

LIDAR 기반의 다중 물체 추적 알고리즘

이재준*, 유지환*

*한국기술교육대학교 기계공학과
e-mail : jaejun0201@koreatech.ac.kr

LIDAR based Multi-object Tracking Algorithm

Jae-Jun Lee*, Jee-Hwan Ryu*

*Dept. of Mechanical Engineering, Koreatech

요 약

본 논문에서는 현대 자율 주행 차량 경진대회에 적용되었던 LIDAR 기반의 다중 물체 추적 알고리즘을 소개한다. 물체 추적은 자율 주행 차량이 외부 환경을 인지하는데 중요한 역할을 한다. 본 논문의 물체 추적 알고리즘은 동시에 여러 개의 물체를 추적할 수 있도록 Multiple Data Association 방식을 사용하였고 순수하게 LIDAR 만으로 동작하기 때문에 밤과 낮 모든 경우에 적용 가능하다. 알고리즘은 Clustering, Data Association, State Estimation, Data Arrangement 총 4 단계로 이루어져 있으며 본 논문에서는 각 단계별로 알고리즘의 동작 방식을 소개한다. 실제 구현에는 Velodyne사의 HDL-32e 이 사용되었고 실제 주행에서 교차로 내의 차량 추적 및 선행 차량의 동향을 추적하는데 적용되었다.

1. 서론

LIDAR 의 원시 데이터(Raw Data)만으로는 차량의 주행 공간 상의 장애물 존재 여부만을 알아낼 수 있는 것에 비해, 원시 데이터를 적절하게 처리하여 분석하면 자율 주행 차량의 주행에 유용한 정보들을 추출할 수 있다. 특히 물체 추적 알고리즘은 LIDAR 의 원시 데이터로부터 차량 주변 물체의 정보 예를 들면 크기, 위치, 속도 등을 추출해 낼 수 있다. 이러한 정보들은 자율 주행 차량이 장애물 회피와 같은 기본적인 동작보다 더 높은 수준의 동작 예를 들면 선행 차량 추종, 교차로 상황 인식과 같은 동작을 구현하는데 필수적이다. 본 논문에서 소개하는 물체 추적 알고리즘은 여러 개의 물체를 동시에 추적하기 위한 추적 알고리즘으로 데이터 군집화(Data Clustering), 데이터 관계 분석(Data Association), 상태 추정(State Estimation), 데이터 정리(Data Arrangement) 총 4 단계로 이루어 진다[그림 1]. 2 장에서는 물체 추적 알고리즘의 각 단계별로 알고리즘의 동작 방식에 대하여 설명한다.

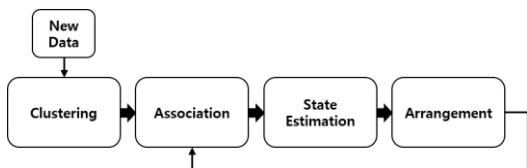


그림 1. 동적 물체 추적 알고리즘

2. 물체 추적 알고리즘

물체 추적 알고리즘은 총 4 단계로 그 중 첫 번째 단계인 데이터 군집화 과정을 제외한 나머지 과정은 재귀(Recursive)적으로 동작한다.

2.1 데이터 군집화(Data Clustering)

데이터 군집화 과정은 LIDAR 센서의 원시 데이터를 각각의 물체 별로 그룹화하는 과정이다. 한 번의 군집화 과정에서 사용되는 Data Point 의 최대 개수는 약 7 만개 정도인데 이 경우 데이터 전체에 대한 계산을 통하여 군집화를 수행하는 기존의 군집화 전략은 실시간 성능을 보장하지 못 하였다. 대신 2D LIDAR 가 회전하면서 동작하는 HDL-32e LIDAR 의 특성을 이용하여 데이터를 시간 순서대로 정렬하고 인접한 시간에 속한 데이터 내에서 군집화를 수행하였다[1]. 데이터를 정렬하면 다음과 같이 $N \times M$ 크기의 Depth 이미지를 만들 수 있다[그림 2]. 예를 들어 (x,y) 위치의 픽셀을 $P(x,y)$ 라고 했을 때, $P(1,1)$ 와 같은 군집에 속하는 데이터를 픽셀을 찾을 때 탐색 영역은 인접한 픽셀인 $P(1,0)$ 과 $P(0,1)$ 뿐이다. 만약 $P(1,0)$ 이 같은 군집에 속하는 픽셀로 판명되었을 경우에는 $P(1,1)$ 과 $P(1,0)$ 을 연결한다. $P(1,1)$ 에 대한 연산이 끝나면 다음 픽셀인 $P(1,2)$ 와 $P(1,1)$, $P(0,2)$ 를 비교하고 같은 군집의 픽셀끼리 연결한다. 이러한 연결과정을 전체 Depth 이미지에 대하여 수행하면 연산시간이 약 10ms 내외의 시간 내에 대략적인 군집을 생성할 수 있다.

(0,0) 20.1m	(0,1) 10.2m	(0,2) NULL	(0,3) NULL	...
(1,0) 10.2m	(1,1) 10.1m	(1,2) NULL	(1,3) NULL	...
(2,0) NULL	(2,1) NULL	(2,2) NULL	(2,3) 6.1m	...
(3,0) 10.2m	(3,1) 10.1m	(3,2) 6.1m	(3,3) 6.2m	...
⋮	⋮	⋮	⋮	⋮

그림 2. LIDAR 데이터의 2D Depth 이미지 표현

여기서 대략적인 군집이라 표현한 이유는 위 알고리즘은 지역 탐색만을 수행하기 때문에 센서의 잡음이나 수풀과 같이 복잡한 형상의 물체에 의해 False Separation을 발생시킨다. 전역 탐색을 통해 생성한 군집의 수와 비교해보면 그 차이가 확연하게 나타난다. 실제로 생성된 군집을 보면 같은 물체이지만 다른 군집으로 할당되어 있는 경우를 쉽게 찾을 수 있다[그림 3]. 실제로 이런 군집화 결과는 잡음처럼 작용하여 다음 단계인 데이터 관계 분석 과정에 심각한 악영향을 미친다.

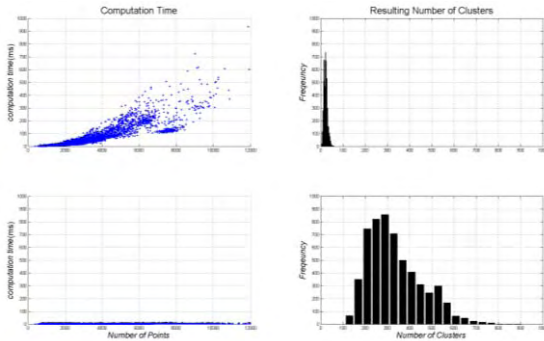


그림 3. 기존의 군집화 알고리즘(위)과 Depth 이미지에서 고속으로 군집화를 수행한 결과(아래)이 부분을 극복하기 위하여 추가적으로 대략적인 군집들을 병합하는 과정을 수행한다. 군집들의 병합과정은 자신과 인접한 다른 군집들을 찾아내어 병합하는 과정이다. 이 과정은 상대 군집의 모든 데이터를 탐색하는 전역탐색이지만 이미 대략적인 군집이 이루어져 있기 때문에 거리가 아주 멀리 떨어져 있는 군집의 단 하나의 데이터를 검사하는 것으로도 해당 군집에 속한 모든 데이터를 탐색 영역에서 제거할 수 있기 때문에 원시 데이터에 대하여 탐색을 수행하는 것에 비해 매우 빠르게 군집화를 수행할 수 있다. 이 한 병합 과정을 통해 우리는 실시간 성능을 만족하면서도 기존의 군집화 과정과 질적으로 유사한 군집화 결과를 얻을 수 있다[그림 4].

2.2 데이터 관계 분석(Data Association)

데이터 관계 분석은 과거의 추적 데이터로부터 예측(prediction)한 현재 추적 물체의 상태 정보를 통해 센서를 통해 관측(observation)된 물체의 상태 정보를 labeling 하는 단계이다. 시간 t에서 생성된 k번째 군집을 C_k^t , 시간 t-1의 i번째 추적 물체 T_i^{t-1} 의 t에서의 상태를 예측한 것을 \hat{T}_i^t 라고 하자. 데이터 관계 분석 과정에서는 군집 집합 $\{C_k^t\}$ 와 추적 리스트 $\{\hat{T}_i^t\}$ 사이의 관계를 분석을 통해 t-1에서의 추적 물체(T_i^{t-1})가 t에서 관측된 군집들 중에 어떤 군집(C_k^t)에 해당하는 지

를 판별하는 과정이다. 관계 분석은 $\{C_k^t\}$ 와 $\{\hat{T}_i^t\}$ 을 특징 공간(Feature Space)에 배치하고 거리가 가까운 군집과 추적 물체를 연결한다. 만약 $C_{i_0}^t$ 와 $\hat{T}_{j_0}^t$ 가 서로 연결되었다면 $C_{i_0}^t$ 는 데이터는 상태 추정(State Estimation) 단계에서 실제 $T_{j_0}^t$ 를 추정할 때 사용된다.

관계 분석 과정에는 2 가지 중요한 요소가 있다. 특징의 선택과 연결 전략이다. 보통 특징의 개수가 많을수록, 즉 특징 공간의 차원이 높을수록 성능이 더 좋

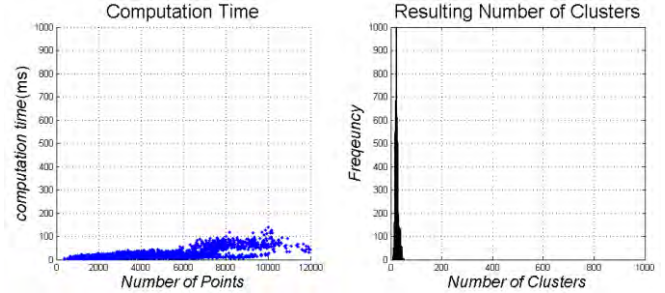


그림 4. Depth 이미지에서 고속으로 군집화를 수행한 후 병합 과정을 거친 결과

을 것이라 생각하기 쉽지만 실제로는 그렇지 않다. 일반적으로 위치 특징이 주로 특징 벡터로 사용되며 높이, 길이, 너비 등이 추가적인 특징으로 사용된다. 하지만 때때로 길이, 너비 등과 같은 특징은 움직이는 물체들에서 심하게 변동하는 특성을 보인다. 그 이유는 LIDAR 로 관측 가능한 추적 물체의 영역이 센서의 측정각도에 영향을 많이 받기 때문이다. 이 측정 각도는 우리 차량의 위치, 자세 그리고 추적 물체의 위치, 자세에 영향을 받는다.

특징 공간에서 군집들과 예측된 추적 물체를 어떤 전략으로 묶을 것인지도 중요하다. 첫 번째 군집부터 차례로 가장 가까운 추적 물체에 할당하는 방법은 가장 간단하지만 가장 지양해야 할 전략이다. 실제로 현재의 군집이 해당 추적 물체에 가장 가까운 군집이라는 것은 데이터 전체를 확인하기 전에는 알 수 없기 때문이다. 그렇기 때문에 모든 군집과 추적 물체의 조합에 대한 거리 계산이 끝난 후에야 할당이 가능하다. 군집 할당은 할당 문제(Assignment Problem)로 해석할 수 있다. 군집의 개수를 N, 추적 물체의 개수를 M 개라고 하자. 할당 결과에 해당하는 변수 a_i 를 정의하자. 만약 $a_i = k$ 라면 군집 C_i 를 추적 물체 T_k 로 할당하게 된다. $A = [a_1, a_2, \dots, a_N]$ 이라 하고, $D(i, a_i)$ 를 C_i 와 T_{a_i} 의 거리라고 할 때, 할당 문제의 목표는 모든 할당 조합의 거리의 합을 최소화하는 A^* 를 찾는 것으로 정의할 수 있다.

$$A^* = \operatorname{argmin}_A \sum_i^N D(i, a_i)$$

본 알고리즘에서는 탐욕 전략을 이용하여 A^* 의 근사치를 찾는 방법을 선택하였다. 본 전략은 가장 작은 거리의 조합부터 차례대로 할당해나가면 A^* 에 근사한 결과를 얻을 수 있을 것이라는 휴리스틱에 기반한다

2.3 상태 추정(State Estimation)

상태 추정 과정은 데이터 관계 분석을 통해 얻은 대

응 관계를 이용하여 측정값(군집)으로 그에 대응하는 추적 물체의 상태를 갱신하는 단계이다. 상태 추정 과정에서 추정하는 추적 물체의 상태 벡터는 다음과 같다.

$$X_t = [x_t \quad y_t \quad \dot{x}_t \quad \dot{y}_t]^T$$

추적 물체의 운동 모델은 선형 등속 운동 모델을 사용하였다. 상태 추정 알고리즘으로는 Kalman Filter 를 이용하였다[4]. 특히 Kalman Filter 의 예측(Prediction) 단계는 데이터 관계 분석(Data Association)단계와 데이터 정리(Data Arrangement)단계에서 추적 알고리즘의 성능을 향상시키는데 활용될 수 있다.

데이터 관계 분석(Data Association) 단계에서 추적 리스트에 있는 물체의 위치를 그대로 사용하는 것보다 Kalman Filter 의 예측 단계로 예측된 위치를 사용하여 더욱 안정적인 결과를 도출할 수 있다.

또한 데이터 정리(Data Arrangement) 단계에서는 센서의 사각지대에 들어가 센서에 나타나지 않는 물체는 예측 단계를 통해 추적을 지속시킬 수 있다.

2.4 데이터 정리(Data Arrangement)

데이터 정리 과정에서는 불필요한 추적 물체의 삭제와 새로운 추적 물체의 등록이 이루어진다. 데이터 정리 과정이 필요한 이유는 불필요한 추적 리소스를 삭제하여 알고리즘의 실시간 성능을 향상시키는 데에 있다.

데이터 관계 분석 과정에서 대응되는 추적 물체를 찾지 못한 군집들은 새롭게 나타난 물체로 간주되어 새로운 추적 물체 목록에 추가된다.

반대로 대응되는 군집을 찾지 못한 추적 물체들은 사라지거나 추적 범위에서 벗어난 것으로 간주되어 추적 물체 목록에서 삭제된다. 이 때 추적 물체의 삭제 시점에 관한 한 가지 유용한 전략을 취할 수 있다. 현 알고리즘에서 추적 물체의 삭제는 추적물체가 사라진 후 약간의 시간 지연(latency) 후에 목록에서 삭제된다. 이 전략의 목적은 센서의 잡음(noise), 가려짐(occlusion), 센서의 사각 지대(blind area) 등에 의해 물체가 잠깐 동안 센서의 시야에서 사라진 후 다시 나타날 경우에도 다시 추적을 재개할 수 있도록 하기 위함이다.

물체가 센서에서 보이지 않는 동안은 Kalman Filter 의 예측 모델을 사용하여 물체의 움직임을 예측하면 나중에 다시 센서에 포착되었을 때 지금까지의 추적 정보를 유지한 채 재 추적이 가능하다.

하지만 때때로 복잡한 환경 또는 잡음이 심한 경우에는 이런 지연전략이 추적 물체 목록을 비대하게 만들어 실시간 성능을 악화시키는 경우가 있다. 이러한 문제는 시간 지연을 추적 물체 별로 차등 분배하여 해결할 수 있다. 중요한 물체에는 더 긴 시간 지연을 부여하고, 그렇지 않은 물체에는 짧게 부여하여 중요하지 않은 정보에 소모되는 리소스를 줄여야 한다. 예를 들면, 관측 횟수에 비례하게 시간을 분배하거나 센서로부터의 거리에 비례하게 시간을 분배하는 전략을 들 수 있다. 이러한 추적 물체 목록 관리 전략을 통해 좀 더 외부 요인에 강인한 물체 추적 시스템의

동작을 구현할 수 있다.

3. 실험 및 결과

본 논문에서는 자율 주행 자동차에 적용 가능한 동적 물체 추적 알고리즘이 소개되었고 실제 자율 주행 차에 구현되었다. 실험은 교통안전공단 주행시험장에서 이루어 졌다.

[그림 5]은 교차로를 지나는 차량을 추적하는 상황이다. 그림에서 확인 할 수 있듯이 추적 알고리즘은 여러 개의 물체를 성공적으로 추적하였고 물체의 궤적, 속도, 위치 등을 추출해 낼 수 있었다.

[그림 6]의 추적 중인 물체 궤적 사이에 있는 막대는 추적 중인 물체가 센서의 시야에서 사라졌음을 의미한다. 당시 고장 난 레이저의 스캔 영역을 물체가 지나면서 센서의 시야에서 사라지는 현상이 발생하였다. 5개의 연속된 막대는 총 5프레임(500ms) 동안 물체가 사라졌음을 의미한다. 이러한 상황에서 물체를 목록에서 바로 삭제하지 않고 예측 과정을 통해 추적을 유지함으로써 물체가 다시 나타났을 때 정보를 잃지 않고도 다시 추적을 재개할 수 있었다.

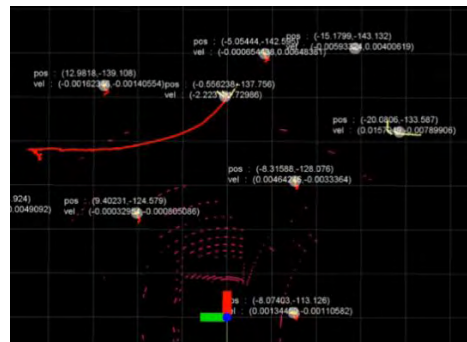


그림 5. 교차로 내 차량의 궤적 및 위치, 속도 추적

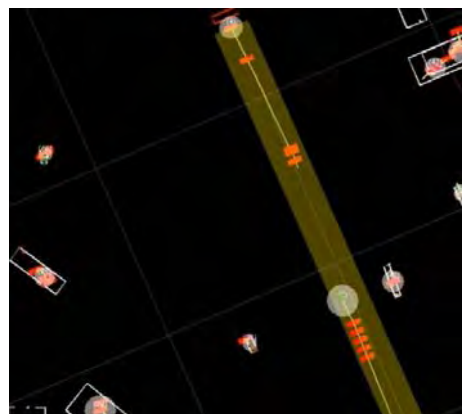


그림 6. 센서의 사각지대에 있는 물체(붉은 색 막대)는 Kalman Filter 의 Prediction 과정을 통해 추적을 지속한다.

사사

본 연구는 산업통상자원부의 재원으로 산업핵심기술 개발사업의 지원을 받아 수행 되었습니다 (10052967, 재난·재해 대응용 특수목적기계 통합제어시스템 개발).

참고문헌

- [1] J. B. Trevor, S. Gedikli, R. B. Rusu, H. I. Christensen, "Efficient Organized Point Cloud Segmentation with Connected Components,"
- [2] F. Bourgeois and J. C. Lassalle, "An Extension of the Munkres Algorithm for the Assignment Problem to Rectangular Matrices." *Communications of the ACM*, vol. 14, no. 12, p. 802, December, 1971.
- [3] S. S. Blackman, "Multiple Target Tracking with Radar Applications," Artech House, 1986.
- [4] S. Thrun, W. Burgard and D. Fox, "Probabilistic Robotics," MIT Press, 2005
- [5] S. S. Blackman, "Multiple Hypothesis Tracking For Multiple Target Tracking," *Aerosp. Electron. Syst. Mag. IEEE*, vol. 19, no. 1, pp. 5–18, 2004.
- [6] T. L. Song, "다중표적 추적필터와 자료연관 기법 동향," vol. 20, pp. 313–322, 2014.