

Superpixel に基づくグラフベースの画像分割における 境界形状項の導入

久保友輔^{†1,a)} 和田俊和^{†1,b)}

概要 : Semantic segmentation を行う DNN をトレーニングするためには大量の領域分割画像が必要になる。この画像を手で作成する際に、あらかじめ物体の境界にまたがらない小領域に分割する GUI が利用できれば、作業量を大幅に減らすことができる。本研究はこのような応用のために、領域分割の精度向上を目指す。領域分割の一手法として、隣接する Superpixel を初期値とする領域を頂点とし、その内部の色分布の類似性を辺の重みとしたグラフに基づいて頂点の統合を反復する手法がある。この手法は動画の領域分割にも容易に拡張できる。しかし、色分布間のみを用いた手法では、照明などの影響による画素値の変化によって、統合が阻害されることがある。このように本来統合されるべきであった領域間の境界形状は複雑になることが多いため、本研究では領域間の色分布の類似性に加えて、領域間の境界形状の複雑さも統合の起こりやすさの基準としてグラフの辺の重み付けをする方法を提案する。BSDS300 データセットを用いた実験を通じて、本手法により精度が向上することを確認した。

キーワード : Superpixel, 画像分割, 境界形状の複雑さ, グラフ構造

Boundary complexity term for graph-based image segmentation

YUSUKE KUBO^{†1,a)} TOSHIKAZU WADA^{†1,b)}

Abstract: Ground-truth data for image segmentation is highly demanded for learning semantic-segmentation network. Manual segmentation and labelling is time consuming task and the results totally depend on workers. For helping this task, we are planning to develop a GUI system and propose a method to improve the segmentation accuracy for this system. In this system, we employ graph-based image segmentation algorithm, because it can easily be extended to consistently segmenting image sequences. The vertices correspond to image regions and the edge is weighted by similarity between color distributions of the adjacent regions. The initial image regions can be superpixels for reducing computational time. Only by using color similarity as the edge weight, however, sometimes fragment regions are left. In such cases, boundary shape between regions which should be merged tend to be complex. This paper introduces a method to weight edges based on boundary shape complexity in addition to color-distribution similarity. We confirmed the effectiveness of our method through experiment on BSDS300 dataset.

Keywords: Superpixel, Image Segmentation, Boundary Complexity, Graph Structure

1. はじめに

画像の領域分割とは、画像を意味的にまとまりのある複数の領域に分割することであり [1], それらの領域は現実世界の物体と対応づいた領域となっていることが望ましい [2]. 本研究では、このような領域をオブジェクトと呼ぶ。領域分割のアルゴリズムは物体認識や画像編集などを目的として、古くから研究されてきた。領域分割を行うことによって、様々な処理を画像内のオブジェクト単位で行うことができ、物体の検出および判別の詳細化に応用することができる。

特に最近では、車の自動運転や視覚監視などの応用を目指して DNN を用いた画像の Semantic Segmentation の研究が盛んに行われているが、学習には画像内の各オブジェクトが占める領域に領域 ID が割り当てられた正解画像が大量に必要となる。この正解画像は描画ツールなどを用いて人手で作成されており、大変な手間がかかるだけでなく、作業によって異なる正解画像が作成されてしまうとい

う問題がある。

この問題を緩和する一つの手法として、対話的な領域分割ツールを用いることが考えられる。このツールによって、予め画像がオブジェクトあるいはオブジェクトを構成する部分単位で領域分割されていれば、これらを単位として領域 ID を割り当てることができ、作業量を大幅に減らすと同時に、作業による結果のばらつきも抑えることができる。

本研究では、以上のように Semantic Segmentation の学習に用いられる正解画像の作成補助に応用することを想定し、人間の一般的な知覚に近い領域分割の実現を目指す。

画像の領域分割の手法は、局所探索型の手法、グラフ構造を用いた手法など様々な手法が提案されている。

局所探索型の手法は画像内の適当な出発点から、画素の色に基づいて、局所的に色分布が異なる領域を形成していく手法であり、膨張型・分割型の領域分割などがある。一般に、局所探索型の手法は初期値依存性が高いという欠点がある。

これに対して、グラフ構造を用いた手法は画像をグラフ

で表現する．このグラフは、画像を Superpixel と呼ばれる過分割領域を頂点とし、隣接する領域を辺で結んで構築されるグラフである．そのグラフを辺の重みに基づいて複数の部分グラフに分割することで領域分割を行う．具体的には、ある頂点に隣接していて、なおかつ色などの属性が類似している頂点を同じ領域とするという処理を再帰的に行う．そのため、色の変化が急な境界で、領域を分割することができる．また、局所探索型的手法と比べて計算量が少ないという利点もある．

グラフを用いた領域分割の先行研究 [3]では、Superpixel と呼ばれる過分割画像の領域を頂点とするグラフで画像を表現し、各領域内の色分布の類似性を用いて辺の重みを計算している．このグラフに基づいて、領域を統合し、色の分布が一樣な領域を形成する．しかし、色分布のみで辺の重み付けをすると、照明変化などで色が変化してしまった領域は統合されずに領域が断片化されるという問題点がある [3]．本来は統合されるべき領域が異なる領域に分割されてしまった場合について詳しく調べると、そのような領域間の境界形状は多くが不安定で複雑になることが分かる．

本研究では、このような領域境界形状の性質に着目し、隣接する領域間の色分布の類似性に加えて、境界形状の複雑さも用いてグラフの辺の重み付けをすることで、領域の断片化を軽減し、分割精度の向上を実現する．

2. 関連研究

2.1 Graph-Based アルゴリズム

P.Felzenszwalb ら [4]による Graph-Based アルゴリズムは、領域間の相違度に基づいて領域統合の判定を繰り返すアルゴリズムであるため、比較的少ない計算量で領域分割を行うことができる．

Graph-Based アルゴリズムは画素を頂点とし、隣接する画素の属性間の相違度を辺の重みとして持つグラフ構造に対し、相違度を介して隣接した画素を領域に統合していき、領域分割を行う手法である．ある領域 R 内部の相違度 $\text{Int}(R)$ を以下のように定義する．ただし、 $\text{MST}(R)$ は領域 R 内の最小全域木(Minimum Spanning Tree)、 $w(e)$ は辺 e の重み(相違度)である．

$$\text{Int}(R) = \max_{e \in \text{MST}(R)} w(e) \quad (1)$$

また、 R の画素数 $|R|$ が1のときは $\text{Int}(R) = 0$ である．以下では、Graph-Based アルゴリズムの具体的な計算手順を説明する．

はじめ1つの頂点(画素)を含む領域が画素数だけある状態で初期化し、各辺の重みについて昇順に、その両端の頂点が属している領域を統合するか否かを評価する．具体

的には、2つの頂点 u, v を結ぶ辺を $e_{u,v}$ とし、 u の属する領域を $R(u)$ としたとき、以下の不等式で評価を行う．

$$w(e_{u,v}) \leq \min(\text{Int}(R(u)), \text{Int}(R(v))) \quad (2)$$

式(2)を満たしていれば $R(u)$ と $R(v)$ は統合され、領域内部の相違度が更新される．しかし、この相違度の定義では、初期状態においてすべての領域内部の相違度が0となってしまう、はじめは重みが0の辺しか統合が起きなくなってしまう．これを軽減するために領域 R 内部の相違度を $R\text{Int}(R)$ として以下の式で再定義する．

$$R\text{Int}(R) = \text{Int}(R) + \delta(R), \quad \delta(R) = \frac{\tau}{|R|} \quad (3)$$

$R\text{Int}(R)$ は $\text{Int}(R)$ と閾値関数 $\delta(R)$ との和になっており、閾値関数 $\delta(R)$ は領域 R の画素数 $|R|$ に反比例する関数であり τ は統合の起きやすさを調整するパラメータである．すなわち、初期は $\delta(R)$ が大きくなるため、統合が起きやすく、統合が進むにつれて $\delta(R)$ が小さくなり $\text{Int}(R)$ に近づいていく．

グラフの辺の本数を m とすると、Graph-Based アルゴリズムの計算量は、 $O(m\alpha(m))$ となる． $\alpha(m)$ はアッカーマン関数の逆関数であるため、 m が十分大きいとき、 $O(m\alpha(m)) \cong O(m)$ と近似できる．画素を頂点とする場合、 m は十分大きくなるため Graph-Based アルゴリズムは線形時間で終了する．したがって、Graph-Based アルゴリズムは比較的高速に分割が行うことができる．

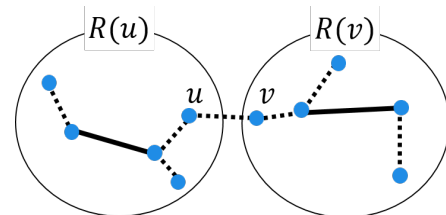


図 1 2領域にまたがる辺の重みが、2領域の何れの領域内部の相違度よりも小さければ統合する．実線の辺が領域内部の相違度を表す．

2.2 Efficient hierarchical graph-based video segmentation

G.Mathias ら [3]による研究では、隣接する Superpixel 同士を辺で結んだ Region Graph (図 3) と呼ばれるグラフを用いて画像を表現する．

このとき、Region Graph の Superpixel u, v を結ぶ辺の重み $w(e_{u,v})$ は以下の式を用いて計算する．ただし H_u は Superpixel u の色ヒストグラム、 n はヒストグラムの bin を表すインデックスであり、 $\sum_n H_u(n) = 1$ を満足する．

$$w(e_{u,v}) = \frac{1}{2} \sum_n \frac{(H_u(n) - H_v(n))^2}{H_u(n) + H_v(n)} \quad (4)$$

式(4)の右辺は2つのヒストグラム間のカイ二乗距離を表している．カイ二乗距離は、2つのヒストグラムが完全に

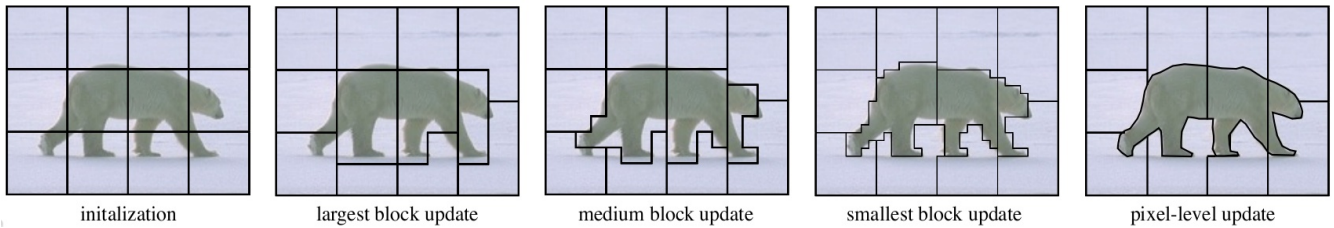


図 2 SEEDS の反復処理の様子 (文献 [5]図5 より引用)

一致している場合 0 をとり、異なっているほど、大きな値をとる。また、式(4)の最大値は、 $\frac{1}{2}(\sum_n H_u(n) + \sum_n H_v(n))$ となり、 $\sum_n H_u(n) = \sum_n H_v(n) = 1$ であるので最大値は 1 となり、色ヒストグラムが全く異なることを表している。すなわち、この手法では、[0,1]の範囲の値を取る相違度を辺の重みとしている。

このようにして得られた Region Graph に対して、前述の Graph-Based アルゴリズムを適用することで領域分割を行う。

さらに、この手法では、時間的に連続する動画フレーム間にも辺を結ぶことによって、異なるフレーム間でも一貫した領域 ID を与えることが出来る領域分割も実現している。

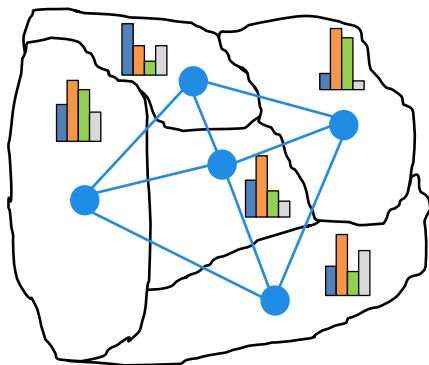


図 3 Region Graph

2.3 SEEDS

M.V. Bergh ら [5]による SEEDS は領域境界に関するエネルギー最適化を行うことによって分割を行う手法である。

はじめに、SEEDS の概要を説明する。SEEDS は領域境界を移動させて、エネルギー関数の値が上がれば、移動させた境界を新しい境界とする、という処理を段階的に格子サイズを小さくしながら反復して行う (図 2)。

具体的な計算手順は以下の通りである。まず、画像を均等な大きさの格子に分割し、境界を初期化する。このときの格子の大きさを幅 W 、高さ H とすると、各段階において、それぞれの格子の中に幅 $W/2$ 、高さ $H/2$ の格子を考える。次に、この半分の大きさの格子の辺を隣の辺へと移動させたとき、エネルギー関数が上昇する場合、境界が更新される。この処理を全ての格子で行うと次の段階へと進む。段

階が進むごとに格子の幅・高さは半分になり、境界の移動がより細かくなっていく。反復回数はパラメータとして与えられるが、ピクセル単位の移動段階が終わると、与えられた反復回数に達していなくてもアルゴリズムは終了する。

ここで用いられるエネルギー関数 $E(s)$ は、次式で定義される。

$$E(s) = H(s) + \gamma G(s) \quad (5)$$

s は分割を表し、 $H(s)$ は色分布項、 $G(s)$ は境界項と呼ばれる項である。 γ は境界項の影響を調整するパラメータである。

分割 s によって分けられた各領域の色ヒストグラム間の相違度が大きくなると、 $H(s)$ は大きな値をとり、領域の境界形状が滑らかになると、 $G(s)$ は小さな値をとる。

次に、SEEDS の空間的な連続性と境界の形状について考える。SEEDS は境界を移動させるアルゴリズムであるため、同じ領域 ID が離れた領域に割り振られることはない。したがって、空間的な連続性も保証される。また、SEEDS は色ヒストグラム間の相違度が大きくなるように境界の移動を繰り返すため、 γ を小さくすると同一オブジェクト内部で隣接する領域間の境界の動きは不安定になる。そのため $\gamma = 0$ のときの SEEDS の Superpixel 画像はオブジェクト内部の境界は複雑な形状になりやすいという性質がある。

3. Superpixel の色と境界形状を併用した画像分割

SEEDS の境界項の重み $\gamma = 0$ での Superpixel 画像は、同一オブジェクト内で互いに隣接する領域の境界形状は複雑になり、異なるオブジェクト間の境界形状は滑らかになるという性質がある。この性質に着目すると、互いに隣接する Superpixel を、同一オブジェクトに属する場合は統合し、異なるオブジェクトに属する場合は統合しない、という計算が実現できるのではないかと、ということが本研究のアイデアである。

また、グラフに基づく領域分割において、隣接する Superpixel の色分布の類似度 (相違度) だけで辺の重み付けをすると、照明変化などによって同一オブジェクト内の

Supapixel が統合されなくなることが知られている。例えば、図 4 に示すように、色分布の類似性のみでグラフの辺に重み付けをただけでは、照明変化によって同一オブジェクト内の Supapixel が統合されず、残ることがある（このことを以下では領域の断片化と呼ぶ）

図 4 について詳しく見てみると、鯨の領域内部の断片化された領域は複雑な形状をしていることがわかる。したがって、色分布間の類似度（相違度）に Supapixel 間の境界形状の複雑さを辺の重みに加えることは、このような問題の解決、あるいは、緩和に役立つはずである。

前述した Graph-Based Algorithm は重みが小さい辺から順に領域統合の判定を行うため、重みが小さい辺ほど統合が起りやすくなる。したがって、境界形状が複雑であるほど辺の重みを小さくすることができれば、より正確に画像をグラフで表現することができると考えられる。3.1 では、境界形状の複雑さを計算する方法を考える。

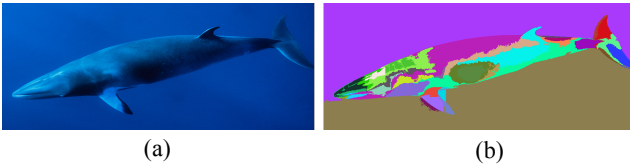


図 4 (a) : 入力画像, (b) : 色分布のみで重み付けをし、分割した画像。照明変化によって、鯨の領域内部に色の変化が生じ、領域が断片化している。

3.1 境界形状項の導入

2つの Supapixel 間の境界は、数学的には平面曲線であり、平面曲線の形状を決定する不変量として曲率がある。以下では曲率に基づき、境界形状の複雑さを計算する方法について述べる。

平面曲線上の点 P における接線と x 軸がなす角を ϕ とし、弧長パラメータを s としたとき（図 5）、P における曲率 κ は以下の式(6)で定義される。

$$\kappa(s) = \frac{d\phi}{ds} \quad (6)$$

そして、 κ の絶対値を曲線全体で s について積分して得られる値は、曲線の全曲率と呼ばれ、全曲率 μ は、

$$\mu = \int |\kappa(s)| ds \quad (7)$$

となる。実際、画像中での座標を用いた計算では、弧長パラメータを用いるのは困難であるため、曲線 \mathbf{p} が任意のパラメータ t について $\mathbf{p}(t) = \begin{pmatrix} x(t) \\ y(t) \end{pmatrix}$ と表されるとき、式(7)を変形し、

$$\kappa(t) = \frac{\dot{x}\ddot{y} - \ddot{x}y}{(\dot{x}^2 + \dot{y}^2)^{3/2}} \quad (8)$$

で求めることができる。ただし、 $\dot{x} = \frac{dx}{dt}, \dot{y} = \frac{dy}{dt}$ である。また、 s と t の関係は、

$$ds = (\dot{x}^2 + \dot{y}^2)^{1/2} dt \quad (9)$$

であるため、式(6), (7), (8), (9)より、

$$\mu = \int \frac{|\dot{x}\ddot{y} - \ddot{x}y|}{\dot{x}^2 + \dot{y}^2} dt \quad (10)$$

が導かれる。この式(10)を用いて、画像内の Supapixel 間の境界形状の複雑さを求める。

前述したとおり、境界形状が複雑であるほど辺の重みを小さくする必要がある。そのため、2つの Supapixel を u, v とし、 u, v 間の境界の全曲率 $\mu_{u,v}$ が大きくなるほど、値が小さくなる関数 $B(e_{u,v})$ を以下に定義する。

$$B(e_{u,v}) = \exp\left(-\frac{\mu_{u,v}}{\sigma}\right) \quad (11)$$

提案手法では、 $B(e_{u,v})$ は境界の形状によって値が変化するため、境界形状項と呼ぶ。 $B(e_{u,v})$ は境界形状の複雑さ、すなわち全曲率 $\mu_{u,v}$ が大きくなれば、0 に近づき、直線、すなわち $\mu_{u,v} = 0$ ならば $B(e_{u,v}) = 1$ となる。 σ は指数関数の広がりを調節するパラメータである。

また、この項は $(0,1]$ の値をとるため、式(4)で示した色分布の類似度と併用する際にも適していると考えられる。

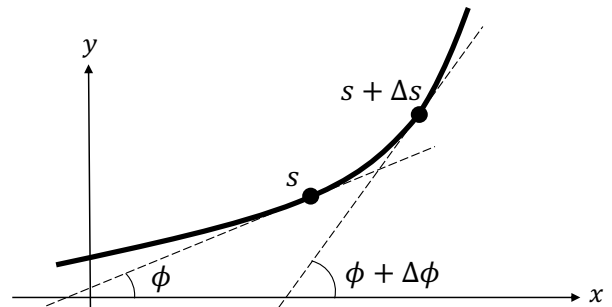


図 5 平面曲線の接線と x 軸がなす角の単位弧長あたりの変化率が曲率を表す。

3.2 色分布項と境界形状項の併用

次に、色分布の類似度と、境界形状の複雑さを併用して重み付けするために、3.1 で述べた境界形状項を色分布の類似度と結合する。以下では、式(4)の右辺を $C(e_{u,v})$ とし、色分布項と呼ぶ。色分布項 $C(e_{u,v})$ と境界形状項 $B(e_{u,v})$ を結合して求める重み $w(e_{u,v})$ を以下に示す。

$$w(e_{u,v}) = \frac{C(e_{u,v})B(e_{u,v})}{\rho C(e_{u,v}) + (1 - \rho)B(e_{u,v})} \quad (12)$$

式(12)の右辺は、 $C(e_{u,v})$ と $B(e_{u,v})$ の重み付き調和平均を計算している。その理由は、重み付き調和平均は、2つの変数が共に 1 に近づけば 1 に近づき、どちらかが 0 に近づけば 0 に近づく関数であるためである。これは言い換えると、色分布が類似しているか、境界形状が複雑であれば、重みを小さくし、色分布も異なり、かつ境界形状が滑らかであれば重みを大きくするということである。これにより、領域統合の判定のときに、境界形状の複雑さを補助的に利用することができる。また、画像によっては色分布項、境界

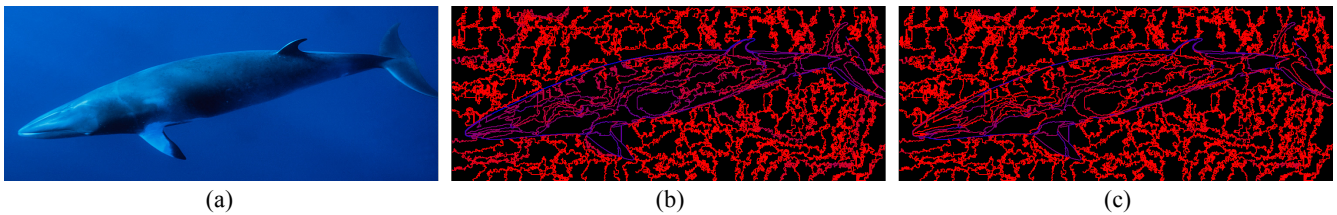


図 6 (a): 元画像, (b): 色分布項のみによる重み付け, (c): 境界形状項を併用した重み付け. 重みが小さい境界は赤色, 大きい境界は青色に近くなるように色付けした. 鯨の領域内部の境界の色を比較すると, (c)の方が(b)よりもより赤く色付けされている.

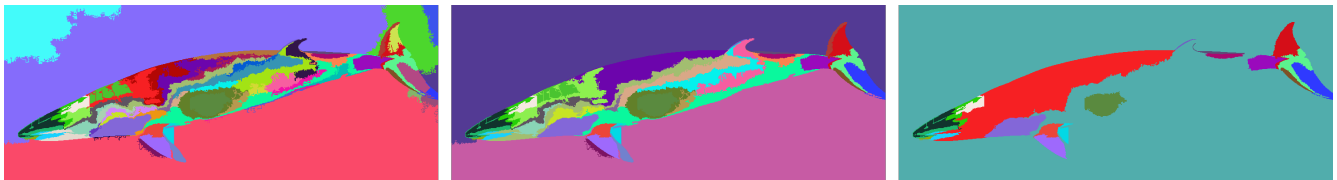


図 7 色分布項のみによる重み付け. 左から順に, $\tau = 0.1, 0.2, 0.3$ と変化させたときの分割画像.

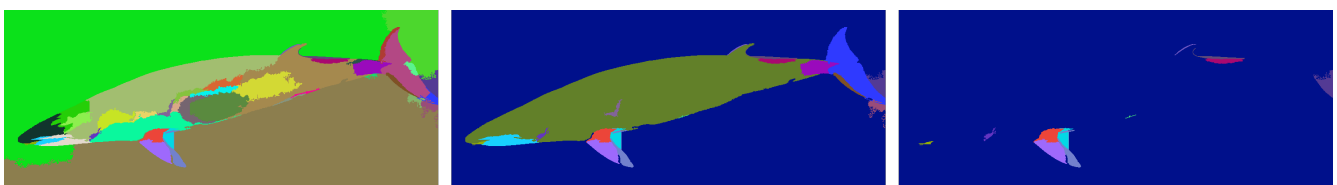


図 8 境界形状項を併用した重み付け. 左から順に, $\tau = 0.1, 0.2, 0.3$ と変化させたときの分割画像.

形状項どちらかを重視した方が良いこともあるため, それぞれの項の重みを調節するパラメータを ρ とする. ただし, $\rho \in [0,1]$ である.

3.3 境界形状項の影響に関する予備実験

はじめに, 境界形状項の併用による重みの変化を確認するため, グラフの重みを境界に色付けした画像を図 6 に示す. 境界の色は, 辺の重みが 0 に近い (統合されやすい) とき赤色に近づき, 1 に近い (統合されにくい) ときは青色に近くなるように色付けした. Superpixel については左右どちらも, SEEDS を用い, 格子の数は 250, 境界項 $G(\cdot)$ の重み $\gamma = 0$ として分割を行った.

図 6 の(b)が色分布項のみ (式(4)) を用いて重み付けをした境界画像, (c)が境界形状項 (式(12)) を併用して重み付けをした境界画像である. ただし, 境界形状項を併用するときのパラメータは, $\sigma = 30, \rho = 0.7$ とした. 図 7 の(b), (c)で比較すると, 境界形状項を併用した境界画像(c)の方が色分布項のみの境界画像(b)よりも鯨の内部の境界がより赤色に近づいていることがわかる. これは, 鯨の領域内部の色分布は部分的に変化しているが, 境界形状は複雑であるという性質を満たしているためと考えられる.

次に, Graph-Based Algorithm を適用し, 分割を行った結果を比較する. 図 7 が色分布項のみによる重み付け, 図 8 が境界形状項を併用した重み付けを行い分割した結果で

ある. また, 図 7, 図 8 どちらも左から, $\tau = 0.1, 0.2, 0.3$ と変化させたときの分割結果である. 図 8 は図 7 と比べて, 鯨の輪郭を保ちつつ, 領域の断片化が少ないことがわかる.

4. 実験

境界形状項を併用した重み付けによって, 分割精度が向上することを確認するため, 既存手法を色分布項のみを用いて重み付け (式(4)), 提案手法を色分布項と境界形状項と併用した重み付け (式(12)) を行い, 比較実験を行う.

実験には, Berkeley Segmentation Dataset and Test (BSDS300) [6]のテスト用のカラー自然画像 100 枚を使用する. 領域分割手法の分割精度を測るための標準的なデータセットである.

被験者の主観によるオブジェクトの定義は異なるため, 正解データのばらつきを考慮した評価が必要となる. BSDS300には, 画像1枚に対し, 被験者5名以上による領域分割の正解データが手動で作成されているため, 複数の被験者による平均的な評価を行うことができる. 以下では正解データ, すなわち被験者によるラベル付けされた画像をGround Truth画像と呼ぶ.

4.1 評価方法

領域分割の精度を測る尺度には, 領域境界についての

Precision および Recall [7]を用いる。Precision, Recall は以下の式(13), (14)で表される。ただし, S_{source}, S_{target} はそれぞれ, アルゴリズムによる分割画像, Ground Truth 画像の境界座標の集合を表す。

$$Precision = \frac{Matched(S_{source}, S_{target})}{|S_{source}|} \quad (13)$$

$$Recall = \frac{Matched(S_{target}, S_{source})}{|S_{target}|} \quad (14)$$

以下では A, B を任意の座標の集合として, $Matched(A, B)$ について説明する。 A のある 1 つの境界座標に注目したとき, その近傍 ϵ ピクセル内に B の境界座標があれば, 1 とカウントする。それを A の全ての境界座標に対して足し合わせた値が $Matched(A, B)$ である。 $|A|$ は集合 A の要素の数であるため, 式(13)はアルゴリズムによる分割画像の境界座標の中で, Ground Truth 画像の境界座標が含まれている割合を表している。同様に式(14)は Ground Truth 画像の境界座標の中で, アルゴリズムによる分割画像の境界座標が含まれている割合を表している。これは, Precision が高ければ領域境界の誤検出が少ないことを意味し, Recall が高ければ, 正しく検出された数が多いことを意味している。

パラメータによって領域数が増えるような画像分割の手法を比較するときは, 一般的に縦軸に Precision, 横軸に Recall をとりプロットした PR 曲線を用いる。PR 曲線が, 全体的に右上に位置していれば, 良いとされる。

4.2 実験条件

PR 曲線をプロットする方法は,

(ア) 式(13),(14)をそれぞれ全画像について分子・分母で総和して計算した値

(イ) 11 点補完平均適合率

の 2 通りを用いた。(ア)は式(13), (14)の計算において $Matched(), |S_{source}|, |S_{target}|$ を各 τ においてデータセットの全画像で総和して計算した値であり, (イ)は Recall を 0, 0.1, 0.2, ..., 1.0 と 11 点に区切り, それぞれの点において補完適合率を平均した値である。(ア)はパラメータ τ ごとの分割精度を評価するのに対し, (イ)は, Recall の各点ごとに分割精度を評価するという違いがある。パラメータ σ, ρ は $\sigma = 200, \rho = 0.5$ に固定した。

既存手法・提案手法共に,

- 分割の細かさを調整するパラメータ τ は $[0,1]$ の範囲を 20 段階で変化させた。
- 境界のずれの許容範囲である ϵ は $\epsilon = 2, \epsilon = 6$ として 2 通りの評価を行った。
- 色分布の類似度の計算には, Lab 色空間のヒストグラムを用いた。

画像ごとの PR 曲線には, Precision と Recall の調和平均である F 値が最大となる点をプロットした。既存手法と提

案手法で分割画像を比較するときは, $\epsilon = 2$ における F 値が最大の点での分割画像を用いる。

4.3 結果と考察

4.2 で述べた条件で実験を行い, 得られた PR 曲線を図 9, 図 10, 図 11, 図 12 に示す。橙色の曲線が提案手法, 水色の曲線が既存手法である。(ア), (イ) どちらの場合も提案手法の PR 曲線が既存手法の PR 曲線よりも上部に位置しており, データセット全体で分割精度の向上が確認できた。

次に, 実際の分割画像で比較を行う。特に F 値の上昇が大きかった分割画像とそのときの PR 曲線を図 13, 図 14 に示す。図 13 の(b)は, 背景の波の領域が断片的に残っているが, (c)ではそれらの領域が背景と統合されている。これは, 断片化された領域の形状が複雑であるため, 境界形状項によって辺の重みが小さく抑えられ, 統合が進んだためと考えられる。その結果, (b)の F 値が 0.616, (c)の F 値が 0.729 と 0.113 上昇している。このように, 提案手法は領域の断片化を解消するのに有効であることが確認できた。

また, 図 13 の(e)も同様に領域の断片化が軽減されている。

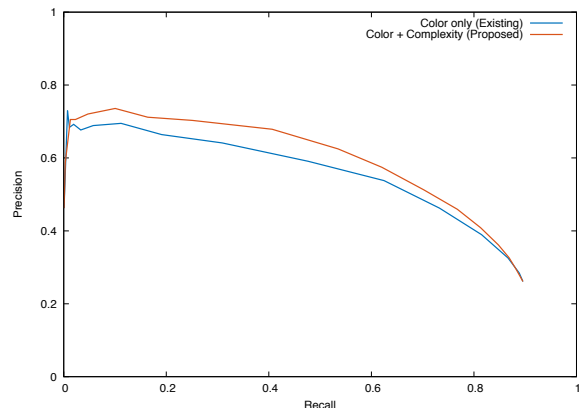


図 9 (ア)の方法で計算した PR 曲線 ($\epsilon = 2$)

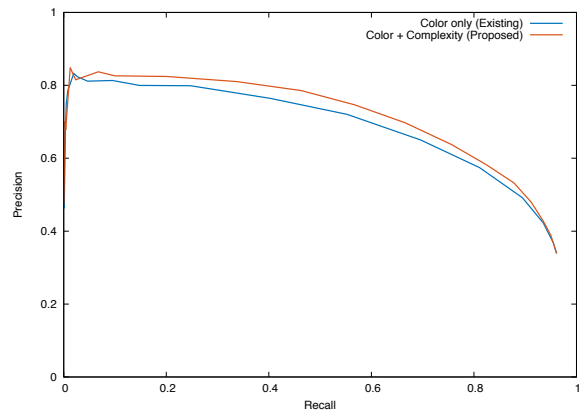


図 10 (ア)の方法で計算した PR 曲線 ($\epsilon = 6$)

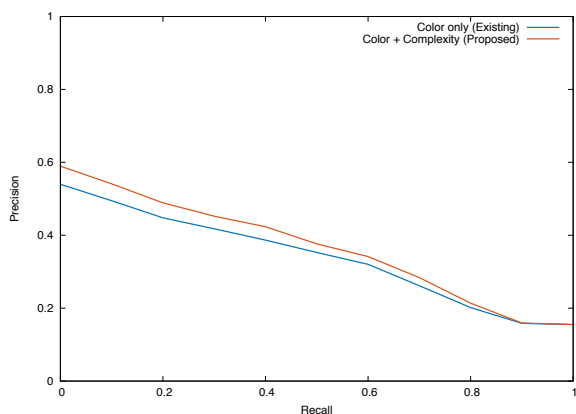


図 11 (イ) : 11 点補完平均適合率 ($\epsilon = 2$)

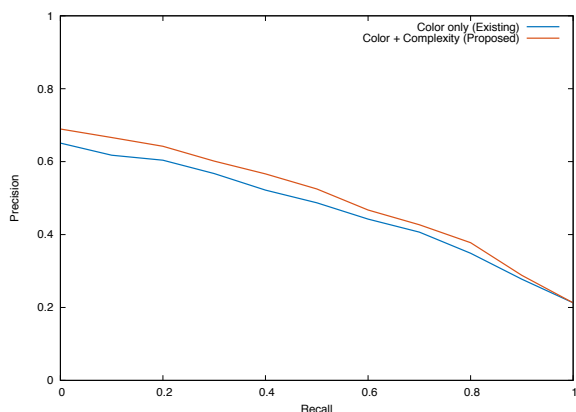


図 12 (イ) : 11 点補完平均適合率 ($\epsilon = 6$)

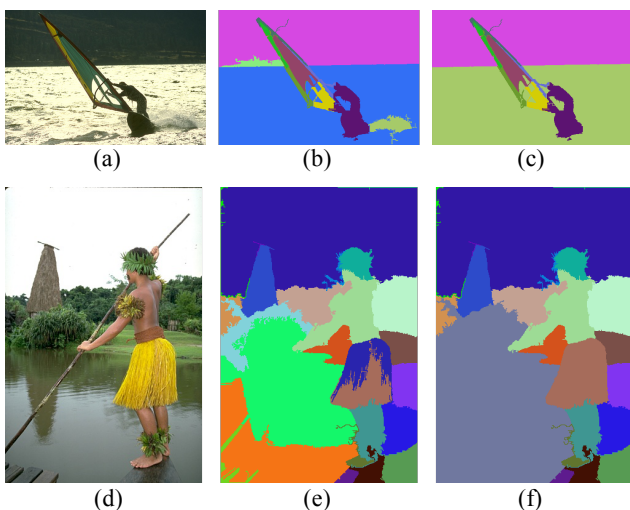


図 13 F 値の上昇値が特に大きい画像.

(a)元画像, (b)既存手法 (F 値 : 0.616), (c)提案手法 (F 値 : 0.729). (b)では統合されず残ってしまっていた波の領域が(c)では統合されている.
(d)元画像, (e)既存手法 (F 値 : 0.529), (f)提案手法 (F 値 : 0.610). (e)では人間の腰の部分や水面に断片的な領域が残っているが, (f)では統合されている.

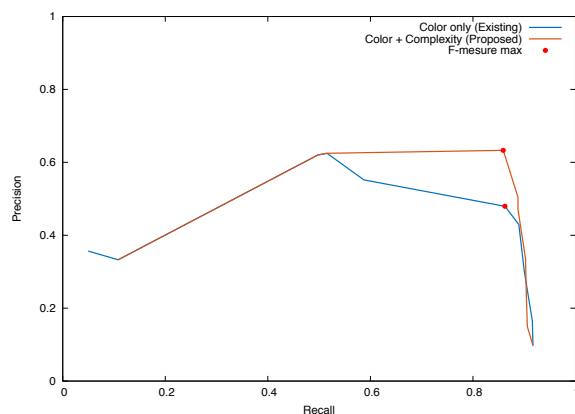


図 14 図 13(a)の画像から得られた PL 曲線, 赤い点は F 値が最大の点を表す ($\epsilon = 2$).

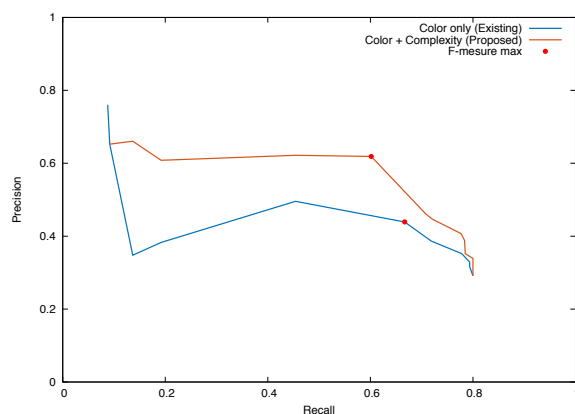


図 15 図 13(d)の画像から得られた PL 曲線, 赤い点は F 値が最大の点を表す ($\epsilon = 2$).

5. まとめ

本研究では, Semantic Segmentation のための正解データ作成を補助する対話的領域分割ツールのために, 人間の直感に近い領域分割アルゴリズムの開発を行った. 基本とするアルゴリズムは, 計算量と精度の観点から, SEEDS によって得られる Superpixel を頂点とするグラフベースの領域分割法とした.

SEEDS の Superpixel の形状はオブジェクト間では滑らかに, オブジェクト内では複雑になりやすいという性質がある. 本研究ではこの性質に着目し, 色と境界形状の複雑さを合わせてグラフの辺の重み付けを行う画像の領域分割法を提案した. Superpixel 間の境界形状の複雑さは曲線の全曲率が大きいほど辺の重みは小さくなるようにした.

実験では画像分割の分野で広く用いられている BSDS300 データセットを用いて行った. 全体の評価は PR 曲線, 11 点補完平均適合率を用いて行い, 提案手法が既存手法よりも高いスコアをとり, 分割精度が向上したことを確認した.

実際の分割画像を比較しても, 照明などの変化による領域の断片化が軽減されていることが確認できた.

パラメータ σ, ρ については、画像ごとに最適な値を推定することが今後の課題である。

参考文献

- [1] B. Peng, L. Zhang and D. Zhang, "A survey of graph theoretical approaches to image segmentation," *Pattern Recognition*, vol. 46, no. 3, pp. 1020 -- 1038, 2013.
- [2] H. Zhang, J. E. Fritts and S. A. Goldman, "Image segmentation evaluation: A survey of unsupervised methods," *computer vision and image understanding*, vol. 110, no. 2, pp. 260--280, 2008.
- [3] M. Grundmann, V. Kwatra, M. Han and I. Essa, "Efficient hierarchical graph-based video segmentation," *2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pp. 2141-2148, 2010.
- [4] F. Pedro and H. Daniel P, "Efficient graph-based image segmentation," *International journal of computer vision*, vol. 59, no. 2, pp. 167-181, 2004.
- [5] M. Van den Bergh, B. Xavier, R. Gemma, B. de Capitani and L. Van Gool, "SEEDS: Superpixels Extracted via Energy-Driven Sampling," *Computer Vision – ECCV 2012. ECCV 2012. Lecture Notes in Computer Science*, vol. 7578, pp. 13-26, 2012.
- [6] M. David, F. Charless, T. Doron and M. Jitendra, "A database of human segmented natural images and its application to evaluating segmentation algorithms and measuring ecological statistics," *Eighth IEEE International Conference on Computer Vision, 2001. ICCV 2001. Proceedings.*, vol. 2, pp. 416-423, 2001.
- [7] E. J. Francisco and J. D. Allan, "Benchmarking image segmentation algorithms," *International Journal of Computer Vision*, vol. 85, no. 2, pp. 167-181, 2009.

正誤表

下記の箇所に誤りがございました。お詫びして訂正いたします。

訂正箇所	誤	正
1 ページ 脚注追加		†1 和歌山大学大学院 システム工学研究科 a) kubo@vrl.sys.wakayama-u.ac.jp b) twada@ieee.org