

가우시안 혼합모델 기반 3차원 차량 모델을 이용한 복잡한 도시환경에서의 정확한 주차 차량 검출 방법

Accurate Parked Vehicle Detection using GMM-based 3D Vehicle Model in Complex Urban Environments

조 영 근¹, 노 현 철², 정 명 진[†]

Younggun Cho¹, Hyun Chul Roh², Myung Jin Chung[†]

Abstract Recent developments in robotics and intelligent vehicle area, bring interests of people in an autonomous driving ability and advanced driving assistance system. Especially fully automatic parking ability is one of the key issues of intelligent vehicles, and accurate parked vehicles detection is essential for this issue. In previous researches, many types of sensors are used for detecting vehicles, 2D LiDAR is popular since it offers accurate range information without preprocessing. The L shape feature is most popular 2D feature for vehicle detection, however it has an ambiguity on different objects such as building, bushes and this occurs misdetection problem. Therefore we propose the accurate vehicle detection method by using a 3D complete vehicle model in 3D point clouds acquired from front inclined 2D LiDAR. The proposed method is decomposed into two steps: vehicle candidate extraction, vehicle detection. By combination of L shape feature and point clouds segmentation, we extract the objects which are highly related to vehicles and apply 3D model to detect vehicles accurately. The method guarantees high detection performance and gives plentiful information for autonomous parking. To evaluate the method, we use various parking situation in complex urban scene data. Experimental results shows the qualitative and quantitative performance efficiently.

Keywords: 3D Vehicle Model, 2D LiDAR, Fully Automatic Parking Ability, L shape feature, Point Clouds Segmentation, Vehicle Detection, Complex Urban Environments

1. 서 론

최근 현대자동차에서 주관하는 무인자율주행차량경진대회 등 자율주행차량에 관련된 연구와 상용차량에 장착되는 운전보조시스템 (Advanced Driver Assistant System: ADAS)에 대해서 많은 관심이 증대되고 있다^[1,2]. 이미 다양한 상용차량 회사에서 운전보조시스템을 상용차에 장착하

여 판매를 하고 있고, 매년 열리는 다양한 모터쇼와 같이 자율주행차량 또는 지능형 자동차와 관련된 행사에서도 여러 운전보조시스템이 소개되고 있다. 그 중 자율주행차량, 운전보조시스템 모두에서 요구되는 능력은 바로 완전 자율화가 가능한 주차환경인식 및 주차능력이다. 이러한 능력을 위해서는 주차환경을 정확하게 인식하는 연구가 필요한데, 센서의 종류에 따라 관련 연구를 살펴보면 다음과 같다.

W. J. Park외 3인은 관련 논문 [3]에서 초음파 센서를 이용하여 빈 공간을 검출하는 주차 보조 시스템을 제안하였다. 초음파 센서는 매우 저렴하기 때문에 상용차량에 많이 이용하고 있지만 분해능이 낮고 외란에도 매우 민감하기 때문에 완전 자율화가 가능한 주차 시스템을 구성하기에

Received : Nov. 7, 2014; Reviewed: Dec. 23, 2014; Accepted: Jan. 27, 2015
* This work was supported by the R&D program of the Korean Ministry of Trade, Industry and Energy(MOTIE) and the Korea Evaluation Institute of Industrial Technology (KEIT). (The Development of Low-cost Autonomous Navigation Systems for a Robot Vehicle in Urban Environment, 10035354)

¹ Electrical Engineering, KAIST, (yg.cho@kaist.ac.kr)

² Robotics Program, KAIST, (rohs@rr.kaist.ac.kr)

[†] Corresponding author: Electrical Engineering, KAIST, Guseong-dong, Yuseong-gu, Daejeon, Korea (mjchung@kaist.ac.kr)

는 한계가 있다.

J. K. Suhr의 1인은 [4]에서 AV(M(Around View Monitor))를 이용하여 바닥의 주차 표식을 인식하는 방법을 제안하였다. 이러한 방법은 다양한 주차 종류를 인식할 수 있다는 장점을 가지고 있으나 주차공간의 점유와 주차 공간에 대한 사전정보가 필요하며 주차 표식이 없는 도로 위의 평행주차나 주차 표식이 훼손된 경우에는 알고리즘 적용이 힘들다.

3차원 LiDAR를 이용한 차량 인식 방법은 A. Petrovskaya의 1인에 의해 관련 논문 [5]에 제안되었다. 3차원 LiDAR는 센서 차량이 이동하지 않아도 3차원으로 업데이트 되는 점군을 획득할 수 있는 장점을 가지고 있으나 센서의 가격이 매우 비싸기 때문에 실제 상용 차량에 적용하기에는 힘든 단점을 가지고 있다.

J. Zhou의 2인은 [6]에서 차량 전방에 지면과 수평하게 장착된 2차원 LiDAR를 이용하여 차량 및 빈 공간을 검출하였다. [6]에서는 차량의 범퍼와 측면의 스캔 정보인 L shape feature를 이용하여 주차환경을 인식하였다. 이러한 방법은 주행 방향에 존재하는 빈 공간을 미리 확인할 수 있다는 장점을 가지고 있으나 복잡한 도심환경에서는 feature의 정확도가 떨어져 올바르게 차량을 검출하기 힘들다는 단점을 가지고 있다. 또한 차량을 2차원으로 검출하기 때문에 주차된 차량 및 빈 공간의 높이 정보를 확인할 수 없다. H. C. Roh는 [7]에서 L shape feature를 구성하는 점군을 선으로 연결하여, 이를 사영(Projection)한 것을 이용해 차량을 검출하였다. 하지만 이 논문에서도 L shape feature의 모호성으로 인해 덩불이 차량으로 인식되는 등의 과검출 문제가 나타났다.

위 관련논문들은 주차 표식을 인식하거나 차량을 2차원으로 검출하는 방법을 사용하여 주차환경에 대한 정보를 획득하는 방법을 사용했다. 복잡하지 않은 주차환경에서는 기존의 방법으로도 충분한 성능을 나타낼 수 있지만 신뢰도 높은 주차환경인식을 위해서는 우선 정확하게 주차 차량을 인식하는 것이 전제되어야 한다. 하지만 일반적인 복잡한 도심환경에서는 차를 포함하여 다양한 도시 구조물이 주차환경에 존재하기 때문에 기존의 feature로는 정확하게 주차된 차량을 검출하는데 어려움이 따르게 된다. 또한 관련논문들을 통해 1) 주차 차량의 정확한 3차원 검출, 2) 두

번째는 주행방향의 차량 전방 인식, 3) 가격경쟁력, 이 3가지 중요 요소를 확인할 수 있었다. 따라서 본 논문에서는 이러한 요소들을 만족하도록 차량 전방에 대해 스캔하는 2차원 LiDAR를 통해 획득한 3차원 점군 지도에서 정확하게 주차 차량을 검출하는 방법을 제안한다. 주차 차량은 기존의 L shape feature와 3차원 객체 분할(segmentation) 정보를 융합하여 객체들 중 차량일 가능성이 높은 후보군을 구성하고 차량 후보군에 새로운 3차원 차량 모델을 적용하는 방법을 이용하여 정확하게 검출하도록 하였다.

2. 3차원 차량 검출

제안하는 방법에 대한 전체 흐름도는 Fig. 1과 같다. 전체 흐름도는 크게 데이터 획득, 차량 후보군 추출, 모델 매칭을 통한 복합 주차 차량 검출로 나눌 수 있다.

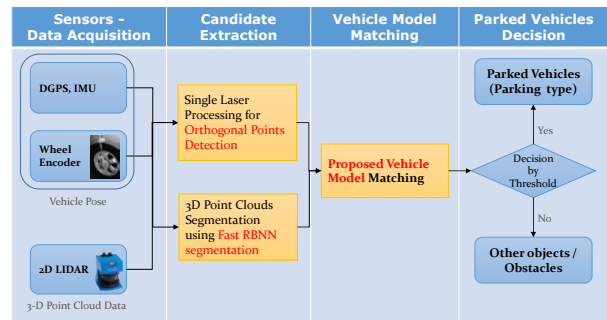


Fig. 1. Overall flow chart of the proposed method

2.1 데이터 획득

본 논문에서 사용되는 데이터는 Fig. 2와 같이 2차원 LiDAR가 지면을 대각선 아래로 바라보게 장착되어 있는 센서 시스템을 이용하여 획득한다^[7]. 센서 시스템이 이동하면서 지면을 대각으로 스캔한 정보가 연속적으로 축적되며, 차량의 위치 정보를 이용하여 갱신하게 되면 3차원 점군을 쉽게 획득할 수 있다. 또한 위치 센서들을 통해 차량의 자세정보(Pose)가 보정하기 때문에 요철구간에 대한 데이터 손실을 방지하며 경사 구간에서도 IMU(Inertial Measurement Unit)로 Pitch 정보를 반영한다. 따라서 주차된 차량과 센서 시스템이 동일한 도로 상에만 존재한다면 경사 정보가 반영된 3차원 점군을 획득할 수 있으며 경사에서의 차량 검출이 가능하다.

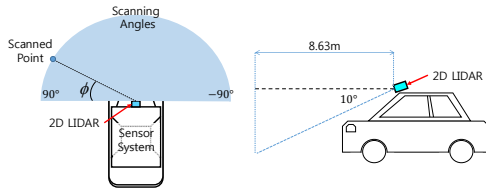


Fig. 2. Sensor installation – front inclined 2D LIDAR

Fig. 3은 2차원 LiDAR가 설치된 센서 시스템을 이용하여 연속적인 거리 정보를 획득하는 과정이다. 여기서 i 번째로 스캔된 정보를 P_i 라고 하고, 해당 스캔 정보 중 k 번째로 측정된 점을 $P_{i,k}$ 라고 한다. 즉, i 번째 스캔에 포함된 총 스캔 정보의 개수가 n 개라고 한다면 스캔 정보 P_i 는 $P_i = \{p_{i,1}, \dots, p_{i,n}\}$ 로 표현할 수 있다.

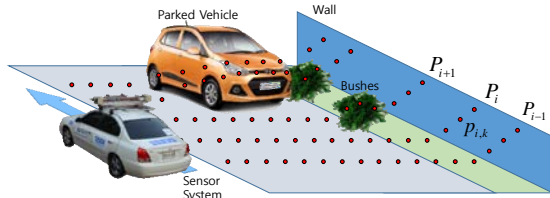


Fig. 3. Data acquisition from the sensor system

2.2 차량 후보군 추출

이번 절에서는 2.1절에서 획득한 3차원 점군 중에서 차량일 가능성이 높은 점군 후보군을 추출하는 과정이다. 관련 논문 [6], [7]에서 사용한 L shape feature는 차량 이외에도 덤불, 건물, 배전함 등 모서리가 직각에 가까운 많은 물체에서도 나타나므로 차량이 아닌 물체를 차량으로 인식하는 과 검출 문제를 야기한다. 하지만 L shape feature은 매우 간단하고 속도가 빠르다는 장점을 가지고 있다. 그러므로 본 논문에서는 L shape feature를 전과정(pre-stage)으로 이용하여 모든 점군이 아닌 차량일 가능성이 높은 우선 물체를 추출하여 3차원 차량 모델과의 매칭을 효율적으로 수행할 수 있도록 한다.

2.2.1 한 줄 스캔 처리 과정

이 과정에서는 획득되는 각각의 스캔 데이터를 가장 먼저 처리하는 부분으로 우선 지면일 가능성이 높은 점군과 오류로 분류될 수 있는 점군들을 먼저 추출한다. 그리고 거리 값의 차이에 따라 한 스캔 내에서 점군들을 분할하

고 각 분할된 점군 중 직각성분(orthogonal)을 가지고 있는 것을 gradient histogram 검출한다.

한 스캔 라인을 유효한 단일 스캔 객체로 분할 하기 위해서는 우선 지면을 제거해야 한다. 지면을 제거하는 방법은 한 스캔라인에 속한 연속적인 점들 중 수평형 특성을 가진 점들을 추출하고, 수평형 특성 점들 중 차량 중심 좌표계로 변환 시켰을 때의 높이 값이 일정 이하인 점들을 지면으로 분류한다. 한 스캔라인에는 스캔 점들이 1부터 n 까지 순차적으로 들어오게 되는데 이러한 점들의 순서를 알기 때문에 점들간의 각도를 구할 수 있다. Fig. 4와 같이 $P_{i,k}$ 의 각도정보는 높이 값의 변화량만 고려하기 위해 $P_{i,k-2}$ 부터 $P_{i,k+2}$ 까지 총 5개의 점들의 y-z plane projection한 것의 사이 각도(α)를 평균한 값을 부여하며 이 각도의 절대값이 $0 \sim 20^\circ$ 사이인 점들을 수평형 점이라 분류한다. 수평형 점들 중 차량중심좌표 기준 높이(z) 값이 $-0.3m$ 인 경우(실측 지면 기준 차량 중심 높이: $0.4m$)를 지면으로 최종 분류한다.

그리고 지면이 제거된 한 스캔라인에 속한 점들은 순차적으로 거리 값의 차이가 $0.2m$ 이하인 점들은 하나의 객체로 묶어주게 된다. 즉, Fig. 5와 같이 한 스캔라인 P_i 에 $S_{i,1}$ 부터 $S_{i,m}$ 까지 m 개의 단일 스캔 객체(Single Laser Segment)가 존재하게 된다. 그리고 이중 단일 스캔 객체에 포함된 점들의 개수가 5개 이하인 객체는 무효 객체로 구분한다. 추출된 단일 스캔 객체들은 Fig. 5와 같이 gradient histogram을 이용하여 직각 객체인지 판단한다⁷⁾.

Fig. 5는 단일 스캔 객체 $S_{i,k}$ 에 대한 gradient histogram을 보여준다. gradient histogram에 사용하는 bin은 지면 제거에서 사용했던 각 점의 각도 정보를 이용하는데 각 bin의 크기를 10° 로 하여 총 36개의 bin으로 histogram을 구성한다. 단일 스캔 객체의 직각 여부는 histogram에서 가장 많은 투

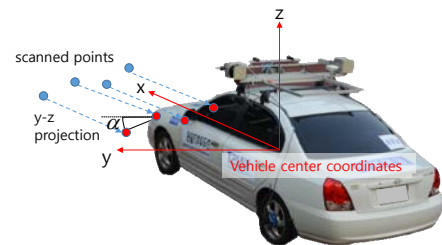


Fig. 4. Horizontal points segmentation by the angle of consecutive points (Blue dots: raw scanned points, Red dots: projected points)

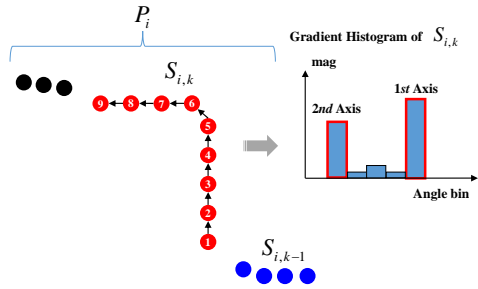


Fig. 5. Gradient histogram of a single laser segment

표(vote)를 받은 두 main axis의 차이가 80° 에서 100° 사이 에 존재하는 경우를 직각 성분이라 판별하게 된다.

2.1.2 3차원 객체 분할

2.1.1항에서 2차원 직각성분을 검출하였다면 [8], [9]와 같 이 3차원 객체 분할(Segmentation) 과정을 통하여 2차원 직 각성분 정보를 3차원 객체로 융합해야 한다. 객체 분할 과 정을 통해 분할된 각각의 객체는 그 객체의 직각성분 존 재 여부를 이용하여 차량 후보군으로 구성하게 된다.

본 논문에서는 [9]에서 제안한 RBNN(Radially Bounded Nearest Neighbor graph)의 방법을 이용하여 객체를 분할하 였다. RBNN은 자신 주위의 특정 반경 이내에 존재하는 모 든 점들을 하나의 객체로 지정하는 방법이다. RBNN은 여 러 객체화 방법 중에 연속적으로 들어오는 데이터를 처리 하기 유용하며 그 처리 과정이 간단하다. 물론 RBNN에는 지면이 제거되지 않는다면 각 물체가 효율적으로 분할되 지 않는다는 단점이 있지만 제안하는 방법에서는 2.1.1 항 에서 이미 지면 추출 과정을 수행하였다. 또한 주차 공간 에서 일반적으로 차량의 주행 속도는 30km 미만이기 때문 에 최대 주행 속도를 30km로 정했으며 센서 시스템의 주 행속도와 2차원 LiDAR의 스캔 속도(37.5Hz) 스캔과 스캔 사 이의 거리는 약 20cm로 예상할 수 있다. 그러므로 본 논문

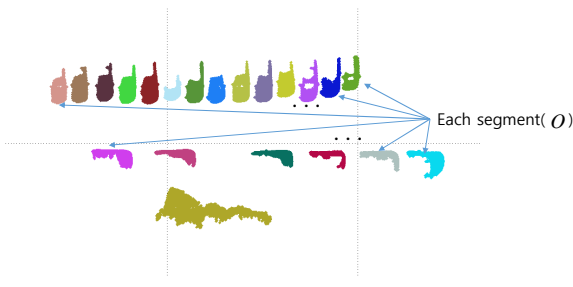


Fig. 6. Segmentation result

에서는 RBNN의 객체화 반경을 0.2m로 정하여 segmentation 을 수행했다.

RBNN 객체분할 과정을 거치게 되면 점군은 Fig. 6와 같 이 각각의 객체로 묶이게 되며 한 객체는 O_j 로 표현된다. 그리고 한 객체는 $O_j = \{P_{ab}, \dots, P_{mn}\}$ 로 3차원 점군들의 집합 으로 이뤄진다.

2.1.3 차량 후보군 추출

2.1.2항에서 분할된 각각의 객체는 2.1.1항에서 추출한 직각성분과 융합하여 차량 후보군으로 구성할 수 있는지 판단되게 된다. 이러한 판단은 각각의 분할된 객체는 객체 내에 2개 이상의 유효한 직각성분의 포함 여부를 따져 결 정하게 된다. 유효한 직각성분이라 함은 객체 내에 포함된 직각성분이 본래의 길이의 90% 이상을 유지하고 있는 경 우를 의미한다.

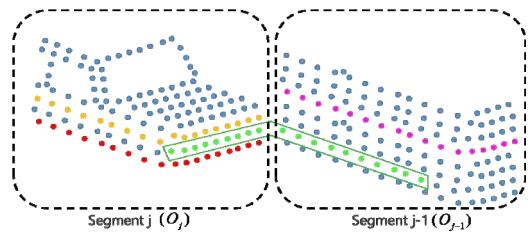


Fig. 7. Candidate examples of segments from RBNN

Fig. 7을 보면 초록색 직각성분은 두 객체에 분리되었기 때문에 유효한 직각성분으로 판단 할 수 없다. 따라서 객 체 O_j 에는 2개, O_{j-1} 에는 1개의 유효한 직각성분을 포함하고 있으며 이중 객체 O_j 만 차량 후보군으로 구성할 수 있다. 차량 후보 C_j 는 식 (1)와 같이 표현될 수 있는데 O_j 는 차량 후보 C_j 를 구성하는 객체를 의미하며 $(x_j, y_j, z_j, \theta_j, \sigma_j)$ 는 후보 C_j 의 Pose 정보를 의미한다. 또한 Pose 정보의 x, y, z는 midrange point로 max point와 min point의 평균값을 의미하며 σ_j 는 차량 후보가 구성되는 순간의 센서 시스템의 Pitch 정 보 (Fig. 9의 y축 기준 방향각)을 이용한다.

$$C_j = \{O_j, x_j, y_j, z_j, \theta_j, \sigma_j\} \tag{1}$$

$$\text{where } \begin{cases} O_j = \{P_{a,b}, \dots, P_{m,n}\} \\ (x_j, y_j, z_j) = \text{midrange}(x, y, z) \\ \theta_j = \arg \max(\text{Length}(S_{i,k}(\theta))) \\ \sigma_j = \text{pitch}(\text{Sensor system}) \end{cases}$$

2.2 3차원 점군에서의 차량 검출

2.2.1 3차원 차량 모델

차량 검출에 사용한 특성 모델은 관련 논문 [10]에서 제안했던 가우시안 혼합 모델(GMM – Gaussian Mixture Model) 기반의 차량 모델이다. 차량모델은 Fig. 9와 같이 총 9개의 가우시안으로 이루어져 있으며 Pose 정보를 이용하여 매칭을 할 수 있다. 차량 모델은 Fig. 8의 a와 같이 부분적으로 스캔된 차량을 대칭적인 외형 특성을 이용하여 혼합한 모델이다. 제안하는 모델은 이러한 대칭 특성과 각 면이 이루는 확신값(Confidence)를 이용하여 Data augmentation 없이 Data accumulation으로 완성형 모델을 구성할 수 있도록 하였다.

우선 Fig. 8의 a와 같이 각각 구분된 옆면과 앞/뒷면의 축적된 데이터에서 Fig. 8의 b와 같이 가우시안 혼합모델을 따로 추출하게 되고 차량의 대칭적 특성에 따라 확신값(Confidence)를 부여된다. 그리고 이 확신값에 따라 선택적으로 가우시안 모델을 추출하여 단순한 변수 혼합과정을 통해 차량의 3차원 완성 모델을 구성할 수 있다. 이렇게 구성한 모델은 차량의 외형적 특성을 잘 반영하고 있기 때문에 차량 검출을 위한 특성으로 사용하기 용이하다. 또한 3차원 차량 모델은 모델의 중심점의 위치와 방향각 (x, y, z, θ, σ)을 이용하여 쉽게 필요한 위치에 적용할 수 있다. 그렇다면 다음 항에서 제안한 3차원 차량 모델을 이용하여 차량을 검출하는 방법에 대하여 살펴보도록 한다.

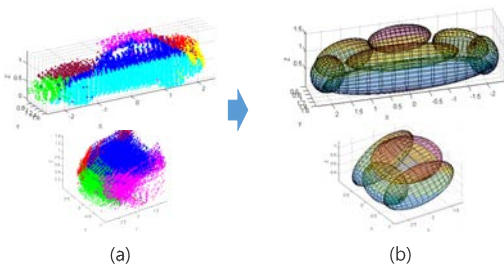


Fig. 8. 3D vehicle model construction. (a) Partial training data (b) Estimated Gaussian models of training data

2.2.2 다양한 주차된 차량 검출

차량의 주차는 일반적으로 평행주차(Parallel Parking), 수직주차(Perpendicular Parking) 그리고 대각 주차(Angled Parking)의 3가지로 분류될 수 있다. 이번 절에서는 차량의

주차 종류에 상관없이 제안한 방법으로 복합적인 주차 환경에서 어떻게 차량을 검출 할 수 있는지 확인한다.

차량은 기본적으로 차량 후보군과 3차원 차량 모델의 매칭(Matching) 과정을 통해 검출한다. 2.2.1항에서 3차원 차량 모델은 중심점 (x, y, z)와 Fig. 9의 z축 기준 방향각 (Yaw: θ)과 y축 기준 방향각(Pitch: σ)으로 구성되어 있으므로, 적용할 모델의 위치 정보는 각 차량 후보의 위치를 이용한다. 후보군에 속한 각각의 차량 후보는 식 (2)와 같이 하나의 위치정보를 가지고 이 위치 정보를 기반으로 3차원 차량 모델과의 매칭을 실시 한다.

$$Pose(C_j) = (x_j, y_j, z_j, \theta_j, \sigma_j) \quad (2)$$

차량 후보군에 속한 객체들이 차량인지 판단하는 조건은 각 객체들과 3차원 차량과의 매칭 점수로 이뤄지게 된다. 본 논문에서는 두 가지 정보를 정규화하여 매칭 점수로 활용하는데 하나는 식 (3)의 a항과 같이 객체를 이루고 있는 점군이 3차원 차량 모델에 포함되는지에 대한 정보이며 다른 하나는 b항과 같이 점군의 법선 벡터(Vector)와 3차원 차량 모델의 법선 벡터 내적을 통한 법선 벡터 유사도를 이용한다. 그리고 두 항 모두 각 객체와 차량 모델에 포함된 점군의 개수로 정규화 하여 0에서 1의 값을 가지도록 설정하였다.

$$score(C_j) = \frac{1}{2} \left(\frac{num(pcds \text{ in GMM})}{num(pcds \text{ in a segment})} \right) + \frac{1}{2} \left(\frac{sum(\max(\sum_{i=1}^9 num(pcds \text{ in GMM}) < \mathbf{n}_i, \mathbf{n}_j >))}{num(pcds \text{ in GMM})} \right) \quad (3)$$

where $\left\{ \begin{array}{l} pcds = \text{point clouds} \\ \mathbf{n}_i = \text{normal vector of each Gaussian model} \\ \mathbf{n}_j = \text{normal vector of each point} \end{array} \right.$

제안하는 점수(Score)에서는 이러한 두 가지 유사도를 통해 점군의 형태뿐만 아니라 점군이 이루는 면의 방향까지 고려할 수 있다. 이미 제안하는 모델에서 차량 검출에

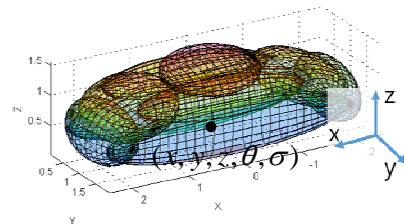


Fig. 9. The complete Gaussian Mixture Model (GMM) based vehicle model which consists of 9 Gaussian models.

필요한 충분한 정보를 지니고 있기 때문에 매칭 후 적합한 Threshold 값을 확인하면 2.1.3항에서 추출한 차량 후보군에서 주차된 차량을 검출할 수 있다.

그리고 일정 Threshold 값 이상의 후보를 차량으로 판단하면 최종적으로 검출한 차량이 가지고 있던 방향각(Yaw: θ)를 이용해 주차 방법을 분류한다. θ 에서 센서 시스템의 주행 경로가 이루는 각도를 보정한 후 주차 차량의 각도의 절대값이 0~30° 인 경우를 평행, 30~60° 를 대각, 60~90° 인 경우를 수직 주차로 분류한다. 마지막으로 일정 Threshold를 넘지 못한 객체들과 차량 후보로 구성되지 못한 객체들은 장애물로 구분하도록 한다.

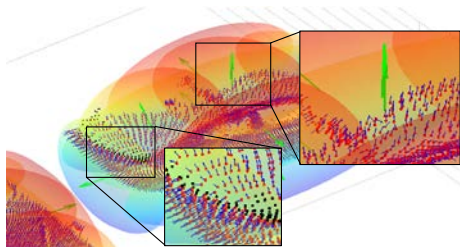


Fig. 10. Scoring method (Red dots: points in the vehicle model, Black dots: points out of the model, Blue arrows: normal vector of red dots, Green arrows: normal vectors of each Gaussian model in the vehicle model)

3. 실험 결과 및 분석

3.1 센서 시스템 및 실험 데이터

Fig. 11는 데이터 획득에 사용된 센서 시스템을 나타내는 그림이다. 거리 센서에는 SICK LMS 200 LiDAR를 사용했으며 위치 센서로는 Xsens MTi IMU, HUACE DGPS와 Autonics Wheel Encoder를 사용했다. 사용한 PC의 성능은 Intel(R) Core i5 (2.8GHz), Ram 4G이며 MATLAB 기반으로 프로그램을 구현하였다.

Fig. 12는 실험 데이터로 활용한 카이스트 캠퍼스 실험 환경으로 A구역, B구역 두 가지 환경에서 획득 했다. Fig. 12의 노란색 상자 (1, 2, 3) 에는 실제 센서 시스템에서 촬영한 실험 데이터를 나타내며 주차장의 주행공간의 폭이 매우 좁으며 아스팔트와 보도블럭 노면으로 이루어져 있다. 또한 다양한 주차 방법과 도시구조물이 위치하고 있기 때문에 제안하는 방법의 검증에 적합하였다.

카이스트 A구역, B구역에서 획득한 정보에는 총 134대

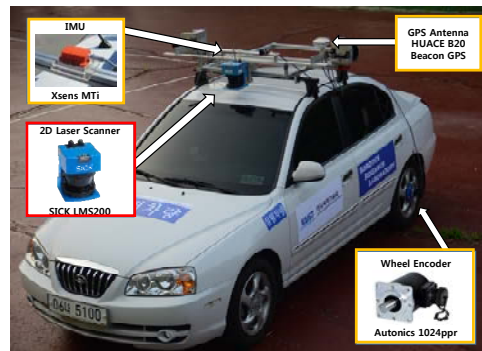


Fig. 11. Experimental sensor system

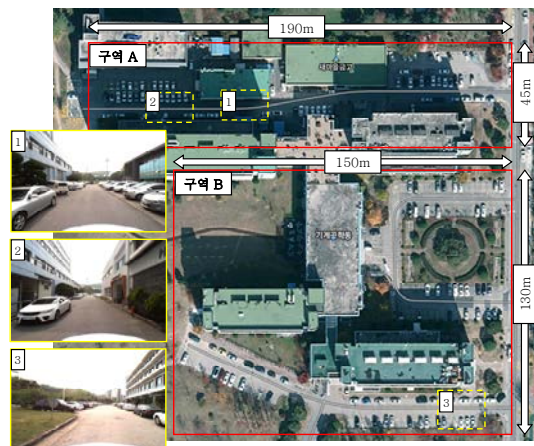


Fig. 12. Experimental environments

의 주차된 차량이 존재 했으며 모두 Ground Truth Labeling 프로그램을 통해 수기로 Ground Truth를 제작하였다. 카이스트 A구역에는 수직주차 36대, 수평주차 22대, 대각주차 4대로 총 62대의 주차된 차량이 포함되어 있으며 카이스트 B구역에는 수직주차 49대, 수평주차 23대로 총 72대의 주차된 차량이 포함되어 있다.

3.2 실험 결과

3.2.1 복합 주차 방법 검출 결과

Fig. 13을 확인하면 제안하는 방법에 대한 복합 주차 방법 검출 결과를 확인할 수 있다 Fig. 13은 Threshold를 0.5로 했을 경우에 대한 결과이다. 파란색 선은 차량의 주행 경로를 의미하며 빨간색은 수직주차, 초록색은 수평주차, 주황색은 대각주차 차량을 의미한다. 그리고 차량으로 검출되지 않은 객체들은 점군의 높이 값에 따라 0m 인 점군은 검은색으로 1.75m인 점군은 갈색의 그라데이션으로 표

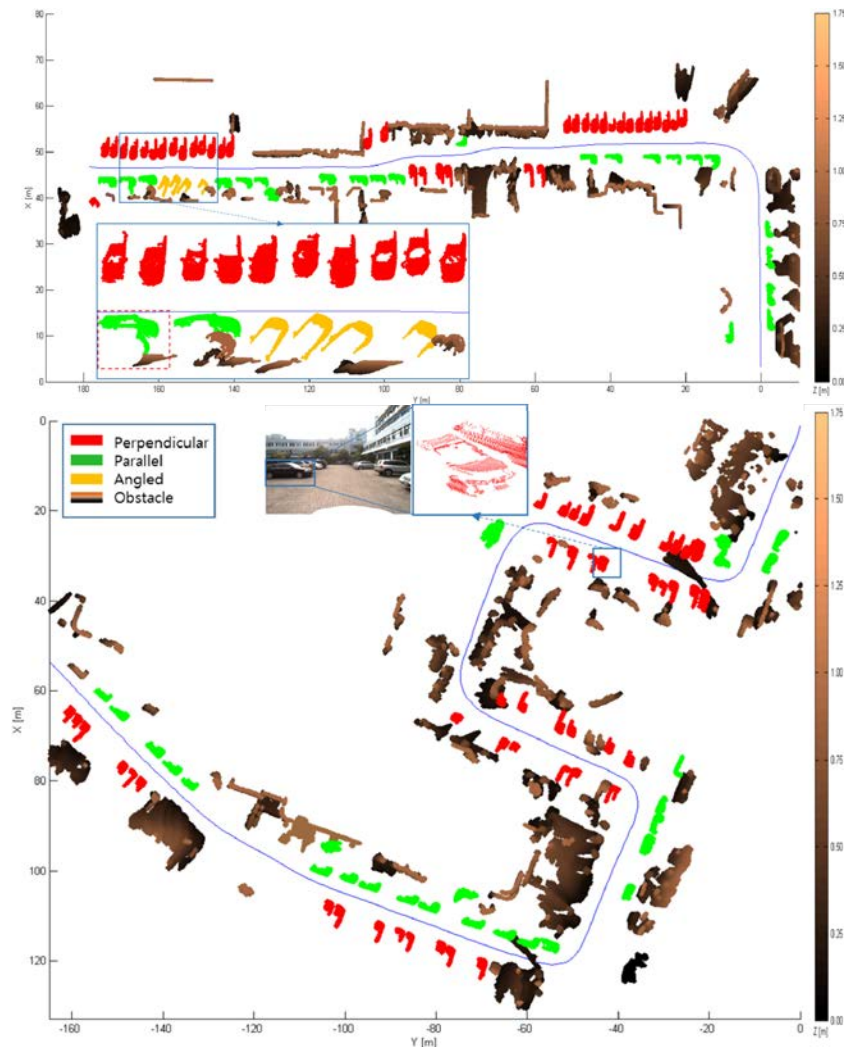


Fig. 13. Experimental result section A(Top) and B(Bottom) (Red: perpendicular parked vehicle, Green: parallel parked vehicles, Orange: angled parked vehicles, Black: obstacles, Blue line: sensor system trajectory)

현하였다.

Ground Truth는 Ground Truth Labeling Program을 이용하여 수기로 제작하였다. 검출 결과의 비교 방법은 논문 [11]에서 제안한 방법으로 검출한 차량과 Ground Truth의 차량이 일치율로 결과를 판단하였다. 비교방법은 검출 결과와 Ground Truth의 일치율이 높을수록 1에 가까워지는데 본 논문에서는 0.9 이상 매칭이 되는 결과를 올바르게 검출된 차량이라고 정하였다.

3.2.2 차량 검출 성능 분석 결과

제안하는 방법의 검출 성능은 Precision과 Recall, 그리고 F1 measure를 이용하여 분석하였다. Precision은 검출 결과에

서 올바르게 검출된 차량의 비율이며, Recall은 Ground Truth 중에서 검출결과에 포함된 차량의 비율을 의미한다. 제안하는 방법의 검출 성능은 Precision과 Recall, 그리고 F1 measure를 이용하여 분석하였다. Precision은 검출 결과에서 올바르게 검출된 차량의 비율이며, Recall은 Ground Truth 중에서 검출결과에 포함된 차량의 비율을 의미하기 때문에 제안하는 방법의 성능을 판단하기에 좋은 지표이다. 또한 F1 measure는 Precision과 Recall을 동등한 비중으로 보고 성능을 한 번에 판단하는 지표이므로 F1 measure를 통해서 종합적인 성능을 확인 할 수 있다.

본 논문에서 차량 검출에 사용하는 차량 모델은 차량에 대한 정보를 충분히 가지고 있기 때문에 검출 방법에는

Threshold 값을 조절하여 제안하는 차량 모델과 점수에 대한 검출 결과를 살펴보았다. 우선 KAIST 구역 A의 데이터에 대하여 확인한 결과는 Fig. 14와 같다. F1 measure를 통해 알고리즘의 성능을 판단하면 Threshold가 0.4일 때 0.954로 약 95%의 검출 성능을 나타내는 것을 확인할 수 있다. 또한 사용자의 목적에 따라 Threshold를 조절하여 알고리즘을 적용할 수 있는데 주차된 차량 중 가장 정확하게 스캔된 차량만을 검출하기 위한 목적으로는 Precision이 높아야 하기 때문에 높은 Threshold를 선택할 수 있으며, 우선 모든 주차된 차량을 검출하는 것이 목적이라면 낮은 Threshold를 선택하면 그 목적에 부합하는 결과를 획득할 수 있다. 구역 A에서의 오검출의 경우는 Fig. 13의 첫 번째 결과의 붉은색 점선 박스와 같다. 이 경우 주차된 차량이 주변 덩불과 붙어있어 덩불까지 차량으로 인식한 경우이다. 이러한 부분은 3차원 객체화 과정에서 덩불과 차량이 하나의 객체라고 인식하였기 때문에 발생한 경우이다.

구역 B에 대해서도 검출결과를 확인하면 Fig. 15와 같다. 구역 B에서는 Threshold가 0.5 일 때 가장 높은 F1 measure를 가지는데 이 때의 값은 0.958로 약 96%의 검출 결과를 나타냈다. 구역 A에서 가장 높은 값을 가졌던 Threshold 0.4에서도 F1 measure는 0.944로 약 94%의 검출 결과가 나오는 것을 확인할 수 있었다. 구역 B에서는 구역 A에서와는 다르게 Threshold가 높아질수록 Recall 값이 확연히 달라지는 것을 알 수 있는데 이는 주차된 차량의 특성에 따라 나타난 결과이다. Fig. 13의 두 번째 결과의 파란색 박스와 같이 LiDAR의 경우 검은색 차량에 대해서는 제대로 레이저가 반사되지 않아 스캔되는 점들의 개수가 다른 색의 차

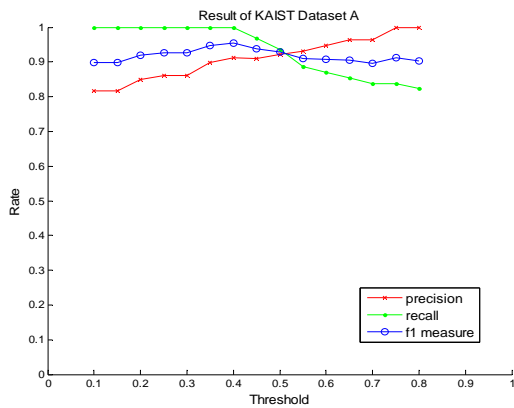


Fig. 14. Precision-recall-f1 measure curve of section A

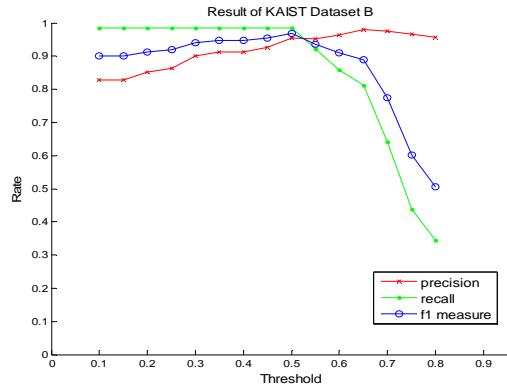


Fig. 15. Precision-recall-f1 measure curve of section B

량보다 적고 산발적으로 분포되게 된다. 그렇기 때문에 법선 벡터를 추정함에 있어서 그 방향이 어긋나는 경우가 발생하게 되고 결과적으로 다른 색의 차량만큼 1에 가까운 굉장히 높은 점수를 가지지 못하는 경우가 발생한다. 하지만 모델이 외형정보를 충분히 지니고 있기 때문에 Threshold가 0.4~0.6으로 설정할 경우 검은색 차량을 포함하여 차량의 구분 없이 검출할 수 있는 결과를 보여주었다.

본 논문에서 제안했던 방법을 이용한 결과와 기존의 방법들의 결과를 비교해보면 Table 1과 같다. 기존의 방법 중 3D LiDAR를 이용한 방법 [5]에는 3개 이상의 Frame이 필요하므로 3개의 Frame일 때의 검출율을 비교하였고, 2D LiDAR의 방법 중 [6]는 범퍼(차량) 검출 결과를 비교하였다. 그리고 마지막으로 [7]의 방법은 구현을 통해 검출 결과를 확인하였다. 비교결과 기존의 방법들에 비해서 차량을 3차원으로 검출하며 검출 결과에서도 향상된 모습을 나타내는 것을 확인할 수 있었다.

그리고 제안하는 알고리즘은 MATLAB을 이용하여 알고리즘이 타당성을 검증하였다. 구역 A에는 3130개의 스캔라인의 총 1129930개의 점군을 처리하는데 연산시간이 287초 이므로 한 스캔라인을 처리하는데 0.09초가 소요되었다. 구역 B에 대해서는 4408개의 스캔라인의 총 1591288개의 점군을 처리하는데 필요한 연산시간은 324초 이므로 한 스캔라인당 0.073초가 소요되었다.

Table 1. Comparison with previous methods

3D LiDAR	2D LiDAR		
[5]	[6]	[7]	Proposed
89.7 %	87.5 %	90.8 %	95.6 %

4. 결론

본 논문에서는 지면을 대각선으로 내려보는 2차원 LiDAR를 통해 획득한 점군 맵에서 2차원 및 3차원 차량 모델을 이용하여 주차된 차량을 정확하게 검출하는 방법을 제안하였다. 이는 지능형 자동차의 완전 자율화가 가능한 주차능력을 위한 기반 기술로 주차된 차량의 종류나 방법에 상관없이 주차된 차량을 검출하고, 추가적으로 어떤 방법으로 차량이 주차되어 있는지에 대한 정보를 제공한다. 또한 주변 환경이 매우 복잡한 도심이나 주차 표시가 없는 도로 위 평행주차의 경우에도 차량을 정확히 검출할 수 있기 때문에 장소에 국한되지 않고 제안하는 방법을 적용할 수 있다. 그리고 본 논문에서는 MATLAB을 사용하여 알고리즘을 검증하였는데 알고리즘을 C언어로 구현한다면 충분히 실시간성을 가질 수 있을 것이라 예상할 수 있다.

또한 더 높은 인식률을 위해서 특징치를 보완하거나 주차 환경에 대한 추가적인 정보를 제공함으로써 좀 더 효율적으로 지능형 자동차에 주차환경에 대한 정보를 전달할 수 있도록 연구 필요할 것으로 보인다.

References

[1] 2013 Korea autonomous vehicle contest website, <http://autonomous.ksae.org/>

[2] Noh Chi-Beom, Kim Min-Ho, Lee Min-Cheol, "Path Planning for the Shortest Driving Time Considering UGV Driving Characteristic and Driving Time and Its Driving Algorithm." Journal of Korea Robotics Society, Vol. 8, No. 1, pp. 43-50, 2013.

[3] W. J. Park, B. S. Kim, D. E. Seo, D. S. Kim and K. H. Lee, "Parking Space Detection Using Ultrasonic Sensor in Parking Assistance System," in IEEE Intell. Veh. Symp., pp. 1039-1044, 2008.

[4] J. K. Suhr and H. G. Jung, "Full-automatic recognition of various parking slot markings using a hierarchical tree structure", in Optical Engineering, vol. 53, num. 3, pp. 037203-1-037203-14, Mar. 2013.

[5] A. Petrovskaya and S. Thrun, "Model based vehicle detection and tracking for autonomous urban driving," in Auton. Robot., vol 26, pp. 123-137, Arp. 2009.

[6] J. Zhou, L. E. Navarro-Serment and M. Hebert, "Detection of Parking Spots Using 2D Range Data," in IEEE 15th Int. Conf. Intell. Transp. Syst., 2012, pp. 1280-1287.

[7] H. C. Roh, C. H. Sung and M. J. Chung, "Fast Vehicle Detection Using Orientation Histogram and Segmented

Line Projection", in Int. Conf. Ubiquitous Robots and Ambient Intelligence(URAI), Nov. 2012.

[8] Bo Gil Seo, Yungeun Choe, Hyun Chul Roh and Myung Jin Chung, "A Clustering Method for Efficient Segmentation of 3D Laser Data", Journal of Korea Robotics Society, Vol. 9, No. 2, pp. 1-10, 2014.

[9] S. U. Ahn, Y. G. Choe, M. J. Chung, "Fast Scene Understanding in Urban Environments for an Autonomous Vehicle equipped with 2D Laser Scanners", The Journal of Korea Robotics Society, Vol. 7, No. 2, pp. 92 - 100, 2012.

[10] Y. G. Cho and M. J. Chung, "A 3D Feature Model for Urban Road Vehicle Detection", Int. Symp. Distributed Autonomous Robotic Systems (DARS), 2014.

[11] B. Douillard, J. Underwood, N. Kuntz, V. Vlaskine, A. Quadros, P. Morton, and A. Frenkel, "On the segmentation of 3D LIDAR point clouds", IEEE Int. Conf. Robotics and Automation (ICRA), pp. 2798-2805, 2011, May.



조영근

2013 인하대학교 전기공학과
(공학사)
2013~현재 한국과학기술원 전기
및 전자공학과(공학석사)
관심분야 : 3D world model, SLAM



노현철

2008 서울시립대학교 전자전기
컴퓨터공학부 (공학사)
2010 한국과학기술원 로봇공학
학제전공 (공학석사)

2011~현재 한국과학기술원 로봇공학학제전공 박사과정
관심분야 : 3D world model, localization



정명진

1973 서울대학교 공과대학전기
공학과 (공학사)
1977 미시간대학교 전기공학과
(공학석사)
1983 미시간대학교 제어공학과
(공학박사)

1976 국방과학연구소 연구원
1981~1983 미시간대학교 CRIM 연구 조교
1983~현재 한국과학기술원 전기 및 전자공학과 교수
관심분야 : service robot, human-robot interaction