

무인로봇을 위한 3D 월드모델에 기초한 Binary 장애지형의 판정

Classification of Binary Obstacle Terrain Based on 3D World Models for Unmanned Robots

진강규* 이현식** 이윤형*** 이영일**** 박용운****
 Gang-Gyoo Jin Hyun-Sik Lee Yun-Hyung Lee Young-Il Lee Yong-Woon Park

Abstract

Recently, the applications of unmanned robots are increasing in various fields including surveillance and reconnaissance, planet exploration and disaster relief. To perform their missions with success, the robots should be able to evaluate terrain's characteristics quantitatively and identify traversable regions to progress toward a goal using mounted sensors. Recently, the authors have proposed techniques that extract terrain information and analyze traversability for off-road navigation of an unmanned robot. In this paper, we examine the use of 3D world models(terrain maps) to classify obstacle and safe terrain for increasing the reliability of the proposed techniques. A world model is divided into several patches and each patch is classified as belonging either to an obstacle or a non-obstacle using three types of metrics. The effectiveness of the proposed method is verified on real terrain maps.

Keywords : Unmanned Robot(무인로봇), 3D World Model(3D 월드모델), Terrain Map(지형맵), Binary Obstacle Terrain (이진장애지형)

1. 서론

최근 행성 탐사, 군사 작전에서의 감시 및 정찰, 제

난구조 등의 임무를 수행할 수 있는 무인로봇의 개발과 사용 예가 증가하고 있다. 무인로봇이 예기치 않은 지형환경에 직면하더라도 목적지까지 가장 안전하고 빠른 경로를 찾아 임무를 수행하기 위해서는 주행성 분석(Traversability Analysis)과 장애물 탐지(Obstacle Detection)는 해결되어야 할 과제이고 지금까지 많은 연구들이 수행되어 왔다.

주행성 분석은 크게 전역 주행성 분석(GTA : Global Traversability Analysis)과 지역 주행성 분석(LTA : Local Traversability Analysis)으로 구분된다. GTA는 위성, 레

† 2009년 2월 26일 접수~2009년 6월 26일 게재승인

* 한국해양대학교(Korea Maritime University)

** 국방기술품질원(DTAQ)

*** 한국항공우주연구원 부산연수원(KPTIP)

**** 국방과학연구소(ADD)

책임저자 : 진강규(ggjin@hhu.ac.kr)

이더 등으로부터 입수되는 계획된 지역의 광역 스케일의 저해상도 지형도로부터 로봇이 통과하기 어려운 산, 호수, 골짜기, 숲 등의 지역을 오프라인적으로 식별하는 것이고, LTA는 반경 수 미터에서 수 십 미터 이내의 지역에 대해 스테레오 비전 카메라나 레이저 거리측정기 등으로 얻어진 고해상도 영상정보를 기반으로 해당 지형의 통과가능 여부를 온라인적으로 판단하는 것이다. 특히 LTA는 근거리 지역을 커버하기 때문에 안전주행을 판단하는 중요한 요소이고 지형들의 물리적, 기하학적 성질에 좌우되게 되므로 기울기(Slope)와 거칠기(Roughness)가 주 연구 대상이었다.

Langer 등^[1]은 최저 및 최고 고도와 기울기를 얻고 이를 지형 셀의 주행가능 여부를 판단하는데 사용하였고, 3D 지형 데이터와 최소자승법으로 추정된 곡면으로부터 기울기를 계산하는 해석적인 방법들은 Gennery^[2], Huajun 등^[3]에 의해 제안되었다. 한편 Hoffman 등^[4]은 로봇의 주행에 영향을 주는 크기, 주파수, 자기상관 함수를 추정하는 벡터 거칠기를 제안하였고, 영상 데이터로부터 추출되는 장애물의 크기와 집적도를 입력으로 퍼지추론하여 거칠기를 추정하는 법이 Seraji^[5]에 의해 제안되었다. Yokoya^[6], Huajun 등^[3]은 프랙탈 이론을 이용하였고, Tunstel 등^[7]은 신경회로망을 결합한 Texture analysis approach를 검토하였다.

장애물 탐지 또한 자율로봇이 가져야 할 핵심 기능으로서 근접 센서를 통해 주행과 안전에 방해가 되는 사물과 지형을 인지하는 것을 말한다. 센서의 한계, 지형환경의 예측불가능성으로 장애물 탐지는 결코 쉬운 문제가 아니며 지난 수년동안 많은 연구들이 있어 왔다. Procopio 등^[9]은 전통적인 비전 접근법에 기계학습기법(Machine Learning Technique)을 결합시키는 연구를 수행하였고, Chang 등^[10]은 격자기반과 센서기반의 장애탐지를 맵핑기법과 결합하는 것을 제안하였고, Rankin 등^[11]은 음각(Negative) 장애물의 야간 탐지 성능을 높이기 위해 스테레오 비전과 열적외선 정보를 융합하는 문제를 다루었고, Dima 등^[12]은 다수의 센서 및 장애물 분류기에 의해 분류된 데이터의 융합을 통해 신뢰도를 높이는 연구를 하였다. 이러한 방법들 중에는 특정 로봇과 사용 환경을 염두에 두고 수행한 연구가 다수이며, 각각 강점과 약점을 가지고 있다.

관련 연구로 최근 저자들은 국방과학연구소가 민관학연 협력으로 추진중인 '네트워크 기반 다목적 견마형 로봇 개발 연구'의 일환으로 3D 월드모델을 기반으

로 지형정보 추출^[13]과 주행성 분석 및 주행속도 추정^[14] 기법을 제안하였다. 지형의 월드모델(World Model)은 같은 크기의 다수 패치(Patch)로 나뉘고, 지형정보 추출 모듈은 패치 지형의 기울기와 거칠기를 추출하고, 퍼지기반의 주행성분석 및 속도추정 모듈은 이를 입력으로 하여 그 지역의 주행속도를 추정한다. 제안한 기법은 로봇의 중심점이 패치의 중앙에 정지해 있다는 가정 하에 패치의 기울기와 거칠기를 추정해 사용하였고, 패치 내에서 로봇 이동 시 발생할 수 있는 장애 상황을 고려하지 못하였다. 따라서 선행연구의 신뢰성을 높이기 위해 로봇이 패치 내에서 가상주행하는 동안 장애여부를 사전에 탐지할 필요가 있다.

이를 위해 3가지 형태의 Binary 장애지형을 판정하는 측도를 제안한다. 극복할 수 없는 장애지형은 진행방향 인접 셀 간의 고도차를 측도로 탐지하며, 차체 접촉 장애지형은 로봇 좌우 앞바퀴 또는 앞바퀴와 중간바퀴 사이에 위치하는 지형의 기하학적 형태로부터 탐지한다. 또한 Roll Over 장애지형은 바퀴가 직접 닿는 지표면의 고도와 최소자승법으로 추정된 평면 방정식의 진행방향과 측면방향의 기울기로부터 판단한다. 특히 장애지형 여부는 해당지역 지형의 기하학적 성질뿐만 아니라 로봇이 통과할 방향과도 밀접한 관계를 가지므로 방향성을 고려하여 '주행가' 또는 '주행불가'를 판정한다.

제안된 알고리즘의 유용성은 실제 월드모델을 이용하여 확인된다.

2. 주행속도추정 서브시스템의 개요

Fig. 1은 견마로봇(이하 로봇)의 외형을 보여주고 있다. 로봇은 정적/동적 장애물이 존재하는 야지에서 주어진 임무를 수행하기 위해서는 자율적으로 주행할 수 있어야 하고, 센서로부터 지형 정보를 획득해 안전하면서도 짧은 경로를 탐색할 수 있어야 하고 또 계획된 경로를 이탈하지 않고 주행할 수 있어야 하기 때문에 하드웨어적으로나 소프트웨어적으로 많은 서브시스템들이 결합된 대단히 복잡하고 첨단화된 시스템이다.

이 중 주행속도추정 서브시스템은 크게 4개의 모듈 즉, 필터링, Binary 장애지형 판정, 지형정보 추출과 퍼지기반의 주행성분석 및 속도추정 모듈로 구성된다 (Fig. 2 참조).



Fig. 1. 견마로봇

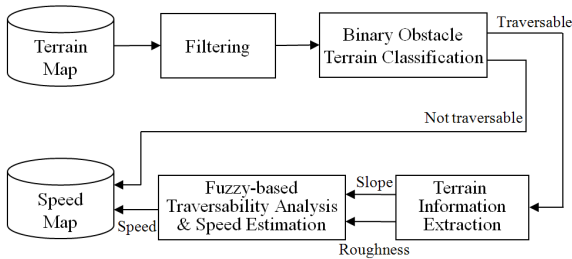


Fig. 2. 로봇을 위한 주행속도추정 서브시스템

필터링은 로봇에 탑재된 센서로부터 신호를 계측하고 처리하는 과정에서 생기는 예기치 않은 임펄스형(Spike-like)의 잡음을 제거해 주고, Binary 장애지형 판정은 필터링된 월드모델을 기반으로 해당 지역으로의 진입이 안전한지 위험한지를 판정한다. 만약 ‘주행불가’이면 그 지역의 속도는 0이 되고 장애판정은 다른 지역으로 넘어가게 되지만, ‘주행가능’이면 퍼지기반의 주행성분석 및 속도추정 모듈은 지형정보 추출 모듈로부터 기울기와 거칠기를 입력받아 주행속도를 추정하게 된다. 여기서 작성된 속도맵은 로봇의 경로계획에 이용된다.

3. 월드모델의 분할과 필터링

가. 월드모델의 분할

로봇은 매 제어 사이클동안 탑재된 비전 센서와 레이저 센서로부터 입수되는 정보를 융합해 직교좌표계(Cartesian Coordinate)의 3D 월드모델(지형 고도맵)을 제공하므로 월드모델은 Fig. 3(a)와 같이 동일한 크기의 여러 개의 사각형 즉, 패치(Patch)로 분할되고 패치는 Fig. 3(b)와 같이 $n \times m$ 셀(Cell)로 구성된다. 각 셀은 위치 정보 (x_i, y_j) 와 고도 정보 $z_{ij} = z(x_i, y_j)$ 를 가지며,

n 과 m 은 로봇의 방향과는 상관없이 패치가 로봇을 완전히 덮을 수 있도록 적절히 선택된다.

본 연구에서는 편의상 각 패치의 중앙에 로봇의 중심점(Robot Center Point)이 있는 것으로 간주한다.

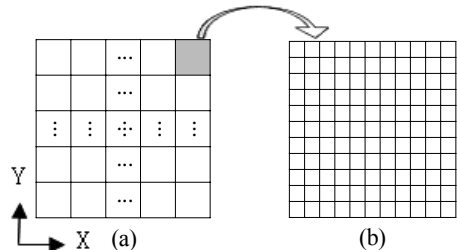


Fig. 3. 월드모델의 분할과 패치

나. 필터링

센서로부터 지형의 영상정보를 획득하고 처리하는 과정에서 예기치 않은 임펄스 잡음이 섞여 왜곡될 수 있는데, 이를 그대로 사용하면 장애탐지에 영향을 줄 수 있다.

잡음을 제거하기 위해 구현이 쉽고, 임펄스 잡음 제거에 효과적인 미디언 필터(Median Filter)를 사용한다. 미디언 필터는 비선형 공간 필터로서 저주파 통과 필터에 비해 경계선이 잘 유지되는 특징이 있다.

미디언 필터는 윈도우의 형태에 따라 수평형, 수직형, 교차형, 블록형, X자형 등으로 구분되며, 일례로 윈도우 크기(Window Size)가 3인 2D 블록형 필터는 Fig. 4와 같이 (i, j) 셀 중심으로 3×3 블록 안의 9개 셀의 고도 값을 오름 또는 내림차순으로 정렬하고 그 중앙값을 취해 (i, j) 셀의 고도 값으로 변경함으로써 잡음을 제거한다. 데이터 정렬은 Quick sort나 Insertion sort 등을 써서 할 수 있고 그 수가 많을 때는 Quick sort가, 적을 때는 Insertion sort가 빠르고 효과적이다. 한편 윈도우 크기는 홀수 값(3, 5, 9,...)을 취한다.

$Z_{i-1,j+1}$	$Z_{i,j+1}$	$Z_{i+1,j+1}$
$Z_{i-1,j}$	$Z_{i,j}$	$Z_{i+1,j}$
$Z_{i-1,j-1}$	$Z_{i,j-1}$	$Z_{i+1,j-1}$

Fig. 4. 고도 맵의 3×3 마스크

4. Binary 장애지형 판정

3D 월드모델에 기초한 Binary 장애지형 판정은 Fig. 3(a)의 모든 패치에 대해 진행되며 패치 지형이 주행에 장애가 되는 환경 요소를 가지고 있는지의 유무 즉, ‘주행가(Traversable)’ 또는 ‘주행불가(Non-Traversable)’를 판단하게 된다. 이 정보는 지형정보 추출과 주행성분석 및 속도추정 모듈의 동작여부를 결정하는데 사용된다. ‘주행가’이면 이후의 두 모듈에 의해 속도가 추정되지만, ‘주행불가’이면 해당 패치에서의 속도는 0이 되고 장애지형 판정은 다음 패치로 넘어가게 된다.

패치 지형의 장애 여부는 로봇과 주변 지형간의 물리적, 기하학적 성질, 로봇의 속도, 마찰계수 등 여러 요인들을 동시에 고려해 판단해야겠지만 이는 현실적으로 대단히 어렵기 때문에 여기서는 단순하면서도 몇 가지 안전을 고려한 측도(Metrics)를 사용한다. 장애 판단을 위한 측도는 Fig. 5와 같이 좌측 앞바퀴가 해당 패치의 좌측 하단(상단)에서 시작하여 Y(X)방향으로 상단(우측) 끝까지 진행하고, 다시 하단(좌측)의 1셀 오른(아래)쪽에서 시작하여 상단(우측) 끝까지 진행하는 순서로 반복하면서 마지막으로 우측 앞바퀴가 우측 상단(하단)에 도달할 때까지 로봇이 가상주행을 하는 동안 계량된다. 여기서는 편의상 로봇이 Y방향으로 움직이는 경우만을 고려하였다. X방향은 이를 참조하면 된다.

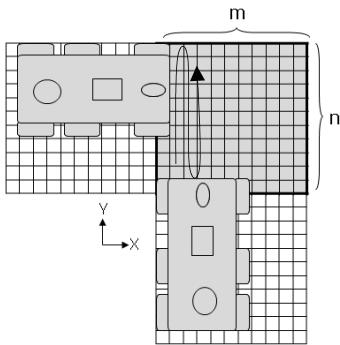


Fig. 5. 로봇의 가상 움직임

가. 극복할 수 없는 장애

극복할 수 없는 장애지형 판정은 Fig. 6과 같이 로봇이 전진할 때 바퀴가 전방의 지형을 넘을 수 있는지의 여부를 판단하는 것으로서 앞바퀴를 기준으로 한다. 식 (1)과 같이 월드모델의 고도정보로부터 진행

방향의 현 위치와 이전 위치 간의 고도차 절대값의 최대값이 극복할 수 없는 높이인 경우 장애물로 간주된다.

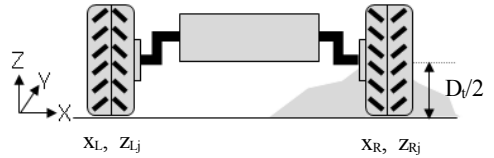


Fig. 6. 극복할 수 없는 장애

$$\Delta \bar{z}_j = \max_{1 \leq i \leq m} |z_{ij} - z_{i,j-1}| \quad (1a)$$

$$b_1 = \begin{cases} 0, & \Delta \bar{z}_j < D_t/2 \quad (1 \leq j \leq n) \\ -1, & \text{elsewhere} \end{cases} \quad (1b)$$

여기서 D_t 는 바퀴의 직경을, $z_{i0}(1 \leq i \leq m)$ 는 이전 패치의 고도를 의미하고, $b_1 = 0$ 는 Y 방향으로 주행이 가능한 패치 지형임을, $b_1 = -1$ 은 주행이 불가능한 지형임을 의미한다.

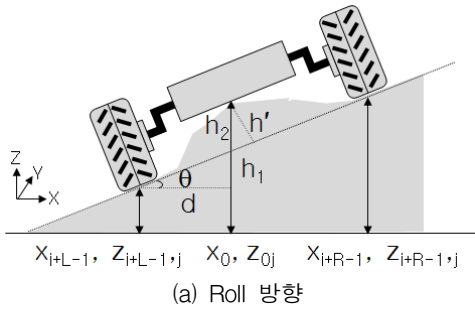
나. 차체 접촉 장애

어떤 패치 지형은 진행방향으로의 고도차는 허용치보다 작아 문제가 안 되지만, Fig. 7과 같이 두 바퀴 사이의 지표면이 차체의 하부 바디에 닿는 경우가 발생되면 심각한 손상을 줄 수 있다. 이와 같이 차체 접촉 장애지형 판정은 지면과의 접촉 여부를 판단하는 측도로서 Fig. 7과 같이 Roll 방향(로봇의 측면방향)은 좌우 앞바퀴를 기준으로, Pitch 방향(로봇의 진행방향)은 바퀴 사이의 간격이 상대적으로 다소 큰 앞바퀴와 중간바퀴를 기준으로 한다.

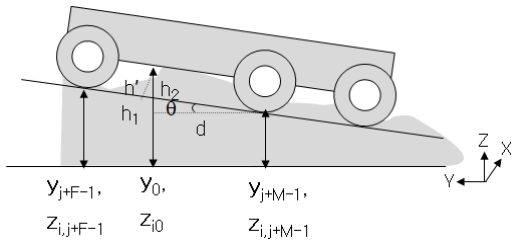
Fig. 7(a)의 경우 기울기 θ 의 근사값은 다음 식으로 쓸 수 있다.

$$\theta = \tan^{-1} \left(\left| \frac{z_{i+R-1,j} - z_{i+L-1,j}}{x_{i+R-1} - x_{i+L-1}} \right| \right) \quad (2)$$

여기서 L과 R은 각각 좌우측 바퀴의 위치를 셀단위로 표시한 것이다. 양 바퀴 사이의 최고 고도값과 이때의 X 방향의 위치를 각각 z_0, x_0 라 하면, 바퀴 하단의 연장선을 기준으로 한 접촉높이 h' 는 다음 식으로부터 구할 수 있다.



(a) Roll 방향



(b) Pitch 방향

Fig. 7. 차체 접촉 장애

$$d = x_0 - x_{i+L-1} \quad (3a)$$

$$h_1 = d \tan \theta \quad (3b)$$

$$h_2 = z_{0j} - z_{i+L-1,j} - h_1 \quad (3c)$$

$$h' = h_2 \cos \theta \quad (3d)$$

결국 h' 가 허용치 h_Y 를 넘게 되면 접촉이 일어나므로 이 지형은 식 (4)를 통해 Binary 장애지형으로 구분되게 된다.

$$b_2 = \begin{cases} 0, & h' < h_Y \\ -1, & \text{elsewhere} \end{cases} \quad (1 \leq i \leq m-W+1, 1 \leq j \leq n) \quad (4)$$

단,

$$h_Y = \begin{cases} D_t/2, & x_{i+L-1} + W_t < x_0 < x_{i+R-1} - W_t \\ D_t/4, & \text{elsewhere} \end{cases} \quad (5)$$

여기서 W 는 차체의 폭을 셀단위로 표시한 값이고, W_t 는 바퀴의 폭을 의미한다.

Pitch 방향의 장애지형 판정은 Roll 방향의 경우와

유사하게 전개될 수 있으며 다음 식을 얻는다.

$$b_3 = \begin{cases} 0, & h' < h_X \\ -1, & \text{elsewhere} \end{cases} \quad (1 \leq i \leq m-W+1, 1 \leq j \leq n) \quad (6)$$

단,

$$h_X = \begin{cases} D_t/2, & y_{j+M-1} + D_t/2 < y_0 < y_{j+F-1} - D_t/2 \\ D_t/4, & \text{elsewhere} \end{cases} \quad (7)$$

여기서 F 와 M 은 각각 앞바퀴와 중간바퀴의 위치를 셀단위로 표시한 것이다.

다. Roll over 장애

지형의 기울기 또한 로봇의 안전 주행에 영향을 미치는 중요한 요소로서, 한계치를 초과한 지역을 주행하다 전복되면 심각한 손상을 입게 된다. 앞의 경우와 같이 로봇이 가상 주행을 하는 동안 Fig. 8과 같이 6개의 바퀴가 지면과 닿는 지점의 (x,y,z) 데이터와 최소자승법으로 Roll 방향과 Pitch 방향의 순시 기울기 (Instantaneous Slope)를 추출한다.

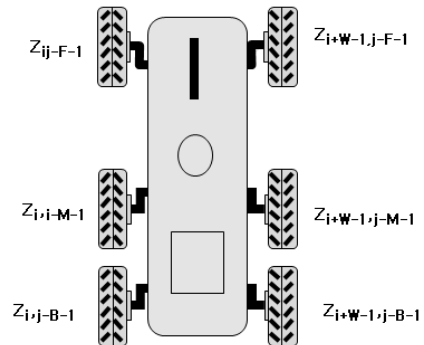


Fig. 8. 바퀴의 위치와 고도

바퀴가 닿는 지형은 3차원 공간에서 식 (8)의 근사 평면 방정식으로 표시될 수 있다.

$$z_{ij} = f(x_i, y_j) = a_0 + a_1 x_i + a_2 y_j + \varepsilon \quad (8)$$

여기서 a_0, a_1, a_2 는 평면 방정식의 미지 계수이며, ε 는 통계적으로 각 셀마다 독립이며, 평균이 0이고 분산이 같은 오차로 간주한다. 주어진 데이터를 식 (8)에 대입하고 행렬-벡터로 표현하면 식 (9)를 얻을 수

있다.

$$\mathbf{z} = \mathbf{Ax} + \boldsymbol{\varepsilon} \quad (9)$$

단,

$$\mathbf{z} = \begin{bmatrix} Z_{i,j-F-1} \\ Z_{i+W-1,j-F-1} \\ Z_{i,j-M-1} \\ \vdots \\ Z_{i+W-1,j-B+1} \end{bmatrix} \in \mathbf{R}^6, \mathbf{x} = \begin{bmatrix} a_0 \\ a_1 \\ a_2 \end{bmatrix} \in \mathbf{R}^3,$$

$$\boldsymbol{\varepsilon} = \begin{bmatrix} \varepsilon_1 \\ \varepsilon_2 \\ \varepsilon_3 \\ \vdots \\ \varepsilon_6 \end{bmatrix} \in \mathbf{R}^6, \mathbf{A} = \begin{bmatrix} 1 & X_i & Y_{j-F-1} \\ 1 & X_{i-W+1} & Y_{j-F-1} \\ 1 & X_i & Y_{j-M-1} \\ \vdots & \vdots & \vdots \\ 1 & X_{i+W-1} & Y_{j-B+1} \end{bmatrix}$$

B는 뒷바퀴의 위치를 셀단위로 표시한 것이다.

오차를 최소화 하는 계수 벡터 \mathbf{x} 를 추정하기 위해 평가함수 J를 다음과 같이 정의한다.

$$J = \boldsymbol{\varepsilon}^T \boldsymbol{\varepsilon} = (\mathbf{z} - \mathbf{Ax})^T (\mathbf{z} - \mathbf{Ax}) \quad (10)$$

식 (10)에서 $\partial J / \partial \mathbf{x} = 0$ 을 만족하는 추정치 $\hat{\mathbf{x}}$ 는 다음 선형 연립방정식의 해가 된다.

$$\mathbf{A}^T \mathbf{Ax} = \mathbf{A}^T \mathbf{z} \quad (11)$$

여기에 QR분해법을 적용하면 해를 쉽게 구할 수 있다. 바퀴의 위치가 서로 달라 $\rho \mathbf{A} = 3$ 이므로 6×3 행렬 \mathbf{A} 는 6×3 직교행렬 \mathbf{Q} 와 3×3 상삼각행렬 \mathbf{R} 의 곱으로 분해($\mathbf{A} = \mathbf{QR}$)될 수 있고 또 직교행렬의 성질 $\mathbf{Q}^T \mathbf{Q} = \mathbf{I}$ 를 이용하면 식 (11)은 다음과 같이 쓸 수 있다.

$$\mathbf{R}\hat{\mathbf{x}} = \mathbf{Q}^T \mathbf{z} \quad (12)$$

\mathbf{R} 이 상삼각행렬이기 때문에 후진대입법을 적용하면 $\hat{\mathbf{x}} = [a_0 \ a_1 \ a_2]^T$ 를 구할 수 있고, Roll 방향의 기울기 α_x 와 Pitch 방향의 기울기 α_y 는 각각 a_1 과 a_2 가 된다.

$$\alpha_x = \frac{\partial \hat{f}(x,y)}{\partial x} = a_1 \quad (13a)$$

$$\alpha_y = \frac{\partial \hat{f}(x,y)}{\partial y} = a_2 \quad (13b)$$

이 값들이 S_x 와 S_y 를 넘게 되면 장애지형으로 판정 된다.

$$b_i = \begin{cases} 0, & |\alpha_x| < S_x \text{ and } |\alpha_y| < S_y \\ -1, & \text{elsewhere} \end{cases} \quad (1 \leq i \leq m-W+1, 1 \leq j \leq n) \quad (14)$$

여기서 S_x, S_y 는 각각 로봇이 Roll 방향과 Pitch 방향으로 주행 가능한 지형의 최대 기울기를 의미한다.

한편 추정치의 오차제곱합(SSE)은 다음과 같다.

$$SSE = \mathbf{z}^T \mathbf{z} - \mathbf{z}^T \mathbf{A} \hat{\mathbf{x}} \quad (15)$$

5. 모의실험

제안된 방법의 유효성을 검증하기 위하여 여러 프레임의 실제 3D 월드모델을 이용하여 모의실험을 실시하였다. Fig. 9는 그중 하나를 그린 것이다. 맵은 모두 100×100 셀로 구성되어 있고, 셀의 크기는 $0.25m \times 0.25m$ 이며, 전체 지역은 $25m \times 25m$ 이다.

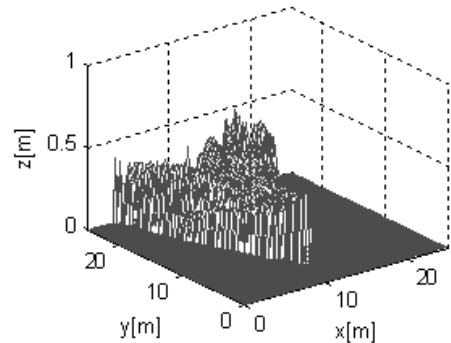


Fig. 9. 필터링 이전의 지형

가. 필터링

Fig. 10은 윈도우 크기 3으로 필터링한 후의 지형을 보여준다. 기대할 수 있듯이 Isolated 잡음이 제거됨을 확인할 수 있다. 윈도우 크기가 클수록 거친 모양이 더 무더지지만 원래의 정보가 왜곡될 수 있고 또 연산부담도 증가하게 된다.

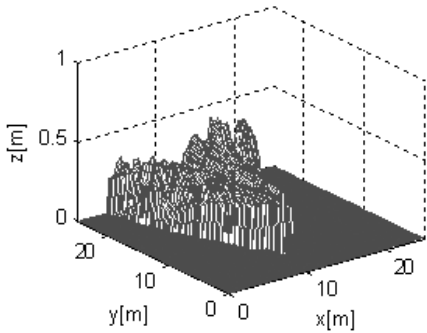


Fig. 10. 필터링 후의 지형(윈도우 크기 = 3)

나. Binary 장애지형 판정

여러 다른 프레임의 월드모델로부터 11×11셀 크기의 패치 4개를 임의로 선택하고 그린 것이 Fig. 11이다. Table 1은 제안한 방법으로 Y방향과 X방향으로의 Binary 장애지형을 판정한 결과를 고도정보와 함께 표시한 것이다.

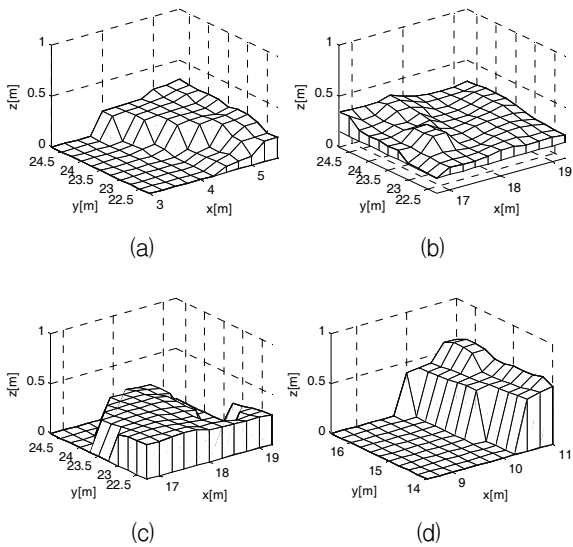


Fig. 11. 월드모델의 11×11 masks

표에서 보면 Fig. 11(a)의 지형은 Y방향으로 b_3 장애와 X방향으로 b_2 장애 요소가 공존하며, (b)는 X방향으로의 b_3 가, (c)는 Roll over 장애물을 제외한 X, Y방향으로 모든 장애요소가, (d)는 Y방향으로의 b_1, b_3, b_4 와 X방향으로의 b_1, b_2 장애 요소가 있음을 보여준다. 그림을 통해서도 (c), (d)가 (a), (b)보다 안전하지 못한 지형임을 직관적으로 느낄 수 있다.

Table 1. Binary 장애지형 판정 결과

Fig.	Binary obstacles(Y/X)				Elevation		
	b_1	b_2	b_3	b_4	Z_{min}	Z_{max}	Z_{ave}
11(a)	0/0	0/-1	-1/0	0/0	0.000	0.312	0.112
11(b)	0/0	0/0	0/-1	0/0	0.154	0.368	0.271
11(c)	-1/-1	-1/-1	-1/-1	0/0	0.000	0.358	0.198
11(d)	-1/-1	0/-1	-1/0	1/0	0.000	0.358	0.198

Fig. 12는 MATLAB GUI 기능을 이용하여 제안한 Binary 장애지형 판정 알고리즘을 구현한 것이다.

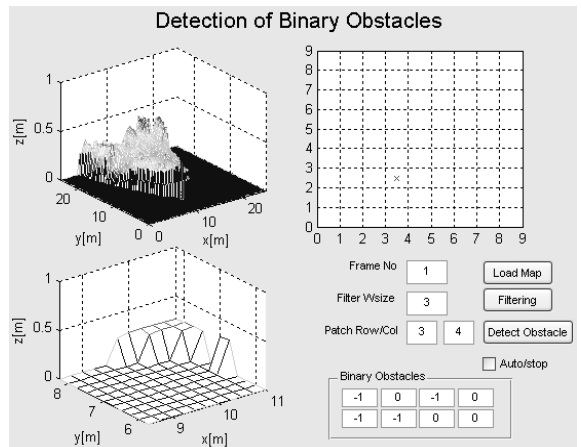


Fig. 12. MATLAB GUI를 이용하여 제안한 방법의 구현

1~5579 프레임의 지형 데이터 중 임의로 선택할 수 있고 선택된 지형은 좌측 상단에 그려진다. 필터 윈도우 크기를 조절하고 버튼을 누르면 필터링된 지형이 좌측 상단에 다시 그려진다. 패치의 행과 열 번호를 입력하고 버튼을 누르면 우측 상단의 패치맵에는 ‘○’ 또는 ‘×’가 그려지고 좌측 하단에는 b_1, b_2, b_3, b_4 장애가 출력된다.

패치맵의 ‘○’와 ‘×’는 각각 ‘주행가’, ‘주행불가’를 의미한다. 한편 하단의 첫째 행은 Pitch 방향, 둘째 행은 Roll방향을 의미한다.

6. 결론

본 연구에서는 저자들이 수행한 로봇의 주행성 분

석 및 주행속도 추정기법의 신뢰성을 높이기 위한 방법으로 월드모델을 기반으로 3가지 Binary 장애지형을 판정하는 측도를 제안하였다. 극복할 수 없는 장애지형의 판정에는 진행방향 인접 셀 간의 고도차를 측도로 사용하였고, 차체 접촉 장애지형 판정에는 바퀴와 지면의 기하학적 형태로부터 판단하였다. Roll over는 지형의 곡면을 평면 방정식으로 표현한 다음 최소자승법으로 방정식의 계수를 얻고 이로부터 판단하였다. 실제 월드모델을 이용하여 시뮬레이션을 수행한 결과 제안된 기법의 유효성을 확인할 수 있었다.

후 기

본 연구는 방위사업청/민군겸용기술센터와 정보통신부/정보통신연구진흥원의 민군겸용기술개발사업의 일환으로 수행하였음.

References

- [1] D. Langer, J. K. Rosenblatt and M. Hebert, "A Behavior-Based System for Off-Road Navigation", IEEE Trans. Robotics and Automation, Vol. 10, No. 6, pp. 776~783, 1994.
- [2] D. B. Gennery, "Traversability Analysis and Path Planning for a Planetary Rover," Autonomous Robots, Vol. 6, pp. 131~146, 1999.
- [3] L. Huajun, Y. Jingyu, Z. Chunxia, "A Generic Approach to Rugged Terrain Analysis Based on Fuzzy Inference", Proc. of the 8th Int. Conf. on Control, Automation, Robotics and Vision, pp. 1108~1113, 2004.
- [4] R. Hoffman and E. Krotkov, "Terrain Roughness Measurements from Elevation Maps", SPIE Mobile Robots, Vol. 1195, 1989.
- [5] H. Seraji, "Traversability Index : A New Concept for Planetary Rovers," Proc. of the 1999 IEEE Int. Conf. on Robotics and Automation, pp. 2006~2013, 1999.
- [6] N. Yokoya and K. Yamamoto, "Fractal-Based Analysis and Interpolation of 3D Natural Surfaces and Their Application to Terrain Modeling", Computer Vision, Graphics, and Image Processing, Vol. 46, pp. 284~302, 1989.
- [7] E. Tunstel, A. Howard and H. Seraji, "Rule Based Reasoning and Neural Network Perception for Safe Off Road Robot Mobility", Expert Systems, Vol. 19, No. 4, pp. 191~200, 2002.
- [8] I. Ulrich and J. Borenstein, "VFH+ : Reliable Obstacle Avoidance for Fast Mobile Robots", Proc. of the 1998 IEEE Int. Conf. on Robotics and Automation, pp. 1572~1577, 1998.
- [9] M. J. Procopio, T. Strohmman, A. R. Bates, G. Grudic and J. Mulligan, "Using Binary Classifiers to Augment Stereo Vision for Enhanced Autonomous Robot Navigation", University of Colorado at Boulder, Tech. Report, 2007.
- [10] T. Chang, T. Hong, S. Legowik and M. N. Abrams, "Concealment and Obstacle Detection for Autonomous Driving", Int. Association of Science and Tech. for Development 1999 Conf. on Robotics and Applications, Santa Barbara, CA, 1999.
- [11] C. S. Dima, N. Vandapel and M. Hebert, "Sensor and Classifier Fusion for Outdoor Obstacle Detection", The 32nd Applied Imagery Recognition Workshop(AIPR2003), pp. 255~262, 2003.
- [12] A. Rankin, A. Huertas and L. H. Matthies, "Nighttime Negative Obstacle Detection for Off-Road Autonomous Navigation," SPIE Defense & Security Symposium: Unmanned Systems Technology IX Conference, 2007.
- [13] 진강규, 이현식, 이윤형 et al., "지형 고도 맵으로부터 기울기와 거칠기 추출 방법", 제어·로봇·시스템학회 논문지, 제14권, 제9호, pp. 909~915, 2008.
- [14] G. Jin, Y. Lee, H. Lee, M. So, "Traversability Analysis for Navigation of Unmanned Robots", Proc. of 2008 Int. Conf. on Instrument, Control and Information Technology, Tokyo, 2008.