Vol.2021-CG-181 No.1
2021/2/16

単視点地形景観画像からの 3D 地形モデルの 2 段階推定

高橋 遼[1,a)]　遠藤 結城[1,b)]　金森 由博[1,c)]　三谷 純[1,d)]

概要：
1 枚の景観画像から 3D 地形モデルを復元できれば、その景観を気軽に 3D で鑑賞できる。しかし既存の単視点深度推定手法では、入力画像中の可視領域の深度しか推定できず、復元形状に欠損が生じてしまう。そこで本研究では、1 枚の地形景観画像から、入力画像中の非可視領域も含めた 3D 地形モデルを推定する、CNN による教師あり学習手法を提案する。3D 地形モデルはテクスチャ付き高さマップで表現する。本研究では、入力画像中で推定しやすい可視領域と、推定しづらい非可視領域を分けて扱うため、2 段階の推定を行う。まず、入力画像の 1) 深度と 2) 影や光源の影響がない色情報を CNN で推定し、その結果から三角形メッシュを計算する。そして、三角形メッシュを真上から平行投影して欠損した高さマップとテクスチャを得る。最後に、高さマップとテクスチャの欠損を別の CNN で補完し、3D 地形モデルを得る。以上により、入力画像の遮蔽領域も含めて 3D 地形モデルを推定できる。

Two-step Estimation of 3D Terrain Model from a Single Landscape Image

1. はじめに

3D 地形モデルは映像制作やゲームなどのアプリケーションで幅広く利用されている。もし 1 枚の画像から 3D 地形モデルを復元できれば、特に AR や VR での応用が考えられ、例えば、観光地で撮影した写真を 3D 表現で再度楽しめるといった利用が挙げられる。しかし、既存の単眼深度推定手法では、入力画像の可視領域の深度しか推定できず、復元する 3D 地形モデルに欠損が生じてしまう。

そこで本研究では、1 枚の地形景観画像から 3D 地形モデルを推定する畳み込みニューラルネットワーク (CNN) による教師あり学習手法を提案する。3D 地形モデルはテクスチャ付き高さマップで表現する。

本研究で解決するべき主な問題は 2 つある。1 つ目は、欠損の無い 3D 地形モデルを推定するにあたって、単一の入力画像から見えない領域は推定する 3 次元情報に欠損が生じる点である。2 つ目は、入力画像の色情報をそのまま用いてテクスチャを推定すると、光源に依らず常に影のついた不自然なテクスチャになってしまう点である。

以上の問題に対処するために、本研究では、入力画像の可視領域から推定した 3 次元情報をもとに、非可視領域の欠損した情報を補完する 2 段階での推定を行う。また、入力画像から影や光源の影響を除去した色を推定し、不自然な推定テクスチャが生じないようにする。さらに、入力画像から見えない地形領域には、形状と色に様々なバリエーションが考えられるため、変分オートエンコーダ (VAE) を用いて欠損部分の多様な補完結果を出力できるようにする。訓練に用いる地形景観画像データセットは、我々の知る限りでは存在しないため、CG ソフトウェアで新たに作成する。提案手法の 2 段階推定の有効性を検証するために、入力画像を別視点へ直接変換する既存手法と定性的かつ定量的に比較し、提案手法がより地形らしい 3D モデルを推定できることを示す。

2. 関連研究

1 枚の景観画像を入力とした 3 次元情報の復元に関して、代表的な研究としては CNN を用いた深度推定 [1], [2], [3], [4], [5], [6] が挙げられる。しかしこれらの手法では、入力画像から見える範囲の深度情報のみ推定

[1] 筑波大学
University of Tsukuba, Tennoudai 1-1-1, Tsukuba, Ibaraki, 305–8573, Japan
[a)] harukaoceansouth@gmail.com
[b)] endo@cs.tsukuba.ac.jp
[c)] kanamori@cs.tsukuba.ac.jp
[d)] mitani@cs.tsukuba.ac.jp

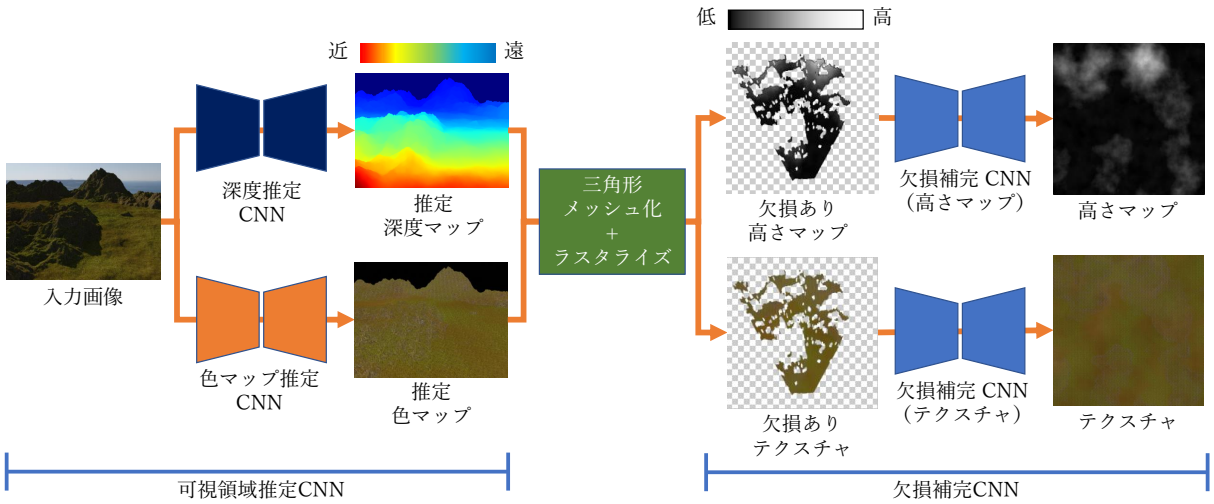

図 1 提案手法の全体像。第 1 段階で可視領域の深度と色を CNN で推定し、その結果から三角形メッシュを構築して真上からラスタライズする。そして第 2 段階で欠損部分を別の CNN を用いて補完する。

し、入力画像の非可視領域は考慮しない。

提案手法は、ある視点において撮影された画像を入力として、真上から見た地形の高さと色情報を推定する、つまり入力画像を別視点へ変換しているとみなせる。このように考えると提案手法は、敵対的生成ネットワーク（GAN）を用いて入力画像を別視点に変換する手法 [7], [8] と関連がある。これらの手法では例えば、地表付近で撮影した画像からその撮影地点の航空画像を生成できる。しかし提案手法と異なり、入力画像における可視領域と非可視領域を明示的に区別しないため、入力画像に含まれる可視領域の 3 次元情報が推定結果に十分に反映されない恐れがある。なお、提案手法のように 1 枚の地形景観画像から 3D 地形モデルを推定する手法は、我々が知る限りでは存在しない。

3. 提案手法

本研究では 1 枚の地形景観画像から、テクスチャ付きの高さマップとして表現された 3D 地形モデルを推定する。入力画像から見えない領域は推定 3 次元情報に欠損が生じるため、まず可視領域の 3 次元情報を推定し、次に非可視領域の欠損を補完する 2 段階の推定を行う。また、入力画像から影と光源の影響を除去することで、推定テクスチャに光源に依存しない不自然な影が入らないようにする。提案手法の全体像を**図 1** に示す。1 段階目では、入力画像の深度マップと色マップをそれぞれ異なる CNN で推定する。色マップとは、入力画像から影と光源の影響を除去した画像である。そして、推定した深度マップと色マップから三角形メッシュを計算し、真上から平行投影することで欠損のある高さマップとテクスチャを得る。2 段階目では、別の CNN で高さマップとテクスチャの欠損をそれぞれ補完し、最終的な 3D 地形モデルを計算する。入力画像から見えない領域には、色と形状に様々なバリエーションが考えられるため、複数の補完結果が出力できるように、変分オートエンコーダ (VAE) を用いて、欠損補完 CNN を訓練する。訓練用データセットが存在しないため、CG ソフトウェア Blender [9] で作成した。本研究で扱う地形景観画像は、問題の単純化のため、山のみの画像を対象とし、川や湖などの水面は含まないものとした。

3.1 深度推定

入力画像の深度推定には、単眼屋内画像の深度推定を行う SARPN [5] を用いる。SARPN は現時点で最も性能の良い手法の一つなので採用した。この手法は、低解像度の深度マップが大域的なシーン構造を捉え、高解像度の深度マップが詳細形状を捉えると考え、マルチスケールの特徴情報を用いて推定を行う。SARPN では、深度から計算した法線による損失関数を用いるが、本研究では推定結果の精度が悪化したため用いない。

3.2 色マップ推定

入力画像の色マップの推定には、入力ラベル画像から GAN を用いて実写画像を合成する SPADE [10] を用いる。SPADE は、通常のバッチ正規化で消失してしまう入力画像の空間情報を保つことで、高精度な結果を実現する。実写画像合成と色マップ推定ではタスクが異なるが、ネットワークモデルが有望だと考え採用した。

3.3 欠損あり高さマップとテクスチャのラスタライズ

入力画像から推定した深度と色マップから三角形メッシュを計算し、真上から平行投影することで、欠損した高さマップとテクスチャを得る。三角形メッシュは深度と色

マップの 1 画素を 1 頂点として、隣接頂点で面を張る。

3.4 高さマップとテクスチャの欠損補完

高さマップおよびテクスチャの欠損補完には、pix2pixHD++ [10] を用いる。pix2pixHD++ は高解像度の出力が可能なネットワークであるため採用した。ただし、転置畳み込みを含むオリジナルの実装では格子状のアーティファクトが発生したため、転置畳み込みをリサイズと畳み込みで置き換えた。入力画像では見えない地形領域については、VAE を用いて様々なバリエーションが得られるようにする。VAE から得られる潜在変数を生成器に入力する際には、潜在変数のチャンネル数を全結合層で調整した後に、入力画像と同じ空間解像度かつ 1 チャンネルの特徴マップに変形して、入力画像とチャンネル方向に結合させた。なお、欠損補完のために pix2pixHD++以外に SPADE も試したが、よい結果が得られなかったため採用しなかった。

3.5 データセット作成

地形景観画像データセットを CG ソフトウェア Blender [9] を用いて作成した。データセットは、ネットワークの入力となる地形景観画像と、それに対応する深度マップと色マップおよび高さマップとテクスチャからなる 4,800 セットである。このデータセットは、深度推定 CNN 以外のネットワークの訓練と推論および、深度推定 CNN の推論と検証に用いた。データ分割に関しては、訓練データが 4,000 セット、検証データが 400 セット、テストデータが 400 セットである。深度推定 CNN の訓練には、別途作成した地形景観画像と、それに対応する深度マップの 3,927 ペアを用いた。高さマップは乱数で疑似的に生成し、地形の規模は 5km 四方とした。地形景観画像の解像度は 320×240、正解の高さマップとテクスチャの解像度は 1024×1024 である。

4. 実験結果

4.1 実験環境・設定

提案手法の実装には Python 言語と PyTorch ライブラリを用い、各 CNN に 1 つずつ GPU を割り当て訓練した。所有する GPU のうち各ネットワークの訓練に必要なメモリを確保できるものを選び、深度推定 CNN には NVIDIA GeForce GTX 1080 Ti を、色マップ推定 CNN とテクスチャ欠損補完 CNN には Quadro RTX 6000 を、欠損高さマップ CNN には Quadro RTX 8000 を割り当てた。深度推定 CNN の学習エポック数は、検証データを用いて 9 エポックに決めた。その他の CNN の学習エポック数は 50 とした。訓練時間について、深度マップと色マップ推定 CNN については、どちらも 1 日程度かかった。高さマップとテクスチャの欠損補完 CNN については、それぞれ 3 日と 5 日程度かかった。最適化は全ての CNN で Adam を用いて行った。深度推定 CNN では学習率を 0.0001 とし、パラメータ β_1 は 0.9、β_2 は 0.999、減衰項は 0.00001 を用いた。その他の CNN では生成器と識別機の学習率はそれぞれ 0.0001 と 0.0004 とし、パラメータ β_1 は 0、β_2 は 0.999、減衰項は 0 を用いた。学習率の調整に関して、深度マップ推定 CNN では、5 エポック毎に 10%学習率を減衰させた。色マップ推定 CNN および高さマップとテクスチャの欠損損完 CNN では、25 エポック以降から線形に学習率を減衰させ、50 エポックで学習率が 0 になるようにした。バッチサイズは、深度マップと色マップ推定 CNN では 4 とし、高さマップとテクスチャの欠損補完 CNN では、それぞれ 2、1 とした。ラスタライズの実装に関しては、欠損高さマップのラスタライズには微分可能レンダリングライブラリ Kaolin [11] を、欠損テクスチャのラスタライズには OpenGL を用いた。なお実装の都合上 Kaolin を用いたが、今後はより高速な OpenGL による実装に変更する予定である。隣接頂点間の距離が大きく離れていると不自然に引き伸ばされた三角形面ができてしまうため、辺の長さが 100m より大きい場合は面を張らないようにした。

比較手法として入力画像の視点変換を行う Selection-GAN [8] を用い、提案手法との比較を定量的かつ定性的に行った。高さマップの定量評価には、絶対誤差 MAE、相対誤差 rel、および推定値と真値の比率 δ が閾値より小さいピクセル画素の割合を計算する精度指標を用いた。閾値には 1.25、1.25^2、1.25^3 の 3 つを用いた。テクスチャの定量評価には、GAN の出力結果の品質を評価する指標として FID を、知覚的な評価指標として LPIPS [12] を用いた。提案手法の推論時には、欠損補完 CNN の生成器に乱数ベクトルを入力し、得られた 10 個の出力の中で最良値となった評価指標が最も多い結果を採用した。

4.2 定性比較

図 2 に提案手法と比較手法の定性比較の結果を示す。推定した 3D 地形モデルのレンダリング時には、入力画像の視点と全体像が見える視点を選んだ。入力画像の視点で比較すると、提案手法の方が正解に近い形状で、かつ細かい色まで再現できていることが分かる。全体像で比較しても、提案手法の結果はより正解に近い結果であることが分かる。

4.3 定量比較

表 1 に提案手法と比較手法の定量比較の結果を示す。評価値は 1) 出力結果全体、2) 入力視点での可視領域のみ、の 2 通りを算出した。なお、FID については出力結果全体でのみ評価した。出力全体での評価と可視領域での評価ともに、全ての評価指標において提案手法が最良値となっており、提案手法は全体的かつ入力画像からみた地形の形状と

表 1 提案手法と比較手法の定量比較。太字は、各指標の最良値を示す。

手法	高さマップ					テクスチャ	
	$\delta < 1.25 \uparrow$	$\delta < 1.25^2 \uparrow$	$\delta < 1.25^3 \uparrow$	MAE ↓	rel ↓	FID ↓	LPIPS ↓
SelectionGAN (全体)	0.28	0.49	0.64	73	0.99	152.8	0.93
Ours (全体)	**0.33**	**0.55**	**0.70**	**63**	**0.78**	**30.1**	**0.13**
SelectionGAN (可視領域のみ)	0.37	0.64	0.80	52	0.47	N/A	0.035
Ours (可視領域のみ)	**0.48**	**0.74**	**0.86**	**40**	**0.37**	N/A	**0.0076**

色を比較手法よりも再現できていることが分かる。

4.4 複数の補完結果

欠損を含む高さマップ・テクスチャに対する、提案手法による複数の補完結果を**図 3・図 4** に示す。 図 3 では、形状の違いをわかりやすく示すため 3D モデルのテクスチャを除外して白一色とし、図 4 では、色のみに着目するため同一の 3D 形状を用いた。 定性比較の結果と同様に、3D 地形モデルのレンダリングには 2 通りの視点を設定した。入力視点で見ると、いずれの出力も、正解 3D モデルに近い形状と色を維持していることが分かる。一方で、全体像で見ると、それぞれ地形形状と色が異なることが分かる。よって提案手法は、入力画像の可視領域の地形を忠実に再現しつつ、非可視領域では地形の様々な形状と色を出力できていると言える。

5. おわりに

本研究では、1 枚の地形景観画像からテクスチャ付き高さマップとして 3D 地形モデルを推定する、CNN による教師あり学習手法を提案した。欠損のない 3D 地形モデルを出力するために、入力画像では欠損してしまう 3 次元情報を補完する 2 段階推定を行った。また欠損補完では、様々な地形の形状と色を出力できるように VAE を用いて訓練を行った。実験により、提案手法が入力画像に忠実で、全体的にもそれらしい 3D 地形モデルを出力できることを定性的かつ定量的に示した。

今後の課題としては、実写画像への汎化性能の向上が挙げられる。現状、本手法は CG データセットを用いているため、実写画像に適用しても、それらしい地形モデルを推定できない。実写画像と CG 画像の統計量の違いにより、汎化性能が低下していると考えられる。この実写と CG 画像の統計量の違いを考慮するために、スタイル変換手法を用いた CG 画像の実写風変換やドメイン適応手法の利用を検討している。また、現在は地形データを手続き的に作成しており、実際の地形と見た目が異なるため、より実写に近いデータセットを作成する予定である。より具体的には、実測の標高数値データや航空写真の利用を考えている。

参考文献

[1] F. Huan, G. Mingming, W. Chaohui, B. Kayhan, and T. Dacheng. Deep Ordinal Regression Network for Monocular Depth Estimation. In *CVPR*, pp. 2002–2011, June 2018.

[2] M. Ramamonjisoa and V. Lepetit. SharpNet: Fast and Accurate Recovery of Occluding Contours in Monocular Depth Estimation. *ICCV Workshops on 3D Reconstruction in the Wild*, 2019.

[3] W. Yin, Y. Liu, C. Shen, and Y. Yan. Enforcing Geometric Constraints of Virtual Normal for Depth Prediction. In *ICCV*, pp. 5683–5692, 2019.

[4] Y. Wu, V. Boominathan, H. Chen, A. Sankaranarayanan, and A. Veeraraghavan. PhaseCam3D — Learning Phase Masks for Passive Single View Depth Estimation. In *ICCP*, pp. 1–12, 2019.

[5] C. Xiaotian, C. Xuejin, and Z. Zheng-Jun. Structure-Aware Residual Pyramid Network for Monocular Depth Estimation. In *Proceedings of the International Joint Conferences on Artificial Intelligence*, pp. 694–700, 2019.

[6] S. F. Bhat, I. Alhashim, and P. Wonka. AdaBins: Depth Estimation using Adaptive Bins. *arXiv:2011.14141*, 2020.

[7] K. Regmi and A. Borji. Cross-view Image Synthesis Using Conditional GANs. In *CVPR*, pp. 3501–3510, 2018.

[8] H. Tang, D. Xu, N. Sebe, Y. Wang, Jason J. J. Corso, and Y. Yan. Multi-Channel Attention Selection GAN With Cascaded Semantic Guidance for Cross-View Image Translation. In *CVPR*, pp. 2417–2426, 2019.

[9] Blender. `https://www.blender.org/`. (最終アクセス日 2020-10-14).

[10] T. Park, M. Liu, T. Wang, and J. Zhu. Semantic Image Synthesis with Spatially-Adaptive Normalization. In *CVPR*, pp. 2337–2346, 2019.

[11] Krishna Murthy Jatavallabhula, Edward Smith, Jean-Francois Lafleche, Clement Fuji Tsang, Artem Rozantsev, Wenzheng Chen, Tommy Xiang, Rev Lebaredian, and Sanja Fidler. Kaolin: A PyTorch library for accelerating 3D deep learning research, 2019.

[12] R. Zhang, P. Isola, A. A. Efros, E. Shechtman, and O. Wang. The Unreasonable Effectiveness of Deep Features as a Perceptual Metric. In *CVPR*, pp. 586–595, 2018.

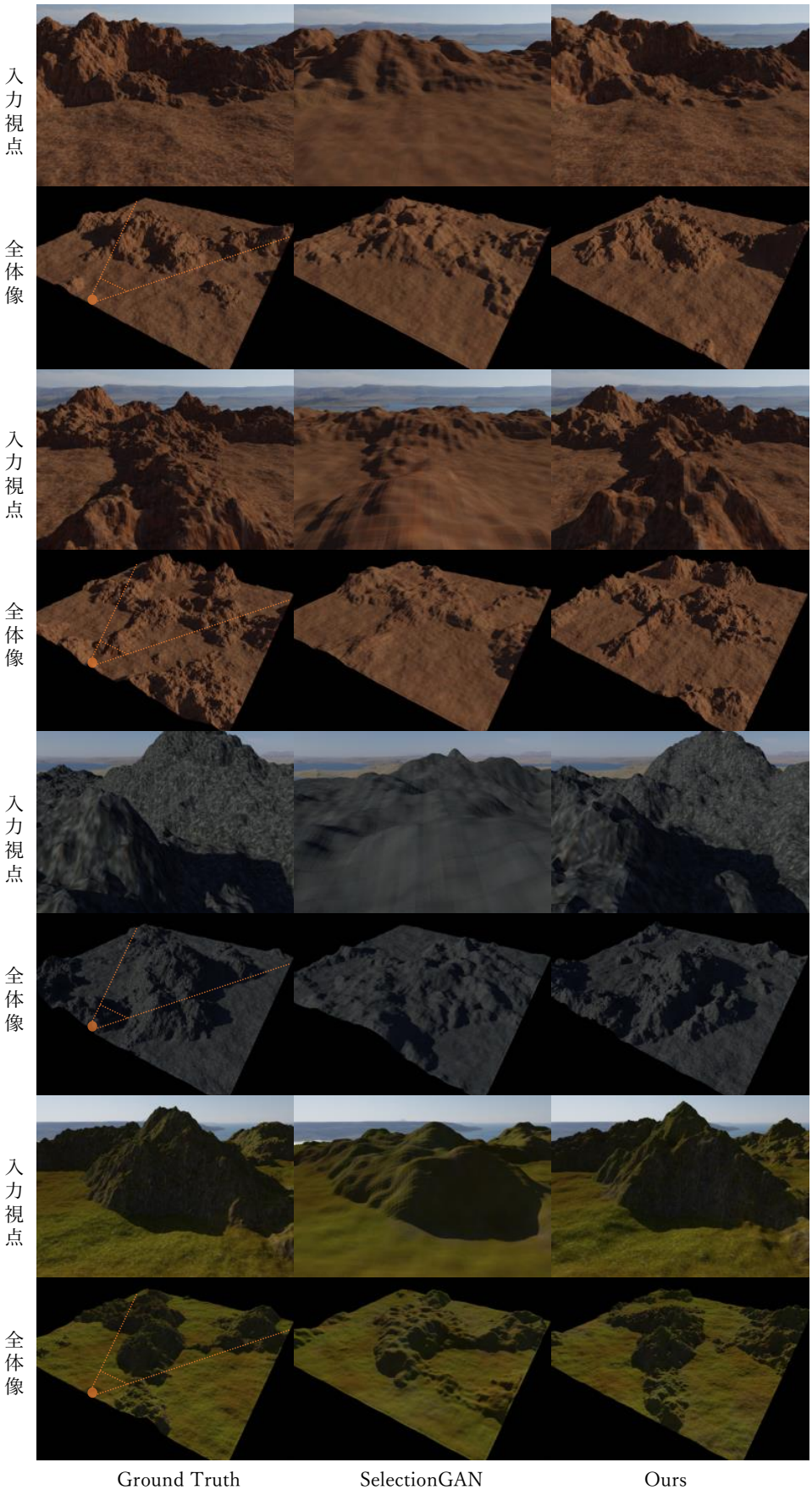

図 2 推定 3D モデルの定性比較。正解 3D モデルの全体像のオレンジ色の部分は、入力視点の位置と視野範囲を表す。

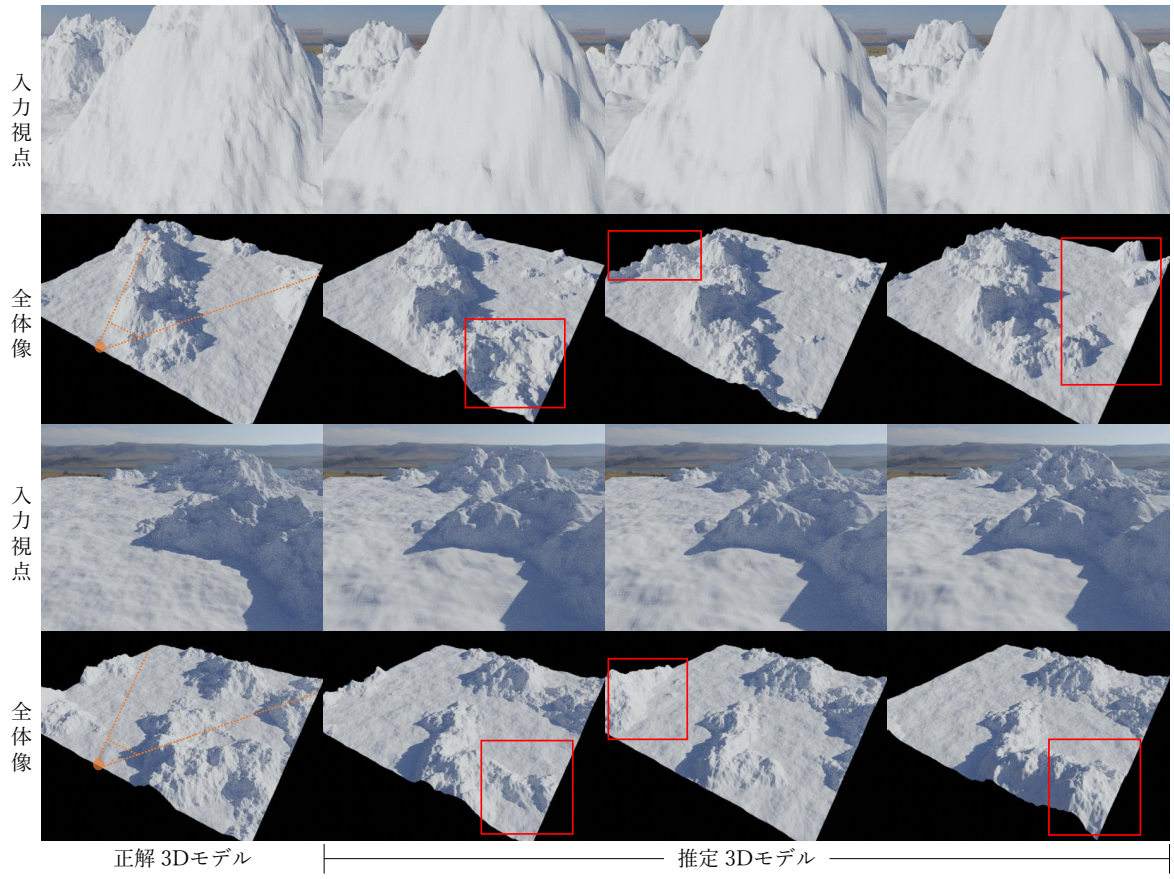

図 3 提案手法による欠損高さマップの複数の補完結果の定性比較。形状の違いをわかりやすく示すために色は白で統一した。赤枠は他の 3D モデルと特に形状が異なる部分。正解 3D モデルの全体像のオレンジ色の部分は、入力視点の位置と視野範囲を表す。

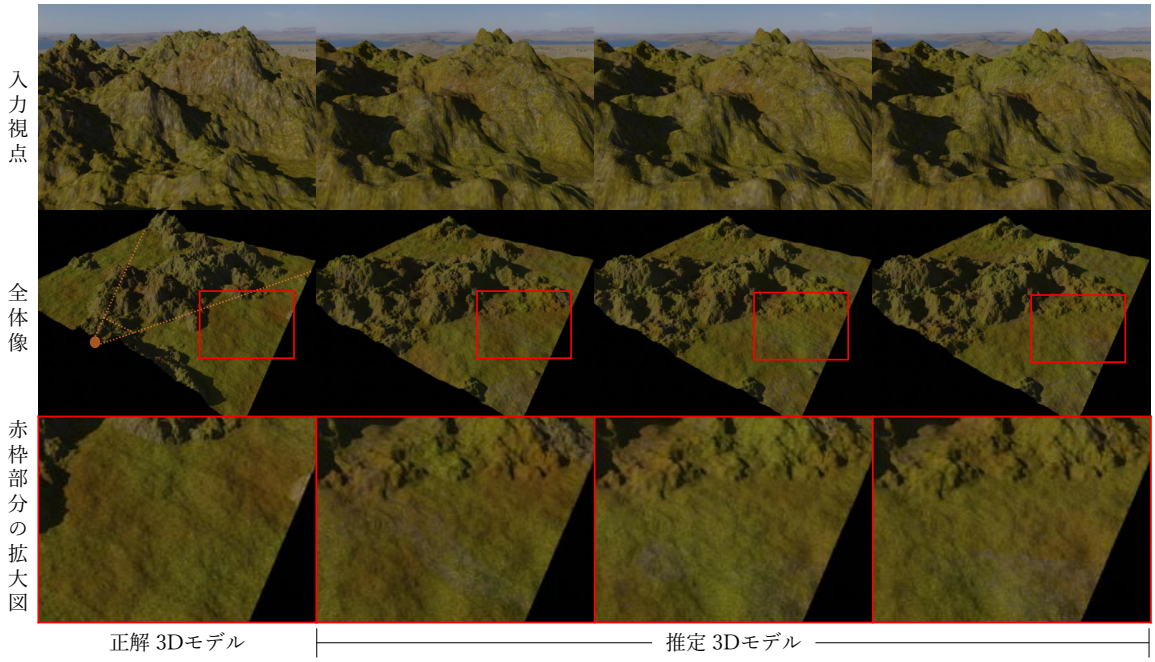

図 4 提案手法による欠損テクスチャの複数の補完結果の定性比較。色の違いに着目するために、推定 3D 形状は同一のものを用いた。3 段目は、2 段目の赤枠部分の拡大図。正解 3D モデルの全体像のオレンジ色の部分は、入力視点の位置と視野範囲を表す。