

인간형 로봇의 이동경로 생성을 위한 장애물 모양의 구분 방법[§]

박 찬 수* · 김 도 익*†

* 한국과학기술연구원 실감교류로보틱스센터

Classification of Obstacle Shape for Generating Walking Path of Humanoid Robot

Chan-Soo Park* and Doik Kim**†

* Interaction and Robotics Research Center, Korea Institute of Science and Technology

(Received June 14, 2012; Revised September 24, 2012; Accepted September 25, 2012)

Key Words: Obstacle Detection(장애물인식), Range Segmentation(영역 분할), Humanoid Robot(인간형 로봇), 3D Depth Map(3 차원 깊이 지도)

초록: 알려지지 않은 실내에서 인간형 로봇의 이동경로 생성을 위해서는 주변 장애물의 형태를 정확히 인식하여 이에 적합한 로봇 움직임을 만들어야 한다. 이 때, 인식된 장애물의 형태에 따라 로봇이 접촉 없이 통과할 수 있고, 발과 접촉하여 통과할 수도 있으며, 회피할 수도 있다. 이를 위해 장애물이 어떤 형태를 갖고 있는지를 분류하여 로봇의 이동경로를 생성할 때 활용 가능한 장애물 인식 및 분류 방법을 제안한다. 특히 장애물 형태를 정확히 인식하기 위한 기존 알고리즘은 많은 계산량으로 실시간 활용에 어려움이 있으며, 불필요한 장애물도 함께 추출하기 때문에 연산자원의 낭비가 불가피하다. 본 연구에서는 장애물 인식의 계산량을 줄이기 위해 장애물의 영역을 분류한 후 정확한 형상이 필요한 장애물에만 크기와 형태를 추출하도록 알고리즘의 적용 범위를 제한하여 계산량을 줄이는 방법을 제안한다.

Abstract: To generate the walking path of a humanoid robot in an unknown environment, the shapes of obstacles around the robot should be detected accurately. However, doing so incurs a very large computational cost. Therefore this study proposes a method to classify the obstacle shape into three types: a shape small enough for the robot to go over, a shape planar enough for the robot foot to make contact with, and an uncertain shape that must be avoided by the robot. To classify the obstacle shape, first, the range and the number of the obstacles is detected. If an obstacle can make contact with the robot foot, the shape of an obstacle is accurately derived. If an obstacle has uncertain shape or small size, the shape of an obstacle is not detected to minimize the computational load. Experimental results show that the proposed algorithm efficiently classifies the shapes of obstacles around the robot in real time with low computational load.

- 기호설명 -

{O} : 원점의 좌표계
 {S} : 센서의 좌표계
 P_{ij} : 센서의 측정지점 좌표
 d_{ij} : 센서의 측정거리
 θ_{ij} : 레이저 광선의 발사 각도

φ_R : 회전 행렬
 h : 목 관절의 높이
 l : 목 관절과 센서간 거리
 φ_i : 목 관절의 틸트 각도
 h_{ij}^a : i 방향의 높이 변화량
 h_{ij}^b : j 방향의 높이 변화량
 P_k : 장애물의 경계점
 P^B : 장애물 경계점의 집합

[§] 이 논문은 2012년도 대한기계학회 IT 융합부문 춘계학술대회(2012. 6. 1.-2., 송실대) 발표논문임.

[†] Corresponding Author, doikkim@kist.re.kr

© 2013 The Korean Society of Mechanical Engineers

아래 첨자

i : 목 관절의 틸트 각도 번호

j : 센서에서 측정된 순서

1. 서 론

휴머노이드 로봇은 두 발로 이동하는 특성을 통해 다양한 환경에서 장애물을 회피하며 이동할 수 있는 장점을 갖고 있다. 지금까지 휴머노이드 로봇의 이동과 관련하여 많은 연구들이 수행되었는데 달리며 이동하는 움직임,⁽¹⁾ 두 팔과 다리로 기어가는 움직임,⁽²⁾ 울퉁불퉁한 지면 및 비탈면을 이동하는 움직임,⁽³⁾ 그리고 계단 등의 이동을 위한 움직임⁽⁴⁾ 등이 구현되어 있는 상황이다. 그러나 이와 같은 연구들은 알려진 실내 환경 내에서만 로봇의 움직임을 생성할 수 있다는 단점이 있으며, 알려지지 않은 환경에서 로봇이 주변 상황과 장애물을 인식하고 이에 적합한 움직임을 스스로 생성하여 이동하는 것과 관련한 연구는 현재 초보적인 수준에서 머물러 있는 상황이다.

알려지지 않은 환경에서 로봇 주변의 장애물을 인식하고 이를 넘어서는 움직임, 밟고 이동하는 움직임, 회피하여 이동하는 움직임을 생성하기 위해서는 로봇 주변 장애물의 모양 및 크기를 정확히 인식할 수 있어야 한다. 이를 위한 기존 연구들은 주로 스테레오 카메라를 활용하여 인간형 로봇 주변에 존재하는 장애물을 측정하고 있지만 동시에 여러 개의 장애물의 크기를 추출할 수 없거나,⁽⁵⁾ 계산량이 너무 많아 실시간 활용이 어렵거나,⁽⁶⁾ 센서의 측정오류 제거를 위한 방법이 검증되지 않아 정확도가 떨어지는 문제가 존재한다.⁽⁷⁾ 특히 기존 방법들은 추출된 선으로부터 제곱 평균 근을 최소화하는 면을 구하거나,⁽⁶⁾ 확률적으로 가장 많은 점을 지나가는 면을 구하기 위해서⁽⁷⁾ 여러 회수의 반복작업을 거쳐 가장 나은 추출결과를 선택하여 사용하기 때문에 장애물 인식에 높은 계산량을 필요로 한다. 또한 스테레오 카메라는 이동하는 로봇에서 활용할 경우 이미지에 잔상이 생겨서 이로 인한 측정 오차가 발생하고, 조도에 따라 측정값이 영향을 받기도 하며, 장애물 표면에 무늬가 존재하지 않는 경우에는 원활한 거리 측정이 어렵다는 근본적인 한계점을 갖고 있다. 레이저 스캐너를 활용하여 로봇 주변의 장애물 크기를 측정하는 경우 스테레오 카메라가 갖고 있는 근본적인 문제들을 극복할 수 있으며 좀 더 정확하게 거리 측정을 할 수 있지만 측정 영역의 해상도는 상대적으로 부족하다는 단점이 있다.

본 논문에서는 2 차원 레이저 스캐너를 활용하

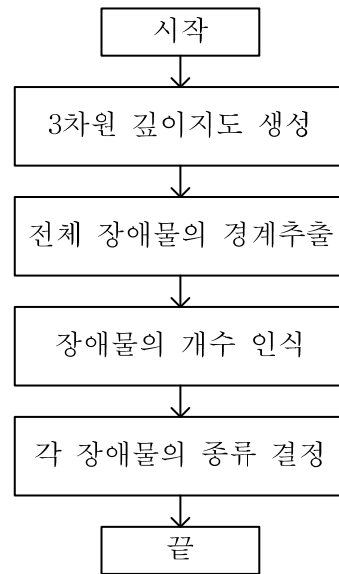


Fig. 1 Flowchart of the obstacle detection

여 알려지지 않은 환경 내의 휴머노이드 로봇 주변에 존재하는 장애물의 모양을 인식하고 분류하는 방법을 제안하였다. 레이저 스캐너로 측정된 장애물을 분류하기 위하여 우선 전체 스캔영역 중 지면을 제외한 장애물이 존재하는 영역을 인식하여 구분하고, 장애물 영역에 존재하는 3 차원 포인트 클라우드의 히스토그램 분포도를 분석하여 장애물 표면의 크기 및 모양을 분류한다.⁽⁸⁾ 장애물은 세 가지 종류로 분류하며 접촉 없이 넘어갈 수 있는 장애물, 평평한 면을 갖고 있어 발과 접촉을 통해 넘어갈 수 있는 장애물, 그리고 복잡한 모양으로 인해 회피해서 통과해야 하는 장애물로 구분한다. 발과 접촉하여 통과 가능한 장애물의 경우 보다 정확한 크기 정보가 필요하기 때문에 이 장애물 영역에 해당하는 3 차원 포인트 클라우드는 높은 계산량이 소모되는 정교한 장애물 인식 알고리즘을 적용하여 정확한 크기 및 모양을 추출한다. 전체 스캔 영역 중 장애물 영역 및 모양을 분류하여 정확한 크기가 필요한 장애물의 영역에 한해 선택적으로 정확한 모양 및 크기를 추출하기 때문에 장애물 추출에 필요한 계산량을 효과적으로 줄여줄 수 있다. 제안된 방법은 센서의 정확도 측면에서 기존의 스테레오 카메라를 활용한 방법보다 정확하게 장애물의 측정이 가능하며, 기존의 추출 방법과 비교하여 훨씬 간단한 방법과 작은 계산량으로 실시간에 가깝게 장애물의 모양 인식 및 크기 추출이 가능하다는 장점이 있다.

2. 장애물 인식방법

본 연구를 위해 휴머노이드 로봇의 머리에 2 차원 레이저 스캐너를 설치하고 목 관절에 팬 (pan), 틸트 (tilt) 움직임을 주어 로봇 주변의 3 차원 깊이 지도 (3D depth map) 을 작성할 수 있는 시스템을 구현하였다. 이를 활용하여 로봇 주변에 위치한 장애물의 형태 및 크기를 인식하는 과정을 나타낸 흐름도를 Fig. 1 에 나타내었다. 우선 레이저 스캐너를 활용하여 로봇 주변의 3 차원 깊이 지도를 측정한다. 이 지도로부터 로봇 주변 장애물과 지면간 경계지점을 추출한 이후, 장애물의 전체 개수를 파악하고, 3 차원 깊이 지도에서 각 장애물이 존재하는 영역을 추출한다. 존재영역이 추출된 각 장애물은 세 가지 장애물의 모양 중 어떤 모양에 해당하는지 인식하여 분류함으로써 전체 과정이 마무리된다.

2.1 3 차원 깊이 지도 생성

레이저 스캐너를 통해 측정된 로봇 주위의 3 차원 깊이 지도는 Fig. 2 와 같이 두 개의 좌표계 {O}와 {S}를 사용하여 표현이 가능하다. Fig. 2(a) 에서, 좌표계 {S}의 원점은 레이저 스캐너의 위치를 의미하며, 좌표계 {S}를 기준으로 i 번째 틸트 방향 각도에서 j 번째로 측정된 측정지점의 좌표인 ${}^S P_{ij}$ 는 다음과 같이 계산할 수 있다.

$${}^S P_{ij} = [d_{ij} \cdot \cos(\theta_{ij}), d_{ij} \cdot \sin(\theta_{ij}), 0]^T \quad (1)$$

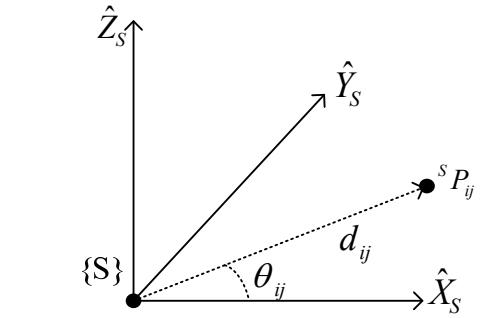
여기서 d_{ij} 와 θ_{ij} 는 각각 목 관절의 i 번째 틸트 각도에서 j 번째로 측정된 측정 거리 및 이 때의 레이저 발사각이다.

본 측정시스템의 기준이 되는 좌표계 {O}는 Fig. 2(b)에 나타나 있으며, 로봇의 두 발 중간지점이 {O}의 원점에 해당한다. 이때 {O}에서 바라보는 P_{ij} 의 위치인 ${}^O P_{ij}$ 는 식 (2)를 통해 구할 수 있다.

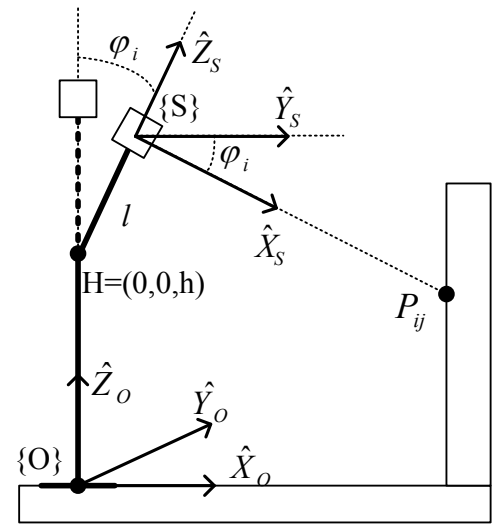
$${}^O P_{ij} = {}^O R \cdot {}^S P_{ij} + {}^O P_{SORG} \quad (2)$$

여기서 ${}^O P_{SORG}$ 는 좌표계 {O}의 원점에서 좌표계 {S}의 원점을 가리키는 벡터이고, ${}^O R$ 은 좌표계 {O}를 기준으로 했을 때 좌표계 {S}의 회전행렬이며 이는 각각 식 (3)과 식 (4)를 이용해 구할 수 있다.

$${}^O P_{SORG} = [l \cdot \sin(\varphi_i), 0, l \cdot \cos(\varphi_i) + h]^T \quad (3)$$



(a) Coordinate system of the laser scanner



(b) Coordinate system of the robot

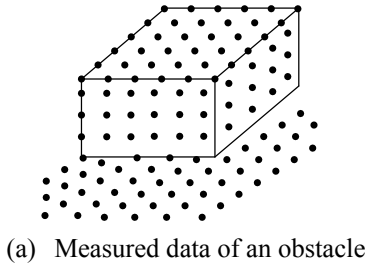
Fig. 2 Coordinate system to generate the 3D depth map around a robot

$${}^O R = \begin{bmatrix} \cos \varphi_i & 0 & \sin \varphi_i \\ 0 & 1 & 0 \\ -\sin \varphi_i & 0 & \cos \varphi_i \end{bmatrix} \quad (4)$$

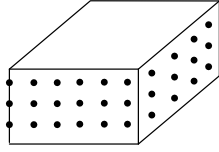
여기서 Fig. 2의 H는 목 관절의 위치, h는 목 관절의 높이, l 은 목 관절에서 센서까지의 거리, φ_i 는 목의 i 번째 틸트 방향의 회전각도이다. 참고로 레이저 스캐너는 240 deg 의 넓은 측정 영역을 갖고 있기 때문에, 목의 틸트 방향 회전만을 이용해도 로봇의 전면 및 좌우 영역까지 모두 스캔이 가능한 특징이 있다.

따라서 좌표계 {O}를 기준으로 바라보는 P_{ij} 의 좌표인 ${}^O P_{ij}$ 는 식 (1)~(4)를 통해 다음과 같이 얻어진다.

$$\begin{aligned} {}^O P_{ij} &= {}^O R \cdot {}^S P_{ij} + {}^O P_{SORG} \\ &= \begin{bmatrix} \cos \varphi_i & 0 & \sin \varphi_i \\ 0 & 1 & 0 \\ -\sin \varphi_i & 0 & \cos \varphi_i \end{bmatrix} \cdot \begin{bmatrix} d_{ij} \cdot \cos(\theta_{ij}) \\ d_{ij} \cdot \sin(\theta_{ij}) \\ 0 \end{bmatrix} + {}^O P_{SORG} \end{aligned}$$



(a) Measured data of an obstacle



(b) Detected boundary of an obstacle

Fig. 3 Boundary detection of an obstacle

$$= \begin{bmatrix} \cos \varphi_i \cdot d_{ij} \cdot \cos(\theta_{ij}) + l \cdot \sin(\varphi_i) \\ d_{ij} \cdot \sin(\theta_{ij}) \\ -\sin \varphi_i \cdot d_{ij} \cdot \cos(\theta_{ij}) + l \cdot \cos(\varphi_i) + h \end{bmatrix} \quad (5)$$

식 (5)를 활용하여 좌표축 {O}를 기준으로 바라본 로봇 주변 모든 점들의 좌표 ${}^0P_{ij}$ 를 구할 수 있고 이를 통해 모든 포인트 클라우드를 포함하는 3차원 깊이도를 얻을 수 있다. 참고로 이후의 간결한 설명을 위해 P_{ij} 는 ${}^0P_{ij}$ 를 의미하는 것으로 정의한다.

2.2 장애물의 경계 추출

장애물의 경계 인식을 위해서는 3 차원 포인트 클라우드 내의 점 중에 인접한 점간 높이 차이를 계산하여 한계값 (threshold value) 보다 클 경우 이 점을 장애물의 경계지점에 위치해 있는 것으로 인식하는 방법이 있다.⁹⁾ 인접한 점간의 높이 차이 계산을 통한 장애물의 경계지점 인식 방법은 작은 계산량으로 간편하게 사용할 수 있으며, 장애물의 경계를 정확하게 추출할 수 있다는 장점이 있다.

Fig. 3은 장애물의 경계를 추출하는 과정을 나타낸다. Fig. 3(a)에서는 3 차원 공간상의 장애물을 레이저 스캐너로 측정된 점의 위치를 표시하였으며, Fig. 3(b)에서는 레이저 스캐너로 측정된 점들 중 장애물의 경계면으로 인식되는 영역의 점을 나타내었다. 장애물의 경계를 추출하기 위한 인접한 점간 높이의 변화량은 다음의 식을 통해 구한다.

$$h_{ij}^a = |z_{ij} - z_{i+1,j}| / \sqrt{(x_{ij} - x_{i+1,j})^2 + (y_{ij} - y_{i+1,j})^2} \quad (6)$$

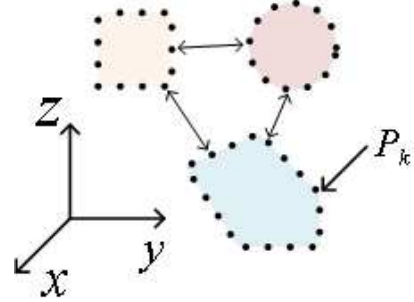


Fig. 4 Count the number of obstacles

$$h_{ij}^b = |z_{ij} - z_{i,j+1}| / \sqrt{(x_{ij} - x_{i,j+1})^2 + (y_{ij} - y_{i,j+1})^2} \quad (7)$$

이 때, (x_{ij}, y_{ij}, z_{ij}) 는 P_{ij} 의 좌표값이고, h_{ij}^a 는 i 값이 증가함에 따른 z -축 방향 높이의 변화량이며, h_{ij}^b 는 j 값이 증가함에 따른 z -축 방향 높이의 변화량이다. 모든 점 P_{ij} 에 대해 h_{ij}^a 와 h_{ij}^b 를 계산하여 한계값을 초과하는 경우가 발생하면 각각 P_{ij} 와 $P_{i+1,j}$, 혹은 P_{ij} 와 $P_{i,j+1}$ 을 장애물의 경계에 위치한 점으로 인식한다. Fig. 3(b)와 같이 구해지는 장애물의 경계에 위치한 모든 점들의 집합을 \mathbf{P}^B 라고 정의한다.

2.3 장애물의 개수 인식

앞 소단원에서 장애물의 경계에 해당하는 점들의 집합 \mathbf{P}^B 를 구했으며 여기에 포함된 k 번째의 점을 \mathbf{P}_k 로 정의한다. \mathbf{P}^B 에는 모든 장애물의 경계에 해당하는 점들이 들어 있으나 그것이 몇 개의 장애물에 대한 경계지점인지는 판단할 수가 없다. 따라서 \mathbf{P}^B 에 속한 점으로부터 장애물이 전부 몇 개가 존재하는지 파악하는 것이 필요하다.

Fig. 4와 같이 \mathbf{P}^B 에 속한 점들은 세 장애물의 경계에 위치한 점으로 이루어져 있다고 가정한다. 이 때 동일한 장애물의 경계점 간에는 서로 가까운 거리에 위치하고 있으며 다른 장애물의 경계점은 서로 먼 거리에 위치하고 있기 때문에 계층화 클러스터링 (hierarchical clustering) 방법¹⁰⁾을 사용하면 \mathbf{P}^B 에 속한 점들이 각각 어떤 장애물에 속한 점인지 구분할 수가 있다. 이를 위해 모든 \mathbf{P}_k 간의 거리를 계산하고 거리가 가까운 순서대로 각 점들을 동일한 클러스터에 포함시켜 나간다. 참고로 여기서 클러스터는 복수의 점으로 이루어진 점들의 집합을 의미한다. 한계값보다 서로 가까운 거리에 위치한 점이나 클러스터를 동일한 클러스터로 합치는 과정을 거치면, 최종적으로 한계값 이상의 거리에 위치한 클러스터들만 남겨진다. 참고로 클러스

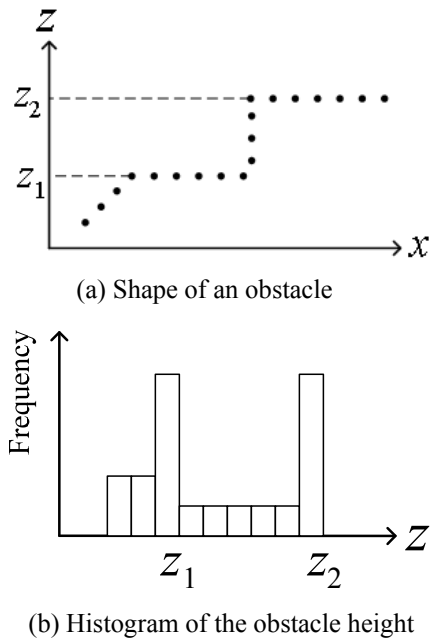


Fig. 5 Shape and histogram of an obstacle

터간의 거리는 각 클러스터에 속한 점간 거리 중 최소값으로 정의한다.

Fig. 4 에는 한계값보다 먼 거리에 위치한 세 개의 클러스터를 각기 다른 색깔로 표시하였고, 이 때 각 클러스터간 거리는 실선의 양방향 화살표로 표시하였다. 이와 같이 최종적으로 남은 클러스터의 개수가 전체 장애물의 개수가 되며, 각 클러스터에 포함된 점들이 각 장애물의 경계면에 존재하는 점으로 결정된다. 또한 각 장애물이 존재하는 면적은 각 클러스터에 포함된 점이 xy 평면에서 차지하고 있는 영역의 외곽선을 통해 얻을 수 있다.

2.4 장애물의 모양 인식

로봇 주변에 존재하는 장애물의 개수가 판별되면, 각 장애물을 모양에 따라 분류하는 작업이 필요하다. 장애물은 크기가 작아서 접촉 없이 넘어갈 수 있는 모양, 계단과 같이 발과 접촉을 통해 넘어갈 수 있는 모양, 형태가 불분명하거나 접촉면이 작거나 높이가 높아서 회피해서 지나가야 하는 모양으로 구분한다. 참고로 장애물을 인식하는 목적에 따라 구분할 장애물의 모양을 결정하게 되는데, 여기서는 로봇의 이동경로 생성에 도움이 될 수 있도록 구분할 장애물의 모양을 결정하였다.

Fig. 5(a)는 임의의 장애물 표면을 레이저 스캐너로 측정했을 때 측정 지점의 좌표를 xz 평면에 나타내었고, Fig. 5(b)는 Fig. 5(a)에 나타난 장애물의 높이 변화에

따른 히스토그램 분포도를 나타내었다. Fig. 5(a)에 표시한 장애물은 높이가 z_1 이나 z_2 에 해당할 경우 수평면이 존재하며, 높이가 z_1 이하의 값일 경우는 경사면이 존재하고, 높이가 z_1 과 z_2 사이에 위치할 경우에는 수직면이 존재한다. 장애물 표면이 수평면일 경우 Fig. 5(b)의 히스토그램에는 해당 면의 높이에서 가장 큰 측정빈도가 나타나며, 이 때 측정 빈도의 크기는 수평면의 면적과 비례한다. 또한 Fig. 5(a)에서 높이가 z_1 과 z_2 사이 구간의 수직면인 경우 히스토그램 막대그래프에는 가장 작은 측정빈도가 나타나며, 높이가 z_1 이하를 갖는 경우처럼 장애물 표면이 경사면일 경우는 수직면일 때보다 다소 높은 측정빈도가 관찰된다.

이와 같이 장애물의 모양을 판단할 때 히스토그램을 활용하면 장애물의 외형에 대한 간략한 정보를 빠르고 쉽게 얻어낼 수 있다는 장점이 있다. 특히 수평면 인식의 경우에는 히스토그램상의 측정 빈도수가 다른 형태의 면보다 확연히 증가하기 때문에 보다 정확한 추출이 가능하고, 측정 빈도수의 크기에 따라 수평면의 면적 또한 추출할 수 있다는 장점이 있다. 참고로 레이저 스캐너와 장애물간 거리가 멀어질수록 센서의 해상도 (resolution) 가 감소하여 수평면에 도착하는 레이저 광선의 개수가 감소한다. 따라서 히스토그램의 측정빈도로 수평면의 면적을 구할 경우에는 센서와 장애물간 거리가 해상도에 미치는 영향도 함께 고려하여 계산해야 한다.

장애물 표면 좌표의 높이값이 변화할 때 히스토그램을 얻은 이후, 장애물의 모양을 다음의 조건을 적용하여 분류한다. 우선 장애물의 최대 높이와 장애물이 존재하는 영역이 한계 높이 (threshold height) 및 한계 면적 (threshold area) 이내의 값을 갖는다면 대상 장애물은 접촉 없이 통과할 수 있는 장애물로 분류한다. 이 때 한계 높이는 로봇이 보행할 때 발이 지면과 떨어져 이동할 수 있는 최대 높이를 의미하며, 한계 면적은 발바닥의 모든 모서리가 들어갈 수 있는 면적을 의미한다. 장애물의 최대 높이가 한계 높이 이내에서 한계 면적 이상의 수평면으로 이루어진 경우 해당 장애물은 발과의 접촉을 통해 통과 가능한 장애물로 인식한다. 위의 두 장애물로 분류되지 않는 장애물은 통과가 어려운 장애물로 분류하여 로봇이 이동시 회피하여 지나가야 하는 장애물로 분류한다.

휴머노이드 로봇이 장애물을 통과하여 이동할 때 장애물의 크기나 표면에 충분한 면적의 수평면이 존재하는지 여부에 따라 생성되는 이동 경로가 달라지는데, 제안된 방법을 활용하면 장애물의 모양을 쉽고 간단하게 인지할 수가 있어 로봇의 이동경로 생성에 간편하게 적용할 수 있다.

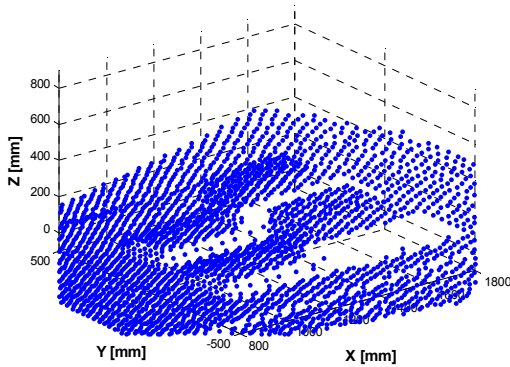


Fig. 6 Point cloud of the obstacles

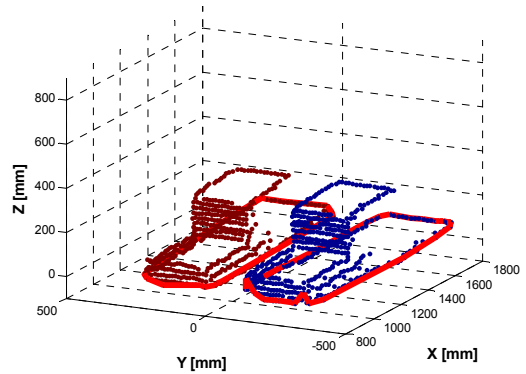


Fig. 8 Count the number of obstacles

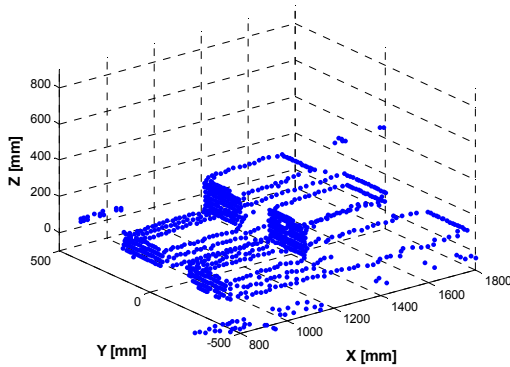


Fig. 7 Boundary detection of the obstacles

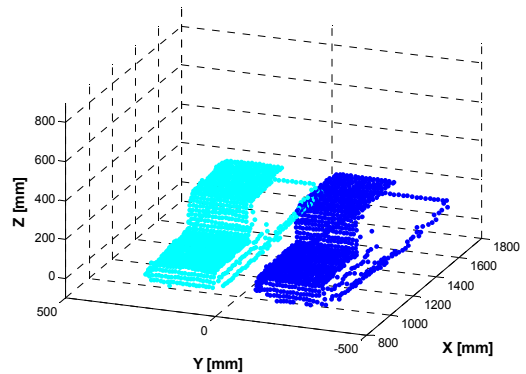


Fig. 9 Point cloud of the detected obstacles

3. 실험

이번 단원에서는 제안된 장애물 모양의 분류 방법을 실험을 통해 검증하였고 그 효과를 확인하였다. KIST의 휴머노이드 로봇인 마루 1을 실험에 활용하였고, Hokuyo사의 UBG-04LX-F01 레이저 스캐너를 거리 측정수단으로 활용하여 로봇 전면부의 3차원 깊이 지도를 생성하였다. 이 때, φ_i 는 $35 \sim 80^\circ$ 의 범위에서 회전하며 로봇 전면으로 약 500~2000 mm 거리 영역에 위치한 물체의 형상을 측정한다. 센서가 한번 틸트 방향으로 회전할 때마다 총 180회의 거리 스캔이 이루어지고, 매 회의 스캔마다 로봇 전면 54° 영역을 0.36°의 분해능으로 150여 회에 걸쳐 거리 측정을 수행한다.

Fig. 6은 장애물을 측정하여 얻은 3차원 깊이 지도를 나타낸다. 두 개의 계단이 로봇의 전면부에 서로 떨어져 위치하고 있으며 각 계단은 두 개의 계단면을 갖고 있다.

Fig. 7은 인접한 점간 높이 차이를 활용하여 장애물의 경계 영역을 추출한 모습을 보여준다. 지면에 해당되는 점들이 제거되고 장애물이 존재하는 영역의 경계지점과 면의 경계지점에 해당하는 점이 온전하게 추출되었다. 다만 센서의 측정오차로 인해 장애물 영역 외부의 지면에 해당하는 영역 중 일부 영역의 값들이 제거되지 않고 남아 있는 것을 확인할 수 있다. 식 (6)과 식 (7)을 활용하여 장애물의 영역을 추출하는 과정에서 센서의 거리측정 오차로 인해 인접한 점간 xy 방향 변화량이 0에 가깝게 측정되는 경우가 간혹 발생하는데, 이로 인해 h_{ij}^* 혹은 h_{ij}^* 의 값이 크게 증가하여 장애물의 경계점으로 인식하기 때문이다. 이로 인해 장애물의 경계점 추출시 Fig. 7과 같은 오차가 발생할 수 있지만 이는 다음 과정에서 제거가 가능하다.

Fig. 8은 장애물의 개수를 인식한 결과를 보여준다. Fig. 7에서 추출된 점들은 계층화 클러스터

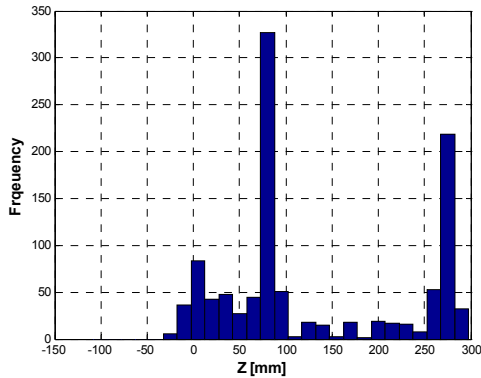


Fig. 10 Histogram of the left stairs

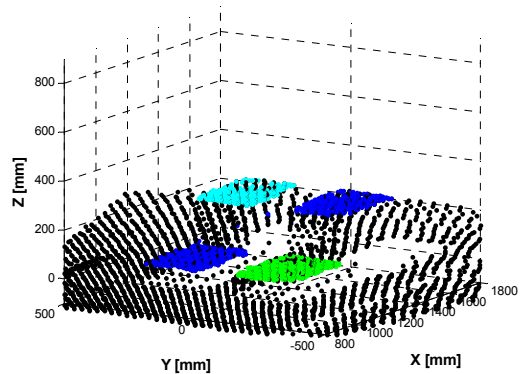


Fig. 12 Extracted planes on the obstacles

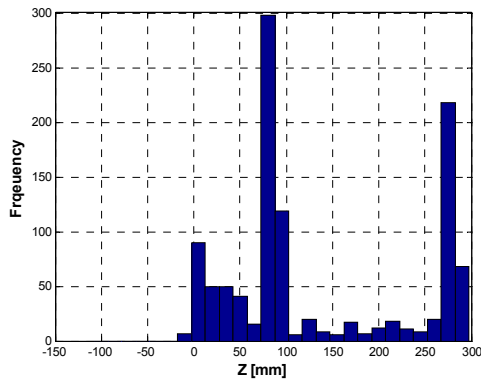


Fig. 11 Histogram of the right stairs

링 과정을 통해 한계거리 이내의 가까운 거리에 위치한 점들을 동일 클러스터로 포함시키는 과정을 거치는데, 최종적으로 생성된 클러스터가 한계값보다 작은 개수의 점을 가지며 미미한 면적상에 존재하는 경우 이 점들은 레이저 스캐너의 측정오차로 판단하여 제거한다. 이 과정을 통해 Fig. 8에서는 서로 다른 두 개의 장애물이 인식되었고 이를 다른 색깔로 표시하였으며, 각 장애물이 존재하는 영역은 붉은 색깔로 표시하였다. 참고로 붉은 색으로 표기된 영역은 각 장애물 위의 점들을 xy 평면으로 투과하여 점들이 xy 평면상에 존재하는 경계영역을 추출하는 방법으로 구한다.

Fig. 9에서는 Fig. 8에서 붉은색 실선으로 표시한 장애물의 경계영역 정보를 활용하여 Fig. 6의 3차원 깊이 지도에 표시된 점들 중 장애물의 경계 영역 내에 존재하는 모든 점을 나타내었다. 초기의 3차원 깊이 지도 내에 존재하는 점으로부터 장애물을 측정된 점만을 온전히 추출할 수 있었으며 분리되어 위치한 장애물은 서로 다른 장애물로

인식한 것을 알 수 있다.

Fig. 9에서 왼쪽 장애물과 오른쪽 장애물 상에 존재하는 점들의 z 축 방향 좌표값 분포는 Fig. 10과 11에서 히스토그램을 활용하여 표시하였다. 두 장애물이 동일한 외형을 갖고 있기 때문에 두 그래프 모두 73~88 mm 영역과 268~283 mm 영역에서 높은 크기의 막대그래프가 관찰되는데 이는 좌우 계단의 두 계단면 높이가 이 구간의 높이를 갖기 때문이다. 히스토그램의 막대그래프 높이가 한계값을 넘는 경우 장애물에서 그 구간의 높이에 일정 면적 이상의 수평면이 존재함을 파악할 수 있다.

히스토그램에서 가장 높은 막대그래프의 높이가 한계값보다 작다는 것은 장애물 표면에 수평면이 존재하지 않거나 그 면적이 충분히 크지 않다는 의미이며 이 경우는 밟고 넘어갈 수는 없는 장애물로 분류된다. 이 때, 장애물의 최대 높이가 한계 높이보다 작으며 장애물의 존재 영역이 한계 면적보다 작은 경우에 한해 접촉없이 넘어갈 수 있는 장애물로 분류되며 그 이외의 장애물은 회피하여 지나가야 하는 장애물로 분류된다.

Fig. 12는 Fig. 6에서 표시한 3차원 깊이 지도 중 Fig. 10과 11의 히스토그램상으로 가장 높은 네 개의 막대그래프에 해당하는 점들을 다른 색으로 구분하여 표현한 것이다. 이처럼 히스토그램을 활용하여 장애물의 수평면 영역을 비교적 정확하게 추출하는 것이 가능하다. 히스토그램을 활용한 장애물 모양 분류 방법은 작은 계산량으로 장애물의 모양을 추출하거나 표면 크기를 추출하는데 유용하게 활용이 가능하다. 클러스터링 방법을 적용하여 센서의 측정오차를 제거하였기 때문에 깨끗한 장애물의 모습 또한 얻을 수가 있었다.

4. 결 론

본 논문에서는 로봇 주변을 측정한 3 차원 깊이 지도로부터 로봇 주변에 위치한 장애물의 존재 영역 및 개수를 추출하는 방법을 제안하였으며, 장애물의 모양을 세 가지 형태로 구별하는 것이 가능하기 때문에 로봇의 이동 경로 생성에 활용할 수 있다. 인접한 점간 높이의 변화량, 계층화 클러스터링, 그리고 히스토그램을 활용하여 장애물의 모양을 구분하는 전 과정을 설명하였으며, 기존의 복잡한 알고리즘을 활용하지 않고 매우 작은 계산량 만으로 간단하게 연산이 가능하도록 구현하였다는 특징이 있다.

계층화 클러스터링을 활용하여 로봇 주변의 장애물 개수를 추출하였으며 구별 가능한 장애물의 최대 개수에는 제한이 없다. 또한 떨어져 위치한 모든 장애물을 구분하는 과정에서 센서의 측정 오차로 판명난 점은 제거가 가능하여 정확한 장애물의 영역만을 추출할 수 있다.

실험을 통해 제안된 방법의 타당성을 검증하였으며, 두 계단을 갖고 있는 서로 떨어진 장애물의 영역을 정확히 추출하였고, 두 장애물 위에 위치한 총 네 개의 계단면에 대해서도 성공적인 추출이 가능하였다.

참고문헌

- (1) Fu, C. and Chen, K., 2008, "Gait Synthesis and Sensory Control of Stair Climbing for a Humanoid Robot," *IEEE Transactions on Industrial Electronics*, Vol. 55, No. 5, pp. 2111~2120.
- (2) Tajima, R., Honda, D. and Suga K., 2009, "Fast Running Experiments Involving a Humanoid Robot," *IEEE International Conference on Robotics and Automation*, pp. 1571~1576.
- (3) Kim, J.Y., Park, I.W. and Oh, J.H., 2007, "Walking Control Algorithm of Biped Humanoid Robot on Uneven and Inclined Floor," *Journal of Intelligent and Robotic Systems*, vol. 48, no. 4, pp 457~484.
- (4) Park, C.-S., Ha, T., Kim, J. and Choi, C.-H., 2010, "Trajectory Generation and Control for a Biped Robot Walking Upstairs," *International Journal of Control, Automation and Systems*, Vol. 8, No. 2, pp. 339~351.
- (5) Okada, K., Ogura, T., Haneda, A. and Inaba, M., 2005, "Autonomous 3d Walking System for a Humanoid Robot Based on Visual Step Recognition and 3D Foot Step Planner," *Proceedings of the IEEE International Conference on Robotics and Automation*, pp. 623~628.
- (6) Gutmann, J., Fukuchi, M. and Fujita, M., 2008, "3d Perception and Environment Map Generation for Humanoid Robot Navigation," *The International Journal of Robotics Research*, Vol. 27, No. 10, pp. 1117~1134.
- (7) Heracles, M., Bolder, B. and Goerick, C., 2009, "Fast Detection of Arbitrary Planar Surfaces from Unreliable 3D Data," *IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 5717~5724.
- (8) Choi, J.-M., Lee, S.-J. and Won, M., 2011, "Self-Learning Navigation Algorithm for Vision-Based Mobile Robots Using Machine Learning Algorithms", *Journal of Mechanical Science and Technology*, Vol. 25, No. 1, pp. 247~254.
- (9) Park, C.-S., Seo, E.-H., Kim, D. and You, B.-J., 2011, "Stair Boundary Extraction Using the 2D Laser Scanner," *IEEE International Conference on Mechatronics and Automation*, pp. 1538~1543.
- (10) Johnson, S. C., 1967, "Hierarchical Clustering Schemes," *Psychometrika*, Vol. 32, No. 3, pp. 241~254.