

딥러닝 기반 화재감지 시스템 구현

강한나라 · 임채영 · 유운섭

국립한경대학교

Implementation of Fire Detection System Based on Deep Learning

Han-na-ra Kang · Chae-young Lim · Yun Seop Yu

Hankyung National University

E-mail : wifi0309@hknu.ac.kr / dlacodud@hknu.ac.kr

요 약

화재는 국소적인 불로부터 시작되어 인적피해와 물질 피해 등 여러 피해를 입히고 있다. 따라서 초기 화재를 신속하고 정확하게 감지해서 조치하는 방법이 요구된다. 본 논문에서는 화재를 빠르게 감지하는데 도움을 주기 위해서 화재감지 방법을 제안한다. 제안하는 방법은 화재 이미지 데이터를 수집한 후 Adam optimizer와 Nadam optimizer CNN, Faster-RCNN, SSD-MobileNet의 4가지 모델링을 사용하여 학습시킨다. 학습시킨 데이터를 바탕으로 화재를 감지한다. 화재의 감지 정확도가 4가지 방법 모두 90% 정도이고 Adam optimizer를 가진 CNN이 가장 좋은 성능을 가진다.

ABSTRACT

Fires originate from local fires and cause various damages, including personal injury and material damage. Therefore, there is a need for a method for quickly and accurately detecting and acting on an initial fire. In this paper, a fire detection method is proposed to help quickly detect a fire. The proposed method collects fire image data and trains it using modeling of CNN with Adam optimizer and Nadam optimizer, Faster-RCNN, and SSD-MobileNet. The fire is detected based on the learned data. As a result of conducting fire detection with various images, it was confirmed that the actual fire detection accuracy was high. CNN with Adam optimizer is best performance among them.

키워드

Fire, Fire Detection, Deep Learning, Convolution Neural Network, Faster-RCNN, SSD-MobileNet

1. 서 론

대부분의 화재는 국소적인 불로부터 시작 되어 인적피해와 물질 피해 등 여러 피해를 입히고 있다. 국소적인 초기 화재를 신속하고 정확하게 감지하여 조치하는 방법은 인적피해와 물질 피해를 줄이는 중요한 역할이 가능하다. 현재의 화재 감지 시스템은 스모그, 불꽃, 열기 등의 감지센서 및 카메라를 통한 영상처리 기반 화재감지 시스템으로 구분이 되는데 이러한 시스템들의 정확도는 매우 낮다.

연막소독, 쓰레기 소각, 연기, 냄새, 방화기, 경보 오동작 등의 이유로 오인출동은 95% 신뢰수준에서 연평균 48,371±5,763건이 발생하였고, 이는 화재진압을 위한 출동보다 약 1.2배 높은 수준이다[1]. 뿐만 아니라 2018년 기준 소화설비가 갖춰진 시설

1,717곳에서 소규모 화재로 인하여 미작동하거나 대규모 화재 시에 동작을 하지 않고 무효한 경우는 1,082 곳으로 절반 이상이 소화설비 작동 정확도가 떨어진다는[2].

따라서 본 논문에서는 화재 감지 시스템의 낮은 정확도 따른 불필요한 사회적 비용을 줄이고 화재를 정확하게 감지할 수 있는 화재 감지 방법에 대한 연구를 한다. Tensorflow를 이용한 딥러닝 시스템을 통하여 화재 감지 시스템을 연구하고 Adam optimizer와 Nadam optimizer의 Custom-Convolution Neural Network (CNN), Faster-RCNN, SSD-MobileNet 이 4가지의 모델링 네트워크를 사용하여 화재 이미지를 학습하여 화재를 감지하는 방법에 대해 연구한다.

II. 관련 연구

2.1 Deep Learning

딥러닝은 여러 비선형 변환 기법을 이용해서 학습 데이터에 대한 높은 수준의 추상화를 통해 데이터의 주요 특징들을 추출한 다음, 데이터 군집화 및 분류를 수행하는 기계학습의 일종으로 Deep Belief Network, Recurrent Neural Network, CNN 등을 활용하는 방법이 있다[3].

2.2 CNN

CNN은 컨볼루션 계층(convolution layer)과 풀링 계층(pooling layer or sub-sampling layer) 입력을 필터로 합성곱하여 특징을 추출하고 이런 합성곱 레이어를 여러 계층으로 연결한다. 합성곱 레이어를 지날 때 마다 저수준의 이미지 특징들이 점차 고수준의 특징들로 만들어진다. 그리고 마지막에는 완전연결 레이어로 최종 결과를 학습한다.

Adam optimizer(Adaptive Moment Estimation)은 RMSProp과 Momentum방식을 결합한 알고리즘이다. 계산해온 기울기의 지수평균을 저장하며, RMS Prop과 유사하게 기울기의 제곱값의 지수평균을 저장한다[4].

Nadam optimizer(Nesterov-accelerated Adaptive Moment Estimation)은 Adam과 NAG를 결합한 알고리즘이다. Nadam은 NAG의 철학을 바탕으로 현재 위치에서 Gradient 값을 계산하는 것이 아닌 Momentum 방향으로 이동한 뒤, 새로운 위치에서 Gradient 값을 계산해내게 된다[4].

2.3 Faster-RCNN

Fast-RCNN에서 가장 큰 계산부하를 차지하는 region proposal 생성을 새로운 방식으로 대체하고, 이를 모델 내부로 통합시키는 것이다. 이에 고안된 것이 바로 Fast-RCNN에서 RPN(Region Proposal Networks)를 통합한 Faster-RCNN이다[5].

2.4 SSD(Single Shot Multibox Detector)

output을 만드는 공간을 나누고(multi feature map) 각 feature map에서 다른 비율과 스케일로 default box를 생성하고 모델을 통해 계산된 좌표와 클래스값에 default box를 활용해 최종 bounding box를 생성한다[6].

III. Tensorflow를 활용한 범용적 화재 감지

TensorFlow를 이용한 화재감지를 하기 위한 4가지의 모델 방법으로 다음과 같은 순서대로 진행한다. 첫 번째는 custom-CNN의 네트워크방법으로는 CONV => RELU => POOL 2번, (CONV => RELU) *2 => POOL 1번, 완전히 커넥트된 레이어 층에 ReLU함수 2번, 마지막으로 softmax activate

function을 사용하여 분류한다. 이미지는 화재 이미지 3863개 비화재 이미지 2688개를 사용하였으며 훈련이미지와 테스트이미지를 8.5:1.5로 하였고 sklearn package의 metrics package의 정밀도, 재현율, F1 score를 구하는 Classification report 명령을 사용하였다. 이미지를 로드 할 때 128*128 픽셀로 고정하며 종횡비는 무시한다. 화재 라벨을 1 비화재 라벨을 0으로 하고 위의 네트워크 모델을 빌드하며 손실함수로 binary_crossentropy를 사용 하고 학습을 할 때의 optimizer를 Adam, Nadam 한번씩 적용해두면서 훈련을 시작한다. 훈련이 끝난 후에 샘플이미지를 로드해서 모델에서 정한 종횡비를 무시한 128*128픽셀로 고정하여 이미지를 예측해서 출력사진에 표시한다.

두 번째는 Faster-RCNN의 방법이다. 화재에 직접 상자를 Labeling한 이미지가 총 600개이며 480개의 train 이미지와 120개의 test 이미지(8:2)로 구성을 하였다. LabelImg 프로그램을 사용하여 이미지를 xml 파일로 저장 후 csv파일로 변환하여 저장 후 미리 Tensorflow detection model zoo(github)에서 다운해놓은 faster_rcnn_inception_v2_coco모델을 통해서 훈련을 시작한다. 훈련이 끝난 후에 폴더에 있는 샘플이미지를 로드하고 이미지에 화재가 예측된 곳에 박스를 그리고 그 위에 화재가 몇 퍼센트로 예측이 되었는지 출력사진에 표시한다. 또한 70% 미만의 퍼센트의 박스는 그려지지 않는다.

세 번째는 SSD-MobileNet의 모델링 방법이다. 이 모델링은 위의 Faster-RCNN의 방법과 같은 object-detection의 방법으로 화재를 검출한다. 위의 Faster-RCNN의 방법과 같은 이미지와 같은 Labeling을 하였으며 구성도 같다. 다만 모델을 Tensorflow detection model zoo (github)에서 다운해놓은 ssd_mobilenet_v2_quantized_300x300_coco모델을 사용하였으며 Faster-RCNN과 이미지의 예측방법도 같다.

IV. Test 및 결과분석

본 화재 감지 시스템의 화재 감지를 위한 학습 환경은 Tensorflow와 Tensorflow-gpu python, OpenCV, Keras 딥러닝 프레임워크를 이용해서 제안된 방법을 구현하였다. 동작을 위한 하드웨어로 Ryzen7-3700x CPU, 16GB RAM Geforce RTX2060 super GPU 8GB VRAM 이 사용되었다.

첫 번째 화재 감지를 위한 Custom-CNN 모델 학습의 성능은 그림1의 그래프처럼 Adam optimizer의 방법으로는 97%의 화재 검출률을 보였고 Nadam optimizer의 경우 92%의 화재 검출률을 보인다. 그림2은 화재 상황이 아니어도 불구하고 시스템은 화재상황으로 인식을 하였다.

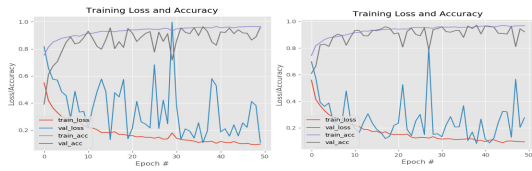


그림 1. Custom-CNN-Adam, Nadam opt graph

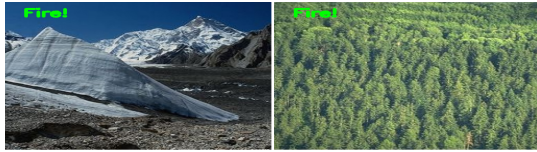


그림 2. 비화재 이미지를 화재로 검출

두 번째 화재 감지를 위한 Faster-RCNN 모델 학습의 성능은 왼쪽 그림 3과 같다. 화재의 손실율은 0.02 밑으로 떨어져서 정확도 98%의 화재 검출을 하며 이미지로 보았을 때 70% 이상의 박스가 표시되는 것을 볼 수 있다. 하지만 유리에 비친 가로등의 빛을 화재로 인식하는 오류도 발생한다.

세 번째 화재 감지를 위한 SSD-MobileNet 모델 학습의 성능은 오른쪽 그림 3과 같다. SSD-MobileNet의 손실율은 20부터 시작해서 40000번의 스텝을 한 결과 2가지의 손실율을 보였으며 화재 이미지 감지로는 특정한 불꽃의 이미지는 99%로 감지하는 반면에 특정 화재 이미지에서는 정확도 70% 미만으로 박스가 표시되지 않는 오류가 발생한다.



그림 3. Faster-RCNN 손실율과 화재 검출

표 1. 제안한 방법과 정확도, 학습시간 비교

Model	accuracy(%)	Training Time (min)
CNN-Adam opt	97%	71
CNN-Nadam opt	92%	71
Faster-RCNN	88%(avg)	93
SSD-MobileNet	87%(avg)	360

표 1은 4가지 방법의 화재 검출율을 나타내고 있고 모두 약 90%의 검출률 성능을 가진다. Adam optimization을 이용한 CNN이 97% 검출률과 학습 시간 71분으로 4가지 방법 중에서 가장 성능이 좋다. 4가지 방법 모두 학습된 이미지라도 빛이나 밝은 이미지, 노을과 같은 적색의 이미지는 화재로

검출되는 경우도 있다. 이를 해결하기 위해서 학습 데이터를 늘리고, 데이터의 모델링 네트워크나 optimizer를 개선해야 한다.

V. 결 론

본 논문에서는 국소적인 화재의 발견에 도움이 될만한 화재 감지 방법을 제안하였다. 수집한 이미지 데이터를 Custom-CNN-Adam opt, Custom-CNN-Nadam opt, Faster-RCNN, SSD-MobileNet을 이용하여 학습 시켰다. 학습된 모델을 사용하여 화재의 정확도를 측정하였다.

실험 결과, Custom-CNN의 경우 대부분의 화재를 잘 감지를 하였으나 몇몇 이미지에서 화재가 아닌 이미지에서 화재로 인식하는 경우가 보였다. 이것은 차후 4000개의 데이터의 양보다 더 늘리고 네트워크의 변경 등을 적용시켜서 발전시킬 수 있을 것이다. optimizer 같은 경우는 Adam opt가 평균적으로 좀 더 나은 정확도를 보여주었다. Faster-RCNN과 SSD-MobileNet은 학습데이터가 600개로 많이 부족했는데 평균 정확도가 낮은 것은 데이터의 부족이 클 것으로 보인다. 향후에 화재 데이터를 늘려 개선하고 발전시킬 수 있을 것이다.

References

- [1] S. Y. Eom, K. J. Kim, and S. K. Lee, "Statistical Analysis and Countermeasure about Fire Mistaken Dispatch," Journal of Korean Institute of Fire Science and Engineering, vol. 27, no. 2, pp. 89-96, Apr. 2013.
- [2] Ministry of Public Safety and Security National Fire Data System, statistics from 2018 [Internet] Available : <http://www.nfds.go.kr/bbs/selectBbsList.do?bbs=B21#>
- [3] Y. J. Kim, E. G. Kim, "Image based Fire Detection using Convolutional Neural Network", Journal of the Korea Institute of Information and Communication Engineering, vol. 20, no. 9, pp. 1649-1656, Sep. 2016.
- [4] S. Ruder "An overview of gradient descent optimization algorithms" [Internet] Available : <https://ruder.io/optimizing-gradient-descent/>.
- [5] S. Ren, K. He, Ross Girshick, and Jian Sun "Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks" [Internet] Available : <https://curt-park.github.io/2017-03-17/faster-rcnn/>.
- [6] W. Liu1, D. Anguelov, D. Erhan, C. Szegedy, S. Reed, C. Y. Fu, A. C. Berg1, "SSD: Single Shot MultiBox Detector" [Internet] Available : <https://taeu.github.io/paper/deeplearning-paper-ssd/>.