

# 자동차 번호판 인식 Project

12121609 김대근

# 전체적인 과정

- R-CNN

1. sliding windows로 검출 영역 설정
2. 해당 영역에 CNN통과
3. 정확도가 기준 이상이 검출되는 영역을 저장
4. 영역들을 정확도를 기준으로 정렬하여 가장 정확도 높은 영역을 출력

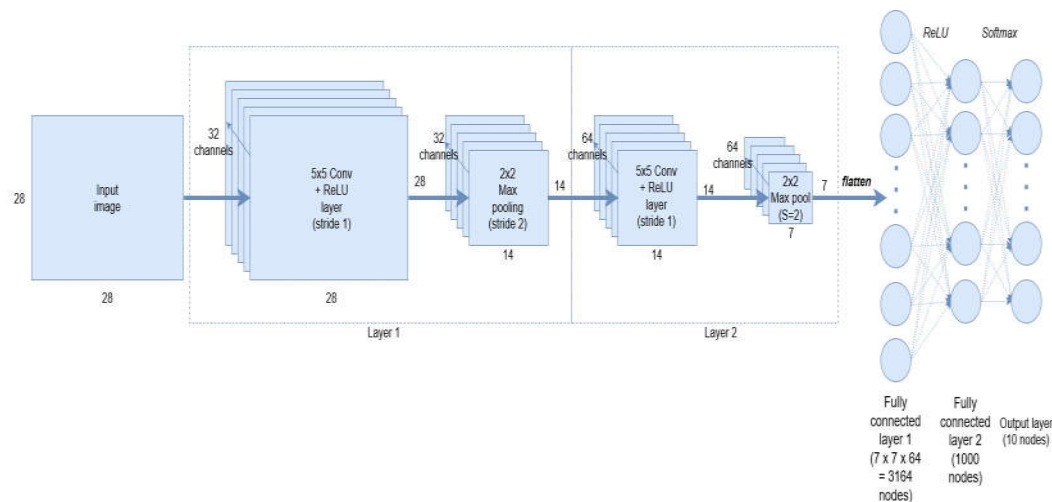
# Tensorflow Object Detection API

- Tensorflow의 object detection API
- Data Set을 TFRecord file format으로 변경
- Detected Box의 좌표값을 추출하는데 실패
- zero부터 설계를 해보기로 결정



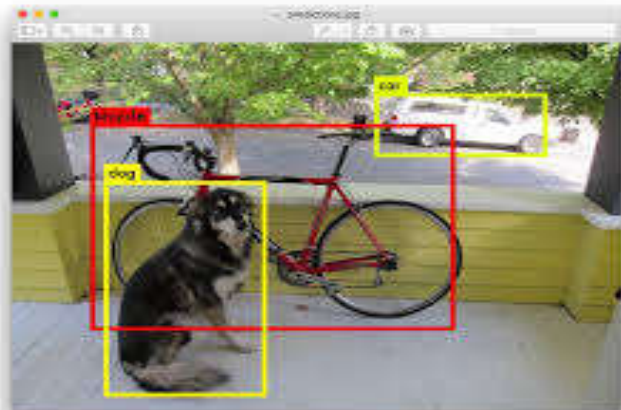
# 직접 신경망 설계

- 숫자를 훈련시키는 신경망 구성을 실습
- 번호판만 따로 잘라 이미지를 만들고 이를 class 0, 번호판 외의 이미지들은 class 1로 설정
- 이렇게 학습해도 결과가 좋지 않게 나왔다.



# Darknet yolo v2

- Darknet의 yolo API를 사용
- Data Set을 만드는 것은 tensorflow보다 용이
- 학습 진행 속도가 너무 느려서 학습 불가능 했다.



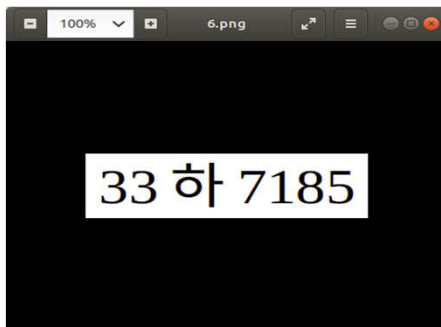
# opencv기반 영상처리로 번호판 인식

- Gaussian blur -> Canny -> findContours
- 만들어진 box의 비율과 넓이를 바탕으로 번호판 문자 아닌 것들은 제외
- 크기가 비슷한 박스들의 간격이 좁은 것들을 모아 번호판의 문자라고 생각하도록 한다



# 숫자 및 한글 인식 방법

- tesseract를 사용
- erosion과 dilation 적용하여 인식이 잘 되도록 했다.
- 번호판이 기울어진 경우에는 잘 인식되지 않았다.
- 완전히 깨끗한 이미지가 아니면 기본적으로 성능이 매우 좋지 않았다.



```
daegu@daegu-laptop: ~/Desktop
File Edit View Search Terminal Help
Warning: Invalid resolution 0 dpi. Using 70 instead.
Estimating resolution as 150
Detected 2 diacritics
(python3) daegu@daegu-laptop: ~/Desktop$ tesseract 4.png outputbase -l kor
tesseract Open Source OCR Engine v4.0.0-beta.1 with Leptonica
Warning: Invalid resolution 0 dpi. Using 70 instead.
Estimating resolution as 150
Detected 2 diacritics
(python3) daegu@daegu-laptop: ~/Desktop$ tesseract 2.png outputbase -l kor
tesseract Open Source OCR Engine v4.0.0-beta.1 with Leptonica
Warning: Invalid resolution 0 dpi. Using 70 instead.
Estimating resolution as 494
Detected 4 diacritics
(python3) daegu@daegu-laptop: ~/Desktop$ tesseract 6.png outputbase -l kor
tesseract Open Source OCR Engine v4.0.0-beta.1 with Leptonica
Warning: Invalid resolution 0 dpi. Using 70 instead.
Estimating resolution as 509
(python3) daegu@daegu-laptop: ~/Desktop$ outputbase.txt
outputbase.txt: command not found
(python3) daegu@daegu-laptop: ~/Desktop$ tesseract 6.png outputbase -l kor
tesseract Open Source OCR Engine v4.0.0-beta.1 with Leptonica
Warning: Invalid resolution 0 dpi. Using 70 instead.
Estimating resolution as 509
(python3) daegu@daegu-laptop: ~/Desktop$
```



# 테스트 결과

- Parking training data의 P1 Type의 번호판만을 이용하여 전처리하고 탐색하는 알고리즘을 만들어서 parking과 cctv 두 가지 경우에서 매우 좋지 않은 결과를 얻었다.

```
Number of bbox corrects: 65
Number of bbox examples: 285
Detection accuracy: 22.81

Number of recognition corrects: 47
Number of recognition examples: 285
Recognition accuracy: 16.49

Average pt: 575.37
Socre: -8.24
```

Parking

```
Number of bbox corrects: 9
Number of bbox examples: 451
Detection accuracy: 2.00

Number of recognition corrects: 0
Number of recognition examples: 436
Recognition accuracy: 0.00

Average pt: 825.02
Socre: -70.51
```

CCTV



# 고찰

- 성능 향상
  - 번호판 유형별 문자 표준 문자 위치를 이용
  - OCR을 Tesseract 대신 ABBYY를 이용
  - 기울어진 번호판의 수평을 맞춤
- 머신러닝의 장점
  - 획득한 이미지의 환경에 영향을 덜 받을 수 있다.
  - 물체의 특징을 스스로 학습하고 이용할 수 있다.
  - 기존의 영상처리 방식보다 정확도와 처리 속도를 많이 향상시킬 수 있다.

# 감사합니다

