

不整地生成 AI を用いたロボットの未知環境適応のための 実験フレームワークに関する研究

○大野 裕之（豊橋技術科学大学） 垣内 洋平（豊橋技術科学大学）

While robots are expected to be used in unknown environments, it is difficult for robots to adapt to such environments and successfully explore them. One way to solve this problem is to use machine learning to plan a run, but there is a limit to the amount of training terrain data that can be prepared for this purpose. In this paper, we propose an experimental framework that uses GAN as generative AI to generate unknown rough terrain to secure a huge amount of learning data and improve the robot's running performance.

1. はじめに

1.1 研究の背景と目的

現在、災害現場や月面などの未知環境におけるロボットの活用が、人の活動領域の拡大やその際に起こりうる人の身の危険性を回避するという点で期待されている。しかし、それら未知の環境に対してロボットが自動操作のみで探査を行うのは困難である。[1] また、それらの環境に対するシミュレーション用に地形データを収集するのにも限界がある。[2] そこで本研究では、ロボットが走行する未知の不整地環境を効率的に生成する方法と、その地形を評価するための走行実験フレームワークを提案する。

1.2 本論文の構成

本論文の構成について説明する。第 2 章ではロボットの走行実験に必要な地形データを効率的に収集する手法について述べる。第 3 章では収集した地形を用いたロボットの走行実験の方法について述べる。第 4 章ではまとめと今後の課題について述べる。

2. 地形データの収集方法

2.1 地形生成の手法

本研究では、効率的に未知環境の地形を収集する手法として、生成 AI を用いた地形のメッシュデータを生成する方法を提案する。生成 AI のモデルには GAN を三次元のボクセルベースに応用した 3D-IWGAN を用いた。[3]

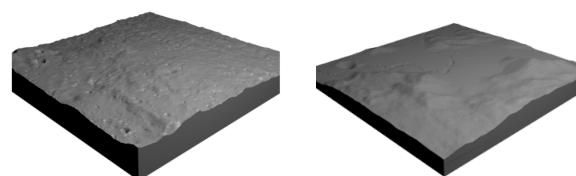
2.2 詳細手順

最初に GAN の学習用データとして NASA の提供している月面のメッシュデータを用意した。25 種類のデータがあるが、それを表面に沿って 4×4 に分割することで学習データ拡張を行った。拡張後の各データに対し、地形表面のメッシュ三角形から頂点座標を抽出し、それを元にボクセル化を行う。このボクセルの三次元座標データを GAN の入力として学習させ、新たな三次元座標データを生成させる。次にこの座標をスプライン曲線で補間し、得られた点を頂点とした三角形メッシュで埋めることで最終的な不整地データを作成する。

2.2.1 NASA の月面データについて

上記で説明した通り、NASA の提供している月面データを利用してデータセットにする。アポロ 11 号等の宇宙船が着陸した地点から $30 \text{ km} \times 30 \text{ km}$ の範囲の高度マ

ップになっている。これらは 3D プリンティング等のために提供されており、ファイルは STL 形式のメッシュデータになっている。以下の図 1 に使用したデータの例を示す。



月面の例 1

月面の例 2

図 1 使用した月面の例

2.2.2 ボクセル化について

本研究では、三次元分布の可視化をする際にその特徴をボクセルによって表現することとした。このボクセルの解像度によって地表面の起伏度合いを調整できることと、その起伏度合いが視覚的に伝わりやすいためである。ここでは月面データをボクセル化する方法について具体的に述べていく。まずはボクセルの解像度を設定する。ここでは仮にボクセルの解像度を $256 \times 256 \times 64$ とする。地表のメッシュ三角形の頂点座標を取得し [図??]、 256×256 の各格子内にある頂点の平均値をその代表点とする。[図??] それらの点の最大値が高さ方向の解像度の 64 マス目となる。このようにして 0 マス目からその格子の点の高さまでを点で埋めることで、0 と 1 で表現された $256 \times 256 \times 64$ の座標データが出来上がる。これに従って $1 \text{ m} \times 1 \text{ m} \times 1 \text{ m}$ の矩形を描くことでボクセルを表現する。

2.2.3 GAN を用いた地形データ生成について

三次元ボクセルの高さ方向の分散により、不整地の起伏度合いを表現した。使用した GAN モデルの構造を以下の図??に示す。データの入力方法は、正規分布に従った乱数の平均と標準偏差の値を入力する方法、画像をエンコードした後の平均と標準偏差の値を入力する方法、深度カメラ等を用いて取得した画像から作成した三次元のボクセル座標をエンコードした後、その平均と標準偏差の値を入力する方法の 3 種類となっている。月面データを $32 \times 32 \times 32$ に分割したボクセルにし、その三次元座標を GAN の入力として学習を行った出力結果が図??である。

2.2.4 メッシュ化について

ボクセル化した地形を元の月面のようになだらかな地表に変換するため、本研究ではボクセルの点からスプライン補間 [3] によって求めた曲線に基づいてメッシュデータに変換する方法を用いた。スプライン補間はある複数の点を全て通る曲線の多項式を求める方法である。仮に xy 平面の三次のスプライン補間を行う場合、以下のような多項式になる。ここで S は曲線が通る各点の y 座標、 n は曲線が通る点の数、そして最終的に求めるのは各項の係数となる。

$$S_0(x) = a_0 + b_0(x - x_0) + c_0(x - x_0)^2 + d_0(x - x_0)^3$$

$$S_1(x) = a_1 + b_1(x - x_1) + c_1(x - x_1)^2 + d_1(x - x_1)^3$$

$$S_2(x) = a_2 + b_2(x - x_2) + c_2(x - x_2)^2 + d_2(x - x_2)^3$$

⋮

$$S_j(x) = a_j + b_j(x - x_j) + c_j(x - x_j)^2 + d_j(x - x_j)^3$$

⋮

$$S_{n-1}(x) = a_{n-1} + b_{n-1}(x - x_{n-1}) + c_{n-1}(x - x_{n-1})^2 + d_{n-1}(x - x_{n-1})^3$$

生成した地形では次数を 10 と設定したため、多項式は以下ようになり、求める係数は $a \sim k$ となる。

$$S_{n-1}(x) = a_{n-1} + b_{n-1}(x - x_{n-1}) + c_{n-1}(x - x_{n-1})^2 + d_{n-1}(x - x_{n-1})^3 \dots + k_{n-1}(x - x_{n-1})^{10}$$

これには Python の数値解析ライブラリである SciPy から `scipy.interpolate.bisplrep` という関数を呼び出すことで自動で計算を行った。この曲線にそって三角形メッシュの頂点を決定してメッシュ化を行う。以下の図??は生成した月面のボクセルデータとそれを元にメッシュ化した月面データの例である。高さの離れた点が隣接していると、スプライン曲線の特性上例の 4 のように極端に鋭利な地形領域ができてしまう場合があった。

2.3 ロボットの構成

使用するロボットには、外界のセンシング機能と不整地に対する一定の走破性能を持ったハードウェアがそれぞれ要求される。そこで使用するロボットには、図??のような径の大きな 2 つのタイヤと 2 つのキャスタ、深度カメラを 4 方向に搭載した。このロボットは、ROS 及び Gazebo シミュレーターの管理団体である Open Robotics とロボットメーカー ROBOTIS(ロボティクス)が共同開発した Turtlebot3[4] の Waffle モデルをもとにしており、各寸法は図??のようになっている。タイヤ径はこの 3.5 倍に設定しており、4 方向の深度カメラは水平方向に対し 45 度下に傾けた状態にしている。深度カメラの画像サイズは 320×320 とした。

2.4 走行計画手法

走行計画として、4 つの深度カメラから得られた画像から走行可能な方位を選択する方法を用いた。事前に生成した地形を用いて走行実験を行い、ある方位に走行可能だった場合は safe(図??) として 0、転倒や脱輪により探索続行不可になった場合はその直前の画像に danger(図??) として 1 をラベリングする。そのデータに対し出力が 0 から 1 になるよう機械学習し、走行中に推論した結果最も 0 に近い値を出力した方位に向かうようにした。例えば図??のような場合には、ロボットは前方向に

進む。

2.5 実験方法

予め生成した複数の不整地上を手動操作でロボットに走行させ、機械学習用データを収集する。画像数は計 10000 枚ほどであり、それぞれに safe と danger をラベリングした。学習モデルには CNN(Convolutional Neural Network) を使用した。上記のデータを用いて学習を行い、再度生成した不整地上をロボットに走行させながらその軌跡を保存する。走行するマップ範囲内での軌跡の含有率をロボットの走破性と定義した。[2] シミュレーションにはロボット用統合 GUI ソフトウェアである choreonoid を用いた。このシミュレーション環境上に月面と $6\text{ m} \times 6\text{ m}$ の壁を周囲に配置し、それをマップの範囲とした。[図??]

2.6 生成地形の評価

本研究では以下の 2 つの点からロボットを走行させることで地形データの評価をすることとした。

- (1) 分散や標準偏差といった統計量が極端に離れている地形を生成した場合、特徴量の範囲が広い汎用的な地形データを収集できるが、学習用データとして用いて走行させた際に実際の地形に対する走破性向上に繋がるかは未知数となる。
- (2) 仮に統計的に近い地形データを生成できたとしても、ロボットの座標や地表面のある隣接した領域どうしの関係性によって走破性は左右する。

学習済みモデルを用いて、生成した不整地上と元の月面上をそれぞれ 2 分間走行させた際の軌跡に対し、マップの走破性を比較することで生成地形の有用性を評価した。使用した元の月面と生成した月面、走行した結果はそれぞれ下図のようになった。ここで、各グラフの中心がロボットの開始位置で、軌跡の止まった点がロボットの走行終了地点である。生成した地形の高さ方向の分散が起伏度合いに影響を与えるため、生成した月面 2 のような分散の小さいものを用いると極端に走破性が上がってしまった。逆に起伏が過度に激しいと、走破性が極端に下がり、かつロボットを配置したスタート地点に依存した走破性になってしまう。

3. おわりに

3.1 まとめ

本研究では 3D-IWGAN を用いて不整地を生成することで、走行計画に必要な学習用データを効率的かつ大規模に用意し、その走破性評価による実験フレームワークの作成を行った。3 章で述べたとおり、生成した地形が走行計画の学習用データとしての有用性があるかは、統計的な特徴量だけで判断するのは難しく、実際にロボットを走行させて評価する必要があることが分かった。ここでは、マップ内をロボットに走行させた軌跡の含有率が近いデータが元の地形データと類似していると判断し、更にその有用性を評価できた。また、本研究での実験フレームワークによって元の月面データの走破結果に近い生成地形を用いることで、少数の地形データで走行計画に必要な機械学習用データを効率的に収集できることが分かった。



図 2 サンプル画像

3.2 今後の課題

シミュレーション環境の実験フレームワークだけでなく、現実環境で実機を用いた検証を行えるような方法を検討する。また、本研究では月面を未知環境の例として取り扱ったが、災害現場等についても同様な実験が行えるか検討する。

参 考 文 献

- [1] “Jaxa 解決すべき技術課題”, <https://www.ihub-tansa.jaxa.jp/introduction/themes.html>
- [2] 西村祐輝: “強化学習を用いた自律多脚車輪型ロボットの脱出行動の環境適応. 2016.”,
- [3] David Meger Edward J. Smith: “Improved adversarial systems for 3d object generation and reconstruction. 2017.”,
- [4] “スプライン補間について”, <https://qiita.com/khidaka/items/84610cd890ecb8443d96>
- [5] “Turtlebot について”, <https://www.besttechnology.co.jp/modules/onlineshop/index.php?fct=photop=191>
- [6] 高橋 友一清水 優: “モデル化した不整地による評価フィールドを用いたレスキューロボットの動作・地図作成能力評価手法の提案. 2014.”,