

---

# 연구성과 발표회

---

26<sup>th</sup> February 2020

전기전자공학부 강민지

전기전자공학부 권미경

스마트운행체공학과 김미선

스마트운행체공학과 손희원

# Index

---

I . 동계 방학 연구 목표

II . 개발 환경

III . 연구 결과

IV . 추후 연구 계획

# I. 동계 방학 연구 목표

---

## AI를 이용한 차선 및 표지판 인식

- YOLO 라이브러리 활용
- ROS를 통한 RaspberryPi3, OpenCR, RemotePC간 통신
- 받은 정보를 사용하여 차선 및 표지판 인식

## II. 개발 환경

- 기존 개발 환경

### Initial Development Environment

#### - Hardware

- turtlebot3 burger
- Raspberrypi 3
- OpenCR
- PiCam
- LiPo 3cell battery

#### - OS (remotePC, raspberrypi3)

- Ubuntu 18.04
- ROS melodic

#### - Software

- OpenCV 3.4.0
- YOLO v3

## II. 개발 환경

- 연구 진행하며 변경된 개발 환경 및 사유

Hardware Environment	변경 사유
<ul style="list-style-type: none"><li>- <b>turtlebot3 Waffle</b></li><li>- <b>Jetson TX2</b></li><li>- OpenCR</li><li>- PiCam</li><li>- LiPo 3cell battery</li></ul>	<ul style="list-style-type: none"><li>: Jetson TX2 보드 탑재 위함</li><li>: YOLO의 원활한 작동을 위함<ul style="list-style-type: none"><li>- Jetson nano가 아닌 TX2 사용 이유<ul style="list-style-type: none"><li>: yolo중 가장 용량이 작은 yolo tiny를 설치하여 작동시켰을 때, 메모리 64GB로도 부족하였고, CPU 학습과정에 문제 발생</li></ul></li></ul></li></ul>

## II. 개발 환경

- 연구 진행하며 변경된 개발 환경 및 사유

OS Environment	Software Environment	변경 사유
<ul style="list-style-type: none"><li>- <b>Ubuntu 16.04</b></li><li>- <b>ROS kinetic</b></li></ul>	<ul style="list-style-type: none"><li>- <b>OpenCV 3.3.1</b></li><li>- YOLO v3</li><li>- <b>TensorRT 4.0</b></li><li>- <b>cuDNN v7.1.5</b></li></ul>	<p>: TX2에 적합하면서도 stable하고 open source가 많은 OS 및 software를 설 치하였음</p>

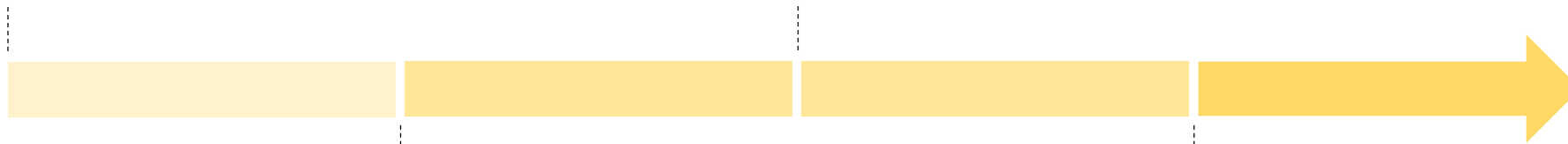
### III. 연구 결과

CNN 및 YOLO  
기본 개념 공부

TX2, RemotePC  
개발 환경 구축

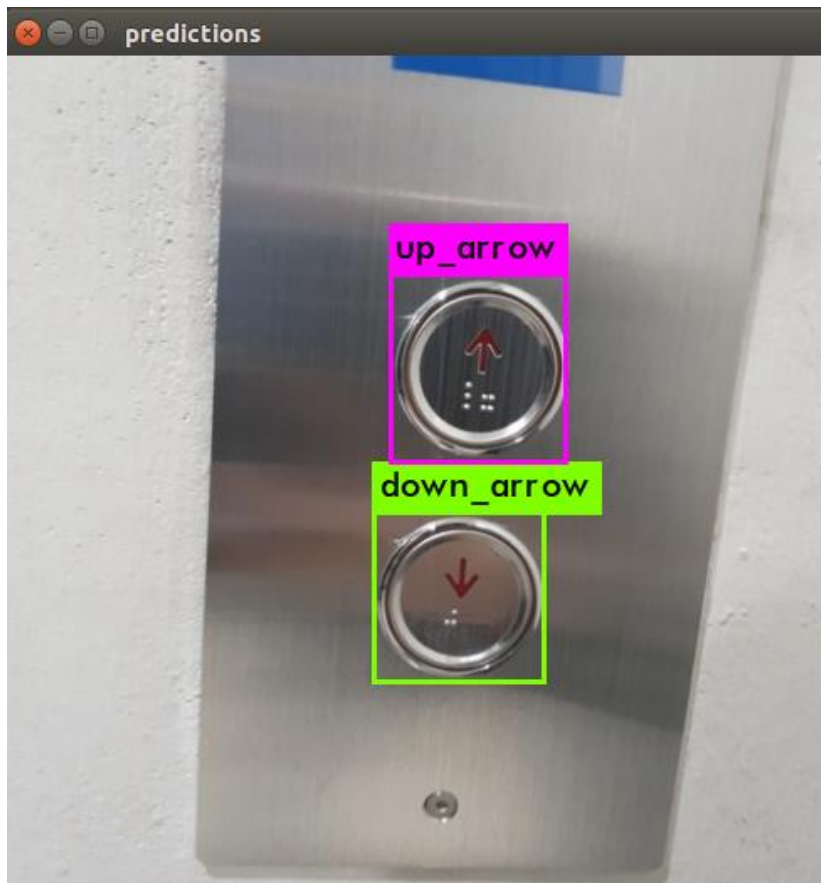
여러 실험을 통한  
보드 선정

Yolo\_mark 예제 실행



### III. 연구 결과

- TX2에서 YOLO 실행 결과



- 엘리베이터 사진을 YOLO로  
image training을 하여 화살표 인식에 성공함



# III. 연구 결과

---

- 성과 분석

- 계획 대비 50% 성공

- 보드를 재선정하고 다시 개발환경을 구축하는 과정에서 많은 시간이 소비됨
    - 영상 하나를 처리하는데 평균 3일이 걸리는 등 처리 속도가 매우 더딤
    - 처리 속도 향상을 위해 노트북에서 처리하고자 환경 구축하는 과정에서 여러 문제가 발생
      - 제공된 노트북의 GPU : intel사의 UHD Graphics 620
      - Intel에서 Ubuntu 환경을 위한 graphic driver를 제공하지 않고,  
개발자를 위한 소스코드만을 제공하여 설치에 어려움을 겪었음

## IV. 추후 연구 계획

---

- 본격적인 프로젝트는 팀을 따로 나누어 진행할 예정

강민지, 권미경	김미선, 손희원
건물 위치 안내 로봇 개발 (부기더가이더)	개인 맞춤형 자동차 및 스마트 홈 개발

## IV. 추후 연구 계획\_강민지, 권미경

---

- 부기더가이더

- 작품 개요

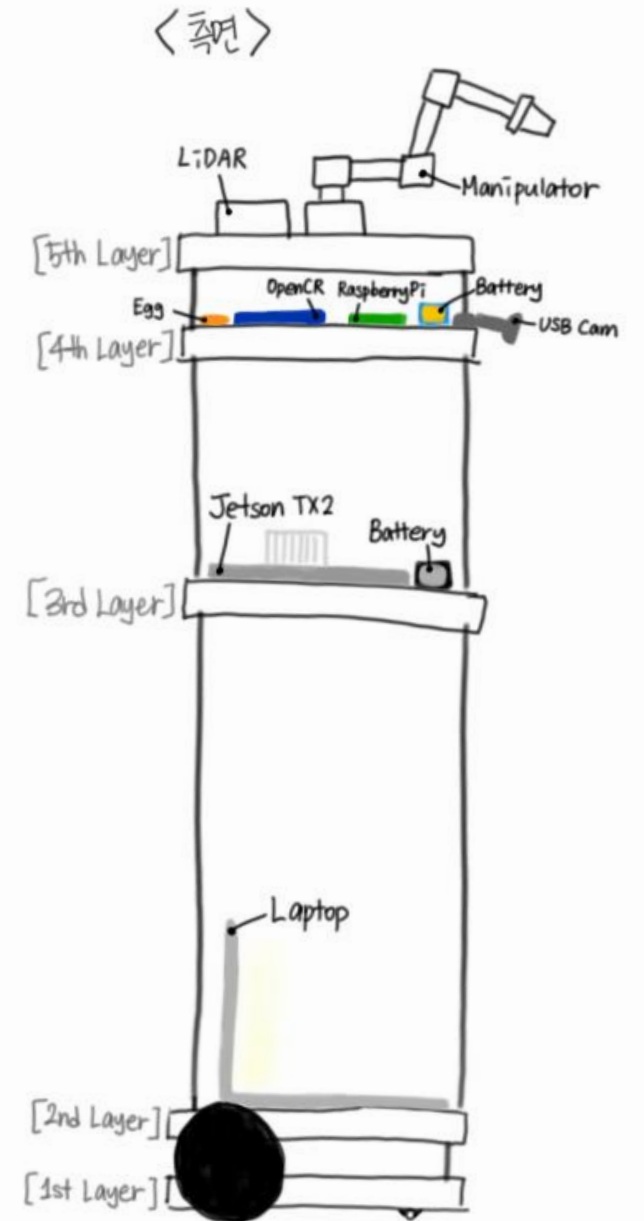
- 건물의 위치를 안내하고자 만들어진 로봇
    - 목표 지점을 입력하면 필요 시 카메라로 층수를 인식하여 엘리베이터로 이동하고, 각 층마다 저장되어 있는 맵을 통해 최적화 된 루트로 길 안내
    - 구성: Turtlebot Waffle Pi, OpenManipulator, USB camera, Jetson TX2

## IV. 추후 연구 계획\_강민지, 권미경

- 부기더가이더

- 구성도

- 5층: Lidar, 매니플레이터
- 4층: 에그(무선라우터 대체), OpenCR, 배터리(3cell, 3700mah), USB camera
- 3층: 젯슨 TX2, 배터리(3cell, 6000mah)
- 2층: RemotePC (목표 지점 입력, 코드 실행)
- 1층: Actuator(바퀴)



## IV. 추후 연구 계획\_강민지, 권미경

---

- 부기더가이더

- 추후 계획

- 라이다가 매니퓰레이터를 장애물로 인식하는 문제 해결 예정
    - 실시간으로 장애물을 피하여 목표지점으로 도착해야 함
    - 엘리베이터 이용 시 층수 인지 및 적정 속도로 승하차

## IV. 추후 연구 계획\_김미선, 손희원

- 개인 맞춤형 자동차 및 스마트 홈 개발



## IV. 추후 연구 계획\_김미선, 손희원

PART 1 ( 영상처리 )	PART 2 ( 딥러닝 )	PART 3 ( 통신 )	PART 4 ( 제어 )
<p>얼굴 인식하여 차주인지 확인</p> <p>↓</p> <p>현재 감정 상태 판별</p>	<p>기분에 따른 개인별 패턴 학습</p> <p>↓</p> <p>패턴에 따른 음악 및 장소 추천</p>	<p>Web Server 구축으로 차와 스마트 홈 간 통신</p> <p>Google Cloud와 Web Server 간 데이터 송/수신</p> <p>↓</p> <p>현재 위치 및 도착 시간 파악</p>	<p>PART 2, PART 3 결과에 맞추어 하드웨어 제어</p>
OpenCV	TensorRT	GPS, http 통신	
ROS			

## IV. 추후 연구 계획\_김미선, 손희원

---

### Development Environment

#### - Hardware

- Jetson TX2 1대
- Jetson nano 1대
- TX2 내장 카메라
- GPS 모듈
- LED, LCD
- 스피커
- LiPo 3cell battery 2개

#### - OS (remotePC, raspberrypi3)

- Ubuntu 16.04 LTS
- ROS kinetic

#### - Software

- OpenCV 3.3.1
- TensorRT 4.0



---

# 감사합니다

wbclair7@konkuk.ac.kr (강민지)  
kmk3942@konkuk.ac.kr(권미경)  
altjs4510@konkuk.ac.kr (김미선)  
dnjs99@konkuk.ac.kr (손희원)

---