



종합학술대회논문집

2020 추계 24권 2호

2020년 10월 30일(금)

온라인 학술대회(ZOOM)

주 최 사단법인 한국정보통신학회

후원기관 한국과학기술단체총연합회, 동의대학교

후 원 사 SK브로드밴드, 대신정보통신(주), (주)대보정보통신, (주)데일리블록체인, SK텔레콤,
네이버시스템(주), 세림TSG(주), 아이씨티웨이, 에스트래픽, 올포랜드, 조인트리,
(주)오픈링크시스템, (주)한스콤정보통신, 진우산전(주), 하이제이컨설팅, LG유플러스

- 30-04** 국내외 사이버보안 훈련 동향 및 고찰
이대성 (부산가톨릭대학교)
- 30-05** 사이버 위기 시나리오 기반 미디어 대응 훈련에 대한 고찰
이대성 (부산가톨릭대학교)
- 31-01** 개인전투체계의 표적 식별 능력 향상을 위한 딥러닝 모델 연구
이종관, 송무준, 김용철, 박상준, 김도경, 백승호 (육군사관학교)
- 31-02** Link-16 성능개량사업에 따른 공중통신중계기 운용방안 제안
권정환, 이준수, 이민우, 임재성 (아주대학교)
- 32-01** 클라우드 기반 스마트 농업을 위한 교육 서비스 모델
김동일, 정희창 (동의대학교, 에어포인트)
- 32-02** 변성기용 반도체 FCL 구조 설계
한만수 (목포대학교)
- 32-03** 복합잡음 환경에서 화소변화를 사용한 스위칭필터 알고리즘
천봉원, 김남호 (부경대학교)
- 32-04** 화소유사성 비교를 사용한 임펄스잡음 제거 알고리즘
천봉원, 김남호 (부경대학교)
- 32-05** S&P 잡음 환경에서 변형된 선형보간법을 이용한 잡음제거 방법
백지현, 김남호 (부경대학교)
- 32-06** 딥러닝 기반 화재감지 시스템과 안드로이드 어플리케이션
강한나라, 임채영, 유운섭 (한경대학교)
- 32-07** 철도 선로 침묵 균열 검출을 위한 3가지 딥러닝 모델 성능 비교
김덕진, 김보천, 이동원, 유운섭 (한경대학교)
- 32-08** 셀프 트레이닝에 도움을 주는 장치 구현
고은성, 김영중, 서방원 (공주대학교)
- 32-09** 스포츠 산업의 해외진출을 위한 경기 기록 활용 방안
김세민, 문정경, 신좌철, 임호균, 이해정, 홍기천, 유강수
(전주교육대학교, 호서대학교, 순천향대학교, 전주대학교)
- 32-10** 초미세먼지 측정 센서 기반 Savitzky-Golay Filter 보정 연구
임병연, 전성우, 이성욱, 송현욱, 정희경 (배재대학교)
- 32-11** 객체 인식 기반 자율주행 차선검출 시스템
전성우, 임병연, 김창수, 정희경 (배재대학교)
- 32-12** 적정기술을 활용한 뉴 스포츠 경기 기록 시스템 설계 : 술런을 중심으로
김세민, 이충호 (전주교육대학교, 한밭대학교)
- 32-13** 스마트기기를 이용하는 응급 정보 서비스 환경 설계
엄상희 (동주대학교)
- 33-01** 다중 카메라 합성 영상 기반 혼합현실 수업
신은지, 조동식 (원광대학교)

딥러닝 기반 화재감지 시스템과 안드로이드 어플리케이션

강한나라 · 임채영 · 유운섭

국립한경대학교

Deep Learning-Based Fire Detection System and Android Application

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

본 논문에서는 라즈베리파이4를 이용한 감시중인 카메라 화면 내에 화재가 발생한 경우, 안드로이드 어플리케이션을 통해 원격지에서 모니터링하는 시스템을 제안한다. 본 시스템은 SSD-MoblieNet 모델을 사용하여 학습시킨 데이터를 바탕으로 라즈베리파이4 카메라로 화재를 감지한 이후, 카메라 캡처를 하여 파이어 베이스 Storage와 Realtime-Database 서버에 전송하며, 어플리케이션에서 알림기능을 수행하며 감지장치에 대한 오작동 여부를 사용자가 판단할 수 있게 정보를 전달하는 기능을 수행한다.

ABSTRACT

In this paper, we propose a remote monitoring system through Android application in the event of a fire in the camera screen being monitored using Raspberry Pi4. Based on data learned using SSD-MoblieNet model, this system detects a fire with a Raspberry Pi4 system including a camera, captures the image of camera, and sends it to Fire Base Storage and Realtime-Database Server. Also it performs notification-functions in the application, and delivers the fire-detection information to the user for user's decision if the detection device is malfunctioning.

키워드

Fire, Fire Detection, Deep Learning, SSD-MobileNet, Android Studio, Java, Application

I. 서 론

대부분의 화재는 국소적인 불로부터 시작 되어 인적피해와 물적 피해 등 여러 피해를 입히고 있다. 국소적인 초기 화재를 신속하고 정확하게 감지하여 조치하는 방법은 인적피해와 물적 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 논문은 기존의 논문인 '딥러닝 기반 화재감지 시스템 구현'에서 비교한 모델 중 라즈베리파이4에 실행할 수 있는 SSD-MobileNet 모델 [3]을 통해서 화재를 감지하고 파이어 베이스 Storage와 Realtime-Database가 연결된 어플리케이션으로 확인하는 방법을 연구한다.

II. 관련 연구

2.1 Deep Learning

딥러닝은 여러 비선형 변환 기법을 이용해서 학습 데이터에 대한 높은 수준의 추상화를 통해 데

이터의 주요 특징들을 추출한 다음, 데이터 군집화 및 분류를 수행하는 기계학습의 일종으로 Deep Belief Network, Recurrent Neural Network, CNN 등을 활용하는 방법이 있다[4].

2.2 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 클래스 값에 default box를 활용해 최종 bounding box를 생성한다[3].

III. 시스템 설계 및 구성

표 1은 이전에 발표한 딥러닝 기반 화재 감지 시스템[5]을 개선해서 성능 비교한 결과이다. 기존 논문에서 제안한 방법에서 라즈베리파이에서 사용할 수 있는 모델은 SSD-MobileNet의 방법이다. 정확도가 낮은 편이지만 마이크로 컴퓨터인 라즈베리파이에서 쓸 수 있으며 기존 논문의 이미지보다 조금 더 라벨링을 개선하였다. 화재에 직접 상자를 Labeling한 이미지가 총 405개이며 324개의 train 이미지와 81개의 test 이미지(8:2)로 구성을 하였다. LabelImg 프로그램을 사용하여 이미지를 xml 파일로 저장 후 csv파일로 변환하여 저장하고 Tensorflow detection model zoo(github)에서 다운한 ssd_mobilenet_v2_quantized_300x300_coco모델을 통해서 훈련을 시작한다. 훈련이 끝난 후에 bazel을 통해서 detect.tflite 파일을 만들고 labelmap과 함께 라즈베리파이의 폴더에 넣고 실행을 한다. 웹캠에서 실시간으로 화재가 몇 퍼센트로 예측이 되었는지 출력창에 표시되며 50% 미만의 퍼센트의 박스는 그려지지 않는다.

표 1. 기존 논문에서 제안한 방법과 정확도, 학습 시간 비교[5]

Model	Accuracy(%)	Frame per second(FPS)
CNN-Adam opt	97%	null
CNN-Nadam opt	92%	null
Faster-RCNN	88%(avg)	0.2
SSD-MobileNet	87%(avg)	4.2

그림1은 안드로이드 어플리케이션 블록도이다. 어플리케이션은 Main Activity, Click Activity로 구성되어있다. Main Activity는 라즈베리파이에서 화재가 감지되면 파이어 베이스의 Storage에 캡처 사진을 저장하고 Realtime-Database에 데이터 값인 현재 시간이 업로드 된다. 새로운 데이터가 추가하게 되면 푸시알림이 울리게 되며 저장된 데이터는 현재 시간의 이름으로 값을 저장하여 중복된 이름을 피하고 ListView에 그 값을 표시하여 저장된 항목을 볼 수 있게 한다. 표시된 리스트를 누르면 두

번째인 Click Activity 화면으로 넘어가게 된다. Click Activity는 Main Activity의 값인 파일 이름을 가져와 String에 저장하면 Storage에 저장된 경로와 URL을 가져와 Glide로 ImageView에 이미지를 넣어 저장된 캡처 사진을 불러온다. 버튼은 2개로 구성되며 119에 신고를 할 수 있는 ACTION_CALL버튼과 오류로 인한 잘못된 촬영의 경우 이미지 삭제 후 초기 상태로 돌아가는 버튼이 있다.

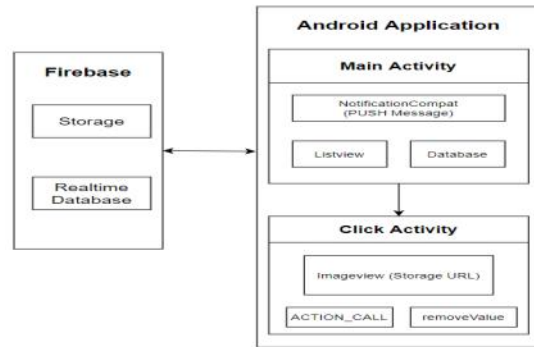


그림 1. 안드로이드 어플리케이션 블록도

그림 2은 화재 감지 시스템의 전체 구성도를 나타낸다. 라즈베리파이를 이용한 딥러닝 기반 화재 감지 시스템의 큰 흐름은 그림1과 같다. 라즈베리파이4로 구현한 카메라, 안드로이드 어플리케이션 그 두 개를 원격으로 연결하는 Data서버가 Firebase이다.



그림 2. 전체 구성도

IV. 구현 결과 및 분석

본 화재 감지 시스템의 화재 감지를 위한 학습 환경은 Tensorflow-Lite와 Tensorflow-gpu, python, OpenCV, Keras 딥러닝 프레임워크를 이용해서 제안된 방법을 구현하였다. 학습과 동작을 위한 하드웨어로 Ryzen7-3700x CPU, 16GB RAM Geforce RTX2060 super GPU 8GB VRAM Raspberry pi 4 4GB, Microsoft HD3000이 사용되었다. 어플리케이션 구현을 위한 소프트웨어 환경은 Android Studio 4.0.1의 Android 4.1(Jelly Bean) SDK 환경에서 구현하였다.

화재 감지를 위한 SSD-MobileNet 모델 학습 성능은 그림 3와 같다. SSD-MobileNet의 손실율은 20부터 시작해서 약 33000번 스텝을 한 결과 평

균 2이하까지의 손실율을 보였으며 화재 이미지 감지로는 특정한 불꽃의 이미지는 99%로 감지하는 반면에 특정 화재 이미지에서는 적색의 물체를 동반한 박스를 그리는 오류가 발생한다. 하지만 그 이외 대부분의 상황에서 높은 수준의 결과를 보여주었고 오류는 이미지 데이터를 늘리거나 정확한 라벨링, 학습 네트워크를 개선해야 한다.



그림 3. SSD-MobileNet 손실율과 화재 검출

카메라에서 감지된 화재 부분이 사진(그림 4)으로 캡처되어 파이어 베이스 Storage에 저장되고 동시에 데이터가 Realtime-Database에 전송이 되어 사용자는 실시간으로 캡처된 사진을 안드로이드 어플리케이션에서 확인할 수 있게 된다. 데이터를 전송하게 되면 어플리케이션에 푸시메시지가 수신되고 푸시메시지를 눌러 어플리케이션을 실행하면 그림 5에 나오는 메인화면을 확인할 수 있다.



그림 4. 라즈베리파이4에서 감지되는 화재사진

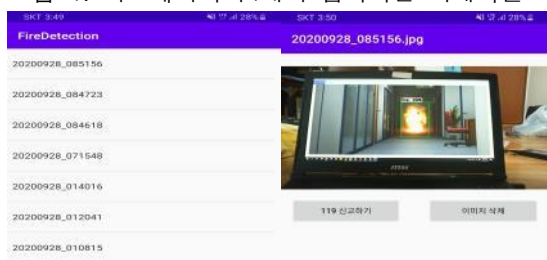


그림 5. 어플리케이션 (좌)메인, (우)Second화면

업로드 된 날짜와 시간이 리스트로 기록되면서 리스트를 터치하면 Second화면으로 넘어가게 되고 사용자가 확인한 후 직접 버튼을 이용해 119에 곧바로 전화가 연결되게 사용할 수 있고 오류로 인하여 잘못된 감지일 경우 이미지 삭제 버튼을 사용하여 이미지를 삭제하고 카메라를 초기상태로 재가동 시킨다.

V. 결 론

본 논문에서는 라즈베리파이를 기반으로 하여 설계한 감시 카메라에 딥러닝 기술을 융합하여 화재를 감지하였으며 어플리케이션을 통한 실시간 확인이 가능한 방법을 제안하였다.

Tensorflow-Lite를 이용한 SSD-MobileNet 모델의 정확도는 라즈베리파이4에서 화재를 잘 감지를 하였으나 몇몇 이미지에서 화재가 아닌 이미지를 화재로 인식하는 경우가 보였다. 이것은 차후에 405개의 이미지보다 데이터양을 늘리고 네트워크 개선 등을 적용시켜서 발전시킬 수 있을 것이다. 어플리케이션의 경우 화재가 감지가 되면 Firebase에서 정보를 가져와 캡처된 사진을 볼 수 있으며 119에 신고 할 수 있는 버튼과 혹시 이미지가 잘못 감지되어 캡처가 되었을 때를 방지하는 이미지 삭제 버튼이 있어서 감지된 부분을 실시간으로 확인을 하며 기계적인 오류를 최종적으로 사람이 판단해 최대한 줄였다. 따라서 본 연구는 정확성과 신속성을 높여 기존의 화재감지 장치들과 차별화되어 사용자에게 편리성과 정확성을 모두 제공할 수 있을 것으로 기대된다.

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] 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/>.
- [4] 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.
- [5] H. N. R. Kang, C. Y. Lim, Y. S. Yu, "Implementation of Fire Detection System Based on Deep Learning", in *Proceeding of Spring Conference of Korea Institute of Information and Communication Engineering*, Vol. 24, No. 1, pp. 338-340, 2020.