

論文 / 著書情報
Article / Book Information

題目(和文)	ニューラルネットワークを用いた学習に基づく動的物体抽出に関する研究
Title(English)	
著者(和文)	岩田治幸
Author(English)	
出典(和文)	学位:博士(工学), 学位授与機関:東京工業大学, 報告番号:甲第3847号, 授与年月日:1998年3月26日, 学位の種別:課程博士, 審査員:
Citation(English)	Degree:Doctor (Engineering), Conferring organization: Tokyo Institute of Technology, Report number:甲第3847号, Conferred date:1998/3/26, Degree Type:Course doctor, Examiner:
学位種別(和文)	博士論文
Type(English)	Doctoral Thesis

ニューラルネットワークを用いた
学習に基づく動的物体抽出に関する研究

平成 10 年 1 月

指導教官 長橋 宏 助教授

提出者 知能科学専攻

学籍番号 95D31046

岩田 治幸

目次

第 1 章 序論	1
1.1 本研究の背景および目的	1
1.2 本研究で用いたシステム構成	4
1.3 本論文の構成	5
第 2 章 従来の物体抽出手法	6
2.1 エッジ検出法	6
2.1.1 エッジ検出法の概要	7
2.1.2 エッジ検出における弛緩法	9
2.2 領域分割法	16
2.2.1 領域分割法の概要	16
2.2.2 分離・統合法	18
2.2.3 動的しきい値処理	21
2.2.4 領域分割における弛緩法	21
2.3 動的モデルを用いる手法	31
2.3.1 Snakes の概要	31
2.3.2 Active Net の概要	31
2.4 本章のまとめ	32
第 3 章 Hopfield 型ニューラルネットワークを用いたカラー画像における境界線検出	34
3.1 境界線検出アルゴリズム	34
3.1.1 自動エッジ追跡によるエッジセグメント検出	34
3.1.2 Hopfield 型ニューラルネットワークを用いた境界線上エッジセグメント選択	37
3.1.3 処理領域移動方法	42

3.2	実験結果	42
3.3	Snakes との対比	53
3.4	本章のまとめ	54
第 4 章 Hopfield 型ニューラルネットワークを用いたカラー画像における動的領域抽出		57
4.1	Hopfield 型ニューラルネットワークを用いた動的領域抽出	58
4.1.1	動的領域抽出アルゴリズム	58
4.1.2	Hopfield 型ニューラルネットワークの目的関数	63
4.1.3	分裂領域への対処方法	64
4.2	実験結果	65
4.3	Active Net との対比	73
4.4	視覚系との対応に関する考察	76
4.5	本章のまとめ	79
第 5 章 抽出対象に応じた動的領域抽出の改良および応用		81
5.1	領域形状に基づく動的領域抽出	81
5.2	複数シードの場合への拡張	83
5.3	移動物体追跡への応用	85
5.4	本章のまとめ	91
第 6 章 階層型ニューラルネットワークの学習を用いた動的領域抽出		92
6.1	領域変形度認識と領域抽出の統合	94
6.2	階層型ニューラルネットワークの構造と入力信号	97
6.3	学習用データ自動生成と学習過程	99
6.4	実験結果	103
6.5	本章のまとめ	104
第 7 章 結論		107
7.1	本研究のまとめ	107
7.2	今後の課題	108
付録 A Hopfield 型ニューラルネットワークを用いた最適化		110

目次	iii
付録 B 階層型ニューラルネットワークにおける学習	114
B.1 Perceptron	114
B.2 誤差逆伝播アルゴリズム	117
謝辞	120
参考文献	121
研究業績	125
図目次	132
表目次	132

第 1 章

序論

1.1 本研究の背景および目的

画像中の物体抽出処理は，画像の認識・理解において重要である．特に，高度ヒューマンインターフェースにおける個人識別・表情認識・視線検出・ジェスチャー認識，画像合成などにおいては，物体領域を正確に抽出できることが不可欠である．また，画像通信，特に，物体と背景を別のオブジェクトとして符号化する，MPEG4[1]などのオブジェクト指向符号化において必要である．割り当てる符号量を物体領域に多く，背景領域に少なくすることで，符号化効率を向上させることができる．

従来の物体抽出，すなわち物体領域と背景領域との境界線検出のためのアプローチは，エッジに基づく手法と領域に基づく手法に大別することができる．

エッジに基づく手法 [2] では，クラックの検出や閉じていない境界線の検出が可能であるなどの利点があるものの，一部がぼやけたり途切れたりしている境界線の検出が効果的に行えないなどの問題が生じる．確率的弛緩法 [3] やマルコフ確率場モデル [4]，Hough 変換 [5] などの手法は，これらの問題へ対処する手法として有効ではあるが，前者の二つの手法は多大な計算コストを要する．Hough 変換 [5] は，あらかじめ境界線形状が既知である場合に有効であるが，やはり計算コストは少ない．

領域に基づく手法は，画像上あるいは特徴空間におけるクラスタリングであり，閉じた境界線を検出できるという大きな利点があるが，偽の輪郭が検出されたりクラックの検出が困難であるなどの問題も生じる．局所的な領域特徴量の均一性に基づく手法 [6] は最も簡単な例であるが，局所的な拘束条件だけを用いて解を得るこ

とに相当し、解が一意に決まる利点があるものの、大局的な解を得られない場合が多い。例えば、画像の不鮮明さやノイズなどのために、階調値などの領域に対する特徴量に局所的なあいまいさが存在すると、領域の細分化や過併合を生じてしまう。また階調値分布が比較的単純な場合でも、しきい値の設定は、人が分割結果に応じて行わなければならない場合が多い。そこで、より大局的な解を自動的な処理によって得るためには、大局的な性質を表す何らかの目的関数を定義して、その最小解を、あらかじめ仮定する拘束条件のもとで求めるという正則化 (regularization) [7] [8] が有効となる。

正則化に基づく従来の手法の一つに、動的モデルを用いる手法がある。動的モデルにおける拘束条件は、制御点あるいは格子点の隣接関係であり、目的関数はモデルの内部エネルギー関数と、画像とモデルの適合性エネルギー関数との線形結合で表される。動的モデルの代表的なものとして、Snakes [9] や Active Net [10] が提案されているが、これらのモデルを用いる場合には、最初に初期モデル、すなわち制御点あるいは格子点の初期状態を境界線近傍に与える必要がある。抽出される輪郭が、制御点あるいは格子点の数および、それらの配置の仕方にできるだけ依存しないようにするためのさまざまな改良が試みられているが [11] [12]、局所解 (local minimum) に陥ることが多く、複雑な背景を有する自然画像から任意形状の輪郭を有効に抽出する手法は、まだ報告されていない。また、分裂する Snakes [11] および Active Net では、複数の均一な領域からなる物体領域を、ひとまとまりにして抽出することはできない。

一方、領域成長法とエッジ検出法を統合することで、それぞれを単独で用いるよりも良い結果が得られることが知られている [13]。しかし、この場合、領域とエッジという異なる評価対象をどのように調和させるかという問題が生じてくる。異なる種類の目的関数を同時に最小化する問題は、組合わせ最適化問題 [14] の一つであり、Hopfield 型ニューラルネットワーク [15]、シミュレーテッドアニーリング (Simulated annealing) を用いたボルツマンマシン [16]、そして遺伝的アルゴリズム [17] などを用いた最適解探索手法が提案されている。本研究では、これらの最適解探索手法のうち、短時間で解を求めることができる Hopfield 型ニューラルネットワークを用いる。

そこで本論文では、まず最初に、動的モデルと同様にエネルギー最小化に基づいた手法ではあるものの、Hopfield 型ニューラルネットワークのエネルギー関数値極小化原理を用いることによって、検出したエッジセグメントの中から境界線上にあるものを選択することで、ひとつながりの線構造を徐々に形成する手法を提案する。

この手法は、背景の影響を受けにくく、ぼやけや途切れに対して有効で、更に、さまざまな形状の境界線を検出することが可能である。次に、Hopfield 型ニューラルネットワークのエネルギー関数値極小化原理を用いることによって、領域成長法とエッジ検出法を統合して、あらかじめ与える初期領域に依存しない動的領域抽出手法を提案する。この手法は、正則化に基づく手法として位置づけられ、ロバストで高精度な処理が可能である。最後に、サンプルとしてあらかじめ与える物体領域の特徴を、階層型ニューラルネットワークを用いた学習によって自動的に獲得した後に、獲得した特徴との適合性に基づき、動的に物体領域を抽出する手法を提案する。具体的には、既に提案した動的領域抽出手法において、学習後の階層型ニューラルネットワークを、獲得した特徴との適合性を調べるために用いる。この手法は、静止画像において、顔のように複数の階調値が均一な領域からなる物体領域を、ひとまとまりにして抽出することが可能である。

1.2 本研究で用いたシステム構成

本研究で用いたハードウェアおよびソフトウェアについて表 1.1 に示す.

表 1.1 研究で用いたハードウェアおよびソフトウェア

ワークステーション ディスプレイ CPU オペレーティングシステム メインメモリ ウィンドウシステム C コンパイラ	SUN SPARCstation10 1152 × 900 ドット R, G, B 各 256 階調 SuperSPARC (36MHz) SunOS4.1.3 128MB X-Window Ver.11 Release 6 GNU C compiler Ver. 2.7.2.1
ページプリンタ	Apple Laser Writer 16/600 PS
イメージスキャナ	EPSON GT-8500
CCD カメラ AD/DA 変換	SONY DXC-151 ADS PIP-9000

イメージスキャナ, CCD カメラで取り込んだ画像および標準画像を, ワークステーション SPARCstation10 で処理して, ページプリンタで結果を出力した.

1.3 本論文の構成

1章「序論」では、本研究の背景と目的、本研究で用いたシステム構成について述べた。

2章「従来の物体抽出手法」では、従来の物体抽出手法の概要について説明した後に、代表的な手法の検証を行なった結果と、それらの特徴および問題点について述べる。

3章「Hopfield 型ニューラルネットワークを用いたカラー画像における境界線検出」では、本研究で提案する境界線検出手法について述べる。この手法は、Hopfield 型ニューラルネットワークのエネルギー関数値極小化原理を用いて、検出したエッジセグメントの中から境界線上にあるものを選択する手法である。

4章「Hopfield 型ニューラルネットワークを用いたカラー画像における動的領域抽出」では、本研究で提案する動的領域抽出手法について述べる。この手法は、Hopfield 型ニューラルネットワークのエネルギー関数値極小化原理を用いて、ピラミッド画像上で動的に領域を抽出する手法である。

5章「抽出対象に応じた動的領域抽出の改良および応用」では、4章で提案する手法において、形状に関する仮定に基づく項を追加して、人物の顔領域を良好に抽出する手法、二つ以上の領域を抽出するための手法、そして移動物体追跡への応用について述べる。

6章「階層型ニューラルネットワークの学習を用いた動的領域抽出」では、本研究で提案する変形度認識による動的領域抽出手法について述べる。この手法は、サンプルとしてあらかじめ与える物体領域の特徴を、階層型ニューラルネットワークを用いた学習によって自動的に獲得した後に、獲得した特徴との適合性に基づき、Hopfield 型ニューラルネットワークのエネルギー関数値極小化原理を用いて、動的に領域を抽出する手法である。

7章「結論」では、本研究のまとめと今後の課題について述べる。

第 2 章

従来の物体抽出手法

画像中の物体領域抽出は，一般に画像のセグメンテーションに含まれ，そのアプローチには次の二つがある．

1. 物体の輪郭に相当するエッジを線として検出する手法
——エッジ検出法 (edge detection methods)——
2. 画像を物体と背景に対応した部分画像に分割する手法
——領域分割法 (region segmentation methods)——
——動的モデル (active model) を用いる手法——

まず 2.1 でエッジ検出法について，次に 2.2 で領域分割法について，最後に 2.3 で動的モデルを用いる手法について，それぞれの概要を簡単に述べた後に，代表的手法を例に，その特徴および問題点について述べる．

2.1 エッジ検出法

物体領域抽出のためのアプローチとしてのエッジ検出は，主として線画などのような 2 値画像に対して広く用いられる手法であり，原画像中のエッジをすべて検出した後に，閉曲線によって囲まれた領域が均一領域であるとする手法である．しかしながら，一般の階調画像に対しては明確な閉曲線が得られることは希であり，実際には有効でない場合が多く，処理対象を限定した場合だけに用いられている．



図 2.1 エッジの分類

2.1.1 エッジ検出法の概要

エッジ検出は、一般的にエッジの強調によって行われる。エッジは、その方向と直角の断面をとり、そこでの階調値の変化に従って区別される。Herskovits, et al. [18] は、典型的なエッジを図 2.1 に示すように分類した。

実際の画像におけるエッジは、理想的な Step 型エッジであることはほとんどなく、ぼやけあるいはノイズを伴っている。また、画像中には階調差の大きなエッジから小さなエッジまで混在するのが普通である。従って、不必要なエッジを取り除いたり、途切れたエッジを連結するための処理を必要とする。

エッジは理想的には面積をもたないが、現実検出されるエッジは面積をもち、あるいはぼやけている場合が多い。これが種々の問題を生じさせる。濃淡変化に対する人間の感じ方についてはさまざまな研究があるが、エッジの抽出と関連してはマッハ現象がある。

エッジを強調する手法としては、微分処理とテンプレートマッチングが良く用いられる。それぞれの手法について簡単に述べる。

微分処理によるエッジ強調

エッジでは、階調値変化が存在する。従って、このような領域は微分処理によって強調できる。画像の微分処理では、演算に用いられる画素の数が少ないために、画像中にノイズがあるとその影響を大きく受ける。そこで、より多くの画素を用いることでノイズの影響を受けにくい処理を行なうことができる。Sobel フィルタは、 3×3 の近傍画素を用いた差分処理によってエッジ強調を行なう代表的なフィルタである。

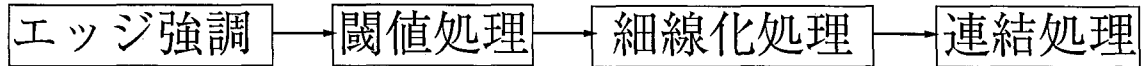


図 2.2 従来のエッジ強調による境界線検出処理

テンプレートマッチングによるエッジ強調

対象物体領域の標準的なテンプレートを用意して、画像とテンプレートとの相関を測ることによって画像中の対象物体領域を検出する手法がある。この手法はテンプレートマッチングと呼ばれる。ここで、対象物体領域をエッジまたは線と考え、それぞれのテンプレートを用意すればテンプレートマッチングによってエッジを強調することができる。

画像のセグメンテーションが目的である場合には、エッジ強調された画像においてエッジ画素を連結して一つの閉じた境界線を構成する必要がある。そのための代表的な処理手順を図 2.2 に示す。エッジあるいは線が存在する場所は、エッジ強調された画像に対し、適当なしきい値処理を行なうことによって求められる。簡単なしきい値処理では、連結した境界線が得られないことが多い。一般に境界線は途切れて断片的に検出される。従って、この断片的な境界線要素を連結する処理が必要となる。断片的な境界線要素が点列または細い線分である場合には、比較的容易に連結することができる。そこで境界線要素の細線化処理が行なわれる。境界線要素を連結するための最も簡単な手法は、境界線要素の端の画素とその近傍に位置する他の境界線要素の端の画素を直線で結ぶ手法である。画像中の境界線は滑らかである場合が多い。従って、境界線要素の方向の連続性を用いることで滑らかな連結が可能となり、良い結果が得られる。境界線要素を結合する手法には、グラフ探索問題として考える手法が報告されている。

検出したい要素の連結性に関する知識、拘束条件を一定の評価関数として与え、これに基づいて経路を求めるグラフ探索問題として、要素の結合を考える手法は、heuristic search を中心として数多くの手法が提案されている。Ramer [19] は、あらかじめエッジ検出オペレータで要素を検出しておき、これに heuristic search を行なった。Ashker & Modestino [20] の手法は、計算コストの軽減を考慮したアルゴリズムを用いている。さらに、探索に用いる経路の評価式に、明るさの差分、曲線の

曲率に加えて、前もって与えられた曲線に関する大局的な情報を取り入れた。

その他にもさまざまな手法が報告されている。例えば Nevatia & Babu [21] は、エッジ要素の検出・強調・結合の一連の処理を提案した。まず 5×5 のサイズで 30° きざみ 6 方向のテンプレート型マスクを使ってエッジ要素を検出する。これに細線化としきい値処理を行ない、残された候補点について 8 隣接点の連結性を調べる簡単な手法で結合している。この他にも、あらかじめエッジの方向についてグループ分けを行ってから結合する手法や、検出された線要素のペアについてコストを計算し、画像中で最小コストのペアを毎回とりだし結合する手法などが報告されている。

境界線検出の手法としては、他に、線を方程式で表現する Hough 変換がある。検出したい線の種類があらかじめ限定されていれば、その線を表す式のパラメータ空間に要素を写像して、この空間でクラスタリングを行ない線を強調することができる。例えば直線の式を傾き a 、切片 b として $y = ax + b$ で表せば、この線上の点 (x_i, y_i) はパラメータ (a, b) 平面で $b = -x_i a + y_i$ の直線に写像される。従って (a, b) 平面上のこれらの直線の交点の座標に基づいて、原画像の直線を定めることができる。直線以外の曲線についてもパラメータ表現ができれば原理的に拡張できるので、円検出、放物線検出、楕円検出などが行なわれている。一般に曲線の場合には、パラメータの数の増大に伴い処理の上で困難が生じるため、何らかの工夫が必要となる。

エッジに関する知識を用いて並列処理によってエッジ強調およびノイズ除去を行なう手法の一つに弛緩法がある。弛緩法については、提案する手法との関連が重要であるので、2.1.2 で実験結果とともに述べる。

2.1.2 エッジ検出における弛緩法

画像処理では、ある処理結果によって対象物体領域に対し適切な解釈を与えることがよく行われる。これは対象物体領域にその特徴によってラベルを付ける処理である。例えばエッジ強調処理の場合、処理結果によって画素がエッジであるか否かを定めることである。一般に、エッジでない画素がノイズなどの影響で十分にエッジらしさを持つことがある。従って、エッジらしさに対してラベルを一意的に定めることができない。しかし、対象物体領域を個別的に見たのでは、その解釈すなわちラベル付けが一意的に定まらないような場合でも、対象物体領域間の関係の情報を考慮すれば適切なラベル付けが可能になる場合もある。弛緩法 (relaxation method) は、この関係を用いてラベル付けのあいまいさを取り除いていく一つの手法である。弛緩法は、完全な並列処理であり、局所的なデータを繰り返し処理することにより、

大局的なデータの処理が行われるところにその特徴がある.

弛緩法によるエッジ強調の概要

まず最初に, $(x_i, y_i) (i \in \{1, 2, \dots, N\}, N: \text{画素数})$ に位置する画素を, 画素 a_i とする. すべての画素に対してエッジ強度 M_i を式 (2.1) に従って求め, エッジの方向 θ_i を式 (2.2) に従って求める. ここで, f_i は a_i の階調値, $f_x(x_i, y_i), f_y(x_i, y_i)$ は a_i における x, y 方向の 1 次微分である.

$$M_i = |\Delta f(x_i, y_i)|$$

$$= \sqrt{f_x^2(x_i, y_i) + f_y^2(x_i, y_i)} \quad (2.1)$$

$$\theta_i = \tan^{-1}\left(\frac{f_x}{f_y}\right) \quad (2.2)$$

すべての画素 a_i に対して, ラベル $\lambda (\lambda \in \{e, n\}, e: \text{エッジラベル}, n: \text{非エッジラベル})$ の確率 $P_i(\lambda)$ の初期値を式 (2.3) および式 (2.4) で定義する.

$$P_i(e) = \frac{M_i}{\max_{k=1 \dots N} [M_k]} \quad (2.3)$$

$$P_i(n) = 1 - P_i(e) \quad (2.4)$$

ラベル確率 $P_i(\lambda)$ の更新は, a_i の近傍画素 a_j のラベル確率 $P_j(\lambda)$ を用いて行なう. まず, $P_i(\lambda)$ に対する $P_j(\lambda)$ の寄与 $Q_i(\lambda)$ を式 (2.5) および式 (2.6) に従って求める.

$$Q_i(e) = \sum_j \{C_1 P_i(e) R_{ij}(e, e) + C_2 P_i(n) R_{ij}(e, n)\} \quad (2.5)$$

$$Q_i(n) = \sum_j \{C_3 P_i(e) R_{ij}(n, e) + C_4 P_i(n) R_{ij}(n, n)\} \quad (2.6)$$

C_k : 適当に設定される定数 ($\sum_k C_k = 1$).

R_{ij} は, 適合係数と呼ばれ, 式 (2.7) から式 (2.10) で定義する.

$$R_{ij}(e, e) = \cos(\alpha - \gamma) \cos(\beta - \gamma) / 2^d \quad (2.7)$$

$$R_{ij}(e, n) = \min[0, -\cos(2\alpha - 2\gamma) / 2^d] \quad (2.8)$$

$$R_{ij}(n, e) = \{1 - \cos(2\beta - 2\gamma)\} / 2^{d+1} \quad (2.9)$$

$$R_{ij}(n, n) = 1 / 2^d \quad (2.10)$$

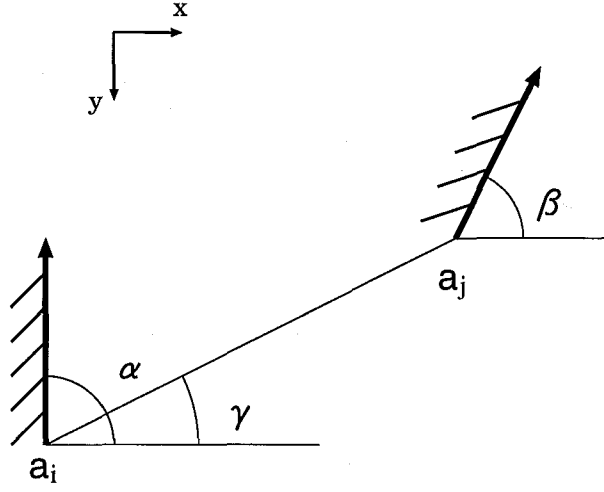


図 2.3 エッジの向き

図 2.3に示す α, β は式 (2.2) で求めた a_i および a_j におけるエッジの方向であり、 γ は a_i から a_j を見たときの方向、 d は a_i と a_j とのチェス盤距離である。

$Q_i(\lambda)$ を -1 から 1 までの値を持つような $Q'_i(\lambda)$ に式 (2.11) および式 (2.12) に従って正規化した後に、 $P_i(\lambda)$ の更新を式 (2.13) および式 (2.14) に従って行なう。

$$Q'_i(e) = \frac{Q_i(e)}{(|Q_i(e)| + |Q_i(n)|)} \quad (2.11)$$

$$Q'_i(n) = \frac{Q_i(n)}{(|Q_i(e)| + |Q_i(n)|)} \quad (2.12)$$

$$P_i^{new}(e) = P_i(e)(1 + Q'_i(e))/S \quad (2.13)$$

$$\begin{aligned} P_i^{new}(n) &= P_i(n)(1 + Q'_i(n))/S \\ &= 1 - P_i^{new}(e) \end{aligned} \quad (2.14)$$

但し、 $S = \sum_{\lambda} P_i(\lambda)(1 + Q_i(\lambda))$ 。

$P_i(\lambda)$ の更新をすべての画素に対して行なった後に、すべての画素における θ_i を式 (2.15) から式 (2.17) に従って更新する。

$$f'_x = W P_i(e) \cos(\alpha) + \sum_j P_j(e) R_{ij}(e, e) \cos(\beta) \quad (2.15)$$

$$f'_y = W P_i(e) \sin(\alpha) + \sum_j P_j(e) R_{ij}(e, e) \sin(\beta) \quad (2.16)$$

$$\theta_i^{new}(e) = \tan^{-1}\left(\frac{f'_x}{f'_y}\right) \quad (2.17)$$

すべての画素に対して、適当な回数の更新を行なった後に、 a_i の階調値 f_i^{last} を式(2.18)に従って求める。

$$f_i^{last} = P_i^{last}(e)(G - 1) \quad (2.18)$$

G :対象画素の階調数.

実験結果

図 2.4に示す 256×320 画素 256 階調の標準画像に対して、Sobel フィルタによるエッジ強調を行なった結果を図 2.5に、 $C_1 = 0.866$, $C_2 = 0.124$, $C_3 = 0.005$, $C_4 = 0.005$, $W = 3.0$ として、5 回繰り返し処理を行なった結果を図 2.6に示す。

弛緩法は、数回の繰り返し処理によって、大局的にエッジを強調してノイズを除去することができる。単に微分を用いてエッジを強調する場合と比較して局所的な曖昧さを除去することが可能である。しかし、図 2.4の左奥の窓の境界線のように、ぼやけた境界線の一部がノイズとして除去されてしまう場合がある。



図 2.4 SCID“カフェテラス”



図 2.5 Sobel



図 2.6 弛緩法によるエッジ強調結果（更新回数 5 回）

2.2 領域分割法

画像の領域分割は、画像をある一定の特徴をもつ小領域ごとに分割する処理であり、画像の理解のための前処理として用いられる。物体領域抽出のための領域分割は、特徴の均一性に注目したアプローチで対象物体領域を直接検出するため、対象物体領域の境界は必ず閉じており、画像の大局的な構造を把握しやすい利点がある。しかし、本来は同一であると考えられる領域を複数の領域に細分した場合や、領域間の境界線が得られずに本来は複数の領域であると考えられる領域を一つの領域にした場合は、後の認識処理を困難かつ複雑にすることになる。従って、画像の領域分割の良否は、画像の認識結果に大きな影響を与える。一般に領域の明確な境界が得られることは希である。また、各領域内でも特徴にばらつきがある場合がほとんどである。このため、計算機による画像の領域分割は非常に難しい工学的課題のひとつであり、以前から多くの研究者達が研究を続けている。しかしながら、あらゆる場合に適応可能で人間が行なう領域分割と同等の性能をもつ汎用的な領域分割法は現在のところ開発されていない。従って、領域分割は、物体領域の階調値が均一あるいは滑らかであり、かつ、物体領域と背景領域の階調値が大きく異なる場合に限り有効である。

2.2.1 領域分割法の概要

領域分割法には、クラスタリングが用いられる。領域分割法を画素のクラスタリングとして捉えたとすれば、クラスタリングを行なう空間の違いに基づいて、二つに分けて考えることができる。すなわち、画像平面であるか、あるいは特徴空間であるかである。

画像平面上でのクラスタリング

画像平面上でのクラスタリングによる領域分割法は、画像の局所的な類似性に基づいて、与えられた画像の平面上で画素のクラスタリングを行ない、画像を特徴の様な連結領域群に分割する手法である。領域成長法 (region growing) [22] と呼ばれることもある。領域の形成過程の違いに注目すると、その手法は、次に列挙する統合、分離、分離・統合、画素結合の四つに大別することができる。

1. 統合法

画像を一旦、小領域 (最小単位は画素) に分離しておいて、隣接した小領域間の類似度を計算し、似た小領域を順次統合することで領域を形成する手法である。

2. 分離法;

画像全体から出発して、領域内で特徴が均一でない限り、部分領域への分割を繰り返す手法である。この手法は、後処理として類似領域の統合を行なわなければ、対象物体領域の抽出には利用できない。

3. 分離・統合法 (Split-and-merge)

1. の統合法は最小領域から出発してボトム・アップ的に統合を繰り返すことによって領域を形成し、また、2. の分離法では、逆に画像全体から出発してトップ・ダウン的に領域を分離していた。Horowitz & Pavlidis [23] はこれらの手法を組み合わせた枠組として分離・統合法を提案している。中間レベルの分割画像から出発して、まず各領域を定数 (平均階調値) で近似する。そして近似精度の低い領域は分離し、似た近似をもつ隣接領域は統合する。分離・統合法は階層型画像データ構造 [24] と関連づけて議論されることが多い。分離・統合法については、提案する動的領域抽出手法との比較において重要であるので、2.2.2 で再び述べる。

4. 画素結合法

隣接した画素間に何らかの関係を導入し、関係の成り立つ画素対を順次結合することで領域を形成する手法である。逐次実行型のアルゴリズムでは領域の結合を行なっているように見えるが、実際には、画素を節点に対応させ、関係のなり立っている画素間に辺をもたせたグラフを構成しており、画像に対してこの関係グラフが構成された時点で分割は一意に決まる。ここで、領域は関係グラフの各連結部分グラフに対応しており、領域の形成はグラフ探索によって達成される。

特徴空間上でのクラスタリング

特徴空間上でのクラスタリングによる領域分割法は、まず画像平面内の画素を特徴空間に写像する。そして、特徴空間内の画素クラスタを識別関数やクラスタリングによって識別し、各クラスタに属する画素にラベル付けを行なった後に、もとの

画像平面に逆写像する。この手法は、画素をパターンと見做したパターン認識理論の応用である。この手法では、しきい値処理と多次元特徴空間でのクラスタリングが代表的である。

1. しきい値処理

特徴空間の分割による画素の分類結果を画像平面に写像する最も簡単な手法はしきい値処理 (thresholding) であり、文字図形などのように階調値分布が比較的単純な場合には、対象物体領域の検出に良い結果を与える。しきい値選択の手法は階調値ヒストグラムを利用するものが多い。しきい値処理については、提案する境界線検出手法において重要であるので、2.2.3 で再び述べる。

2. 多次元特徴空間でのクラスタリング

画像内の各画素について、複数の特徴が与えられている場合には、1次元ヒストグラムに対する処理を反復する手法が有効なことが多い。しかし、クラスタの分布が特殊な場合には、1次元射影だけから各クラスタを検出することは本質的に不可能である。このような場合を想定すると、完全な多次元特徴空間でのクラスタリングが必要である。多次元特徴空間での画素あるいは小領域の分類・クラスタリングによって画像を領域に分割する手法には、マルチスペクトル画像を対象にしたものが多く、リモートセンシング画像の解析に良く利用されている [25]。

領域分割法を利用する際には、与えられた処理対象と処理目的のためのアプローチとして領域分割法が適当であるか否かを判断する必要がある。また、各領域分割手法において、暗に仮定されている画像の性質と処理対象画像の性質が一致しているか否かを各自が判断しなければならない。

2.2.2 分離・統合法

従来の分離・統合法 (Split-and-merge procedure) に関する研究では、階調画像を対象にしたものが多く、領域内での階調値の最大値と最小値との差が、あるしきい値よりも大きい場合に4分割し、また、隣接する領域を合わせた領域内での階調値の最大値と最小値との差がしきい値以下である場合に統合していた。

実験で用いた 256×256 画素のカラー標準画像を図 2.7 に、初期分割を縦横 8 画素間隔で与えたのちに、splitting を行なった結果を図 2.8 に示す。図 2.8 で得られた

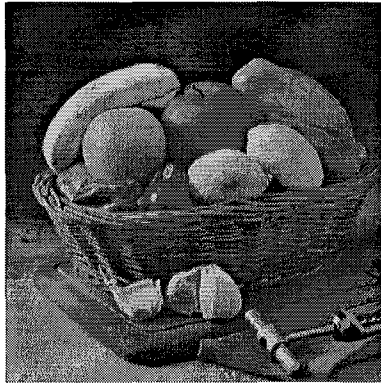


図 2.7 SCID“果物籠”

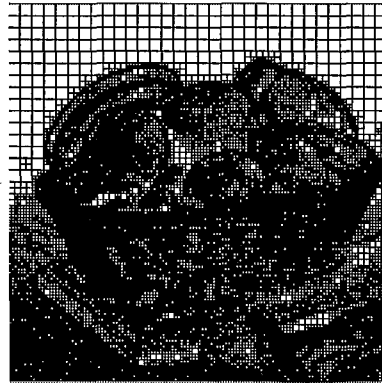


図 2.8 Splited



図 2.9 Grouped

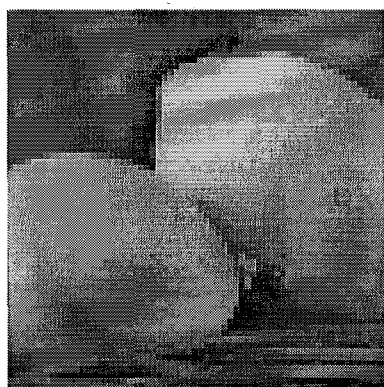


図 2.10 Eliminated

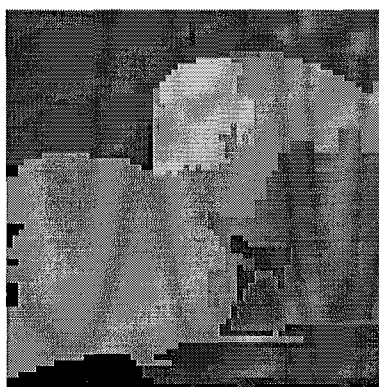
領域に対して merging および grouping を行なった結果を図 2.9 に、さらに 5 画素未満の小領域を除去した結果を図 2.10 に示す。実験では G 成分だけを用い、しきい値を 7 にした。

提案手法ではカラー画像を用いているので、領域内での R, G, B 各階調値の最大値と最小値との差が、すべてしきい値より大きいときに 4 分割し、また隣接する領域を合わせた領域内で R, G, B 各階調値の最大値と最小値との差が、すべてしきい値以下であれば統合する。

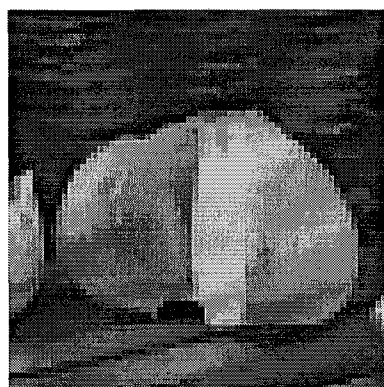
カラー標準画像から 64×64 画素の領域を切り出した画像を図 2.11 に、初期分割を縦横 8 画素間隔で与えたのちに、splitting, merging, および grouping を行ない、さらに 30 画素未満の小領域を除去した結果を図 2.12 に示す。図 2.12 は、各領域にそ



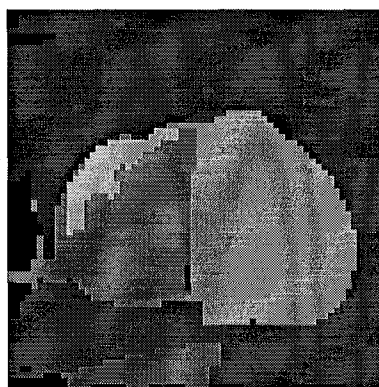
(a) lemon



(a) result 1



(b) garlic



(b) result 2

図 2.11 SCID“果物籠”

図 2.12 実験結果

の R, G, B 平均階調値を割り当てて再構成した画像である。但し、予備実験の結果に基づいて、しきい値を 15 にした。実験結果から、領域の細分化や過併合によって、物体領域の抽出が困難であることが分かる。

2.2.3 動的しきい値処理

特徴空間でのクラスタリングの最も簡単な手法はしきい値処理である。階調値分布が比較的単純な場合でもしきい値の設定は、人が分割結果に応じて行わなければならない。しきい値処理そのものは領域分割法に固有のものではなく、広く画像処理全般で行われる代表的な処理である。

階調画像の領域分割のためのしきい値は、画像全体に対して必ずしも一定ではなく、画像の局所的な性質によって変えなければならない場合が多い。このような画像内の位置に依存したしきい値を用いる手法を、動的しきい値処理 (dynamic thresholding) あるいは局所的しきい値処理 (local thresholding) という。シェーディングのある画像に対しては、動的しきい値処理が特に有効である。Chow & Kaneko [26] は、対象物体領域の境界領域での双峰性ヒストグラムの正規分布の和による最小二乗誤差近似と、そこで得られたしきい値の内挿によって、画像の各点に対してしきい値を動的に決定した。動的しきい値を用いた例には、他に Akita & Kuga [27] などがある。

領域分割性能を検証するために、動的しきい値処理 [26] を図 2.14 に示す指先部分の画像に適用した。図 2.14 の画像は、図 2.13 に示す 512×640 画素 256 階調のカラー標準画像から、図 2.14 に示す 50×60 画素の指先部分を切り出した画像である。図 2.14 の R 成分、 G 成分、 B 成分それぞれについて適用した結果を、図 2.15(a) から (c) に示す。結果は B 成分の階調画像については良好であった。この結果は、肌色同士の色の違いは B 成分に大きく依存していることを示している。しかし、物体領域と背景領域の色に応じて R, G, B のどの成分に基づき処理するのかを決めなければならない。また、ノイズやテクスチャの影響についても考える必要がある。

2.2.4 領域分割における弛緩法

画像の不鮮明さやノイズのために、階調値などの領域に対する特徴量に曖昧さが存在すると、しきい値処理による領域分割では、あまり良い結果が得られない場合がある。この場合、前処理として確率的弛緩法 (probabilistic relaxation) [28] が有効である。



図 2.13 SCID“ポートルート”
(512 × 640 画素)

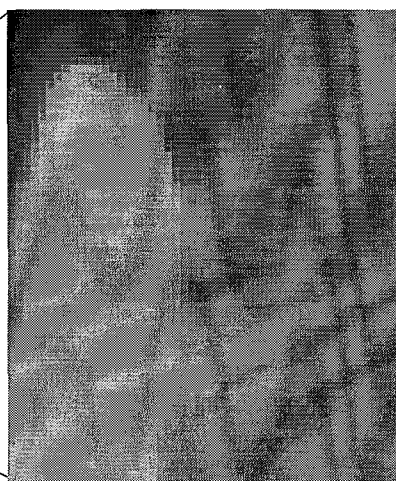


図 2.14 指先部分
(50 × 60 画素)



(a) *R*成分



(b) *G*成分



(c) *B*成分

図 2.15 動的しきい値処理実験結果

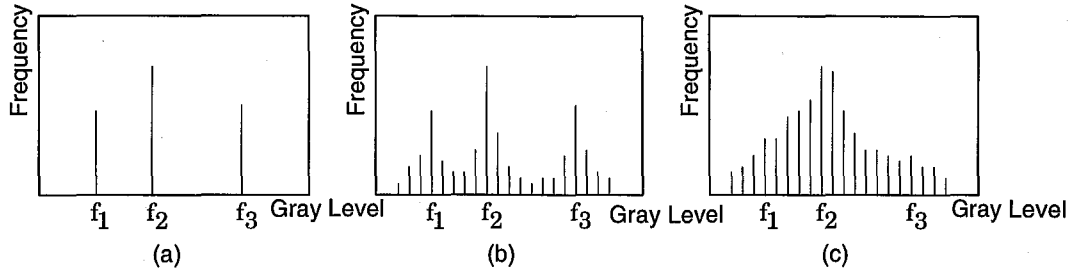


図 2.16 クラスタのヒストグラム分布

一つのクラスタが一つの階調値に対応する理想的な場合の各クラスタの階調値を $f_c (c = 1, 2, 3, \dots, N_c)$ (N_c : クラスタ総数) とする. 図 2.16(a) に示すヒストグラムの画像ならば, 容易にしきい値処理を行うことができる. しかし, 通常一つのクラスタは複数の階調値からなっているために, 図 2.16(b) に示すように各クラスタはヒストグラム上で互いに重なり合う. この場合, しきい値を如何に決定するかが問題となる. 特に, 図 2.16(c) の場合のように一つのピークしかない場合には, しきい値の決定は不可能である.

確率的弛緩法による領域分割の概要

対象画像を階調画像 $f(x, y)$ とし, 画像の階調値を特徴量とした 1 次元の特徴空間を考え, 二つのクラスタへの分割を行なう.

まず, 対象画像すべての画素に対して値域 $[0, 1]$ の初期ラベル確率を式 (2.19) に従って求める.

$$P_i(\lambda_1) = \frac{f_i - b}{w - b} \quad (2.19)$$

a_i : (x_i, y_i) に位置する画素.

f_i : a_i の階調値.

λ_k : クラスタに対応するラベル. ($k = 1, 2$)

$P_i(\lambda_k)$: 画素 a_i のラベルが λ_k である確率. ラベル確率と呼ぶ.

w : $f(x, y)$ の最大の階調値.

b : $f(x, y)$ の最小の階調値.

次に, 画素 a_i のラベル確率を近傍の画素のラベル確率に応じて, より局所的矛

盾が少なくなるように更新する．例えば， λ_1 が優勢な局所領域に大きな $P_i(\lambda_2)$ を持つ画素が孤立して存在している場合，その状態は矛盾している可能性が高いと考え， $P_i(\lambda_2)$ を小さくする．式 (2.20) に示す 8 近傍画素のラベル確率の平均 $Q_i(\lambda_k)$ を用いて，画素 a_i のラベル確率 $P_i^{new}(\lambda_k)$ を式 (2.21) に従って更新する．

$$Q_i(\lambda_k) = \frac{1}{8} \sum_{j=1}^8 P_j(\lambda_k) \quad (2.20)$$

$$P_i^{new}(\lambda_k) = \frac{P_i(\lambda_k)Q_i(\lambda_k)}{\sum_{l=1}^2 P_i(\lambda_l)Q_i(\lambda_l)} \quad (2.21)$$

すべての画素に対するラベル確率の更新を繰り返すことで，ラベル確率が 0 または 1 に収束して，ノイズが除去される．結果的に，しきい値の決定が容易になる．

適当な回数の反復を行なった後に，画素の階調値 f_i^{last} を式 (2.22) に従って求める．

$$f_i^{last} = P_i^{last}(\lambda_1)(G - 1) \quad (2.22)$$

G :対象画素の階調数．

実験結果

図 2.17 に示す画像のヒストグラムを図 2.18 に示す．確率的弛緩法による前処理なしにしきい値を 50 にして，しきい値処理を行なった結果を図 2.19 に，5 回繰り返して処理を行なった結果を図 2.20 に示す．

本研究では，カラー画像を対象としているため，階調画像における確率的弛緩法をカラー画像に応用した．確率的弛緩法を用いてレモンに対応する領域の分割を試みた．あらかじめ与える対象領域内の 1 画素 Q における，画素値と各画素値間での RGB 空間におけるユークリッド距離を，256 レベルに正規化した図 2.21 に示す画像を用いて実験を行なった．+ は Q の位置を表している．各更新回数における f_i^{last} に対して，しきい値を 100 にして 2 値化した結果と f_i^{last} のヒストグラムを，図 2.22 から図 2.24 に示す．図 2.21 のヒストグラムには双峰性がないために，しきい値の決定が困難な例であるが，ラベル確率の更新に従い，ヒストグラムに双峰性が現れ，しきい値の決定が容易になることが分かる．

確率的弛緩法は，画像上で一つの領域を構成する画素の階調値に関して曖昧さが存在すると考え，この曖昧さを近傍の画素の階調値との関係からより矛盾が少なくなるように反復的に修正する．この反復処理により，特徴空間におけるクラスタの広がり小さくなるのでクラスタの発見が容易になり，領域の細分化や過併合を抑制できる可能性がある．



図 2.17 標準画像 f6

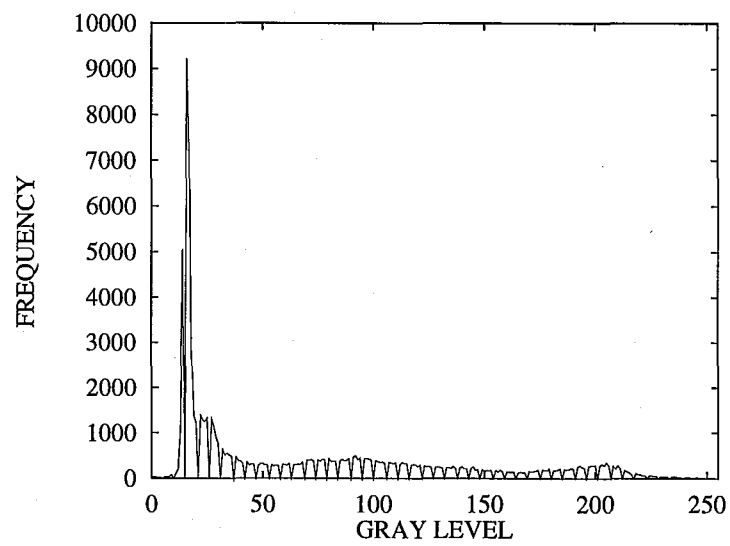


図 2.18 図 2.17 のヒストグラム



図 2.19 しきい値 50

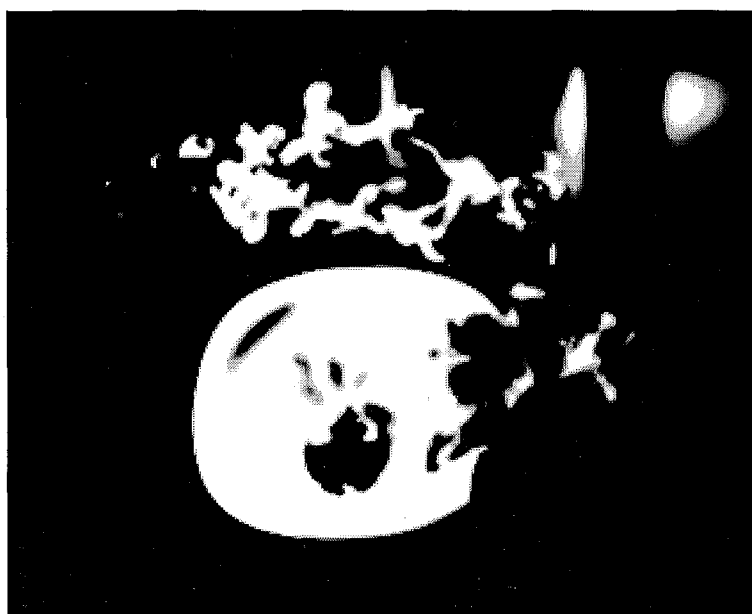


図 2.20 階調画像の処理結果（更新回数 5 回）

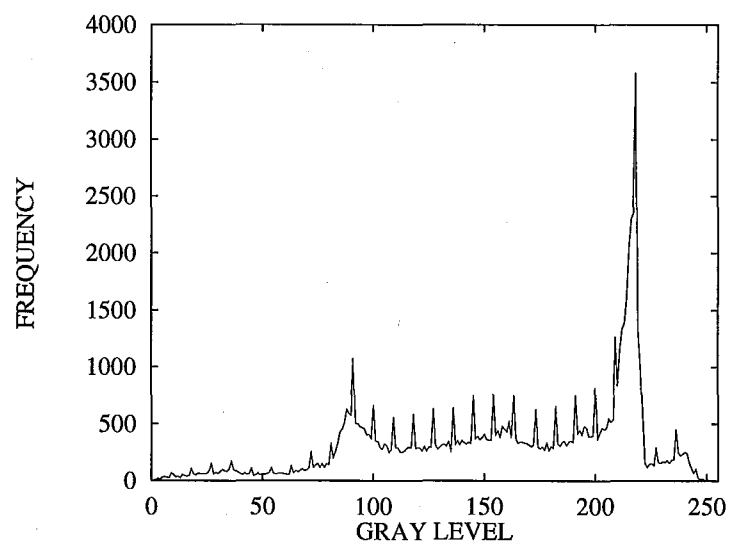


図 2.21 RGB空間ユークリッド距離画像とヒストグラム

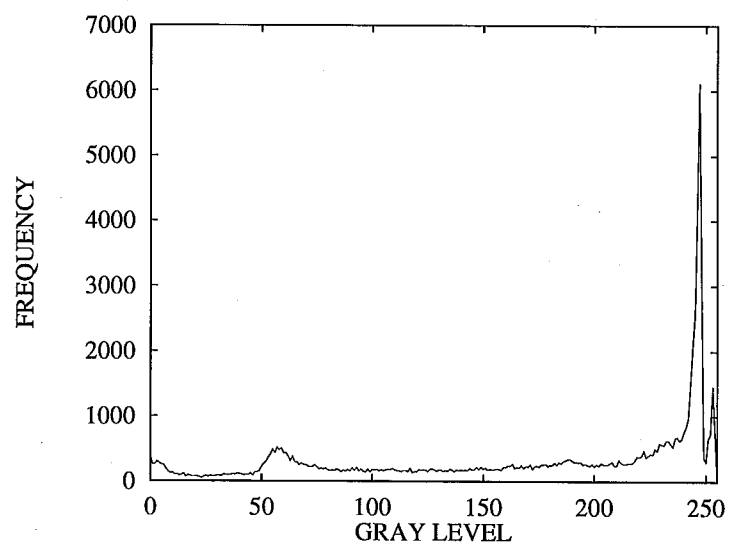
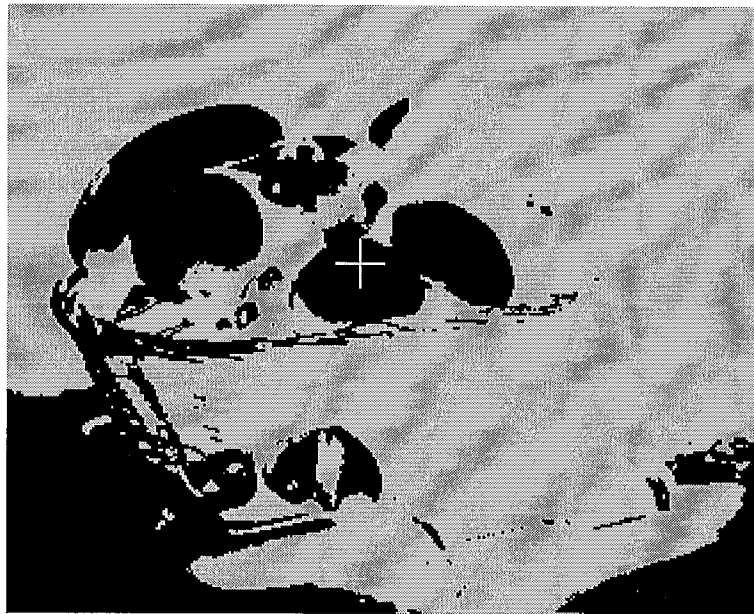


図 2.22 カラー画像の処理結果（更新回数 1 回）

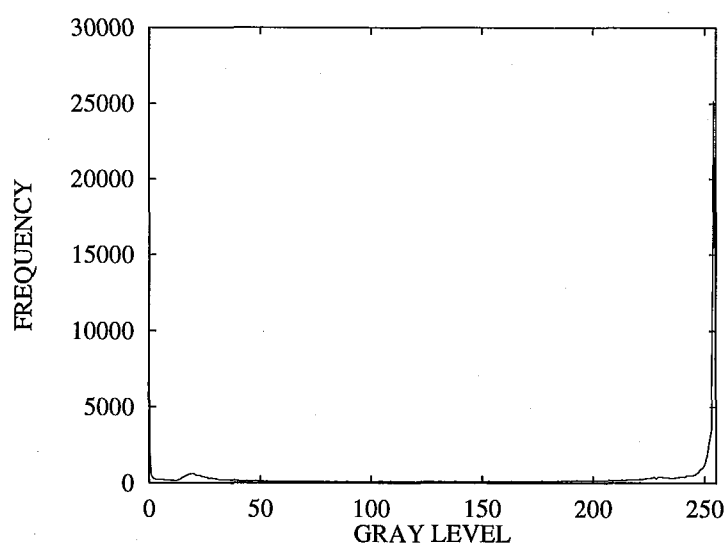


図 2.23 カラー画像の処理結果（更新回数 2 回）

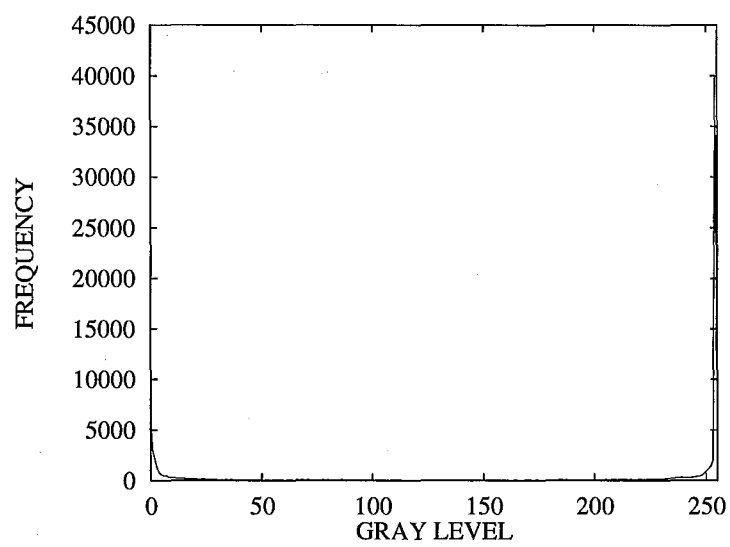


図 2.24 カラー画像の処理結果（更新回数 3 回）

2.3 動的モデルを用いる手法

2.3.1 Snakes の概要

Snakes は、画像平面 (x, y) 上で媒介変数表現された制御点を結んだ輪郭モデル $\mathbf{v}(s) = (x(s), y(s)) (0 \leq s \leq 1)$ である。Snakes では、内部エネルギー $E_{int}(\mathbf{v}(s))$ 、モデルと画像の適合性エネルギー $E_{image}(\mathbf{v}(s))$ 、そして外部強制力を表す外部エネルギー $E_{con}(\mathbf{v}(s))$ によって定義される、式 (2.23) に示すエネルギー関数の値を最小化するようにモデルを変形する。そして、エネルギーの極小状態に対応する輪郭を抽出する。内部エネルギー $E_{int}(\mathbf{v}(s))$ としては、制御点間の距離やモデルの滑らかさなどが用いられ、モデルと画像の適合性エネルギー $E_{image}(\mathbf{v}(s))$ としては、画像の輝度勾配の大きさなどが用いられる。

$$E_{snakes} = \int_0^1 (E_{int}(\mathbf{v}(s)) + E_{image}(\mathbf{v}(s)) + E_{con}(\mathbf{v}(s))) ds \quad (2.23)$$

Snakes には、2.1.1 で述べた手法と比較してノイズの影響を受けにくく、また滑らかに連続した輪郭線が得られるという利点がある。しかし、Snakes を用いて輪郭を抽出する場合、あらかじめ輪郭の近傍に初期輪郭を与えなければならない。また、初期輪郭にできるだけ依存しない手法として、分裂する Snakes [11] が提案されたが、複数の均一な領域からなる物体領域を、ひとまとまりにして抽出することはできない。

2.3.2 Active Net の概要

Active Net [10] のエネルギー関数は、網自身の内部歪みエネルギー E_{int} と、網と画像の適合性エネルギー E_{image} との線形結合によって表される。階調値を用いた場合の E_{image} は $E_{image} = \omega I(x, y)$ で表される。 $I(x, y)$ は格子点 $v(= (x, y))$ における階調値、 ω は実数定数である。 ω の符号によって、格子点が、階調値の高い方へ動くか、あるいは低い方へ動くかが決まる。

画像が、一つの明るい領域と一つの暗い領域から構成されているような場合には、網と画像の適合性エネルギーとして、 $E_{image} = \omega I(x, y)$ を用いて領域を抽出することができる。 $I(x, y)$ は格子点 $v(= (x, y))$ における階調値、 ω は実数定数である。しかし、図 2.25 に示すような画像から抽出する場合には、階調値だけでは不十分である。そこで、文献 [10] では、階調値分布の滑らかさを加味した $E_{image} = \omega G(\text{Max}(I(x, y), V(x, y)))$ を用いた。但し、 G はガウシアンフィルタ、 $V(x, y)$ は格

子点の近傍領域 (9×9 画素) 内での階調値の分散を 256 レベルに正規化した値である. E_{image} は, 階調値が小さくかつ分散の小さい場合に小さな値になり, 階調値が大きいあるいは分散が大きい場合に大きな値になる. これにより, 暗くて滑らかな対象領域を選択的に抽出することができる [10]. 図 2.25 に示した 120×120 画素の階調画像における $V(x, y)$ を図 2.26 に示す. また, 正方格子状の Active Net を用いた場合, 四つの角が残ってしまうので, 放射状 Active Net を用いる.

図 2.25 に示した画像からりんごに対応する領域の抽出を試みた. 各反復回数における結果を図 2.27 から図 2.29 に示す. 但し, 格子点数を 10×35 にした.

Active Net には複数の領域を分離して抽出する能力が備わっていない [10] ため, 内部に対象物体領域が一つだけ含まれるように初期網を配置する必要がある. また, りんごのように一つのほぼ均一な物体領域を抽出することが可能であるが, 分裂する Snakes [11] と同様に, 複数の均一な領域からなる物体領域を, ひとまとまりにして抽出することは不可能である.

2.4 本章のまとめ

本章では, 物体領域抽出のためのアプローチとして, エッジ検出法, 領域分割法, そして動的モデルを用いる手法の概要とそれぞれの代表的な手法について述べた. 代表的な手法としては, 弛緩法, 分離・統合法, 動的しきい値処理法, Snakes, そして Active Net をとりあげ, 実験結果と, その特徴および問題点について述べた.



図 2.25 “果物籠”

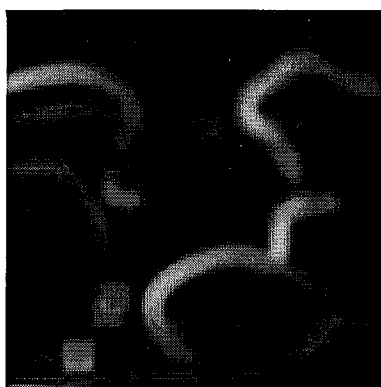


図 2.26 $V(x, y)$

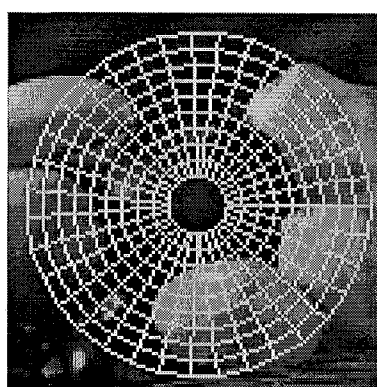


図 2.27 0 回

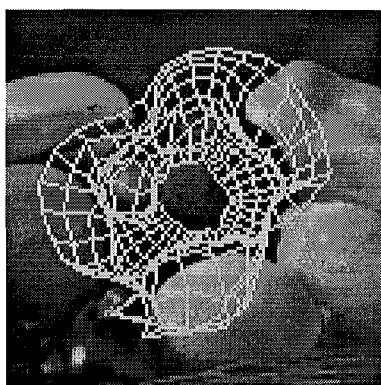


図 2.28 300 回

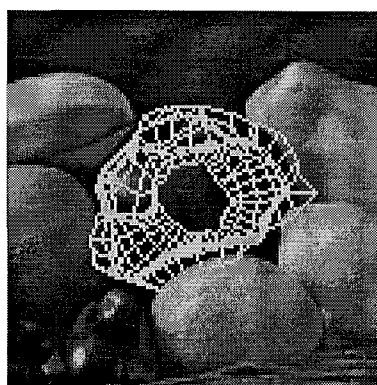


図 2.29 1000 回

第 3 章

Hopfield 型ニューラルネットワークを用いたカラー画像における境界線検出

一部がぼやけたり途切れたりしている境界線は、検出が困難であることが多い。そこで、本章では、カラー画像から、一部がぼやけたり途切れたりしている場合でも、さまざまな形状の境界線を検出することを目的とする。本手法では、境界線上に処理領域を設定する。処理領域を等間隔で分割した小ブロックのそれぞれに、ある長さ以内のエッジ追跡を行い、複数のエッジセグメントを検出する。検出したエッジセグメントに対し Hopfield 型ニューラルネットワークのユニットを割り当て、ニューラルネットワークの目的関数を、エッジセグメント間での色と方向の連続性、そして、エッジセグメントの数に関する制約条件に基づいて定義する。この関数値を極小化することで境界線上にあるエッジセグメントに割り当てられたユニットを発火させる。更に、発火ユニットに従って処理領域を移動する。実験では、ひとつながりの線構造を徐々に形成することで、境界線を良好に検出できることを示す。

3.1 境界線検出アルゴリズム

3.1.1 自動エッジ追跡によるエッジセグメント検出

本手法は、はじめに図 3.1 に示すように、直交座標系 u 軸および v 軸を設定する。次に、境界線上の 1 画素 (u, v) を与え、 S とする。画素 $(u-D+1, v-D+1)$ ($D > 1$) を左上、画素 $(u+D, v+D)$ を右下とする領域を処理領域 W とし、それを縦横適当な間隔 F ($F < D$) で小ブロックに分割する。本論文では予備実験の結果に基づいて、

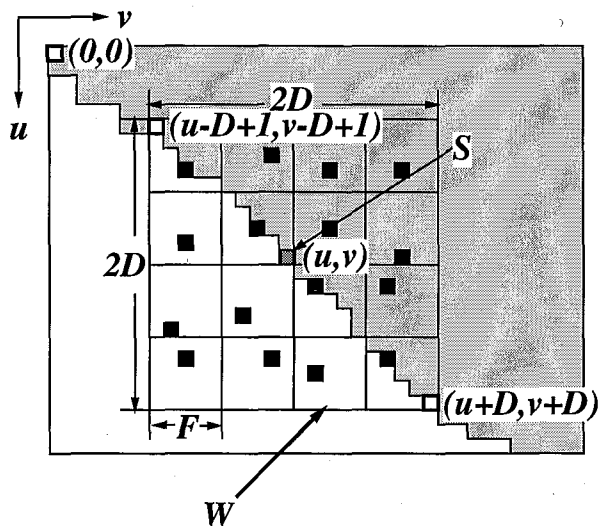


図 3.1 処理領域と小ブロック

$D = 10$, $F = 5$ にする．各小ブロック内の画素をスキャンして，■で示す最も大きく階調値が変化する点を見つけ，エッジ追跡の開始点にする．エッジセグメント検出のためのカラー画像におけるエッジ追跡手法を既に提案した [29]．文献 [29] の手法は，現在追跡しているエッジ画素を中心とする $K \times K$ 画素を， RGB 空間における最近傍法 [30] を用いてクラスタリングする．本論文では予備実験の結果に基づいて， $K = 9$ にする．図 3.2に示す追跡開始点の上下左右の画素 P_a, P_b, P_c, P_d の画素値を調べる． P_a と P_b の画素値間， P_c と P_d の画素値間の RGB 空間におけるユークリッド距離を計算し，それぞれ $L_{a,b}$, $L_{c,d}$ とする．そして $L_{a,b} \geq L_{c,d}$ のとき， P_a と P_b の

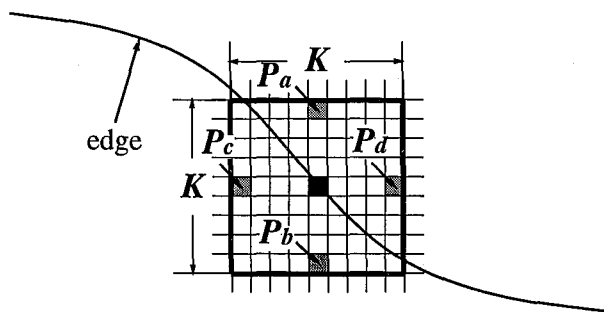


図 3.2 エッジ追跡の初期設定

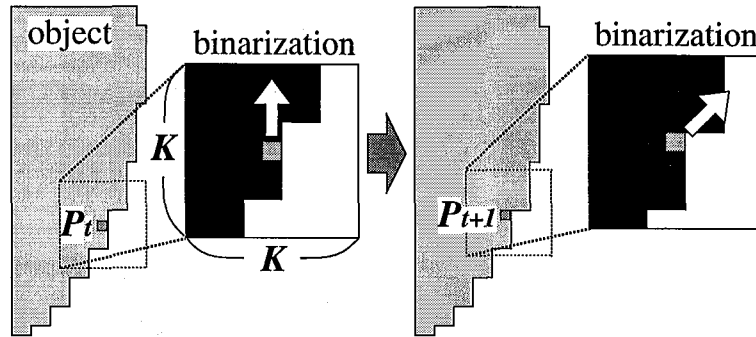


図 3.3 エッジ追跡概念図

一方が物体領域に含まれ、逆に $L_{a,b} < L_{c,d}$ のとき、 P_c と P_d の一方が物体領域に含まれると仮定して一つめのエッジセグメントを検出した後に、他方を物体領域に含まれると仮定して二つめのエッジセグメントを検出する。物体領域に含まれると仮定した画素の RGB 階調値を物体領域に相当するクラス C_{obj} の初期基準 RGB 値とする。つづいて図 3.3 で表される、次の操作 (a) から (c) の一連の処理を 1 ステップとして繰り返す。

- (a) 2 値化：図 3.3 において現在追跡しているエッジ画素 P_t を中心とする $K \times K$ 画素をクラスタリングする。生成された各クラスに属する画素の平均 RGB 値を次の追跡ステップにおけるクラスの初期基準 RGB 値にする。そして、黒く塗りつぶした C_{obj} に属する画素にラベル 1 を、その他すべての画素にラベル 0 を割り当て局所的に 2 値化する。
- (b) 追跡方向決定：(a) で得られた 2 値パターンを用いて、図 3.3 の矢印で示すように追跡方向を決定する。追跡方向は、視覚におけるマッハ効果を考慮して設計した、図 3.4 に示す 45° きざみ八つのパターンマッチングフィルタ $filter(i)$ ($i = 1, 2, \dots, 8$) を用いて決定する。最大値を出力したフィルタに従って、図 3.5 の矢印の方向を追跡方向とする。
- (c) 次のエッジ画素の決定： P_t の 8 近傍画素の中で、(b) で決定した方向にある画素を新たなエッジ画素 P_{t+1} とする。

本論文では、追跡ステップ数の上限をあらかじめ設定する。また、 C_{obj} の要素数が 10 よりも少ない場合には、エッジを見失ったと仮定してエッジ追跡を停止する。そして、適当な長さの追跡結果だけをエッジセグメントとする。従って、一つの開始

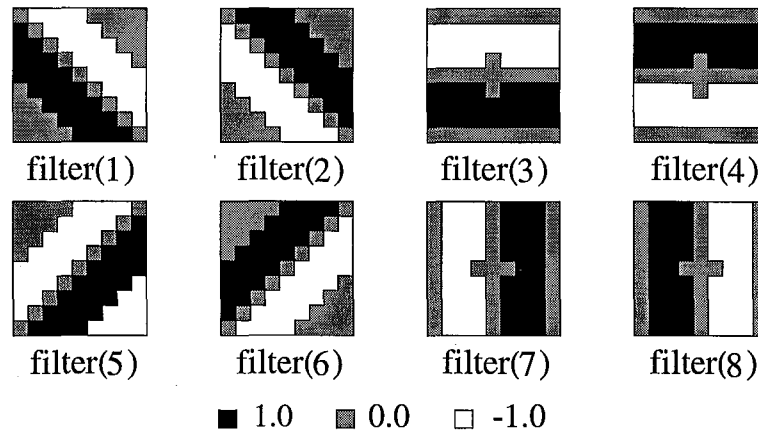


図 3.4 パターンマッチングフィルタ

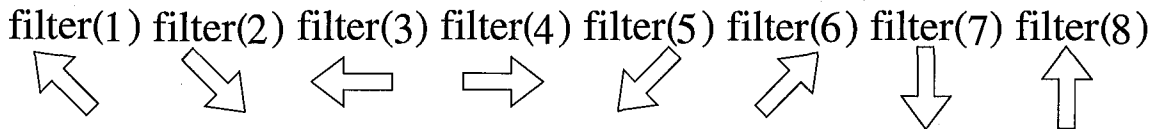


図 3.5 追跡方向

点から 2 方向にエッジ追跡を行い、小ブロックあたりたかだか二つのエッジセグメントを検出する。そして、図 3.6 に示す追跡方向に向かって左側領域 A の平均 RGB 階調値 $\overline{R_l}, \overline{G_l}, \overline{B_l}$ を物体色、右側領域 B の平均 RGB 階調値 $\overline{R_r}, \overline{G_r}, \overline{B_r}$ を背景色と呼び、エッジセグメントの色に関する属性とする。また、操作 (b) で決定した追跡方向の平均値をエッジセグメントの方向とする。

3.1.2 Hopfield 型ニューラルネットワークを用いた境界線上エッジセグメント選択

本節では、Hopfield 型ニューラルネットワークを用いて、検出したエッジセグメントの中から境界線上にあるエッジセグメントを選択する手法について述べる。

本手法では、エッジセグメントの数・物体色・背景色・方向を総合的に評価してエッジセグメントを選択する。そこで、同一物体領域を仮定した場合の境界線上にあるエッジセグメントは仮定条件 1 から仮定条件 5 に示す 5 つの条件を満たしてい

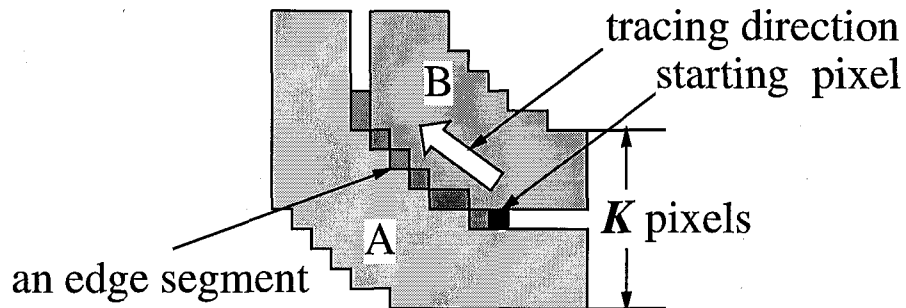


図 3.6 エッジセグメントの色

ると仮定する。仮定条件を表す目的関数を定義して、それを最小化することで、エッジセグメントを選択する。

仮定条件 1: 境界線は、 W を直線的に横切っていることを仮定する。 W の決定法とエッジセグメントの検出法より、 W 内の境界線上には $2D/F$ 個のエッジセグメントが存在することになる。これを仮定すれば、境界線が途切れたりぼやけたりしている場合でも $2D/F$ 個のエッジセグメントによって境界線を構成することができる。また、その他のエッジセグメントをノイズとして除去することができる。

仮定条件 2: 物体領域において色が均一であることを仮定する。すなわち、 W 内で同一物体領域を仮定した場合の境界線上にある二つのエッジセグメントは、物体色が互いに等しいと仮定する。

仮定条件 3: 物体領域における色の均一性だけでなく、背景領域における色の均一性も仮定する。そこで、 W 内で同一物体領域を仮定した場合の境界線上にある二つのエッジセグメントは、物体色だけでなく背景色も互いに等しいと仮定する。

仮定条件 4: 境界線が滑らかであることを仮定する。まず、第 k エッジセグメントが図 3.7 の破線で示す位置にある場合に、第 j エッジセグメントを構成する画素の平均座標と、第 k エッジセグメントを構成する画素の平均座標を結ぶ直線の方角を、図 3.7 の実線の矢印で示すように決定する。このとき、第 j エッジセグメントと第 k エッジセグメントの方角がエッジセグメント間を結ぶ直線の方角に等しいと仮定する。

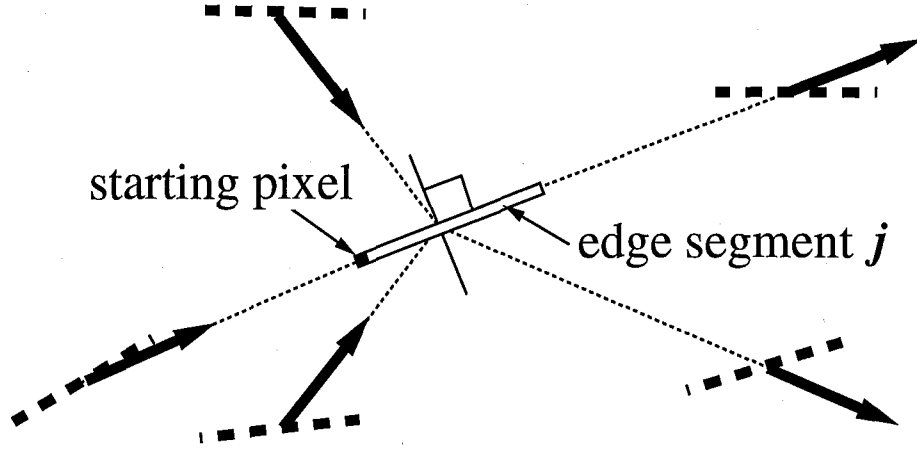


図 3.7 エッジセグメント間を結ぶ直線の方角の定義

仮定条件 5：物体領域と背景領域とは、色が異なっていると考えられる。そこで、任意のエッジセグメントは、物体色と背景色が異なっていると仮定する。

各エッジセグメントに図 3.8 に示す Hopfield 型ニューラルネットワークの一つのユニットを割り当て、第 i ユニットの出力値 U_i が 1 のとき第 i エッジセグメントが境界線上に存在し、 U_i が 0 のとき存在しないと仮定すれば、式 (3.1) から式 (3.5) で表される E_{conn} , E_{left} , E_{right} , E_{dire} , E_{edge} の値が小さくなるほど、仮定条件 1 から仮定条件 5 にそれぞれ近付いたといえる。

$$E_{conn} = \left\{ \sum_{j=1}^N U_j - 2D/F \right\}^2 \quad (3.1)$$

$$E_{left} = \sum_{j=1}^N \sum_{k=1}^N \{ (\overline{Rl_j} - \overline{Rl_k})^2 + (\overline{Gl_j} - \overline{Gl_k})^2 + (\overline{Bl_j} - \overline{Bl_k})^2 \} U_j U_k \quad (3.2)$$

$$E_{right} = \sum_{j=1}^N \sum_{k=1}^N \{ (\overline{Rr_j} - \overline{Rr_k})^2 + (\overline{Gr_j} - \overline{Gr_k})^2 + (\overline{Br_j} - \overline{Br_k})^2 \} U_j U_k \quad (3.3)$$

$$E_{dire} = \sum_{j=1}^N \sum_{k=1}^N \{ \theta_{jk} + \theta_{kj} \} U_j U_k \quad (3.4)$$

$$E_{edge} = - \sum_{j=1}^N T \{ (\overline{Rl_j} - \overline{Rr_j})^2 + (\overline{Gl_j} - \overline{Gr_j})^2 + (\overline{Bl_j} - \overline{Br_j})^2 \} U_j \quad (3.5)$$

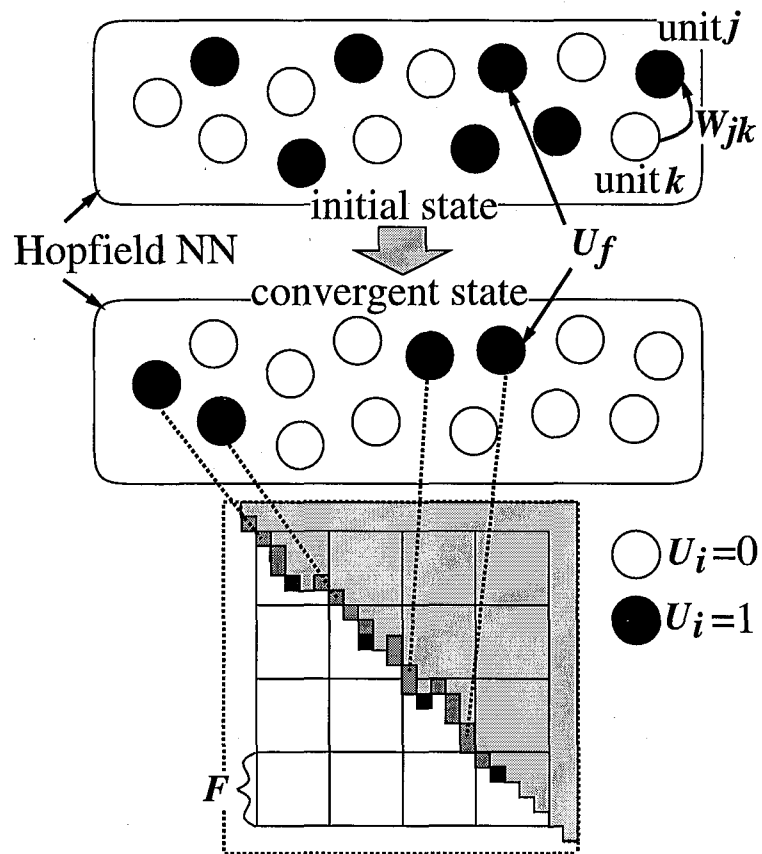


図 3.8 境界線検出概念図

N : エッジセグメント総数.

j, k : $1 \sim N$ の整数.

θ_{jk} : 第 j エッジセグメントと第 k エッジセグメントを結ぶ直線と
第 k エッジセグメントとの方向差 ($0 \sim \pi$ ラジアン).

$$T(x) = \begin{cases} x & \text{if } x < t \\ t & \text{otherwise} \end{cases} \quad (3.6)$$

式 (3.6) は、物体色と背景色との違いに上限を設定して、エッジ強度が強い境界線を常に検出してしまう傾向を抑制するために定義する. しきい値 t は、予備実験の結果に基づいて 1000 にする.

一方、Hopfield 型ニューラルネットワークにおいては、 n 回目の更新における第 j ユニットの出力値 $U_j(n)$ を式 (3.7) に従って決定すれば、式 (3.8) に示すエネルギー関数 E_{NN} の値は決して増加しない (付録 A 参照).

$$U_j(n) = \begin{cases} 1 & \text{if } \sum_{k=1}^N w_{jk} U_k(n-1) - h_j > 0 \\ 0 & \text{otherwise} \end{cases} \quad (3.7)$$

$$E_{NN} = -\frac{1}{2} \sum_{j=1}^N \sum_{k=1}^N w_{jk} U_j U_k + \sum_{j=1}^N h_j U_j \quad (3.8)$$

w_{jk} : 第 k ユニットから第 j ユニットへの結合荷重.

h_j : 第 j ユニットのしきい値.

Hopfield 型ニューラルネットワークの目的関数 E_{obj} を式 (3.9) に示すように定義する.

$$E_{obj} = \alpha E_{conn} + \beta_1 E_{left} + \beta_2 E_{right} + \gamma E_{dire} + \delta E_{edge} \quad (3.9)$$

$\alpha, \beta_1, \beta_2, \gamma, \delta$: 正の実数.

但し、境界線上のエッジセグメントに対応するユニット $2D/F$ 個が 1 を出力している状態において、あるユニットの出力値を変化させた場合の、式 (3.2) から式 (3.5) の変化量の平均値をそれぞれ $\overline{\Delta E_{left}}$, $\overline{\Delta E_{right}}$, $\overline{\Delta E_{dire}}$, $\overline{\Delta E_{edge}}$ とした場合に、 E_{obj} 内の各項が、同等にはたらくようにするために式 (3.10) と式 (3.11) を仮定する.

$$\beta_1 \overline{\Delta E_{left}} = \beta_2 \overline{\Delta E_{right}} = \gamma \overline{\Delta E_{dire}} \quad (3.10)$$

$$\beta_1 \overline{\Delta E_{left}} + \beta_2 \overline{\Delta E_{right}} + \gamma \overline{\Delta E_{dire}} - \delta \overline{\Delta E_{edge}} = 0 \quad (3.11)$$

E_{obj} が E_{NN} と恒等的に等しくなるように、係数比較によって w_{jk} および h_j を決定し、Hopfield 型ニューラルネットワークを構成すれば、ランダムに選んだユニッ

トの状態更新によって、 E_{obj} が極小値になる $U_i (i = 1, 2, \dots, N)$ を求めることができる。 w_{jk} を式 (3.12) に、 h_j を式 (3.13) に示す。

$$\begin{aligned} w_{jk} = & -2\alpha - 2\beta_1 \{(\overline{Rl_j} - \overline{Rl_k})^2 + (\overline{Gl_j} - \overline{Gl_k})^2 + (\overline{Bl_j} - \overline{Bl_k})^2\} \\ & - 2\beta_2 \{(\overline{Rr_j} - \overline{Rr_k})^2 + (\overline{Gr_j} - \overline{Gr_k})^2 + (\overline{Br_j} - \overline{Br_k})^2\} \\ & - 2\gamma \{\theta_{jk} + \theta_{kj}\} \end{aligned} \quad (3.12)$$

$$h_j = -4\alpha \frac{D}{F} - \delta T \{(\overline{Rl_j} - \overline{Rr_j})^2 + (\overline{Gl_j} - \overline{Gr_j})^2 + (\overline{Bl_j} - \overline{Br_j})^2\} \quad (3.13)$$

また、本手法では処理を開始する際に、検出しようとする境界線上にある一つのエッジセグメントに対応するユニットを手動で与え、図 3.8 の U_f で示したように状態更新の間、常にその出力値を 1 に固定して、固定発火ユニットとする。Hopfield 型ニューラルネットワークの初期状態としてランダムな発火状態を与え、収束状態において、 U_f に対応するエッジセグメントを含む、一つの境界線上にあるエッジセグメントに対応するユニットを発火させる。

3.1.3 処理領域移動方法

図 3.1 に示した処理領域 W 内で、境界線上に位置する数個のエッジセグメントに対応するユニットを発火させることができる。そして、その発火ユニットに対応するエッジセグメントの位置に基づいて W を移動する。図 3.9 に示す現在の発火ユニットに対応するエッジセグメントの平均座標 $C2$ を基点として、前回の発火ユニットに対応するエッジセグメントの平均座標 $C1$ から $C2$ へのベクトルの方向に、距離 D 画素の位置にある画素を新たな S とする。移動後は黒丸で示す現在の発火ユニットを U_f にする。但し、最初の処理領域においては U_f を手動で与え、画像中での位置、色、方向を属性として割り当てる。

3.2 実験結果

境界線上にあるエッジセグメントを選択できることを示すために、図 3.10 に示す 512×640 画素、256 階調のカラー標準画像から、 70×70 画素の指先部分を切り出した後に、それを 3 倍に拡大した図 3.11 に示す 210×210 画素の画像全体を処理領域として本手法を適用した。エッジ追跡を行い、総数 452 個のエッジセグメントを検出した。但し、本論文では、エッジセグメントの長さを 4 画素以上 10 画素以下とした。

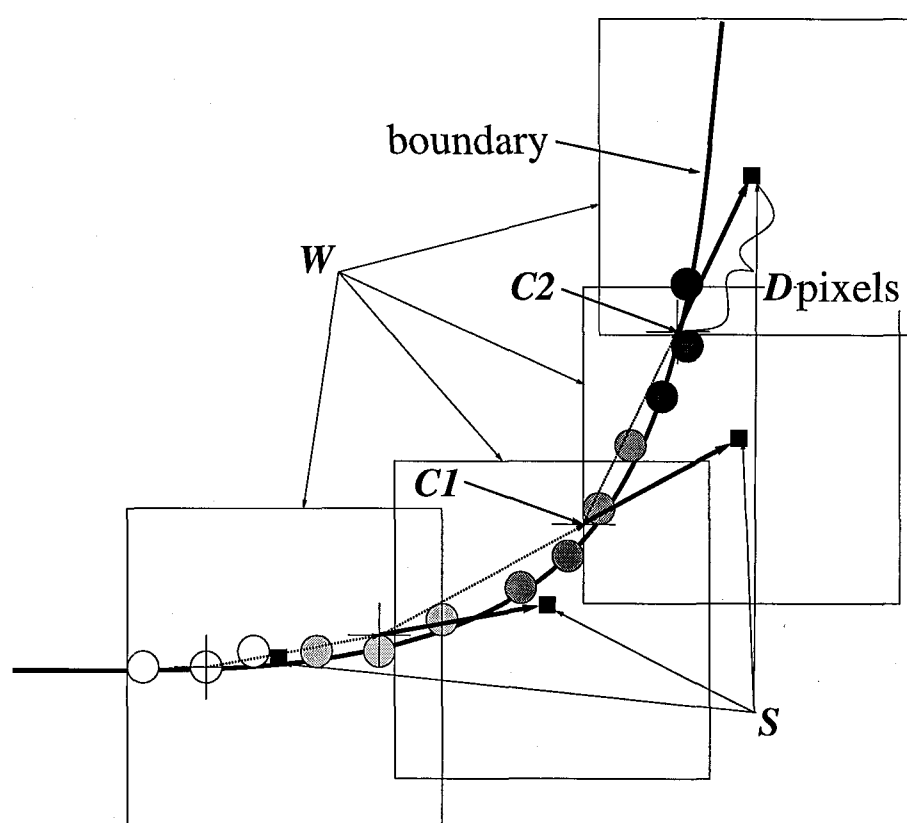


図 3.9 移動概念図



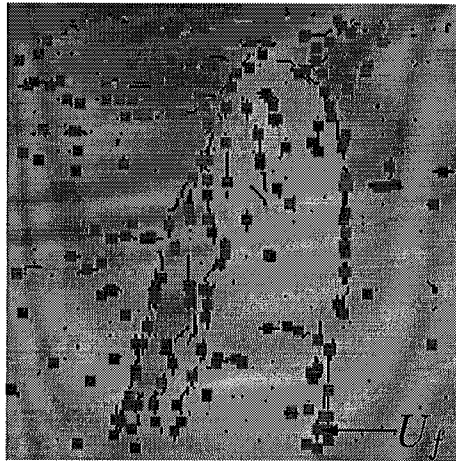
図 3.10 SCID“ポートレート”
(512 × 640 画素)



図 3.11 3 倍に拡大した指先部分
(210 × 210 画素)

境界線を構成するエッジセグメントの属性値を調べた結果に基づき、 $\overline{\Delta E_{left}} = 600$, $\overline{\Delta E_{right}} = 10000$, $\overline{\Delta E_{dire}} = 4$, $\overline{\Delta E_{edge}} = 700$ に設定し、式 (3.10) および式 (3.11) において $\beta_1 = 4.0$ にした場合、 $\beta_2 = 0.24$, $\gamma = 600.0$, $\delta = 10.0$ が求まった。 α を大きくすると、 $2D/F$ 個のユニットが常に発火するようになり、ぼやけや途切れに対して効果的であるが、解の質は低下してしまう。そこで、発火ユニット数と解の質を考慮した予備実験の結果に基づいて、 $\alpha = 100.0$ として Hopfield 型ニューラルネットワークを構成した。但し、境界線が分岐している場合は、パラメータの影響を考慮しなければならない。例えば方向の連続性、すなわち境界線の滑らかさを重視する場合には、 γ の値を大きくする必要がある。

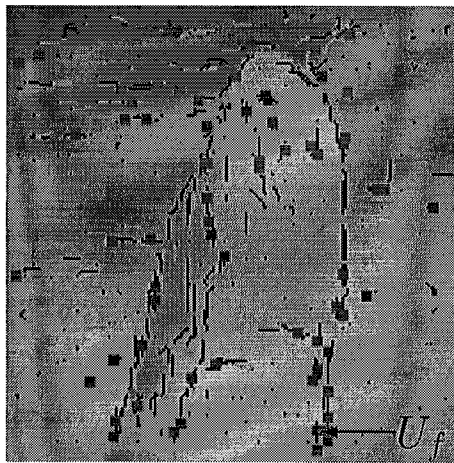
一つのユニットの状態更新を 1 回とし、1 回、100 回、1000 回、10000 回の各更新回数における Hopfield 型ニューラルネットワークの発火状態を図 3.12(a) から (d) に示す。黒く塗りつぶした四角は発火ユニットに対応するエッジセグメントの位置を表している。括弧内に、式 (3.8) に示したエネルギー関数 E_{NN} の各更新回数における値を 10000 で割った値を示す。肌色どうしの境界線において、最も大きく階調値が変化する青色成分だけを用い、10 回の繰り返し処理による確率的弛緩法 [3] を適用した結果を図 3.13 に、本手法を適用した結果を図 3.14 に示す。正方形で囲んだ領域は W を表している。×は、最初に与える W の位置を表している。CPU タイムは、本手法が確率的弛緩法の約 10 分の 1 であった。結果から、本手法を用いて高速



(a) 1 time step (11595)



(b) 100 time steps (9934)



(c) 1000 time steps (720)



(d) 10000 time steps (-24)

図 3.12 ネットワークの発火状態

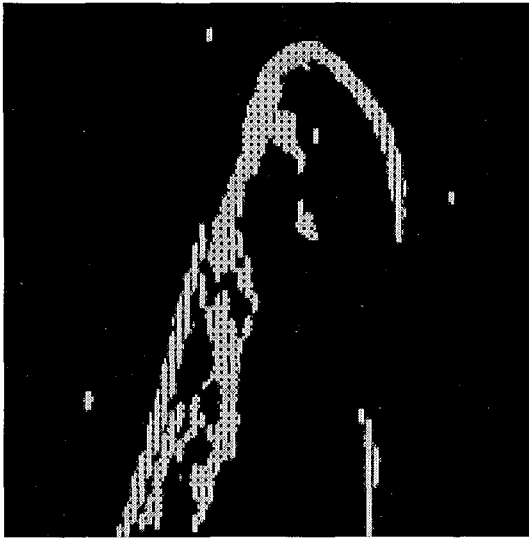


図 3.13 弛緩法

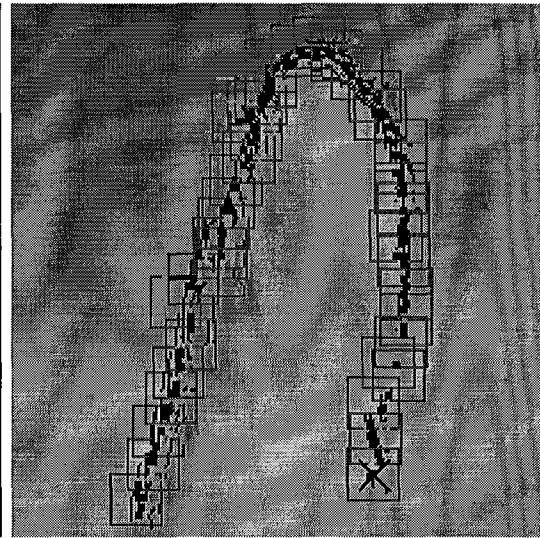


図 3.14 提案手法

かつ高精度に境界線を検出できることが分かる。

Hopfield 型ニューラルネットワークを導入する利点について、全探索との計算コストの比較という観点から述べる。仮りに、一つの開始点から常に二つのエッジセグメントを検出するとすれば、本実験では、 $(2D/F)^2 \times 2 = 32$ 個のエッジセグメントを検出することになる。32 個のエッジセグメントの中から $2D/F = 4$ 個のエッジセグメントを選択する場合、固定発火ユニットを与えることを考えると、組合せの数は ${}_{32}C_4 = 4960$ である。従って、全探索では 4960 回、式 (3.8) のエネルギー関数値を計算しなければならない。一方、本実験では、各 W におけるユニット更新回数を、予備実験で収束を確認できた 1000 回に設定した。従って、式 (3.8) の計算コストは式 (3.7) の約 $N = 32$ 倍であることから、本手法の計算コストは、全探索の場合の $1000/4960/32$ 、すなわち約 $1/150$ となる。但し、Hopfield 型ニューラルネットワークによって求まる解は、初期状態に依存する極小解であり、初期状態に応じて異なる収束解が得られるという問題が生じることがある。しかし、これは解が一意

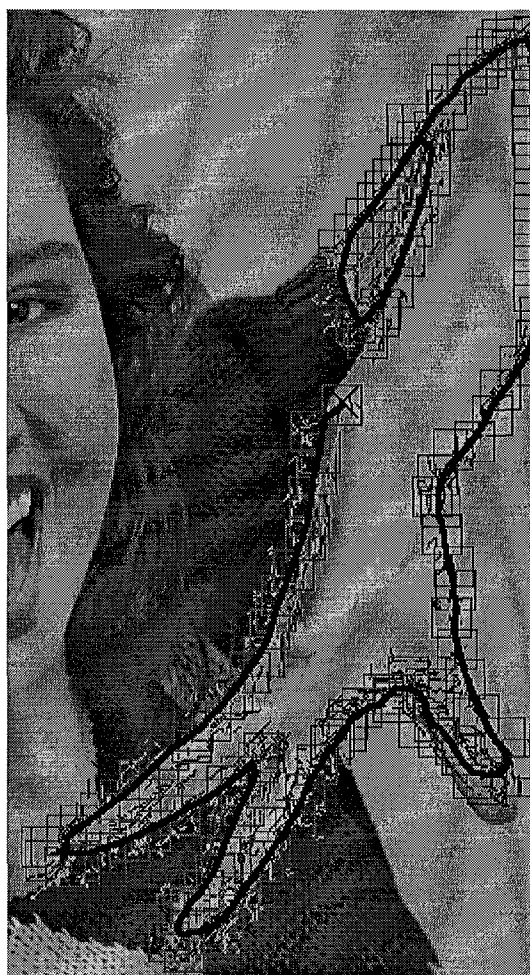
には決まらないという画像処理本来の問題でもある。本手法では、固定発火ユニットによって探索空間を限定するため、処理領域内で境界線が分岐している場合を除き、安定な検出が可能である。また予備実験においても、その安定性が確認された。

本手法を用いて、図 3.10の画像から三つの境界線を検出した結果をそれぞれ図 3.15(a) から (c) に示す。但し、画像の端部分においては、エッジセグメントを仮に生成し、その位置における RGB階調値を属性として割り当てることによって処理を継続した。境界線は、隣接する処理領域において選択されたエッジセグメントの平均座標を、幅 3 画素の直線で結び表現した。

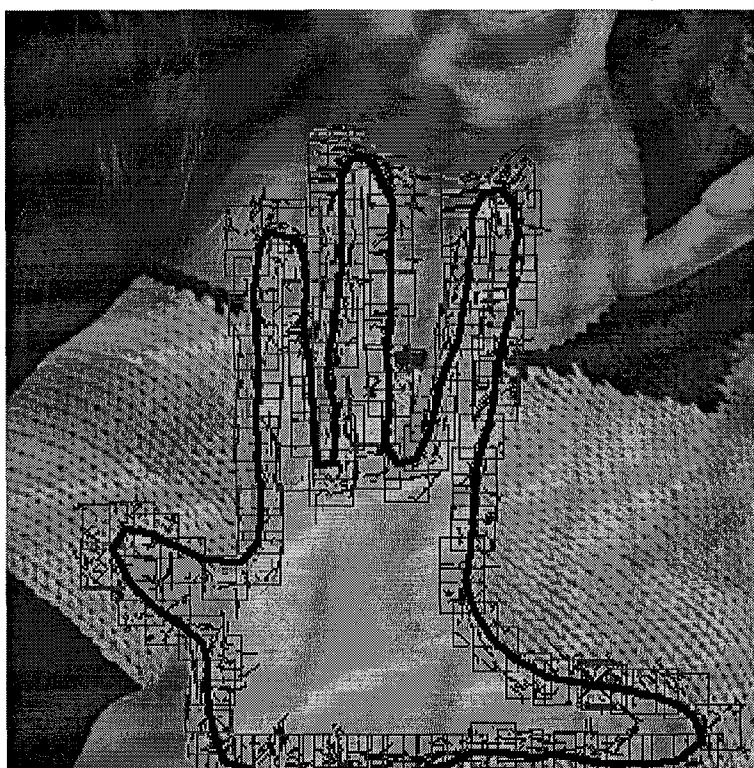
曇天時の湖畔を撮影したカラー写真をイメージスキャナで取り込んだ、図 3.16(a) に示す 600×300 画素、256 階調のカラー画像に対して本手法を適用した。図 3.16(b) に、Sobel フィルタによってエッジ強調した後に 2 値化した結果を示す。図 3.16(c) と (d) に、本手法を用いて境界線を検出した結果を示す。矢印は、 W が移動した方向である。実験結果から、一部がぼやけたり途切れたりしている場合でも、本手法によって山や湖の境界線を検出できていることが分かる。



(a) Face

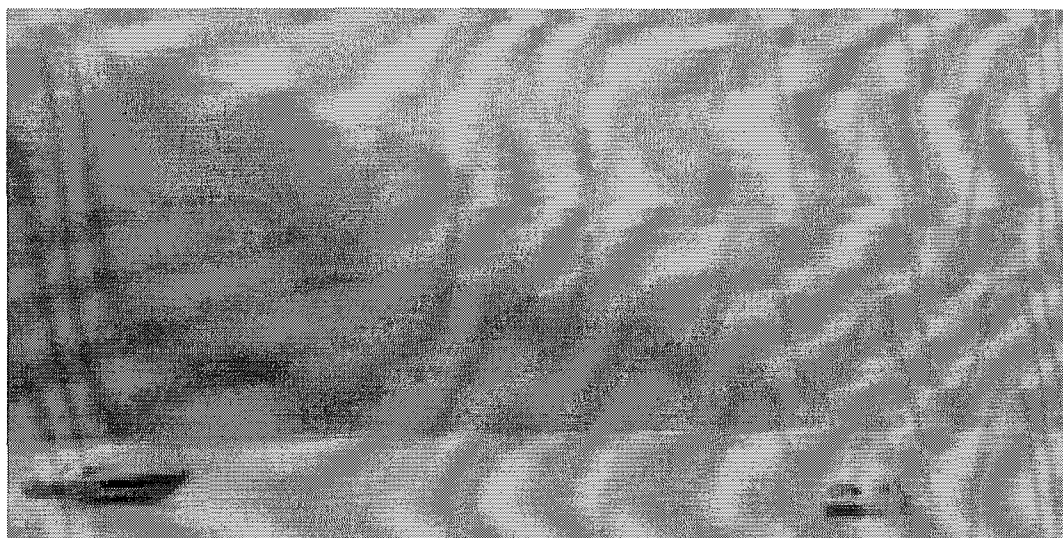


(b) Left arm and hand

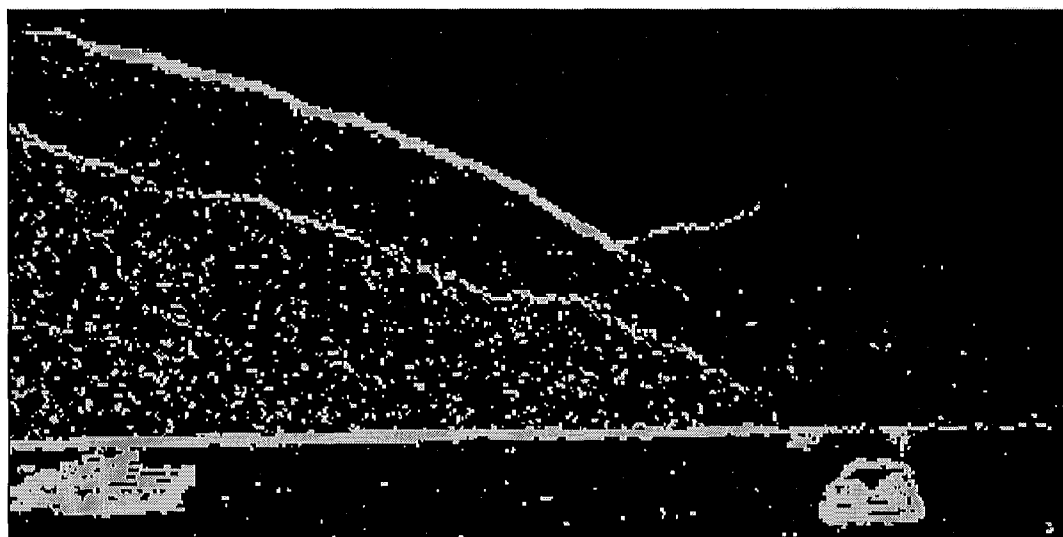


(c) Right hand

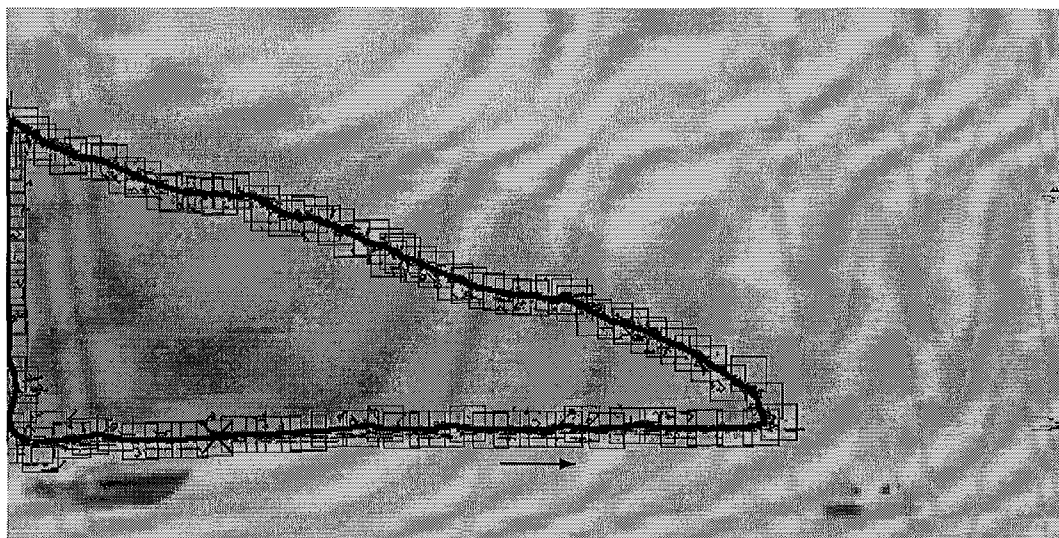
図 3.15 境界線検出結果 1



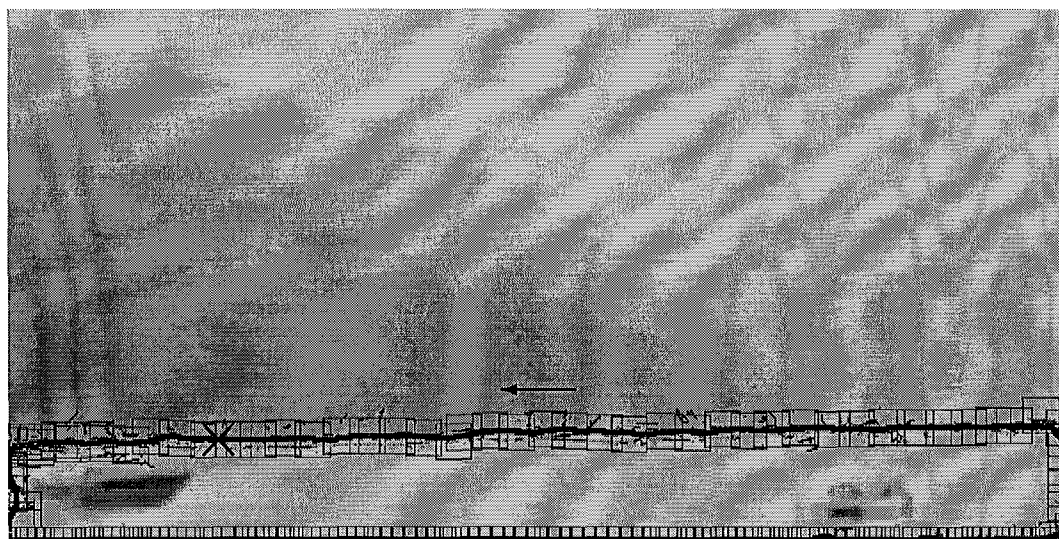
(a) Cloudy lakeside



(b) Sobel



(c) Hill



(d) Lake

図 3.16 境界線検出結果 2

3.3 Snakes との対比

閉じた境界線の検出という観点からは、Kass らによる Snakes[9] が有効な方法として用いられる。しかし、輪郭モデルの閉曲線構造を保ちながら解を求めるため、背景の影響を受けやすく、局所解に陥る可能性が高い。更に、複雑な形状の境界線検出のためには制御点の数や配置の仕方に関して課題が残されている。

Snakes を図 3.10 の画像に適用した。実験では、Snakes のエネルギー関数として、本手法における Hopfield 型ニューラルネットワークの目的関数と同等の関数を定義する。 n 個の制御点の位置を $(x(s), y(s)) (s = 1, 2, \dots, n)$ で表し、エネルギー関数 E_{snakes} を画像エネルギー項 E_{image} と内部エネルギー項 E_{int} を用いて式 (3.14) に示すように定義する。

$$E_{snakes} = \sum_{s=1}^n \{E_{image}(s) + E_{int}(s)\} \quad (3.14)$$

Snakes に関する研究では濃淡画像を対象とする場合が多く、 E_{image} として階調値の 1 次微分を用いる場合が多い。本論文ではカラー画像を対象とすることや、テクスチャやノイズの影響を受けにくくするために、輪郭線内部領域における制御点近傍平均 RGB 階調値 $\overline{R_{in}}, \overline{G_{in}}, \overline{B_{in}}$ と外部領域における制御点近傍平均 RGB 階調値 $\overline{R_{out}}, \overline{G_{out}}, \overline{B_{out}}$ との差を用いて E_{image} を式 (3.15) に示すように定義する。

$$E_{image}(s) = -\phi(s) \{ |\overline{R_{in}}(s) - \overline{R_{out}}(s)| + |\overline{G_{in}}(s) - \overline{G_{out}}(s)| + |\overline{B_{in}}(s) - \overline{B_{out}}(s)| \} \quad (3.15)$$

また本手法では、最初の処理領域における固定発火ユニットの属性値として物体領域の RGB 階調値を与える。これと同等にするために、 $\overline{R_{in}}, \overline{G_{in}}, \overline{B_{in}}$ とあらかじめ与える RGB 階調値との差がすべて 30 よりも大きい制御点 s については $\phi(s)$ の値を 0 にして、輪郭線以外のエッジに制御点が捕まることをできるだけ避ける。

E_{int} としては、隣接する制御点の間隔を収縮させるためのエネルギー項と、式 (3.16) に示す滑らかな曲線にするためのエネルギー項 E_{smooth} の和を用いる場合が多いが、凹部を検出することができない。但し、 $x(n+1) = x(1)$, $y(n+1) = y(1)$, $x(0) = x(n)$, $y(0) = y(n)$ である。

$$E_{smooth} = \varphi(s) \{ (x(s+1) - 2x(s) + x(s-1))^2 + (y(s+1) - 2y(s) + y(s-1))^2 \} \quad (3.16)$$

そこで本論文では、Snakes の内側に向かって制御点を移動させるための内部エネルギー項として面積項 [11] E_{area} を式 (3.17) に示すように定義して凹部を検出する。

そして, E_{smooth} と E_{area} の和を E_{int} とする.

$$E_{area} = -\psi(s)\{x(s)(y(s+1)-y(s-1))-(x(s+1)-x(s-1))y(s)\} \quad (3.17)$$

従って, E_{conn} と Snakes の閉曲線構造, E_{dire} と E_{smooth} , E_{edge} と E_{image} がそれぞれ対応する. 更に Snakes は E_{area} によって収縮する. 本手法における E_{left} および E_{right} に対応するエネルギーについては, 予備実験において良い結果が得られなかったので定義しない.

図 3.17(a) に示す初期輪郭を与えて, Snakes を適用した結果を図 3.17(b) に示す. 図 3.17(c) に示す初期輪郭を与えて, Snakes を適用した結果を図 3.17(d) に示す. 各パラメータは, 予備実験の結果に基づき $\phi(s) = 10.0$, $\varphi(s) = 1.0$, $\psi(s) = 1.0$ に設定した. また, 制御点の数を可変にして, その間隔が 10 画素になるようにした.

Snakes における最大の問題は, 画像に応じてインタラクティブに初期輪郭を設定しなければならないことである. 特に自然画像においては, 背景が複雑であったり, 輪郭上の位置によってエッジ強度が大きく異なる場合が多く, すべての制御点に対して同じパラメータを設定したのでは, 図 3.17(b) の指先部分のように, 輪郭線の正確な検出は困難である. 例えば, エッジ強度の弱い部分を検出するために ϕ を大きくすると, 複雑な背景の部分で輪郭以外のエッジに捕まってしまう, 逆に複雑な背景の部分で輪郭以外のエッジに捕まらないように ϕ を小さくすると, エッジ強度の弱い部分を検出することができなくなる. 更に, 弱いエッジ強度の輪郭線をそれよりも強いエッジ強度をもつ背景の中から検出することは不可能である. この原因としては, Snakes の閉曲線構造が考えられる. すなわち, 制御点の移動に基づいているために, 移動経路上の他のエッジに制御点が捕まる場合がある. 従って, 図 3.17(c) に示すように, 境界線の複雑さに応じて充分な数の制御点を境界線近傍に与えなければならない. 一方, 本手法は, 処理領域内で線構造を徐々に形成しながら解を求める方法である. 従って, 図 3.15 および図 3.16 に示したように, ノイズや背景の複雑さの影響を受けにくい. また, Snakes は制御点の移動経路にあるすべての領域を処理するが, 本手法は境界線近傍を処理するだけで境界線を検出できる.

3.4 本章のまとめ

局所的 2 値化と追跡方向決定を繰り返すエッジ追跡手法によって, カラー画像から複数のエッジセグメントを検出した. 更に, Hopfield 型ニューラルネットワーク

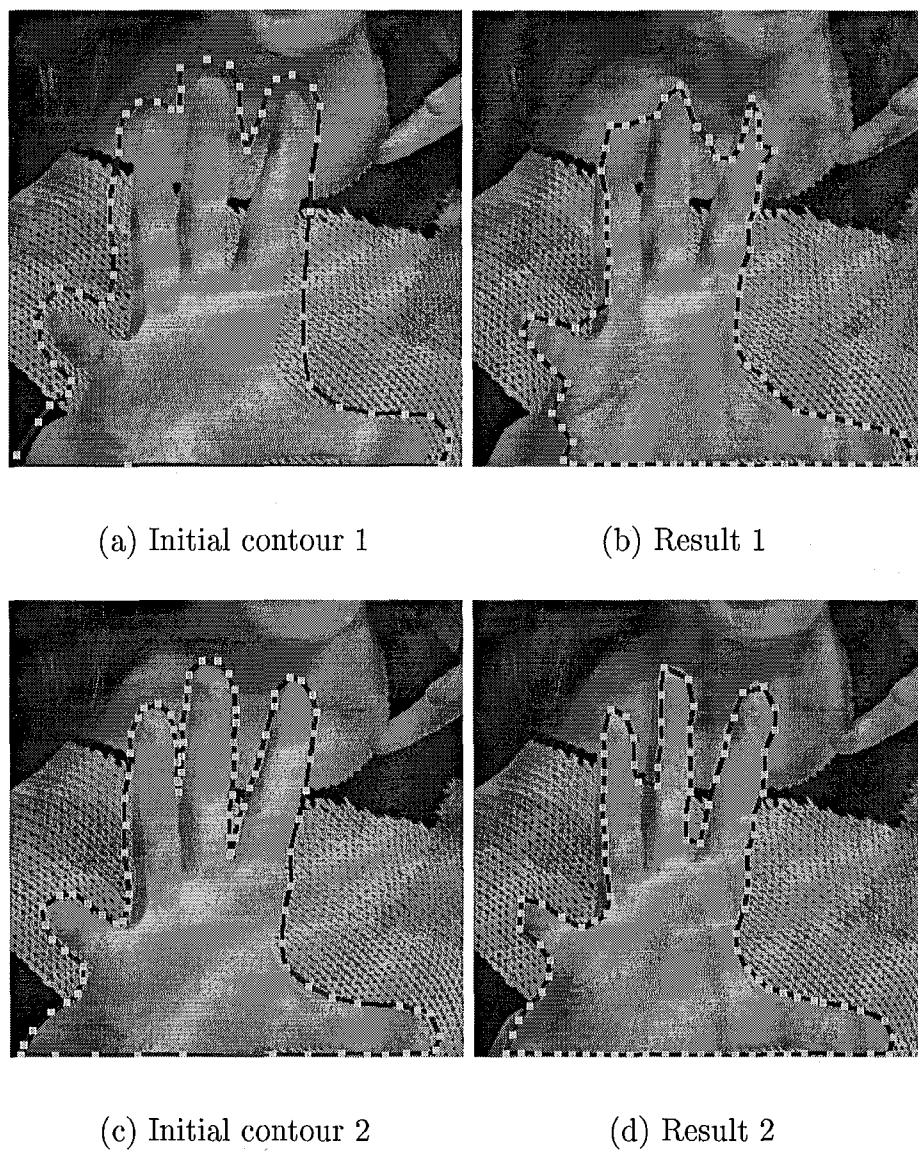


図 3.17 Snakes による処理結果

を用いて境界線上のエッジセグメントを選択する手法を提案し、一部がぼやけたり途切れたりしている場合でもさまざまな形状の境界線を検出できることを示した。

提案した手法は、エッジセグメントを検出することと、処理領域を移動することによって、境界線の近傍だけを処理するので、確率的弛緩法や Snakes よりも高速である。また、処理領域内に存在するエッジセグメントの数を制約条件として、最適なエッジセグメントの組合せを求めることによって境界線を検出するので背景の影響を受けにくい。

第 4 章

Hopfield 型ニューラルネットワークを用いたカラー画像における動的領域抽出

3 章で提案した手法は、処理領域の移動に基づいているために、図 3.15(b) に示した結果のように、閉じた境界線が得られず、物体領域を抽出できない場合があった。また、画像全体を処理領域にすると、探索空間が膨大になってしまう。一方、3 章で提案した手法において、図 3.1 に示した間隔 F とエッジセグメントの長さを最小、すなわち 1 画素に設定すれば、エッジセグメントの選択は、境界画素の選択になる。このとき、探索空間は、画像サイズを S とすれば、 2^S になってしまう。そこで、本章では、任意の境界画素を初期状態として与えて、その近傍において境界画素の選択を繰り返すことで物体領域を抽出する手法を提案する。境界画素の選択においては、3 章で提案した手法と同様に、Hopfield 型ニューラルネットワークを用いる。実験では、Hopfield 型ニューラルネットワークのエネルギー関数値極小化原理を用いて、動的に物体領域を抽出できることを示す。

4.1 Hopfield 型ニューラルネットワークを用いた動的領域抽出

4.1.1 動的領域抽出アルゴリズム

本手法は、最初に、図 4.1(a) に示すように抽出する物体領域内の一画素をシード Q とする。次に Q を含む任意形状領域 A とその他の領域 B とに画像を分け、領域 A に属する画素にラベル 1 を、領域 B に属する画素にラベル 0 を割り当てる。図 4.1(b) から (e) に示すように、領域 A あるいは B 内で、他方の領域に隣接する画素の領域特徴量に対して、Hopfield 型ニューラルネットワークの一つのユニットをそれぞれ割り当て、ユニットの状態更新結果に従ってラベルを変更していくことで領域を変形させる。最後に、収束状態におけるラベルの状態を領域抽出結果とする。図 4.1 の場合、 $\times$ で示す (b) の 6 画素、(c) の 4 画素、そして (d) の 2 画素に割り当てられたラベルを変更していくことで、(e) に示すように求める抽出結果が得られる。但し、本論文では Q をあらかじめ与える。

本手法では、ユニットを割り当てる画素が常に Q を囲んでいるということを前提とする。従って、動的モデルにおける制御点あるいは格子点に相当する境界画素の数は可変であり、初期領域内に物体領域が二つ以上存在していても、一つの物体領域、すなわちシードを含む領域を分離抽出することができる。Hopfield 型ニューラルネットワークの目的関数は、境界線に沿った領域特徴量の連続性、領域の隣接する画素におけるエッジ強度、そして、領域 A においてユニットを割り当てる画素の近傍領域とシード近傍領域との領域特徴量の適合性に基づき定義する。但し、本論文における近傍領域は、注目画素を中心とする 5×5 画素の領域とし、領域特徴量として RGB 階調値を用いる。

Hopfield 型ニューラルネットワークを用いると、局所解が求まるが、大局解が求まる保証はない。更に、原画像のサイズのままで領域を変形させたのでは、収束するまでに膨大な計算時間を要する。そこで本手法では、局所解を避けて、できるだけ大局解 (global minimum) に対応する領域抽出結果を得るとともに、処理を高速に行うために、図 4.2 に示すようなピラミッド画像を用いる。処理する画像が、図 4.2 の P_0 に示すような $X \cdot 2^n \times Y \cdot 2^n$ 画素のカラー画像 $P_0(x, y, c) (x \in \{0, 1, \dots, X \cdot 2^n - 1\}, y \in \{0, 1, \dots, Y \cdot 2^n - 1\}, c \in \{R, G, B\}, n: \text{正定数})$ であるとき、 P_0 から式 (4.1) に示す規則に従って作る、 $X \cdot 2^{n-i} \times Y \cdot 2^{n-i} (0 \leq i \leq n)$ 画素の画像 P_i を、 i 次のピラミッ

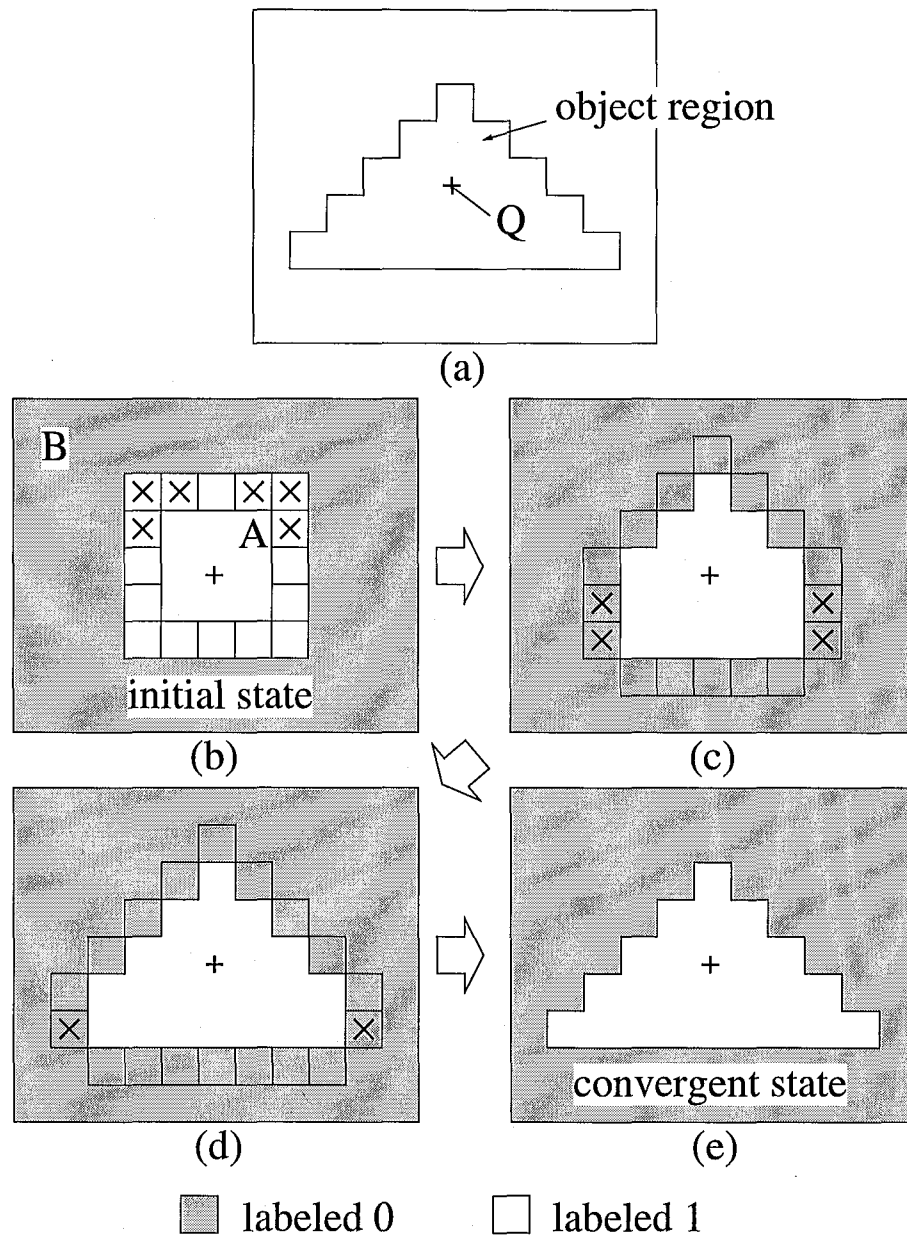


図 4.1 ラベルの変更と領域の変形過程の例

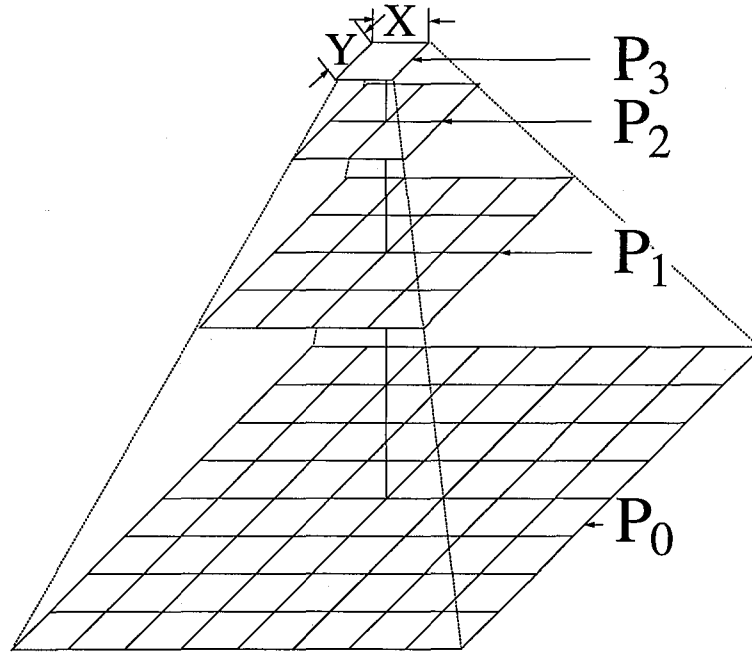


図 4.2 ピラミッド画像

ド画像とする.

$$P_i(x, y, c) = \frac{\sum_{l,m=0}^{2^i-1} P_0(2^i x + l, 2^i y + m, c)}{2^{2i}} \quad (4.1)$$

動的領域抽出アルゴリズムは、次の (1) から (10) に示す処理ステップからなる.

- (1) P_0 中で抽出しようとする領域にある一画素 Q_0 と、用いるピラミッド画像の最大次数 s ($0 \leq s \leq n$) を与える. そして、次数 i の初期値を s にする.
- (2) 図 4.3 の P_2 および図 4.4 に示すように、 Q_i を含む任意形状の領域 A とその他の領域 B とに P_i を分け、 A に属する画素 $P_i(x, y)$ にラベル $L_i(x, y) = 1$ を、その他の画素にラベル $L_i(x, y) = 0$ を割り当てる.
- (3) 領域 A あるいは B 内で、他方の領域に隣接する画素の c 成分に対して、図 4.4 に示すように、0 または 1 を出力するニューラルネットワークの一つのユニットをそれぞれ割り当てる. 図 4.4 の $U_{R,j,c}$ ($R \in \{A, B\}, j \in \{0, 1, \dots, N_R - 1\}$, N_R : ユニット総数) は、領域 R を選んだ場合に c 成分に対して割り当てるユニットを表している.

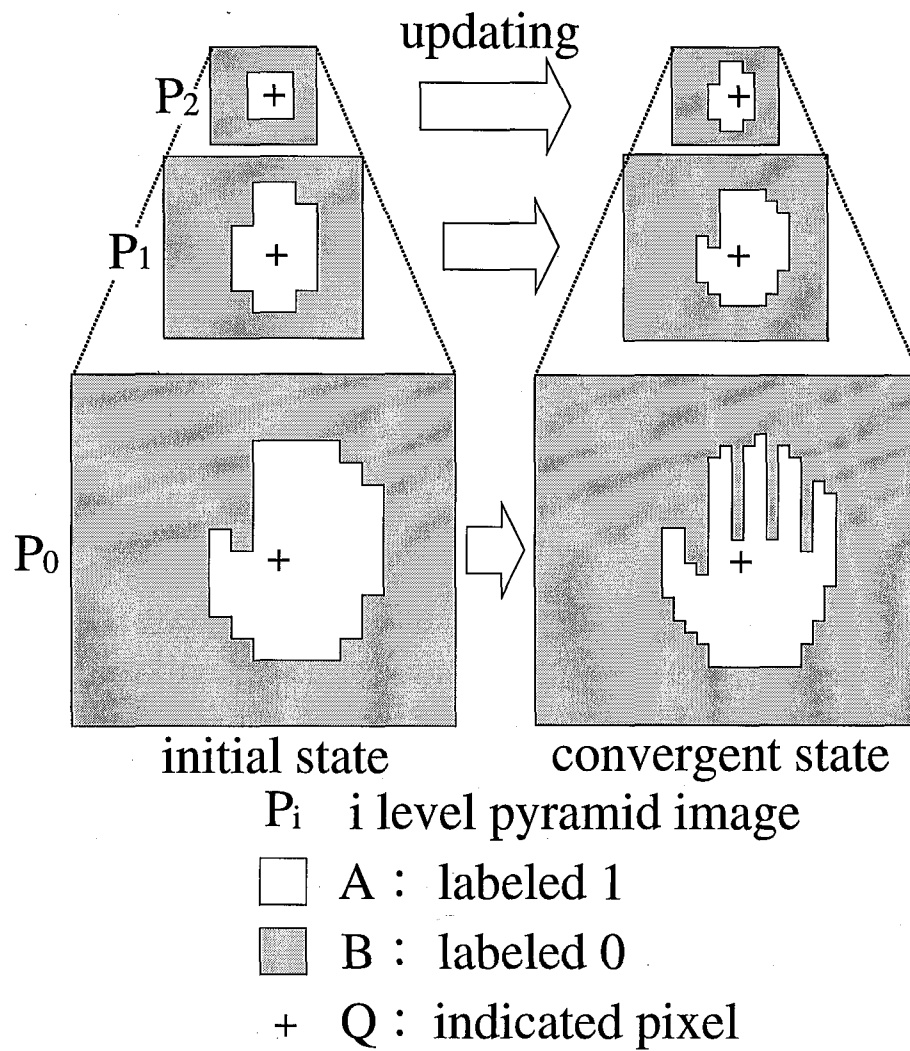


図 4.3 動的領域抽出概念図

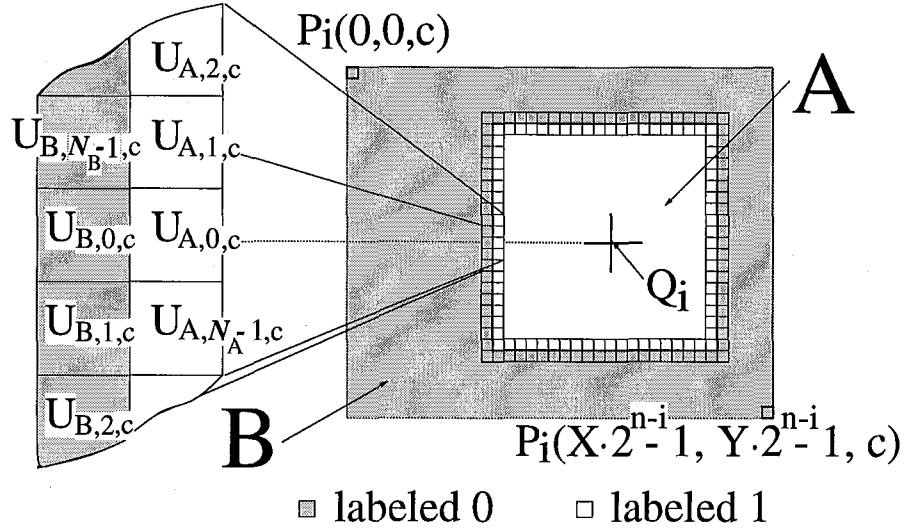


図 4.4 A と B の隣接画素とユニット

- (4) ランダムに選んだ第 j ユニットの状態更新の結果, RGB のいずれか一つの成分に対して割り当てたユニットの出力値 $U_{R,j,c}$ が 0 の場合に, ラベルを 0 ならば 1 に, あるいは 1 ならば 0 に変更する. また, 領域 A に膨張する傾向をもたせるために, 出力値 $U_{B,j,c}$ が RGB 成分すべてについて 1 の場合を除き, 8 近傍画素に対しても同じラベルを割り当てる.
- (5) 処理ステップ (3) と (4) を繰り返す. 本論文では, 予備実験の結果に基づき, 繰り返しステップ数を 1000 にする.
- (6) $i = i - 1$ にする.
- (7) 式 (4.2) に示す規則に従って, P_i のすべての画素にラベル 0 またはラベル 1 を割り当てる. 但し, $l, m \in \{0, 1\}$ とする.

$$L_i(2x + l, 2y + m) = L_{i+1}(x, y) \quad (4.2)$$

- (8) 処理ステップ (3) から (7) を $s-1$ 回繰り返した後に, 処理ステップ (3) と (4) を, 4.1.2 で述べるニューラルネットワークのエネルギー関数値が収束す

るまで行う。但し、0 次のピラミッド画像 P_0 ，すなわち原画像のサイズにおける概略の抽出が既に得られているので，領域 A に膨張する傾向をもたせる必要がない。そこで，8 近傍画素に割り当てたラベルについては，そのままとする。そして，収束状態におけるラベルの状態を領域抽出結果にする。

4.1.2 Hopfield 型ニューラルネットワークの目的関数

Hopfield 型ニューラルネットワークにおいては， n 回目の更新における第 j ユニットの出力値 $U_j(n)$ を式 (4.3) に従って決定すれば，式 (4.4) に示すエネルギー関数 E_{NN} の値は決して増加しない（付録 A 参照）。

$$U_j(n) = \begin{cases} 1 & \text{if } \sum_{k=0}^{N-1} w_{jk} U_k(n-1) - h_j > 0 \\ 0 & \text{otherwise} \end{cases} \quad (4.3)$$

$$E_{NN} = -\frac{1}{2} \sum_{j=0}^{N-1} \sum_{k=0}^{N-1} w_{jk} U_j U_k + \sum_{j=0}^{N-1} h_j U_j \quad (4.4)$$

w_{jk} : 第 k ユニットから第 j ユニットへの結合荷重。

h_j : 第 j ユニットのしきい値。

本論文では， i 次のピラミッド画像において，第 j ユニットを割り当てる画素 $P_{i,\text{unit}}(x_j, y_j)$ の近傍領域で， $P_{i,\text{unit}}(x_j, y_j)$ と同じラベルを割り当てられている画素を $P_{i,\text{same}}(x_j + u, y_j + v)$ ($u, v \in \{-2, -1, 0, 1, 2\}$)，異なるラベルを割り当てられている画素を $P_{i,\text{diff}}(x_j + u, y_j + v)$ ， $P_{i,\text{same}}(x_j + u, y_j + v)$ の平均 RGB 階調値を $\overline{P_{i,\text{same}}}(x_j, y_j, c)$ ， $P_{i,\text{diff}}(x_j + u, y_j + v)$ の平均 RGB 階調値を $\overline{P_{i,\text{diff}}}(x_j, y_j, c)$ とする。そして，最適な領域分割において各領域が RGB それぞれについて満たすべき条件を，仮定条件 1 から仮定条件 3 に示すように仮定する。

仮定条件 1: $\overline{P_{i,\text{same}}}$ は境界線に沿って連続 (continuous) である。

仮定条件 2: $P_{i,\text{unit}}$ におけるエッジ (edge) 強度は，その他の画素よりも大きい。

仮定条件 3: 領域 A において， $\overline{P_{i,\text{same}}}$ とシード近傍領域の平均 RGB 階調値 $\overline{Q_0}$ が適合 (fit) している。

そして， $U_{R,j} = 0$ ($R \in \{A, B\}$) であるとき， $P_{i,\text{unit}}(x_j, y_j)$ が領域 R に属さない，すなわち，ユニットを割り当てる画素として存在しないことを， $U_{R,j} = 1$ であるとき，存在することを仮定すれば，式 (4.5) から式 (4.7) で定義する E_{cont} ， E_{edge} ，そ

して E_{fit} の値が小さくなる程、仮定条件 1 から仮定条件 3 にそれぞれ近付いたといえる。但し、 E_{fit} は、領域 A について構成するニューラルネットワークの場合だけ定義する。

境界線に沿ってある一定距離互いに離れた二つの $P_{i,unit}$ の $\overline{P_{i,same}}$ がすべて等しい場合に、値が 0 になるような E_{cont} を式 (4.5) に示すように定義する。本論文では、予備実験の結果に基づき、この距離を 4 画素に設定する。

$$E_{cont} = \sum_{j=0}^{N-1} \sum_{k=0}^{N-1} \delta_{j \pm 4, k} \left| \overline{P_{i,same}}(x_j, y_j, c) - \overline{P_{i,same}}(x_k, y_k, c) \right| U_j U_k \quad (4.5)$$

$\delta_{j,k}$: クロネッカーのデルタ。

$\left| \overline{P_{i,same}} - \overline{P_{i,diff}} \right|$ を、 $P_{i,unit}$ におけるエッジ強度とする。そして、状態更新の結果 U_j が 0 の場合に、4.1.1 のアルゴリズムに従ってラベルを変更したときのエッジ強度の変化 $\Delta \left| \overline{P_{i,same}} - \overline{P_{i,diff}} \right|$ を用いて、 E_{edge} を式 (4.6) に示すように定義する。

$$E_{edge} = \sum_{j=0}^{N-1} \Delta \left| \overline{P_{i,same}}(x_j, y_j, c) - \overline{P_{i,diff}}(x_j, y_j, c) \right| U_j \quad (4.6)$$

$$E_{fit} = \sum_{j=0}^{N-1} \left| \overline{P_{i,same}}(x_j, y_j, c) - \overline{Q_0}(c) \right| U_j \quad (4.7)$$

ニューラルネットワークの目的関数 E_{obj} を、式 (4.8) に示すように定義する。

$$E_{obj} = \alpha E_{cont} + \beta E_{edge} + \gamma E_{fit} \quad (4.8)$$

α, β, γ : 正の実数。

E_{obj} の最小値を与えるラベルの状態を、最適な領域抽出結果であると仮定する。 E_{obj} が E_{NN} と恒等的に等しくなるように、係数比較によって w_{jk} および h_j を決定してニューラルネットワークを構成すれば、ランダムに選んだユニットの状態を式 (4.3) に従って更新していくことによって、 E_{obj} が極小値になる U_j を求めることができる。 w_{jk} を式 (4.9) に、 h_j を式 (4.10) に示す。

$$w_{jk} = -2\alpha\delta_{j \pm 4, k} \left| \overline{P_{i,same}}(x_j, y_j, c) - \overline{P_{i,same}}(x_k, y_k, c) \right| \quad (4.9)$$

$$h_j = \beta\Delta \left| \overline{P_{i,same}}(x_j, y_j, c) - \overline{P_{i,diff}}(x_j, y_j, c) \right| + \gamma \left| \overline{P_{i,same}}(x_j, y_j, c) - \overline{Q_0}(c) \right| \quad (4.10)$$

4.1.3 分裂領域への対処方法

本手法では、図 4.1 に示したようにラベルを変更していくので、同じラベルを割り当てられている領域が二つ以上に分裂してしまう場合がある。そこで、Q を含む

領域 A と、それを囲む領域 B とに分割するために、分裂した他方の領域を無視するようにする。領域 B に対してユニットを割り当ての場合について、分裂領域の一例と対処方法の概念図を図 4.5 に示す。まず、図 4.5(a) に示す状態において、Q から左方向に調べていき、最初にラベルが変化する■で示す画素をユニット割り当ての開始画素 S としてユニットを割り当てる。次に、E で示す画素に割り当てられたラベルを 0 から 1 に変更すると図 4.5(b) に示すように領域 B は分裂する。再び、図 4.5(c) に示すように、Q から左方向に調べていき、最初にラベルが変化する■で示す画素を S とすると、領域 B から分裂した領域 F に対してユニットを割り当てることになり、結果的にシード Q を含む領域 A と、それを囲む領域 B とを動的に分割することができなくなってしまう。そこで、ユニットを割り当てた画素が Q を囲んでいない場合には、図 4.5(d) に示すように強制的にラベルを変更し、再び S を決定する。領域 A に対してユニットを割り当ての場合についても同様に対処することによって、常に領域 A とそれを囲む領域 B だけを動的に分割することができる。

4.2 実験結果

$s = 2, \alpha = 0.3, \beta = 3.0, \gamma = 0.3$ として、図 4.6 に示す 320×256 画素 256 階調のカラー標準画像において、パパイアとにんにくに対応する領域を、図 4.7 に示す 320×256 画素 256 階調のカラー標準画像において、中央と右側に位置する人物の顔に対応する領域を動的に抽出した様子を、それぞれ図 4.8 から図 4.11 に示す。図 4.3 に示したように、+ は Q の位置を、正方形で囲んだ領域は、あらかじめ与えた領域 A の初期状態を表している。図 (a), (b), (c) において白く塗りつぶした画素は、ラベル 0 を割り当てた画素であり、ユニットを割り当てた画素を黒で表示している。物体領域内で島状に存在するラベル 0 を割り当てた領域と、背景領域内で島状に存在するラベル 1 を割り当てた領域は、ラベルの変更の結果分裂した領域である。図 (d) は、ユニットを割り当てた画素を白で表示した結果である。

次に、図 4.11 の分割において、ピラミッド画像を用いる場合と用いない場合の、エネルギー関数値収束の様子を表すグラフを図 4.12 に示す。本手法では画像を二つの領域 A および B とに分け、それぞれの領域の各領域特徴量、すなわち本論文では RGB 各成分について Hopfield 型ニューラルネットワークを構成する。エネルギー関数値のグラフは、これら 2×3 個のニューラルネットワークのエネルギー関数値の和を表している。但し、エネルギー関数値はユニット総数に比例するので、式 (4.4)

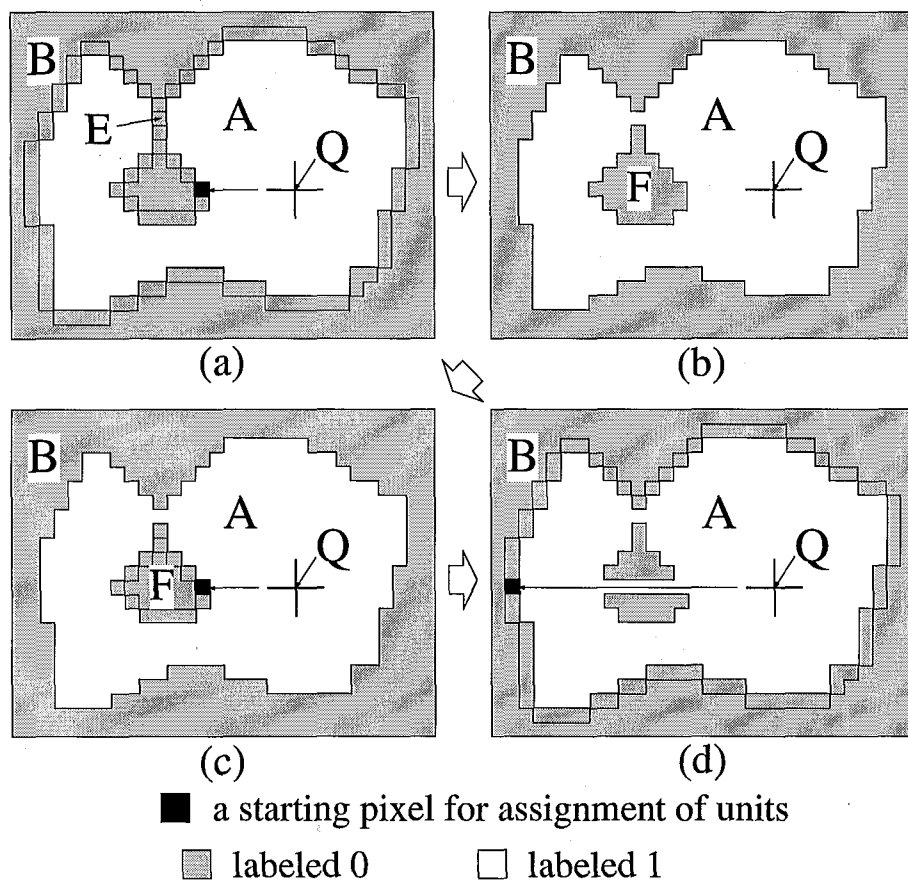


図 4.5 分裂領域と対処方法

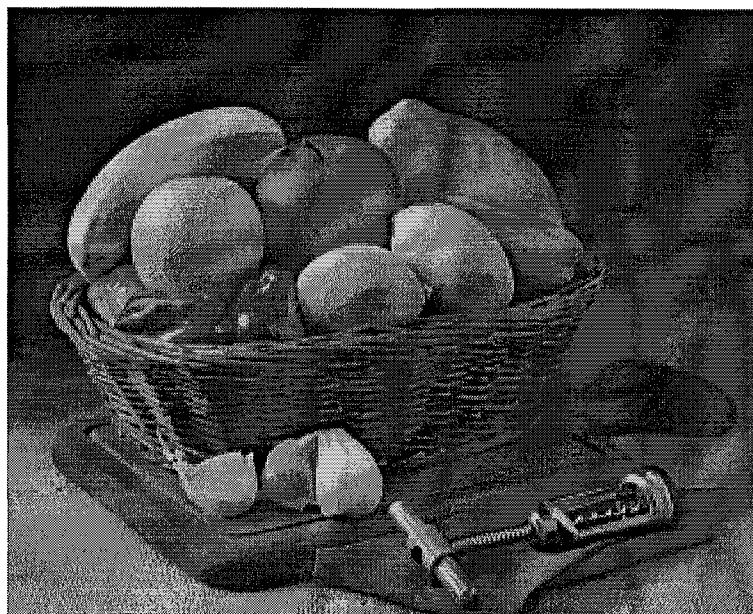


図 4.6 SCID“果物籠”



図 4.7 SCID“ミュージシャン”

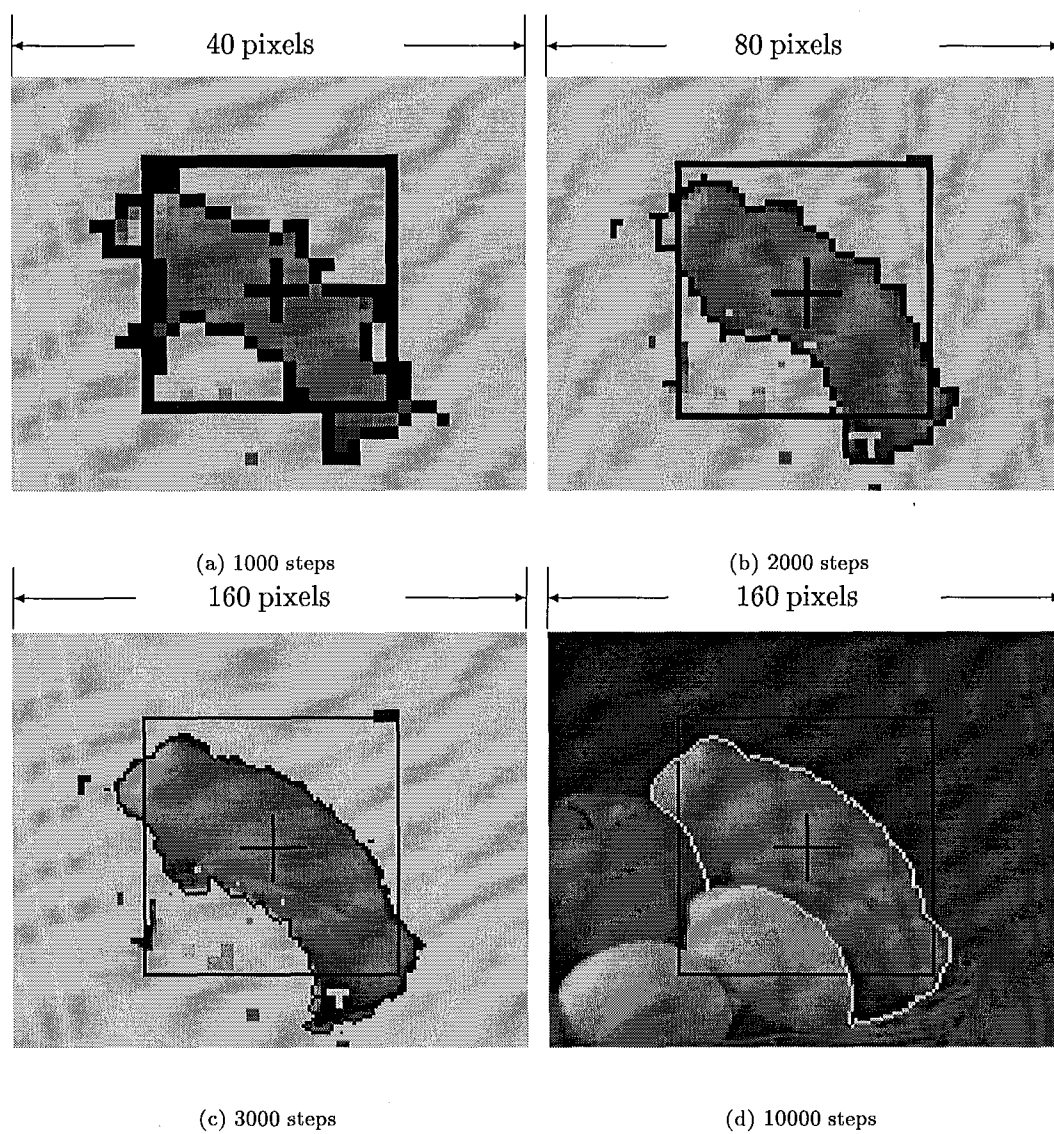


図 4.8 動的領域抽出の様子 (パパイア)

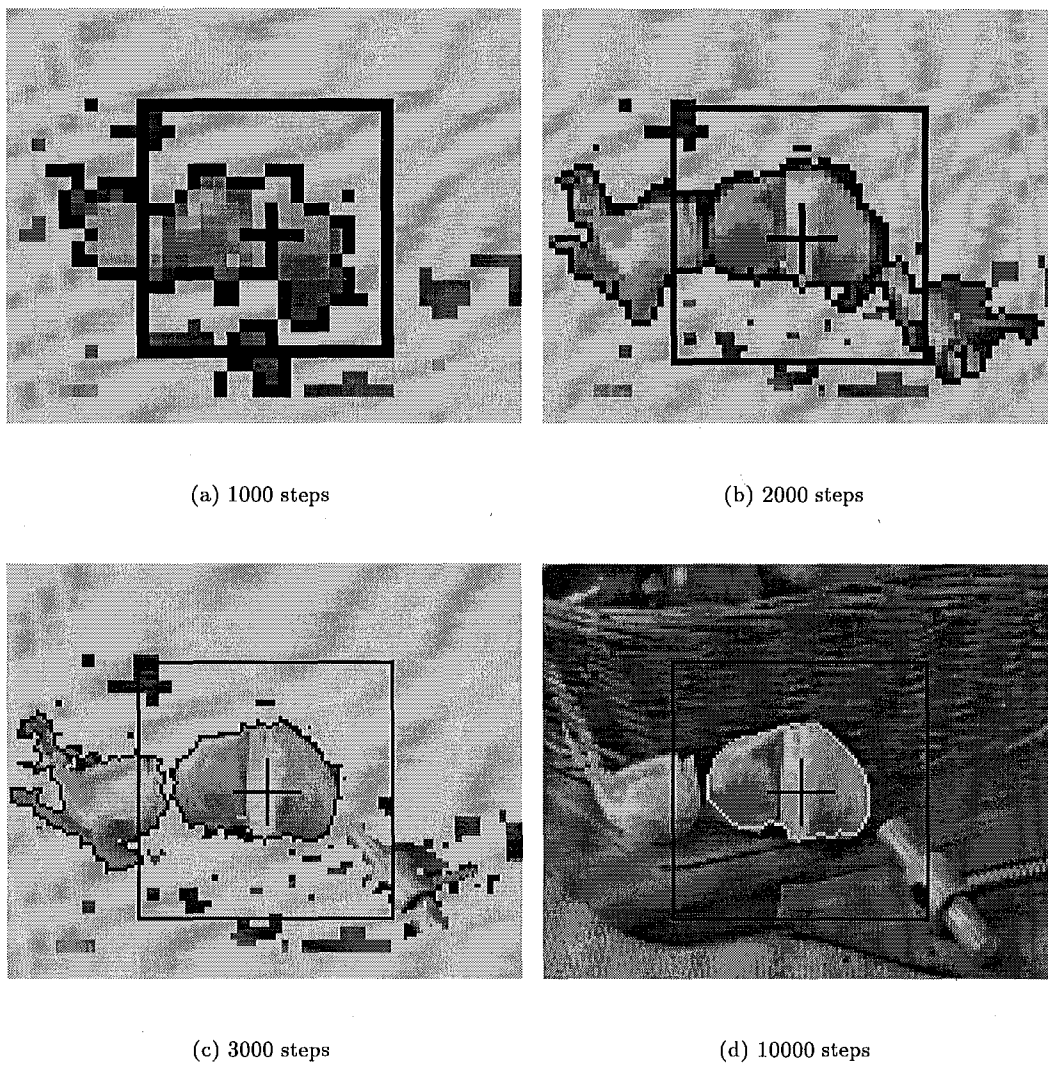


図 4.9 動的領域抽出の様子（にんにく）

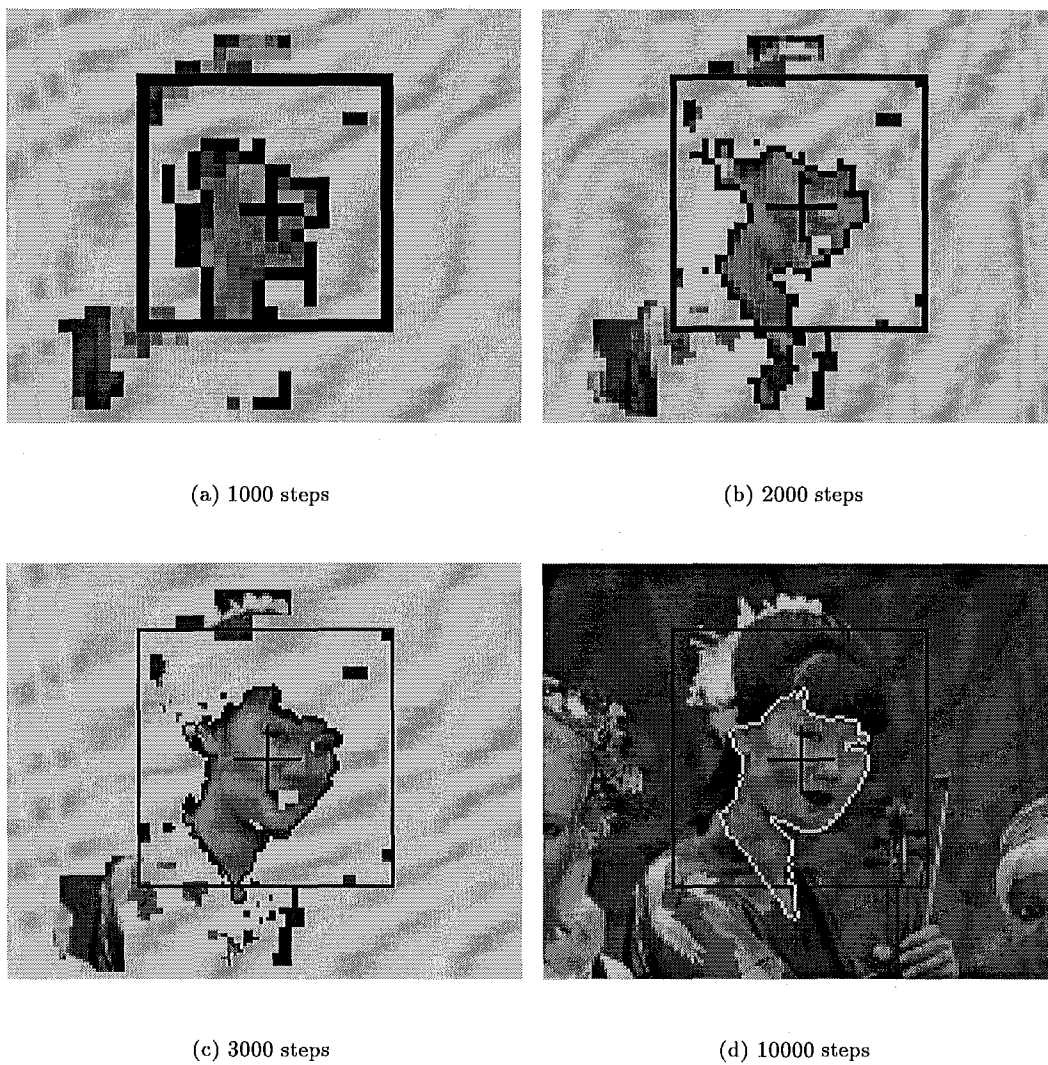


図 4.10 動的領域抽出の様子（顔 1）

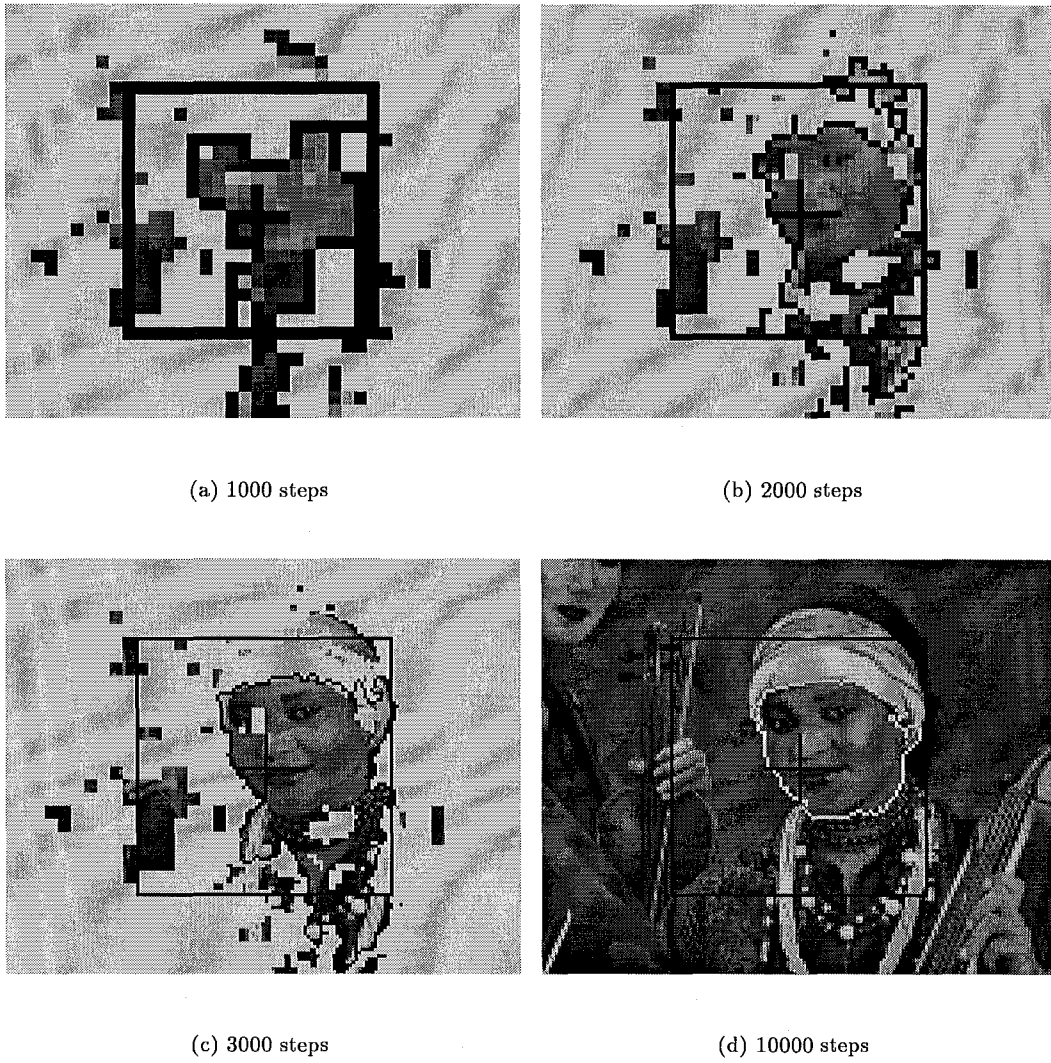


図 4.11 動的領域抽出の様子（顔 2）

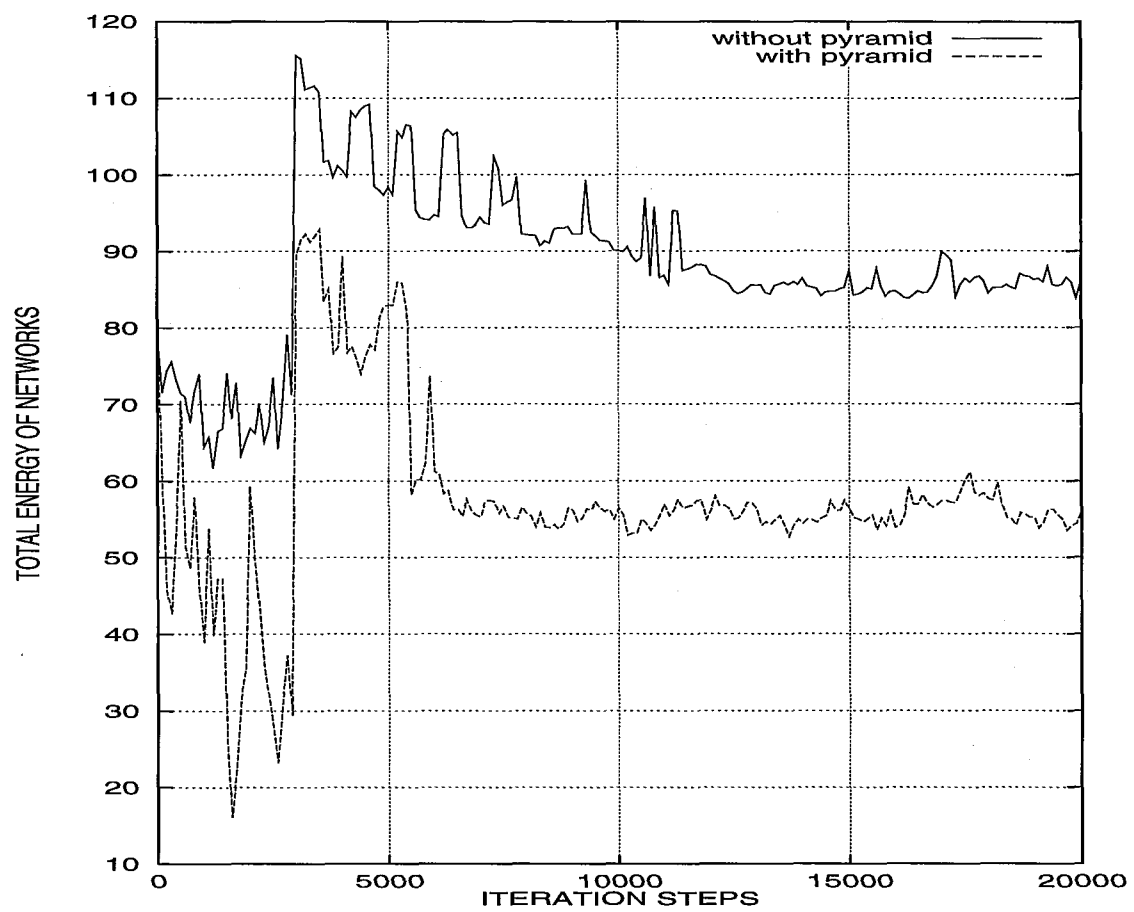


図 4.12 エネルギー関数収束の様子

の値をユニット総数で割った値を用いている。例えば、領域 A の R 成分について構成したニューラルネットワークのエネルギー関数値が減少するようにラベルを変更する場合に、その他のニューラルネットワークのエネルギー関数値が必ずしも減少するわけではない。すなわち、領域 A と B、そして RGB 各成分は競合する。また、領域が分裂あるいは統合するとエネルギー関数値が大きく変化する場合がある。従って、グラフが振動したものと考えられる。また、ステップ数 3000 において大きな値に変移したのは、ラベルの変更範囲を 3×3 画素から 1 画素に変更した結果、 E_{edge} が大幅に増加したからである。グラフから、ピラミッド画像を用いることによって、局所解を避けることができるとともに、収束が速くなっていることが分かる。それぞれのピラミッド画像において収束させることが理想であるが、計算コストを考えると適当なステップ数で概略の大局的抽出が得られれば良いと考え、ピラミッドレベルの変更は一定の繰り返しステップ数で行った。

最後に、図 4.13(a) に示す、カラー写真をイメージスキャナで取り込んだ、 156×156 画素 256 階調の低画質のカラー画像において、顔に対応する領域を動的に抽出した様子を、それぞれ図 4.13(b) から (d) に示す。

パラメータ変更の効果について、簡単に述べる。 α を大きくすることによって、境界線に沿った領域特徴量の連続性が高くなるが、偽の輪郭を生じる場合がある。 β を大きくすることによって、エッジ強度が大きい部分における分割に収束するようになるが、局所解に陥る場合がある。 γ を大きくすることによって、物体領域内部の領域特徴量の均一性が高くなるが、物体領域の細分化を生じる場合がある。 γ を変更した場合の例として、図 4.6 に示した画像において、りんごに対応する領域を抽出した結果を図 4.14 に示す。 RGB 階調値がほぼ等しいりんごと赤ピーマンに対応する領域が隣接しているが、 γ を大きくすることで、りんごに対応する領域だけを抽出することができる。パラメータを変更することで結果が変化する場合があるが、その値をより小さくしたり大きくしたりすることで結果が変化する場合は少ない。従って本手法は、パラメータの変更に関してセンシティブではないと考えられる。

4.3 Active Net との対比

Snakes では、制御点がエッジに向かって移動するのに対し、Active Net では、格子点が階調値の高い領域あるいは階調値の低い領域に向かって移動する。具体的には、2.3.2 で説明した E_{image} を、最外郭格子点（網の最も外側の 1 層）では抽出した

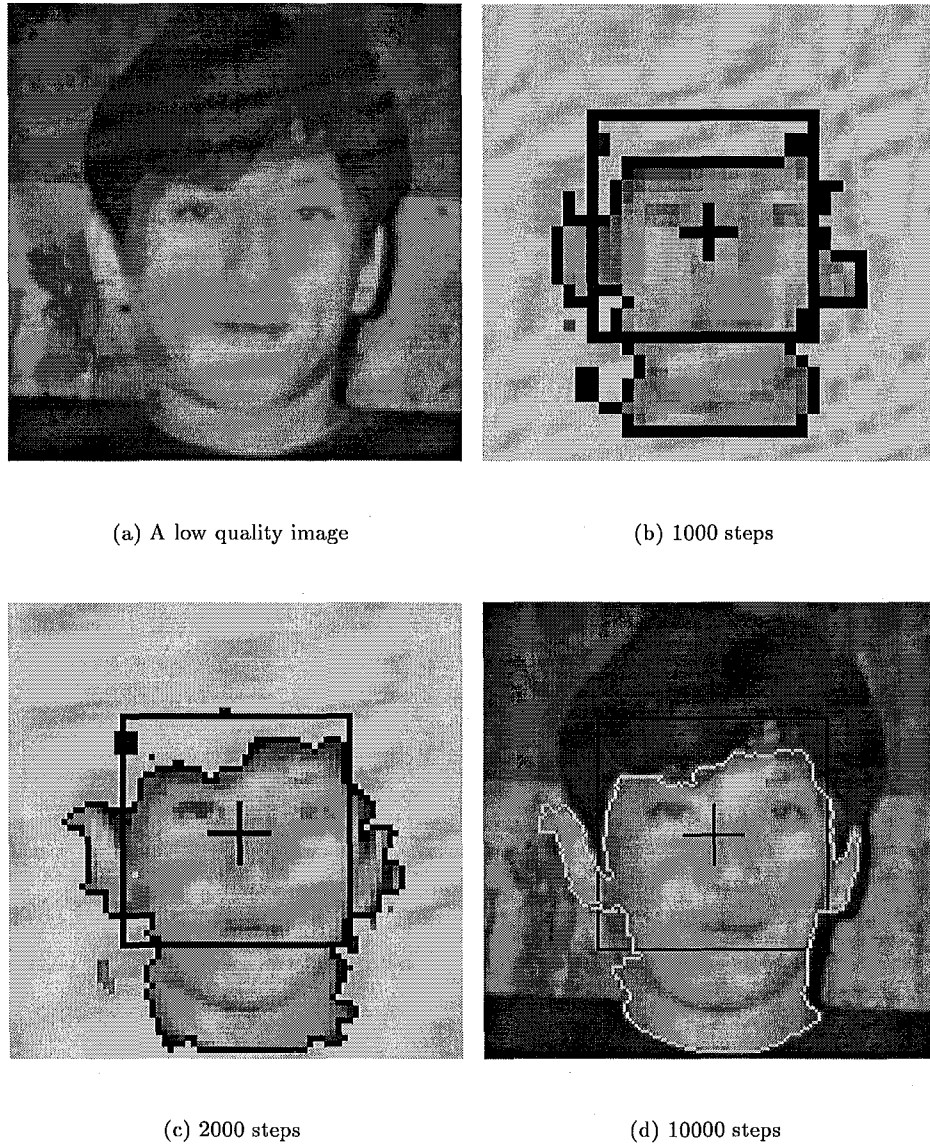
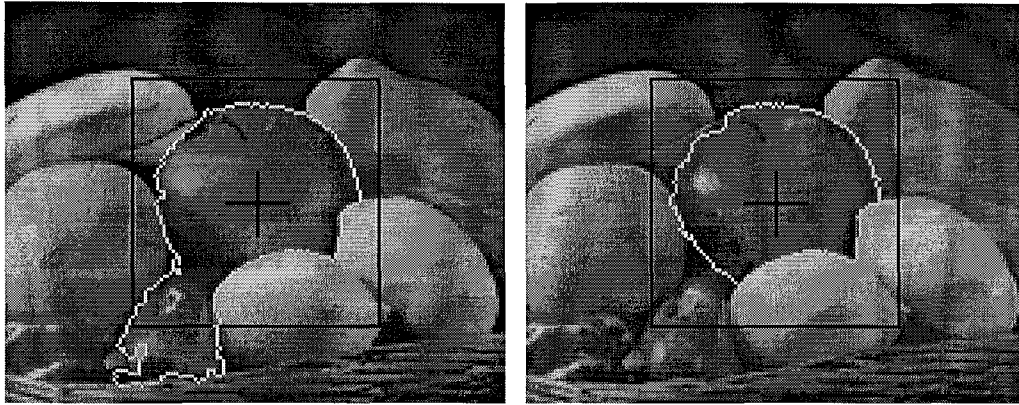


図 4.13 動的領域抽出の様子（低画質画像）

(a) $\gamma = 0.3$ (b) $\gamma = 0.4$ 図 4.14 パラメータ γ の効果

い領域以外の部分に適合し ($\omega < 0$)，内部の格子点では対象物に適合する ($\omega > 0$) ように設定する [10]．これによって，収束状態において，最外郭格子点と 1 層内部の格子点とで対象物の輪郭をはさむように網は変形する．従って，Snakes がエッジに基づいて動作するのに対して，Active Net は領域に基づいて動作するといえる．本章で提案した手法は，領域 A と領域 B の競合に基づいて動作するため，Active Net との関連性が大きい．そこで，提案手法と Active Net を対比した結果について述べる．

RGB 成分すべてを用いるように，2.3.2 で検証した Active Net [10] を拡張した．拡張の方法はいくつか考えられるが，本論文では，Active Net の中心座標における画素値と格子点 $v(= (x, y))$ における画素値間での，RGB 空間におけるユークリッド距離を 256 レベルに正規化した値 $L(x, y)$ を，濃淡レベルの代わりに用いることで，本質的な問題点を明らかにする．

複数の領域を分離して抽出する能力は Active Net には備わっていないので，実用的には何らかの前処理によって，初期網のおおよその位置を前もって決める必要がある [10]．しかし，自然画像において，このような前処理を実現することは非常に困難である．逆に提案手法は，Net を用いていないので，複数の領域を分離することが可能であり，実用性が高いと考えられる．図 4.15 に示す 120×120 画素のカラー画像からりんごに対応する領域の抽出を試みた．格子点数を 10×35 にした場合の Net の初期網と $L(x, y)$ に変換した結果を図 4.16 に示す．格子点数を 10×35 に

して得られた結果を図 4.17 に、 8×35 にして得られた結果を図 4.18 に示す。但し、格子点の間隔は等しくした。従って、図 4.17 の初期網のサイズは、図 4.18 のそれよりも大きい。図 4.17 では、りんごとレモンを分離できていないことが分かる。レモンの陰に対応する部分における $L(x, y)$ が低い値であるために、初期網の内部に対象領域が複数存在することが原因である。この結果は、文献 [10] において述べられているように、Active Net では、複数の領域を分離抽出できないことを示している。一方、図 4.18 において、良い結果が得られたのは、初期網のサイズを小さくした結果、初期網の内部に対象領域が一つだけ存在するようになったからである。レモンに対応する領域の抽出結果を図 4.19 に示す。

また、文献 [10] で用いていた反復法は、SOR 法 (Successive over-relaxation method) である。SOR 法の難点は、ある特殊な問題を除いて、最適な加速係数の値の決定が難しく、係数の微小変動によってその収束性が大きく変化することである [31]。Active Net の動作がパラメータ調整に関して敏感ではないと述べられていたが、実際には非常に敏感であることを、実験で確認した。

一方、本手法は、ラベルの変更に基づく手法であるため、Active Net の網構造に相当するような拘束がない。従って、初期領域内に物体領域が二つ以上存在していても、シードを含む一つの物体領域を抽出することができる。また、ラベルに基づくエッジ強度算出や、三つの仮定条件に基づく領域の競合によって、Active Net よりも高精度な抽出が可能である。

4.4 視覚系との対応に関する考察

式 (4.6) において、ラベルの状態に基づいてエッジ強度を計算することの意味について考察した結果について述べる。

図 4.20 に示す領域 A と領域 B の隣接する画素 P_A および P_B におけるエッジ強度を、微分フィルタ (x, y 方向の差分の 2 乗和の平方根) を用いて求めると異なる値になる。一方、提案手法に従えば同じ値になる。いずれも境界に位置しているため視覚的には同じであるはずである。従って、提案手法におけるエッジ強度を用いることで、微分フィルタを用いる場合よりも、視覚的に自然な境界が得られると考えられる。

次に、式 (4.6) において、エッジ強度そのものではなく、エッジ強度の変化 Δ を用いることの意味について考察した結果について述べる。



図 4.15 “果物籠”

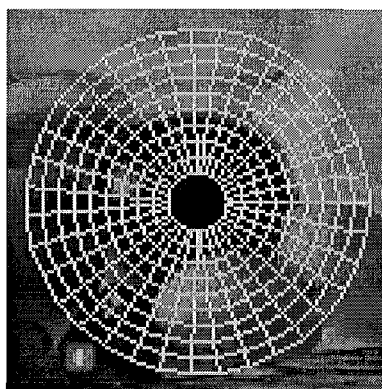


図 4.16 初期網と $L(x, y)$

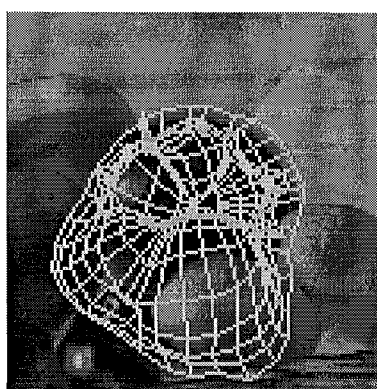


図 4.17 10×35

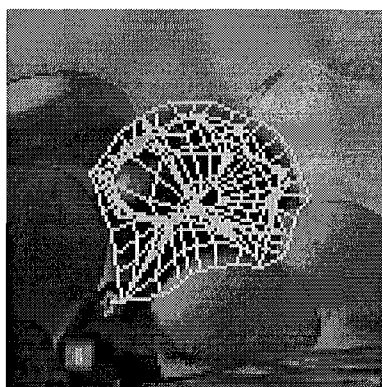


図 4.18 8×35

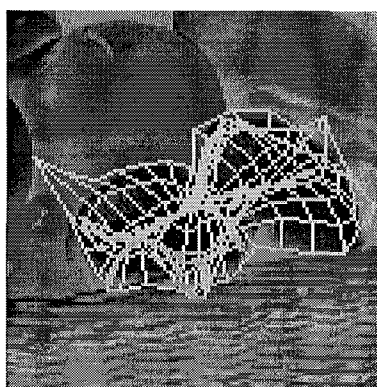


図 4.19 レモン

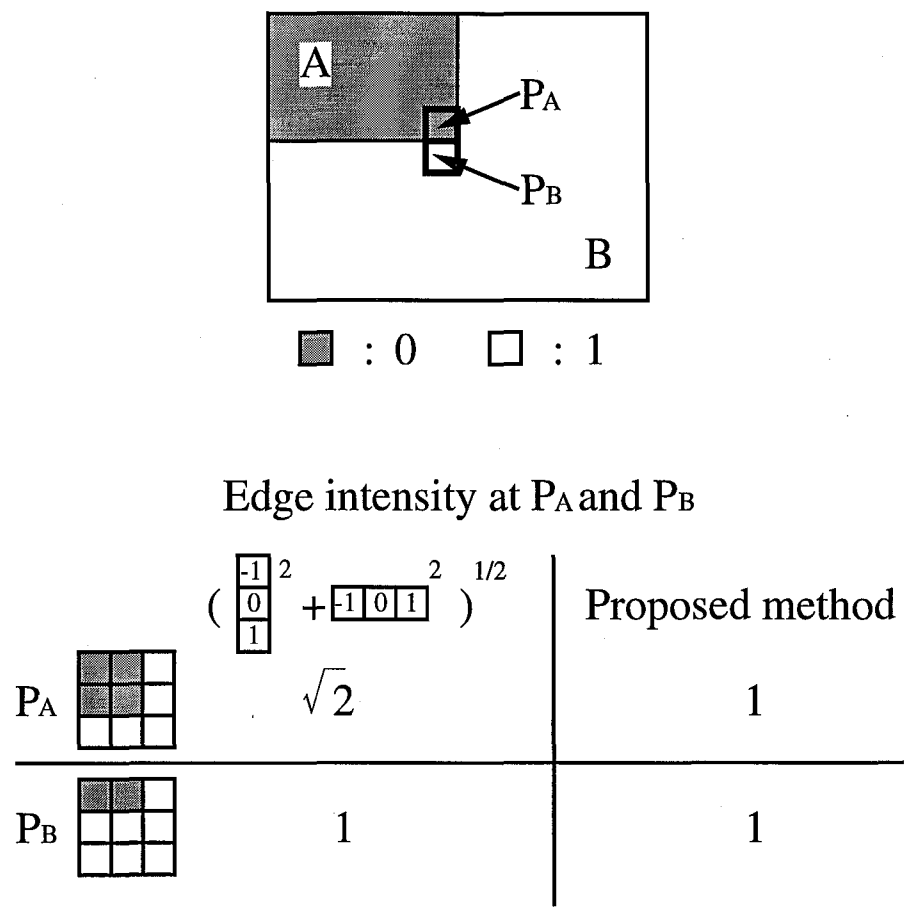


図 4.20 微分フィルタリングとラベルに基づくエッジ強度算出との対比

本論文におけるエッジ強度は、境界線に垂直な方向に関する階調値の1次微分の絶対値に相当し、ラベルを変更した場合のエッジ強度の変化 Δ は、境界から領域内部に向かう、境界線に垂直な方向を正にとった場合の階調値の2次微分（1次微分の絶対値のさらに1次微分）に相当する。従って、図4.21に示すように、領域Aの内部に向かう方向を正にとる場合と領域Bの内部に向かう方向を正にとる場合とは、この2次微分が極小値になる位置が異なる。例えば、前者が極小になる位置は、後者が極大になる位置に一致するため、領域Aあるいは領域Bを選び直す度に、ゼロ交差をはさむ一方の極小から他方の極小への境界の移動が繰り返されることになる。すなわち、エッジ強度そのものを用いる場合のように、領域Aと領域Bの変形によって一つの極大点に境界が近付く場合とは異なり、領域間の競合が生じることになる。その平衡点として、ほぼゼロ交差の位置に境界がおちつくと考えられる。エッジ強度そのものを大きくすることを用いることは、前者に対応するが、2次微分を小さくすることを用いる方が、より良好な結果が得られることを予備実験で確認した。以上の考察は、視覚系におけるゼロ交差検出機構 [32] に対応づけることが可能であると考えられる。

4.5 本章のまとめ

Hopfield 型ニューラルネットワークのエネルギー関数値極小化原理を用いて、ピラミッド画像上で動的に領域を抽出する手法を提案した。本手法は、領域成長法とエッジ検出法を統合した、正則化に基づく手法として位置づけられ、ロバストで高精度な処理が可能である。

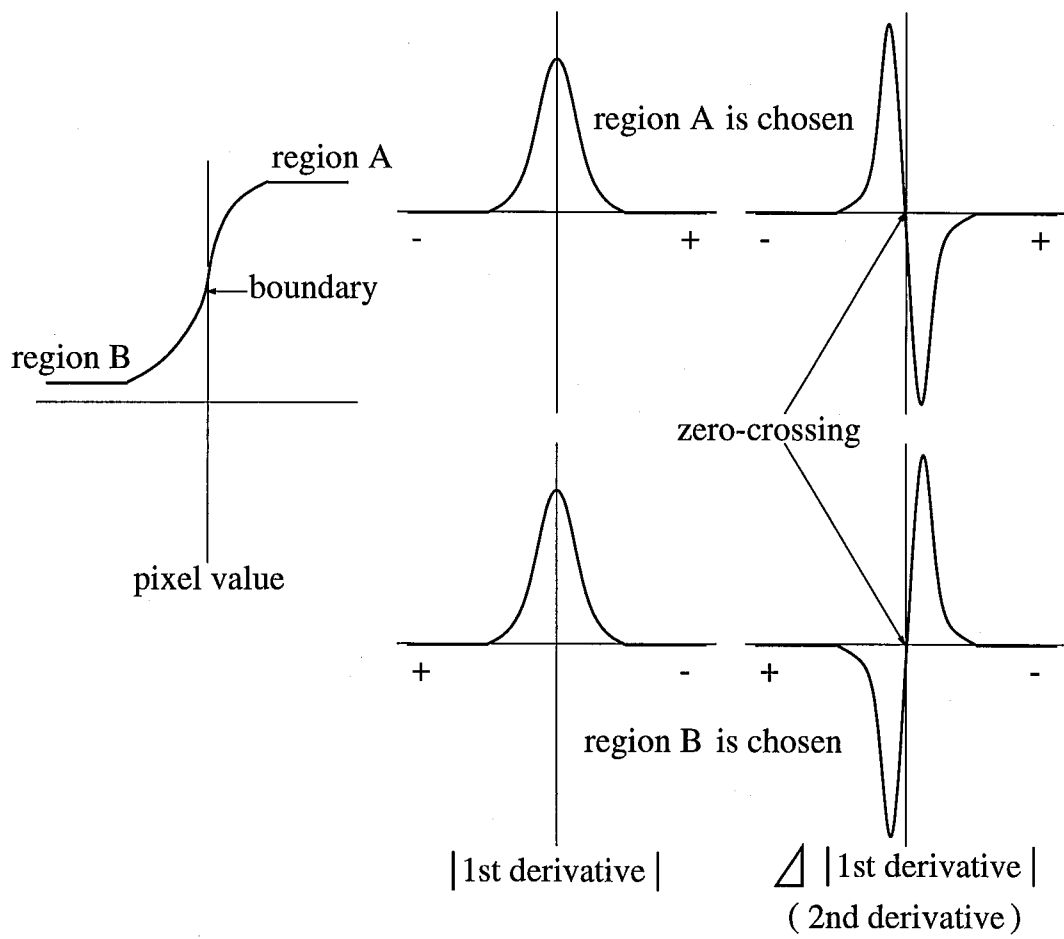


図 4.21 領域の競合とゼロ交差検出

第 5 章

抽出対象に応じた動的領域抽出の改良 および応用

5.1 領域形状に基づく動的領域抽出

4.1 で述べた手法では，陰などの影響のために，良好に物体領域を抽出できない場合があった．一方，局所的な形状に関する記述を用いて表現した物体の輪郭形状と動的輪郭モデルとのマッチングを行なうことで，モデルを物体形状に収束させる手法が最近提案された [33]．文献 [33] の手法は，形状が変化しない場合には有効であるが，形状が変化するような場合に適用することは困難である．そこで本節では，4.1.2 で定義した目的関数に，形状に関する仮定に基づく項を追加して，人物の顔領域を良好に抽出する手法を提案する．

顔領域抽出のための目的関数

本手法では，大局的な顔領域形状に関する一つの仮定として，図 5.1 に示すように横方向の鋭い凹部が存在しないことを仮定する．式 (5.1) に示す E_{face} を最小化して，この仮定に適合するような抽出を行なうことで，陰などの影響を抑制する．

$$E_{face} = -f(i)\{S_i + T_i\}U_i \quad (5.1)$$

$$f(i) = \{(L_i = 1) \cap (S_i, T_i > 3)\} \cup \{(L_i = 0) \cap (S_i, T_i < -3)\} \quad (5.2)$$

$$S_i = (x_{i+5} - x_i)(y_{i+1} - y_i) \quad (5.3)$$

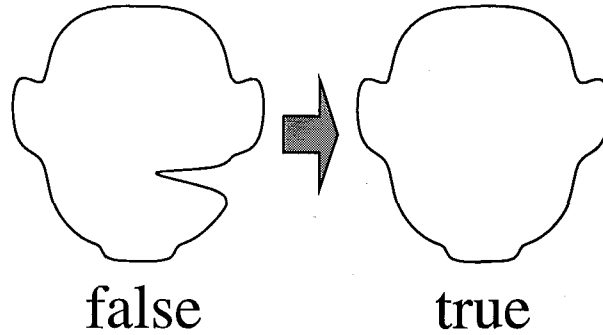


図 5.1 顔形状に関する仮定

$$T_i = (x_{i-5} - x_i)(y_{i+1} - y_i) \quad (5.4)$$

U_i : 第 i ユニットの出力 (0 または 1) .

(x_i, y_i) : 第 i ユニットを割当てた画素の座標.

L_i : (x_i, y_i) におけるラベル (0 または 1) .

4.1.2 で定義した目的関数を E_{simple} として, 本手法における目的関数 E_{obj} を式 (5.5) に示すように定義する.

$$E_{obj} = E_{simple} + \delta E_{face} \quad (5.5)$$

鋭さの異なる三つの凹部に対して, 領域 A にユニットを割り当てた場合 ($L_i = 1$) と領域 B にユニットを割り当てた場合 ($L_i = 0$) について, $-f(i)\{S_i + T_i\}$ の値の例を図 5.2 に示す. 4.1.2 で述べた状態更新規則に従えば, 式 (4.3) におけるしきい値 h_j は $-f(i)\{S_i + T_i\}$ の値に従って増加する. 図 5.2 に示すように, 鋭い凹部において, 領域 A にラベルを割り当てた場合には, $-f(i)\{S_i + T_i\}$ が小さな値になるためラベル変更され難く, 領域 B にラベルを割り当てた場合には, 大きな値になるためラベル変更され易い. 結果的に, 領域 A の鋭い凹部を抑制できる.

実験結果

図 4.7 に示したカラー標準画像において, $\delta = 0.0, 3.0, 6.0, 9.0$ にして, 中央に位置する人物の顔領域を抽出した結果を, それぞれ図 5.3 (a) から (d) に示す. + はあらかじめ与えたシードの位置を, 正方形で囲んだ領域は, あらかじめ与えた領域 A の初期状態を表している. 但し, ユニットの状態更新回数は 15000 回に設定した.

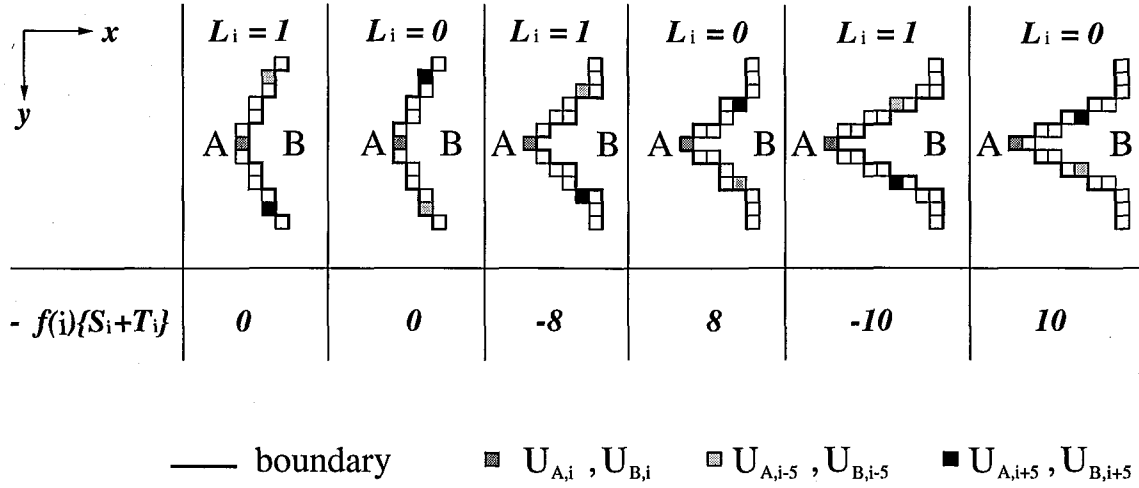
図 5.2 凹部と $-f(i)\{S_i+T_i\}$ の値

図 5.3 (a) では陰の影響のために首の境界線を正確に検出できていないが、図 5.3 (d) では良好に検出できていることが分かる。予備実験の結果、 δ の値を大きくすると、横方向の凹部が検出されなくなるが、エッジ強度の大きな部分における分割に失敗してしまう傾向がみられた。従って、 δ は、抽出精度が低下しない範囲で、最大の値に設定する必要があるといえる。本実験の顔領域抽出では、 $\delta = 10.0$ のときに良好な結果が得られた。

5.2 複数シードの場合への拡張

複数シードの場合へ本手法を拡張する方法について述べる。最初に、シード Q_z ($z = 1, 2, \dots, Z$) (Z : 抽出する領域数) を含む領域とその他の領域とに画像を分ける。そして、シード Q_z を含む領域内の画素 $P(x, y)$ に対してラベル $L(x, y) = z$ を、その他の画素に対してラベル $L(x, y) = 0$ を割り当てる。シードが一つの場合には、シードを含む任意の初期領域を設定することができるが、シードが複数ある場合には、どのように初期ラベルを割り当てるのかを、何らかのルールに基づいて決定する必要がある。本論文では、次の (1) から (3) に示すルールに従って決定する。

- (1) 画素 $P(x, y)$ とすべてのシード Q_z とのチェス盤距離を求める。



図 5.3 動的領域抽出結果

- (2) すべてのチェス盤距離が、あらかじめ与える距離 d よりも大きければ、背景領域に $P(x, y)$ を属させ、ラベル $L(x, y) = 0$ を割り当てる。
- (3) チェス盤距離が d 以下のシードが存在すれば、市街区距離が最も近いシード Q_{nearest} と同じラベル $L(x, y) = \text{nearest}$ を割り当てる。

初期領域を設定した後は、それぞれの領域を変形させることによって、二つ以上の領域を抽出することが可能である。図 4.7 に示したカラー標準画像において、シードを二つ与えて動的に領域を抽出した結果を図 5.4 および図 5.5 に示す。+ はシードの位置を、黒い直線で囲んだ領域は初期領域を表している。特に図 5.5 では、シードを含む二つの領域が互いに干渉し合うことで良好に抽出されていることが分かる。

5.3 移動物体追跡への応用

動画像中の移動物体追跡のための代表的な方法の一つに、フレーム間差分画像を用いる方法 [34] があるが、物体領域内の階調値が均一であったり、背景領域がフレーム間で変化するような場合には良い結果を得られない場合が多い。また、ブロックマッチングなどによって各フレーム間での対応点を決定する方法 [35] があるが、計算コストや精度に関して課題が残されている。そこで、動的領域抽出手法を応用して、背景の影響を受けずに移動物体を追跡する手法を提案する。

動画像中の移動物体追跡アルゴリズムは、次の (1) から (5) に示す処理ステップからなる。

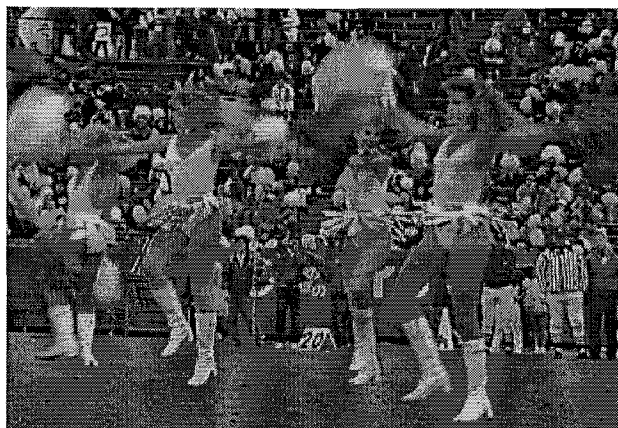
- (1) 第 1 フレーム画像において物体領域内の一画素をシードとする。
- (2) 整数変数 $q (q \in \{1, 2, \dots, T\}, T: \text{フレーム総数})$ の初期値を 1 とする。
- (3) 本手法を用いて、第 q フレーム画像を物体領域と背景領域とに分割する。
- (4) $q=q+1$ にした後、第 $q-1$ フレームで得られたラベルを第 q フレーム画像における初期ラベルとして、動的領域抽出を行う。但し、第 $q-1$ フレームにおいてユニットを割り当てた画素の平均座標をシードの位置とする。また、フレーム間での物体領域移動量が少ない場合には、初期ラベルにおいて、ある程度の抽出が得られているので、第 1 フレームに比べて局所解に陥る可能性が低い。そこで、ピラミッド画像を用いなくて、原画像のサイズで領域を変形させる。



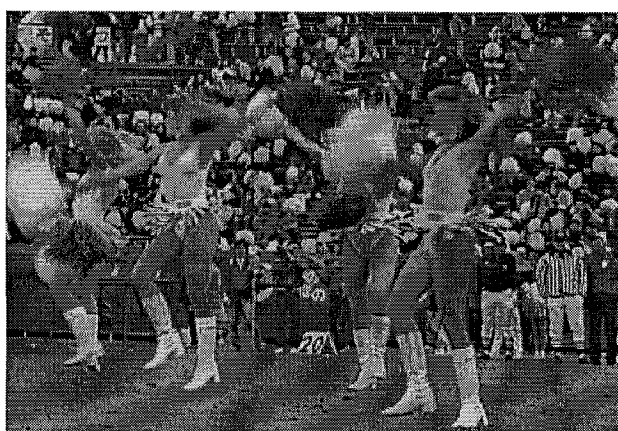
図 5.4 複数シードによる抽出結果 (Cloths)



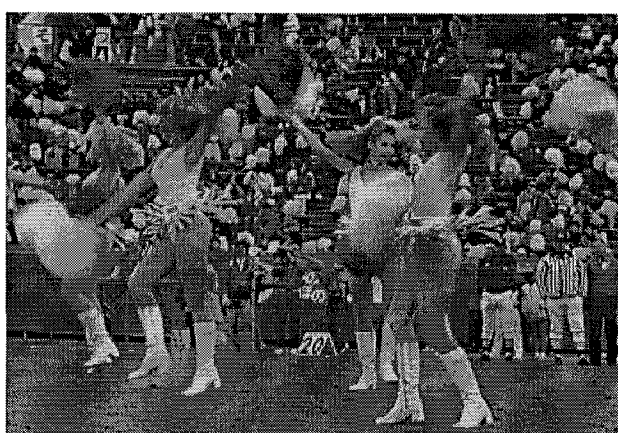
図 5.5 複数シードによる抽出結果 (A violin and a hand)



(a) Cheer093.



(b) Cheer096.



(c) Cheer099.

図 5.6 CCIR”Cheerleaders”

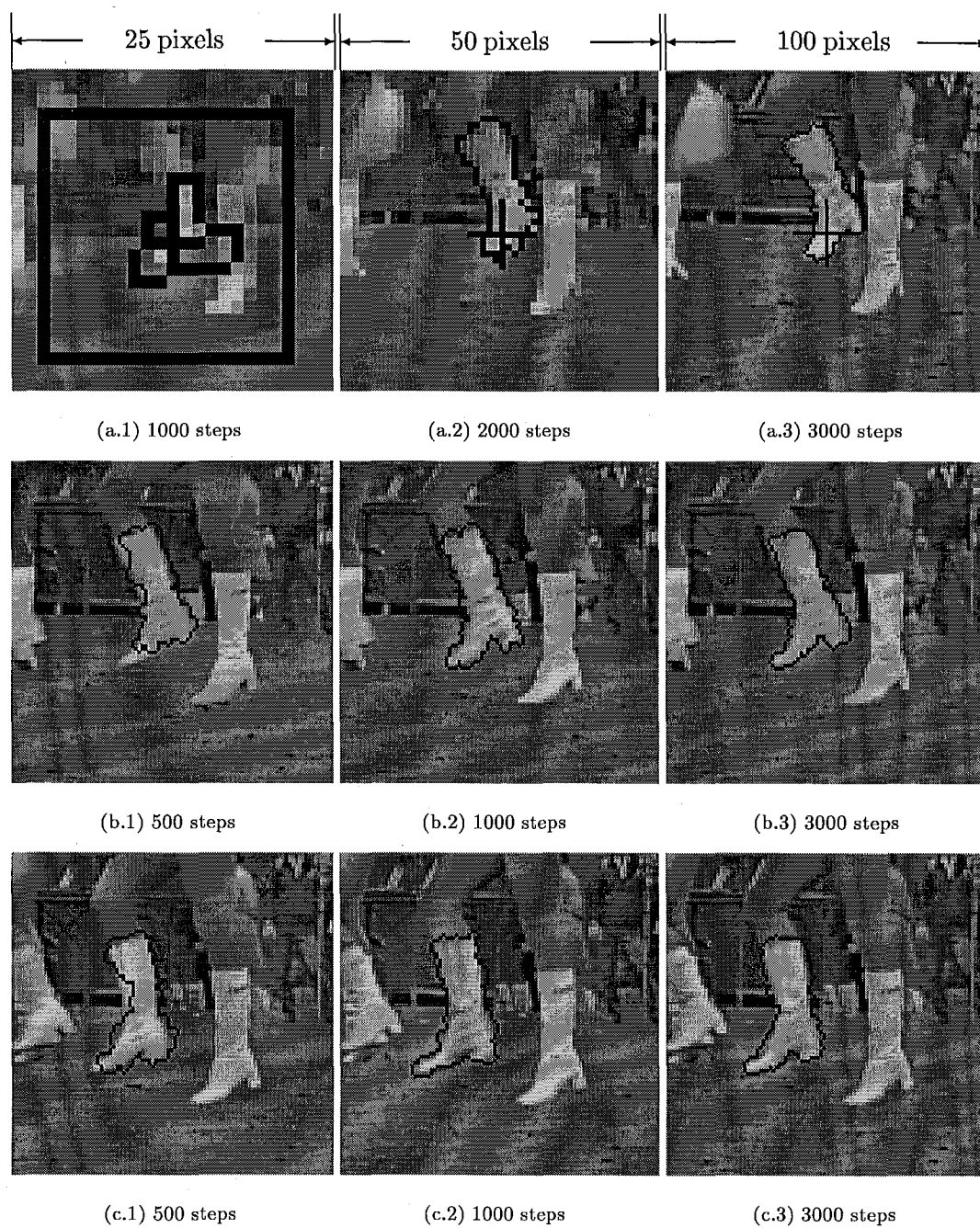


図 5.7 移動物体追跡の様子

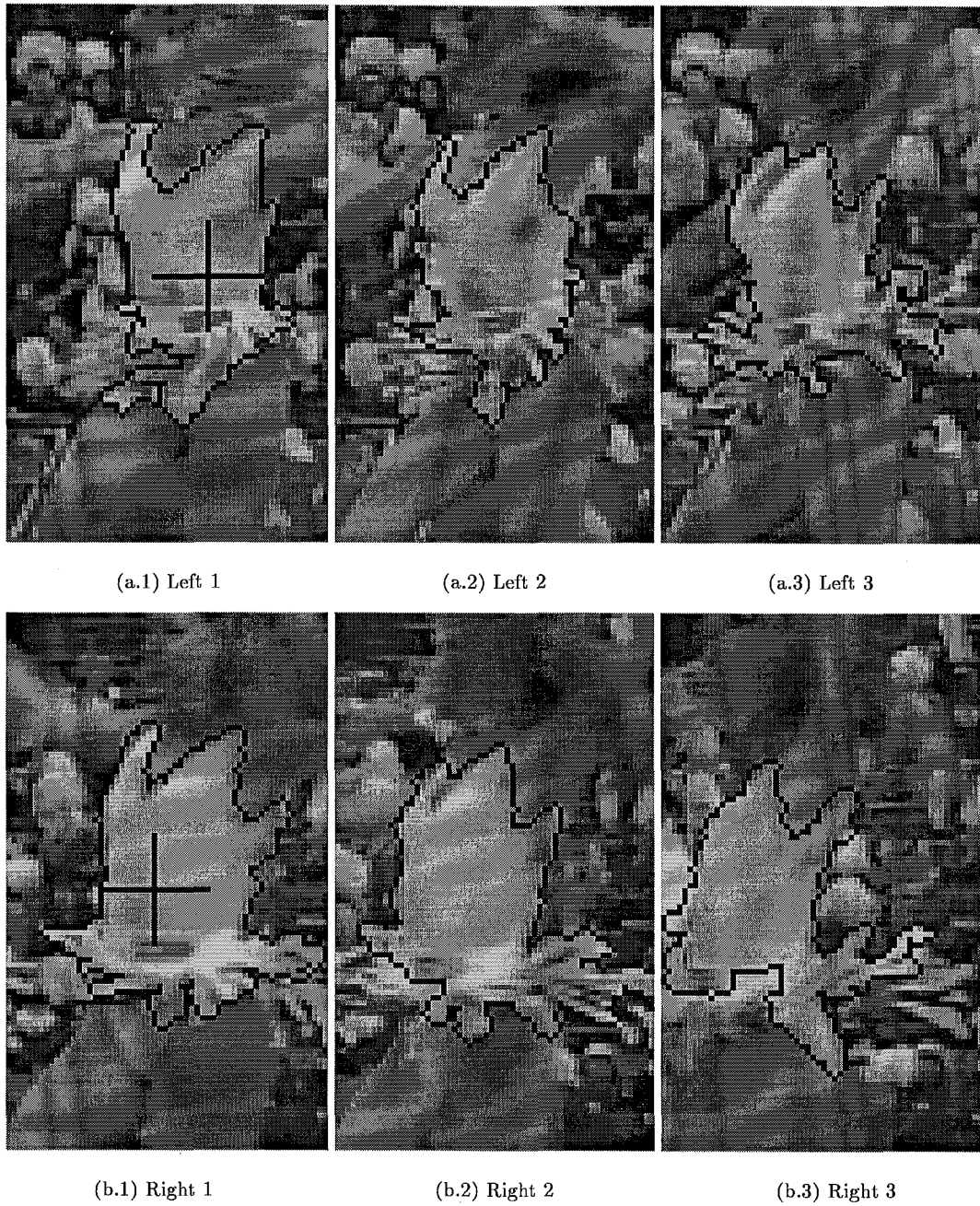


図 5.8 移動物体追跡結果

(5) 処理ステップ (4) を $T-1$ 回繰り返す.

図 5.6 に示す 352×240 画素 256 階調のカラー標準動画像において, 移動物体を追跡した様子を図 5.7 に示す. $+$ は, あらかじめ与えたシードの位置を, 図 5.7(a.1) の 20×20 画素の正方形で囲んだ領域は, あらかじめ与えた領域 A の初期状態を表している. 図 5.7(a) は, 図 5.6(a) に示した第 1 フレームにおいて, ピラミッド画像を用いて動的に領域を抽出した様子である. 図 5.7(b) と (c) は, それぞれ図 5.6(b) と (c) に示した第 2, 第 3 フレームにおいて, ピラミッド画像を用いなくて, 動的に領域を抽出した様子である. 但し繰り返しステップ数は, 各フレームにおける初期状態からの繰り返しステップ数である. 各フレームに要した計算時間は, SUN SPARC station 10 を用いて約 5 秒であった. 更に, 二人の人物の胴体に対応する領域を追跡した結果をそれぞれ図 5.8(a) および (b) に示す.

5.4 本章のまとめ

顔領域抽出のための目的関数を定義して, 4 章で提案した動的領域抽出手法を改良した結果について述べた. 次に, 二つ以上の領域抽出のために, 4 章で提案した動的領域抽出手法を拡張する方法について述べた. 最後に, 動画像中の移動物体追跡に, 4 章で提案した手法を応用して, 良好な結果が得られることを示した. 特に移動物体追跡では, 前フレームにおいて得られた物体領域を変形させるだけでよいため高速である.

第 6 章

階層型ニューラルネットワークの学習 を用いた動的領域抽出

画像の領域分割処理において、物体領域と背景領域との分割を行う場合に、その物体に関する知識をあらかじめ持つことで、良好な結果が得られる場合が多い。動画像を対象とする場合には、オプティカルフローやフレーム間差分画像を用いることで、物体領域を自動的に抽出できる場合がある [11]。しかし、静止画像を対象とする場合に、オプティカルフローやフレーム間差分画像を用いることはできない。従って、静止画像において、複数の均一な領域で構成される物体領域を、ひとまとまりにして抽出するためには、物体領域内のそれぞれの領域が持つ特徴や、領域間の関係などの情報を効率的に利用する必要がある。また、それらの特徴は、自動的に決定できることが望ましい。

4 章で提案した手法は、動的モデルを用いる手法 [9][10] と比較して、高精度な領域抽出が可能である。しかし、物体領域の特徴として色の均一性を仮定していたことから、複数の均一な領域で構成される物体領域を、ひとまとまりにして抽出するためには、シードを複数与えて抽出した領域を、更に統合する必要があった。

そこで、本章では、サンプルとしてあらかじめ与える物体領域の特徴を、階層型ニューラルネットワークの学習（付録 B 参照）によって自動的に獲得した後に、獲得した特徴との適合性に基づき、動的に領域を抽出する手法を提案する。一般的に物体領域の特徴は、変形に伴い次第に失われていくと考えられる。そこで、変形の度合い（変形度）を、特徴に関する適合性を調べるために用いる。本手法における階層型ニューラルネットワークは、この適合性を出力するように学習する。ネットワーク構造および入力信号については 6.3 で、学習データおよび学習過程について

は 6.4 で述べる。

階層型ニューラルネットワークが獲得した特徴に基づく領域抽出は，階層型ニューラルネットワークが記憶したパターンの連想 [36] に相当する．従って，連想するために何らかのキーが必要である．本手法では，従来の動的モデルで用いられてきたエッジ強度などに基づく画像エネルギーの最小化をキーとする．4.1.2 で定義した目的関数において，階層型ニューラルネットワークの出力値を用いることで，変形度の認識と動的領域抽出処理を統合する．そして，画像エネルギーの最小化と，階層型ニューラルネットワークが獲得した特徴との適合，すなわち，変形度の最小化を同時に満たすような分割を，互いをキーとすることで求める．

実験では，階層型ニューラルネットワークが獲得した特徴との適合性を重視すれば，オクルージョンによる不可視部分についても，可視部分からの連想によって抽出できることを示す．また，不可視部分を抽出する場合としない場合との切り替えが，パラメータの変更によって可能であることを示す．更に，本手法を用いた顔領域抽出実験の結果について述べる．

本手法における Hopfield 型ニューラルネットワークの目的関数は，境界線に沿った領域特徴量の連続性，領域の隣接する画素におけるエッジ強度，そして，領域の変形度に基づき定義される．但し，本論文における近傍領域は，注目画素を中心とする 5×5 画素の領域とする．

処理する画像が，図 6.1 の P に示すような $X \times Y$ 画素の濃淡画像 $P(x, y) (x \in \{0, 1, \dots, X-1\}, y \in \{0, 1, \dots, Y-1\}, n : \text{正定数})$ である場合に，本章の動的領域抽出アルゴリズムは，次の (1) から (4) に示す処理ステップからなる．

- (1) 抽出しようとする領域にある 1 画素 Q を与える．
- (2) 領域 A あるいは B 内で，他方の領域に隣接する画素に対して，図 6.1 に示すように，0 または 1 を出力する Hopfield 型ニューラルネットワークの一つのユニットをそれぞれ割り当てる．図 6.1 の $U_{R,j}$ ($R \in \{A, B\}, j \in \{0, 1, \dots, N_R-1\}$, N_R : ユニット総数) は，領域 R を選んだ場合に割り当てるユニットを表している．
- (3) ランダムに選んだ第 j ユニットの状態更新の結果，その出力値 $U_{R,j}$ が 0 の場合に，ラベルを 0 ならば 1 に，あるいは 1 ならば 0 に変更する．
- (4) 6.1 で述べる Hopfield 型ニューラルネットワークのエネルギー関数値が収束するまで，処理ステップ (2) と (3) を繰り返す．

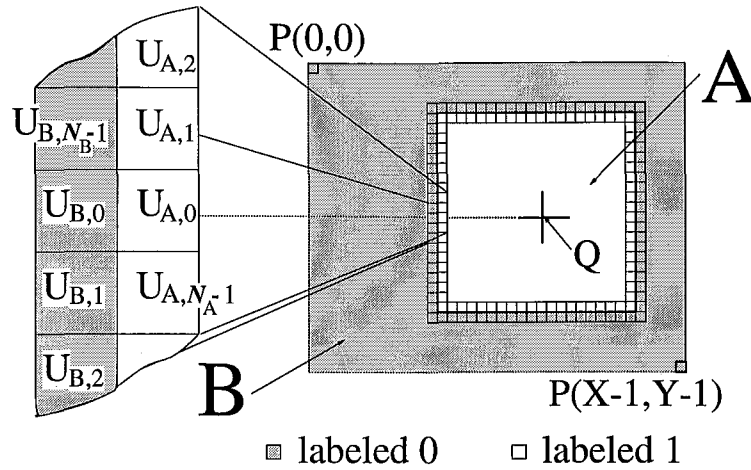


図 6.1 A と B の隣接画素とユニット

6.1 領域変形度認識と領域抽出の統合

本手法は、領域 A の特徴がサンプル物体領域の特徴にできるだけ適合するように、領域 A を動的に変形させる。ここで、サンプル物体領域を変形させることを考える。サンプル物体領域を変形させた領域を入力信号とした場合に、変形度に応じて小さな値を出力するような階層型ニューラルネットワークを構成できれば、その出力値を大きくするように入力信号、すなわち領域 A を変形させることで、最終的にサンプル物体領域の特徴に最も適合する領域 A が得られる。但し、できるだけ多くの変形過程を学習用データとして呈示すると共に、汎化能力を有する学習アルゴリズムを用いる必要がある。本手法では、学習用データを自動的に生成し、誤差逆伝播アルゴリズム [37] を用いて学習を行う。

階層型ニューラルネットワークを用いた領域変形度の認識と、Hopfield 型ニューラルネットワークを用いた動的領域抽出手法を統合する。統合に際して、4.1.2 で、最適な抽出状態が満たすべきであると仮定した、**仮定条件 3**「領域 A において、領域 B に隣接する画素の近傍領域とシード近傍領域の平均階調値が適合 (fit) している。」を、「領域 A の変形度が小さい。」に変更する。すなわち、階層型ニューラルネットワークがサンプル物体領域から獲得した特徴に、領域 A の特徴が適合していることを仮定する。従って、階層型ニューラルネットワークの出力値が大きくなる、す

なわち, 1.0 に近づくように領域 A を動的に変形させることになる.

Hopfield 型ニューラルネットワークにおいては, n 回目の更新における第 j ユニットの出力値 $U_j(n)$ を式 (6.1) に従って決定すれば, 式 (6.2) に示すエネルギー関数 E_{NN} の値は決して増加しない (付録 A 参照).

$$U_j(n) = \begin{cases} 1 & \text{if } \sum_{k=0}^{N-1} w_{jk} U_k(n-1) - h_j > 0 \\ 0 & \text{otherwise} \end{cases} \quad (6.1)$$

$$E_{NN} = -\frac{1}{2} \sum_{j=0}^{N-1} \sum_{k=0}^{N-1} w_{jk} U_j U_k + \sum_{j=0}^{N-1} h_j U_j \quad (6.2)$$

w_{jk} : 第 k ユニットから第 j ユニットへの結合荷重.

h_j : 第 j ユニットのしきい値.

本手法では, 第 j ユニットを割り当てる画素 $P_{\text{unit}}(x_j, y_j)$ の近傍領域で, $P_{\text{unit}}(x_j, y_j)$ と同じラベルを割り当てられている画素を $P_{\text{same}}(x_j+u, y_j+v)$ ($u, v \in \{-2, -1, 0, 1, 2\}$), 異なるラベルを割り当てられている画素を $P_{\text{diff}}(x_j+u, y_j+v)$, $P_{\text{same}}(x_j+u, y_j+v)$ の平均階調値を $\overline{P_{\text{same}}}(x_j, y_j)$, $P_{\text{diff}}(x_j+u, y_j+v)$ の平均階調値を $\overline{P_{\text{diff}}}(x_j, y_j)$ とする. そして, 最適な領域分割において各領域が満たすべき条件を, 仮定条件 1 から仮定条件 3 に示すように仮定する.

仮定条件 1: $\overline{P_{\text{same}}}$ は境界線に沿って連続 (continuous) である.

仮定条件 2: P_{unit} におけるエッジ (edge) 強度は, その他の画素よりも大きい.

仮定条件 3: 領域 A の変形度が小さい.

そして, $U_{R,j} = 0$ ($R \in \{A, B\}$) であるとき, $P_{\text{unit}}(x_j, y_j)$ が領域 R に属さない, すなわち, ユニットを割り当てる画素として存在しないことを, $U_{R,j} = 1$ であるとき, 存在することを仮定すれば, 式 (6.3) から式 (6.6) で定義する E_{cont} , E_{edge} , そして E_{fit} の値が小さくなる程, 仮定条件 1 から仮定条件 3 にそれぞれ近付いたといえる.

注目画素 $P_{\text{unit}}(x_j, y_j)$ から, 境界線に沿ってある一定距離離れた二つの $\overline{P_{\text{same}}}(x_k, y_k)$ の, 少なくとも一方が $\overline{P_{\text{same}}}(x_j, y_j)$ と等しい場合に, 値が 0 になるような E_{cont} を式 (6.3) に示すように定義する. 本論文では, 予備実験の結果に基づき, この距離を 5 画素に設定する.

$$E_{\text{cont}} = \sum_{j=0}^{N-1} \sum_{k=0}^{N-1} f(\delta_{j+5,k} \cdot |\overline{P_{\text{same}}}(x_j, y_j) - \overline{P_{\text{same}}}(x_k, y_k)|, \delta_{j-5,k} \cdot |\overline{P_{\text{same}}}(x_j, y_j)$$

$$- \overline{P_{\text{same}}}(x_k, y_k) \Big|) U_j U_k \quad (6.3)$$

$$f(a, b) = \begin{cases} a & \text{if } a \leq b \\ b & \text{otherwise} \end{cases} \quad (6.4)$$

$\delta_{j,k}$: クロネッカーのデルタ.

$|\overline{P_{\text{same}}} - \overline{P_{\text{diff}}}|$ を, P_{unit} におけるエッジ強度とする. 第 j ユニットを割り当てた画素のラベルを変更したときのエッジ強度の変化 $\Delta |\overline{P_{\text{same}}} - \overline{P_{\text{diff}}}|$ を用いて, E_{edge} を式 (6.5) に示すように定義する.

$$E_{\text{edge}} = \sum_{j=0}^{N-1} \Delta |\overline{P_{\text{same}}}(x_j, y_j) - \overline{P_{\text{diff}}}(x_j, y_j)| U_j \quad (6.5)$$

分割パターンを入力した場合の階層型ニューラルネットワークの出力値を O_n , 第 j ユニットを割り当てた画素のラベルを変更した分割パターンを入力した場合の階層型ニューラルネットワークの出力値を $O_n(j)$ とする. そして, E_{fit} を式 (6.6) に示すように定義する.

$$E_{\text{fit}} = \sum_{j=0}^{N-1} \{O_n(j) - O_n\} U_j \quad (6.6)$$

Hopfield 型ニューラルネットワークの目的関数 E_{obj} を, 式 (6.7) に示すように定義する.

$$E_{\text{obj}} = \alpha E_{\text{cont}} + \beta E_{\text{edge}} + \gamma E_{\text{fit}} \quad (6.7)$$

α, β, γ : 正の実数.

E_{obj} の最小値を与えるラベルの状態を, 最適な領域抽出結果であると仮定する. E_{obj} が E_{NN} と恒等的に等しくなるように, 係数比較によって w_{jk} および h_j を決定して Hopfield 型ニューラルネットワークを構成すれば, ランダムに選んだユニットの状態を式 (6.1) に従って更新していくことによって, E_{obj} が極小値になる U_j を求めることができる. w_{jk} を式 (6.8) に, h_j を式 (6.9) に示す.

$$w_{jk} = -2\alpha f(\delta_{j+5,k} \cdot |\overline{P_{\text{same}}}(x_j, y_j) - \overline{P_{\text{same}}}(x_k, y_k)|, \delta_{j-5,k} \cdot |\overline{P_{\text{same}}}(x_j, y_j) - \overline{P_{\text{same}}}(x_k, y_k)|) \quad (6.8)$$

$$h_j = \beta \Delta |\overline{P_{\text{same}}}(x_j, y_j) - \overline{P_{\text{diff}}}(x_j, y_j)| + \gamma \{O_n(j) - O_n\} \quad (6.9)$$

6.2 階層型ニューラルネットワークの構造と入力信号

本手法における階層型ニューラルネットワークは、図 6.2 に示すように、入力層、隠れ層、出力層の三つの層からなる。

変形の途中にある領域 A だけを用いて特徴を獲得すると、例えば、図 6.3 に示すような場合に、太い線で表す領域 A が物体領域の内部であることを認識できない。そこで、本手法では、図 6.2 に示すように、領域 A と B、それぞれについて入力ユニットを用意して、領域 A と B の変形によって失われる特徴を共に獲得する。

本手法では、物体領域サイズの影響を受けないようにするために、画像 P 全体ではなく、領域 A を含む P の一部を階層型ニューラルネットワークへの入力パターンにする。しかし、図 6.4 の I で示すような矩形領域を入力パターンにすると、A の輪郭と I の辺とが一致する部分では、A が物体領域の内部であることを認識することが困難になる。そこで、本手法では、A に外接する矩形領域よりも、大きめの領域を入力パターンとして用いる。本論文では、図 6.2 に示すように、縦横 10 画素分大きな $W \times H$ 画素 ($W = x_{\max} - x_{\min} + 10, H = y_{\max} - y_{\min} + 10$) の領域を入力パターンにする。

そして、A を含む $W \times H$ 画素の領域を $L \times L$ 個のブロックに分割する。本論文では、図 6.2 に示すように $L = 9$ に設定した。図 6.2 に示す $d_{x,0}, d_{x,1}, d_{x,2}, \dots, d_{x,8}$ および $d_{y,0}, d_{y,1}, d_{y,2}, \dots, d_{y,8}$ は、式 (6.10) から式 (6.14) に従って求める。

$$d_{x,0} = d_{x,8} = W/9 + (W\%9)/2 \quad (6.10)$$

$$d_{x,i} = W/9 \quad (6.11)$$

$$d_{y,0} = d_{y,8} = H/9 + (H\%9)/2 \quad (6.12)$$

$$d_{y,i} = H/9 \quad (6.13)$$

$$i = 1, 2, 3, \dots, 7 \quad (6.14)$$

そして、各ブロック内の画素の平均階調値を $[0.0, 1.0]$ に正規化した値を階層型ニューラルネットワークの入力ユニットへ、図 6.2 の矢印で示すように入力する。従って、階層型ニューラルネットワークの入力層は、 $L \times L \times 2$ 個の入力ユニットからなる。本論文では、162 個である。隠れユニット数を多くすれば、学習用入力データに対しては正しい出力をするようになる。しかし、隠れユニット数が多過ぎると、一般的に汎化能力が低下する。隠れユニット数の決定法は、幾つか提案されているが [39],

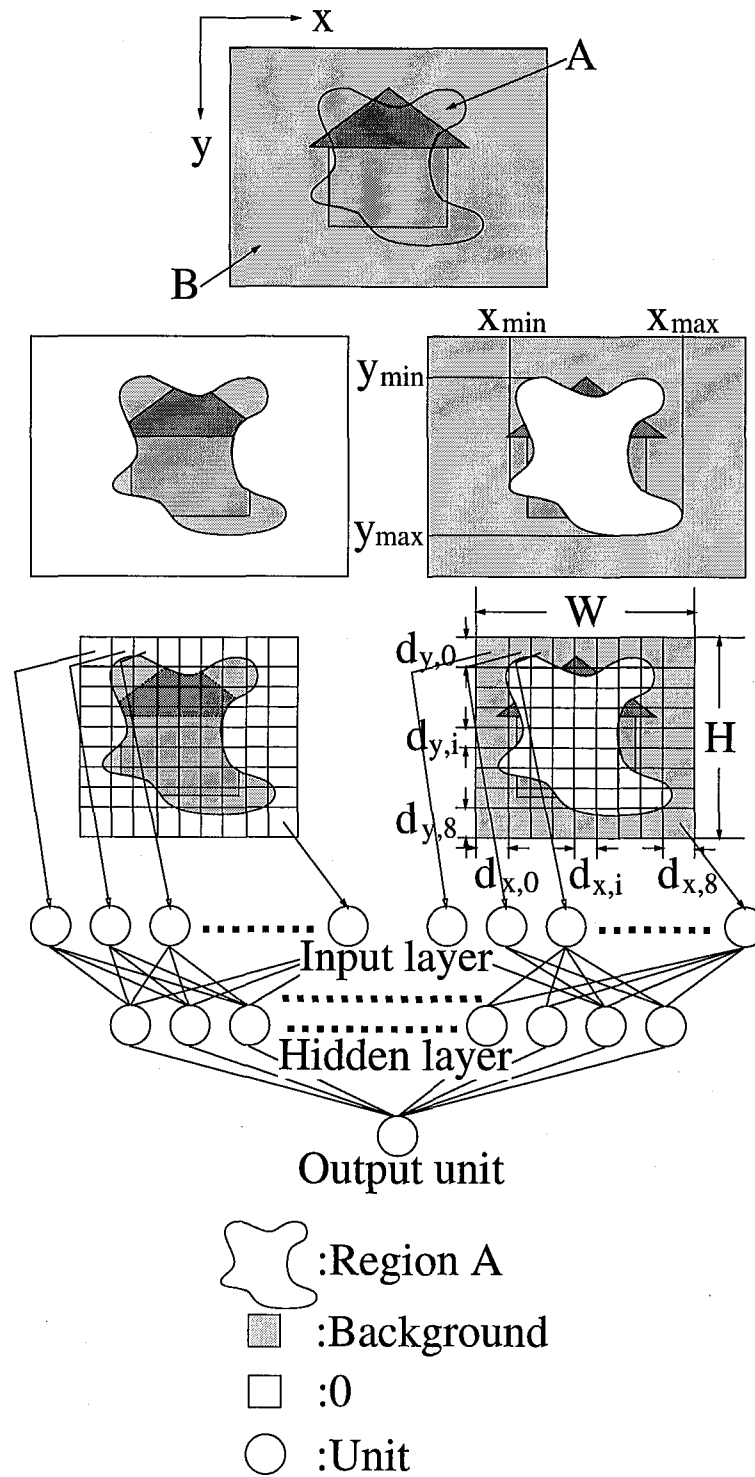


図 6.2 階層型ニューラルネットワークの構造と入力信号

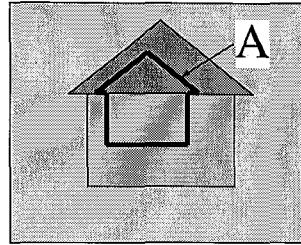


図 6.3 問題点 1

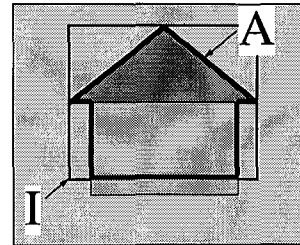


図 6.4 問題点 2

本論文では試行錯誤的に隠れユニット数を 3 に決定して，自動的決定は今後の課題とする．出力ユニットは一つにする．

ユニットの入出力関数は，式 (6.15) に示すロジスティック関数 $f(u)$ とする．

$$f(u) = 1/(1 + e^{-u}) \quad (6.15)$$

u : ユニットへの入力-しきい値

6.3 学習用データ自動生成と学習過程

本手法では，学習用データを自動的に生成する．変形一回につき，領域の隣接する 1 画素のラベルを変更する．サンプル画像における領域の隣接する画素の数を N ，変形回数を T とすれば，変形パターンは N^T 個存在する．膨大な数の変形パターンを，学習用データとしてあらかじめ用意することは，記憶容量の限界のために困難である．そこで，変形領域と変形度の一般化した関係を獲得すると共に，データ量を最小限に抑えるために，二つの変形パターンの生成および学習を 1 ステップとして繰り返す追加学習 [38] を行う．

学習過程は次の (1) から (6) に示す処理ステップからなる．

- (1) 6.2 で述べた階層型ニューラルネットワークの結合荷重およびしきい値を $[-3.0, 3.0]$ のランダムな値に設定して保存する．
- (2) 領域の隣接する画素をランダムに選択して，割り当てられたラベルを変更する．これを T 回繰り返して領域を変形させる．本論文における T は，物体領域と同じ面積の正方形領域を，物体領域と中心が一致するように重ねた場合

に、重ならない部分の画素数とする。言い換えれば、物体領域を変形させて、この正方形領域にするための最小ラベル変更回数である。T を 9 で割った回数毎、すなわちラベル変更回数 $T/9 \times t$ ($t = 0, 1, 2, \dots, 9$) における変形領域を学習用データの入力部とする。但し、背景領域の階調値には制約がない。すなわち背景領域の階調値が如何に変化しても物体領域を抽出できる必要がある。そこで背景領域における階調値は、0 から 255 のランダムな値にする。変形度を $0.1 \times t$ 、出力部すなわち教師信号を $1.0 - 0.1 \times t$ とする。従って、変形前の領域に対する教師信号は 1.0、変形回数 T の領域に対する教師信号は 0.1 である。結果的に、変形領域とその変形度の組み合わせを自動的に生成する。これを、学習用データセット I とする。

- (3) 処理ステップ (2) と同様にして、学習用データセット II を生成する。
- (4) 階層型ニューラルネットワークの結合荷重およびしきい値を読み込み、誤差逆伝播アルゴリズムを用いて学習を行う。学習用データセット I と学習用データセット II を提示した後に、結合荷重およびしきい値を一括修正する。本手法における追加学習では、学習が完全に行われることを仮定しない。そこで、本論文では、修正回数を 100 回に設定する。結合荷重の修正量 Δw は、式 (6.16) および (6.17) に従って求める (付録 B.2 参照)。但し、しきい値については、常に -1.0 を出力するユニットを入力層および隠れ層に、それぞれ一つずつ追加して、それとの結合荷重として修正する。

$$\begin{aligned} \Delta w_{nm}(t) = & -\eta \sum_{I,II} \{(O_n - D)(1 - O_n) \cdot O_n \cdot O_m\} \\ & + \theta \cdot \Delta w_{nm}(t-1) \end{aligned} \quad (6.16)$$

$$\begin{aligned} \Delta w_{ml}(t) = & -\eta \sum_{I,II} \{(O_n - D)(1 - O_n) \cdot O_n \cdot w_{nm}(t) \\ & \cdot (1 - O_m) \cdot O_m \cdot O_l\} + \theta \cdot \Delta w_{ml}(t-1) \end{aligned} \quad (6.17)$$

w_{nm} : 第 m 隠れユニット ($m \in 0, 1, 2, 3$) と出力ユニット間の結合荷重。

η : 学習係数。

D : 教師信号。

O_m : 第 m 隠れユニットの出力。

θ : 慣性項係数。

w_{ml} : 第 l 入力ユニット ($l \in 0, 1, 2, \dots, 162$) と第 m 隠れユニット間の結合荷重.

O_l : 第 l 入力ユニットの出力.

- (5) 階層型ニューラルネットワークの結合荷重およびしきい値を保存する.
- (6) 処理ステップ (2) から (5) を, すべてのサンプル画像について順に行い, 階層型ニューラルネットワークの平均 2 乗誤差があらかじめ与える値 ϵ よりも小さくなるまで繰り返す.

本論文では, 予備実験の結果に基づき, 学習係数 η の初期値を 1.0 にして, 処理ステップ (4) を 100 回繰り返す毎に, 式 (6.18) に従って変更する.

$$\eta = \eta / 10.0 \quad (6.18)$$

また, 慣性項係数 θ は, 0.01 にする.

本論文では, 図 6.5 に示す四つのサンプル画像を用意した. 図 6.5 (b), (c), (d) は, (a) に対する簡単なオクルージョンを想定して, それぞれ (a) の右部, 上下部, 左部を削除することで生成した. 乱数の初期値を変更することで膨大な量の学習用データを生成できる. 図 6.5 (a) について生成した学習用データセットの例を図 6.6 に示す.

学習が進むにつれて, 変形度が大きいほど小さな値を出力するような階層型ニューラルネットワークの結合荷重およびしきい値が得られる. すなわち, 階層型ニューラルネットワークは, 変形によって失われる, サンプル物体領域の特徴を獲得する.

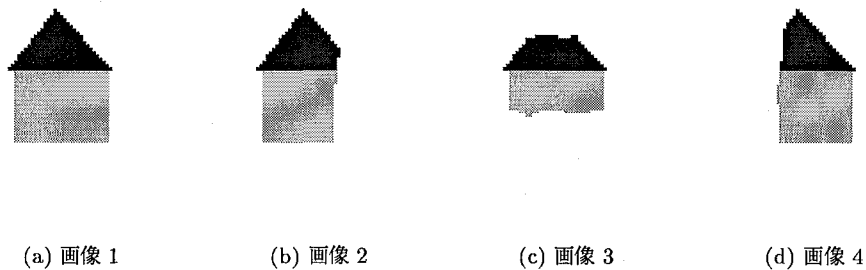


図 6.5 サンプル画像

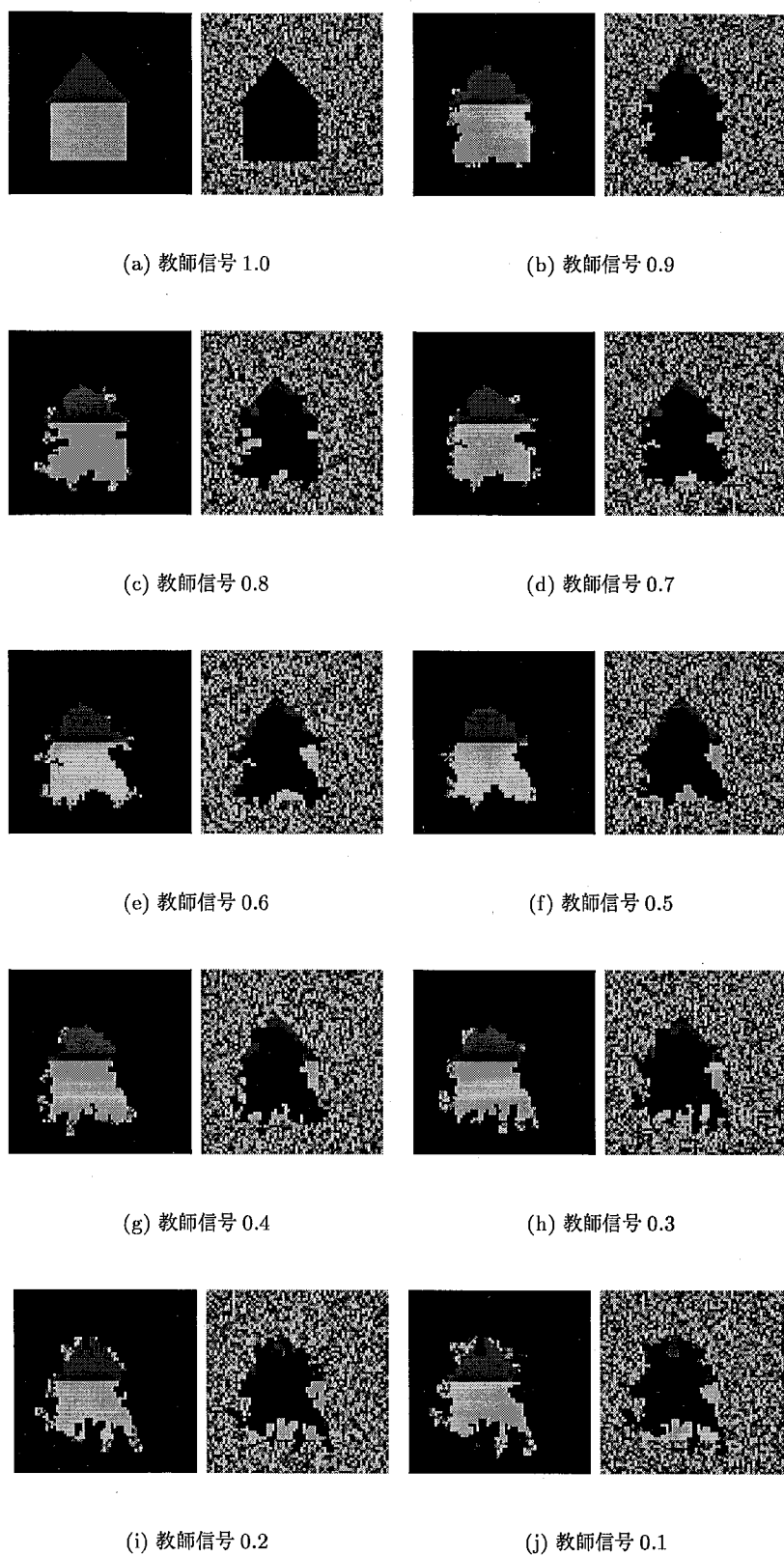


図 6.6 学習用データセットの一例

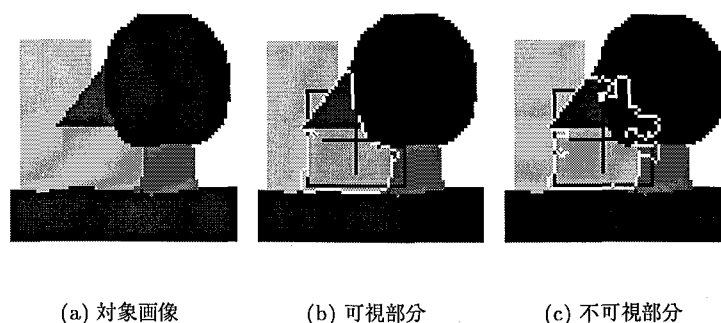


図 6.7 抽出結果

6.4 実験結果

図 6.5 に示したサンプル画像について学習を行った後に、得られた階層型ニューラルネットワークを用いて、図 6.7 (a) に示す 70×70 画素 256 階調の濃淡画像を動的に分割した。図 6.7 (a) は、図 6.5 (a) を家であると仮定して、その左後方にビル、右前方に樹木を配置したシーンを表すパターン画像である。Hopfield 型ニューラルネットワークの目的関数におけるパラメータは、予備実験の結果に基づき、 $\alpha = 0.1, \beta = 1.0, \gamma = 100.0$ に設定した。図 4.1 に示したように、+ で Q の位置を、正方形で囲んだ領域であらかじめ与えた領域 A の初期状態を表している。ユニットを割り当てた画素は白で表示している。画像エネルギー最小化と、サンプル画像から獲得した特徴に対する適合を同時に満たす抽出として、図 6.7 (b) に示す結果が得られる。更に、サンプル画像の特徴に対する適合性をより重視して、パラメータ $\gamma = 3000.0$ にすると、図 6.7 (c) に示す結果が得られ、木で隠れている部分を推定できることが分かる。従って、階層型ニューラルネットワークの出力値を重視すれば、オクルージョンによる不可視部分を、可視部分からの連想によって抽出することが可能であると考えられる。

図 6.5 に示したようなサンプル画像について学習すれば、階層型ニューラルネットワークは、サンプル画像の特徴として、上部が暗く下部が明るいといったような特徴を獲得すると思われる。

以上の実験をふまえ、本章では、さまざまな背景から顔領域をひとまとまりにして抽出した結果について述べる。まず、図 6.5 のサンプル画像を学習した階層型ニューラルネットワークを用いて、図 6.8 (a) に示す人物の顔領域を抽出した。図



(a) ミュージシャン

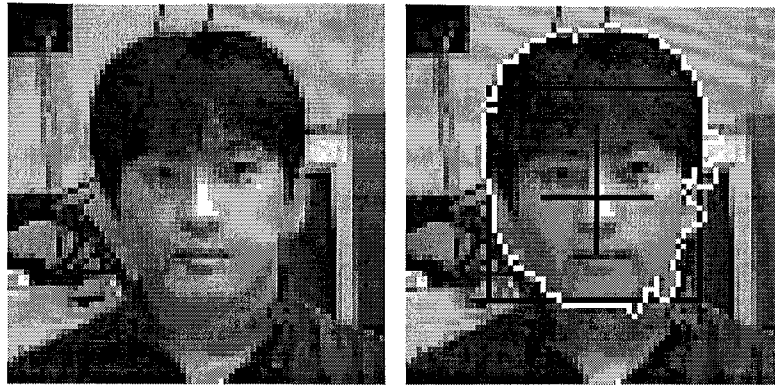
(b) 抽出結果

図 6.8 顔領域抽出結果（標準画像）

6.8 (a) の 100×100 画素の画像は、カラー標準画像“ミュージシャン”の R 成分である。特に頭髮部と背景の階調値がほぼ等しいために、従来手法では抽出が困難な例であるが、図 6.8 (b) に示すように良好に抽出することができる。更に、図 6.8 (b) の顔領域を新たに学習して、図 6.9 (a) および図 6.10 (a) に示す 70×70 画素の画像から顔領域を抽出した結果を、それぞれ、図 6.9 (b) および図 6.10 (b) に示す。背景が複雑な例であるが、顔領域の特徴を用いることで、良好に抽出できることが分かる。また、図 6.10 (b) において、顔領域の左下部がうまく抽出できていない原因としては、背景に強いエッジが存在することと、図 6.8 (b) における顔領域の特徴、特に首部分の特徴との適合が考えられる。従って、画像エネルギーの最小化と、図 6.8 (b) をサンプルとして獲得した特徴との適合を共に満たす抽出結果であるといえる。そこで、図 6.8(b) と図 6.9(b) の顔領域をサンプルとして学習した後に、図 6.10(a) の顔領域を再び抽出すると、図 6.10(c) に示す結果が得られる。従って、さまざまな顔領域を抽出するためには、幾つかのパターンをあらかじめ学習しておく必要があると考えられる。

6.5 本章のまとめ

学習によってあらかじめ獲得したサンプル物体領域の特徴を用いて、複数の均一な領域で構成される物体領域を、ひとまとまりにして抽出できる全く新しい手法を

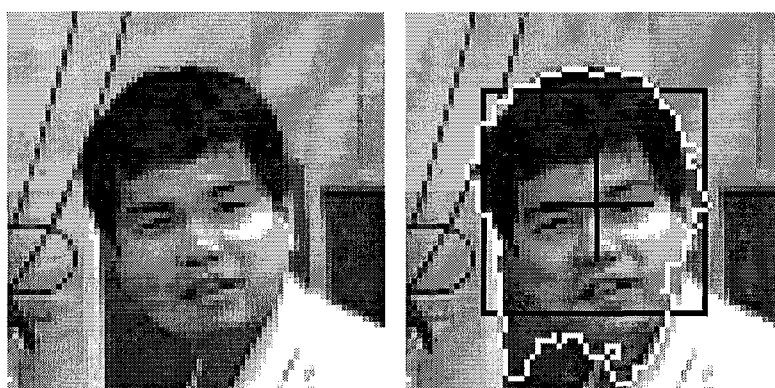


(a) 低画質画像 1

(b) 抽出結果

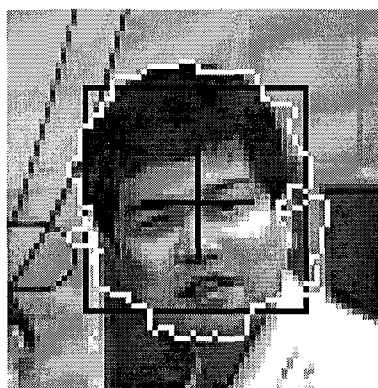
図 6.9 顔領域抽出結果（低画質画像 1）

提案した．本章の実験では，顔領域を抽出したが，より多くの学習用データを学習すれば，さまざまな物体領域を抽出することが可能であると考えられる．更に，ニューラルネットワークや遺伝的アルゴリズムによる物体抽出手法 [40], [41] を用いて，物体領域を矩形領域として抽出することで，物体領域の位置および大きさを推定できる．これらの手法を利用することで推定した位置・大きさと，学習用データ自動生成における変形回数 T に基づき，領域 A の初期状態を与えれば，自動抽出システムを構築することも可能である．



(a) 低画質画像 2

(b) 抽出結果 1



(c) 抽出結果 2

図 6.10 顔領域抽出結果（低画質画像 2）

第 7 章

結論

7.1 本研究のまとめ

一部がぼやけたり途切れたりしている境界線は，検出が困難であることが多い．そこで，3 章では，カラー画像から，一部がぼやけたり途切れたりしている場合でも，さまざまな形状の境界線を検出できる手法を提案した．3 章で提案した手法では，Hopfield 型ニューラルネットワークを用いて選択したエッジセグメントの位置に基づき，処理領域を移動しながら，ひとつつながりの線構造を徐々に形成した．実験では，発火ユニットに対応するエッジセグメントを統合することで，境界線を良好に検出できることを示した．

画像の領域抽出処理を不良設定問題としてとらえる場合，最小解が領域抽出結果に対応するような目的関数を定義して，何らかの拘束条件のもとでそれを最小化する方法が有効である．そこで，4 章では，多目的関数の極小化に有効な Hopfield 型ニューラルネットワークを用いて，動的に領域を抽出する手法を提案した．Hopfield 型ニューラルネットワークを用いる場合には，局所解に陥る可能性が高いので，大局解に対応する領域抽出を行うと共に，かつ高速に処理を行うために，ピラミッド画像を用いた．実験では，Hopfield 型ニューラルネットワークのエネルギー関数値極小化原理を用いて，動的に領域を抽出できることを示した．

4 章で提案した手法では，陰などの影響のために，良好に物体領域を抽出できない場合があった．また，一つの領域抽出に限定した手法であった．そこで，5 章では，形状に関する仮定に基づく項を目的関数に追加することで人物の顔領域を良好に抽出する手法を提案した．そして，二つ以上の領域を抽出するための手法を提案

した。更に、動画像中の移動物体追跡に応用して、良好な結果が得られることを示した。

静止画像において、顔のように複数の均一な領域からなる物体領域を、ひとまとまりにして抽出するためには、抽出しようとする物体領域の特徴を柔軟に決定する必要がある。そこで、6章では、サンプルとしてあらかじめ与える物体領域の特徴を、階層型ニューラルネットワークを用いた学習によって自動的に獲得した後に、獲得した特徴との適合性に基づき、動的に物体領域を抽出する手法を提案した。具体的には、4章で提案した動的領域抽出手法において、学習後の階層型ニューラルネットワークを、獲得した特徴との適合性を調べるために用いた。実験では、オクルージョンによる不可視部分を、可視部分からの連想によって抽出することが可能であることを示した。更に、提案手法によって顔領域を抽出できることを示した。

7.2 今後の課題

物体領域の抽出は、画像認識・理解・画像合成・画像解析などにおいて重要であり、その応用は多岐にわたっている。例えば、セキュリティ管理における個人識別システム、表情認識、画像通信のための情報量圧縮などがある。提案手法の具体的な応用とその際の問題点について、次の(1)から(3)にそれぞれ述べる。

(1) Hopfield 型ニューラルネットワークを用いたカラー画像における境界線検出

画像合成などの目的で、ユーザーがインタラクティブに境界線検出の開始点を与えることで、境界線を検出することができる。この際、境界線が分岐している場合が問題であり、対策としては、分岐が検知される度に、ユーザーが検出しようとする境界線を指定する方法が考えられる。あるいは、6章で提案した手法と同様に、あらかじめサンプルとして与える境界線から、境界線形状や、境界近傍における色の分布などの特徴を獲得して、獲得した特徴に従って検出する境界線を自動的に選択するなどの方法が考えられる。また、物体認識などの目的で、テンプレートマッチングや従来の領域分割手法などを用いて、概略の位置を検出した後に、開始点を自動的に決定できれば、自動的に境界線を検出できる。本手法は、動的モデルと比較して高精度な境界線形状が得られるため、境界線形状に基づく認識において有効である。但し、開始点や分岐の影響で、誤った境界線が検出される可能性があるため、検出した境界線を何らかの方法で評価する必要がある。更に、パラメータの自動決定や、一つの境界線が

途切れているのか、あるいは別々の境界線であるのかを判断するための基準についても考察する必要がある。

- (2) Hopfield 型ニューラルネットワークを用いたカラー画像における動的領域抽出
テンプレートマッチングや従来の領域分割手法などを用いてシードを自動的に検出することで、領域の自動抽出が可能である。但し、複数の均一な領域をひとまとまりにして抽出するためには、更に、抽出した領域を統合する必要がある。また、画像全体に均等にシードを配置した後に、それぞれの領域を変形させると共に、領域の淘汰を行い、最終的に最適な領域分割を得るなどの応用も考えられる。物体の輪郭を正確に抽出できなければ、背景が識別・認識結果に重大な影響を及ぼす。また、画像通信においては、背景が残ることによって、冗長な情報を伝送・再生することになってしまう。そこで、本手法を用いて対象物体領域だけを正確に抽出できれば、背景の影響を取り除いて、識別・認識精度や通信効率を向上させることができると考えられる。

- (3) 階層型ニューラルネットワークの学習を用いた動的領域抽出

本論文では、階層型ニューラルネットワークの出力ユニットを一つにして、一つのカテゴリに属する物体領域を抽出したが、出力ユニットを複数にすることで、複数のカテゴリを学習することも可能である。複数のカテゴリを学習した階層型ニューラルネットワークの出力パターンに基づき、物体領域を抽出すれば、従来別々の処理として位置付けられる場合が多かった、物体抽出と物体認識を、同時に行うことができると考えられる。この際、カテゴリの数と、必要なネットワークの規模など考察すべき内容は多い。更に、カラー画像への適用、学習と抽出精度の関係に関する考察、そして、高速化と大局的解探索のための方策として、ピラミッド画像を用いることについても検討する必要がある。

通信符号化における応用としては、視聴者の意図に応じて伝送記録すべき情報を取捨選択することが考えられる。例えば、テレビ電話やテレビ会議では、関心の対象は顔であることが多いことから、顔領域の自動抽出によって、顔領域に多くの符号量を割り当てる符号化が効率的である。従って、抽出した物体領域に応じて異なる符号量を割り当てることによって、符号化効率が向上すると考えられる。

付録 A

Hopfield 型ニューラルネットワークを用いた最適化

Hopfield 型ニューラルネットワークの重要な応用として、最適化問題への適用がある。最適化問題とは与えられた制約条件のもとで、ある目的関数を最小あるいは最大にする解を求める問題をいう。工学、自然科学、社会科学などで扱われる問題の多くはこのように定式化される [31]。最適化問題は、画像復元などの画像処理、ポートフォリオ選択、遺伝子配列の解析などさまざまな分野でみられる [42]。

ニューラルネットワークによる最適化問題へのアプローチの方法は、次の二つに分類することができる。

- 決定論的な方法—Hopfield 型ニューラルネットワーク
- 確率論的な方法—焼きなまし法 (Simulated Annealing) ・ボルツマンマシン

目的関数は、評価関数とも呼ばれ、目的関数をどのように定義するのが重要である。目的は、複数の目的を同時に含んでいることが多く、各目的がトレードオフの関係にあるときが問題である。このような場合、目的関数は、各目的の荷重和で表現する。

また、Hopfield 型ニューラルネットワークやボルツマンマシンは、制約条件を満たしているとは限らない初期状態から出発して、状態を変化していくに従い、ユニットの結合に関する局所的な制約条件を満たすような状態を求める方法であり、このような方法は 2.1.2 および 2.2.4 で述べた弛緩法 (relaxation method) と類似している。

Hopfield 型ニューラルネットワークは、相互に対称な結合を持ち、各ユニットは、他のユニットからの入力を受け、発火状態において 1 を、休止状態において 0 を出力する。n 回目の更新における第 j ユニットの出力値 $U_j(n)$ を式 (A.1) に従って決定すれば、式 (A.2) に示すエネルギー関数 E_{NN} の値は決して増加しない。

$$U_j(n) = \begin{cases} 1 & \text{if } \sum_{k=1}^N w_{jk} U_k(n-1) - h_j > 0 \\ 0 & \text{otherwise} \end{cases} \quad (\text{A.1})$$

w_{jk} : 第 k ユニットから第 j ユニットへの結合荷重。

h_j : 第 j ユニットのしきい値。

$$E_{NN} = -\frac{1}{2} \sum_{j=1}^N \sum_{k=1}^N w_{jk} U_j U_k + \sum_{j=1}^N h_j U_j \quad (\text{A.2})$$

証明：

時刻 $n-1$ から n において、第 j ユニット U_j のみ変更があり、 $U_{k \neq j}$ には変更がないとすると、j のみ $U_j(n) \neq U_j(n-1)$, $U_{k=j}(n) \neq U_{k=j}(n-1)$ であるから、エネルギー関数を式 (A.3) に示すように表せば、エネルギー関数値の変化 ΔE は式 (A.4) に示すようになる。

$$\begin{aligned} E_{NN} = & -\frac{1}{2} [\{ w_{11} U_1 U_1 + w_{12} U_1 U_2 + \cdots + w_{1N} U_1 U_N \} \\ & + \{ w_{21} U_2 U_1 + w_{22} U_2 U_2 + \cdots + w_{2N} U_2 U_N \} \\ & \vdots \\ & + \{ w_{N1} U_N U_1 + w_{N2} U_N U_2 + \cdots + w_{NN} U_N U_N \}] \\ & + \sum_{j=1}^N h_j U_j \end{aligned} \quad (\text{A.3})$$

$$\begin{aligned} \Delta E = & E_{NN}(n) - E_{NN}(n-1) \\ = & -\frac{1}{2} [\{ w_{1j} U_1(n) U_j(n) - w_{1j} U_1(n-1) U_j(n-1) \} \\ & + \{ w_{2j} U_2(n) U_j(n) - w_{2j} U_2(n-1) U_j(n-1) \} \\ & \vdots \\ & + \{ w_{Nj} U_N(n) U_j(n) - w_{Nj} U_N(n-1) U_j(n-1) \} \\ & + \{ w_{j1} U_j(n) U_1(n) - w_{j1} U_j(n-1) U_1(n-1) \} \end{aligned}$$

$$\begin{aligned}
& + \{w_{j2}U_j(n)U_2(n) - w_{j2}U_j(n-1)U_2(n-1)\} \\
& \quad \vdots \\
& + \{w_{jN}U_j(n)U_N(n) - w_{jN}U_j(n-1)U_N(n-1)\} \\
& + h_j\{U_j(n) - U_j(n-1)\}
\end{aligned} \tag{A.4}$$

更に, $w_{j,k} = w_{k,j}$, $U_{k(\neq j)}(n) = U_{k(\neq j)}(n-1)$ であることから, 式 (A.4) は式 (A.5) に示すように表せる.

$$\begin{aligned}
\Delta E &= -[w_{j1}U_1(n-1)\{U_j(n) - U_j(n-1)\} \\
& + w_{j2}U_2(n-1)\{U_j(n) - U_j(n-1)\} \\
& \quad \vdots \\
& + w_{jN}U_N(n-1)\{U_j(n) - U_j(n-1)\} \\
& + h_j\{U_j(n) - U_j(n-1)\} \\
&= -\{U_j(n) - U_j(n-1)\} \sum_{k=1}^N w_{jk}U_k(n-1) \\
& \quad + \{U_j(n) - U_j(n-1)\}h_j \\
&= -\{U_j(n) - U_j(n-1)\} \left\{ \sum_{k=1}^N w_{jk}U_k(n-1) - h_j \right\}
\end{aligned} \tag{A.5}$$

n 回目の更新における第 j ユニットの出力値 $U_j(n)$ を, 式 (A.1) に従って決定した結果, $U_j(n) \neq U_j(n-1)$ であるとき, ΔE には次の二つのケースがある.

① $U_j(n-1) = 0, U_j(n) = 1$

$\sum_{k=1}^N w_{jk}U_k(n-1) - h_j > 0$ の場合であるから, $\Delta E = -(1-0) \times \text{正} < 0$, すなわち, エネルギーは減少する.

② $U_j(n-1) = 1, U_j(n) = 0$

$\sum_{k=1}^N w_{jk}U_k(n-1) - h_j \leq 0$ の場合であるから, $\Delta E = -(0-1) \times \text{非正} \leq 0$, すなわち, エネルギーは増加しない.

故にエネルギー関数 E_{NN} は決して増加しない.

Q.E.D.

ユニットの動作は非同期的 (逐次ダイナミックス) であり, 同時には一つのユニットだけが動作する. Hopfield らは, 相互作用するスピン系の類推から, このニュー

ラルネットワークに対し、式 (A.2) に示したエネルギー E_{NN} を定義した。相互作用するスピンのモデルは、磁気の現象論に頻繁に登場する。スピンが自由に回転できず、上か下を向くだけの自由度をもつスピンモデルは、発火と静止の二つの状態をもつニューラルネットワークと類似している。この種のモデルは、イジングモデルとして知られている。現実の磁性体と比べると大幅に単純化されているが、統計力学の手段を用いて分析するのに適している。

零温度の非同期ダイナミクスでは、同時に一つのユニットだけが動作し、その度に E_{NN} は単調減少する。Hopfield 型ニューラルネットワークを非同期的に動作させるのは、この性質のためである。このような単調減少関数は、力学系のリアプノフ関数と呼ばれる。この関数の極小値は、アトラクターを表す。系は、アトラクターの一つに向かって移動し、最終的に、それに捕らえられる。提案した境界線検出手法では、この性質を利用して、固定発火ユニットとして特定のユニットを発火状態に維持することで、このアトラクターの一つを指定し、大局的な境界線を検出した。また、最適解をニューラルネットワークが記憶しているパターンであると捉えれば、Hopfield 型ニューラルネットワークによる最適化の過程は、ニューラルネットワークが記憶したパターンを想起する過程と考えることもできる。

付録 B

階層型ニューラルネットワークにおける学習

認識・学習・推論などを行うための階層型ニューラルネットワークに関する研究が行われている。ニューラルネットワークの機能は、ネットワーク構造、ユニット間の結合強度、ユニットで行われる演算によって決定づけられる。本付録では、最も基本的なユニットである MP(McCulloch & Pitts) モデルと、それを用いた Perceptron について述べる。そして、隠れ層における学習則として提案された誤差逆伝播 (Error-Back-Propagation) アルゴリズム [37] について述べる。

B.1 Perceptron

ニューロンは、静止状態と興奮状態という二つの状態のどちらかをとる。McCulloch と Pitts は、この性質に基づいて、図 B.1に示す多入力-1 出力のユニットとして MP モデルを 1943 年に提案した。

$(x_1, x_2, \dots, x_n)$ は入力を、 $(w_1, w_2, \dots, w_n)$ は結合荷重を、 h はしきい値を、そして y はユニットの出力を表している。MP モデルで行われる演算を式 (B.1) に示す。

$$y = f\left(\sum_{i=1}^n w_i x_i - h\right), \quad f(u) = \begin{cases} 1 & u > 0 \\ 0 & u \leq 0 \end{cases} \quad (\text{B.1})$$

McCulloch と Pitts は、図 B.2に示す MP モデルを用いて論理演算が可能であることを示した。

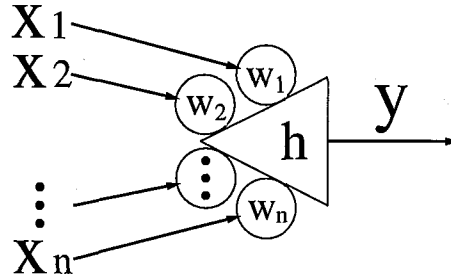


図 B.1 MP モデル

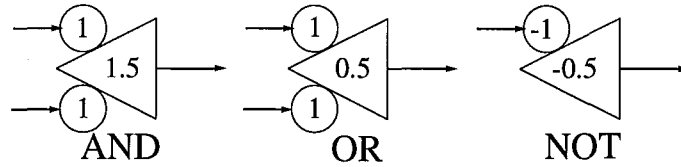


図 B.2 論理演算用 MP モデル

Rosenblatt は 1950 年代末に、学習するパターン認識装置 Perceptron を提案した。Perceptron は図 B.3 に示す三つの層からなり、入力層には 0 と 1 からなる画像パターンを呈示し、隠れ層、出力層の各ユニットは、B.1 で述べた MP モデルに従って動作させる。入力層から隠れ層への結合荷重はランダムに決め、変化させない。第 i 隠れユニットから出力ユニットへの結合荷重 w_{pi} および出力ユニットのしきい値 h_p を、学習によって変化させる。学習は、出力ユニットの出力 y_p が、予め与える 0 または 1 の教師信号 d に一致するように行う。

x_{pi} を第 i 中間ユニットの出力とし、隠れ層のユニット数を 3 と仮定すれば、中間ユニットの出力値の組み合わせによる 3 次元のパターンは図 B.4 の●と○で示す立方体の 8 個の頂点にそれぞれ対応する。 y_p は式 (B.2) で表される。

$$y_p = f\left(\sum_{i=1}^3 w_{pi} x_{pi} - h_p\right) \quad (\text{B.2})$$

y_p は、それぞれのパターンに対応する頂点が、式 (B.3) で表される平面 S のどちら側にあるかによって決まる。

$$\sum_{i=1}^3 w_{pi} x_{pi} = h_p \quad (\text{B.3})$$

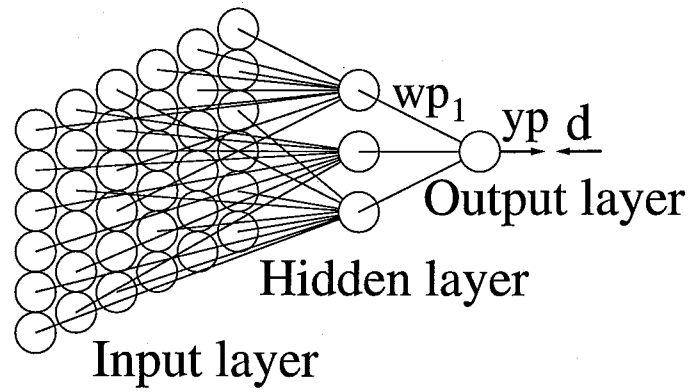


図 B.3 Perceptron の構造

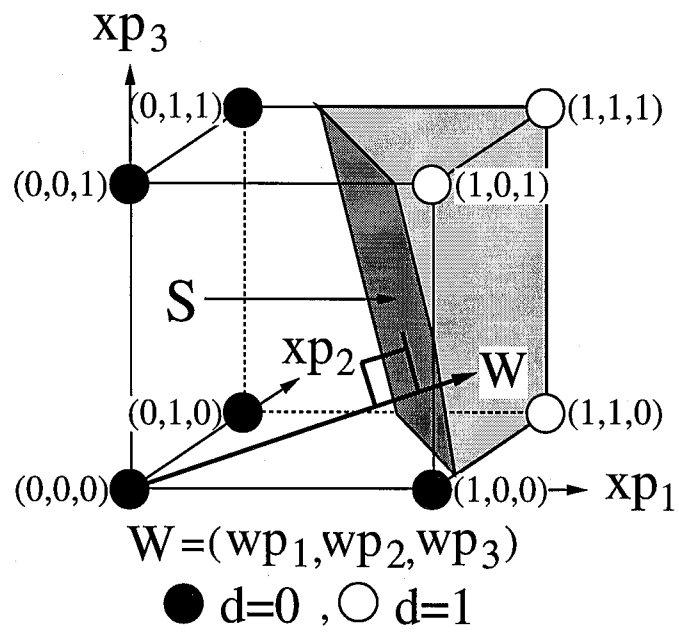


図 B.4 線形分離可能な集合

従って Perceptron の学習は、図 B.4に示す平面 S が、与えられた●の頂点と○の頂点を分離するように、 w_{pi} および h_p を変化させることである。具体的には、まず常に 1 を出力する第 4 隠れユニットを新たに仮定し、しきい値 h_p を $-w_{p4}x_{p4}$ で置き換えて考え、式 (B.2) を式 (B.4) に書き換える。

$$y_p = f\left(\sum_{i=1}^4 w_{pi}x_{pi}\right) \quad (B.4)$$

そして、入力パターンが与えられるごとに結合荷重を、式 (B.5) に示す Δw_{pi} だけ変化する。

$$\Delta w_{pi} = (d - y_p)x_{pi} \quad (i = 1, \dots, 4) \quad (B.5)$$

一般に、図 B.4の●と○で示したように、 n 次元の空間において、ある $n-1$ 次元超平面 S で二つに分離できるような点の集合を、線形分離可能な集合と呼ぶ。

Perceptron は、学習の収束が保証されているが、認識能力は、入力層に与えられたパターンが隠れ層において線形分離可能になるか否かに依存している。

B.2 誤差逆伝播アルゴリズム

Perceptron のように、入力層から隠れ層への結合荷重をランダムな値に設定する場合、入力層に与えられたパターンが隠れ層において線形分離可能になるために必要な隠れユニット数は非常に大きくなる。そこで、1980 年代半ばに、Rumelhart らは、出力層での誤差を利用して、隠れ層における結合荷重を変化させる誤差逆伝播 (Error-Back-Propagation) アルゴリズムを提案した。誤差逆伝播アルゴリズムでは、式 (B.6) に示す出力誤差 2 乗和を最急降下法によって最小化する。

$$E = \frac{1}{2} \sum_{I, \Pi} \sum_{n=0}^{N-1} (O_n - D_n)^2 \quad (B.6)$$

O_n : 第 n 出力ユニット ($n \in 0, 1, 2, \dots, N-1$) の出力。

w_{nm} : 第 m 隠れユニットから第 n 出力ユニットへの結合荷重。

D_n : 第 n 出力ユニットに対する教師信号。

第 m 隠れユニットの出力 O_m に関する E の勾配 $\partial E / \partial O_m$ は、ユニットの入出力関数に微分可能な関数 $f(u)$ を用いると、微分の連鎖法則を適用して式 (B.7) に示すように求まる。

$$\frac{\partial E}{\partial O_m} = \sum_{n=0}^{N-1} \frac{\partial E}{\partial O_n} \cdot \frac{\partial O_n}{\partial u_n} \cdot \frac{\partial u_n}{\partial O_m}$$

$$= \sum_{n=0}^{N-1} \frac{\partial E}{\partial O_n} f'(u_n) w_{nm} \quad (\text{B.7})$$

そして、最急降下学習における結合荷重の修正量 Δw は式 (B.8) から式 (B.10) に示すように求まる。

$$\Delta w_{ml} = -\eta \frac{\partial E}{\partial w_{ml}} \quad (\text{B.8})$$

$$= -\eta \frac{\partial E}{\partial O_m} \cdot \frac{\partial O_m}{\partial u_m} \cdot \frac{\partial u_m}{\partial w_{ml}} \quad (\text{B.9})$$

$$= -\eta \frac{\partial E}{\partial O_m} f'(u_m) O_l \quad (\text{B.10})$$

このアルゴリズムは、4層以上の場合にも同様に適用でき、出力誤差が結合行列を通じて逆向きに伝播していくため、誤差逆伝播アルゴリズムと呼ばれる。ユニットの入出力関数 $f(u)$ として式 (B.11) に示すロジスティック関数を用いると、その微分は、式 (B.12) に示すように簡単に求まる。

$$f(u) = \frac{1}{1 + e^{-u}} \quad (\text{B.11})$$

$$f'(u) = \frac{e^{-u}}{(1 + e^{-u})^2} = \{1 - f(u)\}f(u) \quad (\text{B.12})$$

6章で述べた手法の場合、隠れ層は1層、出力ユニットは一つであるから、結合荷重の修正量 Δw は式 (B.13) および式 (B.14) に示すように求まる。

$$\begin{aligned} \Delta w_m(t) &= -\eta \frac{\partial E}{\partial w_m} + \theta \cdot \Delta w_m(t-1) \\ &= -\eta \frac{\partial E}{\partial O_n} \cdot \frac{\partial O_n}{\partial w_m} + \theta \cdot \Delta w_m(t-1) \\ &= -\eta \sum_{I,II} \{(O_n - D) \cdot f'(u_n) \cdot O_m\} + \theta \cdot \Delta w_m(t-1) \\ &= -\eta \sum_{I,II} \{(O_n - D)(1 - O_n) \cdot O_n \cdot O_m\} \\ &\quad + \theta \cdot \Delta w_m(t-1) \end{aligned} \quad (\text{B.13})$$

$$\begin{aligned} \Delta w_{ml}(t) &= -\eta \frac{\partial E}{\partial w_{ml}} + \theta \cdot \Delta w_{ml}(t-1) \\ &= -\eta \frac{\partial E}{\partial O_m} \cdot \frac{\partial O_m}{\partial w_{ml}} + \theta \cdot \Delta w_{ml}(t-1) \\ &= -\eta \sum_{I,II} \{(O_n - D) f'(u_n) \cdot w_{nm} \cdot f'(u_m) \cdot O_l\} \\ &\quad + \theta \cdot \Delta w_{ml}(t-1) \\ &= -\eta \sum_{I,II} \{(O_n - D)(1 - O_n) O_n \cdot w_m(1 - O_m) O_m \cdot O_l\} \\ &\quad + \theta \cdot \Delta w_{ml}(t-1) \end{aligned} \quad (\text{B.14})$$

アナログニューラルネットワークの出力は入力に対して連続的に変化するために、学習の結果得られるニューラルネットワークの入出力関数は、学習データを滑らかに補間したものになる。これは、ニューラルネットワークの汎化 (generalization) 能力と呼ばれる。例えば、3層ニューラルネットワークの隠れユニット数を十分多くすれば、学習用入力データに対しては正しい出力をするようになる。しかし、隠れユニット数が多過ぎると、結合荷重が大きくなり、学習用入力データに近い他の入力データに対して、大きくかけ離れた値を出力する場合がある。そこで最近では、情報量基準を用いて隠れユニット数を決定する手法が提案されている [39]。

謝 辞

本研究を進めるに当たり，終始熱心な御指導を賜りました長橋宏助教授に心から感謝いたします。貴重な御意見，御討論をいただきました安居院猛名誉教授，長尾智晴助教授，そして永江孝規助手をはじめ，長橋研究室の皆様心から御礼を申し上げます。

参 考 文 献

- [1] 栄藤 稔, “MPEG4 の標準化動向,” 画電学誌, vol.25, no.3, pp.223–228, 1996.
- [2] 白井 良明, “コンピュータビジョン,” pp.25–26, 昭晃堂, 1980.
- [3] B. J. Schachter, A. L. Steven, W. Zucker, and A. Rosenfeld, “An Application of Relaxation Methods to Edge Reinforcement,” IEEE Trans. Syst., Man & Cybern., vol.SMC-7, no.11, pp.813–816, 1977.
- [4] 本郷 節之, 川人 光男, 乾 敏郎, 三宅 誠, “エネルギー学習機能をもつ局所並列確率アルゴリズムによる輪郭線抽出,” 信学論 (D-II), vol.J74-D-II, no.3, pp.348–356, March 1991.
- [5] D. H. Ballard, “Generalizing the Hough Transform to Detect Arbitrary Shapes,” Pattern Recognition, vol.13, no.2, pp.111–122, 1981.
- [6] F. Cheevasuvit, H. Maitre, and D. Vidal-Madjar, “A robust method for picture segmentation based on split-and-merge procedure,” Comput. Vis. Graph. Image process., vol.34, pp.268–281, 1986.
- [7] T. Poggio, V. Torre, and C. Koch, “Computational vision and regularization theory,” Nature, vol.317, pp.314–319, Sept. 1985.
- [8] D. Geman, S. Geman, C. Graffigne, and P. Dong, “Boundary detection by constrained optimization,” IEEE Trans., Pattern Anal. Machine Intell., vol.PAMI-12, no.7, pp.609–628, 1990.
- [9] M. Kass, A. Witkin, and D. Terzopoulos, “Snakes : Active Contour Models,” IJCV, vol.1, no.3, pp.321–331, 1988.
- [10] 坂上 勝彦, 山本 和彦, “動的な網のモデル Active Net とその領域抽出への応用,” テレビ誌, vol.45, no.10, pp.1155–1163, Oct. 1991.

- [11] 荒木 昭一, 横矢 直和, 岩佐 英彦, 竹村 治雄, “複数物体の抽出を目的とした交差判定により分裂する動的輪郭モデル,” 信学論 (D-II), vol.J79-D-II, no.10, pp.1704–1711, Oct. 1996.
- [12] 吉野 和芳, 守田 了, 川嶋 稔夫, 青木 由直, “分裂アクティブネットによる複数物体の抽出,” 信学春季全大, 分冊 7, no.D-595, p.356, March 1993.
- [13] T. Pavlidis and Y. Liow, “Integrating region growing and edge detection,” IEEE Trans., Pattern Anal. Machine Intell., vol.PAMI-12, no.3, pp.225–233, 1990.
- [14] 上坂 吉則, “ニューロコンピューティングの数学的基礎,” 近代科学社, pp.67–76, 1993.
- [15] J. J. Hopfield and D. W. Tank, “Neural computation of decisions in optimization problems,” Biol. Cybern., vol.52, pp.141–152, 1985.
- [16] D. H. Ackley et al., “A learning algorithm for Boltzmann Machines,” Cognitive Science, vol.9, no.1, pp.147–169, 1985.
- [17] “特集 遺伝的アルゴリズム,” 計測と制御, vol.32, no.1, Jan. 1993.
- [18] L. Herskovits, T. O. Binford, “On boundary detection,” MIT Project MAC AI Memo, no.183, 1970.
- [19] U. Ramer, “Extraction of line structures from photographs of curved objects,” CGIP, vol.4, pp.81–103, 1975.
- [20] G. P. Ashker, J. W. Modestino, “The contour extraction problem with biomedical application,” CGIP, vol.7, pp.331–355, 1978.
- [21] R. Nevatia, K. R. Babu, “Linear feature extraction and description,” 6IJCAI, pp.639–641, 1979.
- [22] S. W. Zucker, “Region growing: Childhood and adolescence,” CGIP, vol.5, no.3, pp.382–399, 1976.
- [23] S. L. Horowitz, T. Pavlidis, “A graphtheoretic approach to picture processing,” CGIP, vol.7, no.2, pp.282–291, 1978.

- [24] S. Tanimoto, T. Pavlidis, "A hierarchical data structure for picture processing," CGIP, vol.4, no.2, pp.104-119, 1975.
- [25] 安居院, 中島, 浜, "特徴ベクトルの外積を用いたリモートセンシング画像の領域分割," 画電学誌, vol.8, no.3, pp.196-204, 1979.
- [26] C. K. Chow, T. Kaneko, "Automatic boundary detection of the left ventricle from cineangiograms," Comput. Biomed. Res., vol.5, pp.388-410, 1972.
- [27] K. Akita, H. Kuga, "Towards understanding color ocular fundus images," 6IJ-CAI, pp.7-12, 1979.
- [28] S. Peleg, "A new probabilistic relaxation scheme," IEEE Trans. Pattern Analysis and Machine Intelligence, vol.PAMI-2, no.4, pp.362-369, 1980.
- [29] T. Agui, H. Iwata, and H. Nagahashi, "Edge Trace of Images using Neural Networks," Proc. Intl. Conf. Neural Inf. Process., Seoul, South Korea, pp.922-927, 1994.
- [30] G. H. Pall and D. J. Hall, "ISODATA; An Interactive Method of Multivariate Data Analysis and Pattern Classification," Proc. IEEE Intl. Commun. Conf., pp.116-117, 1966.
- [31] "岩波 情報科学辞典," 岩波書店, pp.360-361, 1990.
- [32] デビッド・マー, 乾 敏郎, 安藤 広志訳, "ビジョン-視覚の計算理論と脳内表現," 第2章, 産業図書, 1987.
- [33] B. Olstad and A. H. Torp, "Encoding of a priori information in active contour models," IEEE Trans., Pattern Anal. Machine Intell., vol.PAMI-18, no.9, pp.863-872, 1996.
- [34] W. B. Thompson, "Combining motion and contrast for segmentation," IEEE Trans., Pattern Anal. Machine Intell., vol.PAMI-2, no.6, pp.543-549, 1980.
- [35] 安居院 猛, 長尾 智晴, 斉藤 智明, 中嶋 正之, "ブロック特徴を用いた歩行人物の抽出と追跡," テレビジョン学会誌, vol.45, no.10, pp.1213-1220, 1991.
- [36] T. Kohonen, "Self-organization and associative memory," Second Edition, Springer-Verlag, Berlin, 1987.

- [37] D.E. Rumelhart, G.E. Hinton, and R.J. Williams, "Learning representations by back-propagating errors," *Nature*, vol.323, no.9, pp.533–536, 1986.
- [38] 池田 和司, "追加学習の漸近論的解析," *信学論 (D-II)*, vol.J80-D-II, no.7, pp.1913–1918, July 1997.
- [39] N. Murata, S. Yoshizawa, and S. Amari, "Network Information Criterion ," *IEEE Trans., Neural Networks*, vol.5, no.6, pp.865–872, Nov. 1994.
- [40] 安居院 猛, 古久保 良隆, 長橋 宏, 長尾 智晴, "2段階神経回路網を用いた白黒階調静止画像からの顔領域の抽出," *画電学誌*, vol.23, no.3, pp.167–175, June 1994.
- [41] 榊永 慎哉, 長尾 智晴, "静止階調画像を対象とした人物の顔領域の自動抽出," *印刷誌*, vol.33, no.6, pp.377–382, June 1996.
- [42] 合原 一幸, "ニューロ・ファジィ・カオス," オーム社, pp.40–54, 1992.

研究業績

論文

- (1) 岩田 治幸, 長橋 宏, 安居院 猛, “ニューラルネットワークを用いたカラー画像における境界線検出,” 映像情報メディア学会誌, vol.51, no.1, pp.113-120, Jan. 1997.
- (2) 岩田 治幸, 長橋 宏, “ニューラルネットワークを用いたカラー画像における動的領域分割,” 電子情報通信学会論文誌 (D-II), vol.J80-D-II, no.11, pp.2995-3003, Nov. 1997.
- (3) 岩田 治幸, 長橋 宏, “ニューラルネットワークの学習とエネルギー極小化による動的領域抽出,” 映像情報メディア学会誌査読中.

国際会議

- (1) T. Agui, H. Iwata and H. Nagahashi, “Edge trace of images using neural networks,” Proceedings of International Conference on Neural Information Processing, ICONIP'94, Seoul, South Korea, vol.2, pp.922-927, Oct. 1994.
- (2) H. Iwata, T. Agui and H. Nagahashi, “Boundary detection of color images using neural networks,” IEEE, Proceedings of International Conference on Neural Networks, ICNN'95, Perth, Western Australia, Australia, vol.3, pp.1426-1431, Nov. 1995.
- (3) H. Iwata and H. Nagahashi, “Motion tracking of color image sequences using neural networks,” IS&T and SPIE, Proceedings of Visual Communications and

Image Processing, VCIP'97, San Jose, California, U. S. A., vol.3024, pp.200–210, Feb. 1997.

研究会

- (1) 岩田 治幸, 長橋 宏, “ニューラルネットワークを用いたカラー画像の動的領域分割,” 電子情報通信学会技術研究報告, PRMU96-31, vol.96, no.140, pp.41–48, June 1996.
- (2) 岩田 治幸, 長橋 宏, “ニューラルネットワークを用いたパターン認識による動的顔領域分割,” 電子情報通信学会技術研究報告, PRMU97-129, vol.97, no.386, pp.1–8, Nov. 1997.

全国大会

- (1) 安居院猛, 岩田治幸, 長橋宏, “ニューラルネットワークを用いた自然画像におけるエッジ追跡,” 電子情報通信学会春季大会講演論文集, 分冊 7, no.D-443, p.179, March 1994.
- (2) 安居院猛, 岩田治幸, 長橋宏, “ニューラルネットワークを用いたエッジ追跡による画像のセグメンテーション,” 電子情報通信学会秋季大会-ソサイエティ先行大会-講演論文集 情報・システム, no.D-12, pp.12–13, Sept. 1994.
- (3) 岩田治幸, 安居院猛, 長橋宏, “ニューラルネットワークを用いた画像中の境界線検出,” 日本印刷学会春期研究発表会講演予稿集, no.A-2, pp.5–8, June 1995.
- (4) 岩田治幸, 安居院猛, 長橋宏, “ニューラルネットワークを用いたカラー画像における境界線検出,” 電子情報通信学会情報・システムソサイエティ大会講演論文集, no.D-20, p.21, Sept. 1995.
- (5) 岩田治幸, 安居院猛, 長橋宏, “ニューラルネットワークを用いたカラー画像における動的領域分割,” 電子情報通信学会総合大会講演論文集, 分冊 2, no.D-344, p.132, March 1996.

- (6) 岩田治幸, 長橋宏, “ニューラルネットワークを用いたカラー動画像中の移動物体追跡,” 電子情報通信学会情報・システムソサイエティ大会講演論文集, no.D-409, p.412, Sept. 1996.
- (7) 岩田治幸, 長橋宏, “領域形状に基づくカラー画像の動的領域分割,” 電子情報通信学会総合大会講演論文集, 分冊 2, no.D-12-66, p.273, March 1997.
- (8) 岩田治幸, 長橋宏, “領域の競合によるカラー画像の動的領域分割,” 映像情報メディア学会年次大会講演予稿集, no.20-9, pp.250-251, July 1997.

目 次

2.1 エッジの分類	7
2.2 従来のエッジ強調による境界線検出処理	8
2.3 エッジの向き	11
2.4 SCID“カフェテラス”	13
2.5 Sobel	14
2.6 弛緩法によるエッジ強調結果（更新回数5回）	15
2.7 SCID“果物籠”	19
2.8 Splited	19
2.9 Grouped	19
2.10 Eliminated	19
2.11 SCID“果物籠”	20
2.12 実験結果	20
2.13 SCID“ポートレート”	22
2.14 指先部分	22
2.15 動的しきい値処理実験結果	22
2.16 クラスタのヒストグラム分布	23
2.17 標準画像 f6	25
2.18 図 2.17 のヒストグラム	25
2.19 しきい値 50	26
2.20 階調画像の処理結果（更新回数5回）	26
2.21 RGB空間ユークリッド距離画像とヒストグラム	27
2.22 カラー画像の処理結果（更新回数1回）	28
2.23 カラー画像の処理結果（更新回数2回）	29
2.24 カラー画像の処理結果（更新回数3回）	30
2.25 “果物籠”	33

2.26 $V(x, y)$	33
2.27 0 回	33
2.28 300 回	33
2.29 1000 回	33
3.1 処理領域と小ブロック	35
3.2 エッジ追跡の初期設定	35
3.3 エッジ追跡概念図	36
3.4 パターンマッチングフィルタ	37
3.5 追跡方向	37
3.6 エッジセグメントの色	38
3.7 エッジセグメント間を結ぶ直線の方法の定義	39
3.8 境界線検出概念図	40
3.9 移動概念図	43
3.10 SCID“ポートレート”	44
3.11 3 倍に拡大した指先部分	44
3.12 ネットワークの発火状態	45
3.13 弛緩法	46
3.14 提案手法	46
3.15 境界線検出結果 1	50
3.16 境界線検出結果 2	52
3.17 Snakes による処理結果	55
4.1 ラベルの変更と領域の変形過程の例	59
4.2 ピラミッド画像	60
4.3 動的領域抽出概念図	61
4.4 A と B の隣接画素とユニット	62
4.5 分裂領域と対処方法	66
4.6 SCID“果物籠”	67
4.7 SCID“ミュージシャン”	67
4.8 動的領域抽出の様子 (パパイア)	68
4.9 動的領域抽出の様子 (にんにく)	69
4.10 動的領域抽出の様子 (顔 1)	70
4.11 動的領域抽出の様子 (顔 2)	71

4.12 エネルギー関数値収束の様子	72
4.13 動的領域抽出の様子（低画質画像）	74
4.14 パラメータ γ の効果	75
4.15 “果物籠”	77
4.16 初期網と $L(x, y)$	77
4.17 10×35	77
4.18 8×35	77
4.19 レモン	77
4.20 微分フィルタリングとラベルに基づくエッジ強度算出との対比	78
4.21 領域の競合とゼロ交差検出	80
5.1 顔形状に関する仮定	82
5.2 凹部と $-f(i)\{S_i + T_i\}$ の値	83
5.3 動的領域抽出結果	84
5.4 複数シードによる抽出結果 (Cloths)	86
5.5 複数シードによる抽出結果 (A violin and a hand)	87
5.6 CCIR”Cheerleaders”	88
5.7 移動物体追跡の様子	89
5.8 移動物体追跡結果	90
6.1 A と B の隣接画素とユニット	94
6.2 階層型ニューラルネットワークの構造と入力信号	98
6.3 問題点 1	99
6.4 問題点 2	99
6.5 サンプル画像	101
6.6 学習用データセットの一例	102
6.7 抽出結果	103
6.8 顔領域抽出結果（標準画像）	104
6.9 顔領域抽出結果（低画質画像 1）	105
6.10 顔領域抽出結果（低画質画像 2）	106
B.1 MP モデル	115
B.2 論理演算用 MP モデル	115
B.3 Perceptron の構造	116

図 目 次	131
-------	-----

B.4 線形分離可能な集合	116
-------------------------	-----

表 目 次

1.1 研究で用いたハードウェアおよびソフトウェア	4
-------------------------------------	---