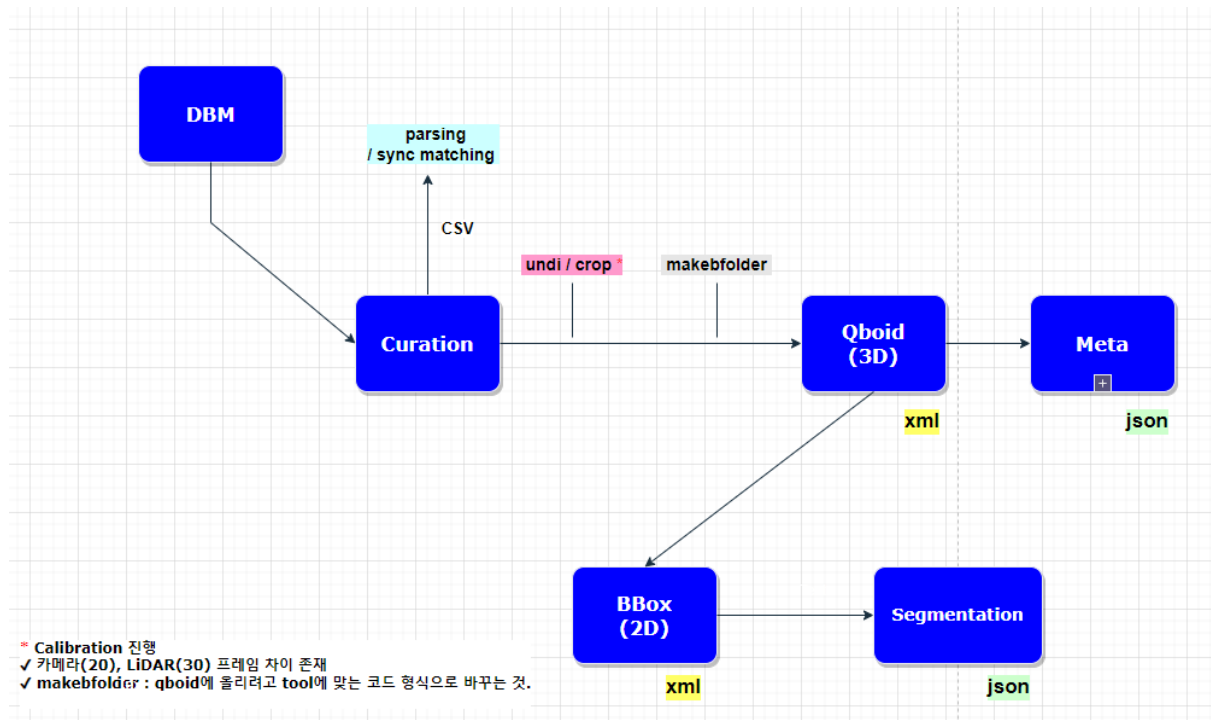


알체라 인턴

자율 주행 모델을 위한 데이터 구축

전체적인 데이터 구축 프로세스



데이터 종류

1. 북미, 유럽 도로 영상 데이터
2. 카메라 영상 데이터 & 라이다 데이터로 구성

데이터 수집 과정

1. 원본 데이터 중에서 bin 파일로 들어온 라이다 파일을 파싱작업을 통해 프레임 별로 쪼갬(그 과정에서 파일 형식은 bin-pcd-bin형태로 변형)
 - **bin 파일** : "binary"의 줄임말로, 데이터를 이진 형식으로 저장한 파일을 의미함. bin 파일에는 텍스트나 이미지 등 다양한 종류의 데이터가 포함될 수 있고 LiDAR 센서에서 생성된 3D 포인트 클라우드 데이터를 저장하는 데 종종 사용되며, 이러한 경우 각 점의 x, y, z 좌표와 그 점의 반사 강도(intensity) 등이 이진 형식으로 저장됨

- **pcd 파일: "Point Cloud Data"의 약자로, 3D 포인트 클라우드 데이터를 저장하기 위해 Point Cloud Library(PCL)에서 사용하는 형식으로 pcd 파일은 ASCII(텍스트) 형식 또는 이진 형식으로 저장될 수 있으며, 각 점에 대한 x, y, z 좌표 외에도 RGB 색상 값 등 추가 정보를 포함할 수 있음**
1. 영상 데이터는 yuv파일 형태로 받아옴
 - a. yuv : 컬러 이미지 파이프라인 의 일부로 사용되는 색 인코딩 시스템이다. 컬러 이미지나 비디오 토크 지각 변환을 통해 색차 컴포넌트의 대역폭 감소를 가능케 하므로 일반적으로 직접적인 RGB 표현을 사용할 때보다 전송 오류나 압축 가공물을 더 효율적으로 마스킹될 수 있게 한다.
 2. 영상 데이터를 큐레이션 작업을 통해 3D → 2D로 변환하는 과정에서 undi(구부러짐)작업을 실시(이 부분에 대한 설명이 제대로 기억이 안남;)한 후에 이질감(?)을 없애고 crop 진행 ⇒ 라이더와 사진 상의 간극을 줄이는데 필요한 작업
 3. **Calibration 정보 활용: Calibration은 센서 데이터가 정확하게 해석될 수 있도록 센서의 오차나 왜곡을 보정하는 과정이다. 예를 들어, 카메라와 LiDAR 센서 사이의 상대적 위치나 방향에 대한 정보가 Calibration 정보에 포함됨**
 4. **makebbfolder : 위의 작업을 진행하고 나온 파일을 큐보이드 툴에서 작업하기 위한 변환 작업**
 5. **csv파일 만드는 원인에 대해 제대로 기억이 나지 않음 보강요망** ⇒ 작업관리도 있지만 nas용량이 전체작업에 대해서 한꺼번에 파싱을 하기에는 버거워서 진행상황에 맞춰서 파싱을 진행하기 위한 용도로도 사용

NAS : 네트워크 연결 스토리지(NAS)는 직원들이 네트워크를 통해 효과적으로 협업할 수 있도록 데이터터를 지속적으로 사용할 수 있게 하는 파일 전용 스토리지 디바이스를 말함

데이터 처리 과정

1. 데이터 수집 과정으로 얻어진 데이터를 시간, 날씨, 도로, 조도, 카메라, 객체, 대상, 움직임, 위치, 방향, 주행 이벤트의 요소들을 기준으로 case 분류 작업 실시
2. 케이스 분류를 한 데이터들을 토대로 cuboid, bbox, segmentation 진행
 - a. cuboid : 객체의 라이더 점군과 이미지를 자체적인 기준에 맞춰서 라벨링을 전용 툴을 사용해서 진행하면 xml 파일 형태로 저장, xml파일에는 박스의 좌표값, 클래스, id, oc, tc값들이 저장되어 있음
 - b. bbox : 2D상황에서의 사진상의 객체의 면적과 대략적인 위치들에 대해서 작업을 진행하면 xml파일 형태로 저장

- c. segmentation : 객체의 아웃라인을 사각형의 형태가 아니라 객체의 형태대로 꼼꼼하게 따는 작업을 진행하면 json파일 형태로 저장, 가장 큰 목적은 형체를 정확히 따는 것이고 그래야 하는 이유로는 양질의 데이터를 위해 잡음(객체 이외의 픽셀들)이 없는 데이터를 확보하는 것이 가장 큰 목표이고 자율 주행 프로젝트에 사용되는 데이터이니 만큼 정밀한 객체 탐지가 필연적이라서 필요한 작업
 - d. **meta data : 날짜, 시간, 날짜 등 작업물에 대한 설명이 들어간 파일(메타데이터 자체가 다른 데이터를 설명해주는 데이터)**
3. 데이터 셋은 위의 4가지 작업을 통해 각각 xml, xml, json, json 파일로 받아지고 1개의 사진당 4개의 파일로 구성이 되어있는데 파일 형태 변환 코드를 통해 json파일 4개로 통일 시킴
- a. 처음부터 json으로 뽑지 않는 이유는 사용하는 툴에서 xml형태로 내보내기 때문!
 - b. ~~xml이 아니라 json으로 하는 이유는 관중이가 말해줬는데 까먹음(더 깔끔했다고 했나 읽기 쉽다고 했나..)~~

b는 밑에 따로 정리했음

4. 위의 데이터들 사이에서 각각의 파일들이 존재하지만 서로의 데이터 안에서 동일 객체에 대한 일관된 데이터 입출력을 위한 기준을 id로 잡아서 id가 주어진 규칙안에서 꼬이지 않도록 확인하는 것이 검수의 가장 큰 목적 ⇒ id가 꼬이면 위의 4가지 파일이 각각 객체마다의 정보를 가지고 있어서 의미를 가지게 되는데 그게 무의미해지는 상황 발생

진행하면서 어려웠던 점

- 라이다 기계의 성능정 한계(**가격대가 심각하게 비쌌 / 1대당 최소 5000만원에서 2억이상**) 때문에 사진과의 프레임 차이 때문에 라이더와 사진간의 간극이 생겨서 작업시 혼선이 생김(**라이다 기계 최대 프레임 20 / 고객사 요청 프레임 30**)
- 대체로 1차 작업된 작업물들에 대해 검수를 하는 작업을 진행했으나 각자 작업에 대한 이해도가 다르기에 작업물의 수준차이가 심했음(**인하우스 선호**)
- 큐보이드 작업에 대한 작업 기준이 상황에 따라 변경되어서 다시 일하는 상황이 생김(대향차선에 대한 기준 변경 등)
- 큐보이드 자체 툴에 대한 여러가지 불편한 사항들이 있음
 - 박스 크기 유지 및 위치를 그대로 가져가기 위해 이전 장의 데이터를 가져오는 L키를 사용하는 순간 프리랜서의 1차작업이 의미가 퇴색되고 검수가 아닌 처음부터 작업해야하는 상황 발생
 - 라벨링된 객체들에 대해 oc, tc를 일일이 넣어줘야함

추가 내용

안면 인식 데이터 구축

- 직접적인 프로젝트를 진행한 것은 아니지만 알체라에서 가장 강점을 두고 있는 안면 데이터에 대해서도 대략적인 구조를 파악
- 전체적인 프로세스는 비슷하나 안면 데이터에서는 얼굴 윤곽, 눈, 코, 눈썹, 입의 요소에 대해 라벨링을 진행하고 이를 json파일로 통일 시킨 후 데이터 구축
- 작업 정밀도 수준은 body skeleton labeling 정도의 수준이지만 안면 인식인 만큼 face landmark의 특색을 띄기도 함 부위가 얼굴로 제한되었을 뿐 라벨링의 정교함은 매우 높은 수준

기타 추가할 내용

xml보다 json 파일을 사용하는 이유

알체라 기준 :

xml -> json파일로 바꾸는 이유

거래처 모델 데이터 학습 포맷이 json이라서 json파일로 변경xml파일도 포맷만 맞으면 학습은 가능함변환시에는 이런식으로 변환됩니다

일반적인 상황 :

1. 간결성: JSON은 XML에 비해 훨씬 간결함. XML에서는 열고 닫는 태그를 모두 작성해야 하는데 동일한 데이터를 표현하는 데 더 많은 문자열이 필요한 반면 JSON은 간결한 구문을 사용하여 동일한 데이터를 표현 가능
2. 읽기 쉬움: JSON의 구조는 사람이 읽기에도, 컴퓨터가 파싱하기에도 더 쉬운편
3. 데이터 구조: JSON은 복잡한 계층적 구조를 가진 데이터를 다루기 쉽게 만들어줌
4. 처리 속도: 일반적으로 JSON을 파싱하고 직렬화하는 것이 XML보다 빠름

but 상황에 따라서는 XML을 더 많이쓰기도 하는데 주로 문서 기반의 시스템에서 XML이 많이 사용됨

