

## 最小切断面を利用した 2 値画像の意味的領域分割

成田 智史<sup>†</sup> 井 尻 敬<sup>†</sup>(正会員)<sup>†</sup> 芝浦工業大学

## Semantic Segmentation for Binarized Image using Minimal Cross Section

Tomofumi NARITA<sup>†</sup>, Takashi IJIRI<sup>†</sup> (*Member*)<sup>†</sup> Shibaura Institute of Technology

〈あらまし〉 本研究では、昆虫や植物などといった自然物の X 線 CT 画像に対する意味的領域分割アルゴリズムを提案する。提案法は、昆虫や植物はくびれを境に意味の異なる領域に分かれることが多いという知見に基づく。提案法は、二値化した X 線 CT 画像（ボクセルデータ）を入力とし、その前景領域を 2 分割する面積が局所最小な切断面（くびれ）の検出を繰り返すことで、意味的領域分割を実現する。このくびれの検出は、Split 処理、Regrow 処理、MinCut 処理という 3 ステップから構成され、Split 処理と Regrow 処理はモルフォロジー演算により実現され、MinCut 処理はグラフカット法により実現される。提案法の精度検証のため、二つの円を組み合わせた人工画像に対して提案法を適用し、提案法による分割結果と正しい分割結果との比較を行った。結果、誤差画素率 2% 未満という高い精度でくびれを検出できることを確認した。また、提案法の有用性を検証するため、昆虫や花のボクセルデータに提案法を適用し、花序内の小花、多肉植物の茎、昆虫の脚などという意味的に異なる領域を分割できることを確認した。

キーワード：意味的領域分割、くびれ、自然物 CT 画像、モルフォロジー処理、グラフカット法

<Summary> This paper presents a semantic segmentation technique for three dimensional (3D) X-ray Computed Tomography (CT) images of natural objects, such as insects or plants. Our technique is based on knowledge that joints of semantically different parts of natural objects are often narrow. Given a binarized 3D CT image, we recursively detect the narrowest cross section that divides the foreground region into two parts. Our narrowest cross section detection consists of three steps; (i) splitting the foreground by erosion operations, (ii) regrowing the split regions by dilation operation and (iii) finding the narrowest cross section in the dilated region by adopting a graph cut method. To evaluate the accuracy of our technique, we adopt it to artificially generated images and found that error pixel rate was less than 2%. To illustrate the feasibility of our technique, we adopt it to 3D CT images of insects and plants. As results, our technique successfully segmented multiple florets from an inflorescence, stems of a succulent plant, and legs of insects.

**Keywords:** semantic segmentation, narrow cross section, x-ray computed tomography images, morphological operations, graph cut algorithm

## 1. はじめに

意味的領域分割とは、画像や形状モデルを意味の異なる複数の部分に分割する処理であり、様々な場面で利用される<sup>1)</sup>。例えば、2 次元画像に対して意味的領域分割を適用すれば、各領域に対して異なる画像処理を施すことができる。また、3 次元画像や形状モデルに対して意味的領域分割を施せば、アニメーション用のリグ配置が容易になる。他にも、意味的領域分割には、生物（昆虫や植物など）の画像データから特定

領域の数や面積を計測するような、生物形状に関する自然科学研究への応用も存在する。

X 線 Computed Tomography (CT) を昆虫/植物のモデリング<sup>2)</sup>や分類学研究<sup>3)</sup>に用いる試みがなされている。これらの研究では、対象物を X 線 CT で撮影し、それを 2 値化することで形状モデルを取得する。もし、これらの研究で得られる自然物の 2 値化画像をさらに意味的に異なる領域に分割できれば、分割された各部位をその後のモデリングや形状/個数/体積などの取得に利用できる。前述のとおり、意味的

領域分割は重要な課題であり、写真<sup>4)</sup>や形状モデル<sup>5)</sup>のための手法が多く発表されている。一方、3次元画像のための手法、特に昆虫や植物のような自然物 CT 画像のための意味的領域分割に関する研究はあまりなされていない。

本研究では、昆虫や植物などの自然物を X 線 CT により撮影し 2 値化を施した画像（ボクセルデータ）を意味的に領域分割する手法を提案する。提案法の鍵となるアイデアは、『昆虫や花は「くびれ」を境に異なる意味を持つ領域に分かれることが多い』という知見を利用することである。具体的には、我々は、ボクセルデータ内のある連結領域に対して、その領域の最小のくびれ（定義は後述）を検出し、そのくびれにより領域を 2 分割する。この連結領域における最小くびれの検出は、収縮処理を繰り返すことで領域全体を大まかに分断する『Split 処理』、分断された領域を再度膨張させる『Regrow 処理』、Regrow 処理により得られた領域内で実際の最小断面を検出する『MinCut 処理』により実現される。提案法では、このくびれ検出による領域の 2 分割処理を再帰的に繰り返すことでボクセルデータ全体からすべてのくびれを抽出する。

提案法の精度を評価するため、くびれが既知である人工画像に対し提案法を適用し、提案法により計算された切断面と正解との比較を行った。結果、提案法は正解と近い断面を生成できることを確認した。また提案法の有用性を確認するため、昆虫や植物のボクセルデータに対して意味的領域分割を行った。タマムシと多肉植物の分割結果を図 1 に示す。タマムシは脚や触覚が分割され、多肉植物は分枝している茎が分割されており、意味的に異なる部位が分割できていることが確認できる。

## 2. 関連研究

画像や形状モデルの意味的領域分割は重要な課題であり、多くの研究がなされている。既存の意味的領域分割手法は、あらかじめ定義した特徴量や規則を利用するもの<sup>5)6)7)8)</sup>と、正解データ群から分割法を訓練する機械学習を用いるもの<sup>4)9)</sup>に大まかに分類できる。

あらかじめ定義した特徴量を用いる手法には、例えば各三角形間の測地距離を利用するもの<sup>5)</sup>や、各面に対して計算した曲率を利用するもの<sup>6)</sup>がある。しかし、これらの既存手法は、その対象データが 3 次元ポリゴンメッシュデータに限定されている。また、Xin ら<sup>7)</sup>は、メッシュ上の測地閉曲線 (Geodesic Loop) を求める手法を提案した。この手法は、ユーザの与えた初期解の更新を逐次的に繰り返す。この手法ではメッシュ上のくびれが検出されるので、検出したくびれを利用してメッシュを分割することで意味的領域分割を実現できる。しかし、この手法では、初期解を入力する必要があるため、多くのくびれを持つ対象 (図 10gh や図 11) には適用しにくい。

本研究の目的はボクセルデータ (3 次元 2 値化画像) の意

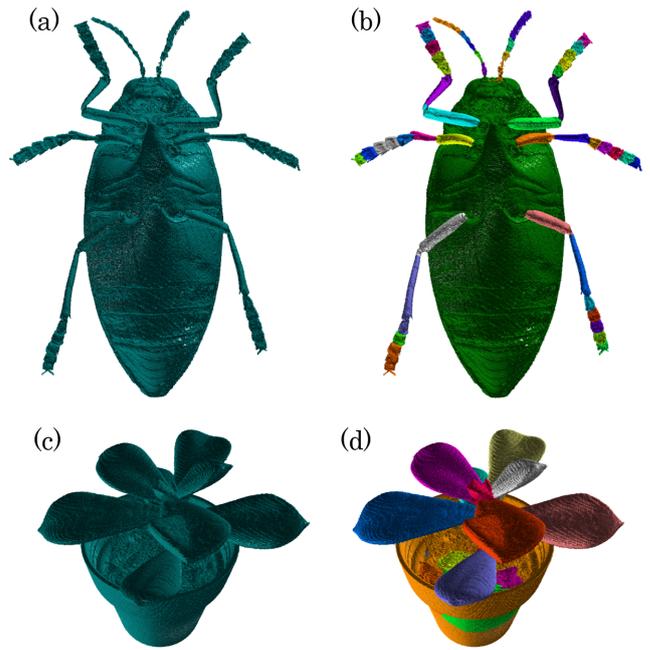


図 1: 提案手法による意味的領域分割結果: (a) タマムシのボクセルデータ, (b) その分割結果, (c) 多肉植物のボクセルデータ, (d) その分割結果.

**Fig. 1** Segmentation results with our technique: (a) input voxel data of an insect, (b) its segmentation result, (c) input voxel data of a plant, and (d) its segmentation result.

味的領域分割であり、ボクセルデータを Marching Cubes 法により表面メッシュ化することで、前述の表面メッシュに対する既存手法を適用することは可能である。しかし、本研究が対象とする自然物のボクセルデータは、前景領域内部の中空領域を含むことが多く、このようなボクセルデータをメッシュ化した場合、多くの孤立した表面メッシュが生成される。既存の表面メッシュに対する意味的領域分割手法<sup>5)6)7)</sup>は、連結した一枚の表面メッシュを想定しているものがほとんどであり、中空領域を含む複雑なボクセルデータには適用しにくい。一方、提案法は、表面メッシュでなく内部の詰まったボクセルデータを直接処理するため、中空領域を持つデータに対しても堅固な意味的領域分割が可能である。

あらかじめ定義した規則を利用する手法のひとつで、本研究と最も関連が深いものが、Schmitt ら<sup>8)</sup>により提案されている。この手法は、細胞組織の 2 次元 CT 画像を二値化した画像から各細胞領域を分割するものであり、前景領域に対

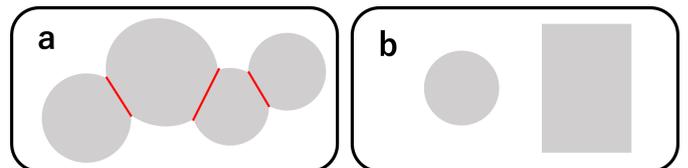


図 2: (a) 複数のくびれを持つ図形と (b) くびれを持たない図形の例 (赤線部がくびれ)。

**Fig. 2** Shapes (a) with and (b) without multiple narrowest cross sections.

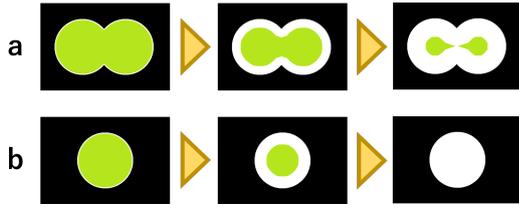


図 3: Split 処理. (a) くびれを持つ領域と (b) くびれを持たない領域.

Fig. 3 Split operation for shapes (a) with and (b) without narrowest cross sections.

して収縮処理を繰り返し施すことで各細胞におけるシード領域を抽出し、各シード領域に対して膨張処理を繰り返すことで各細胞領域の境界を発見する。しかし、この手法は単純な膨張処理のみで領域の境界を決めるため、その境界は面積が局所最小になるとは限らない。また、この手法は異なる複数の半径で収縮処理を施した結果を保持するアルゴリズムであるため、3次元画像への適用には多くのメモリ容量が必要となる。

一方、機械学習を用いる手法では、正解データを利用し、画像<sup>4)</sup>や形状モデル<sup>9)</sup>の分割手法を学習する。しかし、これらの手法は一般的に大量の正解データを必要とし、多くのデータを用意することが困難な自然物 CT 画像には不向きである。

### 3. 提案手法

本研究ではくびれを、領域を2分割する曲面(2次元ならば曲線)の中でその面積が局所的に最小となるもの、と定義する。図 2a に複数のくびれを持つ領域の例を示す。この図において、赤線は領域を2分割し、赤線をどのように微小変化させてもその面積が増加する。そのため、この赤線はくびれとなる。一方、図 2b の図形には、領域を2分割する断面は存在するが、その面積が局所最小となるものは存在せず、これらの図形にはくびれが存在しない。

提案法は、入力されたボクセルデータ(3次元2値画像)の連結領域から最も面積の小さなくびれを検出し、これを境界として連結領域を2分割する。提案法は、(i) 連結領域を大まかに分断する Split 処理、(ii) Split 処理で分断された領域を膨張する Regrow 処理、(iii) Regrow 処理の結果を用いて連結領域における面積最小の切断面(くびれ)を検索する MinCut 処理の3処理を行い、連結領域から面積最小のくびれを検出する。このくびれ検出処理を、2分割された各領域に再帰的に適用することですべてのくびれを検出し、分割された領域に異なるラベル値を付加した画像を出力する。各処理の詳細を紹介する。

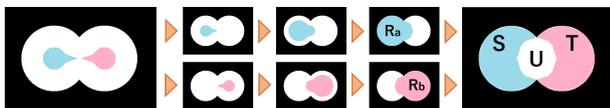


図 4: Regrow 処理.

Fig. 4 Regrow operation.

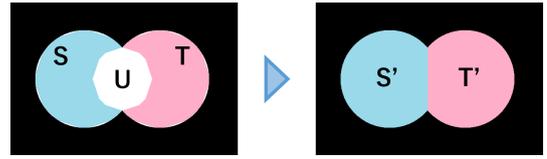


図 5: MinCut 処理の入出力.

Fig. 5 MinCut operation.

Split 処理では、対象領域が複数の連結領域に分断されるか、消滅するまで収縮処理を繰り返す。ここで収縮処理とは、画像内の各画素に対して、その画素値を近傍領域の最小値に更新する処理である。本研究では、半径2画素の球を近傍領域として利用した。対象領域がくびれを持つ場合、この Split 処理により、連結領域は2つ以上の領域に分割される(図 3a)。一方、対象領域がくびれを持たない場合は、複数領域に分割されることなく領域が消滅する(図 3b)。領域が消滅する場合は Regrow 処理と MinCut 処理を行わず、対象領域がくびれを持たないものとして分割処理を終える。

Regrow 処理では、Split 処理で分断された連結領域に膨張処理を施す。ここで膨張処理とは、画像内の各画素に対して、その画素値を近傍領域の最大値に更新する処理であり、本研究では、半径2画素の球を近傍領域として利用した。具体的には、Split 処理にて分断された連結領域のうち、画素数が最も多い2つの連結領域のみを取り出し、これらに対して Split 処理で収縮処理を施したのと同じ回数の膨張処理を適用する(図 4)。本研究では、これらの膨張させた2領域を  $R_a, R_b$  と置き、また、 $S = R_a \cap \bar{R}_b$ ,  $T = \bar{R}_a \cap R_b$ ,  $U = A - S \cup T$  と表記する(図 4)。

MinCut 処理では、Split 処理・Regrow 処理にて生成された領域を利用し、対象領域を2分割する最小切断面を検索する。具体的には、領域全体  $\Omega = S \cup T \cup U$  を、 $S$  と  $T$  をそれぞれ含む2領域  $S', T'$  に分割する面積最小の切断面を領域  $U$  内において検索する(図 5)。

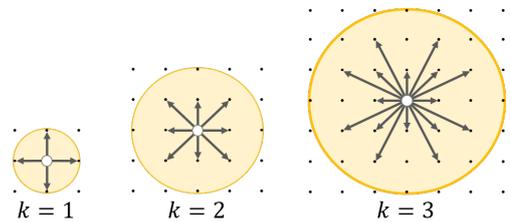


図 6: エッジ配置のための近傍画素.

Fig. 6 Neighboring pixels for edge construction.

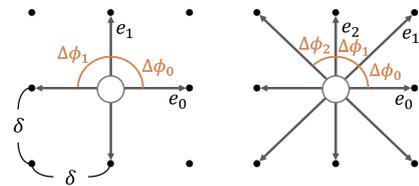


図 7: エッジ容量に関する変数.

Fig. 7 Variables for edge capacities.

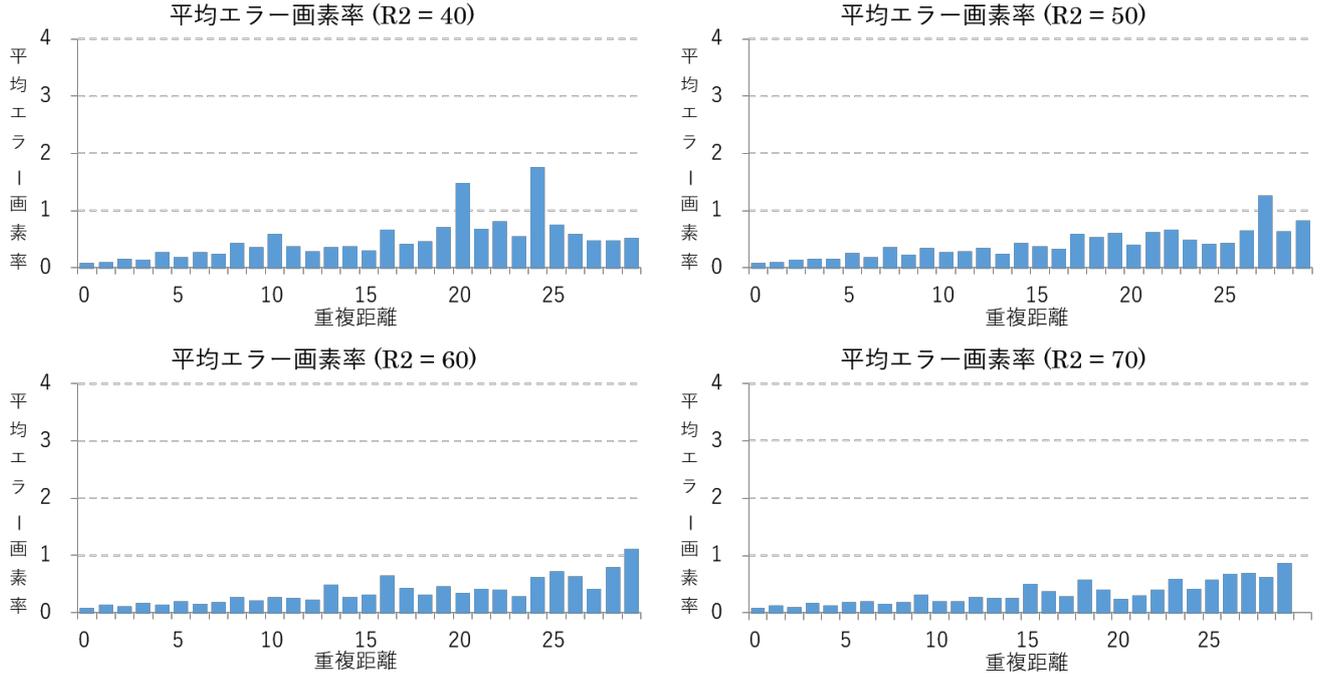


図 8 異なる半径 ( $R2 = 40, 50, 60, 70$ ), 及び, 異なる重複距離 (0 - 29 ピクセル) に対する平均エラー画素率.

Fig. 8 Average error pixel rate for different conditions: radiuses( $R2 = 40, 50, 60, 70$ ) and overlap distances(0 - 29 pixels).

この切断面の計算にはグラフカット法を応用する. まず, 我々は, 始点ノード  $n_s$ , 終点ノード  $n_t$ ,  $\Omega$  内の全画素ノード  $n_p$  より構成されるグラフを構築する. また, 始点ノード  $n_s$  と各画素ノード  $n_p$  の間, 及び, 終点ノード  $n_t$  と各画素ノード  $n_p$  間に以下の容量 (式 (2)) を持つエッジを配置する,

$$capa(n_s, n_p) = \begin{cases} \infty & (n_p \in S) \\ 0 & (n_p \in T), \\ 0 & (n_p \in U) \end{cases} \quad (1)$$

$$capa(n_p, n_t) = \begin{cases} 0 & (n_p \in S) \\ \infty & (n_p \in T). \\ 0 & (n_p \in U) \end{cases} \quad (2)$$

また, 近傍画素間 ( $n_p, n_q$ ) 間にもエッジを配置する. このとき, 半径  $k$  画素の球内における各方向の最近傍画素を近傍として用いる (図 6). この  $k$  の値を大きく設定すると幾何学的に正しい境界を生成できる<sup>10)</sup>が, 計算時間が増大してしまう. 本研究では, 2次元画像については  $k = 3$ , 3次元画像については  $k = 2$  を利用する. 各エッジの容量は, Boykovら<sup>10)</sup>の提案した以下の式 (4) を利用する,

$$capa_{2D}(n_p, n_q) = \frac{\delta^2 \Delta \phi_i}{2|e_i|}, \quad (3)$$

$$capa_{3D}(n_p, n_q) = \frac{\delta^3 \Delta \psi_i \Delta \varphi_i}{\pi|e_i|}. \quad (4)$$

ここで,  $\delta$  は画素のピッチ,  $|e_i|$  はエッジ  $e_i$  の長さ,  $\Delta \phi_i$  は  $e_i$  と  $e_{i+1}$  のなす角度である (図 7). 3次元画像を対象とする場合は式 (4) のとおり, 仰俯角  $\Delta \psi_i$  と方位角  $\Delta \varphi_i$  を利用して, 容量を定義する.

このように定義したグラフを,  $n_s$  と  $n_t$  をそれぞれ含む2つの部分グラフにカットする際のカット容量を考える. すると, このカット容量は, 領域全体を2領域に分割する際の切断面の面積となる<sup>10)</sup>. つまり, このグラフの最小カットを計算することで, 領域全体を2分割する面積最小切断面が取得できる. この最小カット計算には既存手法のグラフカットアルゴリズムを利用する<sup>11)</sup>.

## 4. 結果と考察

### 4.1 精度の評価

提案法の精度を評価するため, 正解が既知である2次元画像データ (図 9a) を人工的に作成し, 提案法の出力結果 (図 9b) と正解 (図 9c) を比較した. 入力画像は2つの円が交差したものであり, 円の半径や2つの円の重なる距離, 回転角度を様々に変化させて生成した. 具体的には, 片方の円の半径  $R1$  を40に固定し, もう1つの円の半径  $R2$  を10刻みで40から70まで変化させた4パターン, 重複距離を0から29まで変化させた30パターン, 各画像を30度ずつ回転させた12パターンの計1440枚の画像を作成した. また, 各画像において, 円内部を前景とし, 2つの円の交点を結ぶ直線

をくびれとして正解画像を作成した。

図 8 に、円の重複距離に対する平均エラー画素率を示す。ここで、エラー画素とは正解データと異なる領域に分割された画素を指し、エラー画素率とは前景領域の全画素数に対するエラー画素数の割合である。この図より、平均エラー画素率は 2% 未満であり、角度によらず十分高い精度で分割できていることがわかる。

#### 4.2 自然物データへの適用例

提案法の有用性を示すため、昆虫や植物のボクセルデータに対して提案法を適用し、結果を観察した。図 1(a, b) と図 10(a-f) に昆虫ボクセルデータの分割結果を示す。分割結果を観察すると、昆虫の頭部・胸部・腹部・脚・触角といった部分を分割できていることが見て取れる。しかし、一方で、領域を必要以上に細かく分割してしまうオーバーセグメンテーションが観察された。これは、ボクセルデータ上の細かな凹凸をくびれとして検出した結果である。このオーバーセグメンテーションの手作業による修正は大きな手間ではないが、この解決は今後の課題である。

次に、図 1(c, d) と図 10(g, h) に植物ボクセルデータに提案法を適用した結果を示す。多肉植物の分割結果 (図 1(c, d)) からは、中心から分岐する各茎が分割できていることが観察でき、ムスカリの花序への適用結果 (図 10(g, h)) からは、密集した小花が分割できていることがわかる。このように、分割対象が密集している場合や数が多い場合、手動による分割は手間がかかるが、提案法によりその自動分割が可能になる。このような意味的分割を施すことで、開花した花数、小花の大きさ、小花の開き具合など、自然科学研究において有用な計測を自動的に行えると考えられる。一方、昆虫への適用例と同様に多少のオーバーセグメンテーションも観察された。

さらに、提案法は、昆虫ボクセルデータの内空領域の分割にも利用できる。ここで内空領域とは、昆虫内部の空気領域である。通常、昆虫標本の CT 画像を閾値により二値化すると、外骨格部分のみが前景として抽出され、内部の空気領域は背景となる。この内部の空気領域は、外骨格の隙間により

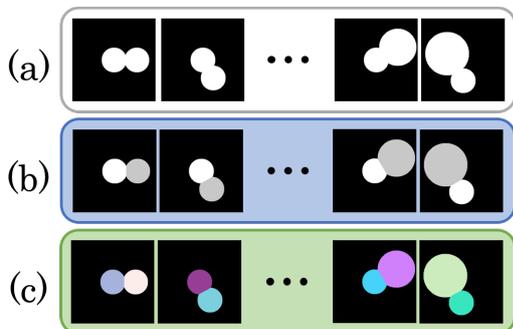


図 9: 評価実験で用いた画像: (a) 入力画像, (b) 分割結果, (c) 正解。

Fig. 9 Images for accuracy evaluation: (a) input images, (b) segmentation results, and (c) ground truth.

外の空間のつながっていることが多く、その分割は容易ではない。図 11 に、この内空領域の分割に提案法を適用した結果を示す。図 11(a-c, g-i) において、白色画素が前景領域を、緑色画素が背景領域を示す。このとおり、背景領域は、内空領域と外部の領域を含んでいる。この緑領域に提案法を適用すると、図 11(d-f, j-l) が得られた。細かな複数の領域に分かれてはいるものの、背景領域が、内空領域と外部領域に正しく分割されていることがわかる。このように内空領域を分割できると、昆虫ボクセルデータの内空領域を埋めることが可能となり、昆虫 CT 画像から不要なポリゴンを削減したメッシュモデルを作成する際に役立つ。

提案法の計算時間を表 1 に示す。計算時間の計測には Windows10, Intel(R) Core(TM) i7-7700K 4.2GHz, RAM64GB を用いた。適用したデータの多くは 20 分以内で分割可能であったが、極端に画素数の多い領域に対しては、7 時間弱の計算時間を要した。これは、画素数の多い領域ほど、Split/Regrow 処理で収縮/膨張処理を繰り返す回数が増え、また MinCut 処理でくびれを検索する領域  $U$  が大きくなるためである。

表 1: 各自然物 CT 画像における分割時間。

Table 1 Timing for segmenting each CT volume.

名前	画像解像度	対象領域画素数	分割数	分割時間(分)
蟻 (図10 ef)	681×681×339	1.5M	24	8
クワガタ (図10 cd)	700×700×328	3.8M	25	12
ムスカリ (図10 gh)	401×402×783	5.4M	82	13
ゾウムシ (図10 ab)	746×811×351	7.1M	30	19
サボテン (図 1 cd)	497×439×483	12.2M	24	13
タマムシ (図 1 ab)	766×927×361	728.4M	47	414

## 5. まとめ

昆虫や花などの自然物がくびれを境界として意味的に異なる領域に分かれることに着目し、ボクセルデータをくびれにより意味的に分割する手法を提案した。提案法は、縮小処理・膨張処理、及び、グラフカット法を組み合わせる事で、2 値化画像の連結領域からそれを切断する最小断面を検出する。提案法の精度を評価するため、正解が既知である人工画像に提案法を適用し、提案法による分割結果と正解を比較した。その結果、高い精度で意味的領域分割を行えることを確認した。また、提案法の有用性を示すため、昆虫や植物の CT 画像に提案法を適用し、昆虫の各部位や花と茎などが正しく分割されることを確認した。

実際のボクセルデータに提案法を適用した場合、提案法は、小さなくびれを検出し、領域を必要以上に細かく分割してしまうオーバーセグメンテーションを引き起こしやすいことが確認されている。また、葉のような薄い領域は付け根がくびれていても、提案法ではこれを検出することができない。これは、Split 処理における収縮処理によってくびれが分断さ

れると同時に領域全体が消滅するためである。Split 処理では、半径 2 ボクセルの球を収縮処理の近傍領域としており、これにより全体が消失してしまうような薄い領域や細い領域は正しく扱うことができない。さらに、前景領域にくびれがなければ分割することができないため、例えば体内の臓器や地中の根といった明確なくびれを持たない領域は分割することが困難である。これらの解決は今後の課題である。現在の実装では、計算時間が長いという課題もある。そのため、計算の高速化も実用化のために重要である。

## 6. 謝 辞

本研究は JSPS 科研費 (15H05924, 17H01848) の助成を受けたものである。本論文の掲載にあたり、重要なお指摘をいただいた査読者の方に感謝いたします。

## 参考文献

- 1) H. Zhu, F. Meng, J. Cai, S. Lu: "Beyond pixels: A comprehensive survey from bottom-up to semantic image segmentation and cosegmentation.", *Journal of Visual Communication and Image Representation*, Vol. 34, pp. 12–27(2016).
- 2) T. Ijiri, S. Yoshizawa, H. Yokota, T. Igarashi: "Flower modeling via x-ray computed tomography.", *ACM Trans. on Graphics*, Vol. 33, No. 4, pp. 48(2014).
- 3) S. Faulwetter, A. Vasileiadou, M. Kouratoras, T. Dailianis, C. Arvanitidis: "Micro-computed tomography: Introducing new dimensions to taxonomy.", *ZooKeys*, No. 263, pp. 1(2013).
- 4) J. Long, E. Shelhamer, T. Darrell: "Fully convolutional networks for semantic segmentation.", *IEEE conference on computer vision and pattern recognition*, pp. 3431–3440(2015).
- 5) S. Katz, A. Tal: "Hierarchical mesh decomposition using fuzzy clustering and cuts.", *ACM Transactions on Graphics*, Vol. 22, No. 3, pp. 954–961(2003).
- 6) A. P. Mangan, R. T. Whitaker: "Partitioning 3D surface meshes using watershed segmentation.", *IEEE Trans. on Visualization and Computer Graphics*, Vol. 5, No. 4, pp. 308–321(1999).
- 7) S. Q. Xin, Y. He, C. Fu: "Efficiently computing exact geodesic loops within finite steps.", *IEEE Trans. on Visualization and Computer Graphics*, Vol. 18, No. 6, pp. 879–889(2012).
- 8) O. Schmitt, M. Hasse: "Morphological multiscale decomposition of connected regions with emphasis on cell clusters.", *Computer Vision and Image Understanding*, Vol. 113, No. 2, pp. 188–201(2009).
- 9) E. Kalogerakis, A. Hertzmann, K. Singh: "Learning 3D mesh segmentation and labeling.", *ACM Trans. on Graphics*, Vol. 29, No. 4, pp. 102(2010).
- 10) Y. Boykov, V. Kolmogorov: "Computing geodesics and minimal surfaces via graph cuts.", *IEEE International Conference on Computer Vision*, pp. 26(2003).
- 11) Y. Boykov, O. Veksler, R. Zabih: "Fast approximate energy minimization via graph cuts.", *IEEE Trans. on pattern analysis and machine intelligence*, Vol. 23, No. 11, pp. 1222–1239(2001).

成 田 智 史

2018 年 芝浦工業大学工学部情報工学科卒。現在、芝浦工業大学大学院理工学研究科電気電子情報工学専攻在学。

井 尻 敬 (正会員)

2004 年 東京工業大学情報科学科卒業、2009 年 東京大学情報理工学系研究科コンピュータ科学専攻終了、博士 (情報理工学)。2009-2012 年 理化学研究所研究員、2012-2015 年 理化学研究所基礎科学特別研究員、2015-2017 年 立命館大学情報理工学部講師、を経て、2017 年より芝浦工業大学工学部准教授。

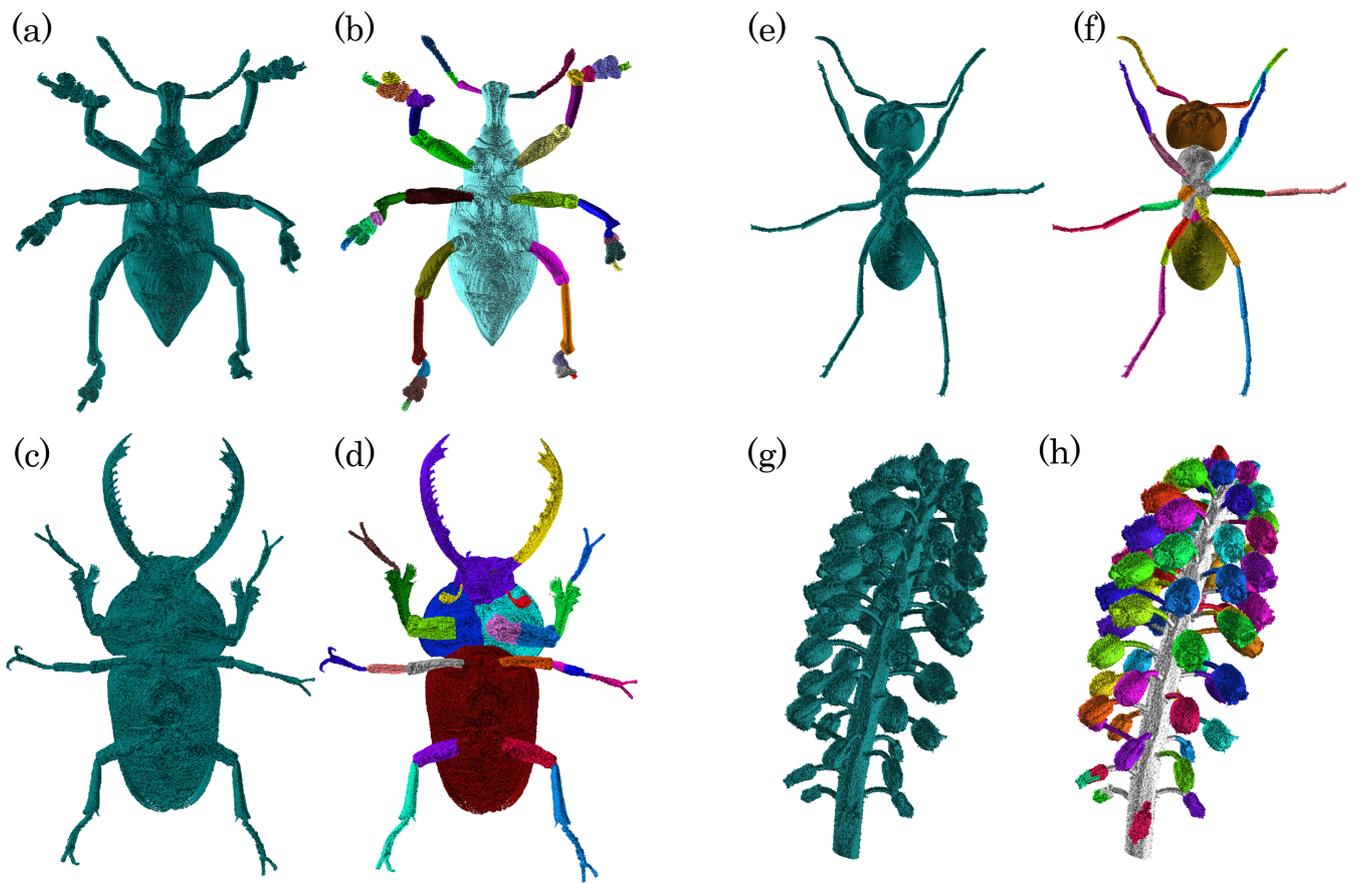


図 10 昆虫と植物のボクセルデータに対して提案法を適用した結果: (a, b) ゾウムシ, (c, d) クワガタ, (e, f) 蟻, (g, h) ムスカリ.

Fig. 10 Segmentation results for insect and plant CT volumes: (a, b) Weevil, (c, d) Stag beetle, (e, f) Ant, and (g, h) Muscari.

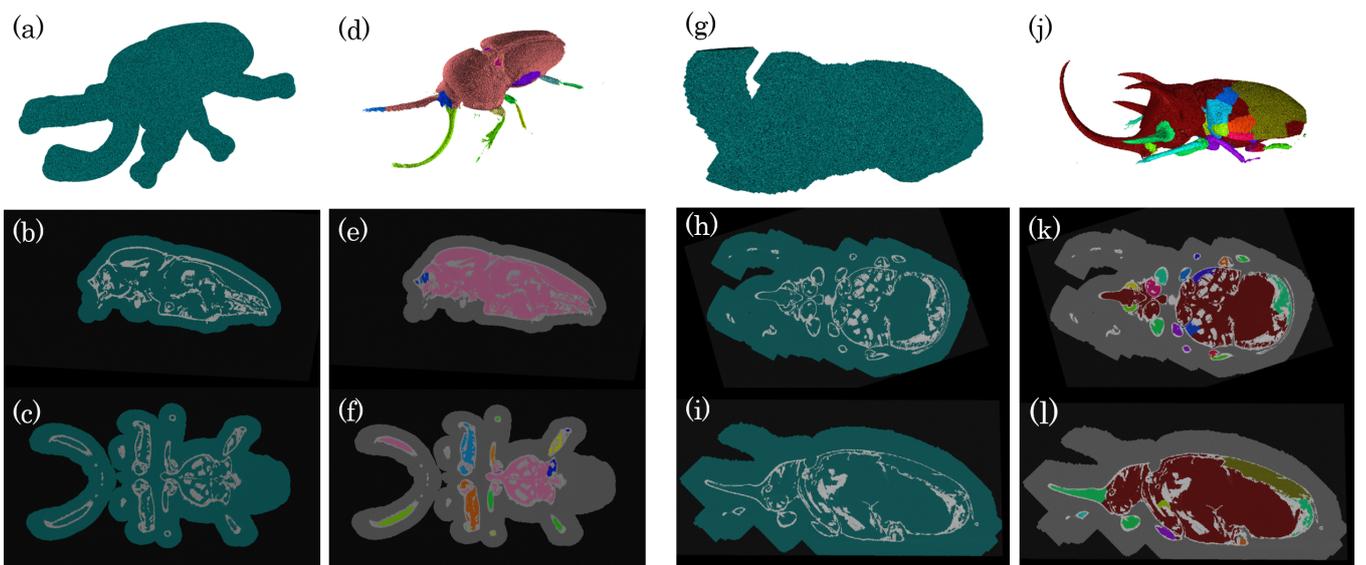


図 11 昆虫の背景領域に対して提案法を適用した結果: (a-c) クワガタ画像の背景領域, (d-f) その分割結果, (g-i) カブトムシ画像の背景領域, (j-l) その分割結果.

Fig. 11 Segmentation results for backgrounds of insect CT volumes: (a-c) background region of stag beetle volume, (d-f) its segmentation result, (g-i) background region of beetle volume, and (j-l) its segmentation result.