

광 마우스 센서를 이용한 이동로봇 좌표추정

Coordinate Estimation of Mobile Robot Using Optical Mouse Sensors

박 상 형, 이 수 영*
(Sang-Hyung Park¹ and Soo-Yeong Yi^{1,*})

¹Dept. of Electrical and Information Engineering, Seoul National University of Science and Technology

Abstract: Coordinate estimation is an essential function for autonomous navigation of a mobile robot. The optical mouse sensor is convenient and cost-effective for the coordinate estimation problem. It is possible to overcome the position estimation error caused by the slip and the model mismatch of robot's motion equation using the optical mouse sensor. One of the simple methods for the position estimation using the optical mouse sensor is integration of the velocity data from the sensor with time. However, the unavoidable noise in the sensor data may deteriorate the position estimation in case of the simple integration method. In general, a mobile robot has ready-to-use motion information from the encoder sensors of driving motors. By combining the velocity data from the optical mouse sensor and the motion information of a mobile robot, it is possible to improve the coordinate estimation performance. In this paper, a coordinate estimation algorithm for an autonomous mobile robot is presented based on the well-known Kalman filter that is useful to combine the different types of sensors. Computer simulation results show the performance of the proposed localization algorithm for several types of trajectories in comparison with the simple integration method.

Keywords: mobile robot localization, optical mouse sensor, Kalman filter algorithm

1. 서론

이동로봇이 자율적으로 움직이기 위해서는 전역공간 상에서 자신의 좌표를 가능한 정확히 알 수 있어야 한다. 여기서 이동로봇의 좌표란 위치와 방향을 포함하는 3변수 상태벡터, (x, y, θ) 를 의미한다. 이동로봇 구동 모터의 속도를 측정하여 누적함으로써 좌표를 추정하는 추측항법(dead-reckoning)은 속도 측정과정에서의 신호잡음 문제뿐만 아니라 각 바퀴의 크기 차이와 지면에서의 미끄러짐과 같은 이유로 위치좌표 추정의 결과가 매우 부정확해질 수 있다. 이를 극복하기 위해 초음파, 스테레오 카메라, 레이저 스캐너와 같은 별도의 다양한 센서들을 이용한 위치추정 알고리즘이 개발되어 왔다. 그러나 대부분의 센서들은 가격이 비싸며 연산량이 많은 문제점이 있다.

이러한 문제점에 대한 대안으로서 광 마우스 센서를 사용하는 방법이 제안되었다[1]. 광 마우스는 영상센서를 이용하여 지면과의 상대적인 이동량을 측정하고, 이를 샘플링 시간으로 나누어 속도를 측정하는 방식으로서, 상대적으로 가격이 저렴하고, 사용이 쉽다는 장점이 있다. 광 마우스 센서 1개만으로는 이동로봇의 방향각 변화율을 포함하는 3차원의 속도 벡터 측정이 불가능하므로 보통 2개 이상을 이동로봇의 바닥면에 배치하여 사용한다. 다수개의 광 마우스 센서를 사용하는 경우에는 추정하고자 하는 미지의 3변수 속도 벡터에 대해 잉여의 센서 데이터가 존재하게 되므로 최적의 속

도 추정값을 얻기 위한 방법이 필요하다. [2,3]에서는 이동로봇 속도 벡터 추정에 필요한 다수개의 광 마우스 센서 최적 배치 방법에 대해 연구하였다.

계측된 속도 벡터 값으로부터 좌표 값을 추정하는 가장 간단한 방법은 속도 값을 단순히 적분하는 것이다. 그러나 계측된 속도 데이터에 잡음이 내재되어 있는 경우 단순 적분에 의한 좌표 추정 결과가 시간이 지남에 따라 매우 부정확해질 수 있다. 그러므로 별도의 센서 데이터를 이용하여 좌표 추정 오차를 보정해 줄 수 있는 방법이 필요하다. [4]에서는 광 마우스 센서를 통해 얻은 이동로봇의 좌표 추정 값을 이동로봇 운동공간에 설치된 별도의 전역 카메라 센서 데이터를 이용하여 보정하기 위한 칼만 필터[5] 기반의 좌표추정 알고리즘을 제안하였다. 일반적으로 모든 이동로봇은 구동 모터에 엔코더 센서를 가지므로 이를 이용하여 이동로봇의 속도를 예측할 수 있다. [6,7]에서는 이동로봇의 엔코더 센서를 통해 예측한 속도 값을 광 마우스 센서의 속도 데이터를 이용하여 보정하는 확장 칼만 필터 알고리즘을 제시하였다. 좌표 값을 얻기 위해서는 보정된 속도 데이터를 별도로 다시 적분해야 한다.

본 논문에서는 광 마우스 센서를 이용한 이동로봇 좌표 추정을 위해 이동로봇의 기구학적 운동방정식을 제한조건으로 갖는 새로운 칼만 필터 알고리즘을 제시한다. 본 논문의 좌표 추정 알고리즘은 이동로봇의 속도 값을 먼저 추정하고, 다시 이를 적분함으로써 좌표 값을 얻는 방식을 벗어나 직접 좌표 값을 추정하기 위해 확장 상태 구조를 갖는다. 또한 본 논문의 알고리즘은 미지의 3변수 좌표 벡터의 개수보다 적은 수의 광 마우스 센서를 사용하는 경우에도 효과적으로 적용될 수 있다. 본 논문의 구성은 다음과 같다. 먼저 II장에서 이동로봇의 기구학적 운동 방정식과 광 마우스 센서 속도 기구학에 관해 설명한다. 또한 운동 방정식으로부터 좌표 값을

* Corresponding Author

Manuscript received February 12, 2016 / revised May 23, 2016 / accepted August 16, 2016

박상형, 이수영: 서울과학기술대학교 전기정보공학과
(scrtrees@gmail.com/suylee@seoultech.ac.kr)

※ 본 논문은 2016 ICROS 전북제주지부 학술대회에서 초안이 발표되었음[9].

※ 이 연구는 서울과학기술대학교 교내연구비의 지원으로 수행되었음.

예측하기 위한 확장 상태 방정식을 도출한다. 센서의 속도 기구학은 서로 다른 위치에 설치된 n 개의 광 마우스 센서로 계측한 속도 데이터로부터 이동로봇 기준점의 속도를 계산하기 위한 것이다. III장에서는 확장 상태 방정식으로 예측된 좌표 값을 광 마우스 센서 속도 계측 값을 이용하여 보정하는 칼만 필터 좌표 추정 알고리즘을 기술하며, IV장에서는 본 알고리즘에 대한 컴퓨터 시뮬레이션 결과를 제시한다. 그리고 V장에서 결론을 맺는다.

II. 이동로봇 운동방정식 및 광 마우스 센서 속도 기구학

1. 이동로봇 운동방정식

2차원 $x-y$ 평면공간에서 차동 바퀴형 이동로봇의 운동 방정식은 다음과 같다.

$$\begin{aligned} \dot{x} &= \cos \theta v \\ \dot{y} &= \sin \theta v \\ \dot{\theta} &= w \end{aligned} \quad (1)$$

여기서 v 와 w 는 이동로봇 두 바퀴의 회전속도에 의한 선속도 및 각속도 입력을 나타낸다. 두 바퀴의 회전속도와 (v, w) 의 상관관계는 다음과 같다.

$$\begin{aligned} v &= \frac{r}{2}(\omega_r + \omega_l) \\ w &= \frac{r}{D}(\omega_r - \omega_l) \end{aligned} \quad (2)$$

식 (2)에서 ω_r 과 ω_l 은 각각 왼쪽 바퀴와 오른쪽 바퀴의 회전속도를 의미하며, r 은 바퀴의 반지름, 그리고 D 는 두 바퀴 사이의 횡방향 거리를 나타낸다. 식 (1)로 표현되는 연속 시간 영역 운동 방정식을 이산시간 영역에서 근사적으로 식 (3)과 같이 나타낼 수 있다.

$$\begin{aligned} x_k &= x_{k-1} + \Delta T \cos \theta_{k-1} v_{k-1} \\ y_k &= y_{k-1} + \Delta T \sin \theta_{k-1} v_{k-1} \\ \theta_k &= \theta_{k-1} + \Delta T w_{k-1} \end{aligned} \quad (3)$$

여기서 ΔT 는 샘플링 간격이며, k 는 이산화된 시간을 나타낸다. 물론 위 식 (3)보다 정교한 이산시간 영역 표현에 관한 연구결과들이 있지만[8], 여기서는 ΔT 가 충분히 작다는 가정하에 간단히 1차 근사에 의한 이산화 방법을 적용한다. 이후의 과정은 보다 정교한 이산화 방법을 적용한 경우에도 동일하게 적용될 수 있다. 식 (3)을 반복 적용하면 식 (4)를 얻을 수 있다.

$$\begin{aligned} x_{k+1} &= x_{k-1} + \Delta T \cos \theta_{k-1} v_{k-1} + \Delta T \cos \theta_k v_k \\ y_{k+1} &= y_{k-1} + \Delta T \sin \theta_{k-1} v_{k-1} + \Delta T \sin \theta_k v_k \\ \theta_{k+1} &= \theta_{k-1} + \Delta T w_{k-1} + \Delta T w_k \end{aligned} \quad (4)$$

식 (3)과 식 (4)를 결합하여 다음과 같은 확장 상태 방정식을 구할 수 있다.

$$X_{k+1} = AX_k + BU_k \quad (5)$$

여기서 상태변수, X_k 와 입력, U_k , 그리고 행렬 A 와 B 는 각각 다음과 같다:

$$\begin{aligned} X_k &= [x_{k-1} \ y_{k-1} \ \theta_{k-1} \ x_k \ y_k \ \theta_k]^T \\ U_k &= [v_{k-1} \ w_{k-1} \ v_k \ w_k]^T \\ A &= \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \end{bmatrix}, \\ B &= \Delta T \cdot \begin{bmatrix} \cos \theta_{k-1} & 0 & 0 & 0 \\ \sin \theta_{k-1} & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ \cos \theta_{k-1} & 0 & \cos \theta_k & 0 \\ \sin \theta_{k-1} & 0 & \sin \theta_k & 0 \\ 0 & 1 & 0 & 1 \end{bmatrix} \end{aligned} \quad (6)$$

2. 광마우스 센서 속도 기구학

광 마우스 센서는 [3]에서 제시한 바와 같이 이동로봇 기준점을 중심으로 같은 거리에 120도 간격으로 C_1, C_2, C_3 의 위치에 설치되어 있다고 가정한다(그림 1). 이동로봇 중심으로부터 각 광 마우스 센서까지의 거리는 L 로 나타낸다. 하나의 광 마우스 센서는 각각의 센서 좌표계를 기준으로 x 와 y , 2축 방향의 속도를 감지하므로 3개의 광 마우스로부터 얻을 수 있는 속도 정보는 총 6개이다.

광 마우스 센서들로부터의 측정 값 벡터를 다음과 같이 나타낸다.

$$S_k = [s_{1,k}^x \ s_{1,k}^y \ s_{2,k}^x \ s_{2,k}^y \ s_{3,k}^x \ s_{3,k}^y]^T \quad (7)$$

여기서 각 성분의 위 첨자는 x , 혹은 y 축 방향 속도 성분을 나타낸다. 예로서 $s_{2,k}^y$ 는 2번째 마우스 센서 y 축 방향 속도 성분의 k 번째 순간 값을 의미한다. 미지의 이동로봇 속도 변수, $(\dot{x}, \dot{y}, \dot{\theta})$ 을 계산하기 위해서는 광 마우스 센서 속도 계측 성분 중에서 3개만 이용하면 되지만 본 논문에서는 각 마우스 센서의 모든 속도 계측 성분을 이용하는 것

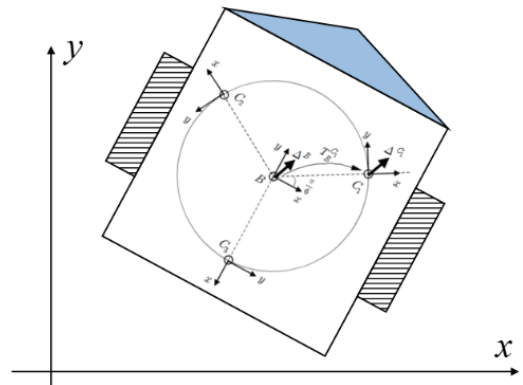


그림 1. 광 마우스 센서 배치.

Fig. 1. Placement of optical mouse sensors.

로 식을 전개한다. 이후의 과정은 어떤 3개의 속도 성분만을 사용하는 경우, 또는 3개 미만의 속도 성분만을 이용하는 경우에 대해서도 적용할 수 있다.

그림 1로부터 이동로봇의 속도와 광 마우스 센서 측정 속도 벡터 (7) 사이의 속도 기구학은 다음과 같이 나타낼 수 있다.

$$S = \begin{bmatrix} s_1^x \\ s_1^y \\ s_2^x \\ s_2^y \\ s_3^x \\ s_3^y \end{bmatrix} = \begin{bmatrix} \frac{\sqrt{3}}{2} & \frac{1}{2} & 0 \\ -\frac{1}{2} & \frac{\sqrt{3}}{2} & L \\ -\frac{\sqrt{3}}{2} & \frac{1}{2} & 0 \\ \frac{1}{2} & -\frac{\sqrt{3}}{2} & L \\ 0 & -1 & 0 \\ 1 & 0 & L \end{bmatrix} \begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{\theta} \end{bmatrix} = G \begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{\theta} \end{bmatrix} \quad (8)$$

이제 식 (5)의 확장 상태 방정식과 식 (8)의 광 마우스 측정 식을 연관시키기 위해 식 (8)의 이동로봇 속도 항을 1차 근사에 의해 이산화하면 식 (9)를 얻을 수 있다. 광 마우스는 영상센서로부터 샘플링 구간 동안 이동한 변위를 계속하고 차분(difference)에 의해 속도를 계산하여 출력하는 것이므로 식 (9)의 같은 이산화 과정은 타당하다고 할 수 있다.

$$S_{k+1} \cong G \begin{bmatrix} \frac{x_{k+1} - x_k}{\Delta T} \\ \frac{y_{k+1} - y_k}{\Delta T} \\ \frac{\theta_{k+1} - \theta_k}{\Delta T} \end{bmatrix} = \frac{1}{\Delta T} G \begin{bmatrix} -1 & 0 & 0 & 1 & 0 & 0 \\ 0 & -1 & 0 & 0 & 1 & 0 \\ 0 & 0 & -1 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x_k \\ y_k \\ \theta_k \\ x_{k+1} \\ y_{k+1} \\ \theta_{k+1} \end{bmatrix} \quad (9)$$

식 (9)는 다음과 같이 벡터 형식으로 다시 쓸 수 있다.

$$S_{k+1} = HX_{k+1}, \quad H = \frac{1}{\Delta T} G \begin{bmatrix} -1 & 0 & 0 & 1 & 0 & 0 \\ 0 & -1 & 0 & 0 & 1 & 0 \\ 0 & 0 & -1 & 0 & 0 & 1 \end{bmatrix} \quad (10)$$

III. 칼만필터 위치추정 알고리즘

식 (5)의 상태 방정식과 식 (10)의 측정 식에 구조적, 비구조적 잡음이 있는 경우 대상 시스템은 아래와 같이 나타낼 수 있다.

$$\begin{aligned} X_{k+1} &= AX_k + z_k + B(U_k + q_k) \\ S_{k+1} &= HX_{k+1} + r_{k+1} \end{aligned} \quad (11)$$

여기서 z_k , q_k , r_k 은 각각 평균값 0과 분산 $Z \in R^{6 \times 6}$, $Q \in R^{4 \times 4}$, $R \in R^{6 \times 6}$ 을 갖는 가우시안 잡음이다. 이동로봇에 있어

서 구조적 잡음은 두 바퀴 반지름, r 이나 두 바퀴 사이 거리, D 에 대한 모델 오차를 나타내며, 비구조적 잡음은 모터 제어기 신호 잡음이나 광 마우스 센서의 측정 잡음을 나타낸다.

이제 식 (11)에 대해서 다음과 같이 상태 추정을 위한 칼만 필터 알고리즘을 적용할 수 있다[5].

$$\hat{X}_{k+1}^- = A\hat{X}_k + BU_k \quad (12)$$

$$P_{k+1}^- = AP_k A' + Z + BQB'$$

$$K_{k+1} = P_{k+1}^- H' (HP_{k+1}^- H' + R)^{-1}$$

$$\hat{X}_{k+1} = \hat{X}_{k+1}^- + K_{k+1} (S_{k+1} - H\hat{X}_{k+1}^-) \quad (13)$$

$$P_{k+1} = (I - K_{k+1}H)P_{k+1}^-$$

여기서 \hat{X}_{k+1}^- 과 \hat{X}_{k+1} 은 각각 사전(priori), 사후(posteriori) 상태 추정 값을 나타내며, $P_{k+1}^- \in R^{6 \times 6}$ 과 $P_{k+1} \in R^{6 \times 6}$ 는 각각 사전, 사후의 상태추정 오차 공분산(covariance) 행렬을 의미한다. 그리고 $K_{k+1} \in R^{6 \times 6}$ 은 칼만 필터의 이득 값을 나타낸다. 식 (12)는 시스템 상태 방정식, 즉 이동로봇의 운동 방정식과 명령 입력을 이용한 예측(prediction) 단계이며, 식 (13)은 센서 측정값을 이용한 오차보정(correction) 단계를 나타낸다.

IV. 컴퓨터 시뮬레이션

앞에서 제안한 좌표 추정 알고리즘의 성능을 검증하기 위해 컴퓨터 시뮬레이션을 행하였다. 시뮬레이션의 목적은 단순 원형 궤적과 보다 복잡한 곡선 궤적에 대해서 (1) 좌표 추정 성능을 검증하고, (2) 광 마우스 센서 속도계측 성분 중 3개 미만을 사용하는 부정형(underdetermined) 경우에 있어서 제안한 알고리즘의 좌표추정 성능을 확인하기 위한 것이다.

두 바퀴 사이의 간격이 $D=10.0 \text{ cm}$ 인 크기를 갖는 이동로봇에 대해서 반지름 1.0 m 인 원을 명령 궤적으로 하였다. 이는 식 (1)에서 입력 값이 $v=1.0 \text{ m/s}$, $w=1.0 \text{ rad/s}$ 에 해당한다. 그리고 그림 1과 같은 광 마우스 센서 배치에서 이동로봇 중심과 각 센서 사이의 거리는 $L=10.0 \text{ cm}$ 로 설정하였으며, 센서 간의 사이각은 120 도로 정삼각형을 이루도록 하였다. 이는 [3]에서 제시한 광 마우스 센서의 최적배치에 해당한다. 그리고 식 (11)에서 상태 변수, 입력 변수, 측정 변수에 대한 가우시안 잡음의 분산 값은 각각 $\sigma_x^2 = 5 \times 10^{-4}$, $\sigma_y^2 = 5 \times 10^{-4}$, $\sigma_\theta^2 = 1.0 \times 10^{-4}$, $\sigma_v^2 = 1.5 \times 10^{-2}$, $\sigma_w^2 = 1.5 \times 10^{-2}$, $\sigma_s^2 = 2.5 \times 10^{-1}$ 로 설정하였다. 잡음의 공분산 행렬은 다음과 같다.

$$Z = \text{diag}[\sigma_x^2 \quad \sigma_y^2 \quad \sigma_\theta^2 \quad \sigma_x^2 \quad \sigma_y^2 \quad \sigma_\theta^2]$$

$$Q = \text{diag}[\sigma_v^2 \quad \sigma_w^2 \quad \sigma_v^2 \quad \sigma_w^2] \quad (14)$$

$$R = \text{diag}[\sigma_s^2 \quad \sigma_s^2 \quad \sigma_s^2 \quad \sigma_s^2 \quad \sigma_s^2 \quad \sigma_s^2]$$

여기서 $\text{diag}[\]$ 는 대각행렬(diagonal matrix)을 의미한다.

그림 2는 광 마우스 센서 속도값 중, 예로서 s_1^x 와 s_1^y 를 그래프로 나타낸 것이다.

또한 단순 원 궤적이 아닌 보다 복잡한 명령 궤적에 대한 좌표추정 성능을 검증하기 위해 $v=1+N_v \text{ (m/s)}$, $w=$

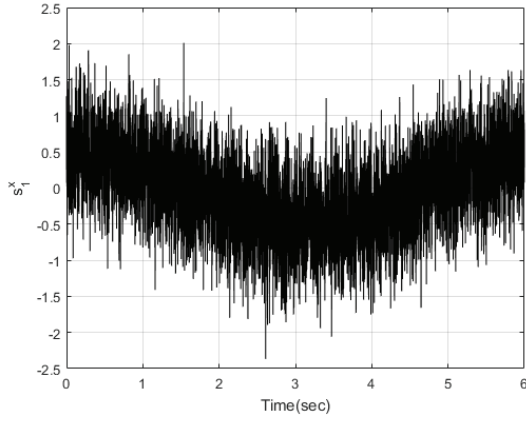
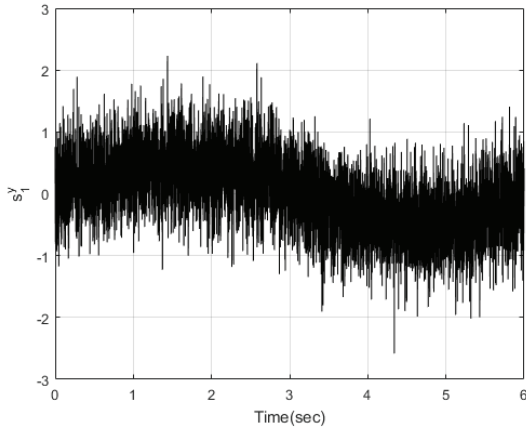
(a) x velocity data from optical mouse sensor at C_1 : s_1^x .(b) y velocity data from optical mouse sensor at C_1 : s_1^y .

그림 2. 원 궤적에 대한 광 마우스 센서 속도 측정 값.

Fig. 2. Velocity data measured by optical mouse sensor for circular trajectory.

$5\sin(2\pi t/6) + N_w$ (rad/s) 인 잡음이 포함된 곡선 명령 궤적에 대한 컴퓨터 시뮬레이션을 수행하였다. 이 때 N_v 와 N_w 는 각각 평균이 0 이고 분산이 20 인 가우시안 잡음이다. 곡선 명령 궤적에 대한 광 마우스 센서의 속도 측정값 중 s_1^x 와 s_1^y 를 그래프를 그림 3에 나타낸다.

1. 좌표추정 결과

그림 4는 원형의 명령 궤적에 대한 본 논문의 좌표추정 방법의 성능을 보여준다. 비교를 위하여 광 마우스 센서 계측값으로부터 계산한 이동로봇 속도 벡터 값을 단순 적분하는 좌표추정 방법의 결과를 함께 제시하였다.

속도 적분 기반의 좌표추정 방법은 다음과 같다. 즉, 광 마우스 센서는 기본적으로 속도값을 계측하며, 속도 계측값과 식 (8)을 이용하여 이동로봇 기준점의 속도를 계산하기 위해서 다음 식과 같이 의사 역행렬(pseudo inverse)을 사용한다 [2,3].

$$\begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{\theta} \end{bmatrix} = (G^T G)^{-1} G^T S \quad (15)$$

속도 적분 기반의 좌표추정 방법은 식 (15)의 결과를 단순

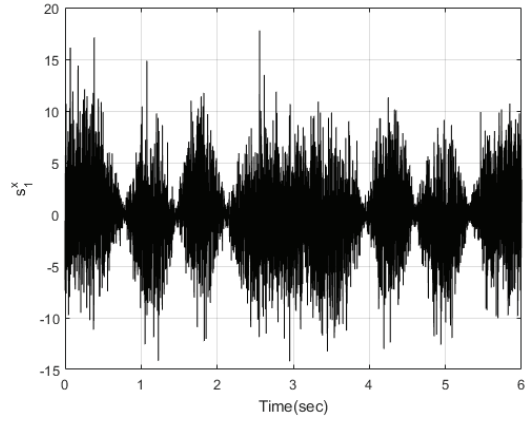
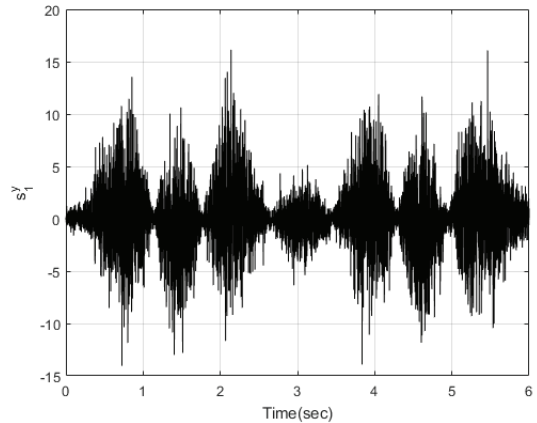
(a) x velocity data from optical mouse sensor at C_1 : s_1^x .(b) y velocity data from optical mouse sensor at C_1 : s_1^y .

그림 3. 곡선 궤적에 대한 광 마우스 센서 속도 측정 값.

Fig. 3. Velocity data measured by optical mouse sensor for curved trajectory.

적분하는 것이다.

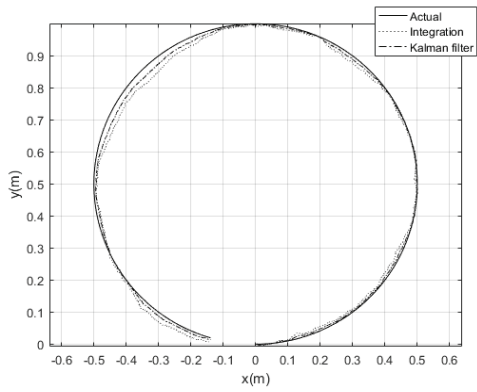
그림 4에서 이동로봇의 운동방정식에 잡음이 포함되므로 실제 궤적(actual trajectory)은 명령궤적과 같이 이상적인 원형 궤적이 되지 않는다는 것을 보인다. 그림에서 보는 바와 같이 본 논문의 칼만 필터 알고리즘에 의한 추정궤적(그림 4(a) $x-y$ 위치 궤적과 그림 4(b) 방향각 궤적)이 실제 궤적에 보다 가까운 결과를 준다. 방향각 추정 값의 오차를 그림 4(c)에 제시하였다.

그림 5는 곡선 명령 궤적에 대한 좌표 추정 결과이다. 이 경우에도 본 논문의 칼만 필터 알고리즘을 이용한 결과가 속도 적분 기반의 알고리즘 보다 더 나은 좌표 추정 성능을 보여준다.

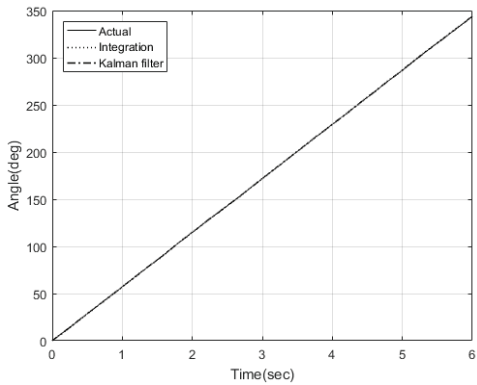
정량적 비교를 위해 표 1에 좌표 추정 오차의 평균과 표준편차를 각각 원형 명령 궤적과 곡선 명령 궤적에 대해 나타내었다. 오차의 평균과 표준편차의 비교에 의해 제시된 칼만 필터 알고리즘의 좌표 추정 성능이 단순 적분 알고리즘의 성능보다 우월함을 확인할 수 있다.

2. 광 마우스 센서 부정형(underdetermined)의 경우 좌표 추정 성능

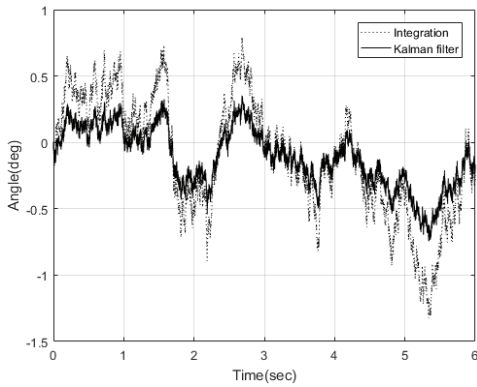
일반적으로 광 마우스 센서로부터 계측되는 속도 데이터는 2축일 뿐이므로 광 마우스 센서를 한 개만 사용하는 경우 식 (8)의 광 마우스 센서 속도 기구학은 부정형이 되고, 따라



(a) $x - y$ trajectory.



(b) Heading angle.



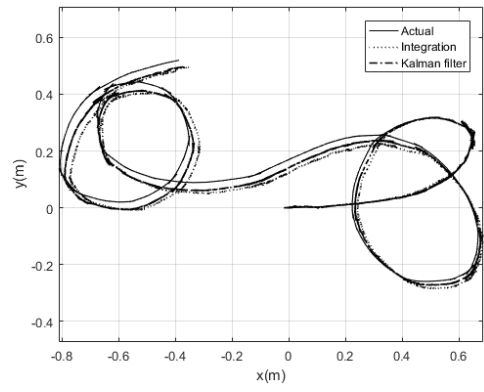
(c) Time - heading angle error.

그림 4. 원 궤적에 대한 좌표 추정 결과.

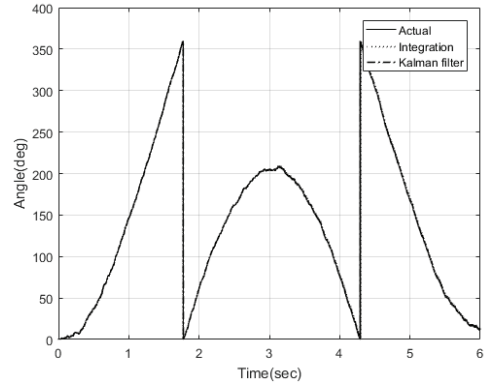
Fig. 4. Coordinate estimation for circular command trajectory.

서 이동로봇의 속도 $(\dot{x}, \dot{y}, \dot{\theta})$ 를 계산할 수 없다. 이 경우 속도 적분에 의한 좌표추정도 불가능하게 된다. 그러나 본 논문의 좌표추정 알고리즘은 속도 센서 데이터뿐만 아니라 이동로봇에 대한 선속도 및 각속도 입력 값 (v, w) 과 운동방정식을 함께 이용하므로 속도계측 센서 데이터가 불완전한 경우에도 좌표추정이 가능하다. 그림 6은 원형의 명령 궤적에 대하여 식 (7)의 광 마우스 센서 계측 데이터 중에서 $S_k = [s_{i,k}^x \ s_{i,k}^y]^T$ 만 이용한 경우, 즉 한 개의 센서만 이용하는 경우 본 논문의 좌표추정 알고리즘의 성능을 보여준다.

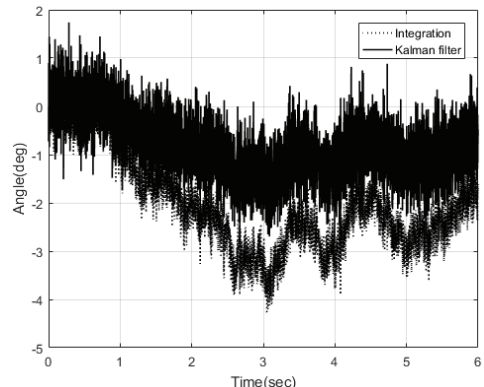
그림 7은 곡선 명령 궤적에 대하여 광 마우스 센서 데이터가 부정형인 경우, 본 논문에서 제안한 좌표추정 알고리즘의 성능을 보여준다.



(a) $x - y$ trajectory.



(b) Heading angle.



(c) Time - heading angle error.

그림 5. 곡선 궤적에 대한 좌표 추정 결과.

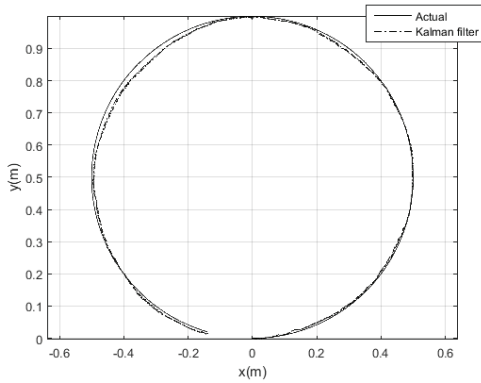
Fig. 5. Coordinate estimation for curved command trajectory.

표 1. 좌표 추정/방향각 오차 평균과 표준편차.

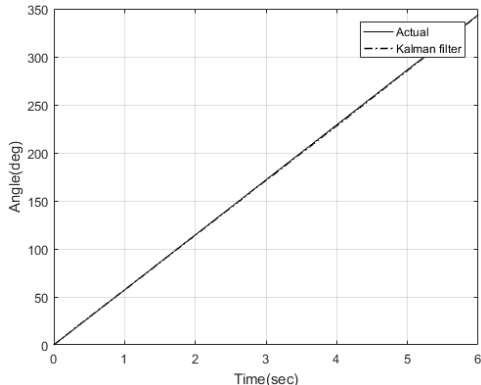
Table 1. Mean and standard deviation of position/heading angle estimation error.

		속도적분 기반 알고리즘		제안 알고리즘
원 궤적	좌표 (m)	평균	0.0210	0.0111
		표준편차	0.0101	0.0057
	방향각 (°)	평균	-0.1198	-0.1146
		표준편차	0.4469	0.2244
곡선 궤적	좌표 (m)	평균	0.0171	0.0072
		표준편차	0.0188	0.0082
	방향각 (°)	평균	-2.3761	-0.9699
		표준편차	1.1106	0.7114

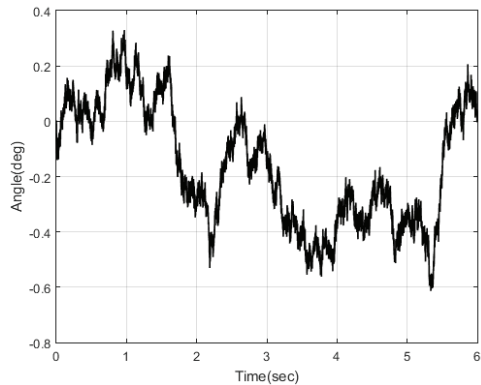
Fig. 1. Mean and standard deviation of position/heading angle estimation error.



(a) x - y trajectory.



(b) Heading angle.



(c) Time - heading angle error.

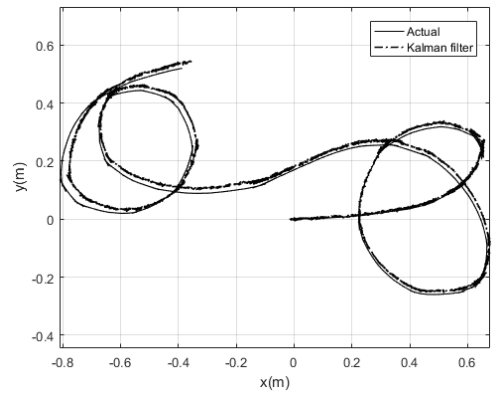
그림 6. 원 궤적에 대한 좌표 추정 결과.

Fig. 6. Coordinate estimation for circular command trajectory.

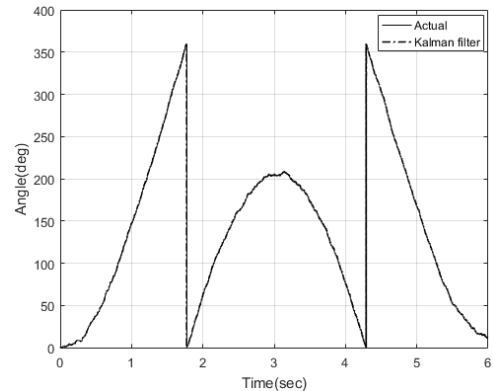
표 2에 그림 6과 그림 7의 위치 및 방향각 추정 오차의 평균과 표준편차를 정량적으로 제시하였다.

V. 결론

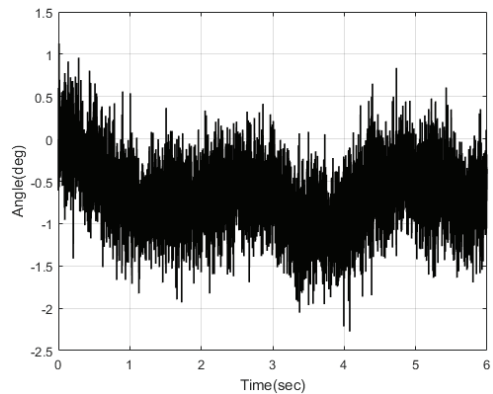
본 논문에서는 광 마우스 센서를 갖는 이동 로봇의 좌표 추정을 위해 칼만 필터 기반 알고리즘을 제안하고, 컴퓨터 시뮬레이션을 통해 기존의 속도계측 값 적분에 의한 좌표추정 성능과 비교함으로써 제안 알고리즘의 우수성을 보였다. 본 논문에서 제안한 방법의 장점은 다음과 같다: 1) 광 마우스 센서 속도 신호의 순간 값만 이용하는 것이 아니라 이동 로봇의 운동 방정식과 선속도 및 각속도 명령 입력 값을 함께 이용하므로, 단순 신호적분 방법에 비해 보다 나은 이동 로봇 좌표 추정 결과를 얻을 수 있다. 2) 추정하고자 하는 이



(a) x - y trajectory.



(b) Heading angle.



(c) Time - heading angle error.

그림 7. 곡선 궤적에 대한 좌표 추정 결과.

Fig. 7. Coordinate estimation for curved command trajectory.

표 2. 칼만 필터 알고리즘 좌표 추정/방향각 오차 평균과 표준편차.

Table 2. Mean and standard deviation of position/heading angle estimation error for Kalman filter algorithm.

원 궤적	좌표 (m)	평균	0.0109
		표준편차	0.0061
	방향각 (°)	평균	-0.1626
		표준편차	0.2184
곡선 궤적	좌표 (m)	평균	0.0188
		표준편차	0.0209
	방향각 (°)	평균	-0.6447
		표준편차	0.4289

동로봇 좌표변수의 개수 보다 적은 수의 광 마우스 센서를 사용하는 경우에도 유사한 좌표 추정 결과를 얻을 수 있다.

칼만 필터 알고리즘은 다양한 센서 데이터들을 융합할 수 있는 수학적 기반이 되므로 본 논문에서 제시한 위치 추정 알고리즘은 광 마우스 센서와 다른 센서들을 융합하여 사용하는데 도움을 줄 수 있을 것으로 기대된다.

REFERENCES

- [1] M. Cimino and P. R. Pagilla, "Location of optical mouse sensors on mobile robots for odometry," *Proc. of IEEE International Conference on Robotics and Automation*, pp. 5429-5434, 2010.
- [2] S. Kim and S. Lee, "Robust velocity estimation of an omnidirectional mobile robot using a polygonal array of optical mice," *International Journal of Control, Automation, and Systems*, vol. 6, no. 5, pp. 713-721, 2008.
- [3] S. Kim and H. Kim, "Optimal optical mouse array for high performance mobile robot velocity estimation," *Journal of Institute of Control, Robotics and Systems*, vol. 19, no. 6, pp. 555-562, 2013.
- [4] D. Sekimori and F. Miyazaki, "Self-localization for indoor mobile robots based on optical mouse sensor values and simple global camera information," *Proc. of 2005 IEEE International Conference on Robotics and Biomimetics(ROBIO)*, pp. 605-610, 2005.
- [5] G. Welch and G. Bishop, "An introduction to the Kalman filter," Technical Report, Department of Computer Science, University of North Carolina at Chapel Hill, 2002.
- [6] S. Lee and J. Song, "Mobile robot localization using optical flow sensors," *International Journal of Control, Automation, and Systems*, vol. 2, no. 4, pp. 485-493, 2004.
- [7] S. Lee and J. Song, "Robust mobile robot localization using optical flow sensors and encoders," *Proc. IEEE International Conference on Robotics and Automation*, pp. 1039-1044, 2005.
- [8] N. Doh, H. Choset, and W. Chung, "Relative localization using path odometry information," *Autonomous Robots*, vol. 21, no. 2, pp. 143-154, 2006.
- [9] S. Park and S. Yi, "Localization of mobile robot with optical mouse sensor," *Proc. of 2012 ICROS Jeonbuk and Jeju Regional Conference (in Korean)*, pp. 46-48, 2016.



박상형

2016년 2월 서울과학기술대학교 전기정보공학과(공학사). 2016년 3월~현재 서울과학기술대학교 석사과정 재학 중. 관심분야는 비선형제어, 이동로봇.



이수연

1988년 2월 연세대학교 전자공학과(공학사). 1990년 2월 KAIST 전기및전자공학과(공학석사). 1994년 8월 KAIST 전기및전자공학과(공학박사). 1995년 3월~1999년 8월 KIST 시스템연구부 선임연구원. 1997년 2월~1998년 2월 Univ. of Southern California 박사후과정. 1999년 9월~2007년 2월 전북대학교 전자정보공학부 부교수. 2005년 6월~2006년 8월 Univ. of Illinois at Urbana-Champaign 방문교수. 2007년 3월~현재 서울과학기술대학교 전기정보공학과 교수. 관심분야는 보행로봇, 로봇비전, 이동로봇.