future of wildland fire detection is analyzed and discussed.

Keywords: AI-based Fire detection, early smoke detection, Faster R-CNN smoke detection, forest fire algorithm

Introduction

Natural wildfires play a fundamental role in many areas, especially due to the rising number of residential developments in the Wildland-Urban-Interfaces (WUI). Early detection of wildland fires can reduce the devastating damages caused by fires, and has been a major research topic for many years. Terrestrial-based video surveillance of forested areas is the most promising technology for an automated and reliable fire detection [1].
Since the effectiveness of any fire detection system is measured by its speed in reporting a fire to the authorities, image-based fire detection systems focus on smoke and prioritize early stage smoke detection reliability and real-time performance.

State-of-the-art detection algorithms extract temporal, spectral and spatial smoke features, as local and global features, from candidate regions in the images and recognize the presence of smoke as in [2],[3],[4]. The decision-making process is then applied to the extracted features to classify smoke incidences. Feature extraction techniques are mainly designed and implemented based on human knowledge. By combining different smoke features, the performance of the smoke detection algorithm can be enhanced and its false alarm rate can be reduced substantially.

The main question in designing feature-based algorithms is if well-defined empirical-based features are sufficient to adapt the system to different environmental conditions.

To extract features automatically and also increase the complex features, deep Neural Networks are applied and examined using a variety of complex dataset.

Convolutional Neural Networks (CNNs) are a category of Neural Networks that have a proven high efficiency in areas such as image recognition and classification [5], but lack the capabilities of object detection and localization.

Region- Based Convolutional Neural Networks (RCNNs), developed by Ross Girschick et al. [6], are one of the first object localization and recognition algorithm family with the ability to extract high-level features. During the recent years, the network structure has been optimized by accelerating the object detection process, reducing the computational costs and enhancing the results.

For the purpose of smoke detection, the state-of-the-art Faster R-CNN network is selected among the three variants of RCNN, which is the pioneer on performance accuracy and speed.

**Faster R-CNN network structure**

Models for object detection using RCNNs are generally based on three main steps: first, find candidate regions (region proposal), then extract features with the help of a CNN network and, finally, classify the objects.

There are a variety of region proposal methods available, some, such as Sliding Window [7], can be computationally very expensive. Other more advanced techniques attempt to segment similar pixels and form a region instead.
Further improvements are achieved by introducing the Selective Search method [8], which is a hierarchical, grouping-based segmentation. In all these methods, the output of the region proposal feeds to a classifier.

Ren et al. [9] expose this region proposal method as a bottleneck for overall processing time. They proposed the Region Proposal Network (RPN) as the next generation method that uses a shared fully connected network instead of any external algorithm for extracting regions. RPN presents a binary classification on each position. Fig. 1 shows the smoke detection framework using RPN.

An RPN uses fixed size bounding boxes, called anchors, which are defined on the feature map. The binary classification is assigned to an anchor based on the Intersection-Over-Union (IoU) overlap with the ground truth. Therefore, anchors play a very important role in the detection accuracy.

![Smoke detection framework with RPN.](image_url)

The second main step of the object detection models deals with the feature extraction using a CNN network. Since training a deep learning network requires a large data set and enormous resources, a pre-trained network can be repurposed as a basis for the detector.
The last convolutional layer of the pretrained network is passed to the RPN network. The output of this network goes through a ROI pooling layer. The final stage is a fully connected layer, consists of a classification and a regression layer, which predict the localization and classification losses of the candidate regions.

The difference between the predicted and actual output is calculated in backpropagation using a cost function. Gradient descent is applied to minimize the cost function by using many iterations and updating weights:

$$w = w - \alpha \nabla J(w)$$  \hspace{1cm} (Eq. 1)

where $w$ is the weight parameter, $\alpha$ is the learning rate and $\nabla J(w)$ is the gradient of the weight parameter $w$. This process is iterated for all the training examples until convergence. Learning rates are defined as steps to reach the minima of the loss function.

The choice of a pre-trained network and fine tuning of the hyper parameters are examined and set to maximize performance on our test set scenarios.

**Data preparation and Model Setup**

The image data set for the learning process of AI-system are provided by the “IQ FireWatch” sensors, a technology for the early detection of forest fires in wildland areas, producing large quantity of data on a daily basis. The image dataset has been labelled using an annotation tool. 80 % of the data is used for training phase and 20 % for the validation. The network is trained using TensorFlow API running on Ubuntu 18.04.

TensorFlow Object Detection API provides a variety of Faster RCNN architectures with pretrained weights. The models need to be adopted and configured for the finetuning process on own data set.

The pretrained Inception_V2 on COCO data set has been used as feature extractor, since the computational complexity and accuracy meets our requirements. The configuration of the parameters plays an important role on the accuracy of the network.

The first stage anchor generator feeds anchors with different scales and aspect ratios for each position in feature map. The form of an anchor is a critical factor in the first stage feature extraction. It assures the transfer of the ground truth in the network. The distribution of the ground truth data in terms of aspect ratio and square root of the area is considered to assign the anchor scale and aspect ratio (Fig. 2).

By analyzing the distribution, median values of the aspect ratio and square root of the areas for smoke incidences are 0.6 and 0.3 respectively. For each parameter 4 values are assigned and 16 different anchor boxes are created for the first stage anchor generator to slide over the image.
The batch size is one of the important hyper parameters in configuring the network. It defines the number of subset training dataset used for the estimation of the error in Gradient Descent. The size varies from 1 (Stochastic Gradient Descent) to the maximum number of training examples (Batch Gradient Descent).

Fig. 2. Histogram distribution of the aspect ratio (a) and area (b) of the ground truth training data for smoke and non-smoke labels.
A small batch speeds up learning but results in the learning process having a higher accuracy variance. A larger batch size, on the other side slows the learning but guarantees a convergence to a more stable model with lower variance in classification accuracy [10]. Instead of using the entire dataset, a fixed number of examples is used as mini batch in our configuration. The magnitude of the learning rate influences the progress of the training process. Too small rates stop the progress of training but large ones never converge.

Among all optimizer, Adam optimizer is an efficient stochastic optimization that assigns individual adaptive learning rates for different parameters using estimation of first and second moments of the gradients [11].

Final proposals may overlap and increase the redundancy. To reduce the number of overlapped detection boxes, non-maximum suppression (NMS) is applied.

**Experimental results and analysis**

The forest fire detection model is trained on 4K images of real scene forest fires with different scene diversity and complexity including weather conditions and water surfaces. A class of non-smoke incidences is defined and labeled, which includes many false positive detections from feature-based algorithms.

An objective function that shows the progress of training is the total loss function, as a combination of localization and classification losses (Fig. 3a). Mean average Precision (mAP) is contemplated as the average precision and is a typical metric for measuring the accuracy of the trained network. Fig. 3b shows the mAP for a typical IoU of 50 %.

The loss function decreases during the 20k steps of the training process. The performance of the model is progressive, since the values are decreasing during training.

mAP reaches a value of 30 for the evaluation. Compared to the model trained on the COCO data set, the performance goal has been reached. To improve the accuracy of the designed system, a sequence of 8 frames per position is analyzed and a final decision-making is reached by combining the results in using NMS.
Fig. 3. Total loss function (a) and mAP with 50% IOU (b) of the training process.

Fig. 4 illustrates the visual results of eight different detections with bounding boxes.

Scene variety, transparency, location and size of the smoke has been considered in the evaluation set. As is evident from the results, even small and semi-transparent smoke incidences are detected. On the last row, a case of missed smoke and false positive detection on water surface have been pointed out. Some complex scenarios are challenging even for human to be detected.
Fig. 4. Smoke detection results of trained faster RCNN network.

**Conclusion and outlook**

In this paper, a Faster RCNN approach has been developed and trained on a real smoke dataset for the purpose of real time smoke detection. A pretrained network is applied as feature extractor and hyper parameters are configured to achieve the high performance on dataset. Experimental results show a promising detection rate. Future work concerns with optimizing the network by analyzing the effect of different feature extractors with and without pretrained weights.

**References**


