Vol.2019-CVIM-215 No.19
2019/1/18

深度画像変換による単一RGBD画像からの植物枝形状復元

井手 絢香[*1]　大倉 史生[*1*2]　松下 康之[*1]　八木 康史[*1]

Abstract – 本研究は、CG モデリングの省力化や植物栽培の高度化のための基盤技術として、植物の枝構造の三次元復元を行うことを目指す。植物の三次元復元は自己遮蔽の多い構造上難しい問題であると知られており、特に葉による遮蔽が多く発生する枝の復元は挑戦的な課題である。植物の三次元復元を行う既存研究では多くの視点数や静的な環境を要求するため、風などの影響による植物の形状変化が発生する環境での運用は難しい。そこで本研究では、単一視点から撮影された RGBD 画像を入力として、深層学習を用いた画像変換技術を利用して植物の深度画像を推定することにより、手軽な入力から植物の枝構造の深度推定を実現する。実験の結果、CG 植物画像および、RGBD カメラで撮影した実植物画像からの枝形状推定の可能性が示された。

Keywords : Pix2Pix、植物形状復元、GAN、深層学習、三次元復元

1 はじめに

本研究は、手軽な入力から植物の三次元構造（枝構造、葉のつき方）を復元することを目的とする。複合現実感（MR）応用に欠かせない三次元モデリング作業において、植物や樹木は手間のかかる対象の一つである。特に、実環境に存在する植物の仮想化には、植物の構造を復元する必要がある。植物画像群を入力とした三次元モデリングについての研究はコンピュータグラフィックス（CG）分野を中心に行われているが、ワンショット撮影などの手軽な入力で、かつ実物体に即した三次元枝形状の推定を行う手法はいまだ実現されていない。

植物形状復元の応用は、MR/CG 分野にとどまらない。果樹などの植物の栽培において、各個体の日々の成長過程を把握し、将来の生育傾向や適切な栽培管理を行うことは、高品質な作物を生産する上で非常に重要である。適切な栽培管理には日々の観察と高度な知識が要求される。しかし、栽培従事者の減少や高齢化に伴い、栽培従事者の省力化、栽培管理ノウハウの継承、品質の向上を両立することが困難になっており、我が国の農業の持続可能性に重大な危機が迫っている。植物の状態（構造や成長過程など）を栽培従事者の代わりに、人の目よりも高頻度・高精度にモニタリングするための基盤技術として、植物の構造を正確に復元することは不可欠である。

一般に植物は自己遮蔽の多い構造をしており、複数台のカメラを用いてもその構造を完全に捉えることは困難である。枝構造推定に関する先行研究 [1] では、複数視点のカメラから得られた画像それぞれについて、深層学習を用いて枝位置の推定を行い、ボクセル空間上で三次元復元を行っている。しかし、この手法は数十視点から撮影された多視点画像を必要とし、撮影中の風の影響などによる植物の形状変化に対応できない。そこで、本研究では、単一画像からの三次元枝形状推定を実現するため、深度情報を含む RGBD カメラで取得した画像を活用する。

提案手法では、[1] でも用いられた Generative Adversarial Network (GAN) による画像変換手法である Pix2Pix [2] を、4 チャネル画像を入力できるよう拡張し、単一 RGBD 画像から遮蔽部分を含む枝の深度画像を推定する。しかし、画像変換の単純な 4 チャネル拡張では、空間的には適切な位置に枝の存在を推定することが多い一方、深度方向の推定値の安定性に問題がみられた。そこで本研究では、損失関数に深度方向の平滑化項を導入し、滑らかな深度を得るよう工夫した。CG 植物画像を対象にした実験の結果、平滑化項の導入による深度推定精度の改善が見られた。また、RGBD カメラで撮影した実植物画像からの深度推定の可能性も示された。

2 関連研究

本章では、植物の三次元復元および画像変換についての関連研究を紹介し、本研究の位置づけを明らかにする。

2.1 植物の三次元復元

植物の三次元形状モデリング技術は、古くから CG 分野で着目されてきた [3]。樹木や植物の構造の複雑さに起因して、手作業での CG モデリングに手間がかかることが課題であり、省力化のための自動・半自動モデリング手法が多く提案されている [4, 5, 6]。特に、多視点植物画像を入力として樹木の枝葉を生成する手法

[*1] 大阪大学
[*2] JST さきがけ

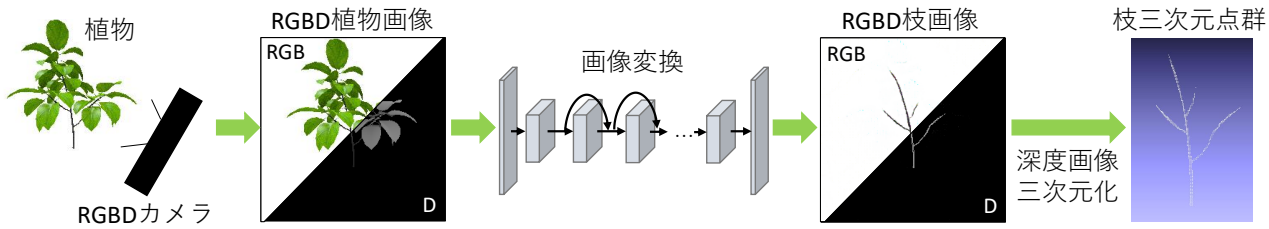


図 1　提案手法の流れ : RGBD カメラで植物の RGBD 画像を取得し、画像変換ネットワークを用いて RGBD 枝画像を生成する。得られた枝の深度画像から対象植物の三次元枝形状を復元する。

は 2000 年台以降活発に研究例が見られる [7, 8, 9]。これらの手法は、観測された樹木のシルエット形状に合致する「見た目がそれらしい」モデルを自動生成する。しかし、実植物を正確に再現したモデルの生成という観点では、「見た目がそれらしい」モデルでは不十分であり、枝構造を正確に再現することが必要である。葉がすべて落ちた状態の植物が観測できる場合は、様々な枝構造の三次元復元手法が提案されている [10, 11] が、実際の植物は多くの葉がついていることから、植物を多視点で観測しても、多くの場合遮蔽が残存する。遮蔽を考慮した枝構造の三次元復元手法として、深層学習を用いた枝位置の推定と三次元復元を組み合わせた手法 [1] や、時系列・多視点の三次元スキャンに基づき時系列植物構造を復元する手法 [12] が提案されているが、数十以上の視点数を必要とするとともに、撮影時の形状変化が発生しない環境を前提としている。

2.2　画像変換

画像変換は、image-to-image translation [2] とも呼ばれ、画像のドメイン間の変換（例えば線画から写真など）を実現する技術である。本技術は、深層学習の登場以前から CG 分野で活発な研究が行われてきた。画像のテクスチャに着目して変換するテクスチャ合成（texture synthesis）やテクスチャ変換（texture transfer）、色調を変換する変換（color transfer）、少し一般的な文脈としてスタイル変換（style transfer）などと呼ばれる手法がその一例である。これらの手法は、深層学習登場以前はパッチの合成による手法が主流であり、image quilting [13] や image analogies [14] に端を発する。これまで、類似パッチ探索の高速化 [15] などの研究が行われているが、近年の深層学習を利用した GAN 研究の進展により、従来と比較し格段に高品質な画像のドメイン間変換が実現できるようになった [2]。深層学習を用いた画像変換では、多数の学習画像群から変換前ドメインから変換後ドメインへの対応を学習し、学習済み画像生成器により変換を行う。

本研究では、CG シミュレーションで生成された葉付き・葉なしの RGBD 植物画像ペアを学習画像として、条件付き GAN の一種である Pix2Pix [2] を改良したネットワークを用いて画像変換を行う。画像生成系の深層学習を深度画像に応用する研究はこれまでも行われている。例えば、Zhang らの研究 [16] では RGB 画像から対象となる空間の表面法線やオクルージョン境界を推定し、深度センサで取得された生の深度データと組み合わせることで、空間の深度推定を行う。一方、本研究のように直接 RGBD 画像間の変換を行う際、深度方向の推定値の安定性に問題が見られた。そのため、本研究ではさらなる損失関数の導入により、RGBD 画像変換の精度および安定性の向上を図る。

2.3　本研究の位置付け

既存の植物の三次元復元手法の多くが植物のシルエット形状に合致する植物モデルを生成する一方、撮影対象の植物の各枝に着目し、構造を詳細に復元する手法が提案されている。本研究は後者を目標とし、既存研究 [1, 12] で前提となる多視点撮影・静的環境という条件を課さない、ワンショット RGBD 撮影による入力に基づく新たな植物枝形状復元手法を提案する。また、提案手法の核となる、画像変換手法を応用した RGBD 画像間の変換は、著者らの知る限りいまだ試みられていない。本研究は、従来試みられなかった深度チャネル間の変換の際に発生する問題を明らかにするとともに、深度推定のための新たな損失関数を提案する。

3　RGBD 画像変換による枝位置推定

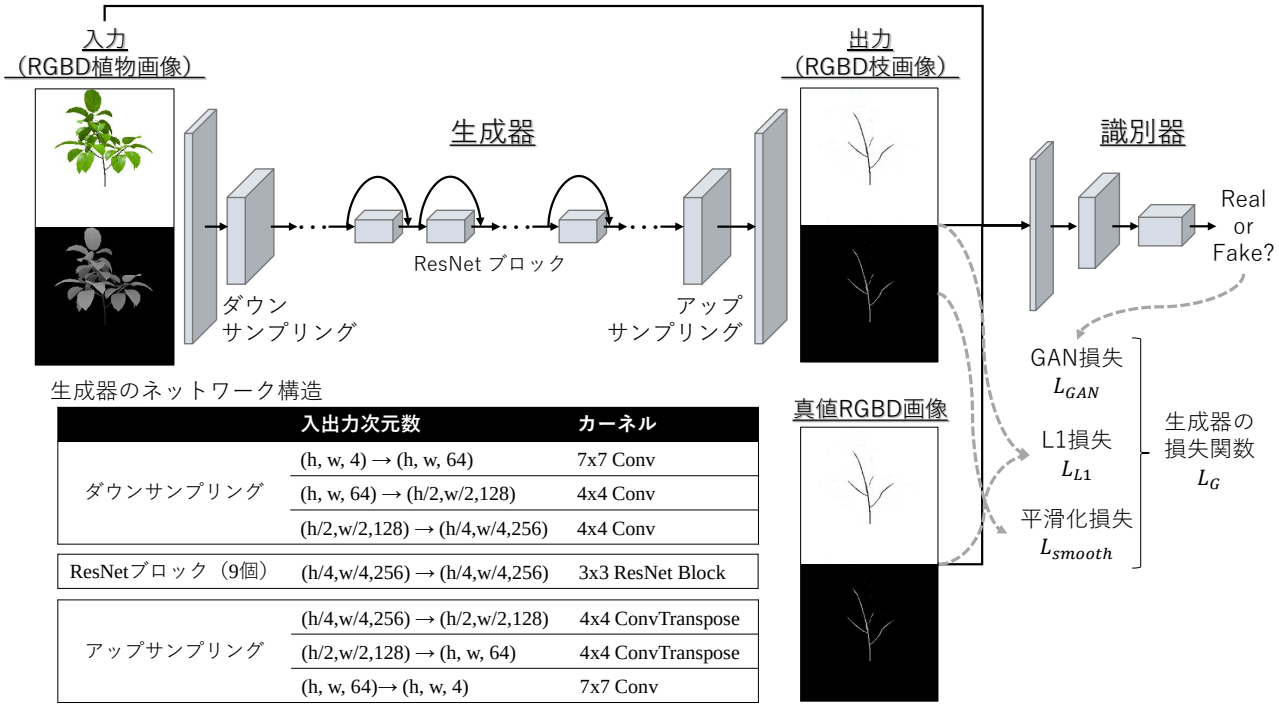
3.1　概要

提案手法の概要を図 1 に示す。提案手法は、RGBD カメラを用いて撮影した RGBD 植物画像をネットワークの入力とし、出力として植物の枝を推定した RGBD 画像を得る。得られた RGBD 枝画像の深度チャネルを三次元に逆投影し、三次元点群を得る。

3.2　RGBD 画像変換

本研究で用いるネットワークは、Pix2Pix [2] の PyTorch 版実装[1]に改良を施したものである。図 2 に、画像変換ネットワークの概念図および構造を示す。具体

[1] https://github.com/junyanz/pytorch-CycleGAN-and-pix2pix



生成器のネットワーク構造

	入出力次元数	カーネル
ダウンサンプリング	(h, w, 4) → (h, w, 64)	7x7 Conv
	(h, w, 64) → (h/2,w/2,128)	4x4 Conv
	(h/2,w/2,128) → (h/4,w/4,256)	4x4 Conv
ResNetブロック（9個）	(h/4,w/4,256) → (h/4,w/4,256)	3x3 ResNet Block
アップサンプリング	(h/4,w/4,256) → (h/2,w/2,128)	4x4 ConvTranspose
	(h/2,w/2,128) → (h, w, 64)	4x4 ConvTranspose
	(h, w, 64)→ (h, w, 4)	7x7 Conv

図 2　RGBD 画像変換：GAN に基づく画像変換ネットワークに RGBD 植物画像を入力し、枝画像に変換する。画像生成器には、表現力の高い ResNet を用いたネットワークを用いる。また、従来生成器の損失関数に用いられた GAN 損失項、L1 損失項に加えて、生成された深度の平滑性を評価する平滑化損失を導入する。

的には、RGB 画像の入力を前提とした実装を拡張し、4 チャネル（RGBD）画像入力を実装するとともに、深度チャネルの平滑化に関する損失関数を導入する。

Pix2Pix のネットワークは、条件付き GAN と呼ばれるネットワークの一種である。Generator で入力となる学習データに近くなるよう画像を作成し、その画像が本物かどうかを Discriminator が判断することを繰り返し、Generator の画像生成能力と Discriminator の判断能力が互いに高め合いながら学習が進むネットワークである。本研究では、Generator のネットワーク構造として、Pix2Pix [2] で採用された U-Net [17] と比較し表現力の高い、ResNet を用いた画像生成器 [18] を利用した。同生成器は、CycleGAN [19] で用いられた Generator の構造と同一である。

本研究では、従来 RGB の 3 チャネル入力に対応する上記ネットワークを 4 チャネル画像入力に対応するように拡張した。RGBD 画像を入力とするにあたり、本研究では RGB 画像と深度画像の位置合わせを行い、さらに入力画像を 4 チャネル 8 ビットの画像として扱うこととした。Microsoft Kinect などの RGBD カメラを用いた撮影においては、事前に計算されたキャリブレーション情報から、深度画像座標上での RGB 値を計算することで、位置合わせ済みの RGBD 画像を作成し、画像変換への入力とする。また、深度値を 8 ビット画像として扱うため、深度チャネルの near clip 面と far clip 面を設定し、その間を線形 256 階調でサンプリングした。

また、従来研究 [1] と同様、植物体を除く背景領域の影響を避けるため、背景領域については各チャネルに最大値を格納する。つまり、RGB の 3 チャネルについては背景領域を白色とし、深度チャネルについては背景領域には最大深度 d_{max} を格納する。本研究では、RGBD カメラ等による撮影時の背景削除の効率化ために、植物領域のバウンディングボックスを手作業で指定し、ボックス内の深度値に基づき、近景以外を背景領域とした RGBD 画像を半自動で作成するツールを実装した。

しかし、Pix2Pix のネットワークを 4 チャネル画像入力に対応するよう拡張するだけでは、枝領域において本来滑らかに変化するべき深度値が隣接画素で大きく異なる値をとることがあり、三次元空間上で植物の枝らしい形状をとるような深度画像を復元することがが難しい。これは、Pix2Pix で用いられる真値画像との L1 損失が、画素の隣接関係を考慮しないことが原因であると考えられる。そこで、本研究では、輝度値の変化を滑らかにするための平滑化項を、Generator の損失関数に新たに導入した。単純に隣接画素との差をとる平滑化項は、画像生成における出力画像の平滑

化を目的として [20] などで導入されているが、そのままでは背景と前景の境界部も含めて滑らかになることにより、ジャンプエッジが表現できない。そこで、提案手法は背景を考慮し、背景以外の画素に対してのみ隣接画素の深度値の差を計算することで問題を回避した。

具体的には、$d_{\mathbf{x}}$ を画素 $\mathbf{x} = (x, y)$ における深度値とすると、本研究で提案する平滑化項は以下の式で表される。

$$L_{smooth} = \sum_{\mathbf{x}} \alpha_{\mathbf{x}} \frac{\sum_{\mathbf{k} \in \mathcal{K}} ||d_{\mathbf{x}+\mathbf{k}} - d_{\mathbf{x}}||_1}{\sum_{\mathbf{k} \in \mathcal{K}} \alpha_{\mathbf{x}+\mathbf{k}}} \quad (1)$$

$$\mathcal{K} = \{(i, j) | i, j \in \{-1, 0, 1\}\} - \{(0, 0)\} \quad (2)$$

ここで、$\mathcal{K}$ は 8 近傍を表現するベクトルの集合である。α は、注目画素または隣接画素が背景である場合に平滑化損失の計算を行わないようにするための係数であり、当該画素の深度が背景深度 d_{max} に近ければ 0、そうでなければ 1 をとる。

$$\alpha_{\mathbf{x}} = \begin{cases} 1 & (d_{\mathbf{x}} \leq 0.9 d_{max}) \\ 0 & (\text{otherwise}) \end{cases} \quad (3)$$

上記の平滑化項の問題点として、深度チャネルの全ての画素を背景としたときの損失が 0 になることが挙げられる。つまり、平滑化項により出現する損失の新たな極小値の影響により、出力画像に枝が生成されない可能性がある。そこで、本研究では、Generator の最適化の前半は通常の画像変換の損失関数を用い、枝の RGB 値および深度値が概ね生成されるようになる最適化の後半に平滑化項を導入することで、出力深度の改善を図る。通常の画像変換で用いられる損失項と合わせ、提案手法で GAN の枠組みで最小化する Generator の損失関数 L_G は以下のように表せる。

$$L_G = \begin{cases} L_{L1} + L_{GAN} & (\text{ep} < \frac{\text{ep}_{\max}}{2}) \\ L_{L1} + L_{GAN} + \lambda L_{smooth} & (\text{ep} \geq \frac{\text{ep}_{\max}}{2}) \end{cases} \quad (4)$$

ここで、ep は現在のエポック数であり、$\text{ep}_{\max}$ は繰り返し最適化の全エポック数を示す。本研究における実験では、$\text{ep}_{\max} = 200$ とした。また、λ は平滑化項の重みを表す。L_{L1} および L_{GAN} は [2] で用いられた損失関数であり、それぞれ生成画像と真値画像の L1 ノルム、Discriminator の出力に基づく損失を表す。両損失関数の実装および、Discriminator の損失関数は Pix2Pix [2] と同一である。

3.3 三次元枝点群の生成

Generator の出力として得られた枝の RGBD 画像のうち、深度チャネルを三次元点群に変換することで、三次元枝点群を生成できる。RGBD 画像の撮影に用いた深度カメラの内部パラメータ（焦点距離 f、光学中心 (c_x, c_y)）を既知とし、非線形の歪みを無視すると、画素 $\mathbf{x} = (x, y)$ の深度値 d に対応する三次元点の位置 $\mathbf{p}$ は、透視投影モデルに基づき以下のように計算できる。

$$\mathbf{p} = (p_x, p_y, p_z) = (\frac{(x - c_x) d_w}{f}, \frac{(y - c_y) d_w}{f}, d_w) \quad (5)$$

$$d_w = \frac{d(d_{far} - d_{near})}{d_{max}} + d_{near} \quad (6)$$

ここで、を d_{near} および d_{far} は、RGBD 画像生成時の near clip 面および far clip 面を示す。Kinect 等の既製 RGBD カメラを用いる場合は、直接撮影された生の深度画像と同様に、内蔵されたキャリブレーションデータおよびライブラリ関数を用いて d_w から三次元点座標に変換可能である。

4 実験

提案手法を用いて得られた深度データの評価を行うため、CG 植物および実植物の RGBD 画像を入力として実験を行った。

4.1 学習条件

Pix2Pix に基づく画像変換の学習には、変換前後の画像ペア群が必要である。しかし、数多くの実植物を葉あり・葉なしの状態で同一位置から撮影することは現実的ではない。そこで、本実験では [1] と同様、CG シミュレーションで生成された植物の RGBD 画像ペア群を学習データとして用いた。本研究では、[6] の手法に基づき、同一の葉をもち枝構造の異なる CG 植物モデルを 10 種類生成した。各植物について、カメラの高さを三段階に変化させ、各々植物の周りから 15 度ずつ 24 枚レンダリングすることで、72 枚の画像を得た。学習データとして用いたのは $10 \times 72 = 720$ の画像ペアであり、各画像の解像度は、256×256 画素とした。CG 画像の作成においては、RGB 画像と深度画像の撮影位置および内部パラメータは同一とし、植物のみがシーンに含まれる、背景テクスチャのない画像を生成した。学習にかかる時間は平滑化項の有無にかかわらずほぼ同一であり、NVIDIA Quadro GP100 (メモリ 16GB) を用いた場合、200 エポックの学習におよそ 2 時間半かかった。

4.2 実験結果

CG 植物をテスト画像として行った生成画像例を図 3 に示す。ここでは、学習に用いた植物と同種（同一の葉を持つ）で異なる枝ぶりの植物モデルを、4.1 節と同じ方法で撮影した画像群を用いた。本研究で提案する平滑化項の有無とその重み λ の値を変化させ比較した。出力結果を三次元に投影した点群より、平滑化項

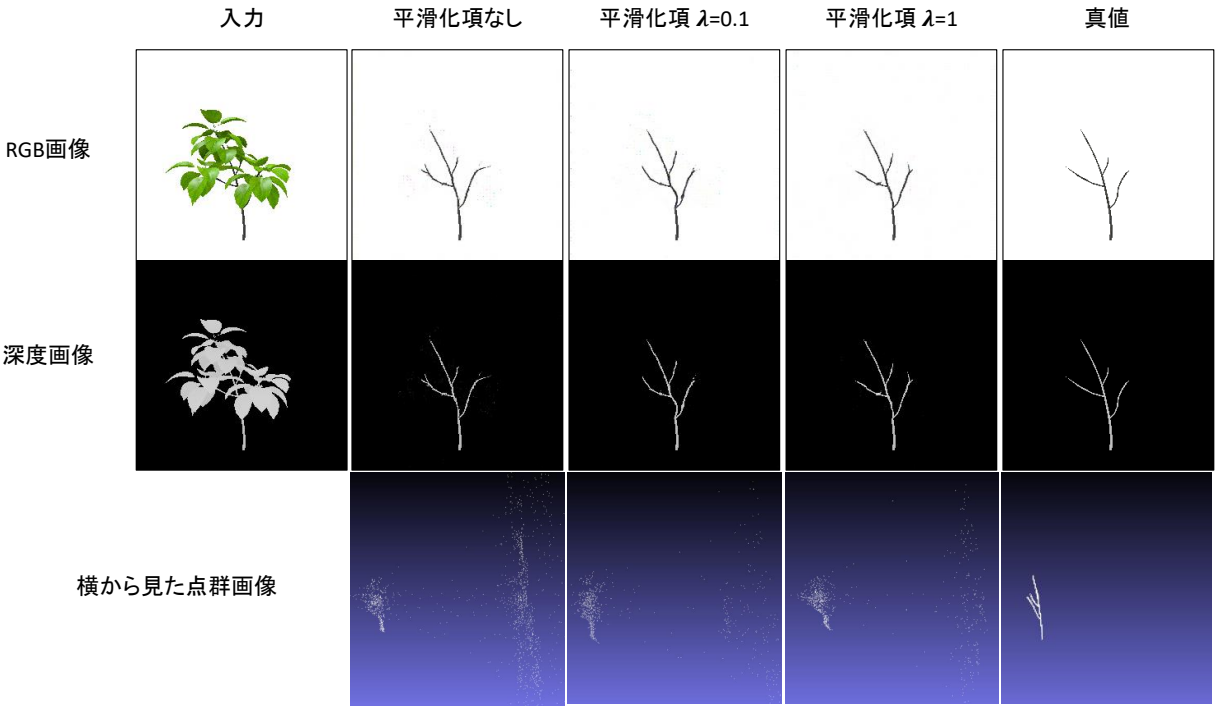


図 3　CG 植物を用いた実験結果

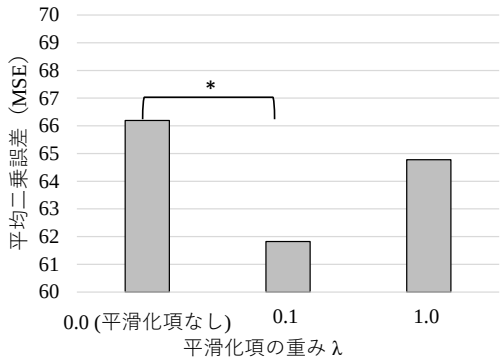


図 4　平滑化項の重み λ による深度画像誤差（MSE）の比較。'*' は対応あり Bonferroni 検定による有意差（有意水準 5%）を示す。

の効果は限定的ではあるものの、導入していない画像と比較すると特に背景・前景間の深度に誤推定される点が少なくなり、深度のばらつきが減っていることがわかる。また、50 枚の RGBD 画像（CG 植物）に本手法を適用した出力深度画像に対し計算された、真値との平均二乗誤差（Mean Squared Error: MSE）の平均を図 4 に示す。異なる平滑化項の重み λ を用いた場合、$\lambda = 0.1$ において、平滑化項を導入しない場合と比較し有意に深度誤差が改善された。

実画像を用いた実験結果を図 5 に示す。本実験では、Microsoft Kinect v2 を用い、植物模型の RGBD 画像を撮影した。観測深度値の安定化のため、連続した数フレームの平均深度を求め、センサに内蔵されたキャリブレーションデータに基づき、深度画像の各フレームに対応する RGB 値を取得することで RGBD 画像を生成した。また、撮影された深度画像の深度分布に基づき背景を削除した画像を提案手法に入力した。図 5 より、異なる形状・種類の CG 植物画像群を学習に用いたにもかかわらず、大まかな枝形状および深度が推定できたことがわかる。特に、平滑化項の導入により、幹（主茎）の部分が太く抽出された。一方、学習データとの植物種や撮影条件の相違により抽出されない、または誤って抽出された枝が見られた。実応用に向けて、学習データと異なる環境への対応や、さらなる高精度化が重要であることが示唆される。

5　考察

本稿では、ワンショット RGBD 撮影により、遮蔽部も含む枝の三次元形状を推定する手法を提案した。提案手法は、変換前後のドメインの画像ペア群を学習に用いる画像変換 Pix2Pix [2] を拡張し、RGBD 画像の葉付き画像から葉なし画像への変換を実現する。深度方向の推定安定性を向上するため、前景･背景を考慮した平滑化項を導入した。CG 植物画像、実画像の双方を用いた実験結果より、平滑化項は深度の安定性だけでなく、推定精度の向上にも寄与することがわかった。

一方、平滑化項の導入による精度向上効果は限定的であり、推定精度には未だ向上の余地がある。今後は、さらなる損失関数の導入や最適化手法の検討が必要である。特に、植物構造の復元を目的とした場合には、

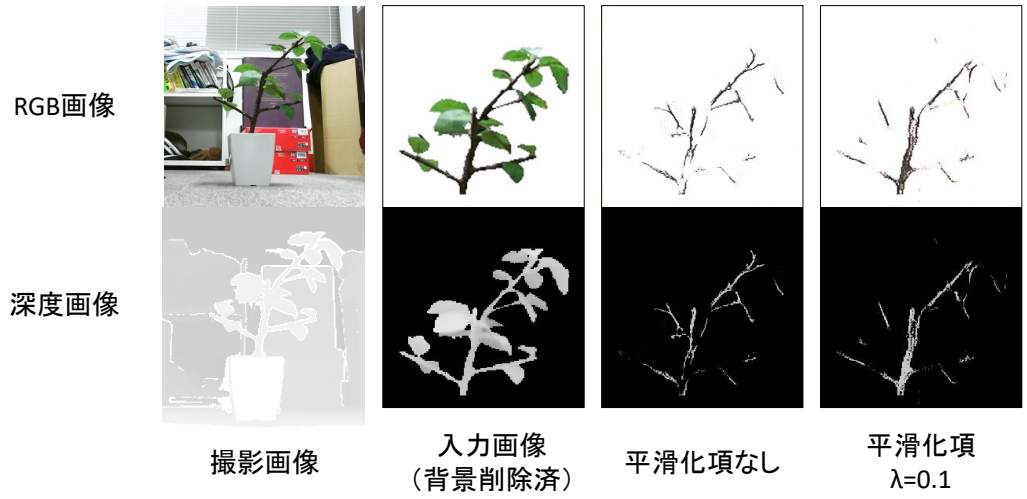


図 5　実植物を用いた実験結果：平滑化項の導入により、幹（主茎）の部分が太く抽出されている。

枝画像の復元だけでなく、枝の分岐点・端点位置の推定が有効であると考えられる。人物の関節位置検出による構造推定 [21] で用いられるような情報の同時推定により、枝構造推定の高精度化を図ることを検討している。

謝辞　本研究の一部は、JST さきがけ JPMJPR17O3 の支援を受けたものである。

参考文献

[1] T. Isokane, F. Okura, A. Ide, Y. Matsushita, and Y. Yagi: Probabilistic plant modeling via multi-view image-to-image translation; CVPR 2018.
[2] P. Isola, J.-Y. Zhu, T. Zhou, and A. A. Efros: Image-to-image translation with conditional adversarial network; CVPR 2017.
[3] K.W. Waite: Modelling natural branching structures; Computer Graphics Forum, 7(2): 105115 (1988).
[4] F. Boudon, P. Prusinkiewicz, P. Federl, C. Godin, and R. Karwowski: Interactive design of bonsai tree models; Computer Graphics Forum, 22(3): 591-599 (2003).
[5] M. Okabe, S. Owada, and T. Igarashi: Interactive design of botanical trees using freehand sketches and example-based editing; Computer Graphics Forum, 24(3): 487-496 (2005).
[6] W. Palubicki, K. Horel, S. Longay, A. Runions, B. Lane, R. Mech, and P. Prusinkiewicz: Self-organizing tree models for image synthesis; ACM Trans. on Graphics, 28(3): 58 (2009).
[7] A. Reche-Martinez, I. Martin, and G. Drettakis: Volumetric reconstruction and interactive rendering of trees from photographs; ACM Trans. on Graphics, 23(3): 720727 (2004).
[8] P. Tan, G. Zeng, J. Wang, S. B. Kang, and L. Quan: Image based tree modeling; ACM Trans. on Graphics, 26(3): 87 (2007).
[9] B. Neubert, T. Franken, and O. Deussen: Approximate image-based tree-modeling using particle flows; ACM Trans. on Graphics, 26(3): 88 (2007).
[10] L.D. Lopez, Y. Ding, and J. Yu: Modeling complex unfoliaged trees from a sparse set of images; Computer Graphics Forum, 29(7): 2075-2082 (2010).
[11] D. Zhang, N. Xie, S. Liang, and J. Jia: 3D tree skeletonization from multiple images based on PyrLK optical flow; Pattern Recognition Letters, 76(1): 49-58 (2016).
[12] Y. Li, X. Fan, N. J. Mitra, D. Chamovitz, D. Cohen-Or, and B. Chen: Analyzing growing plants from 4D point cloud data; ACM Transactions on Graphics, 32(6): 157 (2013).
[13] A.A. Efros, W.T. Freeman: Image quilting for texture synthesis and transfer; In Proc. SIGGRAPH'01, 341-346 (2001).
[14] A. Hertzmann, C.E. Jacobs, N. Oliver, B. Curless, and D.H. Salesin: Image analogies; In Proc. SIGGAPH'01, 327-340 (2001).
[15] C. Barnes, E. Shechtman, A. Finkelstein, and D. Goldman: PatchMatch: A randomized correspondence algorithm for structural image editing; ACM Trans. on Graphics, 28(3): 24 (2009).
[16] Y. Zhang, and T. Funkhouser: Deep depth completion of a single RGB-D image; CVPR 2018.
[17] O. Ronneberger, P. Fischer, and T. Brox: U-Net: Convolutional networks for biomedical image segmentation; Proc. Int'l Conf. on Medical Image Computing and Computer-Assisted Intervention (MICCAI'15).
[18] J. Johnson, A. Alahi, and L. Fei-Fei: Perceptual losses for real-time style transfer and super-resolution; ECCV 2016.
[19] J.-Y Zhu, T. Park, P. Isola, and A.A. Efros: Unpaired image-to-image translation using cycle-consistent adversarial network; ICCV 2017.
[20] A. Mahendran, and A. Vedaldi: Understanding deep image representations by inverting them; CVPR 2015.
[21] S. Wei, V. Ramakrishna, T. Kanade, and Y. Sheikh: Convolutional pose machines; CVPR 2016.