관성센서를 이용한 임베디드 주행 시스템의 오도메트리 오차 보정

Odometry Error Correction of Embedded Navigation System using Inertial Sensor

*주영훈 [1], #송재복 [2]
*Y. H. Ju (neje@korea.ac.kr)[1] , #J. B. Song (jbsong@korea.ac.kr) [2]
[1,2] 고려대학교 기계공학부

Key words : Embedded navigation system, Odometry error correction, Inertial sensor

1. 서론

이동로봇의 주행 시스템에서 이동거리를 측정하는 것을 오도메트리라고 한다. 엔코더를 이용한 추측항법(dead reckoning)은 오도메트리에 있어 매우 일반적인 기법이다. 하지만 이동로봇이 운용되는 환경에서 바닥의 비평탄성, 바닥과 바퀴 사이의 마찰력 변화 등에 의해 미끄러짐 현상이 발생한다. 따라서 엔코더만으로 정밀하고 신뢰성있는 이동로봇의 위치를 결정할 수 없다.

실제 응용에서는 별도의 외부환경을 인식하는 센서를 추가하여 오도메트리 오차를 보정한다. 기본적인 원리는 센서에 의해 획득된 외부 지점(자연표식, 태그, 위성)의 절대 위치 정보에 기반하여, 이동로봇의 현재 위치를 측정하고 보정하는 것이다. 그러한 기법에는 카메라를 통해 인식된 자연표식을 이용하는 기법[1], RFID 태그를 이용하는 기법[2], GPS 를 이용한 기법[3] 등이 있다. 하지만 카메라의 경우 조명의 변화에 민감하고, 표식이 추출되지 않거나 어두운 공간에서는 사용할 수 없다. RFID 태그를 이용한 방식은 초기 설치과정이 필요하고, 태그가 설치되지 않은 공간에서는 사용할 수 없다. 그리고 GPS 는 실내에서 사용할 수 없다.

미세가공기술과 반도체 일괄제작기술의 향상에 따라 다양한 저가의 관성센서가 출시되고 있다. 이것은 저가의 이동로봇 플랫폼에도 용이하게 적용할 수 있다. 이러한 센서로부터 얻은 정보를 처리하는 것은 큰 계산량을 필요로 하지 않으므로, 임베디드 시스템에서도 충분한 결과를 얻을 수 있다. 그리고 로봇의 내부 상태만을 측정하여 위치를 인식하므로, 외부 환경의 변화에 영향을 받지 않는다. 따라서 외부 환경정보에 기반한 위치인식 기법들이 갖는 단점을 보완하는 데에도 적용될 수 있다.

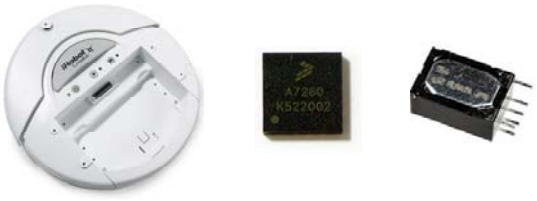
오도메트리 정보의 신뢰성이 낮은 이유는 바퀴와 바닥 사이의 상호작용을 엔코더가 정확히 반영하지 않기 때문이다. 하지만 관성센서의 정보를 이용하면 미끄러짐이 일어난 상황에서 로봇의 실제 운동을 반영하는 관성력을 측정하여 오도메트리의 오차를 보정할 수 있다. 이전의 연구[4]에서는 충분한 초기교정을 거치고, 적절한 오차모델이 적용된 관성센서의 정보를 적분하여 얻어진 변위정보와 엔코더에 의해 얻어진 오도메트리 정보를 확장 칼만 필터(extended Kalman filter) 등의 기법으로 융합하여 보정하였다. 하지만 수치적분 오차, 변화율의 작은 오차 등이 적분되는 과정을 통해 증폭되고 누적된다. 절대적 기준 없이 상대적인 변화량만을 측정하는 센서이므로, 이러한 오차는 끝없이 발산한다.

이러한 문제를 극복하기 위하여, 본 논문에서는 역으로 엔코더의 신호를 미분하여, 모터에 의해 추정되는 로봇 좌표계의 가속도를 구하고, 이를 관성센서로부터 얻은 값과 비교하여 실제 로봇의 운동을 측정하고, 미끄러짐 등의 상황을 감지하여 오도메트리 오차를 보정하는 기법을 제안한다. 관성센서의 값으로부터 변위정보를 얻기 위한 적분을 하지 않으므로, 수치 적분 오차와 편류현상(drift) 등에 의한 오차가 누적되어 발산하지 않고, 오차 항의 개수가 줄어든다.

본 논문은 다음과 같이 구성되어있다. 2 절에서는 본 논문에서 구성한 임베디드 로봇 시스템에 대해 설명하고, 3 절에서는 제안된 알고리듬을 설명한다. 4 절에서는 실험을 통해 제안한 알고리듬의 유효성을 검증하고, 5 절에서 결과에 대해 논의한다.

2. 임베디드 시스템의 구성

저가형 이동로봇에 적용될 수 있는 임베디드 주행 시스템을 구성하였다. 기본적인 구조는 저전력, 저가, 고성능의 특성을 갖는 ARM 코어 기반의 프로세서를 사용하여 구성된 임베디드 시스템과 이동로봇 플랫폼으로 구성된다. 관성센서는 가속도 센서와 각속도 센서를 각각 사용하였다.



(a) (b) (c)

Fig. 1 Configuration of embedded navigation system; (a) iRobot Create, (b) Freescale MMA7260QT, and (c) Murata ENV-05G.

임베디드 시스템은 ARM9 코어 기반의 Intel PXA270 (520MHz) 프로세서를 사용하였고, 운영체제로 임베디드 리눅스를 사용한다. 이동로봇 플랫폼은 Fig. 1(a)에 나타낸 iRobot 사의 개발자용 플랫폼인 Create 를 사용하여 구성하였다. Create 는 시리얼 통신을 통해 간편하게 로봇을 제어하고, 센서정보를 획득할 수 있다. 하지만 엔코더 신호를 내부의 제어시스템에서 계산하여 위치정보만을 제공하기 때문에 직접 바퀴의 속도를 계산할 수 없다. 따라서 이를 분해하여 내부의 엔코더 신호를 직접 AVR 로 연결하고, 엔코더의 펄스를 계수한 값을 시리얼 통신을 통해 임베디드 시스템에 전달하도록 하였다.

Table 1 Specifications of inertial sensors.

	MMA7260QT	ENV-05G
Type	Linear acceleration	Angular rate
Direction	3-axis	1-axis
Range	±1.5g	±70 °/s
Resolution	-	0.1 °/s
Sensitivity	800 mV/g	25 mV/°/s

관성센서는 Fig. 1(b)의 가속도 센서와 Fig. 1(c)의 각속도 센서 두 종류를 사용하였다. 센서의 자세한 스펙은 Table 1 에 제시되어 있다. 이동로봇의 선형 가속도를 정확히 측정하기 위해서 가속도 센서는 로봇의 좌표계 중심에 설치되어야 한다. 각속도 센서는 센서의 감지축과 측정하려는 축(Z 축)이 평행하도록 설치되면 위치에는 영향을 받지 않는다. 설치 위치는 Fig. 2(a)와 같다. 임베디드 시스템에는 센서의 신호를 직접 측정할 수 있는 ADC 기능이 없

기 때문에, 엔코더 신호 계측을 위해 사용한 AVR 을 이용하여, 센서 출력값을 디지털값으로 변환하고, 이를 시리얼 통신을 통해 임베디드 보드로 전송할 수 있도록 하였다. 전체 시스템의 구성은 Fig. 2(b)과 같다.

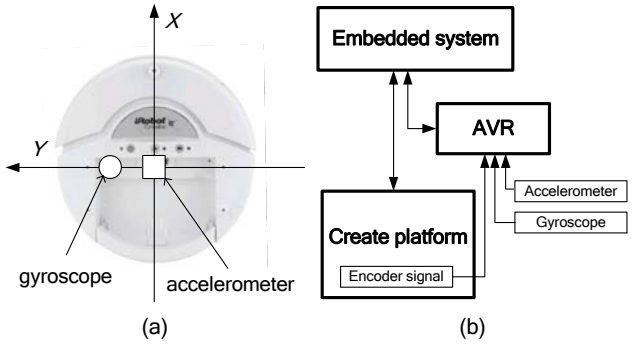


Fig. 2 Embedded navigation system; (a) positions of inertial sensors, and (b) block diagram of system.

3. 오도메트리 오차 보정 알고리듬

본 알고리듬의 기본개념은 엔코더로부터 얻은 이동로봇의 속도를 미분하여, 계산된 가속도와 가속도 센서로 측정한 가속도를 비교하여 오도메트리의 오차를 보정하는 것이다. 차륜구동 방식의 이동로봇에서는 엔코더로부터 얻은 i 번째 시간의 좌우바퀴의 속도 $\upsilon_{r,i}$, $\upsilon_{l,i}$ 로부터 식(1), (2)에 의해 병진속도 υ_i 와 회전속도 ω_i 를 구할 수 있다. b 는 두 바퀴사이의 거리이다. 식(3)을 이용하여 이전 시간에 계산된 속도와 현재 시간에 계산된 속도로부터 엔코더에 의한 가속도 a_i^{enc} 를 계산할 수 있다. 이를 식(4)에서처럼 직접 가속도 센서 a_i^{acc} 로부터 측정한 가속도와 비교하여 엔코더가 정확히 로봇의 운동상태를 표현하고 있는지를 판단한다. 회전운동에 대한 상태는 식(5)를 이용하여 엔코더에 의한 회전속도와 각속도 센서의 출력값을 비교하여 판단한다.

$$\upsilon_i = \frac{1}{2}\left(\upsilon_{r,i} + \upsilon_{l,i}\right) \tag{1}$$

$$\omega_i = \frac{1}{b}\left(\upsilon_{r,i} - \upsilon_{l,i}\right) \tag{2}$$

$$a_i^{enc} = \left(\upsilon_i - \upsilon_{i-1}\right) \Big/ \Delta t \tag{3}$$

$$\Delta a_i = \left(a_i^{enc} - a_i^{acc}\right) \tag{4}$$

$$\Delta\omega_i = \omega_i^{enc} - \omega_i^{gyro} \tag{5}$$

$$\Delta\upsilon_i = \Delta a_i \cdot \Delta t \tag{6}$$

$$\Delta\omega_i = \omega_i^{enc} - \omega_i^{gyro} \tag{7}$$

$$\upsilon_i' = \upsilon_i + \Delta\upsilon_i \tag{8}$$

$$\omega_i' = \omega_i + \Delta\omega_i \tag{9}$$

$$\begin{cases} x(t) = \int_0^t \upsilon(\tau)\cdot\cos\theta(\tau)d\tau \\ y(t) = \int_0^t \upsilon(\tau)\cdot\sin\theta(\tau)d\tau \\ \theta(t) = \int_0^t \omega(\tau)d\tau \end{cases} \tag{10}$$

이렇게 얻어진 가속도의 차이와 각속도의 차이, 즉 엔코더로부터 얻은 값과 관성센서로 얻은 값 사이의 차이가 로봇의 바퀴와 바닥 사이의 미끄러짐에 의해 발생된 오차가 된다. 이를 일정한 기준치과 비교하여, 일정범위 이내일 때는 바닥의 비평탄성에 의한 차이로 가정하여 두 값을 가중평균하여 오도메트리 정보에 반영한다. 하지만 일정값 이상의 차이를 보이게 되면, 식(6), (7)를 이용하여 관성센서에 의한 차이값을 계산하고, 이를 식(8), (9)를 이용하여, 오도메트리 정보를 보정한다. 오도메트리 정보 x, y, θ 는 식(10)와 같이 병진속도와 회전속도를 시간에 따라 적분하여 얻게 된다.

4. 실험결과

제안된 알고리듬을 임베디드 주행 시스템 상에 구현하고, 평탄하지 않는 바닥에서 주어진 경로를 추종하는 실험을 수행하였다. Figure 3(a)는 엔코더로부터 얻은 가속도와 관성센서로부터 얻은 가속도를 비교한 그래프이다. 가속도의 차이를 관찰하여 미끄러짐 현상이 발생하는 것을 감지하였다. Figure 3(b)는 지그재그 형태의 경로를 추종하는 실험을 통해 얻은 로봇의 궤적이다. 미끄러진 영역을 감지하고, 알고리듬에 의해 오도메트리를 보정하여 목적한 경로를 정확하게 추종하는 것을 확인하였다.

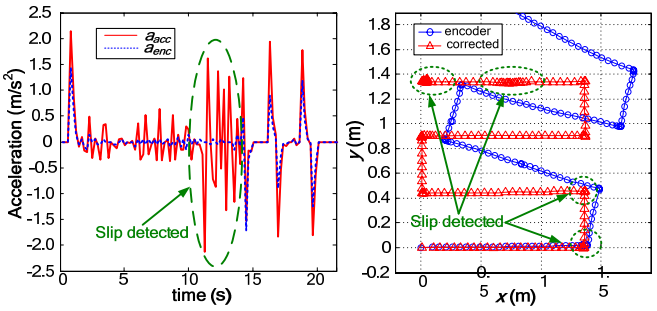


Fig. 3 Experimental results; (a) comparison of acceleration from encoder and accelerometer, and (b) slipped and corrected trajectory of mobile robot on the uneven surface.

5. 결론

본 논문에서는 엔코더로 얻어진 오도메트리 정보의 오차를 관성센서를 이용하여 극복할 수 있는 기법을 제안하였다. 이 기법을 적용하면 비평탄면을 가지고, 마찰계수가 일정하지 않은 환경에서 이동로봇의 실제 위치와 방위를 장시간동안 큰 오차없이 추정할 수 있다. 내부 상태를 측정하는 센서를 사용하므로, 외부 환경의 변화에도 성능은 영향을 받지 않는다.

후기

이 연구는 지식경제부 지원으로 수행하는 21 세기 프론티어 연구개발사업(인간지능 생활지원 지능로봇 기술개발사업)의 일환으로 수행되었음.

참고문헌

1. Hwang, S.-Y. and Song, J.-B., “Upward Monocular Camera based SLAM Using Corner and Door Features,” IFAC, 2008.
2. Hännel, D., Burgard, W., Fox, D., Fishikin, K., and Philipose, M., “Mapping and Localization with RFID Technology,” In proc. of ICRA, vol. 1, pp. 1015-1020, 2004.
3. Sukkarieh, S., Nebot, E. M., and Durrant-Whyte, H. F., “A High Integrity IMU/GPS Navigation Loop for Autonomous Land Vehicle Applications,” IEEE Trans. on Robotics and Automation, vol. 15, no. 3, pp. 572-578, 1999.
4. Barshan, B. and Durrant-Whyte, H. F., “Inertial Navigation Systems for Mobile Robots,” IEEE Trans. on Robotics and Automation, vol. 11, no. 3, pp. 328-342, 1995.