

창업연계공학설계입문 2019 AD Project

이강민 이소영 이수아 조민수



Contents

1. 주제
2. 목표
3. 실행 환경
4. 구현 방식
5. 주행 영상
6. 개선 방안



표지판 인식 자율 주행



지형, 도로의 종류와 주변 환경에 따라
안전을 요구하는 표지판의 조건을 인식하는 자율주행



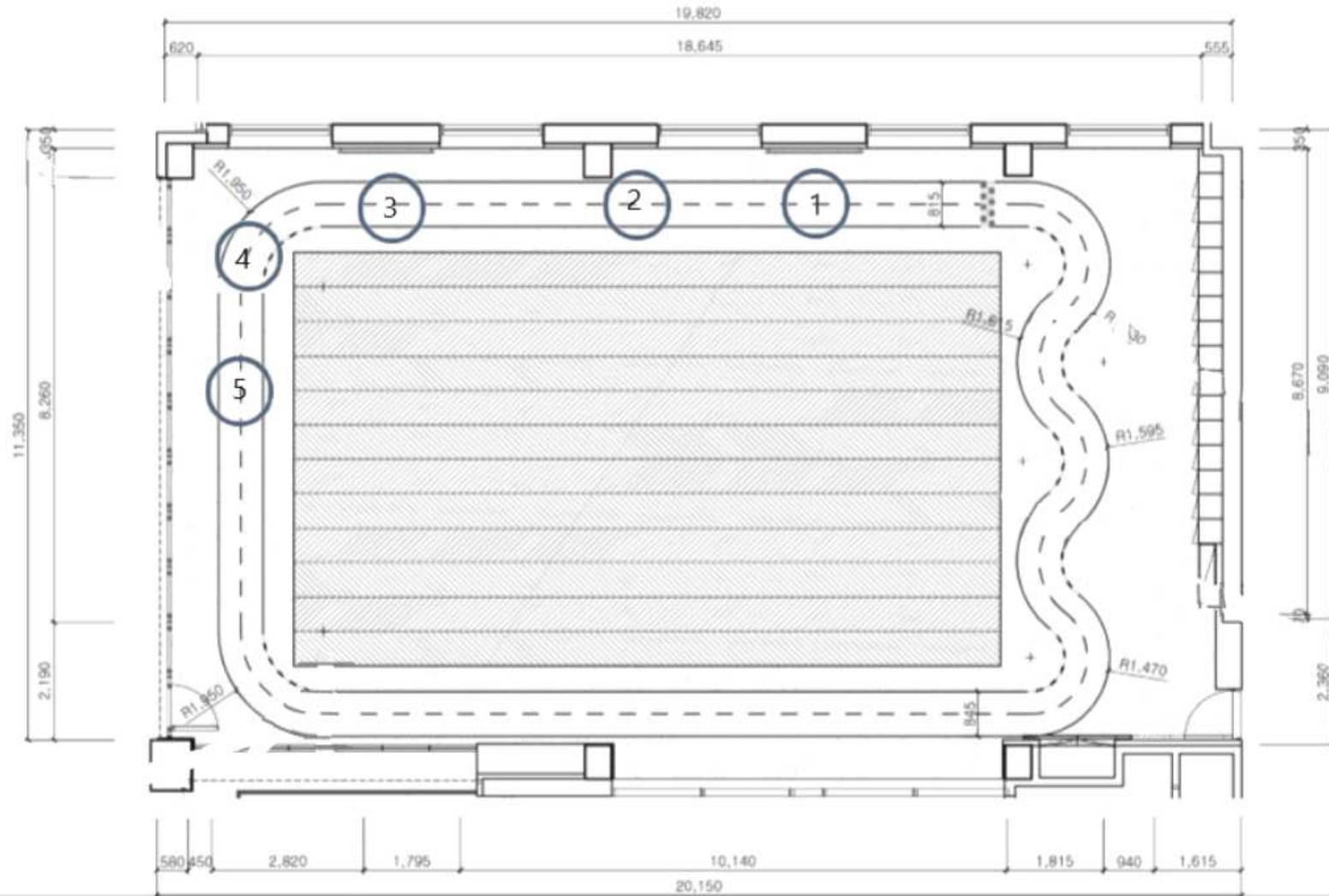
어린이보호구역 / 서행지역 / 단순 감속.가속.정지

표지판 인식 목표 설정

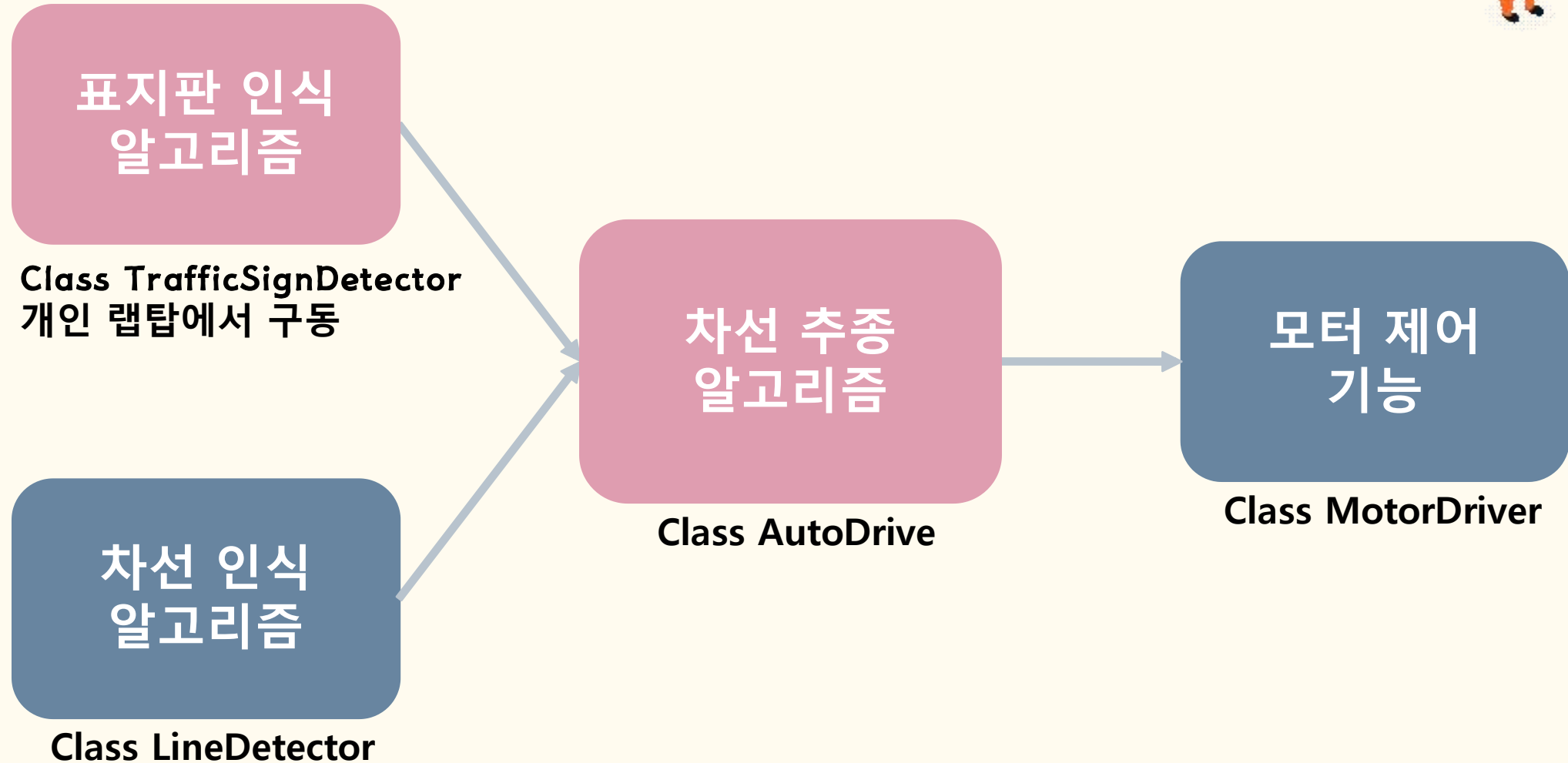


1. 처음 속도는 40으로 주행 시작
2. 첫번째 표지판 : 어린이보호구역 (제한속도 30)
 - 30의 속도까지 감속하며 주행
3. 두번째 표지판 : 전방 서행구역 (사고다발지역)
 - 30의 속도까지 감속하며 주행
4. 세번째 표지판 : 최저속도 50
 - 자율주행차는 50이상의 속도로 가속하여 주행
5. 네번째 표지판 : 최고속도 30 (사고다발지역)
 - 30의 속도까지 감속하며 주행
6. 마지막 표지판 : STOP 정지
 - 최종 주행을 멈춤

자율주행 스튜디오



프로그램 구조



Class TrafficSignDetector



- 'signalDetect' 노드 생성
- '/usb_cam/image_raw' 토픽 구독
- ROI 영역 -> 표지판 인식 (pytesseract 모듈 이용)
- 표지판 정보를 'signal' 토픽에 발행

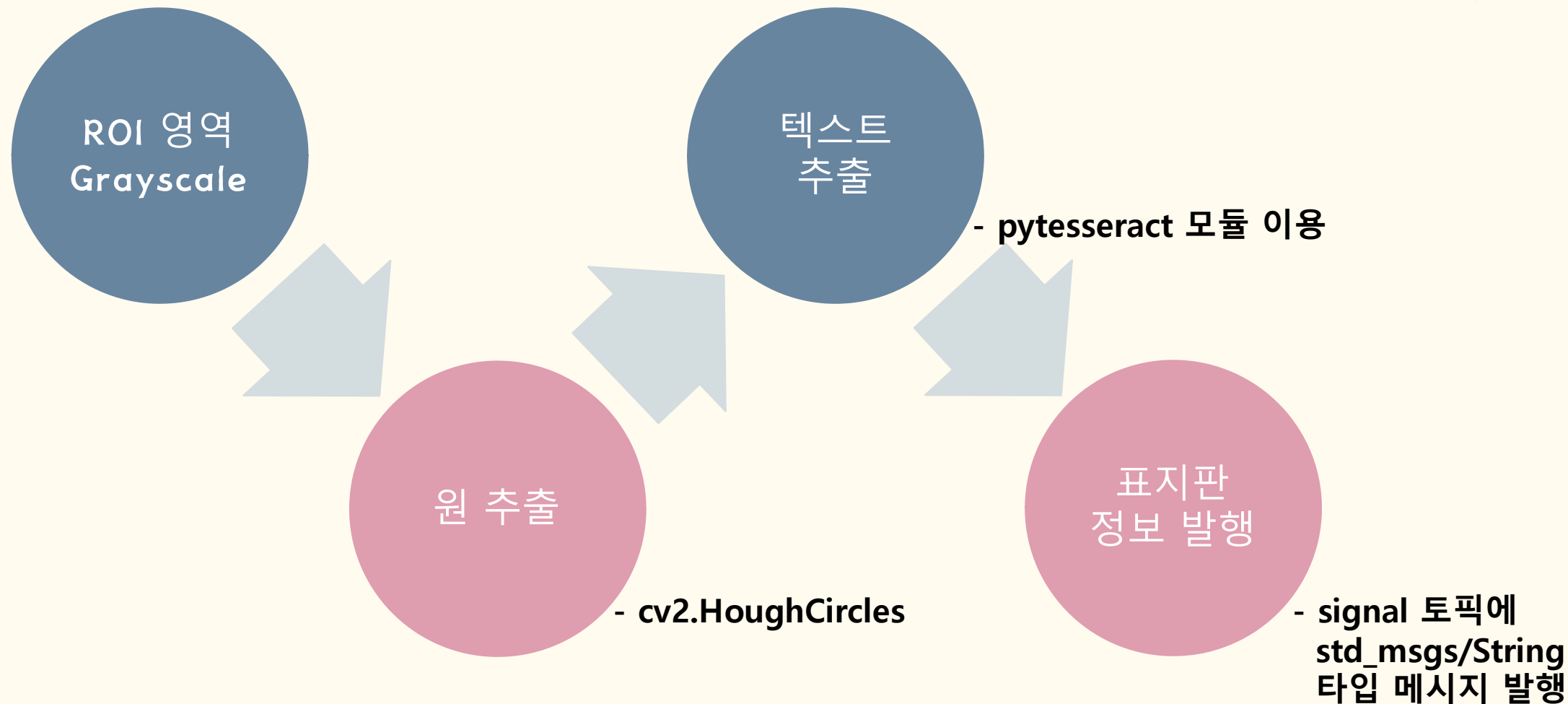
표지판 인식 구현



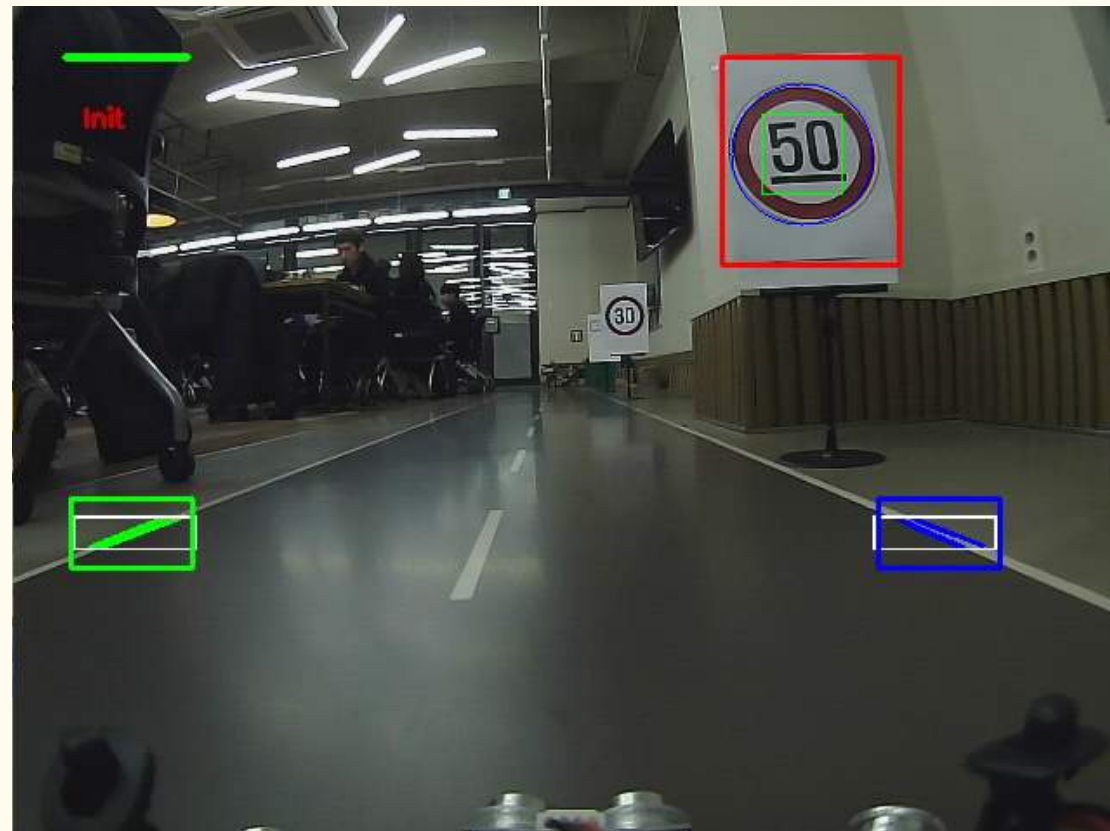
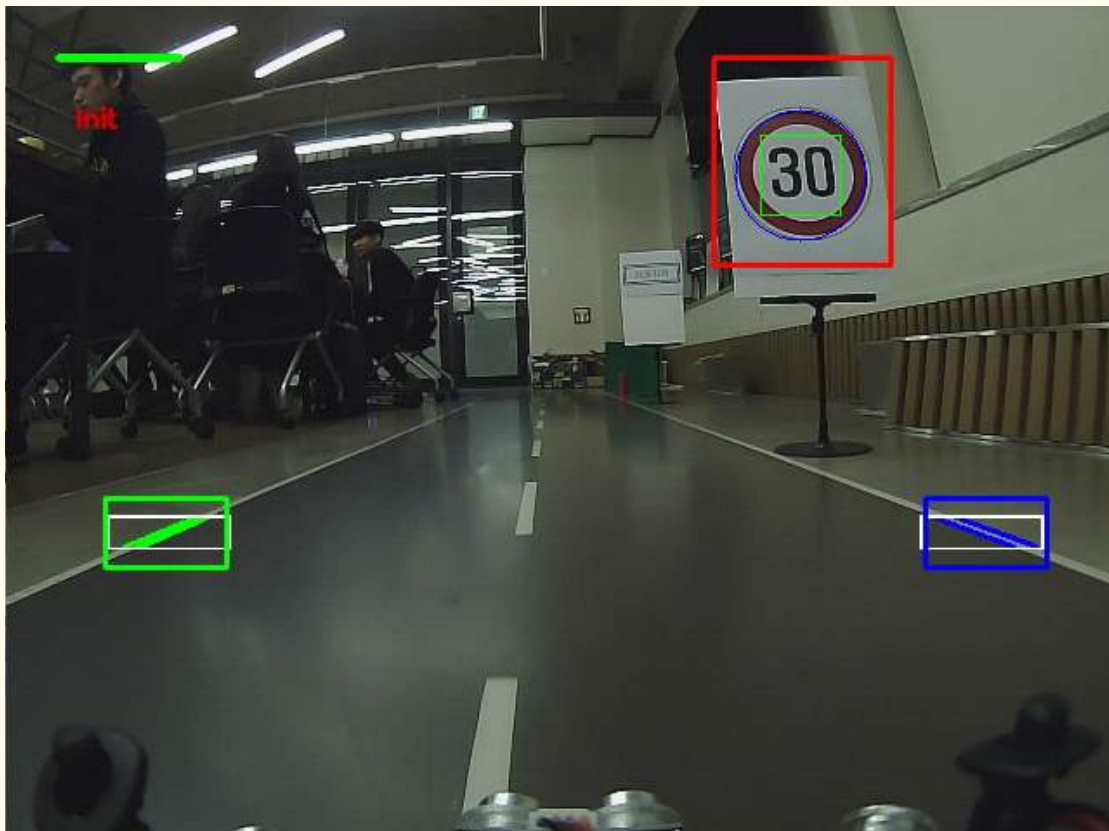
원형
표지판

그 외
표지판

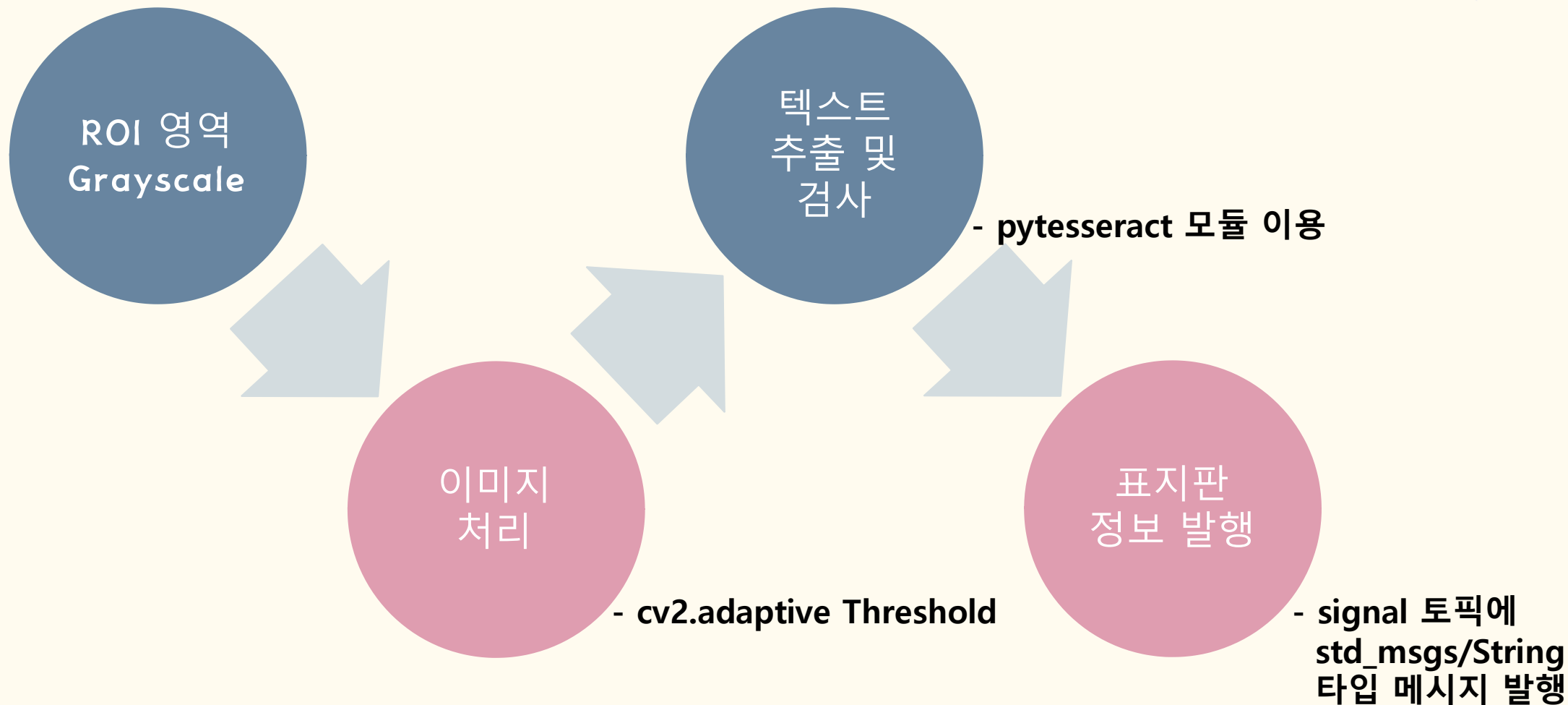
원형 표지판 인식



원형 표지판 인식



그 외 표지판 인식



그 외 표지판 인식



이미지 처리

‘HSV 색상 모델로 변경 후 이진화’, ‘Thresholding’, ‘Adaptive Thresholding’, ‘경계선 검출’ 네 가지 방식을 테스트

→ 가장 적합한 처리 방식은 Adaptive Thresholding



그 외 표지판 인식



글자 필터링

- 잘못 검출된 문자(특수문자, 공백)들을 걸러냄
- '천천히' 대신 '전전이', '찬참미' 등이 인식되어도 '천천히'로 인정

'천천히' 표지판 인식 로그

친미꼬꼬뽀
친미빠대|
친친미룸띠꺼
천천히튼따째
틱꼬다빠

'정지' 표지판 인식 로그

겉쳐뛰°쭈`
징지`흙칭0흥~`|
/참자흙칭0쭈
칩자엮`햏0쭈
참찌낙흙`0?칭/

그 외 표지판 인식



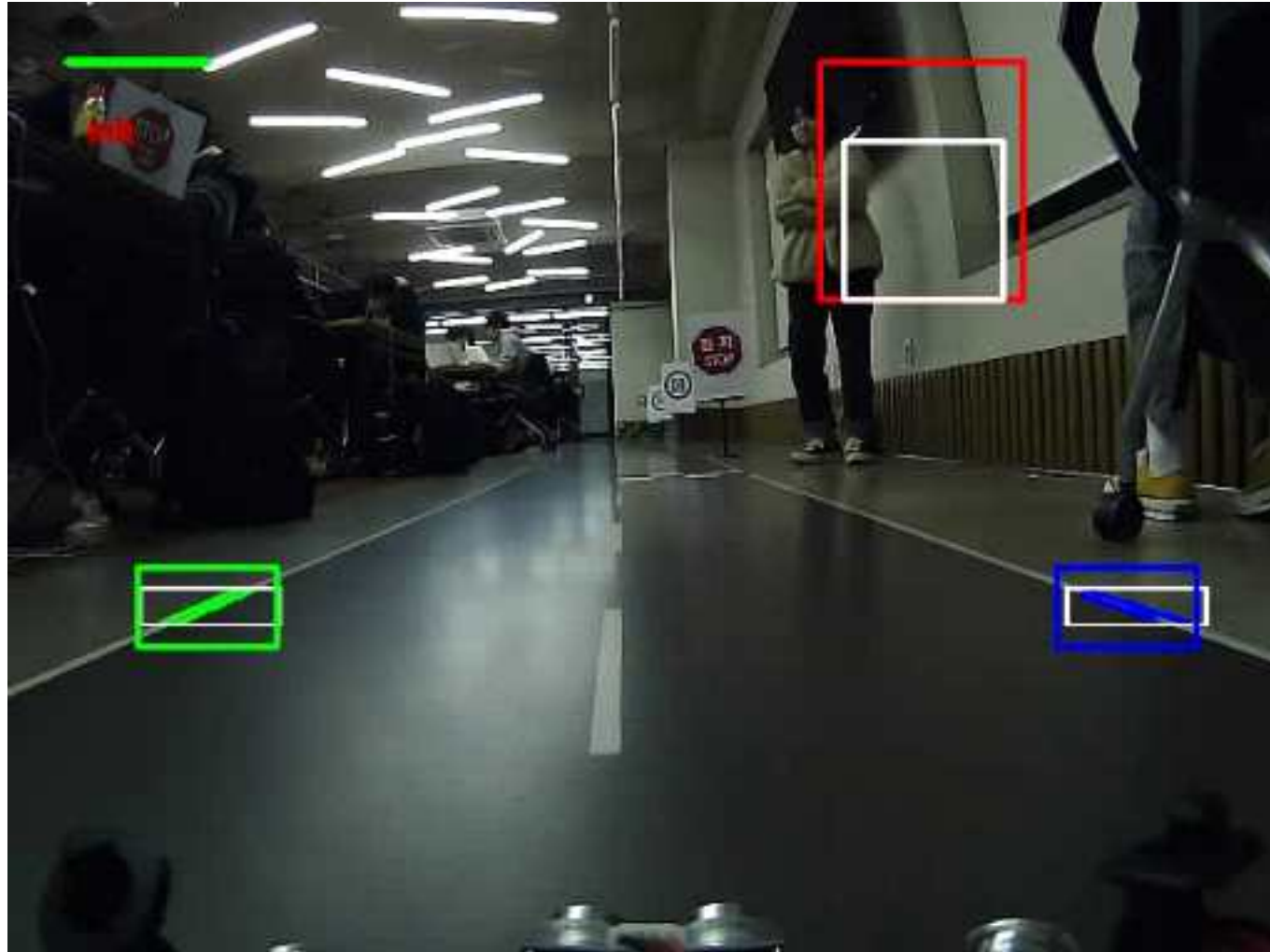
```
self.finds = []

stop = ("stop", [
    ['정', '징', '진', '점', '잠', '짐', '칭', '칭', '친', '침', '참', '침', '성', '섬', '심'],
    ['지', '자', '저', '치', '차', '처', '시']
])
self.finds.append(stop)

slow = ("slow", [
    ['천', '칭', '전', '친', '진', '징'],
    ['천', '칭', '전', '친', '진', '징'],
    ['히', '이', '어', '허', '미', '머']
])
self.finds.append(slow)

child = ("child", [
    ['어', '허', '이', '히'],
    ['린', '턴', '반', '런'],
    ['이', '미', '허'],
    ['보', '브', '부', '모', '므', '무'],
    ['호', '흐', '후']
])
self.finds.append(child)
```

그 외 표지판 인식



5. 결과

주행 영상



개선 방안



1. 머신러닝을 활용한 다양한 표지판 인식

pyteseract의 한계점 → 옰로(yolo)로 그림이나 기호 인식

2. 예외처리

우측 횡단보도의 신호등이 파란신호일때, 우회전 주행 교통법

3. 인식을 향상

자율주행스튜디오 → 실제 도로상황 고려 (날씨/일조량)

END

