

# 차량 OCR 프로그램

실증적 AI 개발 프로젝트



팀: AICN

팀원: 김재민(팀장), 김동경, 이도현

차량 OCR 프로그램

# INDEX

01

프로젝트 요약

02

현재 진행상황

03

다음 회의 일정

# 차량 OCR 프로그램

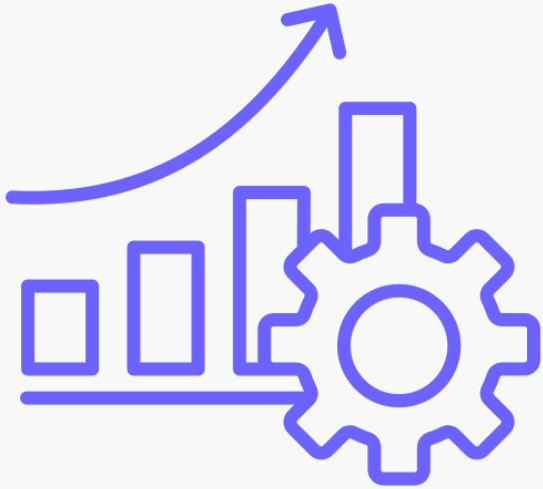
프로젝트 요약



정확도 높은 한국 차량번호 OCR 프로그램

# 차량 OCR 프로그램

현재 진행상황



성능 향상



직관적 UI



정보 보안

# 차량 OCR 프로그램

현재 진행상황

```
(.venv) gimjaemin@MacBook-Air EasyKoreanLpDetector-main % streamlit run server.py
```

메뉴

About

About

파일 업로드

결과 확인 및 수정

## 🚗 차량 번호판 자동 인식 시스템

### 📌 개요

이 웹 애플리케이션은 차량 이미지 또는 폴더(zip 파일)를 업로드하면 YOLO 모델과 OCR 기술을 활용해 차량 번호판을 자동으로 인식해주는 시스템입니다.

### 🔧 사용 방법

- 왼쪽 사이드바에서 '파일 업로드' 메뉴를 선택하세요.
- 이미지 또는 폴더를 ZIP 파일 형태로 압축하여 드래그 앤 드롭 방식으로 업로드합니다.
- 인식 결과 확인 및 수정은 왼쪽 메뉴의 '결과 확인 및 수정' 탭으로 이동하세요.

## 🚗 차량 번호판 자동 인식 시스템

### 📁 이미지 / ZIP 파일 업로드

파일을 이 영역에 드래그 앤 드롭 하세요. (PNG, JPG, JPEG, ZIP 지원, 최대 200MB)



Drag and drop files here

Limit 200MB per file • PNG, JPG, JPEG, ZIP

Browse files

인식 성공



기본 번호

161조7556

수정할 번호

인식 불가



기본 번호

저장할 엑셀 파일명을 입력하세요 (확장자 제외)

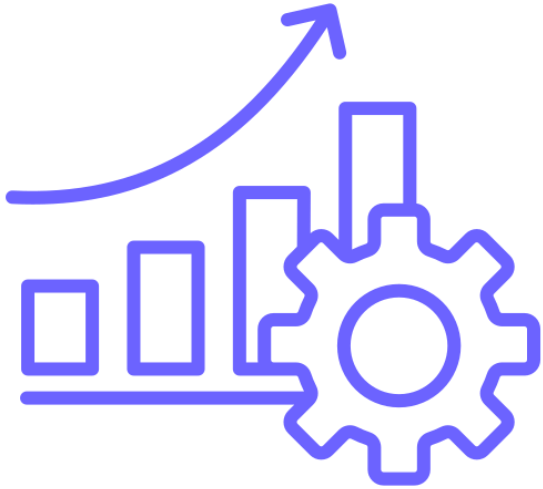
uploaded\_images\_info

엑셀로 저장

	A	B	C	D	E
1	파일명	파일 시간	이미지 경로	이미지 미리보기	차량 번호
2	images-2.jpeg	2025-04-13T23_22_10.615362	uploads/images-2.jpeg		161조7556
3	images-3.jpeg	2025-04-13T23_22_10.775921	uploads/images-3.jpeg		검출된 차 없음
4	images-4.jpeg	2025-04-13T23_22_10.878204	uploads/images-4.jpeg		검출된 차 없음
5	images.jpeg	2025-04-13T23_22_11.108498	uploads/images.jpeg		22누8433
6					
7					

# 차량 OCR 프로그램

현재 진행상황



성능 향상



직관적 UI



정보 보안

# 차량 OCR 프로그램

현재 진행상황

## 번호판 랜덤 생성

```
import os
import sys
import time
import json
import random
import argparse
import urllib.request

from tqdm import tqdm
from colorama import Fore, Style, Back
from PIL import Image, ImageDraw, ImageFont

# Hangul Classification Used for Vehicle License Plates
korean = '가나다라마바사' \
        '아디르래즈' \
        '국노루로복투수' \
        '국노루로복투수'

korean_taxi = '가나다라'
korean_rent = '바사라'
korean_parcel = '네'

os.makedirs('images', exist_ok=True)
new_img_path = 'images/src_plate_img/number_plate_new.png'
old_img_path = 'images/src_plate_img/number_plate_old.png'
# Proceed with the download if the background image is missing
if os.path.isfile(new_img_path) and os.path.isfile(old_img_path):
    pass
else:
    with open('images_url.json', 'r') as f:
        data = json.load(f)
        for fname in data:
            print(f'Download {Fore.GREEN + fname}(Style.RESET_ALL).png image file in images/.' )
            urllib.request.urlretrieve(data[fname], f'images/{fname}.png')

# Hangil font variable declare
hangil_font = ImageFont.truetype('fonts/Hangil.ttf', 100, encoding='unic')

# Noto Sans font variable declare
notosans_font = ImageFont.truetype('fonts/NotoSansKR-Medium.ttf', 120, encoding='unic')

def run(opts):
    count, save_path = opts.count, opts.save_path
    start = time.time()

    # Make folder to saving outputs
    os.makedirs(f'{save_path}/new', exist_ok=True)
    os.makedirs(f'{save_path}/old', exist_ok=True)
    os.makedirs(f'{save_path}/old7', exist_ok=True)
```

출처: [https://github.com/Oh-JongJin/Virtual\\_Number\\_Plate](https://github.com/Oh-JongJin/Virtual_Number_Plate)

번호판 이미지를 템플릿 기반으로 생성

-배경 이미지를 불러와 난수로 번호판 텍스트를 만듦.

-draw.text()로 번호판을 그림.

-번호판 텍스트에 맞춰 파일 저장.

-번호판 조합 다양성 무한대

-훈련 데이터 대량 증강 가능

## 전처리

AI-Hub 데이터 구성

— images/ ← 차량 전체 이미지  
— plates/ ← 번호판 이미지  
— annotations/ ← JSON 라벨

-images/cars/SUV/BMW 폴더에서 차량 이미지 읽기

-번호판 위치 자동 감지 → 번호판 이미지 (plates 폴더) 랜덤 선택

-OCR 최적화 전처리 후 번호판 삽입

-yolo \_ dataset/images/train

과 labels/train에 저장

-data.yaml 자동 생성 → 학습 준비 완료

data.yaml을 YOLO train.py에 넣고 실행  
→ 커스텀 모델 재학습

```
1 # full_pipeline.py: 번호판 생성 + 차량에 합성 + YOLO 학습 전처리까지 한 번에 실행
2 import os
3 import json
4 import random
5 import time
6 import argparse
7 import cv2
8 import numpy as np
9 from tqdm import tqdm
10 from PIL import Image, ImageDraw, ImageFont
11
12 # 번호판 생성 함수
13 def generate_license_plate_images(count, plate_dir):
14     os.makedirs(plate_dir, exist_ok=True)
15     korean = '가나다라마바사' \
16             '아디르래즈' \
17             '국노루로복투수' \
18             '국노루로복투수'
19     new_img_path = 'images/src_plate_img/number_plate_new.png'
20     font_hangil = ImageFont.truetype('fonts/Hangil.ttf', 100)
21     font_noto = ImageFont.truetype('fonts/NotoSansKR-Medium.ttf', 120)
22
23     for i in tqdm(range(count), desc='번호판 생성'):
24         front = f'{random.randint(100, 999)}'
25         middle = random.choice(korean)
26         back = f'{random.randint(1000, 9999)}'
27         full_name = front + middle + back
28
29         image_pil = Image.open(new_img_path)
30         draw = ImageDraw.Draw(image_pil)
31         draw.text((85, -20), front, 'black', font_noto)
32         draw.text((230, 35), middle, 'black', font_hangil)
33         draw.text((375, -20), back, 'black', font_noto)
34         image_pil.save(os.path.join(plate_dir, f'{full_name}.png'), 'PNG')
35
36 # 번호판 위치 감지 함수
37 def enhance_for_ocr(image):
38     gray = cv2.cvtColor(image, cv2.COLOR_BGR2GRAY)
39     thresh = cv2.threshold(gray, 0, 255, cv2.THRESH_BINARY + cv2.THRESH_OTSU)
40     kernel = np.array([[0, -1, 0], [-1, 5, -1], [0, -1, 0]])
41     sharpened = cv2.filter2D(thresh, -1, kernel)
42     return cv2.cvtColor(sharpened, cv2.COLOR_GRAY2BGR)
43
44 # 번호판 위치 감지 함수
45 def detect_plate_bbox(img):
46     gray = cv2.cvtColor(img, cv2.COLOR_BGR2GRAY)
47     blurred = cv2.bilateralFilter(gray, 11, 17, 17)
48     edged = cv2.Canny(blurred, 30, 200)
49     contours, _ = cv2.findContours(edged.copy(), cv2.RETR_TREE, cv2.CHAIN_APPROX_SIMPLE)
50     candidate_max_area = None, 0
```

# 차량 OCR 프로그램

현재 진행상황

## Easy OCR(현재 사용)

설치가 간편하고, Python 코드 몇 줄로 바로 OCR 수행 가능.

한국어 포함 80개 이상의 언어 지원. 한국어 텍스트 인식에 최적화된 언어 모델이 있음.

PyTorch 기반으로 되어 있어서 모델 수정이나 커스터마이징에 유리함.

PaddleOCR보다 상대적으로 경량이며, CPU에서도 속도 괜찮음.

차량번호판처럼 작은 영역에 고정된 포맷이 있는 경우, 정확도가 PaddleOCR보다 떨어질 수 있음.

## Paddle OCR

번호판처럼 포맷이 일정한 고정된 텍스트에는 뛰어난 정확도를 보여줌.  
다양한 모델 (PP-OCRv3, PP-OCRv4, PPOCR-Mobile 등) 제공.

경량 모델부터 대형 고정밀 모델까지 있어, 환경에 따라 선택 가능.

Detection → Recognition 흐름이 잘 짜여 있고, 레이아웃 분석까지 가능.

한국어 라벨이 포함된 모델도 학습 가능하고, 번호판처럼 특정 영역만 따로 훈련할 수 있음.

처음 사용할 때 설치 및 설정이 다소 번거롭고, 모델 크기에 따라 종속성도 많음.



**Paddle OCR 스터디 후 모델에 접목시킬 예정**





# 차량 OCR 프로그램

다음 회의 일정(4/17)



김동경

-AI-Hub 데이터 차량에 번호판을  
정확히 붙여 가공하는 방법 고안



김재민(팀장)

-전처리 전체 및 강화 코드 수정 및  
PaddleOCR 모델 스터디



이도현

-전처리 가공 및 데이터 증강

**감사합니다.**