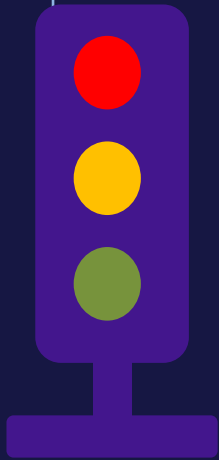


STLS

: Smart Traffic Light System



201411285
유종혁

201411286
유환규

201611308
최준오

201714166
신예슬

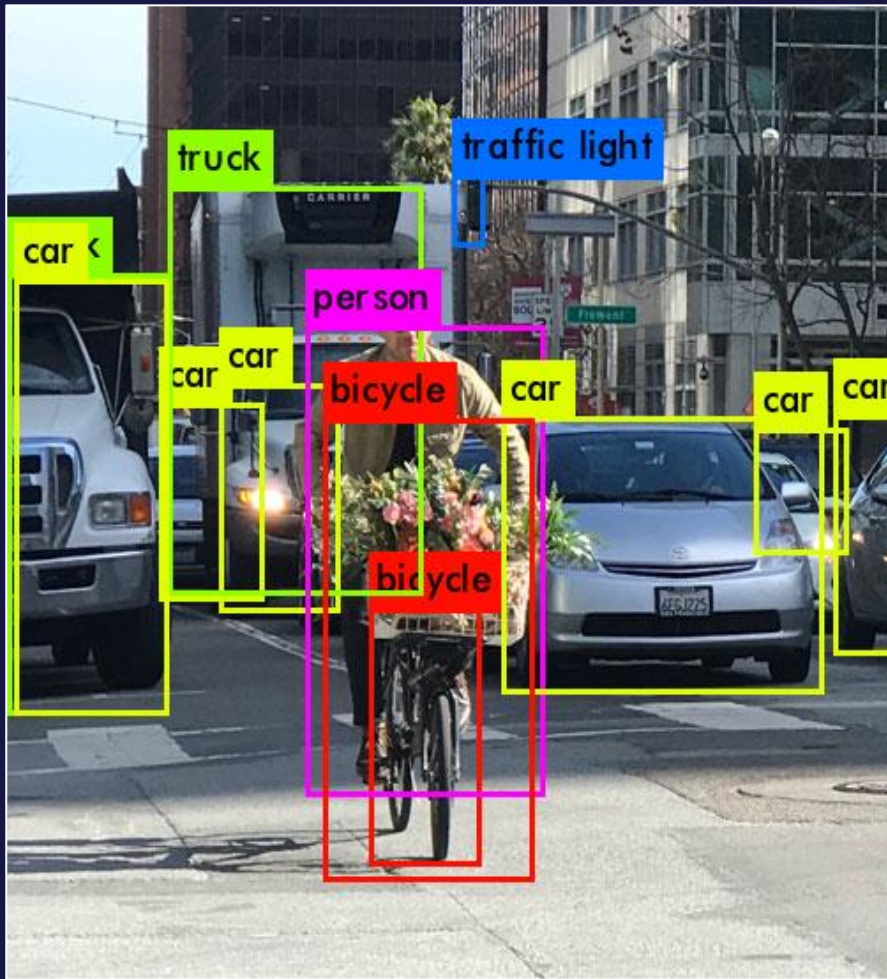


1. 작품명 및 설명

작품명 : Smart Traffic Light System

- 딥러닝 기술(R-CNN)을 응용해 효율적인 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. 작품명 및 설명

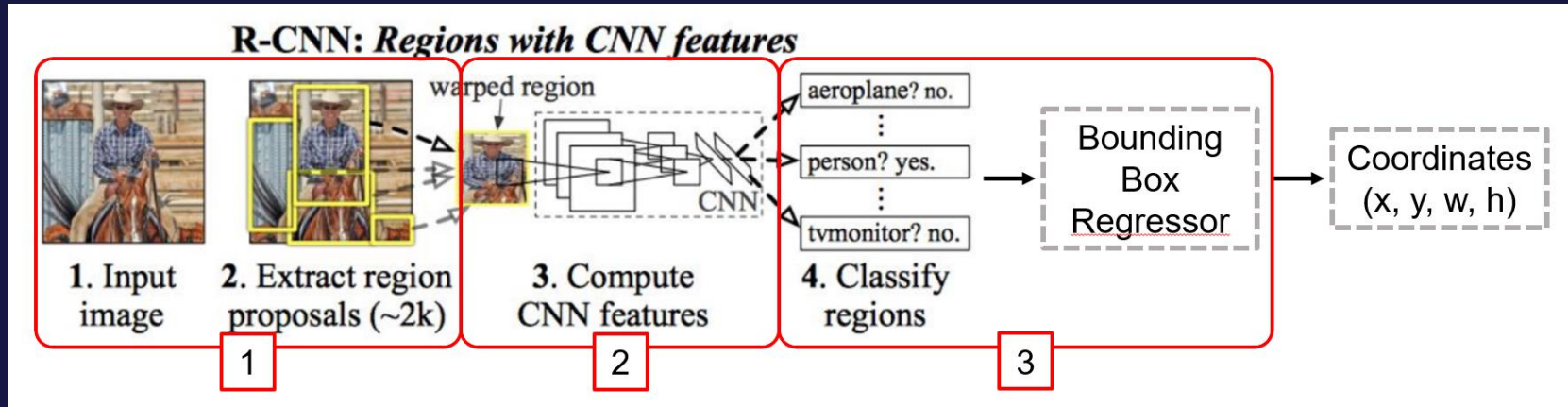


Input : 현재 교통 상황 스냅샷

Output : 스냅샷에 존재하는 객체들을 분류하고 각 객체의 수를 카운팅

Result: 객체의 수를 이용하여 신호별 우선순위를 부여하여 스케줄링

1. 작품명 및 설명



1. Hypothesize Bounding Boxes (Proposals)

: Image로부터 Object가 존재할 적절한 위치에 Bounding Box Proposal (Selective Search) 2000개의 Proposal이 생성됨.

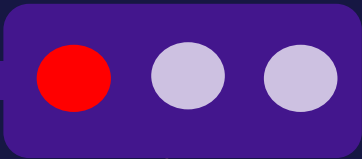
2. Resampling pixels / features for each boxes

: 모든 Proposal을 Crop 후 동일한 크기로 만들 (224x224x3)

3. Classifier / Bounding Box Regressor

: 위의 영상을 Classifier와 Bounding Box Regressor로 처리

하지만 모든 Proposal에 대해 CNN을 거쳐야 하므로 연산량이 매우 많은 단점이 존재 -> 개선된 모델들을 응용해볼 예정



2. SW/ COTS SW/ HW

새로만들 SW :

- 도로상황을 스냅샷으로 캡처 후 서버로 이미지 전송
- 딥러닝(R-CNN) 모델에 스냅샷을 넣어서 객체 분류 및 카운팅
- 보행자 신호등과 도로 신호등의 실시간 스케줄링 시스템

COTS SW / HW :

- Python, Tensorflow, OpenCV, / 웹캠

3. 최종 산출물 형태와 기능

형태 :

- 도로 환경 세트장(카메라, 차량모형, 사람모형)
- 각 신호등을 디스플레이 해줄 프로그램(for PC)

기능 :

- 카메라로 도로 상황을 촬영하고 딥러닝을 활용하여 차량의 종류, 대수, 보행자의 수를 파악
- 실시간으로 신호체계의 스케줄을 변경하여 좀 더 효율적으로 교통상황을 통제



4. Alternative Solutions

노르트홀란드주(州) 지능형 교통 체계
(ITS·Intelligent Transportation System)



교통 상황을 실시간으로 파악해 막힌 곳은 뚫어주고(초록), 차가 없는 쪽은 차가 나타날 때까지 정지 신호(빨강)

그린 웨이브(green wave) 효과: 특정 차량이 어떠한 목적지까지 가면서 빨간불 신호등을 만나지 않고 계속 초록불을 만나게 하는 것이다. 스마트 신호등을 도입하고 나서 교통 체증은 약 20% 감소했다.



5. Project Justification

기존 도로 신호등 시스템 :

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

따라서 실시간으로 변하는 도로상황에 대응하지 못하기 때문에 비효율적이다.



딥러닝을 활용한 실시간 신호등 시스템 :

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



6. Risk Analysis

- 이미지 내 필요한 객체(자동차, 사람, 구급차)를 구분하는 정확도가 떨어질 수 있다.
- 같은 종류의 객체의 수를 카운팅하는 솔루션이 필요하다.
- 자동차로 학습한 분류기가 세트장 내부 자동차를 분류 못 할 수 있다.
- 학습시킬 자동차 이미지 부족 가능성이 있다.
- 딥러닝과 tensorflow API 이해도 부족하다.

Risk Reduction Plan

- 1) 원하는 정확도에 도달하기 까지 학습 데이터 량을 늘린다.
- 2) 딥러닝 모델에 적용할 수 있도록 세트장을 구성한다.
: 카메라의 구도, 카메라의 화질, 자동차와 사람 모형의 형태 등을 고려
- 3) 이미지 처리와 머신 러닝에 대한 추가 공부 필요하다.



7. Success Criteria

- 보행자와 차량 대수를 카운팅하여 현재 교통량을 확인할 수 있어야 한다.
- 일반차와 구급차를 구분하여 구급차 통행 시 녹색신호를 켜줄 수 있어야 한다.
- 카메라로 찍은 이미지의 자동차와 보행자 분류 정확도 70% 이상
- 이미지 인식 후 신호 시스템에 적용까지의 latency 가 5초 이내
- 기존 고정된 스케줄의 신호체계와의 비교 시 신호대기 시간 감소