

GPUによる並列処理に適した ボリュームセグメンテーション

小林亮介[†] 安藤英俊[†]

本論文では、GPUによる並列処理に適した高速なボリュームセグメンテーション方法を提案する。GPUの高速な計算能力を用い、またその並列処理に適した手法を用いることにより、時間がかかるボリュームのセグメンテーションを高速に行うことが可能である。具体的手法として、ウェーブレット変換を用いた局所テクスチャ特徴量の算出を行い、特徴の類似を元にボリュームデータを分割する。セグメンテーションを実際に行うために熱伝導の考え方を用いる。具体的には、熱の定常状態を表すラプラス方程式を反復解法で近似する。また、本手法によるMRIデータのセグメンテーションの結果を示す。

A Volume Segmentation Algorithm Suited for Parallel Processing on GPU

Ryosuke Kobayashi[†] and Hidetoshi Ando[†]

We propose a fast volume segmentation algorithm suited for parallel processing on GPU. Although segmentation of volume data usually needs much computation time, it can be accelerated by using the parallel computation power of GPU, by using a highly parallel algorithm. We use a wavelet-based local texture feature to characterize the region of interest and segment the volume based on the similarity the features. We model the similarity-based segmentation process as solving thermal conduction problem. We show the result of our method through an example of volume segmentation of MRI images by numerically solving Laplace's equation using iterative methods on GPU.

[†]山梨大学
University of Yamanashi

1. はじめに

現在、コンピュータを用いて立体を加工する技術（ボリュームモデリング）は医療の分野をはじめ、さまざまな分野で使われ始めている。例えば、手術をする前にコンピュータ上での切断、取り出し等をシミュレートする手術シミュレーション等がある。CGによるボリュームモデリングによって、実際は一度しか行えないような作業や、作業を行うことが難しい事柄に対するモデリングをシミュレートすることができる。

ボリュームモデリングの例を図1に示す。図1は、りんごのボリュームモデルの切削の様子を示す [1]。



図1 りんごのモデルの切削[1]

Figure 1 Cutting the apple [1].

2. 研究背景

2.1 セグメンテーション

ボリュームモデリングには削る以外にも様々な方法があり、その中にセグメンテーションがある。セグメンテーションとは、立体の中の一部を他から分割することである。立体、特に人体などの複雑な立体の一部、例えば心臓を他と分割するためには、どこからどこまでが心臓なのかという専門的な知識が必要である。また、実際に手で複雑な人体モデルから心臓のみを切り離すためには時間や慣れなども必要になる。そのため、コンピュータによる自動的なセグメンテーションというものが必要であるとされ、様々な研究が行われている。

2.2 ボクセル

また、人体のような複雑な立体を表現するためには、物体の表面だけを描写するモデル（サーフェイスモデル）ではなく、その内部表現まで表現できる立体（ソリッドモデル）が必要になってくる。ソリッドモデルの表現は様々だが、その中でも人体の

ような複雑な立体を表現するのに適した表現方法は、ボクセルを用いた方法である。ボクセルとは、2次元でのピクセルのようなものでありひとつひとつがブロック状のものである。そのブロックが積み重なった集合で物体を構成し、内部が複雑な物体も表現することができる。図2に、ボクセルを用いた立体の例を示す[2]。このような複雑な立体もボクセルを用いれば表現できる。しかし、人体のような複雑なボリュームに対しボクセルを用いた立体の表現を行うと、それだけボクセルの量が増え、モデリングを行うためにそのボクセル一つ一つに対して数値計算等の処理を行わなければならない。そのため処理が重くなってしまうという欠点がある。そのため、ボリュームのモデリングにはGPUを使うことが多い。

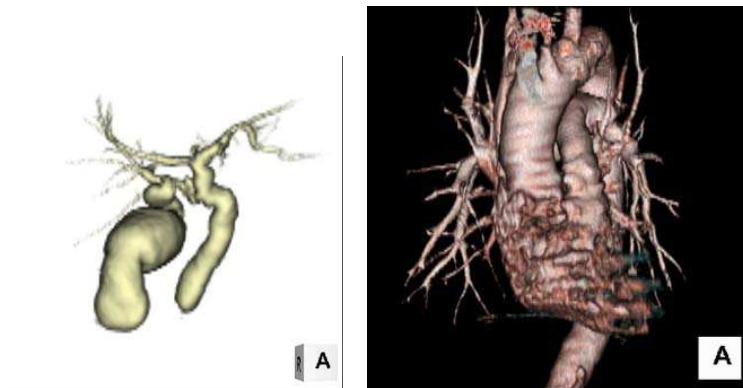


図1 ボクセルによる立体表現[2]
Figure 1 Cube representation with voxel[2].

2.3 GPU

GPUとは、高速な描画のための画像処理装置のことである。GPUは描画に並列処理を行うため、処理効率が良いという特徴を持つ。近年では、GPUの演算能力が飛躍的に向上し、また、画像処理演算の自由度も増してきた。そのため、GPUを画像処理演算以外の汎用計算のために使うというGPGPUの技術が注目されている。

本研究では、GPUを利用し、その並列処理に適した計算方法を用いたセグメンテーションの手法を提案する。

3. 先行研究

GPUを用いない最近のセグメンテーション手法としては、Graph Cuts や Belief

Propagationなどが挙げられる。これらの方法は、ユーザが分割したい領域とそのほかの領域の2つをおおまかに指定し、分割計算を行うものである。Graph Cutsは、それぞれのピクセル（ボクセル）に対してユーザが指定した2つの領域のどちらとつながりが強いを表すt-linkと、ピクセル（ボクセル）間のつながりの強さを表すn-linkのグラフを作成し、そのあとグラフの最小カットを行い分割する手法である。また、Belief Propagationは、ユーザが指定した2つの領域のそれぞれの「らしさ」の情報を周りの画素に伝播し、それぞれの画素がどちらに近いかを確率で求める。しかし、Graph Cutsはグラフの最小カットを行う際に逐次処理を行うので並列処理には向かず、Belief Propagationはノードがループしてしまうと遅くなってしま（画像では画素間をノードでつなぐため、ループがいたるところで起こる）という問題がある。

また、GPUを用いたボリュームセグメンテーションの先行研究としてlevel-set methodsやnonlinear diffusion metricをベースにした方法[3]、[4]がある。この方法は、3次元ボリュームデータから取り出したい箇所を指定し、指定点を決められた境界に達するまで広げることによりある領域を取り出すことができるというものである。図3右はlevel setを用いた方法による脳の取り出し結果を示す。左にnonlinear diffusion metricを用いた方法による大動脈瘤（大動脈の一部がこぶのように膨らむ症状）の取り出し結果を示す。このようにセグメンテーションを行うと、取り出された部分以外の情報がないので、その部分の詳細な情報が見やすくなっている。

しかし、これらの手法は分割を行うために色（輝度）情報のみで分割を行うため、色で区別がつかない境界に対しては十分なセグメンテーションが出来ない可能性がある。

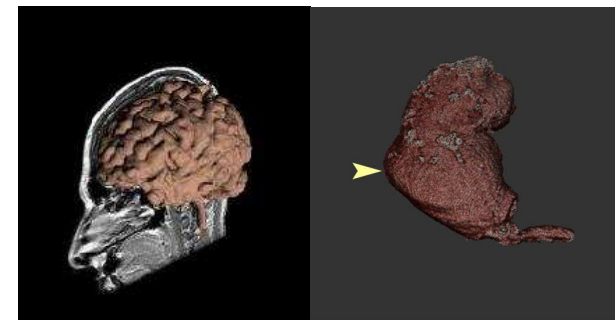


図3 先行研究でのセグメンテーション結果[3][4]
Figure 3 The result of segmentation with previous work[3][4].

4. 提案手法

本研究では、色で区別が出来ない境界に対しても対応できるように、色以外の情報も用いて、GPUの並列処理に適したボリュームセグメンテーションを行う手法を提案する。

色の差が出ない場合でも有効なセグメンテーション手法を行うためには、色以外の画像の特徴(特徴量)を定義する必要がある。本研究では、ウェーブレット変換による局所テクスチャ特徴を用い、特徴量の算出を行った。また、GPUの並列処理に適したセグメンテーション手法には熱伝導を用いた。

本研究での具体的なセグメンテーション計算の流れは、

- (1) ウェーブレット変換
 - (2) 熱伝導率の計算
 - (3) 熱伝導の計算
- となる。

5. ウェーブレット変換による局所テクスチャ特徴の算出

5.1 局所テクスチャ特徴

局所テクスチャ特徴とは、その場所がどのようなテクスチャの集まりなのかを表した特徴である。テクスチャとは、まだら模様、縦縞、横縞、その他様々の模様のことであり、物体によっていろいろな模様を持つ。図 5.1 に局所テクスチャ特徴の具体的な例を示す。例えば、図のそれぞれの赤い図形がそれぞれ水玉模様や縦縞を形成している一部だとする。局所テクスチャ特徴とは、その図形がどのようなテクスチャの一部であるかを示したものである。

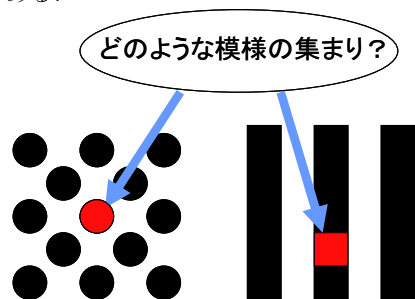


図 5.1 局所テクスチャ特徴
Figure 5.1 Local texture feature.

5.2 ウェーブレット変換

本研究では、Haar Wavelet を用いて局所テクスチャ特徴を定義する。この手法を画像に対して用いると、画像を高周波成分と低周波成分の二つに分けることが出来る。またこの手法は、福田ら[5]による研究により、セグメンテーションに対して有効であると証明されている。図 5.2 に 2 次元画像におけるウェーブレット変換の結果を示す。2 次元画像の X 方向と Y 方向にウェーブレット変換を行った場合、図 5.2 真ん中(Level 1)のように低周波数成分(L)と高周波数成分(H)に分解される。右側の左上に低周波数成分(LL), 右上に X 方向の高周波数成分(HL), 左下に Y 方向の高周波数成分(LH), 右下に XY (斜め) 方向の高周波数成分(HH)が現れる。また、Level 1 の LL に対してウェーブレット変換を行うことで、図 5.2 右(Level 2)のように、LL 成分に対する LL,HL,HH,LH 成分を求めることができ、特徴量を増やすことができる。

これらの情報を元画像のそれぞれの画素に対応させることにより、特徴量を定義することが出来る。図 5.2 は 2 次元画像に対するウェーブレット変換であるが、3 次元の場合は、Z 方向に対してもウェーブレット変換を行うため、図 5.3 のようにさらに特徴量が増える。

Haar Wavelet の分解アルゴリズムを式 (5.1), (5.2) に示す。

$$c_k^{(n)} = \frac{1}{2} (c_{2k}^{(n-1)} + c_{2k+1}^{(n-1)}) \quad (5.1)$$

$$d_k^{(n)} = \frac{1}{2} (c_{2k}^{(n-1)} - c_{2k+1}^{(n-1)}) \quad (5.2)$$

式 (5.1) によって低周波数成分、式 (5.2) によって高周波数成分が得られる。この計算を、XYZ 方向に対して行う。

ウェーブレット変換で得られたそれぞれの特徴量は、熱伝導率を求める際に用いられる。

6. 熱伝導によるセグメンテーション

セグメンテーションの最近の方法として、GraphCuts や Belief Propagation など、さまざまな方法が研究されている。しかし、GraphCuts は逐次処理を行うので並列処理には向かず、Belief Propagation は画像に対して処理が遅くなるという問題がある。本研究では、並列処理に適した手法として、熱伝導によるセグメンテーション方法を提案する。その手法を以下に示す。

まず、それぞれのボクセルをある温度を持った空間であり、ボクセル間には熱を伝導させる物質があると考える。図 6.1 にその考え方を示す。

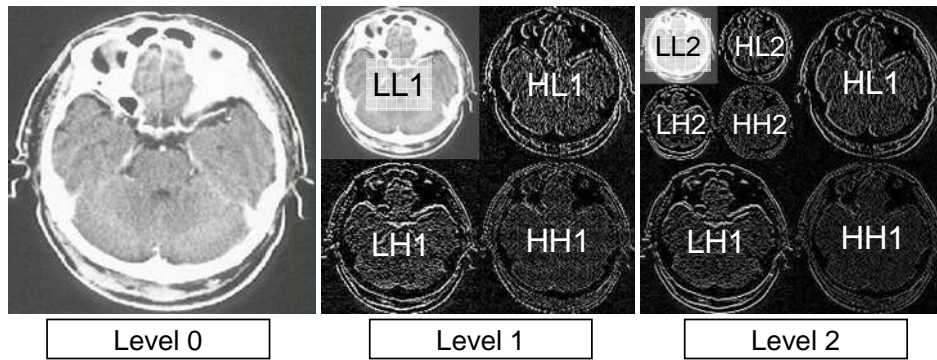


図 5.2 2D ウェーブレット変換
 Figure 5.2 2D wavelet transform.

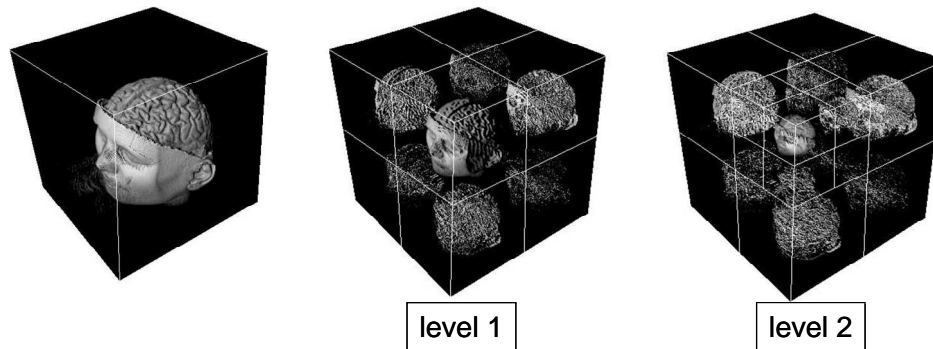


図 5.3 3D ウェーブレット変換
 Figure 5.3 3D wavelet transform.

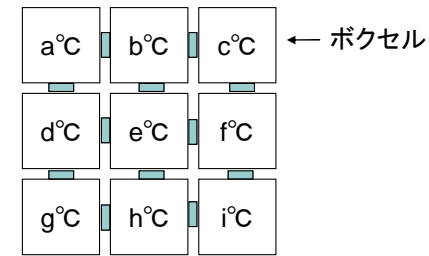


図 6.1 熱とボクセルの関係
 Figure 6.1 Relation of heat and pixel.

また、熱の性質である、

- 温度は高いほうから低いほうへ流れる
- 熱伝導率があり、熱が伝わりやすい物質と伝わりにくい物質がある
- 最終的に熱は安定する

を考える。ある箇所における熱伝導物質の伝導率が図 6.2 (a) ようである場合を考える。熱伝導率が高いところは熱が伝わりやすく、低いところは熱が伝わりにくい。そのため、この時熱源を図 6.2 (b) のように設置した場合、最終的には図 6.2 (c) のような状態で安定するだろうと考えた。

セグメンテーションしたい境目の熱伝導率が低いと定義すれば、定常状態のときにモデルを二つの領域に分けることができる。

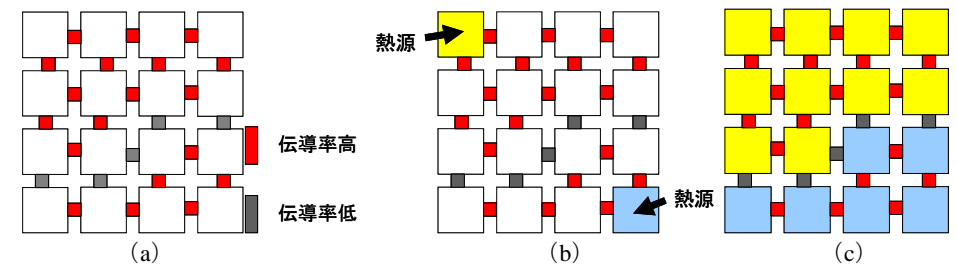


図 6.2 熱伝導によるセグメンテーション
 Figure 6.2 Segmentation with heat conduction.

6.1 ユーザによる領域指定

熱伝導を行う前に、ユーザに分割したい領域とそうでない領域のおおまかな指定を行ってもらふ。ユーザが指定した領域が熱源となる。図 6.3 に領域指定の様子を示す。青い部分が分割したい領域、赤い部分がそうでない領域になる。

6.2 熱伝導率の計算

X,Y,Z 方向で隣り合うボクセル間の熱伝導率は、特徴量の差を利用する。それぞれのボクセルに対して、X 方向の両隣、Y 方向の両隣、Z 方向の両隣との合計 6 つの特徴量の差を求め、ボクセル周辺に対する熱伝導率を計算する。

熱伝導率の計算過程を図 6.4 に示す。特徴量にはウェーブレット変換で求めた局所テクスチャ特徴を用いる。それぞれの局所テクスチャ特徴に対して差を求め、得られた結果を足し合わせる。また、特徴量の差が大きいほど熱伝導率を低くするため、得られた結果の反比例をとって反転させそれを最終的な熱伝導率とする。

6.3 熱伝導の計算

熱伝導の計算を行うためには熱の定常状態を表すラプラス方程式を用いる。ラプラス方程式を式 6.1 に示す。

$$\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} + \frac{\partial^2 u}{\partial z^2} = 0 \quad (6.1)$$

この式は温度分布に変化がなくなった状態（定常状態）を表している。本研究では、このラプラス方程式を n 元連立一次方程式に変換[6]し、反復法で適当な値を持ったデータをラプラス方程式の定常状態へ近似計算する。今回の場合、 n 元連立一次方程式は $n \times n \times n$ の画像データのことで、適当な値を持った画像データはユーザによって指定された熱源である。これをラプラス方程式の解に近似した結果にすることで最終的には定常状態での温度分布を導ける。

図 6.5 に、反復解法でラプラス方程式の定常状態の解に近似していき、熱伝導が行われている様子を示す。熱源は 0°C と 100°C にし、それ以外の場所は温度なしとした状態を最初の値として、熱の計算を行っている（赤が 50°C 以上で青が 50°C 以下）。また、熱伝導率を用いて、熱の伝導の速さを表現する。

熱が安定したら、分割された領域を取り出すことによってセグメンテーションを行う。

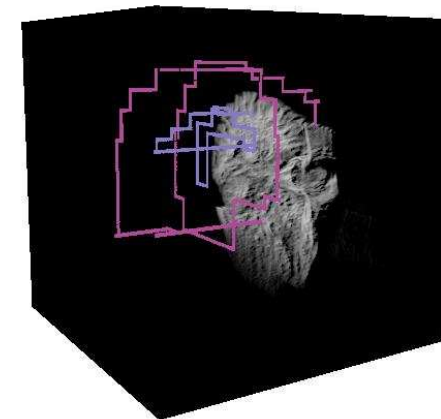


図 6.3 熱源の指定

Figure 6.3 Designate the heat source.

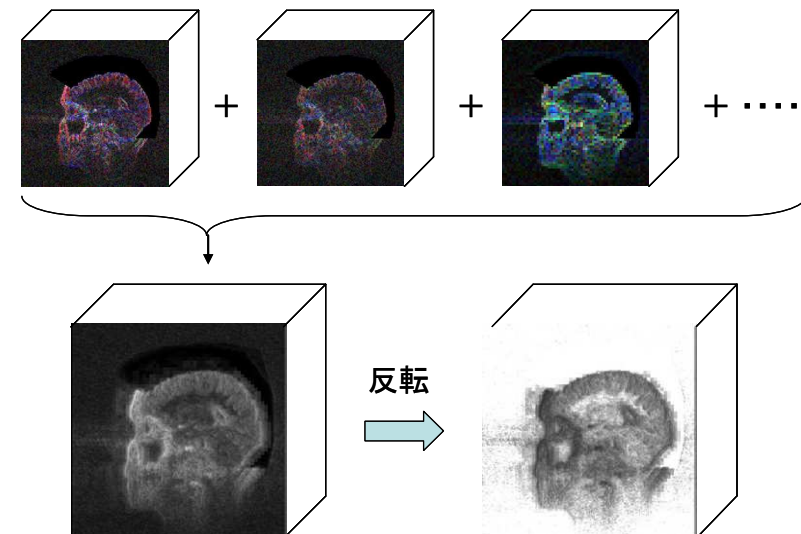


図 6.4 熱伝導率の計算

Figure 6.4 Computation of heat conductivity.

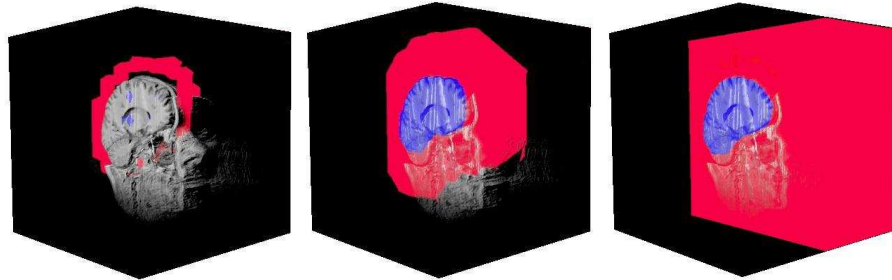


図 6.5 反復法による計算の様子
Figure 6.5 Computation with iteration method.

7. 結果と考察

MRI ボリュームモデルに対するセグメンテーションの結果を図 7.1 に示す。DirectX10, GPU:GeForce 8800GTX, データサイズ $256 \times 256 \times 128$ で実行した結果, 図 7.1 左のようなボリュームモデルから図 7.1 右のように脳だけ切り出されたボリュームモデルの表示を約 30 秒で行うことができた。また, 2 次元 CT, MRI 画像に対してもセグメンテーションを行った。結果を図 7.2 に示す。

全体的に分割したい領域をうまく分割することが出来ているが, 境界であるべきところで熱が漏れてしまったり, 鋭角であるべきところが丸くなってしまう場合が多かった。これは, 境界での熱伝導率がうまく低くならなかった場所があるためであると考えられる。そのため熱伝導率の計算の再検討か, もしくはユーザが熱伝導率の調整を行えるようにすることで改善すると考えられる。

このように, 正確なセグメンテーションを行えない場合もあったが, 今回の方法は熱源をいつでも追加し, 温度分布結果を変えることができるので, 簡単に修正することができる。

8. まとめと今後の課題

本研究では, ボリュームデータに対するセグメンテーションを行った。GPU とその並列処理に適した手法で計算を行った結果, まだ課題はあるが, 適切なセグメンテーションを行うことができた。また, 本研究での反復法はヤコビ法を用いたが, 同じ反復法であり高速な計算が行える共役勾配法などを用いればさらに高速なセグメンテーションを行えると考えられる。

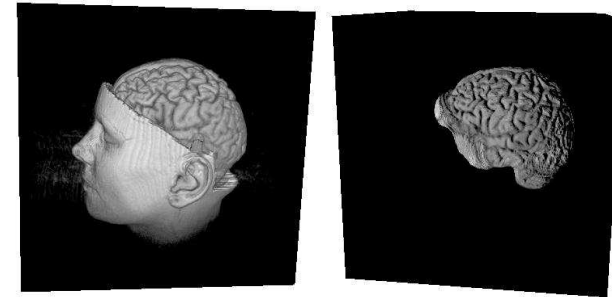


図 7.1 セグメンテーション結果
Figure 7.1 Result of segmentation.

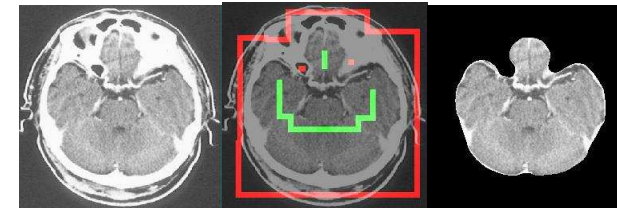


図 7.2 : 2 次元データでの出力結果
Figure 7 Result of 2D data.

9. 参考文献

- 1) 平賀正基, 安藤英俊, “多層深度マップによるリアルタイムボリュームモデリング”, Visual Computing/グラフィクスと CAD 合同シンポジウム 2007 予稿集, pp.269-274, (2007)
- 2) Volume Rndering, <http://www.asahi-net.or.jp/~tv4m-tkhs/VR/VR.html>
- 3) Lefohn,A., Kniss,J., Hansen,C., Whitaker, R. : Interactive Deformation and Visualization of Level Set Surfaces Using Graphics Hardware, (2003).
- 4) Sherbondy,A., Houston,M., Napel,S. : Fast Volume Segmentation With Simultaneous Visualization Using Programmable Graphics Hardware, (2003).
- 5) 福田恵太, 滝口哲也, 有木康雄, “画像セグメンテーションにおけるウェーブレット係数の局所テクスチャ特徴量を用いた GraphCuts”, 電子情報通信学会技術研究報告. PRMU, パターン認識・メディア理解, Vol.98, No.384(20071206) pp. 19-24. (2007)
- 6) 山崎郭滋, 偏微分方程式の数値解法入門, p.131, 森北出版. (1993)