

# 전시장 도슨트 로봇의 경로탐색을 위한 위치추적 보정 알고리즘

정 무 경\*, 이 동 명<sup>o</sup>

## Location Tracking Compensation Algorithm for Route Searching of Docent Robot in Exhibition Hall

Moo Kyung Jung\* Dong Myung Lee<sup>o</sup>

### 요 약

본 논문에서는 전시장에 사용되는 도슨트 로봇 (Docent Robot)의 자율주행 정밀도 향상을 위하여 최소자승법을 적용한 위치추적 보정 알고리즘 (Location tracking Compensation Algorithm based on *Least-Squares Method*,  $LCA_{LSM}$ )을 제안하고, 도슨트 로봇을 사용한 실험을 통하여 그 성능을 분석하였다. 제안한  $LCA_{LSM}$ 은 전시장에서 도슨트 로봇의 자율주행에서 엔코더/자이로 (encoder/gyro,  $E/G$ )에서 발생하는 누적오차를 줄이고 위치추적 정확도를 개선하기 위하여 수집된 로봇의 위치좌표를 최소자승법 (Least-Squares Method, LSM)에 적용하여 보정한다. 실험결과, 제안한  $LCA_{LSM}$ 의 위치추적 평균 오차 감소율은 시나리오 1 (S1) 및 시나리오 2 (S2)에서  $LCA_{KF}$  (Location tracking Compensation Algorithm based on Kalman Filter,  $LCA_{KF}$ ) 보다 4.85% 더 높음을 확인하였다. 또한, 제안한  $LCA_{LSM}$ 의 측정오차에 따른 표준 편차는 S1 및 S2에서  $E/G$ 와  $LCA_{KF}$ 에 비해 훨씬 낮을 뿐 아니라 균일함을 확인하였다. 따라서 제안한  $LCA_{LSM}$ 은 도슨트 로봇이 S1 및 S2의 직선 이동을 할 때  $E/G$  및  $LCA_{KF}$  보다 더 안정적인임을 알 수 있다.

**Key Words** : Least-Squares Method, Kalman Filter, Docent Robot, Location Tracking, Encoder/Gyro

### ABSTRACT

In this paper, a location tracking compensation algorithm based on the Least-Squares Method ( $LCA_{LSM}$ ) was proposed to improve the autonomous tracking efficiency for the docent robot in exhibition hall, and the performance of the  $LCA_{LSM}$  is analyzed by several practical experiments. The proposed  $LCA_{LSM}$  compensates the collected location coordinates for the robot using the *Least-Squares Method (LSM)* in order to reduce the cumulated errors that occur in the Encoder/Gyro sensor ( $E/G$ ) and to enhance the measured tracking accuracy rates in the autonomous tracking of the robot in exhibition hall. By experiments, it was confirmed that the average error reduction rates of the  $LCA_{LSM}$  are higher as 4.85% than that of the  $LCA_{KF}$  in Scenario 1 (S1) and Scenario 2 (S2), respectively on the location tracking. In addition, it was also confirmed that the standard deviation in the measured errors of the  $LCA_{LSM}$  are much more low and constant compared to that of the  $E/G$  sensor and the  $LCA_{KF}$  in S1 and S2 respectively. Finally, we see that the suggested  $LCA_{LSM}$  can execute more the stabilized location tracking than the  $E/G$  sensors and the  $LCA_{KF}$  on the straight lines of S1 and S2 for the docent robot.

\* 이 논문은 2013학년도 동명대학교 연구년 지원에 의하여 연구되었음

• First Author : Tongmyong University, Department of Computer Engineering, evangelionxg@naver.com, 학생회원

o Corresponding Author : Tongmyong University, Department of Computer Engineering, dmlee@tu.ac.kr, 종신회원

논문번호 : KICS2014-06-235, Received June 12, 2014; Revised March 31, 2015; Accepted April 10, 2015

## I. 서론

최근 세계적으로 로봇 시장의 기술 경쟁이 더욱 더 가열됨에 따라, 이동 로봇에 관한 연구가 활발히 진행되고 있다. 이동 로봇은 기구학적 특성에 따라 관절형태의 보행 로봇과 바퀴형태의 주행 로봇으로 나눌 수 있다. 특히 이러한 이동 로봇 분야에서 자율주행 기술의 선점은 경제적 또는 산업적 측면에서 국익에 매우 중요하다<sup>1,2)</sup>.

이동 로봇의 자율주행 기술들은 위치 추적 (localization), 지도생성 (map-building), 경로계획 (path-planning) 기술 등으로 분류되고, 그 중에서도 로봇이 자율주행 중에 위치를 추정하는 위치 추적기술은 로봇의 핵심 기술로 가장 활발히 연구되고 있다. 이 연구들은 주로 전역 위치추정 센서와 지역 위치추정 센서의 융합을 통해 연구가 진행되었으며, 최근에는 위치추적과 지도 생성기술을 함께 수행하는 Simultaneous Localization And Mapping (SLAM)에 대한 연구가 활발히 진행되고 있다<sup>3-5)</sup>.

위치추적에는 일반적으로 이동 로봇의 이동거리 측정을 위하여 엔코더 (encoder)와 이동관성을 측정할 수 있는 가속계 (accelerometer), 회전 관성을 측정할 수 있는 자이로 센서 (gyro sensor), 방위각을 측정할 수 있는 자계 센서인 전자 나침반 (electronic compass)이 각각 이용되는데, 계측방식에 따라 장·단점이 있다.

엔코더는 정밀한 계측이 가능하지만 바닥과 자율주행 장치의 바퀴 구동부 특성 등에 따라 슬립 (slip)에 의한 오차가 발생하고, 가속계 센서는 중력에 의한 가속도를 측정 하므로 정밀도는 높으나 이동 중에는 계측이 불가능하다. 또한 자이로 센서는 측정하는 회전축 외에 다른 축이 회전을 하여도 측정하는 회전축에 영향을 받지 않는 특징이 있어 설치가 용이하나 드리프트 (drift) 현상으로 인하여 바이어스 (bias)와 비정렬 오차 (misalignment error)가 발생한다. 자기 센서인 전자 나침반은 자율주행 장치의 구동부에서의 자기장으로 인하여 오차가 발생한다<sup>6,7)</sup>.

본 논문에서는 전시장에서 도슨트 로봇이 자율주행을 할 경우, 3D 깊이센서를 이용하여 위치추적 누적 오차를 줄이고 이동 궤적의 안정화 및 위치추적의 정밀도를 개선한 위치추적 보정 알고리즘을 제안하고 실제 실험을 통하여 성능을 분석하였다.

## II. 관련 연구

[8]에서는 이동 로봇이 전진 방향으로 이동 할 때 미끄러짐 현상으로 발생하는 방향각 오차의 문제를 자이로 센서를 사용하여 오차를 보정함으로써 경로추적의 성능을 개선하였다. 또한 이동 로봇의 주 바퀴와 보조 바퀴의 회전축이 90°를 이루는 특성과 자이로 센서로 회전 각속도를 이용하여 주행 시 발생하는 미끄러짐에 대한 오차를 보상하였다.

[9]에서는 박물관이나 미술 전시관 등에 사용되는 도슨트 로봇의 개발하였다. 특히 도슨트 로봇의 위치를 추적 할 경우, 기존의 엔코더를 사용하면 이동거리가 짧을 때는 오차의 발생이 적으나 이동거리가 길어지면 누적오차가 계속 커지는 문제가 발생한다. 이러한 문제점을 해결하기 위해서 [10]에서는 도슨트 로봇이 자계 센서의 도움을 받아 마그네틱 와이어로 구성된 경로를 따라 이동 할 때, RFID 리더 (reader)를 사용하여 이동경로에 설치된 태그 (tag)를 인식하게 함으로써 누적오차를 감소시켰다.

[11]에서는 랜드마크를 이용하여 위치추적을 시도 하였는데, 그림 1과 같이 천정 (ceiling) 전역지도를 바탕으로 랜드마크의 인식오차를 제거하고, 칼만필터 (Kalman Filter)를 사용하여 이동 로봇의 흔들림 및 진동 등에 의한 측정잡음을 제거하였다.

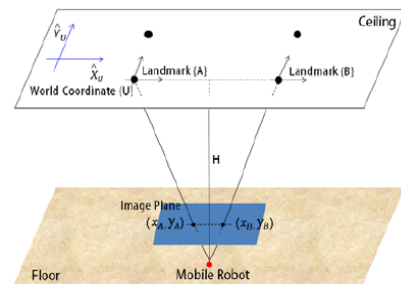


그림 1. 랜드마크를 이용한 위치추적 시스템  
Fig. 1. Location tracking system based on land mark.

## III. 도슨트 로봇의 구성

### 3.1 도슨트 로봇 플랫폼

도슨트 로봇의 플랫폼은 그림 2와 같이 로봇의 표정을 표시하는 LED와 무선 네트워크 연결을 위한 무선LAN, 로봇 주행을 관장하는 구동부, 사용자의 행동패턴을 인지하기 위한 3D 깊이센서 그리고 도슨트 로봇의 운영체제인 ROS (Robot Operating System)의

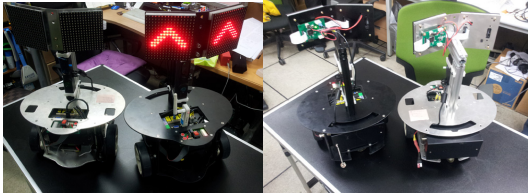


그림 2. 도슨트 로봇 플랫폼  
Fig. 2. Docent robot platform.

구동용 임베디드 모듈 등으로 구성되어 있다. 도슨트 로봇 플랫폼에는 기본 운영체제인 Ubuntu 12.04, 로봇 운영체제인 ROS Hydro 버전, 그리고 장치 드라이버인 OpenNI2, PrimeSense Driver가 설치되어 있다.

### 3.2 엔코더/자이로

ROS기반 도슨트 로봇의 플랫폼은 이동거리 측정을 위해서 기본적으로 엔코더/자이로 센서 (encoder/gyro sensor, E/G)를 사용하였다. 엔코더는 로봇 구동용 초소형 DC 서보모터를 구동하기 위한 고성능 실시간 제어기로 RS232, RS485, Controller Area Network (CAN) 등 다양한 통신 방식을 지원하며, 실시간으로 위치정보 및 속도 정보의 수집이 가능하다.

본 논문에서는 도슨트 로봇의 엔코더로 그림 3의 로보큐브테크의 CUBE-DC2402-DID를, 그리고 자이로 센서로 그림 3과 같이 완전 독립적인 디지털 자이로스코프인 Minfinity사의 CruzCore R1050K를 각각 사용하였다.

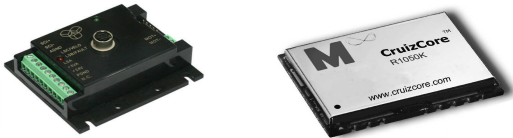


그림 3. 도슨트 로봇 플랫폼에 사용된 엔코더/자이로  
Fig. 3. Encoder/gyro used in docent robot platform.

### 3.3 위치추적 거리 측정 및 방향각 계산

E/G의 출력으로부터 이동 거리 및 방향각을 계산하는 과정은 다음 식 (1) ~ (4)와 같다. 여기에서  $M$ 은 이동거리를,  $V$ 는 속도,  $\theta$ 는 방향각,  $t_0$ 는 초기시간을 각각 나타낸다. 식 (1) 및 식 (2)에서 아래첨자  $r$ 과  $l$ 은 오른쪽과 왼쪽 바퀴의 이동거리를 각각 나타낸다. 식 (3)은 도슨트 로봇이 중심에서 부터 이동한 이동거리를, 식 (4)는 도슨트 로봇의 방향각계산을 각각 나타낸다.

$$M_r(t) = M_r(t_0) + \int_{t_0}^t V_r dt \quad (1)$$

$$M_l(t) = M_l(t_0) + \int_{t_0}^t V_l dt \quad (2)$$

$$M(t) = \frac{M_r(t) + M_l(t)}{2} \quad (3)$$

$$\theta(t) = \theta(t_0) + \int_{t_0}^t \left( \frac{V_l - V_r}{D} \right) dt \quad (4)$$

도슨트 로봇의 위치좌표  $(x, y)$ 는 식 (4)를 이용하여 식 (5)와 식 (6)으로 계산할 수 있다.

$$X(t) = X(t_0) + \int_{t_0}^t V(t) \cdot \sin\theta(t) dt \quad (5)$$

$$Y(t) = Y(t_0) + \int_{t_0}^t V(t) \cdot \cos\theta(t) dt \quad (6)$$

한편 자이로 센서의 출력인 각속도  $\phi$ 는 식 (7)과 같다. 식 (7)에서  $S$ 는 환산계수를,  $v$ 는 출력전압,  $B$ 는 바이어스, 아래첨자  $g$ 는 자이로를 각각 나타낸다.

$$\phi(t) = S_g(v_g(t) - B_g) \quad (7)$$

## IV. 위치추적 보정 알고리즘

### 4.1 최소자승법을 이용한 위치계산

본 논문에서는 E/G 센서로부터 추출한 이동 거리 및 방향각을 근간으로 최소자승법 (Least-Squares Method, LSM)을 이용하여 위치추적 정확도를 개선하고자 한다. 즉, 자이로 센서로부터 얻어지는 방향각  $\theta(t)$ 를 식 (5) 및 식 (6)에 적용하여 얻어지는  $(x, y)$ 를 도슨트 로봇의 위치좌표로 설정한 후, 이를 LSM으로 위치좌표를 보정한다.

LSM은 위치좌표 데이터 집합에 대한 최적의 좌표 근사를 얻는 방법이다.  $n$ 개의 위치좌표 데이터 점들을  $(x_1, y_1), (x_2, y_2), \dots, (x_n, y_n)$ 라 하고,  $x_n$ 과  $y_n$ 를 각각 독립변수와 종속변수로 두면 위치좌표 방정식은 식 (8)과 같이 나타낼 수 있다.  $\alpha$ 와  $\beta$ 는 결정상수를 의미한다.

$$y = \alpha + \beta x \quad (8)$$

그런데 식 (8)에서 위치좌표 데이터 점들  $(x_i, y_i)$  들 사이에는 오차가 존재 할 것이므로 데이터 점들에 대한 오차  $\epsilon_i$ 는 식 (9)와 같이 표현된다.

$$\epsilon_i = y_i - \alpha - \beta x_i \quad (9)$$

식 (9)의 1차 방정식을 근사하는 가장 보편적인 방법은 *LSM*을 이용하는 것이며, 그 방법은 다음과 같다. 먼저 오차의 제곱의 합에 대한 식은 식 (10)과 같이 나타 낼 수 있다.

$$Z = \sum_{i=1}^n \epsilon_i^2 = \sum_{i=1}^n (y_i - \alpha - \beta x_i)^2 \quad (10)$$

식 (10)에서 매개변수  $\alpha$ 와  $\beta$ 에 대해  $Z$ 를 최소화하여 야 하는데, 그 방법은 식 (11) 및 (12) 같이  $\alpha$ 와  $\beta$ 에 대해  $Z$ 를 미분한 후에 0으로 초기화 하는 것이다.

$$\frac{\delta Z}{\delta \alpha} = -2 \sum_{i=1}^n (y_i - \alpha - \beta x_i) = 0 \quad (11)$$

$$\frac{\delta Z}{\delta \beta} = -2 \sum_{i=1}^n (y_i - \alpha - \beta x_i) x_i = 0 \quad (12)$$

그 다음, 식 (11)과 식 (12)는  $\sum_{i=1}^n \alpha = n\alpha$  로  $\alpha$ 와  $\beta$ 에 대한 2개의 1차 방정식으로 표현할 수 있으며, 이는 *Cramer* 규칙을 이용하여 식 (13) 및 식 (14)와 같이 미지수  $\alpha$ 와  $\beta$ 를 나타낼 수 있다.

$$\alpha = \frac{\begin{vmatrix} \sum_{i=1}^n y_i & \sum_{i=1}^n x_i \\ \sum_{i=1}^n x_i y_i & \sum_{i=1}^n x_i^2 \end{vmatrix}}{\begin{vmatrix} n & \sum_{i=1}^n x_i \\ \sum_{i=1}^n x_i & \sum_{i=1}^n x_i^2 \end{vmatrix}} = \frac{\sum_{i=1}^n y_i \sum_{i=1}^n x_i^2 - \sum_{i=1}^n y_i \sum_{i=1}^n x_i y_i}{n \sum_{i=1}^n x_i^2 - \left( \sum_{i=1}^n x_i \right)^2} \quad (13)$$

$$\beta = \frac{\begin{vmatrix} n & \sum_{i=1}^n y_i \\ \sum_{i=1}^n x_i & \sum_{i=1}^n x_i y_i \end{vmatrix}}{\begin{vmatrix} n & \sum_{i=1}^n x_i \\ \sum_{i=1}^n x_i & \sum_{i=1}^n x_i^2 \end{vmatrix}} = \frac{n \sum_{i=1}^n x_i y_i - \sum_{i=1}^n x_i \sum_{i=1}^n y_i}{n \sum_{i=1}^n x_i^2 - \left( \sum_{i=1}^n x_i \right)^2} \quad (14)$$

마지막으로 식 (13)과 식 (14)를 이용하여 구해지는  $\alpha$ 와  $\beta$ 를 식 (8)에 대입한 후, 위치좌표  $(x_n, y_n)$ 를 적용하면 보정된 위치좌표를 얻을 수 있다<sup>[12-13]</sup>.

### 4.2 위치추적 보정 알고리즘

*LSM*을 이용한 위치추적 보정 알고리즘 (*Location tracking Compensation Algorithm based on Least-Squares Method, LCA<sub>LSM</sub>*)의 위치계산 흐름도는 그림 4와 같다.

1단계에서는 도슨트 로봇에 장착되어 있는 *E/G* 센서를 사용하여 이동거리 및 방향각을 구한다. 2단계에서는 1단계에서 구해진 이동거리와 방향각을 통하여 도슨트 로봇의 위치좌표를 구한다. 3단계에서는 먼저 2단계에서 얻어진 위치좌표를 *LSM*에 적용하기 위하여  $n$ 개의 위치좌표를 저장한 다음 저장된 도슨트 로봇의 위치좌표를 *LSM*에 적용하여  $n$ 개의 위치좌표를 보정한다. 즉,  $i \sim (i+k)$  범위에 있는  $k$ 개의 위치좌표를 먼저 보정 한 다음,  $(i+1) \sim ((i+k)+1)$ ,  $(i+2) \sim ((i+k)+2), \dots, (i+n) \sim ((i+k)+n)$ 의 순서대로 위치좌표의 범위를 차례대로 옮겨가면서 위치좌표를 보정한다. 여기에서  $k$ 는 10으로 설정하였다. 마지막 4단계에서는 *LSM*으로 보정된 도슨트 로봇의 최종 위치좌표를 출력한다.

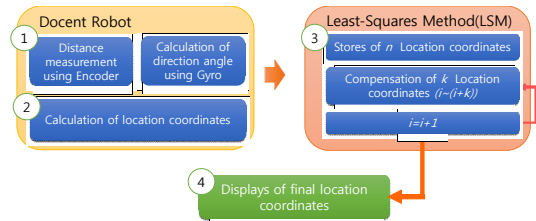


그림 4. 제안하는 *LCA<sub>LSM</sub>*의 위치계산 흐름도  
Fig. 4. Location calculation flows of proposed *LCA<sub>LSM</sub>*.

## V. 실험 및 결과 분석

### 5.1 실험 환경 및 방법

본 논문에서는 제안한 *LCA<sub>LSM</sub>*의 위치추적 정확도 실험은 그림 5와 같이 교내 복도에서 실시되었다. 실험 시나리오는 그림 6과 같이 8m x 10m의 공간에서 도슨트 로봇이 (0, 3)에서 (8, 3)으로 이동하는 시나리오 1(S1)과 (0, 6)에서 (8, 6)으로 이동하는 시나리오 2(S2)로 각각 구성되고, 각 시나리오별 5회 실험을 수행하였다.



그림 5. 실험환경  
Fig. 5. Experimental environment.

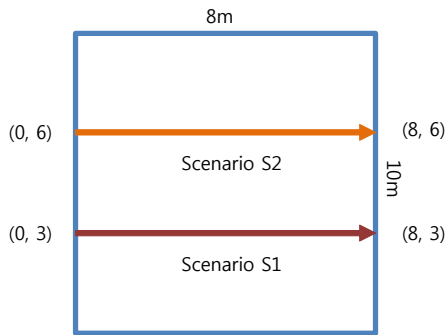


그림 6. 실험 시나리오 S1 및 S2  
Fig. 6. Scenario S1 and S2.

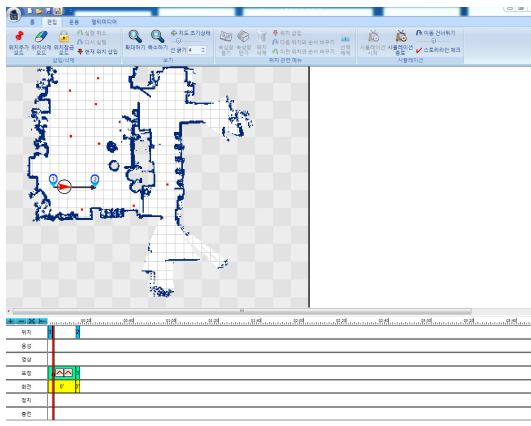


그림 7. 스토리라인 에디터  
Fig. 7. Storyline editor.

위치추적 정확도 실험에서 그림 7과 같은 스토리라인 에디터 (StoryLine Editor)를 사용하여 시나리오별 도슨트 로봇의 위치, 이동로봇의 시작점과 도착점을 설정 한 후, 실제 도슨트 로봇이 이동 할 때의 위치좌표와 발생하는 오차 값을 확인하였다.

## 5.2 실험 결과 및 분석

상기의 실험 시나리오 S1과 S2에 대해 5회 실험 중 각 2회에 대해 제안한  $LCA_{LSM}$ 의 위치좌표 보정 성능을 그림 8 및 그림 9에서  $E/G$  및 칼만필터를 적용한 알고리즘 (Location tracking Compensation Algorithm based on Kalman Filter,  $LCA_{KF}$ )과 비교하였다.

S1의 경우,  $E/G$ ,  $LCA_{KF}$  및 제안한  $LCA_{LSM}$ 의 최대 오차는 표 1과 같이 각각 평균 13.44cm,

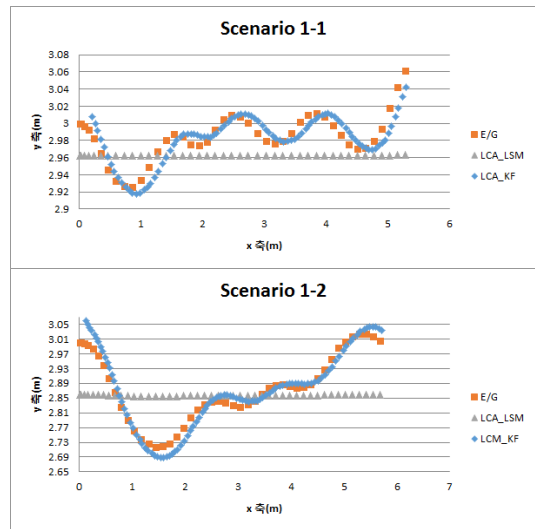


그림 8. S1에 대한 각 알고리즘의 위치좌표 보정 성능 비교  
Fig. 8. Comparisons with compensation performance for S1.

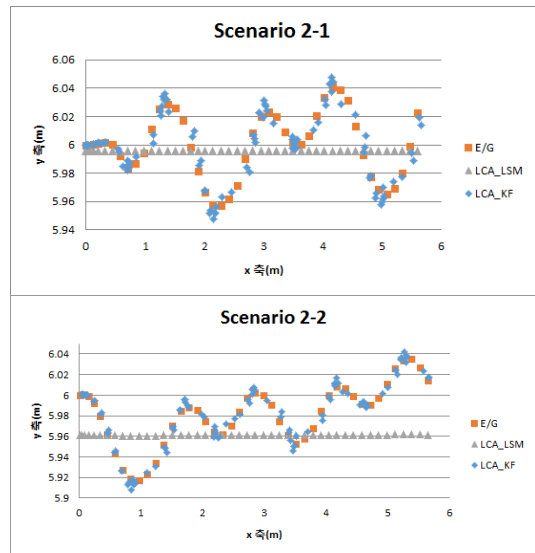


그림 9. S2에 대한 각 알고리즘의 위치좌표 보정 성능 비교  
Fig. 9. Comparisons with compensation performance for S2

13.89cm, 5.35cm으로 나타났다. 그 결과, 제안한  $LCA_{LSM}$ 의 최대 오차는 S1의 실험에서  $E/G$  경우보다 60.19% 감소됨을 확인하였다.

S2의 경우도  $E/G$ ,  $LCA_{KF}$  및 제안한  $LCA_{LSM}$ 의 최대 오차는 표 2와 같이 각각 평균 7.38cm, 7.33cm, 2.30cm으로 나타났다. 그 결과, 제안한  $LCA_{LSM}$ 의 최대 오차는 S2의 실험에서  $E/G$  경우보다 68.83% 감소됨을 확인하였다.

상기의 S1 및 S2의 실험에서 제안한  $LCA_{LSM}$ 은 엔코더/자이로와  $LCA_{KF}$  보다 최대 오차가 큰 쪽으로 감소하고, 가장 적은 오차 값을 가짐을 확인하였다. 그러나  $LCA_{KF}$ 는 S1 및 S2에서 최대 오차가  $E/G$ 와 비슷한 13.89cm, 13.89cm로 전혀 개선되지 않았다.

S1에 대한 위치추적 평균 오차의 비교는 표 3과 같다. 평균 오차는  $E/G$ 는 최대 14.29cm와 최소 0.91cm를,  $LCA_{KF}$ 는 최대 12.83cm와 최소 0.21cm

표 1. S1에 대한 각 알고리즘의 최대 오차 비교  
Table 1. Comparisons of maximum error for S1.

S1	Maximum error(cm)		
	$E/G$	$LCA_{KF}$	$LCA_{LSM}$
S1-1	7.47	7.41	<b>3.80</b>
S1-2	28.19	31.10	<b>14.54</b>
S1-3	15.15	16.26	<b>6.06</b>
S1-4	6.74	6.3	<b>1.38</b>
S1-5	9.68	8.38	<b>0.97</b>
평균	13.44	13.89	<b>5.35</b>

표 2. S2에 대한 각 알고리즘의 최대 오차 비교  
Table 2. Comparisons of maximum error for S2.

S2	Maximum error(cm)		
	$E/G$	$LCA_{KF}$	$LCA_{LSM}$
S2-1	4.30	4.21	<b>0.51</b>
S2-2	8.31	8.10	<b>3.87</b>
S2-3	6.52	6.44	<b>2.33</b>
S2-4	9.63	9.23	<b>1.93</b>
S2-5	8.14	8.71	<b>2.86</b>
평균	7.38	7.33	<b>2.30</b>

표 3. S1에 대한 각 알고리즘의 평균 오차 비교  
Table 3. Comparisons of average error for S1.

실험 횟수	Average error(cm)			Reduction ratio(%)	
	$E/G$	$LCA_{KF}$	$LCA_{LSM}$	$LCA_{KF}$	$LCA_{LSM}$
1	3.77	1.80	<b>1.50</b>	52.25	<b>60.22</b>
2	14.29	12.83	<b>12.21</b>	10.21	<b>14.55</b>
3	9.94	8.49	<b>6.01</b>	14.58	<b>39.53</b>
4	1.37	1.38	<b>1.34</b>	0.58	<b>2.32</b>
5	0.91	0.21	<b>0.18</b>	76.66	<b>80.00</b>

표 4. S2에 대한 각 알고리즘의 평균 오차 비교  
Table 4. Comparisons of average error for S2.

실험 횟수	Average error(cm)			Reduction ratio(%)	
	$E/G$	$LCA_{KF}$	$LCA_{LSM}$	$LCA_{KF}$	$LCA_{LSM}$
1	0.94	0.67	<b>0.54</b>	28.72	<b>42.55</b>
2	3.89	1.80	<b>1.76</b>	53.72	<b>54.75</b>
3	2.67	2.58	<b>2.31</b>	3.37	<b>13.48</b>
4	1.93	0.79	<b>0.75</b>	59.06	<b>61.13</b>
5	4.01	4.06	<b>2.85</b>	1.24	<b>28.92</b>

를, 그리고 제안한  $LCA_{LSM}$ 은 최대 12.21cm와 최소 0.18cm로 각각 나타났다. 따라서 오차 감소율은  $LCA_{KF}$ 의 경우 최대 76.66%와 최소 0.58%를, 제안한  $LCA_{LSM}$ 은 최대 80%와 최소 2.32%로 각각 확인되었다. 이는 제안한  $LCA_{LSM}$ 은 S1에서  $LCA_{KF}$  보다 평균 2.54% 더 오차가 감소된 것이다.

S2에 대한 위치추적 평균 오차는 표 4에서 보는 바와 같이  $E/G$ 는 최대 4.01cm와 최소 0.94cm를,  $LCA_{KF}$ 는 최대 4.06cm와 최소 0.67cm를, 그리고 제안한  $LCA_{LSM}$ 은 최대 2.85cm와 최소 0.54cm로 각각 나타났다. 따라서 오차 감소율은  $LCA_{KF}$ 의 경우 최대 59.06%와 최소 1.24%를, 제안한  $LCA_{LSM}$ 은 최대 61.13%와 최소 13.48%로 각각 확인되었다. 이는 제안한  $LCA_{LSM}$ 은 S2에서  $LCA_{KF}$  보다 평균 7.16% 더 오차가 감소가 된 것이다.

S1 및 S2에서 위치추적 평균 오차의 결과를 볼 때 도슨트 로봇이 주어진 경로를 지속적으로 주행 경우  $E/G$ 만을 사용하면 누적오차가 계속 발생하기 때문에 이에 대한 보정이 필요함을 알 수 있다. 또한, S1 및 S2 실험에서  $LCA_{LSM}$ 은  $LCA_{KF}$  보다 평균 4.85% 오차 감소가 된 것임을 알 수 있다.

다음의 그림 10은 시나리오별 실험에 따른 측정오차의 표준 편차를 분석한 결과이다. 그림 10에서 보는 바와 같이 제안한  $LCA_{LSM}$ 은  $E/G$  비해 훨씬 낮은 표준 편차를 가짐을 알 수 있는데, 이는 도슨트 로봇이 직선 이동을 할 때 보다 더 안정적이고 균일한 위치추적을 할 수 있음을 의미한다.  $LCA_{KF}$ 의 경우, 평균 오차는  $E/G$ 에 비해 상당히 오차개선은 이루어졌지만 표준 편차에서는  $E/G$ 와 비슷한 값을 가지므로 개선이 이루어지지 않음을 알 수 있었다.

제안한  $LCA_{LSM}$ 의 보정성능을 최대 오차, 평균 오차 및 표준 편차 관점에서  $E/G$ 와  $LCA_{KF}$ 와 서로 비교분석 해 보면 다음과 같다.

$LCA_{KF}$ 를 적용한 위치추적의 경우 평균 오차는  $E/G$  보다 감소하는 모습을 볼 수 있으나 표준 편차는  $E/G$ 의 실험결과와 비슷함을 알 수 있다. 그 이유는  $LCA_{KF}$

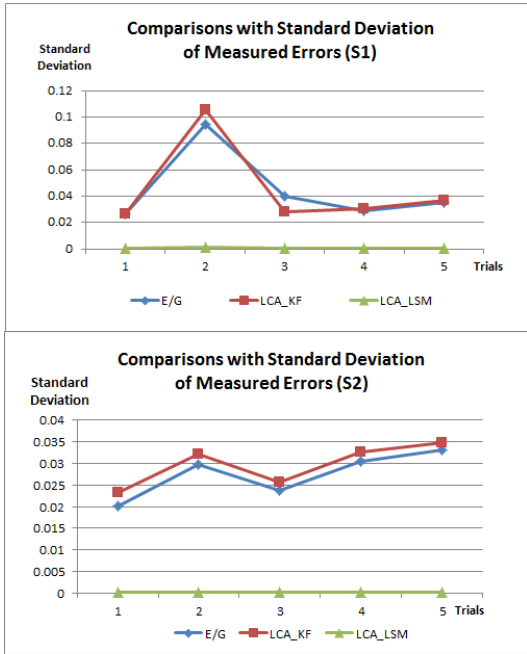


그림 10. 측정오차의 표준 편차 비교 (S1, S2)  
 Fig. 10. Comparisons with standard deviation of measured errors of S1 and S2.

는 보정하기 전의 위치 값 오차를 사용해서 보정을 하기 때문에 결과적으로 도슨트 로봇이 이동경로를 따라 이동하는 경우 보정 전의 결과와 유사한 것으로 판단된다. 즉,  $LCA_{KF}$ 가  $E/G$ 의 경우보다 평균 오차를 감소시키지만 안정적인 이동경로를 추적하는 것은 어렵다. 그러나 제안한  $LCA_{LSM}$ 은, 평균 오차의 감소뿐만 아니라 표준 편차도 큰 폭으로 감소한 것을 알 수 있다. 그 이유는  $LCA_{LSM}$ 은 위치좌표 데이터 수집 과정에서 도슨트 로봇이 시나리오에 따라 경로를 이동 할 때 계속적으로 위치좌표 데이터를 수집하기 때문에  $LCA_{KF}$ 보다 안정적인 위치추적이 가능한 것으로 판단된다. 아울러  $LCA_{LSM}$ 에서 실시간적인 위치추적 기능을 추가하기 위해서는 위치좌표 데이터를 실시간적으로 수집해야 하고 이를 위한 처리시간이 더 소요된다. 그러나 본 논문에서 도슨트 로봇의 이동경로를 추적하는 경우는 실시간성 보다 신뢰도와 안정성이 더 필요하기 때문에  $LCA_{KF}$ 보다 더 합리적이라고 판단된다.

## VI. 결 론

본 논문에서는 전시장에서  $E/G$ 로 자율주행을 실시하는 도슨트 로봇의 누적 오차를 줄이고 이동 궤적의

안정화 및 위치추적의 정밀도를 개선한  $LCA_{LSM}$ 을 제안하고, 도슨트 로봇에서 실제 실험을 통하여 성능을 분석하였다. 실험은 실내 전시장과 비슷한 환경의 대학교 교내 복도에서 가로 세로 8m x 10m의 공간에서 두 개의 시나리오로 각 5회씩 총 10번 실시되었다. 도슨트 로봇의 이동 시나리오는 로봇을 각각 (0, 3)에서 (8, 3)으로, (0, 6)에서 (8, 6)으로 이동하도록 설정하였고, 이 설정환경에서 위치좌표를 수집하였다.

실험결과, 첫째 제안한  $LCA_{LSM}$ 의 위치추적 평균 오차 감소율은 S1 및 S2에서  $LCA_{KF}$ 보다 4.85% 더 높음을 확인하였다. 둘째, 제안한  $LCA_{LSM}$ 은 시나리오 별 실험에 따른 측정오차의 표준 편차에서  $E/G$ 에 비해 훨씬 낮고 균일한 표준 편차를 가져옴을 확인하였으며, 이는 도슨트 로봇이 직선 이동을 할 때 보다 더 안정적인 위치추적을 할 수 있음을 의미한다.

향후, 도슨트 로봇의 이동범위를 보다 더 확대한 실험과, 방향 전환, 곡선 등을 포함한 불규칙 이동 경로에서의 실험이 추가적으로 필요하다고 생각된다. 아울러 도슨트 로봇에 장착된  $E/G$ 의 하드웨어에서 발생하는 누적오차를 줄이기 위한 연구도 시도 할 계획이다.

## References

- [1] J. Kim, J. Do, and S. Kim, "Error minimization of angular velocity using encoders and gyro," *J. Korean Inst. Intell. Syst. (JKIIS)*, vol. 20, no. 6, pp. 814-819, Dec. 2010.
- [2] S. Pack and D. Hwang, "Kinematic structure based DR (encoder, gyro)/vision integrated navigation system for mobile robots," in *Proc. Inst. Control, Robotics and Syst. (ICROS)*, vol. 38 pp. 327-332, Sept. 2009.
- [3] J. Kim, Y. Kim, and S. Kim, "Indoor localization for mobile robot using extended kalman filter," *J. Korean Inst. Intell. Syst. (JKIIS)*, vol. 18, no. 5, pp. 706-7011, Oct. 2008.
- [4] J. Kim, S. Jung, T. Jeon, and S. Kim, "SLAM based on feature map for autonomous vehicle," *J. Korea Inst. Inf. and Commun. Eng. (JKICE)*, vol. 13, no. 7, pp. 1437-1443, Jul. 2009.
- [5] J. Kim, S. Jung, T. Jeon, and S. Kim,

“Path-planning using modified genetic algorithm and SLAM based on feature map for autonomous vehicle,” *J. Korean Inst. Intell. Syst. (JKIIS)*, vol. 19, no. 3, pp. 381-387, Jun. 2009.

- [6] D. Cardarelli, “An integrated MEMS inertial measurement unit,” *IEEE Position Location and Navig. Symp.*, pp. 314-319, 2002.
- [7] R. Hasegawa, “A study on rotation and its application for attitude reference system,” *J. Japan Soc. for Aeronautical and Space Sci.*, vol. 47, no. 540, pp. 16-23, Jan. 1999.
- [8] J. Gwon, T. Kim, and J. Yoo, “Path following control for an omni-directional mobile robot using low-cost gyro sensor,” in *Proc. Inst. Control, Robotics and Syst. (ICROS)*, vol. 39, pp. 222-227, May 2010.
- [9] B. W. Kwak, et al., “Docent robot using magnetic guidance and RFID,” in *Proc. Korean Inst. Intell. Syst. (JKIIS) Fall Conf.*, vol. 20, no. 2, pp. 149-151, Nov. 2010.
- [10] S. Park, S. Hong, and S. Kim, “RFID-based self-localization using gyro,” in *Proc. Ins. Control, Robotics and Syst. (ICROS)*, vol. 8, pp. 181-184, Dec. 2009.
- [11] M. Chae and T. Yang, “A study on precise localization for mobile robot based on artificial landmarks in the ceiling,” *J. Korean Inst. Inf. Technol. (JKIIT)*, vol. 9, no. 8 pp. 85-92, Aug. 2011.
- [12] M. K. Jung, C. Y. Choi, and D. M. Lee, “The compensation algorithm for localization using the least-squares method in NLOS environment,” *J. KICS*, vol. 37-B, no. 4, pp. 309-316, Apr. 2012.
- [13] M. K. Jung and D. M. Lee, “Performance analysis of the localization compensation algorithm for moving objects using the least-squares method,” *J. KICS*, vol. 39C, no. 1, pp. 9-16, Jan. 2014.

**정 무 경 (Moo Kyung Jung)**



2011년 8월 : 동명대학교 컴퓨터공학과 졸업  
 2013년 8월 : 동명대학교 컴퓨터미디어공학과 석사  
 2013년 9월~현재 : 동명대학교 컴퓨터미디어공학과 박사과정

<관심분야> 센서네트워크, 위치추정, 음원인식

**이 동 명 (Dong Myung Lee)**



1982년 2월 : 숭실대학교 전자계산학과(학사)  
 1990년 8월 : 숭실대학교 전산공학과(석사)  
 1997년 8월 : 숭실대학교 전자계산학과(박사)  
 1982년 3월~2000년 2월 : 한국전자통신연구원 책임연구원

2000년 3월~현재 : 동명대학교 컴퓨터공학과 교수  
 <관심분야> 이동통신시스템, 센서네트워크, 위치추정 시스템