

능률적인 3차원 경로계획 알고리즘 개발에 관한 연구

이 승 철, 양 원 영, 김 용 환

중앙대학교 전기공학부

An Efficient 3-D Path Planning Algorithm for Robot Navigation

S. C. Lee, W. Y. Yang, Y. H. Kim

Dept. of Electrical Eng., Chung-Ang University

Abstract - In this paper, an efficient and robust robot path planning technique is discussed. Concentric Ripple Edge Evaluation and Progression (CREEP) algorithm[1] has been elaborated and expanded to carry out 3-D path planning. Like the 2-D case, robot can always find a path, if one exists, in a densely cluttered, unknown and unstructured 3-D obstacle environment. 3-D space in which the robot is expected to navigate is modeled by stacking cubic cells. The generated path is resolution optimal once the terrain is fully explored by the robot or all the information about the terrain is given. Path planning times are significantly reduced by local path update. Accuracy and efficiency of wave propagation in CREEP algorithm are achieved by virtual concentric sphere wave propagation. Simulations in 2-D and 3-D spaces are performed and excellent results are demonstrated.

1. 서론

자율이동로봇(Autonomous Mobile Robot)의 경로계획 알고리즘(path planning algorithm)에 대하여는 과거부터 많은 연구가 행해졌고 현재에도 활발히 진행되고 있다.

기존의 알고리즘은 크게 세 가지로, 로드맵 방법(roadmap approach), 포텐셜 필드 방법(potential field approach), 셀 분해법(cell decomposition approach)으로 분류할 수 있다. 첫째로 로드맵 방법에 속하는 것은 가시도 그래프 방법(visibility graph method)과 자유공간 방법(free space method)이 있다. 가시도 그래프 방법은 장애물을 다면체로 보고 로봇이 다면체의 코너(corner) 꼭지점을 탐지하여 로봇 위치와 목표점과 다면체의 코너 꼭지점을 잇는 가능한 모든 선을 그어 가시도 그래프를 만든 후 그래프 탐색 알고리즘(graph search algorithm)에 의해 최단경로를 정하는 방법이다[2]. 자유공간 방법에서는 장애물들 사이의 자유공간을 보로노이 도형(Voronoi diagram) [3], 일반화된 원통(generalized cylinders)[4], 또는 볼록한 영역(convex region)[5], [6] 으로 표현하는 방법들이 제시되었다. 그러나 이 방법에서는 로봇이 목표점까지 불필요하게 멀리 돌아가는 경향이 있다. 두 번째로 포텐셜 필드 방법에서는 국부적 최소점(local minima)에 빠지는 것을 해결하려는 여러방법들이 제시되었지만 본질적인 한계(inherent limitation)가 여전히 존재한다[7], [8]. 세 번째로 셀 분해법은 로봇의 활동영역을 정사각형 셀(cell)의 배열로 분해해서 각 셀에 특정한 값을 부여함으로써 로봇이 그 셀 값에 따라 경로를 계획하는 방법이다[9]. 이

방법의 가장 큰 장점은 만약 경로가 존재한다면 반드시 찾는다 는 보장이 있다는 것이다. 이 범주에 속하는 것으로서 거리 변환 방법(distance transform)이 있다[10], [11]. 이 방법은 목표점에서 해당 셀까지의 거리가 셀 값이 되고 셀 값은 포워드 마스크(forward mask)와 백워드 마스크(backward mask)에 의해 채워진다. 장애물의 갯수가 많아지고 환경이 복잡해질수록 계산량이 급격하게 증가한다는 단점이 있다.

본 논문에서는 셀 분해법과 가상의 웨이브 전파법(virtual wavefront propagation approach)에 기초한 CREEP 알고리즘을 3차원으로 확장시킨 효율적이고 안정된 3차원 공간에서의 경로 계획 알고리즘을 제안한다.

2. 3차원 CREEP 알고리즘

로봇의 활동영역은 3차원 공간을 정육면체 셀의 적층으로 모델링하였고, 셀의 크기는 장애물이 아닌 셀에 대해서는 로봇이 통과할 수 있게 보장하기 위해서 로봇의 크기와 같게 하였다. 또한 로봇은 제한된 범위 내에서 장애물을 탐지(sensing)할 수 있다고 가정한다.

2.1 가상의 웨이브(Virtual Wave) 발생 및 전파

3차원 공간상에서 i 번째 행과 j 번째 열과 k 번째 깊이 에 있는 셀을 $c(i, j, k)$ 로 표시한다. 하나의 셀은 세 종류로 구분되는 26개의 이웃 셀을 갖는다. 임의의 셀 $c(i, j, k)$ 와 면을 접하며 이웃해 있는 6 개의 셀을 면 이웃 셀(side adjacent cells), $A^s(i, j, k)$ 로 정의하고, $c(i, j, k)$ 와 선을 접하며 이웃해 있는 12 개의 셀을 선 이웃 셀(line adjacent cells), $A^l(i, j, k)$ 로 정의한다. 또한 점에 의해 이웃해 있는 8 개의 셀을 점 이웃 셀(point adjacent cells)이라 정의하고 $A^p(i, j, k)$ 로 표시한다. 예를 들면, 임의의 셀 $c(i, j, k)$ 를 중심으로 보았을 때,

$$A^s(i, j, k) = \{c(i, j-1, k), c(i-1, j, k), c(i, j+1, k), c(i+1, j, k), c(i, j, k-1), c(i, j, k+1)\}$$

$$A^l(i, j, k) = \{c(i, j-1, k-1), c(i-1, j, k-1), c(i, j+1, k-1), c(i+1, j, k-1), c(i-1, j-1, k), c(i-1, j+1, k), c(i+1, j+1, k), c(i+1, j-1, k), c(i, j-1, k+1), c(i-1, j, k+1), c(i, j+1, k+1), c(i+1, j, k+1)\}$$

$$A^s(i, j, k) = \{c(i-1, j-1, k-1), c(i-1, j+1, k-1), \\ c(i+1, j+1, k-1), c(i+1, j-1, k-1), \\ c(i-1, j-1, k+1), c(i-1, j+1, k+1), \\ c(i+1, j+1, k+1), c(i+1, j-1, k+1)\}$$

$$A(i, j, k) = A^1(i, j, k) + A^2(i, j, k) + A^3(i, j, k) \quad (1)$$

이다. 여기에서 $A(i, j, k)$ 는 $c(i, j, k)$ 의 모든 이웃하는 셀들의 집합이다.

먼저 목표점(goal point)인 c_g 에 웨이브 세기(wave strength; WS) 초기값 w_g 를 부여하면 가상의 웨이브가 목표점에서 발생하여 장애물 셀을 제외한 이웃 셀들에게 값을 부여하면서 중심 구 모양으로 영역전체로 퍼져나간다. 임의의 셀 $c(i, j, k)$ 에서 이웃 셀로 웨이브가 퍼져나갈 때, WS 값은 $A^1(i, j, k)$ 에 속한 셀에는 거리값인 1이, $A^2(i, j, k)$ 에는 $\sqrt{2}$ 가, $A^3(i, j, k)$ 에는 $\sqrt{3}$ 이 감소되면서 전파된다. 여기에서 장애물 셀이 아닌 각 셀의 전파도(traversability)를 1로, 장애물 셀은 0으로 가정한다. 웨이브는 장애물에 부딪힐 때마다 갈라져서 그 장애물을 돌아가서 퍼져나간다. 그래서 목표점에서 로봇까지 경로가 존재하는 한 최소한 하나의 웨이브라도 로봇에게 도착하게 된다. 로봇의 입장에서는 가장 강한 웨이브가 들어오는 방향이 최단경로의 방향이 된다.

2.2 웨이브 세기 값(Wave Strength Value) 부여

임의의 셀 $c(i, j, k)$ 는 그 주위의 26개의 셀 중에서 이미 WS 값을 갖고 있는 셀로부터 WS 값을 받을 수 있다. 그러나 간접적으로 장애물에 의해 막혀있는 셀들(indirectly obstructed cells)인 $A'(i, j, k)$ 로부터는 값을 받을 수 없다. 예를 들면, 임의의 셀 $c(i, j, k)$ 가 이웃 셀들로부터 WS 값을 받으려고 할 때 $c(i, j+1, k)$ 가 장애물 셀이라면,

$$A'(i, j, k) = \{c(i-1, j+1, k-1), c(i, j+1, k-1), \\ c(i+1, j+1, k-1), c(i-1, j+1, k), \\ c(i+1, j+1, k), c(i-1, j+1, k+1), \\ c(i, j+1, k+1), c(i+1, j+1, k+1)\} \quad (2)$$

이 된다. 그리고 주어진 임의의 셀 $c(i, j, k)$ ($c(i, j, k) \neq c_g$, $c(i, j, k) \notin W$)의 WS 값 $w(i, j, k)$ 는

$$w(i, j, k) = \max \{ ((w(l, m, n) - 1) \forall c(l, m, n) \in A^1(i, j, k)), \\ ((w(l, m, n) - \sqrt{2}) \forall c(l, m, n) \in A^2(i, j, k) - A^1(i, j, k)), \\ ((w(l, m, n) - \sqrt{3}) \forall c(l, m, n) \in A^3(i, j, k) - A^1(i, j, k)) \} \quad (3)$$

과 같이 계산된다. 여기에서 W 는 이미 WS 값을 받은 셀들의 집합이고, $c(i, j, k)$ 에 $w(i, j, k)$ 를 주는 $c(l, m, n)$ 은 $c(i, j, k)$ 의 부모 셀(parent cell)이라 하며 $p(i, j, k)$ 로 표시한다.

2.3 리플 숫자 부여(Ripple Number Assignment)

웨이브는 점 이웃 셀 방향으로 가장 빠르지만 가장 작은 값을 가지고 퍼져나가고 선 이웃 셀 방향은 중간이고, 면 이웃 셀 방향으로는 가장 느리지만 가장 큰 값을 가지고 퍼져나간다. 이렇게 웨이브가 퍼져나갈 때 장애물을 만나면 웨이브는 갈라져서 퍼져나간다. 그래서 장애물이 아주 많은 환경에서는 수많은 웨이브가 영역 전체에서 다른 속도로 퍼져나갈 것이다. 만약 다른 속도의 웨이브들이 임의의 셀 $c(i, j, k)$ 에 도착할 때, 보통 가장 빠르지만 제일 약한 웨이브가 맨 처음 도착할 것이다. 이때 더 강한 웨이브가 더 늦게 $c(i, j, k)$ 에 도착하면

$w(i, j, k)$ 는 갱신되어야 한다. 결국 약한 WS 값을 받았던 $c(i, j, k)$ 로부터 이미 WS 값을 받은 $A(i, j, k)$ 에 속한 모든 셀들의 WS 값은 갱신되어야 한다. 이렇게 영역 전체에 걸쳐 강하지만 더 늦게 도착하는 웨이브가 이미 부여된 WS 값을 덮어쓰는 것을 막기 위해, 즉 물결의 전파속도를 어떤 상황에서도 일정하게 해주기 위해 우리는 다음의 식에 의해 각각의 셀에 리플 숫자(ripple number)를 부여한다.

$$w_g - s - 1 < w(i, j, k) \leq w_g - s \quad (4)$$

여기에서 $s \geq 1$ 인 정수이고, $c(i, j, k)$ 와 $w(i, j, k)$ 가 주어졌을 때, 만약 위의 조건을 만족하면 리플 숫자 $r(i, j, k)$ 는 $c(i, j, k)$ 와 관련된다. 이때 식 (4)를 만족하는 WS 값을 가진 모든 셀들은 리플 집합(ripple set)(또는 줄여서 리플(ripple)) $R(s)$ 의 원소(element)가 된다. 그리고 $R(0) = \{c_g\}$ 가 된다. $R(s)$ 에 속하는 임의의 셀의 WS 값을 계산하기 위해서는 $R(s-1)$ 과 $R(s-2)$ 에 속한 모든 셀들은 이미 그들의 WS 값을 부여받았어야 하고 고정되어 있어야 한다. 이러한 순차적인 WS 값과 리플 숫자의 부여를 확장(expansion)이라고 정의한다. 같은 리플에 속하는 셀들 사이에 WS 값을 구하는 우선 순위가 없으므로, 같은 리플에 속하는 모든 셀의 WS 값은 병렬로 계산될 수 있다. 그리고 로봇이 항해할 때 WS 값의 갱신은 단지 리플 $R(s)$ ($s \geq$ 현재 경로를 가로막는 새롭게 발견된 장애물의 리플 숫자)에 속하는 셀만 하면 된다. 이러한 명백한 국부적 경로수정은 계산시간을 크게 줄일 수 있다.

다음의 그림 1은 2차원 10×10 해상도(resolution)에서 셀 확장의 예를 보인 것이다. 셀 안의 실수 숫자는 WS 값이고 괄호 안의 숫자는 리플 숫자이다. 같은 리플 숫자의 셀 들을 곡선으로 서로 이어보면 웨이브가 중심원처럼 퍼져나감을 알 수 있다. 그리고 우측 상단의 삼각형모양은 목표점을 나타내고 우측 하단의 원은 로봇의 위치를 나타낸다. 이웃하는 셀 중에서 간접적으로 장애물에 막혀 있는 셀을 제외한 가장 큰 값을 가진 셀이 부모 셀이고, 그 셀을 따라가면 최단 경로가 된다.

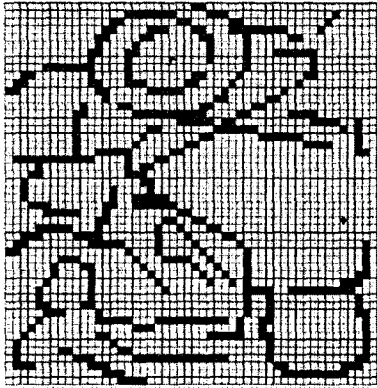
21.99	22.59	23.09	24.09	25.59	26.09	27.09	28.59	29.09	29.09
(10)	(7)	(6)	(5)	(4)	(3)	(2)	(1)	(1)	(1)
18.18	21.99	26.00	26.00	27.00	28.00	29.00	29.00	29.00	29.00
(8)	(8)	(8)	(8)	(3)	(2)	(1)	(1)	(1)	(1)
20.16	21.00	26.00	26.00	27.00	28.00	29.00	29.00	29.00	29.00
(9)	(9)	(9)	(9)	(2)	(1)	(1)	(1)	(1)	(1)
20.00	21.00	22.00	23.00	24.00	27.18	27.59	28.00		
(10)	(8)	(8)	(7)	(6)	(2)	(2)	(2)		
18.58	20.89	21.00	22.58	23.00	26.16	26.58	27.00	26.00	
(10)	(8)	(8)	(7)	(7)	(2)	(2)	(3)	(4)	
19.18	20.18	20.99							
(10)	(8)	(8)							
16.77	19.18	19.99	18.59	17.59	0.00	0.00	0.00	0.00	
(11)	(10)	(10)	(11)	(12)	(20)	(20)	(21)	(10)	
17.77		18.99	14.59		10.89	10.18	6.00	0.00	
(12)		(11)	(13)		(18)	(18)	(20)	(21)	
14.77	16.77		14.58	16.98	11.88	11.18	10.18	0.00	
(13)	(14)		(18)	(14)	(18)	(18)	(18)	(20)	
16.77	19.36	14.36	14.18	14.99	13.99	12.99	10.99	0.00	
(14)	(14)	(15)	(15)	(15)	(16)	(17)	(18)	(20)	

그림 1. 2차원에서의 셀 확장(expansion)

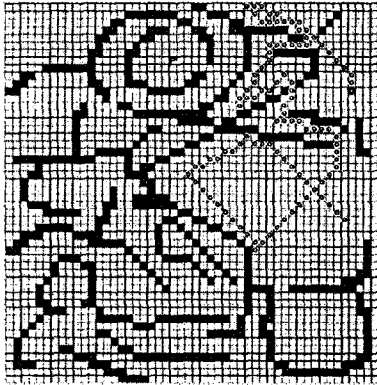
3. 모의 실험 결과(Simulation Results)

앞 절에서 설명한 3차원 CREEP 알고리즘을 C 언어로 구현하여 다양한 장애물 환경에서 실험해 보았다. 막다른 길이 있는 복잡한 장애물 환경에서도 로봇이 효율적으로 목표점을 찾아가는 것을 보이기 위해 그림 2에서는 2차원 50×50 해상도에서의 결과를 보였다. 그림 2(a)에서 위쪽 가운데에 보이는 깃발 모양이 목표점이고, 오른쪽 중간에 보이는 원모양이 로봇

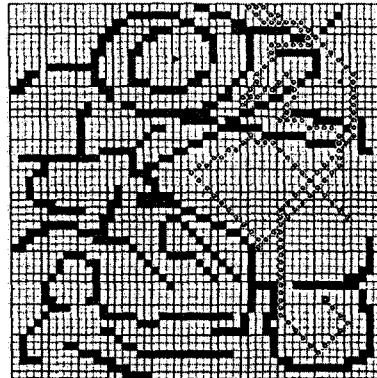
이다. 그림에서 로봇이 탐지한 장애물은 진한 색깔로 바뀌게 된다. 그림 2(b), (c)에서 보는 바와 같이 막다른 길에서도 자유롭게 빠져나온다. 그림 2(d)에서는 경로가 존재하는 한 반드시 목표점까지 찾아간다는 것을 보인다. 그리고 그림 2(e)에서는



(a) 로봇과 목표점과 장애물의 초기배치 (로봇은 장애물에 대한 어떠한 정보도 갖고 있지 않다.)

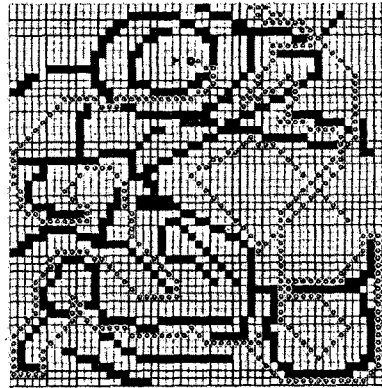


(b) 항해중인 로봇 (진한 색깔은 로봇이 탐지한 장애물)

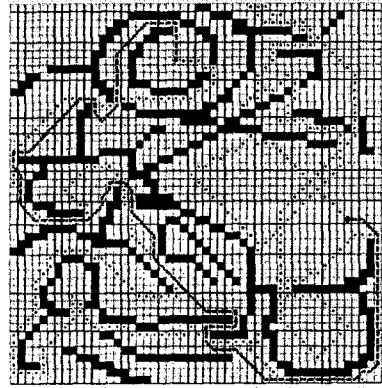


(c) 막다른 길에서도 자유롭게 빠져나옴

처음에 목표점을 찾아갈 때 탐지했던 장애물에 대한 정보를 갖고 다시 로봇의 원래 위치로 되돌아갈 때는 최단경로를 알 수 있다.



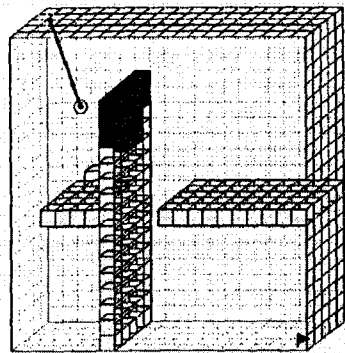
(d) 목표점에 도착



(e) 돌아갈 때는 최단경로로 감.

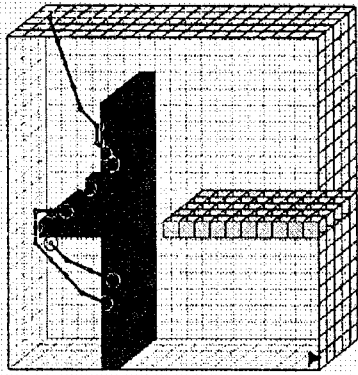
그림 2. 50×50 해상도에서 CREEP 알고리즘에 의한 경로찾기

그림 3에서는 이 논문의 주제인 3차원 CREEP 알고리즘이 3차원 공간상에서도 효율적으로 적용되는 것을 보였다. 그림 3(a)에서 목표점의 위치는 $c(20, 20, 1)$ 이고 로봇의 초기 위치는 $c(1, 1, 5)$ 이다. 3차원에서 그림 3(d)에서 보이는 바와 같이

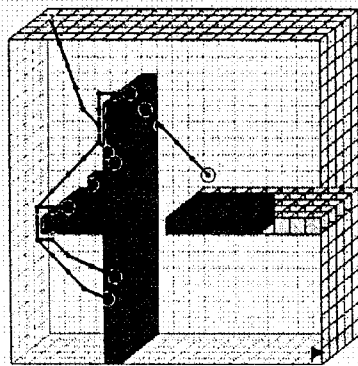


막다른 길에서도 자유롭게 탈출하여 목표점을 찾아간다.

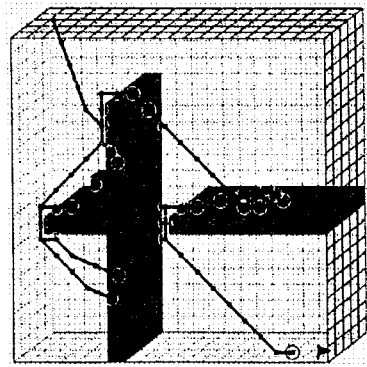
(a)진한 색깔은 로봇이 탐지한 장애물



(b) 막다른 길에서 자유롭게 빠져나옴



(c) 3차원 공간에서 항해중인 로봇



(d) 목표점에 도착

그림 3. 3차원 20×20×5 해상도에서 CREEP 알고리즘에 의한 경로찾기.

4. 결론 및 향후 연구과제

본 연구에서는 자율이동로봇을 위한 통상의 2 차원 평면에서의 경로계획 알고리즘에 기초하여 효율적이고 견실한 3차원 공간에서의 경로계획 알고리즘을 개발하였고, 모의실험으로 그 유용성을 보였다. 과거에 제안된 셀 분해법이나 웨이브 전파법과 달리 CREEP 알고리즘은 대각선 방향으로의 좀 더 정확한 웨이브 전파와 리플 숫자에 의한 웨이브 전파 속도의 동기화를 통해 주어진 해상도 안에서 더욱 정확한 경로를 찾을 수 있다. 그리고 제안된 알고리즘의 리플 숫자에 의한 국부적 경로수정과 WS 값의 병렬계산은 실시간 경로 계획을 위한 큰 장점이다. 또한 위험한 장애물이 있는 환경에서는 퍼텐셜 필드 방법에서처럼 로봇이 장애물과의 안전거리를 유지해야 한다면, 우리는 단지 장애물과 이웃한 셀에는 낮은 웨이브 전파도를 부여하기만 하면 된다. 앞으로 3차원 CREEP 알고리즘을 로봇 매니퓰레이터(manipulator)의 경로계획에 적용하기 위한 연구가 이루어질 것이다. 그것을 위해서 우선은 피아노 운반 문제(piano mover problem)를 풀 수 있도록 알고리즘을 좀더 발전시키는 게 필요하다.

참고 문헌

- [1] 이승철, "능률적이고 안정된 로봇 경로계획 알고리즘 개발에 관한 연구," 대한 전기학회 하계 학술대회 논문집, pp. 823-825, 1995년 7월.
- [2] Oommen B. J. et al, "Robot Navigation in Unknown Terrains Using Learned Visibility Graphs. Part I: The Disjoint Convex Obstacle Case," *IEEE Journal of Robotics and Automation*, Vol. RA-3, No. 6, pp. 672-681, Dec. 1987.
- [3] 이주호, 서삼준, 박귀태, "자율 이동 로봇의 경로계획," 대한 전기학회 하계 학술대회 논문집, pp. 866-870, 1995년 7월.
- [4] Brooks, R. A., "Solving the Find-Path Problem by Good Representation of Free Space," *IEEE Transactions on Systems, Man, and Cybernetics*, Vol. SMC-13, No. 3, March/April 1983.
- [5] Ali, A., Abidi, M., "A 2-D and 3-D Path Planning Algorithm for Robot Navigation and Manipulation," *Technical Report*, Department of Electrical and Computer Engineering, University of Tennessee, Knoxville, 1987.
- [6] Chavex, R. and Meystel, A., "Structure of Intelligence for an Autonomous Vehicle," *Proceedings of the IEEE International Conference on Robotics*, pp. 584-591, 1984.
- [7] Khatib, O., "Real-time Obstacle Avoidance for Manipulators and Mobile Robots," *International Journal of Robotic Research*, Vol. 5, No. 1, pp. 90-98, Spring 1986.
- [8] Koren Y. and Borenstein J., "Potential Field Methods and Their Inherent Limitations for Mobile Robot Navigation," *Proceedings of the 1991 IEEE International Conf. on Robotics and Automation*, Sacramento, California, pp. 1398-1404, April 1991.
- [9] Latombe, J., *Robot Motion Planning*, Norwell, MA: Kluwer Academic Publishers, 1991.
- [10] Jarvis, R.A. and Byrne J.C., "Robot Navigation: Touching, Seeing, and Knowing," *Proceedings of the First International Conference on AI*, Nov. 18-21, 1986.
- [11] Borgefors, G., "Distance Transformations in Arbitrary Dimensions," *Computer Vision, Graphics, and Image Processing*, Vol. 27, pp. 321-345, 1984.