



OLAF

2020 Capstone Design TEAM 11 ELSA

김다훈 김선필 김명수 배한울 윤찬우 주용수 교수님

OLAF 프로젝트 소개

본 프로젝트의 목표는 실내에서 사용자가 원하는 목적지까지 길을 안내하는 로봇을 제작하는 것입니다.



OLAF 시나리오

1



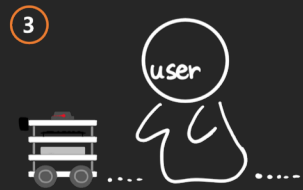
사용자가 웹에서 목적지를 선택합니다.

2



OLAF가 서버로부터 정보를 받습니다.

3



OLAF가 목적지까지 안내를 시작합니다.

OLAF 시스템 구조도

AWS SERVER

Web Server

IONIC

JSON Data Polling

UX / UI: User Input

JSON Server

Node JS

JSON Data Update

OLAF (robot)

IMU sensor

LiDAR sensor

Camera 1

Camera 2

Arduino OpenCR

Modified Firmware: Encoder Correction

Dynamixel

Encoder

Motor

Nvidia Jetson TX2 Board

Ubuntu 16.04 LTS

ROS Core (Kinetic Kame)

Navigation

Global Planner

Room Number Recognition: Tesseract (Camera 2)

Lane Detection (Camera 1)

Wall Detection (LiDAR)

Odometry Info (IMU, Encoder)

Local Planner

Object Recognition: YOLO (Camera 2)

Obstacle Detection (LiDAR)

Operate Node

JSON Request

Path Calculator

Drive to TargetPoint

OLAF 기대 효과

미래관 구관과 신관의 경계처럼 설명하기 어려운 경로를 OLAF에게 직접 안내 받아서 목적지에 쉽게 도달할 수 있다.

실내 자율주행로봇의 관점에서 이동형 방범카메라 로봇, 심부름 로봇 등 여러가지 실내 자율주행이 필요한 로봇의 개발에 바탕이 될 수 있다.

OLAF 기능 설명

Obstacle Detection:

YOLO를 통한 Classification과 LiDAR의 결합으로 장애물의 거리를 판단해 정지

Lane Detection:

미래관 바닥의 타일을 인식하여 통로와 평행하게 주행할 수 있도록 보조

Odometry + LiDAR Control:

Lane Detection에 실패할 경우 Odometry 정보와 LiDAR 벽 인식을 통해 주행

Number Recognition:

Tesseract를 이용해 방 입구의 번호판 영역에서 번호를 추출해 위치정보를 보조

Path Calculation:

목적지를 좌표로 표현하고, 현재 좌표에서 목표 좌표까지의 이동 경로 생성

Web Server 구축 & UX / UI 제작:

사용자의 입력을 받기 위해 IONIC을 활용한 web server 구축 및 UX / UI 제작

Server Connection: JSON server data share

- ROBOT: polling(사용자 입력), update(로봇 운행 상태)

- WEB SERVER: polling(로봇 운행 상태), update(사용자 입력)



OLAF 역할



김다훈

하드웨어 설계 제작
Point op node 제작
OpenCR Firmware
Data Share Protocol



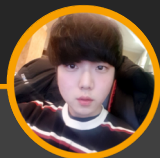
김선필

하드웨어 설계 제작
Point op node 제작
Path Calculation
Lane Detection
RoomNum 인식



김명수

AWS Setting
Web Server UX / UI
JSON Server
Data Share Protocol



배한울

하드웨어 설계 제작
Point op node 제작
OpenCR Firmware
Obstacle Detection



윤찬우

하드웨어 설계 제작
YOLO
Lane Detection
Obstacle Detection

OLAF 하드웨어 구조

