

Utilizing Mean Teacher Semi-Supervised Learning for Robust Pothole Image Classification

Inki Kim*, Beomjun Kim*, Jeonghwan Gwak**

*Student, Dept. of IT-Energy Convergence, Korea National University of Transportation, Chungju, Korea

*Student, Dept. of IT-Energy Convergence, Korea National University of Transportation, Chungju, Korea

**Professor, Dept. of Software, Korea National University of Transportation, Chungju, Korea

[Abstract]

Potholes that occur on paved roads can have fatal consequences for vehicles traveling at high speeds and may even lead to fatalities. While manual detection of potholes using human labor is commonly used to prevent pothole-related accidents, it is economically and temporally inefficient due to the exposure of workers on the road and the difficulty in predicting potholes in certain categories. Therefore, completely preventing potholes is nearly impossible, and even preventing their formation is limited due to the influence of ground conditions closely related to road environments. Additionally, labeling work guided by experts is required for dataset construction. Thus, in this paper, we utilized the Mean Teacher technique, one of the semi-supervised learning-based knowledge distillation methods, to achieve robust performance in pothole image classification even with limited labeled data. We demonstrated this using performance metrics and GradCAM, showing that when using semi-supervised learning, 15 pre-trained CNN models achieved an average accuracy of 90.41%, with a minimum of 2% and a maximum of 9% performance difference compared to supervised learning.

▶ **Key words:** Pothole detection, Semi-supervised learning, Knowledge distillation, Mean Teacher technique, pre-trained models, transfer learning

[요 약]

포장도로에서 발생하는 포트홀은 고속 주행 차량에 치명적인 영향을 미치며, 사망사고를 유발할 수 있는 도로상의 장애물이다. 이를 방지하기 위해 일반적으로는 작업자가 직접 포트홀을 탐지하는 방식을 사용해왔으나, 이는 작업자의 안전 문제와 예측하기 어려운 범주에서 발생하는 모든 포트홀을 입력으로 탐지하는 것이 비효율적이기 때문에 한계가 있다. 또한, 도로 환경과 관련된 지반 환경이 포트홀 생성에 영향을 미치기 때문에, 완벽한 포트홀 방지는 어렵다. 데이터셋 구축을 위해서는 전문가의 지도하에 라벨링 작업이 필요하지만, 이는 매우 시간과 비용이 많이 필요하다. 따라서, 본 논문에서는 Mean Teacher 기법을 사용하여 라벨링된 데이터의 샘플 수가 적더라도 지도학습보다 더욱 강한 포트홀 이미지 분류 성능을 보여준다. 이러한 결과는 성능지표와 GradCAM을 통해 입증되었으며, 준지도학습을 사용할 때 15개의 사전 학습된 CNN 모델이 평균 90.41%의 정확도를 달성하며, 지도학습과 비교하여 2%에서 9%의 차이로 강한 성능을 나타내는 것을 확인하였다.

▶ **주제어:** 포트홀 탐지, 준지도 학습, 지식 증류, Mean Teacher 모델, 사전학습 모델, 전이 학습

- First Authors: Inki Kim, Beomjun Kim, Corresponding Author: Jeonghwan Gwak
- *Inki Kim (cv2@kakao.com), Dept. of IT-Energy Convergence, Korea National University of Transportation
- *Beomjun Kim (zhskfkseh@gmail.com), Dept. of IT-Energy Convergence, Korea National University of Transportation
- **Jeonghwan Gwak (jgwak@ut.ac.kr), Dept. of Software, Korea National University of Transportation
- Received: 2023. 04. 24, Revised: 2023. 05. 12, Accepted: 2023. 05. 15.

I. Introduction

일반적인 비포장도로(Unpaved road)는 아스팔트나 콘크리트 등으로 포장되어있지 않은 도로를 의미한다. 비포장도로는 도로 표면이 고르지 못한 이유로 인해 차량이 흔들리며, 계속되는 흔들림에 의한 진동으로 인해 화물이 손상되거나 승객 부상의 위험 또한 커진다. 대개 흙이나 자갈로 이루어져 있으므로 자갈이나 흙으로 인한 표면의 흠집이나 유리가 깨지는 일도 빈번하게 발생한다.

이러한 단점을 극복하기 위해 도로포장(Road Pavement)은 도로의 표면을 콘크리트, 아스팔트 등으로 다진 것을 의미한다. 대부분은 도로의 내구성 향상 및 차량 운전자와 승객의 승차감 향상 및 통행 속도의 증가에 목적이 있다. 일반적인 도로포장 사용 재료는 주로 콘크리트 및 아스팔트를 통해 도로를 포장한다.

하지만 포장도로에도 단점은 여전히 존재한다. 도로에 움푹 패이거나 손상으로 인해 깨져버린 아스팔트나 콘크리트에 발생하는 포트홀(Pothole)은 도로 관리의 주체인 국가나 지자체에도 재산적인 손해를 발생시키며, 운전자에게는 사망사고까지 발생시킬 수 있는 도로 위의 장애물이다. 일반적으로 도로의 포장의 다짐 부족, 과적 차량 등 다양한 원인으로 균열이 최초에 발생하게 된다. 이때 차량이 통행하면서 발생한 균열은 점점 파임이 발생하게 되며, 주위로 확산하기 시작한다. 특히 아스팔트로 포장된 도로는 습기에 매우 취약하며, 여름에 특히 균열 발생의 확률이 급증한다. 겨울철 또한 눈이 내리고 염화칼슘으로 인해 눈이 녹으며, 이로 인해 다시 재결빙된 눈은 아스팔트를 파괴하여 균열을 발생시킨다. 포트홀은 발생 원인도, 규모도 매우 다양하여 한정된 카테고리로는 분류하여 대처할 수 없고, 예측할 수 없는 위치와 시기에 발생하므로 지속적인 추적 및 관리와 유지보수가 필요하다.

포트홀을 방지하기 위해 콘크리트 도로를 아스팔트 도로로 변경하여 하중이 높은 대형차량이 통행하는 도로의 포장을 대체하는 정책도 존재하지만, 콘크리트는 배수가 느리므로 고인 물이 얼어 미끄럼 사고의 발생 확률이 증가하며, 차량 주행 시 소음이 전혀 발생하지 않는 아스팔트에 비해 콘크리트 도로는 공명음이 발생하여 주변 거주 시설 및 근린시설에 소음피해를 유발할 수도 있는 단점이 존재한다. 또한, 경제적인 비용으로도 차이가 존재하여 모든 도로를 콘크리트 도로로 포장을 변경할 수가 없다.

따라서, 현재 자동차가 주행하는 콘크리트 포장 도로에 포트홀 발생으로 인한 비정상적인 도로의 포장면은 주행 중인 자동차의 안전한 운행에 문제를 발생시킬 수 있고,

운전자 또는 승객에게 안전 문제에 대해 직결적으로 연결되어 있다. 따라서, 도로 표면 위에 존재하는 포트홀을 조기에 감지하고 선제적인 조치를 진행하는 것은 매우 중요하다. 아래 그림 1은 도로 위에서 발생하는 포트홀의 예시이다.



Fig. 1. Example of a Pothole

국가 차원에서도 노후도로 포장보수 확대 및 포트홀의 지속적인 유지보수를 통해 발생 현황은 점차 감소하고 있으나, 관련된 피해소송 및 피해보상의 발생 건수와 금액은 지속해서 증가하고 있다. 따라서, 지속해서 발생하고 있는 포트홀을 효율적으로 유지보수 및 지속적인 추적 관리를 통해 피해를 최소화하고, 담당자 또는 전문가가 직접 수작업 방식으로 포트홀을 탐지하는 것이 아닌, 영상 내에서 나타나는 특징의 유·무를 판별할 수 있는 딥러닝 기반 탐지 방법이 필요하다. 하지만 영상 내에서 나타나는 포트홀의 유무를 판별하기 위해 딥러닝 모델을 구축해야 하고, 딥러닝 모델을 구축하기 위해서는 정상 및 비정상(포트홀)으로 구성된 라벨링된 데이터셋을 학습에 사용하여 테스트 샘플을 분류하는 기존의 지도학습 방법을 사용하는 것은 매우 한계적이다. 도로에서 발생하는 포트홀은 발생하는 그 원인이 매우 다양해 모든 데이터를 확보하기 어려우며, 대량의 데이터셋으로부터 모든 데이터를 라벨링하는 것은 프로세스를 적용하기에 상당히 긴 시간과 적지 않은 비용이 소모된다. 또한, 소량의 라벨링된 데이터를 확보하였다고 하더라도, 라벨링된 데이터가 전체 포트홀에서 나타나는 특징 분포를 학습하기에 불충분할 수도 있으므로 학습 과정에서 경험하지 못하였던 테스트 데이터가 입력된다면 정상적으로 분류하지 못하는 일반화 문제가 발생할 수도 있다.

이에 따라, 준지도학습(Semi-Supervised learning) 기법은 소량의 라벨링된 데이터를 통해 가중치가 부여된 딥러닝 모델을 통해 대량의 라벨링 되지 않은 데이터를 통해 미세하게 결정 경계(Decision Boundary)를 Fitting 시키

는 기법을 의미한다. 따라서, 본 논문에서는 적은 라벨링된 데이터와 대용량의 라벨링 되지 않은 데이터를 혼합해서 사용하는 준지도 학습기법을 사용하여 학습된 딥러닝 모델을 대용량의 라벨링된 데이터로 학습된 포트홀 탐지 성능을 지도 학습(Supervised learning)과 비교한다.

또한 포트홀을 탐지하여 사용자에게 알람을 제공하거나, 보수해야 하는 위치를 제공할 수 있는 장치는 대부분 저전력 기반 엣지 디바이스 위주로 구성되어 있으므로, 많은 연산량과 파라미터를 요구하는 딥러닝 모델은 사용할 수 없다. 따라서, 차량 내부 혹은 엣지 디바이스에서 포트홀을 탐지하기 위해서는 저비용·저전력·고효율 기반 장치의 필요하게 된다. 하지만, 해당 조건을 만족하기 위해서 엣지 디바이스의 가격은 높아지므로, 본 논문에서는 경량화 기법의 하나인 지식 증류(Knowledge Distillation)를 사용하여 연산량과 파라미터는 감소시키고, 효율성은 유지한다. 지식 증류 기법을 사용하기 위해 Mean Teacher Student[1] 모델 기반으로 한 실험을 진행하였다. 해당 지식 증류를 통해 Teacher Network로부터 증류한 가중치를 Student Network로 전이함으로써, 상대적으로 적은 연산량과 파라미터를 기반으로 제한된 Computing Resource에서도 효율적으로 사용할 수 있다.

실험 방법으로는, 각 데이터의 라벨링된 데이터와 라벨링되지 않은 데이터의 비율을 조절하여 지도학습 기반 CNN 모델들의 포트홀 탐지 성능과 준지도학습 기법 기반 Mean Teacher Student를 통해 지식 증류된 딥러닝 모델의 성능과 실제 근거 있는 예측을 확인하기 위한 GradCAM[2]의 결과 비교를 통해 준지도학습 기법 기반 포트홀 탐지 기법의 성능을 입증하였다. 해당 결과로서 제안된 준지도학습 기법 기반 지식 증류된 딥러닝 모델이 적은 수의 라벨링된 데이터를 사용하더라도 효과적인 포트홀 탐지 성능을 향상하는 데 도움이 될 수 있음을 보여준다. 본 논문의 구성은 섹션 2는 Related Work를 서술하며, 실험에 사용된 데이터셋과 Proposed Method는 섹션 3에 제시되어 있다. 실험 결과 및 분석은 섹션 4에 서술되어 있으며, 결론 및 추후 연구 진행 사항은 섹션 5에 서술함으로써 정리하였다.

II. Related Work

포트홀을 탐지하기 위한 연구는 과거부터 지속해서 진행됐으며, 이에 따라 다양한 기법에 따른 포트홀 탐지 방법이 제안되어왔다. 해당 섹션에서는 총 네 가지 주제로

나누어 포트홀 탐지와 데이터셋 구축의 한계를 극복한 방법을 소개하고, 제안된 방법들의 한계점에 관해 서술하고 본 논문에서 제안하는 기법과 비교한다.

국내에서는 포트홀 탐지를 위해 다양한 컴퓨터 비전 기반 포트홀 탐지를 위한 영상 처리 기법을 제안하였다. 진동방식과 유사한 방법으로서, 2014년 서울에서 실시한 대중교통 버스에 포장 파손 감지 센서와 GPS 모듈을 장착하여 센서를 통해 충격을 감지하여 현재 위치를 GPS를 기반으로 하여 유지관리기관에 전송하여 유지보수를 진행하였다. 또한, 택시에는 “차도 모니터링단”을 적용하여 개인택시 운전자가 포트홀을 발견할 때 택시 내에 설치된 버튼을 통해 포트홀의 위치가 자동으로 전송되는 기법을 적용하였다.

하지만, 버스의 경우 포트홀을 통과하였다고 하더라도, 센서에 진동이 감지되지 않거나, 버스가 포트홀을 통과하지 않는다면 감지가 되지 않는다는 단점이 존재한다. 또한, 서울시의 경우 버스 주행차로가 별도로 존재하기 때문에, 일반 주행차선의 경우 포트홀의 검출이 어렵다는 단점이 존재한다. 택시도 마찬가지로, 운전자가 포트홀을 직접 주행하지 못한 경우 포트홀의 정상적인 검출이 불가능하다는 단점이 존재한다. 추가로 맨홀과 과속방지턱을 포트홀로 오탐지 하는 경우도 존재하였다.

또한, 레이저 센서 기반 스캐닝 혹은 적외선 센서를 통한 3D 영상 기반 포트홀 탐지 기술[3-4]이 지속해서 연구 및 제안되고 있지만, 한계적인 센서의 탐지영역의 부족과 높은 장비의 비용과 운용성의 어려움으로 인해 많은 어려움이 존재한다.

전처리 기반 포트홀 탐지 기법은, 영상에서 나타나는 저차원의 특징들을 전처리 기법을 통해 hand-crafted 특징들을 학습시키는 기법이다. S. Nienabar et al.[5]는 차량 내부에서 촬영되어 입력되는 원본 비디오 프레임 영상에서 관심 영역에 해당하는 도로 영역만을 추출하고, 추출된 관심 영역을 통해 Convex Hull 알고리즘을 적용한다. 전처리가 완료된 이미지에 Canny Filter를 적용함으로써, 이미지에서 Contour 정보로 나타나게 된다. 해당 기법은 딥러닝 모델을 최적화하기 위한 학습 과정은 없지만, 실시간으로 적용되기에 많은 연산량이 필요하며, 픽셀들의 연산을 처리하는 양이 많거나, 적정속도와 차간거리를 유지하지 않을 때 검출이 불가능하다는 단점이 존재한다.

K. Vigneshwar et al.[6]은 입력된 포트홀 이미지 크기를 고정되지 않은 상태에서 이미지의 크기를 변경하고, 이미지를 Grayscale로 전환하여 Median Filter, Difference of Gaussian-Filtering을 적용해 데이터셋을 생성하였다. 그 이후 Canny Algorithm, Otsu's

thresholding Algorithm, K-Means Clustering, Fuzzy C-Means Clustering의 분할 성능을 각각 비교하였다. 본 논문에서는 다양한 전처리 기법 기반 포트홀 분할 성능을 제안하였지만, 주행 중인 차량에서 hand-crafted 특징 기반 포트홀 탐지는 연산량으로 인해 불가능에 가깝다.

딥러닝을 기반으로 하는 포트홀 탐지는 국내 및 해외에서는 끊임없이 연구되고 있다. 포트홀 탐지의 특성상 주행 중인 차량의 블랙박스나 같은 옛지 디바이스에서 실시간 탐지를 기반으로 하는 기법과 일반적인 영상을 통해 포트홀을 탐지하는 기법으로 연구가 진행된다. A. A. Shaghorui et al. , M. H. Asad et al.[7-8]은 옛지 디바이스에서 사용할 수 있는 One-Stage 딥러닝 기법을 기반으로 한 SSD(Single Shot Multibox Detector)와 YOLO(You Only Look Once)를 통하여 포트홀 검출 성능을 실험하였다. 학습 시나리오는 모든 데이터를 사용하여 이미지의 해상도만을 변경하여 다양한 조건으로 딥러닝 기법이 포트홀을 탐지할 수 있는 능력을 입증하였으나, 무수히 많은 데이터셋을 수집해야 하는 문제와 Label을 진행해야 하는 한계점을 극복하지 못하였다. N. Çınar and M. Kaya, I. K. Kim et al. [9-10]은 포트홀 탐지를 위한 전이 학습이 학습 데이터셋의 샘플 수가 적더라도 높은 성능을 보유하는 것을 입증하였으나, 급격하게 적은 데이터 샘플로 인하여 강인한 성능의 타당성이 입증되지 않았다. 본 논문에서는 포트홀 탐지를 위한 데이터셋 구축의 한계점과 Label의 한계점을 극복하기 위한 실험을 제안하였으며, 이를 극복하기 위하여 지식 증류 기법 기반의 Mean Teacher Student 기법을 통한 사전학습을 통하여 해당 문제점을 극복하였고, 강인한 성능을 입증하였다.

딥러닝에서의 지식 증류는 Teacher Network에서 추출된 지식을 Student Network로 transfer하는 일련의 모델 경량화 기법들을 이야기한다. 단순히 추출된 지식을 전달하는 것이 아닌, Teacher Network가 이미지를 통해 가중치를 학습시키면서 학습한 Generalization 능력 또한 전달하는 것이 가장 큰 목적이다.

가장 기초가 되는 II model[11]은 입력 이미지 x_i 에 stochastic augmentation을 적용하여 x'_i 에 하나의 이미지 x_i 에 2개의 output인 (z_i, \tilde{z}_i) 을 출력한다. z_i 는 y_i 와 cross-entropy를 계산하고, 2개의 output인 (z_i, \tilde{z}_i) 는 MSE를 계산한다. 이렇게 생성된 Supervised loss와 Unsupervised loss를 더함으로써 합한 loss를 최소화하는 과정이 II model의 연산 과정이다. II model은 입력 이미지에 stochastic augmentation과 dropout을 적용

하여 Consistency를 training 하는 데에 초점을 두었으나, 하나의 Network를 기반으로 동작하기 때문에 Noise가 심하다는 단점이 존재한다.

II model의 개선 버전인 Temporal ensembling[1]은 입력 이미지 x_i 에 stochastic augmentation과 dropout을 적용하는 과정은 동일하나, ensemble predictions인 Z 와, \tilde{z} 인 target vector를 추가로 사용한다. 이를 통해 bias correction을 진행하며, 과거 모델의 결과와 현재 모델 결과를 ensemble하게 되어 성능적으로 개선이 되었다. 하지만, 각 target은 epoch 당 한번 update되므로 학습이 느리다는 단점이 존재한다. 해당 단점을 극복하기 위해 Mean Teacher 기법이 제안되었다. 기존 지식 증류 기법과 가장 큰 다른 점은 teacher model, student model을 사용한다는 점이다. temporal ensembling과 다르게 Mean Teacher는 step마다 Teacher Network의 Weight를 업데이트 하므로, 각 Network 간의 빠른 피드백이 가능하다는 것이 가장 큰 장점이다.

III. Proposed Method

본 논문에서는 이미지상에서 나타나는 포트홀 유무를 분류하는 딥러닝 모델을 학습하기 위해 준지도학습(Semi-Supervised Learning)과 지도학습(Supervised Learning)을 이용한 다양한 CNN(Convolutional Neural Network) 아키텍처들의 결과를 비교하여 분석한다. 딥러닝의 가장 대표적인 방법론인 지도학습은 학습을 위한 데이터셋에 대해서는 높은 정확도를 보여주지만, 학습 과정에서 한 번도 학습되지 않은 테스트 데이터에 대해서는 정확도가 떨어지는 단점을 가지고 있다. 이는 지도학습을 이용한 딥러닝 모델들은 일반화의 오류를 가지고 있다는 것을 의미한다. 또한 지도학습 기법은 필연적으로 라벨링된 데이터(Labeled Data)가 요구되는데, 이를 확보하기 위해서는 라벨링(Labeling) 작업을 수행해야 한다. 하지만 라벨링 작업은 적용 분야에 따라 전문성과 많은 시간을 요구하며, 샘플수가 급증하게 되면 그에 따른 비용도 늘어난다. 포트홀의 경우 각 도로의 주변 상황이나 기상 상황에 따라 이미지에서 보이는 강도, 크기 등이 다르게 나타날 수 있다. 따라서 위에서 설명한 일반화의 오류를 최소화하기 위해 다양한 도로 환경과 기상 상황을 고려한 데이터셋 수집 및 라벨링 작업이 필수적이지만, 모든 상황을 고려한 라벨링 작업은 불가능하다. 따라서 소량의 Labeled Data와 레이블링이 안 된 데이터(Unlabeled data)를 함께 이

용하는 준지도학습을 이용한 CNN 모델들과 지도학습을 이용한 CNN 모델들을 비교하여 준지도학습이 포트홀 분류 일반화 문제에 끼치는 영향과 그에 따른 성능 차이를 GradCAM(Gradient-weighted Class Activation Map)을 이용하여 분석하여 효과를 입증한다.

1. Mean Teacher of Semi-supervised Learning

지도학습과 준지도학습의 포트홀 분류 모델을 비교하기 위해 Mean Teacher 모델을 이용하였다. Mean Teacher 모델은 동일한 아키텍처를 가진 두 개의 모델을 이용하는 것이 가장 큰 특징이다. Fig. 2는 Mean Teacher 모델의 전체적인 아키텍처를 보여준다.

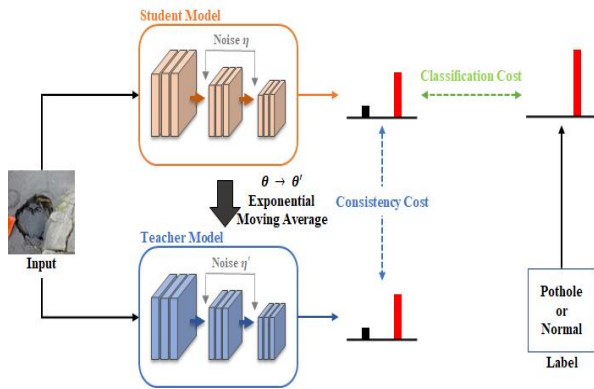


Fig. 2. Mean Teacher Architecture

Fig. 2에서 Classification cost와 Consistency cost는 각각 Labeled data와 Unlabeled data에 대한 오차값으로 오차역전파를 진행하기 위한 Loss function의 결과이다. Classification cost의 경우 일반적인 분류 문제에서 사용되는 Cross-Entropy 등을 이용한다. Consistency cost는 각 모델에서 출력된 결과의 차이를 의미하며, 이를 위해 사용되는 Loss function은 평균제곱오차 등이 있다. 아래 수식은 Consistency cost(J)를 표현한다.

$$J(\theta) = E_{x, \eta', \eta} [\| f(x, \theta', \eta') - f(x, \theta, \eta) \|^2] \quad (1)$$

수식 (1)에서 θ 와 θ' 는 각 모델의 가중치를 의미하며, η 와 η' 는 노이즈를 의미한다. 따라서 각 모델 결과의 차이를 구하여 오차역전파를 위한 최종 loss 값을 구하며, 대부분의 Mean Teacher 모델에서 Total loss를 구하기 위해 가중치 합(Weighted Sum)을 이용한다. 아래 수식은 Classification cost($L_{Classification}$)와 Consistency cost

($L_{Consistency}$)를 이용한 Total loss(L_T)를 표현한다.

$$L_T = L_{Classification} + (W * L_{Consistency}) \quad (2)$$

수식 (2)에서 가중치 W 는 Hyperparameters(하이퍼파라미터)의 개념으로 고정된 가중치를 사용하거나 Sigmoid ramp-up 등을 이용하며, 본 논문에서는 0.2의 고정된 가중치를 사용하여 실험을 진행하였다. 즉, Total loss를 이용하여 오차역전파를 이용한 학습을 진행하는데, 동일한 아키텍처와 가중치를 가진 모델 두 개를 이용한다면 Consistency cost의 의미가 사라질 것이다. 따라서 Total loss는 학생 모델만을 위해 사용되며, 교사 모델은 학생 모델의 가중치를 이동평균 함수 중 하나인 지수이동평균(EMA, Exponential Moving Average)를 이용하여 가중치를 업데이트한다. EMA는 일반적인 이동평균과는 달리 최근 시점에 높은 가중치를 주는 동시에 더 오래된 과거의 값도 낮은 영향력을 끼칠 수 있도록 고려하는 방법이다. 이를 Mean Teacher 모델의 교사 모델을 학습하기 위한 수식으로 나타내면 아래와 같다.

$$\theta'_t = \alpha \theta'_{t-1} + (1 - \alpha) \theta_t \quad (3)$$

수식 (3)에서 t 는 시점을 의미하며 θ 는 학생 모델의 가중치, θ' 는 교사 모델의 가중치를 의미한다. 또한 α 는 수식 (2)의 W 와 같이 하이퍼파라미터의 개념을 가지며, 본 논문에서는 학생 모델의 과거 가중치도 0이 아닌 낮은 수치의 영향력을 가질 수 있도록 0.95로 설정하여 실험을 진행하였다. 따라서 CNN 모델 14가지를 이용하여 Mean Teacher 모델을 이용한 포트홀 이미지 분류를 위한 준지도학습 모델들의 성능 평가 및 분석을 진행하였다. Table 1은 본 논문에서 사용된 Mean Teacher 모델의 하이퍼파라미터를 보여준다.

Table 1. Hyperparameters of Mean Teacher

Parameter	Value
Classification Loss Function	Cross-Entropy
Consistency Loss Function	Mean Squared Error
Weight for Total Loss	0.2
Weight for EMA	0.95

2. Pothole Dataset

포트홀을 탐지하기 위한 인공지능 이미지 기반 포트홀 탐지에서 가장 큰 문제점은 도로에 설치된 맨홀과 도시가스, 광케이블 등의 매설 지점을 표시해주는 라인 마크가

포트홀의 특징과 매우 유사하여 오탐지를 할 가능성이 있다는 것이다. 포트홀의 가장 큰 특징은 인접 도로의 색상과 다른 구멍이 생성되며, 도로 내 불특정 영역에 다수 존재 가능하다는 것이다. 이는 맨홀과 라인 마크에서도 동일하게 적용된다. 또한, 도로 보수 및 공사를 위해 페인트 등을 이용한 표시의 경우에도 CNN 모델들의 성능을 저하할 가능성이 존재한다. Fig. 3은 맨홀과 라인 마크, 페인트를 이용한 표시의 예시를 보여주며, 본 논문에서 학습 및 평가를 위한 데이터셋에 포함하여 실험을 진행하였다.

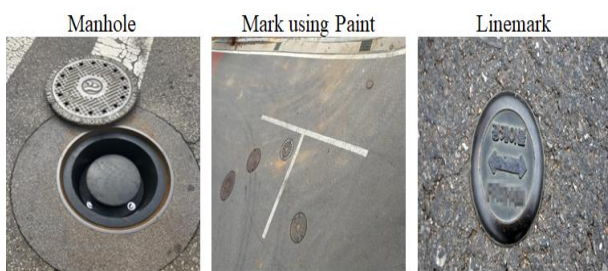


Fig. 3. Examples of Pothole Detection Obstacle on Road

대부분의 이미지 기반 딥러닝 분류 모델들은 카메라의 시점이나 주변 환경에 따라 인식률이 변화한다는 문제점을 가지고 있다. 따라서 최대한 다양한 환경과 시점을 학습하기 위해 인터넷에 공개된 이미지들과 직접 촬영한 사진들을 섞어 데이터셋을 구성하였다. 시점의 경우 크게 두 가지로 나누어지며, 도로 표면을 중점으로 일반 촬영 장비를 통해 수집된 Zoom In(줌인) 시점과 차량 대시보드 상단에 부착된 촬영 장치를 통해 수집된 Dashcam(블랙박스) 시점을 선택하였다. 이유는 이미지 기반 포트홀을 탐지하기 위한 모델을 시스템에 적용한다고 가정하였을 때, 가장 효율적으로 사용할 수 있는 입력 장치는 블랙박스 혹은 차량 앞쪽 범퍼에 장착된 전방 카메라일 것이다. 따라서 줌인 시점의 경우 포트홀이 포함된 도로 사진 600장과 일반 정상 도로 사진 600장을 수집하여 총 1,200장의 이미지를 가진 데이터셋을 구성하였으며, 블랙박스 시점은 포트홀이 포함된 도로 사진 900장과 일반 정상 도로 사진 900장을 수집하여 총 1,800장의 이미지를 가진 데이터셋을 구성하였다. Table 2는 데이터셋의 샘플 분포를 보여준다.

Table 2. Dataset Distribution

Class		Dataset Distribution			
		Training	Validation	Test	Total
Normal	Zoom in	360	120	120	600
	Dashcam	540	180	180	900
Pothole	Zoom in	360	120	120	600
	Dashcam	540	180	180	900

도로 위 맨홀, 라인 마크 등의 장애물은 존재하나 포트홀이 없는 도로는 Normal Class로 정의하였으며, Fig. 4는 Normal Class에 대한 예시를 보여준다. 또한 장애물과 상관없이 도로 위 포트홀이 존재하면 Pothole Class로 정의하였으며, Fig. 5는 Pothole Class에 대한 예시를 보여준다.



Fig. 4. Examples of Normal Class



Fig. 5. Examples of Pothole Class

데이터셋의 모든 이미지는 512 × 512의 크기로 변경하여 학습에 사용하였으며, 학습을 위해 1,800장의 이미지, 검증 및 테스트를 위해 각각 600장의 이미지를 사용하기 위해 총 3,000장의 이미지를 수집 및 분류하였다. 또한 학습을 위한 1,800장의 이미지 중 Labeled data 비율을 각각 20%, 30%, 50%, 100%로 나누어 지도학습과 준지도학습의 성능과 GradCAM을 통한 비교분석을 진행하였다. Table 3은 각 Label 비율에 따른 데이터셋 샘플 수의 분포를 보여준다.

Table 3. Number of Dataset samples by Label Ratio

Label Ratio (Labeled : Unlabeled)	Number of Dataset samples
20 : 80	<ul style="list-style-type: none"> • 360 samples w/ label • 1,440 samples w/o label
30 : 70	<ul style="list-style-type: none"> • 540 samples w/ label • 1,260 samples w/o label
50 : 50	<ul style="list-style-type: none"> • 900 samples w/ label • 900 samples w/o label
100 : 0	<ul style="list-style-type: none"> • 0 samples w/ label • 1,800 samples w/o label

3. Result of Supervised Learning Models

준지도학습 모델들과 성능 비교를 위해 먼저 지도학습을 이용한 CNN 모델들의 포트홀 분류 모델들의 학습 및 성능 평가를 진행하였다. 사용된 CNN 모델들은 AlexNet[12], DenseNet121[13], DenseNet169[13], EfficientNet[14], InceptionV3[15], MnasNet[16], MobileNet[17], RegNet[18], ResNet50[19], ResNet101[19], Resnext50[20], Resnext101[20], ShuffleNet[21], VGG16[22], VGG19[22]으로 총 15개의 모델을 이용하였으며, 모든 모델은 ImageNet을 이용한 사전 학습된 모델의 가중치를 초기 가중치로 설정하였다. Optimizer의 경우 Adam을 이용하였으며, 하이퍼파라미터의 경우 Grid Search 결과 lr(Learning Rate)은 0.0005가 가장 좋은 성능을 보여줬으며, Loss Function은 Cross Entropy를 사용하였다. 훈련은 배치사이즈 16 기준으로 최대 150회를 진행하였으며, Early Stopping을 이용하여 정확도 (Accuracy)를 기준으로 최대 성능을 내는 Checkpoint를 선정하였다. 또한, 정확도가 같은 점수를 가진 모델들의 경우 Loss와 F1 Score를 이용하여 순위를 나누었다. 실험을 위해 사용된 정확도와 F1 Score의 수식은 다음과 같다.

$$Accuracy = \frac{TP + TN}{TP + FN + FP + TN} \quad (4)$$

$$Precision = \frac{TP}{(TP + FP)} \quad (5)$$

$$Recall = \frac{TP}{(TP + FN)} \quad (6)$$

$$F1 - Score = 2 \times \frac{Precision \times Recall}{Precision + Recall} \quad (7)$$

수식 (4), (5), (6)에서 사용되는 TP, TN, FN, FP의 경우 혼동 행렬(Confusion Matrix)에서 사용되는 요소이며, 분류 문제에서 성능 평가를 위해 사용된다. Table 4부터 7까지 각 Labeled data 비율에 따른 지도학습을 이용한

CNN 모델들의 성능을 보여준다.

Table 4. Supervised Learning Result - (20:80)

Model	Accuracy	Precision	Recall	F1 Score
ShuffleNet	0.8183	0.8257	0.8183	0.8173
EfficientNet	0.8067	0.8401	0.8067	0.8018
MobileNet	0.7717	0.7719	0.7717	0.7716
DenseNet121	0.7217	0.7220	0.7217	0.7216
ResNext-50	0.7000	0.7000	0.7000	0.7000
ResNet-101	0.6767	0.6950	0.6767	0.6689
InceptionV3	0.6717	0.6908	0.6717	0.6632
ResNext-101	0.6367	0.6720	0.6367	0.6170
MnasNet	0.6333	0.6485	0.6333	0.6237
RegNet	0.5983	0.7062	0.5983	0.5379
DenseNet169	0.5767	0.5944	0.5767	0.5558
ResNet-50	0.5033	0.6258	0.5033	0.3436
AlexNet	0.5000	0.2500	0.5000	0.3333
VGG-19	0.5000	0.2500	0.5000	0.3333
VGG-16	0.5000	0.2500	0.5000	0.3333
Total	0.6410	0.6162	0.6410	0.5882

Labeled data 비율이 20%인 데이터셋을 이용하여 지도학습을 이용하였을 때 ShuffleNet이 정확도 기준 0.8183으로 가장 좋은 성능을 보여줬으며, EfficientNet이 0.8067을 기록하며 두 번째로 좋은 성능을 보여줬다. 평균 Accuracy는 0.6410으로 부족한 성능을 보여준다.

Table 5. Supervised Learning Result - (30:70)

Model	Accuracy	Precision	Recall	F1 Score
EfficientNet	0.8967	0.9007	0.8967	0.8964
ShuffleNet	0.8483	0.8655	0.8483	0.8465
MnasNet	0.8400	0.8514	0.8400	0.8387
DenseNet121	0.8383	0.8476	0.8383	0.8372
DenseNet169	0.8283	0.8297	0.8283	0.8282
ResNet-50	0.8200	0.8247	0.8200	0.8193
ResNet-101	0.8167	0.8176	0.8167	0.8165
MobileNet	0.7917	0.8419	0.7917	0.7837
InceptionV3	0.7900	0.7910	0.7900	0.7898
ResNext-101	0.7867	0.7871	0.7867	0.7866
ResNext-50	0.7733	0.7805	0.7733	0.7719
RegNet	0.6567	0.6915	0.6567	0.6403
AlexNet	0.5000	0.2500	0.5000	0.3333
VGG-16	0.5000	0.2500	0.5000	0.3333
VGG-19	0.5000	0.2500	0.5000	0.3333
Total	0.7458	0.7053	0.7458	0.7103

Labeled data 비율이 30%인 데이터셋을 이용하여 지도학습을 이용하였을 때 비율이 20%인 데이터셋과는 다르게 ShuffleNet보다 EfficientNet이 더 높은 성능을 보여주었다. EfficientNet은 정확도 기준으로 0.8967을 기록하였으며, Labeled data 비율을 20%만 사용한 ShuffleNet보다 약 0.08의 성능향상을 보여주었다.

Table 6. Supervised Learning Result - (50:50)

Model	Accuracy	Precision	Recall	F1 Score
ShuffleNet	0.9150	0.9179	0.9150	0.9149
ResNet-50	0.9050	0.9083	0.9050	0.9048
MobileNet	0.8767	0.8768	0.8767	0.8767
EffcientNet	0.8733	0.8889	0.8733	0.8721
ResNext-50	0.8583	0.8639	0.8583	0.8578
DenseNet169	0.8450	0.8545	0.8450	0.8440
ResNet-101	0.8300	0.8379	0.8300	0.8290
MnasNet	0.8267	0.8278	0.8267	0.8265
DenseNet121	0.8233	0.8481	0.8233	0.8201
RegNet	0.8200	0.8214	0.8200	0.8198
InceptionV3	0.8200	0.8221	0.8200	0.8197
AlexNet	0.7367	0.7372	0.7367	0.7365
ResNext-101	0.6500	0.6696	0.6500	0.6396
VGG-16	0.5000	0.2500	0.5000	0.3333
VGG-19	0.5000	0.2500	0.5000	0.3333
Total	0.7853	0.7583	0.7853	0.7619

Labeled data 비율이 50%인 데이터셋을 이용하여 지도 학습을 이용한 경우에는 비율이 20%인 데이터셋과 동일하게 ShuffleNet이 가장 높은 성능을 보여주었다. EfficientNet은 정확도 기준으로 0.8733을 기록하며 4번째로 높은 순위를 보여주었으며, ResNet50과, MobileNet이 각 0.9050, 0.8767을 기록하며 준수한 성능을 보여주었다.

Table 7. Supervised Learning Result - (100:0)

Model	Accuracy	Precision	Recall	F1 Score
EffcientNet	0.9667	0.9667	0.9667	0.9667
DenseNet121	0.9583	0.9592	0.9583	0.9583
ShuffleNet	0.9550	0.9577	0.955	0.9549
MobileNet	0.9517	0.9525	0.9517	0.9516
DenseNet169	0.9500	0.9502	0.9500	0.9500
ResNext-50	0.9500	0.9510	0.9500	0.9500
InceptionV3	0.9433	0.9436	0.9433	0.9433
ResNet-50	0.9383	0.9405	0.9383	0.9383
ResNext-101	0.9333	0.9340	0.9333	0.9333
MnasNet	0.9217	0.9268	0.9217	0.9214
ResNet-101	0.9133	0.9134	0.9133	0.9133
VGG-16	0.9017	0.9017	0.9017	0.9017
RegNet	0.8567	0.8577	0.8567	0.8566
AlexNet	0.7117	0.7137	0.7117	0.7110
VGG-19	0.4933	0.2483	0.4933	0.3304
Total	0.8896	0.8744	0.8896	0.8787

Labeled data 비율이 100%인 데이터셋을 이용하여 지도 학습을 이용하였을 때 비율이 30%인 데이터셋과 동일하게 EfficientNet이 가장 높은 성능을 보여주었다. EfficientNet은 0.9667의 성능을 보여주며 지도 학습을 이용한 모델 중 가장 높은 정확도를 달성하였다.

4. Results of Semi-Supervised Learning Models

이미지를 이용한 포트홀 분류 모델들의 지도 학습 모델들과 준지도 학습 모델들의 성능 비교를 위해 Mean

Teacher 모델 기반 CNN 모델들의 학습 및 성능 평가를 진행하였다. 사용된 CNN 모델들과 하이퍼파라미터, 초기 가중치 등의 환경은 지도 학습의 실험 환경과 동일하며, Table 8부터 11까지 각 Labeled data 비율에 따른 준지도 학습을 이용한 CNN 모델들의 성능을 보여준다.

Table 8. Semi-Supervised Learning Result - (20:80)

Model	Accuracy	Precision	Recall	F1 Score
EffcientNet	0.9133	0.9136	0.9133	0.9133
MobileNet	0.8650	0.8680	0.8650	0.8647
InceptionV3	0.8333	0.8351	0.8333	0.8331
ResNet-101	0.8333	0.8437	0.8333	0.8321
ResNext-50	0.8317	0.8374	0.8317	0.8310
ResNext-101	0.8217	0.8225	0.8217	0.8216
DenseNet121	0.8167	0.8328	0.8167	0.8144
ResNet-50	0.8100	0.8107	0.8100	0.8099
DenseNet169	0.7967	0.7989	0.7967	0.7963
ShuffleNet	0.7800	0.7828	0.7800	0.7794
RegNet	0.6883	0.6982	0.6883	0.6844
MnasNet	0.5083	0.7521	0.5083	0.3516
AlexNet	0.5000	0.2500	0.5000	0.3333
VGG-19	0.5000	0.2500	0.5000	0.3333
VGG-16	0.5000	0.2500	0.5000	0.3333
Total	0.7332	0.7031	0.7332	0.6888

Labeled data 비율을 20%만 이용하여 준지도 학습을 이용하였을 경우 정확도 기준 EfficientNet이 0.9133를 기록하며 가장 높은 성능을 기록하였다. 또한 지도 학습을 이용한 EfficientNet과 비교하였을 경우 정확도 기준 약 0.11이 더 높았으며, 평균 정확도는 약 0.09의 큰 폭의 성능 향상을 보여주었다.

Table 9. Semi-Supervised Learning Result - (30:70)

Model	Accuracy	Precision	Recall	F1 Score
EffcientNet	0.9150	0.9207	0.9150	0.9147
MobileNet	0.8883	0.8887	0.8883	0.8883
ResNext-50	0.8750	0.8750	0.8750	0.8750
DenseNet169	0.8700	0.8700	0.8700	0.8700
InceptionV3	0.8550	0.8593	0.8550	0.8546
ResNet-50	0.8500	0.8526	0.8500	0.8497
ResNet-101	0.8433	0.8433	0.8433	0.8433
ResNext-101	0.8400	0.8404	0.8400	0.8400
DenseNet121	0.8300	0.8348	0.8300	0.8294
ShuffleNet	0.8267	0.8320	0.8267	0.8260
MnasNet	0.7883	0.8466	0.7883	0.7790
RegNet	0.7833	0.7885	0.7833	0.7824
VGG-19	0.5000	0.2500	0.5000	0.3333
AlexNet	0.5000	0.2500	0.5000	0.3333
VGG-16	0.5000	0.2500	0.5000	0.3333
Total	0.7777	0.7335	0.7777	0.7435

Labeled data 비율이 30%인 데이터셋을 이용한 준지도 학습의 경우 지도 학습보다 평균 정확도 기준 약 0.03의 성능 향상을 보여주었다. 또한 EfficientNet이 0.9150의 성

능을 보여주며 가장 높은 성능을 보여주었다.

Table 10. Semi-Supervised Learning Result - (50:50)

Model	Accuracy	Precision	Recall	F1 Score
EfficientNet	0.9117	0.9119	0.9117	0.9117
InceptionV3	0.8833	0.8883	0.8833	0.8830
ResNext-50	0.8817	0.8832	0.8817	0.8815
DenseNet121	0.8800	0.8800	0.8800	0.8800
MobileNet	0.8783	0.8785	0.8783	0.8783
ResNext-101	0.8750	0.8759	0.8750	0.8749
DenseNet169	0.8650	0.8742	0.8650	0.8642
ShuffleNet	0.8367	0.8479	0.8367	0.8353
ResNet-101	0.8367	0.8479	0.8367	0.8353
ResNet-50	0.8267	0.8291	0.8267	0.8263
RegNet	0.7950	0.7974	0.7950	0.7946
VGG-16	0.7217	0.7228	0.7217	0.7213
MnasNet	0.6717	0.7633	0.6717	0.6404
AlexNet	0.5283	0.5754	0.5283	0.4411
VGG-19	0.5000	0.2500	0.5000	0.3333
Total	0.7928	0.7884	0.7928	0.7734

Labeled data 비율이 50%인 데이터셋을 이용하였을 때 평균 정확도는 0.7928을 보여준다. 지도학습 모델보다 약 0.007의 성능향상을 보여주며, 라벨링 비율이 높아질수록 성능향상 폭이 작아지는 것을 확인할 수 있었다. 또한 지도학습의 경우 ShuffleNet과 EfficientNet이 각 2번씩 최고성능을 달성하였으나 준지도학습의 경우 EfficientNet이 모든 비율에서 최고성능을 달성하였으며, ShuffleNet의 경우 모든 비율에서 높은 성능을 달성하지 못하였다.

Table 11. Semi-Supervised Learning Result - (100:0)

Model	Accuracy	Precision	Recall	F1 Score
EfficientNet	0.9683	0.9692	0.9683	0.9683
ShuffleNet	0.9667	0.9672	0.9667	0.9667
MnasNet	0.9567	0.9568	0.9567	0.9567
MobileNet	0.9567	0.9568	0.9567	0.9567
InceptionV3	0.9483	0.9483	0.9483	0.9483
DenseNet121	0.9450	0.9454	0.9450	0.9450
DenseNet169	0.9433	0.9434	0.9433	0.9433
ResNet-101	0.9400	0.9407	0.9400	0.9400
ResNext-101	0.9383	0.9384	0.9383	0.9383
ResNext-50	0.9350	0.9376	0.9350	0.9349
ResNet-50	0.9183	0.9197	0.9183	0.9183
AlexNet	0.9050	0.9052	0.9050	0.9050
VGG-19	0.8833	0.8834	0.8833	0.8833
RegNet	0.8567	0.8572	0.8567	0.8566
VGG-16	0.5000	0.2500	0.5000	0.3333
Total	0.9041	0.8879	0.9041	0.8929

Labeled data 비율이 100%인 데이터셋을 이용한 경우 평균 정확도는 0.9040의 성능을 보여주며, 최고성능인 EfficientNet은 0.9667의 성능을 보여주었다. 지도학습과 비교하면 평균 정확도 기준 0.02의 성능향상을 보여주었다.

5. Qualitative Comparison based on GradCAM

본 섹션에서는 GradCAM을 이용하여 지도학습과 준지도학습 모델들을 시각화하여 분석한다. GradCAM은 CNN에서 만든 최종 예측에 가장 많이 기여하는 이미지의 영역을 이해하기 위해 딥러닝에 사용되는 시각화 기술로, 마지막 컨볼루션 레이어의 활성화와 관련하여 대상 클래스 점수의 기울기를 계산한 다음 이러한 활성화에 가중치를 두어 거친 지역화 맵을 생성하는 방식으로 작동한다. 즉, GradCAM에서 생성된 히트맵을 사용하여 CNN 모델이 이미지에서 무엇을 보고 있는지 이해하고 예측을 위해 올바른 기능에 초점을 맞추고 있는지 확인할 수 있으며, 이 정보는 모델의 정확도를 개선하고 보다 강력한 모델을 개발하는 데 사용할 수 있다.



Fig. 6. GradCAM Application Examples

Fig. 6은 GradCAM에 적용된 이미지들을 보여주며, 빨간색 사각형으로 표현된 구역이 포트홀을 나타낸다. 즉, 사각형 내 포트홀의 모양과 위치가 강조되는 히트맵(Heatmap)을 출력할수록 개발자의 의도에 따라 학습됨을 확인할 수 있다. Figs. 7과 8은 각 학습 기법을 통해 훈련된 모델 중 가장 높은 성능을 나타낸 EfficientNet에 GradCAM을 적용하여 시각화된 가중치 정보를 보여준다.

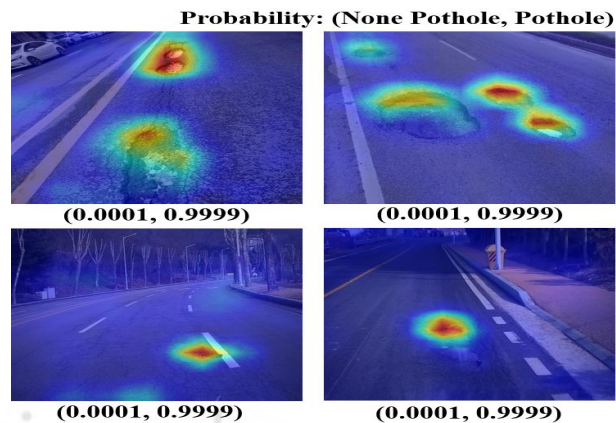


Fig. 7. GradCAM Results on Best Model of Supervised Learning

Fig. 7은 지도학습을 통해 학습된 CNN 모델 중 가장 높은 성능을 보였던 EfficientNet의 GradCAM 결과를 보여 준다. 또한 각 결과 아래에 괄호로 표기된 숫자는 (포트홀이 아닐 확률, 포트홀일 확률)로서, GradCAM 테스트에 사용되는 모든 사진은 포트홀이 포함된 이미지이다. 따라서 괄호 기준 오른쪽의 숫자가 더 높아야 함이 정상적으로 혼련이 되었음을 표현한다. 따라서 지도학습을 이용하였을 때, 포트홀 분류성능이 우수함을 알 수 있지만, GradCAM 결과에서는 포트홀의 전체적인 부분을 탐지하지 못한다. 또한 노이즈처럼 포트홀이 아닌 부분을 강조하는 것을 볼 수 있으며, 이는 학습된 데이터와 다른 환경의 이미지가 입력될 때 성능이 급하락하는 일반화 오류가 발생할 수 있다.

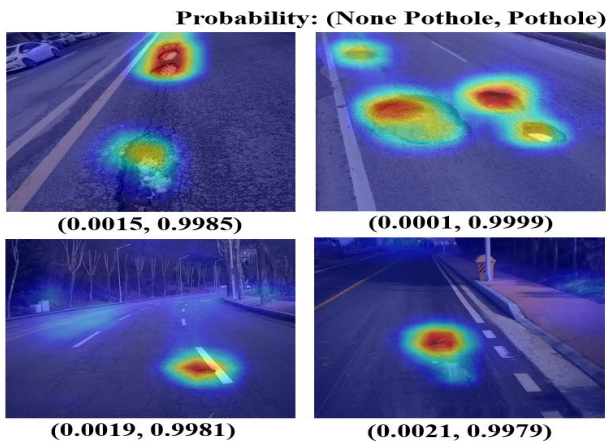


Fig. 8. GradCAM Results on Best Model of Mean Teacher

Fig. 8은 Mean Teacher 모델을 이용하여 준지도학습을 진행한 CNN 모델 중 가장 높은 성능을 보였던 EfficientNet의 GradCAM 결과를 보여준다. 지도학습 모델과 비교하였을 때, 포트홀 영역을 더 정확하고 넓게 표현한다. 또한 지도학습과 비교해 노이즈가 감소하였음을 보여준다. 반면, 포트홀의 확률이 지도학습에 비해 낮은 것을 확인할 수 있다. 이는 성능이 더 낮음을 의미하는 것이 아닌 베이지안 모델의 특성상 추출된 포트홀의 특징들이 학습된 포트홀의 특징들과 얼마나 유사한지를 보여주며, 수치가 낮음에도 더 강인한 성능을 나타냄을 보여준다.

IV. Conclusions

도로 위에 예측할 수 없는 위치에 불규칙하게 발생하는 포트홀을 탐지할 수 있는 강인한 딥러닝 모델 설계를 목표로 지도학습과 준지도학습을 이용하여 15개의 CNN 모델

들의 학습을 진행하고 성능 평가 및 GradCAM을 통한 비교를 진행하였다. Mean Teacher 모델을 활용해 준지도 학습을 한 결과 Accuracy를 기준으로 평균 90.41%를 달성하며, 지도학습과 비교해 평균 정확도가 최소 2%에서 최대 9%의 차이를 보이며 강인한 성능을 나타냄을 확인하였다. 이는 Mean Teacher 모델을 이용해 기존 CNN 모델을 학습하면 포트홀 분류 문제에서 성능 향상의 효과가 있음을 보이며, 동시에 적은 데이터셋으로도 EfficientNet 기준 91.33%의 정확도를 확보할 수 있다는 것을 입증한다. 또한, GradCAM 결과에 따라 지도학습에 비해 준지도 학습을 통해 학습한 모델이 더 높은 신뢰도를 보여줌을 확인하였다. 따라서 준지도학습이 포트홀 분류 문제에 적합함과 효과적임을 입증하였으나, 여전히 다양한 환경에서의 데이터셋 구축은 한계점을 보이며, 옛지 디바이스에서의 적용이 가능하기 위해선 모델의 경량화가 필수적이다. 따라서 위 조건을 충족시키기 위해 Semi-Supervised Learning based Mean Teacher Student Knowledge Distillation 기법 외 다양한 기법을 적용하여 적은 수의 Labeled data로도 강인한 성능을 낼 수 있는 딥러닝 기반 포트홀 분류 및 탐지의 연구는 지속되어야 한다. 본 논문에서 실험에 사용된 방법의 성능은 우수하였지만, 1) Label Ratio가 점차 적어질수록 GradCAM의 실제 Localization의 위치가 비 정확하며, 2) 이미지의 포트홀 탐지와 XAI(eXplainable Artificial Intelligence)를 위한 GradCAM 연산량과 연산 시간이 다소 소요된다는 단점이 존재한다. 또한, 포트홀 탐지는 촬영되는 각도와 장소에 따라 주위 환경(명암, 채도, 그림자 등)에 영향을 받는다. 따라서 연구 지속 필요성과 실험 모델 단점의 보완을 위하여 다양한 각도와 환경에서도 국한되지 않고 강인한 성능을 보유하는 딥러닝 모델을 설계할 계획이다. 그에 따라, 데이터셋의 샘플 수를 늘려 Transformer 기반 딥러닝 모델로 확장 설계하거나, Self-Supervised Learning 기법으로 확장하여 더 큰 딥러닝 모델을 더 작은 데이터셋 샘플 수로 학습하여 더욱 강인한 성능을 보이는 모델 혹은 기법을 개발할 계획이다. 또한, 실제 주행 중인 자동차의 블랙박스나 옛지 디바이스에 적용할 수 있는 경량화 기법이 적용된 딥러닝 모델을 이용하여 실시간으로 포트홀을 탐지하고 분류해낼 수 있는 기법을 설계할 계획이다. 마지막으로, 현실세계에서는 포트홀이 없는 정상 클래스가 대부분의 비중을 차지하고 있으므로, 테스트 상황에서는 불균형이 심한 상태에서의 성능을 측정하고, 이를 개선할 예정이다.

ACKNOWLEDGEMENT

This results was supported by "Regional Innovation Strategy (RIS)" through the National Research Foundation of Korea (NRF) funded by the Ministry of Education (MOE) (2021RIS-001) and Basic Science Research Program through the National Research Foundation of Korea(NRF) funded by the Ministry of Education(Grant No. NRF-2020R111A3074141)

REFERENCES

- [1] A. Tarvainen, H. Valpola, "Mean Teachers Are Better Role Models: Weight-Averaged Consistency Targets Improve Semi-Supervised Deep Learning Results," *Advances in Neural Information Processing System (NIPS 2017)*, Vol. 30, March, 2017, DOI:<http://doi.org/10.48550/arXiv.1703.01780>
- [2] R. R. Selvaraju, M. Cogswell, A. Das, R. Vedantam, D. Parkih and D. Batra, "Grad-CAM: Visual Explanations from Deep Networks Via Gradient-Based Localization," *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, pp. 618-626, March, 2017 DOI:10.1007/s11263-019-01228-7
- [3] K. K. Vupparaboina, R. R. Tamboli, P. M. Shenu, and S. Jana, "Laser-Based Detection and Depth Estimation of Dry and Water-Filled Potholes: A Geometric Approach," *Twenty First National Conference on Communications (NCC)* pp. 1-6, March, 2015 DOI:10.1109/NCC.2015.7084929
- [4] X. Yu, E. Salari, "Pavement Pothole Detection And Severity Measurement Using Laser Imaging," *IEEE International Conference on Electro/Information Technology(EIT)*, Mankato, pp. 1-5, August, 2011, DOI:10.1109/EIT.2011.5978573
- [5] S. Nienaber, M. J. Booyesen, and R. S. Kroon, "Detecting Potholes Using Simple Image Processing Techniques and Real-World Footage," *Proceedings of the 34th Southern African Transport Conference (SATC 2015)*, pp. 6-9, July, 2015, DOI:10019.1/97191
- [6] K. Vigneshwar, B. H. Kumar, "Detection and Counting of Pothole Using Image Processing Techniques," *IEEE International Conference on Computational Intelligence and Computing Research (ICCIC)*, pp. 1-4, December, 2016, DOI:10.1109/ICCIC.2016.7919622
- [7] A. A. Shaghouri, R. Alkhatib, and S. Berjaoui, "Real-Time Pothole Detection Using Deep Learning," *arXiv preprint arXiv:2107.06356*, July, 2021, DOI:10.48550/arXiv.2107.06356
- [8] M. H. Asad, S. Khaliq, M. H. Yousaf, M. O. Ullah, and A. Ahmad, "Pothole Detection Using Deep Learning: A Real-Time and AI-on-the-Edge Perspective," *Advances in Civil Engineering*, April, 2022 DOI:10.1155/2022/9221211
- [9] N. Çınar, M. Kaya, "An automated pothole detection via transfer learning," *2022 International Conference on Decision Aid Sciences and Applications (DASA)*, pp. 1355-1358, March, 2022, doi: 10.1109/DASA54658.2022.9765021.
- [10] I. K. Kim, B. J. Park, B. J. Kim, and J.H. Gwak, "Pothole Detection Method based on an Ensemble of Pretrained Deep Learning Models," *2022 Korean Society for Internet Information (KSII) SPRING Conference*, 2022
- [11] S. Laine, and T. Aila, "Temporal Ensembling for Semi-Supervised Learning," *arXiv preprint arXiv:1610.02242*, March, 2016, DOI:10.48550/arXiv.1610.02242
- [12] A. Krizhevsky, I. Sutskever, and G. Hinton, "ImageNet Classification with Deep Convolutional Neural Networks," (2012) In *Proceedings of the 25th International Conference on Neural Information Processing Systems(NIPS)*, Vol.60, No.6, pp. 84-90, May, 2017, DOI: 10.1145/3065386
- [13] G. Huang, Z. Liu, and Q. Killan, "Densely Connected Convolutional Networks," In *Proceedings of the IEEE conference on computer vision and pattern recognition(CVPR)*, p.4700-4708, August, 2017, DOI: 10.1109/cvpr.2017.243
- [14] M. Tan, V. L. Quoc, "EfficientNet: Rethinking Model Scaling for Convolutional Neural Networks," In *International conference on machine learning(ICML)*, pp. 6105-6114, May, 2019, DOI:10.48550/arXiv.1905.11946
- [15] C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens, and Z. Wojna, "Rethinking the Inception Architecture for Computer Vision," *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 2818-2826, December, 2016, DOI: 10.1109/CVPR.2016.308.
- [16] M. Tan, B. Chen, R. Pang, V. Vasudevan, M. Sandler, A. Howard, and V. L. Quoc "MnasNet : Platform-Aware Neural Architecture Search for Mobile," In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition(CVPR)*, pp. 2820-2828, June, 2019, DOI:10.1109/CVPR.2019.00293
- [17] G. Andrew, M. Zhu, B. Chen, D. Kalenichenko, W. Wang, T. Weyand, M. Andreetto, and H. Adam, "MobileNets: Efficient Convolutional Neural Networks for Mobile Vision Applications," *arXiv preprint*, April, 2017, DOI:10.48550/arXiv.1704.04861
- [18] J. Xu, Y. Pan, S. Pan, S. Hoi, Z. Yi, and Z. Xu, "RegNet: Self-Regulated Network for Image Classification," *IEEE Transactions on Neural Networks and Learning Systems*, pp. 1-6, March, 2022, DOI:10.1109/TNNLS.2022.3158966
- [19] K. He, X. Zhang, S. Ren, and J. Sun "Deep Residual Learning for Image Recognition," In *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 770-778, June, 2016, DOI: 10.1109/CVPR.2016.90
- [20] S. Xie, R. Girshick, P. Dollar, Z. Tu, and K. He "Aggregated

Residual Transformations for Deep Neural Networks," In Proceedings of the IEEE conference on computer vision and pattern recognition(CVPR), pp. 1492-1500, July, 2017, DOI:10.1109/CVPR.2017.634

- [21] X. Zhang, X. Zhou, M. Lin, and J. Sun, "ShuffleNet: An Extremely Efficient Convolutional Neural Network for Mobile Devices," In Proceedings of the IEEE conference on computer vision and pattern recognition(CVPR), pp. 6848-6856, June, 2018, DOI:10.1109/CVPR.2018.00716
- [22] K. Simonyan, and A. Zisserman, "Very Deep Convolutional Networks for Large-Scale Image Recognition," arXiv preprint arXiv:1409.1556, September 2014, DOI:10.48550/arXiv.1409.1556

Authors



Inki Kim received his M.S. degree in Computer Engineering from Korea National University of Transportation (KNUT) in 2023. Since then, he has been pursuing his Ph.D. and conducting research in the field of Deep Learning.

Inki Kim obtained his B.S. degree in Computer Engineering from Korea National University of Transportation (KNUT) in 2020. Since 2023, he has been actively engaged as a Ph.D. student and researcher at the Artificial Machine Intelligence Lab in KNUT. His research interests encompass various areas, including deep learning, time-series-based anomaly detection, and machine learning and deep learning-based segmentation techniques.



Beomjun Kim obtained his B.S. degree in Computer Engineering from Korea National University of Transportation (KNUT) in 2022. Since then, he has been pursuing his Master's degree and conducting research in the field of Deep Learning.

Beomjun Kim enrolled in the Master's Degree program in the Department of IT-Energy Convergence at Korea National University of Transportation, Chungju, Korea, in 2021. He is currently an active researcher at the Algorithmic Machine Intelligence Lab in KNUT. His research interests primarily revolve around deep learning, computer vision, anomaly detection, and defect segmentation.



Jeonghwan Gwak received the Ph.D. degree in machine learning and artificial intelligence from Gwangju Institute of Science and Technology (GIST), Gwangju, Korea in 2014.

From 2002 to 2007, he had worked for several companies and research institutes as a researcher and chief technician. From 2014 to 2016, he worked as a Postdoctoral Researcher at the GIST, and from 2016 to 2017, as a Research Professor. From 2017 to 2019, he was a Research Professor with the Biomedical Research Institute and the Department of Radiology, Seoul National University Hospital, Seoul, South Korea. Since 2019, he has been with the Korea National University of Transportation (KNUT), Chungju, Republic of Korea, as an Assistant Professor, where he has been an Associate Professor, since 2021. He is the Director of the Algorithmic Machine Intelligence Laboratory. His current research interests include deep learning, computer vision, image and video processing, AIoT, fuzzy sets and systems, evolutionary algorithms, optimization, and relevant applications of medical and visual surveillance systems.