

다양한 재료에서 발생하는 연기 및 불꽃에 대한 YOLO 기반 객체 탐지 모델 성능 개선에 관한 연구

권희준¹, 이보희² , 정해영¹ 

¹ 세명대학교 소방방재학과

² 세명대학교 전기공학과

Research on Improving the Performance of YOLO-Based Object Detection Models for Smoke and Flames from Different Materials

Heejun Kwon¹, Bohee Lee², and Haiyoung Jung¹

¹ Department of Fire and Disaster Prevention, Semyung University, Jecheon 27136, Korea

² Department of Electrical Engineering, Semyung University, Jecheon 27136, Korea

(Received December 5, 2023; Revised December 26, 2023; Accepted January 6, 2024)

Abstract: This paper is an experimental study on the improvement of smoke and flame detection from different materials with YOLO. For the study, images of fires occurring in various materials were collected through an open dataset, and experiments were conducted by changing the main factors affecting the performance of the fire object detection model, such as the bounding box, polygon, and data augmentation of the collected image open dataset during data preprocessing. To evaluate the model performance, we calculated the values of precision, recall, F1Score, mAP, and FPS for each condition, and compared the performance of each model based on these values. We also analyzed the changes in model performance due to the data preprocessing method to derive the conditions that have the greatest impact on improving the performance of the fire object detection model. The experimental results showed that for the fire object detection model using the YOLOv5s6.0 model, data augmentation that can change the color of the flame, such as saturation, brightness, and exposure, is most effective in improving the performance of the fire object detection model. The real-time fire object detection model developed in this study can be applied to equipment such as existing CCTV, and it is believed that it can contribute to minimizing fire damage by enabling early detection of fires occurring in various materials.

Keywords: Materials, Object detection, Fire detection, Real_time, YOLOv5

1. 서론

최근 고도의 산업화와 기술 발전으로 건축물과 구조물에서는 가벼우면서 내구성이 뛰어난 다양한 재료가 활용되고 있으며, 특히 PMMA (폴리메틸메타크릴레이트), PC (폴리카보네이트) 및 HDPE (고밀도 폴리에틸렌) 등 다양한 재료가 사용된다. 재료는 특성에 따라 비중, 내열점, 난연 특성 등이 다르게 나타나며, 이는 화재가 발생할 경우

✉ Bohee Lee; bhlee@semyung.ac.kr
Haiyoung Jung; hyjung@semyung.ac.kr

Copyright ©2024 KIEEME. All rights reserved.
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.

다양한 색상 및 형태의 연기와 불꽃이 발생하는 특징이 있다 [1-4]. 그림 1은 일상생활에서 다양하게 사용되는 재료를 보여주며, 그림 2는 재료에 따라 나타나는 화재 양상을 보여준다. 표 1은 재료에 따른 비중, 내열점, 난연 특성을 보여주고 있다.

건축물, 전기기기 및 재료 등은 노후화되면서 재료의 열화 현상을 유발하고 이는 화재 발생 가능성을 높이고 화재 확대의 원인이 된다 [5,6]. 2010~2019년간 발생된 전체 화재 426,521건 중 전기 화재는 98,662건으로 전체 화재의 23%를 차지하였으며, 주요 원인으로는 단락, 과부하, 아크 등이 있다 [7-10]. 이처럼 재료와 화재의 유형에 따라 다양한 열 패턴의 경향, 탄화조직 생성 등의 현상이 나타나며, 이에 따라 색상과 형태가 다양한 연기와 불꽃이 발생한다 [11,12]. 이러한 문제를 해결하고 화재를 신속 정확하게 탐지하기 위해 최근 다양한 인공지능을 활용한 화재객체

탐지에 관한 연구가 수행되고 있다 [13,14]. 인공지능 모델에서 화재로 판단할 수 있는 객체는 연기와 불꽃이 있다. 하지만 연기와 불꽃은 무정형으로 구체적인 형태가 있지 않고 매 순간 형태가 변한다. 또한, 재료의 종류, 화재 장소 및 화재의 크기 등 주변 환경에 따라 색상과 형태가 다양하므로, 이로 인해 초기 화재 상황을 정확하게 판별하는 것은 매우 어려운 문제이다 [15,16].

최근 Zhang *et al.*은 화재를 감지하기 위해 CNN을 활용하는 방식을 제안하였다. 이 연구에서는 전체 이미지를 합성곱 신경망(CNN), 이미지 내 작은 영역을 신경망(NN) 분류기를 사용하여 화재 인식 모델을 개발하였으며, 훈련 및 테스트 데이터 세트에서 각각 97%와 90%의 정확도 성능을 확인하였다. 하지만 CNN 모델은 이미지를 처리할 때 많은 저장용량을 필요로 하며, 복잡한 연산과 계산으로 인해 이미지 처리 속도가 매우 느리다는 문제가 있다. 이러한

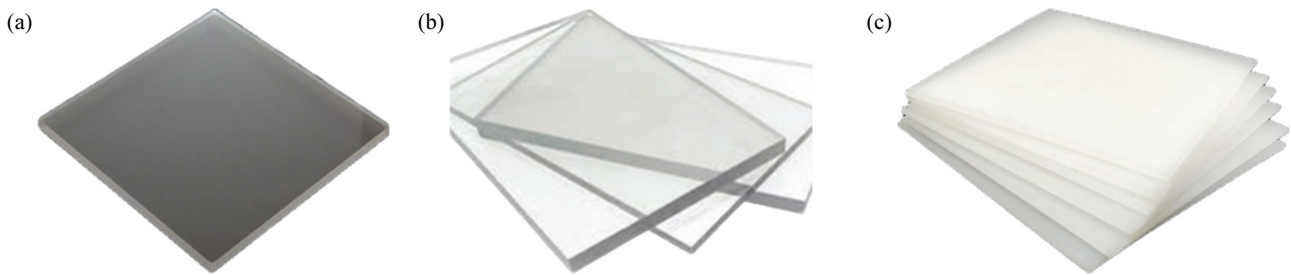


Fig. 1. Different materials: (a) PMMA, (b) PC, and (c) HDPE.

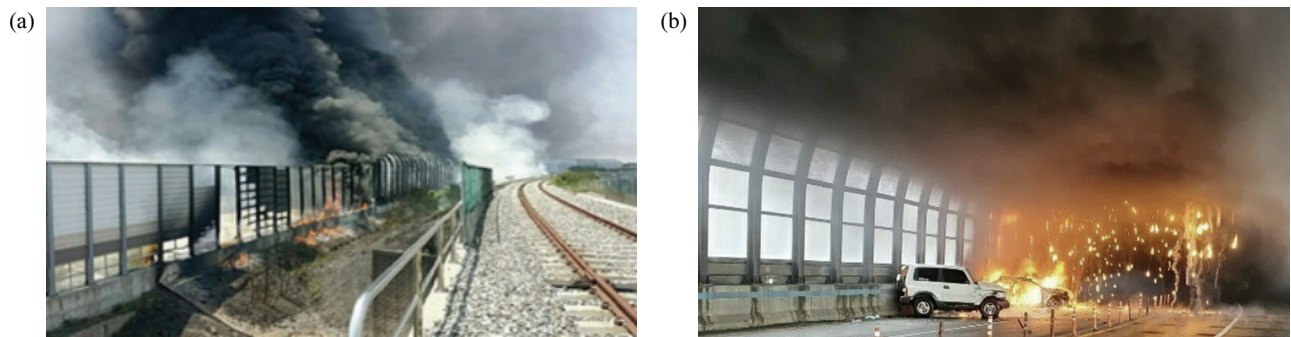


Fig. 2. Fire shape of materials: (a) Gyeongbu expressway fire (PMMA) and (b) Cheongju soundproof tunnel fire (PC).

Table 1. Comparison of physical properties.

Division	Specific gravity	Heat resistant point	Melting point	Fire retardant
PMMA	1.17~1.21	95°C	224°C	Combustibility
PC	1.19~1.23	120°C	230°C	Self-extinguishing
HDPE	0.93~0.97	120°C	130°C	Self-extinguishing

문제를 해결하고자 최근에는 다양한 딥러닝 모델을 활용하여 실시간으로 화재를 탐지하는 다양한 연구가 수행되고 있다 [17]. Li, et al은 실시간 화재 감지 기술의 성능을 향상을 목적으로 딥러닝 기반의 객체 탐지 모델인 Faster R-CNN, R-FCN, SSD 및 YOLOv3를 활용하여 화재 탐지 모델 비교 연구를 수행하였다. 연구 결과 YOLOv3 모델의 성능이 mAP 83.7% FPS 28로 가장 우수한 성능을 보였다. 하지만 실시간 화재를 탐지하기에는 한계가 있다 [18]. 따라서 본 연구에서는 재료에 따라 다양한 색상과 형태로 변화하는 연기와 불꽃에 대해 정확하게 탐지하지 못하는 문제와 화재 탐지 모델의 속도 문제를 개선하고자 실시간 화재객체 탐지 모델에 적합한 object detection 모델과 optimizer를 선정하고, 이후 입력 이미지에 대해 다양한 전처리 방식을 적용하여 화재객체 탐지 모델 성능을 비교 분석하고, 우수한 성능의 실시간 화재객체 탐지 딥러닝 모델을 개발하고자 하였다.

2. 객체 탐지 모델

2.1 R-CNN 모델

R-CNN 모델의 특징은 선택적 탐색을 통해 약 2,000여 개의 후보 영역을 추출하며, 추출된 후보 영역의 입력 크기를 고정한다. 이후 AlexNet를 통하여 각 입력에 대한 후보 영역별 특징을 추출한 후 SVM과 경계 상자 예측을 수행하여 객체를 탐지한다. 이러한 R-CNN 입력 크기를 고정하는 과정에서 정보 손실이 발생하며, 2,000여 개의 후보 영역에 대한 특징 추출로 많은 시간이 소요된다. 또한 여러 단계의 학습으로 인해 대용량 저장 공간 요구 및 많은 시간을 필요로 하며, 이러한 문제를 개선하기 위해 fast R-CNN이 개발되었다. Fast R-CNN은 1장의 이미지를 입력으로 받으며, RoI (region of interest)을 통하여 고정된 크기의 특징을 FC layer로 전달한다. 이후 Softmax와 경계 상자 예측을 수행하여 객체를 탐지한다. 이를 통해 향상된 성능, 적은 저장 공간, 학습시간과 추론시간을 개선하였다. 하지만 후보 객체 영역을 생성하는 과정에서 많은 시간을 소요하며, 실시간으로 객체를 탐지하기에는 너무 느린 속도를 보인다. 이런 문제를 개선하기 위해 경계 상자를 예측하는 동시에 객체를 평가할 수 있는 RPN (region proposal network)을 도입한 faster R-CNN이 개발되었다. Faster R-CNN은 입력 이미지를 CNN 모델을 사용하여 이미지의 특성을 추출하며, 특성 맵(feature map)을 생성한다. 이후 특성 맵에 RPN을 적용하여 객체 후보 영역

을 생성하고 각 객체 후보 영역에 대해서 객체가 존재할 확률을 계산한다. 이후 RoI를 통해 고정된 크기의 특성 맵과 각 후보 영역의 특징을 추출하며, 전결합 계층(fully connected layer)을 통해 객체 분류 및 bounding box 회귀를 위한 정보를 추출한다. 최종적으로 softmax와 bounding box regression를 통해 객체에 대한 클래스 예측과 객체 위치를 기반으로 객체를 탐지한다. 이러한 모델 기법을 2_stage 방식이라 하며, 선택적 탐색(selective search)을 통해 유사한 영역을 그룹화하여 후보 영역을 제안하는 첫 번째 단계와 각 후보 영역별 특징을 추출 후 객체 분류(classification)와 경계 상자 예측(bounding box regression)을 수행하는 두 번째 단계를 거쳐 객체를 탐지하게 된다 [16,19,20].

2.2 YOLO 모델

그림 3은 YOLO의 수행 과정을 보여주고 있다. 먼저 입력 이미지가 들어오면 가로, 세로가 동일한 S×S의 grid로 이미지를 나눈 후 각각의 grid cell에 대해 bounding box 작업과 classification 작업을 동시에 수행한다. 이때, bounding box는 (x,y), (w,h)와 confidence score를 가진다. (x,y)는 bounding box의 중심점이며, (w,h)는 bounding box의 width, height에 대한 상대값에 해당한다. 또한, confidence score는 bounding box가 객체를 포함하는지에 대한 확률값과 bounding box의 예측이 얼마나 정확한가에 대한 정도를 반영한다. 한편, bounding box 작업과 동시에 수행되는 classification은 각 bounding box가 아닌 각 grid cell에 대해 수행된다. 마지막으로 NMS (non-maximum suppression) 알고리즘을 통해 한 객체에 대해 confidence score 가장 높은 하나의 bounding box만을 남기고 그 객체와 관련된 나머지 bounding box를 제거해 최종적인 객체 탐지 이미지를 도출한다. 따라서 YOLO 모델은 입력 이미지에 대해 객체의 특징을 의미하는 객체 분류(classification), 위치를 의미하는 지역화(localization)가 동시에 진행하므로 1_stage 방식이라 하며, 반면 R-CNN 모델과 같이 객체의 특징과 위치에 대한 추출이 각각 진행되는 기법을 2_stage 방식이라 한다. 결과적으로 YOLO 모델과 같은 1_stage 방식은 객체 탐지 속도가 빨라 실시간 객체 탐지에 대해 2_stage 방식에 비해 유리한 방식이다 [21].

2.3 객체 탐지 모델의 성능 비교

화재를 실시간으로 탐지하기 위해서는 높은 정확도와 빠

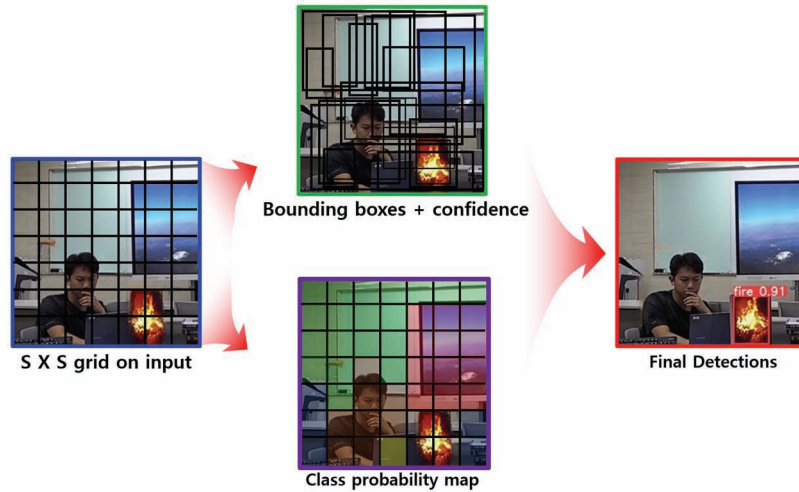


Fig. 3. YOLO algorithm operation principle.

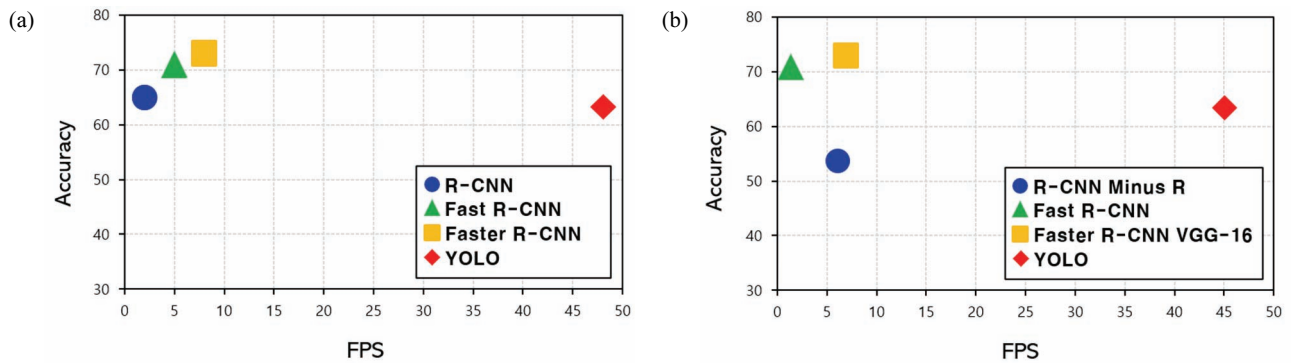


Fig. 4. Compare the accuracy and FPS performance of R-CNN, Fast R-CNN, faster R-CNN, and YOLO: (a) Kumar's the results of each model test and (b) Redmon's the results of each model test.

른 탐지 속도를 필요로 한다. 기존의 연구에서 Kumar, et al은 속도가 가장 빠른 Faster R-CNN보다 YOLO가 속도 측면에서 40 FPS 이상의 성능을 보이는 것을 확인하였고, Redmon et al.은 YOLO가 Faster R-CNN VGG-16보다 속도 측면에서 38 FPS 이상의 성능을 보이는 것을 확인하였다. 따라서 YOLO는 R-CNN 모델과 비슷한 수준의 정확도를 가지며, 또한 높은 속도 성능을 보인다는 것을 확인할 수 있었다. 그림 4는 YOLO 및 R-CNN, Fast R-CNN, Faster R-CNN 모델에 대한 속도와 정확도의 성능 비교 결과를 보여주고 있다 [21,22].

2.4 YOLOv5s6.0

그림 5는 실험에서 사용한 YOLOv5s6.0 모델의 아키텍처이다. YOLOv5s6.0 모델의 backbone은 입력 이미지를

특성 맵으로 변환하는 역할을 수행한다. 입력 이미지 데이터는 backbone의 첫 번째 Conv 모듈에서 연산을 통해 특성 맵이 생성되고, 다음 단계의 Conv 모듈들과 C3 모듈들을 통해 입력 이미지 데이터에 대한 고유의 특징들이 추출된다. 이후 head에서는 backbone에서 생성된 특성 맵에 대하여 탐지하고자 하는 객체에 대해서 객체 분류(classification)와 지역화(localization)에 대한 예측을 수행한다. 각 단계의 수행 과정은 다음과 같다. Backbone에서 생성된 특성 맵은 head의 첫 번째 계층인 Conv 모듈로 전달되고, 다음 단계인 up-sample 모듈에서 특성 맵의 해상도와 객체의 크기가 조절된다. Concat 모듈 단계에서는 여러 계층의 특성 맵들이 결합한 후 객체의 클래스 및 위치에 대한 예측이 동시에 수행된다. 이후 입력 이미지 데이터 내의 객체 크기에 따라 객체 클래스에 대한 예측과 위치에 대한 bounding box의 좌표 및 크기가 출력되어 객체가 탐지된다. 이러한 과정을 통하여 입력되는 이미지는 크기와

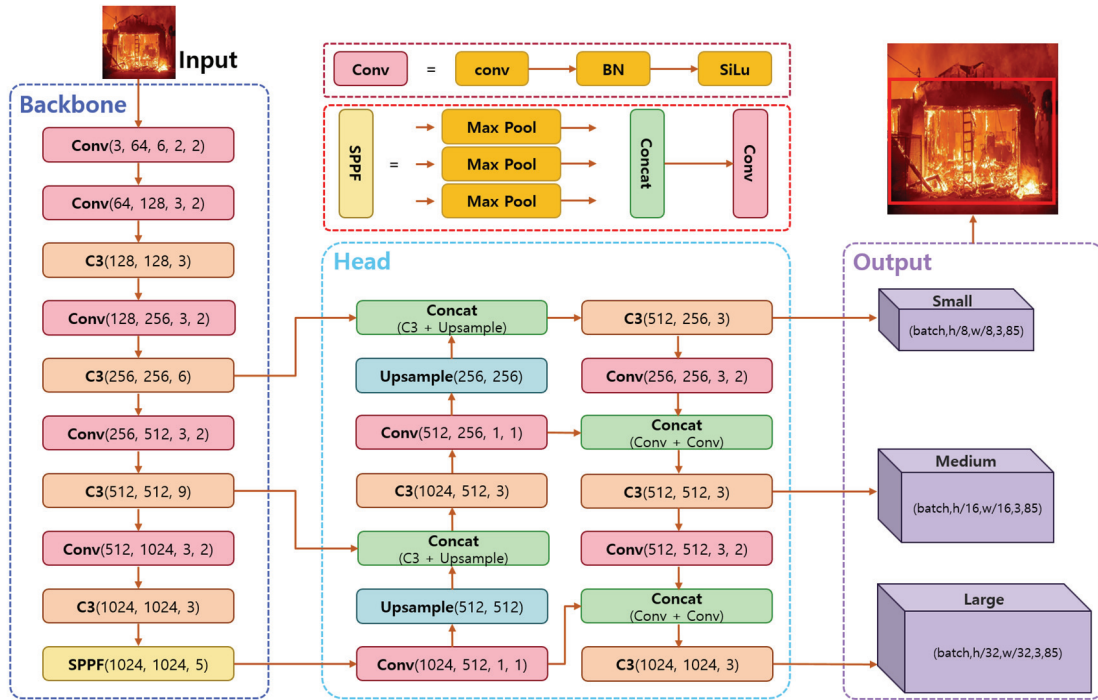


Fig. 5. YOLOv5s6.0 object detection.

상관없이 정확하고 신속한 객체 탐지가 가능하다 [23].

따라서 본 연구에서는 화재객체에 대한 신속한 감지가 가능한 높은 FPS 성능의 YOLOv5s6.0 모델을 실시간 화재객체 탐지 모델로 선정하였다.

3. 실험 내용

3.1 모델 개발 수행 절차

본 논문의 모델 개발 수행 절차는 다음과 같다.

우선 기존 선행연구 분석을 통해 실시간 객체 추적이 가능하고 정확도 성능이 우수한 YOLOv5을 연기 및 불꽃 객체 탐지 모델로 선정하였다. 선정된 연기 및 불꽃 객체 탐지 모델의 성능 개선을 위해 다양한 연기 및 불꽃 이미지 데이터를 수집하였다. 수집된 이미지 데이터에 대해서는 크기 조정, 데이터 라벨링, 이미지 데이터 증강 등의 데이터 전처리 작업을 수행하였다. 또한, 연기 및 불꽃 객체 탐지 모델의 optimizer 선정을 위해 전처리된 이미지 데이터를 활용하여 optimizer별 성능 비교 실험 수행을 진행하였다. 최종적으로 연기 및 불꽃 이미지 데이터에 대해 bounding box 기법, polygon 기법, augmentation 기법을 적용하여 각각에 대해 평가 및 예측 실험을 진행하고 다

양한 성능 평가 지표를 도출하였다.

3.2 데이터 수집

그림 6은 화재 상황에서 발생하는 연기와 불꽃이 포함된 훈련, 검증, 평가 이미지 데이터의 구성을 보여주고 있다. 훈련 데이터셋(training dataset)은 모델이 학습할 데이터셋이며, 검증 데이터셋(validation dataset)은 모델이 학습이 이미 완료된 모델을 검증하기 위한 데이터셋에 해당한다. 마지막으로 평가 데이터셋(test dataset)은 모델의 성능을 평가하기 위해 사용할 데이터셋을 의미한다.

본 연구에 사용한 연기 및 불꽃 이미지들은 Kaggle과 AI_Hub 등과 같은 오픈 데이터셋(open dataset)을 통해 수집하였으며, 수집된 연기 및 불꽃 이미지들은 건축물 내의 다양한 가연물 및 재료 등에서 발생하는 연기 및 불꽃을 촬영한 이미지이다. 또한, 수집된 데이터의 신뢰도 향상을 위해 데이터 전처리 방식으로 bounding box, polygon 및 augmentation 기법을 적용하였다. Bounding box 기법과 polygon 기법에서는 각각 3,500개의 이미지를 사용하였고, augmentation 기법에서는 입력 이미지 데이터에 채도, 밝기, 노출 등을 적용하여 증강된 41,300개의 이미지를 사용하였다. 훈련, 검증, 평가에 활용된 이미지 데이터에 대한 구체적인 구성은 표 2에 나타내었다.



Fig. 6. Image dataset.

Table 2. Dataset conditions.

Image data processing method	Train	Val	Test	Total
Bounding box	2,800	350	350	3,500
Polygon	2,800	350	350	3,500
Augmentation	36,400	4,550	350	41,300

3.3 Data labeling

오픈 데이터셋을 통해 수집된 화재 이미지들에 대해서는 다음과 같은 과정을 수행하였다. 우선 입력 이미지 데이터의 크기를 조정하여 640×640 픽셀 크기로 규격화하고, 이후 bounding box 방식과 polygon 방식을 사용하여 데이터 라벨링을 수행하였다. Bounding box는 물체를 직사각형 모양의 박스 안에 포함되도록 그리는 라벨링 방법으로 각각의 bounding box는 x, y, w, h와 confidence로 구성된다. 여기서 x, y는 bounding box의 중심점을 의미하며, w, h는 bounding box의 넓이와 높이에 해당한다. 또한 confidence는 bounding box 내에 객체가 있는지 없는지에 대한 확률에 해당한다. 이와 같은 bounding box 방식은 객체 외의 주변 배경을 화재로 인식할 수 있어 객체 인식에 대한 정확도가 저하될 수 있는 단점이 있다. 반면 polygon 방식의 경우 객체의 특정 부분을 정확하게 라벨링하여 bounding box를 생성하며, 주변 배경을 최소화할 수 있다는 특징이 있다. 그림 7은 이미지 데이터에 대한 bounding box 방식과 polygon 방식의 라벨링 전처리를 보여주고 있다.

3.4 데이터 증강

화재는 재료의 종류와 주변 환경에 따라 연기와 불꽃의 크기 및 색상이 변화하며, 명확한 형태가 없어 정확하게 연기와 불꽃을 탐지하기 매우 어렵다. 화재 시 발생하는 연기와 불꽃을 정확하게 탐지하기 위해서는 연기와 불꽃의 형태가 명확한 데이터와 충분한 양의 데이터를 필요로 한다. 따라서 데이터 확보에 대한 문제를 해결하기 위해서 본 연구에서는 데이터 증강(augmentation) 기법을 사용하였다. 일반적으로 데이터 증강은 사용된 이미지 데이터셋을 회전(rotating), 뒤집기(flipping), 확대(zooming) 등의 방식을 통해 데이터셋을 증강한다. 본 연구에서는 데이터의 다양성을 확보하고 과대적합을 방지 등을 고려하여 화재 이미지에 대해 채도(saturation), 밝기(brightness), 노출(exposure) 등을 반영하여 데이터셋 증강을 수행하였다. 채도, 밝기, 노출은 각각 $\pm 10\%$, $\pm 20\%$ 를 적용하였다. 채도 조절을 통해 불꽃 색상의 다양성과 색상을 강화하였고, 밝기 조절을 통하여 어두운 환경에서 화재를 감지하도록 하였다. 마지막으로 화재 상황에서 강한 빛으로 인한 오검출을 방지하기 위하여 노출을 조절하였다. 그림 8은 채도, 밝기, 노출 등의 데이터 증강을 적용한 불꽃 이미지들을 보여주고 있다.

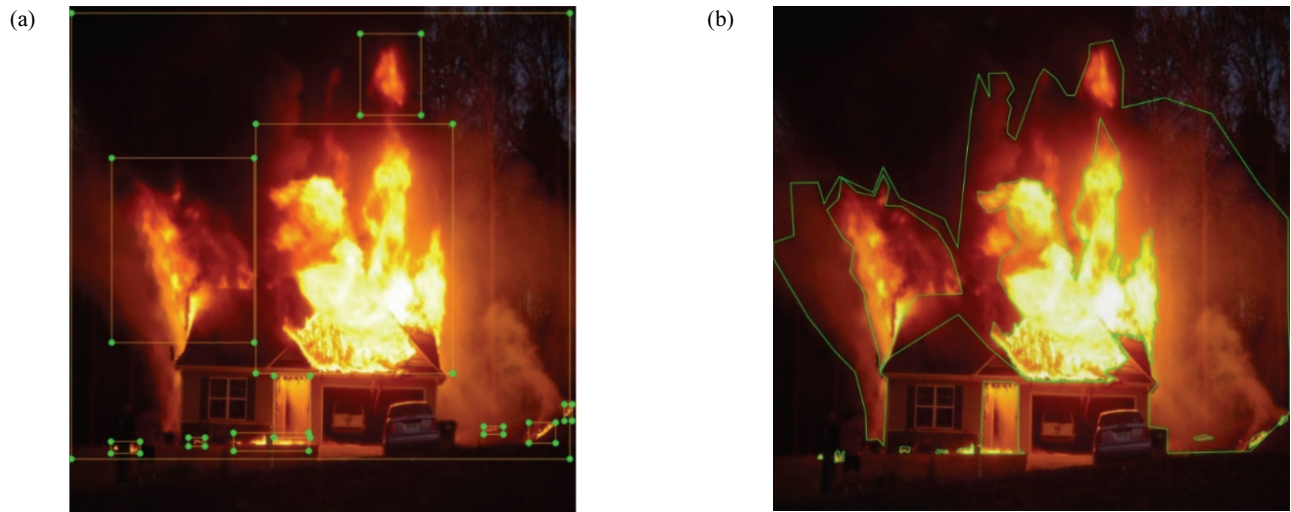


Fig. 7. Data Labeling. (a) Bounding box and (b) polygon.

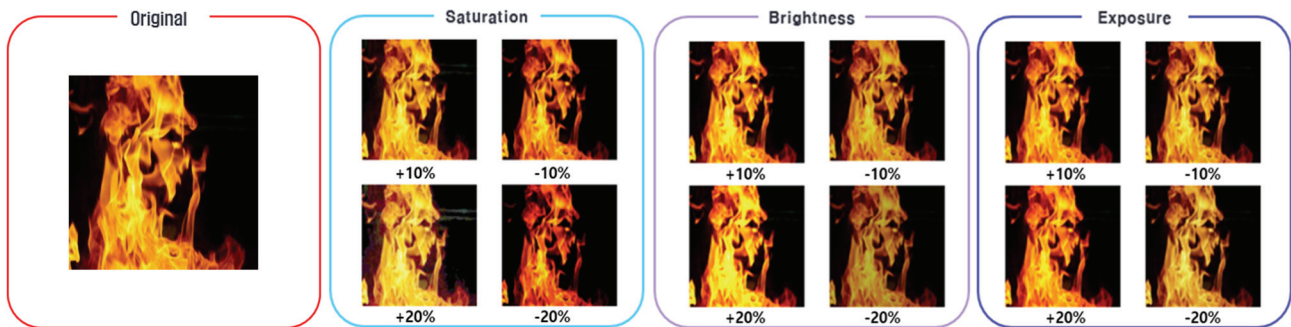


Fig. 8. Data augmentation.

3.5 Optimizer

본 연구에서는 화재객체 탐지 모델로 선정된 YOLOv5s6.0 모델의 손실 함수를 최적화하고 모델을 빠르게 학습시킬 수 있도록 Adam, AdamW, SGD optimizer의 precision, recall, mAP의 성능 비교실험을 수행하여 YOLOv5s6.0 모델의 optimizer 선정하고자 하였다. 실험은 총 3,500개의 화재 이미지에 대하여 train dataset 2,800개, validation dataset 350개, test dataset 350개로 구분하였으며, 데이터 라벨링은 bounding box 기법을 적용하였다. 각 optimizer에 대한 실험 결과는 다음과 같다. Adam은 precision 0.51, recall 0.53, mAP 0.46의 성능을 보였고, AdamW는 precision 0.52, recall 0.53, mAP 0.48

의 성능으로 Adam과 AdamW는 비슷한 성능을 나타냈다. 반면, SGD는 Precision 0.81, recall 0.6, mAP 0.69의 성능으로 Adam과 AdamW보다 높은 성능을 보였다. 따라서 본 연구에서는 YOLOv5s6.0 모델의 optimizer로 SGD를 선정하여 실험을 진행하였다. 그림 9는 optimizer별 객체 검증 성능 결과를 보여주고 있다.

3.6 모델 학습

표 3은 본 연구에서 사용한 화재 탐지 모델의 학습 조건을 보여주고 있다. 화재 탐지 모델은 YOLOv5s 모델을 사용하였고, optimizer는 SGD를 적용하였다.

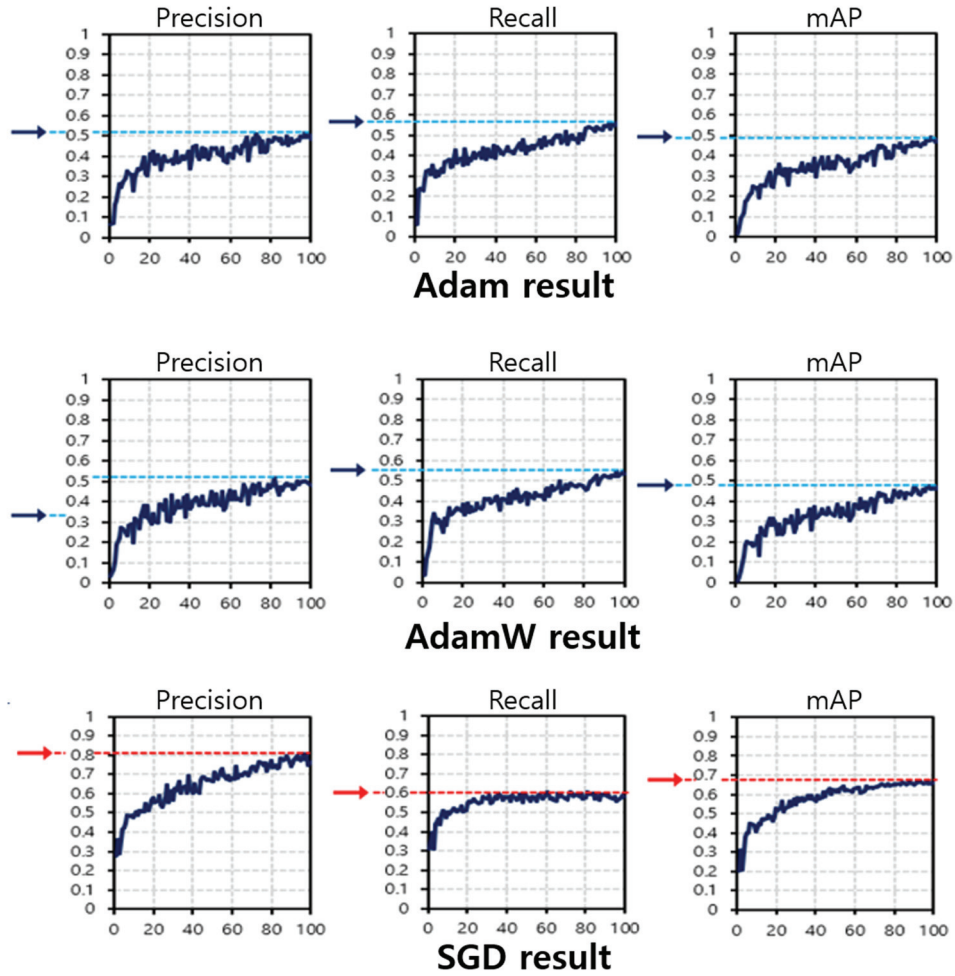


Fig. 9. Test results of each optimizer type.

Table 3. Training fire detection model.

Training parameters	Details
Epochs	100
Batch size	16
Image size (pixels)	640×640
Optimizer algorithm	SGD
Initial learning rate	0.01
Model	YOLOv5s

3.7 성능 평가 지표

모델의 성능을 평가하기 위해서 confusion matrix 분석을 수행하였으며, confusion matrix 분석은 true positive, true negative, false positive, false negative를 통하여 모델을 성능을 판별할 수 있다 [24].

Precision은 정확도를 의미하며 검출 결과들 중 옳게 검출한 비율을 의미한다. Precision에 대한 수식을 식 (1)에 나타냈다. Recall은 재현율을 의미하며, 실제 옳게 검출된 결과물 중에서 옳다고 예측한 것의 비율을 의미한다. Recall의 수식을 식 (2)에 나타냈다.

$$Precision = \frac{TP}{TP+FP} \tag{1}$$

$$Recall = \frac{TP}{TP+FN} \tag{2}$$

여기서, true positive (TP)는 실제 true인 정답을 true라고 예측한 결과이고, false positive (FP)는 실제 false인 정답을 true라고 예측한 결과이다. 또한, false negative (FN)는 실제 true인 정답을 false라고 예측한 결과이고, true negative (TN)는 실제 false인 정답을 false라고 예측한 결과에 해당한다.

한편, PR curve는 confidence의 threshold 값에 따른 precision과 recall의 변화를 나타낸 곡선으로 x축이 recall, y축이 precision이다. PR curve에서는 confidence threshold에 따라 precision과 recall의 trade-off 관계가 있다. 이러한 PR curve의 면적이 AP (average precision)에 해당한다. AP (average precision)는 한 종류의 객체에 대한 모델의 평균 정확도로 객체 탐지 모델의 정확도 평가 지표로 사용되며, AP에 대한 수식은 식 (3)에 나타냈다.

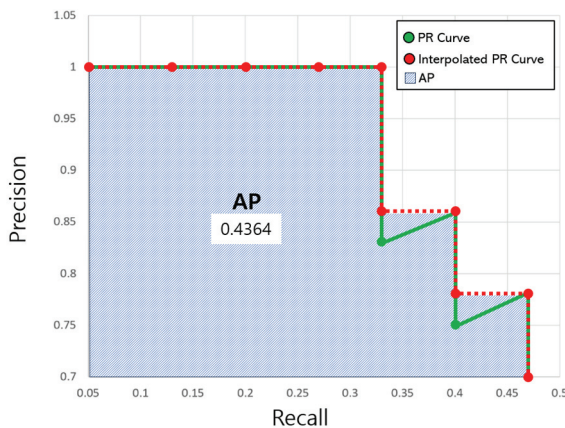


Fig. 10. Examples of PR Curve & AP0.

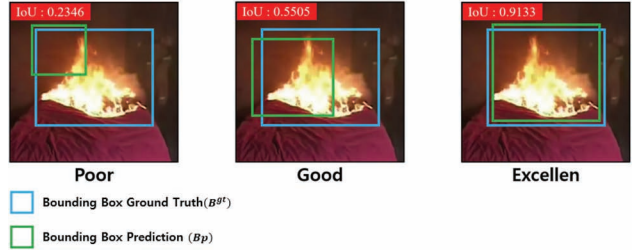


Fig. 11. Examples of IoU.

mAP (mean average precision)는 모든 종류의 객체에 대한 AP의 평균을 의미하며, 식 (4)는 mAP에 대한 수식이다. 그림 10은 PR curve, interpolated PR curve, AP를 보여주고 있다 [25].

$$AP = \sum_{k=0}^{k=n-1} [Recalls(k) - Recalls(k + 1)] \times Precisions(k) \tag{3}$$

$$mAP = \frac{1}{n} \sum_{i=1}^n AP_i \tag{4}$$

IoU (intersection over union)는 객체 탐지 모델에서 2개의 영역이 겹쳐져 있는 정도를 표시하며, 모델이 얼마나 객체 탐지를 정확히 수행했는지에 대한 정확도의 측정 지표로 활용된다. 객체 탐지에서 실제 객체 위치와 예측 위치가 정확히 일치하기 어렵기 때문에 정확도를 간단히 평가하기가 어렵다. 그림 11과 같이 실제 bounding box인 bounding box ground truth와 객체 탐지 모델이 예측한 bounding box인 bounding box prediction의 겹침 정도를 계산하여 객체를 탐지한다. IoU 값은 0.5 이상이면 'Good'으로 보기 때문에 일반적으로 사용하는 IoU의 threshold 값을 0.5로 설정하였다.

4. 실험 및 실험 결과

본 연구에서는 실시간 화재객체 탐지 모델의 성능 향상을 확인하고자 YOLOv5s6.0 모델의 입력 이미지 데이터에 대해 전처리 과정으로 bounding box 방식, polygon 방식, augmentation 방식을 적용하였다. 또한, 전처리 방식별 loss, precision, recall, mAP, FPS 등의 모델의 성능 비교를 수행하였고, 실제 화재 CCTV 영상 이미지에 대해 개발된 YOLOv5s6.0 모델을 적용하여 연기 및 불꽃 검증에 대한 성

능 평가 실험을 진행하였다.

4.1 화재객체 탐지 모델의 학습 및 검증 결과

그림 12는 YOLOv5s6.0 모델에서 입력 이미지 데이터 전처리 방식별 화재객체 탐지 모델에 대한 box_loss와 object_loss 결과이다. 학습 및 검증이 반복될수록 모든 전처리 방식에 대해 box_loss와 object_loss가 모두 감소하는 경향을 보였다. Box_loss의 결과는 bounding box 방식의 training_loss 0.0261, validation_loss 0.0447이고, polygon 방식의 training_loss 0.0133, validation_loss 0.0241이며, augmentation 방식의 training_loss 0.0181, validation_loss 0.0198이었다. 한편, object_loss 결과는 bounding box 방식의 training_loss 0.01471, validation_loss 0.0187이고, polygon 방식의 training_loss 0.0081, validation_loss 0.0101이며, augmentation training_loss 0.0109, validation_loss 0.0094를 확인하

였다. 따라서 bounding box 방식에 비해 polygon 방식과 augmentation 방식이 box_loss, object_loss가 낮아 손실 측면에서 우수한 성능을 보이는 것을 확인하였다.

그림 13은 입력 이미지 데이터 전처리 방식별 화재객체 탐지 모델에 대한 학습 및 검증에 대한 성능 결과이다. 모든 이미지 데이터 전처리 방식에서 학습 및 검증이 반복될수록 precision, recall 및 mAP가 증가하는 경향을 보였다. 실험 결과를 보면 학습 초반 polygon 방식의 precision, recall 및 mAP 성능이 다른 방식들에 비해 빠르게 향상되었다. 이는 연기, 불꽃같이 다양한 색상을 띠고, 형태가 일정하지 않는 무정형의 객체에 대해 polygon 방식은 화재 주변 배경을 최소화기 때문에 초기 학습에 있어 화재 탐지 성능이 상대적으로 우수한 값을 보인 것으로 판단된다. 이후 데이터 수와 데이터의 다양성 등의 한계로 인해 precision, recall 및 mAP 성능의 추가적인 상승으로 이어지지 않았다.

반면, augmentation 방식의 경우 기존의 bounding

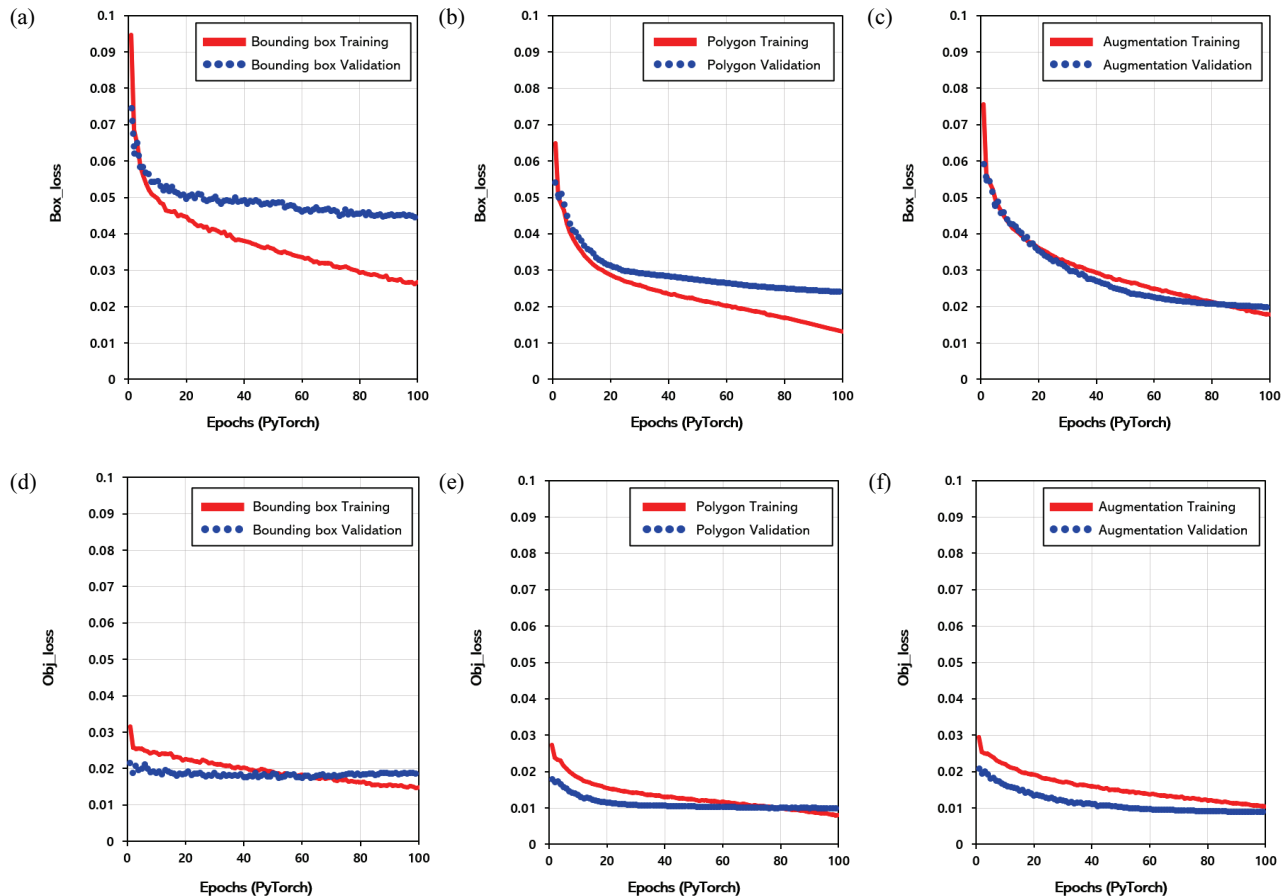


Fig. 12. Training and validation loss. (a) Bounding box, (b) polygon, (c) augmentation, (d) bounding box, (e) polygon, and (f) augmentation.

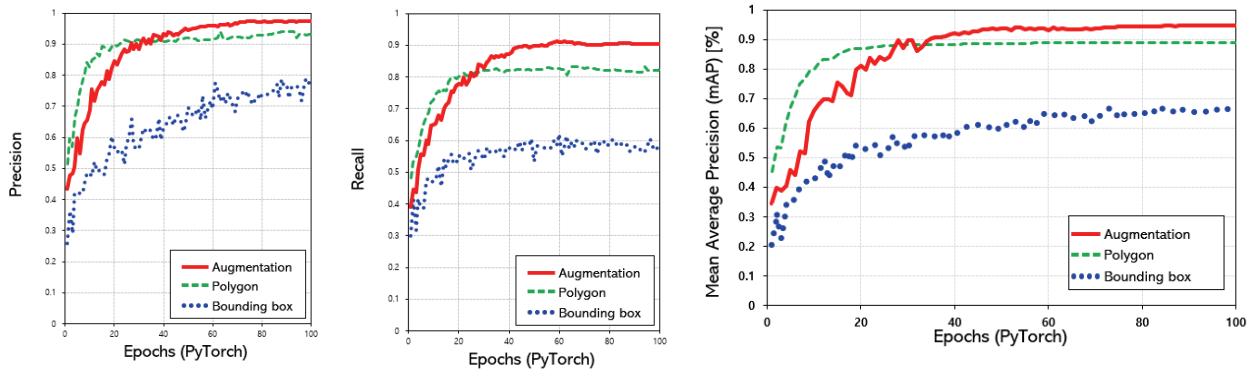


Fig. 13. Precision, recall and mAP results of each data labeling type.

box와 동일한 라벨링을 사용하여 초반 precision, recall 및 mAP의 성능은 polygon 방식에 비해 상대적으로 낮게 나타났지만, 데이터 증강의 통해 데이터 다양성 확보 및 epoch의 반복적인 학습을 통해 epoch 30 시점부터 polygon 방식에 비해 성능이 향상된 결과를 보였다. 결과적으로 augmentation 방식은 precision 0.9737, recall 0.9045, mAP 0.9465로 가장 우수한 성능을 보였으며, 기존의 bounding box 방식보다 precision 0.1823 (23.04%), recall 0.2835 (45.64%), mAP 0.2791 (41.70%) 성능이 개선됨을 확인하였다.

그림 14는 입력 이미지 데이터 전처리 방식별 화재객체 탐지 모델에 대한 FPS와 mAP를 보여주고 있다. FPS의 경우 모든 방식이 비슷한 성능의 FPS를 보이지만, mAP의 경우 augmentation 방식이 0.9509로 가장 우수함을 보였다. 이러한 결과는 FPS가 전처리 시간, 추론 시간 및 NMS (none maximum suppression) 시간 등으로 인해 결정되며, 동일한 파라미터 수와 동일한 YOLOv5s6.0 모델을 통해 학습을 수행했기 때문에 이미지 데이터셋의 수

차이가 FPS에 직접적인 영향을 주지 않았고, 따라서 각 모델에 대한 FPS 성능은 유사한 값을 보였다고 판단된다. 또한, augmentation 방식의 경우 많은 양의 데이터셋에 대해 학습을 수행했기 때문에 입력 이미지에 대한 객체 탐지를 좀 더 수월하게 진행할 수 있어서 비록 전처리 방식별로 FPS 값의 차이는 크지 않지만, 그중에서 augmentation 방식이 가장 빠른 FPS를 보였다고 생각된다. 표 4는 입력 이미지 데이터 전처리 방식별 화재객체 탐지 모델의 성능 결과를 나타냈다.

그림 15는 고흥병원과 이천쿠팡물류센터에서 발생한 전기 합선으로 인한 화재 발생 CCTV 실제 영상이다. Bounding box 방식의 경우 두 장소의 화재를 모두 탐지하지 못하였으며, polygon 방식은 고흥병원 화재의 ① 위치 화재는 인식하지 못하고, ② 위치 화재에 대한 신뢰도는 0.82를 보였다. 하지만 이천쿠팡물류센터의 ③ 위치, ④ 위치 화재에 대해서는 전혀 탐지하지 못하였다. Augmentation 방식은 고흥병원의 ① 위치 화재에 대한 신뢰도는 0.79를 보였고, ② 위치 화재에 대해서 신뢰도는 0.82를 보였다. 또한 쿠팡물류센터 화재의 경우 ③ 위치 화재에 대해서 신뢰도 0.26, ④ 위치 화재에 대해서 신뢰도 0.90으로 두 장소에서 발생한 모든 화재를 높은 신뢰도로 정확하게 탐지하였다.

이러한 실험 결과는 bounding box 방식과 polygon 방식의 경우 객체에 대한 데이터의 다양성과 데이터양의 부족으로 고흥병원 화재와 이천쿠팡물류센터 화재에 대한 낮은 탐지 성능을 보였다고 판단된다. 반면, 채도, 밝기, 노출 등을 통해 화재 이미지 데이터의 양이 충분한 augmentation 방식의 경우 모든 화재에 대해 화재 탐지 성능이 다른 방식에 비해 비교적 우수하게 나타났다.

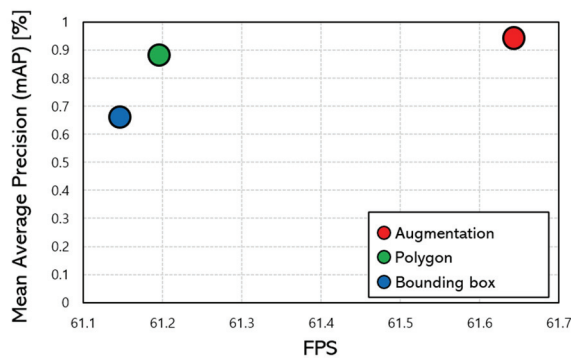
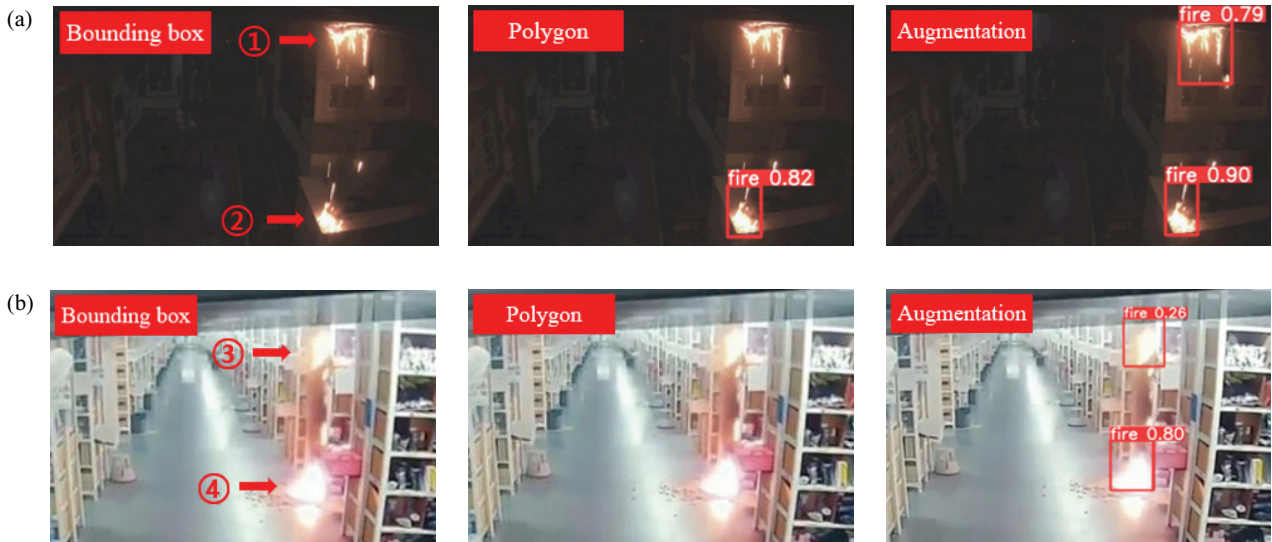


Fig. 14. FPS results of each data labeling type.

Table 4. Performance results by model.

Model	Dataset	Precision	Recall	Last mAP	Best mAP	FPS
Yolov5s6.0	Bounding box	0.7914	0.6211	0.6506	0.6674	61.1468
Yolov5s6.0	Polygon	0.9415	0.8165	0.8897	0.8908	61.1966
Yolov5s6.0	Augmentation	0.9737	0.9045	0.9505	0.9465	61.6426

**Fig. 15.** Validation of model performance through CCTV footage capturing actual fire incidents. (a) Goheung hospital and (b) Coupang distribution center.

5. 결론 및 향후 연구

본 연구에서는 선행연구를 분석하여 실시간 화재객체 탐지에 적합한 object detection 모델을 선정하였으며, 모델 성능을 향상시키기 위해 optimizer 검토를 진행하였다. 모델 개발을 위한 화재 이미지는 AI-Hub와 Kaggle 등 Opendata set을 통해 수집하였으며, 수집한 이미지는 선별을 통해 총 3,500장의 이미지를 사용하였다. 화재 이미지는 resize를 통해 이미지의 크기를 640×640으로 변환하였다. 또한 입력 이미지 데이터에 대해 각각 bounding box labeling, polygon labeling, data augmentation을 통하여 전처리를 진행하였다. 한편, 화재의 다양성을 확보하기 위해 화재 이미지에 대해 augmentation을 수행하였으며, 주변환경에 따라 변화하는 채도, 밝기, 노출을 각각 -10%, +10%, -20%, +20% 등을 적용하였다. 또한 개발 모델에 대한 성능 검증을 위해 고흥병원 화재 및 이천쿠팡 물류센터 화재의 실제 영상 이미지에 대해 모델 성능 실험을 수행하였다. 실험 결과, augmentation 방식은 기존의 bounding box 대비 precision 0.1823 (23.04%), recall

0.2835 (45.64%), mAP 0.2791 (41.70%), FPS 0.4958 (0.81%) 성능이 개선됨을 확인하였다. 이러한 결과로부터 화재로 발생하는 연기 및 불꽃같은 다양한 색상을 띠고, 무정형의 객체 탐지는 데이터를 polygon 방식과 같이 구체적 형태의 전처리 방식도 중요하지만, 데이터의 다양성과 충분한 데이터양이 기반이 되어야 된다고 생각된다.

본 연구에서 개발된 실시간 화재객체 탐지 모델은 기존의 CCTV 등의 장비에 적용될 수 있으며, 다양한 재료에서 발생하는 화재를 조기에 탐지가 가능해 화재 피해 최소화에 기여할 수 있을 것으로 판단된다.

향후 연구에서는 연기와 불꽃의 통해 재료의 종류를 식별하는 모델 및 연기와 불꽃에 대한 검증 정확도 향상과 유체의 유동을 예측하기 위해 polygon 방식보다 정밀한 이미지 데이터 라벨링 방식인 SAM (segment anything model), TAM (track anything model) segment 기법을 적용하고 이에 대한 다양한 데이터 증강 기법을 적용하여 보다 빠르고 정확한 딥러닝 기반의 화재 감지 모델을 개발하고자 한다.

ORCID

Bohee Lee
Haiyoung Jung

<https://orcid.org/0000-0002-8363-437X>
<https://orcid.org/0000-0001-5531-2923>

감사의 글

이 논문은 2023학년도 세명대학교 교내학술연구비 지원에 의해 수행된 연구임.

REFERENCES

- [1] T. Kim, M. Yoon, and J. Lee, *J. Korean Soc. Hazard Mitigation*, **19**, 1 (2019).
doi: <https://doi.org/10.9798/KOSHAM.2019.19.2.1>
- [2] J. S. You, K. S. Kang, J. S. Lee, and Y. J. Chung, *Appl. Chem. Eng.*, **34**, 421 (2023).
doi: <https://doi.org/10.14478/ace.2023.1056>
- [3] J. You and Y. J. Chung, *Fire Sci. Eng.*, **33**, 69 (2019).
doi: <https://doi.org/10.7731/KIFSE.2019.33.1.069>
- [4] A. H. Awad, R. E. Gamasy, A.A.E. Wahab, and M. H. Abdellatif, *Eng. Sci.*, **4**, 34 (2019).
doi: <https://doi.org/10.11648/j.es.20190402.12>
- [5] J. G. Kim, C. I. Park, J. W. Jung, and S. G. Kim, *J. Soc. Disaster Inf.*, **17**, 60 (2021).
doi: <https://doi.org/10.15683/kosdi.2021.3.31.060>
- [6] S. K. Kim, S. G. Choi, and C. H. Lee, *Fire Sci. Eng.*, **31**, 42 (2017).
doi: <https://doi.org/10.7731/KIFSE.2017.31.1.042>
- [7] C. S. Park and T. H. Oh, *Korean J. Secur. Convergence Manage.*, **10**, 249 (2021).
doi: <https://doi.org/10.24826/kscs.10.6.16>
- [8] S. M. Park and S. K. Kim, *Fire Sci. Eng.*, **31**, 111 (2017).
doi: <https://doi.org/10.7731/KIFSE.2017.31.4.111>
- [9] S. K. Kim, D. S. Gum, and C. H. Lee, *Fire Sci. Eng.*, **29**, 57 (2015).
doi: <https://doi.org/10.7731/KIFSE.2015.29.5.057>
- [10] S. Choi, S. Kim, and H. Jung, *Electronics*, **12**, 2579 (2023).
doi: <https://doi.org/10.3390/electronics12122579>
- [11] K. M. Shong, Y. S. Roh, and H. R. Kwak, *J. Korean Inst. Illum. Electr. Install. Eng.*, **20**, 71 (2006).
doi: <https://doi.org/10.5207/JIEIE.2006.20.8.071>
- [12] M. S. Park and C. S. Choi, *Fire Sci. Eng.*, **34**, 101 (2020).
doi: <https://doi.org/10.7731/KIFSE.96dca8eb>
- [13] B. S. Koo, G. H. Jung, A. H. Kim, and H. J. Lee, *Proc. Korea Information Processing Society Conference* (2020) p. 312.
doi: <https://doi.org/10.3745/PKIPS.y2020m11a.312>
- [14] J. S. Park, H. T. Kim, and Y. S. Yu, *J. Korea Inst. Electron. Commun. Sci.*, **6**, 206 (2011).
doi: <https://doi.org/10.13067/JKIECS.2011.6.2.206>
- [15] H. J. Kim, D. H. Lee, A. Niaz, C. Y. Kim, A. A. Memon, and K. N. Choi, *IEEE Access*, **9**, 11694 (2021).
doi: <https://doi.org/10.1109/ACCESS.2021.3051424>
- [16] R. Girshick, J. Donahue, T. Darrell, and J. Malik, *2014 IEEE Conference on Computer Vision and Pattern Recognition* (IEEE, Columbus, USA, 2014) p. 580.
doi: <https://doi.org/10.1109/cvpr.2014.81>
- [17] Q. Zhang, J. Xu, L. Xu, and H. Guo, *Proc. 2016 International Forum on Management, Education and Information Technology Application* (Atlantis Press, 2016).
doi: <https://doi.org/10.2991/ifmeita-16.2016.105>
- [18] P. Li and W. Zhao, *Case Stud. Therm. Eng.*, **19**, 100625 (2020).
doi: <https://doi.org/10.1016/j.csite.2020.100625>
- [19] R. Girshick, *Proc. 2015 IEEE International Conference on Computer Vision (ICCV)* (IEEE, Santiago, Chile, 2015) p. 1440.
doi: <https://doi.org/10.1109/ICCV.2015.169>
- [20] S. Ren, K. He, R. Girshick, and J. Sun, *IEEE Trans. Pattern Anal. Mach. Intell.*, **39**, 1137 (2017).
doi: <https://doi.org/10.1109/tpami.2016.2577031>
- [21] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (IEEE, Las Vegas, USA, 2016) p. 779.
doi: <https://doi.org/10.1109/CVPR.2016.91>
- [22] B. V. Kumar, S. Abirami, R. J. Bharathi Lakshmi, R. Lohitha, and R. B. Udhaya, *IOP Conference Series: Materials Science and Engineering* (IOP Publishing Ltd, Tamil Nadu, India, 2019).
doi: <https://doi.org/10.1088/1757-899X/590/1/012062>
- [23] G. Jocher, A. Chaurasia, A. Stoken, J. Borovec, NanoCode012, Y. Kwon, T. Xie, K. Michael, J. Fang, imyhxy, Lorna, C. Wong, Y. Zeng, V. Abhiram, D. Montes, Z. Wang, C. Fati, J. Nadar, Laughing, UnglvKitDe, tkianai, yxNONG, P. Skalski, A. Hogan, M. Strobel, M. Jain, L. Mammana, and xylieong, *Zenodo, Ultralytics/yolov5: v6.2 - YOLOv5 Classification Models, Apple ML, Reproducibility, ClearML and Deci.ai Integrations*, (2022).
<https://zenodo.org/records/7002879>
- [24] K. M. Ting, *Confusion Matrix, Encyclopedia of Machine Learning and Data Mining* (2017) p. 260.
doi: https://doi.org/10.1007/978-1-4899-7687-1_50
- [25] R. Padilla, S. L. Netto, and E.A.B. da Silva, *Proc. 2020 International Conference on Systems, Signals and Image Processing (IWSSIP)* (IEEE, Niteroi, Brazil, 2020) p. 237.
doi: <https://doi.org/10.1109/IWSSIP48289.2020.9145130>