

ROS2 기반 Frustum-PointPillars 알고리즘을 이용한 실시간 3 차원 객체 탐지

박건석, 황정훈, 박민철*

한국전자기술연구원, 한국전자기술연구원 *한국전자기술연구원

rjstjr950@keti.re.kr, hwangjh@keti.re.kr, *mincheol.p@keti.re.kr

Real-time 3D Object Detection Using Frustum-PointPillars Algorithm Based on ROS2

Gunseok Park, Jung Hoon Hwang, Mincheol Park*

Korea Electronics Technology Institute, Korea Electronics Technology Institute, *
Korea Electronics Technology Institute.

요 약

덤프트릭 자율주행 환경에서 보행자와 같은 소형 객체를 높은 정확도로 인지하기 위해, 본 연구는 카메라 기반 2D 검출 결과와 LiDAR 거리·형상 정보를 결합하는 Frustum-PointPillars 계열의 파이프라인을 ROS 2 에서 구현하였다. 기존 3 차원 탐지기는 전 영역 점군을 처리하므로 연산량이 입력 점군 규모에 비례하여 증가한다. 본 구현은 2D bbox 로 정의된 관심영역(ROI)에 포함되는 점군만 선택하여 PointPillars 인코더에 입력함으로써, 인코딩 대상 점 수를 감소시켰다. 전처리는 Numba JIT 로 구현하여 점군 필터링과 정렬 연산을 가속하였다. 추론은 TensorRT FP16 으로 변환하여 네트워크 실행 시간을 단축하였다. 점군 샘플링은 객체까지의 거리(m)와 ROI 내부 점의 개수를 기준으로 수행하여, ROI 당 입력 점 수를 고정된 상한값 N 으로 제한하였다. 실험에서 전체 파이프라인은 프레임당 처리 시간 100 ms 미만을 유지했고, LiDAR 입력 주기 10 Hz 에서 실시간으로 3D bbox 를 산출하였다. 이는 제한된 연산 자원에서 3D 객체 인지를 수행하기 위한 구현 방법을 제시한다.

I. 서 론

자율주행 기술을 덤프트릭에 적용하기 위해서는, 주변 동적 객체 특히 보행자의 3 차원 위치를 센티미터 단위로 추정할 수 있어야 한다. 덤프트릭은 승용차 대비 차체 높이와 전방·측방 사각지대가 크고, 작업 구역에서는 보행자-차량 혼재 상황이 반복적으로 발생하므로, 보행자 3D 위치 추정 실패는 즉시 충돌 위험으로 연결된다. 따라서 덤프트릭 자율주행 인지 모듈은 저지연(실시간), 고정밀, 고재현율을 동시에 만족해야 한다.

3 차원 객체 탐지 기술은 점군 표현 방식에 따라 발전해 왔다. PointNet[1]은 점군을 직접 입력으로 사용하지만, 입력 포인트 수가 증가하면 연산량이 증가하여 고해상도 LiDAR 스트리밍 환경에서 처리 시간이 커진다. VoxelNet 은 3 차원 격자(voxel) 기반으로 특징을 추출하지만, 3 차원 컨볼루션이 요구되어 실시간 처리에서 지연이 증가한다. Lang 등이 제안한 PointPillars[2]는 점군을 수직 방향으로 기둥(pillar) 단위로 분할하고, 이를 2 차원 조감도(BEV) 피쳐 맵으로 변환한 뒤 2 차원 컨볼루션을 적용함으로써, 실시간성에 유리한 연산 구조를 제공한다.

그러나 덤프트릭 운용 환경에서 보행자는 차량 대비 크기가 작고 ROI 내부 점군 밀도가 낮아, 전 영역 탐색 방식에서는 배경 점군 처리 비중이 증가하고 원거리 보행자에서 유효 포인트 수가 감소하여 탐지 성능이 저하된다. 이를 줄이기 위해 이미지 기반 2D 검출 결과로 3D 탐색 공간을 제한하는 접근이 사용된다. Qi 등은 이미지의 2D 바운딩 박스를 3D 공간의 관심영역으로 제한하여 점군을 분석하는 Frustum PointNets[3]를 제시하였다. 본 연구는 Paigwar 등의 Frustum-PointPillars 개념을 기반으로 하되, 덤프트릭의 ROS 2 스트리밍 환경에서 LiDAR 입력 주기 10 Hz 를 고정적으로 처리하고 제한된 GPU 자원 내에서 실시간 동작이 가능하도록 파이프라인을 재구성하였다.

구체적으로, 본 구현은 3D frustum 을 명시적으로 생성하지 않고, 2D 객체 검출 결과로부터 정의된 ROI 에 포함되는 점군만을 기하학적으로 필터링하여 PointPillars 인코더 입력을 구성하였다. 전처리는 Numba JIT 로 구현하여 ROI 점군 추출과 정렬 연산을 가속하였고, 추론은 TensorRT FP16 으로 변환하여 네트워크 실행 시간을 단축하였다. 또한 ROI 내부 점 개수를 상한 N 으로 제한하는 샘플링을 적용하여,

덤프트릭 작업 구역에서 발생하는 점군 밀도 변동에도 프레임 처리 시간을 일정하게 유지하였다. 이 구성은 덤프트릭 기반 실차 환경에서 보행자 3D bbox 를 실시간으로 산출하기 위한 구현 구조를 제시한다.

II. 본론

본 파이프라인의 인지 처리 흐름은 ROS 2 에서 PointCloud2 와 YOLO 기반 DetectionArray 를 타임스탬프 기준으로 동기화하여 수신하는 단계에서 시작된다. 실시간성을 위해 입력 점군에 적응형 초기 샘플링을 먼저 적용하였다. 매 프레임 10 만 개 이상 포인트를 카메라 좌표계로 변환하는 연산은 지연을 유발하므로, 입력 포인트 수가 임계값을 초과하면 즉시 샘플링을 수행하여 변환 대상 포인트 수를 제한하였다. 그 결과 127,000 개 포인트를 약 40,000 개로 축소하여 좌표 변환 시간을 86 ms 에서 4 ms 이내로 단축했고, 전체 처리 시간을 100 ms 미만으로 유지할 수 있는 조건을 확보하였다. 전처리 코드는 Numba JIT 로 컴파일하여 파이썬 실행 오버헤드를 제거하였다.

좌표 변환 이후에는 2D 검출 바운딩 박스에 대응하는 고정 크기 ROI 를 사용하여 ROI 내부 점군을 추출하였다. ROI 크기는 프레임 간 변경하지 않았으며, 원거리 객체에 대해서도 동일한 규칙으로 ROI 를 적용하였다. ROI 에 포함된 각 포인트에는 2D 박스 중심으로부터의 거리에 기반한 가우시안 중심 확률 피치를 계산하여, 기존 포인트 속성에 추가 1 차원으로 결합하였다. 이후 voxelization 단계에서는 다운샘플링 비율을 포인트 수에 따라 조절하지 않고, 고정된 설정으로 포인트를 정규화하여 pillar 당 최대 100 개 포인트를 갖는 입력 텐서를 생성하였다.

추론 단계의 RPN 은 TensorRT FP16 으로 엔진화하여 지연 시간을 감소시켰다. 추론 결과로 생성된 후보 박스는 GPU 기반 NMS 로 정제하며, NMS 입력은 상위 300 개 후보 박스로 제한하여 연산량을 제어하였다. 또한 보행자 클래스에 대해 yaw 회전각을 0 으로 고정하여 회전각 추정을 생략했고, 3D 코너 계산 및 시각화 투영 과정에서의 삼각함수 연산을 제거하였다. 마지막으로 처리 시간의 지수 이동 평균(EMA)을 이용해 프레임 지연을 모니터링하여, LiDAR 10 Hz 입력 주기에서 안정적으로 동작하도록 구성하였다.

III. 결론

본 연구에서 구현한 Frustum-PointPillars 기반 3 차원 보행자 탐지 파이프라인은 PointPillars 의 pillar 기반 점군 인코딩과 2D 검출 기반 ROI 필터링을 결합하여, 전 영역 점군 처리에서 발생하는 불필요한 연산을 감소시켰다. 초기 단계에서는 입력 점군에 샘플링을 선행 적용하여 카메라 좌표계 변환의 연산량을 감소시켰고, 전처리 과정은 Numba JIT 로 가속하여 파이썬 실행 오버헤드를 제거하였다. 또한 추론 단계는 TensorRT FP16 엔진으로 최적화하여 신경망 실행 지연을 감소시켰다. 그 결과 전체 파이프라인의 프레임 처리 시간은 100 ms 미만으로 유지되었으며, LiDAR 10 Hz 입력 주기에서 보행자 3D bbox 를 실시간으로 산출하였다.

이 결과는 제한된 GPU 자원에서도 3 차원 보행자 인지가 가능함을 실험적으로 확인한 것이며, ROS 2 기반 실차 스트리밍 환경에서의 적용 가능성을 제시한다. 향후에는 보행자에 한정된 탐지 클래스를 차량 및 이륜차로 확장하고, 강우·안개·야간 조건에서의 성능 저하를 줄이기 위해 카메라- LiDAR 융합 과정의 정합 오차 보정과 학습 데이터 구성을 포함한 알고리즘 개선을 수행할 예정이다.

ACKNOWLEDGMENT

본 연구는 산업통상자원부의 기계장비산업기술개발사업(No. 00442974 , 대규모 토석 운반 자동화를 위한 덤프트릭용 자율작업 및 운영시스템 개발)의 지원을 받았습니다.

참 고 문 헌

- [1] C. R. Qi, W. Liu, C. Wu, H. Su and L. J. Guibas, "Frustum PointNets for 3D Object Detection from RGB-D Data," 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 2018, pp. 918-927.
- [2] A. H. Lang, S. Vora, H. Caesar, L. Zhou, J. Yang and O. Beijbom, "PointPillars: Fast Encoders for Object Detection from Point Clouds," 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 2019, pp. 12689-12697.
- [3] A. Paigwar, D. Sierra-Gonzalez, O. Erkent and C. Laugier, "Frustum-PointPillars: A Multi-Stage Approach for 3D Object Detection using RGB Camera and LiDAR," 2021 IEEE/CVF International Conference on Computer Vision Workshops (ICCVW), 2021, pp. 2926-2933.