

JAVA言語で実装化された画像理解システムIUSの 動作概要と、その稼働方法

鈴木 昇一、川俣 博司、大槻 善樹

A Specification about an Image-Understanding System IUS Implemented with Program Language JAVA and Its Operational Method

Shoichi Suzuki, Hiroshi Kawamata, Zenju Otuki

あらまし

画素数だけの認識システムRECOGNITRON[3],[4]の個数からなる集合体として設計されている風景画像の理解システム (Image Understanding System) IUSは、現在、SYMANTEC社製のVisualCafe (Java開発環境) の下で稼働している。個々のRECOGNITRONは一枚の画像内の各々の画素に対応して構成されており、その対応する画素が如何なる事物を表現するのに用いられているかを決定するのに使われている。

本論文では、JAVA言語で実装済みのIUSを稼働させる観点から、その動作仕様と稼働方法が簡単に説明されている。IUSの起動方法、学習機能の説明、シミュレーションの手順、稼働上の諸注意が説明される。IUSを構成する各種プログラムファイルの役割が明らかにされ、シミュレーション結果を解釈するために最小限、必要とされる設定された規則が与えられている。

キーワード

RECOGNITRON モデル構成作用素 類似度関数 大分類関数 カテゴリ選択関数
ファジィ推論ニューラルネット グラフィック・ユーザ・インタフェイス

Abstract

An image-understanding system IUS which can understand the contents of an image in question is designed to be a set of recognizers called RECOGNITRONS[3],[4] whose number is as many as number of pixels in an image in question. Each RECOGNITRON corresponds to only one pixel, and is applied into recognizing to which category its pixel belongs. IUS is operating properly under Java integrated development environment called VisualCafe produced by SYMANTEC Inc..

This paper briefly specifies how IUS implemented with JAVA operates:

(1)The operational method of IUS. (2)The learning ability to understand the image. (3)The procedure for its simulation. (4)Matters that demand special attention.

We shall make clear the roles of all the program-files of which IUS is composed and explain rules at least needed to interpret results obtained from the behavior of IUS.

Key words

RECOGNITRON model-construction operator similarity-measure function binary-state function
category-selection function fuzzy-inference neural net graphic user interface

1. まえがき

S.Suzukiによって構築された“パターン認識の数学的理論[1]～[5]”を適用し、一枚の画像に如何なる事物が存在するかなどを見分ける画像理解システムIUS (Image-Understanding System) が既にJAVA言語で実装されているが[6]、本論文では、このシステムIUSを概観しながら、IUSを稼動をさせる方法が説明される。現在、画素総数だけの認識システムRECOGNITRON[3]、[4]からなる集合体として設計されているIUSはSYMANTEC社製のVisualCafe (Java開発環境) の下で稼動している。

本論文では、JAVA言語で実装済みの画像理解システムIUSを稼動させる観点から、その設計上の仕様と動作方法が簡単に説明されている。IUSの起動方法、学習機能の説明、シミュレーションの手順、稼動上の諸注意が説明される。IUSを構成する各種プログラムファイルの役割が明らかにされ、シミュレーション結果を解釈するために最小限、必要とされる規則とが与えられる。

将来、ソフトウェア、ロボットの眼として採用できるように、改良してゆくつもりであるが、現在、Borland社製JbuilderでのJava統合開発環境 (IDE; Integrated Development Environment) で稼動可能なように、IUSを再構築している途中である。

尚、未定義、或いは説明不足の述語、用語、記号については、4文献[3]～[6]にある。

2. 画像理解システムの操作方法

本章では、画像理解システムIUS[6]の操作方法が説明される。IUSは Microsoft Windows(95,98,ME) を搭載下の、Symantec社のVisual Cafe3.0 (以下VCと略すことがある) がインストールされたパソコンで稼動する。IUSが実装されているパソコンは次の表2.1に掲げられている。個々のパソコンに付けられている番号は1～6となっている。

表2.1 計算機番号と対応する計算機

1 : Gateway Performance 800 (Pentium] 800MHz)
2 : 東芝 BREZZA PV1010JA (Pentium 200MHz)
3 : akia Torando 8280/128 (Pentium] 800MHz)
4 : 富士通 FMV-DESKPEWOR model 936X (Celeron 366MHz)
5 : SONY VAIO PCG-C1 (MMX Pentium 266MHz)
6 : Gateway GATWAY SELECT (AMD Athlon 1.2GHz)

2.1 IUSの起動方法

IUSに入力できる画像は拡張子が[bmp]のMicrosoft-Bitmap形式（以下BMP形式と略すことがある）の、無圧縮24ビットカラー画像のみである。モノクロ画像であっても、24ビットカラーの灰色画像として保存しなければならない。従って、もし、ソフトウェア「Paint Shop」での「グレイスケール」などの機能を使ってモノクロ画像に変換した場合、8ビットに圧縮されてIUSでは使用できなくなってしまう故、デジタルカメラ等で撮影した画像をコンピュータに取り込み、BMP形式で保存しサイズ等の調整を行った後、必ずIUSの「Edit」メニューから「画像のモノクロ化」を選択して、モノクロ化を行い、モノクロ画像に変換しなければならない。

IUSの起動方法は2通りあり、通常は次の2.1.1の方法が推奨される。

2.1.1 Visual Café3.0 からの起動

IUSの構成ファイルの入っているフォルダ「ImageUnderstandingSystemKawamata」（図2.1）をダブルクリックして開き、Visual Café3.0 Project ファイル（拡張子が.vep）の「Image Understanding System.vep」をダブルクリックしてVCを起動する（図2.2）。

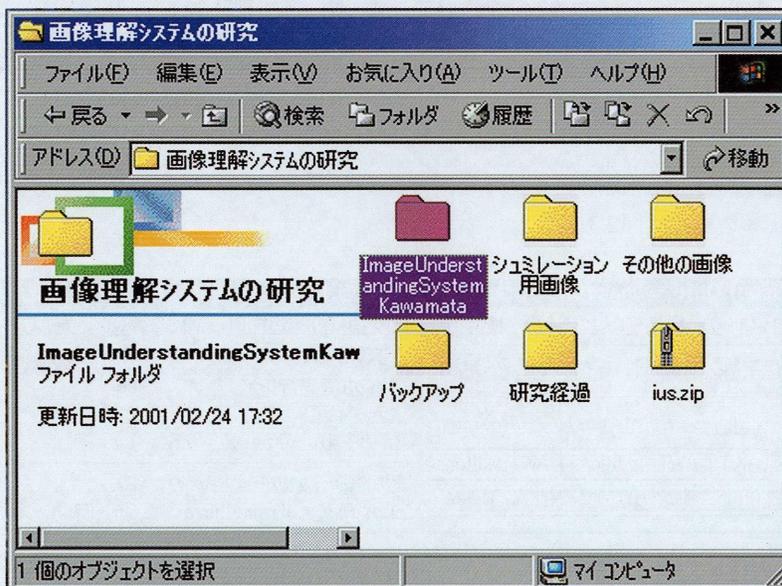


図2.1 IUSフォルダの選択

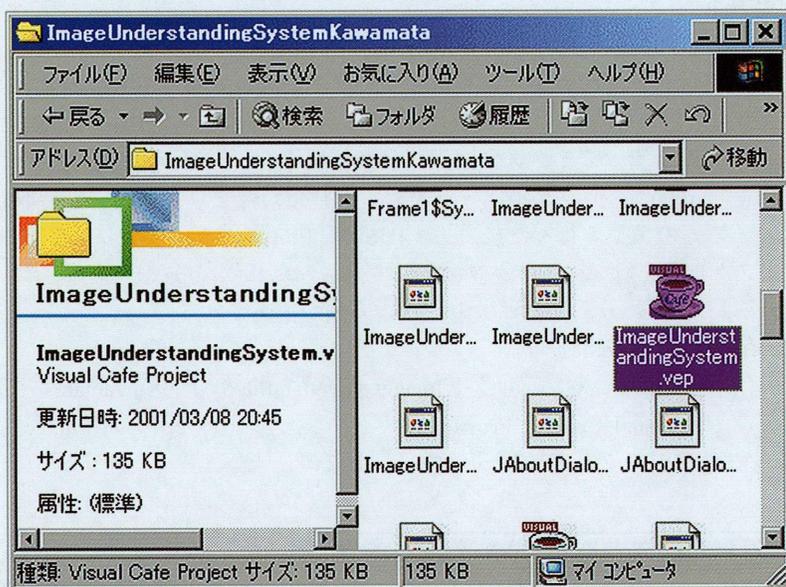


図2.2 IUSプロジェクトファイルの選択

VCが起動したら、画面上段のメニューバーにある「プロジェクト」をクリックし、「実行」メニューをクリックする（図2.3）。

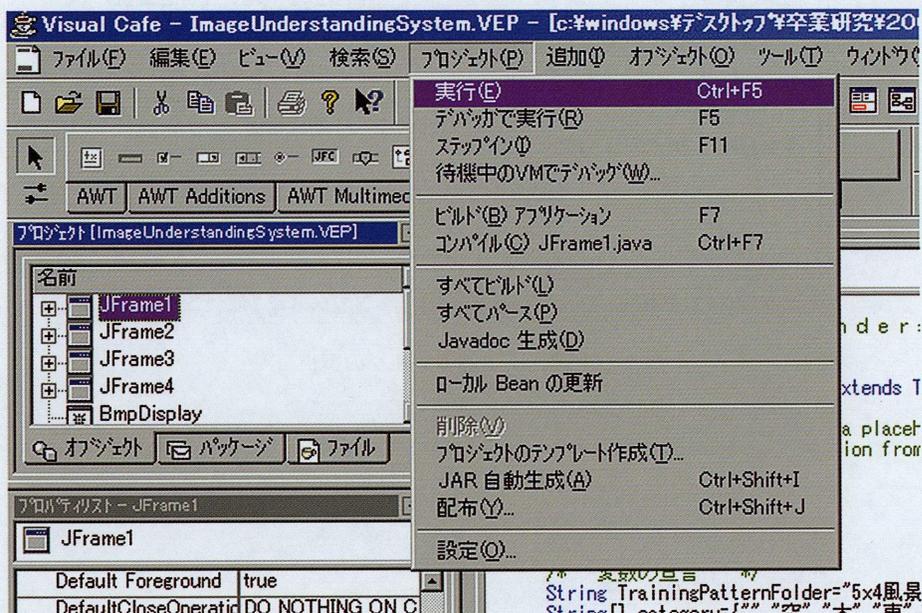


図2.3 VCからの実行

2.1.2 直接起動する場合

IUSの構成ファイルの入っているフォルダ「ImageUnderstandingSystemKawamata」(図2.1)をダブルクリックして開き、図2.4の如く、「RUN-ImageUnderstandingSystem.bat」をダブルクリックする。

プログラムを編集する必要が無い場合はこの方法を使うと、手間が省け短時間で実行できる。

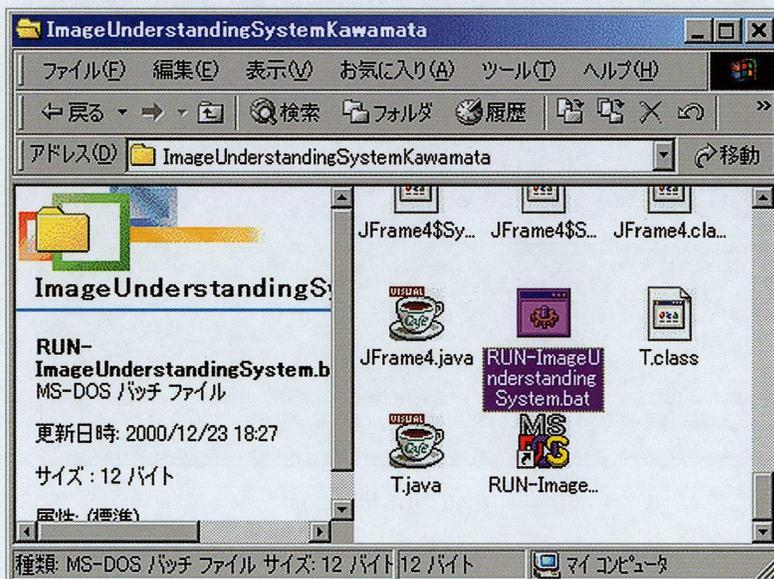


図2.4 直接起動

2.2 機能の説明

IUSを起動すると、図2.5のような画面が表示される。

ここで、画面上段のメニューバーにあるメニューは左から「File」メニュー、「Edit」メニュー、「IU」メニュー、「About」メニュー、「Exit」メニューとなっており、その下のツールバーにあるボタンは左から「Open」ボタン、「モノクロ」ボタン、「About」ボタン、「Exit」ボタン、「一時停止・再開」ボタンとなっている。



図2.5 IUS起動画面

2.2.1 「File」メニュー

「File」メニューをクリックすると、「Open」というメニューアイテムが表示される（図2.6）。これをクリックすると画像ファイルを選択する画面が現れるので（図2.7）、BMP形式のファイルを選択すると、その画像を表示することができる（図2.8）。

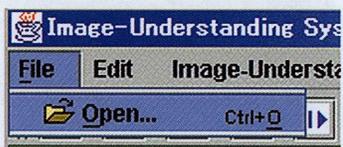


図2.6 「File」メニュー

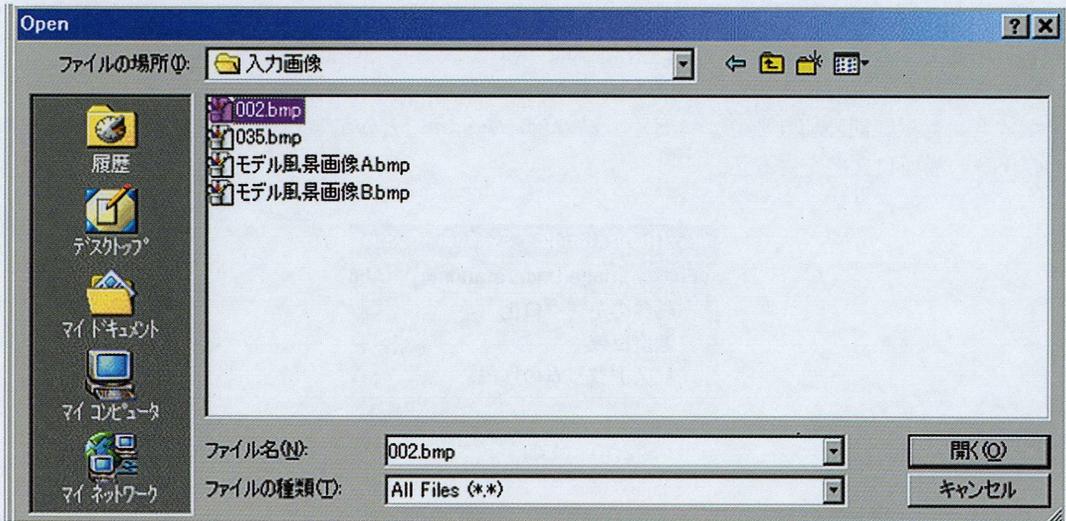


図2.7 ファイル選択

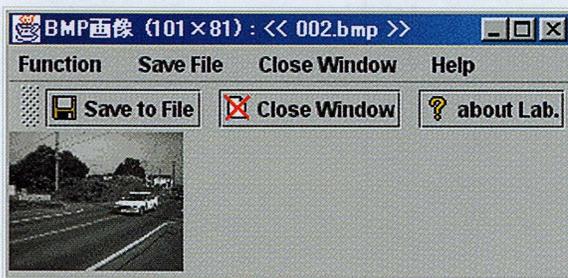


図2.8 画像の表示



図2.9 拡大メニュー

画像を表示しているフレームの「Function」メニューをクリックすると、「拡大」メニューアイテムが表示され（図2.9）、画像を拡大することができる。拡大率はシステムの設定画面の「その他」の画面で変更することができる。

「Save File」メニューからは「Save to File」メニューアイテムのみ選べ、図2.7のような画面が現れ、表示されている画像に名前をつけて保存することができる。BMP形式で保存されるため、拡張子に「.bmp」を付けなければならない。

「Close Window」からはこのフレームだけを消す「この窓だけを閉じる」メニューアイテムを選ぶことができる。

2.2.2 「Edit」メニュー

「Edit」メニューをクリックすると、「画像のモノクロ化」、「濃度反転」、「ヒストグラム作成」というメニューアイテムが現れ、クリックすると図2.7の画面になり、選んだ画像ファイルにそれぞれモノクロ化する処理、濃度を反転処理、ヒストグラムの作成処理を施すことができる。求められたヒストグラムはテキストを表示しているフレーム（以下テキストフレーム）から名前を付

けて保存し、「Excel」などの表計算ソフトから読み込み、表やグラフを作ることができる。テキストを保存する際は、拡張子を「.doc」にするのがよい。

テキストフレーム（図2.10）は、「File」メニューからテキストの保存と読み込み、「Edit」からテキストの選択範囲の切り取り、コピー、貼り付けを行うことができ（図2.12）、テキストエリア上の任意の場所に文字を書き込むこともできる。

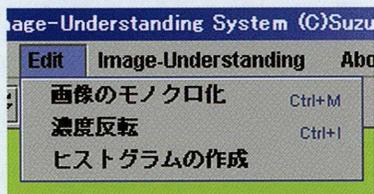


図2.10 「Edit」メニュー

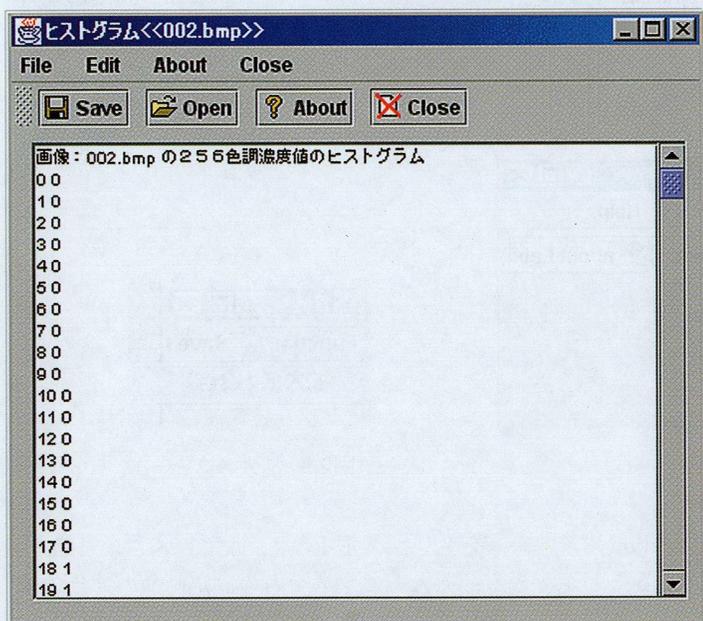


図2.11 テキスト表示用フレーム（テキストフレーム）



図2.12 テキストフレーム「Edit」

2.2.3 「IU」メニュー

「IU」メニューをクリックすると図2.13の画面になり、①「学習」、②「画像の解釈」、③「任意の画素の認識」、④「学習&解釈」メニューアイテムや、「補助機能」からは⑤「代表パターンの表示」、⑥「訓練パターンの表示」、⑦「学習結果の表示」、⑧「訓練パターンの作成」、⑨「 $T\phi$ 画像の表示」、⑩「 $T\phi$ のチェック」、⑪「ノルム距離・内積相違度の表示」、⑫「SMを用いた訓練パターン間の距離」、⑬「システムの状態を表示」、⑭「システムの設定」などのメニューアイテムを選択することができる。

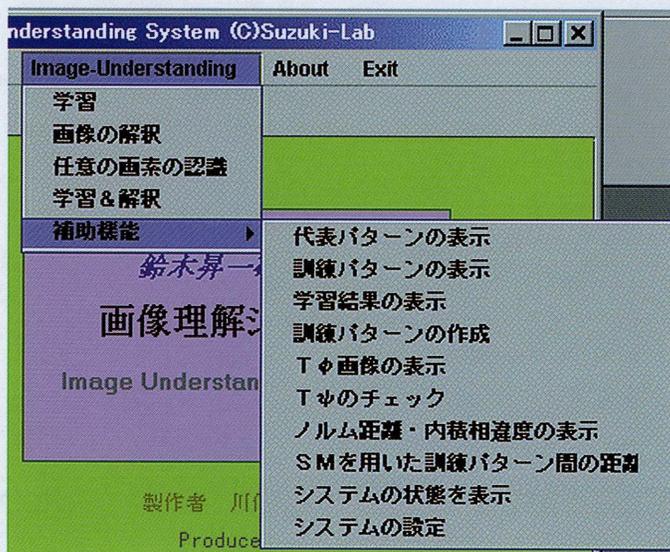


図2.13 「IU」メニュー

① 学習

モデル構成作用素 T には画像内外界の事物 φ のパターンモデル $T\varphi$ を形成する働きがあり、類似度関数 SM には外界の事物 φ のパターンモデル $T\varphi$ が記憶内パターンモデル $T\omega_j$ と似ている程度を測る働きがあり、最後に大分類関数 BSC には外界の事物 φ のパターンモデル $T\varphi$ が帰属する可能性のある候補カテゴリを複数個抽出する働きがある。学習は BSC 内の3助変数である重み $W[j]_k$ 、閾値 $b[j]$ 、標準偏差 $\sigma[j]$ についてなされる。

「学習」メニューアイテムをクリックすると、システムの学習が開始される。学習の進行状況はDOS画面に表示される。学習開始時にテキスト表示用フレームに学習に関する初期設定の情報が表示され、学習が終了するとその学習結果を表示したテキストフレームが表示される。ただし、④の「学習&解釈」とは違い、学習初期設定や学習結果は自動的にファイルに保存されないため、必要に応じてそれらのテキストフレームの「File」メニューから「SAVE」を選ぶか、「SAVE」ボタンをクリックして、ファイルへの保存処理を行う。これも2.2.2の説明と同じく拡張子を「.doc」にすることを推奨する。

学習処理は②や④などと同じく、メインスレッドとは別のスレッドを使って処理しているため、図2.5の画面で「一時停止・再開」ボタンをクリックするとその時点での学習結果を表示し、処理を一時停止することができる。このときIUSが処理を中止するかを尋ねてくるので、「はい」を選択すれば学習を最後まで行わずにその時点で学習を中止することができる。その時閾値 $b[j]$ の最適化は行われませんが、学習終了後の $b[j]$ の最適化の方法を選択する④の「システムの設定」の「関数の選択」の「閾値 $b[j]$ の決定法」は、設定が変更されるとその時点での $W[j]$ 、 $\sigma[j]$ の値を元を選択した方法で $b[j]$ の最適化を行うようになっているので、それを利用して $b[j]$ の最適化を行うことができる。

処理を中止するか尋ねられたときに「いいえ」を選択すると、処理は停止し、「一時停止・再開」ボタンをクリックされると処理を再開する。

なお、スレッド処理を停止するとき、しばしば原因不明のシステム停止が起こるので不用意な使用は避けるべきである。

学習に関する処理は、ソースファイル「ImageUnderstanding.java」のメソッド「int[][] learningProcess (JFrame3 f3)」の部分で行っているの、学習に関する δ や ϵ などのパラメータを変更する場合はメソッド内部でその変数が宣言・計算されている部分を直接書き換える。また、理想出力の値を変更するにはメソッド「double m_y(int j)」の戻り値を変更する。詳しくはプログラムリストに書かれているコメント文を参照。

② 画像の解釈

「画像の解釈」をクリックすると、ファイルを選択するダイアログボックスが開かれるので(図2.7)、画像の解釈を行いたい画像ファイルを選んで「開く」をクリックすると、その画像が表示され、この画像でよいか確認を取ってくるので、「はい」か「いいえ」をクリックする。「いいえ」を選ぶと、何も行われなが、「はい」を選ぶとその画像の解釈が始まる(図2.14)。

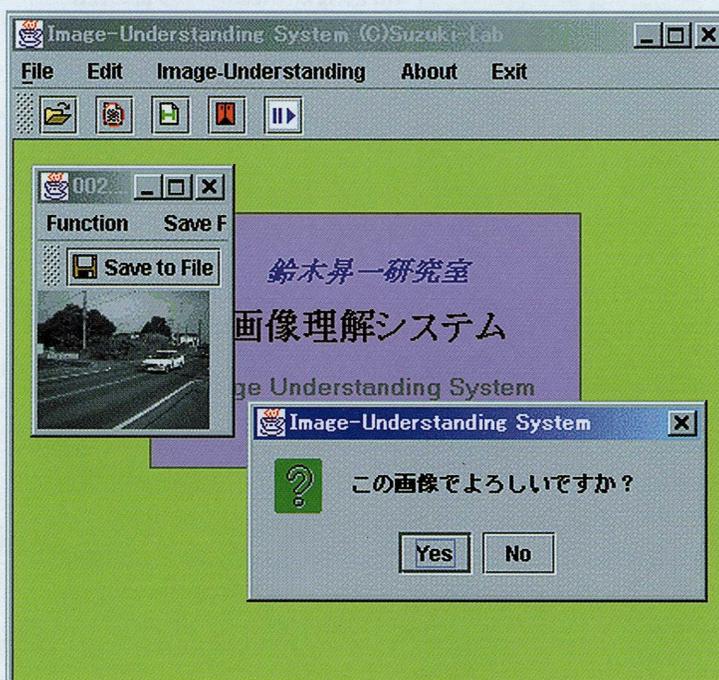


図2.14 画像解釈処理対象の確認

画像の解釈過程はDOS画面に表示されるが、①と同じく「システムの設定」の「その他の設定」の「結果をDOS画面に表示する」のところで「最終結果のみ表示する」を選ぶと各画素の認識過程を非表示にすることもできる。

画像解釈中に「一時停止・再開」ボタンをクリックするとその時点までの画像解釈の結果をカラー画像で表されたラベル情報と、処理対象画像と、その画素の簡単な認識過程の情報が表示され、①と同じく処理を中断するかどうかをたずねてくる。「はい」を選択すると処理はそこで中断され、「いいえ」を選択すると、処理は停止し「一時停止・再開」ボタンを再び押す

と処理を再開することができる。

画像の解釈が終了すると、マウスポインターが指す画素の座標とカテゴリとラベルを表示する機能とクリックした画素について③の処理を施すことができる機能を持った処理対象画像が表示され、ラベルとカテゴリラベルの分布を示したカラー画像とテキストが表示される。この「画像の解釈」も①と同じく、④の「学習&解釈」と同じく自動的にこれらの結果はファイルに保存されないで、必要に応じてそれぞれのフレームにある「SAVE」等のボタンやメニューを選んで保存する。

③ 任意の画素の認識

画像中の1つの画素についての認識過程を詳しく知りたい場合にこれを実行する。

「任意の画素の認識」をクリックすると、ファイルを選択するダイアログボックスが表示されるので(図2.7)、処理したい画像ファイルを選んで開く。画像が表示されたら、認識したい画素をフレーム下部の座標・帰属カテゴリラベル情報などを参考にしながら選び、クリックする。認識が始まると、その認識過程がDOS画面に表示され、認識が終了するとDOS画面に表示される各認識段階の認識過程と文献[6]の7.6節で説明されている値とが表示されたテキストフレームと、画像のラベル分布を表示したテキストフレームと、各認識段階での想起パターンを画像表示したフレームが現れるので(図2.15)、必要に応じてそれらを編集・保存する。

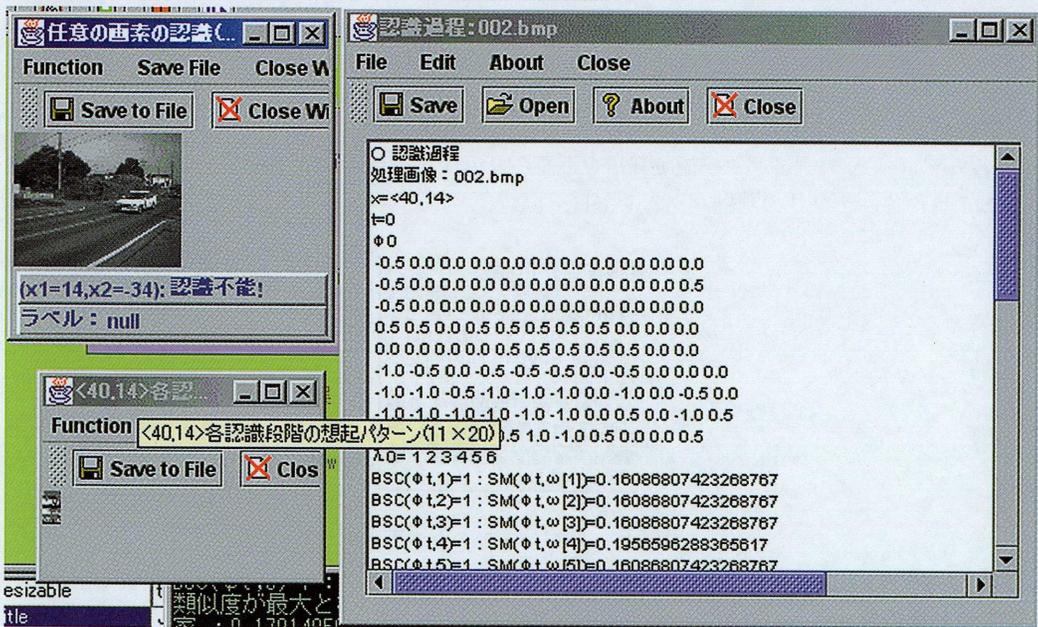


図2.15 任意の画素の認識画面

④ 学習&解釈

①「学習」と②「画像の解釈」を続けて行い、処理結果を自動的にファイルに保存する。

④の「システムの設定」を行った後、これを実行すれば、設定した条件下での実行結果が自動的に求まり、システムの設定状況、学習初期設定、学習結果、ラベル分布のテキスト表示、ラベル分布のカラー画像表現、対象画像が、IUSの構成ファイルの入っている「Image

Understanding System Kawamata」フォルダ内の「結果」の中にそれぞれ「システムデータ.doc」, 「学習初期設定.doc」, 「学習結果.doc」, 「解釈結果.doc」, 「ラベル画像」+処理画像ファイル名, 処理画像ファイル名の名前で保存される。

尚, 代表パターンや訓練パターンの保存はメモリーエラーを起こす恐れがあると判断し, これらを自動的に保存しないようになっている。これらを保存する場合は, ⑤「代表パターンの表示」や⑥「訓練パターンの表示」を使って表示し保存をしなければならない。

⑤ 代表パターン $T_{\omega[j]}$ の表示

各カテゴリ \mathcal{C}_j の代表パターン $\omega[j]$ を上からカテゴリ番号 $j=1$ から順に縦に表示され, 一緒に各カテゴリの代表パターンのモデル $T_{\omega[j]}$ のデータ情報をテキスト表示する。



図2.16 代表パターンの表示画面

⑥ 訓練パターン $\phi[j;q]$ の表示

各カテゴリ \mathcal{C}_j の訓練パターン $\phi[j;q]$ を上からカテゴリ番号 $j=1$ から, 左からパターン番号 $1 \sim r$ まで順に各カテゴリの訓練パターン $\phi[j;q]$ を表示する。



図2.17 訓練パターンの表示画面

⑦ 学習結果の表示

学習で求めた $\omega[j]$, $W[j]$, $b[j]$, $\sigma[j]$ の各カテゴリについての値を表示する。それぞれの値が求めた時刻 t は表示されない。

⑧ 訓練パターン $\phi[j;q]$ の作成

風景画像などからクリックした画素を中心に, 設定した視野の大きさに切り出し, システムの学習に使う訓練パターンを作成する機能。この機能で使う視野の大きさは, 「システムの設定」の「訓練パターンの設定」の「訓練パターン作成機能における視野の大きさ」の p と q の値であり, システムで通常代表パターンの大きさや認識時に使われている p と q とは別に設

定することができる。

初期設定はシステムで設定されている視野の大きさになっている。



図2.18 訓練パターンの作成画面

⑨ パターンモデル画像 $T\phi$ の表示

選んだ画像に⑭の「システムの設定」の「関数の設定」で選択されているモデル構成作用素 T をほどこし、そのデータの画像表示とテキスト表示を行う。

5値の T と S を採用する T を選択している場合は縦横のサイズが奇数で縦×横のサイズが1000,000以下のモノクロ画像であればどんな画像についても処理を行うことができるが、それ以外の T を選んでいる場合は、現在システムに設定されている p と q の大きさの画像しか処理できない。3値の T が処理できない理由は閾値 h^+_{ki} の値が一般的には k と l によって違うため p と q のサイズと違う画像については対応できないためである。これを k と l に関わらず例えば ± 0.5 などに決め、どんなサイズの画像でも処理できるようにするところに改良の余地がある。

⑩ 訓練パターンモデル $T\phi$ のチェック

式(4.9)の全ての $\phi[j;q]$ と $\phi[i;q2]$ ($i \neq j$) について、 $T\phi[j;q] = T\phi[i;q2]$ が存在しないかチェックを行う。該当するものがあればテキストフレームに表示される。

⑪ ノルム距離 $ndis$ ・内積相違度 $idis$ の表示

全ての代表パターン $\omega[j]$ のパターンモデル $T\omega[i]$ と $T\omega[j]$ ($i \neq j$)、および全ての訓練パターン $\phi[j;q]$ のパターンモデル $T\phi[i;q1]$ と $T\phi[j;q2]$ ($i \neq j$) についてのノルム距離(文献[6]の式(4.13))と内積相違度(文献[6]の式(4.41))、

$$(1) \quad ndis(T\omega[i], T\omega[j]) \quad (2.1)$$

$$(2) \quad idis(T\omega[i], T\omega[j]) \quad (2.2)$$

$$(3) \quad ndis(T\phi[i;q1], T\phi[j;q2]) \quad (2.3)$$

$$(4) \quad idis(T\phi[i;q1], T\phi[j;q2]) \quad (2.4)$$

の値を求め表示する。この値については、一旦ファイルに保存してから「Excel」でスペース(空白)を区切り記号とするように設定して、読み込ませれば表形式で印刷できる。

⑫ 類似度関数 SM を用いた訓練パターン間の距離

SM を用いた2つのパターン φ 、 η 間の距離 $SMdis$ を

$$SMdis(\varphi, \eta) \equiv -\log_e \sum_{j \in J} [SM(\varphi, \omega_j) \cdot SM(\eta, \omega_j)]^{1/2} \quad (2.5)$$

とし、全ての訓練パターン $\phi[j;q] \in \Psi_j$ について

$$(1) \varphi \in \Psi_j, \eta \in \Psi_j \quad (2.6)$$

$$(2) \varphi \in \Psi_i, \eta \in \Psi_j (i \neq j) \quad (2.7)$$

を計算し、テキストフレームに表示する。

⑬ 画像理解システムIUSの状態を表示

画像理解システムIUSの設定状況をテキストフレームに表示する。

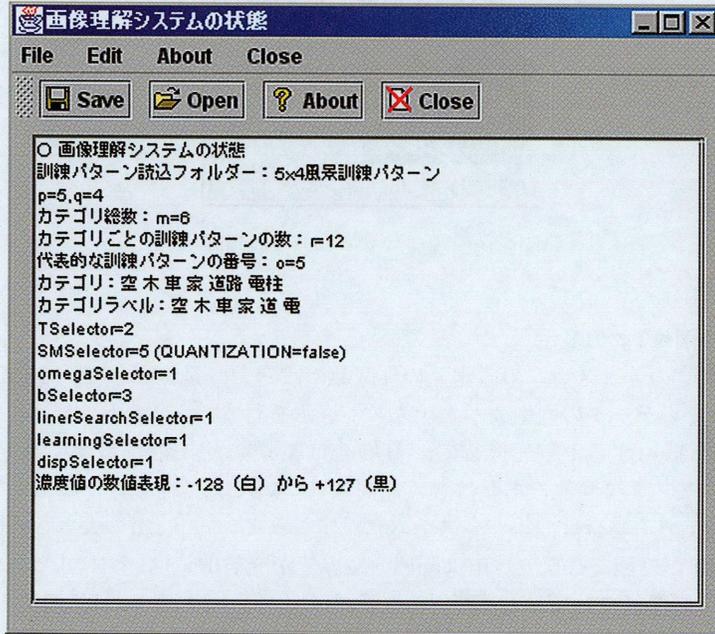


図2.19 画像理解システムの状態

⑭ 画像理解システムIUSの設定

「システムの設定」を選ぶとまず、図2.20の画面が現れる。ここで「関数の選択」, 「訓練パターンの設定」, 「その他の設定」を切り替え、設定の変更を行い、画面下部の「設定」ボタンをクリックすると、その設定が有効になる。「キャンセル」をクリックすると、設定は何も変更されない。

a. 「関数の選択」

ここからは「モデル構成作用素T」と「類似度関数SM」, 「代表パターン ω の決定法」, 「閾値bの決定法」の設定を変更することができる。a-1~a-4の(T1)等の括弧内の文字は、プログラム(ImageUnderstanding.java)上の関数名(メソッド名)である。

a-1 「モデル構成作用素T」

「3値のT」(T1), 「5値のT」(T2), 「Sを採用するT」(T3), 「移動, 回転, 伸縮を修正するT」(T4)「カテゴリごとに修正するT」(T5)の選択肢の中から1つを選ぶことができる。なお、(T4)と(T5)は現時点では未完成であるため選択しても機能しないので、完成次第機能を追加したい。

プログラム上ではTSelectorと表されており、1, 2, 3, 4, 5がそれぞれ5種類のモデル構成

作用素 (T1), (T2), (T3), (T4), (T5) に対応する。

a-2 「類似度関数SM」

「MIを使ったSM」(SM1), 「 ψ を考慮したSM①」(SM2), 「<選択不可>」(SM3), 「 ψ を考慮したSM②」(SM4), 「 ψ を考慮したSM③」(SM5), 「 ψ を考慮したSM④」(SM6)の中から1つを選択し, それを量子化するかどうかを選ぶことができる。この ψ は訓練パターンのことを示している。

(SM3)は現在未使用であるため選択しても機能しない。

プログラム上ではSMSelectorと表されており, 1,2,3,4,5,6がそれぞれ6種類の類似度関数(SM1), (SM2), (SM3), (SM4), (SM5), (SM6)に対応している。

a-3 「代表パターン ω の決定法」

「学習による ω の決定」(ω 1), 「視察による決定」(ω 2)から1つを選択することができる。

(ω 1)を選ぶと, 学習時において代表パターンを変数 o ($b-2$ 参照)番目の訓練パターンを初期値として学習によって求め, 求めた ω を r ($b-3$ 参照)番目の訓練パターン $\phi[j;r]$ に置き換える。

(ω 2)を選ぶと学習時において o 番目の訓練パターン $\phi[j;o]$ を代表パターンとして採用する。

プログラム上では ω Selectorとして表されており, 1, 2が(ω 1), (ω 2)に対応している。

a-4 「閾値 b の決定法」

BSC内の閾値 $b[j]$ の学習時における決定法(最適化)を, 「 ω のみを考慮した b の決定法」(b 1), 「 ψ を考慮した b の決定法」(b 2), 「全ての ψ のBSCを1にする b の決定法」(b 3), 「学習による b の決定法」(b 4)の中から1つを選択する。これを変更し「設定」をクリックすると, その時点での $W[j]$ と $\sigma[j]$ の値を元に $b[j]$ の値を設定された方法で計算するようになっている。

プログラム上ではbSelectorとして表されており, 1, 2, 3, 4がそれぞれ4種類の閾値(b 1), (b 2), (b 3), (b 4)に対応している。

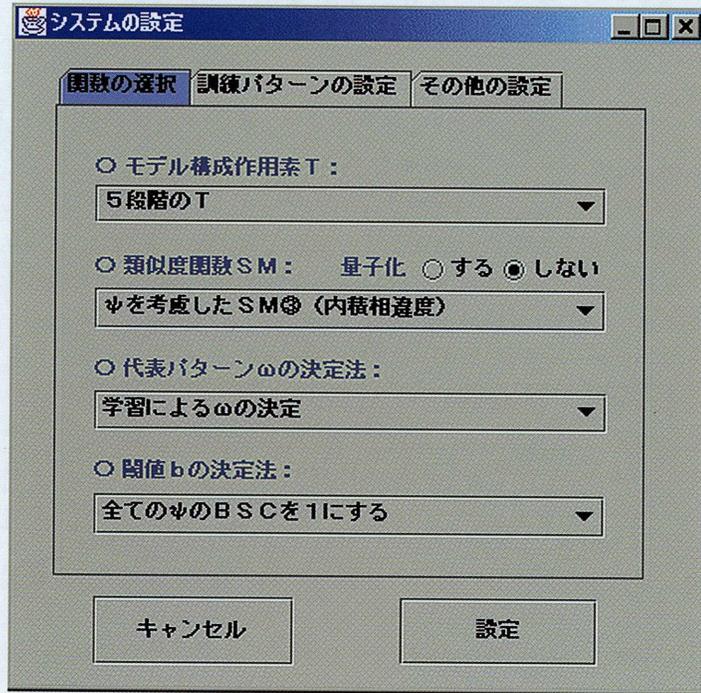


図2.20 関数の選択画面

b. 「訓練パターンの設定」

ここでは「訓練パターンのフォルダ」, 「訓練パターンの代表パターン番号」, 「カテゴリごとの訓練パターンの数」, 「訓練パターン作成機能の視野の大きさ」の設定を行うことができる。

b-1 ~ b-3 を変更し, 「設定」をクリックすると, IUS は訓練パターンの設定変更に伴い初期化されるので, b-1 ~ b-3 の設定は, 学習前に済ませておかねばならない。

b-1 「訓練パターンのフォルダ」

訓練パターンは, IUS の構成ファイルが入っているフォルダ「Image Understanding System Kawamata」内の「Training Patterns」というフォルダを検索し, ここに設定されている名前のフォルダの中から訓練パターンを読み込む。プログラム上では「Training Pattern Folder」と表される。

尚, 訓練パターンのファイル名はカテゴリ名+半角パターン番号+「.bmp」とする。例えば「空」というカテゴリの3番目の訓練パターンであれば「空3.bmp」というファイル名で保存する。

b-2 「訓練パターンの代表パターン番号」

各カテゴリに帰属する訓練パターンのうち, 最もそのカテゴリの特徴を典型的に表していると思われるパターン番号 o を設定する。これは全てのカテゴリに共通であり, カテゴリごとに別々に選ぶことはできない。

このカテゴリ番号 o を持つ各カテゴリの訓練パターン $\psi [j;o]$ がシステムの初期化時にお

ける代表パターン $\omega[j]$ として採用され、学習時における $\omega[j]$ の初期値あるいは $\omega[j]$ そのものとして採用される。

なお、学習によって $\omega[j]$ を決定する場合は ω を 1 にすると学習時に最初に読み込まれる訓練パターンと初期値が一致して、時刻が 0 で終了してしまうため、1 以外の値を選ばなければならない。

b-3 「カテゴリごとの訓練パターンの数」

1つのカテゴリに帰属する訓練パターンの数 r を設定するものである。IUSは訓練パターンをカテゴリ名+1+「.bmp」～カテゴリ名+r+「.bmp」までを読み込むので、例えばこの値を12と設定し、カテゴリ番号4が「家」である場合、IUSは「家1.bmp」～「家12.bmp」を訓練パターン $\phi[4;1] \sim \phi[4;12]$ として読み込む。

b-4 「訓練パターン作成機能における視野の大きさ」

⑧の「訓練パターンの作成」で選んだ画像のある画素をクリックした場合、その画素を中心として切り取られる縦横の大きさを設定するものである。 p は横の幅であり $-p \sim +p$ まで、 q は縦の大きさであり $-q \sim +q$ までを切り取るので、クリックした画素を中心として $(2p+1) \cdot (2q+1)$ の大きさに切り取られる。プログラム上では、システム自体の視野の大きさ p, q と区別するため、 x_p, x_q と表現している。尚、IUS自体の p, q は読み込まれた訓練パターンのサイズにあわせ自動的に決まるので、IUS自体の p と q の大きさを変えるためには、訓練パターンのサイズを変えその訓練パターンを読み込ませる。読み込まれる訓練パターンのサイズが統一されていないとエラーになる。

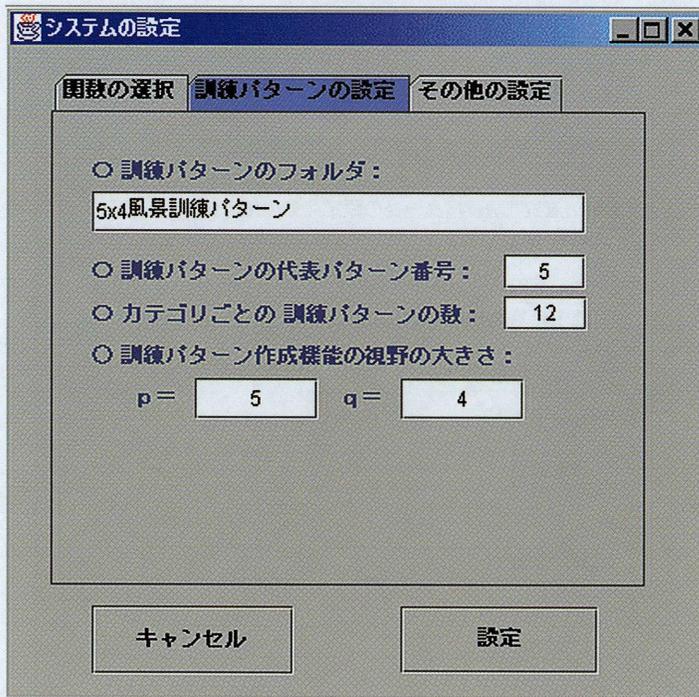


図2.21 訓練パターンの設定画面

c. 「その他の設定」

「結果をDOS画面に表示する」, 「拡大機能の倍率」, 「認識時におけるカテゴリの探索法」, 「学習方法」の設定を変更することができる。

c-1 「結果をDOS画面に表示する」

「学習」や、「画像の解釈」, 「任意の画素の認識や」, 「学習&解釈」を実行するとき、その計算の途中経過をDOS画面に表示するかしないかを設定する。

「途中結果を画面に表示する」を選択すると途中経過を表示し, 「最終結果のみ表示する」を選択すると, 途中の計算経過をDOS画面に表示しない。プログラム上ではdispSelectorという名前で表現されており, 1が表示, 2が非表示を意味する。

c-2 「拡大機能の倍率」

画像表示用フレームの拡大機能の倍率を設定する。プログラム上ではJFrame2.javaのBAIRITUとして表現されている。例えばこれに4を設定したとすると縦4倍, 横4倍の画像になる。整数しか選べない。

c-3 「認識時におけるカテゴリの探索法」

多段階認識時における, カテゴリ帰属知識のカテゴリ番号リスト λ' の探索法を選択する。「線形探索」, 「修正線形探索」から一方を選択する。

プログラム上ではliner SearchSelectorと表されており, 1, 2がそれぞれ「線形探索」, 「修正線形探索」に対応している。

c-4 「学習方法」

大分類関数 (binary-state classifier; rough classifier) BSC (\cdot, j) 内の重み $W[j]$, 閾値 $b[j]$, 標準偏差 $\sigma[j]$ の学習方法を選択する。「逐次学習」, 「非逐次学習」のいずれかを選択する。プログラム上ではlearning Selectorと表され, 1, 2がそれぞれ「逐次学習」, 「非逐次学習」に対応する。

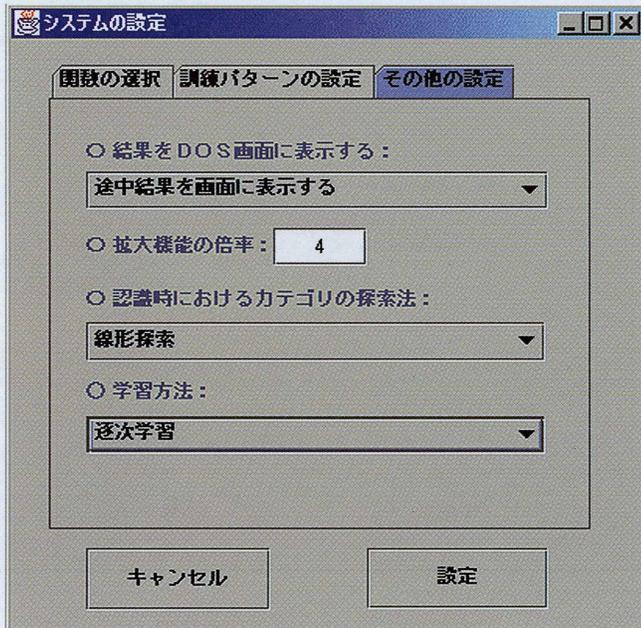


図2.22 その他の設定画面

2.3 シミュレーションを行うための手順

IUSは使用目的によって様々な手順で実行を行うことができるが、シミュレーションを行い、結果を保存する一連の手順の1例を示す。訓練パターンは作成済みとする。訓練パターンの作成方法に関しては「2.2.3 「IU」メニュー」の⑧および、「2.2.3 「IU」メニュー」の⑭—b。訓練パターンの設定を参照。

- (1) IUSを起動する。（「2.1 IUSの起動方法」を参照）
- (2) 「システムの設定」で各種設定を行い、シミュレーションの条件を設定する。（「2.2.3 「IU」メニュー」の⑭を参照）
- (3) 「学習&解釈」をクリックし、処理対象とする画像を選択し、画像を確認し「はい」をクリックする。（「2.2.3 「IU」メニュー」の④を参照）
- (4) 処理が終了したら表示されている処理対象画像の任意の画素をクリックし、認識の「成功例」、「失敗例」、「不定例」、「不能例」などの認識過程を確認しそれを表示しているテキストフレームと、各認識段階における想起パターンの画像を「Image Understanding System Kawamata」フォルダ内の「結果」フォルダに「認識過程.doc」、座標+「.bmp」などのファイル名で保存する。（「2.2.3 「IU」メニュー」の③を参照）
- (5) 「代表パターンの表示」、「訓練パターンの表示」を使って、表示された代表パターン、訓練パターンの画像を「結果」フォルダに保存する。
- (6) 「結果」フォルダ内に作ったファイルから「Word」等のソフトで結果レポート用のドキュメントを作成、保存、印刷する。文字によるラベル情報「解釈結果.doc」は「Word」等のページ設定でフォントサイズや文字数・行数を調整して画像と縦横の比率を同じにするために、「字送り」と「行送り」を同じにする。「Word2000」では、上下左右の余白を20mm、フォントサイズを3.5にし、文字数を101、行数を153にすると字送りと行送りが同じになる。

2.4 IUSを稼動するときの諸注意

IUSを稼動するとき注意しなければならない諸点は次の通りである。

i) カテゴリの設定および、学習に関する δ や ϵ などの変更はプログラム「Image Understanding.java」を直接書き換える。「2.2.3 「IU」メニューの①」およびプログラムに書かれているコメント文を参照。

ii) 「一時停止・再開」ボタンはシステム停止に陥る可能性があるので不用意に使用しないようにしなければならない。

iii) 「学習」や「画像の解釈」などのスレッド処理中にシステムの設定を変更することは可能であるが、実行途中に変更すると結果が正確に出なくなる。それ故、一貫した結果を求める場合はこれらの処理中に変更してはならない。「画像の解釈」実行中に「任意の画素の認識」を行うと、表示や結果が変わってしまうので注意しなければならない。

iv) 画像を同時に多数表示するとメモリーエラーを起こすことがあるので、不要なウィンドウについては閉じておかなければならない。

v) ファイル名やフォルダ名の指定が正しくても、JavaとOSで用いているフォントの違いなどの理由で、ファイルやフォルダが読み込めずにエラーになってしまう場合がある。その場合は適切な名前に変更しなければならない。指定ファイルまでのパスが遠かったりしても読み込めない場合がある。この場合はIUSの構成フォルダになるべく近いところか、デスクトップ付近に移動

してみるとよい。

vi) 浮動小数点の計算によって発生する誤差の修正はほとんど行っていない。

vii) 使用できる画像はMicrosoft Bit Map Fileの24ビット無圧縮カラー画像のみであり、見た目はモノクロ画像でも24ビットカラー情報で保存されている必要がある。また、画像サイズについては、縦×横が1000,000画素以下で、縦、横のサイズはともに奇数でなければならない。

viii) 代表パターン $\omega[j]$ の学習は稀に終了しないことがある。そのときはソースプログラム「Image Understanding. java」のメソッド `omega1(int j,int t)` の「//終了条件補正項」とある部分以下の
`}else if((t-mini_t)>2*r)mini_d=d;// 訓練パターンを2順してもこれまでの最小値を`
`繰り返さないときは最小値を選びなおす`

と書かれている行の、`[2*r]` の「2」を3や4などに増やしてみるとよい。

それでも終了しない場合は、終了しないカテゴリの訓練パターンを変形の少ないものを作り変えてみる。

ix) 「学習」、「画像の解釈」、「学習&解釈」は同時に実行することはできない。これらを実行中に、これら以外の処理を行うことはできるが、「任意の画素の認識」や、「システムの設定」は実行結果に実行条件の一貫性を失わせる恐れがあり、正常な結果を望む場合は実行してはならない。

3. 画像理解システムIUSを構成する各種プログラムファイルの役割と、シミュレーション結果を解釈するために必要とされる知識

大概が画像入出力プログラムImageProg-16を作成し、鈴木がIUSの基本設計を行い[5]、川俣がImageProg-16を改良しながら、JAVA言語を用い鈴木と相談しながらその基本設計に改良を加え、IUSを実装し、実際にシミュレーション[6]を実行している。本章では、画像理解システムIUSを構成する各種のプログラムファイルの役割が明らかにされ、シミュレーション結果を解釈するために最小限、必要とされる規則が説明される。

3.1 画像理解システムIUSを構成するプログラムファイルの役割

本研究でJava言語によって実装された画像理解システムIUSを構成する各種プログラムファイルの役割が簡単に説明される。

①Image Understanding.java … 画像理解システムの主要な処理を行うクラス。帰属するJFrame1クラスのインスタンス(cif)から与えられた命令を実行する。

②JFrame1.java … プログラム起動時に表示されるフレーム。将来的に複数の画像理解システムを同時に作動させることを想定したプログラム設計上の構造では、従属するImage Understandingクラスのインスタンス(IU)に人間が命令を与えるための窓口としての役割を持つ。プロジェクト「ImageUnderstandingSystem(.vcp)」におけるメインプログラムでもある。

③JFrame2.java … 画像表示用フレーム。保持している画像オブジェクト(currentDisp)を表示し、任意の画素の認識や、訓練パターンの切り出し、マウスポインタの指す画素の座標や帰属するカテゴリを表示する機能を持つ。保持している画像オブジェクトに拡大や保存の命令を与えることもできる。

④JFrame3.java … テキスト表示・編集用フレーム。実行結果の表示や、文字の書き込み、テキ

ストの保存や読み込みを行う。

⑤JFrame4.java … 画像理解システムの設定を変更する機能を持つフレーム。正確には、従属するImageUnderstandingクラスのインスタンス(IU)の機能設定を変更するためのクラス。

⑥BmpDisplay.java … 画像クラス。画像本体の濃度値データやサイズ、解釈結果のラベルや拡大比率などを保持している。なお、画像理解システムが処理対象として実際操作する濃度値の2次元配列型データは、このクラスのインスタンスが持つ画像本体の1次元配列の濃度値データを数値計算用に加工し、2次元配列にコピーしたものを処理している。また、拡大された画像については拡大画像自体の1次元データから拡大比率を元に対応する座標を割り出し、元の大きさの2次元配列を復元し、処理対象としている。従って、拡大画像そのものの2次元配列は存在しない。

⑦BitMapHandle.java … Microsoft Bit Map ファイル形式の24ビット無圧縮カラー画像データを読み込み、バイト配列から画像情報を生成するためのクラス。

⑧JaboutDialog.java … システムの所属・名称を表示するためのクラス。

⑨T.java … モデル構成作用素Tの追加・拡張を行うクラス。これを使用する場合は、「Image Understanding.java」のメソッドT (double[][] phi) の2行上の

```
//T Tn=new T(this) ;
```

文の最初の//を外さなければならない。

3.2 シミュレーション結果を解釈するための仕様

IUSを稼動し、そのシミュレーション結果が得られた場合、この結果を解釈するために必要とされる最小限の規則を説明しよう。

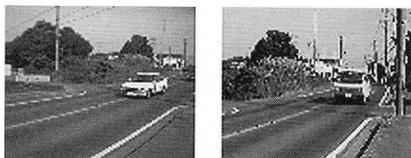
本IUSで採用されている全カテゴリは空、木、車、家、道路、電柱の6種類である。それぞれのカテゴリの番号と、シミュレーション結果表示のための文字および色との対応関係は表3.1の通りである。

表3.1 採用したカテゴリ

カテゴリ番号	カテゴリ	ラベル文字	ラベルカラー
1	空	空	
2	木	木	
3	車	車	
4	家	家	
5	道路	道	
6	電柱	電	
内部表現 -1	認識不定	*	
内部表現 0	認識不能	?	

採用した自然風景画像については、CASIOのデジタルカメラQV-10A，とSONYのモバイルコンピュータVAIO PCGC1-Rに内蔵されているCCDカメラで撮影した風景自然画像を，横×縦のサイズを101×81に縮小し，輝度値を取出して256階調単色画像に変換したものを使っている．人工画像はWindowsに付属しているソフトウェア「ペイント」で作成している．これら画像については以下のようにになっている．

・認識に用いた画像一覧



002.bmp

035.bmp

・学習に用いた画像一覧



003.bmp

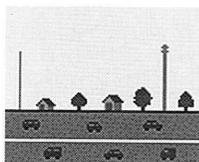
031.bmp

032.bmp

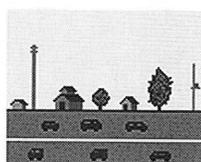
035.bmp

059.bmp

・人工画像（人工風景画像A.bmpから学習）



人工風景画像A.bmp



人工風景画像B.bmp

対象とする一枚の画像についてIUSが行った画像解釈結果のヘッダー部分にある「結果 #- #- #」は，例えば「結果1-2-3-004」なら，1月2日に鈴木昇一研究室の番号3の計算機で，画像004.bmpを処理した，という意味である．シミュレーションに使われたパソコンの番号については，表2.1のようにになっている．

「○ 画像理解システムの状態」のところにある各種「Selector」の意味は，次のようになっている．

- I) Tselector : 1=3値のT, 2=5値のT, 3=Sを採用するT, 4=移動, 回転, 伸縮を修正するT, 5=カテゴリごとに修正するT
- II) SMSelector : 1=MIを使ったSM, 2=訓練パターンを考慮したSM①, 3=なし, 4=訓練パターンを考慮したSM②, 5=訓練パターンを考慮したSM③, 6=訓練パターンを考慮したSM④

- Ⅲ) QUANTIZATION : true=SMを量子化する, false=SMを量子化しない
- Ⅳ) omega Selector : 1=学習による $\omega[j]$ の決定, 2=視察による $\omega[j]$ の決定
- Ⅴ) bSelector : 1=代表パターンのみを考慮した $b[j]$ の決定法, 2=訓練パターンを考慮した $b[j]$ の決定法, 3=全ての訓練パターンについてBSCを1にする $b[j]$ の決定法, 4=学習による $b[j]$ の決定法
- Ⅵ) linerSearchSelector : 1=線形探索, 2=修正線形探索
- Ⅶ) learningSelector : 1=逐次学習, 2=非逐次学習

4. むすび

大概が画像入出力プログラムImageProg-16を作成し、鈴木がIUSの基本設計を行い[5]、川俣がImageProg-16を改良しながら、JAVA言語を用い鈴木と相談しながらその基本設計に改良を加え、IUSを実装し、実際にIUSを稼動しそのシミュレーションを実行した。その報告は既に、文献[6]でなされている。本論文はこの文献[6]での説明不足を補う役目を背負っている。

本論文では、JAVA言語で実装済みの画像理解システムIUSを稼動させる観点から、その動作方法が簡単に説明された。現在、IUSはSYMANTEC社製のVisualCafe (Java開発環境) の下で稼動しているが、Borland社製JbuilderのJava統合開発環境IDEで稼動可能なように、IUSを再構築している途中である。

将来、ソフトウェア・ロボットの眼として採用できるように、改良してゆくつもりである。

文 献

- [1] 鈴木昇一：“認識工学”，柏書房，Feb.1975
- [2] 鈴木昇一：“ニューラルネットの新数理”，近代文芸社，Sept.1996
- [3] 鈴木昇一：“パターン認識の数理的一般解決”，近代文芸社，Jun.1997
- [4] 鈴木昇一：“認識知能情報論の新展開”，近代文芸社，Aug.1998
- [5] 鈴木昇一：“風景画像から知識を抽出し、解釈するシステムの、ファジィ推論ニューラルネットによる構成”，情報研究（文教大学・情報学部），no.23, pp.183-265, 2000
- [6] 鈴木昇一,川俣博司,大槻善樹：“風景画の理解に関するJAVA言語によるRECOGNITRONの計算機シミュレーション”，情報研究（文教大学・情報学部），no.27, pp.73-110,, Mar.2002

(論文題目 JAVA言語で実装化された画像理解システムIUSの動作概要と、その稼動方法, 文教大学情報学部・情報研究. No.28への投稿論文, 投稿年月日2002年9月2日)

(すずきしょういち 文教大学情報学部, かわまたひろし 富士ソフトABC株式会社, おおつきぜんじゅ 文教大学情報学部非常勤講師)