



Smoke Objection Detection in Deep Learning for Real-Time Wildfire Environments Using Faster R-CNN

N. Nithya^{1*}  R. Priya² 

^{1,2}Vels Institute of Science, Technology and Advanced Studies, Chennai, India

^{1*}nithya18290@gmail.com

²priyaa.research@gmail.com

Abstract. The increasing frequency and severity of wildfires have highlighted the urgent need for intelligent, real-time monitoring systems. These systems are capable of detecting early signs of forest fire activity. This paper experiments with the real time datasets from various websites with deep learning techniques. The dataset is a real dataset that comprises UAV and CCTV imagery. Here the data geometric augmented dataset is passed into the model. In this paper, experimental analysis and evaluation are performed to predict and distinguish the regions containing smoke and non-smoke. The proposed approach here uses the deep learning model called Faster Region-Based Convolutional Neural Network (Faster R-CNN) architecture to accurately recognise and localise smoke related to environmental conditions. The model is trained and validated on the dataset, and it ensures a strong performance on the both aerial and ground-level perspectives. Evaluation metrics scale for these deep learning models includes precision, recall, f1-score and accuracy. They are used to find the model's performance for near real-time deployment in wildfire surveillance systems. The results of this model show the efficiency of the smoke object detection on early detection capabilities.

Keywords: Object Detection, Real time dataset, Faster R-CNN, Deep Learning.

1 Introduction

Forest fires have become increasing from day by day. These frequent activities are severe and it is mostly due to the effects of climate change, deforestation, and anthropogenic activities. They poses significant threats to ecosystems, human health, and infrastructure [1]. Rapid and accurate detection of forest fires is essential to reduce their adverse impacts by enabling timely intervention. However, traditional fire monitoring systems such as satellite-based observation platforms and ground-based thermal or optical sensors often suffer from delays, low spatial resolution, and limited adaptability to real time dynamic environments [2].

The current structure of object detection models frequently explores single-stage detectors for edge-device speed, but they can sometimes struggle with false positives when dealing with semi-transparent objects like smoke. So an alternative method for

© The Author(s) 2026

R. Vasanth Kumar Mehta et al. (eds.), *Proceedings of the International Conference on Intelligent Systems for a Sustainable Future (ISSF 2026)*, Atlantis Highlights in Intelligent Systems 16,

https://doi.org/10.2991/978-94-6239-693-7_63

this is using Faster R-CNN two-stage alternative. Highlighting is that pairing it with a ResNet50 V1 backbone provides deep, hierarchical feature extraction, making it highly accurate at on the actual smoke plumes from look-alikes from cloud and fog.

Recent advances in computer vision and deep learning have led to the emergence of intelligent, automated systems for visual wildfire detection using image-based. Among deep learning architectures, region-based convolutional neural networks (R-CNNs), particularly the Faster R-CNN model, have demonstrated strong performance in object detection tasks across complex and chaotic scenes [3].

In this paper, we use the technology deep learning framework for real-time forest fire object detection [4], it is very massive in size and they are collection of UAV and CCTV images capturing wildfire smoke and flame events across multiple geographic locations. The climatic environments like smog, fog and lightning prevent the traditional methods from correctly identifying the smoke areas. In order to attain high accuracy and prediction here we employ the Faster R-CNN deep learning model which identifies the place where smoke appears in the images. With a high rate of prediction on these models we can make timely efforts for reducing the risk.

2 Related Works

In the year 2022, Mukhriddin Mukhiddinov et.al [5] proposed a wildfire smoke detection system using UAV images and an optimised YOLOv5 model. The dataset used in their study was collected from Google, Kaggle, containing 6,000 images with 3,285 smoke and 2,715 non-smoke samples. The optimised YOLOv5 was designed to improve accuracy and to reduce false positives scores. By using UAV images in the model, it proved the efficient of identification between smoke from non-smoke environments.

S. N. Saydirasulovich et.al [6] (2023) presented a paper using the images with Unnamed Aerial Vehicle and the deep learning model YOLOv8 model. The dataset, for their work, was sourced from Google, Kaggle websites, which comprises of 6,000 images on smoke and non-smoke categories. After the model training and testing they found that it strengthened to improve detection precision, feature extraction, and real-time performance under different aerial conditions. By using the image UAV-based data, the system demonstrated strong learning and prediction on early smoke detection. The experiment drew attention on the effectiveness of advanced deep learning architectures in wildfire monitoring and rapid response in case of smoke triggers.

Y. Kim et al. [7] (2023) experimented their work on detecting the smoke in wildfire environments using UAV images and the deep learning model called YOLOv7. The authors also created their own dataset, consisting of 6,500 UAV images that include both smoke and non-smoke cases. After the training, validation and testing performances, they found that the YOLOv7 model shows a high accuracy in real time environments even in complex forest settings. The system was able to identify smoke as soon as possible with higher precision than the previous traditional methods.

Y. Al-Smadi et al. [8] (2023) analysed the smoke detection using the deep learning models YOLOv3 and YOLOv5. The datasets used for this work are the Kaggle wildfire smoke dataset which consists of 737 images. The results for this model highlight that the model YOLOv5 outperformed YOLOv3 in detection accuracy and processing speed when compared to YOLOv3. The experiment highlighted the optimisation of model parameters for small and imbalanced datasets. Overall, the findings on their work concluded with a valid effectiveness on lightweight YOLO models for real-time wildfire smoke detection.

C. Bahhar et al. [9] (2023) the authors proposed a hybrid wildfire detection method which combined a staged YOLO model with Ensemble CNN and ABi-LSTM. The dataset used for their work FLAME which comprises of 39,375 images. The model improved the accuracy by integrating spatial and temporal learning. Their system showed good predictions on detecting the reliability in video-based scenarios. The approach of their method resulted in a great performance for real time wildfire and smoke analysis.

In 2024, Kumar et al. [10] proposed a solution for the traditional problems on detecting wildfires using drones. The model used for this deep learning technique is DeepLabv3 architecture which analyzes the D-Fire dataset obtained from Kaggle web resources, which aims to identify and segment smoke before a fire spreads. The result of the model is computationally efficient and therefore supports real-time processing on the drones. It also proves that they can effectively implemented in all the remote regions without spending on heavy computational power..

S. Chaturvedi et.al [11] (2024) the authors used the deep learning model called hybrid Convolutional Transformer which was used to predict and classify the early fire and smoke detection using the MODIS smoke dataset. The architecture of the deep learning model combined CNN's spatial feature extraction with Transformer based global attention for its improved sensitivity. Its feather light design leads to the path of faster inference with reduced computational cost. The approach proved that their work gives efficient results for large satellite based wildfire monitoring.

N. Ahmad et.al [12] (2025) the authors experimented with the deep learning model called FireNet-KD, a Swin Transformer model enhanced through multi-source knowledge distillation for wildfire detection. The authors trained the model with FLAME dataset comprising of 44,000 samples, it integrated multiple teacher networks to boost feature learning. The model achieved a good generalisation across diverse fire and non-fire scenes. The results of the model implies the accuracy for real-world wildfire detection tasks.

3 Methodology

The wildfire object detection in this experiment has undergone a set of procedures for finding the smoke plumes in the images. This study utilises the real time dataset image dataset, available on web resources. The dataset comprises approximately 29,500 instances and includes six key attributes: Image, Annotations, Image Name, Partner String Classes, Camera, and Date.

3.1 Architecture Description

To optimise computational efficiency and reduce the overall workload, this experiment adopts a batch processing approach. Batch processing ensures reduced time consumption, improved workflow management, and more efficient training.

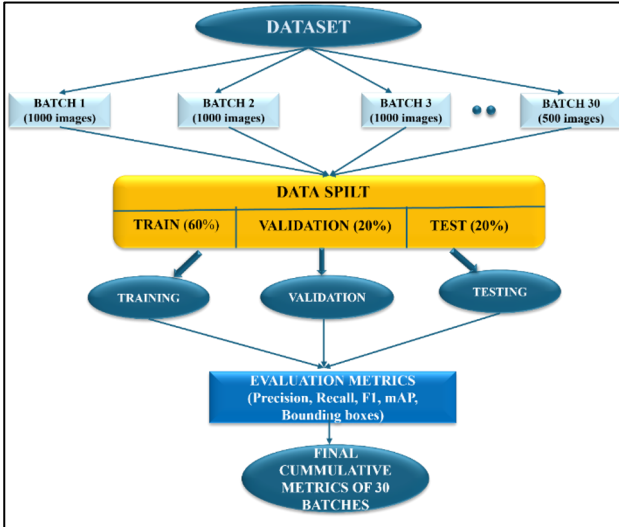


Fig. 1. Methodology Pipeline

Fig. 1 describes the workflow in batches. In this setup, the dataset is divided into 30 batches, each containing 1,000 images. Within each batch, the data is further split into 60% for training, 20% for validation, and 20% for testing. Every batch undergoes independent training (100 epochs), validation, and testing, with evaluation metrics recorded for each process. Finally, the results from all 30 batches are aggregated into a single cumulative evaluation metric, providing a comprehensive performance assessment for the entire dataset.

This work is done in the Google Colab environment with Python coding. This dataset has predefined annotation boxes as an attribute. When applying this annotation box in this model, they have to adopt the absolute pixel coordinates format. This involves scaling of YOLO annotations into absolute pixel format[13]. The general syntax form for the YOLO annotation for these boxes are given as:

$$\langle class_id \rangle \langle x_center \rangle \langle y_center \rangle \langle width \rangle \langle height \rangle \quad (1)$$

These annotations are transformed into Faster R-CNN syntax into the following as:

$$[x_min, y_min, x_max, y_max] \quad (2)$$

The formulas for changing the annotations from YOLO to Faster R-CNN is given as:

$$x_{min} = (x_{center} - \frac{width}{2}) * image\ width \quad (3)$$

$$y_{min} = (y_{center} - \frac{height}{2}) * image\ height \quad (4)$$

$$x_{max} = (x_{center} + \frac{width}{2}) * image\ width \quad (5)$$

$$y_{max} = (y_{center} + \frac{height}{2}) * image\ height \quad (6)$$

These converted annotations may then be stored directly into the PyTorch Dataset class as bounding box and label dictionaries. Performing this process ensures the accurate preservation of the datasets for further processes.

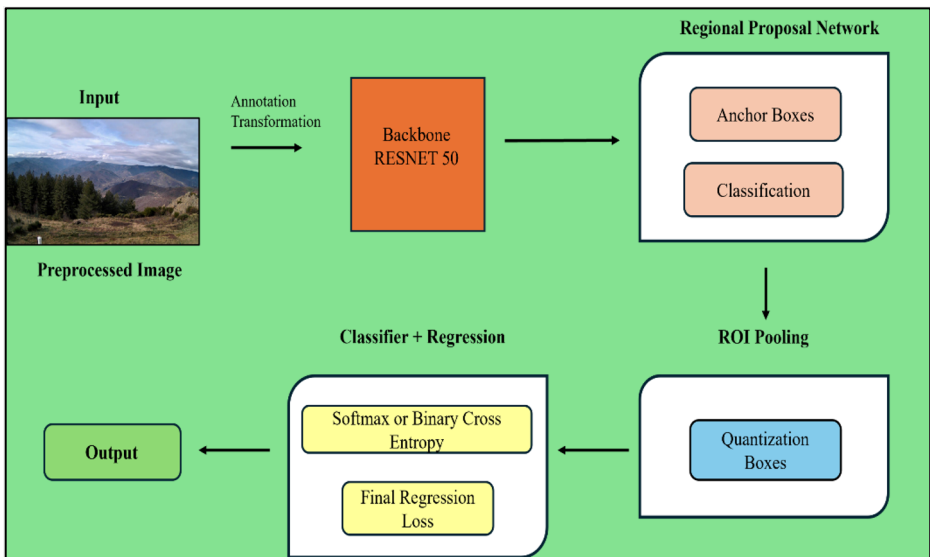


Fig. 2. Framework of Faster R-CNN in Wildfire Object Detection

3.2 Regional Proposal Network

In the Regional Proposal Network (RPN), the anchors generation of interest is created on the image where it finds smokes or is related to smoke patterns [14]. Each anchor is a candidate bounding box that might contain an object called smoke or non-smoke. For each anchor, the anchor labelling is created. The anchors with the value $\text{IoU} \geq 0.7$ are considered as positive proposals. And anchors with the value $\text{IoU} \leq 0.3$ are considered as negative proposals. After this step comes the step called training the RPN. Here, the positive proposals are trained as positive anchors with objectness score value = 1. The negative proposals are trained as negative anchors with the value = 0. From all the created anchors, only the most promising candidates which has a high objectness score, good overlap with ground truth are passed to the next stage.

Each anchor of this layer is checked first to find whether it contains an object. The objectness score is predicted by the model by using a small classifier.

$$P = \sigma(W_c * F + b_c) \quad (7)$$

Where F is the feature and the symbol sigmoid takes the score between 0 and 1. After this process the RPN predicts the four offsets to adjust the anchor box.

$$(t_x, t_y, t_w, t_h) = W_x * F + b_r \quad (8)$$

These offsets shift the anchors centre and size. Finally the refined regional proposal box is computed by applying the offsets to the anchor.

$$x = x_a + t_x w_a, \quad y = y_a + t_h h_a \quad (9)$$

By performing this process it provides a sharper and more accurate proposal to the next layer of the model.

3.3 ROI Pooling

After the RPN, the image is transferred to this layer. In this layer, the region of Interest and pooling take the corresponding candidate regions(proposals from the RPN) and convert them into a fixed size(7×7 or 14×14) feature map. This process ensures that every region contributes uniformly, regardless of the original size of the bounding box [15]. This avoids bias toward larger or smaller bounding boxes. First the proposal coordinates are scaled to the feature map resolution using

$$x' = \frac{x}{s}, y' = \frac{y}{s} \quad (10)$$

Next the proposal is divided into equal pooling bins computed as

$$h_{\text{bin}} = \frac{k}{k}, \quad w_{\text{bin}} = \frac{w}{k} \quad (11)$$

Then the max pooling is applied inside each bin to extract the strongest activation using

$$\text{Output}(i, j) = \max_{\text{bin}(i, j)} F(x', y') \quad (12)$$

After the above process the pooled values are arranged to form a fixed $k \times k$ feature map for each proposal. This fixed size representation helps the next layers to perform classification and bounding box regression consistently.

3.4 Classifier and Regression

The classifier Head is used for the classification task between the smoke and the background. The purpose of this classification is to decide what objects fall in the smoke class and what backgrounds fall under the non smoke class. In this study, we use a binary classification method, with a score for smoke as 1 and background as 0. Here, softmax is used to convert the classifier logits into probabilities. And next comes the regression head which is used for finding the bounding box on the test dataset. It also predicts how to modify the anchor box to fit the object. Here, the linear layer output technique is used to smooth bounding boxes.

Here the anchors which are generated by the layer is coarse, so to smooth this the regression head approximates the coarse values into a fine image parameters (d_x, d_y, d_w, d_h) . Here d_x and d_y represent the box centre, whereas d_w and d_h represent the width and height. To refine the coarse anchor boxes the bounding box is employed. These anchors are defined as:

$$d_x = \frac{x^* - x_a}{w_a}, \quad d_y = \frac{y^* - y_a}{h_a}, \quad d_w = \log\left(\frac{w^*}{w_a}\right), \quad d_h = \log\left(\frac{h^*}{h_a}\right) \quad (13)$$

4 Performance Metrics

Generally, in the performance metrics evaluation on the object detection task, we use the following traditional metrics such as precision, recall, f1-score. The accuracy here plays an essential role in the model's assessment.

4.1 Precision

Precision is a scale which used to determine whether the smoke predictions made by the model are actually correct or not. It helps us understand whether the system is predicting too much of wrong predictions.

$$\text{Precision} = \frac{TP}{TP+FP} \quad (14)$$

where TP represents the number of true positives and FP denotes false positives [16]. In wildfire detection, the values which having higher precision indicates fewer false alarms, which is important for avoiding unnecessary firefighting responses.

4.2 Recall

Here the model's capacity is to detect all smoke instances in the images using this metric scale called Recall.

$$\text{Recall} = \frac{TP}{TP+FN} \quad (15)$$

where FN refers to false negatives. In real-world wildfire monitoring, a high recall is essential, as missed detections can delay response times and allow fires to spread uncontrollably [16]. Therefore, an optimal wildfire detection model seeks a balance between precision and recall to ensure both accuracy and sensitivity.

4.3 F1-Score

To evaluate the model accuracy in the model F1-score is used. It is a statistical scale metric that uses both precision and recall considerations. From these considerations the term harmonic mean is calculated between the false positives and false negatives [16]. It is mathematically given as:

$$F1 = 2 \times \frac{(Precision \times Recall)}{(Precision + Recall)} \quad (16)$$

4.4 Accuracy

This is a performance measure that finds how well the smoke and non-smoke images are predicted compared with the ground truth.

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \quad (17)$$

5 Results

The study of Faster R-CNN with the wildfire dataset with a ResNetV1 backbone demonstrates a highly efficient performance on the challenging dataset, resulting an mAP that reflects its capabilities to handle real world environments. It shows it handles noise better than models trained on synthetic data. While recent literature frequently highlights single-stage detectors like YOLO for their extreme processing speeds, our study emphasize that the two-stage region proposal approach is more reliable at minimizing dangerous false negatives. By successfully detecting unstructured, semi-transparent smoke plumes. Apart from these features the ResNet50 architecture allowed our system to successfully distinguish between early stage smoke from atmospheric conditions like fog or low hanging clouds which backbones often struggle with false positives. Ultimately, these results illustrate that the study shows a high precision and strong F1-scores for critical, early warning wildfire detection systems. Here, the results on wildfire detection display the outputs of the comparison of the Real time dataset dataset image with its ground truth and predicted truth. Also, the training, validation and mAP curves are taken into account for calculating the efficiency of the model built.

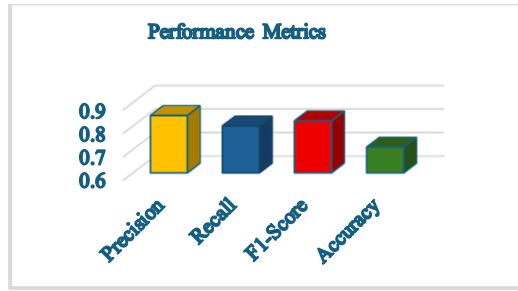


Fig. 3. Percentage of Metrics with real time dataset on Faster R-CNN

The Fig. 3 shows, the results show that our model is performing really good. With an accuracy of 71%, it is clear the system correctly predicting the smoke patterns. The right call most of the time. The precision score shows that when the model identifies smoke patterns, it is almost always correct, while the recall confirms that it isn't missing many targets. And F1-score is also high, we can see there is a good balance in the model.

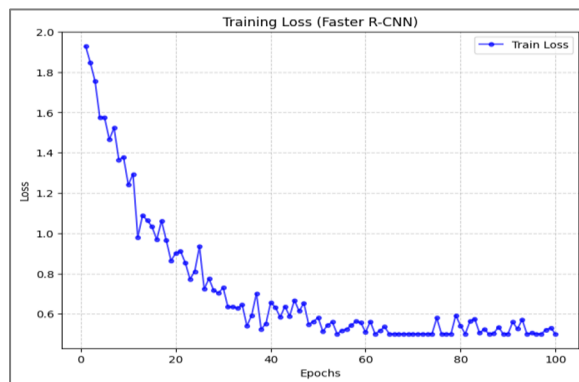


Fig. 4. Graph of the Training Loss Curve

From the Fig. 4, the loss curve display the training curve of object detection model over 100 epochs. Initially, the model begins with a high loss value of around 1.9, indicating early-stage learning with notable errors. As training increase, the loss gradually decreases, reflecting improved model optimization and reduced prediction errors. By the final epoch, the loss freeze near 0.3, displaying a good performance.

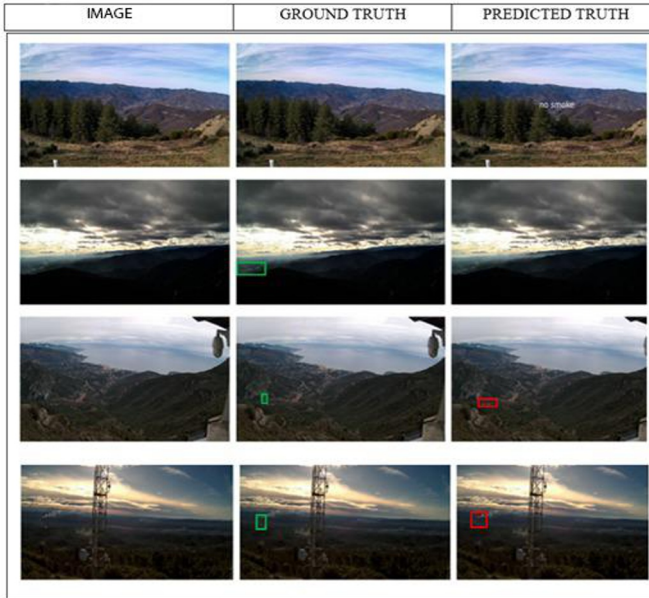


Fig. 5. Qualitative Analysis on Ground truth and Predicted truth

The Fig. 5 compares the real time dataset wildfire image with its ground truth and the predicted truth. Here, the bounded boxes in the colour red show the area of smoke from the test set on wildfire detection.

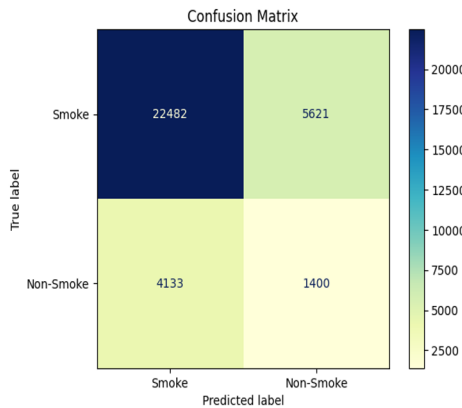


Fig. 6. Confusion matrix of Classification Performance on Real time dataset with Faster R-CNN

In Fig. 6, the confusion matrix indicates that 22,482 smoke images were correctly classified, while 5,621 were missed, and 4,133 non-smoke images were misclassified as smoke. This performance shows a reasonable balance between detection sensitivity and false alarm control in wildfire smoke recognition tasks.

6 Conclusion & Future Enhancements

This study concludes that this study confirms using Faster R-CNN with a ResNet50 V1 backbone is highly effective for detecting early signs of wildfire smoke. By testing the model on the real time images, we showed that our approach handles complex backgrounds much better than lighter, faster models. Our goal is to optimize this system so it can run efficiently on smaller devices while keeping the high accuracy we achieved here. In conclusion, the model accomplished a reliable performance in identifying smoke and non-smoke regions with a good accuracy of 71%. Also, the model shows a good generalisation capability, which shows the model does not fall under underfitting nor overfitting. The confusion matrix obtained from the model here indicates that the model can effectively detect most smoke instances, although there were a few misclassifications that still occur due to visual similarities. For future enhancement, the model can be improved by integrating other advanced augmentation methods to provide better efficiency and maturity, which helps in reducing false detections. Applying advanced multimodal architectures in the future will enhance the detection and robustness in complex real-time wildfire scenarios.

References

1. Alam, G.M.I., Tasnia, N., Biswas, T., Hossen, M.J., Tanim, S.A., Miah, M.S.U.: Real-Time Detection of Forest Fires Using FireNet-CNN and Explainable AI Techniques. *IEEE Access* 13, 51150–51181 (2025)
2. Redmon, J., Farhadi, A.: YOLOv3: An Incremental Improvement. *arXiv preprint arXiv:1804.02767* (2018)
3. Ren, S., He, K., Girshick, R., Sun, J.: Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 39(6), 1137–1149 (2017)
4. Lostonlen, M., Isla, N., Guillen, J., Veith, F., Buc, C., Barriere, V.: Scrapping the Web for Early Wildfire Detection: A New Annotated Dataset of Images and Videos of Smoke Plumes In-the-wild. *arXiv preprint arXiv:2402.05349* (2024)
5. Mukhiddinov, M.: A Wildfire Smoke Detection System Using Unmanned Aerial Vehicle Images Based on the Optimised YOLOv5. *Sensors* 22(23), 9384 (2022)
6. Saydirasulovich, S.N., Mukhiddinov, M., Djuraev, O., Abdusalomov, A., Cho, Y.I.: An Improved Wildfire Smoke Detection Based on YOLOv8 and UAV Images. *Sensors* 23(20), 8374 (2023)
7. Kim, Y., et al.: Forest fire Smoke Detection Based on Deep Learning Approaches and Unmanned Aerial Vehicle Images. *Sensors* 23(12), 5702 (2023)
8. Al-Smadi, Y., Alauthman, M., Al-Qerem, A., Aldweesh, A., Quaddoura, R., Aburub, F., Mansour, K., Alhmiedat, T.: Early wildfire smoke detection using different YOLO models. *Machines* 11(2), 246 (2023)
9. Bahhar, C., Ksibi, A., et al.: Wildfire and smoke detection using Staged YOLO model and Ensemble CNN. *Electronics* 12(1), 228 (2023)
10. Kumar, A., Perrusquía, A., Al-Rubaye, S., Guo, W.: Wildfire and Smoke early detection for drone applications: A Light-weight Deep Learning Approach. *Engineering Applications of Artificial Intelligence* 136, 108977 (2024)
11. Chaturvedi, S.: Ultra-Lightweight Convolution-Transformer Network for early fire smoke detection. *Fire Ecology* 20, 83 (2024)

12. Ahmad, N.: FireNet-KD: Swin Transformer-Based Wildfire Detection with Multi-Source Knowledge Distillation. *Fire* 8(8), 295 (2025)
13. Elizaroshan, D., Raj Kumar, J.S.: Forest Wildfire Detection from Satellite Image Using Deep Learning. *International Journal of Advanced Research in Science, Communication and Technology (IJAR SCT)* 3(1), 578–585 (2023)
14. He, K., Zhang, X., Ren, S., Sun, J.: Spatial Pyramid Pooling in Deep Convolutional Networks for Visual Recognition. In: *European Conference on Computer Vision (ECCV)*, LNCS 8691, pp. 346–361. Springer (2014)
15. He, K., Gkioxari, G., Dollár, P., Girshick, R.: Mask R-CNN. In: *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, pp. 2980–2988 (2017)
16. Padilla, R., Netto, S.L., da Silva, E.A.B.: A Survey on Performance Metrics for Object-Detection Algorithms. In: *2020 International Conference on Systems, Signals and Image Processing (IWSSIP)*, pp. 237–242 (2020)

Open Access This chapter is licensed under the terms of the Creative Commons Attribution-NonCommercial 4.0 International License (<http://creativecommons.org/licenses/by-nc/4.0/>), which permits any noncommercial use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons license and indicate if changes were made.

The images or other third party material in this chapter are included in the chapter's Creative Commons license, unless indicated otherwise in a credit line to the material. If material is not included in the chapter's Creative Commons license and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder.

