

計算機による屋外風景写真の認識

— 画像の領域分割と記述的モデルを用いた解析 —

大田友一 金出武雄 坂井利之

(京大・情報工)

1. まえがき

我々は、建物、木、空、道路、自動車を含む大学構内のカラー風景写真の認識を、物体の性質、物体間の関係等を記述的に表現したモデルを用いて領域解析法 (region analysis) により行っている。領域解析法では、入力データとして与えられるシーンを、物体の表面に対応する「意味のある領域」に分割し、その領域と物体との対応をつけることにより認識を行なう。入力画像データは、まず、明るさの情報のみに基づいて一様な性質を持つ数百個の領域に分割する。解析過程ではモデルに“最もよく”マッチングする解釈 (interpretation) を各領域に対して与えていき、同じ解釈を与えられた隣接する領域同志を必要ならマージするというようにして、region growing 的な手法により認識を行なう。

従来、region growing では、隣接する領域同志がマージ可能かどうかを判定する際の一つの制約条件を与えるものとしてセマンティックな情報を導入した bottom-up 的な処理が行われてきた。我々は、屋外風景についての知識を、その知識が本来持っている構造を保存したかたちで記述し、また、解析過程において、解析結果をデータ構造上に蓄積していくことにより、モデルが top-down 的に働くことを可能にした region growing システムを作成している。その基本的部分をインプリメントしたので、結果を報告する。

2. 領域解析法の考え方

領域解析の手法によりシーンの認識を行なう最初の試みは 1970年に Brice 等により行われた [1]。彼らは、ブロックのシーンを、まず、明るさの一様な領域に分割し、ヒューリスティック region growing により意味のある領域 (以後、意味領域と呼ぶ) にまとめた後、認識を行なった。しかし、屋外風景などの複雑なシーンでは、ヒューリスティックのみにより意味領域に分割することは困難である。

Yakimovski は、1973年、領域分割段階に認識処理を持ち込んだセマンティック region growing の方法により屋外風景写真の認識を行なった [2]。Tenenbaum 等も同様に、領域分割と認識を組合わせた IGS (Interpretation Guided Segmentation) と呼ぶ方法によりシーンの認識を行なっている [3]。しかし、たとえば「建物」の認識を考えた場合、次の2つの事に気がつく。

- ①「建物」では、形が重要な特徴となるが、彼らのセマンティック region growing の方法では、1個の領域の形としてしか形の情報の利用ができず、対象物体の形を表現することは原理的に不可能である。
- ②「建物」には「窓」という副次的な部品が付いているが (これを以後、object-subject の関係と呼ぶ)、「窓」の認識は「建物」という限定された世界の中で行なうのが有効である。しかし、セマンティック region growing では、このような取扱いはできない。

Yakimovski 等の方法では、領域間の境界が消去可能かどうかの判定条件に、「境界の強さ」などの他に「意味」という一つのパラメータを加えるかたちでセマンティックが導入されているにすぎず、region growing の bottom-up 的な性質をそのまま引き継

いた制御を行なっている。形や object-subobject の関係を認識処理に利用するには、region growing を、単に「一様領域の境界を消去して意味領域にまとめる」ものとしてではなく、「一様領域を入力としてモデルにマッチングする対象物を抽出し、認識を行なう処理であり、領域間の境界は認識の結果として消える」ものとしてとらえることが重要である。

我々は、本来 bottom-up な性格を持つ region growing に top-down な制御を導入した認識システムを構成することを考えた。top-down 制御を行なうためには、

- ①対象とする世界の知識が、その知識が本来持つ構造を反映した構造で表現されていること --- 記述的モデル
- ②解析プログラム自身が、解析処理の進行状況を把握していること --- 解析結果の記述の形成

が必要となる。解析過程は、図1に示すように、まず bottom-up にシーンの中から object 存在の争がかりをつかみ、次に top-down にその object の細部構造を明らかにするというかたちで進行する。

対象とする世界についての知識をいかに表現するかは、人工知能研究上、大きな問題である。屋外風景認識の問題では、モデルは、サブルーチンの集り[4]、グラフ表現[5]、種々の拘束条件の集り[3]、条件付き確率の集り[2]として表現されてきた。我々が採用しようとする記述的モデルの代表的なものとしては、セマンティックネットワークがある。

しかし、画像解析においては、使用できる知識として yes/no 的に絶対的なものはほとんどなく、リンクの有無だけで物体が持つべき性質を表現することは困難である。我々は、単純なリンクではなく、fuzzy な真理値を持つ述語により性質を表現する方法をとった。そして、セマンティックネットワークのノードに相当する部分を knowledge block と呼び、我々のモデルの基本単位とした。図2は、knowledge block 間の関係を概念的に示したものである。

各々の knowledge block は、「建物」という物体、「コンクリート」という物質、「灰色」という性質など、屋外風景の世界の一つの事柄に関する知識を含み、それらの事柄の関係を示す explicit または implicit なリンクで結ばれている。knowledge block による表現法の特徴の一つは、図2にあるような物体と物質との間の属性の引継ぎのような一般的な知識だけでなく、解析プログラムが処理の特定の局面で使う ad hoc な規則もモデル中に記述することができ、解析過程の柔軟な制御が可能となり、解の探索空間を小さくし、効率よい処理が行なえるようにできることである。

Minsky の frame system [6] の考え方からも分るように、top-down な制御方式をスマートに実現するには、モデル自身が認識処理の主導権をとって解析をすすめるという概念でシステムを構成することが有用である。我々のモデルもこの考え方で構成されており、いわば、knowledge directed なかたちで認識処理を行なう。

画像処理の面からみれば、物体の細部の特徴を正確に抽出し正しく記述するためには、画像データとして十分な解像度のものを用いなければならず、取扱うべき画像データの量は膨大なものとなる。従って、前処理として入力シーンを

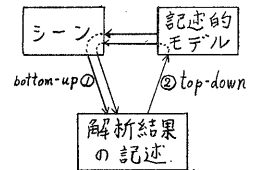


図1. 認識システムの動作

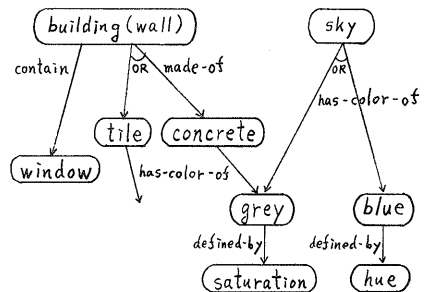


図2. knowledge block 間の関係

一様領域に分割するプログラムは、データ量を適当な程度に減少させると共に、画面中の構造的な情報の抽出を行ない解析処理をできるだけ容易にするという役割を持ち、充分に強力なものが要求される。

3. 前処理としての領域分割

3.1. Recursive Thresholding

前処理プログラムの役割は、赤(R)緑(G)青(B)のフィルタを通して得られた(R,G,B画面とよぶ)3枚の画像(各々256x256,各点5bit)を、明るさの情報と若干のヒューリスティクスを用いて一様な性質を持つ数百個の領域に分割することである。

ある物体の表面に対応する部分は、入力のR,G,B画面ではほぼ一様な性質を持ち、各画素をR,G,Bの三次元空間内にプロットすれば1つのクラスタを形成すると考えられる。このクラスタを検出し、もとの画面内で対応する画素を抽出すれば領域を切出すことができる。しかし、三次元空間内でのクラスタの検出は若干困難な問題であり、これをR,G,B空間のある軸上に写像して得られるヒストグラム(一般に複数個を用いる)上でクラスタ、すなわち、ピークを検出し、領域の切出しを行ない、切出された領域と残りの領域について同じことを繰り返して一様領域に分割するのがrecursive thresholdingの方法である。多種類のヒストグラムを用いて繰り返し閾値処理して一様な性質の領域を抽出するのは良く使われる有効な方法で、色彩画面の分割にはOhlander[7]によって利用されている。我々のアルゴリズムも基本的には同じであるが、色彩を表わす軸の選ぶ方と、最良の分割の選択に際しての画面上での評価を導入した。

3.2. 写像軸の選択

ヒストグラムを写像する軸の選ぶ方は興味ある問題である。Ohlanderは、R,G,B, intensity, hue, saturation にテレビ信号のY,I,Qを加えた9つのヒストグラムを用いている。しかし、hue, saturationのように除算を含む非線形なパラメータの場合には、ヒストグラム上に特異な極値を生じ真のピークの検出が難しい。

(3-1)式で表わせる線形変換を考えると、クラスタの分離能力が最大となるのはXの分散が最大となる場合である。よく知られているように、Xの分散の最大値は、R,G,B値の共分散行列の最大固有値に等しく、その時の固有ベクトルが、

$$X = \alpha R + \beta G + \gamma B \quad (\text{ただし } \alpha^2 + \beta^2 + \gamma^2 = 1) \quad \text{--- (3-1)}$$

分散最大軸の方向余弦(α, β, γ)を手える。固有値の大きい順から対応する軸を各々X,Y,Z軸とよぶことにし、それらの有効性を調べた。

2枚の異なるタイプのシーンa,b(aは図7と類似のシーン、bは図3に示した海辺のシーン)に対して、R,G,B,X,Y,Zの6軸を使う場合と、R,G,Bの3軸のみを使う場合について比較実験を行なった。いずれの場合にも、分割結果には大差なかった。各軸のヒストグラム上で検出されたピークが分割に使われた回数を表1に示した。X軸はほとんどの場合、intensityの方向を向いており、これに垂直なY軸は色合いを表現していると考えてよい。bの絵でY軸がよく選ばれているのは、bの絵が原色に富んでいるためと考えられる。X,Y,Zの3軸のみを使う場合も、ほぼ同様の結果を得た。ただし、計算時間は、R,G,B軸のみを使う場合の約2倍かかる。以上の実験結果から、ここで考えているようなシーンの場合には、R,G,Bの3軸のみを使えば充分であると考えて分割処理をした。しかし、色合いの豊富なシーンの場合には、hue, saturationなどのかわりとして、intensityの

軸に垂直な幾つかの軸に写像したヒストグラムを用いることが考えられる。

3.3. 分割アルゴリズムの詳細

図4に分割処理の概念図を示す。分割処理はrecursiveな処理を行なうために、分割可能かどうかをヒストグラムにより調査する領域を示すテンプレート(以後、単にテンプレートと呼ぶ)を積上げる画像スタックを用いて行なう。良好な分割結果を得られるよう、テクスチャ部の予備抽出、ヒストグラム作成時における小窓での走査、ピーク選択時における抽出領域評価のヒュリスティックスの導入などのアルゴリズムが工夫してある。



図3. 海辺の風景 b (水着女性とパラソル、6軸で分割)

ヒストグラム	R	G	B	X	Y	Z
aの絵 6軸	18	9	18	14	0	0
aの絵 3軸	28	15	20	/		
bの絵 6軸	25	34	37	21	7	0
bの絵 3軸	37	32	45	/		

表1. ヒストグラムの選択回数

テクスチャ部の予備抽出 水のようなテクスチャ部分に上述の分割処理をそのまま適用すると、多数の小さな領域に細分されてしまい、解析過程におけるそれらの取扱いが非常に複雑なものとなる。従って、テクスチャ部分はあらかじめ抽出しておき、その部分に対しては、あまり分割が行なわれなないようにする。

テクスチャ部の抽出は、次の方法で行なう。

- Green 画像を図5-aのラプラシアン演算子で微分し、閾値Tで2値化する。

$$T = \left(\frac{\text{微分値のヒストグラムのモード値}}{\text{モード値}} \right) + (\text{標準偏差}) \times 1.2 \quad \dots (3-2)$$

- 得られた2値化微分画像に図5-bの9x9の窓をあてはめ、9個の部分窓のうち8個以上に最低1個の1があれば、真中の点をテクスチャ部であるとする。

分割の閾値の選択 R, G, B

3つのヒストグラム上の山と谷をすべて検出し、その各々に対して(3-3)式でスコアを計算する。(図6参照)

$$\text{スコア} = \frac{2P - V_a - V_b}{2P} \times \frac{W - NP/P}{W} \quad \dots (3-3)$$

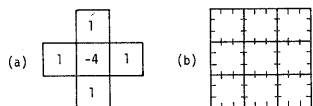


図5. ラプラシアン演算子と9x9の窓

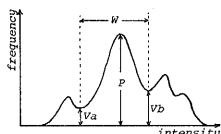


図6. ヒストグラムの山と谷の検出

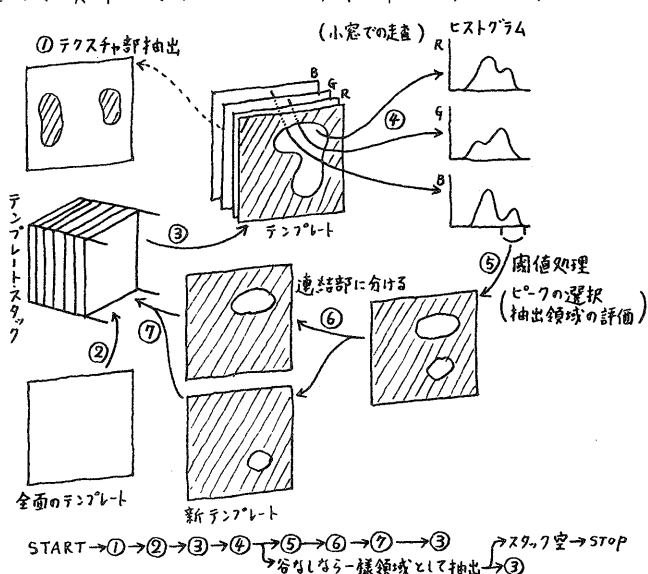


図4. Recursive Thresholding

ただし、 NP は山に含まれている点の数である。式の第1項は谷の相対的な深さを表わし、第2項は山の急峻さを表わす。谷の切れ込みが深く、山が細く鋭いほど大きなスコアになる。 R, G, B 3つのヒストグラムを通じてスコアが上位3つの山を選び分割の候補とする。スコアの良い山が、必ずしも元の画面上でまよりの良い領域に対応するとは限らないので、領域のまよりの良さに関するヒューリスティックな評価値を導入し、3つの候補の各々について実際に領域の抽出を行なって評価して最もまよりのある領域に対応する山を最良の閾値と決定する。

領域のまよりの良さを高速に評価する方法として、*looseness* と呼ぶ値を(3-4)式により定義した。

$$looseness = \left(\frac{F - O}{O} \right)^2 + \left(\frac{O - S}{O} \right)^2 \quad \text{ただし } F: \text{1回の fusion 処理後の総面積} \quad \dots (3-4)$$

O : 2値画面中に存在する領域の総面積
 S : 1回の shrinking 処理後の総面積

第1項は、2値画面中に存在する領域の平均的な細かさを表わし、第2項は、各領域が含む虫喰いの程度を表わしている。最小の *looseness* を持つものを最良の分割であるとする。

大領域に含まれる小さな領域の検出 ヒストグラムをとる領域が大きい場合には、ヒストグラムに存在すべき小さな谷が、主要な山に隠されて検出できなくなることがしばしばある。従って、面積が十分に大きな領域の場合には、全体のヒストグラムで谷がみつからなくても、 32×32 の小さな窓でその領域を走査して各々の窓でヒストグラムをとり、谷を見つけることを試みる。この方法により、大領域に埋もれている小さな領域も発見し、分割することができる。

図7のカラー風景写真を領域分割した結果が図8である(領域間の境界が印字してある)。

3.4. 領域分割結果の記述

領域分割した結果は、テーブルに整理して記述し解析過程で各種の性質が高速に計算できるようにする。一樣領域に分割された画面は、

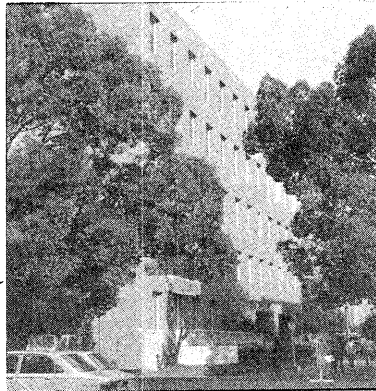


図7. 屋外風景写真

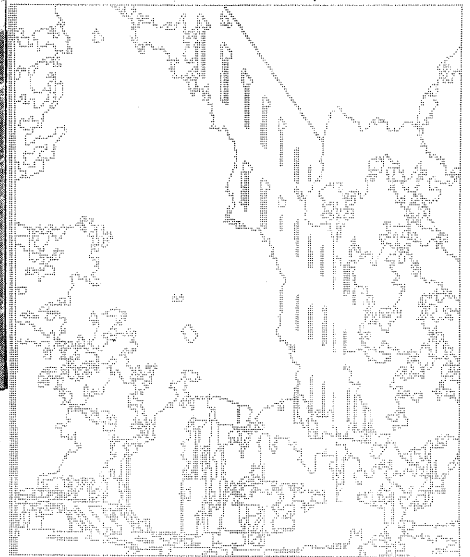


図8. 領域分割結果(一樣領域)

region, boundary (region

テーブル	記述内容
region table	面積、 R, G, B 平均値、テクスチャ度、周囲長、重心位置、画素位置分布の共分散行列、外接長方形の位置、周囲の boundary 番号、含む hole の数と番号
boundary table	長さ、(4方向) chain cord、両側の region 番号、両端の vertex 番号、あてはめた line 数と番号
vertex table	座標、出入する boundary の番号
hole table	周囲長、周囲の boundary の番号、外側の領域番号
line table	長さ、方向、原点からの垂直距離、両端の座標、あてはめた boundary の番号、あてはめ時の誤差

表2. 5テーブルの記述内容

の境界)、vertex (boundaryの端点)、hole (regionが他のregionに含まれる)、line (boundaryにあてはめた直線セグメント)の5項目で記述する。個々の region、boundary などには

通し番号をつけておく。項目別に5つのテーブルをつくる。テーブルの内容は、表2に示すとおりである。

4. 解析プログラムの構成

4.1. 制御構造とモデル

解析プログラムの役割は、前処理により得られた一様領域の各々に対して、モデルに基づいて解釈を与えながら region growing の手法により解析をすすめることである。解析は、まず、bottom-up 的に行ない、解析結果の記述をデータ構造上に蓄積していき、object の存在が抽出されると、今度は解析結果から feedback をかけるかたちで top-down にその object の細部構造を明らかにするというように動作する。top-down 制御を導入したことにより、次の利点が考えられる。

- 物体としての形状の情報が利用できる。
- 「建物」とその「窓」のような object-subobject の関係を解析過程で有効に利用し、効率よい処理を行なえる。
- 小さい領域の解釈を効率よく行なえる。

解析プログラムの構成を図9に示した。

データ構造上に解析結果の記述を含む data block の階層を形成することにより、解析の進行状況を記録していく。階層は、図9に示す7つのレベルからなり、解析プログラムの1つの処理は、階層中のあるレベルに存在する data block に作用し、それを別のレベルに移すか、上位のレベルの data block と関係づけるかするかたちで実行する。

解析処理に先立ち、前処理により分割された一様領域の各々に対して data block を作成し、homogeneous region レベルにおく。

Ask ステップ は、解析過程の最初に homogeneous region レベルから比較的大きな面積の領域 (の data block) を key-region として選り出し、key region レベルにおく。key-region は、シーン中に存在する object の主要な部分を反映し、その性質を充分に受け継ぎ、object 発見の手がかりとなると期待できる。各々の key-region について、モデル中に記述されたすべての object との間で、色、テクスチャなど、その領域単独で評価できる知識 (F-ASK rule として記述してある) を用いて ask-operation を行ない、各々の object に対してマッチングの確かさのスコアを計算しておく。

Select ステップ は、homogeneous region レベルまたは、key region レベルにある、まだ解釈を与え

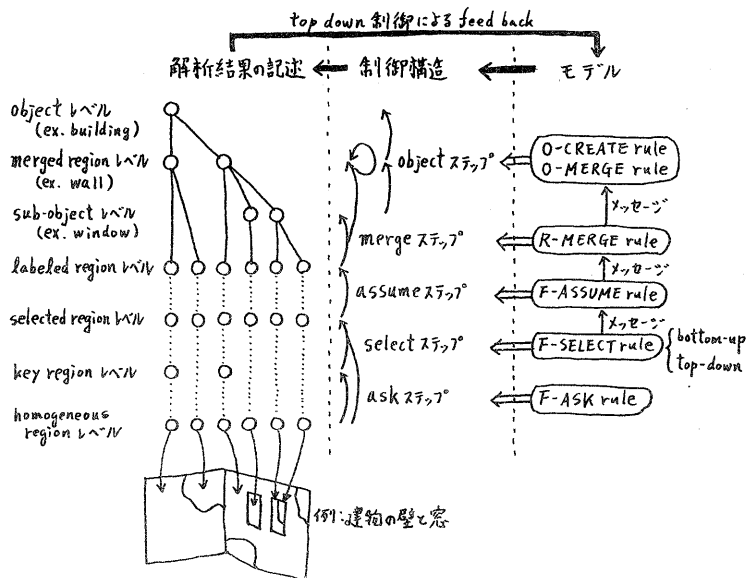


図9. 解析プログラムの構成


```

1 *OBJECT-1
2 (*SKY *TREE *ROAD *CAR *BUILDING)
3 *OBJECT-2
4 (*B-WINDOW *C-WINDOW *WHEEL)
5 *BASIC
6 (F-START ((ACT 100 (FL-SET (ADD 1000 (GET *RGN CONF-VAL))))
7 (ACT (GREATERP (*AREA *RGN) 2000)(FL-ADD 1000))
8 (ACT (SAME *OBJ (QUOTE *SKY))(FL-ADD 200))))
9 F-SELECT((ACT 100 (FL-SET (ADD 1000 (GET *RGN CONF-VAL))))
10 (ACT (GREATERP (*AREA *RGN) 1000)(FL-ADD 100))))
11 *SKY
12 (CLASS 1 F-PX 10
13 F-ASK ((GEN (*UPPER *RGN)(100 . 30))
14 (GEN (*SHINING *RGN)(100 . 20) R)
15 (GEN (M-OR (*BLUE *RGN)(*GREY *RGN))(100 . 20))
16 (GEN (*NO-TEXT *RGN)(100 . 70) R))
17 R-MERGE ((ACT 100 (FL-SET 2000)))
18 O-MERGE ((ACT 100 (PROG2 (FL-SET 2000)
19 (SETQ *S-MAIL OCCLUDED))))
20 *BUILDING
21 (L-ATB (OR *CONCRETE *TILE) CLASS 1 F-PX 20
22 OBJECT-2 (*B-WINDOW)
23 F-ASK ((GEN (*MIDDLE *RGN)(60 . 30) R)
24 (STR (*MANYHOLE *RGN)(100 . 20) R)
25 (STR (*MANYLINE *RGN)(40 . 20) R))
26 F-SELECT ((ACT (*HOLE *RGN *RGN2)
27 (PROG2 (FL-SET 3000)(SETQ *STOP 10)
28 (SETQ *S-MAIL WINDOW)(SKIP 10)))
29 (ACT { ... } { ... })
30 (ACT (*LARGE *RGN)
31 (FL-SET (ADD 1000 (*CONPART *RGN *RGN2))))
32 F-ASSUME((ACT (M-NOT (SAME *MODE (QUOTE *BUILDING)))
33 (SKIP 3))
34 (ACT (M-OR (SAME *R-MAIL WINDOW)(SAME *R-MAIL WALL))
35 (PROG2 (FL-SET 100)(SETQ *STOP 10)(SKIP 10)))
36 (ACT (*SHADOW *RGN *RGN2)(PROG2 (FL-MULT 15)(SKIP 10)))
37 (ACT 100 (SKIP 1))
38 (STR (M-IF (M-AND (SAME (GET *RGN2 LABEL)
39 (QUOTE *BUILDING))
40 (*CONTACT *RGN *RGN2))
41 (*SHADOW *RGN *RGN2)(60 . 40) R-R)
42 (ACT (M-IF (M-AND (SAME (GET *RGN2 LABEL)
43 (QUOTE *SKY))
44 (*CONTACT *RGN *RGN2))
45 (M-NOT (*LINEAR *RGN *RGN2))
46 (PROG2 (FL-SET 0)(SKIP 10)) R-R DEMON 100)
47 (STR (M-IF (SAME (GET *RGN2 LABEL)
48 (QUOTE *BUILDING))
49 (*SAMEPROP *RGN *RGN2))(80.20) R-R))
50 R-MERGE ((ACT 100 (FL-SET 2000)))
51 O-MERGE ((ACT 100 (FL-SET 2000)))
52 O-CREATE((ACT (GREATERP (*AREA *RGN) 2000)
53 (FL-SET 2000))))
54 *B-WINDOW
55 (CLASS 2
56 F-ASSUME ((ACT (M-AND (SAME *R-MAIL WINDOW)
57 (M-OR (*MANYLINE *RGN)
58 (*DARKER *RGN *RGN2)))
59 (PROG2 (FL-SET 100)
60 (SETQ *STOP 10)(SKIP 10)))
61 (ACT (M-AND (SAME *R-MAIL WALL) ..... ))
62 R-MERGE ((ACT 100 (FL-SET 2000)))
63 O-CREATE ((ACT 100 (FL-SET 2000))) )
64 *CONCRETE
65 (F-ASK ((GEN (M-AND (*BRIGHT *RGN)(*GREY *RGN))
66 (60 . 20) R)))
67 *TEXTURAL
68 (DF-PROP ((*X)(PR-TEXTR 20 40 90 100 *X)))
69 *BRIGHT
70 (DF-PROP ((*X)(FUZZY1 (PIC-DATA 1 *X) 30 50)))
71 *GREY
72 (DF-PROP ((*X)(PR-SAT -1 0 8 15 *X)))
73 *BLUE
74 (DF-PROP ((*X)(PR-HUE 20 30 32 48 *X)))
75 *HOLE
76 (DF-PROP ((*X *Y)(REL-DATA 9 *X *Y)))
77 *SHADOW
78 (DF-PROP ((*X *Y)(M-AND (*DARKER *X *Y)(BELOW *X *Y)
79 (*SAMECLR *X *Y))))

```

The fuzzy truth values and the probabilities in the model are multiplied by the factor 100.

図10. モデルの一部

の絆組内で矛盾なく動作させることができる。

4.2. モデルの構成

図10は屋外風景写真の解析に用いたモデルの一部である。モデルは、この様にLISP的な表現法で記述する。

はじめの2行はモデルに記述してあるobjectのknowledge blockのリストであり、次の2行は、同じくsub-object(objectの付随的な部品)のリストである。*SKY, *BUILDING, *B-WINDOW, *CONCRETE, *BLUEなどは、いずれもknowledge blockであり、それは、図2に例示した構造を形成している。knowledge block間のリンクは、explicitなものか、または、そのknowledge blockが含むrule中に記述されたimplicitなものになっている。図10において、各々のknowledge blockに続くリストが、そのblockが持つ知識の記述である。記述されている知識の種類はフラッグを付加して示してある。L-ATBは属性の引き継ぎを表わす属性リンクを示し、F-ASK, F-SELECT, F-ASSUMEなどは、各々、askステップ、selectステップ、assumeステップで使われるべきruleのリストを従えている。

ruleには、GEN, STR型のruleとACT型のruleがある。GEN, STR型のruleは、それが含む述語のfuzzyな真理値と、その知識の確かさを示す重み値とにより、スコアを更新する。ACT型のruleは、その中に記述された式により、任意の処理を行なう。ruleには、その評価方法を規定するフラッグが含まれており、そのruleが使っている知識が、色などのように他領域と独立に評価できるものか(R)、すでに解釈されている領域との間の関係を調べるものか(R-R)、それとも、将来に解釈される領域との間で評価されるべきものか(DEMON)を示している。

制御プログラムとモデルの間で情報の受渡しをするには、特別の意味を持たせたグローバル変数を使う。ある時点で考

慮すべき領域を限定するためには *RGN, *RGN2, rule間でのメッセージの授受を行なうためには *S-MAIL, *R-MAIL, 制御プログラムが分岐の探索を行なうのを制御するためには *STOPなどがある。

5. 解析過程の実際

図8の領域分割された画像を図10の一部を示したモデルにより解析した結果が図11である。この実験では、領域の色、テクスチャ度、位置、面積、簡単な形状、境界線の形状、領域同志の位置関係などの性質を解析に用いた。物体としての形状の性質はまだ用いていない。青い、黄色い、暗い、ジグザグなどの種々の概念を knowledge block により定義して使用した。

解析過程の最初に ask ステップにより、面積 300以上の領域が 30個、key-region として選ばれ、ask operationにより、そのうちの 23個が正しい解釈に対して最高のスコアを得た。しかし、シーンの方角は暗く、ask operation のみで正しい解釈を得るのは難しい。

解析プログラムは、select, assumeなどのステップからなる過程に入り、「空」、左側の「木」、右側の「木」、「建物」の上部、「道路」、「自動車」、「建物」の下部の順に object を発見し、解析を行なった。ask operation ではあいまいであった領域も、assume operation で領域間の関係についての知識を用いることで正しく解釈された。「道路」と「建物」の境界が混乱しているのは、領域分割時にうまく分割されなかった部分があるためである。

制御プログラムの各ステップが、モデルに基づいていかに解析をすすめるかを図12(図8の一部)を用いて例示しよう。A, B, C, D とマークされているのは key-region である。A はすでに「空」と解釈され、select ステップが bottom-up モードで B を選ぶ selected region レベルに置いたところであるとする。また、ask ステップで、B は MANYHOLE, MANYLINE などの性質を満たし、「建物」として十分に高いスコアを得ているとする。

Assume ステップでは、領域 B への各 object のマッチングのスコアを、各々の F-ASSUME rule を評価して更新する。*BUILDING の F-ASSUME rule 評価では、まず、31行目の ACT 型 rule の述語がみだされ、32~34行の 3 rule をスキップする(*MODE には、top-down モードで select ステップを行なった object の knowledge block がセットされる。従って、現在は NIL)。そして、35, 36, 37行目の順に rule を評価する。36行目の rule では、R-R のフラグが在るので、*RGN に B の data block を、*RGN2 には merged region レベルに存在するすべての data block を順々にセットし、述語を評価する。A が *RGN2 にセットされた時、A のラベルは *SKY であり、A と B は接しているので NOT LINEAR の性質がチェックされるが、これは明らかに満たされず、スコアは変化しない。結局、*BUILDING が最高のスコアを得て、B のラベルとして割付けられる。

領域 B の data block が top-down 処理の主導権を得ると、select ステップでは *BUILDING の F-SELECT rule を用いて、B に接している領域から 1つ選択する。領域 F のスコア評価には、F の data block を *RGN へ、B ののを *RGN2 へセットして rule を評価する。F は B の hole になっているので、最初の rule の述語が満足され、スコアには 3000 を、*S-MAIL にはメッセージ WINDOW をセットする。また、F に対しては、以降の rule はすべてスキップする。*STOP に 10 がセットされるので、これ以上、他の領域を探索することを中止し、F を最高スコアとして選択する。

Fに対する assume ステップでは、*B-WINDOW の43行目の rule の述語がみだされ、Fは「窓」として最高のスコアを得る。

領域E(壁の一部)もBのholeであるので、Fと同様にして選択されるが、今度は *BUILDINGが最高のスコアになり、*BUILDINGのラベルを得て、merge ステップでBにマージされる。

6. むすび

我々は、従来、bottom-up的な制御のみによっていた region growing に top-down な制御を導入し、その有効性を示した。また、とかく複雑になりがちな top-down 制御法の記述が容易に行なえ、かつ、直観的にもわかりやすいモデルの記述方式と制御構造を作成して屋外風景写真の解析に利用し、一応の成果を得た。また、前処理のために開発した recursive thresholding による領域分割プログラムは、領域解析システムの前処理プログラムとして十分に強力である。

現段階では、「物体としての形状」の性質は用いていない。従って、建物、自動車など、その形状が重要な特徴となる object の場合には、領域としての形状のみにより正しい解釈を得ることに若干の無理がある。我々は、このシステムに、早急に物体としての形状を評価する知識を付加し、その実効を確かめたい。

最後に、3節の領域分割プログラムの整備、特に、字像軸の選択に関する実験については、京大情報工学科4回生、林孝次君の協力に感謝する。

<参考文献>

- [1] Brice, C.R. and Fennema, C.L., "Scene Analysis Using Regions," *Artificial Intelligence*, Vol. 1, No. 3, 1970, pp. 205-226.
- [2] Yakimovsky, Y. and Feldman, J., "A Semantics-based Decision Theory Region Analyzer." *Proc. 3IJCAI*, Aug. '73.
- [3] Tenenbaum, J.M. and Barrow, H.G., "Experiments in Interpretation-Guided Segmentation." *Tech. Note 123, SRI*, '76.
- [4] Bajcsy, R. and Lieberman, L., "Computer Description of Real Outdoor Scene." *Proc. 2IJCPR*, Aug. '74, pp. 174-179.
- [5] Preparata, F.P. and Ray, S.R., "An Approach to Artificial Nonsymbolic Cognition." *Information Science*, Vol. 4, '72, pp. 65-86.
- [6] Minsky, M., "A Framework for Representing Knowledge." *A.I. Memo 306, A.I. Lab. MIT*, June 1974.
- [7] Ohlander, R., "Analysis of Natural Scenes." *Ph.D. Dissertation, CMU*, Apr. 1975.

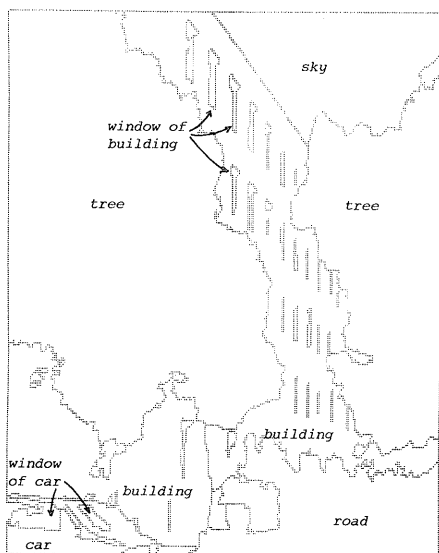


図11. 解析結果(意味領域)

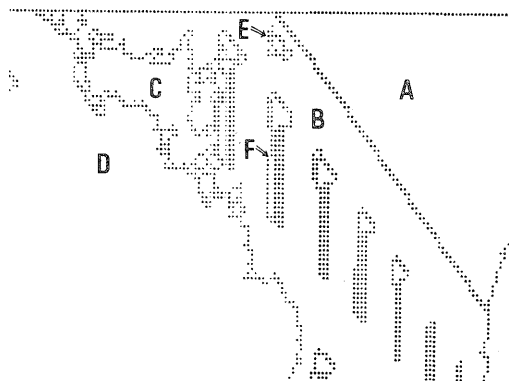


図12. 領域分割結果(図8)の一部