

## 차량 정보 인식 및 추적 기법을 이용한 인공지능 학습용 교통정보 데이터 자동정제

김 대 진<sup>1</sup> · 김 준 화<sup>2\*</sup><sup>1</sup>동국대학교 영상문화콘텐츠연구원 조교수<sup>2</sup>건양대학교 인공지능학과 조교수

# Automatic Refinement Traffic Data for Artificial Intelligence Training Using Vehicle Information Recognition and Tracking Techniques

Dae-Jin Kim<sup>1</sup> · Jun-Hwa Kim<sup>2\*</sup><sup>1</sup>Assistant Professor, Research Institute for Image & Cultural Contents, Dongguk University, Seoul 04626, Korea<sup>2</sup>Assistant Professor, Department of Artificial Intelligence, Konyang University, Daejeon 35365, Korea

### [요 약]

인공지능을 이용한 서비스들이 우리 생활에 밀접하게 다가선 이 시점에서, 양질의 데이터 수집은 중요한 요소가 되었다. 특히, 교통데이터를 이용한 인공지능 학습모델들이 내비게이션, 교통정보, 자율주행, 교통사고 예방, 범죄차량추적 등 다양하게 활용될 수 있으며, 성능고도화를 위해서는 양질의 학습데이터 확보와 효율적인 정제방법이 필수적이다. 따라서 본 논문에서는 인공지능 학습용 교통정보 데이터의 자동정제 방법을 위한 연구를 진행하였다. YOLOv5 기반의 객체탐지, IOU기반의 객체 추적, EfficientNet을 이용한 차종인식, 가변임계치를 이용한 LPR 병렬처리를 활용하여 데이터 정제를 수행하였으며, 수작업으로 했을 때보다 시간 및 비용 절감을 가져올 수 있었다. 또한, 차량 정보 인식 및 추적기법의 다양한 정제 방법을 통해서 더욱 효과적인 양질의 데이터 정제가 되었음을 확인하였다.

### [Abstract]

Artificial intelligence services are currently becoming increasingly integrated into our lives. Hence, the collection of quality data is becoming crucial. In particular, artificial intelligence learning models that use traffic data can be employed in various fields, including navigation, traffic information, autonomous driving, traffic accident prevention, and criminal vehicle tracking. Hence, collecting high-quality learning data and developing efficient refinement methods to improve performance are essential. Therefore, this paper proposes an automatic refinement method of traffic information data for artificial intelligence learning. We used YOLOv5 object detection, IOU tracking, vehicle classification, and variable thresholds in a parallel algorithm for license plate recognition to perform data refinement and demonstrated savings in terms of both time and cost, compared with manual work. The results confirm that various refinement methods of vehicle information recognition and tracking methods are more effective for quality data refinement.

**색인어** : 자동정제, 객체탐지, 객체추적, 객체식별, 차량번호인식**Keyword** : Automatic Refinement, Object Detection, Object Tracking, Object Classification, License Plate Recognition<http://dx.doi.org/10.9728/dcs.2025.26.3.677>

This is an Open Access article distributed under the terms of the Creative Commons Attribution Non-Commercial License (<http://creativecommons.org/licenses/by-nc/3.0/>) which permits unrestricted non-commercial use, distribution, and reproduction in any medium, provided the original work is properly cited.

Received 09 January 2025; Revised 31 January 2025

Accepted 12 February 2025

**\*Corresponding Author, Jun-Hwa Kim**

Tel: +82-42-600-8508

E-mail: junhwakim@konyang.ac.kr

## I. 서론

최근 들어 인공지능은 혁신 기술로써 과학기술발달을 이끌어 가고 있다. 인공지능의 발달은 산업 전반에 영향을 미치게 되었고, 이중 가장 중요한 요소는 데이터에 있다. 데이터 학습을 통해서 인공지능 모델의 성능이 좌우되고, 비즈니스 모델에 맞추어서 인공지능 데이터가 정확하게 수집되어야 인공지능 학습모델을 구동시킬 때 신뢰성 있는 학습 결과를 확보할 수 있다. 특히 교통데이터의 경우 내비게이션, 교통정보, 자율주행, 교통사고 예방, 범죄차량추적 등 인공지능 기술을 이용하여 다양하게 활용될 수 있으며, 학습모델에 가장 중요한 요소인 고품질 학습데이터 획득 및 효율적인 정제방법이 필수적이다[1],[2]. 그러나, 데이터 정제작업을 진행하는 데 있어 AI 학습을 위한 필요목적이 항상 같지 않고, 비즈니스 모델이나 연구의 방향성에 따라서 똑같은 이미지라고 해도 다른 항목으로 정제해야 하는 경우가 대다수이다. 또한, 차량, 표지판, 신호등을 중심으로 라벨링이 진행되었으나, 같은 이미지에서 차량 제조사별로 라벨링(Labeling)을 할 수도 있고, 이륜차와 사륜차로 구분해서 라벨링을 진행할 수도 있다[14]. 본 논문은 환경, 계절, 침두/비침두, 도로 유형들에 따른 교통정보를 구분하기 위해서 차량 번호판, 외관 인식을 위한 사전 자동정제작업을 목적으로 한다. 이와 같은 이유에서 자동정제작업을 하는 대부분 논문들은 공통점을 찾기가 쉽지 않다.

교통데이터의 경우 차량과 같은 이동수단의 정보가 필수적이고 이동수단의 고유성을 나타내는 정보는 차량 번호판이다. 따라서, 차량번호인식(LPR:License Plate Recognition), 객체추적(Object Tracking), 차종인식(Vehicle Classification)의 인공지능 기술을 적용하여 데이터 정제가 사전에 이루어진다면, 원시데이터 수집 후 수작업으로 정제가 이루어지는 과정을 간소화 할 수 있고, 표준화된 절차를 통해서 범용적으로 활용할 수 있는 데이터셋 구축 프로세스를 구축할 수 있다. 본 논문에서는 인공지능 학습용 교통정보 데이터 구축 시 정제과정에서 효율성 증대를 위해 LPR 병렬처리, 객체추적, 차종인식을 통해서 데이터를 자동정제할 수 있는 시스템을 연구해 본다. 본 논문의 구성은 2장에서 인공지능 학습용 교통정보 데이터를 정제하기 위한 기존 사례 및 문제점을 제시하고 3장에서는 인공지능 학습용 교통정보 데이터 정제를 위한 인공지능 알고리즘을 알아보고, 4장에서는 정제 이미지에 대한 종류 및 기준을 설정하고 제안한 알고리즘을 기반으로 구현해 보며, 주어진 환경에서의 성능 비교를 하였고, 5장에서는 결론을 지었다.

## II. 기존 사례 연구

일반적인 인공지능 학습용 데이터 구축 단계를 보면 그림 1과 같다. 이 과정에서 데이터 획득 및 정제과정은 대부분 사

람의 수작업 때문에 대부분 처리되고 있다. 학습용 데이터 중 교통정보 데이터는 수작업에 의한 시간 및 비용 소비가 많으므로 차량번호 인식을 이용하여 교통정보 데이터 획득 및 정제를 진행하고 있다[3].

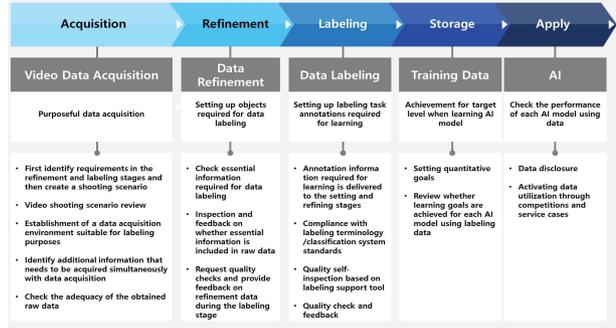


그림 1. 인공지능 학습용 데이터 구축 단계[1]

Fig. 1. Build data for AI learning

### 2-1 정제 시스템에 LPG 적용

그림 2에서는 교통정보 데이터의 정제를 위해서 차량번호 인식을 이용한 정제 프로세스를 보여주고 그 세부 내용은 아래와 같다.

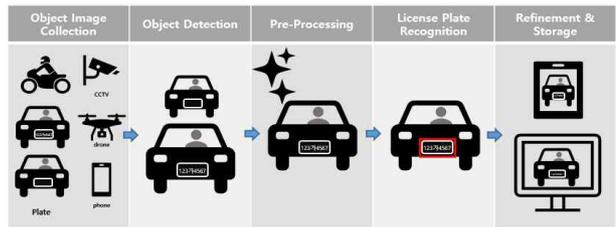


그림 2. 차량번호 인식을 이용한 교통정보 데이터의 정제

Fig. 2. Refinement of traffic information data using LPR

#### 1) 교통정보 데이터 획득

CCTV로부터 교차로, 이면도로 등의 장소로부터 데이터를 획득하게 된다. 이때 자동차의 종류, 날씨(우천, 안개, 일출/일몰), 주/야간, 교통량(침두/비침두)등 다양한 조건들에 맞추어서 데이터 획득을 진행한다.

#### 2) 차량번호 인식을 위한 사전이미지 처리[3]

교통데이터의 경우 수작업에 의한 시간 및 비용을 줄이기 위해서 차량번호 인식을 진행하고, 이때 번호 인식의 정확도를 높이기 위해서 이미지 처리를 진행하였다. 많은 논문에서 성능개선을 위한 아래와 같은 방안들이 진행되었다.

##### • 화질 개선

번호판의 글씨가 잘 보일 수 있도록 주/야간에 상관없이 일정한 밝기를 유지하는 것이 중요하다. 특히 주간의 경우 햇살

이 강하게 비추고, 각도에 의해서 햇빛이 직접적으로 번호판에 비치는 경우 번호판에 백화 현상이 일어나 글씨를 잘 알아볼 수 없는 경우가 발생한다. 이 부분은 카메라 내 노출(Exposure) 교정을 이용하여 해결한다. 따라서, 현장마다 주변 환경의 상황에 맞추어서 노출 정도를 가변으로 조절할 수 있다.

• **색상 개선**

교통데이터의 차량 번호판 인식의 경우 대부분 흑백 이미지로 처리한다. 이는 밤 또는 날씨가 흐린 경우에도 똑같이 적용되기 때문에 흑백 이미지를 사용하여 인식 처리를 진행하는 것이 일반적이다. 또한, 컬러 카메라 사용 시 비용이 증가 때문에 흑백 카메라를 사용하여 입력받는다. 그러나, 최근 전기자동차와 같이 컬러 정보를 가지고 있는 번호판들이 많아지면서 번호판 인식 시 중요 정보로 사용할 수 있다. 따라서 중간에는 컬러 모드를, 야간 또는 어두운 날에는 흑백 모드를 사용하여 인식률을 개선할 수 있다. 주/야간 판정은 히스토그램(Histogram) 정보를 이용하여 판정할 수 있고, 중간에 컬러 이미지를 번호 인식하는 경우 RGB나 HSV 이미지 포맷을 사용하여 인식한다.

• **노이즈 개선**

획득된 CCTV 교통데이터를 보면 번호판에 이물질이 묻어 있는 경우를 많이 본다. 특히 비/눈이 내리거나, 공사현장에 출입하는 경우 이런 현상이 더욱 많이 발생한다. 이는 번호판 인식률을 떨어뜨리는 결과를 가져온다. 이 경우 고주파(High-pass), 저주파(Low-pass), 미디언(Median-pass), 가우시안 필터링(Gaussian-filtering) 등을 통해서 번호들이 더 잘 보이도록 개선한다.

• **이미지 변환**

CCTV의 경우 카메라가 설치된 각도에 따라 차량의 번호 각도가 다르게 보인다. 교통데이터의 차량번호 인식의 경우는 교차로나 이면도로의 특성에 맞추어서 CCTV가 설치되기 때문에 주차 관제용 차량번호 인식보다 다각도의 번호판 이미지획득이 필요하다. 이러한 문제를 해결하기 위해서 다각도 번호판 이미지에서 전처리 보정 없이 번호 추출 후 학습에 사용하는 방법이 있고, 또는 위치 옮김(Translation)과 회전(Rotation)을 사용한 기준 이미지로 학습하여 사용하기도 한다.

**3) 차량 번호판 추출**

카메라로부터 습득한 입력 이미지로부터 번호판을 추출한다. 주로 에지 검출(Edge Detection) 방법을 통해 번호판을 유추한다. 침식(Erosion), 팽창(Dilation), 캐니(Canny) 필터, 열림(Opening), 닫힘(Closing)을 이용하여 해당 영역을 유추하고, 이 부분의 특징 벡터를 추출한다. 이때, CCA(Connected Component Analysis), DFT(Digital Fourier Transform), Adaboost등을 이용하여 특징 벡터를 추출할 수 있다. 최근 들

어서는 YOLO[4], SSD[5]와 같은 객체탐지 모델을 사용하여 차량 번호판 자체를 직접 추출하기도 한다.

**4) 차량번호판 내 숫자, 한글 및 지역 정보 추출**

우리나라의 경우 차량 번호판이 규격화되어 있지만, 그 종류가 다양하다. 번호, 한글만 있는 경우도 있지만, 지역 정보가 있기도 하고, 2줄인 경우도 있으며, 색상을 통해서 영업용/전기자 등을 구분하기도 한다. 따라서, 이들 정보를 각각의 블랍(Blob)으로 추출해야 한다. 이때, 가로/세로 투영(Projection)이나 형태학적 얇음(Morphological Thinning), 형태학적 두꺼움(Morphological Thickening)을 통해서 번호 부분의 블랍을 추출한다.

**5) 차량 번호판 인식**

차량 번호판으로부터 번호 및 한글이 있는 각각의 블랍들을 획득하고, 블랍별 인식을 진행한다. 템플릿 매칭(Template Matching), SVM(Support Vector Machine), 신경망(Neural Network) 등을 이용하여 인식할 수 있고, 최근에는 딥러닝(Deep Learning)을 이용하여 차량 번호 인식을 연구한다.

**2-2 기존 시스템의 문제점 제시**

데이터 구축 시 고려해야 할 부분은 데이터의 활용 목적에 맞게 종류를 결정하고, 누가, 어디서, 언제, 어떠한 방법으로 획득할지를 결정하며, 인공지능 모델러와 상의 후 기준에 맞추어서 데이터 수집하는 과정이 이루어진다. 이때 정확한 지침에 맞추어서 데이터 획득 및 정제가 이루어져야만 한다. 그러나, 데이터 획득 및 정제과정은 대부분 사람의 수작업으로 처리되기 때문에 사람마다 주관적 기준이 적용되어 데이터 품질이 낮아질 수 있다. 특히 교통데이터의 경우 차량 번호판 인식을 이용하여 정제 기준을 제시하였고, 이를 통해서 수작업에 의한 시간 및 비용을 절감할 수 있다. 그러나, 차량번호 인식을 적용했음에도 오인식, 객체 겹침 등 다양한 요소에 의해서 정제 효율이 많이 떨어진다. 따라서, 다양한 인공지능 기술(객체추적, 차종 인식, LPR 병렬처리)을 활용하여 교통데이터에 대한 효율적인 자동정제를 진행하여 이러한 문제를 해결할 수 있다.

**III. 교통정보 데이터 자동정제 방법에 관한 연구**

**3-1 객체탐지 및 추적을 이용한 정제**

**1) 객체탐지(Object Detection)**

객체탐지의 실시간성과 인식 정확도를 어느 정도 보장할 수 있는 인공지능 모델로서 One Stage Detector의 가장 대표적인 YOLO(You Only Look Once)v5를 사용한다[4]. 이

모델의 특징은 Neural Network Single Forward만으로 다수의 객체탐지를 할 수 있고, 이미지 전체 Context로 탐지할 수 있어서, 빠르게 탐지할 수 있으며, 상대적으로 False Positive가 덜 나온다.

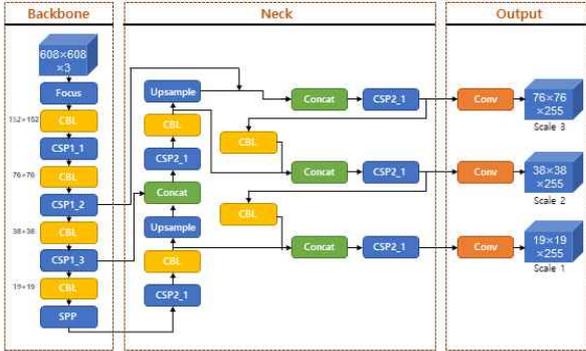


그림 3. YOLOv5 네트워크 모델  
 Fig. 3. YOLOv5 network model

그림 3을 보면 YOLOv5의 네트워크 모델을 나타낸다. YOLOv5 알고리즘은 크게 3개의 부분(Backbone, Neck, Output)으로 구성된다. Backbone에서 입력 영상의 Feature 정보를 추출하고, Neck은 추출된 Feature 정보를 기반으로 특징 맵 3개의 축척을 생성한다. 마지막으로 Output 부분에서는 생성된 Feature 정보를 기반으로 객체탐지 결과를 보인다.

Backbone은 Feature Map을 추출하는 Convolution Network로 다중 Convolution과 Pooling을 통해서 4개의 Map Layer를 구성한다. 이때 152x152, 76x76, 38x38, 19x19픽셀 Layer를 생성한다. Neck으로부터 더 많은 Context를 얻고 Loss 값을 줄이기 위해서 Backbone으로부터 받은 Map Layer는 PAN(Path Aggregation Network)과 FPN(Feature Pyramid Network)의 피라미드 구조를 사용할 수 있다. 각각을 살펴보면, PAN은 하위 Feature Map으로부터 강인한 Localization Feature 정보를 상위 Feature Map에 전달하여 융합 Feature 정보를 구성하는 방식이고, FPN은 상위 Feature Map에서 하위 Feature Map까지 모든

의미 있는 Feature 정보를 구성하는 방식이다. Feature 정보는 76x76x255, 38x38x255, 19x19x255 차원의 융합 Layer를 가진다. 다차원이 융합 Layer에서 탐지 과정이 이루어져서, 크고 작은 모든 객체에 대해서도 SOTA(State of the Art) 모델이 된다. 그림 4를 보면 COCO 데이터셋[6]을 기준으로 했을 때 YOLOv5의 성능지표가 나온다.

YOLOv5는 s, m, l, x, 4가지 모델이 제공되며 서로 다른 정확도와 속도가 나온다. 정제의 정확성을 높이기 위해서 본 논문에서는 x 모델을 사용하였다.

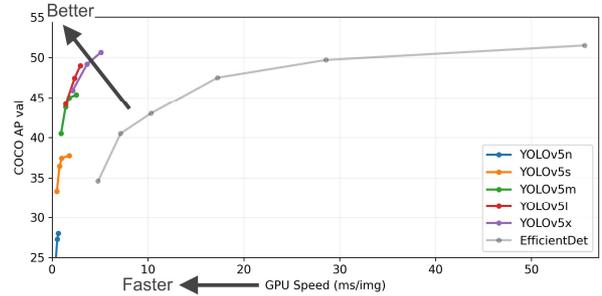


그림 4. YOLOv5 모델별 성능지표(COCO 데이터셋)[4]  
 Fig. 4. YOLOv5 performance by model(COCO)

2) IOU(Intersection over Union) 기반의 객체추적

IOU Based Object Tracker[7],[8]은 추적하고자 하는 객체에 대한 정보를 IOU를 기반으로 추적하는 원리이다. IOU가 높은 경우에 객체 간 겹치는 구간이 많이 차지하게 되고, 프레임 간의 객체 위치가 현재를 기준으로 해서 많이 벗어나지 않기 때문에 객체추적에 이용할 수 있다. 그림 5에서는 IOU 기반의 객체추적의 원리를 보여주고 있다.

IOU 기반의 객체추적 알고리즘의 최대 장점은 추가 학습이 이루어지지 않아도 되어 쉽게 적용 가능하다는 점이다. 하지만 그림 5(a)에서 보듯이 객체추적(Object Tracking) 중에 객체탐지(Object Detection)과정에서 미탐(False Negative)이 발생하면 연결성이 종료되어 ID 교체(Switch)문제가 발생한다. 이러한 문제를 해결하기 위해서 객체추적의 연결성이

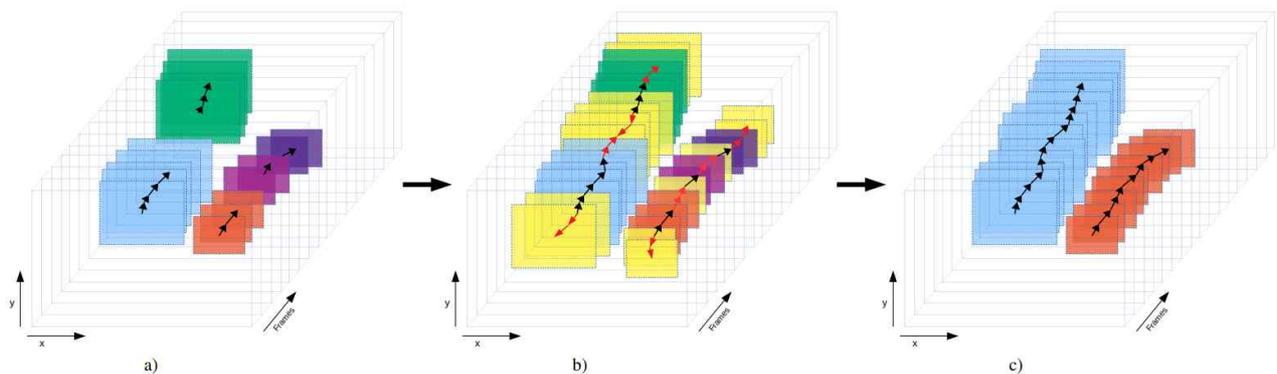


그림 5. IOU기반의 객체추적 기본원리[8]  
 Fig. 5. Basic principle of the extension for the IOU Tracker

종료된 순간부터 일정 프레임 동안 추적기(Tracker)의 정보를 객체탐지에 반영한다[8]. 만약 반영 후에 일정 프레임 이상 객체탐지가 되지 않아서 객체 추적이 불가능한 경우에는 추적기가 마지막으로 연결성이 종료된 장소에서 객체탐지가 종료된 것으로 판단한다. 이런 원리를 통해서 객체추적이 이루어지기 때문에 추적기가 반영해야 하는 프레임 수가 많아져서 검색할 객체의 숫자가 많아지면 리소스를 많이 차지하는 단점이 있다.

이 외에도 객체탐지를 기반으로 하는 객체 추적방법에는 특징기반의 객체추적(Feature based Object Tracking)[9], 광학 흐름 기반의 객체추적(Optical flow based Object Tracking)[10], Association을 이용한 객체추적[11] 등이 있으나 정체의 특성상 객체가 명확하게 드러나야 인공지능 모델 성능이 좋아지기 때문에 객체 겹침(Overlap)에 의한 ID 손실은 데이터를 명확하게 구분한다는 면에서 정체 시 더 도움이 될 수 있다. 본 논문에서 객체추적을 이용할 때 처음 객체탐지 시 임계값을 0.95로 설정하였고, 객체추적에 사용하는 IOU 임계값을 0.6으로 설정하였으며, 객체가 소실될 수 있는 최소 프레임수를 7로 설정하여 실험하였다.

3-2 차종인식을 이용한 정체

실생활 속에서 자료를 수집하다 보면 원하는 환경이 항상 구성되지는 않는다. 특히 교통데이터의 경우는 CCTV를 위쪽에서 도로면을 찍고, 교통 혼란 때문에 차량데이터가 겹쳐 보이는 경우가 많이 발생할 수 있다. 이 경우 인공지능 학습데이터를 위한 정제작업을 위해서는 각각의 객체들이 똑같은 ID를 가지면서 객체 겹침이 일어나지 않는 데이터를 수집하는 것이 중요하다. 만약 객체 겹침이 일어났더라도 다시 객체가 보이면 해당 ID를 그대로 유지하기 방안이 필요하다. 이러한 경우에는 식별(Classification) 기술을 이용하여 해당 ID를 그대로 유지할 수 있다.

식별을 위해서 객체 탐지 시 YOLOv5를 사용하였고, 정확도를 높이기 위해서 식별 알고리즘으로 EfficientNet[12] 사용하였다. 그림 6을 보면 EfficientNet의 모델사이즈별 성능을 확인할 수 있다(B0~B7은 모델 사이즈를 나타냄).

EfficientNet의 성능 측정방법을 Top-k(k=1) Accuracy로 했을 경우, B7에서 84.3%의 정확도가 나왔고, 다른 SOTA 알고리즘들과 비교했을 때보다 적은 Parameter 수가 나온다.

인공지능 모델의 성능향상을 높이기 위한 모델링 조정 방법은 3가지 방안이 있다.

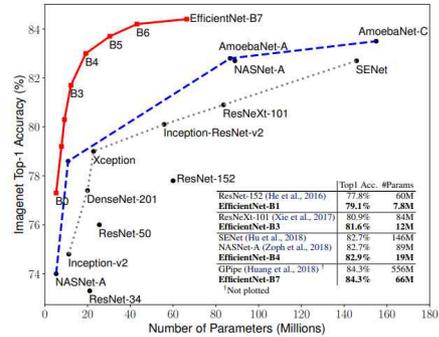


그림 6. Classification 성능지표 비교(ImageNet)[12]  
Fig. 6. Classification performance compare(ImageNet)

1) Depth(d)

Network가 깊어지면 좀 더 많은 복잡한 특징을 추출할 수 있고 다른 Task에 일반화하기 좋지만, Vanishing Gradient Problem이 생기게 된다. 이를 해결하기 위해 Resnet의 Batch Normalization, Skip Connection 등의 방법이 있지만, 모델이 너무 깊어지게 되면 성능의 한계를 가지게 된다.

2) Width(w)

Width(Channel)는 Scale Down(MobilenetV2 등)을 통해서 작은 모델을 만들 때 사용하고, 넓은 Channel을 통해서

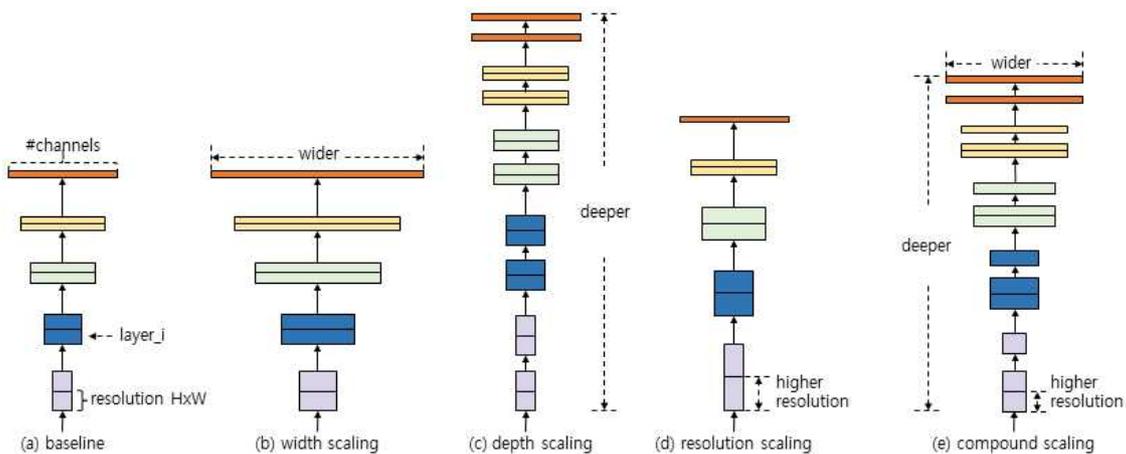


그림 7. 복합 스케일링  
Fig. 7. Compound scaling

는 섬세한 특징까지 대표할 수 있으나, Width를 계속 증가하는 경우 그 성능은 빠르게 포화(Saturation) 상태가 된다.

**3) Resolution(r)**

해상도 큰 이미지로 학습시킬 경우보다 높은 정확도를 얻을 수가 있다. 그러나 속도, 메모리 등의 문제들이 발생할 수 있다. 따라서, 학습 시 적정 크기의 이미지로 학습하며, 본 논문 실험 시에는 480x480 크기의 이미지를 사용하였다.

EfficientNet에서는 그림 7과 같이 Depth, Width, Resolution의 복합스케일링(Compound Scaling)을 진행한다. 본 논문에서 적용 시에는 Baseline 모델로 Resnet50을 사용하였고, 복합스케일링을 진행 시 Parameter로 d=1.4, w=1.2, r=1.3을 사용하였다. 이때 학습용 데이터셋은 스펀포드차량데이터셋[13]을 사용하였다.

**3-3 LPR 병렬처리를 이용한 정제**

인공지능 학습용 교통정보 데이터 정제 시 인덱스(Index)가 될 수 있는 정보는 차량번호이다. 따라서, 2-1의 LPR 알고리즘을 통해서 차량번호 인식을 진행한다. 수많은 번호 인식 알고리즘이 연구되고 있으나, 100% 번호 인식이 되는 경우는 없다. 현장의 상황, 이물질 묻음 정도, 카메라의 성능, 빛의 노출 정도 등 수많은 요인에 의해 인식률이 달라진다. 따라서 본 논문에서는 기존 번호 인식 엔진이 가지는 문제점을 개선하기 위하여, 전처리 부분에서 가변 임계치 병렬 알고리즘을 적용하여 번호 인식률을 높이는 연구를 진행하였다[3].

**1) 가변임계치 병렬 알고리즘**

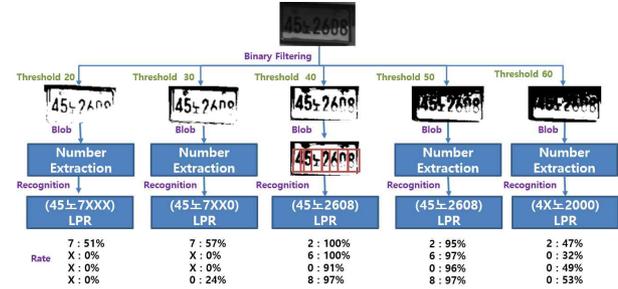
단일 LPR을 이용한 번호 인식보다 병렬처리를 통한 LPR을 진행했을 때 인식률이 더 높게 나오는 것을 확인할 수 있다[3]. 제안하는 전처리 과정에서는 이진화 필터 적용할 때 N개의 가변 임계치(Variable-Threshold)를 병렬처리하고 N개의 번호 인식 처리를 진행한다. 이때 이진화 필터의 임계값은 식 1과 같다.

$$Th = \frac{\sum plate}{size} + \text{가변임계치 (variable - threshold)} \quad (1)$$

if  $\in put > Th$  then  $\in put = 255$   
 if  $\in put < Th$  then  $\in put = 0$   
 plate : 번호판 영역의 pixel 값  
 size : plate의 모든 pixel의 수

특히 외부 변수에 의해서 추출된 번호판 원본이 번호 인식에 적합하지 않은 경우가 많이 발생하고, 차량번호 인식의 성능이 낮게 나올 때 이진화 필터, 가변 임계치 병렬처리를 적용하여, 번호 인식에 더욱 적합한 이미지를 추출할 수 있다. 본 논문 실험 시 N=5, 가변 임계치(Th)는 20부터 60까지 10씩 증가하여 다변화하여 적용하였다.

그림 8에서는 가변 임계치 병렬처리 과정을 나타낸다. 이진화 필터에 가변 임계치 병렬처리를 적용하면, 번호 인식에 더욱 적합한 이미지를 추출 및 인식하기 때문에 LPR의 성능을 높일 수 있다.



\*Korean license plate recognition photo

**그림 8.** 가변 임계치 병렬처리[3]

**Fig. 8.** Variable thresholds filtering in parallel process

**IV. 시스템 구현 결과**

**4-1 정제 이미지 종류 및 기준**

차량 객체 이미지의 경우는 그림 9와 같이 검출 영역(빨간색 영역)에 진입된 차량이 완전히 사라진 이후 차량에 대해서 새롭게 정제를 진행한다.



\*Korean license plate recognition photo

**그림 9.** 차량 객체 이미지 정제 기준

**Fig. 9.** Criteria for vehicle object image refinement

차량 번호판 이미지 정제의 경우는 그림 10과 같이 정면, 후면, 이륜차 번호판으로 나눌 수 있다.



\*Korean license plate recognition photo

**그림 10.** 차량 번호판 이미지 정제 기준

**Fig. 10.** Criteria for vehicle plate image refinement

차량 외관 이미지 정제의 경우는 그림 11과 같이 정면, 후면, 옆면으로 나눌 수 있다.



그림 11. 차량 외관 이미지 정제 기준  
Fig. 11. Criteria for vehicle exterior image refinement

그림 9~11과 같이 인공지능 학습용 교통정보 데이터 수집 및 정제를 할 수 있으며, 이때 그림 12는 올바른 정제 예시를 보여준다.

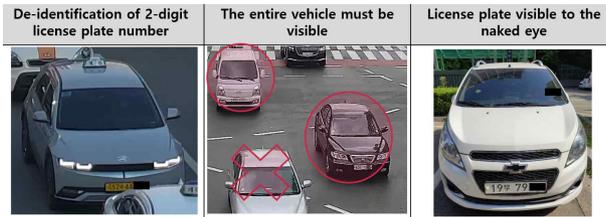


그림 12. 올바른 정제 예시  
Fig. 12. Example of correct refinement

#### 4-2 인공지능 학습용 교통정보 데이터 자동정제 연동

CCTV 교통데이터 중 그림 12와 같이 가장 확실한 데이터를 추출하여 데이터 학습에 사용해야 하므로 전체 영상을 역으로 재생함으로써, 카메라에 가장 근접한 데이터부터 수집한다. 이는 차량들이 카메라 가까이 다가왔을 때, 더욱 적합한 정제 데이터가 수집될 확률이 크기 때문이다. 정제과정을 보면, 역으로 추출된 프레임들을 기준으로 YOLOv5 알고리즘을 이용하여 원하는 객체 데이터를 추출한다. 이때 기준 객체탐지의 임계값  $\sigma_h \geq 0.95$ 인 객체만 사용하며, 탐지된 객체에 대해서 BBox 정보(x, y, width, height)를 가지고 있고 객체의 BBox 중심점인 BBox\_Start\_Center\_Position 값을 저장한다. 그리고, 역으로 재생된 프레임 순으로 객체추적을 진행하며,  $\sigma_{IOU} \geq 0.6$ 인 객체인 경우 같은 객체라고 판단하여 이 객체들은 정제 시 사용하지 않는다. 이때  $\sigma_{IOU} < 0.6$ 인 마지막 객체의 BBox의 중심점 BBox\_End\_Center\_Position 값을 저장한다.  $\sigma_{IOU} < 0.6$ 이면 새로운 객체인지 기존의 객체 연장인지를 판단해야 한다. 이 경우에 EfficientNet 식별 기술을 사용하여 객체 속성을 결정한다. 프레임 영역 내에 기준 객체가 존재하는 경우 이 프레임은 정제 후보 프레임이 된다. 물론 한 프레임 내에 기준객체와 미기준객체가 같이 존재할 수 있다. 이때 기준객체가 많이 존재할수록 적합한 정제 데이터라고 할 수 있다. 이렇게 추출된 정제 프레임 내 객체들의 인덱스 정보를 부여하기 위해서 객체별 차량번호 인식을 수행한다. 번호 인식의 인식률을 높이기 위해서 LPR 병렬처리를 이용하며, 번호 인식에 실패하거나, 동일객체의 차량번호 인식결과가 같은 경우 해당 객체는 선택하지 않는다. 이러한

과정을 통해서 인식된 차량번호는 어노테이션(Annotation) 시 json, xml과 같은 컨테이너에 인덱스 정보로 사용한다.

#### 4-3 알고리즘 선택 정제 비교

인공지능 학습용 교통정보 데이터 자동 정제 시 사용하는 알고리즘들을 아래와 같이 선택적으로 결정할 수 있다.

4-1의 기준으로 하여 만족하는 정제 파일이 나오는 것을 확인해 보았다. 위 알고리즘으로 정제된 파일 1000개를 확인하였다. ①의 경우는 프레임 내 모든 객체가 탐색 되기 때문에 객체 겹침, 차량중복인식 및 차량번호 오인식이 나는 경우가 많이 발생하였다. ②의 경우는 차량번호 인식이 되는 객체가 있을 때만 정제하였으나, 차량중복인식이 많이 되어서 정제되었다. ③의 경우 가장 성능향상 정도를 볼 수 있었으나, 차량번호판이 잘 안 보일 때에도 정제가 되었다. ④의 경우 객체가 겹치지 않으면서, 차량번호판이 가장 잘 보이는 파일들을 선택하여 정제하였다. ⑤의 경우 차량 외관에 대한 검증 프로세스를 거쳐서 보다 정밀하게 정제된 것을 확인할 수 있다. 이 부분은 Object Tracking시 ID switch 되는 부분에 대한 보정이 이루어진 것이다.

표 1. 성능 비교(1000개 파일기준)

Table 1. Performance comparison

Algorithm	①	②	③	④	⑤
Correct number of refined files	126	325	598	878	902

- ① Object Detection 만이용
- ② Object Detection + LPR 병렬처리 이용
- ③ Object Detection + Object Tracking 이용
- ④ Object Detection + Object Tracking + LPR 병렬처리 이용
- ⑤ Object Detection + Object Tracking + LPR 병렬처리 이용 + 차중 인식

## V. 결 론

인공지능 학습용 교통정보 데이터는 학습에 가장 중요한 요소이기 때문에 본 논문에서는 고품질 학습데이터 획득을 위한 학습데이터 정제방법을 제시하였다. 차량 정보 인식 및 추적 기법을 이용하여 교통데이터 정제 시 자동화를 진행하였으며, 수작업으로 했을 때보다 시간 및 비용 절감할 수 있었다. 또한, 제안한 알고리즘을 조합했을 때 더욱 효과적인 양질의 데이터가 정제되었음을 확인하였다.

또한, 지금까지는 Vision AI 알고리즘을 사용하여 인공지능 학습용 교통정보 데이터를 정제했지만, 향후에는 VLM (Vision Language Models) 멀티 모달 인공지능 모델들을 이용하여 해당 부분의 효율성 및 사용성을 더욱 높일 수 있도록 연구를 진행할 것이다.

참고문헌

[1] Y. Koh, J. Park, H. Oh, H. Yoo, H. Park, K. Seo, ... and Y. Koo, A Guide to Building Datasets for AI Training, National Information Society Agency, Daegu, February 2021.

[2] D.-J. Kim and Y.-G. Jeon, "Vision AI-Based Automatic Selective De-Identification," *Journal of Digital Contents Society*, Vol. 24, No. 4, pp. 725-734, April 2023. <http://dx.doi.org/10.9728/dcs.2023.24.4.725>

[3] D.-J. Kim and H. Moon, "Implementation of Pre-and-Post Processing Algorithm to Improve LPR(License Plate Recognition) Rate," *The Transactions of the Korean Institute of Electrical Engineers*, Vol. 68, No. 12, pp. 1594-1600, December 2019. <https://doi.org/10.5370/KIEE.2019.68.12.1594>

[4] GitHub. Ultralytics YOLO Vision [Internet]. Available: <https://github.com/ultralytics/yolov5>.

[5] W. Liu, D. Anguelov, D. Erhan, C. Szegedy, S. Reed, C.-Y. Fu, and A. C. Berg, "SSD: Single Shot MultiBox Detector," in *Proceedings of the 14th European Conference on Computer Vision (ECCV 2016)*, Amsterdam, Netherlands, pp. 21-37, October 2016. [https://doi.org/10.1007/978-3-319-46448-0\\_2](https://doi.org/10.1007/978-3-319-46448-0_2)

[6] COCO - Common Objects in Context. COCO Dataset [Internet]. Available: <https://cocodataset.org/#home>.

[7] N. Wojke, A. Bewley, and D. Paulus, "Simple Online and Realtime Tracking with a Deep Association Metric," in *Proceedings of 2017 IEEE International Conference on Image Processing (ICIP)*, Beijing, China, pp. 3645-3649, September 2017. <https://doi.org/10.1109/ICIP.2017.8296962>

[8] E. Bochinski, T. Senst, and T. Sikora, "Extending IOU Based Multi-Object Tracking by Visual Information," in *Proceedings of the 15th IEEE International Conference on Advanced Video and Signal Based Surveillance (AVSS)*, Auckland, New Zealand, pp. 1-6, November 2018. <https://doi.org/10.1109/AVSS.2018.8639144>

[9] G. Koch, R. Zemel, and R. Salakhutdinov, "Siamese Neural Networks for One-Shot Image Recognition," in *Proceedings of the 32nd International Conference on Machine Learning (ICML '15)*, Lille, France, July 2015.

[10] D.-A. Beaupré, G.-A. Bilodeau, and N. Saunier, "Improving Multiple Object Tracking with Optical Flow and Edge Preprocessing," arXiv:1801.09646, January 2018. <https://doi.org/10.48550/arXiv.1801.09646>

[11] Y. Xu, A. Ošep, Y. Ban, R. Horaud, L. Leal-Taixé, and X. Alameda-Pineda, "How to Train Your Deep Multi-Object Tracker," in *Proceedings of 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*,

Seattle: WA, pp. 6786-6795, June 2020. <https://doi.org/10.1109/CVPR42600.2020.00682>

[12] A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, ... and N. Houlsby, "An Image Is Worth 16x16 Words: Transformers for Image Recognition at Scale," arXiv:2010.11929v2, June 2021. <https://doi.org/10.48550/arXiv.2010.11929>

[13] Kaggle. Stanford Cars Dataset [Internet]. Available: <https://www.kaggle.com/datasets/jessicali9530/stanford-cars-dataset>.

[14] K. Kim and H. Kim, "AI Model-Based Automated Data Cleaning for Reliable Autonomous Driving Image Datasets," *Journal of Broadcast Engineering*, Vol. 28, No. 3, pp. 302-313, May 2023. <https://doi.org/10.5909/JBE.2023.28.3.302>

김대진(Dae-Jin Kim)



1998년 : 대전대학교 대학원 (공학사)  
2000년 : 동국대학교 대학원 (공학석사)  
2010년 : 대전대학교 대학원 (공학박사)

2017년~현 재: 동국대학교 영상문화콘텐츠연구원 조교수  
※ 관심분야: 코덱, 멀티미디어 플랫폼, 콘텐츠 DNA, 워터마크, 딥러닝, 번호인식, 주차관제시스템 등

김준화(Jun-Hwa Kim)



2019년 : 동국대학교 (공학사)  
2020년 : 동국대학교 대학원 (공학석사)  
2023년 : 동국대학교 대학원 (공학박사)

2023년~2024년: 동국대학교 박사후연구원  
2024년~현 재: 건양대학교 인공지능학과 조교수  
※ 관심분야: 영상처리, 컴퓨터비전, 신호처리, 딥러닝 등