자율이동로봇의 안전주행을 위한 주행성 맵 작성

Development of a Traversability Map for Safe Navigation of Autonomous Mobile Robots

진 강 규*
(Gang-Gyoo Jin^{1,*})

¹Division of IT, Korea Maritime and Ocean University

Abstract: This paper presents a method for developing a TM (Traversability Map) from a DTM (Digital Terrain Model) collected by remote sensors of autonomous mobile robots. Such a map can be used to plan traversable paths and estimate navigation speed quantitatively in real time for robots capable of performing autonomous tasks over rough terrain environments. The proposed method consists of three parts: a DTM partition module which divides the DTM into equally spaced patches, a terrain information module which extracts the slope and roughness of the partitioned patches using the curve fitting and the fractal-based triangular prism method, and a traversability analysis module which assesses traversability incorporating with extracted terrain information and fuzzy inference to construct a TM. The potential of the proposed method is validated via simulation works over a set of fractal DTMs.

Keywords: autonomous mobile robot, digital terrain model, terrain information, traversability map

I. 서론

인간의 끊임없는 도전정신과 과학과 기술의 비약적인 발전에 힘입어 외부 환경에 대한 인식과 판단 기능을 갖춘 자율이동로봇에 대한 연구개발이 활발히 진행되고 있으며, 주요 응용분야로서는 행성 탐사, 재난 수색 및 구조, 군사용 감시, 정찰 및 지원, 환경관리, 농작물의 파종과 수확 작업 등이 있다[1-5].

이동로봇은 일반적으로 열악한 환경에서 운전되고 경우에 따라서는 조작자와의 통신이 어려운 환경에서도 임무를 성공적으로 완수해야 하므로 예상치 못한 상황에 대처하기 위해서는 자신의 능력에 더욱 의존하는, 즉 자율행동 결정 능력을 가져야 한다.

이동로봇이 자율적으로 동작하기 위해서는 환경감시(environment sensing), 위치인식(localization), 경로계획(path planning), 장애물 회피(obstacle avoidance) 및 경로추적(path tracking) 등의 기능을 가져야 한다[1,2]. 특히 야지(rough terrain)에서 동작될 경우에는 실시간적으로 안전한 경로계획과 경로추적을 위한 주행속도 계산이 필요하고 이를 위해서는 지형의 특성, 경로의 곡률, 로봇과 지형과의 상호작용 등 몇 가지 사항들이 제약 조건과 더불어 고려되어야 한다[4]. 그 중에서도 센서로부터 수집한 지형 데이터로부터 지형특성을 추출하고 이를 바탕으로 주행성을 분석하는 것이 어떤 다른 것보다도 우선되어야 할 사항이다.

주행성 분석은 크게 전역 주행성 분석과 지역 주행성 분석으로 나뉜다. 전역 주행성 분석은 위성, 레이더 등을 이

용하여 사전에 계획된 지역에 대한 광역 스케일의 저해상도 지형도를 얻고, 여기에서 산, 호수, 골짜기, 숲 등을 식별하고 로봇의 통과여부를 오프라인적으로 판단하는 것이고, 지역 주행성 분석은 스테레오 비전 카메라나 LiDAR, InSAR 등의 원격 센서로 반경 수 미터에서 수 십 미터 이내 지역의 고해상도 영상정보나 디지털 지형 모델(DTM: Digital Terrain Model)을 구축하고 해당 지형의 통과여부를 온라인적으로 판단하는 것이다[1,10]. 특히 지역 주행성 분석은 지형의 물리적, 기하학적 성질에 크게 좌우되므로 몇 가지 지형특성들이 연구되어 왔고, 그 중에서도 기울기와 거칠기가 주요 관심의 대상이 되어왔다.

기울기와 거칠기는 대개 영상정보나 DTM으로부터 추출된다. 영상정보를 기반으로 하는 연구로 Seraji [3]는 스테레오 비전으로부터 지형 기울기를 얻고 또 장애물의 크기와 밀집도도 추출하여 퍼지추론과 결합해 거칠기를 추정하였고, Williams와 Howard [6]는 극지방의 지형 기울기 추정에 싱글 카메라 영상을 사용하였고, Lee 등[7]은 스테레오 비전으로 도로의 기울기를 추정하고 이를 자유주행공간 검출에 사용하였다. 한편 DTM을 기반으로 하는 연구로 Hoffman과 Krotkov [8]는 Fourier 해석법으로 크기, 주파수와 자기상관(autocorrelation) 세 요소를 가지는 벡터 거칠기를 제안하였고, Huajun 등[9]과 Jin 등[10]은 기울기 추정에 최소자승법을 또 거칠기 추정에 프랙셔널 브라운 운동(fractional Brownian motion)을 이용하였다. 이들 모두는 나름대로 만족스러운 성과를 제시하고 있으나 개선의 여지가 있는 것은 사실이다.

본 논문에서는 DTM으로부터 자율이동로봇의 경로계획과 주행속도 결정에 기초자료가 되는 주행성 맵을 얻는 문제를 다룬다. 이를 위해 DTM은 다수의 패치로 분할되고,

* Corresponding Author

Manuscript received September 30, 2013 / revised January 13, 2014 / accepted January 20, 2014

진강규: 한국해양대학교 IT공학부(ggjin@kmou.ac.kr)

기울기와 거칠기는 특정 패치로부터 어느 한 방향으로 나아갈 때 지나게 되는 인접 패치들도 포함시켜 추출된다. 특히 기울기는 로봇의 진행방향과도 밀접한 관계를 가지고 있어 방향성을 고려하여 최소자승법으로 추정하고, 거칠기는 프랙탈 이론(fractal theory)에 기초한 삼각프리즘법으로 추정한다. 두 지형정보가 추정되면 이들을 입력으로 주행성 지수를 출력하는 퍼지시스템을 설계하고 이를 바탕으로 주행성 맵을 작성한다. 제안된 알고리즘의 유효성은 가상의 프랙탈 지형을 이용하여 검증된다.

II. 지형 정보의 추출

자율이동로봇은 열악한 환경에서 임무를 수행하게 되고, 임무 수행 중 센서로부터 획득한 3D 지형 데이터를 기반으로 안전하면서도 짧은 경로를 스스로 탐색하고 또 주행속도를 추정할 수 있어야 한다. 이를 위해서는 해당 지형의 특성을 추출하고 추출한 데이터를 기반으로 주행성을 판단할 수 있어야 한다.

본 연구에서 제안하는 주행성 맵의 작성은 크게 3 단계로 이루어지며, 그림 1은 이를 보여준다. DTM 분할(DTM partition)은 DTM을 같은 크기의 패치(patch)로 분할해주며, 지형정보 추출(terrain information extraction)은 기울기와 거칠기 정보를 추출해주며, 주행성 분석(traversability analysis)은 두 지형정보를 바탕으로 주행성 지수(traversability index)를 계산하고 최종적으로 주행성 맵(TM: Traversability Map)을 만들어 낸다.

1. DTM의 분할

매 샘플링 시각 센서로부터 획득되는 DTM의 X-Y 평면은 등간격으로 분할되어 $W \times W$ 셀(Cell)로 구성되며, 각 셀

그림 1. 주행성 맵의 작성 과정.

Fig. 1. Procedure of a traversability map development.

그림 2. 3D 지형 고도 모델.

Fig. 2. 3D terrain elevation model.

그림 3. 로봇의 진행 가능 방향.

Fig. 3. Possible traversing direction of the robot.

그림 4. 지형정보 추출에 사용되는 윈도우.

Fig. 4. Windows used for extracting terrain information.

은 그림 2와 같이 위치 정보 (x, y) 와 고도 정보 $z = f(x, y)$ 를 가지며, 로봇 위치는 그림 3의 하단 중앙 아래쪽에 있는 것으로 간주된다.

주행성 분석을 효과적으로 수행하도록 DTM은 같은 크기의 패치로 분할되므로(그림 3 참조), 패치는 $V \times V$ 셀로 구성된다. 이때 V 는 홀수 정수로서 로봇의 진행방향과는 상관없이 패치가 로봇을 완전히 덮을 수 있는 크기로 하며, $W = K \cdot V$ 인 것으로 간주된다. 만약 W 가 V 의 정수배가 아닌 경우에는 로봇으로부터 먼 거리에 있는 상단과 좌우 가장자리 쪽의 셀을 버린다. 이렇게 함으로써 DTM에는 K^2 개의 패치가 있게 되고, 패치에는 V^2 개의 셀이 내포된다.

그림 3은 특정 패치상에서 로봇이 진행할 수 있는 방향(3, 5, 8개)을 도식적으로 그린 것이다.

기울기와 거칠기는 패치단위로 추출되며 로봇이 특정 방향으로 전후진할 때 지나게 되는 인접 패치들도 포함시킨다. 따라서 진행방향에 따라 선택되는 패치 수는 2개 또는 4개이고, 그림 4는 이를 음영으로 표시한 것이다.

2. 지형의 기울기(terrain slope) 추출

기울기는 수평면에 대한 노면의 기울어진 정도로서 자율이동로봇의 안전 주행에 크게 영향을 미치는 요소이다. 기울기가 클수록 바퀴와 노면 간의 슬립이 커져 그 지역을 통과하기가 어렵고, 허용치를 넘게 되면 무게 중심이 불안정한 영역으로 이동해 전복(roll over)된다.

기울기는 스테레오 비전 카메라로부터 얻는 영상정보로부터 추출하거나[5-7] LiDAR (Light Detection and Ranging), InSAR (Interferometric Synthetic Aperture Radar) 등 원격 센서로 수집된 DTM으로부터 계산될 수 있으며[8-10] 상승지형인 경우에는 양, 하강지형의 경우에는 음의 값으로 표시

그림 5. 좌표 변환.

Fig. 5. Axis transformation.

된다.

본 논문에서는 그림 4에서 음영으로 표시된 윈도우 크기의 데이터에 최소자승법을 적용하여 로봇의 진행방향(pitch direction)과 측면방향(roll direction)의 기울기를 산출한다. 이때 진행방향이 그림 5의 Y'축 방향이라면 X-Y 좌표상의 위치 (x_i, y_j) 는 X'-Y' 좌표상의 (x_i', y_j') 로 변환되며 여기에 식 (1)이 사용된다.

$$\begin{bmatrix} x_i' \\ y_j' \end{bmatrix} = \begin{bmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{bmatrix} \begin{bmatrix} x_i \\ y_j \end{bmatrix} \quad (1)$$

그림 4에서 음영 지역들의 윈도우 크기를 $N \times M$, 내포된 모든 셀들의 인덱스 집합(index set)을 $I_{ij} = \{(i,j) \mid 1 \leq i \leq N, 1 \leq j \leq M\}$, I_{ij} 의 크기를 $Q = N \cdot M$ 라 하면 이 지역의 노면을 대표하는 수학적 모형은 다음과 같이 나타낼 수 있다.

$$z_{ij} = f(x_i, y_j) = a_0 + a_1 x_i + a_2 y_j + \varepsilon, \quad \forall (i,j) \in I_{ij} \quad (2)$$

여기서 a_0, a_1, a_2 는 계수이고, ε 는 각 셀마다 독립인 오차이다.

Q 개의 데이터 쌍 $\{(x_i, y_j), z_{ij}\} (1 \leq i \leq N, 1 \leq j \leq M)$ 을 식 (2)에 대입하고 이를 행렬-벡터로 표현하면 식 (3)과 같이 쓸 수 있다.

$$\mathbf{z} = \mathbf{A}\mathbf{x} + \boldsymbol{\varepsilon} \quad (3a)$$

단,

$$\mathbf{z} = \begin{bmatrix} z_{11} \\ \vdots \\ z_{ij} \\ \vdots \\ z_{NM} \end{bmatrix} \in \mathbb{R}^Q, \quad \mathbf{x} = \begin{bmatrix} a_0 \\ a_1 \\ a_2 \end{bmatrix} \in \mathbb{R}^3, \quad \boldsymbol{\varepsilon} = \begin{bmatrix} \varepsilon_1 \\ \varepsilon_2 \\ \vdots \\ \varepsilon_Q \end{bmatrix} \in \mathbb{R}^Q \quad (3b)$$

$$\mathbf{A} = \begin{bmatrix} 1 & x_1 & y_1 \\ \vdots & \vdots & \vdots \\ 1 & x_i & y_j \\ \vdots & \vdots & \vdots \\ 1 & x_N & y_M \end{bmatrix} \quad (3c)$$

오차제곱합이 최소가 되는 $\mathbf{x}$ 를 구하기 위해 평가함수를 식 (4)와 같이 정의하면

$$\mathbf{J} = \mathbf{e}^T \mathbf{e} = (\mathbf{z} - \mathbf{A}\mathbf{x})^T (\mathbf{z} - \mathbf{A}\mathbf{x}) \quad (4)$$

$\mathbf{J}$ 를 최소로 하는 추정치 $\hat{\mathbf{x}}$ 는 $\mathbf{A}^T \mathbf{A}$ 의 역행렬이 존재하는 경우에 다음 식으로 구해진다.

$$\hat{\mathbf{x}} = (\mathbf{A}^T \mathbf{A})^{-1} \mathbf{A}^T \mathbf{z} \quad (5)$$

이로부터 진행방향의 기울기 $\alpha_p = \hat{x}_2 = \hat{a}_2$ 와 측면방향의 기울기 $\alpha_R = \hat{x}_3 = \hat{a}_1$ 를 구할 수 있고, 로봇의 전복은 이들 값 중 큰 것에 의해 좌우되므로 기울기 지수(slope index) α 는 다음과 같이 정의된다.

$$\alpha = \min[\max(\frac{|\alpha_p|}{S_p}, \frac{|\alpha_R|}{S_R}), 1] \quad (6)$$

여기서 S_p, S_R 은 각각 진행방향과 측면방향의 최대 허용 기울기를 의미하고, α 는 다음 장에서 다룰 퍼지시스템의 입력신호로 사용되므로 편의상 0과 1사이의 값을 갖도록 하였다. 또 식 (6)에서 절대치를 취한 이유는 전복방향은 고려대상이 아니기 때문이다.

3. 지형의 거칠기(terrain roughness) 추출

거칠기는 노면의 지질학적 불규칙성을 나타내주는 정도로서 특히 자율이동로봇이 험한 야지를 주행할 경우에는 안전에 크게 영향을 주는 요인 중 하나가 된다. 일반적으로 노면이 거칠수록 해당 지역으로 주행하기가 더 어렵고, 이때 주행속도를 높이면 전복될 위험성이 커진다.

현재까지 문헌에 보고된 거칠기 추정 방법으로는 영상 데이터로부터 추출되는 장애물의 크기와 밀집도로 퍼지추론하는 법[5], Fourier 해석을 이용하는 법[8], 프랙탈을 이용하는 법[9,10] 등이 있다.

최근 프랙탈 차원으로 지형의 복잡성을 정량화하는 방법이 널리 사용되고 있고, 그 대표적인 방법으로는 Box-counting법, ε -blanket법, Spectral 해석법과 프랙탈 브라운 함수(fractal Brownian function) 접근법, 삼각프리즘법(TPM: Triangular Prism Method) 등이 있다. 이 중에서도 TPM의 가장 신뢰성이 높은 것으로 알려져 있다[12,13].

프랙탈 차원을 추정할 때 샘플링 전략은 DTM으로부터 어떤 스텝크기(step size)를 어떻게 취할 것인지를 정하는 방법으로서 성능에 직접적인 영향을 주며 현재까지 널리 사용되고 있는 방법으로는 기하학적 스텝(geometric step)법, 제수 스텝(divisor step)법과 산술적 스텝(arithmetic step)법이 있다.

Clarke의 기하학적 스텝법[11]은 스텝크기 δ 로 1과 $(N-1)/2$ 사이의 2의 거듭제곱을 취하며 그 최대값을 기준으로 나머지 모든 δ 에 대해 고정 커버리지(fixed coverage)를 적용하므로 인해 셀 활용률이 낮은 것이 단점이다. 제수 스텝[12]법은 $N-1$ 의 제수($N-1$ 은 제외)를 δ 로 취함으로써 100% 셀 활용률이 보장되지만, 특정 N 에서는 회귀분석 데이터가 적은 것이 단점이다. 한편 산술적 스텝법[13]은 1과 $(N-1)/2$ 사이의 정수를 δ 로 취함으로써 회귀분석 데이터를 충분히 확보할 수 있지만, δ 가 클 때 연산부담이 증가하고 셀 활용률이 낮은 것이 단점이다. 이를 개선하고자 저자는 전체 셀 활용률을 높이면서 스텝수를 줄여 연산시간을 줄이고

또 임의 크기의 윈도우를 사용할 수 있는 수정 산술적 스텝(MAS: Modified Arithmetic Step)법을 제안하였다[14].

거칠기 추정법으로 본 논문에서는 MAS 샘플링 전략을 채용하는 TPM을 이용한다. 윈도우 크기가 $N \times M$ 이고 $L = \min(N, M)$ 이면 MAS법은 1과 $(L-1)/2$ 사이의 정수 중 셀 활용률, 즉 식 (7)의 CR (Coverage Ratio)이 문턱 값(threshold) 이상인 것만을 δ 로 취하게 된다. 이렇게 함으로써 스텝크기 집합은 $\Delta = \{\delta | 1 \leq \delta \leq (L-1)/2 \text{ and } CR \geq \text{threshold}\}$ 가 되고 $n = \dim(\Delta)$ 이 된다.

$$CR = \frac{[\text{int}(\frac{N-1}{\delta})\delta + 1][\text{int}(\frac{M-1}{\delta})\delta + 1]}{N \times M} \times 100\% \quad (7)$$

여기서 $\text{int}(\cdot)$ 는 정수 부분을 취하는 연산자이다.

$\delta \in \Delta$ 마다 4개의 삼각프리즘 상단의 표면적 합 S_δ 가 계산되므로 결국 n 개의 데이터 쌍 (δ_i, S_i) 를 얻게 되고, 이를 대표하는 log-log 좌표상의 1차 회귀식은 다음과 같이 쓸 수 있다.

$$\log(S) = a + H \cdot \log(\delta) + \epsilon \quad (8)$$

여기서 H 는 상수로서 직선 기울기, a 는 절편, ϵ 는 오차를 의미한다. n 개의 데이터 쌍을 가지고 식 (8)은 다음과 같이 다시 쓸 수 있다.

$$\mathbf{z} = \mathbf{A}\mathbf{x} + \boldsymbol{\epsilon} \quad (9a)$$

단,

$$\mathbf{z} = \begin{bmatrix} \log(S_1) \\ \log(S_2) \\ \vdots \\ \log(S_n) \end{bmatrix} \in \mathbb{R}^n, \quad \mathbf{x} = \begin{bmatrix} a \\ H \end{bmatrix} \in \mathbb{R}^2, \quad \boldsymbol{\epsilon} = \begin{bmatrix} \epsilon_1 \\ \epsilon_2 \\ \vdots \\ \epsilon_n \end{bmatrix} \in \mathbb{R}^n \quad (9b)$$

$$\mathbf{A} = \begin{bmatrix} 1 & \log(\delta_1) \\ 1 & \log(\delta_2) \\ \vdots & \vdots \\ 1 & \log(\delta_n) \end{bmatrix} \quad (9c)$$

여기에 최소자승법을 적용하면 식 (5)와 같은 추정치 $\hat{\mathbf{x}}$ 를 얻고 $H = \hat{x}_2$ 로부터 프랙탈 차원 D 는 다음 식과 같이 계산된다.

$$D = 2 - H \quad (9)$$

다음은 MAS 샘플링 전략을 채용하는 TPM의 전체 알고리즘을 보여준다[14].

Procedure of TPM with the MAS strategy

Step 1 get $N \times M$ data;

Step 2 calculate $L = \min(N, M)$ and Δ with the MAS strategy;

for $i = 1$ to $\dim(\Delta)$

Step 3 let $\delta = \Delta(i)$ and $S(i) = 0$;

Step 4 find covering area(b_{row} , e_{row} , b_{col} , e_{col}) with N , M and δ ;

for $\text{row} = b_{\text{row}}$ to $e_{\text{row}} - \delta$ step δ

for $\text{col} = b_{\text{col}}$ to $e_{\text{col}} - \delta$ step δ

Step 5 get four corner elevations h_a , h_b , h_c and h_d of the selected square;

Step 6 calculate $h_e = (h_a + h_b + h_c + h_d)/4$;

Step 7 calculate the top surface area of the four prisms A with h_a , h_b , h_c , h_d , h_e and δ ;

Step 8 let $S(i) = S(i) + A$;

end

end

end

Step 9 apply a data fitting method to derive H and D ;

앞서 계산된 D 는 2와 3 사이의 값을 가지므로 퍼지시스템의 입력신호로 사용하도록 편의상 0과 1 사이의 값으로 사상한다. 따라서 지형의 거칠기 지수(roughness index)는 다음 식으로 정의된다.

$$\beta = \max[D - 2, 0] \quad (10)$$

지표면이 거칠수록 β 값이 더 커지고, 반대로 평탄할수록 더 작아지며, 평면인 경우에는 $\beta = 0$ 가 된다.

III. 주행성 분석

여기서는 앞 장에서 추출한 두 가지 지형정보와 퍼지기법으로 주행성 지수를 얻고 이로부터 TM을 작성하는 문제를 다룬다.

1. 입력공간의 퍼지분할

자율이동로봇은 외부환경의 불확실성과 센서의 부정확성이 내재된 정보로부터 올바른 판단을 유도해 내야 하므로, 이런 형태의 문제 해결 방식으로는 퍼지기법이 효과적이므로 주행성 분석을 위해 퍼지이론을 사용한다.

퍼지시스템은 2개의 입력신호를 받아 추론을 통해 1개의 출력신호를 만들어 내도록 설계되며, 기울기 지수 α 와 거칠기 지수 β 가 입력, 주행성 지수 γ 가 출력이 된다. 지식베이스를 구축하고 이를 추론하기 위해서는 우선 입력공간을 퍼지분할하고 이에 따른 각 퍼지집합의 소속함수를 정의할 필요가 있다. α , β 는 각각 4개의 퍼지집합 {FLAT, SLOPED, STEEP, VERY STEEP}와 {SMOOTH, MODERATE, ROUGH, BUMPY}로 분할되고, 바깥쪽의 소속함수는 시그모이드형, 안쪽은 가우스형을 사용하였다. 그림 6은 퍼지분

그림 6. 입력변수들의 퍼지분할.

Fig. 6. Fuzzy partitions of the input variables.

표 1. 주행성 분석용 16 퍼지규칙.

Table 1. 16 fuzzy rules for traversability analysis.

$x_1 \backslash x_2$	SMOOTH	MODERATE	ROUGH	BUMPY
FLAT	T_5	T_4	T_3	T_2
SLOPED	T_4	T_3	T_2	T_1
STEEP	T_3	T_2	T_1	T_1
VERY STEEP	T_2	T_1	T_1	T_1

할된 공간과 소속함수들의 모양을 보여준다.

2. 주행성 지수의 결정

규칙베이스에 축적된 지식은 퍼지규칙으로 불리며, 통상 ‘if-then’ 조건문으로 기술된다. 주행성 분석용 규칙베이스는 다음과 같은 Takagi-Sugeno 퍼지모델로 나타낼 수 있다.

$$\text{Rule } i: \text{if } x_1 \text{ is } A_{1i} \text{ and } x_2 \text{ is } A_{2i}, \\ \text{then } \gamma = Q_i \quad (1 \leq i \leq \ell) \quad (11)$$

여기서 $x_1 (= \alpha)$ 와 $x_2 (= \beta)$ 는 언어변수, A_{1i} 와 A_{2i} 는 각각 i 번째 퍼지규칙에서 전건부 변수들의 퍼지값, ℓ 은 퍼지규칙의 개수이고 A_{1i} 는 {FLAT, SLOPED, STEEP, VERY STEEP} 중 하나, A_{2i} 는 {SMOOTH, MODERATE, ROUGH, BUMPY} 중 하나가 된다. 한편 Q_i 는 스칼라 값으로서 $\{T_1, T_2, T_3, T_4, T_5\}$ 중 하나가 되며 T_1, T_2, T_3, T_4, T_5 는 각각 0, 0.25, 0.5, 0.75, 1의 값을 갖는다.

퍼지규칙은 전문가의 지식과 경험을 기반으로 작성되었고 표 1은 $\ell = 16$ 개의 규칙을 나타낸 것이다.

따라서 입력 쌍 (x_1, x_2) 에 대해 식 (11)의 추론 결과는 다음 식과 같다.

$$\gamma = \frac{\sum_{i=1}^{\ell} \zeta_i Q_i}{\sum_{i=1}^{\ell} \zeta_i} \quad (12a)$$

단,

$$\zeta_i = \prod_{j=1}^2 A_{ji}(x_j) \quad (12b)$$

여기서 $A_{ji}(x_j)$ 는 변수 x_j 의 퍼지집합 A_{ji} 에 대한 소속정도이다. 그림 6에서 볼 수 있듯이 인접한 퍼지집합 간에 서로 포갠(overlap)이 일어나도록 퍼지분할 함으로써 항상

$\sum_{i=1}^{\ell} \zeta_i > 0$ 의 관계가 성립되어 식 (12a)는 유효하다.

그림 7은 두 입력 α, β 를 0과 1사이에서 변경할 때 퍼지추론 결과를 3D 공간상에 그린 것이다.

그림에서 γ 값이 클수록 해당 지역으로의 진행이 더 순조롭고 $\gamma = 0$ 은 ‘통과불가(non-traversable)’, $\gamma = 1$ 은 ‘완전 통과(completely traversable)’를 의미한다.

TM은 프로그래밍 관점에서 0과 1사이의 γ 값이 저장되어 있는 기억장소이고, 특정 패치로부터 나아갈 수 있는 방향이 최고 8개이므로 총 $8K^2$ 개의 기억공간이 필요하다. 매 샘플링 시각 새로운 DTM이 얻어지고 γ 가 계산되면 이를 기반으로 TM이 작성된다. 특히 가장자리나 모서리 패치의 진행할 수 없는 방향과 대응되는 기억공간에는 0이 저장된다.

그림 7. 퍼지시스템의 입출력 관계.

Fig. 7. I/O relationship of the fuzzy system.

IV. 시뮬레이션 및 결과 검토

여기서는 프랙탈 DTM을 사용하여 시뮬레이션을 실시하고 그 결과를 검토한다. 프랙탈 DTM은 midpoint displacement법[15]으로 만들어지고, MAS 문턱 값은 80%로 하였다.

1. 프랙탈 패치를 이용한 시뮬레이션

먼저 그림 4와 같은 지형정보 추출 환경을 고려해서 서로 다른 3개의 프랙탈 DTM으로부터 각각 37×19 , 19×37 , 37×37 윈도우를 임의로 취하고 그린 것이 그림 8의 좌측 그림이다. 셀 크기가 $20\text{cm} \times 20\text{cm}$ 이면 이들은 각각 $7.4\text{m} \times 3.8\text{m}$, $3.8\text{m} \times 7.4\text{m}$, $7.4\text{m} \times 7.4\text{m}$ 를 커버한다. 고도는 로봇에 장착된 센서를 기준으로 아래쪽을 음으로 하였다. 각각의 고도 정보는 표 2에 요약해 놓았다. 그림에서 보면 (a)는 (b)

그림 8. 패치 지형과 프랙탈 플롯.

Fig. 8. Patch regions and fractal plots.

표 2. 추정된 지형정보와 주행성 지수.

Table 2. Estimated terrain information and traversability index.

Fig.	Window	Elevation(m)			D	α	β	γ
		z_{min}	z_{max}	z_{ave}				
8(a)	37×19	-1.31	-0.42	-0.90	2.42	0.41	0.42	0.50
8(b)	19×37	-1.44	-0.89	-1.17	2.70	0.02	0.70	0.44
8(c)	37×37	-1.30	-0.85	-1.04	2.32	0.06	0.32	0.77

와 (c)에 비해 상대적으로 급경사 지형이고, (b)는 (a)와 (c)에 비해 기복이 심한 지형임을 직관적으로 느낄 수 있다.

그림 8의 우측 그림은 TPM의 프랙탈 플롯(fractal plot)을 그린 것이고, 표 2는 로봇이 Y방향으로 진행할 때 패치 지형들의 지형정보와 주행성 지수를 구해놓은 것이다. 이때 로봇의 진행방향과 측면방향의 최대 허용 기울기를 각각 $S_F = 0.70(35^\circ)$, $S_R = 0.47(25^\circ)$ 로 하였다. 표 2에서 보면 그림 8(a)의 기울기 지수 α 가 그림 8(b)와 8(c)보다 더 크고 또 그림 8(b)의 거칠기 지수 β 가 그림 8(a), 8(c)보다 크게 나왔는데 이 결과는 직관과 일치하고 있는 것을 알 수 있다. 결과적으로 γ 값이 제일 큰 8(c)가 Y 방향으로 가장 주행하기 좋은 지형임을 알 수 있다.

2. 프랙탈 DTM을 이용한 시뮬레이션

다음은 95×95 셀로 구성된 DTM상에서 시뮬레이션을 실시하였고, 그림 9는 이를 그린 것이다. 전과 동일한 셀 크기를 적용하면 DTM은 19m×19m 지역을 커버하고, 최저 고도(z_{min}), 최고 고도(z_{max}), 평균 고도(z_{ave})는 각각 -1.50m, -0.53m, -1.01m이다. DTM을 5×5 패치로 분할하였고, 패치 하나는 19×19 셀이고 크기는 3.8m×3.8m이다.

5×5로 분할된 패치지형에 앞에서 제안한 방법을 적용하고 편의상 TM을 비트맵으로 표시하였다. TM에는 패치당 3×3의 사각 공간이 할당되고 진행방향에 따른 계산된 γ 의 비트맵 표시 위치는 그림 10의 좌측 그림과 같고, 중앙의 빈 공간(사용하지 않음)은 비트맵에서는 흰색으로 표시하였다.

계산된 γ 는 0.41~0.75 사이의 값들이었고, 그림 10의 우측 그림은 이를 흰색과 검정색 사이 5개 스케일의 비트맵으로 그린 것이다. TM의 비트맵은 빈 공간을 제외하고 흰색에 가까울수록 γ 값이 작아 그 지역과 방향으로는 주행이 원활하고, 검정색에 가까울수록 γ 값이 커서 주행이 어렵다는 것을 시각적으로 보여준다.

그림 9. 프랙탈 DTM.

Fig. 9. Fractal DTM.

그림 10. 프랙탈 DTM으로부터 작성한 TM.

Fig. 10. TM developed from a fractal DTM.

V. 결론

본 연구에서는 자율이동로봇에 장착된 센서로부터 입수한 DTM으로부터 경로계획과 주행속도 결정의 기초자료가 되는 TM을 얻는 문제를 다루었다. DTM은 다수의 패치로 분할되고, 분할된 패치상에서 로봇의 진행방향을 고려하여 기울기와 거칠기가 추출되었다. 기울기는 최소자승법으로, 거칠기는 TPM으로 추정하였다. 추정된 지형정보를 바탕으로 주행성 지수를 계산해주는 퍼지시스템을 설계하였고 최종적으로 TM을 작성하였다.

프랙탈 DTM상에서 시뮬레이션을 수행한 결과 제안된 기법의 유효성을 확인할 수 있었다.

REFERENCES

- [1] S. Singh, R. Simmons, T. Smith, A. Stentz, V. Verma, A. Yahja, and K. Schwehr, "Recent progress in local and global traversability for planetary rovers," *Proc. of the 2000 IEEE International Conference on Robotics & Automation*, CA, pp. 1194-1200, 2000.
- [2] D. Langer, J. K. Rosenblatt, and M. Hebert, "A behavior-based system for off-road navigation," *IEEE Transactions on Robotics and Automation*, vol. 10, no. 6, pp. 776-783, 1994.
- [3] S. Yaghoubi, N. A. Akbarzadeh, S. S. Bazargani, S. S. Bazargani, M. Bamizan, and M. I. Asl, "Autonomous robots for agricultural tasks and farm assignment and future trends in agro robots," *International Journal of Mechanical & Mechatronics Engineering*, vol. 13, no. 3, pp. 1-6, 2013.
- [4] S. H. Joo and J. H. Lee, "A high-speed autonomous navigation based on real time traversability for 6x6 skid vehicle," *Journal of Institute of Control, Robotics and Systems (in Korean)*, vol. 18, no. 3, pp. 251-257, 2012.
- [5] H. Seraji, "Traversability index: a new concept for planetary rovers," *Proc. of the 1999 IEEE International Conference on Robotics and Automation*, pp. 2006-2013, 1999.
- [6] S. Williams and A. M. Howard, "A single camera terrain slope estimation technique for natural arctic

- environments,” *Proc. of the 2008 IEEE International Conference on Robotics and Automation*, CA, pp. 19-23, 2008.
- [7] K. Y. Lee and J. W. Lee, “Stereo-vision based road slope estimation and free space detection on road,” *Journal of Institute of Control, Robotics and Systems (in Korean)*, vol. 17, no. 3, pp. 119-206, 2011.
- [8] R. Hoffman and E. Krotkov, “Terrain roughness measurements from elevation maps,” *SPIE Mobile Robots*, vol. 1195, 1989.
- [9] L. Huajun, Y. Jingyu, and Z. Chunxia, “A generic approach to rugged terrain analysis based on fuzzy inference,” *Proc. of the 8th International Conference on Control, Automation, Robotics and Vision*, pp. 1108-1113, 2004.
- [10] G. Jin et al., “Slope and roughness extraction method from terrain elevation maps,” *Journal of Institute of Control, Robotics and Systems (in Korean)*, vol. 14, no. 9, pp. 909-915, 2008.
- [11] C. Clarke, “Computation of the fractal dimension of topographic surfaces using the triangular prism surface area method,” *Computers & Geosciences*, vol. 12, no. 5, pp. 713-722, 1986.
- [12] W. Ju and N. S.-N. Lam, “An improved algorithm for computing local fractal dimension using the triangular prism method,” *Computers & Geosciences*, vol. 35, no. 6, pp. 1224-1233, 2009.
- [13] C. W. Emerson, N. S.-N. Lam, and D. A. Quattrochi, “A comparison of local variance, fractal dimension, and Moran's I as aids to multispectral image classification,” *International Journal of Remote Sensing*, vol. 26, no. 8, pp. 1575-1588, 2005.
- [14] G. Jin, “Triangular Prism method based on an enhanced sampling method,” *Journal of Korean Institute of Intelligent Systems*, vol. 23, no. 2, pp. 93-99, 2013.
- [15] D. Saupe, *Algorithms for Random Fractals, The Science of Fractal Images*, In H.-O. Peitgen and D. Saupe, Editors, Springer-Verlag, 1988.

진강규

1977년 한국해양대학교 기관학과 졸업.
1985년 (미)FIT 전기·전자·컴퓨터공학과 석사. 1996년 (영)웨일즈대 전기·전자·시스템공학과 박사. 1996년~현재 한국해양대학교 IT공학부 교수. 관심분야는 지능제어, 유전알고리즘을 이용한 최적화.