

画像解析を用いたジグソーパズルの組立

押田 至啓

Assembly Algorithm of Jigsaw Puzzle Using Image Analysis

Yoshihiro Oshida

Digital image processing and image analysis are widely used in various fields. In this paper, algorithm to assemble a jigsaw puzzle with image analysis is described. In this method, the contour of the puzzle piece is used for searching the fit piece. Piece fitting is achieved using the information on the projection of the piece contour, the distance between corners of the piece and the number of the picture elements between corners of the piece.

1. 緒言

近年、コンピュータの急速な進歩とともに大容量の画像データを取り扱うことが容易になり、画像解析処理技術は実用的な技術として広く用いられるようになってきている。コンピュータを用いたデジタル画像処理はデジタル信号処理¹⁾の1分野であるが、2次元データである画像情報を用いて、形状認識、パターンマッチング、文字認識等を行い、これを工学的、医学的分野で各種機器に組み込むなど、その応用分野は多岐にわたっている。この画像解析技術の基礎となっている技術は2値化法、2次元フィルタリング、輪郭抽出法、輪郭解析、パターン認識などがあり、従来より様々な方法が提案されている。²⁾

本論文では、2次元画像処理技術を統合的に組み合わせた処理の一応用として、ジグソーパズルの組み合うピースを検出し、ジグソーパズルを完成するためのアルゴリズムについて述べる。ジグソーパズルの組立においては表面に描かれた図の情報が大きな役割を果たすが、本方法では各ピースの輪郭のみに注目し、辺の凹凸等の形状の特徴と辺の長さから対応する辺を特定し、パズルの組立を行う。

2. 基礎となる画像解析方法

2.1 全体の処理の流れ

ジグソーパズル組立を行うための処理の全体的な流

れを図1に示す。

まず、スキャナによりジグソーパズルの各ピースを2次元の画像情報としてコンピュータに取り込む。本法では各ピースの形状情報のみを用いるため、後の処理の簡便さも考慮し、色分布情報のない裏面をデータとして取り込む。取り込んだ画像を2値化処理した後、局所方向符号を求めて輪郭（エッジ）抽出を行う。局所方向符号を用いて局所曲率符号を求め、符号化処理する。これにより、輪郭の曲率を符号化する。この符号化された輪郭データを用い、G-オペレーションによりピースの4つの角を検出し、角間距離を求めるとともに、角間の凹凸を検出する。ピースの4辺の凹凸の組み合わせにより各ピースをグループ分けし、分類されたピースの凹凸の組み合わせと、角間距離の比較により対応するピースを決定し、パズルを組み立てる。

2.2 2値化処理

本法では各ピースの輪郭情報のみを用い、その凹凸形

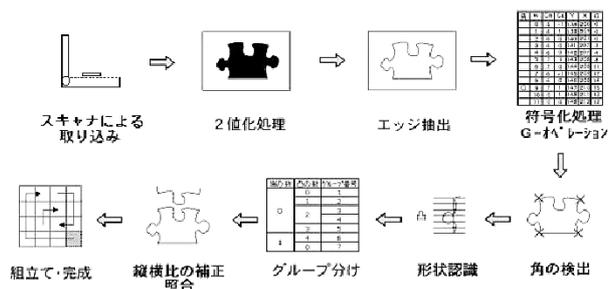


図1 全体の処理の流れ

状と、寸法を用いて対応するピースを特定する。このため入力した2次元画像データを、しきい値以上は1、しきい値以下は0とすることにより2値化する。

2.3 局所方向符号³⁾

2値化画像の輪郭のみを符号化する方法として、Freemanのチェーン符号を用いて局所方向符号 Cn_i を求める。これは2次元配列データとして量子化された隣接する2個の画素間の位置関係を符号化したものである。図2に示すように、図中の斜線で表した画素に隣接する画素は8個であるため、隣接する画素を量子化された8つの局所方向を用いて表す。この局所方向符号は図に示すとおり、右方向を0とし、そこから反時計回りに1から7までの数値を用いて表す。

2.4 局所曲率符号⁴⁾

格子状画素平面上に輪郭が連結しているときの3個の連続する画素による局所角は、局所方向が8方向に量子化されていることから、図3に示す5通りになる。

局所曲率符号の値を図4に示す。図中の斜線の画素の局所曲率符号は、●の画素を基準として斜線の画素に続く画素の位置により図のように符号化する。すなわち、角度が45°増すごとに1、2、3となる。また、連続した画素の連続する進行方向に対し左曲がり+、右曲

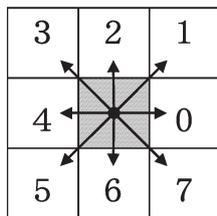


図2 局所方向符号

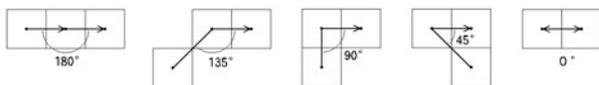


図3 量子化された局所角

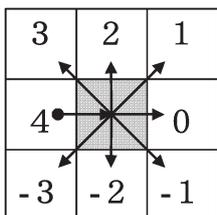


図4 局所曲率符号

がり-の符号になる。この関係を局所方向符号を用いて表せば、連続する画素の内*i*番目の画素の局所曲率符号 Ca_i は次式で表される。

$$Ca_i = \text{octmod}(Cn_i - Cn_{i-1} + 11) - 3 \quad (1)$$

ただし、 $\text{octmod}(x)$ は x を8で割った剰余を示す。

この局所曲率符号は、離散化された曲線の局所的な曲がり具合の特徴を表している。

2.5 G-オペレーション⁵⁾

連続した曲線の大域的な曲率を求めるためには、局所的な特徴を曲線に沿って平滑化する必要がある。この平滑化の方法としてGallusらにより提案されたのがG-オペレーションである。連続する画素の内*i*番目の画素のG-オペレーションは次式により定義される。

$$G_i = MCa_i + \sum_{k=1}^{M-1} (M-k)(Ca_{i-k} + Ca_{i+k}) \quad (2)$$

ただし、 M は平滑化に用いる曲線上の連続した画素の数である。

このG-オペレーションは離散化された画像データから得られる局所曲率を曲線に沿って平滑化を行った結果として得られる。曲線上を反時計回りに回るとき、曲線が左曲がりならば+の値、右曲がりならば-の値になる。また、曲線の曲率半径が小さいほど絶対値が大きな値となる。すなわち、この値の絶対値が大きいところが曲線の角の部分であると考えられることができる。

3. 処理方法

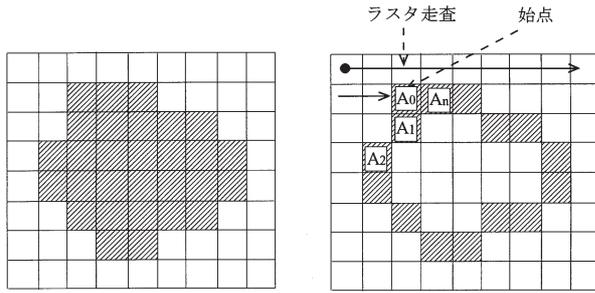
3.1 2値化処理

本方法では取り込んだ画像がピース裏面であるため、画像データはピース部分ではほぼ一樣な値となっている。また、周辺部分は異なった一定値となっている。このため代表的なピースの輝度データ値の全画面に対するヒストグラムを作成し、ピース裏面と周辺それぞれに対応する輝度の中間値をしきい値とする。

3.2 輪郭の抽出と符号化処理

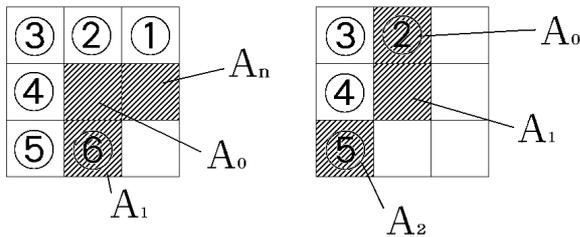
2値化した画像データから局所方向符号を求め、これを用いて局所曲率符号を求める。

図5 (a) に示すような2値化画像データの場合、まず、図5 (b) に示すように画像左上の画素から右方向にラスタ操作していき、最初にピース面に当たる画素を始点と定める。この始点を画素 A_0 とし、輪郭線を構成



(a) 2 値化画像 (b) ラスタ走査

図5 輪郭の抽出



(a) A₀での処理 (b) A₁での処理

図6 輪郭の抽出と符号化処理

する画素の連なりをA₁, A₂, ..., A_nとする。

画素A₀に連なる画素A₁は、図6 (a)に示すようにA₀を中心にして右上の画素から反時計回りに隣接する画素を参照し、ピース面と認識した画素が連なる輪郭線の画素A₁となる。このとき画素A₀における局所方向符号は図2の関係より6となる。次に図6 (b)に示すように画素A₁を基準にして同様の処理を行うことにより、連なる輪郭線の画素A₂を求めることができ、画素A₁における局所方向符号5を得る。この操作を始点の画素A₀に到達するまで繰り返せば、ピースの輪郭線上の一連の局所方向符号を求めることができる。また、輪郭線を構成している画素を認識することができる。

得られた一連の局所方向符号を用いて、式(1)により、局所曲率符号を求める。たとえば図6 (b)の画素A₁における局所曲率符号は

$$Ca_1 = \text{octmod}(5 - 6 + 11) - 3 = -1 \quad (3)$$

となる。この演算を輪郭線上の全ての画素について行えば、輪郭線上の一連の局所曲率符号の値を求めることができる。ただし、画素A₀における局所曲率符号は輪郭線の最後尾の画素A_nの局所方向符号を用いて求める。

3.3 G-オペレーションおよび角の検出

得られた輪郭線上の一連の局所曲率符号の値を用い、式(2)によりG-オペレーションの値を求める。ただ

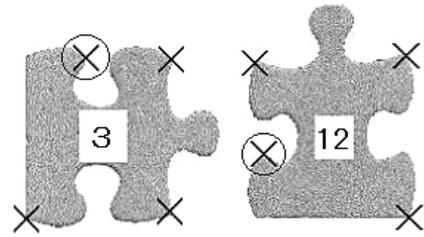


図7 角の誤認

し、輪郭線の始点、終点付近では連続した輪郭線として処理しなければならないため添字の値を調整しなければならない。すなわち、 $i-k$ の値が負の数になる時には輪郭線上の全画素数 n を加えて $i-k+n$ とする。また、 $i+k$ の値が n を超えるときは n を引いて $i+k-n$ とする。

このようにして得られた輪郭線上のG-オペレーションの値が大きい画素が角の部分に相当すると考えられる。したがって、G-オペレーションの値の大きい順に4個の画素を、4角形ピースの4個の角と考えることができる。しかし、実際のピースでは図7に示すように、値の大きい点4点を×印で表せば、角以外の急峻な部分(○印)を角と誤認する場合がある。ただし、4角形ピースの場合、角間の直線距離はほぼ同程度の値になるため、しきい値を設けて、この値より近い点は除外することとする。

4個の角を検出することにより、角間直線距離と輪郭線に沿った角間画素数の値を得ることができる。

3.4 辺の形状認識とグループ分け

ピースの各辺は、凹か凸あるいは直線のいずれかになっている。組み合わせるピースを選択するためには、ピースの各辺の形状を認識しておかなければならない。本研究で用いた認識法を次に示す。

図8に示すように、連続する2つの角を直線で結び、これを等分した点の座標を求める。図では簡単のため2つの角を結んだ直線の方法をY軸としている。また、本

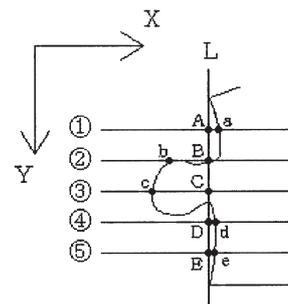


図8 辺の形状認識

表1 ピースの形状によるグループ分け

端の数	凸の数	ピースの形	グループ番号	端の数	凸の数	ピースの形	グループ番号
0	0		1	1	0		7
	1		2		1		8
	2		3			9	
			4			10	
	3		5			11	
4		6		12			
2	0		13	2	0		15
					1		16
	1		17		1		17
					2		18

法では6等分した点を用いた。図では直線Lが角を結んだ直線であり、直線①～⑤が6等分する線である。直線Lと直線①～⑤との交点をそれぞれA～Eとし、直線①～⑤と輪郭線との交点をそれぞれa～eとする。このとき直線①～⑤上の2つの交点の間の距離が最も大きい点（図ではC-c）での位置関係を調べる。図のようにcの方がピースより外側にあれば凸、内側にあれば凹、また、交点間の距離がほぼ0であれば直線であると認識することができる。

この操作を他の3辺についても行い、ピースの4辺の凹凸の形状を認識することができる。

次に、組み合わせるピースを探索する操作を容易にするためピースの形状によりピースをグループ分けする。ピースは4辺の凹凸、直線形状により表1に示すような18のグループに分類することができる。これにより、組立時に適合するピースをある程度絞り込むことができる。

3.5 パズルの組立

形状別に分けたグループの内、2つの辺が直線になるもの（グループ15～18）はパズル全体の四隅になるピ

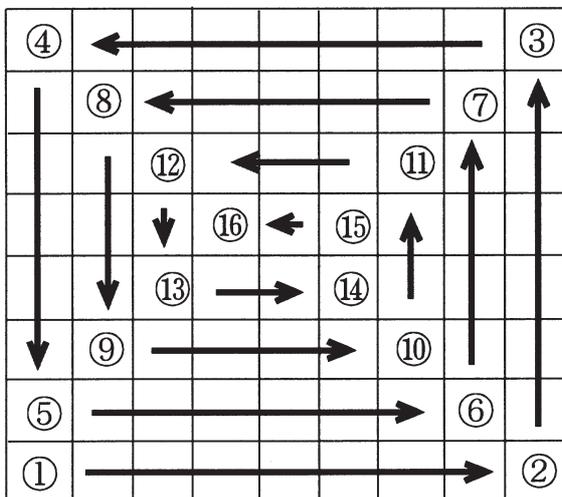


図9 ピースの組立手順

ースであり、4枚存在する。この内の任意のピースを1枚選び、これを組立の基点とする。図9に示す①の位置にこのピースを設置し、組立の進行方向（図中の矢印）にある辺、すなわちこのピースの右の辺に組み合う辺を持ったピースを選ぶ。このとき候補となるピースは、外周であることから、1辺が直線であるグループ7～14、あるいは2辺が直線になるグループ15～18の中に含まれる。順次右方向に組み合うピースを選んで行く。グループ15～18に含まれるピースが選ばれたとき、パズルの隅（図9②の位置）に到達しているため、対応を調べる辺を反時計回りに90°回転し、上の辺について適合するピースを選択する。

対応をとる辺を反時計回りに90°回転しながら、組み合わせるピースを選定する作業を繰り返し、図9⑤の位置まで到達した時点で外周の組立が終わる。

次に、⑤以降の内周のピースについては、進行方向の辺の適合に加えて、進行方向右側の辺とすでに選定を終わった辺との適合を調べ、組み合わせるピースをグループ1～6の内から選定する。この選定操作を順次螺旋状に内部に向かって繰り返し、パズルの中心（図9⑩）の位置に到達したとき組立は完了したことになる。

適合する辺を選定する基準は次の通りである。

- i) 対応する辺の凹凸形状の一致（凹と凸の組み合わせ）
- ii) 角間直線距離の一致
- iii) 輪郭に沿った角間画素数の一致

対応する2辺を、優先順位 i) ii) iii) で比較し、この基準を満たすものを適合するピースと選定する。

4. 処理結果

4.1 入力画像

組み立てるジグソーパズルは図10に示すような20ピース（5×4）のものを用いた。入力画像は図に示すジグソーパズルの裏面であり、ほぼ同じ輝度の信号として取り込まれる。この20枚のピースをそれぞれスキャナを用いて画像データ化し、コンピュータに取り込む。入力は各ピースごとに行い、取り込み時の方向はランダムとし、図中のピース番号がほぼ正立する方向に設置し画像を入力した。

なお、入力時のスキャナの解像度は200dpiであり、約0.13mmの寸法精度で取り込んでいる。各ピースの1辺の長さが約50mmであることから、形状を比較する精度は充分であると考えられる。

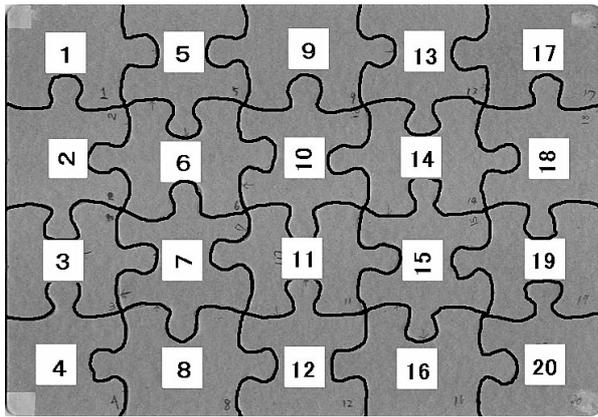


図10 ジグソーパズル (原型)

4. 2 2値化処理

入力画像と2値化処理後の画像の1例を図11に示す。

入力画像がほとんど輝度変化のない画像であることから、図に示すとおり正確に2値化されていることが分かる。

4. 3 G-オペレーションと角検出およびグループ分け

2値化画像から局所方向符号 C_n 、局所曲率符号 C_s 、画面上のX座標値、Y座標値およびG-オペレーションを求めた結果の一例を表2に示す。ただし、処理の妥当性



(a) 入力画像 (b) 2値化画像

図11 入力画像と2値化処理後の画像

表2 形状表の一例

角	w	Cn	Ca	Y	X	G
0	5	1	138	280	2	
1	4	-1	139	279	0	
2	4	0	139	278	0	
3	4	0	139	277	0	
4	4	0	139	276	0	
5	4	0	139	275	0	
6	4	0	139	274	0	
7	4	0	139	273	0	
8	4	0	139	272	1	
9	5	1	139	271	1	
10	4	-1	140	270	-1	
11	4	0	140	269	0	
12	4	0	140	268	0	
13	4	0	140	267	1	
14	4	0	140	266	1	
15	4	0	140	265	1	
16	4	0	140	264	1	
17	5	1	141	263	2	
18	4	-1	141	262	0	
19	4	0	141	261	1	
20	4	0	141	260	2	
21	4	0	141	259	4	
22	5	1	141	258	6	
23	4	-1	142	257	7	
24	4	0	142	256	9	
26	5	1	142	254	14	
27	4	-1	143	253	14	
28	5	1	143	252	17	
29	4	-1	144	251	18	
30	6	2	144	250	20	○
31	6	0	145	250	18	
32	7	1	146	250	16	
33	6	-1	147	251	14	
34	7	1	148	251	12	
35	6	-1	149	252	8	
36	6	0	150	252	6	
37	6	0	151	252	6	
38	7	1	152	252	4	
39	6	-1	153	253	1	
40	6	0	154	253	-1	
41	6	0	155	253	0	
42	7	1	156	253	0	
43	6	-1	157	254	-1	
44	6	0	158	254	-1	直線部
45	6	0	159	254	1	
46	7	1	160	254	0	
47	6	-1	161	255	-1	
48	6	0	162	255	-1	
49	6	0	163	255	0	
50	7	1	164	255	0	

を検証するため、実際の入力ピース画像ではなく、データ点数を少なくしたデータを用いた。表においてwは輪郭上の画素数番号である。この表より、○印の画素が角であることになる。

図10のピースNo.3に対する実際の処理結果出力を図12に示す。B[]は輪郭線上での画素のカウント値、x_1[], y_1[]はそれぞれ角の位置の全画面上におけるx座標、y座標の値である。また、凹凸は3, 4で述べた方法により検出した結果であり、0は直線、1は凹、2は凸を表している。グループはこの凹凸のパターンから得られたグループ番号であり、このピースはグループ9に属するという結果が得られ、正しく認識されていることが分か

```

*w(輪郭画素数)=2176
B[0]= 527
B[1]= 891
B[2]=1517
B[3]=2076

角の位置 k=0 x_1[0]=29.00 y_1[0]=35.00
角の位置 k=1 x_1[0]=28.00 y_1[0]=399.00
角の位置 k=2 x_1[0]=427.00 y_1[0]=387.00
角の位置 k=3 x_1[0]=428.00 y_1[0]=37.00

角の画素 角間直線距離 角間画素数 凹凸 グループ
527,      364,      364,      2,      9
891,      399,      626,      1,      9
1517,     350,      559,      0,      9
2076,     399,      627,      1,      9
    
```

図12 ピースNo.3の演算結果

表3 グループ分け結果

グループ番号	実際のピース番号	演算結果のピース番号
1	—	—
2	—	—
3	—	—
4	—	—
5	—	—
6	6,7,10,11,14,15	6,7,10,11,14,15
7	—	—
8	—	—
9	2,8,9,16,18	2,8,9,16,18
10	—	—
11	—	—
12	—	—
13	—	—
14	—	—
15	—	—
16	1,4	1,4
17	17,20	17,20
18	—	—

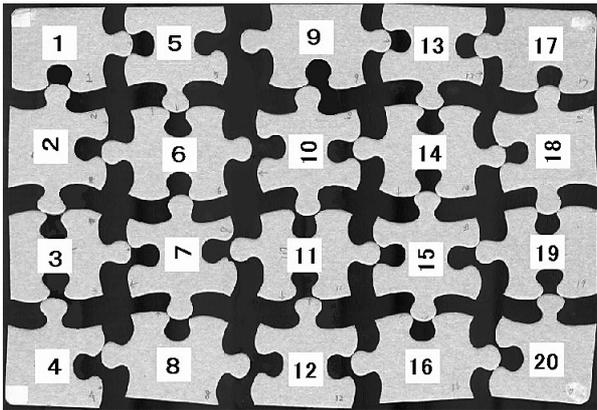


図13 組立結果

る。また、角間の直線距離と輪郭線に沿った画素数は辺の形状情報として凹凸、グループとともに記録し、次の組立処理に用いる。

同様の処理を全てのピースに対し行い、辺の凹凸の組み合わせからグループ分けを行った結果を表3に示す。表から分かる通り、全てのピースが正しくグループ分けされている。

この結果、本法により各ピースの形状は正しく認識され、その辺の特徴を用いて正しくグループ分けを行うことができることが検証された。

4.4 組立処理

3.5で述べた方法で適合するピースを選定した結果を図13に示す。図は、演算処理により出力されたピースの配置結果に従って実際のピースを並べ直した結果である。

元のパズルである図10と結果の図13を比較すれば、正しく組み合わせられていることが分かる。

以上の結果より、本法によりジグソーパズルを正確に組み立てることができることが分かった。

5. 結 言

画像解析処理の一応用として、ジグソーパズルを組み立てるアルゴリズムを示した。本法はジグソーパズル各ピースの形状のみに注目し、ピース各辺の凹凸情報と角間直線距離、角間画素数のみを用いて適合するピースを選定する方法である。市販のジグソーパズルに対し、この方法でピースの裏面画像のみを用いて正確に組み立てることができることを示した。ただし、今回用いたパズルは、20ピースのもので比較的規模の小さいパズルであった。さらに規模の大きいジグソーパズルを対象にすれば、辺情報の特徴が似通ったピースが多数存在することが予想される。このため、本法で適合する複数の候補のピースを選んだ後、ピースを1個に絞り込む最後の段階で表面の図画情報を用いる必要があると考えられる。すなわち、表面の画像の連続性や色分布情報等を入力し、輪郭情報と組み合わせることにより、多ピースのパズルにも適用することができると考えられる。

また、本法は処理を簡略化するために、各辺の凹凸の情報のみを用いており、凹凸の寸法的な情報は用いていない。適合する辺の選定に寸法情報を用いたパターンマッチング等を適用すれば、処理は煩雑になるものの多ピースのものにも有効であると考えられる。さらに、ジグソーパズルの組み立てのみでなく、対応する辺の探索等に应用することができると考えられる。

参考文献

- 1) 齋藤正男, デジタル画像処理 (1986), 東海大学出版会.
- 2) 鳥脇純一郎, 画像理解のためのデジタル画像処理 [I] (1987), 昭晃堂.
- 3) 森俊二, 坂倉柊子, 画像認識の基礎 [I] (1986), オーム社, p147.
- 4) 森俊二, 坂倉柊子, 画像認識の基礎 [I] (1986), オーム社, p148.
- 5) 森俊二, 坂倉柊子, 画像認識の基礎 [I] (1986), オーム社, p150