

미지의 환경에서 하이브리드 맵을 활용하는 모바일 로봇의 탐색

박정규*, 전홍석**, 노삼혁***

Mobile Robot Exploration in Unknown Environment using Hybrid Map

Jung Kyu Park*, Heung Seok Jeon**, Sam H. Noh***

요약

모바일 로봇은 자신의 임무를 수행하기 위해 탐색 기능을 가지고 있어야 한다. 탐색은 감시 로봇, 구조 로봇, 자원 탐사 로봇 등에 사용될 수 있다. 로봇이 환경을 탐색하기 위해서는 지도와 같은 환경에 대한 정보를 가지고 있어야 한다. 그러나 기존에 많이 사용되는 그리드 지도는 용량이 너무 커서 모바일 로봇에 사용하기 힘들다는 문제를 가지고 있다. 본 논문에서는 저 사양의 모바일 로봇에서 사용할 수 있는 하이브리드 지도를 제안한다. 또한 제안하는 하이브리드 지도를 사용하여 모든 영역을 탐색하는 방법을 제안하고 있다. 제안하는 방법은 로봇의 작업 환경을 현재 처리할 수 있는 영역과 처리 할 수 없는 영역으로 나누고, 나누어진 영역을 탐색하는 동시에 지도를 업데이트 하며 영역을 확장해 나간다. 탐색이 완료된 영역은 하이브리드 맵 형태로 저장한다. 생성한 하이브리드 맵을 사용하여 로봇은 장애물이 없는 영역으로 이동 경로를 생성할 수 있다. 실험결과에 따르면 기존 그리드 지도에 비해 약 6%의 메모리만을 사용해 환경 지도를 생성할 수 있었다.

▶ Keywords : 모바일 로봇, 하이브리드 맵, 매핑

Abstract

Mobile robot has the exploration function in order to perform its own task. Robot exploration can be used in many applications such as surveillance, rescue and resource detection. The workspace that robots performed in was complicated or quite wide, the multi search using the several mobile robots was mainly used. In this paper, we proposed a scheme that all areas are

•제1저자 : 박정규 •교신저자 : 전홍석

•투고일 : 2013. 3. 13, 심사일 : 2013. 3. 26, 게재확정일 : 2013. 4. 10.

* 홍익대학교 컴퓨터공학과 (Dept. of Computer Engineering, Hongik University)

** 건국대학교 컴퓨터공학과 (Dept. of Computer Engineering, Konkuk University)

*** 홍익대학교 컴퓨터정보공학부 (School of Computer & Information Eng., Hongik University)

※ 본 논문은 중소기업청에서 지원하는 2012년도 산학연공동기술개발사업(No. 2012-A424-0024)의 연구수행으로 인한 결과물임을 밝힙니다.

searched for by using one robot. The method to be proposed extract a area that can be explored in the workspace then the robot investigates the area and updates the map at the same time. The explored area is saved as a hybrid map that combines the nice attributes of the grid and topological maps. The robot can produce the safe navigation route without the obstacles by using hybrid map. The proposed hybrid map uses less memory than a grid map, but it can be used for complete coverage with the same efficiency of a topological map. Experimental results show that the proposed scheme can generate a map of an environment with only 6% of the memory that a grid map requires.

▶ Keywords : Mobile Robot, Hybrid Map, Mapping

I. 서 론

로봇이 환경에 대한 사전 정보가 없는 상태에서 환경에 놓이고 임무를 수행하기 위해서는 로봇 스스로 지도를 생성할 수 있어야 한다. 이를 수행하기 위해서 Simultaneous Localization and Mapping (SLAM) 알고리즘이 사용된다. SLAM 알고리즘은 로봇이 가지고 있는 센서를 사용하여 주변 환경 정보를 파악하고 그 내용을 바탕으로 지도를 생성하며 동시에 지도에서 자신의 위치를 추정하는 것을 말한다 [1-3][18]. 그러나 SLAM 알고리즘에 생성되는 지도는 대부분 고해상도의 그리드 지도를 사용하기 때문에 지도의 크기에 따라 지도의 업데이트 시간과 이동 경로 생성에 계산 시간이 많이 소비되는 문제점을 가지고 있다. 예를 들어 많이 사용되는 FastSLAM과 DP-SLAM의 경우 내부에서 사용되는 그리드 맵으로 약 30M 정도 크기의 파일이 사용되고 있다 [4][5]. 이러한 그리드 맵은 저사양의 모바일 로봇에서 사용이 어렵다. 또한 SLAM 알고리즘은 미지의 환경을 모두 매핑하기 위해서 로봇이 어떻게 이동해야 하는가에 대한 방법을 제시하지는 않는다. 이런 이유로 SLAM 알고리즘만을 사용해서 미지의 환경을 모두 표현할 수 있는 지도의 생성을 보장할 수 없다.

로봇이 동작하는 환경의 모든 영역을 매핑하기 위해서는 탐색 기능이 필요하다[1]. 그림 1은 로봇 탐색에서 많이 사용되는 방법을 표시하고 있다. 그림 1과 같은 방법은 쉽게 구현할 수 있지만 모든 영역의 탐색을 보장할 수 없다. 탐색 기능은 모바일 로봇의 핵심기술의 하나로 많은 연구가 이루어졌다. 대표적인 예로 Distance Transform 기반의 연구들을 들 수 있다[6][7]. 이 연구들은 로봇 탐색에서 많이 사용되지

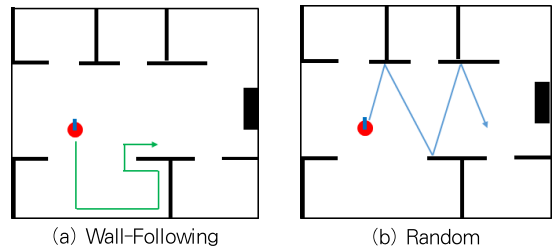


그림 1. 모바일 로봇의 탐색 경로
Fig. 1. Exploration path of mobile robot

만 로봇이 사전에 제작된 맵을 가지고 있음을 가정하고 있다. 앞의 연구를 기반으로 하는 로봇은 미지의 환경에 적용할 수 없는 문제점을 가지고 있다. 미지의 환경에서 동작하는 탐색 연구로 Sector-Based 기반의 연구를 들 수 있다[8]. 이 연구는 미지의 환경에서 동작하기는 하지만 로봇이 동작 환경의 외부 경계를 모두 순환하여 맵을 생성 후 탐색하는 것을 가정하고 있다. 미지의 환경의 탐색 연구에서 가장 활용되는 연구로는 Frontier-Based가 있다[9-11]. Frontier-Based 기반의 연구에서는 Frontier를 탐색했던 공간(Open Space)과 탐색하지 못한 영역(Unexplored Space)의 사이 공간으로 정의하고 있다. 로봇이 미지의 영역을 탐색하면서 다음에 이동할 공간으로 가장 가까운 Frontier 영역을 선택한다. Frontier-Based 기반의 연구는 하나의 로봇에 적용하여 사용할 경우 로봇이 많은 영역을 반복해서 이동해야 하는 문제와 복잡한 영역에서 경로 생성에 문제가 발생할 수 있다.

로봇 탐색에서는 로봇은 이동 경로를 생성하기 위해 지도를 사용한다. 로봇 분야에서 많이 사용되는 지도는 그리드 지도(Grid Map)와 토폴로지 지도(Topological Map) 두 가지 형태로 분류할 수 있다. 첫 번째는 그리드 지도는 환경을

작게 나누어진 그리드로 표현한다[12][19]. 각 그리드의 셀은 환경에서의 물체의 유무 또는 물체의 유무에 대한 확률을 표현한다. 두 번째는 토폴로지 지도는 환경을 그래프 형태로 표현한다[13]. 노드는 영역 또는 랜드마크를 표현한다. 노드와 노드 사이의 연결은 링크로 표현한다.

앞의 두 가지 지도 형태는 각각 장점과 단점을 가지고 있다. 그리드 지도는 센서 데이터만을 사용하여 쉽게 생성이 가능하다. 그리드 지도는 다양한 형태로 현재까지 많이 사용되고 있다. 그러나 환경이 커지고 복잡해짐에 따라 그리드 지도 크기가 커지면 경로 생성에 많은 계산 시간이 요구된다[4].

이에 반해 토폴로지 지도는 환경을 그래프 형태로 유지하기 때문에 그리드 지도 보다 작은 양의 메모리를 사용하여 더 큰 환경을 표현할 수 있다. 그러나 그래프의 노드는 해당 영역의 정확한 환경을 표현하지 못하기 때문에 로봇의 정확한 임무 수행에 적합하지 않을 수 있다. 예를 들어 노드 내에서 로봇의 정확한 위치를 모르기 때문에 노드 내의 이동이 불가능하다[13].

따라서 본 논문에서는 미지의 환경에서 로봇이 하이브리드 맵을 사용하여 모든 영역을 탐색하는 매핑 방법을 제안하고자 한다. 제안하는 방법은 제한된 환경, 예를 들어 벽으로 막혀 있는 영역, 뿐만 아니라 열려 있는 미지의 환경에서도 동작할 수 있다. 또한 제안하는 방법은 그리드 기반의 지도의 장점인 환경의 표현력을 갖으며 토폴로지 지도의 저 메모리 사용, 빠른 경로 검색의 특징을 수용하는 하이브리드 매핑이다. 하이브리드 매핑은 기존 그리드 매핑과 다르게 고정된 크기의 셀을 사용하지 않지만 그리드 맵의 환경 표현력을 유지한다.

본 논문의 나머지 구성은 다음과 같다. 다음 2장에서는 논문에서 제안하는 알고리즘에 대해 설명한다. 3장에서는 실험 결과를 설명하고, 마지막으로 4장에서는 결론과 향후 계획에 대해서 설명한다.

II. 하이브리드 매핑

1. 하이브리드 맵 구조

본 논문에서 제안하는 하이브리드 맵은 그리드 맵의 정확한 맵의 표현력과 토폴로지 맵의 저 메모리 사용, 빠른 경로 검색을 포함하고 있다.

논문에서 제안하는 맵은 그리드 맵을 기반으로 사각형 정보를 추출하여 토폴로지 형태로 맵을 유지하고 있다 (Rectangle-Based Hybrid Map, RcB-Map). 로봇이 구

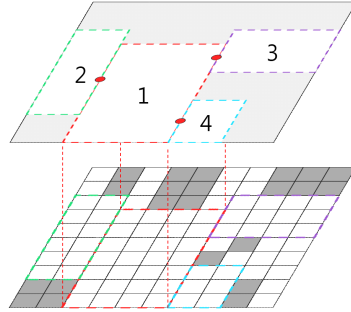
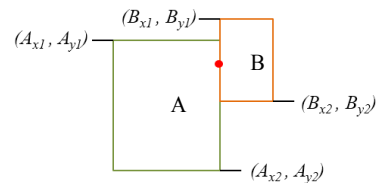


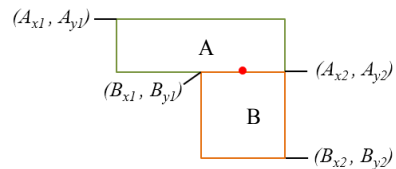
그림 2. 그리드 맵과 제안하는 하이브리드 맵
Fig. 2. Grid Map and Proposed Hybrid Map

동되어 이동을 시작하면 SLAM 알고리즘 기반으로 그리드 맵을 생성한다[5]. 대부분의 SLAM 알고리즘은 환경을 정확히 표현하기 위해 고해상도의 지도를 사용하고 있다. 예를 들어 DP-SLAM의 경우 3000x3000 크기의 지도를 생성하며, 약 30MB의 이미지 파일을 사용한다. 임베디드 보드나 일반 모바일 로봇에서는 하드웨어 사양의 제약으로 앞의 경우의 고해상도의 지도를 생성하거나 이를 활용하기 어렵다. 본 연구에는 고해상도의 맵을 사용하지 않고 저해상도의 그리드 맵을 이용하며 이를 토폴로지 맵으로 전환하여 사용한다.

그림 2는 본 논문에서 제안하는 하이브리드 맵과 일반적인 그리드 맵을 표현하고 있다. 로봇은 그림 2의 그리드 맵을 생성하고 그 맵을 기반으로 하이브리드 맵을 생성한다. 생성된 하이브리드 맵은 노드와 인접 중심점 (Adjacent Center Point, ACP)으로 표현하고 있다. 노드는 그리드 맵에서 공



(a) 노드가 좌우로 인접할 때



(b) 노드가 상하로 인접할 때

그림 3. 노드와 ACP
Fig. 3. Node and ACP

백인 사각형을 표현하고 ACP는 인접하는 두 노드 접합면의 중심점을 의미한다. 그림 3은 A, B 노드가 좌우 또는 상하로 인접할 때 ACP를 표현한 것이다. ACP는 수식 (3, 4)를 통해서 구할 수 있다.

제안하는 하이브리드 맵이 기존의 토폴로지 맵 형태와 다른 점은 다음과 같다. 일반 토폴로지 맵에서 노드는 표현하고자 하는 영역의 기본 정보만을 유지하지만 하이브리드 맵에서는 사각형을 하나의 노드로 지칭하고 해당 영역의 좌표 정보를 유지하여 그리드 맵의 정확한 표현을 보여주고 있다. 또한 ACP를 사용하여 노드간의 연결을 나타내며 노드간 이동경로 생성에 사용될 수 있다. 하이브리드 맵의 두 노드 A, B가 수식 (1), (2)와 같을 때 두 노드가 좌우로 인접하는 경우는 수식 (3)으로 상하로 인접하는 경우는 수식 (4)를 이용하여 ACP를 구할 수 있다.

$$node : A(A_{x1}, A_{y1}, A_{x2}, A_{y2}) \tag{1}$$

$$node : B(B_{x1}, B_{y1}, B_{x2}, B_{y2}) \tag{2}$$

$$ACP(x, y) = (\frac{A_{x1} + B_{x2}}{2}, A_{y2}) \tag{3}$$

$$ACP(x, y) = (A_{x2}, \frac{B_{y1} + A_{y2}}{2}) \tag{4}$$

제안하는 하이브리드 맵은 수식 (5)와 같이 정의 할 수 있다.

$$G = \{ V, A \} \tag{5}$$

V 는 식 (6)과 같이 노드의 ID, 노드의 좌표 정보 R, 노드 리스트 정보와 인접 노드 리스트로 표시할 수 있다.

$$V = \{ ID, R, pNext, pAnode \} \tag{6}$$

A 는 V 노드와 인접한 노드의 리스트로 노드의 V 노드의 ID, ACP, V 노드와 ACP의 거리 값, A 노드 리스트를 포함하며 식 (7)과 같이 표시할 수 있다.

$$A = \{ ID, ACP, nDistance, pNext \} \tag{7}$$

로봇이 동작하며 맵이 업데이트되고 V 노드가 새로 생긴 V 노드 리스트를 업데이트하고, 기존 노드와의 인접 여부를 확인하여 A 노드 리스트를 업데이트 한다. 하이브리드 맵의 업데이트는 알고리즘 1과 같이 표현할 수 있다.

알고리즘 1. 지도 업데이트
Algorithm 1. Map Update

```

1. MapUpdate (R1)
2.   vnode = MakeVnode(R1);           // (6)
3.   VnodeListAdd(vnode);
4.   if (IsACP(vnode))                // (3), (4)
5.     anode = MakeAnode(vnode, acp); // (7)
6.     AnodeListUpdate(anode);
7. End
    
```

3. 센서 모델

본 연구에서 하이브리드 맵은 그리드 맵을 기반으로 토폴로지 형태의 맵을 기반으로 하고 있다. 그리드 기반의 맵을 제작하기 위해서 로봇이 가지고 있는 센서를 사용하여 환경의 값을 측정하고 그리드 셀의 값으로 전환해야 한다. 센서의 값을 그리드의 한 셀(x, y)의 확률로 표현하기 위해 그림 4와 같은 센서 모델을 도입하였다. 센서 값 측정은 로봇에서 많이 사용하는 센서인 레이저 스캐너를 사용하였다.

로봇의 현재 좌표가 $X_t(x, y)$ 이고 센서의 한 점의 거리 측정 값이 d , 해당 각도가 θ 일 때 t 시간의 센서의 측정 값은 $Z_t(x, y, d, \theta)$ 로 표현할 수 있다. 측정값을 기반으로 그리드 지도의 셀은 식 (8)과 같이 확률로 표현할 수 있다. 센서 값 d 가 r 보다 작을 때는 0.7의 확률을 부여하고 현재 셀의 확률 값과 더한 값을 최근 값으로 업데이트 한다. 그 다음 센서 값 d 와 r 의 값이 같을 때는 0.05의 확률 값을 할당하고 셀을 업데이트 한다. 마지막으로 d 가 r 보다 클 경우에는 센서 값의 오류로 취급하여 확률 값을 0으로 부여하고 셀의 값은 업데이트 하지 않는다. 이 내용은 수식 8과 같이 정리할 수 있다.

$$p(Z_t | X_t) = \begin{cases} 0.7 & d < r \\ 0.05 & d = r \\ 0 & d > r \end{cases} \tag{8}$$

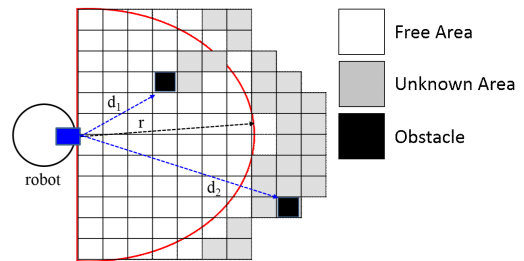


그림 4. 센서 모델
Fig. 4. Sensor Model

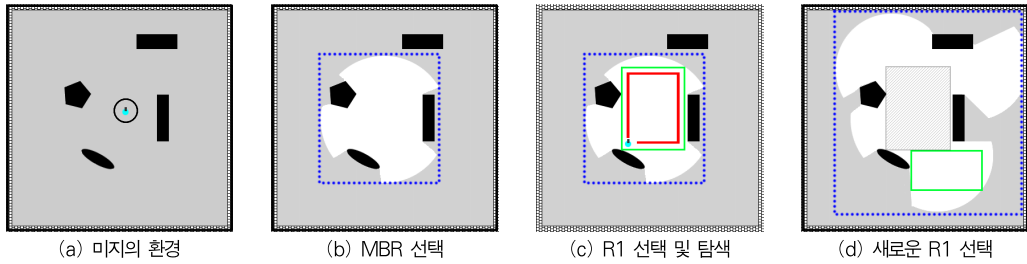


그림 5. RcB-Map 알고리즘의 처리 단계
Fig. 5. Processing Steps of the RcB-Map Algorithm

4. 탐색 및 하이브리드 맵 생성

이 절에서는 그림 5를 이용하여 미지의 환경을 탐색하고 하이브리드 맵을 생성하고 업데이트하는 방법을 설명한다.

그림 5의 (a)의 미지의 환경에 로봇이 위치하면 로봇은 자신의 센서를 사용하여 환경을 파악하고 그 정보를 이용하여 맵을 업데이트 해야 한다. 또한 생성된 맵을 이용하여 자신의 위치를 파악하고 맵을 확장할 수 있는 이동 경로를 생성할 수 있어야 한다. 로봇은 현재 위치에서 자신이 가지고 있는 센서와 앞 절에서 설명한 센서 모델을 사용하여 최초 맵을 생성한다.

이때 생성된 맵을 기반으로 그림 5(b)와 같이 최소의 사각형 영역(Minimum Bounding Rectangle, MBR)을 설정한다. 이때 MBR은 현재 생성된 지도 영역에서 로봇이 파악할 수 있는 최소의 사각형으로 공백 영역과 장애물 등을 포함한다. MBR 영역은 알고리즘 2를 이용하여 구할 수 있다. 구해진 MBR 영역은 현재 환경 상태를 표현하는 최소의 영역으로 환경이 변할 때 계속 변경될 수 있는 영역을 의미한다.

알고리즘 2. MBR 구하기
Algorithm 2. Getting MBR

1. GetMBR (Map)
2. GetMinPosition(x1, y1) free or occupied area from MAP
3. GetMaxPosition(x2, y2) free or occupied area from MAP
4. return (x1, y1, x2, y2)

그림 5(b)의 MBR 영역은 그림 6와 같이 그리드 맵으로 표현할 수 있다. MBR 영역이 파악되면 로봇은 새로운 이동 경로를 생성하여 그 경로를 이동하며 새로운 정보를 획득하고 지도를 확장해야 한다. 새로운 이동 경로를 선택하기 위해 MBR 영역을 Rectangle Tiling Scheme을 사용하여 그리드 맵을 작은 크기의 사각형 형태의 영역으로 분해한다[17].

그림 6의 점선 사각형은 분해된 사각형 중에서 일부를 표

현한다. 분해된 사각형 중에서 장애물을 포함하지 않은 가장 큰 영역을 선택하여 R1이라 명하고 이 영역을 로봇의 탐색 영역으로 선택한다. 그림 6에서는 가운데 점선 사각형(초록색 점선)이 R1으로 선택한다. 선택된 R1 영역의 정보를 이용하여 하이브리드 맵을 업데이트 한다. 이때 하이브리드 맵에는 수식 1과 같이 R1의 ID, 사각형 좌표, 노드 리스트 정보, 인접 노드 리스트가 저장된다.

R1 이 선택되면 해당 영역을 탐색을 통해서 로봇은 환경 지도를 확장할 수 있어야 한다. 선택된 R1 내부는 장애물이 없는 Free 영역으로 로봇이 자유롭게 이동할 수 있는 영역이다. 로봇은 그림 5(c)과 같이 선택된 R1 영역을 빠른 시간 내 많은 영역을 탐색하고 정보를 획득하기 위해 Wall following 으로 탐색 하면서 새로운 주변 정보를 획득하고 그리드 맵과 하이브리드 맵을 업데이트 한다.

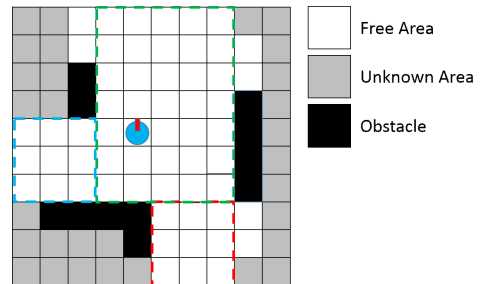


그림 6. MBR 영역의 그리드 맵
Fig. 6. Grid Map of the MBR area

최초의 R1 영역의 탐색이 끝나고 맵이 업데이트 되면 로봇은 새로운 맵을 기반으로 MBR 영역을 확인한다. 그림 5(d)와 같이 MBR 영역이 변경되었다면 새로운 MBR 영역을 기반으로 Rectangle Tiling을 수행하여 새로운 R1 을 선택한다. 이때 이전에 탐색했던 R1 은 하이브리드 맵의 하나

의 노드로 저장한다. 탐색된 R1 장애물이 없는 Free 영역이지만 탐색이 완료된 영역이기 때문에 장애물 영역으로 간주하여 새로운 R1 영역을 구할 때 Free 영역에 포함하지 않는다. 새로운 R1 영역을 선택할 때는 탐색의 효율성을 고려한다. 이때 효율성은 현재 로봇의 위치와 새로운 영역으로의 이동 거리와 R1의 크기를 포함한다. 알고리즘을 수행하며 더 이상의 R1을 구하지 못할 때 수행을 종료한다. 앞에 설명한 내용을 정리하면 알고리즘 3과 같다.

알고리즘 3. 미지의 환경에서 로봇 탐색

Algorithm 3. Robot Exploration in Unknown Environment

```

1: Init scan and make local map (map)
2: mbr = GetMBR (map); // algorithm 2
3: rectangles = RectangleTiling(map, mbr)
4: R1 = getR1(rectangles);
5: while (R1 != null)
6:   WallFollowing(R1);
7:   MapUpdate(R1)
8:   if (mbr is changed) Then
9:     goto line 2
10:  endif
11:  R1 = getR1(rectangles)
12: end while
13: End

```

III. 실험 결과

실험에서는 본 논문에서 제안하는 방법을 평가하기 위해 시뮬레이터를 제작하여 사용하였다. 시뮬레이터는 Visual C++로 제작하였고 본 논문의 알고리즘과 Frontier-Based 알고리즘[9][11]을 수행할 수 있다. 로봇이 동작하는 실험 환경은 15m x 15m의 일반 주거 환경을 모델링 하였다. 사용 로봇은 Pioneer 3-DX[14], 맵 제작을 위한 센서는 Hokuyo 레이저 라인지 스캐너 [15]를 모델링하여 사용하였다.

그림 7은 제안한 RcB-Map 알고리즘을 수행한 결과를 보여주고 있다. 로봇이 어떤 영역에 주어지면 로봇은 자신의 센서를 사용하여 그림 7(a)와 같이 주변을 탐색 하고 이를 바탕으로 그림 7(e)와 같이 이 최초 R1을 선택한다. 선택된 R1을 Wall-Following으로 탐색하여 그림 7(b)와 같이 그리드 지도를 확장하며 그림 7(f)와 같이 RcB-Map을 확장한다. 이때 3개의 Vnode를 사용하며 약 194B의 메모리를 사용하고 있다. 그림 7(c)와 환경의 3/4 영역을 탐색한 이후에는 그림 7(g)와 같이 7개의 Vnode 사용하여 환경을 표현하고 있다. 이 작업을 반복하여 7(d)와 같이 환경의 탐색이 완료되면 그림 7(h)와 같은 RcB-Map 맵을 작성할 수 있다. 이때 사

용된 Vnode는 약 50개로 5420B의 메모리를 사용하고 있다. 이때 사용된 메모리는 300x300 그리드 지도와 비교 했을 때 약 6% 정도이다.

그림 8은 Frontier-Based 알고리즘을 수행한 결과이다. Frontier-Based 알고리즘은 이미 탐색된 지역과 탐색되지 않은 지역 사이의 영역을 Frontier라 언급하고 현재 위치에서 가장 가까운 Frontier 영역으로 이동하며 새로운 영역 탐색을 시도한다. 이번 실험에서 대상 공간을 모두 탐색하는 이동거리만 보면 Frontier-Based가 1750번 RcB-Map 27,000번 이동을 하였다. 또한 로봇의 턴 횟수를 보면 Frontier 122 번 RcB-Map이 240번 정도이다. Frontier-Based 알고리즘이 총 이동 거리는 짧지만 복잡한 구역에서의 이동시 많은 턴이 발생하여 로봇의 제어가 힘들다는 문제점을 가지고 있다. 또한 Frontier 영역으로 이동시 환경에 따라 바로 이동할 수 없는 문제가 발생한다. 이에 반해 RcB-Map 알고리즘은 복잡한 구역도 로봇이 다닐 수 있는 영역으로 나누어 처리함으로써 Frontier-Based에 비해 안정된 이동 경로를 생성할 수 있다.

IV. 결론

본 논문에서는 미지의 환경에서 탐색 임무를 수행할 수 있는 탐색 알고리즘과 하이브리드 매핑 방법을 제안하였다. 제안한 매핑 방법은 그리드 지도와 토폴로지 지도의 장점을 수용하여 저 사양의 하드웨어에서 사용될 수 있다. 실험을 통해 RcB-Map이 기존 그리드 지도에 비해서 약 6%(300x300 그리드 지도)의 메모리만을 사용하여 환경 지도를 생성할 수 있음을 보였다.

향후에는 임베디드 보드의 처리 능력과 배터리를 고려하는 이동 경로 생성 방법을 연구할 것이다 또한 다른 탐색 알고리즘과 비교 수행을 하여 새로운 실험 결과를 얻을 것이며, 결과를 바탕으로 RcB-Map을 지속적으로 개선할 예정이다.

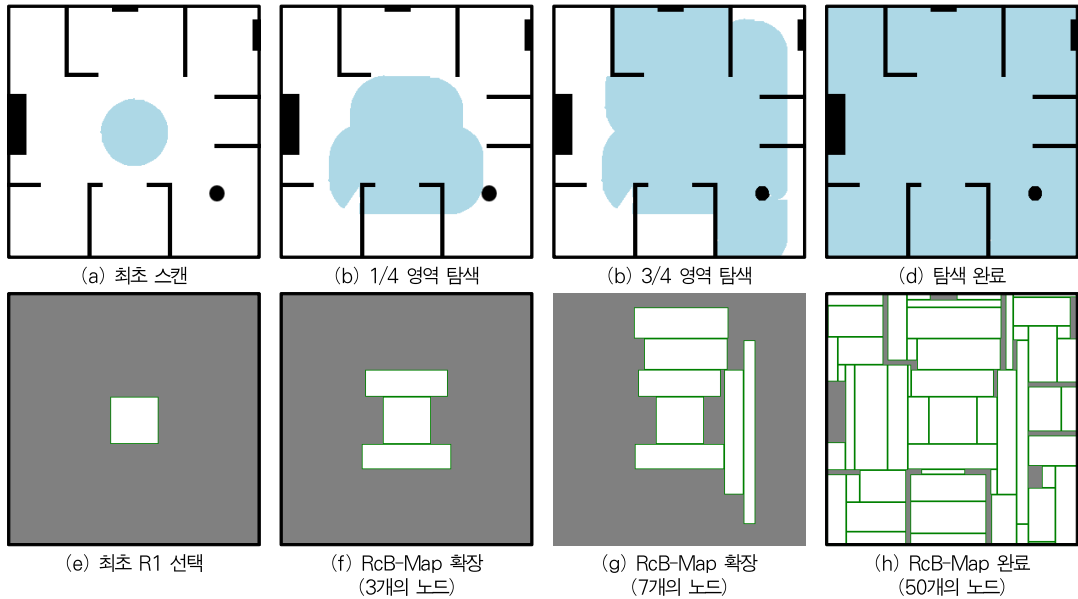


그림 7. RcB-Map 알고리즘의 결과
Fig. 7. Results of the RcB-Map Algorithm

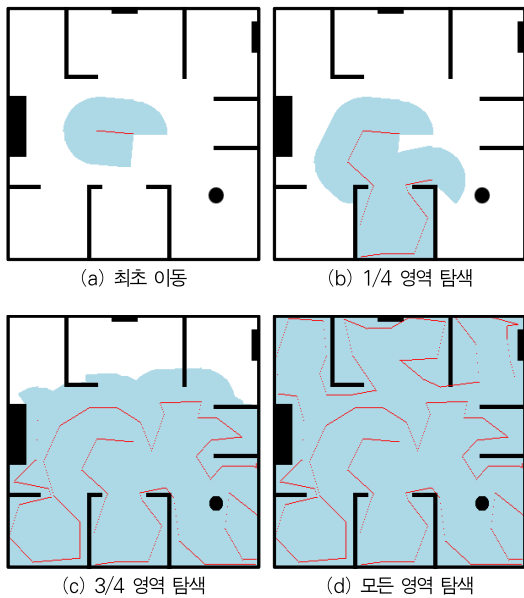


그림 8. Frontier-Based 알고리즘 결과
Fig. 8. Results of Frontier-Based Algorithm

참고문헌

- [1] D. Fox, J. Hightower, L. Liao, D. Schulz, and G. Borriello, "Bayesian Filtering for Location Estimation", IEEE Pervasive Computing, 2003.
- [2] S. Thrun, D. Fox, W. Burgard, and F. Dellaert, "Robust Monte Carlo Localization for Mobile Robots," in Proceedings of the National Conference on Artificial Intelligence, 2001.
- [3] G. Dissanayake, P. Newman, H. F. Durrant-Whyte, S. Clark, and M. Csobra, "A Solution to the Simultaneous Localisation and Mapping (SLAM) Problem," IEEE Transactions on Robotics and Automation, 2001.
- [4] M. Montemerlo, S. Thrun, D. Koller, and B. Wegbreit, "FastSLAM: A Factored Solution to the Simultaneous Localization and Mapping Problem," in Proceedings of the AAAI National Conference on Artificial Intelligence, 2002.
- [5] A. Eliazar and R. Parr, "DP-SLAM: fast, robust simultaneous localization and mapping without predetermined landmarks," in Proceedings of the

- 18th international joint conference on Artificial intelligence, 2003.
- [6] A. Zelinsky, R. A. Jarvis, J. C. Byrne, and S. Yuta, "Planning Paths of Complete Coverage of an Unstructured Environment by a Mobile Robot," in Proceedings of the International Conference on Advanced Robotics, 1993.
- [7] A. Zelinsky, "Using path transforms to guide the search for findpath in 2D," International Journal of Robotics Research, 1994.
- [8] T. Lee, S. Baek and S. Oh, "Sector-based maximal online coverage of unknown environments for cleaning robots with limited sensing," Robotics and Autonomous Systems, 2011.
- [9] A. Mobarhani, S. Nazari, A. H. Tamjidi and H. D. Taghirad, "Histogram Based Frontier Exploration", in *Proceedings IEEE International Conference on Robotics and Automation (ICRA)*, pages 1128-1133, 2012.
- [10] M. Moors, D. Fox, R. Simmons, and S. Thrun, "Collaborative multi-robot exploration," in Proceedings of the IEEE International Conference on Robotics and Automation, 2000.
- [11] L. Jeanpierre and A. Mouaddib, "Distributed Value Functions for Multi-Robot Exploration," in *Proceedings IEEE International Conference on Robotics and Automation (ICRA)*, pages 1544-1550, 2012.
- [12] A. Elfes, "Using Occupancy Grids for Mobile Robot Perception and Navigation," Computer, Volume 22 Issue 6, Pages 46-57, 1989.
- [13] S. Thrun, "Robotic Mapping: A Survey," Exploring Artificial Intelligence in the New Millenium, 2002.
- [14] Pioneer 3-DX, <http://www.mobilerobots.com>
- [15] URG-04LX, <http://www.hokuyo-aut.jp>
- [17] M. Orłowski, "A New Algorithm for the Largest Empty Rectangle Problem," Algorithmica, vol. 5, pp. 65-73, 1990.
- [18] J. K. Park, H. S. Jeon and Sam H. Noh, "A Robot Coverage Algorithm Integrated with SLAM for Unknown Environments," The Korea Society of Computer and Information, vol. 15(1), pp. 61-69, 2010.
- [19] H. I. Kim, "Path Planning for Cleaning Robots Using Virtual Map," The Korea Society of Computer and Information, vol. 14(11), pp. 31-40, 2009.

저 자 소개



박 정 규

2002 홍익대학교 컴퓨터공학과 석사
2006-현재 홍익대학교 컴퓨터공학과 박사 과정
관심분야: 지능형 로봇, 운영체제, 임베디드 시스템
Email: parkjk@cs.hongik.ac.kr



전 흥 석

2001 홍익대학교 전자계산학과 박사
2007-2008 아이오와대학교 컴퓨터공학과 객원 교수
2002-현재 건국대학교 자연과학대학 컴퓨터공학과 교수
관심분야: 지능형 로봇, 임베디드 시스템
Email: hsjeon@kku.ac.kr



노 삼 혁

1986 서울대학교 컴퓨터공학과 학사
1993 매릴랜드대학교 컴퓨터공학과 박사
1993~1994 조지워싱턴대학교 객원 조교수
1994-현재 홍익대학교 정보컴퓨터공학교수
관심분야: 운영체제, 플래시메모리소프트웨어, 차세대지장장치
Email: samhnoh@hongik.ac.kr