An Efficient Backbone for Early Forest Fire Detection Based on Convolutional Neural Networks

Quy Quyen Hoang, Quy Lam Hoang, and Hoon Oh *

Department of Electrical, Electronic and Computer Engineering, University of Ulsan, Ulsan, South Korea
Email: hqquyen@gmail.com (Q.Q.H.), quylam925@gmail.com (Q.L.H.)
*Correspondence: hoonoh@ulsan.ac.kr (H.O.)

***Abstract*—Forest fires cause disastrous damage to both human life and ecosystem. Therefore, it is essential to detect forest fires in the early stage to reduce the damage. Convolutional Neural Networks (CNNs) are widely used for forest fire detection. This paper proposes a new backbone network for a CNN-based forest fire detection model. The proposed backbone network can detect the plumes of smoke well by decomposing the conventional convolution into depth-wise and coordinate ones to better extract information from objects that spread along the vertical dimension. Experimental results show that the proposed backbone network outperforms other popular ones by achieving a detection accuracy of up to 52.6 AP.**

***Keywords*—convolutional neural network, object detection, forest fire detection, backbone network, depth-wise convolution**

I. Introduction

The forest fire is a disaster that causes devastating damage to human life and serious losses to the economy and ecosystem [1]. Unfortunately, the fire remains largely undetected until it has spread over a large area, making extinguishing difficult and sometimes impossible. Therefore, it is essential to devise a method that can effectively detect forest fires in the early stages. This paper proposes a variant of a vision-based fire detection model that relies on Convolutional Neural Networks (CNNs) to reduce the time it takes to detect forest fire and to improve detection accuracy with low time complexity.

Various methods of fire detection have been proposed so far [2–4]. The early approaches primarily relied on the fire lookout towers with tools, such as Osborne Fire finder [5]. This approach is obviously inefficient because it is subject to human error. There have been other approaches using sensors to detect temperature, smoke, flames, etc. However, it is almost impossible to build a dense sensor network in a vast forest area [6]. Additionally, fire detection time may be delayed until the detection parameter values exceed the threshold. Recently, despite facing an imbalance between detection accuracy and computational efficiency, vision-based approaches using artificial intelligence have been attracting attention [7, 8].

Vision-based approaches can be divided into two categories: traditional approaches and CNN-based ones. The former approaches relied on image processing techniques to explore features of fire and smoke such as color, shape and motion. For instance, RGB [9], YCbCr [10], or Lab [11] models were based on chromatic and dynamic features to extract fire and smoke pixels. Zhang *et al.* [12] used wavelet and fast Fourier transform methods to analyze the contours of the fire area in videos. Foggia *et al.* [13] introduced a multi-expert framework that combines color, shape and motion properties to increase the performance of the system. One recent approach utilized a background subtraction and color segmentation mechanisms to detect regions containing motion [14]. These approaches may be suitable for devices with low computational power, such as drones or surveillance cameras; however, it may not be appropriate to use the same feature extraction algorithm for different forest fire scenarios. Additionally, these methods require a careful image pre-processing step to ensure detection accuracy.

Generally, the latter approaches performed better compared to the former ones. Sharma *et al.* [15] adapted existing backbone networks, VGG16 [16] and Resnet50 [17], to develop fire detection system. However, these backbones produced a large number of parameters. Three approaches [18–20] modified backbones based on popular classification networks such as SqueezeNet [21], GoogleNet [22], and MobileNetV2 [23], respectively. The modified backbones allowed the models to be implemented on low computational devices while limiting their accuracy. Recently, some approaches used a two-channel CNN [24] and an inception mechanism [25] to build backbones for fire detection. They both achieved good accuracy, but focused on detecting only overt or near-obvious fire after the fire reached a certain size.

In summary, the fire detection models discussed above tried to use popular backbone networks; however, they may not be suitable for forest fire detection because they do not take into account the specific characteristics of

Manuscript received April 6, 2023; revised May 20, 2023; accepted June 4, 2023.

forest fires. For example, it is important to recognize plumes of smoke that are actually caused by a forest fire because they can be considered an early sign of a fire. In addition, those models require high computational load.

This paper proposes a new backbone network for the early detection of forest fires with low computational load, that can serve as a feature extraction module. The proposed backbone network has two key features to detect a forest fire early. First, the backbone can extract multiple views from input data over different branches to form different receptive fields and enable multi-scale representations. Second, the backbone decomposes the conventional convolution into depth-wise and coordinate ones and can better extract information from objects that spread along the vertical dimension, such as plumes of smoke. According to experiments, it was shown that the proposed backbone allowed the model to achieve up to 52.6 Average Precision (AP), which far exceeds the accuracies of other popular backbones such as ResNet [17], Inception [22] and ConvNext [26].

The rest of the paper is organized as follows. Section II presents the background; Section III describes the proposed backbone architecture in detail; and Section IV presents experiment studies and analyzes results, and is followed by conclusion in Section V.

II. Background

A. Forest Fire Detection Model

The forest fire detection model consists of three modules, Backbone, Neck and Head, as shown in Fig. 1. The Backbone consists of one input and four hierarchical stages denoted as $S_1, \ldots, S_4$. A feature map at S_i is constructed from S_{i-1} while a feature map at S_1 is built by extracting features from input (image). In this way, the object information in a feature map becomes more abundant at the higher stages.

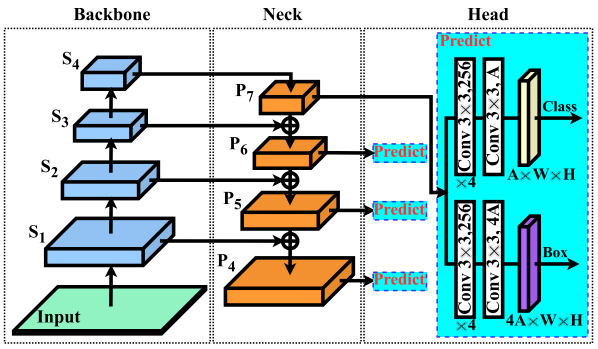

Figure 1. The architecture of the forest fire detection model.

The Neck consists of four levels denoted as $P_4, \ldots, P_7$. A feature map at P_7 is built by applying convolutions from the feature map at S_4 in the backbone. A feature map at P_i is constructed by up sampling the feature map in P_{i+1} and adding it to the corresponding feature map in the Backbone. In this way, the model can not only balance information in different feature maps, but also scales the variation of the objects to detect.

The Head includes two specific tasks: object classification and bounding box regression. Each one is represented as a small convolutional network that consists of five serially connected convolution layers (with 4 × "Conv 3×3, 256") whose output is either a class feature map represented by A × W × H (Anchors × Width × Height) or a box feature map by 4A × W × H, where 4 indicates 4 relative offset values between the anchor and the ground truth box. The upper branch of each prediction convolutional network within the Head determines the probability of the presence of a specific object at each spatial position and the lower branch regresses the offset from each anchor box to a nearby ground truth object.

In this paper, we focus on developing a new Backbone which is suitable for extracting features from the images of fires and smokes in the forest.

B. Motivation

Recent studies for fire detection usually use popular backbone networks which are designed for object classification on ImageNet [27] dataset to develop the fire detection model. This dataset does not contain the smoke fire class. Therefore, those backbones have not been trained and tuned to extract specific features for forest fires, thus limiting the accuracy. Besides, ImageNet is a huge dataset containing more than one million images with one thousand classes. The designers tried to increase the number of layers and use the large kernel in their backbone to extract more information. This results in a large number of parameters and thus requires large computational power. From this point of view, it seems that using the existing widely used backbone for forest fire detection models is not effective.

To address these problems, we propose a new backbone that has the following characteristics. First, one large kernel is replaced by many smaller kernels in order to allow effective feature extraction with fewer parameters. Second, the conventional convolution are decomposed into depth-wise and coordinate ones. The decomposition allows the module to better extract information about objects that spread along vertical dimensions, such as plumes of smoke. It also helps the model reduce the number of parameters. Third, the accuracy is improved by using residual structure and splitting technique. Finally, the model is trained and adjusted to get higher performance on forest fire dataset that contain images of both smoke and fire.

III. Proposed Method

A. Backbone Architecture

The overall structure of Backbone is shown in Fig. 2. The spatial dimension of the input is progressively reduced across stages. The input image is first processed by the stem block to quickly reduce spatial input without losing feature information. The stem uses three 3×3 kernels where each convolution has a stride size of 2, 1 and 1. It has the same effective receptive field as a single 7×7 kernel, but uses greater depth and fewer parameters. The use of three smaller kernels allows more information to be

extracted from the input image and makes the model easier to deploy on devices with less computational power. Batch normalization (*BN*) and rectified linear unit (*ReLU*) are additionally applied to the output of each convolution layer to speed up and stabilize the training process. At the end of the stem block, a 3×3 max pooling is applied to reduce the size of the feature map.

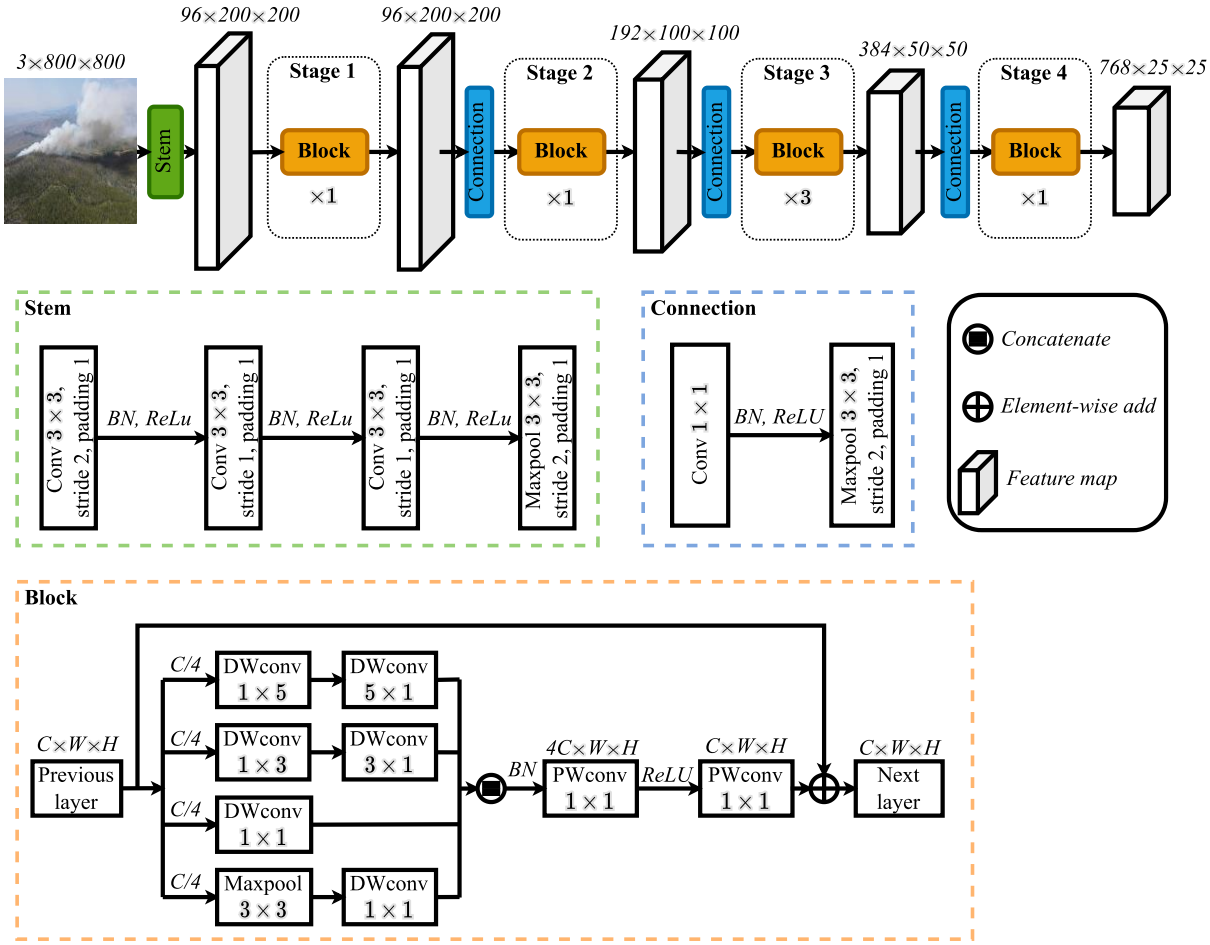

Figure 2. The proposed Backbone structure for forest fire detection.

The feature map generated from the stem block is comprehensively extracted through a hierarchical structure of four stages, each stage consisting of one or more blocks. Note that the third stage has three blocks. Each block is constructed using a residual structure that avoids the vanishing gradient problem and allows deeper layers to be obtained. The feature map from the previous layer is divided into four branches to make the model easy to extract features from multiple views. The four branches are treated by four different operators. The preceding two branches represent depth-wise convolutions (DWconv) and coordinate convolutions, indicating the decomposition of convolution operation. This decomposition can relax the computational complexity while still preserving the properties of convolution. The remaining two branches are used to blend information along spatial and channel dimensions.

Specifically, depending on the number of channels in the input, the feature map is split into four parts corresponding to four branches. The first branch and the second branch use the combination of one DWconv 1×5 and one DWconv 5×1, and the combination of one DWconv 1×3 and one DWconv 3×1, respectively. The splitting allows the module to better extract information, especially from the smoke plumes in which information spreads along the vertical dimension. Moreover, this technique helps the model reduce the number of parameters. The third branch uses one DWconv 1×1, and the last branch uses a combination of one max-pooling 3×3 and one DWconv 1×1. The outputs of all branches are concatenated along the channel dimension to produce a feature map. Then, two PWconv 1 × 1s are added serially to mix the information along channel dimension.

Two adjacent stages are connected by a connection block that consists of one max pooling 3×3 and one Conv 1×1. This reduces the size of the feature map by a half and doubles the number of channels.

A. *Neck and Head*

The Neck and Head in our model are inherited from RetinaNet [28]. In particular, the Neck adopts Feature Pyramid Network (FPN) which detects objects of different scales through different pyramid levels. A pyramid is constructed with four levels from 4 to 7, denoted as

$P_4, \ldots, P_7$, where each level has 256 channels. The Head requires a multi-task learning using two tasks referred to as object classification and bounding box regression. Each one is a small convolutional network attached to each FPN level whose output is either a class feature map represented by A × W × H (Anchors × Width × Height) or a box feature map by 4A × W × H, where A = 9 and 4 indicates 4 relative offset values between the anchor and the ground truth box. Details of the Neck and Head are referred to the paper [28].

B. Loss function

The focal loss (*FL*) function [28] is used in the model since it is suitable for smoke detection scenario where foreground and background classes are extremely imbalanced during training. $FL(p_t)$ as the focal loss function for classification score p_t, is expressed as follows:

$$FL(p_t) = -(1-p_t)^{\gamma} \log(p_t), \quad (1)$$

where $-(1-p_t)^{\gamma}$ is the modulating factor, with tunable focusing parameter $\gamma = 2$, and

$$p_t = \begin{cases} p & if\ y = 1 \\ 1-p & otherwise, \end{cases} \quad (2)$$

where $y \in \{\pm 1\}$ specifies the ground-truth class and $p \in [0, 1]$ is the model's estimated probability for the class with label $y = 1$.

From the papers [28, 29], we borrow the bounding box regression loss function, denoted by L_1, to measure difference between offsets and ground truth boxes. Then, the total loss, L_{total}, is expressed as a linear combination of $FL(p_t)$ and L_1:

$$L_{total} = \alpha FL(p_t) + \beta L_1, \quad (3)$$

where α and β are balancing terms. According to experiments in [28, 29], the optimal values of both α and β are given 1.

IV. Experiment

A. Dataset

The dataset with 4,350 images of forest fires is created from two sources, one from HPWREN Public Database [30] and another manually collected from the other sources on the Internet. Then, it is divided into a training set of 4,132 images and an evaluation set of 218 images. The data set appropriately considered a variety of forest fire scenarios by including forest fire images with a variety of fire intensity, time of day, and smoke shape.

B. Experimental Setup

The model was implemented by using the Python programming language and Pytorch framework. Then, it was trained and evaluated by using a computer equipped with a GeForce RTX 3060 GPU card. The training process went through 60 epochs with a batch size of 6. The learning rate was initialized with 2.5×10^{-3} and then was decreased by 10 times after 40 epochs and 100 times after 55 epochs.

The proposed backbone was compared with various existing backbones, including Resnet50 [17], ConvNext [26], VGG16 [16], EfficientNet [31], InceptionV1 [22], and InceptionV4 [32], using the same Head and Neck inherited from RetinaNet and with the same implementation settings such as number of epochs and learning rate for fair comparison.

C. Experimental Results

In this paper, we used three metrics to evaluate the performance of the models. The first metric is Average Precision (*AP*) that is commonly used to measure the accuracy of the object detection model. A higher AP score indicates better accuracy. The other two metrics are the Giga floating-point operations per second (*GFLOPs*) and the number of parameters (*#Parameters*), which are used to evaluate the computational complexity of the model.

Table I shows the performance comparison of our proposed model and various popular CNN models by using a forest fire dataset. Overall, our model achieved the best AP, AP^{50} and AP^{75} values while keeping good values in #Parameters and GFLOPS. Resnet50 as a default backbone used in RetinaNet, and VGG16 also achieved pretty good AP values; however, they required the higher #Parameters and GFLOPS. On the other hand, EfficientNet and InceptionV1 generated much fewer parameters, but were able to achieve considerably low APs, 44.0 and 41.2, respectively. It can be concluded that the proposed model is quite promising for forest fire detection by achieving the highest performance with low computational power.

TABLE I. Performance Comparison of the Proposed Backbone and the Other Backbones

Backbone	AP	AP^{50}	AP^{75}	#Parameters(M)	GFLOPS
Proposed	***52.6***	***86.0***	***51.4***	***18.52***	***192.71***
Resnet50	50.2	83.7	48.3	36.10	204.36
VGG16	49.7	83.7	48.8	142.93	431.82
Convnext	48.0	81.0	46.3	19.61	150.11
EfficientNet	44.0	70.9	42.0	14.58	35.75
InceptionV1	41.2	69.4	40.4	16.13	65.25
InceptionV4	41.0	66.4	40.3	52.92	190.43

The qualitative test results for forest fires are shown in Fig. 3 that includes 15 test images. The proposed model was able to detect various shapes of smokes and/or fires correctly regardless of daytime or nighttime. Moreover, the model could detect small smokes such as 7, 9, 13, and 14, blurred smokes such as 4, 13 and 14, and far-away smokes such as 2, 7, 9, 13, and 14, that are difficult for humans to discern. The detection of smoke implies that the model can detect a forest fire at an early stage.

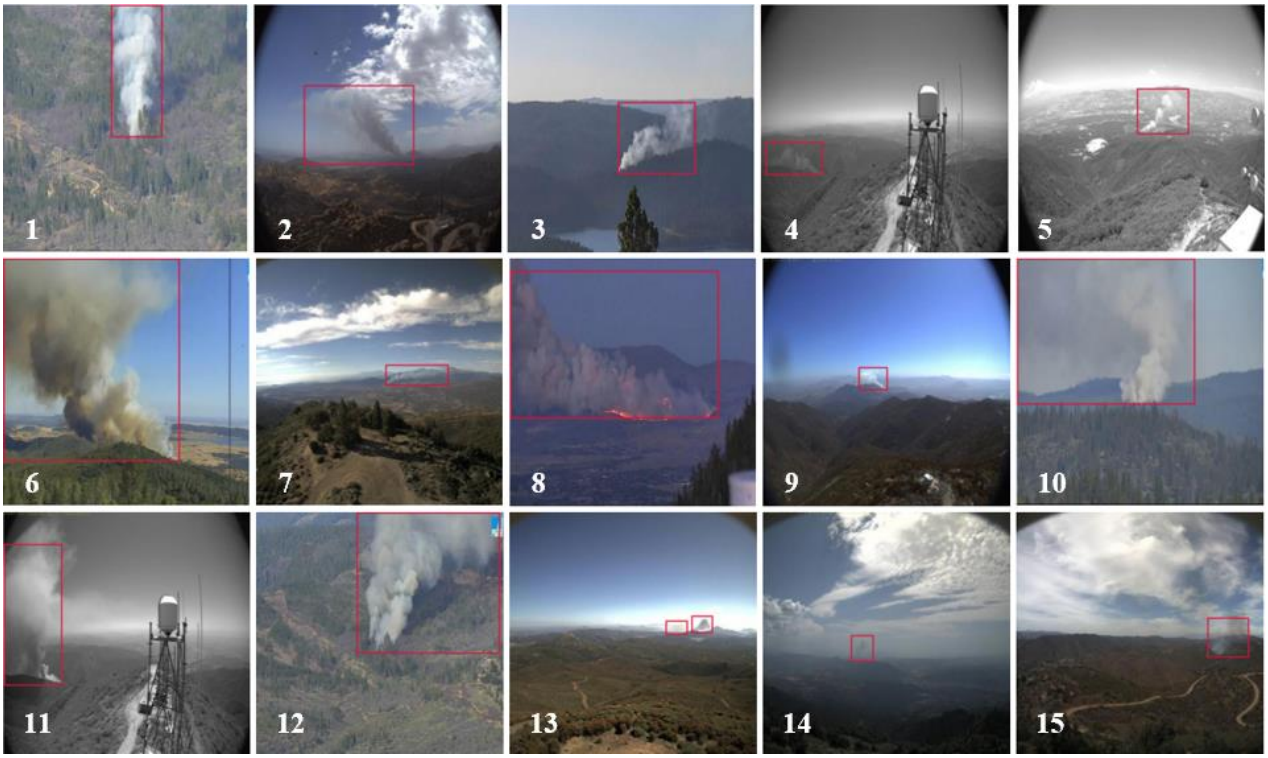

Figure 3. The qualitative results for forest fire detection on our dataset.

D. Ablation Study

We also conducted an ablation study to examine the effect of using splitting strategy, depth-wise and coordinate convolutions over the basic backbone. According to the results in Table II, when the additional techniques were used together, the proposed model not only improved the accuracy by 5.4% compared to the basic model, but also reduced the number of parameters by 34.3% and GFLOPS by 14.2%. This demonstrates the importance of the proposed additional techniques.

TABLE II. ABLATION STUDY ON THE BACKBONE MODULE WITH DIFFERENT TECHNIQUES

Basic	Splitting	Coordinate	Depth-wise	AP	#Parameters(M)	GFLOPS
√				49.9	28.21	224.49
√	√			50.7	20.93	200.60
√	√	√		51.5	19.72	196.62
√	√	√	√	52.6	18.52	192.71

V. CONCLUSION

This paper proposed a variant of a vision-based fire detection model that relies on CNN for early and efficient detection of forest fires. The proposed model focused on structuring the Backbone module newly while using the Neck and Head modules as they are. Specifically, we applied a splitting technique as well as the use of depth-wise and coordinate convolutions to efficiently detect different types of smoke from forest fires. The proposed model was evaluated using a dataset that contains 4,350 images of forest fires. According to the experiment results, the proposed forest fire detection model performed better than the existing models in terms of accuracy and computational cost reduction. This suggests the possibility of applying our early forest fire detection model to low-power embedded devices such as wildlife surveillance cameras.

CONFLICT OF INTEREST

The authors declare no conflict of interest.

AUTHOR CONTRIBUTIONS

Quy Quyen Hoang proposed and implemented the research idea. Quy Lam Hoang processed the experimental data and wrote the manuscript. Hoon Oh supervised the research and revised the manuscript. All authors read and approved the final version.

FUNDING

This result was partially supported by "Regional Innovation Strategy (RIS)" through the National Research Foundation of Korea (NRF) funded by the Ministry of Education (MOE) (2021RIS-003). It was also partially supported by Institute of Information & communication Technology Planning & Evaluation (IITP) grant funded by the Korea government (MSIT) (2020-0-00869, Development of 5G-based Shipbuilding & Marine Smart Communication Platform and Convergence Service).

REFERENCES

[1] Facts + Statistics: Wildfires. [Online]. Available: https://www.iii.org/fact-statistic/facts-statistics-wildfires
[2] V. Chowdary and M. K. Gupta, "Automatic forest fire detection and monitoring techniques: A survey," in *Proc. Intelligent*

Communication, Control and Devices, Springer, 2018, pp. 1111–1117.
[3] A. A. A. Alkhatib, "A review on forest fire detection techniques," *International Journal of Distributed Sensor Networks*, vol. 10, no. 3, 597368, 2014.
[4] P. Barmpoutis, P. Papaioannou, K. Dimitropoulos, and N. J. S. Grammalidis, "A review on early forest fire detection systems using optical remote sensing," *Sensors*, vol. 20, no. 22, 6442, 2020.
[5] History of the Osborne Firefinder. [Online]. Available: http://www.nysforestrangers.com/archives/osborne%20firefinder%20by%20kresek.pdf
[6] K. Bouabdellah, H. Noureddine, and S. Larbi, "Using wireless sensor networks for reliable forest fires detection," *Procedia Computer Science*, vol. 19, pp. 794–801, 2013.
[7] 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, 2020.
[8] A. Olugboja, Z. Wang, and Y. Sun, "Parallel convolutional neural networks for object detection," *Journal of Advances in Information Technology*, vol. 12, no. 4, pp. 279–286, November 2021. doi: 10.12720/jait.12.4.279-286
[9] T.-H. Chen, P.-H. Wu, and Y.-C. Chiou, "An early fire-detection method based on image processing," in *Proc. 2004 International Conference on Image Processing, ICIP'04*, IEEE, 2004, vol. 3, pp. 1707–1710.
[10] V. J. I. J. O. E. T. Vipin and A. Engineering, "Image processing based forest fire detection," *International Journal of Advanced Research in Engineering & Technology*, vol. 2, no. 2, pp. 87–95, 2012.
[11] C. Yuan, Z. Liu, and Y. Zhang, "UAV-based forest fire detection and tracking using image processing techniques," in *Proc. 2015 International Conference on Unmanned Aircraft Systems (ICUAS)*, IEEE, 2015, pp. 639–643,
[12] Z. Zhang, J. Zhao, D. Zhang, C. Qu, Y. Ke, and B. Cai, "Contour based forest fire detection using FFT and wavelet," in *Proc. 2008 International Conference on Computer Science and Software Engineering*, IEEE, 2008, vol. 1, pp. 760–763.
[13] P. Foggia, A. Saggese, and M. Vento, "Real-time fire detection for video-surveillance applications using a combination of experts based on color, shape, and motion," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 25, no. 9, pp. 1545–1556, Sept. 2015. doi: 10.1109/TCSVT.2015.2392531
[14] M. A. Mahmoud and H. Ren, "Forest fire detection using a rule-based image processing algorithm and temporal variation," *Mathematical Problems in Engineering*, vol. 2018, 7612487, 2018, https://doi.org/10.1155/2018/7612487
[15] J. Sharma, O.-C. Granmo, M. Goodwin, and J. T. Fidje, "Deep convolutional neural networks for fire detection in images," in *Proc. International Conference on Engineering Applications of Neural Networks*, Springer, 2017, pp. 183–193.
[16] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," arXiv:1409.1556v6 [cs.CV], 2014.
[17] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. the IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 770–778.
[18] K. Muhammad, J. Ahmad, Z. Lv, P. Bellavista, P. Yang, and S. W. Baik, "Efficient deep CNN-based fire detection and localization in video surveillance applications," *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. 49, no. 7, pp. 1419–1434, July 2019. doi: 10.1109/TSMC.2018.2830099
[19] K. Muhammad, J. Ahmad, I. Mehmood, S. Rho, and S. W. J. I. A. Baik, "Convolutional neural networks based fire detection in surveillance videos," *IEEE Access*, vol. 6, pp. 18174–18183, 2018. doi: 10.1109/ACCESS.2018.2812835
[20] K. Muhammad, S. Khan, M. Elhoseny, S. H. Ahmed, and S. W. Baik, "Efficient fire detection for uncertain surveillance environment," *IEEE Transactions on Industrial Informatics*, vol. 15, no. 5, pp. 3113–3122, May 2019. doi: 10.1109/TII.2019.2897594
[21] F. N. Iandola, S. Han, M. W. Moskewicz, K. Ashraf, W. J. Dally, and K. Keutzer, "SqueezeNet: AlexNet-level accuracy with 50x fewer parameters and< 0.5 MB model size," arXiv:1602.07360v4 [cs.CV], 2016.
[22] C. Szegedy *et al.*, "Going deeper with convolutions," in *Proc. the IEEE Conference on Computer Vision and Pattern Recognition*, 2015, pp. 1–9.
[23] M. Sandler, A. Howard, M. Zhu, A. Zhmoginov, and L.-C. Chen, "Mobilenetv2: Inverted residuals and linear bottlenecks," in *Proc. the IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 4510–4520.
[24] Y.-Q. Guo, G. Chen, Y.-N. Wang, X.-M. Zha, and Z.-D. Xu, "Wildfire identification based on an improved two-channel convolutional neural network," *Forests*, vol. 13, no. 8, p. 1302, 2022.
[25] D.-L. Nguyen, M. D. Putro, X.-T. Vo, T.-D. Tran, and K.-H. Jo, "Fire warning based on convolutional neural network and inception mechanism," in *Proc. 2022 International Conference on Multimedia Analysis and Pattern Recognition (MAPR)*, IEEE, 2022, pp. 1–5.
[26] Z. Liu, H. Mao, C.-Y. Wu, C. Feichtenhofer, T. Darrell, and S. Xie, "A convnet for the 2020s," in *Proc. the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022, pp. 11976–11986.
[27] O. Russakovsky *et al.*, "ImageNet large scale visual recognition challenge," *International Journal of Computer Vision*, vol. 115, no. 3, pp. 211–252, 2015.
[28] T.-Y. Lin, P. Goyal, R. Girshick, K. He, and P. Dollár, "Focal loss for dense object detection," in *Proc. the IEEE International Conference on Computer Vision*, 2017, pp. 2980–2988.
[29] S. Ren, K. He, R. Girshick, and J. Sun, "Faster R-CNN: Towards real-time object detection with region proposal networks," arXiv:1506.01497v3 [cs.CV], 2015.
[30] High performance wireless research and education network. [Online]. Available: http://hpwren.ucsd.edu/index.html
[31] M. Tan and Q. Le, "Efficientnet: Rethinking model scaling for convolutional neural networks," in *Proc. International Conference on Machine Learning*, 2019, pp. 6105–6114.
[32] C. Szegedy, S. Ioffe, V. Vanhoucke, and A. A. Alemi, "Inception-v4, inception-resnet and the impact of residual connections on learning," in *Proc. Thirty-First AAAI Conference on Artificial Intelligence*, 2017.