



Title	三次元画像データの領域分割に関する研究
Author(s)	菊池, 隆文; Kikuchi, Takafumi; 坂本, 雄児 他
Citation	北海道大學工學部研究報告, 141, 137-153
Issue Date	1988-07-29
Doc URL	https://hdl.handle.net/2115/42112
Type	departmental bulletin paper
File Information	141_137-154.pdf



三次元画像データの領域分割に関する研究

菊池 隆文 坂本 雄児 青木 由直
(昭和63年3月31日受理)

Research on Regional Extraction of Three-Dimensional Image Data

Takafumi KIKUCHI, Yuji SAKAMOTO and Yoshinao AOKI
(Received March 31, 1988)

Summary

Recently with the development of CT scanner and MRI technique three-dimensional image data have become increasingly available. With the development of 'under' snow radar technique, it has become possible to acquire three-dimensional data of targets in accumulated snow. However, we are lacking in systems by which we can display three-dimensional data accurately. We have been conducting research work which may enhance the display of three-dimensional imaging. We have attempted to extend FIR filter for three-dimensional imaging. We are working on a three-dimensional image processing including edge extraction. Following edge extraction, the processing of three-dimensional segmentation is greatly enhanced. The selection of outstanding data from the source will make it possible to display a three-dimensional image more readily. Three-dimensional FIR filtering is reported in the first section. In the second section three-dimension domain partition and a method by which compression of three-dimensional image data are reported.

1. 序 論

従来画像とは二次平面上のデータをさし、画像処理もそれを対象にした二次元のものが大半であった。二次元の画像処理、特にデジタルの画像は、コンピュータの発達とデジタル画像の需要の伸びとともに様々な技法が開発されてきている。

近年では、データ採集技術の発達とともに、特に医療関係のCTスキャナやMRI装置、ホログラフィックレーダ技術等、採取されるデータが二次元データにとどまらず、三次元としての属性を備えるものが現れてきた。いわゆる三次元画像である。

しかし、三次元の画像データを採取することができるようになって、現在のところ三次元画像をうまく表示する事のできるディスプレイシステムはまだ多くない¹⁾。現在のところ三次元画像データを表示する方法は、プログラムを用いて立体的に表示するかもしくは三次元データを二

次元のスクリーンに投影して表示する方法が主である。しかし、双方の場合とも物体の表面を表示する場合には特に問題はないが、立体の内部を表示する場合は、断面を表示するか立方体や円柱の形で部分を抜き出して表示する、もしくは透視像を表示する方法が主であり画像データの持つ情報の多くが失われてしまう。またこれらの表示方法では、希望の領域を表示するためには、多くの画面の切り替えを必要とする。

このようにデータの採取が可能になっても、表示システムの不足から三次元の画像システムはまだ一般的とは言えない状況にある。

従来著者らは三次元画像データを採取可能な雪中レーダシステム²⁾や、三次元画像データを高速に二次元画像に変換して表示することのできる、三次元メモリシステムの研究を行って来ている。雪中レーダシステムは、データを採取しそのデータを三次元メモリシステム等で表示する段階まで完成している。そこで画像処理部分の次の段階として、

1. 採取された積雪の三次元画像データ中の埋没物体の計数。
2. 特定のターゲットの画像データ中からの抽出。
3. 抽出されたターゲットの識別。

などの処理を目指している。

そのため、三次元画像処理技法の研究に着手し、三次元画像データから特定の部分を取り出すため、領域の分割と抜き出しについて新しい方式を提案し、実験でその有効性を確かめている。

2. 三次元フィルタリング

本報告の研究において、まずデジタルフィルタの三次元化について検討を行った。デジタルフィルタリングは、デジタル画像処理の中でも、最も基本的なもののひとつであり、画像処理そのものから、他の画像処理の下準備としてまで様々な用途で用いられている。

2.1 デジタルフィルタリング

ここで必要とされている画像処理においては対象は、すでにデジタルメモリに落とされた離散化されたデータである。したがって、フィルタリングはデジタルフィルタを用い、また専用のハードウェア回路は用いず、パーソナルコンピュータ上のソフトウェアにて行った。

フィルタリングの実現には二次元のフィルタリングにおいてもよく用いられる、加重マトリクスを用いた方法を採用した³⁾。これは、中心画素とその近傍にのみフィルタを作用させる方法であり、通常 3×3 もしくは 5×5 などのものがよく用いられる。

理想的には、フィルタリングは全空間領域に対して行うべきであるが、高性能なコンピュータを用いても一点に対するフィルタリングにおいて、全空間領域を考慮に入れることは膨大な計算量を考えると不可能である。そこでマトリクスの大きさの決定には窓関数の考え方をを用い、中心画素から遠い画素についてはその影響を考えないものとし、中心画素とその近傍にのみフィルタを作用させる。実際には、中心画素とその26近傍にフィルタを作用させる $3 \times 3 \times 3$ の加重マトリクスを用いた。

ここで小さなマトリクスを用いるということは、急激な空間周波数の変化にしか対応できないことを意味する。したがって、 $3 \times 3 \times 3$ のマトリクスを用いたフィルタリングでは、画像全体に渡るような緩やかな変化にはあまりよく対応できないことになる。しかし、画像の辺縁の処理や先に述べたような全体の計算時間などの問題点から、あまり大きなマトリクスを採用すること

は問題点が多い。

2.2 使用したフィルタ

FIR フィルタの実験は、エッジの抽出を主眼に行った。二次元の画像処理におけるエッジ抽出にはゾーベルオペレータ (Sobel operator) やラプシアン (Laplacian) などのフィルタがよく用いられる。そのためこの実験においても、これらのフィルタを三次元に拡張し、双方の結果を比較した。

2.2.1 ラプラシアン

ラプラシアンは二次微分のフィルタで、三次元の場合、

$$\nabla^2 f \equiv \frac{\partial^2 f}{\partial x^2} + \frac{\partial^2 f}{\partial y^2} + \frac{\partial^2 f}{\partial z^2} \quad (2.2.1.1)$$

と表され、三次元のデジタル画像に対するラプラシアンは次のように表される。

$$\begin{aligned} \nabla^2 f(i, j, k) &\equiv \Delta_x^2 f(i, j, k) + \Delta_y^2 f(i, j, k) + \Delta_z^2 f(i, j, k) \\ &\equiv [\Delta_x f(i+1, j, k) - \Delta_x f(i, j, k)] + [\Delta_y f(i, j+1, k) \\ &\quad - \Delta_y f(i, j, k)] + [\Delta_z f(i, j, k+1) + \Delta_z f(i, j, k)] \\ &\equiv [f(i+1, j, k) - f(i, j, k) - \{f(i, j, k) - f(i, j-1, k)\}] \\ &\quad + [f(i, j+1, k) - f(i, j, k) - \{f(i, j, k) - f(i, j-1, k)\}] \\ &\quad + [f(i, j, k+1) - f(i, j, k) - \{f(i, j, k) - f(i, j, k-1)\}] \\ &\equiv [f(i+1, j, k) + f(i-1, j, k) + f(i, j+1, k) + f(i, j-1, k) \\ &\quad + f(i, j, k+1) + f(i, j, k-1)] - 6f(i, j) \end{aligned} \quad (2.2.1.2)$$

したがって、三次元のラプラシアンの加重マトリクスは図 2.2.1 のようになる。

$k = -1$	$k = 0$	$k = 1$																											
<table border="1" style="border-collapse: collapse; text-align: center;"> <tr><td>0</td><td>0</td><td>0</td></tr> <tr><td>0</td><td>-1</td><td>0</td></tr> <tr><td>0</td><td>0</td><td>0</td></tr> </table>	0	0	0	0	-1	0	0	0	0	<table border="1" style="border-collapse: collapse; text-align: center;"> <tr><td>0</td><td>-1</td><td>0</td></tr> <tr><td>-1</td><td>6</td><td>-1</td></tr> <tr><td>0</td><td>-1</td><td>0</td></tr> </table>	0	-1	0	-1	6	-1	0	-1	0	<table border="1" style="border-collapse: collapse; text-align: center;"> <tr><td>0</td><td>0</td><td>0</td></tr> <tr><td>0</td><td>-1</td><td>0</td></tr> <tr><td>0</td><td>0</td><td>0</td></tr> </table>	0	0	0	0	-1	0	0	0	0
0	0	0																											
0	-1	0																											
0	0	0																											
0	-1	0																											
-1	6	-1																											
0	-1	0																											
0	0	0																											
0	-1	0																											
0	0	0																											

図 2.2.1 三次元ラプラシアン

2.2.2 ゾーベルオペレータ

ゾーベルオペレータは、画像中の勾配を求めるフィルタの一種であり、一次の偏微分が最大値をとる方向とその大きさをとる。三次元画像の勾配の大きさと向きを求めるために次の式を用いる。

$$\max(|\Delta_x f(i, j, k)|, |\Delta_y f(i, j, k)|, |\Delta_z f(i, j, k)|) \quad (2.2.2.1)$$

さらに、雑音に強いエッジ検出オペレータを作るためには、注目する画素の近傍領域内の濃度の和や平均値の差を計算するとよい。そこで、 $\Delta_x f(i, j, k)$ 、 $\Delta_y f(i, j, k)$ 、 $\Delta_z f(i, j, k)$ を次のように定義する。

さらにこの式に、斜め方向の変化を加味したものが、Sobel オペレータである。

このオペレータは、X 成分用、Y 成分用、Z 成分用をそれぞれ画像に作用させ、出力の絶対値

の大きいものを結果として採用する。

$$\begin{aligned} \Delta_x f(i, j, k) \equiv & f(i-1, j-1, k-1) + f(i-1, j, k-1) + f(i-1, j+1, k-1) \\ & + f(i-1, j-1, k) + f(i-1, j, k) + f(i-1, j+1, k) + f(i-1, j-1, k+1) \\ & + f(i-1, j, k+1) + f(i-1, j+1, k+1) - (f(i+1, j-1, k-1) \\ & + f(i+1, j, k-1) + f(i+1, j+1, k-1) + f(i+1, j-1, k) + f(i+1, j, k) \\ & + f(i+1, j+1, k) + f(i+1, j-1, k+1) + f(i+1, j, k+1) \\ & + f(i+1, j+1, k+1)) \end{aligned} \quad (2.2.2.2)$$

$$\begin{aligned} \Delta_y f(i, j, k) \equiv & f(i-1, j-1, k-1) + f(i, j-1, k-1) + f(i+1, j-1, k-1) \\ & + f(i-1, j-1, k) + f(i, j-1, k) + f(i+1, j-1, k) + f(i-1, j-1, k+1) \\ & + f(i, j-1, k+1) + f(i+1, j-1, k+1) - (f(i-1, j+1, k-1) \\ & + f(i, j+1, k-1) + f(i+1, j+1, k-1) + f(i-1, j+1, k) + f(i, j+1, k) \\ & + f(i+1, j+1, k) + f(i-1, j+1, k+1) + f(i, j+1, k+1) \\ & + f(i+1, j+1, k+1)) \end{aligned} \quad (2.2.2.3)$$

$$\begin{aligned} \Delta_z f(i, j, k) \equiv & f(i-1, j-1, k-1) + f(i, j-1, k-1) + f(i+1, j-1, k-1) \\ & + f(i-1, j, k-1) + f(i, j, k-1) + f(i+1, j, k-1) + f(i-1, j+1, k-1) \\ & + f(i, j+1, k-1) + f(i+1, j+1, k-1) - (f(i-1, j-1, k+1) \\ & + f(i, j-1, k+1) + f(i+1, j+1, k+1) + f(i-1, j, k+1) + f(i, j, k+1) \\ & + f(i+1, j+1, k+1) + f(i-1, j+1, k+1) + f(i, j+1, k+1) \\ & + f(i+1, j+1, k+1)) \end{aligned} \quad (2.2.2.4)$$

こうして得られた三次元のゾーベルオペレータは図 2.2.2 のようになる。

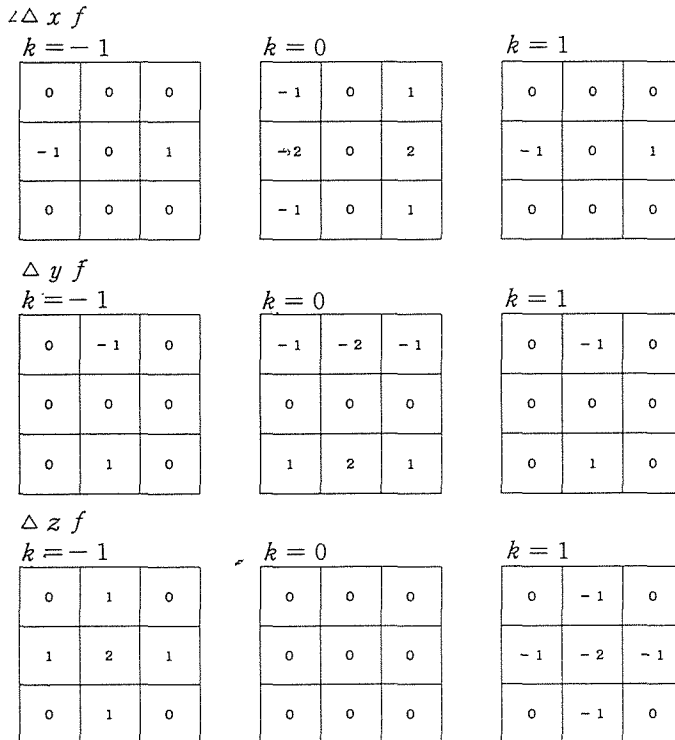


図 2.2.2 三次元ゾーベルオペレータ

2.3 フィルタの比較

前述の三次元 FIR フィルタを、16ビットのパーソナルコンピュータおよび高速 IP (Image Processor) 上で、ソフトウェア的に実現した。

ラプラシアンとゾーベルオペレータの双方を三次元に拡張したが、それを実際に用いて性能の比較を行った。

比較に用いたデータは、CT スキャナの出力を一度ネガフィルムにしたものをビデオカメラを用いてデジタル化したもので、像は人体の胴体部分の横断面である。データは一面素あたり256階調、画像の大きさは、X 方向に256ドット、Y 高向に155ドット、Z 方向は、1cm おきに15枚である。

ここで XY 方向には画素間の距離はほぼ等高的であるのに対し、Z 方向に関しては、XY 方向に比較してはるかに大きな値となっている。そのため Z 方向に関しては、データの補間を行っている。

写真 2.3.1 はソースデータ中の一枚の XY プレーンを抜きだしたもので、写真 2.3.2,

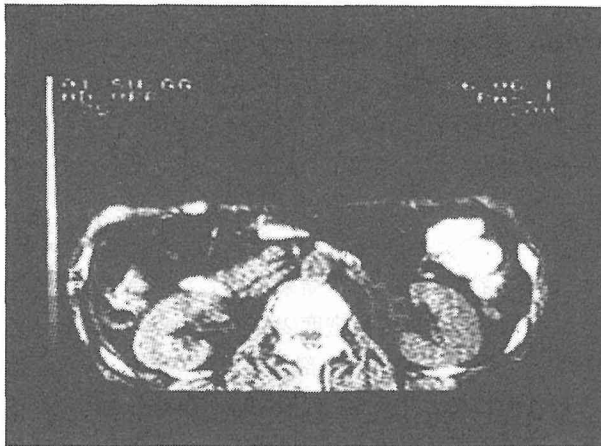


写真 2.3.1 人体の CT 画像

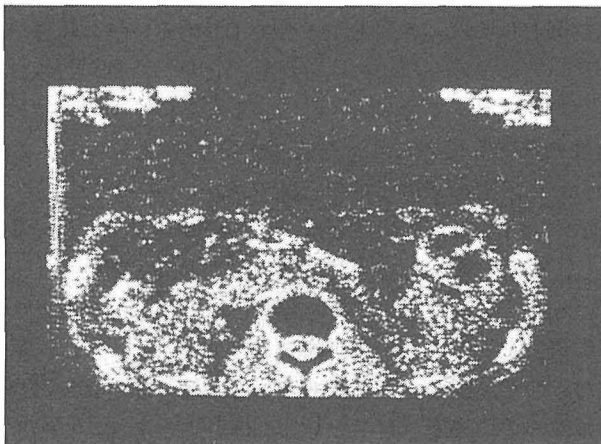


写真 2.3.2 ラプラシアンの出力

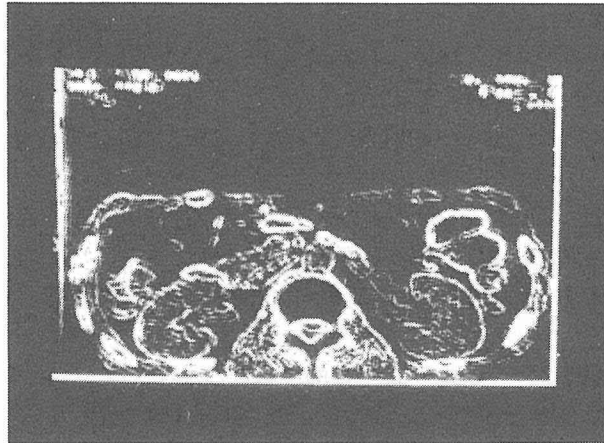


写真 2.3.3 ゾーベルオペレータの出力

2.3.3はラプラシアンおよびゾーベルオペレータを作用させてエッジを取り出した結果から、ソースの写真に対するXYプレーンを抜き出したものである。

写真を見てもわかるとおり、ラプラシアン、ゾーベルオペレータの双方の場合において、画像中のエッジが抽出できることがわかる。しかし、ゾーベルオペレータを作用させた結果に比較して、ラプラシアンを作用させたものは、画像中のノイズに強く反応してノイズによるエッジまで抽出していることがわかる。

2.4 ノイズ除去

ゾーベルオペレータによるフィルタリングの結果を見ても、やはりノイズによるものと思われるエッジが現れている。また同じ領域内でも密度変化の激しい部分において不必要なエッジが抽出されている部分が存在する。(写真中央の白い部分が背骨であり、それをささむようにして一対の空豆状の臓器が写っているがこれは腎臓で、内部組織が複雑なため中に多くのエッジが抽出されている。このエッジは、物体の形状を知る上では必ずしも必要ではない。)

このような雑音によるエッジや、形状認識の際には必要とはされないエッジを除去するために、本当のエッジ情報を保持したままノイズの除去を行うことが望ましい。

ノイズの除去の簡単な例としては、各要素がすべて1である一種の積分フィルタの加重マトリクスを用いたフィルタリングによるものがあげられる。しかし、この方法は、平滑化によってノイズだけではなくエッジのデータまでもが失われてしまうという欠陥を持つ。

そこで、二次元の画像処理でも用いられる選択的局所平均化(エッジを保ったスムージング(edge preserving smoothing))の三次元拡張を試みた。

これは図2.4.1に示すような9個の局所領域(4つの5角形、4つの6角形、正方形1つ)内の濃度の分散を計算し、分散が最小となった領域の平均濃度を新たな $f(i, j)$ の濃度とする方法である。この方法を用いると、エッジやノイズを含むような領域は分散が大きくなりエッジのない局所領域の濃度の平均値によって、ノイズが除去されて行く。

これを三次元拡張したものが、図2.4.2である。計15個の局所領域の濃度の分散を計算し、分散が最小になる局所領域の平均濃度を新たな $f(i, j, k)$ の値とするのである。

この処理を写真2.3.1のデータに施した結果が、写真2.4.1である。写真2.4.1をみる

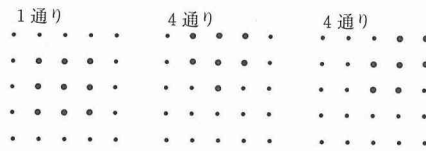


図 2.4.1 二次元の局所領域

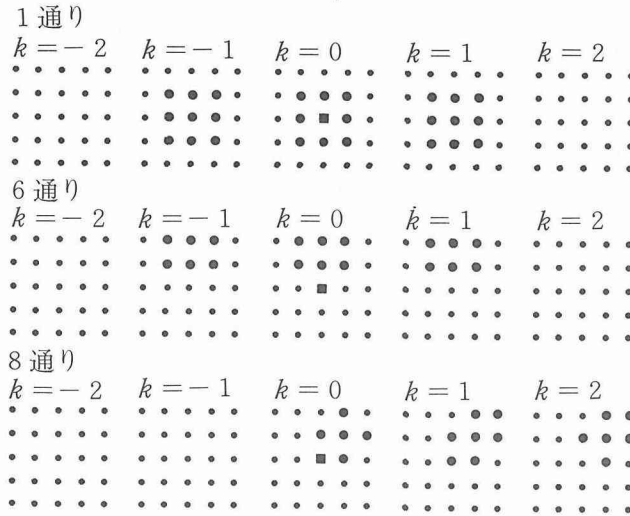


図 2.4.2 三次元の局所領域

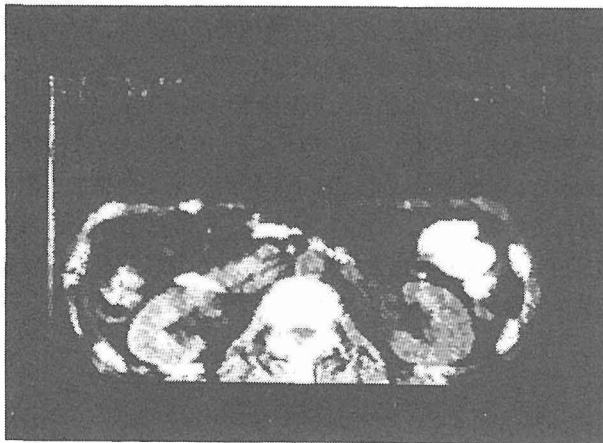


写真 2.4.1 平滑化後の画像

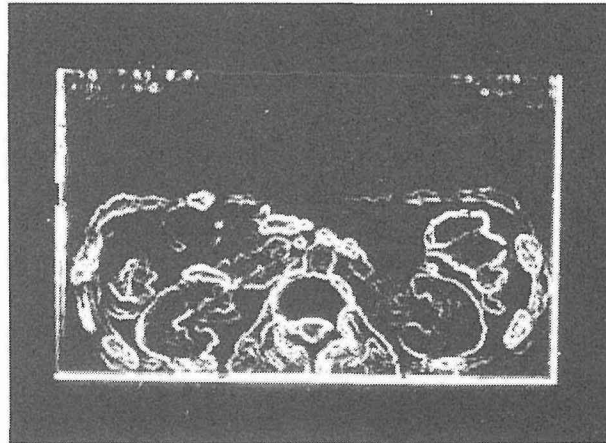


写真 2.4.2 平滑化後に抽出したエッジ

と、腎臓内の急激な密度の変化がなくなり均一化されていることが確認できる。

写真 2.4.2 はノイズを除去したデータに対してゾーベルオペレータを作用させてエッジを抽出した例である。腎臓のあたりにおいて、不必要なエッジが減少していることがわかる。

2.5 ま と め

一連の実験によって、FIR フィルタの三次元化の可能なことが確認され、また実際にエッジ抽出フィルタにおいて良好な結果を得ることができた。

また、フィルタリングを行う際に、あらかじめ選択的局所平均化の処理を行うことによって、エッジ情報を損失する事なく、画像よりノイズを除去できることが確認できた。

最後に、シミュレーションデータに対してエッジ抽出を行い三次元メモリシステムで表示した例と、ゾーベルオペレータによるフィルタリングの際に生じる $\Delta_x f$, $\Delta_z f$ の結果をそれぞれ表示する。データはいずれの写真も、1 画素 1byte, $32 \times 32 \times 32$ ドットである。

写真 2.5.1 は、雪中レーダで得られた三次元画像データに対してゾーベルオペレータを作用させてエッジを抽出した例である。写真 2.5.2 は $\Delta_x f$ を作用させた結果で、X 方向のエッジが抽出されていることがわかる。写真 2.5.3 は $\Delta_y f$ を作用させた結果でこのデータは Y 方向の変化が少ないので反応の現れている領域は小さい。写真 2.5.4 は $\Delta_z f$ を作用させた結果で、 $\Delta_x f$ の場合と同じように、Z 方向のエッジが抽出されていることが見て取れる。最後の写真 2.5.5 は、これらのエッジデータから、最終的に得られたエッジデータである。

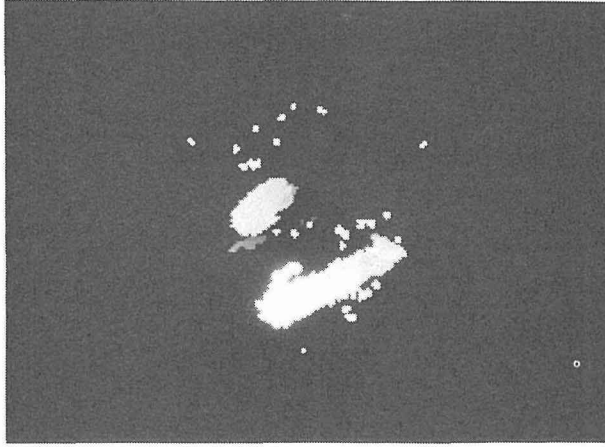


写真 2.5.1 ソースデータ

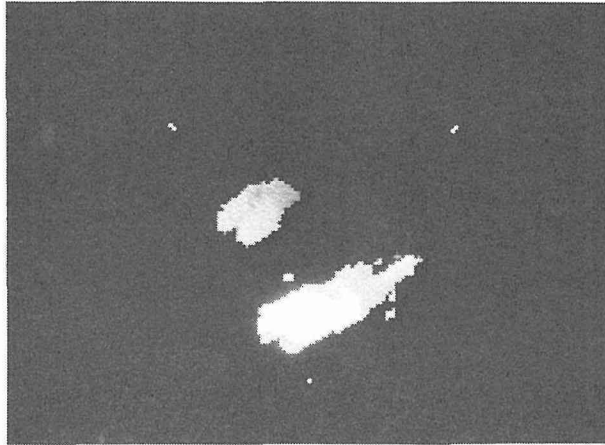


写真 2.5.2 x 方向のエッジ

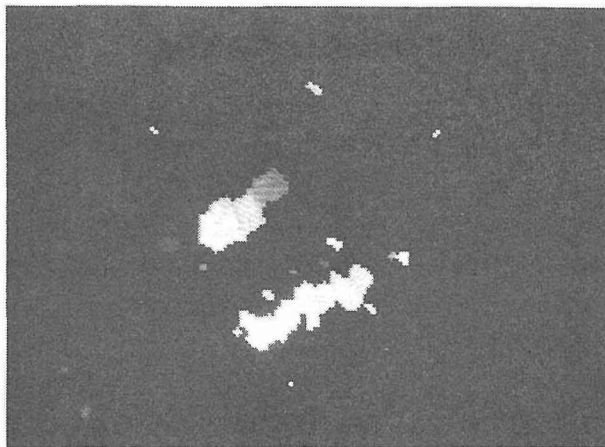


写真 2.5.3 y 方向のエッジ

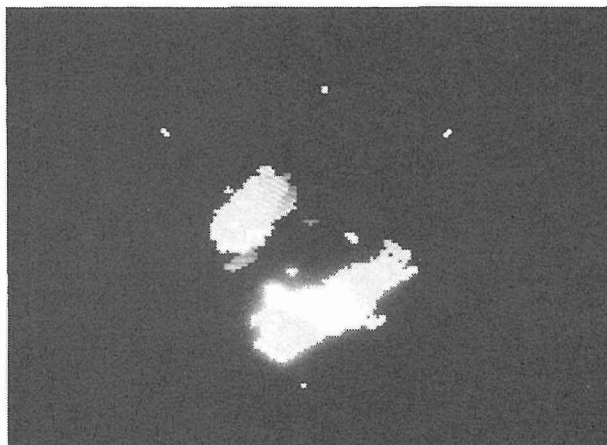


写真2.5.4 z方向へのエッジ

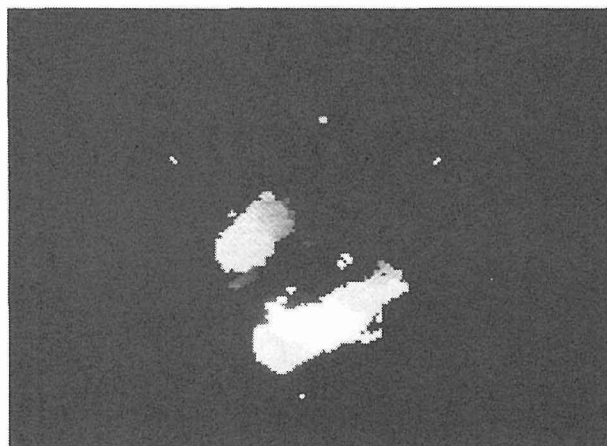


写真2.5.5 エッジデータ

3. 三次元データの領域分割

FIR フィルタを三次元化することによって、画像中からエッジを抽出することに成功したため、次の段階としてフィルタリングによって得られたデータを用いて、三次元画像の領域の分割を試みた。

二次元画像処理においても、中心画素の近傍から同じような値を持つ画素を抽出する領域の分割は行われてきた。写真を領域分割すると、写っている物体を抜き出すことになる。三次元画像データについても同じことがいえ、三次元画像データの領域分割を行うということは、データが幾つのパーツからできているかを表すことになる。

3.1 画像の領域分割

領域の分割の簡単な方法としてはしきい値処理があげられる。しきい値処理はCT像中の骨のように特徴ある値を持つ領域の抽出は行うことができるが、密度変化の激しい部分やエッジが緩やかに変化しているような部分の抽出には不向きである。そこで、中心画素のみではなく、近傍

の情報を取り入れた領域分割の方法が考えられている。

単純領域拡張法はその一つである。この方法は次のようなアルゴリズムによる。

1. 画像をラスタスキャンし、どの領域にも所属していない画素を求める。
2. その画素の濃度とその周囲（4近傍あるいは8近傍）で、まだどの領域にも属していない画素の濃度と比べてその差があるしきい値以下ならひとつの領域として統合する。
3. 新たに統合された画素に注目し2の操作を繰り返す。
4. 2, 3の操作をそれ以上領域が広がらなくなるまで反復する。
5. 1に戻り新たな領域の出発点となる画素を捜す。

しかしこの方法では、領域間の濃度変化が緩やかな場合やコントラストの弱いエッジの隙間があると、二つの領域は統合されてしまう。

この欠点をのぞくためには、2において領域の周辺部の画素の濃度とその周囲の画素の濃度を比較するのではなく、すでに存在する領域の平均濃度と隣接する画素の濃度を比較するとよい。

しかし、この方法では、領域探索の開始点によって最終的な領域分割結果が変わってしまうという欠点が残されている。

特にここで対象とする、雪中レーダのデータからの再生像は、それがホログラムデータからの再生像であるという理由から、辺縁のぼやけた像になっている。そこで、このような辺縁のぼやけた像でも領域の分割できるような方法を考案した。

3.2 エッジデータを用いた領域の分割

さきに述べた方法では、なだらかなエッジの変化が問題になっていた。しかし、画像データからエッジの抽出を行うと、なだらかな変化であってもある程度のエッジを抽出することができる。

そこで、基本的には領域拡張法であるが、単に画素のデータを追跡するだけでなく、あわせてエッジのデータを参照する方法を考案した。

この方法は次のようなアルゴリズムによる。

1. 画像をラスタスキャンし、どの領域にも所属していない画素を求める。
2. その画素の濃度とその周囲（4近傍あるいは8近傍）で、まだどの領域にも属していない画素の濃度と比べて次にあげる条件を満たせばひとつの領域として統合する。
 - A. 画素の濃度差の絶対値が画素用のしきい値以下で、かつ中心画素のエッジデータの絶対値が、領域分離用のエッジ用のしきい値より小さい。
 - B. 画素の濃度差の絶対値が画素用のしきい値以上であっても、中心画素のエッジデータの絶対値が、領域統合用のエッジ用のしきい値以上である。
3. 新たに統合された画素に注目し2の操作を繰り返す。
4. 2, 3の操作をそれ以上領域が広がらなくなるまで反復する。
5. 1に戻り新たな領域の出発点となる画素を捜す。

この方法によると、単純領域拡張法では、2の段階でリジェクトされるほど急激に画素間の値が変化していても、その分エッジのデータが大きくなるので、Aの条件によってひとつの領域として統合される。これは雪中レーダの辺縁の領域統合に役立つ。また隣あう画素が似たような濃度を持っていても、なだらかなエッジ上の点でエッジデータにある程度の値がでていれば、Bの条件によって領域の統合がなされない確率が高くなる。

3.3 実 験

この方法を実際に雪中レーダで得られたデータを用いて有効性を確認した。写真3.3.1は用

いた原画像で、積雪に埋没した二本の金属缶をターゲットとしたものである。原画像の大きさは、1画素256階調、XYZすべての方向に32画像の等方的なデータで、各辺の長さはほぼ50cmである。

写真3.3.1は得られたデータをそのまま表示したもので、雪の表面からの反射や積雪の層からの反射で、ターゲットを視認することは難しい。写真3.3.2は、しきい値処理によって、画素の濃度がしきい値よりも大きい部分のみを取り出したものである。

雪中レーダのデータの場合、このしきい値処理のみでもターゲットのおおよその形を認識することができるが、実際の大きさよりもやや小さな領域が抽出される。これがより複雑な画像データで、様々な濃度の領域が入り組んで存在しているような場合は、この方法は、部分の取り出しには不利である。

そこで、この雪中レーダのデータをエッジ抽出のフィルタにかけて、エッジデータを取り出す。さらにこのエッジデータとソースデータを用いて領域の分割を行う。ここでは雪以外の領域（ターゲット）のみを抽出する。

さらに写真3.3.3、写真3.3.4は先のアルゴリズムを用いて物体をひとつだけ抽出した例である。

単にソースデータにしきい値をかけただけで物体を抜き出すと、先にも述べた通り、雪中レーダのデータでは本来の大きさよりも小さな領域が取り出されてしまう。またそれに合わせてしきい値を下げると、雪の層の反射などが現れてしまい、物体の抽出が困難になる。

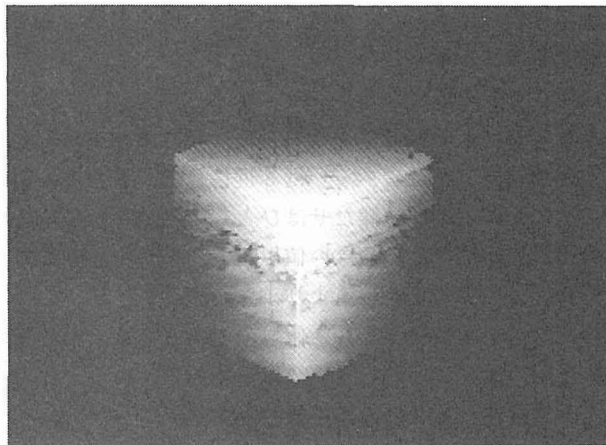


写真3.1.1 雪中レーダのデータ
(しきい値処理なし)

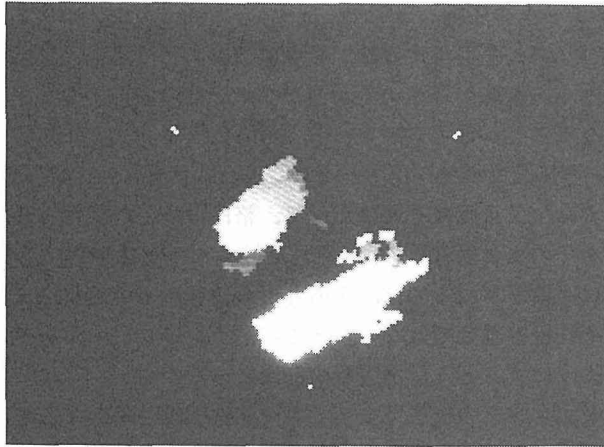


写真 3.3.2 雪中レーダのデータ
(しきい値処理あり)

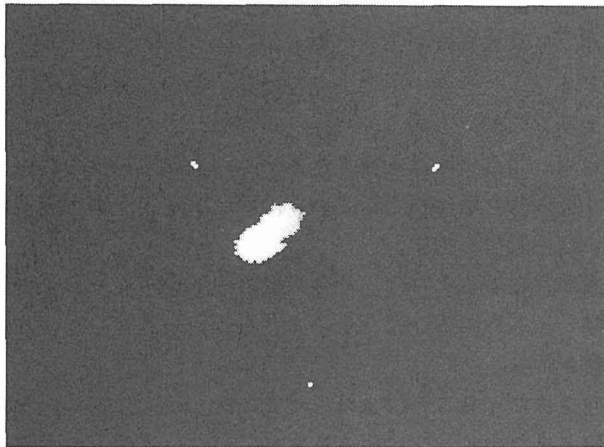


写真 3.3.3 領域の取出し 1

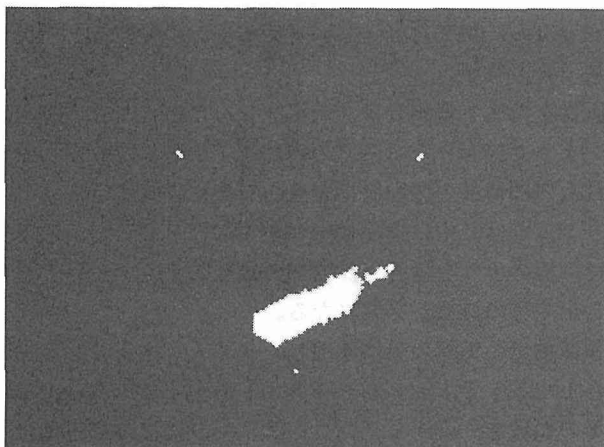


写真 3.3.4 領域の取出し 2

3.4 テンプレートデータの圧縮

この領域抽出アルゴリズムを実現する際、領域の抽出を行う場合に、出力結果に直接画素の濃度を出力するのではなく、領域が存在するか否かで1, 0の値を持つデータを出力した。我々はこのデータが、型になることからテンプレート (templet) データと呼んでいる。このように一度テンプレートを作り、直接画素の値を出力しない理由は、データ保存時の記憶容量のためである。

三次元の画像データは、二次元のそれに比べて、格段に多くの記憶容量を必要とする。もし、抽出した領域データをそのまま保存しては、それだけ多くの記憶容量を消費してしまう。しかし、テンプレートデータに必要とされるのは、その外形の情報のみで、データ自体が二値化されているので内部の情報はそれほど重要ではない。そこで、テンプレートデータを保存する際に外形の情報のみを保存するようにすると、記憶容量の節減につながる。

そこでチェーンコーディングの技法によってデータを圧縮し保存する方法を考案した。

この方法は、テンプレートデータの表面の形状をある一点を起点としてそのつながりを方向で記憶しておく方法である。後日領域の抽出を改めて行う場合は、テンプレートを復元する作業が必要になるが、記憶容量の節約のメリットの方が大きいものとする。チェーンコーディングによるデータ圧縮のアルゴリズムは次のようになる。

1. 圧縮するテンプレートデータの中の一点を指定、手近な表面上の一点を起点とする。この起点を最初のチェーンの始点とする。
2. チェーンに指定点、始点の座標を書き込み、さらに始点を検索の中心画素とする。
3. 中心画素の周り26近傍を順次検索していき、テンプレートの表面上にあり、まだチェーンに組み込まれていない画素を捜す。
4. そのような画素が存在するならば、その方向コードをチェーンの後尾につけ加え、中心画素をその画素に移し、3へ戻る。またそのような画素が存在しなければ、チェーンの最後尾に未満コードを書き込む。
5. チェーンを逆向きにたどることにより、すでにチェーンに組み込まれた画素を中心画素とする。その26近傍にまだチェーンに組み込まれていない表面上の画素がないか捜す。もしそのような画素が存在するならば、新たなチェーンを作り、現在の中心画素が、何番目のチェーンの何番目の要素であるかを書き込み、その時の中心画素を始点として2へゆく。もしそのような画素がなかったら、現在たどっているチェーンの分岐点まで戻り、5を繰り返す。1で指定した点に戻ったら終了。

逆にチェーンから画像を再生する場合は、チェーンをたどってその方向に応じて点を打っていき、すべてのチェーンをうち終わったら、その内部を1データで塗りつぶすことによって行われる。チェーンに囲まれた内部外部の区別は指定点の有無によって行う。

但し、このアルゴリズムでは、中に違う領域がすっぽりと入っているような場合にはその情報は失われてしまうので、そのような場合は、内部の領域のテンプレートもともに作っておき、再生の際その領域を抜き出す等の処置が必要になる。

写真3.4.1は、領域抽出の実際の際に使用したシミュレーションデータである。データの大きさは、一画素あたり1バイトで、 $32 \times 32 \times 32$ ドットのサイズである。このデータに対して領域の抽出を行った後、その領域を一度チェーンコーディングによって圧縮し、再び再生したのが写真3.4.2である。

この場合チェーンコーディングの結果、元はソースデータと同じ32kbyteの容量を必要として

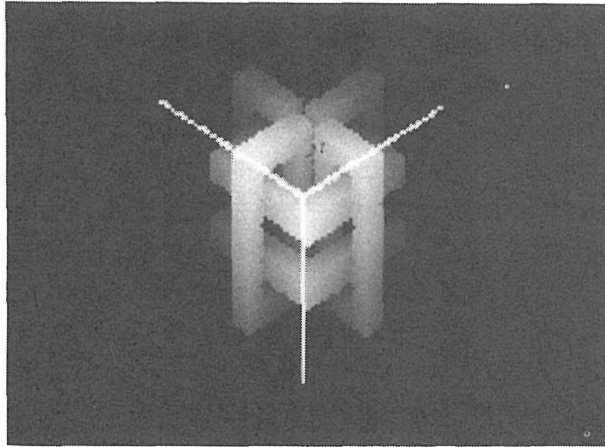


写真3.4.1 シミュレーションデータ

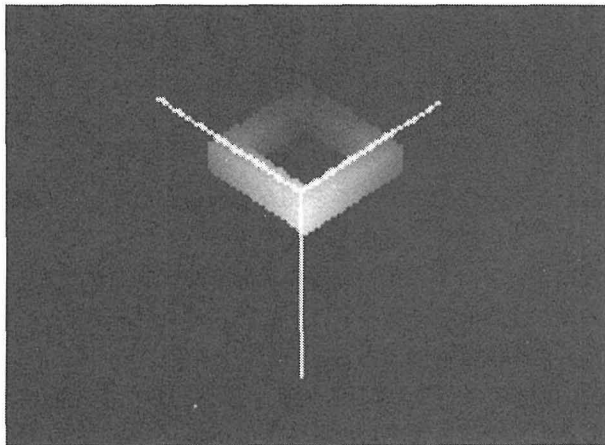


写真3.4.2 チェーンコーディングからの再現像

いたデータを、1 kbyte 強まで圧縮することができた。

3.5 ま と め

一連の実験において、エッジデータを取り入れた領域拡張法において、輪郭の不明瞭なデータにおいても、より正確な領域の分割が行えることを確認でき、雪中レーダによって得られたデータより、積雪中の埋没物体の抽出を行うことに成功した。

さらには、形状のみの保存であるならば、チェーンコーディングを用いてデータの圧縮を行うことによって、記憶容量を節減できることを確認した。

4. 結 論

今回の研究によって、三次元の画像処理における有効な手法をいくつか開発する事ができた。FIR フィルタの三次元化においては、加重マトリクスによるフィルタを三次元化することに成功して、特に画像中からのエッジの抽出でよい結果を得ることができた。

三次元画像の領域分割においては、雪中レーダの三次元画像データより、任意の領域（ターゲット）を抽出できるようになったことから、次の段階である埋没物体の認識への足がかりができた。

領域の抽出ができたということは、その領域中に含まれる Voxel 数を数えることにより、その物体のおおよその体積を知ることができることを意味する。さらに、領域内の Voxel の値を知ることによって、その値から物体の材質を推測することができる。

このような情報を得ることによって、まず材質の条件から、積雪中の埋没物の材質が金属、コンクリート、木、もしくは動物であるのかが推測でき、さらに体積の条件から、金属であれば、自動車なのか、消火栓なのか、動物であれば、人間の成人か、猫や犬などの小動物なのかを推測できる。

さらに、消火栓や自動車のように、ほぼ形や体積の決まった物体であれば、取り出された領域に再びフーリエ変換を施すことにより、パターンマッチングによって、種類の同定を行うことができる。

さらに、物体を抽出して残った雪のデータを調査することができる。強い反射をもつ物体がなくなった後の雪のデータにも、積雪の層などの情報が残っており、その情報を活用することができるようになる。これによって、その年の降雪の状況を知ることができるようになる。

さらに今回の研究では、主な対象を雪中レーダで得られたデータとしたが、他の三次元画像データに対しても、これらの手法は有効であると思われる。

序論でも述べた通り、医療用の CT や MRI 装置などは一度のスキャンによって、高分解能の三次元画像データを得ることができるようになってきている。しかし、いくら分解能が高くなっても、表示方法が断面表示や、透過表示のままでは、人間がその画像を認識するためには熟練を必要とする。

そこで、より容易に三次元画像を認識するために、今回開発した三次元画像データの領域分割を用いて、特定の部分のみを抽出して表示する方法が考えられる。例えば、心臓の検診においては、肺や脊椎、肋骨等のデータは不必要な情報なので、三次元画像の中から心臓だけを取り出して表示することによって、認識が容易になる。

このように、必要な部分のみを取り出して表示することにより、三次元画像データの認識性ははるかに高まるであろう。

しかし正確な領域の抽出には、さらに高い解像度の画像や抽出すべき領域の平均的な Voxel 値や、隣接する周囲の領域の平均的な Voxel 値についてなど多くの情報が必要となり、そのためのデータベースが必要となるだろう。

さらに今後の三次元画像処理に期待するものとしては、三次元画像データの展開などが考えられる。三次元画像データから単一の部分を抽出して表示しても、そのままではまだ視認性が十分であるとはいえない。

例えば心臓だけを取り出して観察しても、その内部を見ることはできず、心臓内部の疾患を知

るためには断面の表示などが必要になってしまう。そこでこのような問題を解決するために、三次元データの展開を行って、内部の表示を行う事が必要であり、今後の課題である。

参 考 文 献

- 1) 高橋佳成, 坂本雄児, 青木由直, テレビジョン学会技術報告, Vol. 12, No. 9 (1988), pp. 33-38.
- 2) 坂本雄児, 田尻克博, 澤井孝哉, 青木由直, 電子情報通信学会論文誌, Vol. J70-B, No. 12(1987), pp. 1544-1550.
- 3) 青木由直, 波動信号処理 (森北出版, 1986), pp. 188-194.