



Received: 2024/11/05
Revised: 2024/11/11
Accepted: 2024/11/26
Published: 2024/12/31

***Corresponding Author:**

Soon-Kook Hong
Dept. of Mechanical System Engineering, Republic of Korea Naval Academy
1 Jungwon-ro, Jinhae-gu, Changwon-si, Gyung-sangnam-do, 51704, Republic of Korea
Tel: +82-55-907-5314
E-mail: hks753@gmail.com

Abstract

본 논문에서는 이전 논문(파이썬과 MOY Graph를 이용한 항적 분석방법)에서 설명된 사진 이미지로부터 Canny 에지 이미지를 추출하는 방식을 이용하여 필요한 경계 이미지를 추출하는 방식을 다시 설명한다. 이렇게 얻어진 Canny 에지 이미지를 MNIST 데이터셋을 통해 학습된 Adam 옵티마이저를 기준으로 하는 CNN 기반 머신러닝을 이용하여 숫자 이미지 분포를 확인하는 방법을 제시한다. 또한 이 결과를 통하여 구름 이미지와 미사일 피격으로 인한 이미지를 구별할 수 있음을 설명하고 있다.

This paper remarks a method for extracting the required edge images using the Canny function applied to photo images, as described in a previous work(On an Analysis Method for Ship Wakes with Python and MOY Graphs). Following this, we propose an approach to identify the distribution of digit images using a CNN-based machine learning model trained on the MNIST dataset with the Adam optimizer. Furthermore, we demonstrate that this program is able to distinguish between cloud images and images resulting from missile strikes.

Keywords

미사일 피격 이미지(Missile Strike Image), Canny 함수(Canny Function), MNIST 데이터셋(MNIST Dataset), Adam 옵티마이저(Adam Optimizer), CNN(Convolutional Neural Network)

Acknowledgement

이 논문은 2023년 정부(방위사업청)의 재원으로 국방기술진흥연구소의 지원을 받아 수행된 연구임 (KRIT-CT-23-030)

파이썬을 이용한 미사일 피격 이미지 추출 및 CNN 기반 머신러닝을 이용한 명중정보 분석 연구

The Research on Analyzing Missile Strike Images via Python and Hit Information by Using a CNN-based Machine Learning

김세라¹, 최영두², 장원석³, 한민석⁴, 전민규⁵, 이현기⁵, 홍순국^{6*}

- ¹해군사관학교 기초과학과 부교수
- ²해군 소령/해군사관학교 전자제어공학과 조교수
- ³한화시스템 해양시스템1팀 수석연구원
- ⁴해군사관학교 전자제어공학과 부교수
- ⁵해군사관학교 기계시스템공학과 조교수
- ⁶해군 대령/해군사관학교 기계시스템공학과 교수

Sera Kim¹, Youngdo Choi², Wonseok Jang³, Min-Seok Han⁴, Min-Gyu Jeon⁵, Hun-Kee Lee⁵, Soon-Kook Hong^{6*}

- ¹Associate professor, Dept. of Natural Science, Republic of Korea Naval Academy
- ²LCDR, ROK Navy/Assistant professor, Dept. of Electronics and Control Engineering, Republic of Korea Naval Academy
- ³Chief engineer, Naval System Team 1, Hanhwa Systems
- ⁴Associate professor, Dept. of Electronics and Control Engineering, Republic of Korea Naval Academy
- ⁵Assistant Professor, Dept. of Mechanical System Engineering, Republic of Korea Naval Academy
- ⁶CAPT, ROK Navy/Professor, Dept. of Mechanical System Engineering, Republic of Korea Naval Academy

1. 서론

현대 전쟁에서 미사일 피격 이미지 확인의 중요성은 그 어느 때보다 커졌다. 전쟁이 진화함에 따라 미사일 공격의 정밀도와 파괴력이 증가하고 있으며, 이러한 공격의 효과를 정확하게 평가하는 것이 방어뿐만 아니라 전략적 계획 수립에도 필수가 되었다.

근래의 러시아-우크라이나 전쟁에서 볼 수 있듯이, 러시아의 탄도 미사일이나 우크라이나의 대응 방어 시스템을 포함해 양측이

사용하는 미사일 시스템의 파괴력은 실시간으로 미사일 피격을 식별하고 분석하는 능력의 중요함을 보여주었다. 이런 시기에 미사일 피격 이미지 확인은 다음과 같이 여러 면에서 핵심적인 역할을 할 수 있다.

첫 번째로 미사일에 의한 피해 평가를 확인하기 위하여 필요할 수 있다. 미사일의 명중률을 분석함으로써 군사 부대는 군사 시설, 인프라, 또는 민간 지역에 가해진 피해를 정확하게 평가하고, 이 결과를 통해 현재 전략을 조정하고 자원을 배치하며 방어 시스템을 강화할 수 있다. 또한 실시간으로 미사일 피격을 분석함으로써 목표가 완전히 무력화되었는지, 추가 조치가 필요한지를 판단할 수 있다. 특히 이 평가 작업은 민간인 보호를 위하여 필수적이다. 전쟁 지역에서 민간인의 안전을 보장하는 것은 가장 중요한 문제 중 하나이기 때문이다. 미사일 피격 분석은 군사 및 인도주의 단체가 민간 지역이 타격을 받았는지 확인하는 데 도움을 주며, 향후 민간 보호 구역에 대한 공격을 막을 수 있다.

대표적인 예로 Fig. 1에서 볼 수 있는 러시아-우크라이나 전쟁 중 폴란드 동부 우크라이나 국경지대 마을인 프세보도프에 떨어진 미사일 피격 사건을 들 수 있다. 이 피격으로 인해 폴란드 민간인 2명이 사망하였으며, 다국적 후속조사를 통해 우크라이나발(發) 미사일로 판정되었다[1].



Fig. 1. The missile strike in Poland[1]

두 번째로 피격 이미지의 활용은 미사일 방어 기술 발전에 도움이 된다. 이미지 확인은 미사일의 궤적, 속도, 명중 지역에 대한 이해를 돕고, 이를 바탕으로 더 효과적인 요격 시스템을 개발할 수 있다. 이스라엘의 아이언 돔(Iron Dome)이나 미국의 첨단 미사일 방어 시스템 등은 정확한 미사일 피격 데이터를 통해 기술을 개선하고 있다. 따라서 현대 전쟁에서 미사

일 피격 이미지 분석의 중요도는 점점 커져갈 것이며, 다양한 분쟁을 해결하는 데 핵심적인 도구가 될 수 있을 것이다.

저자의 이전 연구[2]는 파이썬을 이용하여 영상 이미지에서 필요한 정보만을 추출하여 매듭 이미지로 변환하는 기술을 소개하였으며, 본 논문에서는 이 기술을 적용하여 미사일 피격 시 발생하는 연기 이미지에서 필요한 정보만을 남기고 단순화하여 일반적인 구름 이미지와 구별하는 방법을 살펴보고자 한다. 구체적으로는 생성된 단순화된 이미지에 MNIST 데이터셋을 이용한 CNN 기반 머신러닝을 적용하여 해당 이미지가 미사일의 피격으로 생성된 이미지 여부를 빠르게 판정 가능한 방법을 소개하고자 한다. 이 방법은 고화질 이미지에 적용 가능한 머신러닝 기술의 발전을 통해 미사일의 명중정보를 파악하는 수준 역시 더욱 향상될 것이라 기대된다.

2. 해당 연구를 위한 배경 정보

2.1 영상 이미지의 필요 이미지 추출법

[2]는 선박이 움직인 수면 위 흔적 이미지에서 매듭 이미지를 추출하기 위하여 파동의 형태를 분석하는 방법을 제안하였다. 바로 항적에서 보이는 다양한 파형 이미지를 파이썬으로 특정 부분 이미지를 강조하고 필요 없는 부분을 생략하여 기하학적 매듭 그래프(MOY graph)로 변환하는 방법으로, 이 이미지 변화를 통하여 이미지 정보를 수학적 대상으로 보고 항적 정보를 담은 이미지들을 통하여 특정 수학적 군을 구성함을 참고문헌 [3]에서 확인할 수 있었다. 이 방법을 이용하여 이 논문에서는 미사일 피격 시 등장하는 연기 이미지에서 필요한 부분을 추출한 후 일반 구름 이미지와의 차이를 확인하고자 하였다.

2.2 이미지 판독 머신러닝 데이터셋 - MNIST

MNIST(Mixed National Institute of Standards and Technology)는 손글씨(hand writing) 숫자 이미지를 인식하기 위한 딥러닝 및 머신러닝 알고리즘을 평가하는 데이터셋이다. MNIST는 28×28 픽셀 사이즈의 손글씨 숫자 흑백 이미지 70,000장으로 구성되어 있다. 이 중 60,000장이 학습용(training

set), 10,000장이 테스트용(test set)으로 구성된 이 데이터셋은 숫자 이미지 인식 분야에서 매우 중요한 기준으로 여겨진다.

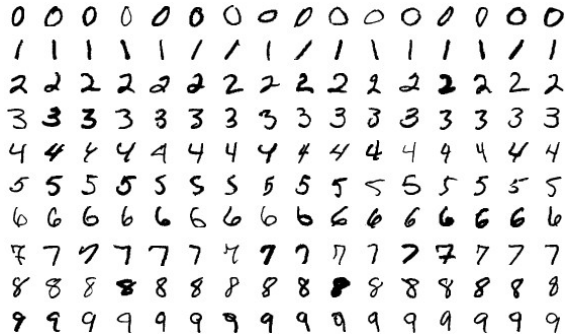


Fig. 2. A subset of the MNIST dataset[4]

이 데이터셋의 장점은 간단하고 접근성이 높다는 것이다. MNIST는 공개 데이터셋으로 빠른 학습과 테스트가 가능하다. 또한 작은 모델로도 충분히 실험이 가능하여 데이터와 시간 소모가 적다. 게다가 MNIST는 오랫동안 연구된 데이터셋이기 때문에 학계와 산업에서 표준으로 사용되어 결과를 비교하기 좋아 해당 연구에서도 사용하게 되었다.

3. Canny 에지 이미지 추출 및 CNN기반 머신러닝을 통한 미사일 피격 이미지 판정

[2]에서의 이미지 추출은 파이썬의 Canny 함수를 이용하였다. 이를 이 논문에서는 ‘Canny 에지(edge) 이미지’라고 부르도록 하겠다. 사용한 주요 코드들은 다음과 같다.

Canny 에지 검출을 통해 그레이스케일 이미지의 경계를 추출하였다. sigma 값을 2.0으로 설정하여 적절한 수준의 가우시안 블러를 적용함으로써 노이즈를 제거하고, 라이브러리 matplotlib의 subplots와 imshow 함수를 사용하여 원본 이미지와 Canny 에지 검출 결과를 시각화하여 이미지의 경계 정보를 강조하였다. 그레이스케일 이미지를 NumPy 배열로 변환하기 위해 np.array 함수를 사용하여 각 픽셀의 밝기 값을 포함하는 2D 배열을 생성하여 데이터 분석의 효율성을 높이고, 이미지 처리 작업의 정밀성을 확보할 수 있었다.

여기에 앞서 언급한 MNIST 데이터셋을 이용하여 미사일 피격 이미지에서 숫자 이미지 정보를 추출하

기 위하여 CNN 기반 머신러닝을 정의하였다. 이는 TensorFlow와 Keras를 사용하여 MNIST 데이터셋에 대해 CNN 모델을 구현하기 위한 코드로 구성하였다. 이 연구에서 제작된 코딩이 하는 작업을 구체적으로 설명하면 다음과 같다.

첫 단계로 분석하고자 하는 Canny 에지 이미지를 MNIST 데이터셋에 적합한 28×28픽셀 크기의 조각으로 분할하도록 한다.

두 번째로 각 28×28픽셀 사이즈 이미지 내의 각 픽셀 값을 0과 1 사이로 정규화하여 표현함으로써 다음 단계에서 CNN 모델이 이를 더 쉽게 학습할 수 있도록 한다.

세 번째 단계로 사용할 CNN 모델을 특정하였다. 여러 개의 Conv2D와 MaxPooling2D 레이어로 구성하여 CNN 모델이 두 번째 단계에서 생성된 각 이미지를 추출하여 숫자 이미지로 인지하게 하였다. 바로 Adam 옵티마이저와 sparse categorical cross-entropy 손실 함수를 사용하여 MNIST 데이터셋을 학습하였다. 이 모델은 5 에포크 동안 훈련되었다.

각 이미지 조각에 대해 앞서 정의한 CNN 기반 머신러닝을 이용하여 인식된 숫자 이미지의 개수의 분포를 display-distribution 함수를 이용하여 막대그래프로 만들었다.

각 숫자 이미지 학습 결과 3, 4, 6, 8, 9 숫자의 분포는 미사일 피격에서 나오는 연기 이미지와 대조군인 구름 이미지 모두에서 비슷한 분포가 관측되어 이 논문에서는 관측 대상에서 제거하였다. 해당 프로그램에서 폭발 이미지에서 나오는 밝은 화염점은 숫자 0으로 인지되고 파편 궤적은 숫자 7로 인지되어 미사일 피격 이미지를 대표하는 첫 번째 숫자 그룹(그룹 1)은 숫자 0과 7을 포함하는 집합으로 정의하였다. 반면 구름의 Canny edge 조각들에서 잘 발견되는 이미지인 2와 5를 확인하기 위하여 두 번째 숫자 그룹(그룹 2)는 숫자 2와 5로 구성하였다. 즉, Canny 에지 이미지를 분석하여 나오는 숫자 이미지들을 크게 그룹 1과 그룹 2로 분류하여 각 그룹의 원소 분포를 확인함으로써 원본 이미지가 미사일 피격에 의한 결과인지 아닌지를 확인한다.

예를 들어 앞서 본 폴란드 미사일 피격 사진 이미지인 Fig. 1에서 Canny 에지 이미지를 추출하면 Fig. 3와 같은 결과를 얻을 수 있다. Fig. 3를 28×28픽셀 사이즈로 분할하여 Fig. 4와 같이 여러 조각을 낸 뒤 해

당 조각들에 포함된 Canny 에지 이미지가 앞서 구현된 CNN 기반 머신러닝에서 각 조각들이 어떤 숫자 이미지로 해석되는지 분석하였다.



Fig. 3. Canny edge extraction from Fig. 1

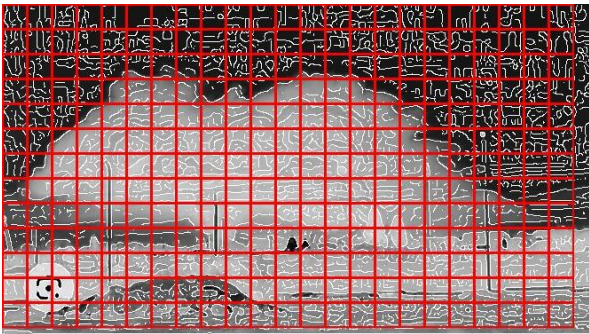


Fig. 4. Image segmentation in Fig. 3

이 숫자들의 분포를 그룹 1과 그룹 2로 분류하고 각 그룹에 속한 이미지 수를 바 형태 그래프로 구현하면 Fig. 5와 같은 그래프를 얻을 수 있다. Fig. 5를 통해 주어진 Fig. 1의 이미지는 그룹 1에 속하는 숫자 이미지가 그룹 2에 비해 상대적으로 더 많이 있다는 사실에 근거하여 미사일 피격 이미지로 판단을 내리도록 할 수 있다.

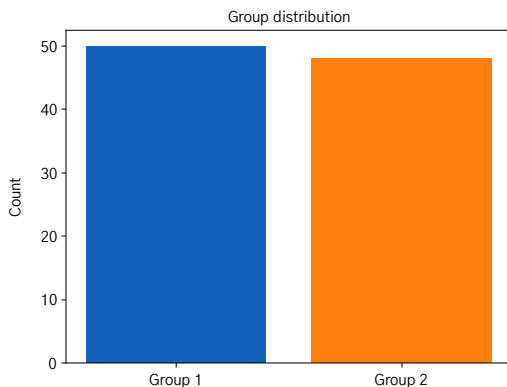


Fig. 5. The distribution of digit images in Fig. 4

다음으로 이 외의 다른 피격 이미지들과 구름 이미지를 바탕으로 해당 코딩에서 표현하는 그룹 1과 그룹 2의 분포 정도가 잘 표현되는지 살펴보았다. 올해 이란의 이스라엘 미사일 공격 시 촬영된 사진인 Fig. 6를 바탕으로 해당 프로그램을 실행하면 Fig. 7과 같은 분포 그래프를 얻을 수 있다.



Fig. 6. The missile strike in Israel[5]

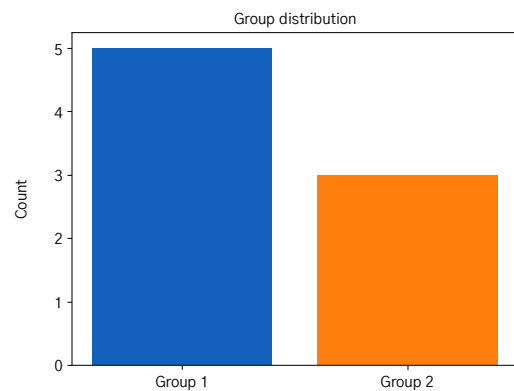


Fig. 7. The distribution of digit images in Fig. 5

앞서 기대한 것처럼 그룹 1에 속하는 숫자의 이미지가 그룹 2에 속하는 숫자의 이미지에 비해 많음을 알 수 있다. 이렇듯 피격 이미지에 대하여 그룹 1로 인지되는 이미지의 수가 그룹 2에 비해 크게 나올 경우 미사일 피격으로 판단하도록 한다면, 이를 바탕으로 무인시스템 등에서 피격확인 프로그램에 적용이 가능할 것으로 보인다.

반대로 이 결과와 비교하여 구름의 이미지를 대상으로 해당 프로그램을 실행한 결과는 다음과 같다. 일반적으로 연기 이미지와 유사한 모양을 가지는 구름은 적운과 적란운이다. 그 예시로서 Fig. 8과 같은 구름(적운)의 이미지에 대한 Canny 에지 이미지는 Fig. 9과 같으며, 이 CNN 기반 머신러닝에 적용시켰을 때의 결과는 Fig. 10의 분포 그래프와 같다.



Fig. 8. A cumulus 1 [6]

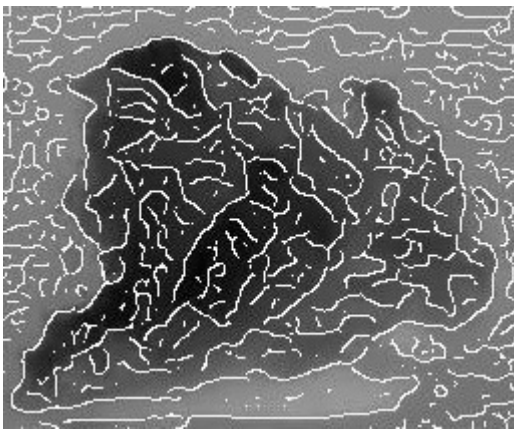


Fig. 9. Canny edge extraction from Fig. 8

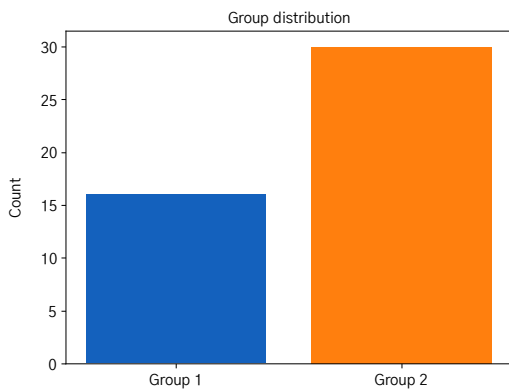


Fig. 10. The distribution of digit images in Fig. 9

즉, 예상한 대로 구름의 이미지에 대해서는 2와 5로 인지되는 이미지 조각이 더 많은 결과를 얻게 됨을 해당 그래프를 통해서 알 수 있다. 따라서 주어진 CNN을 통해 일반적인 구름과 미사일 피격으로 나오는 연기 이미지 분류가 가능함을 확인할 수 있다.

추가로, 흔히 볼 수 있는 또다른 구름 이미지와 이에 대한 분포 그래프 결과는 Figs. 11-14와 같다.



Fig. 11. A cumulus 2 [7]

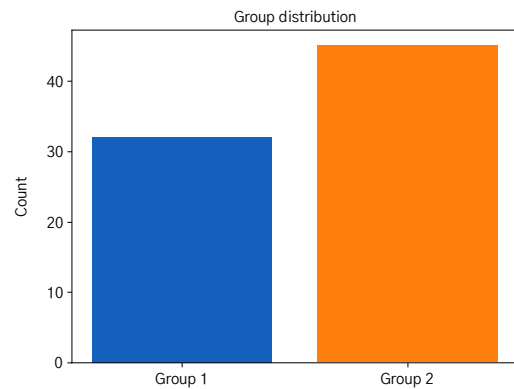


Fig. 12. The distribution of digit images in Fig. 11



Fig. 13. A cumulonimbus cloud [6]

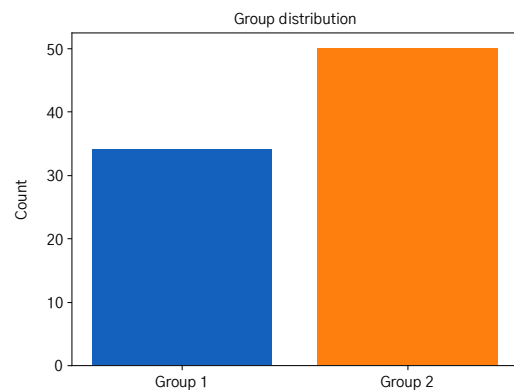


Fig. 14. The distribution of digit images in Fig. 13

이 미사일 피격 판정 프로그램은 북한군 미사일 또는 어뢰 피격 이미지를 판단하는 데에도 적용 가능하다. Fig. 15의 경우 2024년 9월 19일 공개된 SRBM인 화성포-11다-4.5의 시험발사 결과에 해당하는 이미지이다. 해당 이미지에 대하여 프로그램을 적용하여 확인한 결과는 Fig. 16과 같다.



Fig. 15. The missile drill strike in North Korea[8]

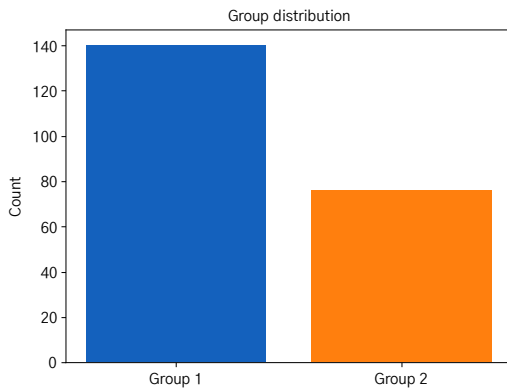


Fig. 16. The distribution of digit images in Fig. 15

또한 각각 어뢰 피격 이미지와 천안함 어뢰 피격 이미지인 Fig. 17과 Fig. 19에 대해서도 유사하게 그룹 1에 속하는 숫자인 0과 7의 이미지가 그룹 2에 비하여 상대적으로 많이 등장함을 확인할 수 있다.



Fig. 17. The torpedo strike[9]

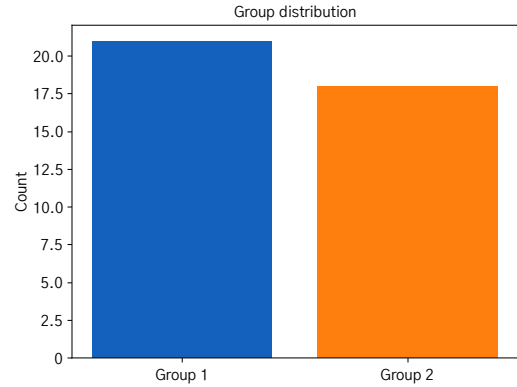


Fig. 18. The distribution of digit images in Fig. 17



Fig. 19. Attack on ROKS Cheonan[10]

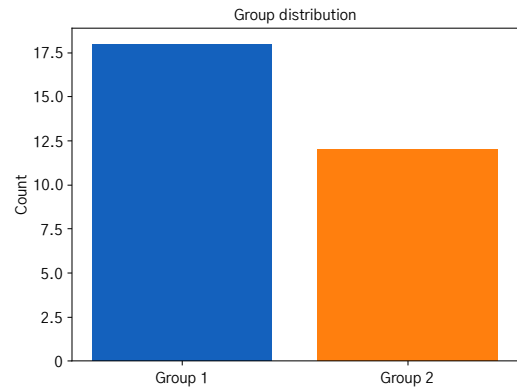


Fig. 20. The distribution of digit images in Fig. 19

4. 결론

미사일 피격으로 발생하는 연기 이미지와 일반적인 구름 이미지를 구별하기 위하여 본 연구는 파이썬을 통하여 사진에서 Canny 에지 이미지를 추출한 후 이를 숫자 이미지인 MNIST 데이터셋을 학습한 CNN 기반 머신러닝을 통하여 여러 개의 숫자 이미지들로 인식하는 방법을 제안하였다. 또한 이 방식에서 구별 기준이 될 수 있는 숫자 이미지(0, 7 / 2, 5)가 무엇인지 이 논문에서 정의하여 분류 가능성을 보

였다.

더불어 해당 프로그램을 통해 구별 가능한 여러 개의 이미지를 제시하였다. 다만 이 논문의 예시에는 미사일 피격 이미지의 수가 적지만 계속해서 다양한 미사일 피격 이미지 및 구름 이미지를 확보하는 중이며, 현재까지 수집된 이미지들(20여개)로는 그룹 1과 그룹 2의 분포를 통해 모두 구별 가능성을 확인하였다. 충분한 데이터 확보 후 후속 연구에서 해당 프로그램의 판독 정확도를 제시하고자 한다.

이 논문에서 사용된 MNIST 데이터셋은 숫자 이미지에만 특화되어 있어, 글자나 다양한 객체 인식을 위한 일반화된 성능을 확인하기 어렵다. 이를 해결하기 위하여 더 복잡한 데이터셋(CIFAR-10, ImageNet 등)을 학습한 후속 연구를 계획 중이며, 이를 통해 단순한 명중정보를 넘어서 각 미사일 피격에 따른 연기 이미지를 분류하는 등의 작업도 수행할 수 있는 머신러닝 개발도 가능하리라 기대한다.

참고문헌

- [1] https://news.sbs.co.kr/news/endPage.do?news_id=N1006972561
- [2] 김세라. 파이썬과 MOY graph를 이용한 항적 분석방법. Journal of KNST (2023) Vol. 6, No. 3, pp. 235-239.
- [3] 김세라, 선박항적이 가지는 특수 군에 대한 연구. Journal of KNST (2023) Vol. 6, No. 4, pp. 376-379.
- [4] https://ko.wikipedia.org/wiki/MNIST_%EB%8D%B0%EC%9D%B4%ED%84%B0%EB%B2%A0%EC%9D%B4%EC%8A%A4
- [5] <https://www.newslock.co.kr/news/articleView.html?idxno=81092>
- [6] [https://namu.wiki/w/%EC%A0%81%EC%9A%B4\(%EA%B5%AC%EB%A6%84\)?from=%EC%A0%81%EC%9A%B4](https://namu.wiki/w/%EC%A0%81%EC%9A%B4(%EA%B5%AC%EB%A6%84)?from=%EC%A0%81%EC%9A%B4)
- [7] <https://namu.wiki/w/%EC%A0%81%EB%9E%80%EC%9A%B4>
- [8] <https://v.daum.net/v/20240919155702844>
- [9] <https://www.youtube.com/watch?app=desktop&v=2VQDAzOCYyo>
- [10] <https://www.youtube.com/watch?app=desktop&v=AkW46jl6nyg>