

1인 미디어 카메라 로봇

스마트모빌리티 정희성
스마트모빌리티 전준혁
스마트모빌리티 황찬기

목차

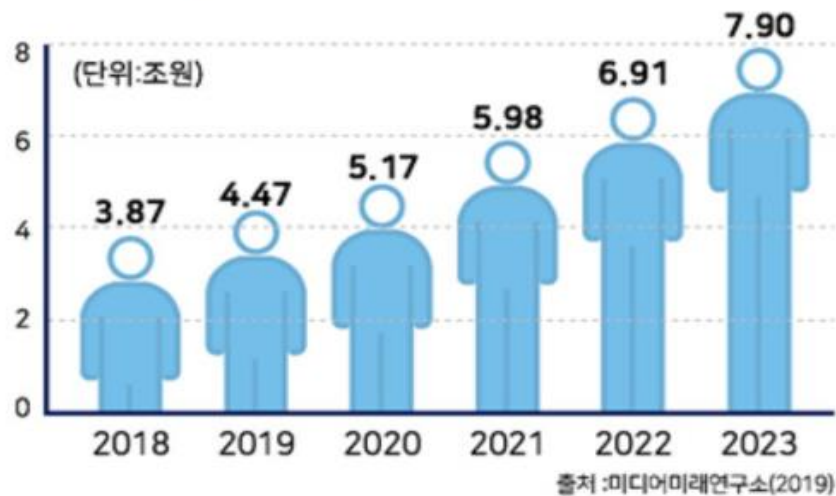
1. 선정이유
2. 사용부품
3. 기능
4. 블록도
5. 얼굴 검출 및 인식
6. 손 인식
7. ROS통신
8. 동작 영상
9. Q&A

1. 선정이유

1인 미디어산업 2.5조 규모로 성장...과기부 지원 대폭 확대

기사입력 : 2023년03월07일 16:41 | 최종수정 : 2023년03월07일 16:41

1인 미디어 시장 규모



- 국내 유튜브나 인스타 라이브 등을 이용한 1인 미디어에 사람들이 관심을 많이 가지고, 가지고 도전하는 사람들도 많아 지고 있다.
- 1인 미디어의 특성 상 기존 장비를 사용 시, 사용자가 직접 조정을 해야 하는 불편함이 있는데, 이러한 불편함을 개선해보며 학기 중 배웠던 raspberry pi를 활용하여 만들어 보고자 주제를 선정하게 되었습니다.

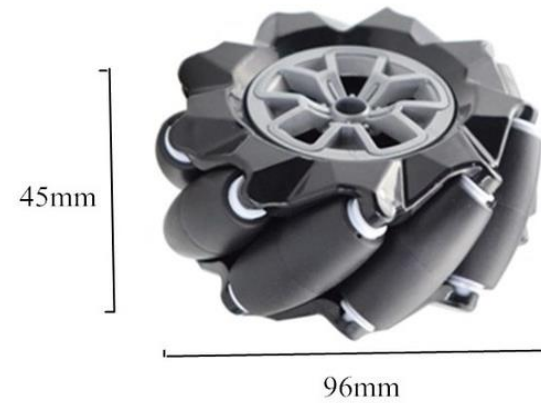
2. 사용부품



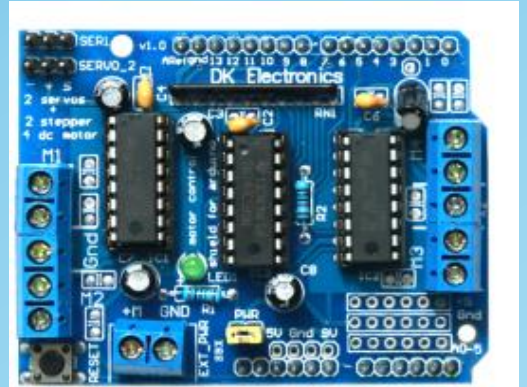
12V350rpm motors



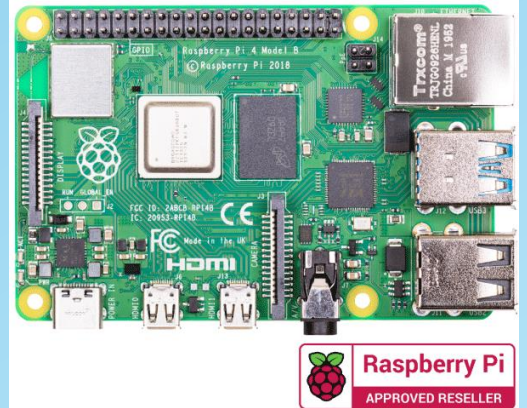
96mm mecanum wheel



Weight: 150g

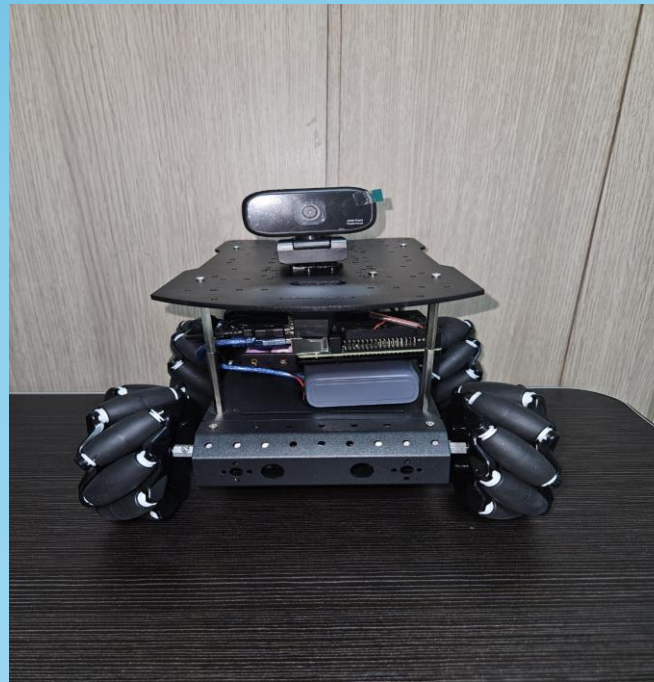


8 GB Raspberry Pi 4 Model B

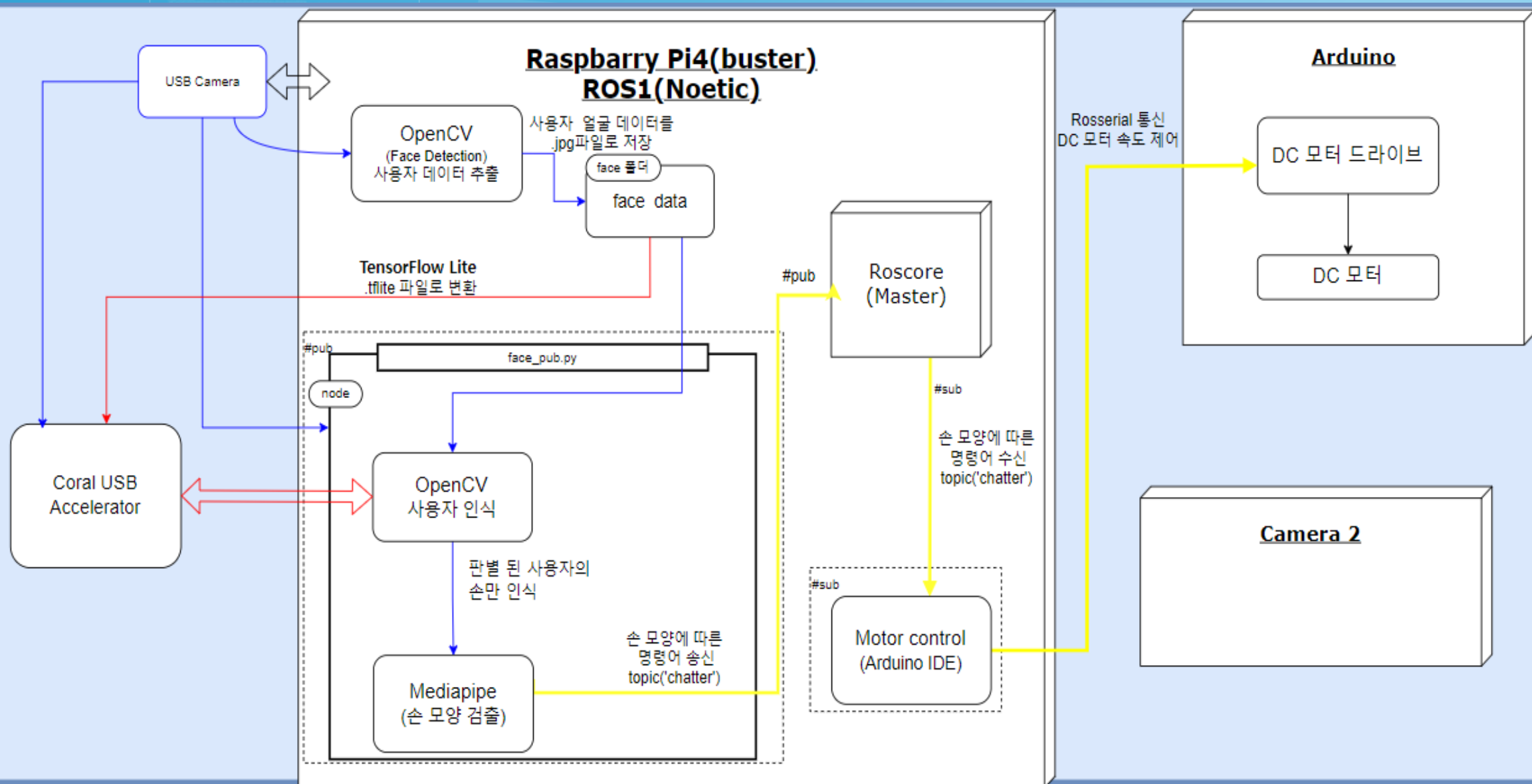


3. 기능

- 사용자의 손모양을 자동으로 인식 인식된 손 모양에 따라 카메라를 설치한 주행로봇이 자동으로 거리를 조절해준다.



4. 블록도



5. 얼굴 검출 및 인식

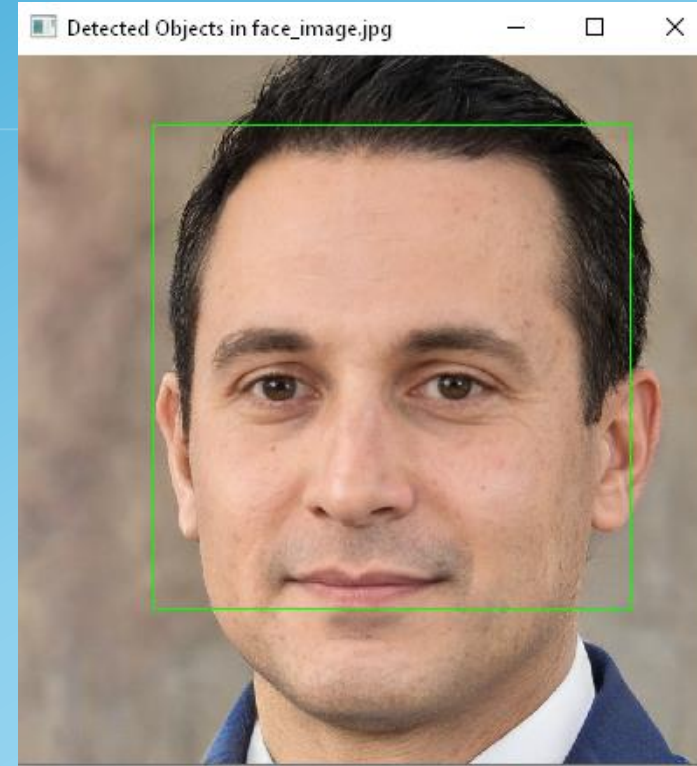
- 사용자의 얼굴을 미리 학습을 진행한 뒤, 학습된 사용자가 인식이 되면 잠금이 해제

5. 얼굴 검출 및 인식(Used library functions)

- ▣ 1. Haar Cascade Classifier (Opencv Library)
- ▣ 2. LBPH (Local Binary Pattern Histograms)
- ▣ 3. Dlib (Development Library)

1. Haar Cascade Classifier (Opencv Library)

1. 카메라를 통하여 사용자의 얼굴을 검출하기 위해 Haar Cascade Classifier 를 사용
2. 검출된 사용자 얼굴 이미지들을 지정된 디렉토리에 저장
3. 디렉토리에서 이미지 파일을 읽어와 흑백 이미지로 변환한 후 학습 데이터로 사용할 수 있게 배열에 저장, 모델 생성
4. 각 배열에 저장된 이미지들을 정수형 데이터로 변환 학습데이터 레이블

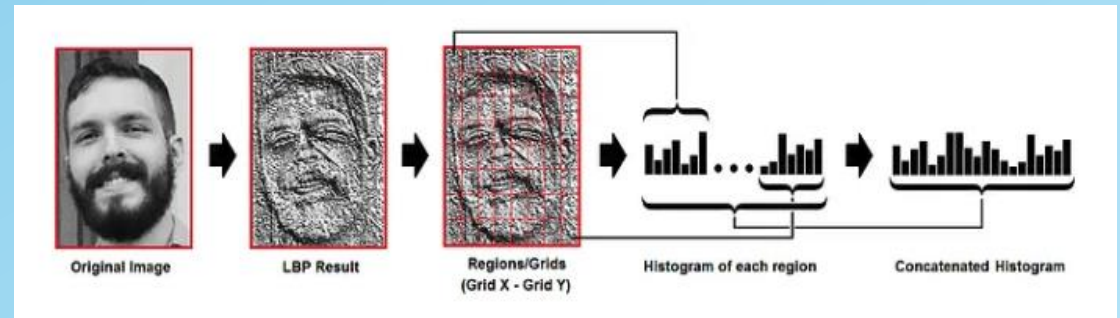
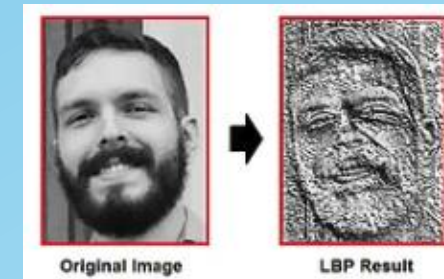
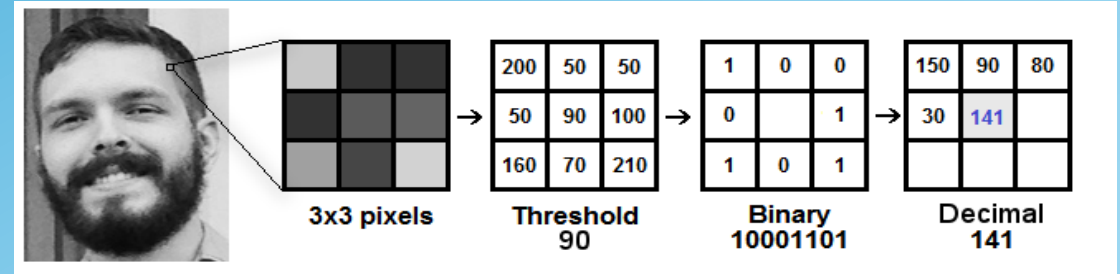


객체 검출을 위한 특징 기반 분류기의 하나

2. LBPH (Local Binary Pattern Histograms)

1. LBPH 알고리즘을 사용 얼굴 인식
모델 객체를 생성

2. 저장된 모델 데이터를 사용 사용자의
얼굴을 인식하는 패턴 학습



LBPH 알고리즘의 원리

1. Local Binary Pattern

각 픽셀에 대한 이진 패턴을 생성

이진 패턴은 픽셀의 이웃 픽셀과의 비교를 기반으로 생성

각 픽셀에 대해 이웃 픽셀들과의 비교를 통해 생성된 이진 패턴은 해당 지역의 텍스처를 나타냄

2. Histograms

각 이미지 영역의 LBP 패턴을 히스토그램으로 표현 이미지를 그리드로 나눔

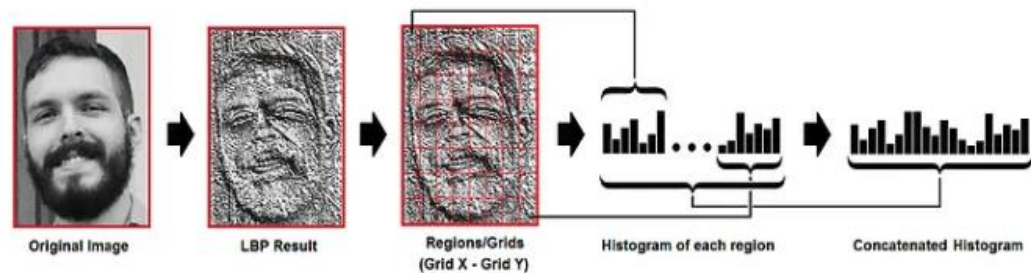
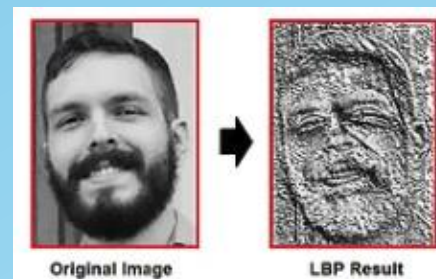
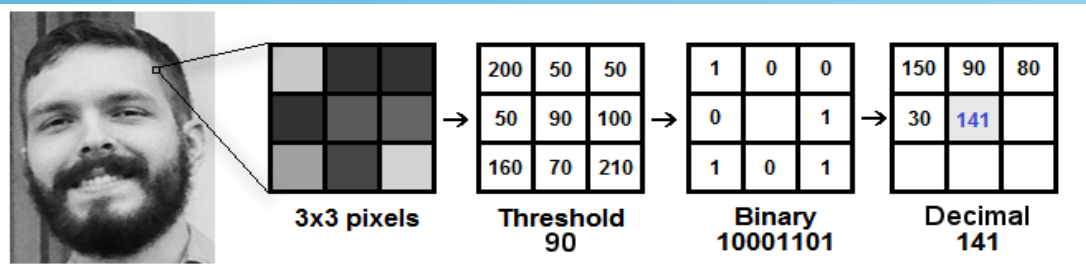
각 그리드에 대한 LBP 히스토그램을 계산

3. Local Patterns and Labels

LBP 패턴은 이웃 픽셀 간의 상대적인 밝기 차이에 기반, 특정 픽셀의 밝기가 이웃 픽셀보다 크면 1, 그렇지 않으면 0으로 표시

4. 인식

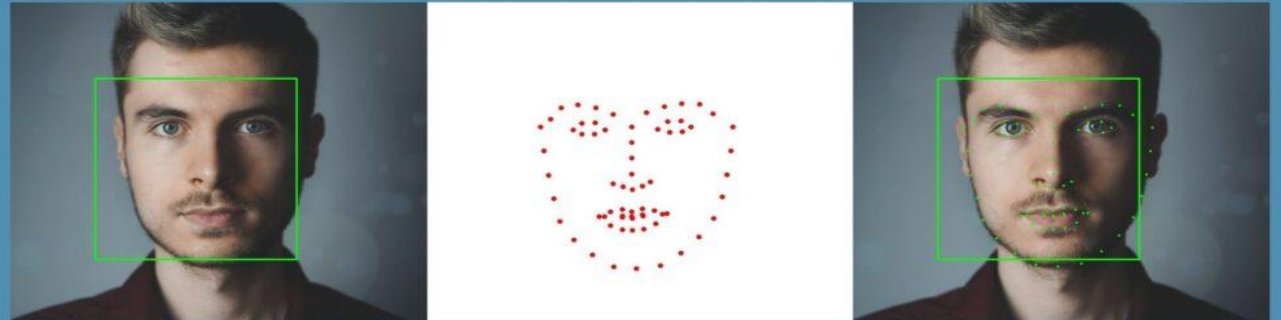
LBPH를 사용하여 이미지의 각 지역에 대한 텍스처 히스토그램을 생성한 후, 다양한 이미지들 간의 텍스처 차이를 측정



3. Dlib (Development Library)

1. 카메라를 통해 사용자의 얼굴을 확인
2. 학습된 알고리즘을 통해 사용자의 얼굴 인식

Face Landmark Detection using Dlib



C++로 개발된 고성능의 머신러닝 및 이미지 처리 라이브러리로, 얼굴 감지, 얼굴 인식, 이미지 유사성 평가, 객체 검출 등의 기능을 제공

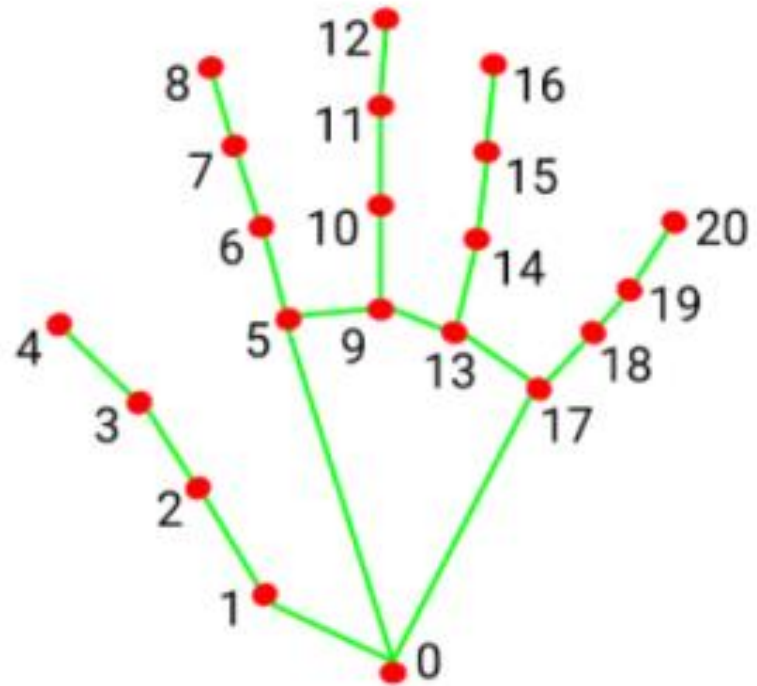
6. 손 인식

- 사용자의 손을 미디어 파이프를 사용하여 검출, 검출된 손에 랜드마크를 표시
- 표시된 랜드마크의 좌표를 비교하여 손 모양을 구분, 구분된 손 모양에 따라 명령어 출력

1. MediaPipe

- MediaPipe Hands는 고성능 손, 손가락 추적 솔루션으로 이번 작품에서 로봇의 동작을 조종 할 수 있는 기능을 맡고있다.
- 전체 이미지에 대해 손바닥을 감지한 후 손 랜드마크 모델은 2차 분석을 통해 감지된 손 영역 내에서 21개의 3D 손 너클 좌표에 대한 정확한 키포인트 위치 파악을 수행한다. 이 모델은 머신 러닝으로 학습하며 부분적으로 보이는 손이나 겹치는 손의 경우에도 인식할 수 있다.
- 랜드마크 모델을 활용하여 사용자의 손 모양을 판별하고, 모양에 따른 로봇 동작 명령어를 출력한다.

Hand landmark



- 손가락 랜드마크 번호간 길이를 비교 손가락을 접었는지 폈는지 판단
ex) 검지 -> 0번~8번 길이가 0번~6번 길이보다 짧으면 접힌 걸로 판단
- 손가락을 하나만 폈을 경우 영상 속 손가락의 랜드마크 x축을 비교하여 좌,우 방향을 판단

손 모양 인식 영상

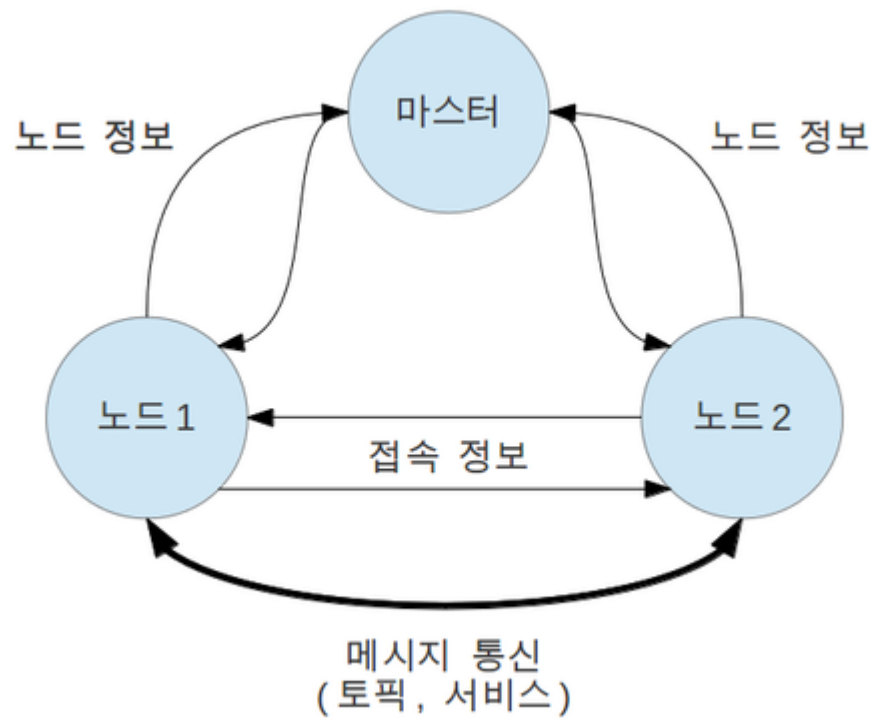


7. ROS 통신

- 작품은 로봇 응용 소프트웨어 개발을 지원하는 운영체제인 ROS를 사용하여 여러 node사이의 통신을 진행
- 메인보드인 raspberry pi와 주행로봇의 arduino사이의 통신은 roserial을 사용.

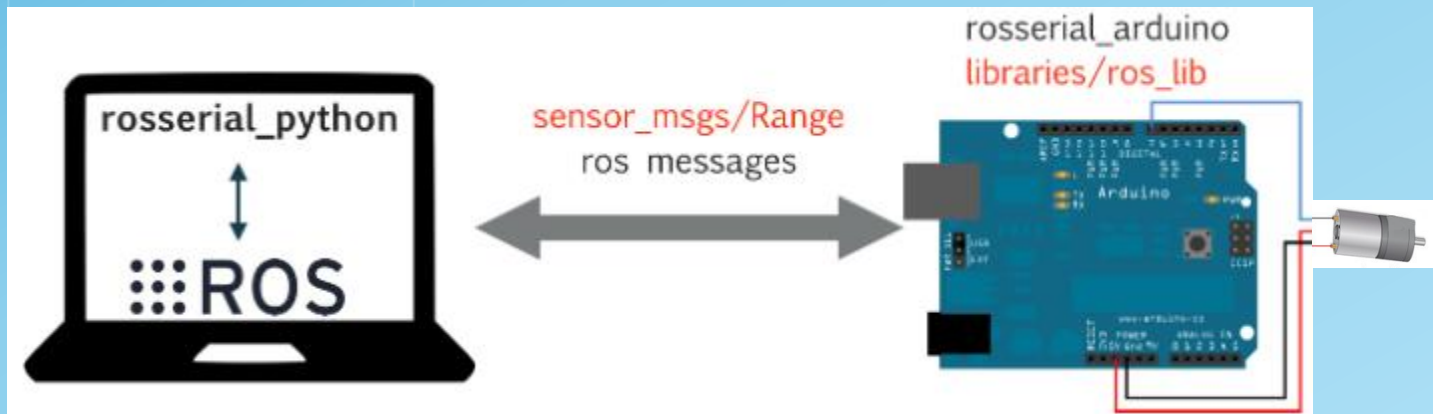
ROS

- ROS에서 사용되는 통신 시스템은 메시지(msg), 액션(action), 서비스(svc)로 크게 3가지로 나타낼 수 있다.



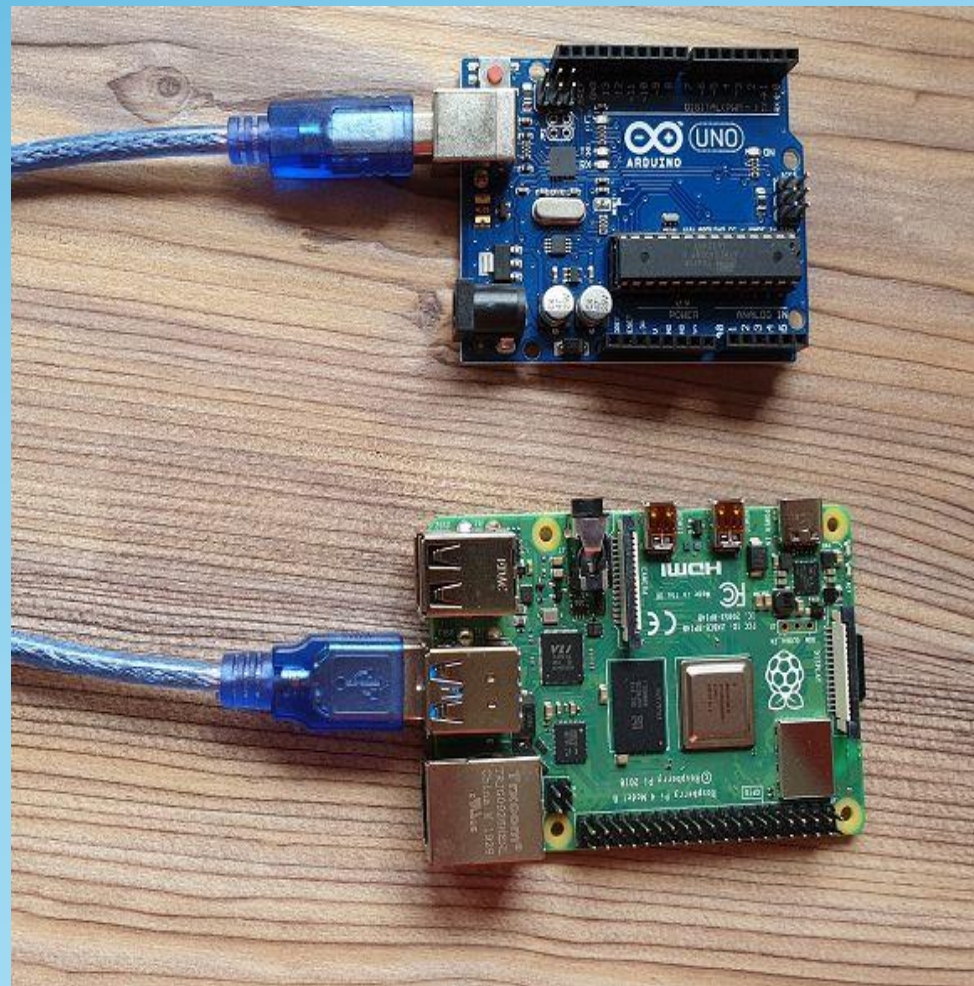
Rosserial

- 마이크로 컨트롤러와 컴퓨터 간의 메시지 통신을 위해 변환 작업을 수행하는 패키지
 - * 제어기 -> 시리얼(rosserial 프로토콜) -> PC(ROS 메시지로 재전송)
 - * 제어기 <- 시리얼(rosserial 프로토콜) <- PC(ROS 메시지를 시리얼로 변경)
- 일반적으로 마이크로 컨트롤러는 ROS에서 기본 통신으로 사용하는 TCP/IP보다 시리얼 통신을 많이 사용해 Rosserial과 같은 중계자 역할이 필요하다.

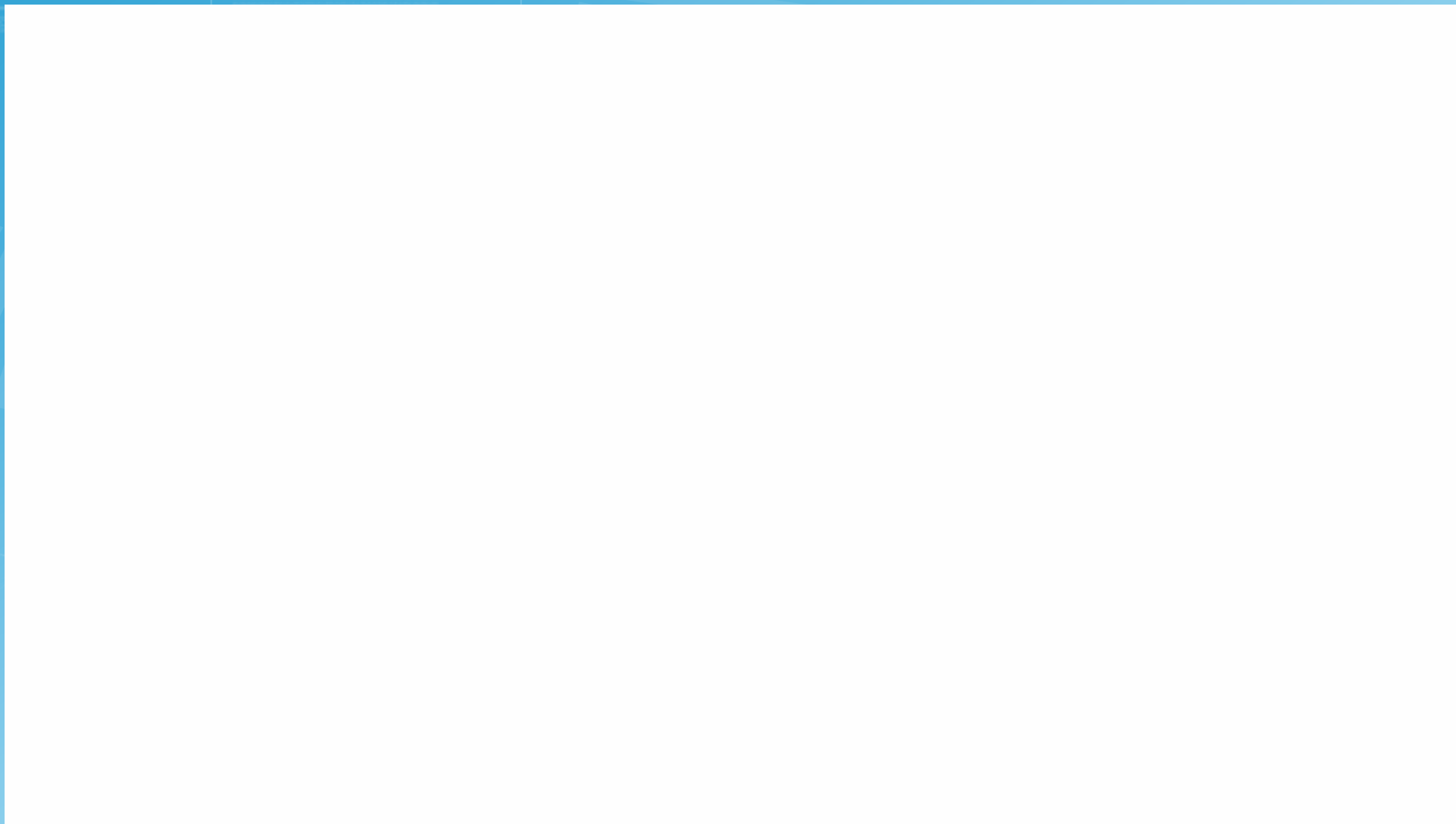


ROS 통신 과정

- ① 라즈베리파이에서 Roscore(master)를 실행
- ② 라즈베리파이와 아두이노를 USB 포트를 이용하여 연결
- ③ 라즈베리파이와 아두이노 Rosserial 관련 패키지를 설치
- ④ port 및 통신속도와 topic을 정의
- ⑤ 지정된 topic으로 메시지를 송수신



손 모양에 따른 Ros 메시지 출력 영상



8. 동작 영상

Q&A

