

aParkings-AI

인턴 김수빈

인턴 이재혁

INDEX

1. 프로젝트 소개
2. 진행 일정
3. 적용 기술
4. 시나리오
5. 결과
6. 개선 사항

1. 프로젝트 소개

■ aParkings-AI 프로젝트 소개

■ 주차 구역에 대한 정보 제공

- 주차 구역 내 자동차의 유무를 파악한 후 그에 대한 정보를 UI로 제공

■ 주차 위치를 확인 할 수 있는 QR 코드 제공

- 주차장 내 기둥에 부착된 QR코드를 통해 위치에 대한 이미지 제공

■ 목표

■ 불필요한 시간 절약

- ▶ 주차 구역을 찾는 시간 감소
- ▶ 주차 위치를 찾는 시간 감소

2. 진행 일정

■ 일정표

1 / 27.	1 / 28.	1 / 29.	1 / 30.	1 / 31.
설날 연휴.	승희 : + - 여러 명 사람 인식. 재혁 : + - 주차면 인식. 수빈 : + - QR 위치를 도면 위에 표시.	승희 : + - 여러 명 사람 인식. 재혁 : + - 주차면 인식. 수빈 : + - QR Scanner가 도면을 받아와서 Activity 위에 이미지 표시.	승희 : + - 사람 인식에 대한 정확도 향상. 재혁 : + - 주차면 위의 자동차 인식. 수빈 : + - QR Scanner가 도면을 받아와서 Activity 위에 이미지 표시.	승희 : + - 사람 인식에 대한 정확도 향상. 재혁 : + - 주차면 위의 자동차 인식. 수빈 : + - QR Scanner가 도면을 받아와서 Activity 위에 이미지 표시.
..	승희 : + - User table (인식된 시간, 횟수) 코드 연구. 재혁&수빈 : + - 주차장 도면 프로그래밍. (SenTerm 사진으로 입력 받아 도면 위 표시). - SenTerm 과 Jetson Nano 통신.	승희 : + - User table (인식된 시간, 횟수) 코드 연구. 재혁&수빈 : + - 주차장 도면 프로그래밍. (SenTerm 사진으로 입력 받아 도면 위 표시). - SenTerm 과 Jetson Nano 통신.	승희 : + - 얼굴인식시 카메라 자동 촬영 코드 연구. 재혁&수빈 : + - 주차장 도면 프로그래밍. (SenTerm 사진으로 입력 받아 도면 위 표시). - SenTerm 과 Jetson Nano 통신.	승희 : + - 얼굴인식시 카메라 자동 촬영 코드 연구. 재혁&수빈 : + - 주차장 도면 프로그래밍. (SenTerm 사진으로 입력 받아 도면 위 표시). - SenTerm 과 Jetson Nano 통신.
2 / 10.	2 / 11.	2 / 12.	2 / 13.	2 / 14.
승희 : + - SenTerm 과 Jetson Nano 통신. 재혁&수빈 : + - aParkings-AI 애플리케이션 개발.	승희 : + - SenTerm 과 Jetson Nano 통신. 재혁&수빈 : + - aParkings-AI 애플리케이션 개발.	승희 : + - aSafes-AV SenTerm camera test. 재혁&수빈 : + - aParkings-AI 애플리케이션 개발.	승희 : + - aSafes-AV SenTerm camera test. 재혁&수빈 : + - aParkings-AI 애플리케이션 개발.	승희 : + - aSafes-AV 완성 목표. 재혁&수빈 : + - aParkings-AI 애플리케이션 완성 목표.
2 / 17.	2 / 18.	2 / 19.	2 / 20.	2 / 21.
2 / 24.	2 / 25.	2 / 26.	2 / 27.	2 / 28.
문서 작업.	문서 작업.	문서 작업.	문서 작업.	문서 작업.

2. 진행 일정

■ 1월

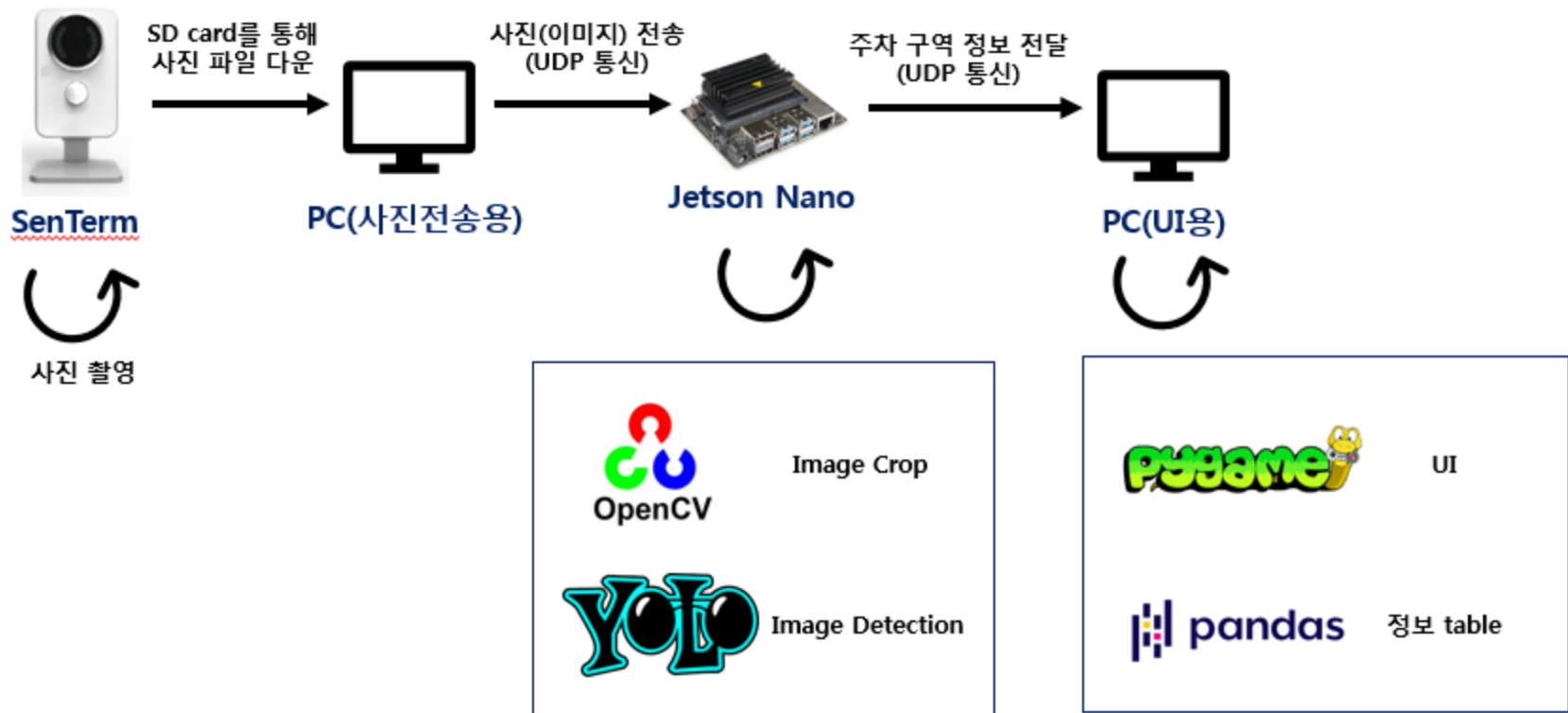
- SenTerm에 대한 이해
 - USN 시스템 기초 실습 및 응용
- Jetson Nano에 대한 조사 분석 (오픈소스 참고, 동영상 시청 등)
- 개발을 위한 기초 지식 습득 (Yolov3, OpenCV 등)
- 개발 환경 세팅
 - 전 이현우 인턴 자료 참고 및 추가적인 자료는 문서에 작성

■ 2월

- 주차 도면 및 QR code에 대한 UI 생성
- OpenCV를 활용한 image crop 개발
- Jetson과 PC 사이의 socket 통신 구축 (TCP → UDP 변경)
- Yolov3를 활용한 object detection 개발
- Pygame를 활용한 UI 개발
- Pandas를 활용하여 정보 테이블 생성

3. 적용 기술

■ aParkings-AI 흐름도



3. 적용 기술

■ 적용 기술

■ UDP Socket 통신

■ PC → Jetson Nano

- ▶ SenTerm에서 촬영한 이미지를 SD카드를 통해 PC에 입력 후 PC를 통해 Jetson Nano로 전송

■ Jetson Nano → PC

- ▶ 이미지 처리에 대한 결과(주차 구역 내 주차 유무에 대한 정보)를 UI용 PC로 전송

■ 이미지 처리 (Jetson board)

■ OpenCV

- ▶ 이미지의 특정 위치(주차 구역)를 자른 이미지 파일을 Jetson nano의 crop_img directory에 저장

■ Yolov3

- ▶ crop_img directory에 이미지 파일이 자동차인지 판단하여 자동차 유무 결정
- ▶ 자동차 유무에 대한 리스트 저장

3. 적용 기술

■ 적용 기술

■ UI (PC)

■ OS

- ▶ Directory에 저장된 file list를 불러오기 위해 사용한 파이썬 모듈

■ Pygame

- ▶ 주차 구역에 대한 정보(자동차 유무)가 표시된 주차 도면을 스크린으로 출력하기 위해 사용한 모듈

■ Pandas

- ▶ 주차장 정보 테이블을 데이터 프레임으로 출력하기 위해 사용한 모듈

■ Datetime

- ▶ 자동차가 인식된 시간을 저장하기 위해 사용한 파이썬 모듈

■ QR Code

- QR code 생성을 도와주는 파이썬 모듈

4. 시나리오

■ aParkings-AI

- 운전자(user)에게 주차장 도면을 통해 빈 주차 구역을 보여준다.
(초록색 : 주차 가능, 빨간색 : 주차 불가능)



- 운전자가 빈 자리에 주차하면 일정 시간 후 주차장 도면에 표시된다.



4. 시나리오

■ aParkings-AI

- 주차장 도면에 자동차 유무가 반영되면 주차장 정보 테이블을 출력한다.

주차 가능: 143 / 152				
	주차구역	입차		출차
0	0	2020-02-25 14:46:15	2020-02-25 14:51:41	
1	4	2020-02-25 14:46:15		0
2	7	2020-02-25 14:46:15		0
3	8	2020-02-25 14:46:15	2020-02-25 14:53:42	
4	10	2020-02-25 14:46:15	2020-02-25 14:53:42	
5	13	2020-02-25 14:46:15	2020-02-25 14:53:42	
6	15	2020-02-25 14:46:15	2020-02-25 14:51:41	
7	22	2020-02-25 14:46:15	2020-02-25 14:53:42	
8	23	2020-02-25 14:46:15		0
9	1	2020-02-25 14:51:41		0
10	2	2020-02-25 14:51:41		0
11	5	2020-02-25 14:51:41		0
12	12	2020-02-25 14:51:41	2020-02-25 14:53:42	
13	9	2020-02-25 14:53:42		0
14	11	2020-02-25 14:53:42		0
15	24	2020-02-25 14:53:42		0

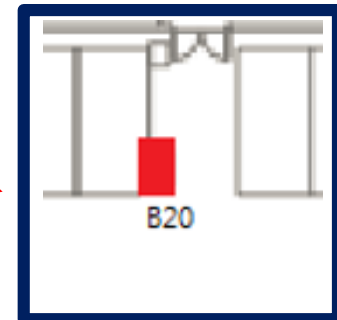
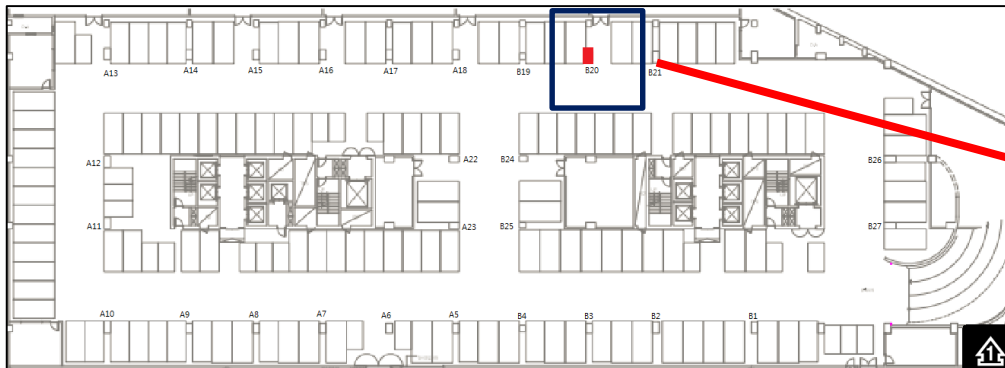
4. 시나리오

■ QR code

- 운전자가 주차 후 근처 기둥에 부착된 QR 코드를 휴대폰으로 촬영



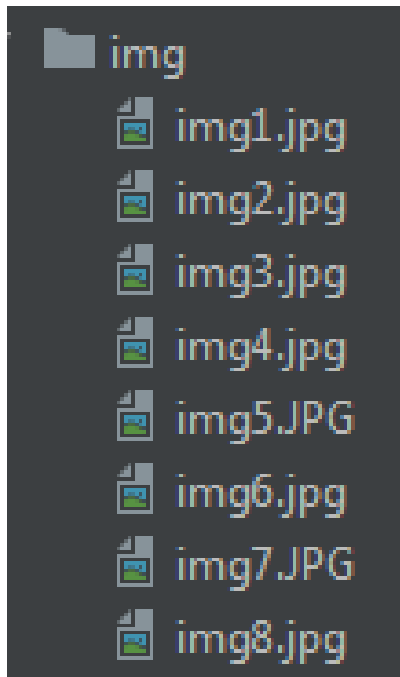
- 운전자의 휴대폰에 주차 위치 정보를 담은 도면 출력



5. 결과

■ PC(Client)가 Jetson Nano(Server)로 SenTerm에서 촬영한 이미지 전송

- SenTerm에서 촬영한 사진을 저장

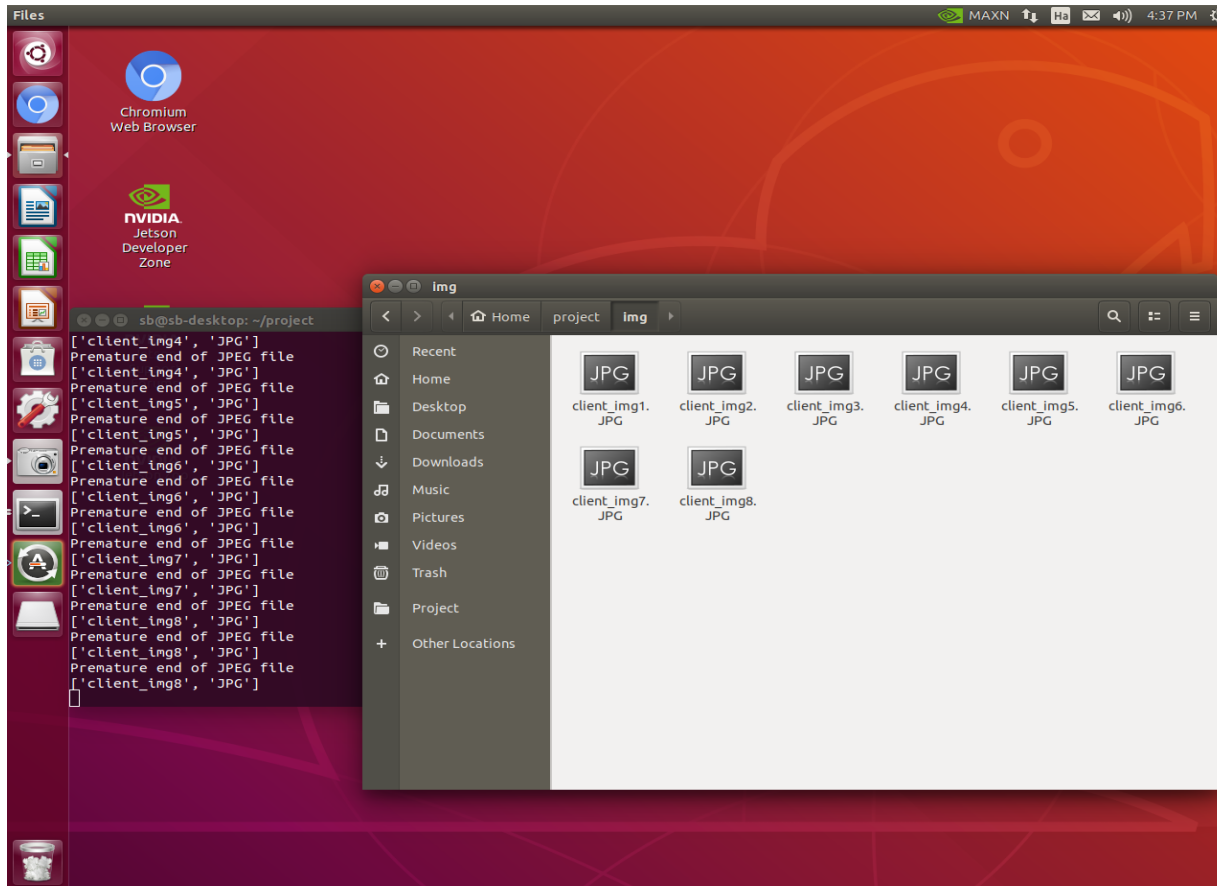


- PC(Client)에서 Jetson Nano(Server)로 이미지 전송

```
img1.jpg
end
img2.jpg
end
img3.jpg
end
img4.jpg
end
img5.JPG
end
img6.jpg
end
img7.JPG
end
img8.jpg
end
```

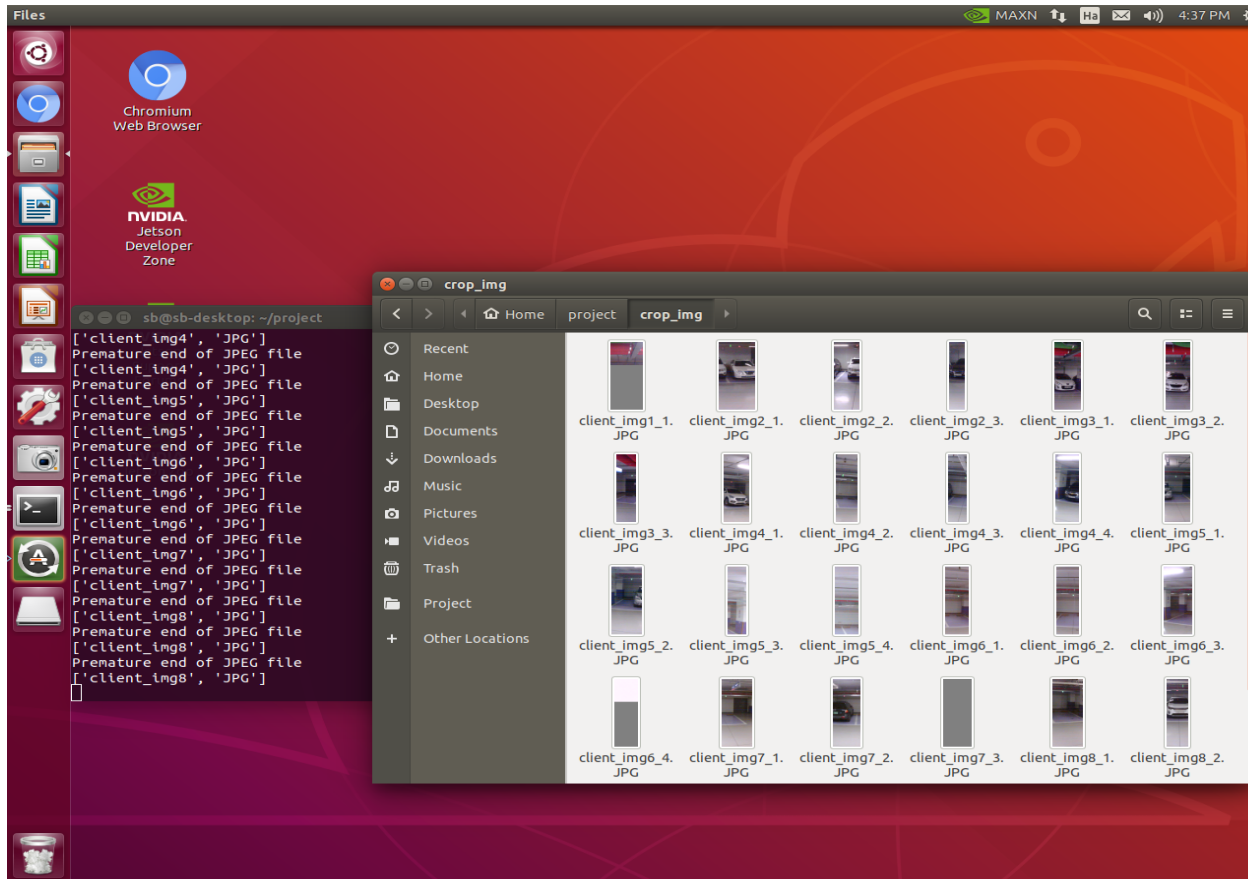
5. 결과

- Jetson Nano(Server)는 PC(Client)가 보낸 이미지를 img directory에 저장



5. 결과

- Jetson Nano(Server)에서 이미지를 주차 구역에 맞게 자른 후 crop_img에 저장



5. 결과

- UI PC(Client)가 'start'를 Jetson Nano(Server)로 보내면 Server는 Yolo 수행 후 주차 상황에 대한 정보를 UI Client로 전송

```
b'start'
client_img2_1.JPG car
client_img2_1.JPG car
client_img2_2.JPG car
client_img2_2.JPG car
client_img3_1.JPG car
client_img3_2.JPG car
client_img4_1.JPG car
client_img4_3.JPG car
client_img5_1.JPG car
client_img8_2.JPG car
client_img8_3.JPG car
[0, 1, 1, 0, 1, 1, 0, 1, 0, 1, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 0, 0
, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0]
```

5. 결과

- Jetson Nano(Server)로부터 정보를 받은 UI PC(Client)는 pygame을 이용해 주차장 도면에 정보 표시

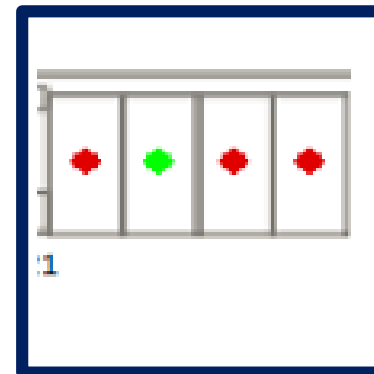


5. 결과

■ aParkings-AI

■ HPB 실내 주차장 예시 상황

■ 7,8,10번 구역 입차 (1.29 촬영)

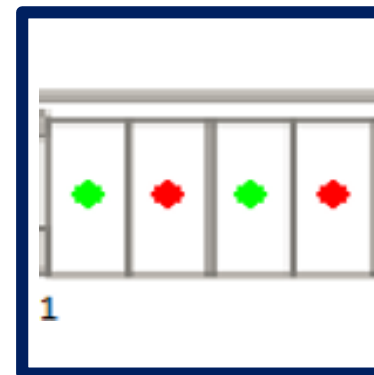
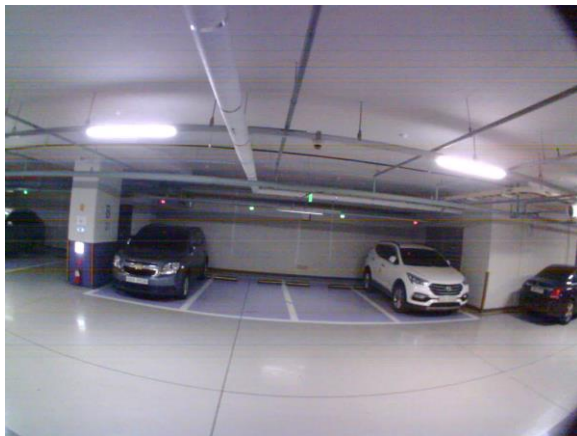


5. 결과

■ aParkings-AI

■ HPB 실내 주차장 예시 상황

■ 8,10번 구역 출차, 9번 구역 입차 (1.29 촬영)



5. 결과

- Server로부터 정보를 받은 UI Client는 pandas를 이용해 테이블로 정보 출력

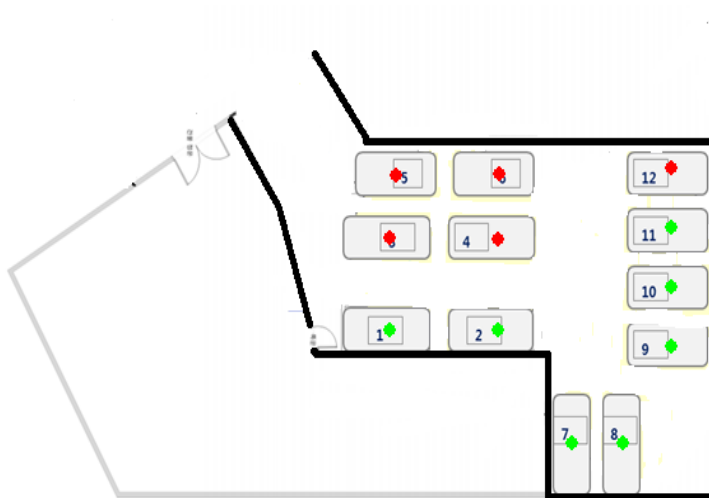
주차 가능: 143 / 152

	주차구역	입차	출차
0	0 2020-02-25 14:46:15	2020-02-25 14:51:41	
1	4 2020-02-25 14:46:15		0
2	7 2020-02-25 14:46:15		0
3	8 2020-02-25 14:46:15	2020-02-25 14:53:42	
4	10 2020-02-25 14:46:15	2020-02-25 14:53:42	
5	13 2020-02-25 14:46:15	2020-02-25 14:53:42	
6	15 2020-02-25 14:46:15	2020-02-25 14:51:41	
7	22 2020-02-25 14:46:15	2020-02-25 14:53:42	
8	23 2020-02-25 14:46:15		0
9	1 2020-02-25 14:51:41		0
10	2 2020-02-25 14:51:41		0
11	5 2020-02-25 14:51:41		0
12	12 2020-02-25 14:51:41	2020-02-25 14:53:42	
13	9 2020-02-25 14:53:42		0
14	11 2020-02-25 14:53:42		0
15	24 2020-02-25 14:53:42		0

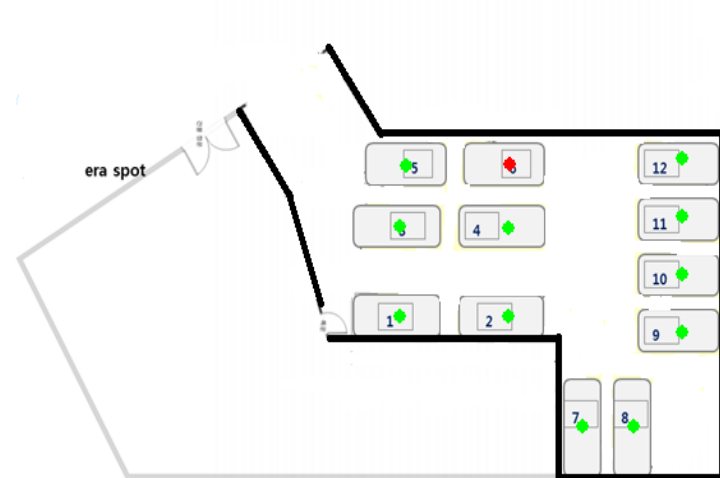
5. 결과

■ JWB 실외 주차장 실험

1번 사진 상황



2번 사진 상황



5. 결과

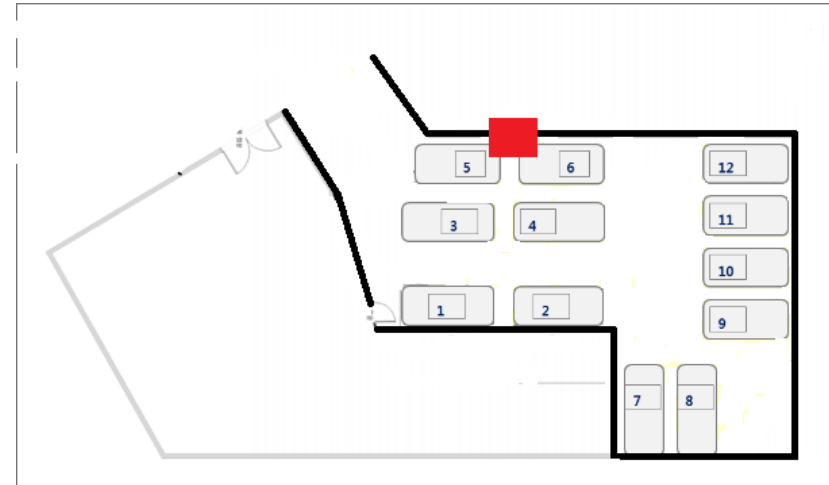
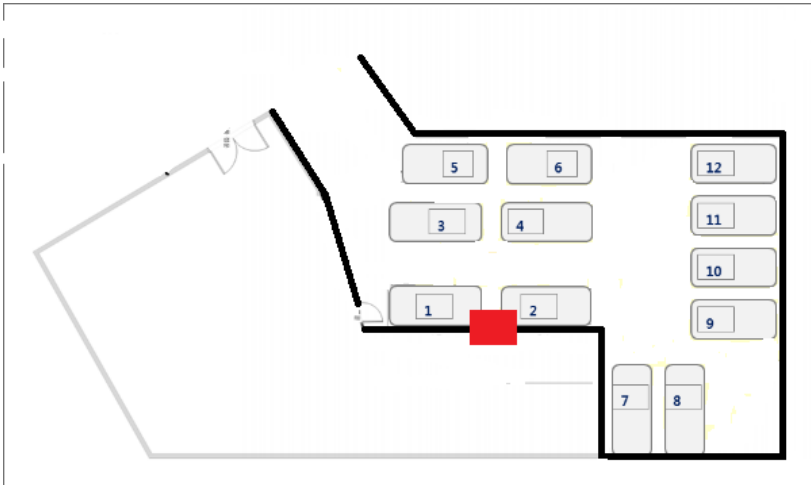
■ JWB 주차장 실험

■ 테이블

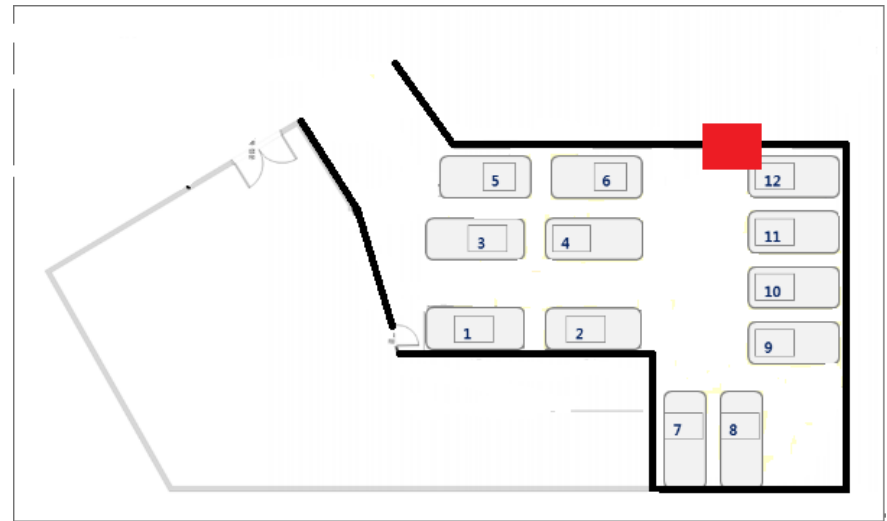
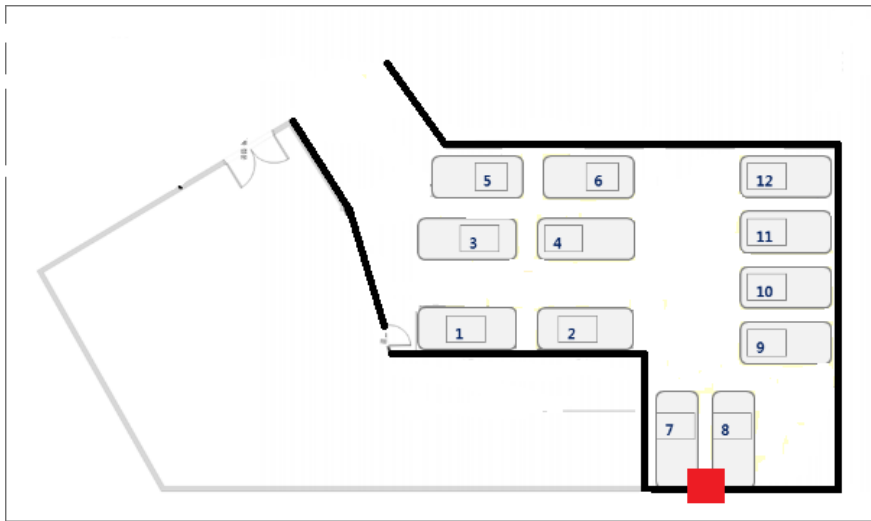
주차 가능: 11 / 12

주차구역				입차	출차
0	2	2020-02-27 11:05:37		2020-02-27 11:08:08	
1	3	2020-02-27 11:05:37		2020-02-27 11:08:08	
2	4	2020-02-27 11:05:37		2020-02-27 11:07:19	
3	5	2020-02-27 11:05:37			0
4	11	2020-02-27 11:05:37		2020-02-27 11:07:19	

5. 결과



5. 결과



6. 개선 사항

■ 개선 사항

■ SenTerm과 Jetson Nano의 UDP 통신 구축

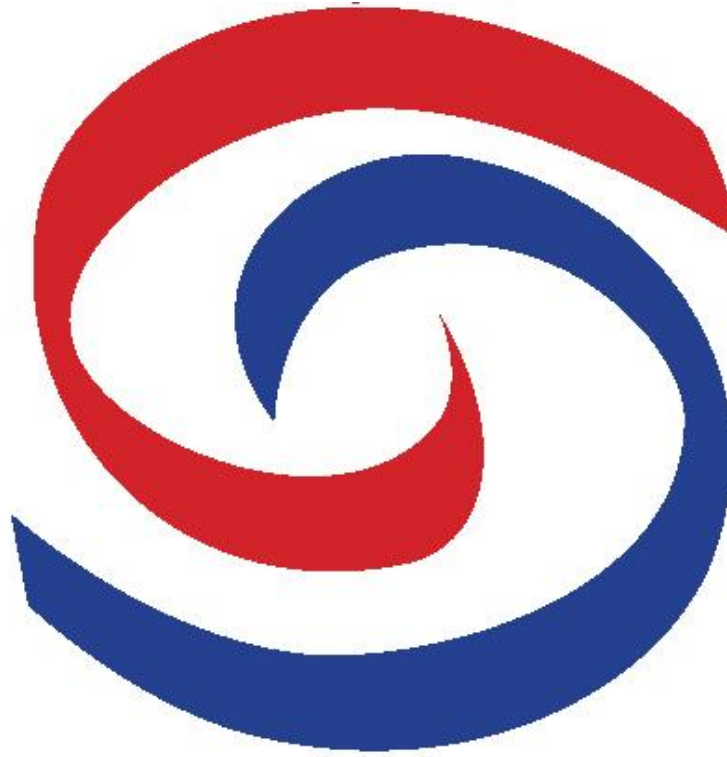
- 현재 SenTerm에서 촬영한 사진을 SD card를 통해 PC로 옮긴 후 전송
- C와 Python 간의 통신 또는 현재 짜여진 server의 코드를 C로 변경

■ QR 코드 테이블

- QR 코드를 촬영하면, 촬영한 순서대로 데이터 기록(Queue 형식)

■ Yolov3의 Object detection 인식률 개선 필요

- 야외 주차장의 경우 주차구역이 불규칙적으로 지정되어 있기 때문에 규칙적인 실내 주차장의 사진들과 비교 했을 때 인식률이 저조



감사합니다