차량 모델 및 LIDAR를 이용한 맵 매칭 기반의 야지환경에 강인한 무인 자율주행 기술 연구

The Research of Unmanned Autonomous Navigation's Map Matching using Vehicle Model and LIDAR

박재웅*, 김재환, 김정하
 (Jae-Ung Park¹, Jae-Hwan Kim¹, and Jung-Ha Kim¹)
¹Kookmin University

Abstract: Fundamentally, there are 5 systems are needed for autonomous navigation of unmanned ground vehicle: Localization, environment perception, path planning, motion planning and vehicle control. Path planning and motion planning are accomplished based on result of the environment perception process. Thus, high reliability of localization and the environment perception will be a criterion that makes a judgment overall autonomous navigation. In this paper, via map matching using vehicle dynamic model and LIDAR sensors, replace high price localization system to new one, and have researched an algorithm that lead to robust autonomous navigation. Finally, all results are verified via actual unmanned ground vehicle tests.

Keywords: UGV, autonomous navigation, vehicle model, LIDAR, map building, map matching, localization

1. 서론

첨단 기술의 발전 및 IT 산업의 발전으로 다양한 분야에서 무인화 산업의 연구 개발 및 적용이 이루어지고 있다. 특히, 국방산업 관련 분야와 극한의 환경 속에서 수행되는 분야들을 선두로 무인화 산업 개발이 가속화 되고 있는 상황이며, 이들 무인화 산업의 핵심은 무인 차량을 위한 자율주행 기술 개발 및 적용이라고 할 수 있다. 본 연구에서 다루고 있는 무인 차량은 무인지상차량(unmanned ground vehicle)으로 군사적 용도로 특수 제작된 지상차량이 아니라, 일반 상용 SUV를 이용하여 무인 차량으로 개조하였으며, 스스로 환경을 인식하고 판단하여 주어진 임무를 수행할 수 있는 자율주행 알고리즘이 이식된 무인 자율주행 차량이 되겠다. 본 연구와 같이 상용 차량을 개조해 무인 자율주행 차량을 개발한 대표적인 사례가 DARPA Challenge이며, 미국 국방성 산하 고등연구 계획국(defense advanced research project agency) 주관으로 개최된 무인 자율주행 차량 대회이다. DARPA Challenge는 미국의 국방시스템에 무인 자율주행 차량을 도입하고 사용하기 위한 인프라 확장을 목적으로, 2004, 2005년에는 야지환경을 대상으로 한 DARPA Grand Challenge를, 2007년에는 도심환경을 대상으로 한 DARPA Urban Challenge를 개최하여 높은 수준의 무인화 산업의 기술력을 확보할 수 있었다[1,2]. 국내에서도 이러한 연구가 국방과학연구소를 중심으로 활발하게 진행되고 있으나 아직 미국의 연구 결과 수준에 크게 못 미치고 있는 상황이다. 최근에는 국방분야 외에 민수분야에서

도 이러한 연구가 시작되었으며, 특히 지능형 자동차 개발을 목표로 완성차 업체 및 관련 부품업체가 연구에 박차를 가하고 있다. 2010년에는 현대기아자동차에서 국내 대학을 대상으로 무인 자율주행 자동차 연구경진대회를 열어 포장 및 비포장도로가 혼재된 약 4km 정도의 코스에서 장애물 회피, 차선 및 속도 준수 등 다양한 임무를 수행할 수 있는 무인 자율주행 자동차를 개발하도록 하였으며, 그 결과 국내 관련 연구의 활성화 및 연구 수준의 향상에 크게 기여하였다.

이러한 무인 자율주행이 가능하기 위해서는 기본적으로 5가지 기술이 필요하며, 각각은 현재 차량의 위치를 인식할 수 있는 위치인식(localization)기술, 주변의 지형, 장애물, 차선 등의 환경을 인식할 수 있는 환경인식(environment perception)기술, 목적지까지의 최적의 경로와 장애물을 회피할 수 있는 경로를 생성하는 경로계획(path planning)기술, 인식된 환경과 추종해야 할 경로 및 차량 상태를 고려해 최적의 주행 명령을 생성하는 주행계획(motion planning)기술, 차량 상태 정보 획득 및 명령 추종을 위한 구동기 제어를 하는 차량제어(vehicle control)기술이 있다. 이 중, 경로계획과 주행계획은 위치인식과 환경인식 결과를 기반으로 이루어지기 때문에, 위치인식과 환경인식의 높은 신뢰도는 전체적인 자율주행의 완성도를 판단할 수 있는 기준이 되고 있다[3,4]. 위치인식의 경우 실외 자율주행을 하고 있기 때문에, GPS (Global

* 책임저자(Corresponding Author)

논문접수: 2011. 2. 20., 수정: 2011. 3. 10., 채택확정: 2011. 3. 29.

박재웅, 김재환: 국민대학교 자동차공학전대학원

(parkju502@hanmail.net/johnkim@kookmin.ac.kr)

김정하: 국민대학교 자동차공학과(jhkim@kookmin.ac.kr)

※ 본 논문은 지식경제부 산업원천기술개발사업[과제번호10035354, 실외환경에 강인한 도로기반 저가형 자율주행기술개발]과 2010년 국민대학교 교내연구비의 지원에 의해 수행되었음.

그림 1. 무인 자율주행 차량.

Fig. 1. Example of unmanned ground vehicle.

Positioning System)를 주 시스템으로 사용하고 있으며, INS (Inertial Navigation System)와 융합하여 GPS를 통한 위치 인식이 불가능한 경우에도 지속적인 위치인식이 가능하도록 개발하고 있다. 특히, 험준한 지형에서 운용되는 국방 무인 차량의 경우는 우거진 나무들에 의해서 GPS를 이용한 위치인식이 용이하지 못하고, 차량의 roll, pitch, yaw의 변화가 심한 비포장 도로에서 INS의 누적 오차도 크게 발생하는 상황이 지속적으로 발생할 수 있기 때문에 신뢰도가 높은 고가의 DGPS (Differential GPS) 및 INS를 사용하고 있다. 환경인식은 크게 카메라와 LIDAR (Light Detection And Ranging)를 주 센서로 사용하고 있으며, 특히 레이저 기반의 LIDAR 센서는 높은 신뢰도의 거리 정보와 빠른 데이터 획득 속도, 긴 측정 거리, 넓은 측정 범위와 높은 분해능을 가지고 있어 환경인식에 있어 중요한 역할을 담당하고 있다. 최근에는 거리 정보 외에 반사율을 이용한 인식 및 3D LIDAR 센서를 이용한 인식까지 이루어지고 있다[5].

이에 본 논문은 위치인식이 어렵거나 불확실한 환경에서 고가의 위치인식 센서를 사용해야 하는 단점과 GPS 및 INS가 고장 났을 경우 더 이상 자율주행이 불가능해지는 문제점을 극복할 수 있는 강인한 무인 자율주행 기술 개발에 연구 목적을 두었다. 이를 위해 차량 모델과 기하학적 모델 그리고 조향각 센서 및 휠 엔코더를 이용해 차량의 현재 위치 및 헤딩(heading)을 추정하고, 실시간 추정된 현재 위치 및 헤딩을 이용해 글로벌 맵(global map)과 LIDAR 센서로부터 생성된 로컬 맵(local map)과의 맵 매칭(map matching)을 통해 자율주행을 할 수 있는 알고리즘을 개발하였다. 그리고 실제 무인 자율주행 차량을 이용하여 그 결과를 검증하였다.

II. 시스템 설계

1. 시스템 아키텍처

무인 자율주행 차량은 서론에서 설명하였듯이, 위치인식, 환경인식, 경로계획, 주행계획, 차량제어기술 등이 유기적으로 융합되어 시스템을 운영하고 있으며, 다수의 이중 센서 및 구동기 그리고 컨트롤러로 구성되어 있다. 또한 더 나아가 다수의 무인 자율주행 차량도 운영할 수 있어야 하기 때문에, 시스템 아키텍처를 체계적으로 잘 구축해야 한다. 따라서 본 기관에서는 현재 미국이 군사용 목적으로 제한한 삼군

그림 2. 무인 자율주행 차량을 위한 시스템 아키텍처.
Fig. 2. System architecture for unmanned ground vehicle.

통합형 무인 시스템 아키텍처인 JAUS (Joint Architecture for Unmanned System)를 참조한 시스템 아키텍처를 사용하였다[6].

무인 자율주행 차량을 구성하는 Node는 4개이며 위치인식을 수행하는 Navigation Node, 환경인식을 수행하는 Environment perception Node, 경로계획 및 주행계획을 수행하는 Arbiter Node 마지막으로 차량제어를 수행하는 Vehicle control Node로 구성된다.

2. 개발 환경

본 연구에 사용된 차량은 현대자동차의 베라크루즈를 사용하였으며, 시스템 하드웨어 구성은 그림 3과 같다.

LIDAR 센서는 SICK社의 LMS291S_05를 사용하였으며, 사양은 그림 4와 같다.

LIDAR 센서는 차량에 총 6개를 장착하였으며, 장착 위치는 그림 5와 같다.

차량의 이동 거리 측정을 위한 엔코더는 Autronics社의 E30S Series를 사용하였으며, 사양은 그림 6과 같다.

엔코더는 4바퀴 모두에 장착되었으며, 외부 충격에 의한

그림 3. 무인 자율주행 시스템의 하드웨어 구성.

Fig. 3. System hardware configuration of unmanned ground vehicle.

- Acquisition of Obstacle Position Data
- SICK LMS291S_05
- Maximum 80m Detection
- Angular Range: 100°/180°
- Angle Resolution: 0.25°/0.5°/1°
- Distance Resolution: 1cm
- Acquisition Time: 75Hz/37Hz/19Hz

그림 4. LIDAR 센서 사양.

Fig. 4. LIDAR sensor specification.

그림 5. LIDAR 센서 장착 사진.

Fig. 5. LIDAR sensors installation location.

	Type		Incremental
	Resolution	(P/R)	100
	Output Phase		A, B, Z
	Power Supply	VDC	5 ± 5%
	Max. freq response	kHz	300
	Adjusting a zero position	rpm	5000

그림 6. 엔코더 사양.

Fig. 6. Encoder specification.

그림 7. 엔코더 장착 위치.

Fig. 7. Encoder installation location.

	Angular Range	deg	-780 to +780
	Angular Resolution	deg	0.1
	Accuracy	deg	± 1.4
	Power Supply	VDC	+ 8 ~ 16
	Calibration control		Via CAN bus
	Adjusting a zero position		Via CAN bus

그림 8. 조향각 센서 사양.

Fig. 8. Steering wheel angle sensor specification.

그림 9. 조향각 센서 장착 위치.

Fig. 9. Steering wheel angle sensor installation location.

훼손 및 이물질에 의한 오염을 방지하기 위해 바퀴 내부에 장착하였다. 장착 모습은 그림 7과 같다.

차량의 조향각 측정을 위한 조향각 센서(steering wheel angle sensor)는 비접촉 다중회전센서(non-contact multi-turn angle sensor)를 사용하였으며, 사양은 그림 8과 같다.

조향각 센서는 스티어링 칼럼과 스티어링 휠 사이에 장착되었으며, 장착 모습은 그림 9와 같다.

III. 차량 위치 및 heading 추정

본 연구에서는 차량 모델과 기하학적 모델을 기반으로 차량의 현재 위치 및 heading을 추정하였다. 차량 모델은 선회반경을 구하기 위해 사용하였으며, 구해진 선회반경을 기하학적 모델에 적용하여 차량의 현재 위치 및 heading을 추정하였다.

1. 차량 모델

차량 모델은 그림 10과 같이 바이시클 모델(bicycle model)을 적용하였으며, 이를 위해서 다음과 같은 몇 가지 전제조건을 두었다. 우선 수평면내를 운동하는 차량에 있어서 좌우

그림 10. 바이시클 모델.

Fig. 10. Top view of the bicycle model.

타이어의 옆 미끄럼각이 같으며 그 값은 충분히 작다고 가정하였다. 또한 차체의 물과 일정 속도로 주행하는 차량의 타이어 트레드도 무시하게 되면, 전후륜의 좌우륜이 등가적인 차량의 전후륜과 차축과의 교점에서 각각 집중된 전후륜 2륜의 차량 모델을 얻을 수 있다[7].

그림 10에서 P는 차량 중심점, r 은 P점에서의 yaw rate, β 는 P점에서의 타이어 옆 미끄럼각, V 는 차량 속도, δ 는 조향각, l_f , l_r 은 각각 차량중심점에서부터 전후축간의 거리를 나타낸다. 또한 차량 전후륜 타이어의 옆 미끄럼각이 β_f , β_r 이고, 이들 타이어에 작용하는 코너링 포스(cornering force)는 Y_f , Y_r , 코너링 파워(cornering power)는 K_f , K_r 로 나타냈다. 실조향각 $\delta \ll 1$ 인 차량에 대한 옆 미끄럼각과 코너링 포스는 다음과 같이 나타낼 수 있다.

$$\beta_f = \beta + \frac{l_f r}{V} - \delta \quad (1)$$

$$\beta_r = \beta - \frac{l_r r}{V} \quad (2)$$

$$Y_f = -K_f \beta_f = -K_f \left(\beta + \frac{l_f r}{V} - \delta \right) \quad (3)$$

$$Y_r = -K_r \beta_r = -K_r \left(\beta - \frac{l_r r}{V} \right) \quad (4)$$

이때, 좌우 타이어 자체의 특성과 좌우 타이어에 작용하는 코너링 포스가 같다면, 전후륜 코너링 포스는 각각 $2Y_f$, $2Y_r$ 로 나타낼 수 있다. 이 힘은 y축방향, 즉 차량의 횡방향에 작용하기 때문에 차량의 횡방향 운동은 다음과 같다.

$$mV \left(\frac{d\beta}{dt} + r \right) = 2(Y_f + Y_r) \quad (5)$$

$$I \frac{dr}{dt} = 2(l_f Y_f - l_r Y_r) \quad (6)$$

여기에서 m 은 차량의 관성질량, I 는 차량의 yawing 관성모멘트를 나타낸다. 위 식에서 정의된 타이어의 옆 미끄럼 각과 타이어에 작용하는 코너링 포스에 의한 바이시클 모델의 차

량 횡방향 운동과 연직축 둘레의 yawing 운동은 다음과 같이 나타낼 수 있다.

$$mV \frac{d\beta}{dt} + 2(K_f + K_r)\beta + \left\{ mV + \frac{2}{V}(l_f K_f - l_r K_r) \right\} = 2K_f \delta \quad (7)$$

$$2(l_f K_f - l_r K_r)\beta + I \frac{dr}{dt} + \frac{2(l_f^2 K_f + l_r^2 K_r)}{V} r = 2l_f K_f \delta \quad (8)$$

식 (7)과 (8)에서 m, K_f, K_r, l_f, l_r, I 는 차량의 고유 특성 값이기 때문에 상수 값이며, V 와 δ 는 각각 엔코더와 조향각 센서로부터 입력 받게 된다. 따라서 식 (7)과 (8)을 연립하여 해를 구하면, β 와 r 을 얻게 된다.

차량 모델을 통해 최종적으로 얻고자 하는 값은 선회반경이기 때문에, 식 (9)와 같이 앞서 계산된 yaw rate r 과 엔코더를 통해 획득된 차량 속도 V 를 대입하면 선회반경 ρ 를 구하게 된다.

$$\rho = \frac{V}{r} \quad (9)$$

2. 기하학적 모델

차량의 위치 및 heading 추정을 위해 그림 11과 같은 선회반경을 이용한 기하학적 모델을 사용하였다. 차량은 하나의 강체로 가정하였으며, 차량 중심점 P 를 기준으로 위치와 heading을 계산하였다.

차량 모델로부터 얻어진 선회반경 ρ 와 차량의 각 타이어에 장착된 엔코더로부터 계산된 평균 순간 이동거리 값 L 로부터 회전각 θ 를 구할 수 있다.

$$\theta = \frac{L}{\rho} (\text{rad}) \quad (10)$$

이를 바탕으로 이전 위치 (x_1, y_1) 으로부터 차량의 선회중심점 O 를 기준으로 θ 만큼 회전한 현재 위치 (x_2, y_2) 를 구할 수 있다.

$$\begin{bmatrix} x_2 \\ y_2 \end{bmatrix} = \text{Rot}(\theta) \begin{bmatrix} x_1 \\ y_1 \end{bmatrix} \quad (11)$$

또한 현재 위치 P_2 에서의 heading각 ψ_2 는 P_1 에서의 heading각 ψ_1 를 기준으로 θ 만큼 회전한 각도가 된다.

그림 11. 기하학적 모델.

Fig. 11. Geometrical model.

$$\psi_2 = \psi_1 + \theta \quad (12)$$

IV. LIDAR 기반의 맵 빌딩

1. 환경정보 검출 및 로컬 맵 생성

레이저 기반의 LIDAR 센서는 TOF (Time of Flight) 방식으로 방출된 단일 레이저 펄스가 센서 범위 내에 존재하는 사물의 표면에 반사되어 돌아오는 시간을 거리로 환산해 사용자에게 제공해준다. 따라서 point cloud 형태로 물체와의 상대적인 거리 정보를 제공하게 되는데, 이 거리정보를 이용하여 차량 주변의 환경을 인식하게 된다. 검출된 환경정보는 실시간 일정한 크기와 셀(cell)을 가지는 로컬 맵에 맵핑(mapping)하게 되는데, 셀에는 장애물 유, 무에 따른 불(boolean)값이 입력된다. 로컬 영역의 크기는 20.5m×20.5m이며, 셀 크기는 0.5m×0.5m로 최종적인 로컬 맵 크기는 그림 12와 같이 41×41이다.

그림 13은 장착된 6개의 LIDAR 센서 중, 회피해야 할 장애물을 검출하는 4개의 LIDAR 센서로부터 검출된 환경에 대

그림 12. 로컬맵 설정.

Fig. 12. Local map settings.

그림 13. LIDAR 센서에 의해 검출된 환경 (1).

Fig. 13. Detected environment (1) from LIDAR sensors.

그림 14. LIDAR 센서에 의해 검출된 환경 (2).

Fig. 14. Detected environment (2) from LIDAR sensors.

한 그래프와 로컬 맵이다.

그림 14는 주행 가능 영역을 검출하기 위해 차량 지붕에 장착돼서 지면을 스캔하는 2개의 LIDAR 센서로부터 검출된 그래프이며, 가장 오른쪽 그래프는 회피해야 할 장애물 정보와 도로 정보를 분리한 모습이다.

2. 맵 빌딩

그림 13과 14에서 알 수 있듯이 LIDAR 센서로부터 획득된 환경정보는 높은 정확도를 나타내고 있다. 따라서 이동하는 무인 차량의 정확한 현재 위치와 heading을 알고 있다면, 스캔된 환경정보를 이동 및 회전변환 시켜서 하나의 맵에 지속적으로 업데이트하게 되면 전체 주행환경이 맵핑된 글로벌 맵을 얻을 수 있다. 이와 같은 기술을 맵 빌딩(map building)이라고 한다[8,9]. 글로벌 맵의 크기는 주행 환경을 모두 포함할 수 있는 비교적 큰 맵으로 설정하고, 셀 크기는 로컬 맵과 동일한 0.5m×0.5m이며 셀에는 장애물 유, 무에 따른 불 값이 입력된다.

글로벌 맵은 크게 두 가지로 구분하여 사용하게 되는데, 하나는 절대적인 기준이 되는 맵으로 로깅을 통해 데이터베이스화한 글로벌 맵이며, 다른 하나는 실시간 위치인식을 통해 맵핑되어 사용되는 글로벌 맵이다. 우선 데이터베이스된 글로벌 맵은 맵 매칭에 있어서 기준으로 사용되기 때문에 최대한 위치 오차가 없는 맵 빌딩을 해야 한다. 따라서 본 연구에서는 고가의 RTK (Real Time Kinematic)-GPS와 INS를 연동한 위치인식 시스템을 사용하여 맵 빌딩을 했다. 실시간 글로벌 맵 빌딩에서는 기본적으로 앞서 소개한 차량 모델을 이용해 추정된 위치 및 heading을 사용하였다. 그러나 최초 주행 시작 위치에서는 절대적인 위치와 heading이 있어야 하기 때문에, GPS를 이용해 초기 위치 및 heading을 얻는 초기화 단계를 거쳐야 한다. 그림 15는 맵 빌딩 순서도이다.

3. 필터 설계

높은 신뢰도의 거리 정보를 출력하는 LIDAR 센서이지만, 야지 환경에서 이동 및 회전을 하면서 계속된 거리 정보이기 때문에 계속되는 과정 혹은 맵핑하는 과정에서 노이즈가 포

그림 15. LIDAR 기반의 맵 빌딩 순서도.

Fig. 15. LIDAR-based map building flow chart.

함되게 된다. 따라서 정확한 글로벌 맵을 생성하기 위해서는 이러한 노이즈를 제거할 수 있는 필터를 설계해야 한다. 본 연구에서는 고정된 장애물은 차량이 주행을 하고 있어도 검출 영역에서 벗어나기 전까지 동일한 절대 위치에 연속적으로 맵핑이 된다는 점을 이용해 노이즈를 제거하고 비교적 정확한 장애물 윤곽을 맵핑하도록 하는 필터를 설계하였다.

필터 설계를 위해 글로벌 맵과 동일한 크기의 임시 맵을 생성하고 글로벌 맵 빌딩과 동일하게 맵핑을 수행한다. 그러나 임시 맵은 글로벌 맵과는 달리 셀에 불 값이 아닌 1byte의 정보를 담을 수 있도록 했으며, 그 셀에는 맵핑된 누적 횟수가 입력된다. 필터는 이 임시 맵을 이용하여 설정된 threshold에 따라 실제 글로벌 맵에 맵핑을 할 것인지에 대한 판단을 하게 된다. Threshold는 기본적으로 장착된 LIDAR 센서의 위치와 각도 그리고 지형 종류에 따라 초기 threshold(A)를 가지게 되며, 현재 차량의 속도에 따라 threshold(B)와 heading 변화각에 따라 threshold(C) 값이 실시간 변화되도록 설계하였다.

4. 글로벌 맵의 데이터베이스화를 위한 맵 수정

글로벌 맵 중 데이터베이스로 만들 맵은 실제 지도와 같이 도로 윤곽이 매끄럽고 정확하게 표현되어야 한다. 그러나 LIDAR 센서 데이터가 Point cloud 형태로 들어오기 때문에 앞서 필터링 과정을 거쳐도 그림 17의 좌측 그림과 같이 장애물들의 윤곽이 매끄럽지 못하게 생성된다. 따라서 오프라인을 통해 글로벌 맵을 수정하게 되며, 장애물들의 윤곽을 선으로 매끄럽게 만들고, 윤곽으로만 표현된 장애물 위치를 그 내부까지 확장시켜주는 수정 과정이 이루어진다. 그림 17의 오른쪽 그림은 최종적으로 완성된 글로벌 맵이다.

그림 16. 필터 설계 알고리즘.

Fig. 16. Filter design algorithm.

그림 17. 글로벌 맵 수정 전(a)과 후(b).

Fig. 17. Global map modify, before (a) & after (b).

V. 맵 매칭을 통한 헤딩 보정

1. 헤딩 보정의 필요성

오프라인으로 수정 완료된 글로벌 맵은 데이터베이스화되어 Environment perception Node로 입력된다. 그리고 자율주행을 하는 동안 차량 모델을 통해 얻은 추정된 현재 위치와 헤딩각을 이용해 차량 기준점에서 로컬 맵 크기만큼의 영역을 읽어오게 된다. 그리고 그 맵은 실시간 LIDAR 센서로부터 생성되는 로컬 맵과 매칭을 하게 된다. 만약 차량 모델로부터 계산된 차량의 절대 위치와 헤딩각이 오차가 없다면 글로벌 맵에서 읽어온 맵과 실시간 생성된 로컬 맵의 매칭률은 매우 높을 것이다. 그러나 차량 모델을 통해 추정된 차량 위치와 헤딩은 그 특성상 오차를 포함하게 된다. 오차의 원인은 크게 두 가지로 구분할 수 있으며, 하나는 모델링 설계에 있어서 무시되었던 변수들에 의한 오차와 실시간 센서에 의해 계속된 값들이 가지는 오차에 기인한 것이다. 특히, 헤딩의 경우에는 식(12)에서 알 수 있듯이, 이전 헤딩에 지속적으로 누적된 값이기 때문에 오차 또한 지속적으로 누적될 수밖에 없다. 그림 18은 헤딩 오차의 누적으로 인해 발생한 대표적인 예로써, 동일한 경로를 왕복하였을 때 시작위치와 복귀위치의 헤딩이 일치하지 않는 모습을 확인할 수 있다. 이러한 현상이 발생하는 원인은 차량 모델을 설계함에 있어서 평면을 가정하였기 때문에 그림 18과 같이 급격한 경사를 지날 때 오차가 발생하며, 엔코더를 이용한 오도미터(odometer)의 경우에는 과속 방지턱을 넘을 때와 같이 순간적으로 pitching 현상이 발생하였을 경우에 오차를 발생하기 때문이다.

따라서 본 연구에서는 그림 19와 같이 맵 매칭을 통해서 추정된 헤딩이 얼마나 실제와 벗어나고 있는지를 확인하고, 발생된 헤딩 오차만큼을 실시간 보상에 줄 수 있는 알고리즘을 연구했다[10].

2. 실시간 맵 매칭 기법

맵 매칭을 위해서 가장 먼저 선행되어야 하는 단계는 글로벌 맵에서 현재 차량의 위치와 헤딩에 해당하는 로컬 맵을

그림 18. 헤딩 누적 오차로 인한 잘못된 맵 빌딩.

Fig. 18. Incorrect map building caused by accumulated heading error.

읽어오는 단계이다. 방법은 맵 빌딩을 하던 방식과 유사하다. 글로벌 맵 빌딩의 경우, 로컬 좌표계의 LIDAR 센서 데이터를 위치인식 시스템에서 획득된 절대 좌표계로 변환하고, 마지막으로 글로벌 맵에 맵핑하게 된다. 맵 매칭의 경우, 차량 모델에서 계산된 차량의 절대 위치 및 헤딩을 이용해 글로벌 맵에서 로컬 맵 크기만큼을 읽어오고, 최종적으로 읽어온 맵을 로컬 좌표계로 회전하여 실시간 생성되는 로컬 맵과 동일한 위치 및 방향으로 될 수 있도록 한다.

3. 헤딩 오차 산출

맵 매칭에 있어서 가장 신뢰할 수 있는 정보는 현재 LIDAR 센서로부터 생성된 로컬 맵이다. 위치인식을 통해 글로벌 맵에서 읽어온 맵은 기본적으로 위치인식 시스템의 오차를 포함하고 있기 때문이다. 따라서 로컬 맵을 기준으로 글로벌 맵에서 읽어온 맵을 매칭하게 되면, 랜드마크(Landmark)의 위치를 이용해 추정된 헤딩이 얼마나 오차를

그림 19. 헤딩 오차를 구하기 위한 맵 매칭 과정.

Fig. 19. Map matching process for calculation of heading error.

그림 20. 헤딩 오차 산출.

Fig. 20. Calculation of heading error.

가지고 있는지를 계산할 수가 있다. 본 연구에서는 도로의 형상을 랜드마크로 사용하였다. 만약 직선 도로에서 무인차가 도로의 중앙으로 도로와 나란하게 주행을 하고 있다면 로컬 맵의 중앙 열을 중심으로 도로의 폭만큼 직사각형 모양의 도로 형상이 생성될 것이다. 이 때, 추정 heading이 약 5°정도 오차를 포함하고 있다면, 글로벌 맵에서 읽어들인 맵은 로컬 맵과 비교해서 약 5°정도 기울어진 평행사변형 모양으로 도로 형상이 나타날 것이다. 따라서 두 개의 맵을 매칭하게 되면, 도로의 형상이 서로 일치하지 않음을 알 수 있으며, 더 나아가 그 각의 차이까지 추정할 수 있다.

heading 각의 오차 값을 찾는 방법에 대해 기술하기 전에 heading 오차 값의 특성을 먼저 알아보겠다. 차량 모델을 통해 얻은 heading은 지속적인 오차 누적을 포함하기 때문에, 맵 매칭 이탈 각도 지속적으로 커지게 된다. 즉, 최초 무인 차량이 출발할 때에는 맵 매칭율이 거의 100% 수준이나 주행 거리가 증가할 수록 맵 매칭율도 지속적으로 감소하게 되는 경향을 보이는 특성을 가지고 있다. 따라서 이러한 특성을 이용해 실시간 읽어들인 글로벌 맵을 차량 기준으로 좌, 우 방향으로 일정 각도로 회전을 시켜서 로컬 맵과 매칭을 시켜보면서 가장 매칭율이 높은 각도를 찾게 된다. 좌, 우 방향으로 회전하는 범위는 실험을 통해 거리에 따른 차량 모델을 통해 얻은 heading 오차 값을 산출하여 정하며, 회전하는 각도 분해능은 실험을 통해서 0.5°로 선정했다.

매칭율에 대한 계산은 로컬 맵과 동일한 크기의 임시 맵을 생성하여, 로컬 맵에서 도로 값을 가지고 있는 각 셀을 기준으로 읽어들인 글로벌 맵에 해당되는 셀이 도로이면 '0' 값을 입력하고, 도로가 아니라면 가장 인접한 도로까지의 상대 거리 값을 넣도록 했다. 이 임시 맵을 이용해 표준편차를 구하고, 각 표준편차들을 바탕으로 차량 위치를 기준으로 얼마큼 heading 오차가 발생했는지 추정하게 된다.

VI. 실험 결과

1. 실험 환경

본 연구에 대한 검증 및 실험을 위해 비교적 도로와 그 외

그림 21. 실험 환경.

Fig. 21. Test environment.

곽이 잘 구분되며, 비포장 도로로 직진과 곡선 구간이 혼재된 환경 조건을 가진 자동차 트랙을 이용하였다. 이 트랙의 총 주행 거리는 약 1.4km이며, 도로 폭은 최대 8m에서 최소 4m로 자율주행 실험에 있어서 임의로 장애물을 설치할 수 있는 폭을 가지고 있다. 그림 21은 이 자동차 트랙의 항공 사진으로 적색 선은 주행 경로이며, 시작 위치와 도착 위치는 같으며, 비교적 폭이 넓은 세 지점에 장애물을 설치하였다.

2. 차량 모델 설계의 검증

차량 모델을 통해서 얻은 차량 위치와 heading은 기본적으로 yaw rate r 에 의해서 결정된다는 것을 앞서 본문에서 설명했다. 따라서 차량 모델이 얼마나 잘 설계되었는지를 검증하기 위한 검증 인자로 yaw rate을 사용하였으며, 고가의 INS에서 나오는 yaw rate을 비교 대상으로 사용하였다. 그림 22는 차량 모델에서 계산된 yaw rate과 INS에서 얻은 yaw rate을 비교한 그래프로, 가속도에 의한 영향까지 고려될 수 있도록 속도와 조향을 급격하게 변화시키면서 데이터를 획득했다.

그림 22에서 가장 위에 있는 그래프가 yaw rate을 비교한 그래프로 모든 구간에서 INS에서 얻은 값과 거의 일치하는 모습을 확인할 수 있다. 가장 큰 차이를 보이고 있는 구간은 원으로 표시한 구간으로, 속도 그래프에서 알 수 있듯이 급정지를 한 구간이다. 이 구간에서 유심히 봐야 할 것은 조향은 큰 변화가 없었다는 것이다. 즉, 급정지에 따른 관성에 의해 INS의 yaw rate값이 실제보다 증폭되어 검출되었으며, 차량 모델에서 얻은 값이 더 정확한 결과를 나타내고 있다.

3. 무인 자율주행을 통한 실시간 맵 빌딩 결과

LIDAR 센서를 이용한 맵 빌딩 결과를 바탕으로 차량 모

그림 22. 차량 모델 설계 검증을 위한 yaw rate 결과.

Fig. 22. Result of yaw rate test for verify vehicle model.

그림 23. 글로벌 맵 생성.

Fig. 23. Generated global map.

델과 맵 매칭을 결합한 위치 및 heading 각 추정의 신뢰도를 판단하였다. 우선 RTK-GPS와 INS를 연동한 위치인식 시스템을 이용하여 실험 환경을 맵 빌딩했으며, 그림 23과 같이 오프라인 수정을 통해 최종적인 글로벌 맵을 생성했다. 글로벌 맵은 임의로 놓은 장애물 정보는 포함하고 있지 않으며, 도로 윤곽만을 정확하게 맵핑하였다.

이렇게 생성된 글로벌 맵은 실시간 차량 모델을 통해 얻은 현재 차량의 위치와 heading에 해당하는 위치에서 로컬 맵 크기만큼 읽어오게 되며, LIDAR 센서로부터 생성된 로컬 맵과 맵 매칭을 하게 된다. 그리고 맵 매칭을 통해 추정된 heading의 오차 값을 산출하여 그림 24와 같이 오차 각만큼을 보정하며 목적지까지 무인 자율주행을 수행하였다.

그림 24에서 흰색은 도로이며 녹색은 글로벌 맵, 적색은 LIDAR 센서 데이터로, 좌측 그림은 heading 오차로 도로 윤곽이 일치하지 않는 상황이며, 우측 그림은 오차 보정을 통해 도로 윤곽이 거의 일치하고 있는 모습을 보여주고 있다.

그림 25는 최종적으로 자율주행을 통해 실시간 맵핑된 글로벌 맵이다. 맵핑된 글로벌 맵에서 각 코너의 도로 부분이 검은색으로 생성된 것은, 지면을 스캔하는 LIDAR 센서 데이터가 코너링 중에 발생한 rolling에 의해 정확하게 도로를 분리하지 못해서 발생한 결과이다. 이러한 이유 때문에 실제적인 자율주행에 있어서는 지면을 스캔한 데이터는 사용하지 않았으며, 맵 빌딩에만 이용하였다. 그리고 그림 21과 같이 세 지점의 장애물도 정확하게 맵핑된 것을 확인할 수 있으며, 전체적으로 차량 모델과 맵 매칭을 통해 생성된 글로벌 맵이 RTK-GPS와 INS를 연동하여 맵 빌딩된 글로벌 맵과 거의 일치하고 있음을 확인할 수 있다.

그림 24. 맵 매칭을 통한 오차 보정.

Fig. 24. Error compensation using map matching.

그림 25. 자율주행을 통한 맵 빌딩 결과.

Fig. 25. Result of real-time map building.

VII. 결론

본 연구에서는 차량 모델과 LIDAR 센서를 이용한 맵 매칭 기반의 위치인식 알고리즘을 제시하였으며, 이를 통해서 안정적인 무인 자율주행을 성공하였다. 특히, 높은 나무들에 의한 GPS 음영 지역이 많은 야지 환경이나, 터널 같이 GPS 수신기 불가능한 지역에서도 정확한 글로벌 맵만 보유하고 있다면 자율주행이 가능할 수 있음을 확인하였다. 또한 실험에서는 최초 초기화 과정에서만 GPS 정보를 사용하여 알고리즘을 검증하였지만, 실제적으로는 GPS 및 INS를 이용한 위치인식 시스템과 연동하여 보다 신뢰성이 높은 위치인식 시스템으로 사용할 수 있으며, 더 나아가 저가형 GPS 및 IMU (Inertial Measurement Unit)와 연동하여 현재 실외환경에서 주로 사용되고 있는 고가의 위치인식 시스템을 대체할 수 있는 저가형 시스템 개발도 가능할 것이다. 그리고 LIDAR 센서 외에 카메라를 사용한 비전 시스템을 이용해 차선 및 색상 정보까지 맵 빌딩에 포함시키는 연구를 현재 진행하고 있으며, 이 연구가 성공한다면 더 다양한 환경에서도 신뢰성 높은 자율주행이 가능해 질 것이다. 맵 매칭 경우에도 본 논문에서는 heading 오차 산출에 적용한 내용만 기술하였지만, 실제적으로는 경로 생성 알고리즘에도 적용하여 자율주행을 수행하였으며, 이 외에도 이동 장애물 tracking 등 다양한 부분에 있어서 적용 가능하다. 따라서, 본 연구는 실외환경에서 강인한 자율주행이 가능할 수 있는 기법을 맵 매칭 기법에서 찾았으며, 이 기법을 차량 모델을 이용한 저가형 위치인식 기술에 적용하여 그 가능성을 확인하였다.

참고문헌

- [1] C. Crane, D. Armstrong, A. Arroyo, A. Baker, D. Dankel, G. Garcia, N. Johnson, J. Lee, S. Ridgeway, E. Schwartz, E. Thorn, S. Velat, and J. H. Yoon, "Team gator nation's autonomous vehicle development for the 2007 DARPA urban challenge," *Journal of Aerospace computing, information, and communication*, vol. 4, no. 12, pp. 1058-1085, Dec. 2007.
- [2] C. Urmson, J. Anhalt, D. Bagnell, C. Baker, R. Bittner, M. N. Clark, J. Dolan, D. Duggins, T. Galatali, C. Geyer, M. Gittleman, S. Harbaugh, M. Herbert, T. M. Howard, S. Kolski, A. Kelly, M. Likhachev, M. McNaughton, N. Miller, K. Peterson, B. Pilnick, R. Rajkumar, P. Rybski, B. Salesky, Y. W. Seo, S. Singh, J. Snider, A. Stentz, W. Whittaker, Z. Wolkowicki, and J. Ziegler,

- “Autonomous driving in urban environments: boss and the urban challenge,” *Journal of Field Robotics Special Issue on the 2007 DARPA Urban Challenge, Part 1*, vol. 25, no. 8, pp. 425-466, June 2008.
- [3] D. Ferguson and A. Stentz, “Field D*: An interpolation-based path planner and replanner,” *Proc. of the International Symposium on Robotics Research*, vol. 28, pp. 1926-1931, Oct 2005.
- [4] H. Najjaran and A. Goldenberg, “Real-time motion planning of an autonomous mobile manipulator using a fuzzy adaptive Kalman filter,” *Robotics and Autonomous Systems*, vol. 55, no. 2, pp. 96-106, Feb. 2007.
- [5] J. Lalonde, N. Vandapel, D. Huber, and M. Hebert, “Natural terrain classification using three -dimensional ladar data for ground robot mobility,” *Journal of Field Robotics*, vol. 23, no. 10, pp. 839-861, June 2006.
- [6] Joint Architecture for Unmanned System (JAUS) Reference Architecture, Version 3.0.
- [7] U. Kiencke and L. Nielsen, *Automotive Control Systems, For Engine, Driveline, and Vehicle*, 2nd Ed., Springer, 2004.
- [8] S. Lee, S. Lee, and D. Kim, “Recursive unscented kalman filtering based slam using a large number of noisy observations,” *International Journal of Control, Automation, and Systems*, vol. 4, no. 6, pp. 736-747, Dec. 2006.
- [9] J. Lee, S. Lee, J. Kim, and C. Han, “The development of a map building algorithm using LADAR for unmanned ground vehicle,” *Journal of Institute of Control, Robotics and Systems(in Korean)*, vol. 15, no. 12, pp. 1246-1253, Dec. 2009.
- [10] S. Park, M. Park, and S. Park, “Object-spatial layout-route based hybrid map and global localization for mobile robots,” *International Journal of Control, Automation, and Systems*, vol. 7, no. 4, pp. 598-614, Aug. 2009.

박재웅

2005년 인천기능대학교 자동차과 졸업.
2008년 국민대학교 산업기술대학원 자동차공학과 공학석사. 2008년~현재 국민대학교 자동차공학전문대학원 박사과정 재학중. 관심분야는 무인자율주행차량의 차량시스템 설계, 차량제어.

김재환

2007년 국민대학교 자동차공학과 졸업.
2008년 국민대학교 자동차공학전문대학원 전자제어시스템전공 석사. 2008년~현재 동 대학원 박사과정 재학중. 관심분야는 무인자율주행차량, 환경인식.

김정하

1981년 성균관대학교 기계공학과 졸업.
1986년 University of Cincinnati 공학석사.
1990년 University Pennsylvania 공학박사.
현재 국민대학교 자동차공학과 교수.
관심분야는 차량전자제어시스템, 무인차량.