

일반논문 (Regular Paper)

방송공학회논문지 제25권 제5호, 2020년 9월 (JBE Vol. 25, No. 5, September 2020)

<https://doi.org/10.5909/JBE.2020.25.5.776>

ISSN 2287-9137 (Online) ISSN 1226-7953 (Print)

## 가상 데이터를 활용한 번호판 문자 인식 및 차종 인식 시스템 제안

이 승 주<sup>a)</sup>, 박 구 만<sup>b)†</sup>

### Proposal for License Plate Recognition Using Synthetic Data and Vehicle Type Recognition System

Seungju Lee<sup>a)</sup> and Gooman Park<sup>b)†</sup>

#### 요 약

본 논문에서는 딥러닝을 이용한 차종 인식과 자동차 번호판 문자 인식 시스템을 제안한다. 기존 시스템에서는 영상처리를 통한 번호판 영역 추출과 DNN을 이용한 문자 인식 방법을 사용하였다. 이러한 시스템은 환경이 변화되면 인식률이 하락되는 문제가 있다. 따라서, 제안하는 시스템은 실시간 검출과 환경 변화에 따른 정확도 하락에 초점을 맞춰 1-stage 객체 검출 방법인 YOLO v3를 사용하였으며, RGB 카메라 한 대로 실시간 차종 및 번호판 문자 인식이 가능하다. 학습데이터는 차종 인식과 자동차 번호판 영역 검출의 경우 실제 데이터를 사용하며, 자동차 번호판 문자 인식의 경우 가상 데이터만을 사용하였다. 각 모듈별 정확도는 차종 검출은 96.39%, 번호판 검출은 99.94%, 번호판 검출은 79.06%를 기록하였다. 이외에도 YOLO v3의 경량화 네트워크인 YOLO v3 tiny를 이용하여 정확도를 측정하였다.

#### Abstract

In this paper, a vehicle type recognition system using deep learning and a license plate recognition system are proposed. In the existing system, the number plate area extraction through image processing and the character recognition method using DNN were used. These systems have the problem of declining recognition rates as the environment changes. Therefore, the proposed system used the one-stage object detection method YOLO v3, focusing on real-time detection and decreasing accuracy due to environmental changes, enabling real-time vehicle type and license plate character recognition with one RGB camera. Training data consists of actual data for vehicle type recognition and license plate area detection, and synthetic data for license plate character recognition. The accuracy of each module was 96.39% for detection of car model, 99.94% for detection of license plates, and 79.06% for recognition of license plates. In addition, accuracy was measured using YOLO v3 tiny, a lightweight network of YOLO v3.

Keywords : YOLO, Synthetic Data Generation, Object Detection, Text Recognition

## I. 서론

최근 신형 자동차 번호판이 추가됨에 따라 주차관계 시스템의 문자인식 기능이 오인식 되는 사례가 발생되었다. 이러한 오인식은 새로 추가된 번호판 양식의 글자수 증가와 향후 추가예정인 반사필름식 번호판 등의 재질 변화에 따른 오작동으로 분석된다. 또한 대부분의 자동차 번호판 문자인식 관련 시스템에서는 차량의 번호판만을 이용하여 출입 허가를 여부를 판단한다. 그러나 차량번호판을 불법으로 교체한 차량의 경우 기존 시스템으로 구별하기 어렵다. 또한 기존 자동차 번호판 문자인식은 영상처리를 통하여 번호판 영역을 추출 후 DNN(deep neural network)을 통한 문자인식 방법<sup>[1]</sup>과 딥러닝을 활용하여 객체인식 네트워크를 통한 번호판 영역 검출 후 OCR(optical character recognition) 네트워크를 이용한 문자인식 방법<sup>[2]</sup>을 사용하였다. 이러한 방법은 번호판 인식의 배경이 변하거나 번호판의 폰트가 변하면 인식률이 하락되는 단점이 있다.

따라서 본 논문에서는 이러한 문제를 해결하기 위하여, 자동차 번호판 인식과 차종 인식 시스템을 제안한다. 본 시스템은 환경변화에 강인할 수 있도록 번호판 검출 및 문자 검출을 모두 딥러닝을 사용한 객체인식 네트워크를 사용하였다. 또한 여러 폰트에 대응할 수 있도록 가상데이터 생성 시 여러 폰트를 추가하였다. 자동차 번호판 문자인식은 그림 1과 같이 2006년 이후 보급된 흰색 바탕의 지역명이 제외된 데이터로 범위를 한정하며 새로 추가된 8자리 신형 번호판을 인식한다. 문자 인식 학습데이터는 특징점 매칭(feature matching)을 통한 유사도가 높은 폰트를 사용하여

가상으로 차량 번호판을 생성하였다. 가상으로 데이터를 생성함으로써, 데이터 수집 및 정리에 소요되는 시간을 줄였다. 그리고 차종 인식을 위하여 학습데이터를 제조사 홈페이지 홍보사진을 사용하여 학습 후 차량 전면의 사진을 이용하여 테스트하였다. 본 시스템은 딥러닝 기반의 객체 검출(object detection) 중에 1-stage 방법인 YOLO v3<sup>[3]</sup> 사용하였다. YOLO v3는 대표적인 2-stage 방식의 R-CNN(R-CNN<sup>[4]</sup>, Fast R-CNN<sup>[5]</sup>, Faster R-CNN<sup>[6]</sup>) 계열 방법과 비교하여 정확도는 낮으나 검출 속도가 빠른 장점이 있다. 실시간으로 진입하는 차량의 번호판과 차종을 추론하기 위하여 정확도 보다 속도에 초점을 두었다. 본 시스템의 장점은 RGB 카메라 1대로 차종 및 차량번호판 문자인식을 수행한다. 본 시스템을 도로 교통 관계 시스템, 주차 관계 시스템에 설치된 번호판 인식용 카메라를 활용하면 추가 카메라 장착 없이 서비스 할 수 있을 것으로 예상된다. 또한 현재 무인 주차관계시스템을 보급하고 있는데, 제안하는 시스템을 사용하면 저렴한 비용으로 구축할 수 있을 것으로 예상된다.

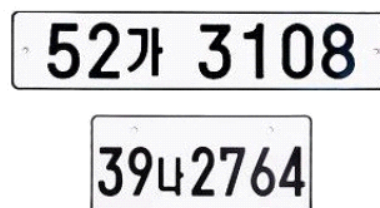


그림 1. 2006년 이후 보급된 번호판  
Fig. 1. License plates distributed since 2006

## II. 관련 연구

### 1. 딥러닝 기반 객체 검출

딥러닝 기반의 객체 검출은 객체 검출 절차에 따라 총 2가지로 구분한다. 가장 먼저 등장한 객체 검출 방법은 2-stage 객체 검출 방법이며, 이후 검출 속도에 초점을 맞춰 제안된 1-stage 객체 검출 방법이 있다. 2-stage 방법은 후보 영역 검출(region proposal) 후 후보영역에 포함된 객체를 분류(region classification)하며, 1-stage와 비교하여 검출 정확도는 높지만 속도가 느린 단점이 있다. 대표적인 2-stage

a) 서울과학기술대학교 일반대학원 미디어IT공학과(Dept. of Media IT Engineering, The Graduate School, Seoul National University of Science and Technology)

b) 서울과학기술대학교 전자IT미디어공학과(Dept. of Electronic IT Media Engineering, Seoul National University of Science and Technology)

\* Corresponding Author : 박구만(Gooman Park)

E-mail: gmpark@seoultech.ac.kr

Tel: +82-2-970-6430

ORCID:https://orcid.org/0000-0002-7055-5568

\* 동 연구는 다부처사업으로 수행중인 재난치안용 멀티콥터 무인기 특화 임무장비 기술 개발(과제번호 10080094) 연구결과 중 일부임.

\* This study is part of the research result of the multi-copter un-manned special mission equipment development project (No. 10080094)

· Manuscript received July 16, 2020; Revised August 31, 2020; Accepted August 31, 2020.

객체검출 방법은 R-CNN(R-CNN, Fast R-CNN, Faster R-CNN, Mask R-CNN<sup>[7]</sup>) 계열이 있다. 1-stage 방법은 2-stage의 후보영역 검출과 객체 분류를 한번에 수행하여 검출속도가 빠르지만 정확도가 낮은 단점이 있다. 대표적인 1-stage 객체 검출 방법은 YOLO(YOLO v1<sup>[8]</sup>, YOLO v2<sup>[9]</sup>, YOLO v3)와 SSD<sup>[10]</sup>가 존재한다. 본 논문에서 제안하는 시스템은 실시간 객체 검출을 기반으로 수행하기 때문에 YOLO v3를 사용한다.

## 2. 문자 검출(Text Detection)

문자 검출은 글자 단위(character based)의 검출과 단어 단위(word based)의 검출방법이 있다. 글자 단위 검출방법에는 대표적으로 Lukas Neumann<sup>[11]</sup>이 있으며, ER detector를 활용하여 문자 영역 후보 생성 후 합성 폰트(synthetic font)를 사용하여 OCR을 학습한다. 반면, 단어 단위의 검출은 객체 검출을 이용하여 단어를 검출한다. 대표적인 방법으로는 R-CNN 기반의 Max Jaderberg<sup>[12]</sup>와 YOLO 기반의 Ankush Gupta<sup>[13]</sup>가 존재한다. 각각의 논문에서 앵커 박스 (Anchor Boxes)를 포함한 다양한 파라미터를 수정하여 문자를 검출율을 향상시켰으나, 본 논문에서는 YOLO v3를 기반으로 파라미터 수정 없이 글자 단위로 문자를 검출한다.

## III. 전체 시스템 구조도

그림 2는 본 논문의 전체 시스템을 구조도로 나타낸다. 차량 전면 이미지를 이용하여 차종 인식과 차량 번호판을 인식한다. 차종 인식(Car model recognition)은 입력받은 전면 이미지를(Image)를 사용하여 제조사와 차량 모델명을 추론한다. 동시에 차량 번호판 영역 검출(Car license plate area detection)은 차량 번호판 문자 인식을 위하여 이미지 내 차량 번호판 영역을 찾는다. 만약 번호판 영역을 찾지 못하였다면, 차량 번호판 문자 인식 모듈을 실행하지 않는다. 차량 번호판 영역을 찾았다면, 처음 입력 받은 이미지에서 차량 번호판 영역을 추출하여 차량 번호판 문자 인식(Car license plate character recognition) 네트워크에 입력한다. 최종 출력에서 차종인식 결과와 차량 번호판 문자 인식 결과를 표출한다.

## IV. 딥러닝 기반의 차종 인식 모듈

### 1. 학습 데이터 구성

학습에 사용된 데이터의 출처는 제조사 홈페이지와 자동

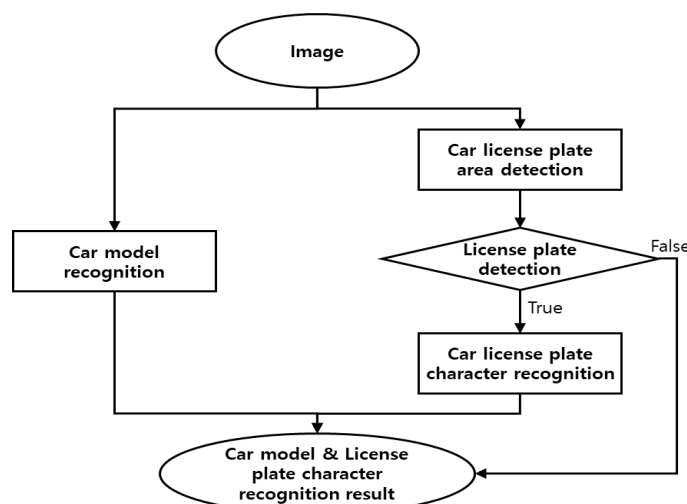


그림 2. 전체 시스템 구조도

Fig. 2. Overall system structure

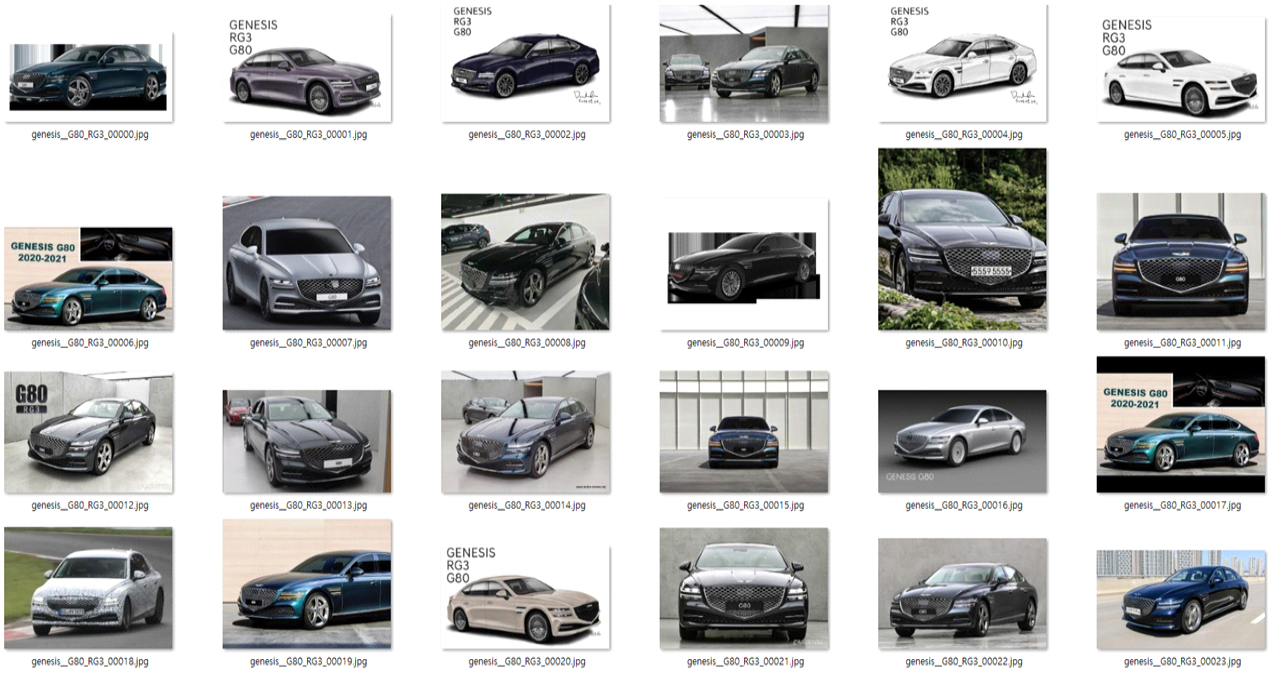


그림 3. G80\_RG3 클래스의 학습 데이터  
Fig. 3. Training data of G80\_RG3 class

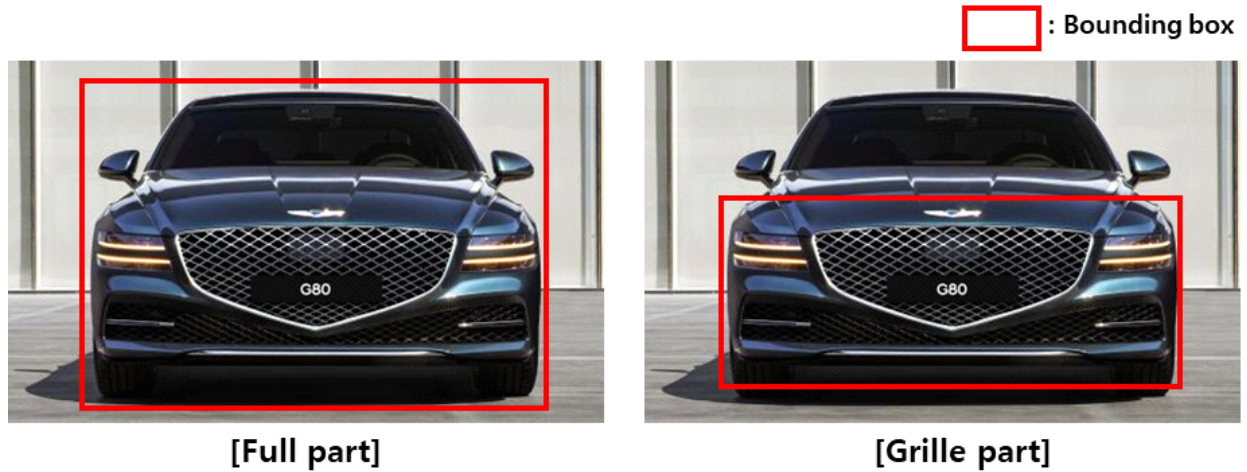


그림 4. 차종인식 모델별 RoI 설정  
Fig. 4. RoI setting by car model recognition model

차 관련 인터넷 뉴스 기사에 포함된 차량 전면부 이미지를 사용하였다. 그림 3은 G80\_RG3 클래스(class)의 차종 학습데이터를 나타낸다. 본 장의 차종인식 실험은 그림 4와







같이 차량 전면부 전체(Full part)와 차량 전면 그릴(Grille part)로 구성하여 테스트하였다. 표 1은 각 모델의 학습데이터와 클래스 번호, 학습데이터 수를 정리하였다.

표 1. 차종인식 모델별 학습 데이터 정보

Table 1. Training data information by car model

Class index	Manufacturer	Car_model	Number of image data
0	Hyundai	SANTAFE_TM	48
1	Hyundai	Grandeur_IG	39
2	Kia	All_new_carnival	41
3	Kia	The_New_Sorrento	44
4	Hyundai	Sonata_New_Rise	59
5	Kia	All_New_Morning_JA	78
6	Hyundai	Avante_AD	47
7	Hyundai	The_New_Grand_Starex	54
8	Ssangyong	Tivoli_Armor	65
9	Kia	K5_2nd_generation	75
10	Ssangyong	Rexton_sports	59
11	Hyundai	Kona	72
12	Kia	The_New_K3	51
13	Genesis	G80	8
14	Benz	E-Class_w213	16
15	Kia	All_New_K7	47
16	Renaultsamsung	QM6	72
17	Chevrolet	The_Next_Spark	55
18	Hyundai	Sonata_DN8	52
19	Kia	The_New_Carnival	39
20	Hyundai	The_New_Avante_AD	64
21	Hyundai	Palisade	68
22	Kia	All_New_K3	28
23	Ssangyong	Rexton_Sports_Khan	26
24	Kia	K7_Premier	49
25	Hyundai	All_New_Tucson	52
26	Chevrolet	The_New_Spark	38
27	Kia	The_New_K5_2nd_generation	62
28	Hyundai	The_New_Grandeur_IG	26
29	Kia	K5_3rd_generation	47
30	Kia	Celltos	42
31	Renaultsamsung	The_New_QM6	49
32	Hyundai	Avante_CN7	36
33	Renaultsamsung	XM3	34
34	Genesis	GV80	21
35	Genesis	G80 RG3	24

표 2. 차종인식 모델 데이터 증강 파라미터 정보  
Table 2. Car model data augmentation parameter information

Parameter Name	Parameter Value	Augmentation Result	Remarks
Add	-40 ~ 40		Brightness
Affine(translate)	-0.04 ~ 0.04		Move position
Affine(scale)	0.4 ~ 0.9		Resize
Affine(rotate)	-2 ~ 2		Rotation
GaussianNoise	0 ~ 0.1		-
GammaContrast	0.2 ~ 1.75		-

## 2. 증강 파라미터

차종인식을 위한 학습데이터는 한 클래스에 최소 8장에서 최대 80장으로 구성되어 있다. 이러한 데이터로 학습시 학습데이터 불균형과 학습데이터와 유사한 환경에서만 좋은 성능을 보인다. 따라서 다양한 환경에서의 성능 향상과 데이터 불균형을 해결하기 위하여 데이터를 증강(data augmentation)하였다. 증강 파라미터는 표 2와 같이 적용하였으며, 각 클래스마다 20,000장으로 구성하였다.

## 3. 학습 및 테스트시 네트워크 파라미터

학습과 테스트에 사용된 네트워크의 이미지 입력크기는 416\*416\*3(width, height, channel)으로 동일하며, 1 에폭(=11,000 iteration) 반복하여 학습하였다. 클래스 수는 앞서 정리한 표 1과 같이 총 36개이며, 학습률(learning rate)은 0.001, 배치(batch)는 64로 설정하였다.

## V. 딥러닝 기반의 번호판 검출 모듈

본 장의 번호판 검출 모듈은 그림 5와 같이 번호판의 영역을 추출(crop)하는 기능을 수행한다.

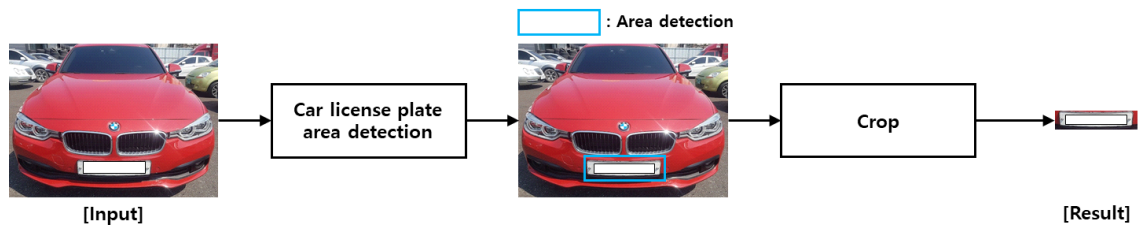


그림 5. 번호판 검출 모듈 구조도  
Fig. 5. Structure of license plate detection module



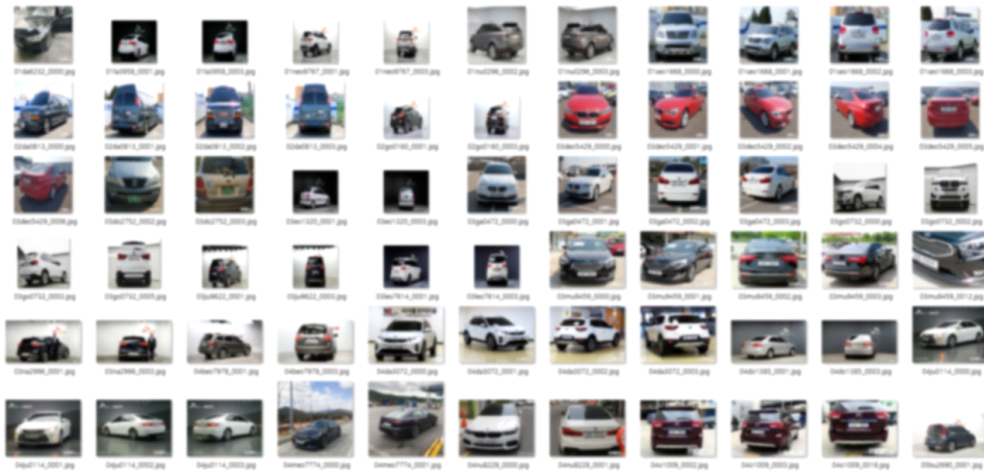


그림 6. 번호판 검출 학습 데이터  
Fig. 6. Learning data of car license plate detection

### 1. 학습데이터 구성

번호판 검출을 위한 학습데이터는 중고차 거래 사이트에서 차종 구분 없이 총 2,173장을 수집하여 학습데이터로 사용하였다. 그림 6은 학습데이터의 원본 일부이며, 사적인 데이터가 포함되어 있어 블러(blur)처리 하였다. 클래스는

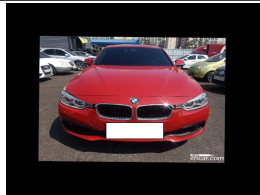
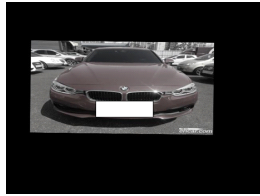
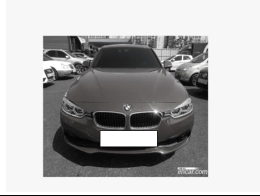
총 1개이며, 번호판 영역만 바운딩 박스에 포함시켰다.

### 2. 증강 파라미터

다양한 환경에 강인한 모델을 만들기 위하여 학습데이터를 증강하였다. 각각 다른 파라미터로 구성하여 총 3개의

표 3. 번호판 검출 모델 데이터 증강 파라미터 정보

Table 3. Car license plate detection model data augmentation parameter information

Model Name	Parameter Name	Parameter Value	Augmentation Result	Remarks
Plate_0(190913)	GammaContrast	1 ~ 1.5		background : black
	Affine(translate)	-0.06 ~ 0.06		
	Afiine(scale)	0.4 ~ 0.8		
	Affine(rotate)	-		
	GaussianBlur	-		
	Grayscale	-		
Plate_1(190928)	GammaContrast	1 ~ 1.5		background : black
	Affine(translate)	-0.06 ~ 0.06		
	Afiine(scale)	0.4 ~ 0.8		
	Affine(rotate)	-1 ~ 1		
	GaussianBlur	0 ~ 2.0		
	Grayscale	0 ~ 1		
Plate_2(190929)	GammaContrast	1 ~ 1.5		background : white
	Affine(translate)	-0.06 ~ 0.06		
	Afiine(scale)	0.4 ~ 0.8		
	Affine(rotate)	-1 ~ 1		
	GaussianBlur	0 ~ 2.0		
	Grayscale	0 ~ 1		

모델의 학습 데이터를 생성하였다. 표 3은 각 모델별 데이터 증강 파라미터를 정리하였다. 데이터 증강 후 각 모델별 학습데이터는 총 1,000,000장으로 구성하였다.

### 3. 학습 및 테스트시 네트워크 파라미터

학습과 테스트에 사용된 네트워크의 이미지 입력크기는  $416 \times 416 \times 3$ 으로 동일하며, 1 에폭(=16,000 iteration) 반복하여 학습하였다. 클래스 수는 총 1개이며, 학습률은 0.001, 배치는 64로 설정하였다.

## Ⅶ. 딥러닝 기반의 번호판 문자 인식 모듈

### 1. 폰트 비교

번호판 문자 데이터를 가상으로 생성시 실제 데이터와 유사한 폰트를 사용하기 위하여, 특징점 매칭 기반의 이미지 유사도 방법인 SIFT<sup>[14]</sup>, ORB<sup>[15]</sup>, FLANN<sup>[16]</sup>을 사용하였다. 상업적으로 사용가능한 폰트 100종을 확보하여 각 클래스 별 실제 데이터와 가상데이터의 유사도를 측정하였다. 그림 7은 유사도 측정 결과 중 상위 5개의 값을 그래프로 나타내었다.

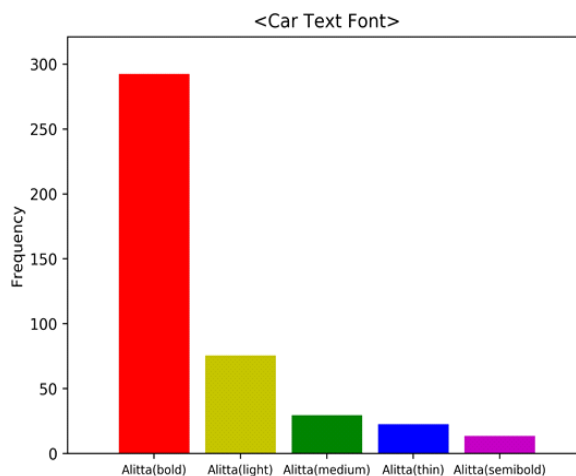


그림 7. 특징점 매칭을 이용한 폰트 유사도 비교  
Fig. 7. Comparing Font Similarities with Feature Matching

### 2. 학습데이터 구성

유사도가 높은 상위 3개 폰트를 사용하여 가상 데이터를 생성하였다. 그림 8은 생성된 가상 번호판 데이터를 나타낸다. 또한 가상 데이터와 실제 데이터를 비교하기 위하여, 실제 데이터를 수집하였다. 인식할 문자의 클래스는 총 50개이며, 표 4에 정리하였다.

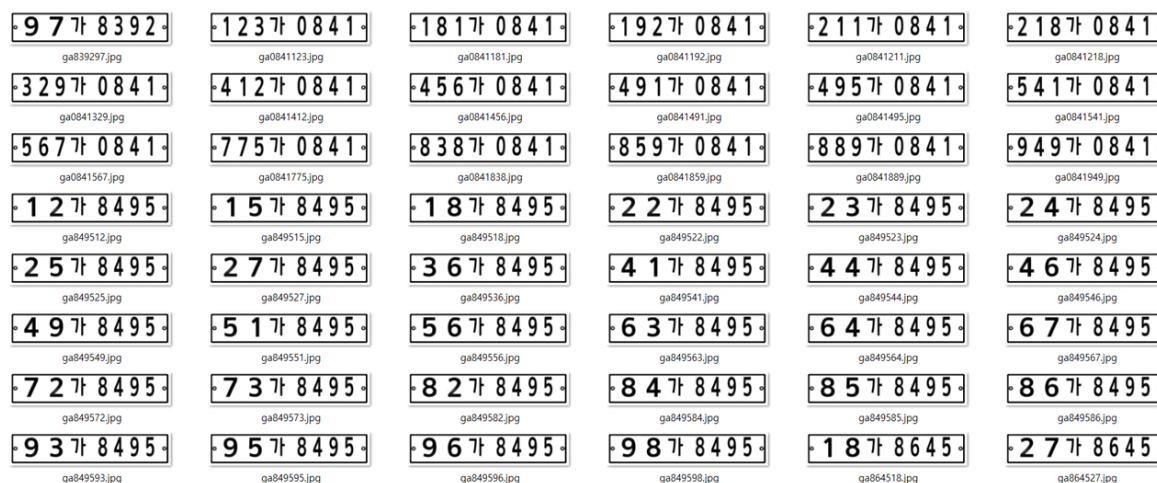


그림 8. 가상 생성 번호판  
Fig. 8. Synthetic generated license plate




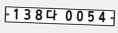

표 4. 번호판 인식 모델별 학습 데이터 정보

Table 4. Training data information by license plate recognition model

Class index	Class name	Class index	Class name	Class index	Class name
0	0	17	deo	34	nu
1	1	18	leo	35	du
2	2	19	meo	36	lu
3	3	20	beo	37	mu
4	4	21	seo	38	bu
5	5	22	eo	39	su
6	6	23	jeo	40	u
7	7	24	go	41	ju
8	8	25	no	42	a
9	9	26	do	43	ba
10	ga	27	lo	44	sa
11	na	28	mo	45	ja
12	da	29	bo	46	bae
13	la	30	so	47	ha
14	ma	31	o	48	heo
15	geo	32	jo	49	ho
16	neo	33	gu		

표 5. 번호판 인식 모델 데이터 증강 파라미터 정보

Table 5. License plate recognition model data augmentation parameter information

Parameter Name	Parameter Value	Augmentation Result	Remarks
GammaContrast	1 ~ 1.5	-	Brightness
Affine(translate)	-0.06 ~ 0.06	-	Move position
Affine(scale)	0.4 ~ 0.8		Resize
Affine(rotate)	-3 ~ 3		Rotation
GaussianBlur	0 ~ 2.0		-
Grayscale	0 ~ 1	-	background : white

### 3. 증강 파라미터

선행 연구인 가상 데이터 생성을 통한 딥러닝 기반 문자 인식 시스템 제안<sup>[17]</sup>에서 좋은 성능을 보인 Model\_9의 증강 파라미터 값을 사용하여 가상 데이터와 실제 데이터 각각 1,500,000장으로 증강시켰다. 표 5는 증강 파라미터를 정리하였다.

### 4. 학습 및 테스트시 네트워크 파라미터

학습과 테스트에 사용된 네트워크의 이미지 입력크기는 416\*416\*3으로 동일하며, 1 에폭(=24,000 iteration) 반복

하여 학습하였다. 클래스 수는 총 50개이며, 학습률은 0.001, 배치는 64로 설정하였다.

### 5. 문자 인식 결과 후처리

YOLO v3의 결과값은 객체의 클래스 확률(class probability)을 기준으로 내림차순으로 정렬한 값을 출력한다. 본 장의 문자 인식 모듈에서는 객체의 위치 기반으로 정렬 후 객체의 클래스 이름(class name, object name)을 추출한다. 그림 9는 문자 인식결과를 객체 위치 기반으로 정렬하는 과정을 도식화 하였다. 그림에서 ①, ②는 객체의 클래스 확률에 따른 순서를 나타내며, 파란색 박스는 검출된 객체의

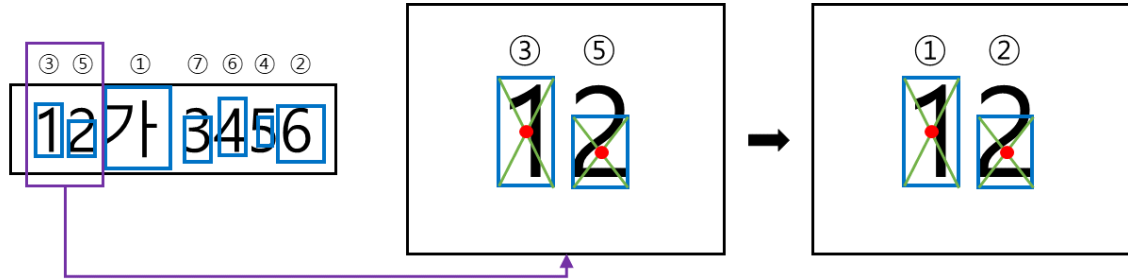


그림 9. 문자 인식 결과 후처리 과정  
Fig. 9. Text recognition result post-processing

위치를 나타낸다.

## Ⅶ. 실험 및 분석

각각의 실험에서 사용한 테스트 데이터는 학습 데이터에서 사용하지 않은 중고차 거래 사이트에서 확보한 데이터를 사용하였다. 각 실험에 따른 테스트 데이터 구성은 각각의 실험결과에 명시하였다. 또한 학습파라미터에서 학습횟수를 1 에폭으로 고정한 이유는 학습 시간 단축과 각 실험의 모델마다 동일한 조건으로 실험하기 위한 목적을 갖는다. 본 실험에서는 1 에폭의 값이 최적의 반복학습 횟수였으나, 각 테스트에 환경 및 학습 데이터 구성에 따라 1 에폭 이상의 반복횟수를 필요로 할 수 있다.

### 1. 차종 인식 성능

차량 전면 전체 영역과 그릴 영역을 나누어 학습하였다. 테스트 데이터는 36개의 클래스에 해당되는 차종을 각 50장씩 구성하여 총 1,800장을 기준으로 실험하였다. 그림 10은 테스트시 입력이미지를 Gray와 RGB로 변경하여 각각의 모델에 2가지 입력 이미지에 대한 결과를 나타낸다. 미세한 차이로 차량 전면 그릴 모델에 RGB 채널의 입력 이미지를 사용하였을 때 좋은 성능을 보인다. 이러한 결과는 그림 11과 같이 테스트 데이터에 차량 보닛(bonnet)이 개방된 이미지가 포함되어 있어, 정확도에 영향을 미치는 것으로 분석된다.

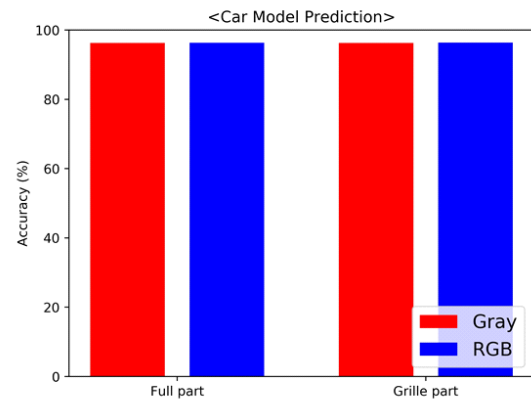


그림 10. 차종 인식 정확도  
Fig. 10. Car model recognition accuracy

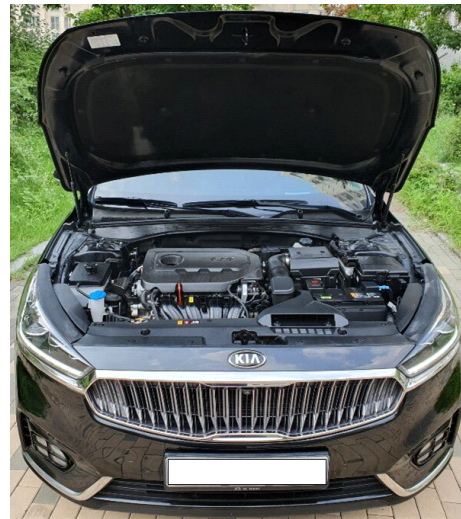


그림 11. 차종 인식 테스트 데이터(보닛 개방)  
Fig. 11. Car model recognition test data (bonnet open)

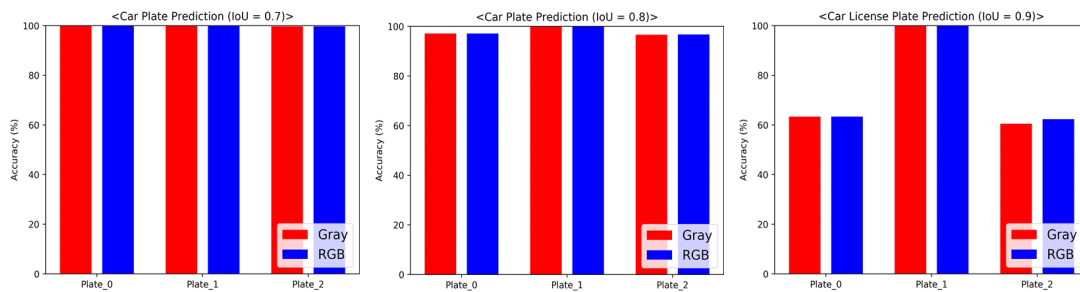


그림 12. 번호판 검출 정확도

Fig. 12. License plate detection accuracy

## 2. 번호판 검출 성능

앞서 정리한 표 3의 데이터 증강 파라미터에 따른 총 3개의 모델(Plate\_0, Plate\_1, Plate\_2)을 실험하였다. 테스트 데이터는 차종에 구분없이 총 5,000장의 차량 전면과 후면 데이터로 구성하였다. 테스트시 예측값과 정답값의 IoU (Intersection over Union) 임계값을 0.7, 0.8, 0.9로 설정하여 정확도를 측정하였다. 차종인식과 마찬가지로 입력이미지를 RGB, Gray로 변환하여 각각 성능 측정하였다. 그림 12는 IoU에 따른 번호판 검출 정확도를 나타낸다. IoU 0.7일 때, 입력이미지의 채널에 관계없이 미세한 차이로 Plate\_1 모델이 99.998%를 기록하여 가장 좋은 성능을 보인다. IoU 0.8일 때는 IoU 0.7과 동일하게 Plate\_1 모델이 99.996%를 기록하여 약간의 정확도가 하락되었지만 입력이미지의 채널과 관계없이 동일하게 높은 성능을 보였다. IoU 0.9일 때는 Plate\_1 모델이 다른 모델에 비해 월등하게 높은 성능을 보였으며, 입력이미지의 채널과 관계없이 동일한 정확도를 기록하였다. Plate\_2 모델에서 입력이미지가 RGB일 때, Gray 대비 약 2% 성능차이를 보였다.

## 3. 문자 인식 성능

특징점 매칭 기반 이미지 유사도 측정에서 상위 3개의 폰트를 선택하여, 가상데이터 생성 후 학습한 모델을 기준으로 정답값과 예측값의 결과를 비교하였다. 테스트 데이터는 IoU 0.9 기준 번호판 검출에서 정상적으로 검출된 데이터 중 3,000장의 이미지로 구성하였다. 그림 13은 문자 인식 성능을 나타낸다. 그림 7과 비례하여 유사도가 높은

폰트가 좋은 성능을 보인다. 그러나 현저한 차이를 보이지 않으며 3개의 모델은 76~79%의 정확도를 기록한다. 문자 인식의 경우, 입력이미지가 Gray일 때 RGB 대비하여 1% 정도의 정확도 향상이 있었다. 이러한 결과는 번호판 이미지가 흰색과 검은색으로 이루어져 있기 때문으로 분석된다. 또한 실제 데이터를 기반으로 학습한 모델(Real)은 가상 데이터 모델보다 높은 성능을 기록하였다. 이는 가상 데이터의 폰트가 실제 데이터의 폰트보다 유사도가 낮기 때문에 정확도가 하락된 것으로 파악된다.

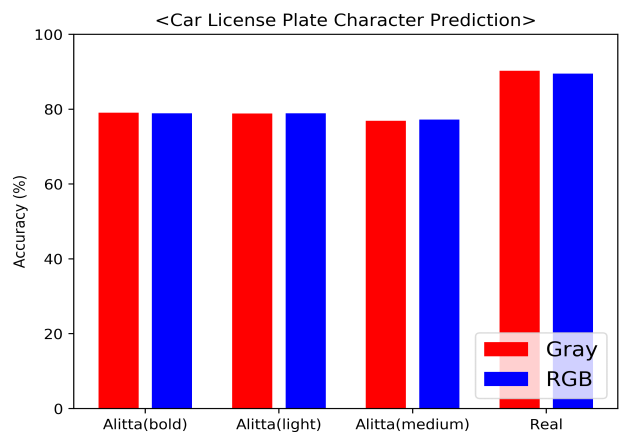


그림 13. 번호판 문자 인식 정확도

Fig. 13. License plate character recognition accuracy

## 4. 모델 경량화

앞서 실험한 각 모듈에 대하여 경량화 모델인 YOLO v3 tiny를 사용하여 정확도 및 모델 용량에 대하여 비교하였다. 표 6은 각 모듈별 실험 결과에서 가장 높은 정확도를 기록

표 6. 경량화된 모델 정확도 비교

Table 6. Lightweight model accuracy comparison

Model Name	YOLO v3 accuracy	YOLO v3 tiny accuracy	Model size (YOLO v3 / tiny)	Remarks
Grille part (RGB)	96.399%	91.071%	235MB / 33.4MB	Car model recognition
Plate_1 (Gray, IoU=0.9)	99.994%	30.525%	234MB / 33MB	Car license plate area detection
Alitta(bold, Gray)	79.067%	68.036%	235MB / 33.5MB	Car license plate character recognition

한 모델의 학습데이터를 사용하여, YOLO v3 tiny를 통한 학습한 결과의 정확도 값을 정리하였다. 본 절의 실험에서 YOLO v3 tiny를 사용한 번호판 검출의 경우 97.02%(IoU=0.7), 81.88%(IoU=0.8), 30.52%(IoU=0.9)를 기록하였다. IoU 0.9에서 낮은 검출 결과를 보였지만, 해당 모델의 검출 결과 영역에 5~10 화소를 더한 영역으로 번호판 영역을 추출하면 문자 인식 네트워크의 입력에 사용할 수 있다. 그러나 문자 인식 네트워크에서는 YOLO v3에 비교하여 YOLO v3 tiny의 정확도가 낮으므로 경량화 네트워크로 대체할 수 없을 것으로 분석된다.

## VIII. 결 론

본 논문에서는 가상 데이터를 활용한 번호판 문자 인식 및 차종 인식 시스템을 제안하였다. 차종 인식 모듈에서는 RGB 채널의 입력이미지에서 좋은 성능을 보이며, 번호판 검출 및 문자 인식 모듈에서는 Gray 채널의 입력이미지에서 좋은 성능을 보인다. 또한 문자 인식의 경우 특징점 매칭 기반 유사도 비교 결과를 이용하여 유사도 상위 3개 폰트를 가상 데이터로 생성하였다. 실험에서 특징점 매칭 기반 유사도 비교 결과와 폰트별 문자 인식 성능이 비례하는 것을 확인하였다. 마지막으로 임베디드 시스템 등의 저사양의 시스템 혹은 실제 서비스에 활용하기 위하여 경량화 네트워크인 YOLO v3 tiny를 이용하여 정확도를 측정하였다. 차종 인식 네트워크는 약 91%로 별도의 파라미터 변경 없이 사용이 가능하였다. 그러나 번호판 검출 모듈에서는 약 30.52%(IoU=0.9)를 기록하여, 객체 검출 좌표에서 5~10 화소(pixel) 정도 검출 결과 영

역을 넓히면 경량화 네트워크로 대체 가능하다. 그러나 문자 인식 모듈의 경우, 68.03%를 기록하여 YOLO v3 대비 약 11%의 성능 하락이 있었다. 따라서 추후 연구에서는 학습데이터를 고정 한 후 네트워크의 입력 크기, 네트워크 구조를 변경하여 해당 모듈의 네트워크를 개선하는 실험을 진행할 계획이다. 또한 문자 인식 정확도를 개선하기 위하여 높은 유사도를 갖는 폰트를 혼합하여 성능을 개선할 계획이다.

## 참 고 문 헌 (References)

- [1] Jung-Hwan Kim, Joonhong Lim, "License Plate Detection and Recognition Algorithm using Deep Learning," *Journal of IKEEE*, 2019, 23.2: 642-651, <http://dx.doi.org/10.7471/ikeee.2019.23.2.642>
- [2] Xuan-Truong Dang, Eung-Tae Kim, "Robust Motorbike License Plate Detection and Recognition using Image Warping based on YOLOv2," *JOURNAL OF BROADCAST ENGINEERING*, 2019, 24.5: 713-725, <https://doi.org/10.5909/JBE.2019.24.5.713>
- [3] REDMON, Joseph; FARHADI, Ali, "Yolov3: An incremental improvement," arXiv preprint arXiv:1804.02767, 2018.
- [4] GIRSHICK, Ross, et al, "Rich feature hierarchies for accurate object detection and semantic segmentation," *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 580-587, 2014.
- [5] GIRSHICK, Ross, "Fast r-cnn," *Proceedings of the IEEE international conference on computer vision*, pp. 1440-1448, 2015.
- [6] REN, Shaoqing, et al, "Faster r-cnn: Towards real-time object detection with region proposal networks," *Advances in neural information processing systems*, pp. 91-99, 2015.
- [7] HE, Kaiming, et al, "Mask r-cnn," *Proceedings of the IEEE international conference on computer vision*, pp. 2961-2969, 2017.
- [8] REDMON, Joseph, et al, "You only look once: Unified, real-time object detection," *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 779-788, 2016.
- [9] REDMON, Joseph; FARHADI, Ali, "YOLO9000: better, faster, stronger," *Proceedings of the IEEE conference on computer vision and pat-*

- tern recognition, pp. 7263-7271, 2017.
- [10] LIU, Wei, et al, "Ssd: Single shot multibox detector," *European conference on computer vision*, Springer, Cham, pp. 21-37, 2016, [https://doi.org/10.1007/978-3-319-46448-0\\_2](https://doi.org/10.1007/978-3-319-46448-0_2).
- [11] NEUMANN, Lukas; MATAS, Jiri, "Real-time scene text localization and recognition," *2012 IEEE Conference on Computer Vision and Pattern Recognition, IEEE*, pp. 3538-3545, 2012, doi:10.1109/CVPR.2012.6248097.
- [12] JADERBERG, Max, et al, "Reading text in the wild with convolutional neural networks," *International journal of computer vision*, 116.1: 1-20, 2016, doi:10.1007/s11263-015-0823-z.
- [13] GUPTA, Ankush; VEDALDI, Andrea; ZISSERMAN, Andrew, "Synthetic data for text localisation in natural images," *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 2315-2324, 2016.
- [14] LOWE, David G, "Distinctive image features from scale-invariant keypoints," *International journal of computer vision*, 60.2: 91-110, 2004, <https://doi.org/10.1023/B:VISI.0000029664.99615.94>.
- [15] RUBLEE, Ethan, et al, "ORB: An efficient alternative to SIFT or SURF," *2011 International conference on computer vision, Ieee*, pp. 2564-2571, 2011, doi:10.1109/ICCV.2011.6126544.
- [16] MUJA, Marius; LOWE, David G, "Scalable nearest neighbor algorithms for high dimensional data," *IEEE transactions on pattern analysis and machine intelligence*, 36.11: 2227-2240, 2014, doi:10.1109/CRV.2012.60.
- [17] Seungju Lee, Gooman Park, "Proposal for Deep Learning based Character Recognition System by Virtual Data Generation," *JOURNAL OF BROADCAST ENGINEERING*, 25.2: 275-278, 2020, <https://doi.org/10.5909/JBE.2020.25.2.275>.

## 저 자 소 개



### 이 승 주

- 2016년 2월 : 상명대학교 컴퓨터소프트웨어공학과 공학사
- 2018년 9월 ~ 현재 : 서울과학기술대학교 미디어IT공학과 석사과정
- ORCID : <https://orcid.org/0000-0001-8969-4296>
- 주관심분야 : 컴퓨터비전, Object Detection, Deep Learning



### 박 구 만

- 1984년 : 한국항공대학교 전자공학과 공학사
- 1986년 : 연세대학교 전자공학과 석사
- 1991년 : 연세대학교 전자공학과 박사
- 1991년 ~ 1996년 : 삼성전자 신호처리연구소 선임연구원
- 1999년 ~ 현재 : 서울과학기술대학교 전자미디어공학과 교수
- 2006년 ~ 2007년 : Georgia Institute of Technology Dept.of Electrical and Computer Engineering. Visiting Scholar
- 2016년 ~ 2017년 : 서울과학기술대학교 나노IT디자인융합대학원 원장
- ORCID : <https://orcid.org/0000-0002-7055-5568>
- 주관심분야 : 컴퓨터비전, 실감미디어