

2N-6

# 描画処理プロセスにおける感性要素の図的表現

長谷部 潔      鴨志田 稔      柳沢 隆夫      櫻本 肇

芝浦工業大学

## 1. はじめに

ソフトをより使いやすいものとするためには、システムとユーザとのインタフェースがフレキシブルでなくてはならない。システムとは「論理世界」の中に存在するものであり、それを動かすユーザは、「感性世界」の中に存在するものである。そして、ユーザがシステムを通して実行させた結果は、システム内のウインドを通してしかユーザには伝わらない。そのため、システムを使いやすいものとするためには、ウインドを見ただけでシステム内の処理プロセスを把握し、ユーザ自身が持っている感性基準と比較できなければならない。

本論文では、画像システム記述言語"WELL-PPP"<sup>[1]</sup>を用いて、この様な考えを基にして作られたシステムの図的表現について述べる。

## 2. システムの図的表現の説明

「システム内の処理の流れ」、「ウインド表示の流れ」、「ユーザのオペレーションの流れ」の関係を明確にするために、この図は、これらの三つの流れを平行して表記する。そして、それぞれの流れにおける幾つかの状態を四角で囲み、それを矢印で結んで全体としてのオペレーションの流れを表わす。図1参照。

以下、三つの流れについて具体的に述べる。

### 2-1. システム

このシステムは、カーネルとinterface function between object (IFO) という二つのパートで構成されている。カーネルは、ネットワーク構造の管理、generic関数の実行要求、次にやるべきオペレーションの生成を行なう。IFOでは、カーネルからの要求でactual関数に変換し、それに必要なデータをユーザに要求し、それが揃った時点で関数を実行する。ここでは、この二つのパートを区別せず、以下の様な大まかな流れだけを表記する。

ここでは、以下の様なシステム内の大まかな流れを表記する。

- 1) ユーザからgeneric関数がリクエストされる。
- 2) 「generic関数+制約条件」でactual関数に変換する。
- 3) 変換されたactual関数が必要とするデータを捜す。
- 4) データが無い場合、ユーザにデータをリクエストする。
- 5) ユーザからデータを受け取った時点でactual関数を実行する。

ユーザが入力した時、そのデータが不適切なものだった場合<sup>[2]</sup>は、システム内のvalidity checkによって却下され、データ入力する前の状態に戻る。

このvalidity checkとは、関数が実行する前にユーザが入力したデータがシステムの妥当かどうかを判定する機能である。validity checkは、以下の場合に起こる。

- (a) 二つのラインが互いに接近している場合  
この場合、原因として二つの場合がある。ユーザが

誤って起こしてしまった場合とユーザが故意にやった場合である。

(b) 一つ又は二つの主要点で円や曲線を選択した場合

(c) 三点が全て一直線に並んだ主要点で円や曲線を選択した場合

この二つの場合では、円や曲線は指定できないので直線に選択し直してもらるか又は指定する主要点の数を増やしてもらわなくてはならない。

(d) 補色に近い色を対極に置いてしまった場合

この場合、二つの色の間に無色部分が出来てしまう。システムは、ユーザにデータの再設定を要求するか、そのまま続けるのかを尋ねなくてはならない。

(e) 輝度データを見た場合、滑らかではない場合

この場合、高次の曲線で結ぶか、又はユーザにデータを与え直してもらう。

ユーザへのメッセージは、メッセージ\_ウインドを通して伝えられる。

これらの流れは、ユーザからは見えないものであり、ユーザから見えるものとは区別するために点線の矢印で表わす。

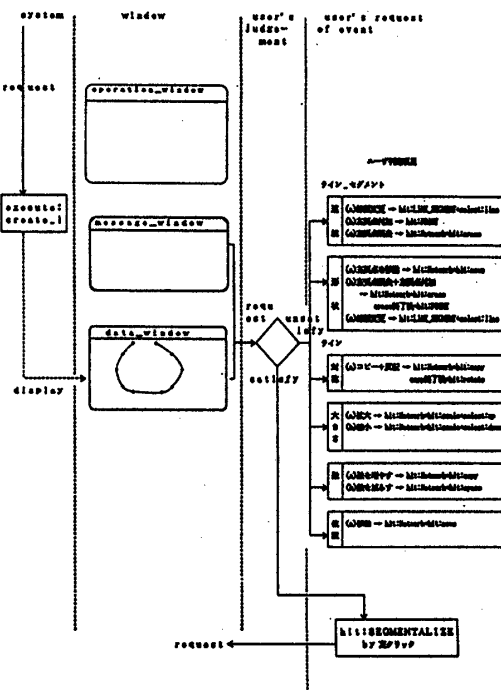


図1. システムの図的表現

### 2-2. ウインド

ここでは、現在のウインド状態を記し、どの様な情報がユーザに送られているのかを表わす。ウインドには、オペレーション\_ウインド、データ\_ウインド、メッセージ\_ウインド、選択ウインドの四種類があり

[3]、それぞれのウインドは、ユーザに異なる情報を与える。具体的には以下の通りである。

(a) オペレーション\_ウインド：オペレーション\_ウインドは、コントロール部、ネットワーク部の二つに分かれる。コントロール部では、データ\_ウインドの呼び出し、ネットワークの切り替え、現在のネットワーク名とプロセス名を表示する。ネットワーク表示部は、オブジェクトとその関数のネットワークを表示し、現在の描画過程オブジェクトを赤字で、次に行うべき関数を青わくで表している。ネットワーク表示部で示されているnext-candidate-operationsをマウスでヒットすることにより、その関数をリクエストする。

(b) データ\_ウインド：ユーザへのデータ入力要求やユーザからのイベントを受け付ける。データ入力は、マウスの左クリックをヒットすることで行われる。ユーザは、このウインドを通して、自分が作った絵を評価することとなる。

(c) 選択ウインド：関数を実行させるために必要なデータがシステム内に存在しない場合、このユーザ\_ウインドを通してユーザにリクエストする。

(d) メッセージ\_ウインド：選択ウインドと併用して表示され、ユーザに制約条件の入力を促す。又、ユーザが間違っただけを入れvalidity checkで不適切と判断された場合やオペレーション操作を間違っただけ、ユーザに意志を尋ねるメッセージを表示したりユーザに対処のしかたの説明を表示したりする。

### 2-3. ユーザ\_オペレーション

ここでは、ユーザが行うオペレーションの流れとウインド上に写し出されたオブジェクトに対するユーザの判定項目を記す。

#### 2-3-1. オペレーションの種類

ユーザ\_オペレーションは、user's judgement と user's request of eventの二つに分けられる。

user's judgementとは、ユーザが対象画像に対して持つ評価である。user's request of eventとは、評価後の一連のオペレーションである。ユーザは、感覚的意識により対象画像を判定、すなわち図1のuser's judgementを経て、eventのrequestが行われる。感覚的意識が活性化するか否かは、satisfyまたはunsatisfyによって判定される。

#### 2-3-2. ユーザ判定項目

実行処理の結果として得られた対象画像は、データ\_ウインドに写し出され、ユーザによってsatisfy、unsatisfyという二つの感性要素で評価される。対象画像が不満足な場合、ユーザがそれを修正する基準としてユーザ判定項目を設定する。このユーザ判定項目とは、対象画像の不満となり得る点を具体的に上げ、それぞれの場合における修正オペレーションの手順を記したものである。ユーザは、この判定項目を参考にして修正問題、並びに修正するため次にやるべき一連のオペレーション手続きを決定するのであるのである。いくつかのオブジェクト状態に対して、多くのユーザ判定項目がある。例えば幾つかのユーザ判定項目を以下に記す。

判定項目と次にやるべきオペレーション

#### (1) ライン\_セグメントに対する判定項目

##### <連続性>

不連続な場合

- (a) ラインの種類を変える方法  
→hit:LINE\_SEGMENT + select: line
- (b) 主要点を新たに加える方法  
→hit:POINT
- (c) 主要点を幾つか消す方法  
→hit:Network + hit:erase

##### <形状>

線をふくらませたい、又はへこませたい場合

- (a) 主要点を移動する方法  
→hit:Network + hit:move

(b) 一度ライン\_セグメントを消してから新たに主要点を加える方法

- hit:Network + hit:erase
- erase終了後 + hit:POINT

##### <線>

(a) 線の種類を変える方法

- hit:LINE\_SEGMENT + select:line

#### (2) ラインに対する判定項目

##### <対称性>

(a) コピーしてから反転する方法

- hit:Network + hit:copy
- copy終了後 + hit:reverse

##### <大きさ>

(a) 拡大する方法

- hit:Network + hit:scale
- + select up

(b) 縮小する方法

- hit:Network + hit:scale
- + select down

##### <数>

(a) 増やす方法

- hit:Network + hit:copy

(b) 減らす方法

- hit:Network + hit:erase

##### <位置>

(a) 移す方法

- hit:Network + hit:move

### 2-3-3. オペレーションの流れ

ユーザ\_オペレーションの流れは、具体的に以下の図2の通りである。

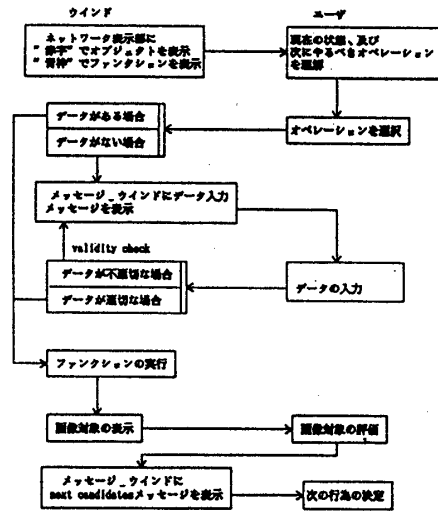


図2. オペレーションの流れ

### 3. まとめ

本論文では、描画処理システムにおけるオペレーションの流れとそれに対するユーザの操作方法を図を使って報告した。このマニュアルを見ることでシステムをよりユーザフレンドリーな物とすることであろう。次の段階として、このマニュアルを基にして全てのnetworkにおけるvalidity checkを作製する。

### 文献

- [1] 榎本、鴨志田：“分野記述型言語の構造” 情報処理学会第44回全国大会 1992
- [2] 関、鴨志田、榎本、宮村：“カラー画像の直接修正法とデータの妥当性” 情報処理学会第44回全国大会 1992
- [3] 鴨志田、丹羽、榎本：“オブジェクトネットワークによる画像システム記述言語” 情報処理学会第44回全国大会 1992