

ウィンドウシステム“未”用UI評価環境によるペンインタフェースの一評価

5W-5

二村尚, 中島一彰, 早川栄一, 並木美太郎, 高橋延匡

(東京農工大学 工学部 電子情報工学科)

1. はじめに

アプリケーションプログラム（以下AP）のユーザインタフェース（以下UI）の設計において、プロトタイプの実現、テスト、評価を反復的に行なう手法が用いられる[1]。この反復は、一般的なモデルだけでなく、AP固有の機能やターゲットユーザの特徴を理解し、UIの設計に反映するために行なわれる。

そこで筆者らは、実現されたAPをユーザに操作させて、操作の過程を分析するユーザテストを支援する環境（UI評価環境[2]）をペンウィンドウシステム“未”[3]上で実現した。本稿では、“未”用APである「分散手書きKJ法システム」[4]のユーザテストを行ない、UI評価環境に対する要求、再設計と「分散手書きKJ法システム」の一評価について述べる。

2. “未”用APにおけるペン入力の一評価

“未”上では、手書きUIを利用したAPとして「分散手書きKJ法システム」が実現されている。このAPではペンで文字、図の入力や操作が可能であり、現在もUIの改良が行なわれている。

その改良の一例を挙げる。分散手書きKJ法システムでは、APの操作をペンのストロークで行なうこと（ジェスチャ入力）が可能であるが、前の版ではジェスチャ入力は、ペンの側面にあるサブスイッチを押しながら操作を行なう必要があった。これは、ペンのストロークがジェスチャ入力なのか、文字や記号などの入力なのかを判断するためであるが、操作中にスイッチが必要なのか不要なのかを意識しなくてはならないという欠点があった。APの設計者は、次の版ではジェスチャをサブスイッチなしで入力できるように改良した。

今回はこの改良による効果を示す一つのデータとして、ユーザの作業時間の変化を、実際に実現されているシステムおよび環境で調査し、比較評価を行なう。

3. ユーザテストによるUI改良前後の比較評価

3.1 テスト方法

分散手書きKJ法の作業として、通常行なわれる作業をテスト用のタスクとして設定し、その作業を2種類の

インタフェースで被験者に実行させて、その作業時間を測定する。これにより、同じ作業を行なう時間が全体としてどの程度異なるのか、入力のどの部分に違いがあるのかを調査する。

本評価で設定したタスクを図1に、そのテストによって測定する項目を図2に示す。

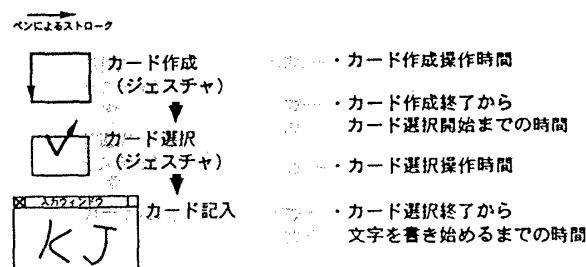


図1 テストユーザに与えたタスク 図2 測定項目

3.2 テスト中に記録するデータ

図3の時間を調査するために、ペンストロークの入力時刻（開始時刻、終了時刻）を記録する。また、改良前の版でサブスイッチのon/offがどのタイミングで行なわれているのかを記録する。

4. UI評価環境の再構成

UI評価環境は、ユーザテストによるユーザの入力の履歴と、入力に対するAPの動作の履歴をファイルに保存し、ユーザテストによるUI評価を支援する環境である。この環境を利用して、3.2で述べたデータを収集するために、一部の機構を再設計し、データの記録、集計を行なう環境を実現した。次にその詳細について述べる。

4.1 ユーザの入力履歴を記録する機構の再構成

試作した環境では、ユーザの入力履歴として、ウィンドウシステムからAPに発行される入力データ（イベントデータ）と、受け取った時刻を記録する機構を提供した。今回はそれに加え、ウィンドウシステムをUI評価用に改造し、ウィンドウシステムの入力デバイスの処理を行なう機構で入力時刻の記録を可能にした。

4.2 APの動作履歴として記録するデータ

ユーザテストで記録した入力時刻は、入力によって実行された機能別に集計する。その集計を支援するデータとして、入力履歴に対して次に示すデータを記録するように、APを改造した。

- ・実行されたAPの機能名
カード作成、選択など、入力に対するAP動作の内容を判断して集計する。
- ・操作の対象となったカードの識別子
カードの作成から、カードの選択を行なうまでの時間を計算する場合、操作の対象となったカードがどれかを判断して、カード別に時間を集計する。

4.3 記録したデータの集計に対する要求

テスト中に記録した入力履歴の人手による集計は、非常に手間がかかる。そこで、APの動作履歴や、入力が発生したウィンドウ名を利用して作業時間や作業間隔を自動的に計算するツールを作成する。ツールによる集計の例を図3に示す。

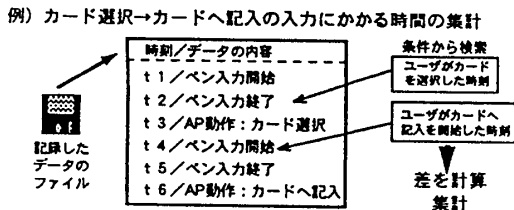


図3 データ集計ツールの概要

5. ユーザテストの実施

2. で述べたユーザテストを、5名のユーザに対して行った。テストは図 のタスクを10回入力させ、それを2種類のインタフェースで入力させた。また、10回入力し終えた後は、休憩と簡単な練習をさせて、双方の操作が混同しないようにした。カードへ記入する内容はあらかじめ用意したものを与えて、内容を考える時間を排除した。

6. テスト結果

図2で示した項目を表1、2に示す。サブスイッチを使用した場合のカード作成操作とカード選択操作の時間は、スイッチを押してからストロークの入力後スイッチを離すまでの時間である。そこで表3では、ストロークの入力時間だけを計算して示した。

テストユーザA～Eはおよそテスト環境の経験が多い順番に並んでいる。Aは普段から使用しており、Eは今回初めて使用した。結果を表1, 2, 3に示す

7. 結果の考察

表1, 2のすべての項目において、サブスイッチを使用

表1 サブスイッチを使用した場合の平均入力時間

テストユーザ	A	B	C	D	E
カード作成操作	2.11	2.00	2.87	2.20	2.44
作成→選択	2.07	2.69	2.23	3.97	4.50
カード選択操作	0.75	0.71	1.05	1.04	1.05
選択→文字入力	2.14	2.20	3.22	2.59	2.06

表2 サブスイッチを使用しない場合の平均入力時間

テストユーザ	A	B	C	D	E
カード作成操作	1.44(68)	1.38(69)	1.16(58)	1.24(56)	1.00(41)
作成→選択	1.22(58)	1.27(47)	1.34(61)	3.20(81)	3.82(85)
カード選択操作	0.41(55)	0.40(56)	0.49(47)	0.36(33)	0.33(31)
選択→文字入力	2.08(97)	1.44(65)	3.18(99)	2.16(83)	1.87(80)

() 内は表1の値を100とした数値

表3 ストロークの平均入力時間

テストユーザ	A	B	C	D	E
作 サブスイッチあり	155	148	202	143	155
成 サブスイッチなし	144(93)	138(93)	116(57)	124(87)	100(65)
選 サブスイッチあり	40	38	60	55	54
択 サブスイッチなし	41(100)	40(101)	49(82)	36(65)	33(61)

() 内は上段の値を100とした数値

した場合(以下旧版)の方が、使用しない場合(以下新版)より時間がかかっている。特にカード作成、選択の操作時間の違いが大きく、慣れていないユーザほどこの傾向がある。一方カードの選択から文字入力までの時間は、サブスイッチを押す操作がないので、新版と旧版の差が小さい。

表3で示したストロークの入力時間は、サブスイッチの操作に慣れているユーザにとってあまり変わらないが、慣れていないユーザはストロークの入力時間も長くなっている。つまり、サブスイッチを押す操作によって、入力速度にも影響があることを示している。

8. おわりに

本稿では、ユーザテストによって「分散手書きKJ法」のUI改良の効果を検証した。またUI評価環境を使用してユーザテストにおけるデータの収集と集計を自動的に行うことができた。

今後は、さらに多くのテストを行ない、手書きUIの評価方法とその評価を支援する環境に対する要求について検討する。

参考文献

[1] Jacob Nielsen: The Usability Engineering Life Cycle, the IEEE Computer Society, COMPUTER March 1992, Vol.25, No3
 [2] 二村, 他: 手書きユーザインタフェースの評価を支援する環境の実現, 情報処理学会ヒューマンインタフェース研究会資料61-4, 1995
 [3] 早川, 他: ペンインタフェース研究・開発のためのウィンドウシステム“未”(HITSUJI)の設計と開発, 情報処理学会論文誌, Vol.36No.4, 1995
 [4] 中島, 他: ペンインタフェースを用いた分散KJ法システム, プログラミングシンポジウム, 1995