Vol.2021-CG-183 No.2
2021/8/25

CG シーンの学習に基づく法線と輪郭線の推定と 3 次元モデリングへの応用

山梨 傑[1]　　岡部 誠[1]

概要：我々は写真や絵画などの 2 次元画像から 3 次元形状をモデリングするための手法を研究している。2 次元画像から 3 次元形状を再構築することはコンピュータビジョンにおける本質的な課題のひとつである。近年、畳み込みニューラルネットワークに基づく深層学習によって、単一画像からでも精度の高い深度マップが得られるようになった。しかし、それらの深度マップから 3 次元形状をポリゴンモデルとして起こしてみると必ずしも良い形状が得られない。そこで、我々は単一画像から深度マップではなく、法線マップと輪郭線マップを推定し、それらの情報を基にポアソン方程式を解くことで 3 次元形状を得るような手法を提案する。法線マップと輪郭線マップの推定に用いる深層学習モデルは U-Net であり、前者の推定には平均二乗誤差、後者の推定には Dice 係数を損失関数に用いる。コンピュータグラフィックスによって生成したデータセットを用いて学習と評価を行ったので報告する。

キーワード：画像処理、深層学習、3 次元形状モデリング

1. はじめに

2 次元画像から 3 次元形状を再構築することはコンピュータビジョンにおける本質的な課題のひとつである。また、ゲームや映像などを生産するコンテンツ産業においても、画像から容易に形状がモデリングできるようになれば、生産工程の効率化につながる。

近年、畳み込みニューラルネットワーク(CNN)に基づく深層学習によって、単一画像からでも精度の高い深度マップが得られるようになり、多くの論文が発表されている[1,4]。また、人の顔[2]や人体の形状[3]に限定した手法では、単一画像からでも正確な 3 次元形状モデルを得ることができる。また、3 次元形状とそのレンダリング画像との関係を学習する手法[5,6,7]もある。これらの手法を観察してみると、対象形状を人の顔や人体、また自動車や飛行機などのいくつかの工業製品などに絞った場合には精度の高い 3 次元形状が得られることが分かる[2,3,5,6,7]。一方で、対象が絞り込まれていない場合、例えば、車載カメラや室内の画像から推定された深度マップは精度が高いとは言えず、そこから 3 次元形状をポリゴンモデルとして起こしてみると必ずしも良い形状が得られない[1,4]。また、対象を工業製品に絞り込んだ場合[5,6,7]でも、形状の細部のモデリングに関してはまだ課題が残っているものと思われる。

そこで我々はより一般的な対象を扱いつつ、形状の細部までをモデリングできるような手法の研究を行っている。今回は単一画像を入力とし、深度マップではなく、法線マップと輪郭線マップを推定し、それらの情報をポアソン方程式を解くことで積分し、3 次元形状を得るような手法を提案する。法線マップとは 3 チャンネルの画像であり、画像中の物体表面の 3 次元法線ベクトルを各ピクセルが保持している。輪郭線マップとは 1 チャンネルの画像であり、各ピクセルは、そのピクセルが画像中の物体の輪郭線上に位置しているかどうかをバイナリで保持している（位置していれば１、位置していなければ０）。

このようなアプローチをとる理由は、Koenderink らが *Pictorial Relief* [8]にて報告している人間の視覚認識の実験結果に触発されてのことである。この実験では、単一画像から絶対的な深度を推定することは人間にとって難しく、一方で、単一画像から法線や相対的な深度を推定することは人間にとって簡単である、という事実を確認している。従って、我々人間の脳を模倣したニューラルネットワークも、深度マップの推定は難しいが、法線マップや輪郭線マップの推定は正確に行えるのではないか、と考えた。法線マップの推定手法としては、イラストにシェーディングを行うために CNN の一種である U 字型ネットワーク(U-Net)[9]を用いて法線マップを推定する手法[10]や、単一の RGB-D 画像から法線マップを推定する手法[11]などがある。

提案手法において、法線マップと輪郭線マップの推定に

1 静岡大学大学院総合科学技術研究科工学専攻
数理システム工学コース

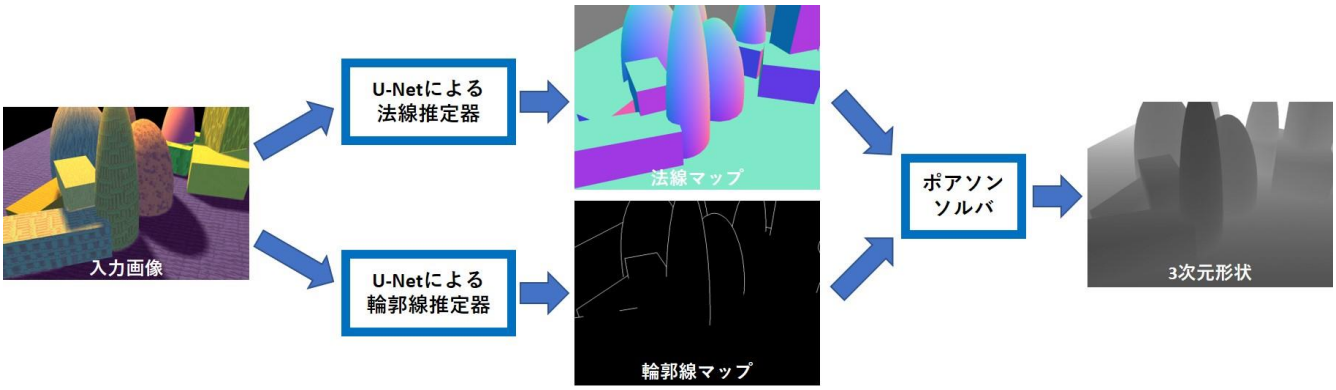

図 1．提案手法の概要。入力画像から U-Net によって法線マップと輪郭線マップを推定した後、それらから合成した勾配情報をポアソンソルバで積分することで 3 次元形状を得る。

用いる深層学習モデルは U-Net[9]であり、前者の推定には平均二乗誤差、後者の推定には Dice 係数を損失関数に用いる。学習と評価には、コンピュータグラフィックス(CG)によって生成されたデータセットを用いる。今回のデータセットにおいては、法線マップと輪郭線マップの両方について高い精度で推定することができた。しかし、データセットに含まれないような種類の画像に対しては失敗することがあり、その結果を踏まえて更にデータセットに改良を加えていきたいと考えている。

2. 提案手法

図 1 に提案手法の概要を示す。提案手法はまず、入力画像を U-Net によって構築された法線マップ推定器と輪郭線マップ推定器に適用する。推定された法線マップと輪郭線マップから、最終的に求めたい 3 次元形状（深度マップ）の勾配情報を合成する。その勾配情報をポアソンソルバで積分することで深度マップを得て、それをポリゴン化することで最終的な 3 次元形状を得る。

2.1 法線マップ推定器

法線マップ推定器に RGB の 3 チャンネルから成る画像を入力すると、同じサイズの 3 チャンネルから成る法線マップを出力する。法線マップの各ピクセルは、入力画像中の物体表面の 3 次元法線ベクトルを保持している。

推定器には U-Net を用いた。U-Net の構造は Ronneberger らが提案したもの[9]と同様であるが、各畳み込み層の直後にバッチ正規化層を追加した点が異なっている。

損失関数には平均二乗誤差を用いた。推定された法線マップを$\hat{N}$ 、正解の法線マップをN 、画像空間内の全ピクセルの集合をP、ピクセルの位置をpとすると、損失関数は

$$L = \frac{1}{|P|}\sum_{p\in P}|\hat{N}_p - N_p|^2$$

と定義される。

2.2 輪郭線マップ推定器

輪郭線マップ推定器に RGB の 3 チャンネルから成る画像を入力すると、同じサイズの 1 チャンネルから成る輪郭線マップを出力する。輪郭線マップの各ピクセルは、そのピクセルが入力画像中の物体の輪郭線上に位置しているかどうかをバイナリで保持している（位置していれば 1、位置していなければ 0）。

推定器には U-Net を用いた。法線マップ推定器の U-Net と異なる点は、最終層のチャンネル数が 3 から 1 になった点のみである。

損失関数には画像のセグメンテーションでもよく使われる Dice 係数を用いた。推定された輪郭線マップを$\hat{C}$ 、正解の輪郭線マップをC 、画像空間内の全ピクセルの集合をP、ピクセルの位置をpとすると、損失関数は

$$L = 1 - 2\frac{\sum_{p\in P}\hat{C}_p C_p}{\sum_{p\in P}\hat{C}_p + \sum_{p\in P}C_p + 1}$$

と定義される。

3. 実験結果

3.1 データセット

今回の実験で用いたデータセットは、CG で描かれた 70000 個のシーンから成る。そのうち 60000 個を法線マップ推定器及び輪郭線マップ推定器の学習に、残りの 10000 個をそれぞれの推定器の評価に用いた。

図 2．データセットは 1 つのシーンにつき入力画像、法線マップ、輪郭線マップの 3 つの画像から成る。

各シーンは図 2 のように、入力画像、法線マップ、輪郭線

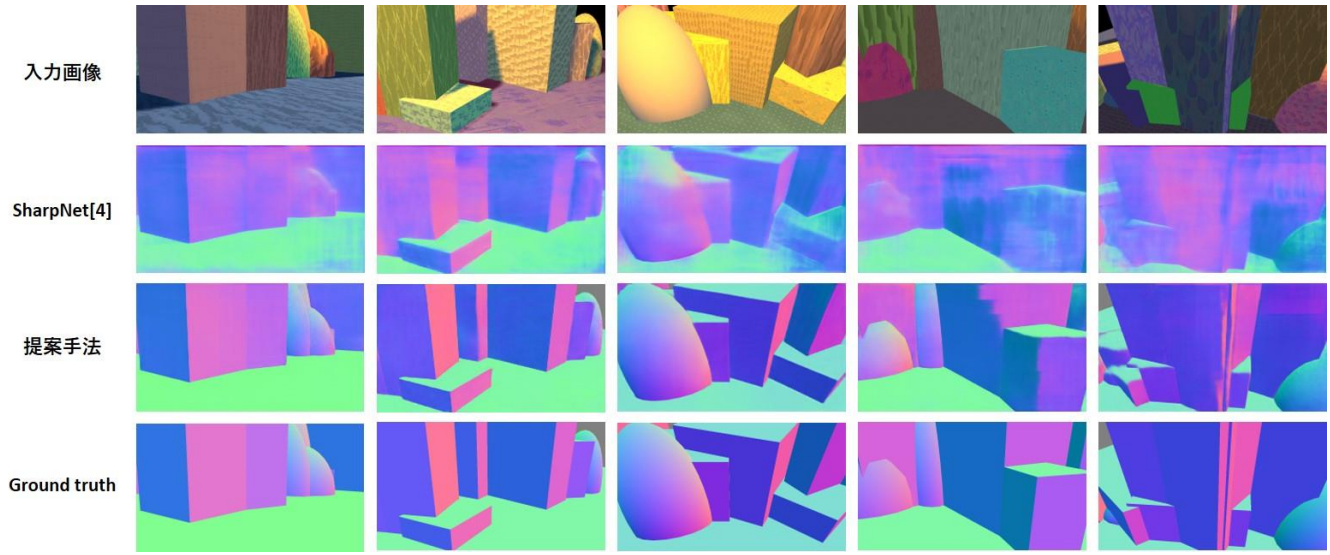

図３．5 つのシーンについて法線マップの推定結果を示す。我々のデータセットに関しては、SharpNet[4]に比べると提案手法によって推定された法線マップの方がシャープで正確である。一番右の列に失敗例を示す。

マップの 3 つの画像から成る。各画像の解像度は 256×144 ピクセルである。各シーンは地面を表す平面と、その上に配置された 18 個の 3 次元物体から成る。地面の色とテクスチャはランダムに設定される。各 3 次元物体は直方体または楕円体であり、その位置、スケール、方向、色、テクスチャはランダムに設定される。また、レンダリングの際のカメラの位置と方向はランダムに設定される。全シーンについて、照明は同一パラメータに設定された点光源 1 つから成る。入力画像のレンダリングには局所照明モデルと、アンチエイリアシング処理を施したシャドウマップを用いた。輪郭線マップは、入力画像のレンダリングの際に使われたデプスバッファを用いて計算した。デプスバッファのx軸方向とy軸方向の微分を計算し、その値の大きいピクセルには輪郭線として 1 を与え、それ以外のピクセルには輪郭線でないとして 0 を与えた。

3.2 法線マップ推定器の学習と推定結果

Keras で実装した U-Net を 60000 個のシーンに対し学習した。NVIDIA GeForce GTX 1080 Ti を使用したところ、1 エポックあたり約 30 分を要し、42 エポック目で validation loss が 0.0025 で最小となった。このモデルを用いて推定された法線マップを図 3 に示す。我々のデータセットに関しては、提案手法はほとんどの場合に Ground truth に近い法線マップを精度よく推定できていることが分かる。また、SharpNet[4]に比べても、提案手法によって推定された法線マップの方がシャープで正確である。一番右の列には失敗例を示す。ここでは特に暗い箇所において、正しい法線が推定できていないが、この入力画像においては我々人間でも法線の推定が難しい。

3.3 輪郭線マップ推定器の学習と推定結果

Keras で実装した U-Net を 60000 個のシーンに対し学習した。NVIDIA GeForce GTX 2080 Ti を使用したところ、1 エポックあたり約 24 分を要し、41 エポック目で validation loss が 0.0536 で最小となった。このモデルを用いて推定された輪郭線マップを図 4 に示す。我々のデータセットに関しては、提案手法はほとんどの場合に Ground truth に近い輪郭線マップを精度よく推定できていることが分かる。また、比較として SharpNet[4]によって推定された境界線マップ(boundary)を二行目に示す。注意として、SharpNet[4]によって推定された境界線マップは物体の境界線が白くなっているが、我々の輪郭線マップは物体の境界線であっても、その面が滑らかにつながっている場合白くならないため、結果が一致していない箇所がある。提案手法によって推定された輪郭線マップの方がはっきりと輪郭線を捉えており、正確である。一番右の列には失敗例を示す。ここでは特に手前の物体と奥の物体に距離がある箇所において、正しい輪郭線が推定できていないが、この入力画像においては我々人間でも輪郭線の推定が難しい。

4. まとめ

本論文では、U-Net を使い一枚の画像から法線マップと輪郭線マップを推定した。どちらも今回の実験で用いたような簡単な CG シーンの場合、既存手法を上回る結果が得られた。一方で、人間でも判断が難しいような箇所や複雑なシーンの場合、あまり望ましい結果が得られなかった。今後はデータセットの改善やネットワークの改善などを行い、上記のような画像であっても正確に推定ができるよう精度の向上を目指していきたい。

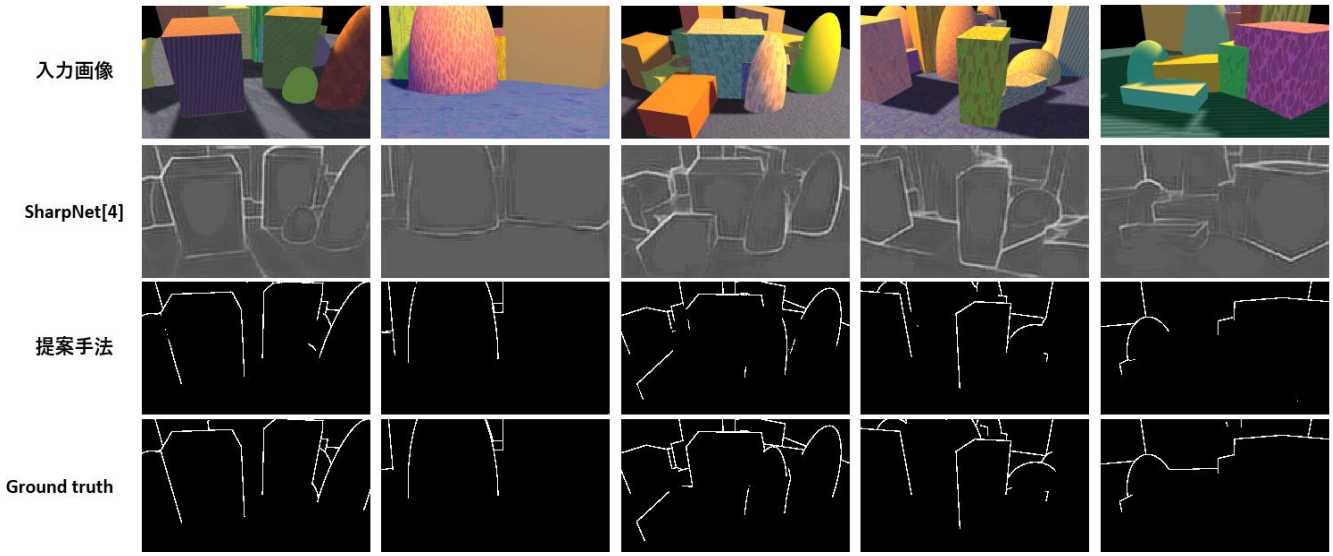

図４．５つのシーンについて輪郭線マップの推定結果を示す。我々のデータセットに関しては、SharpNet[4]に比べると提案手法によって推定された輪郭線マップの方がはっきりと輪郭線を捉えており正確である。一番右の列に失敗例を示す。

尚、法線マップと輪郭線マップが得られた後、それらから勾配情報を合成し、ポアソンソルバによって積分することで深度マップと 3 次元形状が得られるが、それらのレンダリング結果については研究会での口頭発表にて報告する。

参考文献

[1] Shariq Farooq Bhat, Ibraheem Alhashim, Peter Wonka, “AdaBins: Depth Estimation using Adaptive Bins”, CVPR 2021.

[2] Bernhard Egger, William A. P. Smith, Ayush Tewari, Stefanie Wuhrer, Michael Zollhoefer, Thabo Beeler, Florian Bernard, Timo Bolkart, Adam Kortylewski, Sami Romdhani, Christian Theobalt, Volker Blanz, Thomas Vetter, “3D Morphable Face Models - Past, Present and Future”, SIGGRAPH 2021.

[3] Shunsuke Saito, Zeng Huang, Ryota Natsume, Shigeo Morishima, Angjoo Kanazawa, Hao Li, “PIFu: Pixel-Aligned Implicit Function for High-Resolution Clothed Human Digitization”,

[4] Michaël Ramamonjisoa, Vincent Lepetit, “SharpNet: Fast and Accurate Recovery of Occluding Contours in Monocular Depth Estimation”, ICCVW 2019.

[5] Xinchen Yan, Jimei Yang, Ersin Yumer, Yijie Guo, Honglak Lee, “Perspective Transformer Nets: Learning Single-View 3D Object Reconstruction without 3D Supervision”, NIPS 2016.

[6] Hiroharu Kato, Yoshitaka Ushiku, Tatsuya Harada, “Neural 3D Mesh Renderer”, CVPR 2018.

[7] Lars Mescheder, Michael Oechsle, Michael Niemeyer, Sebastian Nowozin, Andreas Geiger, “Occupancy Networks: Learning 3D Reconstruction in Function Space”, arXiv:1812.03828 [cs.CV].

[8] J. J. Koenderink, “Pictorial Relief”, Phil. Trans. of the Roy. Soc.: Math., Phys, and Engineering Sciences, 356(1740), pp. 1071-1086, 1998.

[9] Olaf Ronneberger, Philipp Fischer, Thomas Brox, “U-net: Convolutional networks for biomedical image segmentation”, Medical Image Computing and Computer-Assisted Intervention (MICCAI), Springer, LNCS, Volume 9351, pages 234-241, 2015.

[10] Matis Hudon, Rafael Pagés, Mairéad Grogan, and Aljosa Smoli. Deep normal estimation for automatic shading of hand-drawn characters. In: ECCV Workshops (2018).

[11] Jin Zeng, Yanfeng Tong, Yunmu Huang, Qiong Yan, Wenxiu Sun, Jing Chen, and Yongtian Wang. Deep surface normal estimation with hierarchical RGB-D fusion. In: CVPR (2019).