

グラフカットを用いたユーザ入力不要の画像領域分割

小川 真[†] 久保田 光一[‡]

中央大学大学院 理工学研究科[†] 中央大学 理工学部[‡]

1 はじめに

画像における領域分割とは、一枚の画像から特徴量（色、テクスチャなど）を用いて対象となる前景領域を抽出することであり、一般物体認識や画像合成など画像処理技術の前処理として利用されている。

領域分割問題をエネルギー最小化問題と捉えて解を求める手法が多く提案されており、その手法の一つとしてグラフカットによる領域分割が挙げられる。グラフカットでは、各領域からエネルギー関数を定義し、大域最小解を求めることが可能であり、領域と境界の両方の特性を用いた領域分割が実現する。

グラフカットを用いた手法として、Rother らにより GrabCut[1] が提案されている。この手法ではユーザ入力により対象物体を含む矩形領域を大まかに指定し、その外側を背景領域、内側を前景候補領域として学習する。前景と背景それぞれの色分布を混合正規分布モデル (Gaussian Mixture Model, GMM) を用いてモデル化し、グラフカットを行う。領域分割の結果を再学習し、グラフカットを繰り返し行うことで正確な前景領域の抽出を可能にしているが、ユーザ入力が必要であるため処理を行うのに必要な手間が増えてしまう。そこで、本研究ではユーザ入力による矩形領域の選択操作を必要としない GMM の構築方法を提案する。

2 提案手法

以下に提案手法の処理の流れを示す。

- (1) 入力画像に対し、 k -means clustering を用いてラベリングを行う。
- (2) ラベリングされた画像を格子状に分割する。
- (3) 分割されてできた各小領域に対して、再ラベリングを行う。
- (4) 割り当てられたラベルをもとに、背景候補領域と前景候補領域を設定し、GMM を構築する。
- (5) 構築した GMM をもとにグラフカットを行い、出力画像を得る。

(1) から (4) までが本研究での主とするところであり、ユーザ入力に代わって自動的に処理を行っている。(1) は節 2.1、(2) から (4) は節 2.2 で説明する。

2.1 k -means clustering によるラベリング

入力画像の全ピクセルに対して、0 または 1 でラベリングを行う。特徴量として、ピクセルの RGB 色空間を利用している。

2.1.1 k -means clustering

k -means clustering[2] は幅広い分野で使われているクラスタリングを行うアルゴリズムである。 k はクラス数を表している。以下に処理の流れを簡潔に示す。

- (1) 各データにランダムにクラスを割り当てる。
- (2) 各クラスの重心をそれぞれ計算する。
- (3) 各データと各重心との距離を計算し、最も近い重心のクラスに分類し直す。
- (4) クラスの割り当てが変化しなかった場合、もしくは変化量が事前に設定した一定の閾値を下回った場合に、収束したと判断して処理を終了する。

2.2 小領域への分割と再ラベリング

まず、縦の分割数と横の分割数を設定する。それらの値をもとに節 2.1 で得られたラベリングされた画像を幅が等間隔になるように格子状に分割する。つまり、小領域の数は縦の分割数と横の分割数の積と等しくなる。

次に、各小領域に含まれるピクセルに割り当てられたラベルの 0 と 1 の数を調べ、多いほうのラベルでその小領域に含まれる全てのピクセルを再ラベリングする。各小領域に対して一つのラベルが割り当てられることになる。

割り当てられたラベルのうち、「背景らしい」ほうを背景候補領域として、もう一つを前景候補領域として学習を行い、GMM を構築する。ここで、「背景らしい」とは、節 2.1 で得られたラベリングされた画像の最も外側（枠部分）のラベルを集計し、多いほうのラベルのことである。

入力画像と再ラベリング後の画像を図 1 に示す。黒い部分が背景候補領域、それ以外が前景候補領域を表している。縦の分割数を 10、横の分割数を 8 としている。

Image Segmentation using Graph Cuts without User Input
[†]Makoto OGAWA, Graduate School of Science and Engineering, Chuo University

[‡]Koichi KUBOTA, Faculty of Science and Engineering, Chuo University



入力画像 再ラベリング後の画像
図1 入力画像と再ラベリング後の画像

3 実験

3.1 実験概要

10枚の画像に対して、提案手法と従来手法であるGrabCutの比較実験を行った。従来手法ではユーザの入力として矩形領域の指定のみを行うものとし、繰り返し回数は5回とする。なお、提案手法のグラフカット処理は1回のみ行う。

実験で使う画像及び従来手法の入力は全てMicrosoft Research CambridgeのGrabCutデータセット[3]を利用する。このデータセットには領域分割をした際の結果の画像も含まれている。その画像を正解とし、各手法のそれぞれの画像に対する誤検出率を求めた。以下に誤検出率を求める式を示す。

$$\text{誤検出率} [\%] = \frac{N_B + N_F}{N} \times 100 \quad (1)$$

ここで、 N_B は背景領域の誤検出ピクセル数、 N_F は前景領域の誤検出ピクセル数、 N は全ピクセル数を表す。

提案手法の誤検出率は縦の分割数及び横の分割数を2から10まで動かしたときの、最も値が低いものを選択している。

3.2 実験結果

各画像に対する誤検出率を調べた結果を図2に示す。横軸が画像を示す番号、縦軸が誤検出率を表している。従来手法より大幅に誤検出率が低いものもあれば、反対に大きく上がってしまったものもある。提案手法ではピクセルの色情報をもとにGMMを構築しているため、背景領域と前景領域で色が大きく異なっている画像に対しては強いが、似たような色の場合に良い結果になりにくいと考えられる。

誤検出率の差が顕著だった8番の画像の領域分割の結果を図3に示す。このように背景に複雑なエッジが含まれている画像の場合、従来手法では背景と前景の境界がわからず、前景部分を大きく残してしまうことがある。それに対し、提案手法では背景の複雑なエッジに惑わされずに領域分割できている。しかし、前景領域に大きく異なった色を複数含むため、それぞれに別のラベルが割り当てられてしまい、その結果、一部分が欠けてしまったと考えられる。

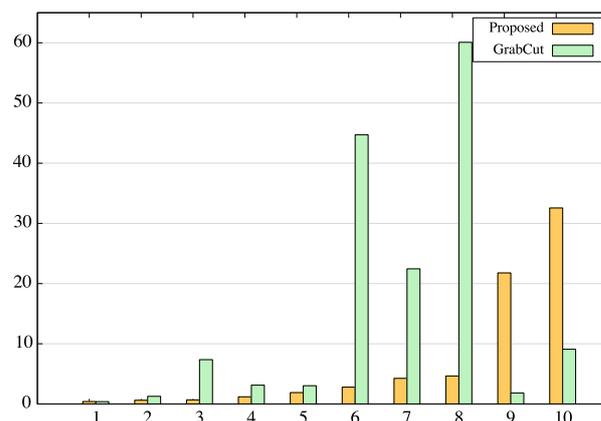


図2 各画像に対する誤検出率



提案手法 従来手法
図3 8番の画像の領域分割の結果

今回の実験では、縦の分割数と横の分割数を適当に選択したが、その値によって誤検出率が大きく変わることがある。今後の課題として、各画像に適切な分割数を選択するための実験や、より一般的な画像を対象とした実験を追加で行い、有効性をさらに検討する必要がある。

4 おわりに

本研究では、 k -means clustering を用いてユーザ入力による矩形領域の選択操作を必要としない領域分割の手法を提案した。また、実験によりその有効性を評価した。

参考文献

[1] C. Rother, V. Kolmogorov, and A. Blake. “Grab-Cut”: Interactive Foreground Extraction Using Iterated Graph Cuts. *ACM Transactions on Graphics (Proc. SIGGRAPH2004)*, 23(3):309–314, 2004.

[2] J. B. MacQueen. Some Methods for classification and Analysis of Multivariate Observations. *Proceedings of 5th Berkeley Symposium on Mathematical Statistics and Probability*, University of California Press, 1:281–297, 1967.

[3] <http://research.microsoft.com/en-us/um/cambridge/projects/visionimagevideoediting/segmentation/grabcut.htm>