基準調査領域の非交差和領域の最適イメージ切り出しアルゴリズム

Algorithms for computing the optimal image segmentation of nonintersecting union of base monotone regions

全 順美* 葛西 亮生* コルマン マティアス* 徳山 豪*
Jinhee Chun Ryosei Kasai Matias Korman Takeshi Tokuyama

1 はじめに

$n \times n$のピクセルグリッド平面$G$からのイメージ切り出し問題は、画像処理における最重要問題の一つである。様々な実用的なアプローチが知られているが、Asanoら[1]はイメージ切り出し問題を組み合わせて最適化問題として定式化し、パラメトリック最適化の手法を利用して解決する枠組みを提案し、先端的なアルゴリズム理論の応用として注目されている。画像はピクセルグリッド$G$における輝度分布として表される。具体的には、各ピクセル$p = (i, j)$に対して、実数値$b(p) = b_{ij}$を与える。色画像の場合は、適当なピクセルへの影響を考慮して輝度を計算し、その後でピクセルに対する特性分布によってうけることができる。

$G$のピクセル集合の適当な族$\mathcal{F}$を考え、$\mathcal{F}$の要素$S$をイメージ領域と呼ぶ、最適化問題として定式化するときには、イメージ領域$S$に対してある目的関数を与え、その最小化もしくは最大化を行う。具体的にAsanoらの利点はクラス内分割と呼ばれるもので、$S$内の輝度の分割和と$G \setminus S$内の輝度の分割和の和を最小化する。パラメトリック最適化の手法を用いる、次のような問題を解くことが必要になる。

最大重み領域問題：グリッド$G$の各ピクセル$(i, j)$

に実数値の重み$w_{ij}$を与える。このとき、$S \in \mathcal{F}$

で、重み和

\[ w(S) = \sum_{(i, j) \in S} w_{ij} \]

を最大化するものを求める。

パラメトリック最適化の理論から、適当なパラメータ値$\theta$が存在し、$w_{ij} = b_{ij} - \theta$に対する最大重み領域がクラス内分割を最小化することが示される。さらに、このパラメータ値は、最大重み領域問題が最悪$O(n^2)$回（実用的には$O(\log n)$回）解くことがでられると考えられる。

したがって、最大重み領域問題の効率よく解くためには、クラス内分割の最小化が可能である。しかしながら、最大重み領域問題は領域族$\mathcal{F}$の性質に計算複雑度が大きく依存し、たとえば$\mathcal{F}$として連結なグリッド全てを含まない族を取ると、NP完全となる[1]。一方、最大重み領域問題が多形性時に解ける場合も多くあり、過去にはそのような領域族が利用されてきた。代表的なものとして、基準調査領域、単調領域、単調領域のネットワーク構築においては基準領域のグリッドサイズ$N = n^2$に対して線形時間で計算することができます。

ここで、各列に対して、$0 \leq f(j) \leq n$に対応させる関数$f$を考える。このような関数はグリッド上の曲線を定義し、そのような曲線の下側領域で表されるピクセル集合を$(x, y)$、あるいはグリッドの底辺を基とする基準調査領域（base monotone region）と呼ぶ。具体的には、$S = \{(i,j) : i \leq f(j)\}$として表される領域である。同様にグリッド境界の4つの辺それぞれを基として基準調査領域を考えることができる。二つの関数$f, g$を用いて、$S = \{(i,j) : g(j) \leq f(j)\}$と表される領域を単調領域と呼び、$x$単調領域である。かつグリッド位置で単調領域に$S$単調領域と呼ぶ。これらの領域に関する最大重み領域問題はデータマイニングや放射線医療における最適化においても広く用いられている[5, 6, 2, 1]。たとえば、現行の放射線治療においては、放射線の照射は正方形の照射窓を上下から伸びる遮蔽膜で遮れ、照射される対象の腫瘍に放射線を当てると共に、重要な臓器には放射線を遮ることができる。これには、Interleaf collimatorが、遮蔽板の位置は上記の二つの関数$f, g$で定義され、照射される部分$x$単調領域と呼ばれます。なお、いくつかの$x$単調領域での照射を重ねることにより、必要な量を腫瘍に照射し、かつ臓器への副作用を最小化することが求められ、一方照射回数の最小化も要求される。この最小化問題は困難な計算問題であり、現在の放射線治療の性能の関係で$x$単調領域にはいくつかの制限がついて、最大重み領域問題を基盤として実用解法を設計することが可能である。

しかしながら、これらの領域の生成は非常に特殊であり、より複雑なイメージの切り出しを行うためには、これらの領域を複数用いて表示できるような図形を構成することが望ましい。しかしながら、$x$軸を基とした基準調査領域と$y$軸を基とした基準調査領域の和集合として表示できる図形の性質を考えると、複雑な交差パターンを持つ図形が生成され、対応する最大重み領域問題はNP困難問題とならない[3]。

本論文では4つのグリッド境界辺を基とする基準調査領域を考え、それらの非交差和で表示される（すなわち、基準調査領域に分割される）図形の組み合わせを考える（図1）。たとえば、上下2つの辺を基とする基準調査領域の非交差和（この場合は交差和でも同じである）を考えると、その組合せは$x$単調領域であり、重
表 1: アルゴリズムの計算量

<table>
<thead>
<tr>
<th>分割数</th>
<th>時間計算量</th>
<th>空間計算量</th>
</tr>
</thead>
<tbody>
<tr>
<td>3分割以下</td>
<td>(O(N))</td>
<td>(O(N))</td>
</tr>
<tr>
<td>4分割</td>
<td>(O(N^2))</td>
<td>(O(N))</td>
</tr>
<tr>
<td>5分割</td>
<td>(O(N^{1.5}))</td>
<td>(O(N^{1.5}))</td>
</tr>
</tbody>
</table>

2 問題の定義と基本となるアルゴリズム

ここでは、重みを表す \(n \times n\) 行列 \(W = (w_{ij}) = (W(i,j))\) と考え、この行列を入力とする。本論文では、各箇所に対して重みを与える行列作成を前提しているが、行列を作成するため、通常の平滑度の \((x,y)\) 座標の関数であることに注意を必要がある。便宜上、\(W(i,j) = W(i+1,j) = W(i,j+1) = W(i,j)\)（\(i = 1\) と \(j = 1\)）とする。行列作成の任意の箇所の重みを選んだ集合 \(S = \{(i_1,j_1), \ldots, (i_k,j_k)\}\) と \(S \subseteq \{1,2,\ldots, n\} \times \{1,2,\ldots, n\}\) を考えたとき、\(S\) の重み \(w(S)\) を

\[ w(S) = \sum_{i=1}^{k} w(i,j) \]

と定義する。今、4 つの非交差な部分集合 \(U, D, L, R\) を考え、

\[ w(U \cup D \cup L \cup R) \]

が最大になるような \(U, D, L, R\) を求めたい。ただし、ここで \(U, D, L, R\) は4つの境界を持つ基準領域であり、それぞれ適当な条件を記述すること、以下のようになる。

\[(i,j) \in U \implies ((i+1,j) \in U) \lor (i = n)\]
\[(i,j) \in D \implies ((i-1,j) \in D) \lor (i = 0)\]
\[(i,j) \in L \implies ((i,j-1) \in L) \lor (j = 0)\]
\[(i,j) \in R \implies ((i+1,j) \in R) \lor (j = n)\]

直感的には、切り出し領域の \(U, D, L, R\) の各部分に色を割り当てると、4つの境界の各長さが4のベンキを（それぞれ横、縦的方向に）塗ってできる行列の色塗り問題と捉えることができる。ベンキの刷毛は途中で申し上げはならないとし、境界の辺から開始して色塗りを行うと、一つの色で塗られる領域は基準領域となる。目的は、色領域の長さを最大化する那の色塗りを考え、色に関して塗られた領域の重みを最大にするのである。本論文で与えるアルゴリズムの説明とその図示に便利なので、以下、領域切り出し問題と解釈する。

手順は、1色のみを用いた色塗りの場面を想定する。これは基準領域に関する色塗り問題であり、既存方法である。例として、\(w(D)\) の最大値について考えると、列ごとに分

割して計算することができる。すなわち、第 \(i\) 列成分の \(i\) 行目までのプレフィックス和

\[ \text{sum}(i,j) = \sum_{k=0}^{i} W(k,j) \]

を計算し、連続式

\[ D(i,j) = \max\{D(i-1,j), \text{sum}(i,j)\} \]

を利用して、第 \(j\) 行目のプレフィックス最大値

\[ D(i,j) = \max\{D(i-1,j), \text{sum}(i,j)\} \]

を、最大を達成する行番号と共に計算する。プレフィックス最大値を達成する部分を各列において塗れば、領域が計算できる。同様にして、

\[ U(i,j) = \max\{U(i,k), \sum_{k=0}^{n+1} W(l,j)\} \]
\[ L(i,j) = \max\{L(i,k), \sum_{k=0}^{n+1} W(l,k)\} \]
\[ R(i,j) = \max\{R(i,k), \sum_{k=0}^{n+1} W(l,k)\} \]

も求められる。これらの4つのテンポルは以下のより複雑な場合のアルゴリズムを設計し利用する。これらのテンポルの構成にかかる計算量は、明らかに時間計算量、空間計算量ともに \(O(n^2) = O(N)\) であり、ピクセル数に対して線形である。

2.1 2色での色塗り

次に、ピクセルグリッド平面の2色での最適な色塗りについて考える。2色の塗り方は大きく以下の2つのパターンに分類できる（図2）。

1. 隣接する位置の境界から色塗り（D または R）
2. 対面する位置の境界から色塗り（U または D）

これの2つの場合について、1色での色塗りの結果を用いて計算時間 \(O(n)\) で求められる。対面位置の場合は簡単であり、既存の \(x\) 軸単調領域に関する最大重み領域問題と等価であるので、省略する。障害する位置からの塗り分けの例として \(w(D \cup L)\) の最大値について考え、ピクセルグリッド平面の \((i,j)\) 成分までに制限したときの最適値 \(D(i,j)\) について考えると、\((i,j)\) 成分についての状況で、以下のように場合分けができる。

(1) \((i,j) \in D\)（図3左）
(2) \((i,j) \in L\)（図3中央）
図 3: 隣接位置からの塗り分けの場合分け

(3) \((i, j) \notin (D \cup L)\) （図 3 右）

(1) の場合,

\[
DL(i, j) = DL(i, j - 1) + D(i, j)
\]

と表すことができる。同様に、(2) の場合,

\[
DL(i, j) = DL(i - 1, j) + L(i, j)
\]

と表すことができる。以上の 3 の場合、同様に塗り分けを考えるときの基準値を算出するのに役立つ。据え置きの塗り分けの計算値は、それぞれの境目の塗り分けの基準値である。

図 4: 3 色塗り問題を解く動的計画法の実行時の分類

3.2 3 色での塗り分け

一般性を失わず、上廻り、左廻り、下廻りの 3 つの境界塗りから塗り分けの順番順に数値を挙げてみることにする。

\[
(i, j) \in L \Rightarrow j \leq LeftTop
\]

である。これに注目すると、3 色の塗り分けの部分の塗り分けの部分の計算によって、動的計画法を用いて高速に求めることができる。

\[
UDL(j) = \max \{DL(i, j - 1) + U(i, j)\}
\]

と表すことができる。したがって、結局、UDL(j) に対する基準値は、

\[
UDL(j) = \max \{DL(i, j) + UL(i + 1, j)\}
\]

と表すことができる。
FIT2009（第8回情報科学技術フォーラム）

図 5: 4色塗りのパターンの分類

\[ (i, j) \in R \Rightarrow j \geq RightTop \]
\[ (i, j) \in U \Rightarrow i \geq UTop \]
\[ (i, j) \in D \Rightarrow i \leq DownTop \]

と定義し、これら4つの値に注目すると、4色塗りは以下の3つのパターンに分類できる。
1. \( LeftTop < RightTop \) (図5左上)
2. \( DownTop < UTop \) (図5右上)
3. \( LeftTop \geq RightTop \) かつ \( DownTop \geq UTop \) (図5右下)

\( LeftTop < RightTop \) の場合

この場合、ある \( j \) 行（\( LeftTop \leq j < RightTop \)）を基準として3色塗りの部分問題に分解して考えることができる。この場合の4色塗りの最適値 \( UDLR_3 \) は、

\[ UDLR_3 = \max_{0 \leq j \leq n} \{ UDL(j) + UDR(j+1) \} \]

と表すことができる。これは、前節の結果、そこで求めまった配列 \( UDL, UDR \) を利用すると、時間計算量、空間計算量ともに \( O(N) \) で求められる。

\( DownTop < UTop \) の場合

この場合も、ある \( i \) 行（\( DownTop \leq i < UTop \)）を基準として3色塗りの部分問題に分解することができ、塗り分けの最適値 \( UDLR_2 \) は、

\[ UDLR_2 = \max_{0 \leq i \leq n} \{ DLR(i) + ULR(i+1) \} \]

と表すことができる。これも時間計算量、空間計算量ともに \( O(N) \) で求められる。

図 6: 4色塗り分け問題の2色塗り分け問題への分割

\( LeftTop \geq RightTop \) かつ \( DownTop \geq UTop \) の場合

この場合、部分問題への分割方法として以下の2つの手法を考えることができ、それぞれに従ってアルゴリズムを設計することができる。この2つのアルゴリズムの性能にはトレードオフがあるので、それぞれを取り扱うことにする。

(i) 2色塗り分けへの分割
(ii) L 字型の3色塗り分けへの分割

3.1 2色塗り分けへの分割を利用したアルゴリズム

\( UTop, DownTop, LeftTop, RightTop \) の位置と、塗り分けの数の図に注目すると、4色塗り分け問題は図6のように2つの2色塗り分け問題4色塗り問題に分解できる。また、\( UTop, DownTop, LeftTop, RightTop \) の位置を固定して考えたとき、図7のように、2色塗り分けのパターンは2つ存在する。4色塗り問題の中央部には他の色も存在するため、中央部の4色塗り問題を2つに分割したとき、各々の領域を4色塗り問題2色塗り問題に分割し、各々の分割された領域での色塗りの状況を考慮すれば、各々の領域での色塗り問題2色塗り問題に分割することができる。

したがって、この場合の4色塗り分けの最適値 \( UDLR_3 \) を求める式は以下のようになる。

\[ UDLR_3 = \max_{1 \leq RT \leq DT \leq n} \left\{ DL(DT, RT - 1) + DR(UT - 1, RT) + UL(DT + 1, LT) + UR(LT, RT + 1), DL(UT - 1, LT) + DR(DT, LT + 1) + UL(RT, RT - 1) + UR(DT + 1, RT) \right\} \]

この場合の時間計算量は、パラメータ \( RT, LT, UT, DT \) について考慮しなければならないので \( O(n^4) = O(N^4) \) である。空間計算量については2色塗り分けの結果を記憶しておくので \( O(n^2) = O(N) \) となる。\( LeftTop < RightTop, DownTop < UTop \) の場合も含めると、結果4色塗り分けの最適値 \( UDLR_3 \) は、

\[ UDLR = \max\{ UDLR_1, UDLR_2, UDLR_3 \} \]

で与えられる。合計の時間計算量は \( O(n^4) = O(N^4) \)、空間計算量は \( O(n^2) = O(N) \) となる。

(第1分冊)
3.2 L字型の3色塗り分けへの分割による高速化アルゴリズム

LeftTop ≥ RightTop かつ DownTop ≥ UpTop の場合の計算時間の高速化を考える。塗り分けの解に注目すると、図8のように2つのL字型の3色塗り分けに分割することができる。この性質に注目して高速化アルゴリズムを考える。

まず、L字型の3色塗り分けについて考える。図9のように、L字型はULR，ULR，ULR，ULR，ULRの4パターンに分けられる。それぞれのパターンに対して、パラメータs，t，u を図9のように定義する。例えばしてDLR(s,t,u)について述べると、

- s:左側の高さ
- t:右側の高さ
- u:高さがsである、最も右の位置

DLR(s,t,u):図のL字型に対する3色塗り分けの最適値となる（ただし、s>t）。DLR(s,t,u)を求める漸化式を考える。DownTopに注目すると、0≤DownTop≤s-1の場合とDownTop=sの場合が考えられる。0≤DownTop≤s-1の場合、

\[ DLR(s,t,u) = DLR(s-1,t,u)+L(s,u) \]

が成り立つ。DownTop=sの場合は、集合Rに対するプロッカーに注目し、2色塗り分け問題に分割して考えることができる（図10）。集合Rに対するプロッカーは、集合Dの高さをt以上とする。最も右の列jであり、

\[ b = \max \{ j | (i,j) \in D \} \ (i \geq t) \]

で表され、j≤bである(i,j)成分は集合Rの要素となることはできない。また、常にb＜uが成り立つ。プロッカーbに注目すると、

\[ DLR(s,t,u) = \max_{0\leq b<s} \{ DL(s,b)+DR(t,b+1) \} \]

が成り立つ（図10）。さらに、b＜uの場合について、図10上の空白領域に注目すると、

\[ DLR(s,t,u) = \max_{0\leq b<u} \{ DL(s,b)+DR(t,b+1) \} \]

が成り立つ。2つの場合をまとめて、DLR(s,t,u)を求める漸化式はs>tに対して、

\[ DLR(s,t,u) = \max \{ DLR(s-1,t,u)+L(s,u), \]

\[ DLR(s,t,u-1), \]

\[ DL(s,u)+DR(t,u+1) \} \]

で与えることができる。ここで、初期条件はs=tの場合であり、

\[ DLR(t,t,u) = DLR(t) \]

で与えられる。これは、パラメータs，t，uについて計算、記憶し、前もって求めておいた2色塗り分けのテーブルを利用すると、
テープル $DLR_L$ の要素を含む $O(1)$ で計算を行えるので、時間計算量、空間計算量ともに $O(n^2) = O(N^{1.5})$ である。
同様にテープル $DLR_R$, $DLR_{ULR}$, $ULR_R$, は計算することができる。これらを用い、L 字型分割を組み合わせて $UDLR_R$ を求める。L 字型の組み合わせパターン2つについて考えると (図11), $UDLR_R$ は以下の式で与えられる。

$$UDLR_R = \max \left\{ \begin{array}{l}
\max_{0 \leq s \leq s' \leq n} \{ DLR_L(s, t, u) \\
+ ULR_R(s + 1, t + 1, u + 1) \}, \\
\max_{0 \leq s \leq s' \leq n, 1 \leq u \leq n + 1} \{ DLR_R(s, t, u) \\
+ ULR_L(s + 1, t + 1, u - 1) \} \right\}
\right.$$

したがって、$UDLR_R$ を時間計算量、空間計算量ともに $O(n^2) = O(N^{1.5})$ で求める事ができる。

以上をまとめると、下記の定理を得る。

定理2 4つ以上の境界辺からの基準面領域に分割できる領域の中で、重み和を最大化するものを $O(n^{1.5})$ 時間で $O(N^{1.5})$ の作業領域を用いて計算することができる。また、$O(N)$ の作業領域を用いて $O(N^2)$ 時間で計算することができる。

4 結論と拡張

本論文では、4つの方角からの基準面領域の非交差和の最適切り出し問題が効率的に解けることを示した。理論的には、交差を許した場合の領域の最適切り出しの NP 困難性と比較すると、計算量が何倍か節約の関連の解明として興味深い成果と考えられる。応用面においても、たとえば、放射線診断における Interleaf collimator は現在では左右2方向のみからの基準面領域の非交差和を遮蔽する領域としている。医療機器としての Interleaf collimator の耐故障性 (leafの衝突による破損回避) の問題で、現在では左右2方向であるが、原理的には4方向からの遮蔽を行う機器の開発も可能であり、本論文の成果はそのよう