

딥러닝 기반의 프레임 유사성을 이용한 화재 오탐 검출 개선 연구

Fase Positive Fire Detection Improvement Research using the Frame Similarity Principal based on Deep Learning

이 영 학*, 심 재 창*

Yeung-Hak Lee*, Jae-Chnag Shim*

Abstract

Fire flame and smoke detection algorithm studies are challenging task in computer vision due to the variety of shapes, rapid spread and colors. The performance of a typical sensor based fire detection system is largely limited by environmental factors (indoor and fire locations). To solve this problem, a deep learning method is applied. Because it extracts the feature of the object using several methods, so that if a similar shape exists in the frame, it can be detected as false postive. This study proposes a new algorithm to reduce false positives by using frame similarity before using deep learning to decrease the false detection rate. Experimental results show that the fire detection performance is maintained and the false positives are reduced by applying the proposed method. It is confirmed that the proposed method has excellent false detection performance.

요 약

화염 및 연기 감지 알고리즘 연구는 다양한 모양, 빠른 확산 및 색상으로 인해 컴퓨터 비전에서 어려운 과제이다. 일반적인 센서 기반 화재 감지 시스템의 성능은 환경 요인(실내 및 화재발생 위치)에 따라 크게 제한된다. 이러한 문제를 해결하기 위해 딥러닝 방법을 적용하였으며, 이것은 물체의 형상을 특징으로 추출하므로 비슷한 형상이 프레임내에 존재하면 오탐으로 검출 될 수 있다. 본 연구는 화재 오탐 검출 개선을 위해 딥러닝 사용 전과 후에 프레임 유사성을 이용하여 오탐을 줄이는 새로운 알고리즘을 제안한다. 실험결과 제안된 방법을 적용하여 화재 검출 성능은 유지를 하면서 오탐 부분이 최소 30% 까지 감소하는 결과를 얻을 수 있었다. 제안된 방법의 오탐 검출 성능이 뛰어나다는 것을 확인하였다.

Key words : deep learning, faster r-cnn, fire detection, smoke detection, ssim

* Dept. of Computer Engineering, Andong National University

★ Corresponding author

E-mail: jcschim@andong.ac.kr, Tel : +82-54-820-5645

※ Acknowledgment

This work was supported by the National Research Foundation of Korea(NRF) grant funded by the Korea government (MSIT) (No. NRF-2018R1D1A1B07051087)

Manuscript received Mar. 13, 2019; revised Mar. 18, 2019; accepted Mar. 21, 2019.

This is an Open-Access article distributed under the terms of the Creative Commons Attribution Non-Commercial License(<http://creativecommons.org/licenses/by-nc/3.0>) which permits unrestricted non-commercial use, distribution, and reproduction in any medium, provided the original work is properly cited.

1. 서론

화재는 인위적이든 자연적이든 세상에서 가장 큰 재앙 중의 하나이다. 그 이유는 많은 경제적 손실을 야기시키며 때로는 소중한 생명을 앗아가기 때문이다. 특히, 자동 화재 감지 또는 모니터링에 대한 연구는 화재로 인한 사상자 및 재산 피해를 보호하기 위해 예전부터 주목을 받았다.

건물, 공장 및 실내 공간에서 사용되는 기존 화재 감지 시스템은 제한된 지역에 여러 개의 광학 및 이온 센서들(온도, 연기, 불꽃 등)을 사용한다. 전통적인 화재 감지 방법은 일반적으로 간단하고 저렴

하며 정확성 때문에 점(위치) 기반의 센서들이 적용되어 광범위하게 사용되어왔다. 센서들은 아주 근접되어 있는 연기와 화재에 의해 생성된 입자의 존재를 감지한다. 근접성을 요구하는 이온화에 의한 화학반응을 이용하는 것이다. 그리고 CCTV를 이용한 감시 체계를 사용하고 있다. 이 방법은 수십대의 카메라를 설치하여 중앙관제 시스템에서 여러 사람이 CCTV영상을 관찰하는 것이다. 센서 기반은 화재의 위치에 따라 센서의 반응이 늦을 수도 있으며, CCTV인 경우는 항상 많은 모니터링 요원이 감시를 하고있어야하는 단점이 있다.

IT 기술의 진화에 따라 컴퓨터 성능 향상 및 영상 분석 기술이 지난 10년간 크게 향상되었다. 컴퓨터 비전 기반의 화재 탐지 시스템은 기존의 방법들 보다 여러 가지 장점을 제공한다[1]. 지금까지 카메라를 사용하여 전통적인 문제를 해결하기 위해 비디오, 이미지, 색상, 텍스처, 기하학, 모션 그리고 신경망과 같은 다양한 접근방법이 사용되었다. Yamagishi and Yamaguchi [2], Celik [3], Chen [4]는 화염 영역을 구분하기 위해 색상 정보를 사용하였다. Liu and Ahuja [5]와 Yuan [6] 푸리에 계수 분석을 불꽃 윤곽선 추출에 적용하였다. Ugur [7]과 Dedeoglu [8]은 불꽃을 분석하기 위해 시간 및 공간 웨이블릿 방법을 제안하였다. Chen [4] 및 Çelik [9]은 화재의 불꽃과 연기를 인식하는 색상 모델을 확립하였다. 이러한 방법은 신경망, Adaboost, SVM 등의 분류를위한 기존의 알고리즘에 대한 입력으로 사용되는 다양한 특징 추출 알고리즘 또는 다차원 특성 벡터를 구축하는 것이다 [10, 11]. 따라서 기존의 비디오 기반 화재 탐지 시스템은 특징 추출을 위해 다양한 전문 기술들을 사용하였다.

화재 검출을 위한 또 다른 접근 방법은 특징 추출을 위해 특별한 알고리즘을 사용하지 않고 딥러닝 알고리즘을 사용하는 것이다. 딥러닝 알고리즘은 비디오 영상의 화재 및 연기를 감지하기 위해 데이터의 형상 혹은 표현을 학습하는 것이다. 인식율이 높은 CNN(Convolutional Neural Networks)은 비디오 영상에서 위상 특성을 추출할 수 있는 딥러닝 학습의 변형이다. 그러나 초기 화재 위치, 연기 전과 방향, 화재 크기, 화재 발생률 등에 대한 정보는 제공할 수 없다.

LeCun [12]이 제안한 CNN은 에러의 감쇠를 줄

였으며, Krizhevsky [13]는 CNN을 물체 인식에 적용하였다. CNN은 정규화된 값을 획득하기 위해 ReLU (Rectified Linear Unit), 활성화 함수(Activation Function) 그리고 Dropout을 이용하다[14]. 이러한 진보된 방법은 기존의 물체 인식방법을 획기적으로 향상 시켜왔다. 본 논문은 진보된 딥러닝 기반의 화재 검출에서 오탐을 줄이는 새로운 알고리즘을 제안하며 흐름도는 그림 1에 나타내었다.

본 논문에서 화재 검출을 위해 기본적으로 딥러닝 방법을 이용할 것이며, 오탐 제거를 위한 프레임 미분의 새로운 알고리즘을 적용하였다. 2장에서는 딥러닝에 프레임 비교 방법 대해 소개합니다. 3장에서는 실험 및 제안된 알고리즘 적용 결과를 기술하며, 4장에서 결론을 나타낸다.

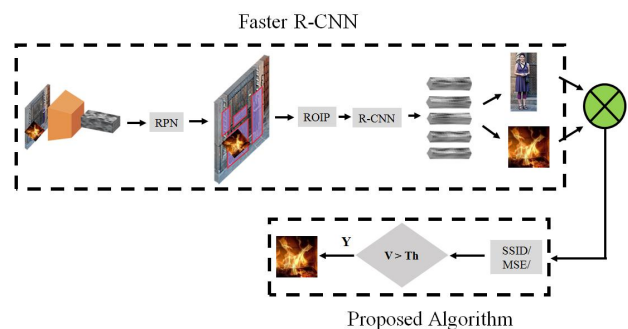


Fig. 1. The architecture of proposed algorithm. 그림 1. 제안된 알고리즘 구조도

II. 본론

1. Faster R-CNN

영상을 분류하는 것보다 영상 내에 있는 객체를 구별하는 것이 어려운 경우가 많다. 기본적으로 딥러닝의 R-CNN은 여러 단계를 거친다. 우선, R-CNN은 객체 존재가 가능한 영역에 대해 영역 선택(region proposal) 또는 바운딩 박스를 생성한다. 바운딩 박스들은 선택적 검색 알고리즘을 사용하여 찾는다. 이것은 비슷한 색상, 밝기, 패턴 등의 인접한 픽셀을 결합하여 찾는 방법이다. 두 번째 방법은 추출된 바운딩 박스를 CNN의 입력으로 사용하기 위해 크기의 단일화이다. 세 번째는 SVM을 사용하여 선택한 영역을 분류하는 것이다. 마지막으로, 분류된 객체의 바운딩 박스가 정확한 좌표를 설정하도록 선형 회귀 모델을 사용한다.

Faster R-CNN은 단순히 영역 제안을 생성하는

부분을 모델내에서 통합하는 영역제안 네트워크(RPN)라는 새로운 방법을 적용하는 방법이다. 이 방법은 객체를 탐지하기 위한 RPN 네트워크의 새로운 응용 프로그램이다. RPN의 역할은 입력 이미지에서 객체를 제안하는 부분의 사각형과 객체 스코어를 출력하는 것이다. 이것은 완전 연결된 네트워크이며 Faster R-CNN과 함께 convolutional 계층을 공유하도록 설계되었다. 훈련된 RPN은 영역제안 부분의 품질 향상과 물체 검출의 정확도를 향상시킨다. 일반적으로 Faster R-CNN은 외부의 느린 선택 검색(CPU에 의해 계산 됨)을 사용하지만, 내부의 빠른 RPN(GPU에 의해 계산 됨)을 사용하여 속도를 향상시킨다. RPN는 마지막 convolutional 계층 다음에 위치한다. 그 이후에 ROI Pooling, 분류 그리고 바운딩 박스 가 위치하며, 그림 2와 같다.

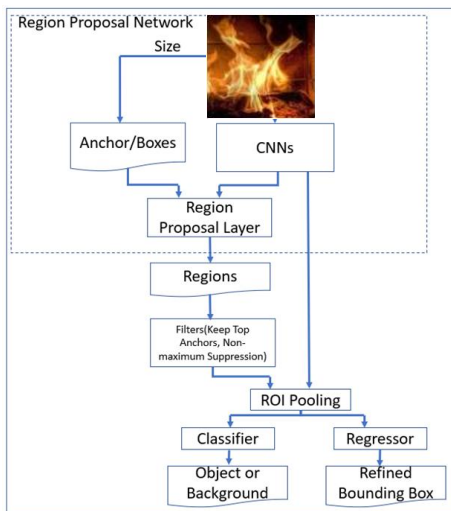


Fig. 2. The Architecture of Faster R-CNN.
그림 2. Faster R-CNN 구조도

RPN은 3x3 윈도우를 이용한 컨볼루션 계산에 의해 입력 영상을 256 또는 512 크기의 특징을 추출한다. 여기서 얻어진 결과는 box classifier 계층과 box regress 계층으로 사용된다. box regression은 미리 정의된 참조 box 이름들을 사용하며, 이 box는 슬라이딩 창의 각 위치에서 바운딩 박스 후보로 사용된다. 이것은 같은 크기의 슬라이딩 윈도우를 움직이는 중심 위치를 이용하여 미리 정의된 다양한 비율 / 크기의 앵커 박스를 적용하여 특징들을 추출한다. 9 개의 앵커 박스(3 가지 크기와 3 가지 비율)가 사용된다. 앵커 박스는 슬라이딩 윈도우의 각 위치에서 바운딩 박스의 후보로 사용됩니다.

라벨링된 영상 데이터는 앞에서 기술한 Faster R-CNN 모델을 사용하여 처리되고, .meta 파일은 학습 결과로서 생성된다. 최종적으로 얻어진 그래프 파일(“.pb”파일)은 이전 단계에서 얻은 “.meta” 파일을 사용하여 생성된다. 마지막으로 “.pb”파일을 사용하는 개체 감지 결과(바운딩 박스 및 개체 점수)가 입력 이미지의 모니터에 표시됩니다.

2. 유사도 및 움직임 계산

SSIM(Structural Similarity)[15]은 압축과 변환으로 인한 왜곡과 원본 이미지의 유사도를 측정하는 방법이며, 이것은 MSE(Mean Square Error)와 Peak Signal 노이즈 비율(PSNR) 방법 보다 높은 정밀도를 나타낸다. 이것은 시각적 유사성을 정량화하기 위한 원본 영상(Y)에 대한 테스트 영상(X)의 평가이다. 1.0에 가까운 값은 테스트 영상이 원본 영상과 비슷함을 의미하며, 0.0에 가까운 값은 테스트 영상이 원본 영상과 매우 다르다는 것을 의미한다. SSIM 수식은 아래와 같이 정의된다.

$$l(x,y) = (2\mu_x\mu_y + C1) / (\mu_x^2 + \mu_y^2 + C1) \quad (1)$$

$$c(x,y) = (2\sigma_x\sigma_y + C2) / (\sigma_x^2 + \sigma_y^2 + C2) \quad (2)$$

$$r(x,y) = (\sigma_{xy} + C3) / (\sigma_x\sigma_y + C3) \quad (3)$$

여기서 μ_x 와 μ_y 는 평균 픽셀 값, σ_x 와 σ_y 는 표준편차 그리고 σ_{xy} 는 x와 y의 공분산 값이다. C1, C2 그리고 C3는 분모와 분자가 0이 되는 것을 막기 위한 상수이다. $l(x,y)$ 는 밝기 차이의 관계, $c(x,y)$ 는 대비 차이의 관계 그리고 $r(x,y)$ 는 x와 y 사이의 구조적 변화에 대한 유사도이다. 이들을 이용한 구조적 유사도는 식 (4)와 같다.

$$SSIM = [l(X,Y)]^\alpha [c(X,Y)]^\beta [r(X,Y)]^\gamma \quad (4)$$

여기서 α, β, γ 는 각 항의 중요도를 나타내는 것으로써 본 논문에서는 1.0을 사용하였다.

III. 실험 및 결과

본 논문은 딥러닝 기반의 Onvif 카메라 이용 화재 감지 시스템에서 발생하는 오탐율을 최소화하기 위한 프레임내의 관심영역 유사성을 이용한 새로운 알고리즘을 제안하였다. 연구에서 사용된 시스템은 Intel Core i7-7700 (3.5 GHz), 메모리 16G,

Geforce TITAN-X 이며, 중요 프로그램은 OpenCV 및 Python 3.5 이다. 본 연구에 사용된 불꽃 및 연기 데이터베이스는 인터넷과 일반 노지 및 공장에서 직접 녹화된 비디오를 사용하였다(100 미터가 넘는 차량도 포함). 비디오 녹화 장치는 휴대폰 카메라와 Cannon G5카메라를 이용하였다.

제안 된 알고리즘을 구현하기 위해 다음과 같은 과정을 수행하였다. 첫 번째, 라벨링 데이터 작업이다. 비디오로부터 프레임 단위로 영상을 추출 한 후, 각각의 영상으로부터 불꽃과 연기에 해당 부분들을 LabelImg 프로그램을 사용하여 라벨링을 한다. 라벨링 영역의 예를 그림 3에 나타내었다. 본 논문에서 사용된 라벨링 후보는 불꽃, 연그라인더 및 용접 불꽃 그리고 사람을 대상으로 하였다. 라벨링 결과는 이미지 이름과 함께 각 사각형의 네 점 좌표와 객체의 이름이 포함되어 .xml 파일에 저장된다.

둘째, 라벨링된 영상들에 대한 훈련과정이다. 레이블링 결과의 .xml 파일은 텐서플로의 학습 데이터 형식으로 변환되어야합니다. 훈련 과정에서 입력 영상은 JPEG 또는 PNG 파일입니다. 이러한 영상의 메타 데이터 및 레이블은 별도의 파일에 저장되므로 훈련 데이터를 읽을 때 메타 데이터 또는 레이블 파일과 별도로 이미지 파일을 읽어야하기 때문에 코드가 복잡해집니다. 또한 영상을 JPG 또는 PNG 형식으로 읽고 매번 디코딩하면 성능이 저하될 수가 있다. TFRecord 파일 형식을 사용하면 위의 성능 저하를 방지하고 보다 쉽게 개발할 수 있다. TFRecord 파일 형식은 이미지의 높이와 너비, 파일 이름, 인코딩 형식, 이미지 바이너리, 영상에서 대상 물체의 위치를 나타내는 레이블 사각형 위치 값을 저장한다. 이 과정을 통해 전체 학습 데이터는 70% 훈련 데이터와 30% 유효성 검증데이터로 분류되어 저장된다. 훈련에 사용된 모델은 FAST R-CNN ResNet(Deep Residual Network) 선택하였으며, 적은 종류의 물체와 검출율이 가장 높은 것이 특징이다. 화재와 관련하여 5 종류의 물체를 검출해야하므로 카테고리는 5로, 학습 단계는 20,000 으로 설정하였다. 훈련에 사용된 화재 영상은 12,160 장을 사용하였다.

세 번째, 훈련 된 모델의 추출하는 것이다. 학습 프로세스는 소정의 포인트마다 학습 결과를 나타내는 체크 포인트를 저장한다. 각각 체크 포인트는

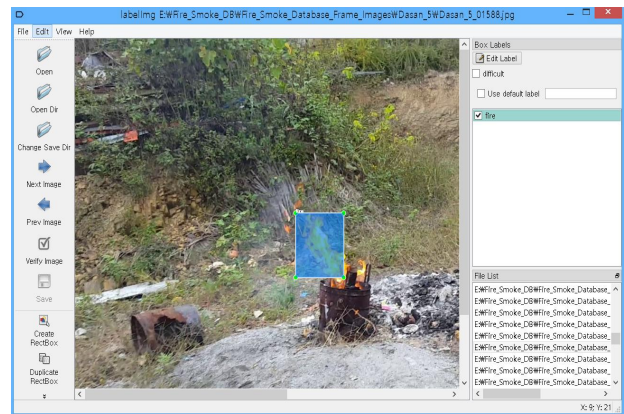


Fig. 3. Example of Labeling area for dataset image.
그림 3. 영상 데이터의 라벨링 영역 예

Tensorflow 모델 파일 형식의 모델에 대한 meta 정보를 가지고 있으며, 다시 학습 할 수 있다. 그러나 메타 파일에는 많은 불필요한 정보가 있기 때문에 실제 모델을 사용하기 위해서는 메타 파일을 개선해야할 점이 많다. 최종적으로 메타 데이터를 제외한 모델과 가중치를 결합하는 “.pb”파일이 생성된다.

이 논문에서, 실험 데이터는 실제로 공장에서 촬영된 비디오 영상을 사용하였다. 그림 3은 실험에 사용된 비디오의 프레임 예를 나타내었다.



Fig. 4. Example of the frame sequences of test videos.
그림 4. 테스트 비디오의 연속 프레임 예

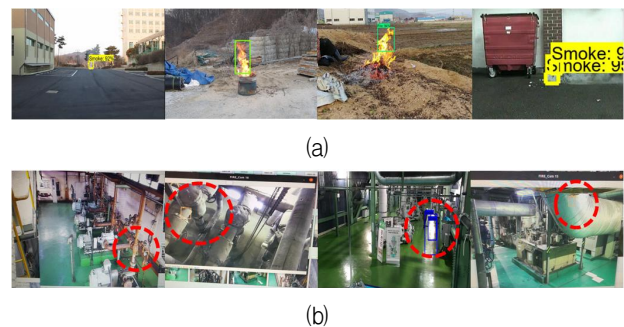


Fig. 5. The frame sequence result of the Faster R-CNN, (a) the result of true positive, (b) the result of false positive.

그림 5. Faster R-CNN의 연속 프레임 결과 (a) true positive 결과, (b) false positive 결과

Fater R-CNN 모델을 기반으로 완성된 “.pb”파일을 사용하여 화재 검출 실험을 수행하였다. 그림 5에 딥러닝을 적용한 화재 검출 결과들을 나타내었다.

Table 1. Faster R-CNN results for video images (Frames).

표 1. Faster R-CNN video 테스트 결과

Input Video	Ground Truth	True Positive	True Negative	False Positive
Video 1	1183	1158	0	25
Video 2	14997	0	14891	106
Video 3	14658	0	14492	166

그림 5 (a)는 촬영된 여러 가지 비디오를 이용하여 화재 (불꽃과 연기)검출 여부를 실험 한 결과이다. 화재 검출 범위는 30미터 이내로 하였다. Faster R-CNN의 검출 임계값은 40%이상으로 하였다. 그림 5 (b)는 딥러닝 훈련결과를 적용하여 오탐이 검출된 결과이다. 오탐은 여러 곳에서 나타났지만 그 유형은 2가지로 나누어진다. 첫 번째, 햇빛의 반사에 의해 연기 혹은 불꽃으로 검출되는 경우이다. 두 번째, 시설물 가운데서 연기와 불꽃처럼 형상과 색상이 비슷한 경우이다. 그림 5 (b)는 모두 상기 사항에 해당한다. 그림 5 (a)와 (b)에서 사용된 1개의 비디오 영상 결과를 대한 화재 검출 결과를 표 1에 나타내었다. Video 1의 경우 정밀도는 98%과 같으며, 계산 식은 식 (5)와 같다. 그리고 화재이지만 화재로 인식 못하는 경우가 약 2%정도 나타났다. 그리고 오탐의 경우는 나타나지 않았다. 이 경우는 반사 영역, 연기 그리고 불꽃과 같은 유사형태가 없기 때문이다. Video 2 경우 False Positive는 106개 그리고 Video 3 경우 False Positive는 166개가 각각 나타났다. Video 2와 Video 3은 공장 내부에서 촬영된 동영상을 대상을 하였다. 표 1에 서 Ground Truth는 전체 이미지의 프레임 수를 나타낸다. True Positive(TP)는 화재가 검출되는 경우를 나타내고, True Negative(TP)는 화재로 인식 못하는 경우, False Positive(FP)는 화재가 아닌 것을 화재로 검출한 경우이다.

$$Precision = \frac{TP}{TP+FP} \quad (5)$$

Video 1인 경우는 연속 프레임에서 발생된 것이

아니며, 비디오가 30fps이므로 충분히 보상이 가능하다. 그러나 Video 2인 경우는 계속 알람이 울리므로 사용자 스트레스가 높아지게된다. Video 2에서 발생하는 오탐율을 저하시키기위하여 본 논문에서는 다음과 같은 특성을 이용한다.

첫째, 움직임 특성을 확인한다. 화재가 발생할 경우 움직임이 존재하므로 30픽셀 이상의 블록이 생성되면 이를 화재 후보 상태로 등록한다. 그림 6에 화재 발생 경우와 화재 발생이 없는 경우에 대해 프레임 차를 각각 나타내었다. 여기서 사용한 프레임 차의 크기는 10이다.

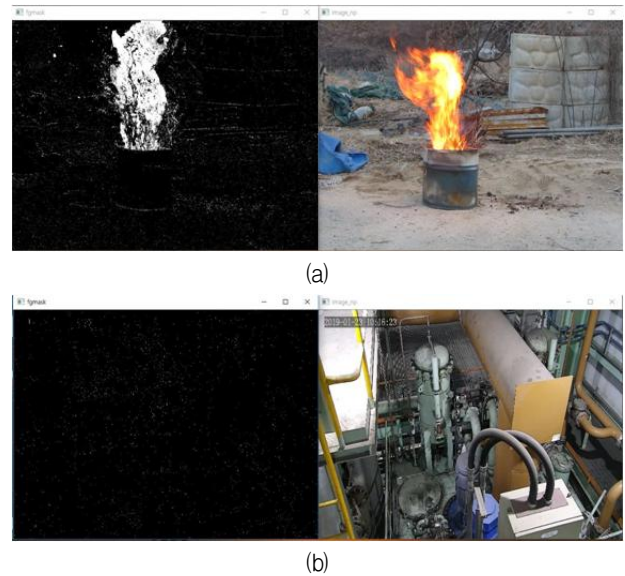


Fig. 6. The result of frame compare,

(a) In case of fire, (b) In case of non-fire.

그림 6. 프레임 비교 결과, (a) 화재가 발생한 경우, (b) 화재가 발생하지 않은 경우.

둘째, 화재 후보 상태가 1이면 딥러닝 과정을 수행하여 화재 영역의 바운딩 박스가 관심 영역으로 저장되며, 다른 프레임과의 비교는 SSIM 지수와 평균 제곱(MSE) 오차를 이용하여오탐 여부를 판별하게 된다. 실험에서 하나의 배열에 10장의 프레임 영상이 저장되며, 첫 번째, 중간, 그리고 마지막 프레임을 이용하여 SSIM 지수와 평균 자승 오차(MSE)를 사용한 유사성 계산은 아래 식과 같다.

$$S_k = ssim(f_i, f_j) \quad (6)$$

$$M_k = MSE(f_i, f_j) \quad (7)$$

$$V_s = \begin{cases} 1, & \text{if } 0.5 \leq S < 0.91 \\ 0, & \text{else} \end{cases} \quad (8)$$

$$V_M = \begin{cases} 1, & \text{if } M_k < Th \\ 0, & \text{else} \end{cases} \quad (9)$$

$$V^p = \begin{cases} 1, & \text{if } V_S = 1 \text{ and } V_M = 1 \\ 0, & \text{else} \end{cases} \quad (10)$$

여기서 k 는 1, 2, 3이며, f_i 와 f_j 는 프레임 영상이다. 화재가 검출되면 앞과 뒤 프레임 유사성은 떨어지게되고 평균 제곱 에러는 높게 나타난다. 논문에서 사용된 Th 값은 8000이며, 실험 값이다.

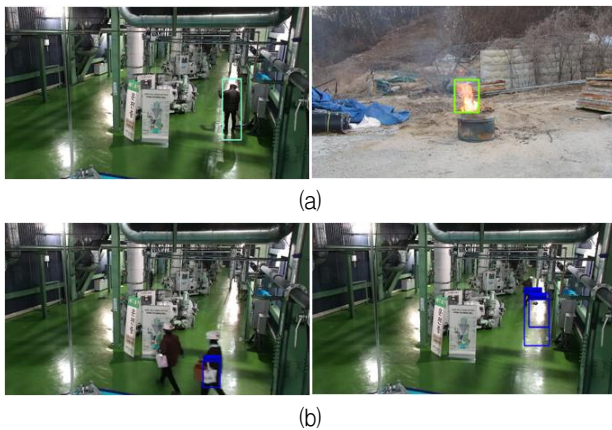


Fig. 7. The result of adapted proposal algorithm using video files, (a) good result, (b) bad result.
 그림 7. 제안된 알고리즘이 적용된 비디오 파일 실험 결과, (a) 좋은 검출 결과, (b) 나쁜 검출 결과

Table 2. Proposed algorithm results for video images (Frames).
 표 2. 제안된 알고리즘을 적용한 video 테스트 결과

Input Video	Ground Truth	True Positive	True Negative	False Positive
Video 1	1183	1158	25	0
Video 2	14997	0	14997	0
Video 3	14658	0	14539	119

제안된 알고리즘을 이용한 실험 결과를 표 2에 나타내었다. 제안된 알고리즘을 사용한 결과 Video 2에서 검출된 오탐율이 0%로 떨어졌으며, Video 1에서 화재 검출은 그대로 지속되었다. 그러나 Video 3에서는 오탐율이 0.8%로 나타났다. 추가적인 8가지의 비디오에 대한 오탐율 실험 결과는 0%를 기록했으며, 비디오 실험에서 사용한 영상 예를 그림 8에 나타내었다.

Video 3 실험에서 오탐이 없어지지 않은 이유는 역광부분에 사람의 움직임이 있으며, 이로 인하여 형상이 변형되어 연기로 검출되는 경우이다. 그리고 장바구니 그리고 헬멧등이 경우에 따라 오탐



Fig. 8. The example of videos to test for proposed algorithm.
 그림 8. 제안된 알고리즘을 적용한 다른 비디오 영상 예

로 감지되었다. 여러 사항을 고려하여 최종 임계값을 조정하거나, 추가적인 색상 및 특징 분석을 이용한 알고리즘이 요구된다.

IV. 결론

화재는 작은 불씨가 많은 재산과 생명을 앗아가는 무서운 자연환경이다. 본 논문은 딥러닝 기반으로 화재를 검출할 경우 오탐이 발생하는 부분을 줄이고자 새로운 알고리즘을 제안하였다. 일반적인 딥러닝 방법을 사용 할 경우, 영상 내에 연기혹은 불꽃과 비슷한 형상이 존재 할 경우 화재로 오탐이 발생한다. 이 문제를 해결하기 위해 논문에서 첫째, 움직임을 이용하였다. 정지된 영상에서 화재가 발생하면 움직임이 발생한다. 둘째, 화재로 검출된 영역(바운딩 박스)에 대한 프레임 유사도 및 평균 제곱 에러를 이용한 새로운 알고리즘을 적용하였다. 기존의 일반적인 딥러닝만을 적용했을 때 보다 제안된 알고리즘의 오탐 확률이 현저히 줄어든다는 것을 실험을 통해 알 수 있었다. 추후 과제으로써는 오탐을 좀 더 줄이기 위해 화재의 색상 분석과 프레임의 상관 관계를 이용한 실험 및 딥러닝 모델 연구가 필요하다.

References

[1] B. C. Ko, S. J. Han, and J. Y. Nam, "Modeling and Formalization of Fuzzy Finite Automata for Detection of Irregular fire Flames," *IEEE Transaction on Circuits and System Video Technology*, vol. 21, no. 12, pp. 1903–1912, 2011. DOI: 10.1109/TCSVT.2011.2157190

[2] H. Yamagishi and J. Yamaguchi, "A Contour Fluctuation data Processing Method for Fire Flame Detection using a Color Camera," *IEEE*

26th Annual Conference on IECON of the Industrial Electronics Society, p. 824, 2000.

DOI: 10.1109/IECON.2000.972229

[3] T. Celik and H. Ozkaramanli, "Automatic Fire Detection in Video Sequences," *Proceeding of European Signal Processing Conference*, Italy, 2016.

[4] T. Chen, P. Wu, and Y. Chiou, "An Early Fire-Detection Method Based on Image processing," *2004 International Conference on Image Processing*, pp. 1707-1707, 2004. DOI: 10.1109/ICIP.2004.1421401

[5] C. B. Liu and N. Ahuja, "Vision Based Fire Detection," *2004 International Conference on Pattern Recognition*, p. 34, 2004.

DOI: 10.1109/ICPR.2004.1333722

[6] F. N. Yuan, G. X. Liao, Y. M. Zhang, "Feature Extraction for Computer Vision Based Fire Detection," *Journal of University Science and Technology of China*, pp. 39-43, 2006.

[7] T. B. Ugur and Y. Dedeoglu, "Computer Vision based Method for Real-time Fire Flame Detection," *Pattern Recognition Letters*, Vol. 27, no. 1, pp. 49-58, 2006.

DOI: 10.1016/j.patrec.2005.06.015

[8] N. Dedeoglu and B. U. Toreyin, "Real-time Fire and Flame Detection in Video," *ICASSP'05*, pp. 669-672, 2005.

[9] T. B. CelikT and Y. Dedeoglu, and A. E. Cetin, "Wavelet Based Real-time Smoke Detection in Video," *EUSIPO 2005*, 2005.

DOI: 10.1109/ICASSP.2005.1415493

[10] C. Yu, Z. Mei, and X. Zhang, "A Real-time Video Fire Flame and Smoke Detection Algorithm," *Procedia Engineering*, Vol. 62, pp. 891-898, 2013.

DOI: 10.1016/j.proeng.2013.08.140

[11] S. Frizzi, R. Kaabi, M. Bouchouicha, J. Ginoux, E. Moreau, and F. Fnaiech, "Convolutional Neural Network for Video Fire and Smoke Detection," *IECON 2016*, 2016.

DOI: 10.1109/IECON.2016.7793196

[12] Y. LeCun, L. Bottou, Y. Bengio, and P. Hsfner, "Gradient-based Learning Applied to Document Recognition," *Proceeding of IEEE*, vol. 86, no. 11,

pp. 2278-2324, 1998. DOI: 10.1109/5.726791

[13] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet Classification with Deep convolutional neural Network: Advanced in Neural Information Processing System 25," *26th Annual Conference on Neural Information Processing Systems*, pp. 1106-1114, 2012.

[14] G. E. Hinton, N. Srivastava, A. Krizhevsky, S. Ilya, and R. Salakhutdinov, "Improving neural networks by preventing co-adaptation of feature detectors," *Clinical Orthopedics and Related Research 2012*, 2012.

[15] H. S. Kim and J. S. Park, "Intensity-based efficient Video Quality Assessment for Variable bitrate Streaming," *Korean Institute of Next Generation Computing*, vol. 11, no. 5, pp. 63-71, 2015.

BIOGRAPHY

Yeung-Hak Lee (Member)



1988 : BS degree in Electronic Engineering, Yeungnam University.
1991 : MS degree in Electronic Engineering, Yeungnam University.
2003 : PhD degree in Electronic Engineering, Yeungnam University.

1991~1995 : Research Engineer, LG Precision.

2005~2006 : Postdoc, Cardiff University.

2010~2016 : Professor, Kyungwoon University.

2017~ : Research Engineer, Andong National University.

Jae-Chang Shim (Member)



1987 : BS degree in Electrical Engineering, Kyungpook National University.

1990 : MS degree in Electrical Engineering, Kyungpook National University.

1993 : PhD degree in Electrical Engineering, Kyungpook National University.

1994~ : Professor, Andong National University.