

이동로봇의 동시간 위치인식 및 지도작성(SLAM)

본 고에서는 이동로봇의 동시간 위치인식 및 지도작성(Simultaneous Localization and Mapping: SLAM) 기술에 대하여 다룬다. 이동로봇의 SLAM을 위하여, 로봇과 랜드마크의 상태를 상태공간 영역에서 같이 기술하는 방법과 센서로부터 입력된 정보를 이용하여 로봇이 상태를 추정하는 기법을 소개한다. 실제 로봇을 통한 예제를 통하여 로봇의 상태와 특징점을 동시에 추정하는 것을 보여준다.

■ 임현, 이영삼*
(인하대학교 전기공학부)

1. 서론

SLAM은 로봇의 위치인식과 위치한 환경에 대한 지도를 동시에 작성하는 것을 의미한다. 이는 우리가 일상에서 행하는 미지의 환경 탐색과 매우 유사한 방법이다. 인간이 미지의 환경에 놓였을 때 위치한 환경내의 구분 가능한(distinguishable) 특징점(feature)을 인식하여 이를 기준으로 위치를 추정하게 되는데 이는 위치인식과 지도작성이 선후관계를 가지지 않는다는 것을 의미한다. 만약 우리가 흰색의 무한 평면에 놓였다고 가정해 보자. 우리는 무한한 흰색 평면에서 크게 눈에 띄는 특징들을 발견하지 못할 것이며 따라서 내가 처음 위치한 곳으로부터 내가 어떻게 이동해왔는지 알 수 없을 것이다. 또한 유사한 특징점이 많이 존재하는 경우 우리는 스스로의 위치인식을 실패한다. 일례로, 사막과 같이 비슷한 특징점들이 펼쳐지는 환경에서는 사람조차 그 방향감각을 상실하고 만다. 이와 같이, 환경을 인식가능한 정보로 변환(mapping)하고 이로부터 자기의 위치를 추정하는 것(localization)은 사람도 종종 실패하는 어려운 과제이다.

로봇이 SLAM을 가능하게 하기 위해서는 센서가 가지는 불확실성(uncertainty)을 다루는 것이 중요하다. 또한 차원이 높은 센서정보로부터 2~3차원 정도의 정보를 추려내어 이를 지속적으로 대응시키는 데이터 조합(照合, data association) 문제도 해결해야 한다. 그리고 관찰된 특징점 자료들을 효율적으로 관리하는 방법도 고안해야 한다. 이 처럼 로봇이 자기의 위치를 추정하기 위해서 많은 기법이 동원되어야 함을 알 수 있다. 본 고에서는 센서와 로봇 자세의 불확실성을 다루는 기법으로 확장 칼만 필터(Extended Kalman Filter, EKF)를 사용하는 방법을 소개한다. 확장 칼만 필터는 비선형 모델의 상태추정에 널리 이용되어 왔다. 하지만 최근에는 Unscented Kalman Filter 등이 비선형 모델에 대해서 더 나은 결과를 보여주고 있다. 데이터 조합이나 자료관리는 본 고의 범위를 벗어나므로 다루지 않는다.

2. SLAM 문제 설정

SLAM은 위치인식과 지도작성을 동시에 하는 것이다. 먼저 본 절에서는 두가지 문제를 각각 정의하고 이것을 통합적으로

기술하는 방법을 소개한다.

2.1 문제정의

로봇이 환경 안에서 이동하면서 갯수를 미리 알 수 없는 랜드마크들의 상대적인 측정값을 장착된 센서를 통하여 관찰하며 이동하는 상황을 생각해 보자. 그리고 다음의 변수를 설정하자.

- \mathbf{x}_k : k 시점에서 로봇의 위치와 자세를 표현하는 상태벡터
- \mathbf{u}_k : 로봇의 상태를 \mathbf{x}_k 로부터 \mathbf{x}_{k+1} 시점으로 이동시키기 위해서 시점에서 적용한 제어 입력 벡터
- \mathbf{m}_i : i 번째 랜드마크의 위치를 기술하는 벡터. 특징점의 위치는 시간에 불변(time invariant)하다고 가정한다.
- $\mathbf{z}_{i,k}$: k 시점에서 관찰된 i 번째 랜드마크의 위치 정보. 다수개가 관찰될 수 있다. 이를 하나의 벡터인 \mathbf{z}_k 로 표현한다.
- $\mathcal{X}_{0:k}$: k 시점까지 기록된 로봇 상태의 집합
 $\{\mathbf{x}_0, \mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_k\} = \{\mathcal{X}_{0:k-1}, \mathbf{x}_k\}$
- $\mathcal{U}_{0:k}$: k 시점까지 기록된 제어 벡터의 집합
 $\{\mathbf{u}_0, \mathbf{u}_2, \dots, \mathbf{u}_k\} = \{\mathcal{U}_{0:k-1}, \mathbf{u}_k\}$
- \mathcal{M} : 모든 랜드마크 벡터의 집합
 $\{\mathbf{m}_1, \mathbf{m}_2, \mathbf{m}_3, \dots, \mathbf{m}_n\}$
- $\mathcal{Z}_{1:k}$: k 시점까지 기록된 관측 벡터의 집합
 $\{\mathbf{z}_1, \mathbf{z}_2, \dots, \mathbf{z}_k\} = \{\mathcal{Z}_{1:k-1}, \mathbf{z}_k\}$

2.2 위치인식 문제

위치인식(localization)은 랜드마크들의 위치가 사전에(a priori) 알려져 있는 경우 이를 기준으로 로봇의 위치를 추정하는 것이다. 우리가 일상 생활에서 주로 하는 것이 이 위치인식 문제이다. 우리는 미리 알고 있는 건물이나 구분 가능한 특징들

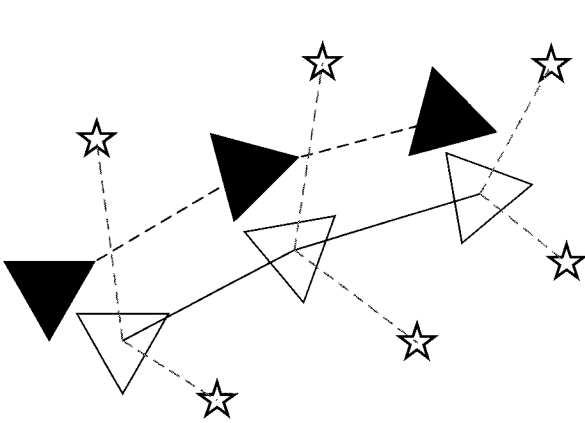


그림 1. 위치인식 문제의 도식화

면, 복도의 끝점)으로 부터 자신의 위치를 추정한다. 이를 위치 인식 문제라 할 수 있다.

로봇의 위치인식은 측정값 \mathbf{z}_k 로부터 로봇의 위치 \mathbf{x}_k 를 알아내는 것이다. 이 과정에서 기준이 되는 값은 사전에 위치가 알려진 모든 랜드마크들의 집합 \mathcal{M} 이다. 이를 결합 사후 확률(joint posterior probability)의 형태로 표현하면 (1)과 같다.

$$P(\mathbf{x}_k | \mathcal{Z}_{1:k}, \mathcal{U}_{0:k-1}, \mathcal{M}, \mathbf{x}_0) \quad (1)$$

즉, 위치인식 문제는 초기치 \mathbf{x}_0 와 $k-1$ 시점까지의 제어 입력, 관측벡터와 사전에 위치가 알려진 랜드마크를 통하여 매 k 시점마다 로봇의 위치를 추정하는 것이다. 로봇의 위치인식 과정을 그림으로 표현하면 그림 1과 같다. 그림 1은 그 위치가 미리 알려진 랜드마크를 관찰한 정보를 토대로 자신의 위치를 추정하는 것을 나타내고 있다.

그림 1에서 점선은 추정된 경로와 관찰을 나타낸다. 즉, 위치가 알려진 랜드마크를 관찰하여 로봇의 위치를 알아내는 것이 위치인식 문제이다. 이때 로봇의 위치추정의 불확실성은 센서의 오차로부터 기인한다.

2.3 지도작성 문제

지도작성 문제는 로봇의 위치가 미리 알려진 경우 이를 이용하여 관찰된 특징점들의 전역 위치를 추정하는 것이다. 이는 마치 GPS 정보로부터 우리가 지도를 작성하거나 측량을 하는 것과 같은 것이다. 우리는 GPS 정보와 같은 신뢰할 만한 데이터를 기반으로 상대 좌표의 관찰 정보를 기록하여 관찰 대상의 지구 상에서의 절대좌표를 알아낸다. 이와 같이 기준점과 상대좌표로 관찰된 결과를 누적하여 로봇이 위치한 환경을 모델링 할 수 있게 된다. 지도작성 문제는 (2)와 같이 정의 할 수 있다.

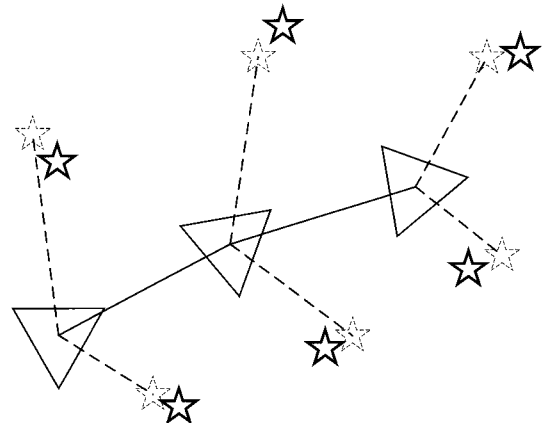


그림 2. 지도작성 문제의 도식화

$$P(\mathcal{M} | \mathcal{X}_{0:k}, \mathcal{Z}_{1:k}, \mathcal{U}_{0:k-1}) \quad (2)$$

즉, 지도작성 문제는 위치와 관측정보 그리고 제어입력으로부터 랜드마크 집합 \mathcal{M} 을 추정하는 것이다. 로봇의 지도작성 과정을 그림으로 표현하면 그림 2와 같다. 그림 2는 알려진 로봇의 위치에서 관찰한 정보를 이용하여 랜드마크의 위치를 추정하는 것을 나타내고 있다.

그림 2에서 점선은 랜드마크의 추정 위치를 의미한다. 즉, 위치가 알려진 로봇의 경로에서 관찰한 랜드마크의 정보를 전역 좌표를 알아내는 것이 지도작성 문제이다. 이때 지도의 부정확성은 센서의 오차로부터 기인한다.

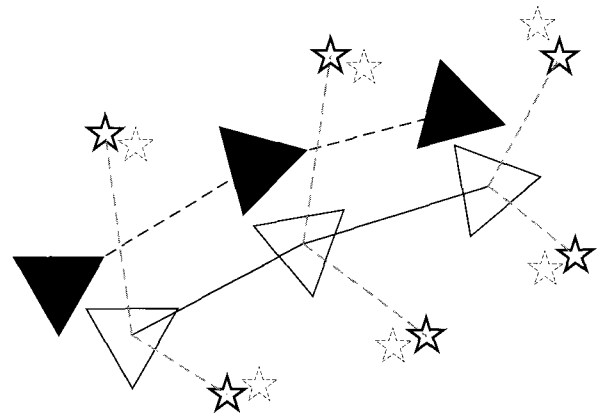


그림 3. SLAM 문제의 도식화

2.4 The Simultaneous Localization and Mapping(SLAM) 문제

대부분의 로봇 위치인식 그리고 지도작성 문제는 따로 해결할 수 없다. 위치인식 문제는 랜드마크의 위치가 미리 알려져야 한다는 가정을 가지고 있으므로, 전역 좌표계(world coordinate)와 특징점을 일치시키는 교정작업이 필요하게 된다[1, 2, 3]. 게다가, 한정적인 공간에서만 위치인식이 가능해질 것이다. 지도작성 문제는 로봇의 위치가 미리 알려져야 한다는 가정을 가지고 있다. 이것은 실외의 경우 GPS와 같은 위성 항법장치의 도움으로 가능하지만 실내의 경우 그렇지 않다.

따라서 이 둘의 관계는 서로 분리하여 생각할 수 없다. 로봇은 지속적으로 자신의 위치를 확인하기 위해서 시간에 불변한 특징들을 기준으로 삼아야 한다. 이를 위해서 지도작성을 해야 한다. 또한 지도작성을 위해서 자신의 위치를 알아야 한다. 따라서 위치인식과 지도작성은 위치한 환경내에서 로봇의 위치를 알기 위해서 모두 필요한 과정이다.

SLAM을 확률의 형태로 표현하면 k 시점에서 (3)의 확률 분포를 구하는 것이다 [6]. 이는 그림 3과 같다.

그림 3은 로봇의 위치와 랜드마크의 위치를 동시에 추정하는 것을 보여주고 있다. 그림에서 실선으로 표현된 경로는 로봇의 절대경로이다. 이 값은, 우리가 알아낼 수 없다. 하지만 측정값(랜드마크와 연결된 실선)을 통하여 로봇의 이동경로를 추정하고(점선) 이에 따라 랜드마크의 위치도 추정(점선으로 된 별)할 수 있다. 이때 불확실성은 로봇 주행거리계와 센서 오차로부터 유발된다.

$$P(\mathbf{x}_k, \mathcal{M} | \mathcal{Z}_{1:k}, \mathcal{U}_{0:k-1}, \mathbf{x}_0) \quad (3)$$

(3)의 확률분포는 관측벡터 $\mathcal{Z}_{1:k}$ 과 초기값 그리고 적용된 모

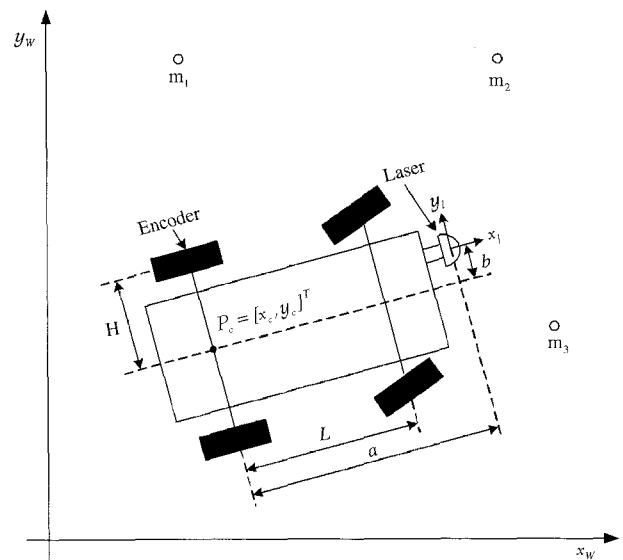


그림 4. 로봇 모델



그림 5. (5)에서 데이터 취득을 위해서 사용된 차량

든 제어입력 $U_{0:k-1}$ 가 주어진 상태에서 랜드마크의 위치와 k 시점에서의 로봇 상태벡터 \mathbf{x}_k 의 결합확률을 나타낸다. (3)의 확률분포를 구하기 위해서 재귀적 방법이 요구된다. 이는 매 k 시점마다 새로운 정보가 도착하기 때문이다. 만약 재귀적 해법이 아닌 방법이라면, 시점 k 가 무한대에 가까울 경우 해를 풀기 어렵다. 재귀적 해법을 통한 \mathbf{x}_k 의 계산은 $k-1$ 시점의 결합 확률분포 $P(\mathbf{x}_{k-1}, \mathcal{M} | \mathcal{Z}_{1:k-1}, U_{0:k-2})$ 와 \mathbf{u}_{k-1} 그리고 관측벡터 \mathbf{z}_k 를 Bayes 정리를 이용하여 계산할 수 있다. 이 계산에는 상태 공간 모델과 관측 모델(observation model)이 필요하다.

관측 모델은 로봇 상태벡터 \mathbf{x}_k 와 랜드마크 집합 \mathcal{M} 이 주어졌을 때 관측 벡터 \mathbf{z}_k 의 확률 분포를 의미한다. 이는 다음과 같은 확률식으로 기술된다.

$$P(\mathbf{z}_k | \mathbf{x}_k, \mathcal{M}) \quad (4)$$

로봇의 움직임 모델(motion model)은 다음과 같은 상태공간 모델의 확률분포로서 기술할 수 있다.

$$P(\mathbf{x}_k | \mathbf{x}_{k-1}, \mathbf{u}_{k-1}) \quad (5)$$

위 식은 상태 천이가 Markov 과정임을 의미한다. 즉, 현재 상태 \mathbf{x}_k 는 오직 이전 상태 \mathbf{x}_{k-1} 과 입력 벡터 \mathbf{u}_{k-1} 로서 기술된다. 또한 랜드마크 집합과 관측에 독립이다.

이제 SLAM 문제는 일반적인 두 단계의 반복으로 구현할 수 있다. 바로 예측(prediction, time-update)과 보정(correction, measurement-update) 단계이다. 예측단계는,

Time-update

$$P(\mathbf{x}_{k+1}, \mathcal{M} | \mathcal{Z}_{1:k}, U_{0:k}, \mathbf{x}_0) = \int P(\mathbf{x}_{k+1} | \mathbf{x}_k, \mathbf{u}_k) P(\mathbf{x}_k, \mathcal{M} | \mathcal{Z}_{1:k}, U_{0:k-1}, \mathbf{x}_0) d\mathbf{x}_k \quad (6)$$

로 구성된다. $P(\mathbf{x}_k, \mathcal{M} | \mathcal{Z}_{1:k}, U_{0:k-1}, \mathbf{x}_0)$ 는 사전 확률분포(a priori probability distribution)로서 로봇의 움직임 모델을 통하여 구해진다. 보정 단계는 측정치 \mathbf{z}_k 가 가능해졌을 때 확률 분포를 다음과 같이 구한다.

Measurement-update

$$P(\mathbf{x}_{k+1}, \mathcal{M} | \mathcal{Z}_{1:k+1}, U_{0:k}, \mathbf{x}_0) = \frac{P(\mathbf{z}_{k+1} | \mathbf{x}_{k+1}, \mathcal{M}) P(\mathbf{x}_{k+1}, \mathcal{M} | \mathcal{Z}_{1:k}, U_{0:k}, \mathbf{x}_0)}{P(\mathbf{z}_{k+1} | \mathcal{Z}_{1:k}, U_{0:k})} \quad (7)$$

위 식은 누적된 관측 벡터 $\mathcal{Z}_{1:k+1}$ 와 제어 입력 $U_{0:k}$ 를 이용하여 결합 확률분포 $P(\mathbf{x}_{k+1}, \mathcal{M} | \mathcal{Z}_{1:k+1}, U_{0:k}, \mathbf{x}_0)$ 를 구한다.

이로부터 로봇 상태 \mathbf{x}_k 와 맵 벡터 \mathcal{M} 을 동시에 구한다.

3. SLAM 문제의 해법

3.1 Extended Kalman filter based SLAM

Vehicle motion을 다음과 같은 확률 분포로 설정한다 [7].

$$P(\mathbf{x}_{k+1} | \mathbf{x}_k, \mathbf{u}_k) \Leftrightarrow \mathbf{x}_{k+1} = \mathbf{f}(\mathbf{x}_k, \mathbf{u}_k, \mathbf{w}_k) \quad (8)$$

여기에서 $\mathbf{f}(\cdot)$ 은 로봇의 기구학 모델을, \mathbf{w}_k 는 공분산이 \mathbf{Q}_k 이고 평균이 0인 가우시안 잡음이다. 관측 모델은

$$P(\mathbf{z}_k | \mathbf{x}_k, \mathcal{M}) \Leftrightarrow \mathbf{z}_k = \mathbf{h}(\mathbf{x}_k, \mathcal{M}, \mathbf{v}_k) \quad (9)$$

와 같이 기술되며 $\mathbf{h}(\cdot)$ 은 상태벡터와 관측의 관계를 나타낸다. \mathbf{v}_k 는 covariance가 \mathbf{R}_k 인 가우시안 잡음이다. 기구학 모델과 관측모델의 비선형 식 안에 잡음이 포함된 이유는 vehicle motion equation의 경우 외부에서 관측가능한 잡음을 모델링 하는 것 보다 명령을 내린 제어추종 값을 지정된 시간안에 tracking 하지 못하는 것에서 오차가 발생한다고 보기 때문이다. 측정 잡음 역시 마찬가지로, 로봇 상태의 불확실성과 복잡한 센서 모델 내부에서 발생한다고 보는 것이다.

EKF를 통하여 $P(\mathbf{x}_k, \mathcal{M} | \mathcal{Z}_{1:k}, U_{0:k-1}, \mathbf{x}_0)$ 의 기대값

$$\begin{bmatrix} \hat{\mathbf{x}}_{k|k} \\ \hat{\mathbf{m}}_k \end{bmatrix} = E \left[\begin{bmatrix} \mathbf{x}_k \\ \mathcal{M} \end{bmatrix} \middle| \mathcal{Z}_{1:k} \right] \quad (10)$$

와 covariance

$$\begin{aligned} \mathbf{P}_{k|k} &= \begin{bmatrix} \mathbf{P}_{\mathbf{x}\mathbf{x}} & \mathbf{P}_{\mathbf{x}\mathbf{m}} \\ \mathbf{P}_{\mathbf{x}\mathbf{m}}^T & \mathbf{P}_{\mathbf{m}\mathbf{m}} \end{bmatrix}_{k|k} \\ &= E \left[\begin{pmatrix} \mathbf{x}_k - \hat{\mathbf{x}}_k \\ \mathcal{M} - \hat{\mathbf{m}}_k \end{pmatrix} \begin{pmatrix} \mathbf{x}_k - \hat{\mathbf{x}}_k \\ \mathcal{M} - \hat{\mathbf{m}}_k \end{pmatrix}^T \middle| \mathcal{Z}_{1:k} \right] \end{aligned}$$

를 계산한다. 여기서 $\hat{\mathbf{m}}_k$ 는 측정 값을 통하여 구한 랜드마크의 위치이다. EKF의 시간갱신(time-update)은

$$\begin{aligned} \hat{\mathbf{x}}_{k+1|k} &= \mathbf{f}(\hat{\mathbf{x}}_{k|k}, \mathbf{u}_k) \\ \mathbf{P}_{\mathbf{x}\mathbf{x}, k+1|k} &= \nabla \mathbf{f}_{\mathbf{x}} \mathbf{P}_{\mathbf{x}\mathbf{x}, k|k} \nabla \mathbf{f}_{\mathbf{x}}^T + \nabla \mathbf{f}_{\mathbf{u}} \mathbf{Q}_k \nabla \mathbf{f}_{\mathbf{u}}^T \end{aligned}$$

이다. $\nabla \mathbf{f}_{\mathbf{x}}$ 는 \mathbf{f} 를 $\hat{\mathbf{x}}_{k|k}$ 에 대하여 편미분 한 결과이고 $\nabla \mathbf{f}_{\mathbf{u}}$ 는 \mathbf{f} 를 \mathbf{u}_k 에 대하여 편미분 한 결과이다. Map vector \mathcal{M} 은 시불변이라고 가정하면 시간 갱신할 필요가 없다. 움직이는 랜드마크에 대한 처리는 본고를 벗어나므로 다루지 않는다.

EKF의 관측-갱신(observation-update)은

$$\begin{aligned} \begin{bmatrix} \hat{\mathbf{x}}_{k+1|k+1} \\ \hat{\mathbf{m}}_{k+1} \end{bmatrix} &= \begin{bmatrix} \hat{\mathbf{x}}_{k+1|k} \\ \hat{\mathbf{m}}_k \end{bmatrix} + \mathbf{K}_{k+1} \left[\mathbf{z}_{k+1} - \mathbf{h} \left(\hat{\mathbf{x}}_{k+1|k}, \mathcal{M}_k \right) \right] \\ \mathbf{P}_{k+1|k+1} &= \mathbf{P}_{k+1|k} - \mathbf{K}_{k+1} \mathbf{S}_{k+1} \mathbf{K}_{k+1}^T \\ \mathbf{S}_{k+1} &= \nabla \mathbf{h}_x \mathbf{P}_{k+1|k} \nabla \mathbf{h}_x^T + \mathbf{R}_{k+1} \\ \mathbf{K}_{k+1} &= \mathbf{P}_{k+1|k} \nabla \mathbf{h}_x^T \mathbf{S}_{k+1}^{-1} \end{aligned}$$

과 같다. $\nabla \mathbf{h}_x$ 는 \mathbf{h} 를 $\hat{\mathbf{x}}_{k+1|k}$ 과 $\hat{\mathbf{m}}_k$ 에 대하여 편미분 한 결과이다.

4. SLAM 문제의 구현

4.1 로봇 및 센서 모델링

본 절에서는 간단한 이동 로봇 모델을 통하여 SLAM의 구현에 대해서 알아본다. 그림 3에 나타난 로봇은 제어입력 벡터 $\mathbf{u}_k = [v_k \ \omega_k]^T$ 를 적용하여 이동되며 이는 각각 로봇의 중심속도와 조향 입력을 의미한다. 이 로봇의 프로세스 모델은 식 (11)과 같다.

$$\begin{bmatrix} \dot{x}_c \\ \dot{y}_c \\ \dot{\theta} \end{bmatrix} = \begin{bmatrix} v \cos \theta \\ v \sin \theta \\ \frac{v}{L} \tan \omega \end{bmatrix} \quad (11)$$

Index k 는 프로세스 모델식의 유도 편의상 생략한다. 관찰정보는 레이저 스캐너로부터 측정되므로 로봇 좌표는 레이저스캐너 중심으로 변환되어야 한다. 따라서 2차원 공간에서 위 프로세스 모델에 의한 좌표 갱신은

$$\begin{bmatrix} x_l \\ y_l \end{bmatrix} = \begin{bmatrix} x_c + a \cos \theta - b \sin \theta \\ y_c + a \sin \theta + b \cos \theta \end{bmatrix} \quad (12)$$

의 2차 회전변환의 형태를 갖는다. 이것의 미분값은

$$\begin{bmatrix} \dot{x}_l \\ \dot{y}_l \end{bmatrix} = \begin{bmatrix} \dot{x}_c - (a \sin \theta + b \cos \theta) \dot{\theta} \\ \dot{y}_c + (a \cos \theta - b \sin \theta) \dot{\theta} \end{bmatrix} \quad (13)$$

따라서 그림 3에 나타난 로봇의 상태공간 모델은 다음과 같다.

$$\mathbf{x}_{k+1} = \mathbf{f}_v(\mathbf{x}_k, \mathbf{u}_k) = \begin{bmatrix} x_{k+1} \\ y_{k+1} \\ \theta_{k+1} \end{bmatrix} = \begin{bmatrix} f_1 \\ f_2 \\ f_3 \end{bmatrix}$$

여기에서 f_1, f_2, f_3 는 각각

$$\begin{aligned} f_1 &= x_k + \Delta t \cdot v_k \cos \theta_k \\ &\quad - \Delta t \frac{v_k}{L} \tan \omega_k \cdot (a \sin \theta_k + b \cos \theta_k) \\ f_2 &= y_k + \Delta t \cdot v_k \sin \theta_k \\ &\quad + \Delta t \frac{v_k}{L} \tan \omega_k \cdot (a \cos \theta_k - b \sin \theta_k) \\ f_3 &= \theta_k + \frac{v_k}{L} \tan \omega_k \end{aligned}$$

을 의미한다. 이 식에서 Δt 는 샘플링 시간을 의미한다.

이동로봇의 센서는 거리와 각도를 측정할 수 있는(range and bearing sensor) 레이저 스캐너로 가정한다. 랜드마크는 2차원 위치를 가지며 다음과 같다.

$$\mathbf{m}_i = \begin{bmatrix} m_{i,x} \\ m_{i,y} \end{bmatrix} \quad (14)$$

측정 모델 $\mathbf{z}_{i,k}$ 는 k 시점에서 관측한 i 번째 랜드마크의 관측 벡터이다. 이는 다음과 같다.

$$\begin{aligned} \mathbf{z}_{i,k} = \mathbf{h}_i(\mathbf{x}_k, \mathcal{M}) &= \begin{bmatrix} r_i \\ \phi_i \end{bmatrix} \\ &= \begin{bmatrix} \sqrt{(m_{i,x} - x_k)^2 + (m_{i,y} - y_k)^2} \\ \tan^{-1} \left(\frac{m_{i,y} - y_k}{m_{i,x} - x_k} \right) - \theta_k + \frac{\pi}{2} \end{bmatrix} = \begin{bmatrix} h_1 \\ h_2 \end{bmatrix} \end{aligned}$$

$\frac{\pi}{2}$ 를 더해준 것은 레이저 스캐너의 경우 그림 4에서 좌표계 (x_l, y_l) 에서 y_l 축을 시작으로 회전하면서 측정하기때문에 좌표계가 세계 좌표계와 90도 회전을 이루고 있다고 볼 수 있다. 따라서 $\frac{\pi}{2}$ 를 더해줌으로서 축을 일치시킬 수 있다.

4.2 SLAM

4.2.1 결합 SLAM 벡터 설정

시간 k 에서 결합 SLAM 상태벡터(joint SLAM state vector)를 다음과 같이 표현하자.

$$\begin{aligned} \tilde{\mathbf{x}}_{k+1} &= \mathbf{f}(\tilde{\mathbf{x}}_k, \mathbf{u}_k) \\ &= \begin{bmatrix} \mathbf{f}_v(\mathbf{x}_k, \mathbf{u}_k) \\ \hat{\mathbf{m}} \end{bmatrix} \end{aligned}$$

이는 기존의 로봇 상태벡터 \mathbf{x}_k 에 map vector $\hat{\mathbf{m}}$ 을 확장시킨 것으로 $\tilde{\mathbf{x}}_k$ 로 표현한다.

4.2.2 Prediction stage

Motion model에 대한 prediction 은

$$\tilde{\mathbf{x}}_{k+1|k} = \mathbf{f}(\tilde{\mathbf{x}}_{k|k}, \mathbf{u}_k)$$

이다. Covariance의 time-update는 다음의 식으로 얻어진다.

$$\mathbf{P}_{k+1|k} = \nabla \mathbf{f}_{\tilde{\mathbf{x}}} \mathbf{P}_{k|k} \nabla \mathbf{f}_{\tilde{\mathbf{x}}}^T + \nabla \mathbf{f}_{\mathbf{u}} \mathbf{Q}_k \nabla \mathbf{f}_{\mathbf{u}}^T \quad (15)$$

EKF의 적용을 위하여 이동 로봇 모델의 자코비안을 다음과 같이 구한다. 결합 SLAM 상태벡터 $\tilde{\mathbf{x}}_k$ 에 대한 자코비안 $\nabla \mathbf{f}_{\tilde{\mathbf{x}}}$ 는 다음과 같이 구성된다.

$$\nabla \mathbf{f}_{\tilde{\mathbf{x}}} = \frac{\partial \mathbf{f}}{\partial \tilde{\mathbf{x}}_{k|k}} = \begin{bmatrix} \frac{\partial f_1}{\partial(x,y,\theta,\mathbf{m}_i)} \\ \frac{\partial f_2}{\partial(x,y,\theta,\mathbf{m}_i)} \\ \frac{\partial f_3}{\partial(x,y,\theta,\mathbf{m}_i)} \end{bmatrix} = \begin{bmatrix} \nabla \mathbf{f}_{v_x} & \mathbf{0}_{vm} \\ \mathbf{0}_{vm}^T & \mathbf{I}_m \end{bmatrix}$$

이 식에서 $\mathbf{0}_{vm}$ 은 $\mathbb{R}^{dim(\mathbf{x}) \times dim(\mathbf{m})}$ 의 영행렬이다. \mathbf{I}_m 은 $\mathbb{R}^{dim(\mathbf{m}) \times dim(\mathbf{m})}$ 이다. 각각의 자코비안은 다음과 같다.

$$\begin{aligned} \nabla \mathbf{f}_{v_x} &= \frac{\partial \mathbf{f}_v}{\partial \mathbf{x}_{k|k}} = \begin{bmatrix} \frac{\partial f_1}{\partial x} & \frac{\partial f_1}{\partial y} & \frac{\partial f_1}{\partial \theta} \\ \frac{\partial f_2}{\partial x} & \frac{\partial f_2}{\partial y} & \frac{\partial f_2}{\partial \theta} \\ \frac{\partial f_3}{\partial x} & \frac{\partial f_3}{\partial y} & \frac{\partial f_3}{\partial \theta} \end{bmatrix} \\ &= \begin{bmatrix} 1 & 0 & \Phi_{13} \\ 0 & 1 & \Phi_{23} \\ 0 & 0 & 1 \end{bmatrix} \\ \nabla \mathbf{f}_{v_u} &= \frac{\partial \mathbf{f}_v}{\partial \mathbf{u}_k} = \begin{bmatrix} \frac{\partial f_1}{\partial v} & \frac{\partial f_1}{\partial \omega} \\ \frac{\partial f_2}{\partial v} & \frac{\partial f_2}{\partial \omega} \\ \frac{\partial f_3}{\partial v} & \frac{\partial f_3}{\partial \omega} \end{bmatrix} \\ &= \begin{bmatrix} \Psi_{11} & \Psi_{12} \\ \Psi_{21} & \Psi_{22} \\ \frac{\tan \omega}{L} & \frac{v(1+\tan^2 \omega)}{L} \end{bmatrix} \end{aligned}$$

여기에서,

$$\begin{aligned} \Phi_{13} &= -\Delta t \cdot v_k \sin \theta_k \\ &\quad -\Delta t \frac{v_k}{L} \tan \omega_k (a \cos \theta_k - b \sin \theta_k) \\ \Phi_{23} &= \Delta t \cdot v_k \cos \theta_k \\ &\quad +\Delta t \frac{v_k}{L} \tan \omega_k (-a \sin \theta_k - b \cos \theta_k) \end{aligned}$$

이고,

$$\begin{aligned} \Psi_{11} &= \Delta t \cdot \cos \theta_k - \frac{\Delta t}{L} \tan \omega_k (a \sin \theta_k + b \cos \theta_k) \\ \Psi_{12} &= -\Delta t \cdot \frac{v_k}{L} \cdot (1 + \tan^2 \omega_k) (a \sin \theta_k + b \cos \theta_k) \\ \Psi_{21} &= \Delta t \cdot \sin \theta_k + \frac{\Delta t}{L} \tan \omega_k (a \cos \theta_k + b \sin \theta_k) \\ \Psi_{22} &= \Delta t \cdot \frac{v_k}{L} \cdot (1 + \tan^2 \omega_k) (a \cos \theta_k + b \sin \theta_k) \end{aligned}$$

이다. 즉, map vector는 로봇 상태 벡터를 편미분 했을 때 영벡터 이므로 다음과 같이 covariance update를 나타낼 수 있다.

$$\mathbf{P}_{k+1|k} = \begin{bmatrix} \nabla \mathbf{f}_{v_x} \mathbf{P}_{vv} \nabla \mathbf{f}_{v_x}^T + \nabla \mathbf{f}_{v_u} \mathbf{Q}_k \nabla \mathbf{f}_{v_u}^T & \nabla \mathbf{f}_{v_x} \mathbf{P}_{vm} \\ \mathbf{P}_{vm}^T \nabla \mathbf{f}_{v_x}^T & \mathbf{P}_{mm} \end{bmatrix}$$

4.2.3 Update stage

이어서 측정 모델의 자코비안을 구한다. 측정모델의 자코비안 $\nabla \mathbf{h}_{\tilde{\mathbf{x}}_a}$ 는 다음과 같다.

$$\begin{aligned} \nabla \mathbf{h}_{\tilde{\mathbf{x}}_a} &= \frac{\partial \mathbf{h}_i}{\partial \tilde{\mathbf{x}}_{a,k+1|k}} \\ &= \begin{bmatrix} \frac{\partial h_1}{\partial(x,y,\theta,\mathbf{m}_i)} \\ \frac{\partial h_2}{\partial(x,y,\theta,\mathbf{m}_i)} \end{bmatrix} = \begin{bmatrix} \nabla \mathbf{h}_{v_x} & \cdots & \nabla \mathbf{h}_{m_i} & \cdots \end{bmatrix} \\ \nabla \mathbf{h}_{v_x} &= \frac{\partial \mathbf{h}_i}{\partial \tilde{\mathbf{x}}_{k+1|k}} = \begin{bmatrix} \frac{\partial h_1}{\partial x} & \frac{\partial h_1}{\partial y} & \frac{\partial h_1}{\partial \theta} \\ \frac{\partial h_2}{\partial x} & \frac{\partial h_2}{\partial y} & \frac{\partial h_2}{\partial \theta} \end{bmatrix} \\ &= \begin{bmatrix} \frac{x-m_{i,x}}{r_i} & \frac{y-m_{i,y}}{r_i} & 0 \\ \frac{m_{i,y}-y}{r_i^2} & \frac{x-m_{i,x}}{r_i^2} & -1 \end{bmatrix} \\ \nabla \mathbf{h}_{m_i} &= \frac{\partial \mathbf{h}_i}{\partial \mathbf{m}_i} = \begin{bmatrix} \frac{x-m_{i,x}}{r_i} & \frac{y-m_{i,y}}{r_i} \\ -\frac{(y-m_{i,y})}{r_i^2} & \frac{x-m_{i,x}}{r_i^2} \end{bmatrix} \end{aligned}$$

이 식에서 $r_i = \sqrt{(m_{i,x} - x)^2 + (m_{i,y} - y)^2}$ 이다. 로봇과 map vector의 사후확률은 다음과 같이 계산된다.

$$\begin{aligned} \tilde{\mathbf{x}}_{k+1|k+1} &= \tilde{\mathbf{x}}_{k+1|k} + \mathbf{K}_{k+1} [\mathbf{z}_{k+1} - \mathbf{h}(\tilde{\mathbf{x}}_{k+1|k})] \\ \mathbf{P}_{k+1|k+1} &= \mathbf{P}_{k+1|k} - \mathbf{K}_{k+1} \mathbf{S}_{k+1} \mathbf{K}_{k+1}^T \\ \mathbf{S}_{k+1} &= \nabla \mathbf{h}_{\tilde{\mathbf{x}}_a} \mathbf{P}_{k+1|k} \nabla \mathbf{h}_{\tilde{\mathbf{x}}_a}^T + \mathbf{R}_{k+1} \\ \mathbf{K}_{k+1} &= \mathbf{P}_{k+1|k} \nabla \mathbf{h}_{\tilde{\mathbf{x}}_a} \mathbf{S}_{k+1}^{-1} \end{aligned}$$

4.2.4 State augmentation

기존에 관찰되지 않았던 새로운 랜드마크가 관찰되면 이를 적절히 처리한 후 SLAM 상태벡터에 추가시켜 주어야 한다. 먼저 극좌표에서 정의된 측정값과 측정값의 covariance를 마지막 행에 추가하여 확장한 다음의 상태변수를 고려하자.

$$\begin{aligned} \hat{\mathbf{x}}_{aug} &= \begin{bmatrix} \tilde{\mathbf{x}}_k \\ \mathbf{z} \end{bmatrix} \\ \mathbf{P}_{aug} &= \begin{bmatrix} \mathbf{P}_{vv} & \mathbf{P}_{vm} & \mathbf{0} \\ \mathbf{P}_{vm}^T & \mathbf{P}_m & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{R} \end{bmatrix} \end{aligned}$$

관측된 거리, 방위각 정보는 다음과 같은 변환을 통하여 world coordinates로 등록(registration)된다.

$$\begin{bmatrix} m_{x,i} \\ m_{y,i} \end{bmatrix} = \mathbf{g}_i(\mathbf{x}_k, \mathbf{z}) = \begin{bmatrix} x_k + r_i \cdot \cos(\phi_i + \theta_k - \frac{\pi}{2}) \\ y_k + r_i \cdot \sin(\phi_i + \theta_k - \frac{\pi}{2}) \end{bmatrix}$$

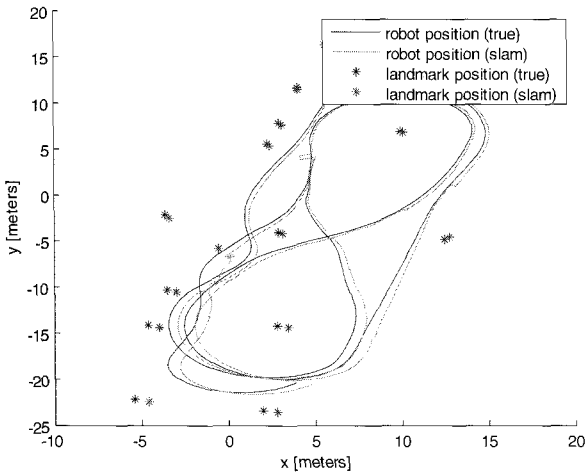


그림 6. (5)에서 기록한 로봇의 상태 자료를 기반으로 SLAM을 수행한 모습

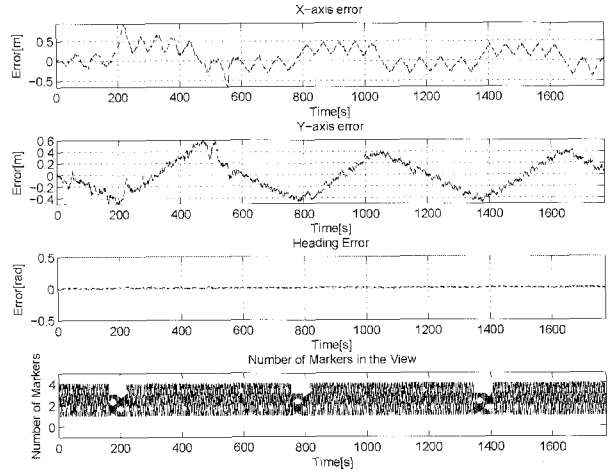
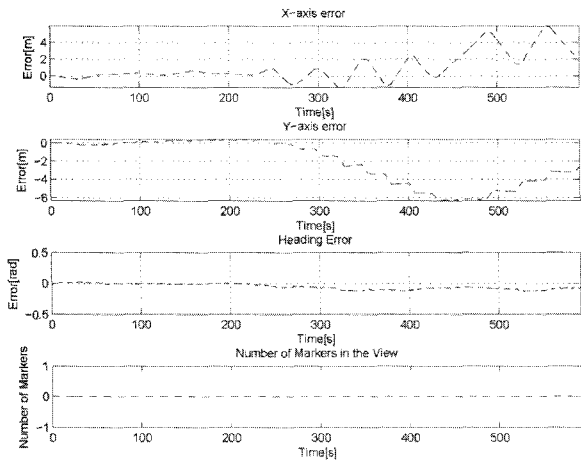
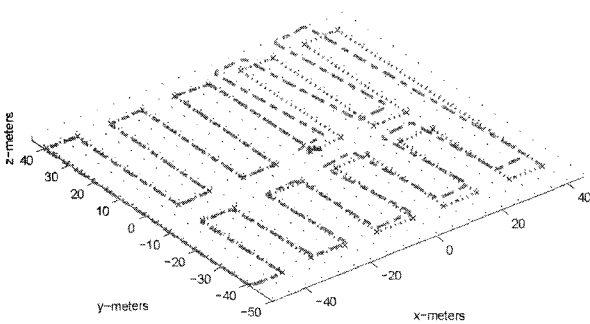
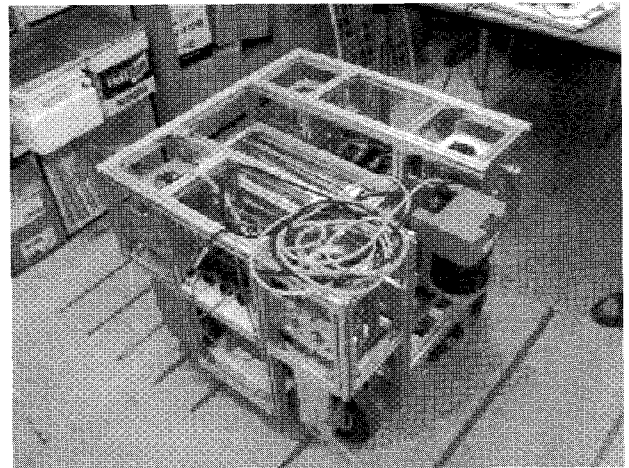


그림 8. 한 프레임에 fiducial marker가 2개 이상인 경우의 오차 그래프



(a) 오차가 계속 증가하고 있다



(b) 추정된 경로(dash-dot)와 ground truth 경로(dotted)

그림 7. (4)에서 주행거리계 정보만 이용하여 추정된 결과

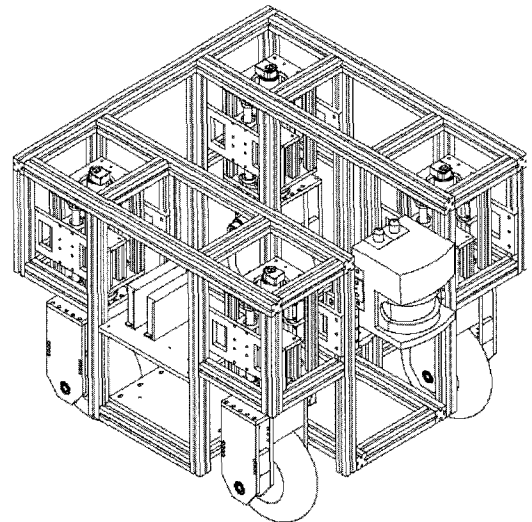


그림 9. (4)에서 대상으로 하는 모바일 로봇 (위: 실제사진, 아래: CAD 모델)

새롭게 관찰된 랜드마크의 관찰값 $z_{i,k}$ 를 이용하여 다음과 같은 SLAM 상태벡터 확장을 계산한다.

$$\tilde{\mathbf{x}}_k^+ = \mathbf{f}_i(\hat{\mathbf{x}}_{aug}) = \begin{bmatrix} \tilde{\mathbf{x}}_k \\ \mathbf{g}_i(\mathbf{x}_k, \mathbf{z}) \end{bmatrix}$$

covariance update는

$$\mathbf{P}_a = \nabla \mathbf{f}_{\mathbf{x}_{aug}} \mathbf{P}_{aug} \nabla \mathbf{f}_{\mathbf{x}_{aug}}^T$$

와 같다. 자코비안 $\nabla \mathbf{f}_{\mathbf{x}_{aug}}$ 는 다음과 같이 정의 할 수 있다.

$$\nabla \mathbf{f}_{\mathbf{x}_{aug}} = \frac{\partial \mathbf{f}_i}{\partial \mathbf{x}_{aug}} = \begin{bmatrix} \mathbf{I}_v & 0 & 0 \\ 0 & \mathbf{I}_m & 0 \\ \nabla \mathbf{g}_{\mathbf{x}_v} & 0 & \nabla \mathbf{g}_{\mathbf{z}} \end{bmatrix}$$

이 식에서 자코비안 $\nabla \mathbf{g}_{\mathbf{x}_v}$, $\nabla \mathbf{g}_{\mathbf{z}}$ 는 각각 다음과 같다.

$$\nabla \mathbf{g}_{\mathbf{x}_v} = \frac{\partial \mathbf{g}_i}{\partial \mathbf{x}_v} = \begin{bmatrix} 1 & 0 & -r_i \cdot \sin(\phi_i + \theta_k - \frac{\pi}{2}) \\ 0 & 1 & r_i \cos(\phi_i + \theta_k - \frac{\pi}{2}) \end{bmatrix}$$

$$\nabla \mathbf{g}_{\mathbf{z}} = \frac{\partial \mathbf{g}_i}{\partial \mathbf{z}} = \begin{bmatrix} \cos(\theta_k + \phi_i) & -r \sin(\theta_k + \phi_i) \\ \sin(\theta_k + \phi_i) & r \cos(\theta_k + \phi_i) \end{bmatrix}$$

5. Simulation Result

5.1 Laser scanner

실제 실험을 위해서 그림 5와 같은 차량을 이용하여 취득된 레이저 스캐너 정보와 차량의 엔코더 신호를 이용하였다. 이 데이터는 [5]에서 취득되었으며 SLAM 연구를 위하여 저자가 제공하고 있다. 데이터는 MATLAB의 mat 형식으로 저장되어 있으며 총 112초 동안 기록되어 있다. 이 차량의 모델과 센서모델은 4절에 나타나 있다. 실험 결과는 그림 6과 같다.

Ground truth 데이터는 GPS 를 이용하여 취득되었다. 그림 6은 GPS 데이터와 SLAM 을 이용하여 추정된 경로 데이터를 겹쳐 그린 것이다.

5.2 Monocular vision

레이저 스캐너는 거리와 방위각이 반환되는 이점이 있어 SLAM 에 많이 사용되어 왔다. 하지만 최근에는 고가의 레이저 스캐너 대신 상대적으로 저가의 카메라를 이용하여 SLAM 을 수행하려는 연구가 진행되어 왔다. [4]에서는 Laser scanner 대신 한 대의 카메라를 이용한 SLAM이 제안되었다. 본 절에서는 카메라를 센서로 이용했을 경우에 대한 SLAM에 대하여 소개한다. 그림 7은 [4]에서 그림 9에 대한 로봇 모델로 모의 실험한 결과로 엔코더 값에 의한 로봇의 위치추정시 오차를 보여주고 있다. 실험 시간이 증가함에 따라 로봇의 위치정확도는 매우 낮아

지고 있다.

그림 8은 [4]에서 제안한 영상 센서 모델에 대하여 모의 실험을 수행한 결과이다. 실내에서 0.5m 범위의 오차의 로봇 위치추정이 가능한 것을 볼 수 있다.

6. 결론

본 고에서는 이동로봇의 SLAM에 대하여 소개 하였다. 전달의 명확성을 위해서 위치인식 문제와 지도작성 문제를 각각 정의하고 이를 결합한 SLAM 에 대하여 확률적으로 기술하였다. 칼만 필터와 Bayes 필터의 연관성에 대해서도 부록에 기술 하였다. 본 고에서 소개한 방법은 EKF를 이용한 방법이었으나, 다른 방법으로 non-parametric 방법인 Particle filter나 선형화 과정을 통계적 방법으로 변환하여 칼만 필터를 이용하는 Unscented Kalman filter 등이 이용 될 수 있다.

Derivation of the Bayes Filter

본 절에서는 조건부 베이스 규칙에 대하여 유도하는 과정을 설명한다. 조건부 베이스 규칙(conditional Bayes rule)은 다음과 같다.

$$P(x|y, z) = \frac{P(y|x, z) P(x|z)}{P(y|z)}$$

이는 y 와 z 가 주어졌을 때 x 의 확률 분포를 의미한다. 이는 SLAM에서 다음과 같이 이용 할 수 있다.

$$\begin{aligned} & P(\mathbf{x}_k, \mathcal{M} | \mathcal{Z}_{1:k}, \mathcal{U}_{0:k-1}, \mathbf{x}_0) \\ &= P(\mathbf{x}_k, \mathcal{M} | \mathbf{z}_k, \mathcal{Z}_{1:k-1}, \mathcal{U}_{0:k-1}, \mathbf{x}_0) \\ &= P(\mathbf{z}_k | \mathbf{x}_k, \mathcal{M}, \mathcal{Z}_{1:k-1}, \mathcal{U}_{0:k-1}, \mathbf{x}_0) \\ &\quad \times \frac{P(\mathbf{x}_k, \mathcal{M} | \mathcal{Z}_{1:k-1}, \mathcal{U}_{0:k-1}, \mathbf{x}_0)}{P(\mathbf{z}_k | \mathcal{Z}_{1:k-1}, \mathcal{U}_{0:k-1}, \mathbf{x}_0)} \end{aligned} \quad (16)$$

상태 \mathbf{x}_k 와 \mathbf{z}_k 는 Markov 과정이므로, 다음을 만족한다.

$$\begin{aligned} P(\mathbf{x}_k | \mathbf{x}_{k-1}, \mathbf{x}_{k-2}, \dots, \mathbf{x}_0) &= P(\mathbf{x}_k | \mathbf{x}_{k-1}) \\ P(\mathbf{z}_k | \mathbf{x}_k, \mathbf{x}_{k-1}, \dots, \mathbf{x}_0) &= P(\mathbf{z}_k | \mathbf{x}_k) \end{aligned}$$

따라서 (16)에서 $P(\mathbf{z}_k | \mathbf{x}_k, \mathcal{M}, \mathcal{Z}_{1:k-1}, \mathcal{U}_{1:k}, \mathbf{x}_0)$ 는 $P(\mathbf{z}_k | \mathbf{x}_k, \mathcal{M})$ 이 된다. 마찬가지로, $P(\mathbf{z}_k | \mathbf{x}_k, \mathcal{M}, \mathcal{Z}_{1:k-1}, \mathcal{U}_{1:k}, \mathbf{x}_0)$ 는 $P(\mathbf{z}_k | \mathcal{Z}_{1:k-1}, \mathcal{U}_{1:k})$ 이 된다. 정리하면 (7)이 된다. (7)은 사후 확률분포(posterior probability distribution)을 의미하며, 물리적으로는 외부 환경에 대한 관측이 이루어진 후의 내부 상태 \mathbf{x}_k 의

확률 분포 갱신을 의미한다. $P(\mathbf{x}_k, \mathcal{M} | \mathcal{Z}_{1:k}, \mathcal{U}_{0:k-1}, \mathbf{x}_0)$ 를 구하기 위해서 total probability theorem을 이용한다.

$$P(\mathbf{x}_k, \mathcal{M} | \mathcal{Z}_{1:k}, \mathcal{U}_{0:k-1}, \mathbf{x}_0) = \int P(\mathbf{x}_k | \mathbf{x}_{k-1}, \mathcal{Z}_{1:k}, \mathcal{U}_{0:k-1}) \times P(\mathbf{x}_{k-1}, \mathcal{M} | \mathcal{Z}_{1:k-1}, \mathcal{U}_{0:k-2}, \mathbf{x}_0) d\mathbf{x}_{k-1}$$

상태는 Markov process이므로

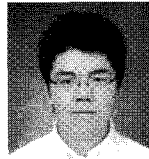
$$P(\mathbf{x}_k | \mathbf{x}_{k-1}, \mathcal{Z}_{1:k}, \mathcal{U}_{0:k-1}) = P(\mathbf{x}_k | \mathbf{x}_{k-1}, \mathbf{u}_{k-1})$$

따라서, (6)과 같이 된다.

참고문헌

- [1] "Stargazor™," Hagisonic, South Korea, Daejeon. [Online]. Available: <http://www.hagisonic.com/>
- [2] "U-sat™," Korea LPS, South Korea, Pusan. [Online]. Available: http://korealps.co.kr/bbs/products_model.html
- [3] "Evolution robotics. northstar." [Online]. Available: <http://www.evolution.com/products/northstar/>
- [4] 임현, 양지혁, 이영삼, 김진걸, "한 대의 카메라와 Fiducial 마커를 이용한 SLAM", 제어·로봇·시스템학회 논문지 제 15권, 제4호, 2009. 04.
- [5] J. Guivant, E. Nebot, and S. Baiker, "Autonomous navigation and map building using laser range sensors in outdoor applications," *Journal of Robotic Systems*, vol. 17, pp. 3817-3822, 2000.
- [6] Durrant-Whyte, H.; Bailey, T., "Simultaneous localization and mapping: part I," *Robotics & Automation Magazine, IEEE*, vol. 13, no. 2, pp. 99-110, June 2006.
- [7] Bailey, T.; Durrant-Whyte, H., "Simultaneous localization and mapping (SLAM): part II," *Robotics & Automation Magazine, IEEE*, vol. 13, no. 3, pp. 108-117, Sept. 2006.

저자약력



임현

- 2008년 인하대학교 전자공학 공학사
- 2008년~현재 인하대학교 대학원 전기공학과 석사과정 재학중
- 관심분야 : SLAM, Optimal Estimation, Structure from Motion(SfM), 모바일로봇



이영삼

- 1997년 인하대학교 전기공학 공학사
- 1999년 인하대학교 전기공학과 석사
- 2003년 서울대학교 전기컴퓨터공학부 (박사)
- 2004년~현재 인하대학교 전기공학부 조교수
- 관심분야 : 임베디드 시스템, 디지털 제어, CACSD (Computer-aided Control System Design) 소프트웨어