International Journal of 3D Printing Technologies and Dijital Industry
Uluslararası 3B Yazıcı Teknolojileri ve Dijital Endüstri Dergisi

REAL TIME FIRE DETECTION USING FASTER R-CNN MODEL

H. Saygın Sucuoğlu[a], İsmail Böğrekci[a], Pınar Demircioğlu[a*]

[a]Aydın Adnan Menderes University, Faculty of Engineering, Mechanical Engineering Department, Aydın, TURKEY

[*]*Corresponding Author: pinar.demircioglu@adu.edu.tr*

(**Geliş/Received**: 22.10.2019; **Düzeltme/Revised**: 02.12.2019; **Kabul/Accepted**: 09.12.2019)

ABSTRACT

The aim of this study is to develop a real time fire detector using Faster R-CNN (Faster region-based convolutional neural network). For machine learning process of the system; 1,000 images (including fire and non-fire scenes) 80 and 20% for training and validation, respectively were used. The machine learning process was conducted using a system with the specifications of NVidia GeForce GTX 1070 Ti with 17 GB onboard memory. The required environments and tools (Python 3.5, Tensorflow 1.13.1, OpenCV, CUDA-cuDNN toolkits) were installed on the Anaconda virtual environment. The fire scenes on the images were labeled as fire and non-fire using LabelImg software. The metrics of the training process were obtained from the Tensorboard. The total loss value decreased from 2 to 0.02 with the steps of 40,000 at training. As the loss function was lower than the level of 0.05, inference graph was frozen and exported to detect the fire source. The developed real time fire detector model was tested in real time using lighter as fire source. In the test results; the 99% of accuracy was obtained using developed Faster R-CNN fire detector model.

Keywords: Faster R-CNN, Loss Function, Real time Fire Detection, Tensorflow and Tensorboard

1. INTRODUCTION

Firefighting is an extremely hard and dangerous task. After starting, it is almost impossible to control and it is extremely hard to recover the damaged area and loss lives. Therefore, the most efficient way for firefighting is to detect the fire source before it spreads and reaches the point of no return. For this reason, some early fire detection systems have been developed [1]. Luo & Su developed an intelligent security system for buildings [2]. The multiple sensors and fused sensory data were used in system in order to both detect the fire and generate reliable fire detection signal. Khoon et al. [3] developed an Autonomous firefighting mobile platform. It had capability to patrol and monitor the prescribed area and to search for the fire occurrence with flame sensors. Chang et al. [4] designed and manufactured a FSR (Fire searching robot) using task oriented design (TOD) methodology. Kim et al. [5] composed a portable fire evacuation guide system that can monitor indoor fires. Roberto et al. [6] projected a multi sensor data fusion technique for fire detection. The detection system was based upon temperature, luminosity and flame measurements. Kumar et al. [7] produced a gesture controlled robotic system with the capabilities of flame detection and fire extinguishment. Necsulescu et al. [8] developed an autonomous fire detection robotic system equipped with thermal infrared sensor to detect potential threats of fire and find out the source. Although; the proposed the sensor based fire detection systems can detect the fire sources, they are generally limited to indoor usages. They have also limitations according to location and type of sensors used in the system [9, 10]. Currently; the fire detection methods have been replaced from sensor based to computer vision based due to researches and developments in digital camera, image and video processing techniques. Image based fire detection systems can be categorized as computer vision, color model and machine learning. In the computer vision based systems; the fire source is detected using both color and characteristics of the fire extracted from the motion of flames [11]. The performance of these systems are better than that of the color based. However; as many heuristic characteristic values are used in these systems, it is too hard to apply them to the real environmental conditions. The color based systems generally use the models such as RGB (Red-Green and Blue) and HSI (Hue, Saturation and Intensity). The detection accuracy of these systems are too low since the color of the flames can change depending upon environmental conditions and burning material [12-14]. Machine learning based systems detect the fires using some developed algorithms such as Bayesian network, SVM (Support vector machine) and CNN (Convolutional neural networks). While former systems having high errors, the latter systems can significantly reduce the false detection rate and provide the best results [15].

Frizzi et al. [16] proposed a system using CNN for flame and smoke detection. They combined the convolution and max pooling. In this structure, an RGB color image was processed through two convolutional operations with the kernel size of 3x3. A max pooling with the stride value of two was added the convolutional layers of two and five. The output of the fully connected layers were connected to the layer of three to produce the class labels.

Zhong et al. [17] developed a fire detection system based on CNN in video sequence. In the structure, A RGB model based fire image was taken through three convolutional layers with the kernel size of 11x11, 5x5 and 3x3, respectively. Then, a window for max pooling with the size of 3x3 was adapted to structure to reduce the cost of the computing. In the fully connected layer stage, the dropout method was adopted to the system to reduce the over smoothing. 40,000 and 4,000 images were used for training and testing, respectively.

In this study; a fire detection model was studied and developed using Faster R-CNN (Faster region-based convolutional neural network). Totally 1,000 images (containing fire and non-fire scenes) 80% and 20% for training and validation, respectively from different scenes and environment were used to create structure of the model. A lighter was used as a fire source. The model was tested in real time using a webcam. Test results show that the fire can be detected at 99% of accuracy.

2. MATERIAL AND METHOD

2.1. Faster R-CNN

Faster RCNN is an object detection model developed by Ross Girshick, Shaoqing Ren, Kaiming He and Jian Sun in 2015. This architecture can be used for object detection with convolution neural network. Faster R-CNN is composed from three parts as convolution layers, region proposal network and classes & bounding box prediction.

CNN (Convolutional neural network) is one of the main thing to do image recognition and classification. CNN image classification takes an input image, process it and classify under certain categories. The input image is understood as an array of pixels. In CNN, each input image passes it through a series of convolution layers with filters (Kernals), Pooling, fully connected layers. Then it applies Softmax function to classify an object with probabilistic values between 0 and 1(Figure 1).

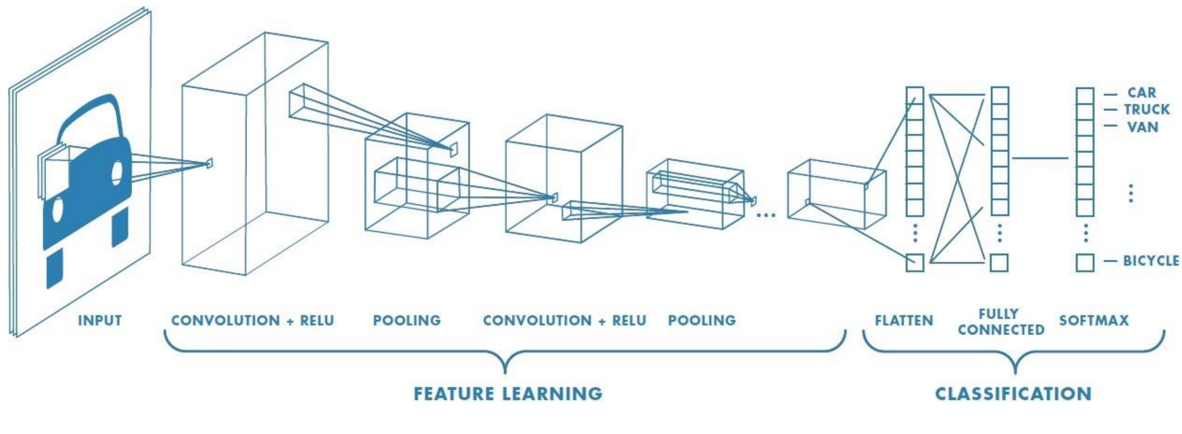

Figure 1. CNN with layers [18]

Convolution is the first footstep to extract features from an input image. Convolution holds the relationship between pixels via learning image features using small squares of input data. Pooling layers is used to reduce the number of parameters when the input images are too large. These layers can be different types as max pooling that takes the largest element from the rectified feature map, average pooling takes the average value and sum pooling used the sum of the values. In the fully connected layer; the matrix is flattened into the vector. The vector features are combined to create the new model. After all; an activation function e.g. the softmax is used to categorize the output [19].

RPN (Region proposal network) is the second component of the Faster R-CNN structure. RPN is a neural network sliding on the last feature map of the convolution layers. It predicts that there is an object or not and bounding boxes of the objects (Figure 2).

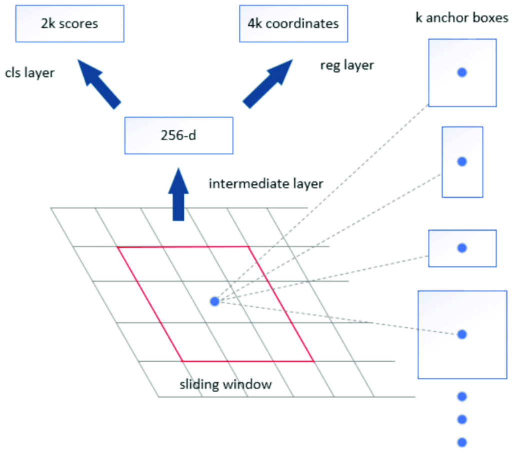

Figure 2. Region proposal network [20]

In the classes and bounding box prediction; fully connected neural network that obtained from RPN as input regions is used to predict the class of the object and bounding box.

2.2. Structure of the Fire Detector

The flowchart of the fire detection process with the developed fire detector was shown in Figure 3.

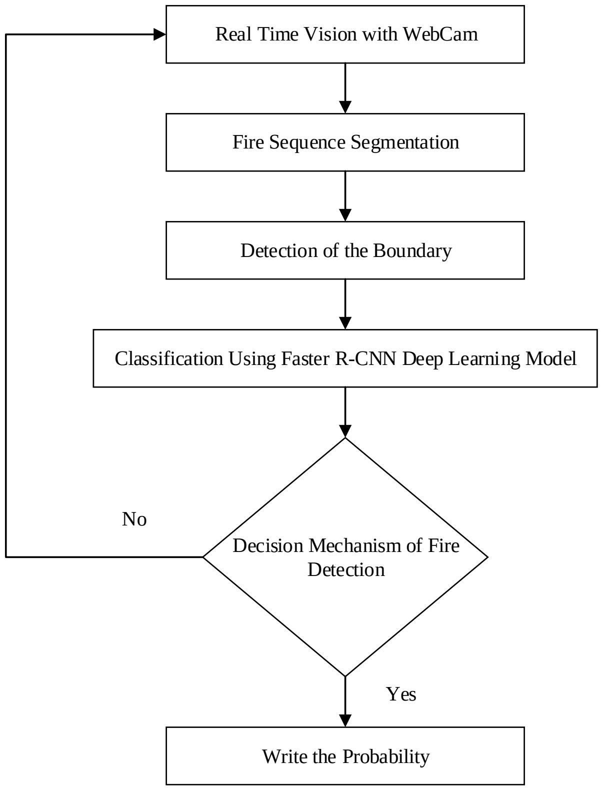

Figure 3. The flowchart of the fire detection process with fire detector.

According to the structure the real time vision was acquired via webcam. Then the fire sequence segmentation was applied to get the candidate of the fire source. In the next step, the boundary of the detected fire candidate was created. The classification applied with the model of Faster R-CNN deep learning model. The decision mechanism determined about it is a fire or not. Then, in the correct fire candidate the system determine the probability and put the score on the scene.

2.3. Experimental Setup and Dataset

The experiment was conducted using a system with the following specifications; NVidia GeForce GTX 1070 Ti with 17 GB onboard memory. The required environments and tools (Python 3.5, Tensorflow 1.13.1, OpenCV, CUDA-cuDNN toolkits) were installed on the Anaconda virtual environment. The fire scenes on the images were labeled as fire and non-fire using LabelImg software. 80% of the total 1,000 images were used for training and the 20% of the images were dedicated for test process. The selected images from the learning process were shown in Figure 4. The converter function selected as Softmax, number of steps for the learning was determined as 40,000 and the number as epochs was used as 3.

Figure 4. Selected images from training process

The Tensorboard visualization toolkit of the tensorflow can be used to view the progress of the training job (Figure 5). In this study, it was used to obtain the metrics (loss function and accuracy).

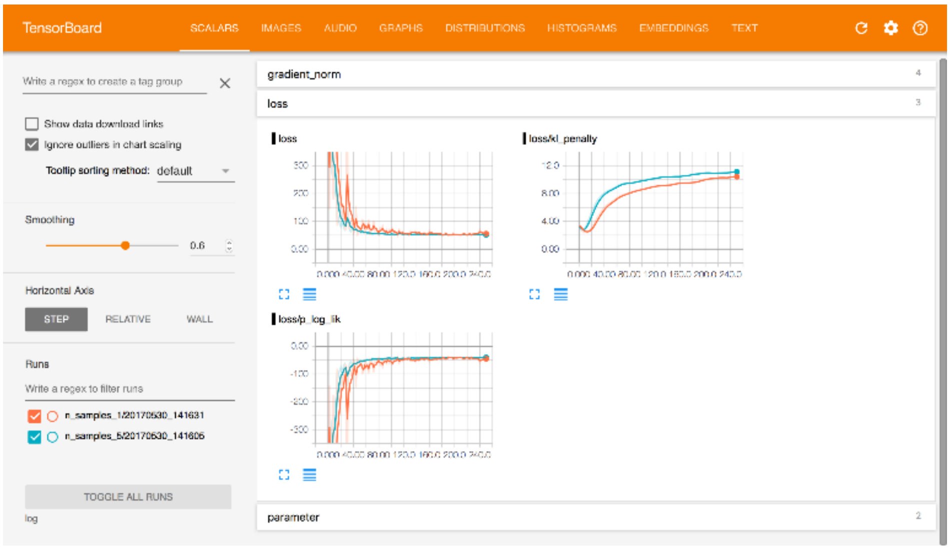

Figure 5. Tensorboard

3. RESULTS AND DISCUSSION

The obtained total loss graph from Tensorboard was given in Figure 6. X-axis shows the number of steps while Y-axis indicates the total loss.

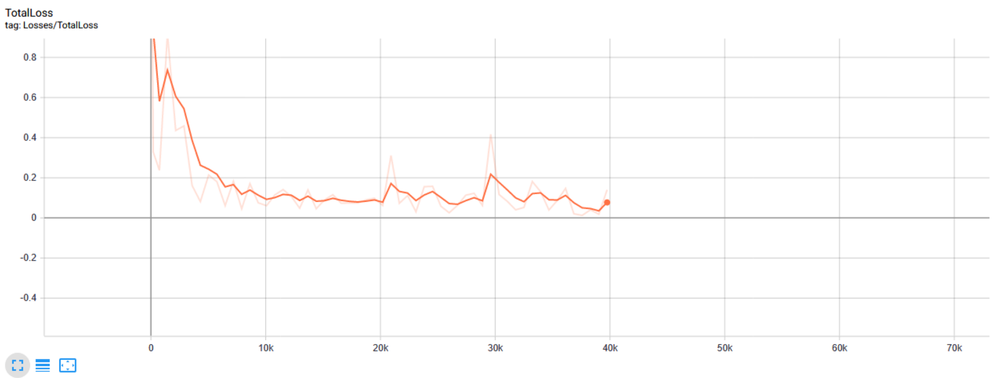

Figure 6. Total loss values in the training process

It was inferred from the graph that the while the total loss value was 2 at the starting of the training it was decreased to level of 0.02 at the end of the training at the steps of 40,000. As the loss function was lower than the level of 0.05, the inference graph was frozen and exported to detect the fire source. In the real time experiments, the lighter was used as fire source. The test scene images from real time fire detection test was shown in Figure 6.

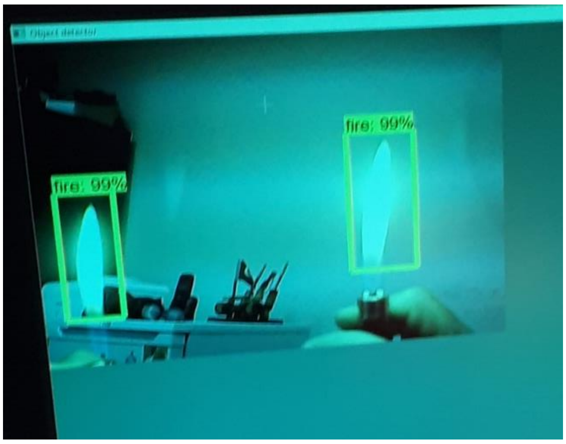

Figure 7. Fire detection test results

As shown in the Figure 7, the fire sources were classified as a fire with the detection score of 99%.

4. CONCLUSION

In this research study, a Fire Detector AI Model (FDAIM) was developed using the Faster R-CNN object detection model. The system was trained with 800 images containing fire and non-fire images. 200 images were used in validation. The metrics of the created model were obtained from the Tensorboard tool of the Tensorflow. The total loss value of the model was found as 0.02 with 40,000 number of steps and 3 epochs. After training process, the system was tested with lighter and the fire sources were classified with the score of 99%.

REFERENCES

1. Tan, C.F., Liew, S.M., Alkahari, M.R., Ranjit, S.S.S., Said, M.R., Chen W., Rauterberg, G.W.M., Sivarao D.S. “Fire Fighting Mobile Robot: State of the Art and Recent Development”, Australian Journal of Basic and Applied Sciences, Vol.10, Pages 220-230, 2013.

2. Luo, R.C., Su, K.L. “Autonomous Fire-Detection System Using Adaptive Sensory Fusion for Intelligent Security Robot”, IEEE/ASME Transactions on Mechatronics, Vol. 12, Pages 274-281, 2007.

3. Khoon, T.N., Sebastian, P., Saman, A.B.S. “Autonomous Fire Fighting Mobile Platform”, Procedia Engineering, Vol. 41, Pages 1145-1153, 2012.

4. Chang, P.H., Kang, Y.H., Cho, G.R., Kim, J.H., Jin, M., Lee, J. “Control Architecture Design for a Fire Searching Robot using Task Oriented Design Methodology”, SICE-ICASE International Joint Conference, IEEE, Pages 3126 - 3131, Busan, 2006.

5. Kim, Y. D., Kim, Y. G., Lee, S. H., Kang, J. H., An, J. “Portable Fire Evacuation Guide Robot System”, The 2009 IEEE/RSJ International Conference on Intelligent Robots and Systems October 11-15, St. Louis, USA, 2009.

6. Roberto, G.F., Branco, K.C., Machado, J.M., Pinto, A.R. “Local Data Fusion Algorithm for Fire Detection through Mobile Robot”, Test Workshop (LATW), 14th Latin American, 1-6, Cordoba, 2013.

7. Kumar, P.S, Ratheesh, B.R., Gobinath, B., Kumaran, K.M., Venkataraman, S. “Gesture Controlled Fire Extinguisher Robot with Audio and Video Capture”, IOSR Journal of Electronics and Communication Engineering, Pages 101-105, 2007.

8. Necsulescu, D. S., ur Rehman, A., & Sasiadek, J. “Fire detection robot navigation using modified voting logic”, In Informatics in Control Automation and Robotics (ICINCO), 2014 11th International Conference on IEEE, No. 1, Pages 140-146, 2014.

9. Frizzi, S., Kaabi, R., Bouchouicha, M., Ginoux, J. M., Moreau, E., Fnaiech, F. "Convolutional Neural Network for Video Fire and Smoke Detection", IECON 2016 - 42nd Annual Conference of the IEEE Industrial Electronics Society, Pages 877-882, 2016.

10. He, K., Zhang, X., Ren, S., Sun, J. "Deep residual learning for image recognition", In Proceedings of the IEEE conference on computer vision and pattern recognition, Pages 770-778, 2016.

11. Kim, Y. J., Kim, E. G. "A Study on Fire Detection Using Faster R-CNN and ResNet", Information, Vol.21, Issue 1, Pages173-180, 2018.

12. Celik, T., Demirel, H. "Fire detection in video sequences using a generic color model", Fire Safety Journal, Vol. 44, Pages 147-158, 2009.

13. Horng, W. B., Peng, J. W., Chen, C. Y. "A new image-based real-time flame detection method using color analysis", In Networking, Sensing and Control, Proceedings, 2005 IEEE, Pages 100-105, 2005.

14. Töreyin, B. U., Dedeoglu, Y., Güdükbay, U., Cetin, A. E. "Computer vision based method for real-time fire and flame detection", Pattern recognition letter, Vol.27, Pages 49-58, 2006.

15. Kim, Y. J., Kim, E. G. "Image based fire detection using convolutional neural network", Journal of the Korea Institute of Information and Communication Engineering, Vol. 20, Pages 1649-1656, 2016.

16. Frizzi, S., Kaabi, R., Bouchouicha, M., Ginoux, J. M., Moreau, E., Fnaiech, F. "Convolutional neural network for video fire and smoke detection", IECON 2016 - 42nd Annual Conference of the IEEE Industrial Electronics Society, (pp. 877-882). 2016.

17. Zhong, Z., Wang, M., Shi, Y., Gao, W. "A Convolutional Neural Network-Based Flame Detection Method in Video Sequence", Signal, Image and Video Processing, Vol.12, Issue 8, Pages1619-1627, 2018.

18. Understanding of Convolutional Neural Network (CNN) — Deep Learning, https://medium.com/@RaghavPrabhu/understanding-of-convolutional-neural-network-cnn-deep-learning-99760835f148 , October 21, 2019.

19. What Is a Convolutional Neural Network? https://ch.mathworks.com/solutions/deep-learning/convolutional-neural-network.html , October 21, 2019.

20. Faster RCNN Object detection, https://towardsdatascience.com/faster-rcnn-object-detection-f865e5ed7fc4 , October 21, 2019.