

<http://dx.doi.org/10.17703/JCCT.2023.9.6.301>

JCCT 2023-11-37

## 증강형 딥러닝 기반 미세먼지 예측 시스템

### Dust Prediction System based on Incremental Deep Learning

장성봉\*

Sung-Bong Jang\*

**요약** 딥러닝은 심층신경망(Deep Neural Network)을 구축하고 대량의 훈련 데이터를 수집한 후, 구축된 신경망을 오랫동안 학습 시켜야 한다. 만약, 학습이 제대로 진행되지 않거나 과적합이 발생하면, 학습은 실패하게 된다. 현재까지 개발되고 있는 딥러닝 도구를 사용할 경우, 훈련데이터 수집과 학습에 많은 시간이 소요된다. 하지만, 모바일 환경의 급격한 도래와 센서 데이터의 증가로 인해, 신경망 학습에 걸리는 시간을 획기적으로 줄일 수 있는 실시간 증강형 딥러닝 기술에 대한 요구가 급격하게 증가하고 있다. 본 연구에서는 미세먼지 센서를 장착한 아두이노 시스템을 사용하여 실시간 증강형 딥러닝 시스템을 구현 하였다. 구현된 시스템에서는 미세먼지 데이터를 5초마다 측정하고 최대 120개가 축적이 되면, 기존에 축적된 데이터와 새로이 측정된 데이터를 데이터셋으로 사용하여 학습을 수행하도록 하였다. 학습 수행을 위한 신경망은 입력층 1개, 은닉층 1개, 출력층 1개로 구성하였다. 구현된 시스템에 대한 성능을 평가하기 위해 학습 시간과 평균 제곱근 오차(root mean square error, RMSE)를 측정 하였다. 실험 결과, 평균 학습 오차는 0.04053796이었으며, 학습 주기당(1 에포크) 평균 학습 시간은 3,447 초 정도의 시간이 걸렸다.

**주요어** : 실시간 기계 학습, 딥러닝, 증강형 데이터, 심층 신경망

**Abstract** Deep learning requires building a deep neural network, collecting a large amount of training data, and then training the built neural network for a long time. If training does not proceed properly or overfitting occurs, training will fail. When using deep learning tools that have been developed so far, it takes a lot of time to collect training data and learn. However, due to the rapid advent of the mobile environment and the increase in sensor data, the demand for real-time deep learning technology that can dramatically reduce the time required for neural network learning is rapidly increasing. In this study, a real-time deep learning system was implemented using an Arduino system equipped with a fine dust sensor. In the implemented system, fine dust data is measured every 30 seconds, and when up to 120 are accumulated, learning is performed using the previously accumulated data and the newly accumulated data as a dataset. The neural network for learning was composed of one input layer, one hidden layer, and one output. To evaluate the performance of the implemented system, learning time and root mean square error (RMSE) were measured. As a result of the experiment, the average learning error was 0.04053796, and the average learning time of one epoch was about 3,447 seconds.

**Key words** : Real-time Machine Learning, Deep Learning, Incremental Data, Deep Neural Networks

\*정회원, 금오공과대학교 산학협력단 교수 (제1저자)  
접수일: 2023년 10월 6일, 수정완료일: 2023년 10월 19일  
게재확정일: 2023년 11월 1일

Received: October 6, 2023 / Revised: October 19, 2023

Accepted: November 1, 2023

\*Corresponding Author: [sungbong.jang@kumoh.ac.kr](mailto:sungbong.jang@kumoh.ac.kr)  
Dept. of Industry-Academy, Kumoh Institute of  
Technnology, South Korea

## I. 서 론

2020년 개최되었던 세계경제 포럼에서는 미래 세상에서는 AI 기술이 더욱더 핵심 역할을 할 것으로 예상하였으며, 이에 따라, 우리나라에서도 교육과 산업 영역에서 AI를 활용하기 위한 시도들이 더욱 활발히 일어나고 있다. AI의 핵심기술인 기계학습은 학습 데이터의 부족, 신경망 구조의 복잡함, 오래 걸리는 학습시간 등으로 실생활에 활용하기에는 매우 어려운 기술로 인식되었다. 하지만, 수년 전부터는, 학습에 필요한 비정형 데이터의 방대한 축적, 병렬처리와 분산처리 기술 발전, 중앙처리장치(Central Processing Unit)의 획기적 성능 향상 등으로 실생활 적용 가능한 기술로 커다란 관심을 받고 있다[1]. 딥러닝은 심층신경망(Deep Neural Network)을 구축하고 대량의 훈련 데이터를 수집한 후, 구축된 신경망을 오랫동안 학습 시켜야 한다. 만약, 학습이 제대로 진행되지 않거나 과적합이 발생하면, 학습은 실패하게 된다. 현재까지 개발되고 있는 딥러닝 도구들을 사용할 경우, 훈련데이터 수집과 학습에 많은 시간이 소요된다[2][3]. 하지만, 모바일 환경의 급격한 도래와 센서 데이터의 증가로 인해, 신경망 학습에 걸리는 시간을 획기적으로 줄일 수 있는 방안이 필요하며, 데이터를 한번에 수집한 후, 한번만 학습하는 정적인 딥러닝 모델 대신에, 증강되는 데이터를 일정한 시간마다 업데이트 한 후, 업데이트된 데이터를 사용하여 점진적으로 학습하는 증강형 딥러닝에 대한 요구가 급격히 증가하고 있다[4-6]. 본 연구에서는 증강형 딥러닝 모델을 사용한 미세먼지 학습 및 예측 시스템을 구현하고 학습 성능을 측정하였다.

## II. 이론고찰

Y. Li [7]는 매우 빠른 속도로 생성되며 내용 및 분포 특성이 모두 고속 동적으로 변경되며 실시간으로 처리되어야 하는 실시간 환경에서, 증강된 학습 데이터를 업데이트하고 이를 실시간으로 학습하기 위한 새로운 딥러닝 기법을 제안하였다. 해당 기법에서는 벡터 형태로 수집된 데이터를 텐서라는 3차원 공간 데이터로 변경하고 심층 신경망 매개변수와 구조를 점진적으로 업데이트 하면서 학습을 진행 한다. 증강 딥러닝에서 발생하는 신경망 매개변수 폭증을 방지하기 위해 1차원

근사 접근 방식(first-order approximate approach)을 사용하여 증강된 데이터의 특성을 빠르게 학습 하도록 하였다. 성능 평가를 위해, 제안된 기법을 구현하고 온라인에서 증강 데이터를 사용하여 학습한 결과, 딥러닝 모델을 효율적으로 업데이트 하였으며, 실시간 데이터 스트림 학습을 효율적으로 수행함을 보였다.

Z. Chen [8] 연구에서는 실내 측위를 정확하고 효율적으로 수행하기 위한 증강형 딥러닝 접근 방식을 제안 하였다. 1단계에서는 합성곱 신경망(Convolutional Neural Network, CNN)을 사용하여 RF(Radio-Frequency) 측정에서 얻은 실제 데이터 세트를 기반으로 고유한 특징을 추출하고 두 번째 단계에서는 첫 번째 단계에서 얻은 환경 유형에 대한 지식을 사용하여 실내 측위를 수행하도록 하였다. 실험 결과, 제안된 방식이 현지화 정확도와 계산 효율성이 크게 향상되어 벤치마크 방법보다 성능이 우수함을 보여 주었다.

M. Baharani[9]의 논문에서는 고주파 전력 전자 컨버터의 실시간 신뢰성 모델링 및 예측을 위한 심층 학습 신뢰성 인식(Deep RACE) 딥러닝 모델을 제안 하였다. 제안된 기법에서는 거의 실시간에 가까운 안정적 패턴 인식을 생성하기 위해 알고리즘이 포함된 통합 시스템(클라우드에서 에지 노드까지)으로 솔루션을 제공 하였다. 또한, 소자 저항 변화에 기반한 집단 신뢰도 훈련과 추론을 위해 쌓여진 장단기 메모리를 기반으로 한 딥러닝 알고리즘을 제안 하였다. 나사에서 제공하는 고주파 데이터 세트를 사용하여 실제 딥러닝 모델을 구축하고 실험한 결과, 기존에 사용된 기법(칼만 필터 또는 입자 필터)에 비해 훨씬 더 높은 정확도를 나타내었다. G. A. Tahir [10]는 식품 이미지 자동 인식을 위한 새로운 증강형 딥러닝 모델을 제안 하였다. 제안된 모델에서는 전이 학습, 클래스 증강 데이터, 증강 데이터의 특징 선택을 위한 개방형 연속 학습 기법을 사용 하였다. 제안 된 기법의 성능을 평가하기 위해, 실제 모델을 구현하고 4종류의 표준 식품과 파키스탄 식품 데이터셋을 사용하여 훈련 시킨후, 모델을 평가 하였다. 실험 결과, 제안된 기법이 증강된 클래스와 데이터를 학습 하는데 더 우수한 성능을 보였으며, 도메인 변경에 효율적으로 적응할 수 있음을 보였다. A. A. Darem[11]은 악성 코드를 탐지하기 위해 개발된 기존의 정적인 딥러닝 모델들은 수시로 진화하는 악성코드를 탐지하기에는 많은 문제점이 있음을 지적하였다. 이를 해결하고 수시

로 진화하는 악성코드의 변종을 효율적으로 발견하기 위해, 적응적 행동 기반 증강 배치 학습 기법을 제안하였다. 제안된 기법에서는 악성코드 실행 API를 수집하고 이로부터 특징들을 수집하고 악성코드라고 판단되면, 최초 특징을 데이터로 저장한다. 만약, 해당 악성코드가 진화하여 변경될 이를 이에 대한 특징들은 다시 기존의 유사한 악성코드 API와 비교하여 증강 데이터를 수집한다. 수집된 증강 데이터는 적응적 배치 증강 학습 모델을 사용하여 학습 시킨다. 성능 평가 결과 기존의 정적 딥러닝 모델에 비해 탐지율 및 효율성 측면에서 훨씬 우수함을 보여 주었다. 실험결과, 월 1.35회만 업데이트 해도 평균 99.41%의 악성 코드를 탐지할 수 있었다. S. Younan[12]은 웨어러블 기기를 장착한 개인의 일상 생활 인식 정확도 향상을 위해, 기존의 정적 모델이 아닌 증강형 딥러닝 모델을 제안 하였다. 제안된 기법에서는 표준화된 일상 생활 데이터를 사용하여 딥러닝 모델을 조정하고 개인 맞춤형 데이터를 점진적으로 수집하여 학습 모델을 훈련 시킨다. 제안된 모델의 장점은 모든 데이터를 사용하여 처음부터 다시 학습할 필요 없으며, 정확도 면에서 기존의 정적모델보다 우수하다는 점이다. 실험 결과, 기존 모델보다 최대 19%정도 성능이 향상 되었음을 보이고 있다.

이 이외에도 매우 많은 관련 연구가 이루어지고 있으며, 다양한 분야에 증강형 딥러닝이 응용되고 있다 [13-15]. 미세먼지 수집 및 예측 분야에서는 아직까지 증강형 딥러닝을 적용한 연구가 없었다.

### III. 연구 방법

본 과제에서는 아두이노 보드에 미세먼지 센서를 장착하고 일정한 시간 간격으로 증강되는(Incremental)미세먼지 측정값을 수집하고 이를 아두이노 CPU로 학습시키고 학습된 결과는, USB를 통해 PC로 전송 하도록 하였다. 구축된 하드웨어 시스템은 [그림 1]과 같다.

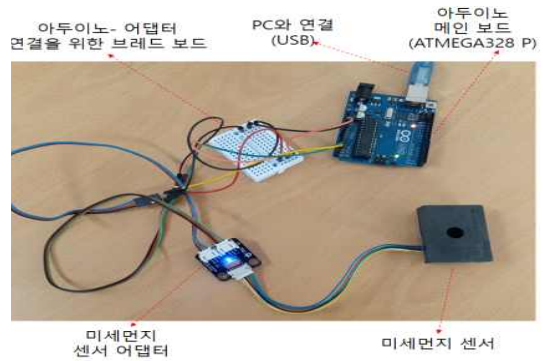


그림 1. 실험을 위한 하드웨어 구성  
 Figure 1. Hardwares for experiment

미세먼지 데이터 수집, 데이터 전처리, 통신 기능은 C++언어로 구현 하였으며, 아두이노 개발 환경인 스케치에서 컴파일 하여 이진 코드로 만든다. 만들어진 이진 코드는 스케치를 사용하여 아두이노 플래시 메모리에 심는다. 미세먼지 데이터를 읽고 학습하는 부분은 C 언어로 구현하여 아두이노 센서에서 실시간으로 동작 되도록 구현 하였다. 전체 연구 방법은 [그림 2]와 같다. 각 단계에 대한 자세한 설명은 다음에서 설명한다.

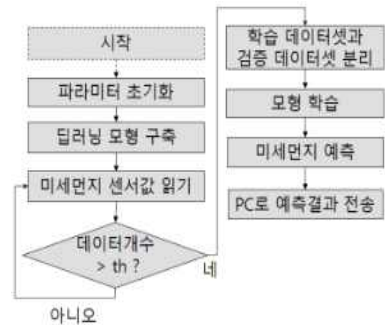


그림 2. 증강된 데이터를 사용한 딥러닝 과정  
 Figure 2. Incremental and real-time deep learning process

첫째, 활성화 함수, 비용 함수, 초기 가중치, 학습률과 같은 딥러닝 하이퍼 파라미터를 초기화한다. 본 연구에서 사용한 딥러닝 파라미터값은 [표 1]과 같다.

표 1. 실험에서 사용된 딥러닝 파라미터  
 Table 1. Deep learning parameters for experiment

파라미터 이름	파라미터 값
활성화 함수	시그모이드(sigmoid)
학습 주기	100
최적화 함수	SGD(Stochastic Gradient Descent)

둘째, 딥러닝 모델을 구축 한다. 실시간 학습을 위해서는 학습에 걸리는 시간을 최대한 짧게 하고 오차는 최소로 하도록 신경망을 구성해야 한다. 모델을 복잡하게 구성할 경우, 오차는 줄어들지만 학습에 시간이 너무 오래 걸린다. 본 연구에서는 1개의 입력층, 1개의 은닉층, 1개의 출력층을 가진 학습 모델을 구축하여 사용 하였다. 은닉층의 개수를 더 늘리려 했으나, 아두이노 장비의 메모리가 부족하여, 더 이상 늘리수가 없었다. 실험에 사용된 신경망 모델은 [그림 3]과 같다.

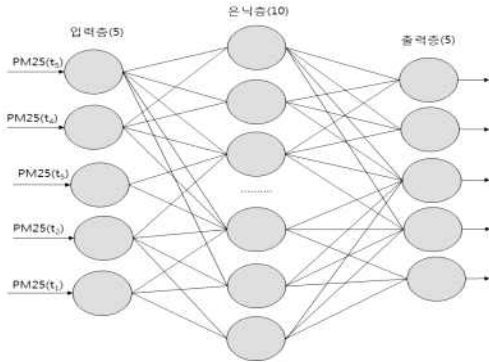


그림 3. 실시간 딥러닝을 위한 심층 신경망 모델  
Figure 3. Deep learning model for real-time deep learning

세번째 단계에서 미세먼지 센서 데이터를 읽고 버퍼에 저장한다. 버퍼의 크기는 110으로 설정하였으며, 버퍼가 모두 차게 되면 학습을 수행 하도록 하였다. 버퍼가 채워지지 않을 경우에는 계속 데이터를 읽도록 하였다. 최초 모두 버퍼가 채워지면, 학습을 진행한다. 한번의 학습을 진행한 후, 5초 간격으로 미세먼지 데이터를 수집 증강한 후, 20개가 되면, 다시 학습을 수행한다. 즉, 100초 마다 증강된 데이터를 사용하여 학습을 주기적으로 수행하도록 하였다.

네번째, 데이터를 표준화하고 학습데이터와 검증 데이터를 분리한다. 데이터 표준화는 데이터값 사이의 크기 차이로 인해 발생하는 편차를 줄이기 위해 0과 1 사이의 정규화된 값으로 변경한다. 정규화 값을 생성하기 위해 다음과 같은 공식을 사용한다.

$$\text{표준화값} = (\text{원래값} - \text{최소값}) / (\text{최대값} - \text{최소값}) \quad \text{--- (1)}$$

표준화된 값을 학습 데이터와 검증 데이터로 나눈다. 본 실험에서는 데이터의 80%는 학습 데이터로 20%는 검증 데이터로 사용하였다.

다섯번째 단계에서 학습을 수행 한다. 학습 수행을 위한 비용함수는 평균제곱오차 계산함수를 사용하였다. 평균 제곱오차 함수의 공식은 다음과 같다.

$$\text{평균제곱오차} = \frac{1}{\text{총데이터수}} \sum_{n=1}^{\text{총데이터수}} (\text{정답값}_n - \text{예측값}_n)^2 \quad \text{--- (2)}$$

위의 공식에서 총데이터수는 학습에 사용된 데이터값 개수를 의미하며, 출력층의 노드숫자와 동일하다. 정답값은 목표 데이터 값을 의미하며, 예측값은 학습된 심층 신경망 모델을 사용하여 예측한 값을 의미한다. 본 연구에서는 수집된 미세먼지 데이터셋에 대해 100번의 주기를 학습한 후, 에포크 평균 오차값을 측정하였다. 에포크 평균 오차값은 다음과 같이 구하였다.

$$\text{평균오차} = \frac{1}{\text{총에포크수}} \sum_{i=1}^{\text{총에포크수}} \text{에포크평균제곱오차}_i \quad \text{--- (3)}$$

마지막 단계에서는 증강된 데이터를 사용한 학습 결과를 USB를 통해서, PC로 전송하도록 하였다. PC로 전송된 데이터는 아두이노 스케치의 시리얼 모니터 기능을 사용하여, 화면에 출력되도록 하였다. 화면에 출력된 내용은 캡처하여, 메모장으로 복사한 후, 저장 하였다.

본 연구에서 제안한 증강형 미세먼지 시스템과 일반적인 미세먼지 시스템의 차이점은 다음과 같다. 첫째, 증강형 미세먼지 시스템에서는 실시간 생성되는 데이터를 일정한 시간 간격으로 수집하여 학습 데이터로 사용한다. 두 번째, 예측을 위한 딥러닝 모델 학습이 한번에 그치지 않고 데이터를 수집할 때마다, 주기적으로 이루어 진다. 셋째, 주기적으로 학습을 수행할 때마다 오래된 데이터는 빼고 가장 최근에 수집한 데이터를 사용한다.

#### IV. 연구 결과

본 연구에서는 스케치를 사용하여 딥러닝 코드를 작성 및 컴파일 한 후, 아두이노 보드에 업로드 하였다. [그림 4]는 증강 딥러닝 코드 작성, 컴파일, 업로드 화면을 보여주고 있다.

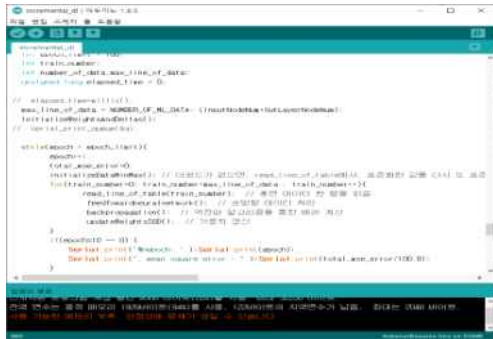


그림 4. 증강형 딥러닝 코드 작성, 컴파일, 업로드 화면  
 Figure 4. Screen for editing, compiling, and uploading incremental deep learning codes

실험 결과 데이터는 스케치에서 제공하는 “시리얼 모니터” 기능을 이용하여 PC로 전송한 후, 화면에 출력 되도록 하였다. [그림 5]는 5분 단위로 센서 데이터를 읽고 딥러닝 모델을 학습한 후, 실험 오차와 학습 시간을 PC로 전송하는 화면을 보여 주고 있다.

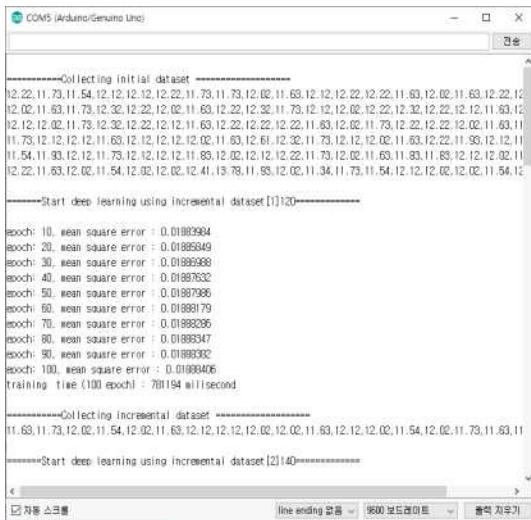


그림 5. 증강 데이터 수집과 학습 결과 출력 화면  
 Figure 5. Screen for displaying incremental data collection and learning results

구현된 시스템의 성능 평가를 위해, 20번의 실시간 학습을 수행한 후, 오차값을 측정하여 보았다. 측정결과 는 [표 2]과 같다.

표 2. 실험 결과

Table 2. Experiment results

실험 횟수	평균제곱오차 (MSE)	학습 시간 (milisecond(1/1000초))
1	0.01888406	781194
2	0.02173845	1012558
3	0.01401501	1242822
4	0.01841318	1472918
5	0.01828224	1703024
6	0.01556840	1934979
7	0.01399929	2167228
8	0.00778674	2399071
9	0.01396008	2631815
10	0.07070976	2864382
11	0.06959473	3097751
12	0.06826161	3330824
13	0.02984795	3563486
14	0.03493240	3796659
15	0.03371150	4029742
16	0.03127457	4262624
17	0.03454056	4493869
18	0.02402401	4725429
19	0.06524055	4958355
20	0.08534449	5190201
21	0.07224658	5421772
22	0.08092285	5655326
23	0.07218444	5888989
24	0.05742757	6122057
평균	0.04053796	3447794

24 번의 증강학습 결과 평균제곱오차값은 0.04053796 이었으며, 학습 시간은 평균 3447794(milisecond)로 약 3447초가 걸렸다. 실험 결과에서 알 수 있듯이, 오차값 은 평균 에상을 벗어나지 않았으나, 학습시간은 매우 오래 걸렸음을 알 수 있다. 학습 시간이 오래 걸린 이유는 CPU성능의 차이에서 비롯되었다. 실험에서 사용한 아두이노의 CPU성능은 일반 PC의 1/1000정도 밖에 되지 않는다. 따라서, 1000배 이상 느릴 수밖에 없다. 증강형 딥러닝은 주로 성능이 좋지 않은 소규모 장비에 주로 사용되기 때문에, 학습 시간이 길 수 밖에 없지만, 스스로 알아서 진화해 나가기 때문에 인간의 개입을 최소화 할 수 있는 장점이 있다.

## V. 결 론

센서 빅데이터의 증가, IoT환경, 모바일 컴퓨팅 환경의 등장으로 전통적인 딥러닝 학습 방법에서 실시간 딥러닝 학습에 대한 요구가 급격하게 증가하고 있다. 본 연구에서는 임베디드 시스템 내부의 센서에서 실시간

으로 데이터를 수집하고 학습을 수행하는 실시간 딥러닝 시스템을 구현 하였다. 구현된 시스템에서는 입력층 1개, 은닉층 1개, 출력층 1개로 구성된 심층 신경망을 구축하고 5초마다 미세먼지 데이터를 수집한 후, 축적된 데이터와 새로이 축적된 데이터를 사용하여 모형을 학습 시켰다. 성능 평가를 위해 학습 시간과 평균 제곱근 오차(root mean square error, RMSE)를 측정 한 결과, 평균 학습 오차는 0.04053796이었으며, 평균 학습 시간은 3447초 정도의 시간이 걸렸다. 실험 결과, 학습 시간이 매우 길게 걸렸음을 알 수 있다. 추후 연구로는 학습 시간을 줄일 수 있는 방안을 찾는 것이다.

## References

- [1] S.-H. Kim, C. Lee, C.-H. Youn, "An Accelerated Edge Cloud System for Energy Data Stream Processing Based on Adaptive Incremental Deep Learning Scheme," *IEEE Access*, Vol.8, pp.19534-195358,2020(DOI: 10.1109/ACCESS.2020.33771)
- [2] Y. Cao, M. Jia, P. Ding, X. Zhao, Y. Ding, "Incremental Learning for Remaining Useful Life Prediction via Temporal Cascade Broad Learning System With Newly Acquired Data," *IEEE Transactions on Industrial Informatics*, Vol. 19, No. 4, pp. 6234-6245, 2023(DOI:10.1109/TII.2022.3201977)
- [3] X. Wang, J. Zhou, J. Fan, "IDUDL: Incremental Double Unsupervised Deep Learning Model for Marine Aquaculture SAR Images Segmentation," *IEEE Transactions on Geoscience and Remote Sensing*, Vol. 60, 2022(DOI: 10.1109/TGRS.2022.3203071)
- [4] M. Data, M. Aritsugi, "T-DFNN: An Incremental Learning Algorithm for Intrusion Detection Systems," *IEEE Access*, Vol. 9, pp. 154156-154171, 2021(DOI: 10.1109/ACCESS.2021.3127985)
- [5] Y. Xiang, Y. Miao, J. Chen, Q. Xuan, "Efficient Incremental Learning Using Dynamic Correction Vector," *IEEE Transactions on Geoscience and Remote Sensing*, Vol. 8, pp. 2309-23099, 2020(DOI:10.1109/ACCESS.2019.2963461)
- [6] M. Masana, X. Liu, B. Twardowski, M. Menta, A. D. Bagdanov, J. V. D. Weijer, "Class-Incremental Learning: Survey and Performance Evaluation on Image Classification," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 45, No.5, pp. 5513-5533, 2023(DOI:10.1109/TPAMI.2022.3213473)
- [7] Y. Li and M. Zhang, "Online Real-Time Analysis of Data Streams Based on an Incremental High-Order Deep Learning Model," *IEEE Access*, Vol. 6, pp. 77615 - 77623, 2018(DOI: 10.1109/ACCESS.2018.2883666)
- [8] Z. Chen, M. I. AlHajri, M. Wu, N. T. Ali, R. M. Shubair, "A Novel Real-Time Deep Learning Approach for Indoor Localization Based on RF Environment Identification," *IEEE Sensors Letters*, Vol. 4, No. 6, pp.735-740, 2020(DOI:10.1109/LENS.2020.2991145)
- [9] M. Baharani, M. Biglarbegian, B. Parkhideh, H. Tabkhi, "Survey of deep learning software tools,"Real-Time Deep Learning at the Edge for Scalable Reliability Modeling of Si-MOSFET Power Electronics Converters," *IEEE Internet of Things Journal*, Vol. 4, No. 6, pp. 7375-7385, 2019(DOI: 10.1109/JIOT.2019.2896174).
- [10]G. A. Tahir and C. K. Loo, "An Open-Ended Continual Learning for Food Recognition Using Class Incremental Extreme Learning Machines," *IEEE Access*, Vol.8, pp. 82328-82346, 2020 (DOI: 10.1109/ACCESS.2020.2991810)
- [11]A. A. Darem, F. A. Ghaleb, A. A. Al-Hashmi, J. H. Abawajy, S. M. Alanazi, A. Y. Al-Rezami, "An Adaptive Behavioral-Based Incremental Batch Learning Malware Variants Detection Model Using Concept Drift Detection and Sequential Deep Learning," *IEEE Access*, Vol.9, pp. 97180 - 97196, 2021(DOI:10.1109/ACCESS.2021.3093366)
- [12]S. Younan and M. Abu-Elkheir, "Deep Incremental Learning for Personalized Human Activity Recognition on Edge Devices," *IEEE Canadian Journal of Electrical and Computer Engineering*, Vol.45, No.3, pp. 215-221, 2022(DOI:10.1109/ICJECE.2022.3199227)
- [13]Q. Zhang, L. T. Yang, Z. Chen, P. Li, "Incremental Deep Computation Model for Wireless Big Data Feature Learning," *IEEE Transactions on Big Data*, Vol. 6, No. 2, pp. 248-257,2020(DOI: 10.1109/TBDATA.2019.2903092)
- [14]X. Li, S. Dong, Q. Su, M. Yu, X. Li, "Adaptive Threshold Hierarchical Incremental Learning Method," *IEEE Access*, Vol. 11, pp.12285 -12293, 2023(DOI: 10.1109/ACCESS.2023.3242688)
- [15]L. Guan, Y. Wu, "AReduce the Difficulty of Incremental Learning With Self-Supervised

Learning," *IEEE Access*, Vol. 9, pp. 128540-128549,  
2021(DOI: 10.1109/ACCESS.2021.3112745)

※이 연구는 금오공과대학교 대학 학술연구비  
로 지원되었음(2021년)