

차선 영역이 분할된 AVM 영상을 이용한 차선 수준 차량 위치 추정 기법

도종용[†]·강정원*·정석우**·정지원**·심현철*

Lane-Level Vehicle Localization Using Segmented AVM Images

Jongyong Do[†], Jungwon Kang*, Seokwoo Jung**, Jiwon Jung**
and David Hyunchul Shim*

Key Words: Vehicle Localization(차량 위치 추정), Lane region-segmented AVM images(차선 영역이 분할된 AVM 영상), Iterative Closest Point(ICP) algorithm

Abstract

In this paper, we present an algorithm for lane-level vehicle localization using lane region-segmented AVM images. First, we segment an input AVM image into lane regions. Then, we estimate a vehicle pose through ICP-based matching between the lane regions and the digital map. Contrary to previous methods that used restricted lane models, the proposed method facilitates vehicle localization even in arbitrary shaped road sections, as we utilize pixel-wise lane region itself without the use of the lane model. Through experiments in real roads, we verified the effectiveness of the proposed method.

1. 서론

주행 영역에 대한 지도를 기반으로 자율주행을 하면 센서를 이용한 주변 환경 인식이 부정확한 경우에도 지도 정보를 활용하여 주행이 가능하고, 영역 내에서는 어떤 경로든 임의로 정할 수 있다. [2]의 EureCar는 실시간으로 차선을 검출하여 경로를 생성하지만 주차 공간과 같이 검출한 차선을 주행을 위한 제어에 직접 활용하기 힘든 경우에는 경로계획이 어려우므로, 지도를 이용한 차량 위치 추정이 유용하게 쓰일 수 있다.

차량 측위와 관련된 방법은 크게 Kalman 필터를 이용한 방식[1], [3], [7]과 Particle 필터를 이용한 방식(Monte-Carlo Localization)[4], [6]으로 나눌 수 있다.

지도 기반의 자율주행에서의 화두는 실제 차량이 위치해 있는 곳과 지도에 나타나는 차량의 위치를 정확하게 일치시켜 차량이 지도 정보를 따라 주행할 때 실제 도로에서도 지도에서 예측한 상황대로 주행할 수 있도록 하는 것이다.[5] 본 연구에서는 저가형 센서인 AVM 카메라를 이용한 차선 수준 차량 위치 추정기법에 대해서 다룬다. 2장에서는 관심 영역에 대한 차선 지도를 작성하는 방법에 대해 언급하고, 3장에서 작성한 지도와 AVM 영상을 정합하여 차량의 위치를 추정하는 기법을 다룬다. 그리고 4장에서는 실시간으로 저장한 AVM 영상에 개발한 알고리즘을 적용한 결과를 정리하였다.

2. 차선 지도 작성

2.1 AVM 카메라 시스템

위치 추정을 위한 센서로 AVM 카메라를 사용하여 차량의 모든 방향에서 평면 도로 상의 차선, 노면표시 정보를 획득하였다. (Figure 1.)

[†]한국과학기술원 기계항공공학부

E-mail : djy0129@kaist.ac.kr

TEL : (042)350-3764 FAX : (042)350-3764

* 한국과학기술원 KI로보틱스연구소

** 한국과학기술원 기계항공공학부



Figure 1. AVM camera system&AVM image

AVM은 노면 표시 정보를 직관적으로 획득가능하며 Field of View(FOV)가 넓은 장점이 있고, 저가형 센서 기반 위치 추정에 관한 연구에 적합하다.

2.2 차선 지도 작성

차량 측위를 위한 지도는 실시간 운용을 위해 차선 정보만 나타나있는 간결한 구조로 작성하였고, 이를 바탕으로 차선 수준의 차량 측위를 수행하였다. 우선 목표환경 위성사진 이미지와 목표환경에서 획득한 센서 데이터를 이용하여 그린 지도 이미지 상에서, 차선의 좌푯값을 매뉴얼하게 추출하여 이웃한 점들을 이어 지도를 작성하였다. 지도를 구성하는 점들은 각 구간별로 번호가 매겨져 있고, 한 구간을 정의하는 점이 두 개 이상이므로, 본 논문에서는 이 형태의 지도를 차선 객체지도라 부른다. 이 차선 객체지도는 점들을 이어서 차선 지도 형태를 나타내기는 하지만 점과 점 사이의 구간에 대한 정보는 없으므로 모든 지점에 대해서 좌표를 얻을 수 있도록 한 픽셀 당 1cm의 스케일로 격자를 정의한 차선 격자 지도를 작성하였다.

위성사진은 위성위치에 따라 이미지가 다르고 정밀도가 떨어지지만 목표환경에 대한 광역적 구조 정보를 포함하므로 지도의 전체적인 틀을 잡기에 유용했다. 센서 데이터는 차량 루프랙 양 옆에 장착된 두 대의 라이다로부터 얻은 것으로 intensity 정보를 이용하여 차선을 검출하였고, 3차원 복원 정보를 이용하여 연석을 검출하여 Figure 2와 같은 형태로 나타냈다. 라이다 데이터 기반의 지도에서 같은 지점을 여러 번 지나갈 때 지도가 밀리면서 해당 지점이 이동하는 현상이 발생하였는데, 이는 지도 작성에 쓰인 라이다가 1채널로 충분한 데이터를 얻기 어려웠고, 정밀한

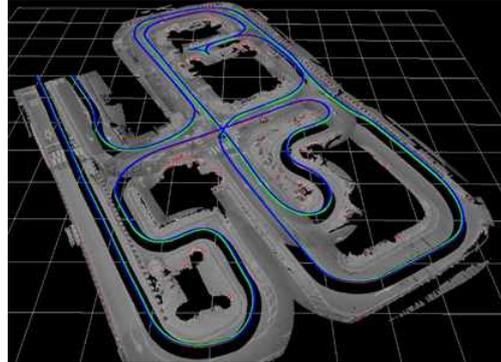


Figure 2. Lidar data based map

DGPS와 같은 GPS를 사용하지 않았기 때문에 오차가 누적되면서 발생한 현상이다. 즉, 라이다 기반 지도에서 구간 구간마다 지역적으로는 차선 지도에 활용 가능한 수준의 정보를 얻을 수 있었지만 광역적으로는 오차가 크게 나타났다.

따라서 광역적으로는 위성사진, 지역적으로는 센서 데이터를 참고하여 이미지 한 픽셀 당 1cm 스케일로 차선을 정의하는 객체지도를 작성하였다. 지도 전체를 M , 지도를 구성하는 객체들을 B , 객체를 구성하는 점을 v 라고 정의하면, 점들이 모여서 객체를 이루고 객체들이 모여서 지도 형태를 완성하므로 차선 객체 지도를 식 (1)~(3)으로 표현할 수 있고 도식화하면 아래 Figure 3과 같이 나타낼 수 있다.

$$M = \{B_1, \dots, B_n\} \quad (1)$$

$$B_i = \{v_i^{(1)}, \dots, v_i^{(m_i)}\} \quad (2)$$

$$v_i^{(j)} = (x_i^{(j)}, y_i^{(j)}) \quad (3)$$

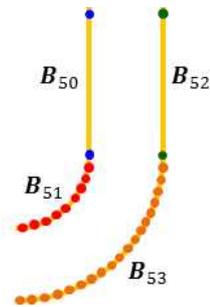


Figure 3. Schematic of lane object map

차선 객체지도는 선을 두 점으로 정의하고 곡선을 여러 개의 점으로 정의하고 있으므로 지도 모든 부분에 대한 정보를 포함하고 있지는 않다. 따라서 지도를 간격이 일정한 격자로 나뉘어서 점이 정의되지 않은 구역까지 좌푯값을 가질 수 있도록 하는 작업이 요구되어, 차선 객체지도를 바

탕으로 UTM 좌표계 상의 좌표 정보를 포함하는 격자 지도를 생성하였다.

3. AVM 이미지를 이용한 차량 측위

AVM 이미지를 이용한 차량 측위는 우선 지도 상에서 차량이 있을만한 후보 위치를 몇 곳 선정 한 후, AVM 이미지에서 검출한 차선과 지도 간의 ICP 정합을 통해 차량이 존재할 가능성이 가장 높은 위치를 현재 차량의 위치로 추정한다. 아래 Figure 4.는 이러한 과정을 도식화 한 것이다.

3.1 현재 차량 후보 위치지정

현재 차량의 후보 위치는 바로 이전 위치에서 GPS 위치 변화량만큼 이동한 위치, 현재 GPS에 의한 위치, 그리고 앞의 두 위치의 중간 지점으로 설정한 임의의 위치 세 가지가 있다. 이전 위치에서 GPS 위치변화량만큼 이동한 위치를 정할 때 GPS 위치변화량 계산은 식 (4)~(10)과 같다.

$$\delta_{trans} = \sqrt{(\Delta y_{t,t-1}^{gps})^2 + (\Delta x_{t,t-1}^{gps})^2} \quad (4)$$

$$\delta_{rot1} = atan2(\Delta y_{t,t-1}^{gps}, \Delta x_{t,t-1}^{gps}) - \theta_{t-1}^{gps} \quad (5)$$

$$\delta_{rot2} = \theta_t^{gps} - \theta_{t-1}^{gps} - \delta_{rot1} \quad \text{where} \quad (6)$$

$$\Delta y_{t,t-1}^{gps} = y_t^{gps} - y_{t-1}^{gps}, \Delta x_{t,t-1}^{gps} = x_t^{gps} - x_{t-1}^{gps} \quad (7)$$

$$x_t^{(1)} = x_{t-1} + \delta_{trans} \cdot \cos(\theta_{t-1} + \delta_{rot1}) \quad (8)$$

$$y_t^{(1)} = y_{t-1} + \delta_{trans} \cdot \sin(\theta_{t-1} + \delta_{rot1}) \quad (9)$$

$$\theta_t^{(1)} = \theta_{t-1} + \delta_{rot1} + \delta_{rot2} \quad (10)$$

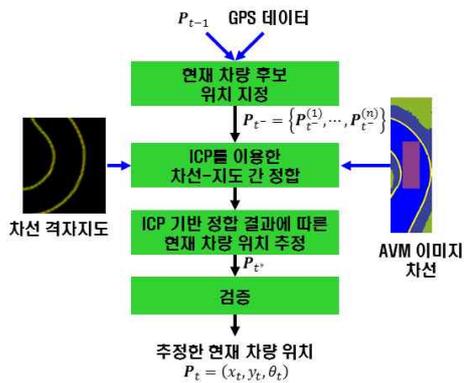


Figure 4. AVM images based vehicle localization process

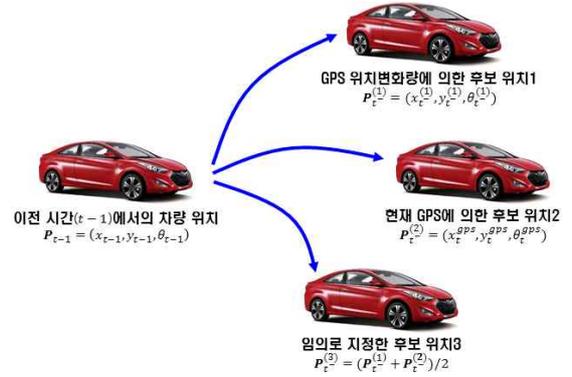


Figure 5. Past vehicle location and current vehicle candidate locations

3.2 후보 위치들로부터의 ICP 기반 현재 차량 위치 추정

이전 위치에서 GPS 위치 변화량만큼 이동한 위치를 후보 위치1로 두고, ICP에 의한 위치 보정을 시도한다. 후보 위치1에 대해 ICP가 성공하면 보정된 위치를 현재 차량의 위치로 설정하고, 후보위치1에 대한 위치 보정을 실패하면 나머지 후보 위치들 중 ICP에 의한 위치 보정이 가장 잘 된 위치를 현재 위치로 설정한다. 여기서 ICP 성공은 AVM-LGM 간의 정합 점수가 높다는 것을 뜻한다. 모든 후보 위치에 대한 ICP 보정이 실패할 경우에는 알고리즘적으로 할 수 있는 부분이 없기 때문에 GPS 위치 변화량만큼 이동한 위치(후보 위치1)를 차량의 현재 위치로 사용한다. 현재 위치에서의 GPS 정보(후보 위치2)를 차량 위치로 사용하면 GPS 데이터 오프셋이 발생하여 실제 차량의 위치와의 오차가 클 수 있으므로 GPS 위치 변화량을 사용하는 것이다.

$$s_m^{(i)} : P_t^{(i)} \text{ (후보위치 } i) \text{에서의 ICP 정합점수} \quad (11)$$

$$R^{(i)}, T^{(i)} : P_t^{(i)} \text{에서의 ICP 변환결과} \quad (12)$$

$$\delta_{trans}, \delta_{rot1}, \delta_{rot2} : \text{GPS 위치 변화량} \quad (13)$$

일 때, 후보 위치에 대한 ICP 적용 과정은 아래 그림과 같이 나타낼 수 있고, ICP 기반 정합 결과에 따른 현재 차량의 위치는 아래와 같은 함수 형태로 표현할 수 있다.

$$P_t^* = f(P_{t-1}, R, T) \quad (14)$$

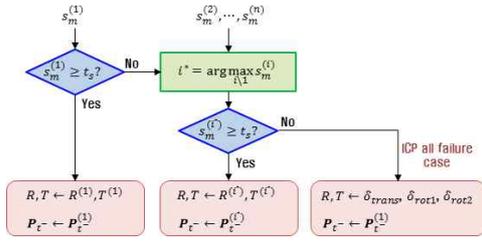


Figure 6. Estimation current location from candidate locations

3.3 후보 위치에서의 ICP 기반 차량 위치 보정
 차량의 최종 위치는 차량의 후보 위치에 대해 ICP 알고리즘을 적용하여 결정한다. 우선, AVM 이미지에서 분할된 차선영역에 대한 세션화(thinning)을 통해 골격화된 차선을 획득하고, ICP 알고리즘을 사용하여 AVM 이미지로부터 얻은 차선 점들과 차선 격자 지도의 차선 점들 사이의 변환 관계를 계산하여 차량 위치를 보정한다.

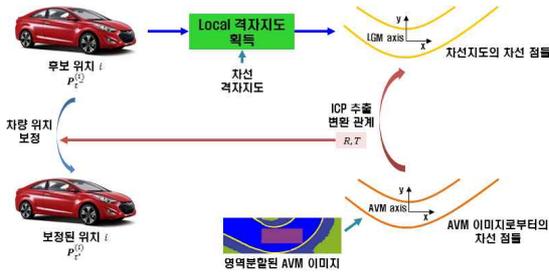


Figure 7. Vehicle localization by using ICP Algorithm

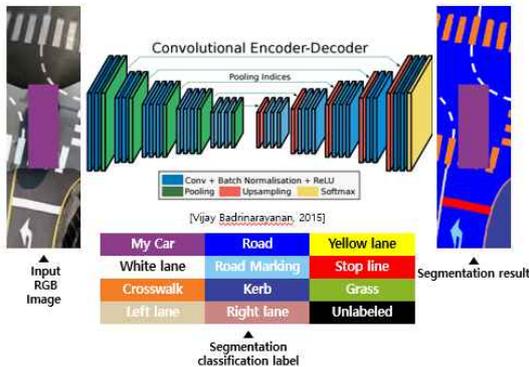


Figure 8. AVM Image Segmentation Deep Convolutional encoder-decoder network [9]

AVM 이미지로부터의 차선 영역 분할은 딥 러닝 네트워크 학습 기법[9]을 이용하였다. 주행 영역에 대한 AVM 이미지에 나타난 노면 종류별 색을 다르게 해서 다량의 학습 데이터를 만든

후, 딥 러닝 네트워크를 학습시켜 실시간으로 들어오는 AVM 이미지에서 노면 표식을 종류별로 분리할 수 있게 하였다. 그리고 분할된 AVM 이미지에서 차선에 대한 분할 정보만 사용하여 AVM 차선 지도를 얻었다. (Figure 8.)

수평성분만 있는 직선 차선 구간에서는 ICP 결과에서 불확실성이 발생하여 차량 진행방향에 대한 오차가 커진다. 이러한 ICP 불확실성 문제를 해결하기 위해 AVM 이미지에서 차선을 이루는 점들의 수평 정도를 계산하고 이를 차량 위치 보정에 활용하는데, AVM 이미지의 차선 점들과 차선 지도 차선 점들 사이의 정합 점수와 AVM 이미지의 차선 점들의 수평 정도에 따라 ICP 결과를 선택적으로 사용한다. 여기서 정합 점수란 차량 위치에 대한 Local Grid Map(LGM)과 실시간 AVM 이미지를 비교하여 차선 모양이 일치하는 정도로 본 연구에서는 정합 점수가 0.2(t_s , 20%) 이상이 되어야 ICP가 성공한 것으로 판단하였다.

AVM 이미지로부터 분리한 차선 정보를 세션화한 결과를 AVM pnt라 하고, AVM 이미지와 대응되는 위치에서 지역적인 차선 격자 지도를 LGM pnt라고 하면, ICP를 적용하는 과정은 먼저 실시간으로 들어오는 AVM 이미지에 대해서 AVM pnt의 수평, 수직 정도를 필터로 추출하는 것으로 시작한다. 그 다음 AVM pnt와 LGM pnt 간의 정합 정도로 점수를 매기게 되는데, 점수가 낮으면 ICP 적용을 포기한다. 점수가 높다면 AVM pnt의 수평 성분이 높을 경우($t_h > 0.8$, Figure 9.)는 차량의 횡방향으로만 ICP를 적용하는 Forced $T_x = 0$ ICP를 사용하고, 수평 성분이 낮을 경우는 종방향, 횡방향으로 완전한 ICP를 사용한다. 그 과정을 수식으로 나타내면 식 (15)~(24)과 같다.

$$G_{x1}: \text{차선의 한 점에서의 x축방향 픽셀 intensity 합의 미분값(backward)} \quad (15)$$

$$G_{x2}: \text{차선의 한 점에서의 x축방향 픽셀 intensity 합의 미분값(forward)} \quad (16)$$

$$G_{y1}: \text{차선의 한 점에서의 y축방향 픽셀 intensity 합의 미분값(backward)} \quad (17)$$

$$G_{y2}: \text{차선의 한 점에서의 y축방향 픽셀 intensity 합의 미분값(forward)} \quad (18)$$

$$G_x = \max(|G_{x1}|, |G_{x2}|) \quad (19)$$

$$G_y = \max(|G_{y1}|, |G_{y2}|) \quad (20)$$

$$h_i = \begin{cases} 1 & \text{if } G_y \geq G_x \\ 0 & \text{otherwise} \end{cases} \quad (21)$$

s_m : AVM-LGM 차선 점들 간 정합 점수 (22)

n_p : AVM 이미지로부터 분할된 차선(세선화)을 이루는 점(픽셀)의 개수 (23)

s_h : $|\sum h_i|/n_p$ (수평인 지점의 비율) (24)

ICP 적용 과정을 아래 Figure 9와 같이 나타낼 수 있다.

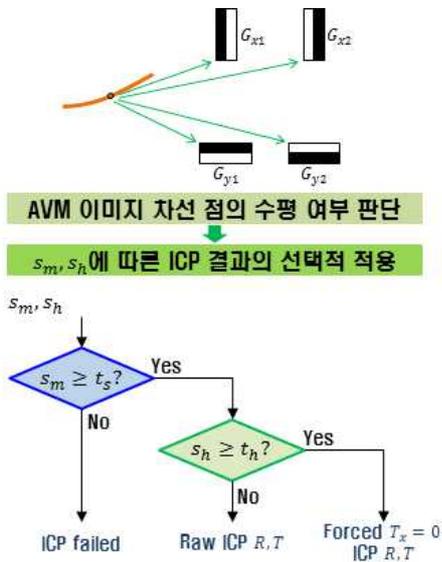


Figure 9. ICP based vehicle localization considering horizontal component of AVM lane images

3.4 현재 차량 위치 추정에 대한 검증

현재 차량 위치에 대한 검증으로 추정된 현재 차량 위치가 GPS에 의한 차량 위치와 차이가 심할 경우에 위치를 보정하도록 하였고, 차량이 지도상에서 사전에 정의해놓은 접근 불가능 격자에 위치하면 접근가능 격자로 위치를 보정하였다.

4. 실험 결과

4.1 실험환경

실험은 인천광역시 서구 소재, 신진 자동차 운전 전문학원에서 진행하였다. 실험환경은 운전 시험장 특성상 차선 간격이 좁고, 직각구간, S자 곡선구간, T자 주차구간, 교차로 등 다양한 코스를 포함한다. 그리고 시험장 전체 모양은 직사각형으로, 크기는 가로 64m, 세로 99m이다. 이에 대한 차선 격자 지도는 총 1364의 점과 73개의 차선 객체로 구성했다.



Figure 10. satellite picture & Lane grid map

4.2 실험결과 및 분석

Figure 11~14는 지도의 일부 구간에서의 위치 추정 결과를 나타낸 것으로, 빨간 경로가 GPS 정보, 파란 경로가 차량 위치 추정 알고리즘 적용 결과이다. Figure 11은 수평 구간으로 Forced $T_x = 0$ 이 적용되어 차량 진행 방향이 GPS 위치 변화량과 같은 경향을 보이는 것을 확인할 수 있다. Figure 12에서는 파란 경로가 급격히 변화하는 현상을 확인할 수 있는데, 이는 Figure 9에 나타냈듯이 후보위치1에서 ICP 보정을 실패한 후, 후보위치2에 대한 로직으로 넘어가서 ICP 보정을 성공한 경우에 나타나는 현상이다. 실험결과 지도 영역 전체적으로 AVM 이미지와 지도상에서 차량의 위치가 일치하는 것을 확인할 수 있었고, 특히 지도상에서 위치 추정 결과가 차선을 넘어가지 않고 계속해서 보정되는 것을 확인할 수 있었다. 그리고 교차로나 주차 구간에서도 본 알고리즘이 정상적으로 작동하였다. (Figure 13., 14.)

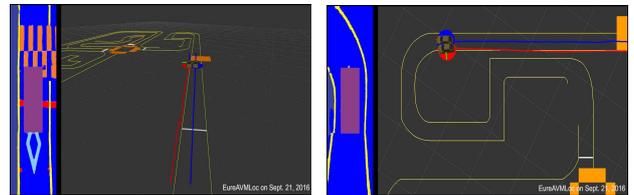


Figure 11. Localization verification with real-time AVM images: case(1) Forced $T_x = 0$



Figure 12. Localization verification with real-time AVM images: case(2) candidate location change

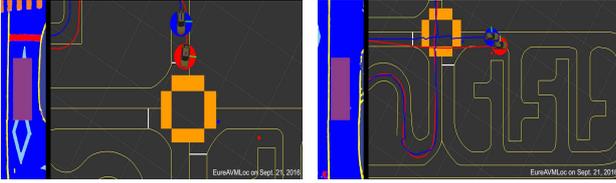


Figure 13. Localization verification with real-time AVM images: case(3) intersection without lane



Figure 14. Localization verification with real-time AVM images: case(4) T-shape parking lot

5. 결론

본 연구에서는 Monte-Carlo Localization과 ICP 알고리즘을 이용한 차선 수준 차량 측위 방법을 제안하였다. 차량 후보 위치와 ICP 기반의 최적화된 위치 추정을 통해 측위에 있어서 일정 수준 이상의 정확도를 유지하려고 하였지만 지도 오차에 대한 uncertainty 모델링의 부재로 실제 구간과 동일한 형상의 지도를 작성하기 어려웠고, 특히 곡선 차선 구간에서 측위 오차가 크게 나타났다. 그리고 좁은 FOV를 갖는 AVM 카메라를 사용하여 2차선 이상의 도로에서 차량의 위치에 대한 모호성이 생겼고, 전 구간에서 차선이 명확한 환경에서만 본 알고리즘을 적용 가능하다는 한계점이 발생하였다.

제안한 알고리즘을 일반 도로 환경에 적용하기 위해서는 위의 한계점을 극복해야 한다. 우선 자동화된 지도 작성 방법 개발하여 지도 작성 시간을 줄이고, 지도 오차에 대한 uncertainty 모델링을 통해 지도의 정확도를 향상시켜 [8]과 같은 정밀 도로 지도 기반의 자율주행이 가능하도록 해야 할 것이다. 그리고 FOV가 넓은 AVM 카메라를 사용하고, 차선뿐만 아니라 모든 노면 표시를 활용하여 위치 추정 결과가 일반 도로에서도 보장될 수 있도록 개선해야 할 것이다.

후 기

This material is based upon work supported by the Ministry of Trade, Industry & Energy(MOTIE, Korea) under Industrial Technology Innovation

Program. No.10051867, Development of real-time localization and 3-D mapping system with cm-level accuracy based on digital maps and vision data for autonomous driving.

참고문헌

- [1] Gruyer, D., Belaroussi, R. and Revilloud, M., 2016, "Accurate lateral positioning from map data and road marking detection", *Expert Systems with Applications*, Vol. 43, No.1 pp. 1~8.
- [2] Unghui Lee, Sangyol Yoon, Inwook Shim, Seunghak Shin, Jongwon Choi, Jiwon Oh, David Hyunchul Shim, Inso Kweon, and Seibum Choi, 2013, "KAIST EureCar: Development of an Autonomous Car including Around-Environment Recognition and Collision Avoidance on the Complicated Road", *Robot and Human*, Vol. 10, No.2, pp. 20~31
- [3] Kim, S. W., Kim, D. W., Lee, K. W. and Yi, K. S., 2015, "Lane-Level Localization and Steering Control for Automated Vehicles in Urban Environments", *Transactions of Korean Society of Mechanical Engineers*, Vol. 2015, No. 11, pp. 688~693.
- [4] Jo, K., Jo, Y., Suhr, J. K., Jung, H. G. and Sunwoo, M., 2015, "Precise Localization of an Autonomous Car Based on Probabilistic Noise Models of Road Surface Marker Features Using Multiple Cameras", *IEEE Transactions on Intelligent Transportation Systems*, Vol. 16, No. 6, pp. 3377~3392.
- [5] Levinson J. and Thrun, S., 2010, "Robust Vehicle Localization in Urban Environments Using Probabilistic Maps", *IEEE International Conference on Robotics and Automation (ICRA)*, pp. 4372~4378.
- [6] Chausse, F., Laneurit, J. and Chapuis, R., 2005, "Vehicle localization on a digital map using particles filtering", *IEEE Intelligent Vehicles Symposium*, pp. 243~248.
- [7] Najjar, Maan E. El and Bonnifait P., 2005, "A Road-Matching Method for Precise Vehicle Localization Using Belief Theory and Kalman Filtering", *Autonomous Robots*, Vol. 19, Issue 2, pp. 173~191.
- [8] Schreiber, M., Knoppel, C. and Franke, U., 2013, "LaneLoc: Lane marking based localization using highly accurate maps", *2013 IEEE Intelligent Vehicles Symposium (IV)*, pp. 449~454.
- [9] Badrinarayanan, V., Kendall, A. and Cipolla R., 2014, "SegNet: A Deep Convolutional Encoder-Decoder Architecture for Robust Semantic Pixel-Wise Labelling", *Computer Vision and Pattern Recognition, ID. 1468*.