

STLS : Smart Traffic Light System



2nd Implementation



201411285 유종혁



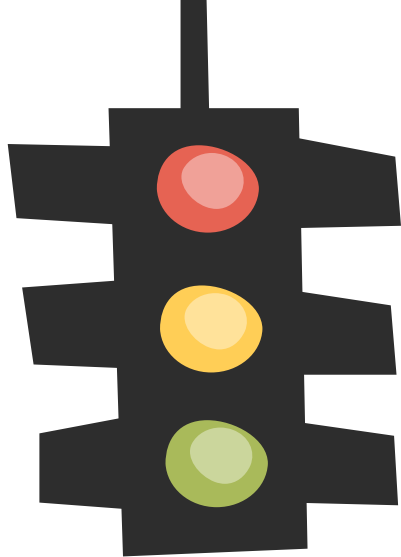
201411286 유환규



201611308 최준오



201714166 신예슬



STLS : Smart Traffic Light System

2nd Implementation



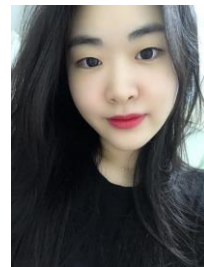
201411285 유종혁



201411286 유환규

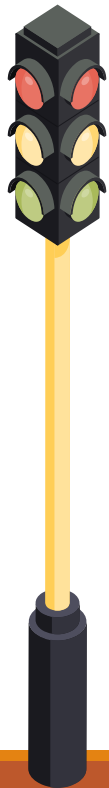


201611308 최준오



201714166 신예슬

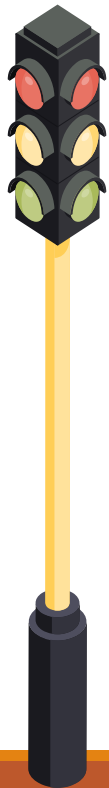
목차



1. 프로젝트 소개
2. 프로젝트 구조
3. 추적성 분석
4. System Test & Pass/Fail Criteria
5. 데모 시나리오
6. 데모 영상

1. 프로젝트 소개

바쁜데 신호등은 언제 켜지는 거야



기존 도로 신호등 시스템 :

기존 도로 신호 시스템은 항상 정해진 스케줄 대로 운영된다.
(초록불 1분 30초, 좌회전 신호 2분 등)

따라서 실시간으로 변하는 도로상황에 대응하지 못하기 때문에 비효율적이다.

대기하는 차량이 많은 곳에 신호를 우선적으로 줄 순 없을까?



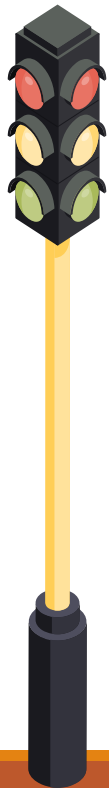
딥러닝 기술을 이용해서 만들어 볼까?



실시간 신호등 시스템 :

딥러닝 기술을 이용해서 실시간으로 차량의 존재 여부 뿐만 아니라 대수를 파악하고 그에 따라 신호체계를 동적으로 변경하여 효율적인 교통체계를 구축한다.

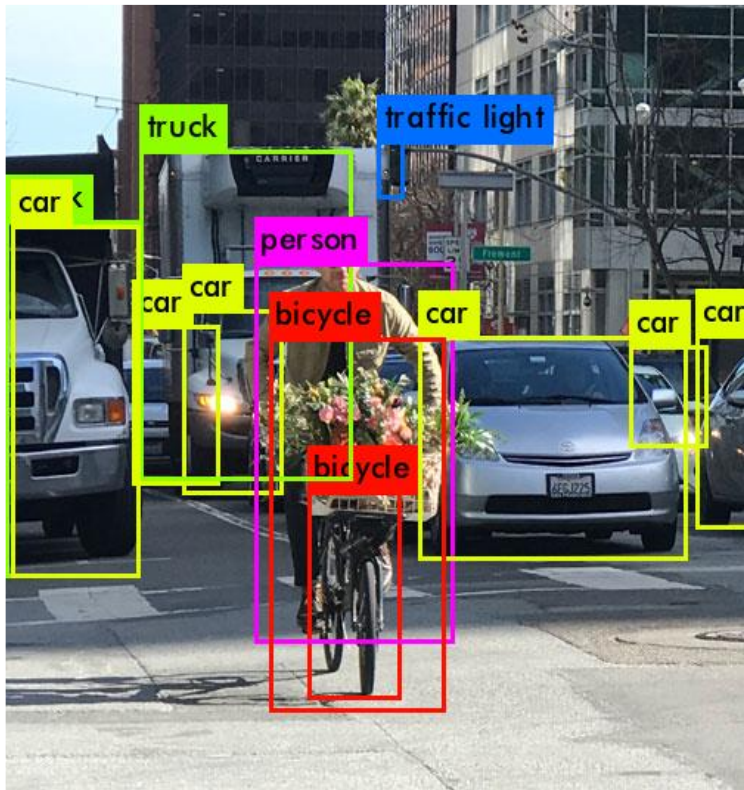
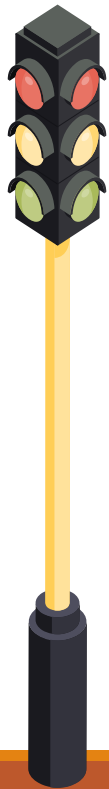
1. 프로젝트 소개



작품명 : Smart Traffic Light System

- 딥러닝 객체 검출 기술(Retinanet)을 응용해 효율적인 실시간 신호체계를 구축한다.
- 실시간으로 도로 상황을 촬영하여 자동차의 대수를 파악하고 교통량이 많은 도로에 초록 신호를 우선적으로 부여한다.
- 도로에 구급차가 진입 시, 해당 도로에 높은 우선순위를 부여한다.
- 횡단보도의 보행자 수를 파악하고 교통량을 고려해서 보행자 신호를 조작한다.

1. 프로젝트 소개



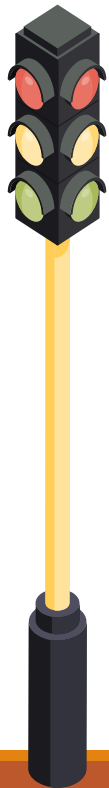
Deep Learning Object Detection

Input : 현재 교통 상황 스냅샷

Output : 스냅샷에 존재하는 객체들을 분류하고 각 객체의 수를 카운팅한 결과

=> 객체의 수를 이용하여 도로 신호별 우선순위를 부여하여 신호 스케줄링

1. 프로젝트 소개



프로젝트 구성

1. 딥러닝 서버

- Google Colab GPU 런타임 이용
- Tensorflow + Keras + Retinanet

2. 신호 관리 시스템

- 파이썬 GUI : PyQt5
- 신호 스케줄링 알고리즘 구현
- 딥러닝 서버와 통신

3. 시뮬레이션 세트장

- 사거리 도로 형태
- 120cm x 90cm 크기
- 차량 모형과 사람 모형 활용
- 4대의 웹캠으로 도로상황 캡처



2. 프로젝트 구조

Component & Deployment Diagram

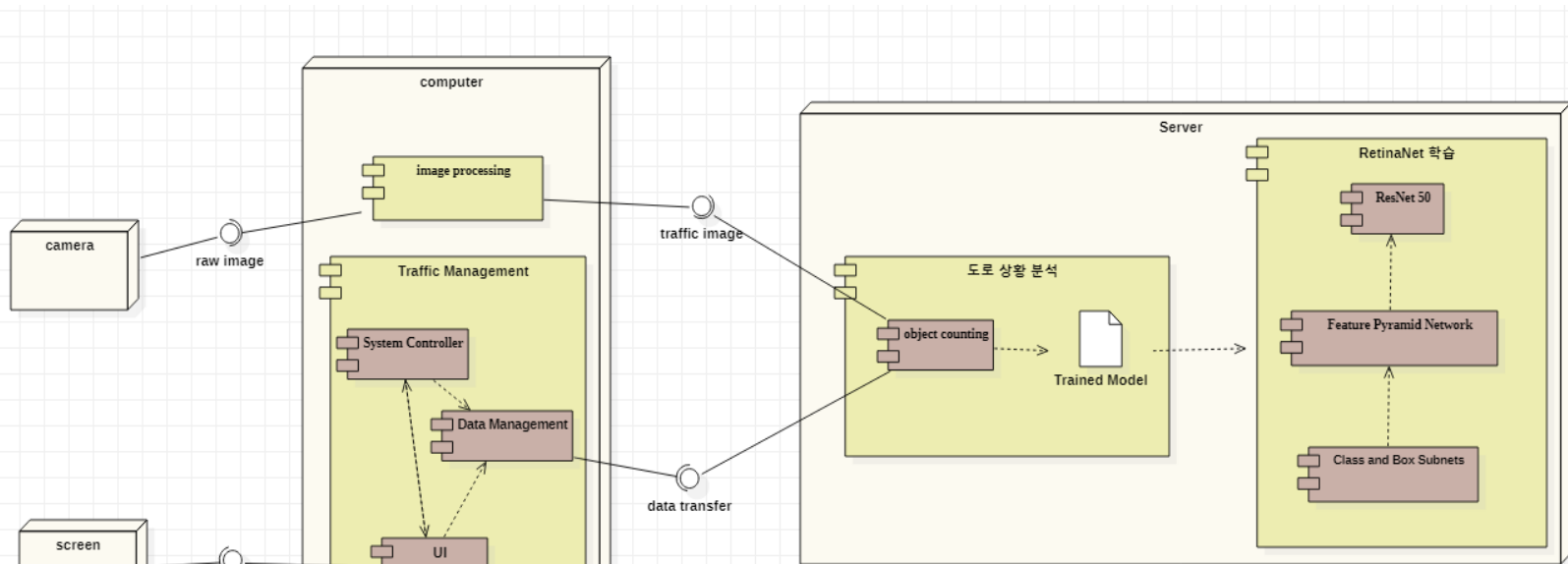
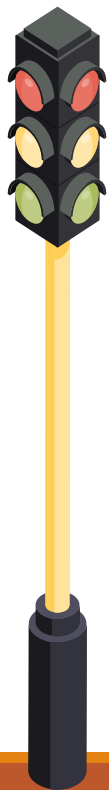
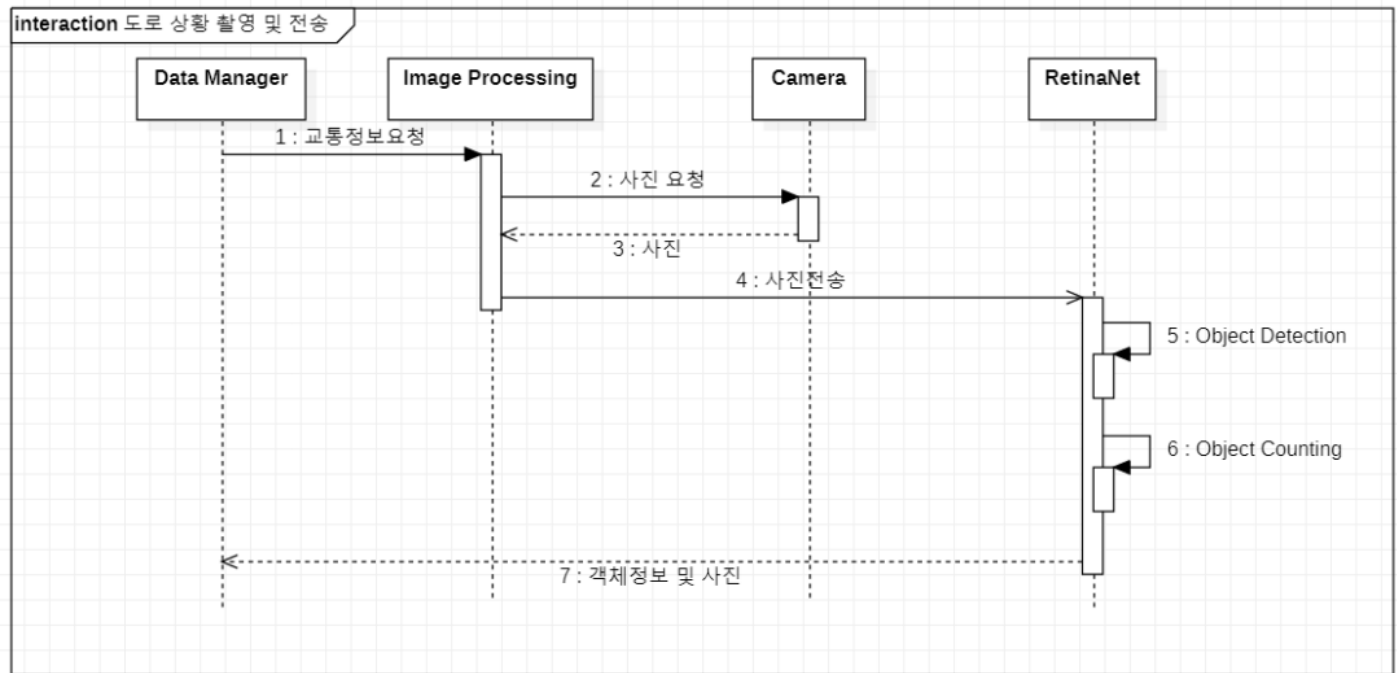
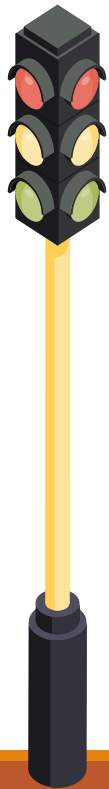


Image processing : 카메라에서 사진을 촬영하여 서버로 전송
Traffic Management : 서버에서 도로상황정보를 받아 신호 스케줄링을 실행
도로 상황 분석 : 사진으로 Object Detection 및 counting을 진행
RetinaNet 학습 : MS COCO 데이터셋 및 구글차 데이터셋으로 딥러닝 학습

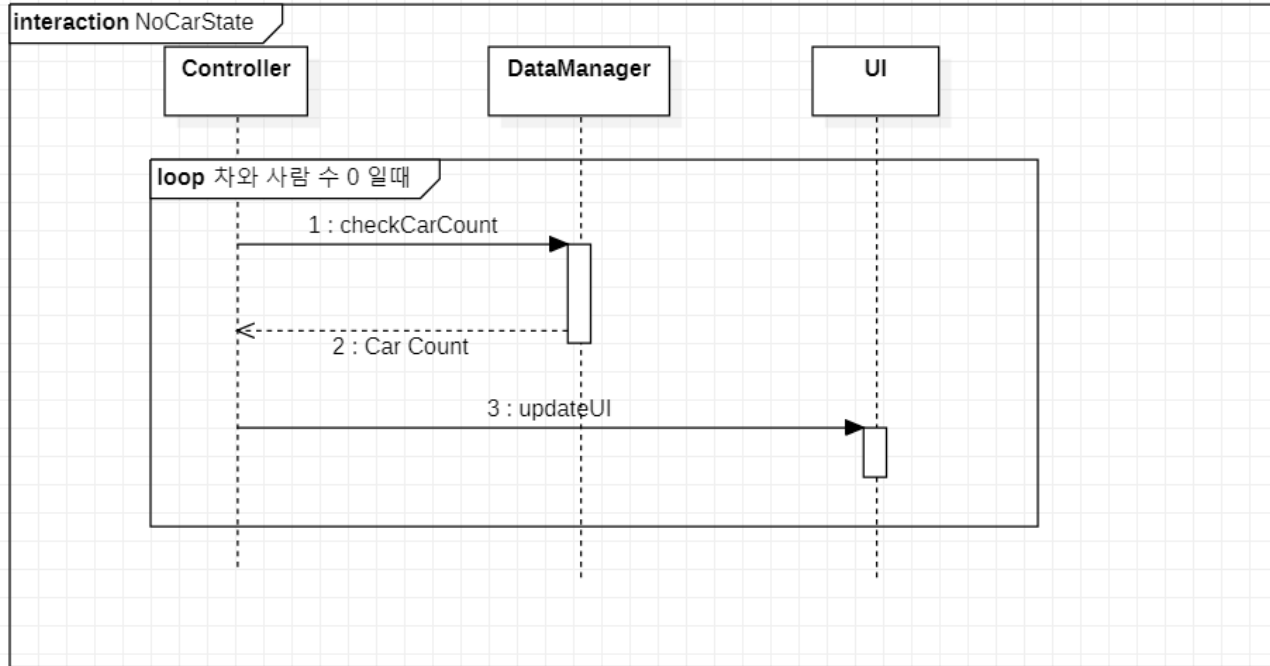
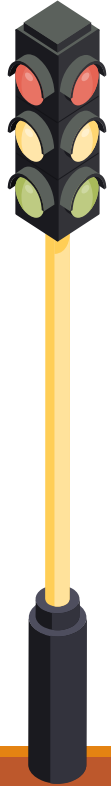
2. 프로젝트 구조

System Sequence Diagram 1. 도로 상황 촬영 및 전송



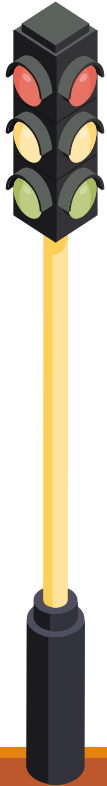
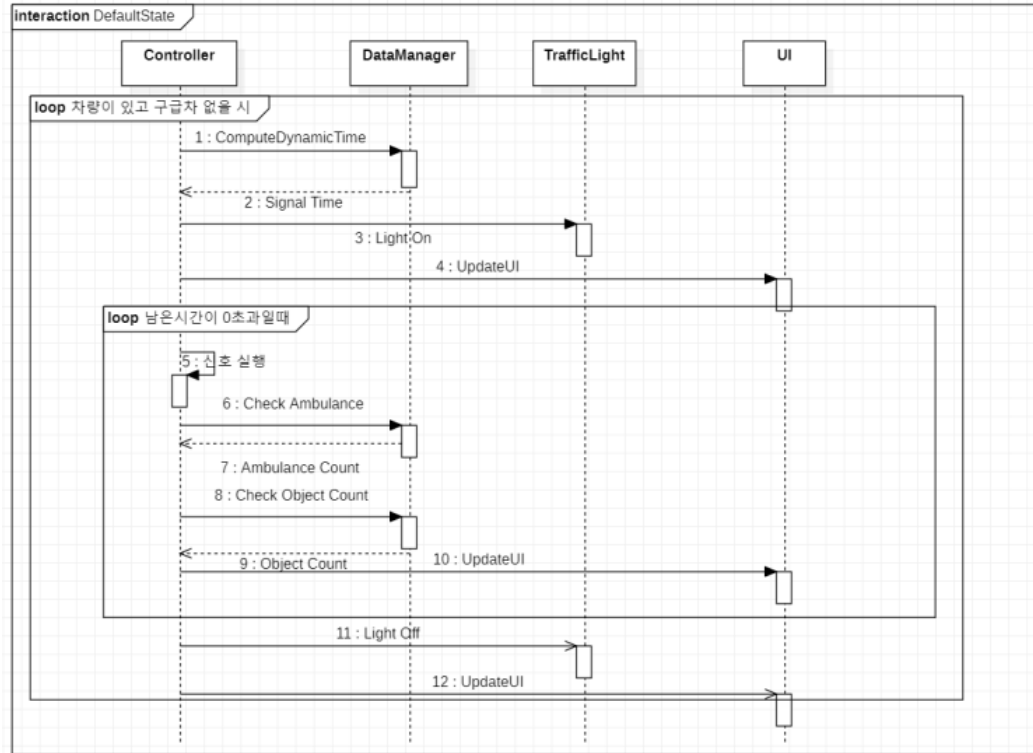
2. 프로젝트 구조

System Sequence Diagram 2. 도로에 차와 사람이 없을 때



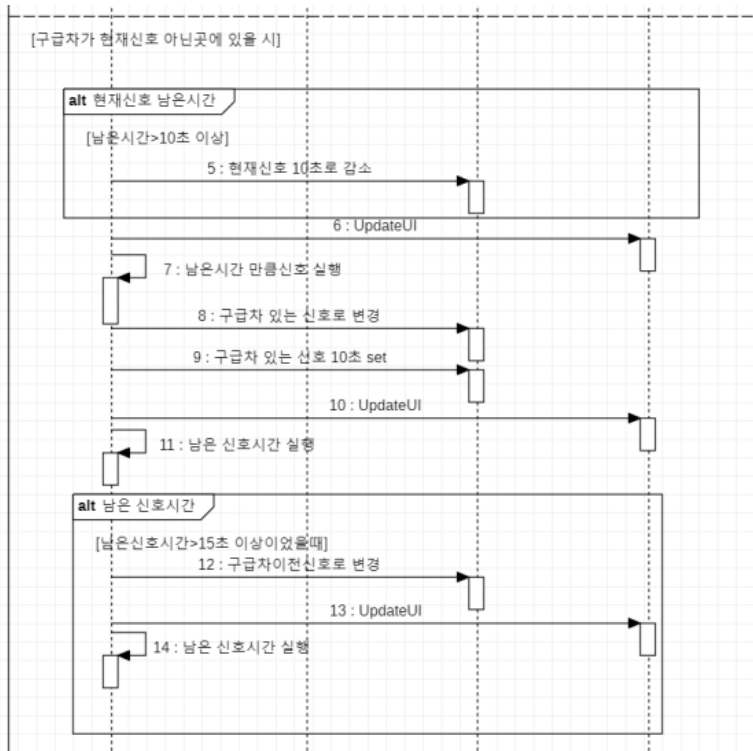
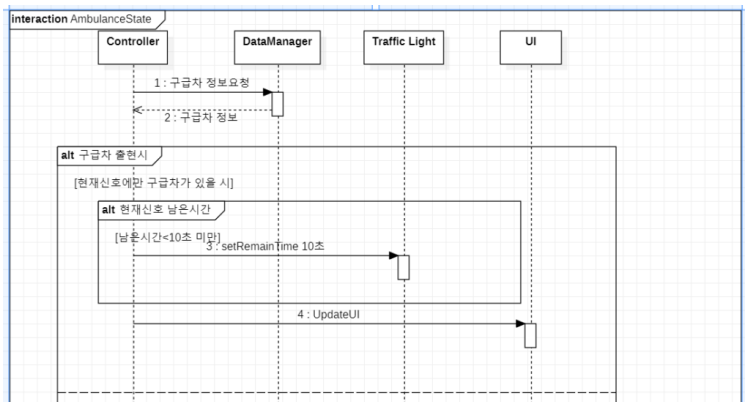
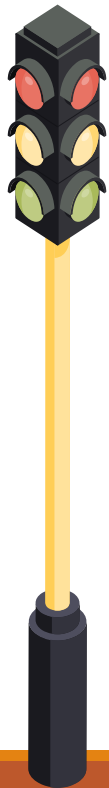
2. 프로젝트 구조

System Sequence Diagram 3. 기본 스케줄링 알고리즘



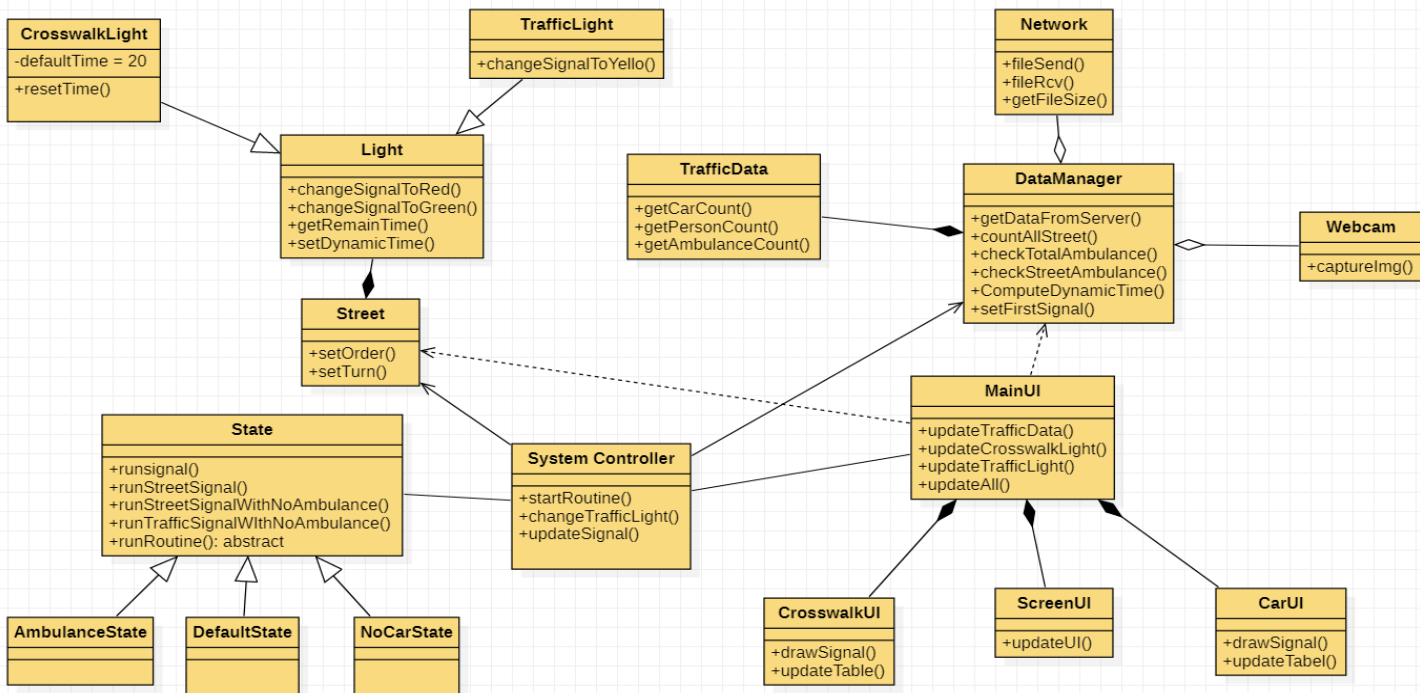
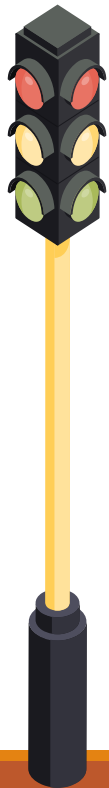
2. 프로젝트 구조

System Sequence Diagram 4. 구급차 출현 시 예외상황



2. 프로젝트 구조

Traffic Management 클래스 다이어그램



2. 프로젝트 구조

Data Management Module

Light Class

클래스: 차량신호등, 보행자 등 => 클래스 Street 집합연관

설명: 신호등 관련 클래스

기능: 신호변경(적색, 녹색, 황색)

기능: 신호유지시간 설정

기능: 현재 신호유지시간 저장 => 구급차 신호 변경할 때 필요

TrafficData Class

설명: 거리정보 관련 클래스

기능: 차량, 구급차, 사람 정보

DataManager Class

설명: 4개의 거리정보를 관리한다.

연관: TrafficData, Image Processing

기능: 서버로부터 거리 정보 얻어오기

기능: 현재 차량이 있는지 있는 알려주는 기능

기능: 구급차가 현재 있는지 알려주는 기능

기능: 해당 거리에 구급차가 있는지 알려주는 기능

기능: 해당 거리의 동적시간을 계산해주는 기능

기능: 어떤 거리가 차량이 가장 많은가 알려주는 기능

기능: 거리정보 보여주기

System Controller Module

Controller Class

설명: 전반적인 신호 제어를 담당

연관: DataManager, Light, View

기능: 신호 조작 기능

기능: DataManager한테 Data 요청

State Class

설명: 상태에 따른 여러가지 스케줄링 실행

연관: Controller

클래스: NocarState

클래스: DefaultState

클래스: AmbulanceState

UI Module

MainUI, CarUI, CrosswalkUI Class

설명: 화면 표시 관련 클래스

의존: TrafficData, Light

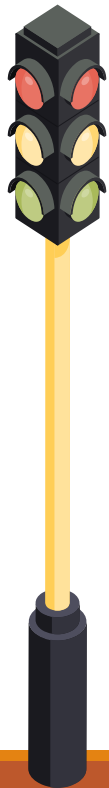
Image Processing Module

Video class

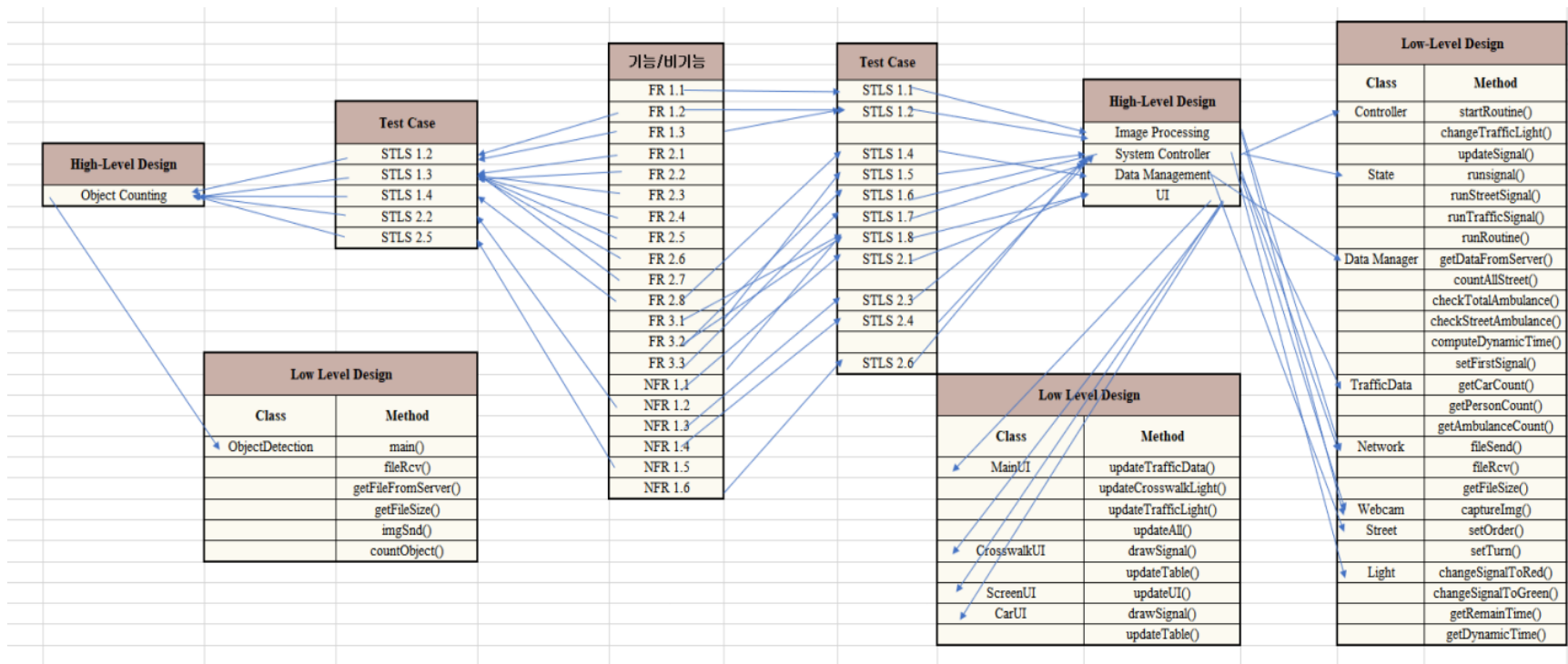
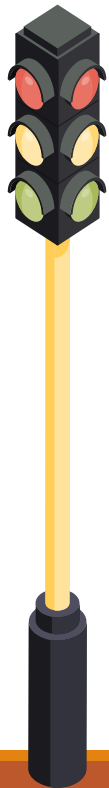
기능: 이미지 캡처

Network class

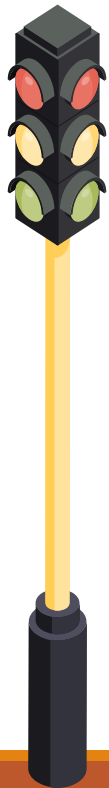
기능: 연결 설정 후 4개의 이미지 전송



3. 추적성 분석

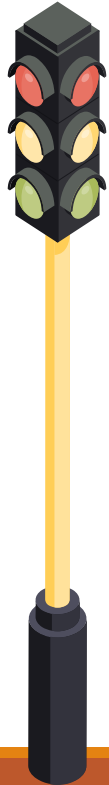


4. System Test & Pass/Fail Criteria



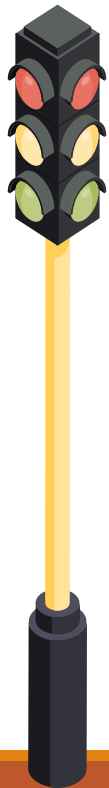
TEST no.	TEST DESCRIPTION	PASS/ FAIL
STLS_1.1	도로 상황을 4대의 카메라로 촬영하여 메인 컴퓨터에 전송할 수 있어야 한다. 사진 안에는 도로의 차량들과 사진 기준 도로의 왼쪽 보행자들을 담고 있어야 한다.	○
STLS_1.2	이미지를 서버에 전송할 수 있어야 한다. 이때 4개의 도로 상황 이미지가 항상 일정하게 정렬되어 전송하여야 한다.	○
STLS_1.3	이미지에서 찾은 객체들의 수를 세고 신호 스케줄러로 보낸다.	○
STLS_1.4	사거리에서 모든 방향에서 차량(또는 보행자)이 존재한다고 가정했을 때 기본적으로 동작하는 신호 스케줄링이다.	○
STLS_1.5	도로에 구급차가 나타났을 때 신호 스케줄링은 현재 신호를 잠시 멈추게 하고 구급차가 나타난 도로 쪽에 신호를 주어야 한다.	○
STLS_1.6	도로에 차나 보행자가 없을 경우 효율성을 높이기 위해서 도로들의 신호를 추가적으로 관리한다	○
STLS_1.7	신호 스케줄러가 정한 신호 스케줄대로 비가 동작한다.	○

4. System Test & Pass/Fail Criteria



TEST no.	TEST DESCRIPTION	PASS /FAIL
STLS_2.1	모니터를 통해 화면을 분할하여 도로 신호와 보행자 신호를 한 번에 표시할 수 있어야 한다.	○
STLS_2.2	딥러닝을 이용한 Object Detect의 Processing 시간을 확인한다. => 사진 당 0.09초 X 4장 = 0.5초 이내	○
STLS_2.3	분석데이터를 이용한 신호 스케줄링 걸리는 시간을 확인한다.	○
STLS_2.4	구급차 통과시 신호 변경 요청을 반영하는데 필요한 시간을 측정한다. => 웹캠 이미지 캡처 후 딥러닝 서버로 전송 시 네트워크 지연 및 객체 검출 시간으로 인하여 기준이었던 1초를 초과했다. 테스트 결과 구급차 출현 시 5초 정도 경과 후 신호변경이 반영되므로 실제 적용에는 무리가 없다고 판단된다.	△
STLS_2.5	Object detect의 정확도를 측정한다. => 기존 ms coco 데이터셋을 이용한 전이학습된 모델에 구급차 검출을 위한 전이학습을 시키는 과정에서 데이터셋 크기의 불균형과 차량과 구급차의 공통된 특징으로 인해 객체 검출 정확도가 조금 하락했다. 하지만 임계값 조절을 통해 정상적인 시뮬레이션이 가능하도록 보완했다.	△
STLS_2.6	기존 신호시스템보다 실시간 신호시스템 반영 시 개선 비율을 측정한다.	○

4. System Test & Pass/Fail Criteria



STLS 2.6: 기존 신호시스템보다 실시간 신호시스템 반영 시 개선 비율을 측정한다.

	대기시간	차량 대수	사람 수
거리1	120	2	0
거리2	120	2	0
거리3	120	2	1
거리4	120	6	2

-> 기존 신호 시스템의 시뮬레이션 결과

	대기시간	차량 대수	사람 수
거리1	110	2	0
거리2	121	2	0
거리3	123	2	1
거리4	77	6	2

-> 실시간 신호 시스템의 시뮬레이션 결과

기존 신호시스템의 시뮬레이션 결과와 실시간 신호 시스템의 시뮬레이션 결과를 비교하였을 때 차량 대수가 많은 거리4의 대기시간이 120초에서 77초로 현저히 줄었으며 그에 비해 다른 도로의 대기시간은 증가했다. 하지만 전체 차량 평균대기시간은 120초에서 97.5초로 개선됐다.

5. 데모 시나리오

시뮬레이션 가정:

ㄱ. 도로의 교통량은 4번 > 1번 = 3번 > 2번 이다.

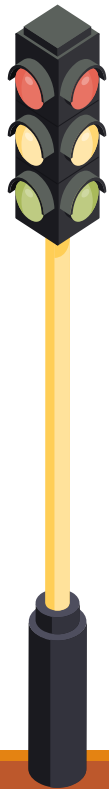
각 도로는 신호가 바뀌는 것을 기준으로 차량이 추가된다.

도로1 : 2번에 한대 / 도로2 : 3번에 한대

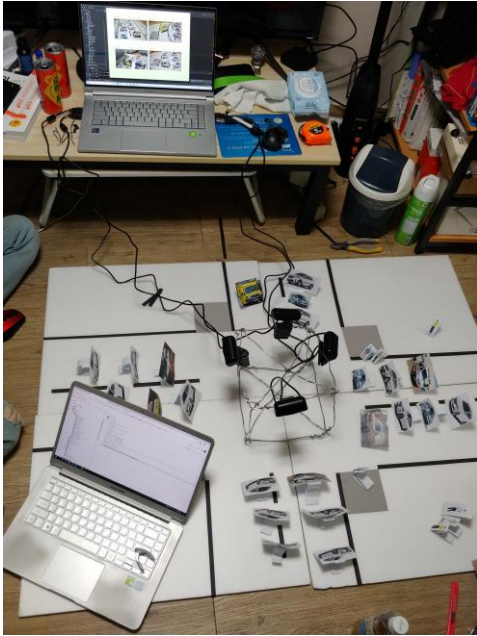
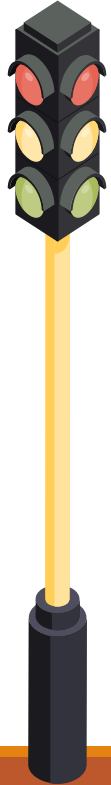
도로3 : 2번에 한대 / 도로4 : 1번에 한대

ㄴ. 보행자의 최대수는 도로3번에 1명, 도로4번에 2명이고 신호 변경 시 마다 1명씩 추가된다.

1. 시뮬레이션 가정대로 신호등시스템을 두 사이클 진행하며 동적으로 시간이 변경되는 것을 측정한다.
2. 구급차가 현재 녹색신호인 곳에서 발견됐을 시, 구급차 있는 곳의 신호가 10초 미만일 경우 10초로 연장되는 상황을 확인한다. (2번 도로에 테스트)
3. 현재 녹색신호가 아닌 도로에서 구급차가 발견됐을 시에 현재신호 남은 시간을 10초로 줄인 후 구급차가 있는 신호로 이동하는 것을 확인한다.(2번 신호일때 3번 도로에 테스트)
4. 3번과 같은 상황에 대하여 구급차가 3번 도로와 4번도로에 동시 발견 시에 대하여 확인한다.
5. 모든 도로에 차량과 사람이 없을 때 모든 신호가 적색신호로 바뀌는 것을 확인하고 특정 도로에 차량이 진입하면 그 도로부터 신호체계를 재개하는 것을 확인한다. (한 사이클 진행 후 3번도로에 차량 진입)



5. 데모 세트장 구성 및 UI



인공지능 신호통제시스템 관리

시스템 시간: 18:52:4

차량 수	구급차 수	사람 수	신호	호출시간	부대시간
차량신호1 차대	0	0	10	40	
차량신호2 차대	0	0			
차량신호3 차대	0	0			
차량신호4 차대	0	0			
보행자신호 녹색	10	20			
보행자신호 적색	0	20			
보행자신호 녹색	0	20			
보행자신호 적색	0	20			

보행자신호: [버튼] 차량신호: [버튼]

답변: [버튼] default: [버튼]

1.png 차량 수: 0개 구급차 00개 사람 수: 0개 2.png 차량 수: 0개 구급차 00개 사람 수: 0개

3.png 차량 수: 0개 구급차 00개 사람 수: 0개 4.png 차량 수: 0개 구급차 00개 사람 수: 0개

차량신호1
신호: 녹색
부대시각: 40
호출시각: 3.0

차량신호2
신호: 적색
부대시각: 0
호출시각: 0

차량신호3
신호: 적색
부대시각: 0
호출시각: 0

차량신호4
신호: 적색
부대시각: 0
호출시각: 0

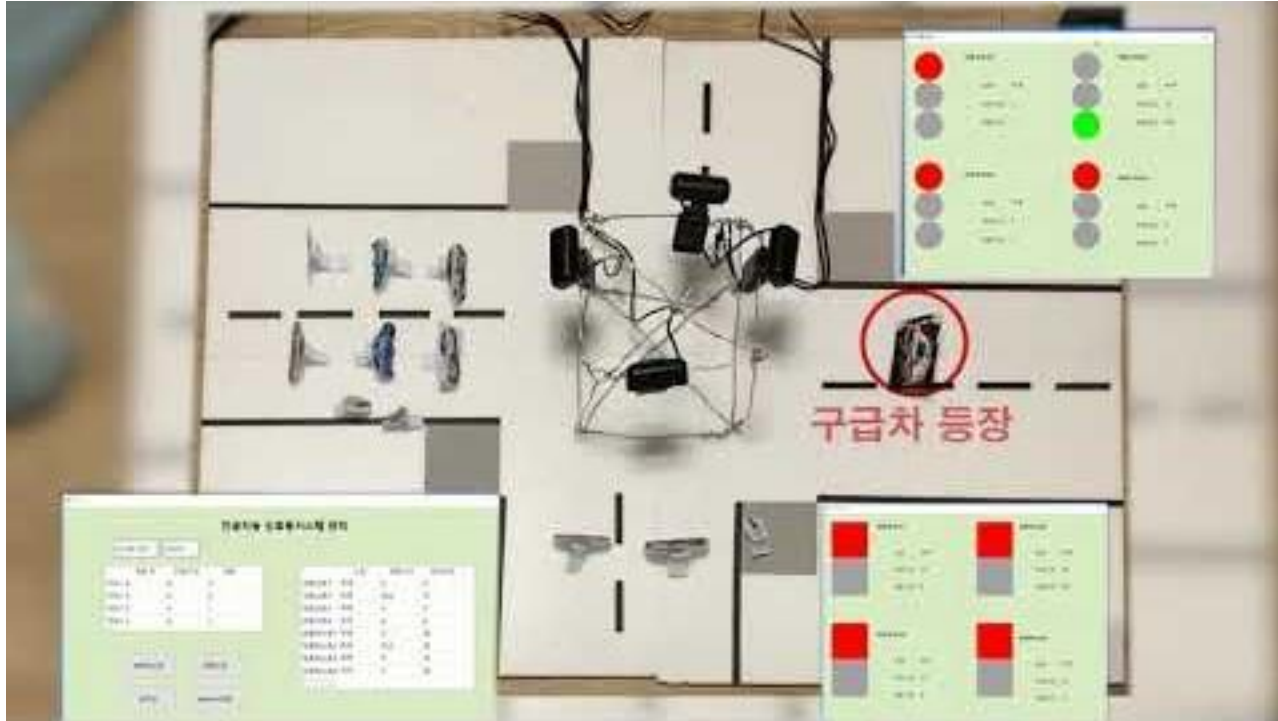
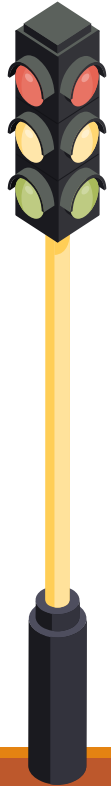
보행자신호1
신호: 녹색
부대시각: 20
호출시각: 3.0

보행자신호2
신호: 적색
부대시각: 20
호출시각: 0

보행자신호3
신호: 적색
부대시각: 20
호출시각: 0

보행자신호4
신호: 적색
부대시각: 20
호출시각: 0

6. 데모영상



유튜브 링크 : <https://youtu.be/tQtdUiGZ5bg>

동영상 파일 링크 :

https://drive.google.com/file/d/1fPEDrupV8wopMHUVEdHz_wZoOp6uE339/view?usp=sharing