

빅데이터와 머신러닝 기반 Smoke 모니터링 예측시스템

Big Data and Machine Learning-based Smoke Monitoring Prediction System

정다비치*, 이원기**, 이주호***, 광경민****, 노영주*****

Davichi Jeong*, Won-Ki Lee**, Ju-Ho Lee***, KyungMin Gwak****, Young J. Rho*****

{davichiar, shkuei773, dlwngh3536, k010511, yrho}@kpu.ac.kr

요 약

여러 산업현장에서 생산적 목적을 위하여 가스를 사용하는 것은 불가피하다. 가스의 종류에는 안전한 가스부터 위험한 가스까지 여러 종류가 존재하는데 위험한 가스의 경우는 누출 사고 발생 시 커다란 인명피해와 재산피해를 불러온다. 이러한 사고를 막기 위하여 산업 현장에서는 감지 센서와 관리 프로그램을 사용하여 이러한 위험을 줄이고자 노력을 하지만 센서의 오작동이 존재하고, 누출 사고의 원인이 무엇인지 정확히 알기 어렵다. 이러한 문제들을 극복하기 위하여 생성된 데이터들을 머신러닝 기법을 통하여 가스 확산 시뮬레이터를 제작하여야 하는데 그러기 위한 신뢰할 수 있는 데이터들이 필요하다.

본 논문에서는 신뢰할 수 있는 데이터를 얻기 위하여 시중에서 판매하고 있는 가스 탐지기를 사용하여도 되지만 데이터를 수집하기 위한 방식 자체도 연구의 한가지 목적으로 만들어 가스 확산 시뮬레이터를 진행하기 전 단계인 Smoke 탐지기를 제작하고자 한다.

키워드 : Smoke, 라즈베리파이, Tensorflow, Classification

I. 서 론

가스 누출 사고 사례 중 하나로, 빙그레 공장에서 누출된 암모니아 가스는 압축식 냉동기의 냉매로 이용되는 무수 암모니아인데, 호흡기 질식을 일으킬 수 있으며 심할 경우 화상을 입을 수도 있다. 암모니아 가스가 인체에 미치는 영향은 그 농도(ppm)와 접촉 시간에 따라 달라지며, 적용할 수 있는 해독제는 별도로 없다.

위와 같이 고농도의 유해가스는 누출되었을 시에 인체에 치명적이기 때문에 누출의 범위와 위치를 정확하게 아는 것이 중요하다.[1] 또한 대구 염소가스 누출사고 사례가 있는데, 이 경우에도 유출된 가스의 성분 파악이 늦어지면서 대피 명령 등 후속조치를 제때 내리지 못했다.

소방관계자는 "원 탱크에서 바로 누출됐다면 어떤 물질인지를 알기 때문에 바로 대처가 가능한데 누출된 가스가 어떤 성분인지를 몰라 이를 확인하느라 조치가 늦어졌다"고 밝혔다.[2] 또한 누출된 가스의 성분을 확인하지 못하여 대응이 늦어질 수도 있다.

우리는 산업 현장에서나 일상에서 이용하고 있는 가스에 의한 사

고 및 확산 방지를 진행하려 하고, 이는 사회, 경제적으로 효과적인 산업 현장 사고 제어 가능할 뿐만 아니라 환경적으로 가스 제어가 가능하기 때문에, 위험 가스 차단 가능하다. 이는 라즈베리파이 카메라를 이용하여 Smoke 측정 가능하고, 관련 센서와 같이 이용한다면, 가스 발생 위치와 농도를 알 수 있을 것이다.[3]

II. 본 문

1. 관련 연구

관련 연구 목표는 총 5가지로 나타낼 수 있다.

- (1) 라즈베리파이 카메라, 센서를 이용하여 Smoke 감지
- (2) Tensorflow를 이용하여 Camera에서 Smoke 부분만 추출
- (3) 라즈베리파이 Smoke 센서, Image를 통해 학습 데이터 생성
- (4) 학습 데이터 생성 한 것을 토대로, Model 생성
- (5) Model 생성 된 것을 토대로, 카메라 촬영시 Smoke 농도 LOW/HIGH 표시

관련 기술 사례로, 가스 카메라에서 가장 유명한 FLIR 열화상 카메라 (CO2)의 주요특징으로 이산화탄소의 누설을 즉시 시각적으로 보여주고 누설 부위를 정확하게 추적하여 검출한다. 하지만 FLIR에서 제한적으로 카메라를 제공해주며, 현재 CO2나 다른 가스 탐지 카메라는 판매 중지된 제품이 대부분이다.

*한국산업기술대학교 컴퓨터공학과
 **한국산업기술대학교 컴퓨터공학과
 ***한국산업기술대학교 컴퓨터공학과
 ****한국산업기술대학교 스마트팩토리학과
 *****한국산업기술대학교 컴퓨터공학과 교수

1.1 기존 유사 프로그램과의 비교

표 1. 유사 프로그램과의 비교

	가스안전 지킴이 가스누설검지기 가스누출감지기	적외선식 불꽃감지기	FLIR 열화상 카메라	GDKPU [가스 탐지 분석기]
구분	가정용 탐지기	산업용 화재 탐지기	산업용 가스 탐지기	개발 예정 탐지기
탐지 방법	LP가스, 도시가스 등이 유출되어 공기 중의 일 정치 이상이 발생시에 알림	자외선과 적외선의 전자 기파를 측정하여 화재를 탐지	열화상 카메라를 이용 하여 가스의 농도를 나 타냄	카메라를 이용하여 가 스 농도 정도 나타냄 - 가격적 우위
목표	가정의 가스밸브 유출 탐지	산업 현장에서 불이 나기 쉬운 곳에서 더 큰 화재 를 미연에 방지하고자 함	가스 유출의 경우 이미 지를 통해 보고 위험한 상황인지 몰라 대응이 늦어지는 것을 방지	가스 유출의 경우 이미 지를 통해 보고 위험한 상황인지 몰라 대응이 늦어지는 것을 방지

1.2 개발 이슈

첫 번째 이슈는 현 모듈형 열화상카메라서는 제한으로 CO (일산화탄소), LPG, Smoke와 같은 가스 종류 탐지가 가능하다는 것이다. 이에 개발 이슈를 적용하여 CO2 탐지나 다른 가스 탐지보다, 감지 가능한 CO, LPG, Smoke 관련 가스 탐지에 집중하도록 이 문제를 해결하고자 했다.

두 번째 개발 이슈는 가스의 농도 측정 방식이다. 가스를 영상으로 보게 되면 진한 부분과 연한 부분이 있는 것을 확인 할 수 있는데 이것을 이용하여 가스의 농도를 머신러닝, 클러스터링 기법을 통하여 학습시켜 사용하고자 한다. 농도 구분 방법은 센서를 통해 이미지를 학습시키고, 이를 통해서 가스의 농도를 센서 없이도 얻게 만드려 한다.

마지막 개발 이슈로 비주얼라이징이 있다. 만일 가스를 탐지하고, 일정 수준 이상의 ppm (농도)이 나올 경우 이를 울리는 기능 말고, 비주얼라이징, 즉 비주얼 적으로 보여줄 부분에 대해서 상의하게 되었는데, 이는 Android 같은 외부적 기기에서 실시간으로 농도를 탐지할 수 있도록 수정하도록 한다.

2. 세부 설계 및 구현

2.1 개발 환경



그림 1. 개발 환경 구성도

첫 실험 환경은 연구실에서 진행하였으며, Smoke(보이는 연기)를 측정해야 하기 때문에, 전자담배 같은 연기를 통해서 라즈베리파이로 찍을 수 있는 환경을 조성했다.

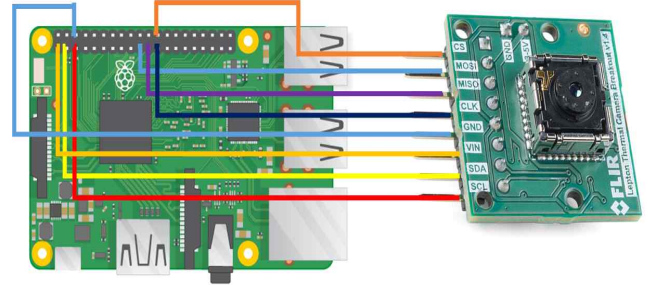


그림 2. 라즈베리파이, 카메라 회로도

데이터를 수집하기 위한 장치로 라즈베리파이3 B를 사용하였으며, <그림 2> 과 같이 회로도를 구성했다. 이후 PICAra[4], FLIR[5] 카메라를 이용해 연기를 측정하였다. 이를 통해서 사진 데이터를 수집하였다.

2.1.1. HW 설계, 구축

SoC	BMC2837
CPU	Qu0ad Cortex A53 @ 1.2GHz
Instruction set	ARMv8-A
GPU	400MHz VideoCore IV
RAM	1GB SDRAM
Ethernet	Micro-SD
Wireless	802.11n/Bluetooth 4.0
Video Output	HDMI/Composite
Audio Output	HDMI/Headphone
GPIO	40

그림 3. 라즈베리파이3 B 사양

정기적으로 Smoke Image를 찍어 서버로 전송하는 데이터로 만들기 위하여 사용한 라즈베리파이3 B는 <그림 3> 과 같은 사양을 이용했다. 컴퓨터의 USB 장치를 이용하여 공급받는다. Smoke Image를 서버로 보내는 소프트웨어로는 Python을 사용했다. 라즈베리파이3 B와 호환되는 카메라인 PICAra는 Python 소프트웨어로 움직이며, 이를 통해서 Camera를 640x480 해상도로 라즈베리파이에서 실시간으로 촬영하는 영상을 볼 수 있도록 한다.

2.1.2. 유스 케이스



그림 4. 유스 케이스 (전체 FLOW)

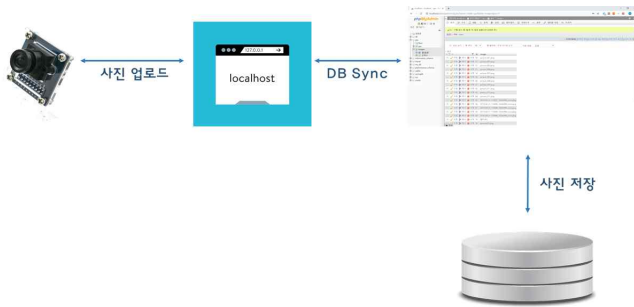


그림 5. 유스 케이스 (HW FLOW)

2.1.3. 시스템 구성도

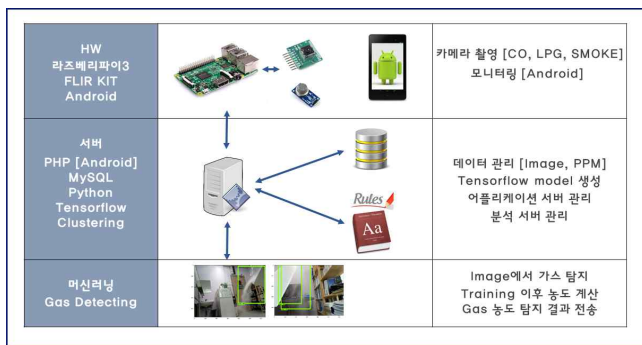


그림 6. 시스템 구성도

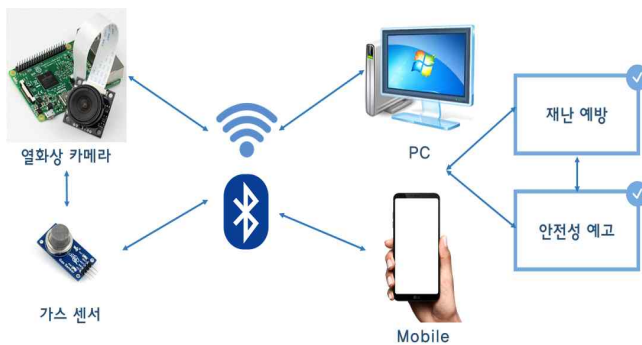


그림 7. 시스템 구성도 (간략)

2.1.4. Gas Detection Flow

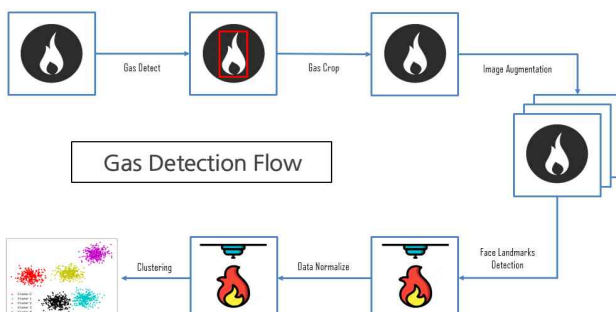


그림 8. Gas Detection Flow

III. 모의 실험

3.1 실험 방법

실험 방법으로는 총 3가지로 나누어 진행하였다.

첫 번째로는 라즈베리파이이며, FLIR KIT을 이용한 카메라 구현, 라즈베리파이3으로 개발했으며, MySQL과 연동하여 PICamera 라이브러리 사용, 전송하는 방식을 사용했으며 Window Local Server로 진행되었다.

두 번째로는 머신러닝인데 이는 Image를 받아서, Smoke 부분만 Tensorflow[6], Faster-RCNN[7] 모델 이용해서 추출했으며 Image 데이터와 농도 관련 데이터를 받아서 학습하고, LOW/HIGH 모델로 구현했다. 이는 MacBook Local Server로 진행했다.

세 번째로는 Server와 DB인데, 이는 Local 서버 구축 및 MySQL을 이용한 DB를 구축했으며, 일정 PPM 농도이상일 시에 경고음을 보내는 식으로 구성하였다.

마지막 Android에서 시각화한다.

3.1.1. 실험 환경 설계

- (1) CO, LPG는 위험하기 때문에 시연은 Smoke로 진행
- (2) MacBook Server (머신러닝), Window Server (MySQL)
- (3) Android (안드로이드)를 이용해서 비주얼라이징
- (4) 라즈베리파이를 이용해서 카메라 촬영, 서버 전송 확인
- (5) Python 서버에서 Smoke 탐지 확인
- (6) 농도 LOW/HIGH가 제대로 나오는지 확인
- (7) 일정 수준 넘을 경우 경보음 울리거나 확인, Android로 비주얼라이징

4. 실험 결과 및 분석

우선 Pylepton을 이용하여 이미지를 캡처한 뒤, MySQL과 서버와 연동해 Image를 업로드 한다. DB에 저장된 내용은 PHPMyAdmin을 통해서 확인 가능하다. 이를 통해서 라즈베리파이 에 있는 사진을 DB로 전송할 수 있다.

라즈베리파이 카메라에서 사진을 찍어서 DB에 올리게 되면 (Local), 사진을 받아 Smoke 부분만 Detecting 한다. Detecting 한 후, Keras[8] 기법을 사용해서, LOW, MID, HIGH IMAGE로 구분하는 작업까지 이루어진다.

Smoke Detection, Classification 관련 결과가 출력되면, 그 내용을 Android로 전송해야 하기 때문에, 관련 내용을 다시 DB에 전송한다. Smoke Detection 부분에는 Object Detection 기법이 들어가며, 이 부분에 대한 모델은 Faster R-CNN을 사용한다. 총 이미지는 500장 정도이며, 이 이미지를 Smoke 라벨링하여 감지한다.[9]

4.1 Smoke Detection 구현 결과

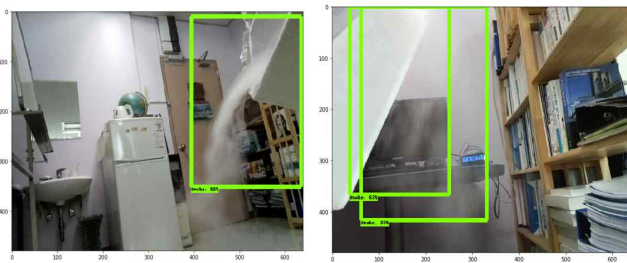
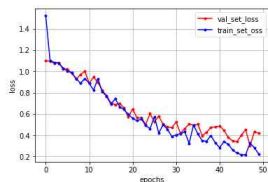


그림 9. Smoke Detection 실제 구현 결과

<그림 9> 는 실제 Smoke Detection 한 결과이며, 이처럼 Smoke 정확도가 몇 퍼센트인지, 그에 따라서 박스로 잡히는 영역이 달라진다. 이를 통해서 이 Smoke의 농도가 어느 정도 되는지 판단한다.

4.2. Smoke Classification 구현 결과



```
[1,000 0,000 0,000]
0
[0,000 1,000 0,000]
1
[0,000 1,000 0,000]
1
[0,000 0,000 1,000]
2
[0,000 0,000 1,000]
2
2019-06-08,19H2M44.jpg low
2019-06-08,19H2M35.jpg mid
2019-06-08,19H5M26.jpg mid
c:\prsrapp\c0873n\we9413.jpg high
c:\pru43\ndv9373oe453221.jpg high
```

그림 10. Smoke Classification 실제 구현 결과

<그림 9> 는 실제 Smoke Detection 한 결과이며, 이처럼 Smoke 정확도가 몇 퍼센트인 Smoke Classification 구현은 Keras 이미지 모델 분류를 사용하였으며, Smoke Detection에서 나온 이미지 Crop 된 사진을 LOW, MID, HIGH로 가스 분류를 해서 데이터 셋을 만든다. 단 이때 면적에 영향을 받지 않고 농도의 높낮이로 구분한다.

위 그래프는 훈련을 할수록 얼마나 loss 값이 줄어드는지를 확인할 수 있는 그래프이다 <그림 10> 처럼 훈련을 진행했을 때 최종 정확도는 약 85%이며, 실제로 이미지를 테스트하게 되면 mid, low, high 순서대로 데이터가 나오는걸 볼 수 있다. 지, 그에 따라서 박스로 잡히는 영역이 달라진다. 이를 통해서 이 Smoke의 농도가 어느 정도 되는지 판단한다.

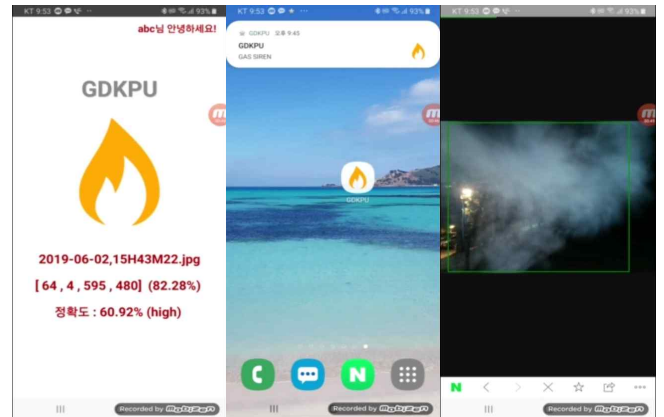


그림 11. Android 비주얼라이징

라즈베리파이에서 받은 사진 데이터, Smoke Detecting 된 데이터와 Smoke Classification 된 데이터를 모아서 Android에서 알림기능을 실행시킨다.

기본적으로 APP을 실행시키게 되면 알림 기능이 작동하며, 이는 뒤로 가기 2번으로 종료 시킬시 서비스는 종료된다. (홈 버튼 누를시 서비스 지속)

GDKPU 제목 밑의 사진을 누르게 되면 IMAGE를 볼 수 있으며, 만일 연기가 잡혔다면 CROP 된 부분도 보여준다. (GDKPU 제목을 누르게 되면 내용을 생략한 채로 제목만 보이게 할 수 있으며, 제목을 다시 누르게 되면 다시 나타난다.) 첫 번째 내용은 image 이름, 두 번째 내용은 CROP 된 사각형 위치, 사각형이 차지하는 이미지의 전체 비율이다. 세 번째 내용은 CROP 된 사각형의 정확도이며, 마지막은 이 Smoke의 위험도를 나타낸다. (LOW, MID, HIGH)

만일 위험도가 MID나 HIGH가 나온다면 APP을 나온 상태에서 서도 경고 PUSH 메시지가 나온다.

IV. 결 론

5.1. 기대 효과

표 2. 기대 효과

사용자 측면	<ul style="list-style-type: none"> - 사용자 측면에서, 가스가 누출된 것인지 농도로 판별 가능하며 - 가스 농도 측정의 방식의 변화를 확인 가능하여 안전하게 가스관련 서비스를 이용할 수 있습니다. [신뢰도 증가]
비즈니스 측면	<ul style="list-style-type: none"> - 비즈니스 측면에서, 개인 뿐 아니라 기업이나 업체에서 가스 누출 관련 좀더 쉬운 방법으로 판별 가능하기 때문에, 이 프로그램이 많이 활성화가 되면 가스 관련 산업에 변화를 일으킬 수 있다고 생각합니다.
개발자 측면	<ul style="list-style-type: none"> - [가스 누출 Program] 은 최근 가장 활발하게 연구가 이루어지고 있는 머신러닝, AI에 관한 프로젝트입니다. - 머신러닝 중에서도 데이터 가공, 관련 공부를 하다보면 머신러닝에 관한 기초가 쌓여 자후 이 프로젝트가 아니더라도, 빅데이터 관련 분야의 프로젝트를 진행할 때도 많은 도움이 될 것입니다.

5.2. 향후 연구 과제

가스 오염을 구분해 내는 것이 최종 목표이지만 가스 오염을 구분해내기 위한 가스 확산에 대한 데이터가 존재하지 않는다. 그를 위하여 가스 확산 데이터를 수집하는 것은 최우선적으로 하게 되었다. 그를 위하여 먼저 가스의 데이터를 수집할 수 있는 기기를 제작하여야 하는데, 데이터를 수집하는 방식에서 대해서 살펴보면 화학식 반응에 의하여 발생하는 에너지를 전기적인 에너지로 변환시키는 원리에 의하여 가스를 감지하는 전기화학식 센서가 존재하고, 가연성 가스와 산소와 반응하여 생기는 반응열을 전기신호로 변환해서 감지하는 방식인 접촉연소식 가스센서, 적외선을 발사하고 다른 한쪽에서는 적외선을 감지하여 적외선 사이에 유독한 물질이 들어와 적외선을 흡수하게 된다면 감지기에서 알아차려 컴퓨터로 전송하고 컴퓨터에서 물질의 종류를 분석하여 어떠한 물질인지 알아차리는 방식이 있다.

우리가 이번 연구에서 만들고자 하는 방식은 가스(Smoke)의 농도를 파악할 수 있는 것을 만들고자 했다. Smoke의 농도를 알아보기 위하여 카메라에 잡히는 Smoke의 이미지들과 다른 기기들로 Smoke 농도를 측정한 결과를 모아 머신러닝 기법을 사용하여 학습을 했고, 향후에 더 나아가 카메라로 파장을 이용하여 가스를 탐지하고, 종류까지도 분석할 예정이다.

icle&id=62&Itemid on Oct. 17, 2019.

※ 본 연구는 중소기업부의 중소기업혁신사업의 부분적 지원으로 진행되었다.

참 고 문 헌

- [1] “빙그레 공장 폭발사고...암모니아 가스 노출 시 인체 영향은?” 라포르시안. Retrieved at <http://www.raportian.com/news/articleView.html?idxno=1612> on Feb.13, 2014.
- [2] “대구 염소가스누출사고... 누출대처 ‘화 키워’” 노컷뉴스. Retrieved at <https://www.nocutnews.co.kr/news/4339386> on Dec.11, 2014.
- [3] 광경민, 노영주. (2018). 가스 확산 모사를 위한 센서 데이터 수집 연구. 한국정보과학회 학술발표논문집, (), 449-451.
- [4] PI Camera, Retrieved at <https://picamera.readthedocs.io/en/release-1.13/> on Oct. 17, 2019.
- [5] FLIR Korea. Retrieved at <https://www.flirkorea.com/> on Oct. 17, 2019.
- [6] TensorFlow, Wikipedia. Retrieved at <https://en.wikipedia.org/wiki/TensorFlow> on Oct. 17, 2019.
- [7] Chungkeun Lee, H. Jin Kim, Kyeong Won Oh. (2016). Comparison of faster R-CNN models for object detection. 제어로봇시스템학회 국제학술대회 논문집, (), 107-110.
- [8] Keras. Retrieved at <https://www.tensorflow.org/guide/keras?hl=ko> on Oct. 17, 2019.
- [9] Image Dataset, Retrieved at http://wildfire.fesb.hr/index.php?option=com_content&view=art