

추계적 페트리넷을 통한 동적 환경에서의 지능적인 환경정보의 갱신

Intelligent Update of Environment Model in Dynamic Environments through Generalized Stochastic Petri Net

박중태*, 이용주**, 송재복**

Joong-Tae Park, Yong-Ju Lee, Jae-Bok Song

Abstract - This paper proposes an intelligent decision framework for update of the environment model using GSPN(generalized stochastic petri nets). The GSPN has several advantages over direct use of the Markov Process. The modeling, analysis, and performance evaluation are conducted on the mathematical basis. By adopting the probabilistic approach, our decision framework helps the robot to decide the time to update the map. The robot navigates autonomously for a long time in dynamic environments. Experimental results show that the proposed scheme is useful for service robots which work semi-permanently and improves dependability of navigation in dynamic environments.

Key Words : Map update, Generalized stochastic petri net (GSPN), decision framework

1. 서론

인간과 공존하며 유용한 서비스를 제공하는 서비스 로봇에 있어서 안전하고 신뢰성 있는 자율주행은 필수적인 요소이다. 신뢰성 높은 자율주행을 위해서는 실패하지 않는 위치추정 기술이 최우선적으로 확보되어야 한다. 로봇이 자신의 위치를 알지 못하는 상태에서는, 목표지점으로 정확히 이동할 수 없기 때문이다. 따라서 로봇의 위치를 추정하기 위한 다양한 방법들이 제시되었다. 노드와 에지로 구성되어 있는 위상지도에서 각 노드에 저장되어 있는 영상의 특징을 입력영상과 비교하여, 마코프 위치추정으로 로봇이 어떤 노드에 있는지 추정하는 방법이 제안되었다[1]. 또한, 스테레오 카메라를 사용하여 SIFT(Scale Invariant Feature Transform) 특징을 추출하고 [2], 이를 이용하여 위치추정을 수행하는 방법도 제시되었다 [3].

위와 같은 위치추정 기술은 많은 로봇에 적용되어 다양한 연구 및 실험으로 그 성능이 검증되었다. 그러나 기존의 위치추정은 환경정보를 모델링한 지도와 실제 환경과의 차이가 작아야만 성공한다. 즉, 어떤 요인으로 실제 환경에 변화가 생겨 로봇이 갖고 있는 환경지도가 실제 환경과 많이 달라지면, 로봇이 위치추정에 실패할 확률이 높아진다. 만약, 로봇 스스로 변화된 환경을 인식하여, 변화된 환경정보를 기존의 지도에 반영할 수 있으면 위와 같은 문제를 해결할 수 있다. 따라서 본 논문에서는 로봇 스스로 환경의 변화를 판단하여

지도를 갱신할 수 있는 방법을 제안한다.

본 논문은 다음과 같이 구성된다. 2장에서는 물체 인식 기법인 SIFT와 거리센서를 이용해 환경의 변화를 검출할 수 있는 방법을 소개한다. 3장에서는 본 논문에서 제시한 지능적 환경정보 갱신을 위한 프레임워크에 대해 다룬다. 4장에서는 실험을 통해 본 논문에서 구현한 프레임워크의 합당성과 함께 결론을 제시한다.

2. SIFT를 이용한 변화된 환경의 검출방법

본 절에서는 가변적인 환경에서 환경의 변화를 검출할 수 있는 방법에 대해 소개한다. 로봇은 주행 중 비전센서로부터 입력되는 영상정보에서 지도에 등록된 물체를 인식하여 현재 로봇의 위치에서 인식된 물체까지의 상대거리(D_{mea})와 상대각도(θ_{mea})를 구한다. 또한, 현재로봇으로부터 인식된 물체의 지도상의 위치까지의 상대거리(D_{est})와 상대각도(θ_{est})를 구한다. 즉, θ_{est} , D_{est} 는 예측한 상대각도와 거리이고, θ_{mea} , D_{mea} 는 측정된 상대각도와 거리이다. 만약, 물체의 위치가 실제위치와 다르면, 예측치와 측정치의 차이가 많아진다. 따라서 식 (1)과 같이 두 정보의 차가 정해진 문턱치를 넘게 되면, 지도상에 등록된 물체의 위치가 변경되었다고 판단할 수 있다. 그림 1은 D_{mea} , θ_{mea} , D_{est} , θ_{est} 구하는 과정을 나타낸 것이다.

$$\begin{cases} |\theta_{est} - \theta_{mea}| > 30^\circ \\ |D_{est} - D_{mea}| > 2m \end{cases} \quad (1)$$

저자 소개

- * 박중태 : 高麗大學 機電工學科 碩士課程
- ** 이용주 : 高麗大學 機械工學科 博士課程
- ** 송재복 : 高麗大學 機械工學科 教授 · 工博

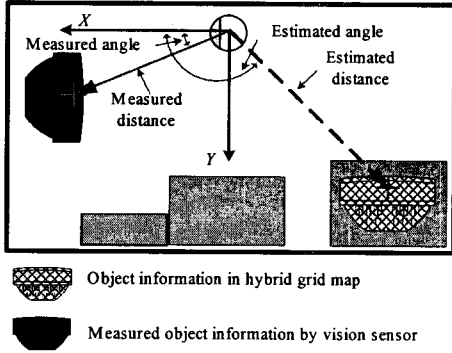


Fig. 1 Criterion for change in object position.

위와 같이 로봇은 주행 중 비전센서의 입력영상에서 인식된 물체의 위치와 지도상의 물체의 위치와의 차이를 계속해서 검사한다. 하지만 위치추정 오차 및 오정합으로 인하여 식 (1)의 조건을 만족할 수 있기 때문에, 비전센서로 들어오는 영상이 식 (1)의 조건에 일치할지라도 지도에 등록된 물체의 위치가 변경되었다고 확신할 수는 없다. 따라서 물체의 정합결과가 식 (1)의 조건에 부합하면, 이를 비정상 정합 (abnormal matching)이라 정의하고, 지도에 등록된 물체의 위치 불확실성을 증가시킨다. 이와 반대로 물체의 정합결과가 식 (1)의 조건에 부합된다면 정상정합(normal matching)으로 인식하여, 지도에 등록된 물체의 위치 정확성을 증가시킨다. 정상 정합 및 비정상 정합에 대한 정의는 표 1에 기술되어 있다.

Table 1 Description of terminology.

Normal matching	Abnormal matching
Object recognizer matches the vision feature whose position is unchanged.	Object recognizer matches the vision feature whose position is changed.

3. 지능적 환경정보 갱신 판단 프레임워크

지능적 환경정보 갱신을 위해서는 로봇 스스로 환경의 변화를 감지하여 적절한 시기에 지도에 반영시킬 수 있어야 한다. 본 절에서는 일반화된 추계적 페트리넷 (Generalized Stochastic Petri Net; GSPN)을 통해 지도갱신 판단 프레임워크를 설계하는 방법을 다룬다. GSPN은 추계적 페트리넷 (SPN)의 모든 전이를 시간과 관련된 전이와 시간에 무관한 즉시 전이로 분류한 것이다. 페트리넷에 관한 자세한 이론은 참고문헌 [4]에 소개되어 있다.

GSPN을 통해 설계한 프레임워크는 그림 2와 같으며, 표 2에 그림 2에 대한 설명이다. 그림 2에서 마킹의 처음 상태는 $M_0 = (1, 0, 1, 0, 0, 0)$ 이다. 즉, 로봇은 다음 작업을 위해 대기하는 중이며, 물체인식 또한 정상정합 상태임을 나타낸다. 이후 트랜지션 t_0 이 발생하여 P0의 토큰이 P1으로 이동하게 되면 마킹의 상태는 $M_1 = (0, 1, 1, 0, 0, 0)$ 이 되고, 로봇은 주행하기 시작한다. 그러나 t_1 이 발생하면 마킹의 상태는 $M_2 = (0, 0, 1, 0, 1, 0)$ 이 되고, 로봇은 지도갱신 작업을 수행한다. 이는 로봇의 행동은

어떠한 트랜지션이 발생했는지에 따라 달라짐을 의미한다. 따라서 정확한 지도 갱신을 위해서는 로봇 스스로 어느 시점에 트랜지션 t_i 을 발생시키는지 결정하는 것이 중요하다. 결정방법은 아래와 같이 표현된다.

$$U' = \begin{bmatrix} 0 & 0 & \frac{\lambda_4}{(\lambda_2 + \lambda_4)} & \frac{\lambda_2}{(\lambda_2 + \lambda_4)} & 0 & 0 \\ 0 & 0 & \frac{\lambda_2}{(\lambda_2 + \lambda_6)} & 0 & \frac{\lambda_6}{(\lambda_2 + \lambda_6)} & 0 \\ \frac{\lambda_7 \times p}{(\lambda_2 + \lambda_7)} & \frac{\lambda_7 \times (1-p)}{(\lambda_2 + \lambda_7)} & 0 & 0 & 0 & \frac{\lambda_2}{(\lambda_2 + \lambda_7)} \\ \frac{\lambda_3}{(\lambda_3 + \lambda_5)} & 0 & \frac{\lambda_5}{(\lambda_3 + \lambda_5)} & 0 & 0 & 0 \\ 0 & \frac{\lambda_3}{(\lambda_3 + \lambda_6)} & 0 & 0 & 0 & \frac{\lambda_6}{(\lambda_3 + \lambda_6)} \\ \frac{\lambda_7 \times p}{(\lambda_3 + \lambda_7)} & 0 & \frac{\lambda_3}{(\lambda_3 + \lambda_7)} & 0 & \frac{\lambda_7 \times (1-p)}{(\lambda_3 + \lambda_7)} & 0 \end{bmatrix} \quad (2)$$

$$Y = Y \cdot U' \quad \sum_{i=1}^5 y_i = 1 \quad (3)$$

$$\pi_i = \frac{y_i \cdot m_i}{\sum_{j=1}^5 y_j \cdot m_j} \quad \text{where, } m_i = \frac{1}{\sum_{t_i \in E(M_i)} \lambda_k} \quad (4)$$

$$f_j = \sum_{i, t_i \in E(M_i)} \lambda_j(M_i) \cdot \pi_i \quad (5)$$

$$\begin{aligned} f_{4(\text{normalstate})} &= \lambda_4 \cdot \pi_1 \\ f_{5(\text{mapupdate})} &= \lambda_5 \cdot \pi_2 \end{aligned} \quad (6)$$

여기서, f_j 는 트랜지션 t_j 의 점화율이며, $\lambda_j(M_i)$ 는 마킹 M_i 에서의 t_j 의 점화율이다. 따라서 로봇은 주행 작업을 끝낸 후 트랜지션 t_4 와 t_5 의 점화율을 구하여, t_5 의 점화율이 높으면 t_1 을 발생시켜 지도갱신 작업을 수행하면 된다. 식 (6)은 t_4 와 t_5 의 점화율을 구하는 식이다.

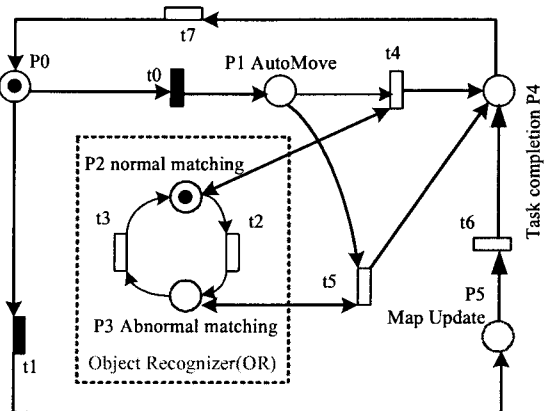


Fig. 2 Decision Framework using GSPN.

Table 2 Description for Fig. 2.

Place	Description	
P0	State: navigation available.	
P1	State: running AutoMove.	
P2(P3)	State: normal (abnormal) matching.	
P4	State: Automove completed.	
P5	State: Map update	
Transition	Description	Firing rate
t0(t1)	Start AutoMove(Map update)	-
t1	AutoMove completed	λ_1
t2(t3)	Normal(abnormal) matching.	$\lambda_2(\lambda_3)$
t4,t5	Automove complete.	λ_4, λ_5
t6	Map update complete.	λ_6
t7	Initialization.	λ_6

4. 실험 및 결론

4.1 실험

다음 실험은 주거환경 내에서 로봇 스스로 환경의 변화를 감지하여 새롭게 지도를 갱신하는 과정을 나타낸 것이다. 본 실험에서는 7m X 8m의 일반적인 실내 주거환경에서 로봇이 40분 동안 자율주행을 수행하였다.

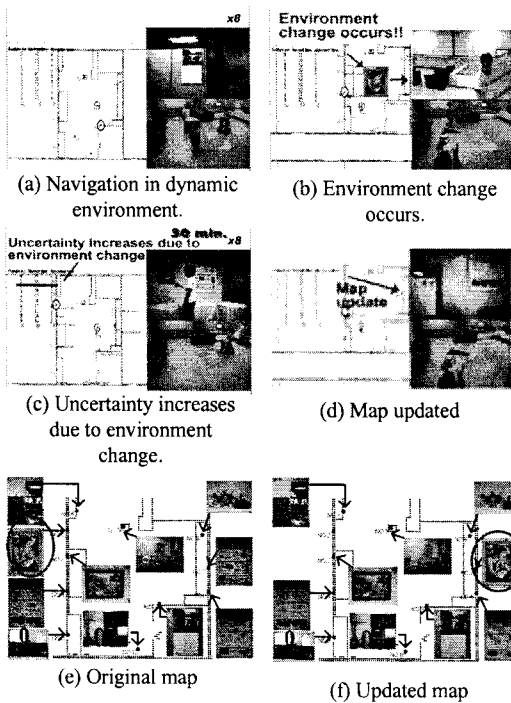


Fig. 5 Experiment of map update.

그림 5(e)는 환경을 표현한 혼합 격자지도이며, 위치추정에 사용되고 있다. 실험 초반에는 실제 환경과 지도와는 정확히 일치한다. 로봇은 그림 6(a)처럼 자율주행을 시작한다. 이후

로봇은 주행 중 위치추정에 사용되는 물체정보들의 위치 정확성과 불확실성에 대한 정보들을 축적한다. 이 과정을 수행하는 도중 실제 환경이 변하지 않았지만, 위치추정 오차 및 오정합으로 인해, 로봇의 위치추정에 사용되는 물체들의 위치 불확실성이 증가할 수 있다. 설계된 프레임워크는 각 물체의 위치 정확성과 부정확성에 대한 정보들을 축적하여, 실제 환경의 변화가 발생했는지를 판단하게 된다. 그림 5(b)는 지도에 등록된 물체의 실제 위치가 외부 요인에 의하여 변경된 경우이다. 이후 로봇은 프레임워크를 통해 환경이 변화되었다고 판단하여, 그림 5(d)와 같이 지도를 갱신한다. 그림 5(f)는 갱신된 지도이다.

4.2 결론

본 연구에서는 가변적인 환경에서 로봇 스스로 변화된 환경정보를 인식하고, 새롭게 인식된 환경정보를 이용하여 기존의 지도를 갱신할 수 있는 기법을 제안하였다. 제안된 지능적 지도갱신 프레임워크는 일반화된 추계적 베트리넷 기법을 통해 설계되었으며, 실험을 통하여 다음과 같은 결론을 얻을 수 있었다.

- 1) 일반화된 추계적 베트리넷의 정량적 평가방법을 통해 로봇 스스로 최적의 지도 갱신 시기를 판단할 수 있는 능력을 개발하였다.
- 2) 로봇 스스로 환경의 변화를 감지하여 기존의 지도를 갱신함으로써, 실제 환경과 지도간의 차이를 줄일 수 있었다. 이를 통해 위치추정의 성공률을 향상시키는 효과를 얻었다. 결과적으로 위치추정 성능의 향상으로 인해, 자율주행의 신뢰성 또한 향상되었다.
- 3) 본 논문에서 제안한 지도갱신 프레임워크는 가변적인 환경에서 반영구적으로 운용되어야 하는 여러 형태의 서비스 로봇에 유용하다.

후 기

이 연구는 산업자원부 지원으로 수행하는 21세기 프론티어 연구개발사업(인간기능 생활지원 지능로봇 기술개발사업)의 일환으로 수행되었습니다.

참 고 문 헌

- [1] Kosecka, J, Li, F, "Vision based topological Markov localization," Proc. of IEEE Int'l Conf. on Robotics and Automation, vol. 2, pp. 1481-1486, 2004.
- [2] Lowe, D.G, Se, S, "Distinctive image features from scale invariant keypoints," Int'l Journal of Computer Vision, vol. 60 no 2, pp. 91-110, 2004.
- [3] Lowe, D.G, Se, S, "Vision-Based global localization and mapping for mobile robots," Proc. of IEEE Transactions on Robotics, vol. 21, pp. 217-226, June, 2005.
- [4] Wang, J, "Timed Petri Nets Theory and Application," Kluwer Academic Publishers, 1998.