|  |
| --- |
| **자율주행 및 머신러닝을 통한**  **책 분류 로봇** |

**2019. 12. 27**

**비트교육센터**

**과정명 : 머신러닝을 활용한 자율주행 머신 개발자 양성과정**

**성명 : 김장겸, 박주언, 여재영, 윤상민**

**목 차**

**1. 프로젝트 선정 개요..........................................................................................○**

**2. 프로젝트 범위 및 구성....................................................................................○**

**3. 제안 시스템.......................................................................................................○**

**1) 소프트웨어 아키텍처....................................................................................○**

**2) 구현................................................................................................................○**

**4. 개발 도구 및 일정.............................................................................................○**

**1) 개발 도구.........................................................................................................○**

**2) 개발 일정..........................................................................................................○**

**5. 기타.....................................................................................................................○**

**1) 시스템 요구사양...............................................................................................○**

**2) 참고문헌...........................................................................................................○**

**6. 개발인원............................................................................................................○**

**1. 프로젝트 선정 개요**

도서관이나 서점에는 수많을 책을 사람들이 하나하나 분류 하기에는 너무나 많은 양과 다양한 종류가 있어서 인력을 사용해서 책을 분류하는건 많은 인적자원과 시간을 소비하는 작업이다.

본 프로젝트는 머신러닝 및 자율 주행 기술을 활용하여 이러한 작업을 도와줄 수 있는 로봇을 개발하면 어떨까 하는 아이디어에서 시작하였다.



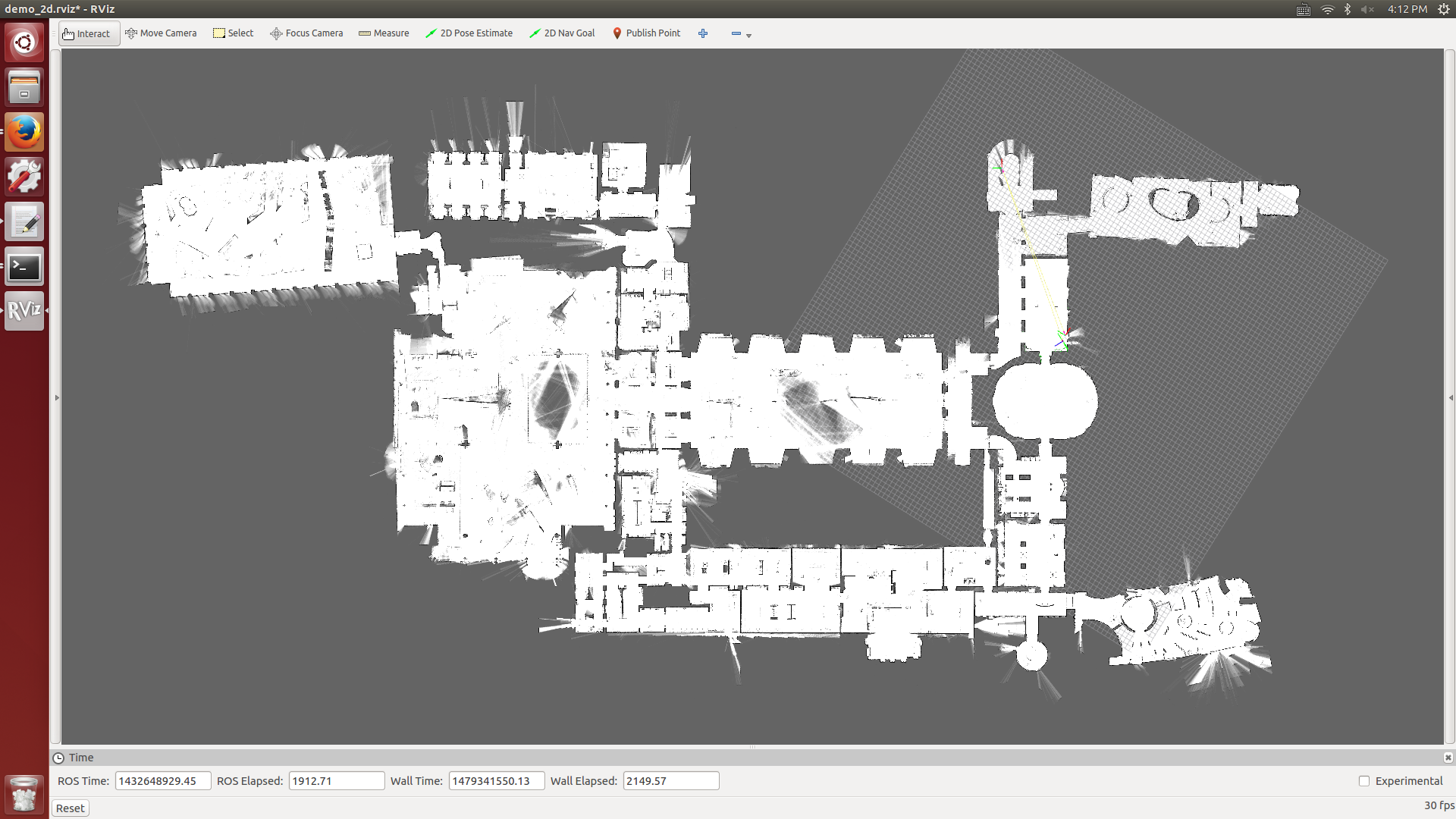
전체적인 진행과정은 자율주행과 이미지 분류, 이미지 인식을 이용하여 자율주행으로 원하는 지점으로 이동 후 이미지 분류를 통해 사물을 개별적으로 분류하고 마지막으로 분류한 이미지를 인식하여 데이터베이스에 있는 정보와 비교를 해 사용자에게 책이 잘못된 위치에 있다는 것을 알려줌을 목표로 하였다.

이 프로젝트는 책장이나 선반에 물체를 분류하고 인식하는 것이므로 나중에 더 나아가서 물류센터에서 물류를 분류하는 등 분류할 물건이 많고 종류가 다양하며 공간이 넓은 곳에서 사용할 수 있을 것으로 보인다.

**2. 프로젝트 범위 및 구성**

프로젝트 목표가 도서관이나 서점을 목표로 하였으므로 대표적인 자율주행인 차선주행을 쓰기 적합하지 않으므로 2D Lidar 및 IMU 센서 데이터를 활용하였다.

Google에서 제공하는 Google Cartographer로 map을 제작 하였으며 map을 불러와 저장된 데이터와 현재 데이터를 비교하여 현재 위치 및 각도를 파악한다.

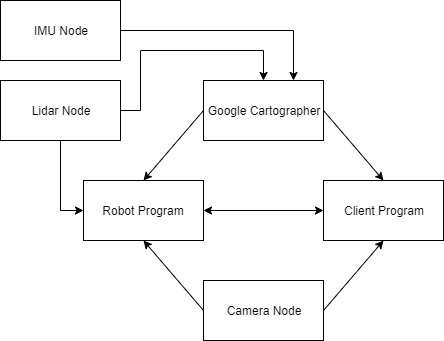


<Google\_Cartographer로 만든 map>

위에 과정이 끝난 후 지정된 책장 위치로 자율 주행을 한 후에 책장을 인식하여 촬영을 시작하고 촬영 후 edge검출로 책을 나눈 후 Vocabulary Tree 기반 이미지 비교 알고리즘을 통해 잘못 꽂힌 책이 있는지 검사한다.

**3. 제안 시스템**

**3.1 소프트웨어 아키텍처**



본 프로젝트에서 작성한 소프트웨어는, 위 구조도와 같이 다양한 프로그램들이 연동되어작동한다. 프로그램들 간의 통신은 로보틱스 분야에서 널리 사용되고 있는 ROS를 사용하여 구현하였다.

1. 각종 센서 노드(IMU,Lidar, Camera Node)

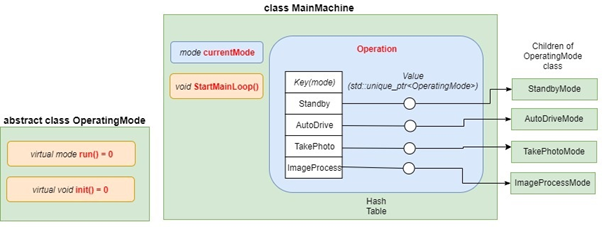
각각의 센서에서 읽은 데이터를 다른 프로그램으로 주기적으로 전송한다.

2. GoogleCartographer

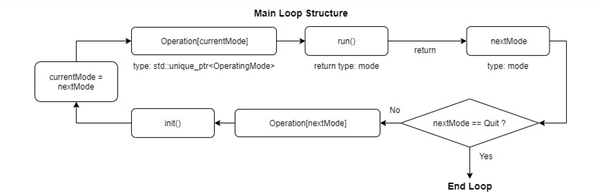
Google사에서 만든 오픈 소스 SLAM 라이브러리로, IMU 센서와 2D-Lidar 센서 데이터를 사용하여 맵 정보와 로봇의 위치 정보를 계산한다.

3. Robot Program

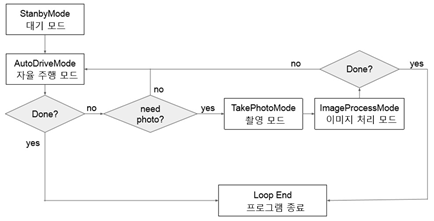
로봇을 컨트롤하는 프로그램으로 대기모드, 자율주행 모드, 촬영 모드, 이미지 처리 모드의4가지 모드를 가지고 있다. 프로그램의 전체 클래스 구조는 아래 그림에 간략히 나타내었다. 전략 패턴을 이용하여 각 모드별 실제 기능을 담당하는구현부와 전체 모드들을 관리하고 모드들 간의 공통적으로 필요한 기능을 담당하는 MainMachine 클래스를 분리하여 코드 중복을 줄이고 기능의확장성을 용이하게 하였다.



Robot Program은 기본적으로 하나의 메인 루프를 갖는데, 그 구성은 다음 그림과같다.각 모드 들의 구현부는 공통적으로 init 메소드를 통해 각 기능에 필요한 초기화 과정을 거치고, run 메소드를 통해 실제 기능을 수행한다.run 메소드는 다음으로 실행할 모드 정보를 enum class로 반환한다.



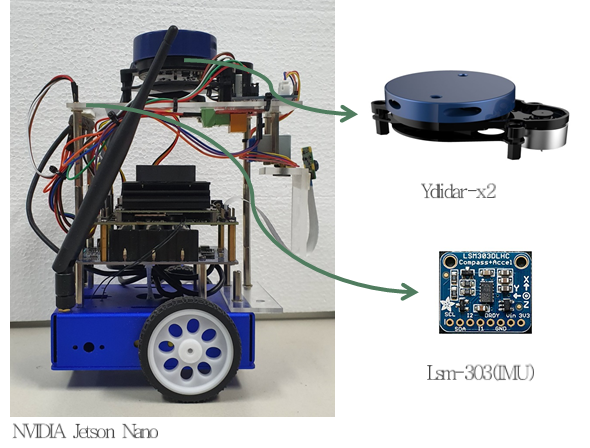
모드 간의 전환은 아래에 나타낸 순서도와 같이 이루어진다.



<프로그램 순서도>

**3.2 구현**

**(1) 기계**



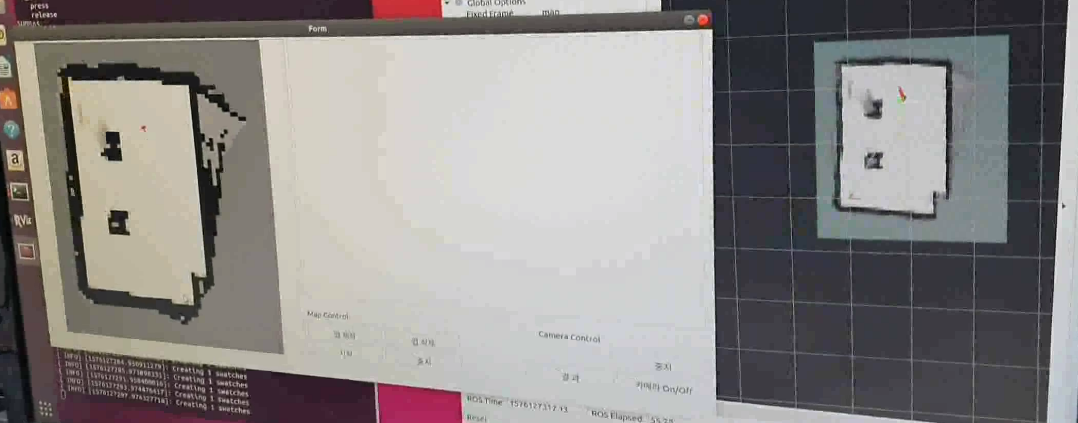
jetson nano를 활용한 jetbot 사용하여 자동차의 기반으로 사용했고 Ydlidar 센서와 Lsm-303(IMU)을 아크릴 판과 지지대를 사용하여서 책장 높이에 맞게 설계하였다.

앞에 사진을 찍는 카메라는 pi-camera를 사용했으며 나중에 더 자세한 각도 조절을 할 수 있게 step motor와 pi-camera를 겹합하여 완성하였다.

**(2) 자율 주행 & Cartographer**

Lidar와 IMU의 정보를 가지고 Google Cartographer에 보내주어 map 제작한다.

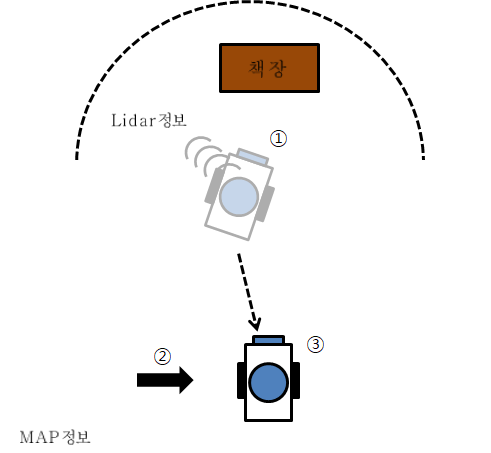
jetbot을 사용하여 map을 돌아다니며 센서들의 정보로 map을 만들고 저장하면 나중에 map을 불러들일때 현재 정보와 불러드린 정보를 비교하여 자기 자신의 현재 위치를 잡고 원하는 지점을 정하고 지정된 위치까지 A\* 알고리즘을 활용하여 선택한 지점까지 최단 경로로 이동한다.



<map 제작>

**(3)촬영**

먼저 자율주행을 통해 지정한 책장의 좌표로 이동한다. 그 후 Lidar 센서와 Map data를 이용하여 정확한 책장의 위치와 각도를 계산한다, 계산 결과를 바탕으로 촬영에 적합한 위치로 이동하고 책장 방향으로 회전 후 사진을 촬영한다.



촬영

**(4)Objetc Detection**.

촬영한 이미지에서 배경과 책장 부분을 분리하기 위해 Object Detection을 사용하였다. Object Detection 알고리즘으로는 Deep Learning 기반 알고리즘인 YOLO 를 사용하였다. YOLO는 다른 Deep Learning 기반 Object Detection 알고리즘들에 비해 동작속도가 빨라 리얼 타임 동작이 가능하다는 장점을 갖고 있어 본 프로젝트에서 채택하였다. 본 프로젝트에선 YOLO를 사용하기 위해 Tensorflow 기반의Open Source 프로젝트인 Darkflow를 사용하였다. 사용한 모델은 직접 촬영한 이미지를 이용하여 학습하였으며 학습에 사용된 이미지는 약 2500장이다.

학습 한 모델을 Client Program에서 동작하여, 촬영한 이미지에서 책장 부분만을 인식하고 그 정보를 Robot Program으로 전송하여 책장 부분 이미지만을 추출 할 수 있도록 한다.



<책장 인식 결과>

**(5) 개별 책 분리**

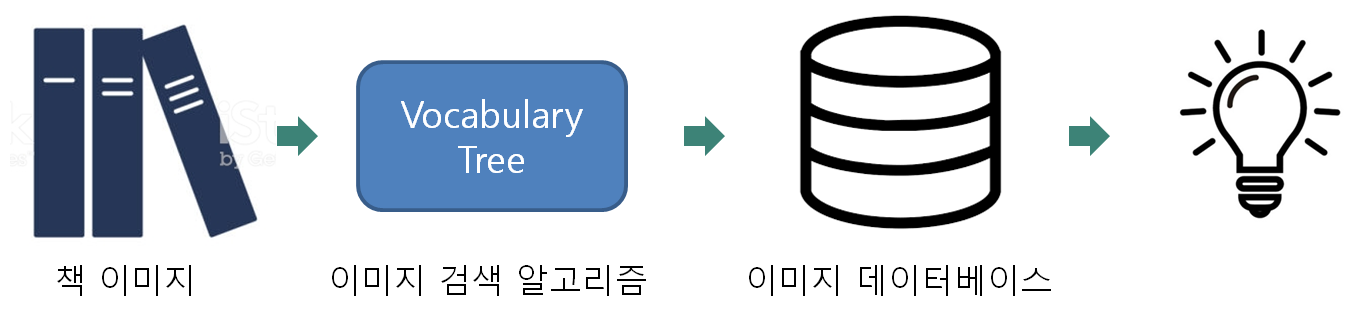
Opencv의 Edge검출을 활용하여 책을 개별로 분류한다. 먼저 책장 이미지에서 Canny Edge Detection을 사용하여 Edge를 검출한다. 검출한 Edge 이미지에서 짧은 Edge들을 제거한 Edge 이미지를 만든다. 이는 책 표지의 텍스트와 그림 등을 제거하고 책과 책 사이에 생기는 긴 Edge만을 추출하기 위함이다. 짧은 Edge를 제거한 Edge 이미지에서 Hough 변환을 통해 수직에 가까운 직선들을 추출하여 책들을 분리할 선들을 만든다. 이 때 전처리 전의 Edge 이미지를 기준으로 선들로 나눠지는 구역 안에 Edge 성분이 일정 픽셀 이하인 구역은 책이 아닌 것으로 판단하고 무시한다.



<edge 검출을 활용한 책 분류>

**(6) 책 인식**

분리해낸 책 이미지는 데이터베이스에 있는 미리 저장된 책 이미지들과 비교하여 가장 유사한 이미지를 찾는 방식으로 책을 인식한다.

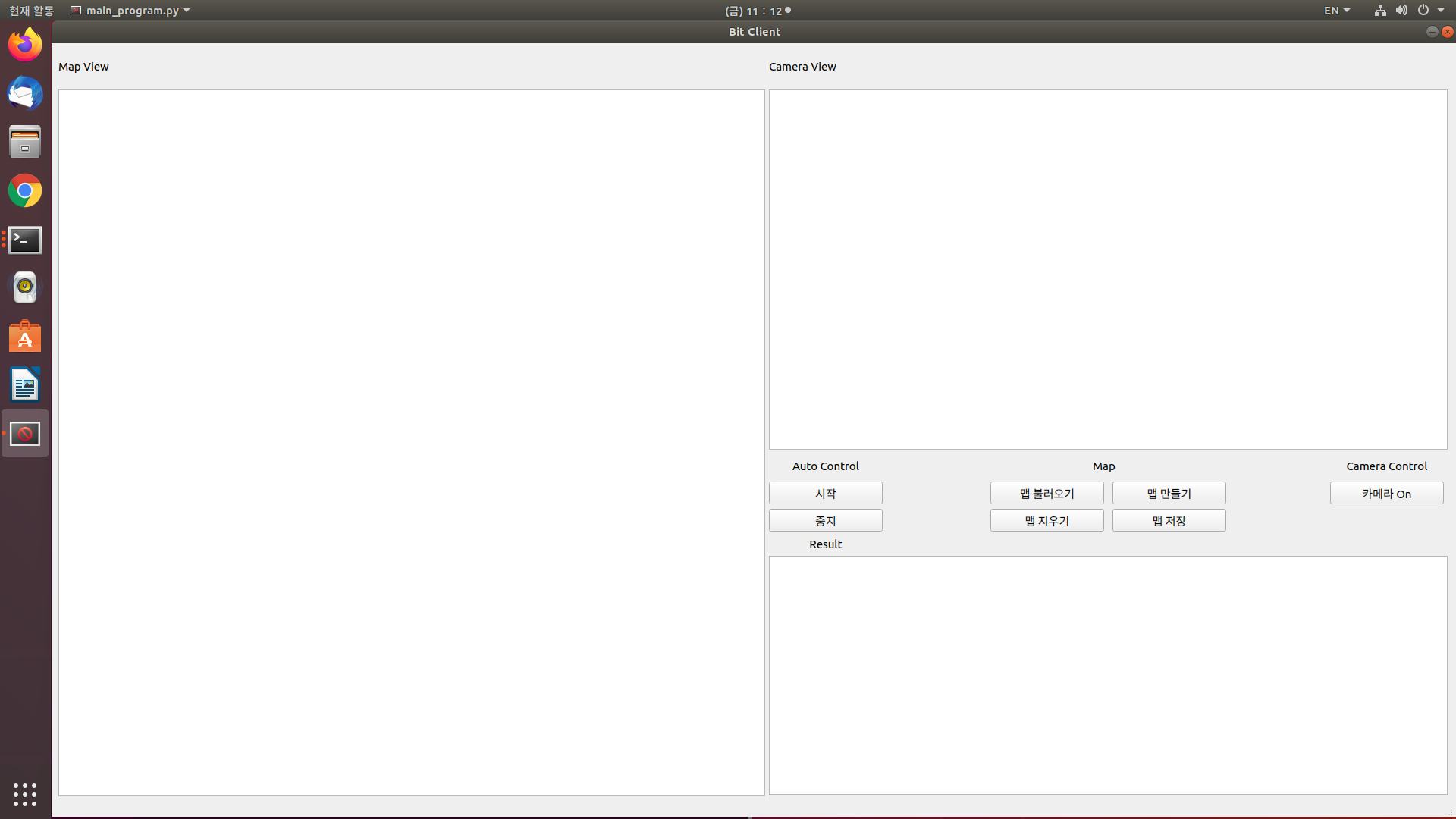


<책 분류 & 인식 전체 과정>

데이터베이스 내에서 유사한 이미지를 찾는 검색 알고리즘으로는, 대규모 이미지를 효율적으로 검색할 수 있는 Vocabulary Tree 알고리즘을 사용하였다. Vocabulary Tree 알고리즘은 컴퓨터 비전 분야에서 널리 사용되고 있는 Bag of Words 알고리즘에 기반한 알고리즘이다. 각각의 이미지에서 특징점을 분리한후 비지도학습 머신러닝 알고리즘인 K-Means Clustering을 사용하여 Visual Words를 만들고, 각각의 Visual Wods들의 빈도수를 벡터화 하여, 벡터간의 유사도를 측정하거나 다른 머신러닝 알고리즘의 입력값으로 이를 활용한다. 본 프로젝트에서는 특징점 추출 알고리즘으로 OpenCV Lab의 ORB를 사용하였다. 이는 빠른 속도와 적은 메모리 사용량을 가지고 있으며 상용이용이 가능한 라이센스를 갖고 있기 때문이다. ORB는 특징점의discripter 벡터를 256비트의 이진수 형식으로 나타내기 때문에 일반적인 K-Means Clustering을 적용할 수 없다. 이를 해결하기 위해 관련 논문을 참고하여 이진수 형식의 특징 벡터에 대해서도 K-Means Clustering과 유사한 Clustering이 가능한   
K-Majority 알고리즘을 구현하여 사용하였다.

**(7) Clinet / GUI**

GUI 구성에는 PySide2를 사용하였다.

PySide2는 Python으로 Qt를 사용할 수 있게 하는 Qt에서 제공되는 Library로 이를 사용하여 쉽게 GUI를 구성할 수 있으며 QtCreator를 통해 GUI구성 후 생성된 .ui 파일을 Python 파일로 변환하여 사용할 수도 있으며 본 프로젝트에서는 해당 방식으로 GUI를 구성하였다.

<완성된 UI>

Client 프로그램에 구현 된 기능은 다음과 같다,

맵 제작 기능

- Google의 Cartographer를 사용하여 맵을 작성하는 기능.

맵 불러오기 기능

- 맵 제작 기능으로 제작 된 맵을 load.

자동 주행 기능

- 불러온 맵을 탐색하여 책장의 위치로 이동하여 본 프로젝트의 목표인 책이 정위치에 존재하는 지를 파악하는 기능.

카메라 뷰 기능

- 주행중인 로봇이 촬영하는 영상을 Client에서 확인하기 위한 기능, 버튼을 통해 On/Off가 가능.

결과 확인 기능

-Result화면에 자동 주행 기능으로 확인 된 책들의 정보를 확인 할 수 있는 기능.

**4. 개발 도구 및 일정**

**4.1 개발 도구**

개발 언어: C++, Python

library/framwork : ROS(mellodic), OpenCV 4.0, Tensorflow 1.14(darkflow), Cartographer, YOLO v2

개발 도구 : Atom, Visual Studio 2019, QtCreator

**4.2 개발 일정**

|  |  |  |
| --- | --- | --- |
| 개발 일정 | 개발 내용 | 기타 |
| 10. 01 ~ 10.11 | 프로젝트 주제 및 언어, 도구 선정 |  |
| 10.14 ~ 10.18 | jetbot 조립, 사용법 연구 |  |
| 10.21 ~ 10.25 | jetbot에 lidar, step motor등 아크릴을 사용하여 연결(로봇 제작) |  |
| 10.28 ~ 11.15 | Ros와 cartographer 사용 |  |
| 11.18 ~ 11.22 | darkflow를 사용한 책장인식 |  |
| 11.25 ~ 11.29 | map 인식을 통한 자율 주행 연습 |  |
| 12.02 ~ 12.13 | 책 분류 후 책인식 프로그램 작성 |  |
| 12.16 ~ 12.20 | ui 제작 및 구동 확인 |  |
| 12.23 ~ 12.26 | 모든 프로그램 연결 후 최종 확인 |  |

**5. 기타**

**5.1 시스템 요구사항**

사용 장비 : NVIDIA Jetson Nano, NVIDIA Jetson TX2, NVIDIA GPU-RTX 2070

개발 환경: Windows10, Ubuntu 18.04

**5.2 참고 문헌**

[1] D. Chen, S. Tsai, K.-H. Kim, C.-H. Hsu, J.P. Singh, and B. Girod."Low-cost asset tracking using location-aware camera phones." Number1, San Diego, California, USA, 2010.

[2] Sam S. Tsai , David Chen , Huizhong Chen, Cheng-Hsin Hsu, Kyu-HanKim , Jatinder P.Singh , Bernd Girod,"Combining image and text features: a hybrid approach to mobile book spinerecognition", Proceedings of the 19th ACM international conference onMultimedia, November 28-December 01, 2011, Scottsdale, Arizona, USA

[3] D. Nister and H. Stewenius."Scalable recognition with a vocabulary tree." In Proc. CVPR, 2006.

[4] Li Fei-Fei, RobFergues,Antonio Torralba,"Recognizing and Learning Object Categories", ICCV 2005.

[5] Grana C, Borghesani D, Manfredi M, Cucchiara R,"A fast approach for integrating ORB descriptors in the bag of wordsmodel." In: Proc. SPIE 8667, Multimedia Content and Mobile Devices. SPIE,Burlingame, California, USA, pp 866709-866709-8, 2013.

[6] E. Rublee, V. Rabaud, K. Konolige, G. Bradski,"ORB: An efficient alternative to SIFT or SURF", Proc. IEEE Int.Conf. Comput.Vision, pp. 2564-2571, Nov. 2011.

**6. 개발 인원**

|  |  |
| --- | --- |
| **사 진** | **인 적 사 항** |
|  | 성 명 : 김장겸  최종학력 :동양미래대 학사학위  연 락 처 : 010-7455-5310 / E-mail : kjk7455@gmail.com  담당업무 : Client Program/ GUI  사 용 툴 : Atom Editor, Qt\_creater |
|  | 성 명 : 박주언  최종학력 : 동경대학 대학원 박사과정 중퇴  연 락 처 : 010-9080-0000 / E-mail bluegbgb@gmail.com  담당업무 : 팀장/이미지 인식/전체 아키텍쳐 설계  사 용 툴 : Visual Strudio, 2019, Visual Strudio Code |
|  | 성 명 : 여재영  최종학력 : 동양미래대학 졸업  연 락 처 : 010-6549-6541 / E-mail : blossombudle@naver.com  담당업무 : 자율주행  사 용 툴 : visualstudio 2019, Qt\_creater |
|  | 성 명 : 윤상민  최종학력 : 동아대학교 졸업  연 락 처 : 010- 9481-8776/ E-mail heroes0104@naver.com  담당업무 : 객체인식  사 용 툴 : tensolflow, darkflow |