Fire Warning Based on Convolutional Neural Network and Inception Mechanism

Abstract—**Fire is a dangerous disaster that takes many lives and human property. Fire happens everywhere, especially in areas with high temperatures or hot sun. Fires can be caused by humans or by nature. Therefore, an early warning of fire is necessary to reduce the damage. Research in many different fields has long been focused on fire alerts. This paper proposes a fire alarm system based on a lightweight convolutional neural network. The design takes the advantage of convolution layers, depthwise separable convolution layers, inception module, and softmax function to optimize network parameters while ensuring feature extraction and classification. This network is trained and evaluated on FireNet dataset with an accuracy of 97.14%. In addition, this work also builds and implements the fire video testing systems on low-computation devices such as CPU-Based personal computer and embedded devices.**

Index Terms—**Convolutional neural network, fire classification, inception network, fire warning system.**

I. Introduction

Among natural and man-made disasters, fire is one of the major disasters affecting human life [1]. Fires tend to increase year by year, occurring everywhere, and in many areas such as factories, warehouses, houses, apartment buildings, amusement parks, and forests. The main cause of fires can be natural disasters (earth warming, volcanoes, thunder) or human causes (careless in daily life, electrical short, burning garbage, burning forests for farming). In order to minimize accidental damage from fire, many studies have focused on detecting and early warning of the fire source. These methods mainly rely on the operation of physical sensors such as temperature sensors, smoke sensors, and flame sensors [2]. However, the sensor-based methods can trigger false fire warning because it is not capable of distinguishing between fire and smoke. Besides, these methods have a certain delay when assessing the fire level to make warning decisions. From that analysis, this paper proposes a vision-based method for remote fire detection based on a lightweight convolutional neural network (CNN). This network is trained and evaluated on FireNet dataset, then tested on a real-time warning system without high latency.

The main contributions of this paper are shown as:

1 - Proposes a lightweight CNN architecture for fire classification. It comprised of feature extraction and classification modules.

2 - Applies the depthwise separable convolution layers in the inception module to reduce greatly the number of parameters.

3 - Designs a simple real-time fire warning system that deploys on the CPU and embedded device with high accuracy.

The remaining part of the paper is organized: Section II reviews the related works to fire detection methods and their strengths and weaknesses. Section III introduces the detail of the proposed approach. Section IV presents the experiment and analysis the results. Finally, Section V concludes the paper and future works.

II. Related Work

A. Traditional approach

Traditional machine learning approaches are greatly based on the color and motion of fire. The study in [3] proposes an RGB model based on chromatic and disorder measurement for fire-pixels and smoke-pixels extraction. [4] uses two different color models to distinguish between fire and smoke, then replaces heuristic rules with a fuzzy logic method to speed up the classifier's processing. In fire motion analysis, [5] uses wavelet analysis and disorder characteristics to detect fire and smoke. The authors in [6] apply the method based on the Lucas-Kanade optical flow algorithm to detect fire in the real-time video stream from a monocular camera. Similarly, the work in [7] proposes two novel optical flow estimators to overcome the insufficiencies of classical optical flow models when applied to fire detection. These approaches are easy to deploy with low computational and sensor devices. However, the similarity between the color and motion of the fire with the background image is a major obstacle, making the accuracy significantly less. On the other hand, it also requires careful image preprocessing to ensure fire detection accuracy.

B. CNN-based approach

In the field of computer vision, object detection is a hot topic of research with the popular application of convolutional neural networks. Fire detection is one such topic with a variety of proposed methods. The authors in [8] propose a method of forest fire detection with a fine-tuned CNN network. [9] apply two trained network models, VGG16 and Resnet50 to build a fire detection system in images. [10]–[13] refines various variations of popular classification networks (AlexNet, SqueezeNet, GoogleNet, and MobileNetV2) to build fire detection and surveillance systems. These approaches achieve high performance but produce a large size of network parameters and they are incompatible operate on CPU and embedded devices.

III. Proposed Methodology

The proposed fire classification network is described in Fig. 1. The network consists of two sub-modules: feature extraction and classification.

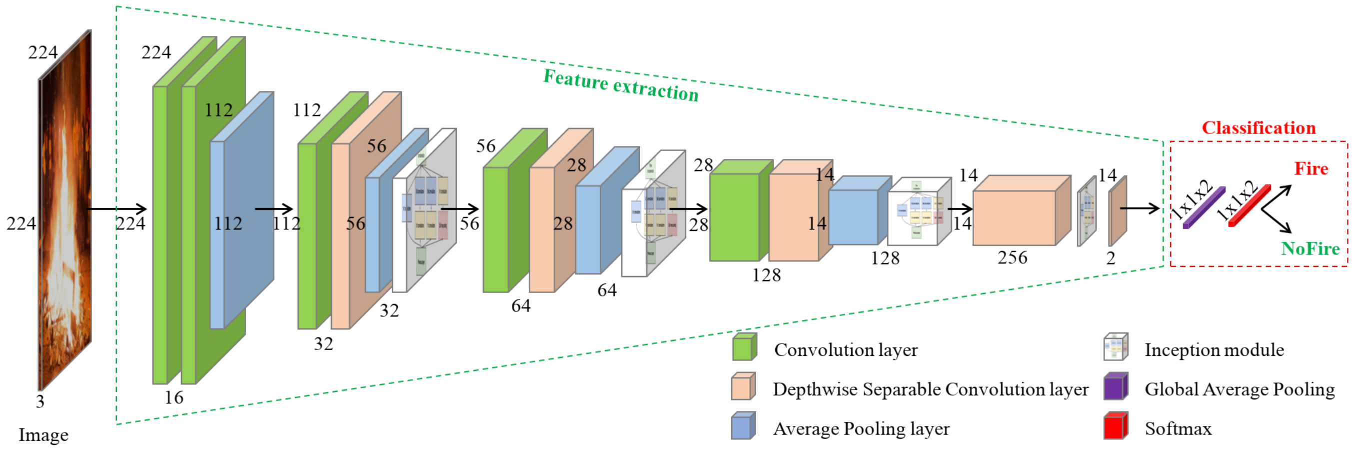

Fig. 1. The proposed fire classification network. This network consists of two sub-modules: feature extraction and classification.

A. Feature extraction module

Unlike traditional methods, CNNs are capable of extracting feature maps from raw images without any preprocessing. This makes it easier for later tasks to be applied including classification tasks. The feature extraction module in this paper focuses on exploiting the outstanding features of convolution layers, depthwise separable convolution layers (DWConv) [14], and inception mechanism [15] to optimize network parameters. Therefore, it can be deployed in low computing devices. Specifically, this module is designed based on five main blocks that are sequentially stacked according to the depth of the network. The blocks have similar structures except for the first and the last block. The first block uses two 7×7 convolution layers followed by batch normalization (BN), a ReLU activation function, and ends with a 3×3 average pooling layer. This block uses convolution layers with a large kernel size to increase the receptive field during the original raw image extraction. The input image after going through this block will be halved in size from 224×224 to 112×112.

The next three similarly structured blocks include a convolution layer, a DWConv layer, an average pooling layer, and an inception module. These three blocks act as intermediate-level feature extraction. The convolution layer in these blocks uses kernels varying from 5×5 to 3×3 while the average pooling layer keeps the same 3×3 kernel. The size kernels change by decreasing in order to optimize the network parameters. The inception module still follows the original architecture as shown in Fig. 2. However, the convolution layers are completely replaced by DWConv. Each inception module consists of four branches, each with a different structure. From bottom to top, the first branch uses only one $1 \times 1 \times 32$ DWCon, the second branch uses a combination of a $3 \times 3 \times 32$ max pooling layer and $1 \times 1 \times 32$ DWCon layer, the third branch uses a combination of two DWCon layers ($1 \times 1 \times 24$ and $3 \times 3 \times 32$), and the last branch is a combination of two DWCon ($1 \times 1 \times 24$ and $3 \times 3 \times 32$). The outputs of all branches are concatenated following the channel dimension

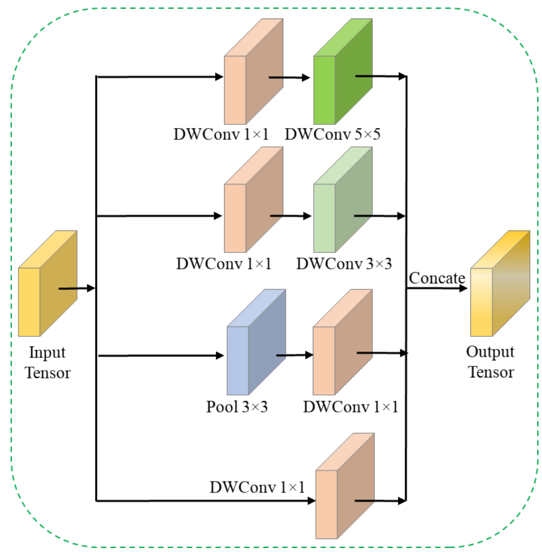

Fig. 2. The architecture of the inception module based on depthwise separable convolution.

to produce a fine-tuned feature map. This technique aims to enrich the amount of information on feature maps after fully extracted by the previous convolution. With this proposal, a large number of network parameters were reduced, but useful information is still ensured in each feature map level. This result is proved in the ablation study section. Through these three modules, the feature map further reduces the dimension to 14×14 and increases the number of channels to 128.

The last block in the feature extraction module composed of two DWConv layers interspersed is an inception module. This module further enriches feature information and reduces the dimension of the feature map to 14×14 with a channel number of 2.

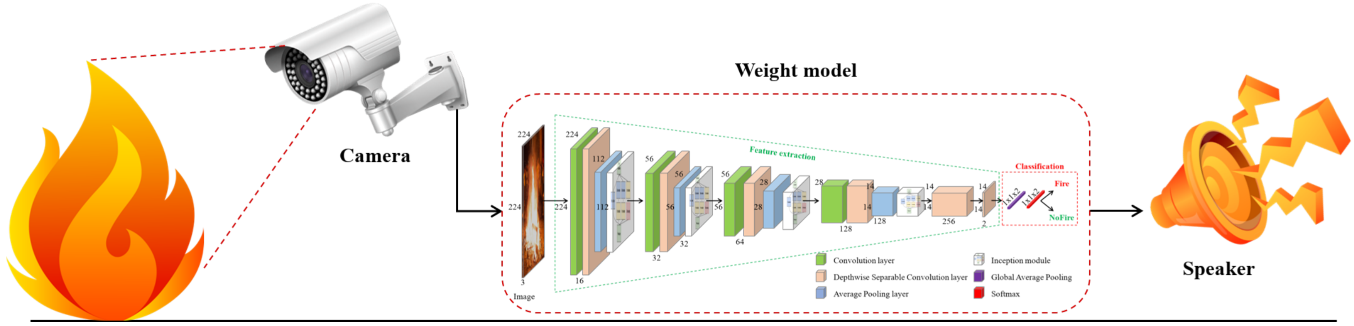

Fig. 3. The overall of fire video testing system. This system consists of camera, trained weight model, and speaker.

B. *Classification module*

In the popular classification networks so far, the fully connected layers are still used to perform the classification task in CNN. However, this technique increases a very large number of network parameters leading to limited applicability in low computing devices. To solve this problem, the classification module in this paper replaces all fully connected layers with a single global average pooling (GAP) layer. Accordingly, from the feature map, the output of the feature extraction module with size $14 \times 14 \times 2$ will be processed by GAP and generate a $1 \times 1 \times 2$ feature map. Then, a softmax function will be applied on this feature map to calculate the probability of each object appearing in the image corresponding to the classes in the dataset ($NoFire$ and $Fire$).

C. *Loss function*

The loss function is cross-entropy loss. It calculates the difference between the predicted value and the target value during training. This function is described in detail as follows:

$$L_{class} = -\sum_{c=0}^{1} p_c^{*}.log(p_c), \tag{1}$$

where c presents the index of a each class (0 to 1). p_c^t is the target indicator (0 or 1). p_c is the predicted probability from the proposed network. log denotes a natural logarithm function.

D. *Fire video testing system*

The overview of the fire video testing system is depicted in Fig. 3. The system is simply designed with three sub-modules: input, trained fire classification model, and output. The input is a set of YouTube videos recorded indoors and outdoors. The model was trained on the FireNet dataset. The output is the red text signal displayed on the screen and the warning sound to the speaker if the system detects any fire. This system is completely similar to the real-time testing system.

IV. EXPERIMENT

A. *Dataset*

The dataset used for training and evaluation in this experiment is a part of FireNet dataset [16]. This dataset contains 2,425 images including 1,124 fire images (Fire class) and 1,301 non-fire images (NoFire class). The images in the dataset are collected from many sources on the internet (Google, Flickr) and extracted from other datasets. The content of the photos is taken from many different environments and different contexts, it shows the diversity in fire disaster conditions. This work divides the dataset into a training set with 1,940 images (80%) and an evaluation set with 485 images (20%). To ensure classification accuracy and avoid overfitting, this experiment also applies several images augment techniques such as shift, random zoom, and random brightness.

B. *Experimental setup*

The proposed network is implemented using the Python programming language on the Keras framework. This network uses a GeForce GTX 1080Ti GPU for training and evaluation. For speed testing, it is also deployed on an Intel Core i7-4770 CPU @ 3.40 GH CPU (personal computer) and an Nvidia Maxwell GPU (Jetson Nano device). The training goes through 200 epochs and the batch size is 16. Adam optimization method is applied to the weight update process. The learning rate is initialized with 10^{-3} and then decreases after 10 epochs by a factor of 0.60 times if the accuracy is not improved.

C. *Experimental results*

The process of training and evaluating the proposed network is performed on the mentioned dataset. Besides, for a fair comparison with other popular classifier CNN networks, these networks are refined, then trained and evaluated on the same dataset. As a result in Table I, the proposed network achieved an accuracy of 97.14% with just over 400K network parameters. This result shows that it outperforms mobile networks (SqueezeNet, MobileNet, NASNetMobile) and even outperforms the best performing network VGG19 with 1.17% accuracy, but VGG19 is up to 50 times the proposed network parameter. The qualitative results of fire classification is presented in Fig. 4 and the ability to classify each class of the proposed network is shown in Fig. 5.

For safety reasons, this experiment only performed real-time system speed testing with fire simulation videos. In the system built as described in section III-D, the experiment

Fig. 4. The qualitative results of fire classification on FireNet dataset.

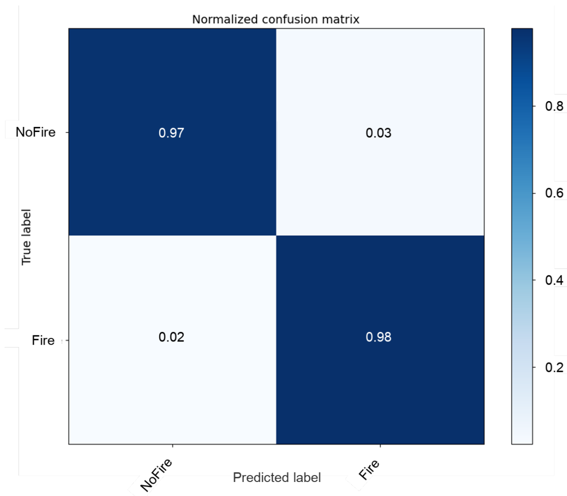

Fig. 5. The confusion matrix of proposed network.

TABLE I
COMPARISON RESULT OF FIRE CLASSIFICATION NETWORK WITH OTHER POPULAR NETWORKS. THE RED COLOR INDICATES THE BEST COMPETITOR.

Network	Parameters	Accuracy (%)
SqueezeNet	258,874	92.77
Proposed	**409,176**	**97.14**
MobileNetV2	3,571,778	93.95
MobileNetV1	4,288,714	91.93
NASNetMobile	5,362,334	90.42
DenseNet	8,097,354	95.13
VGG16	15,250,250	95.46
VGG19	20,559,946	**95.97**
Xception	22,969,906	95.97
InceptionV3	23,911,210	92.10
LeNet	78,432,080	89.41
AlexNet	833,965,362	93.28

achieved 26.67 frames per second (FPS) and 17.17 FPS on CPU-Based and Jetson Nano devices, respectively. Fig. 6 shows several sample results when experimenting on simulation videos. Through this, the experiment also shows that technical conditions such as image exposure, color similarity, and camera quality and resolution are factors that directly affect the accuracy and processing speed of the proposed system.

D. Ablation study

To evaluate the effect of the inception mechanism on the entire proposed network and DWConv on each inception module, this experiment also conducted several ablation studies. In ablation study 1, this work increases and decreases the number of inception modules according to the depth of the network and then compares the results. As shown in Table II, when the number of inception drops below four, the network parameters are of course greatly reduced, but the accuracy also decreases. The best accuracy reaches 96.30% with only one last inception of the network. In contrast, when increasing the number of inception to five, the network increased to 93,984 parameters but the accuracy decreased by 0.84%.

In ablation study 2, the convolution layers in the original inception module are replaced with depthwise separable convolution layers. The results in Table III show that this has improved the number of network parameters by 210,800 parameters and increased the accuracy by 1.17%. From the

Fig. 6. The result of the video testing system on CPU-Based personal computer.

TABLE II
ABLATION STUDY 1 ON DIFFERENT NUMBER OF INCEPTION MODULE. THE RED COLOR INDICATES THE BEST ARCHITECTURE.

Inceptions	Parameters	Accuracy (%)
0 (Stem)	215,768	95.46
1 (Last)	246,584	96.30
2	263,960	95.96
3	347,640	96.13
4 (Proposed)	**409,176**	**97.14**
5	503,160	96.30

above experiments, this paper chooses the number of inception modules to be four and replaces all convolution layers in each inception module with depthwise separable convolution layers to achieve the most balanced results in terms of network parameters and accuracy.

TABLE III
ABLATION STUDY 2 ON EFFECT OF DWCONV ON TO INCEPTION MODULE. THE RED COLOR INDICATES THE BEST ARCHITECTURE.

Inceptions	Parameters	Accuracy (%)
Conv	619,976	95.97
DWConv	409,176	**97.14**

V. CONCLUSION AND FUTURE WORK

This paper has proposed a compact convolutional neural network for fire classification that includes feature extraction and classification modules. The network is designed based on the advantages of the convolution layers, depthwise separable convolution layers, and inception network. The proposed network is trained and evaluated on FireNet dataset with high accuracy and negligible latency when tested in a video testing system. In the future, this fire classification network will be further developed to improve recognition even when the context is mixed between fire and smoke.

ACKNOWLEDGMENT

This work was supported by “Region Innovation Strategy (RIS)” through the National Research Foundation of Korea (NRF) funded by the Ministry of Education (MOE)(2021RIS-003).

REFERENCES

[1] M. Cavallini, M. Papagni, and F. Baruffaldi Preis, “Fire disasters in the twentieth century,” *Annals of burns and fire disasters*, vol. 20, pp. 101–3, 06 2007.

[2] F. Khan, Z. Xu, J. Sun, F. M. Khan, A. Ahmed, and Y. Zhao, “Recent advances in sensors for fire detection,” *Sensors*, vol. 22, no. 9, 2022.

[3] T.-H. Chen, P.-H. Wu, and Y.-C. Chiou, “An early fire-detection method based on image processing,” in *2004 International Conference on Image Processing, 2004. ICIP ’04.*, vol. 3, pp. 1707–1710 Vol. 3, 2004.

[4] T. Çelik, H. Özkaramanlı, and H. Demirel, “Fire and smoke detection without sensors: Image processing based approach,” in *2007 15th European Signal Processing Conference*, pp. 1794–1798, 2007.

[5] A. Rafiee, R. Dianat, M. Jamshidi, R. Tavakoli, and S. Abbaspour, “Fire and smoke detection using wavelet analysis and disorder characteristics,” in *2011 3rd International Conference on Computer Research and Development*, vol. 3, pp. 262–265, 2011.

[6] S. Rinsurongkawong, M. Ekpanyapong, and M. N. Dailey, “Fire detection for early fire alarm based on optical flow video processing,” in *2012 9th International Conference on Electrical Engineering/Electronics, Computer, Telecommunications and Information Technology*, pp. 1–4, 2012.

[7] M. Mueller, P. Karasev, I. Kolesov, and A. Tannenbaum, “Optical flow estimation for flame detection in videos,” *IEEE Transactions on Image Processing*, vol. 22, no. 7, pp. 2786–2797, 2013.

[8] Q. Zhang, J. Xu, L. Xu, and H. Guo, “Deep convolutional neural networks for forest fire detection,” in *Proceedings of the 2016 International Forum on Management, Education and Information Technology Application*, pp. 568–575, Atlantis Press, 2016/01.

[9] J. Sharma, O.-C. Granmo, M. Goodwin, and J. T. Fidje, “Deep convolutional neural networks for fire detection in images,” in *Engineering Applications of Neural Networks* (G. Boracchi, L. Iliadis, C. Jayne, and A. Likas, eds.), (Cham), pp. 183–193, Springer International Publishing, 2017.

[10] K. Muhammad, J. Ahmad, and S. Baik, “Early fire detection using convolutional neural networks during surveillance for effective disaster management,” *Neurocomputing*, 12 2017.

[11] 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, 2019.

[12] K. Muhammad, J. Ahmad, I. Mehmood, S. Rho, and S. W. Baik, “Convolutional neural networks based fire detection in surveillance videos,” *IEEE Access*, vol. 6, pp. 18174–18183, 2018.

[13] K. Muhammad, S. Khan, M. Elhoseny, S. Hassan Ahmed, and S. Wook Baik, “Efficient fire detection for uncertain surveillance environment,” *IEEE Transactions on Industrial Informatics*, vol. 15, no. 5, pp. 3113–3122, 2019.

[14] F. Chollet, “Xception: Deep learning with depthwise separable convolutions,” in *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 1800–1807, 2017.

[15] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. E. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich, “Going deeper with convolutions,” *CoRR*, vol. abs/1409.4842, 2014.

[16] A. Jadon, M. Omama, A. Varshney, M. S. Ansari, and R. Sharma, “Firenet: A specialized lightweight fire & smoke detection model for real-time iot applications,” *CoRR*, vol. abs/1905.11922, 2019.