

ペンウィンドウシステムのためのプログラム開発支援系

7J-7

平井孝史, 早川栄一, 並木美太郎, 高橋延匡

(東京農工大学)

1. はじめに

入力デバイスとしてペンを用いたウィンドウシステムであるペンウィンドウシステムは、今まで人が行ってきた「紙とペン」に近い自然な感覚で入力・操作ができることから、今までにない UI をもった AP を実現できる可能性があると考えられる。

筆者は、「ペンウィンドウシステムの AP の開発を容易にする」「ペンウィンドウシステムの UI の構築・実験の支援をする」ことを目標に研究を進めており、ペンウィンドウシステム“未”[1]上で、画面設計などを支援する開発支援系の初版を実現した。

本稿では、初版から得られた問題点をもとに、開発支援系第二版について述べる。

2. 初版の問題点

開発支援系初版においては、ペンウィンドウシステムの AP の開発の手間を削除するために、

- ・画面設計
- ・イベント登録 (UI の定義)
- ・結果を C のソースとして出力

などの機能を実現した。これから得られた問題点として、次のことがある。

(1) 思考の中断が発生する

初版では、メニューから文字列や四角形などを選択し、それをウィンドウに配置していくことによって画面設計を行っていた。この方法だとメニューとウィンドウとの間をペンが頻繁に行き来することになり、その結果思考を妨げることがあった。

(2) ペンの特徴を活かせていない

初版では、ペンの特徴を用いたこととして、編集作業をジェスチャで行うことができたが、作成される AP 側にペンを用いた特徴を反映させることがあまりできなかった。入力デバイスとしてペンを用いているにもかかわらず、単なるポインティングデバイスとしてしか使っていないようなところがあった。

(3) ストロークデータの扱いが不十分

入力デバイスとしてペンを用いているため、画面設計における画面の構成要素としてストロークデータを扱うことが考えられるが、初版ではそのペンの特徴であるストロークデータを扱うための機構は特に用意されていなかった。

3. 設計方針

問題点をふまえて、次のような設計方針をたてた。

(1) 「紙とペン」を指向する

ウィンドウシステムに限らず、AP の開発時にはその画面構成などのイメージを紙に描くことがよくある。そのような過程において、「紙とペン」だけによる行為は人の思考を妨げることなく進められる。思考を中断させることなく開発を進めるには、ペンを積極的に用いて「紙とペン」のように使用できることが必要である。

(2) 手書きの UI の構築支援をする

UI においてだけでなく、単なる画面表示に関しても、システム固有のものであるフォント（種類や大きさ）や図形などにとらわれない自由な表現が可能になる。つまりこれはペンを用いた手書き入力による利点だと考えられる。ことごとくによってペンの特徴を活かしたものができると考えられる。

(3) ストロークデータを扱う

ストロークデータはペン特有のデータであり、ペンによる自由なストロークデータの入力はそれだけでペンの特徴を用いているとも考えられる。そのストロークデータを扱うため、ストロークデータの結合・分離・オブジェクトとしての扱いなど、特別の機構を持つようにする。

4. 設計

設計方針をもとに、初版での画面設計と UI の定義を次のようにした。

4.1 画面設計

画面設計を行う際、従来のものは文字列や四角形など画面に表示させるものをメニューなどから選択し表示させるものであった。しかしここでは方針に述べたようなことから、入力にはペンによる手書きのストロークデータを扱えるようにする。すなわち画面設計をするべきウィンドウにペンで直接入力し、それがそのままその AP の画面として有効になるようにする (図 1)。

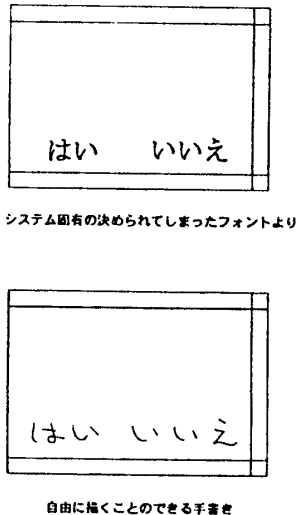


図 1

4.2 UI 定義

“未”はプログラミングモデルとしてイベント駆動方式を採用している。そのため、ウィンドウのある領域に何か入力があったら (何かイベントが起きたら) ある関数を実行するといった、イベント登録が必ず必要になる。実際には、(1) イベント領域を指定し、(2) その領域に実際に行うように入力し、(3) そのときの処理を記述する、といった順番で行う (図 2)。

- (1) ウィンドウに表示しているものにあわせてペンによりイベント領域の指定をする。“あわせて”と書いたが、なにもびったりあわせる必要はない。実際に表示しているものよりも大きくしたり小さくしたり、ユーザの自由に行う。そうすることで、従来の枠から離れた UI ができる可能性がある。
- (2) イベントの種類は一つではないし、イベント登録にはさまざまなパラメータの設定が必要になる。イベント領域に実際にそのイベントを起こすことによ

って、イベントの種類とイベント登録の際のさまざまなパラメータの設定を自動化することができる。

(3) テキストエディタを起動し、イベント領域に実際にイベントが起きたときの処理を記述する。“未”上の AP は C 言語で書かれているため、ここでの処理も C 言語で記述をする。

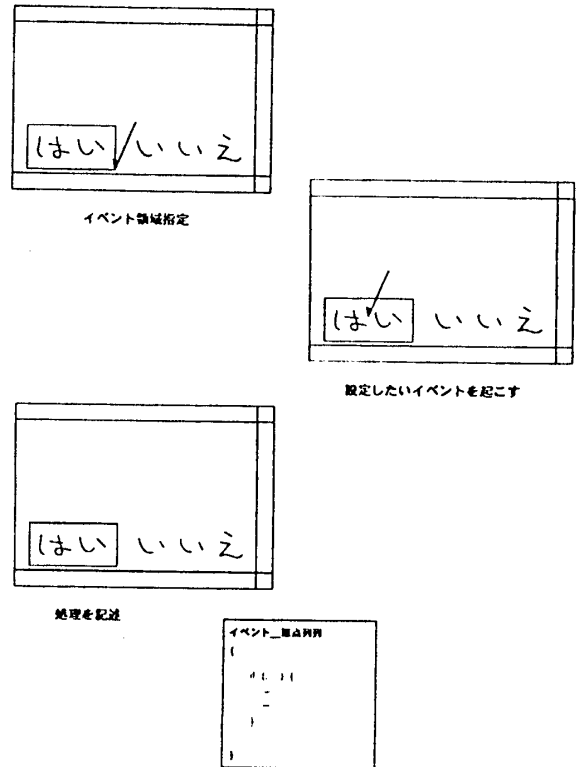


図 2

4. おわりに

本稿では、手書き入力を重視したペンウィンドウシステムのための開発支援系について述べた。今後は実現を行う予定である。

参考文献

- [1] 河又 ほか：表示一体型液晶タブレットを用いた“未”ウィンドウシステムの設計と実現，情報処理学会ヒューマンインタフェース研究会，45-17，1992
- [2] 荒井 ほか：手書きヒューマンインタフェースの構築支援環境，情報処理学会ヒューマンインタフェース研究会，35-7，1991