

(19) 日本国特許庁 (J P)

(12) 公開特許公報 (A)

(11) 特許出願公開番号

特開2000-172395

(P2000-172395A)

(43) 公開日 平成12年6月23日 (2000.6.23)

(51) Int.Cl. <sup>7</sup>	識別記号	F I	テームト* (参考)
G 0 6 F 3/00	6 5 5	G 0 6 F 3/00	6 5 5 A 5 C 0 8 0
G 0 9 G 3/20		G 0 9 G 3/20	U 5 C 0 8 2
5/00	5 1 0	5/00	5 1 0 G 5 E 5 0 1

審査請求 未請求 請求項の数 8 書面 (全 12 頁)

(21) 出願番号 特願平10-377798

(22) 出願日 平成10年12月10日 (1998. 12. 10)

(71) 出願人 597043408

株式会社インターナショナル インテリジ  
エント インフォメーション  
高知県高知市帯屋町1丁目9番6号

(72) 発明者 山口 人生

神奈川県平塚市土屋2946番地 神奈川大学  
理学部情報科学科内

(74) 代理人 594191548

山口 人生

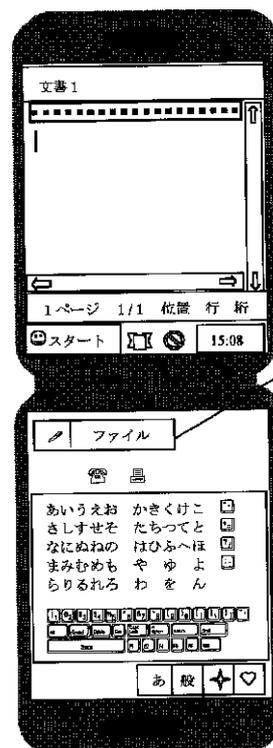
最終頁に続く

(54) 【発明の名称】 携帯情報端末向けのコンパクトGUI

(57) 【要約】 (修正有)

【課題】 小さな携帯情報端末や、スマートフォン向けコンパクトGUIシステムを構築する。

【解決手段】 メニューバー (+ ツールバー) 及び各メニューを開いたコマンド一覧表や各項目を開いたダイアログボックスの位置を開いたコンテンツ表示部を動かすことなく自由に移動でき、かつ、コンテンツ表示部をメニューバー (+ ツールバー) やコマンド一覧表及びダイアログボックスの上に重ねて表示できる、またこの逆も可能となる。さらにそれらをそれぞれ縮小、拡大できる。同一ソフトで異なる二つの画面を開いた場合、それぞれに対応する同一のメニューバー (+ ツールバー) は別々に表示されるのではなく一つに統合して表示される。対象とする携帯情報端末が二つの独立したディスプレイ部を持つ場合、メニューバー (+ ツールバー) をコンテンツ表示部画面が開かれているディスプレイとは別のディスプレイに分割して表示できる。



6

## 【特許請求の範囲】

【請求項1】本発明のコンパクトGUIモデルでは、各応用ソフトを開いた時、通常、画面の上部に表示されている"メニューバー(+ツールバー)"を、その画面から独立させて表示する。つまり、各応用ソフトを開いた時、"メニューバー(+ツールバー)"画面と(従来の画面では、それ以外の部分にあたる)"コンテンツ表示部"画面が、ディスプレイ上で別画面として、同時に開かれることになる。(以後、"ツールバー"が標準で表示されるソフトと、"ツールバー"が標準で表示されないソフトを、まとめて同時に議論するために、"メニューバー(+ツールバー)"という表現を用いる。議論の力点はあくまでも、"メニューバー"に置かれている点に注意せよ。また、"メニューバー"の代役として、例えば、"メニューボックス"や"メニューアイコン"、その他の"メニュー"表示形式を考慮することができる。本発明の各請求項目は、それらの一般的な表示法に適用できる汎用的なものである。ただ、議論の判り易さのため、従来の"メニューバー形式"を基準に話を進める。)

【請求項2】さらに、本発明のコンパクトGUIでは、"メニューバー(+ツールバー)"を(、同時に開いた"コンテンツ表示部"を動かすことなく、)ディスプレイ上で、独立、且つ、自由に移動させることができる。また、各メニューを開いた時の"コマンド一覧表"や、各コマンドを開いた時の"ダイアログボックス"の位置も(、同時に開いた"コンテンツ表示部"を動かすことなく、)自由に移動できるようにする。(独立移動性)

【請求項3】その結果、特に、"コンテンツ表示部"やその他の(ソフトを開いた)画面を"メニューバー(+ツールバー)"や"コマンド一覧表"や"ダイアログボックス"の上に重ねて表示できる。また、その逆も可能になる。(相互重合性)

【請求項4】さらに"メニューバー(+ツールバー)"や"コマンド一覧表"や"ダイアログボックス"をそれぞれ、自由に縮小、拡大できるようにする。(独立縮小・拡大性)

【請求項5】さらに、同一ソフト(、例えばインターネットのブラウザソフト、)で異なる二つ(複数)の画面を同時に開いた場合、それぞれに対応する同一の"メニューバー(+ツールバー)"は、それぞれ二つ(複数)が別々に表示されるのではなく、一つの合体"メニューバー(+ツールバー)"として表示される。言い換えれば、同じソフトで、別々の画面を二つ(以上)同時に開いても、同じ一つの"メニューバー(+ツールバー)"が両者(それら)を共通に制御できるようにする。(統合表示性)

【請求項6】この統合表示性の一例として、具体的には、次のようなGUIモデルを考慮することができる。

(1)あるソフトE(例えば、インターネットブラウザ

ー)をディスプレイ上で初めて開く時には、集合S(E)={ "Eメニューバー(+ツールバー)"、"Eコンテンツ表示部" }が、"初期状況"で同時に表示される。

(2)これらの画面を適宜使用する。その結果、集合S(E)内の各要素の表示形式、及び、それらの表示順序等の関係は初期状況から変移する。さらに、この間、場合によっては、別のソフトFを開くことで、ディスプレイ画面上で、集合S(E)が集合S(F)の下にくるような事態が生じるかもしれない。このような一般的な状態を仮に、"現状況"<sub>n</sub>と呼ぼう。

(3)現状況<sub>n</sub>で、同じソフトEから、さらに第二の画面を開くとする。この時、

3-1. 集合S(E)内の"Eメニューバー(+Eツールバー)"がディスプレイ画面上、最前部に(その位置のまま)浮上してくる。

3-2. この"Eメニューバー(+Eツールバー)"に、第二画面で生成される、新たな"Eコンテンツ表示部2"が新要素として追加される。

20 3-3. その結果、2つの要素{"Eメニューバー(+Eツールバー)"、"Eコンテンツ表示部2"}からなる集合S(E)(2)が生成される。

3-4. S(E)(2)内の各要素の表示順は、"Eコンテンツ表示部2"が最前部になる。

3-5. この結果、最初"Eメニューバー(+Eツールバー)"と組んでいた"Eコンテンツ表示部"は単体でもとの場所に取り残されることになる。この結果の状況を<sub>n+1</sub>としよう。

30 (4)さらに別の操作を続けて、一般の状況<sub>m</sub>になったとする。ここで、依然として、S(E)(2)と先の"Eコンテンツ表示部"は(画面の重なりの意味で)離れているものと想定する。この<sub>m</sub>もとで、"Eコンテンツ表示部"を使用したい時は、その"Eコンテンツ表示部"画面をタッチすればよい。この時、"Eコンテンツ表示部"はディスプレイ上、最前部に浮上し、その真下に"Eメニューバー(+Eツールバー)"が浮上して、もとの集合S(E)を再構成する。

40 (5)最後に、画面上の単一の"Eメニューバー(+ツールバー)"が、残りの(二つ以上ある同一種類画面のうち)どの"Eコンテンツ表示部X"画面を制御しているのかを一意に決定できねばならない。

(6)これの決定は簡単である。以上の設計により、"Eメニューバー(+ツールバー)"とペアを組んでいる"Eコンテンツ表示部X"画面は常に一意決定されている。("Eメニューバー(+ツールバー)"の直前か直後にある。)この画面を制御するように設計すればよい。

50 【請求項7】特に、対象とする携帯情報端末が、二つの独立したディスプレイ部を持つ、折畳式のモデルの場合には、"メニューバー(+ツールバー)"部分を、"コ

コンテンツ表示部"画面が開かれているディスプレイとは別のディスプレイの方へ分割して表示させる機能を実現できる。その結果、付随した"コマンド一覧表"や"ダイアログボックス"も別画面のほうに表示される(分割表示性)(特願平9-279300"新携帯情報端末:モデル1"を参照せよ。)

【請求項8】コンパクトGUIの視点から、"メニューバー(+ツールバー)"のスマートで知的な表示法の問題が生じる。いかにして、スペースを小さく、しかも、見やすく、使いやすくするかという点は、別の特許になる。ここでは、従来のように、"メニューバー(+ツールバー)"を初期画面に表示するのではなく、「"メニューショウ"を初期画面上で左上隅に表示する。」という、究極の最小化を我々のコンパクトGUIとして提案する。この"メニューショウ"とは、一メニュー分の文字数(せいぜい、4~5文字)を記すに足るだけの表示面積を有する電子掲示板のようなものであり、この掲示板の上を、各(区切られて、一ユニットになっている)メニューが右から左へと、一定の速度で流れながら表示される仕組みになっている。(町の電子広告を想起してほしい。)つまり、各メニューは、ある一定の(短い)時間間隔で、このメニューショウ上に(瞬間的に入れ替わるのではなく、流れるように)繰り返し登場する。各メニューを開きたい時は、従来と同様、目的のメニューがメニューショウ上に登場した時に、そのメニューをタッチすればよい。このメニューショウに請求項1~7が適用される。(メニューショウに関する詳しい説明は実施例を参照せよ。)

【発明の詳細な説明】

【0001】

【産業上の利用分野】本発明は、情報通信端末としての携帯情報端末(PDA)の表示画面上で、GUI(グラフィカルユーザインターフェイス)をコンパクトに実現するシステムに関する。

【0002】

【従来の技術】従来、パソコン等で、何らかの応用ソフトを、ディスプレイ上に開いた場合、通常、画面の上部に"メニューバー(+ツールバー)"が表示されていた。(以後、"ツールバー"が標準で表示されるソフト\*

「このメニューバーや一覧表やダイアログボックスは、(開いた表示画面全体を動かさない限り、)画面上で移動させることが出来ない」①

且つ、

「この一覧表やダイアログボックスは、(開いた表示画面全体を縮小・拡大した場合でも、)画面上で縮小や拡大をさせることが出来ない」②

という事実がある。

【0009】

【発明が解決しようとする課題】ところで、従来の携帯型パソコンと比較して、携帯情報端末のディスプレイ部分の表示面積はかなり小さい。このような、小さな表示面積しか持たないディスプレイ上で、同じGUIモデル

\*と、"ツールバー"が標準で表示されないソフトを、まとめて同時に議論するために、"メニューバー(+ツールバー)"という表現を用いる。議論の力点はあくまでも、"メニューバー"に置かれている点に注意せよ。また、"メニューバー"の代役として、例えば、"メニューボックス"や"メニューアイコン"、その他の"メニュー"表示形式を考えることができる。本発明は、それらの一般的な表示法に適用できる汎用的なものである。ただ、議論の判り易さのため、従来の"メニューバー形式"を基準に話を進める。)

10 【0003】例えば、各社のインターネットのブラウザソフトや各社のワープロソフト、表計算ソフト、その他のオフィス用ソフトまで、総てこの「画面の上部に"メニューバー(+ツールバー)"が表示される」というモデルでGUIが統一されていた。

【0004】この事実により、この"メニューバー(+ツールバー)"+"(その他の)コンテンツ表示部"というGUIモデルは、かなり便利なものであり、汎用性があることが見て取れる。

20 【0005】実際、パソコンばかりでなく、従来の携帯情報端末でも、この"メニューバー(+ツールバー)"+"コンテンツ表示部"GUIモデルが基本的に採用されている。

【0006】多分、今から流行り出すであろう、スマートフォンの世界でも、この基本GUIモデルは採用されるに違いない。

30 【0007】このような状況のもとで、従来の(携帯型)パソコンでは、このようなメニューバー中の任意のメニュー(例えば、"ファイル"メニュー)を開いた時に出現する、各コマンド(例えば、"新規作成"、"開く"、"閉じる"、"名前を付けて保存"、"印刷"、"送信"、"プロパティ"等)の一覧表、及び、各コマンドを開いた時に出現する"ダイアログボックス"は、ディスプレイ画面中でかなりの面積を占めていた。そして、当然のことながら、この一覧表やダイアログボックスの下になったコンテンツ表示部は、ユーザの目から隠されて、ユーザからは見えなくなっていた。

【0008】ここで、大事な点として、従来のGUIでは、

「このメニューバー(+ツールバー)を表示した場合、各メニューを開いた時のコマンド一覧表や各項目を開いた時のダイアログボックスは、(たとえ、パソコン向けGUIと比べて、簡易化されているとしても、)ディスプレイ部分の大部分を占め、ほとんどのコンテンツ部分がユーザからは見えなくなるという事

50

態が生じる。

【0010】つまり、パソコンの場合と比較して、携帯情報端末では、ディスプレイ画面全体に対する、コマンド一覧表やダイアログボックスで隠される部分の相対的な割合が増加する。場合によっては、ディスプレイ画面一杯に広がる場合もある。この結果、GUIとしての不便さが増すことになる。

【0011】このような事態を避けるため、まず、メニューバーやコマンド一覧表やダイアログボックスで使用する文字やアイコンを相対的に小さくするという手段をとるのが普通であろう。しかしながら、使用する各文字やアイコンの大きさに関しては、これを人が見る限り、ある一定以上のサイズは確保しておかねばならない。つまり、ディスプレイのサイズが小さくなったからといって、それに比例して文字サイズを小さくすることは不可能であるという現実がある。たとえ、無理に縮小しても、ユーザインターフェイスの観点から、不適切になる。一瞥して、読み難くなるのだ。

【0012】次に、(文字は"GUIの意味で極小"の大きさのまま、)画面全体に占めるメニューバーやコマンド一覧表やダイアログボックス部分の割合をできるだけ小さくすることを試みるのが常套手段であろう。

【0013】このためには、(もう文字は小さくできないのであるから、)例えば、従来の一覧表やダイアログボックスをさらに多重に階層化して、小出しに表示するという手段が採用できるかもしれない。(具体的に、どのような区分けで階層化するかを考えれば、これは中々の難問であることが判る。)ただし、(もし出来たとしても、)この階層化により、GUIとしての便利さは減少するという点を注意せねばならない。つまり、目的の項目がどこにあるのかを探るのが大変になる。

【0014】また、ソフト自体を簡易化して、メニューの数や各一覧表のコマンド数やダイアログボックスの内容を出来るだけ減らすという戦術もある。これは、従来の携帯情報端末用GUIで採用されている戦術である。しかし、これにも限度がある。そもそも、この戦術\*

「たとえ簡易ソフトでも、(ある種のメニューに対応した)コマンド一覧表やダイアログボックスは、いくらでも長く、大きくなる場合がある」③

のだ。

【0020】以上の洞察により、次のことが理解できる。つまり、従来のGUIを、従来の基本思想のまま、画面の小さな携帯情報端末向けに改変して移植すると、インターフェイスの"感性"、及び、"便利さ"の意味で、欠点を抱え込むことになる。

【0021】我々は、ソフトウェア、特にGUI部分の改変によって、以上述べたような基本的な欠陥を克服したい。

【0022】ところが、一方において、本モデルがあまりにも斬新すぎて、従来のGUIモデルとの整合性がまったく取れないようでは困る。特に、従来の応用ソフト

\*は、ソフトとしての諸機能の多様性を犠牲にして成り立つ手段であり、そうそう気安くは使えない。つまり、この場合はGUIの便利さとソフトの諸機能とのバスターになるわけである。

【0015】以上の意味で、従来のGUIモデルに基づく、コマンド一覧表やダイアログボックス部分の表示面積の極小化は、便利さと不便さのバスターの意味で、すでに行き着くところまで行っている。

【0016】これが、今から流行るであろう、スマートフォンや小型携帯情報端末のように、ディスプレイ画面がさらに小さい場合だと、不便さはさらに増すことになる。つまり、GUIとしての"メニューバー(+ツールバー)"方式の便利さと、不便さが逆転しかねない事態になる。

【0017】さらに言えば、従来の携帯情報端末でも、同じソフトで、異なる画面を二つ以上同時に開く機能を装備する場合がある。この場合、従来のGUIでは、"メニューバー(+ツールバー)"がそれぞれの画面に付属している関係上、その画面の表示場所に、その画面の数だけ出現した。このような状況で、各々のコマンド一覧表やダイアログボックスを開けば、デメリットは、ますます悪化することが目に見えている。

【0018】つまり、一画面をディスプレイ上で精一杯大きく開いて、やっとメリットとデメリットの均衡が取れるような、ギリギリのGUIを何とかして採用できたとしても、同時に二画面の表示では、どうしようもなくなる。

【0019】その上、例えば、インターネットのブラウザソフトで不可欠の機能である、"お気に入り"(良く使うホームページの登録機能)メニューを、簡易化ソフトで省略するわけにはいかないであろう。このメニューのコマンド一覧表は、御存知のように、登録されたホームページを区分けしたフォルダの数だけ生成され、且つ、各項目を開くと、そのフォルダに登録されたホームページの名称一覧ダイアログボックスが生成される。つまり、

ウェアが、まったく再利用できないようでは困る。具体的に言えば、"メニューバー(+ツールバー)"、もしくは、その変形を使用するという原理だけは、踏襲したい。

【0023】以上の思想に基づき、本特許では次のような新しいGUIシステムを採用する。

【0024】

【課題を解決するための手段】上記目的を達成するために、本発明のコンパクトGUIモデルでは、各応用ソフトを開いた時、通常、画面の上部に表示されている"メニューバー+(ツールバー)"を、その画面から独立させて表示する。

【0025】つまり、各応用ソフトを開いた時、"メニューバー(+ツールバー)"画面と(従来の画面では、それ以外の部分にあたる)"コンテンツ表示部"画面が、(比較的小さな液晶)ディスプレイ上で別画面として、同時に開かれることになる。

【0026】一枚目の画面を開いた時、最初に表示される"メニューバー(+ツールバー)"のディスプレイ上\*

「開いた"メニューバー(+ツールバー)"と"コンテンツ表示部"を、ディスプレイ上で、それぞれ独立、且つ、自由に移動させることができる。」④(独立移動性)

【0029】さらに、各メニューを開いた時の"コマンド一覧表"や、各項目を開いた時の"ダイアログボックス"の位置も(、同時に開いた"コンテンツ表示部"と

「"コンテンツ表示部"やその他の(ソフトを開いた)画面を"メニューバー(+ツールバー)"や"コマンド一覧表"や"ダイアログボックス"の上に重ねて表示できる。また、その逆も可能になる。」⑤(相互重合性)

一言で言えば、独立画面相互の重なり具合は、自由にできるように設計する。

「"メニューバー(+ツールバー)"や"コマンド一覧表"や"ダイアログボックス"をそれぞれ、自由に縮小、拡大できるようにする。」⑥(独立縮小・拡大性)

(従来のGUIでは、上記②の通りである。)

【0032】以上が、"メニューバー(+ツールバー)"を"コンテンツ表示部"から独立させるという意

「同一ソフト(、例えばインターネットのブラウザソフト、)で異なる二つの画面を同時に開いた場合、それぞれに対応する同一の"メニューバー(+ツールバー)"は、それぞれ二つが別々に表示されるのではなく、一つの合体"メニューバー(+ツールバー)"のみが表示される。」⑦(統合表示性)

言い換えれば、同じソフトで、別々の画面を二つ(以上)同時に開いても、同じ一つの"メニューバー(+ツールバー)"が両者(それら)を共通に制御できるようにする。(従来は、開いた画面の数だけ、"メニューバー(+ツールバー)"が表示された。)

【0034】第五の機能は、携帯情報端末の装置としての形状、すなわち、ハードウェアに依存する。上記④～

「"メニューバー(+ツールバー)"部分を、"コンテンツ表示部"画面が開かれているディスプレイとは別のディスプレイの方へ分割して表示させる機能を実現できる。」⑧(分割表示性)

(特願平9-279300"新携帯情報端末:モデル1"を参照せよ。)

これは、同一画面上での自由な移動を保証した、上の第一機能④(独立移動性)とは本質的に異なるという事実注意到注意せよ。

【0036】上記④～⑧の五機能がシステム特許の意味で新規性・進歩性を有していることは明白である。これらの諸機能を実現するための、ソフトウェアの具体的な内容に関しては、著作権の問題になり、ここでは詳述しないし、する必要もない。要は、この道の専門家として、以上5つの新機能が実現可能かどうかを保証すればよからう。

\*の位置と"コンテンツ表示部"のディスプレイ上の位置は、ほぼ従来通りとする。つまり、上下にくっ付いた形で表示される。

【0027】それでは、我々の新GUIと従来のGUIとでは、実質的にどこが違うのか?以下の点が本質的に違うのである。

【0028】第一に、我々の新GUIでは、  
「開いた"メニューバー(+ツールバー)"と"コンテンツ表示部"を、ディスプレイ上で、それぞれ独立、且つ、自由に移動させることができる。」④(

独立に、)自由に移動できるようにする。(従来のGUIでは、上述①の通りである。)

【0030】第二に、その結果、特に、  
「"コンテンツ表示部"やその他の(ソフトを開いた)画面を"メニューバー(+ツールバー)"や"コマンド一覧表"や"ダイアログボックス"の上に重ねて表示できる。また、その逆も可能になる。」⑤(相互重合性)

【0031】第三に、

味である。さらに、次のような本質的な効果をあげることが可能になる。

【0033】第四に、

「同一ソフト(、例えばインターネットのブラウザソフト、)で異なる二つの画面を同時に開いた場合、それぞれに対応する同一の"メニューバー(+ツールバー)"は、それぞれ二つが別々に表示されるのではなく、一つの合体"メニューバー(+ツールバー)"のみが表示される。」⑦(統合表示性)

⑦の4つの新機能は、携帯情報端末のディスプレイ部分が小さいという前提だけで有用な機能であった。それに対し、第五の機能は、対象とする携帯情報端末が、二つの独立したディスプレイ部を持つ、折畳式のモデルの場合に実現する。

【0035】このようなモデルの携帯情報端末では、

【0037】というわけで、この場で、プロとしての保証をしておこう。「上記の(コンパクトGUI向け、)新しい5機能は、総て実現可能である事実をここに保証する。」以上で発明の開示は終わった。

【0038】

【作用】上記のように構成された新機能を使用する場合、まず、第一機能④に関しては、これ以上追加する注意事項はない。

【0039】第二機能⑤に関してであるが、例えば、"メニューバー(+ツールバー)"が"コンテンツ表示部"の背後にあり、"メニューバー(+ツールバー)"の一部分だけが画面上に顔を出しているような状況を想

定せよ。この場合、顔を出している "メニューバー (+ ツールバー)" の一部をタッチすると、"メニューバー (+ ツールバー)" が自動的に "画面の最上部、且つ、画面の最前部" に移動し、しかも、タッチされた個別メニューの "コマンド一覧表" が開かれるように設計する。つまり、後ろに隠されている "メニューバー (+ ツールバー)" を使用する際、一度、画面の最上部、且つ、最前部へ (タッチで) 移動さす必要はない。

【0040】これに関連して、次のような気になる事態が生じる。従来の GUI では、使いたい画面 A をタッチすれば、その画面の "Aメニューバー (+ ツールバー)" + "Aコンテンツ表示部" がワンセットで最前面に出てきた。ところが、我々のコンパクト GUI では、"メニューバー (+ ツールバー)" と "コンテンツ表示部" は独立している。この状況のもとで、例えば、別画面 B の背後にある、"Aコンテンツ表示部" 画面を使用 \*

$S(A) = \{ "Aメニューバー (+ ツールバー)", "Aコンテンツ表示部" \}$  として取り扱う。」 ⑨

(場合によっては、{ "Aメニューバー"、"Aツールバー"、"Aコンテンツ表示部" } と 3 要素に分解しても良い。しかし、以下の議論は 2 要素分解を基準に話を進めていく。それで話の根本が変化することはない。)

【0043】よって、別のソフト B を開いて、同じディスプレイ上に、集合

$S(B) = \{ "Bメニューバー (+ ツールバー)", "Bコンテンツ表示部" \}$

が追加表示された場合、 $S(A)$  と  $S(B)$  の要素が、"表示画面の重なり" の意味で交わり合うことはない。

【0044】そして、例えば、 $S(A)$  が  $S(B)$  より、"表示画面の重なり" の意味で背後にある時、 $S(A)$  の任意の要素をタッチして、画面の最前部へ移行した時、(集合として、まとまって移動するのであるから、)  $S(A)$  の他の要素も同時に前面へ浮上してくることになる。

【0045】それでは、 $S(A)$  内の 2 要素の内、どちらをタッチしても、(画面前部への集団移行の意味で、) まったく同じ効果を生むのか? 幸い (残念) ながら、事はそれほど単純ではない。では、どう違うのか。

【0046】前後の重なり順が違ってくるのである。つまり、どちらかをタッチすることにより、 $S(A)$  内の 2 要素は同時に最前部に移行してくるが、タッチした要素の方が、2 要素内で、さらに最前部 (画面の一番手前側) として表示される。

【0047】つまり、タッチ以前の  $S(A)$  内の "Aメニューバー (+ ツールバー)" と "Aコンテンツ表示部" の画面上の順序がどのようなものであっても、タッチ後は、タッチされた方が最前部に来るのだ。

【0048】そして、上で述べたように、特に "Aメニューバー (+ ツールバー)" の特定のメニューをタッチした時は、同時にそのメニューが開かれるのである。

\* したいとする。この "Aコンテンツ表示部" をタッチして最前部に持ってきた時、これに対応する、残りの "Aメニューバー (+ ツールバー)" 部はどうなるのかという基本技術に関する疑問である。

【0041】具体的には、例えば、最初、"Aメニューバー (+ ツールバー)" 部が画面 B の背後にある時、"Aコンテンツ表示部" を最前部に移した後も、"Aメニューバー (+ ツールバー)" 部はそのまま、元の位置 (画面 B の背後) に留まるのか? これでは、少しばかり、使い勝手が悪くないか?

【0042】この本質的な疑問に対する解答は次のようなものとなる。我々のコンパクト GUI では、「何らかのソフト A を開いた時、独立して表示される "Aメニューバー (+ ツールバー)" と "Aコンテンツ表示部" は、"理論上"、常に一つのまとまった 2 要素集合

【0049】また、"Aコンテンツ表示部" をタッチした場合は、ディスプレイ画面内でタッチ前の位置のまま、最前部へ浮かび上がってくることになる。この時、"Aメニューバー (+ ツールバー)" の画面内の位置は変化せず、開かれもせず、そのまま、"Aコンテンツ表示部" の真下へ浮上してくることになる。

【0050】第三機能⑨に関しては、一つだけ作用上の注意をしたい。"メニューバー (+ ツールバー)" は "コンテンツ表示部" やその他の画面の下に隠せるのであるから、"メニューバー (+ ツールバー)" の縮小機能は、実用上、役に立たないと思うかもしれない。しかしながら、現実はそうではない。縮小した "メニューバー (+ ツールバー)" を "コンテンツ表示部" 上の任意の位置に重ねる場面を想定してみよ。ましてや、"コマンド一覧表" や "ダイアログボックス" の場合は、これを開いた後で、縮小できることのメリットは、(従来の GUI が、一覧表やダイアログボックスを縮小出来なかったという事実を思い起こせば、) 大きい。

【0051】さて、第四機能⑩実現のためには、そもそも、いかにして "メニューバー (+ ツールバー)" を単一のものに統合するかという点が重要になってくる。これの解決のために、次のような手段を採用する。

【0052】(1) あるソフト E (例えば、インターネットブラウザ) をディスプレイ上で初めて開く時には、集合  $S(E) = \{ "Eメニューバー (+ ツールバー)", "Eコンテンツ表示部" \}$  が、"初期状況" で同時に表示される。

(2) これらの画面を適宜使用する。その結果、集合  $S(E)$  内の各要素の表示形式、及び、それらの表示順序等の関係は初期状況から変移する。さらに、この間、場合によっては、別のソフト F を開くことで、ディスプレイ画面上で、集合  $S(E)$  が集合  $S(F)$  の下にくるよ

うな事態が生じるかもしれない。このような一般的な状態を仮に、"現状"<sub>n</sub>と呼ぼう。

(3) 現状<sub>n</sub>で、同じソフトEから、さらに第二の画面を開くとする。この時、

3-1. 集合S(E)内の"Eメニューバー(+ツールバー)"がディスプレイ画面上、最前部に(その位置のまま)浮上してくる。

3-2. この"Eメニューバー(+ツールバー)"に、第二画面で生成される、新たな"Eコンテンツ表示部2"が新要素として追加される。

3-3. その結果、2つの要素{"Eメニューバー(+ツールバー)"、"Eコンテンツ表示部2"}からなる集合S(E)(2)が生成される。

3-4. S(E)(2)内の各要素の表示順は、"Eコンテンツ表示部2"が最前部になる。

3-5. この結果、最初"Eメニューバー(+ツールバー)"と組んでいた"Eコンテンツ表示部"は単体でもとの場所に取り残されることになる。この結果の状況を<sub>n+1</sub>としよう。

【0053】さらに別の操作を続けて、一般の状況<sub>m</sub>になったとする。ここで、依然として、S(E)(2)と先の"Eコンテンツ表示部"は(画面の重なりの意味で)離れているものと想定する。この<sub>m</sub>もとで、"Eコンテンツ表示部"を使用したい時は、その"Eコンテンツ表示部"画面をタッチすればよい。この時、"Eコンテンツ表示部"はディスプレイ上、最前部に浮上し、その真下に"Eメニューバー(+ツールバー)"が浮上して、もとの集合S(E)を再構成する。

【0054】最後に、画面上の単一の"Eメニューバー(+ツールバー)"が、残りの(二つ以上ある同一種類画面のうち)どの"Eコンテンツ表示部X"画面を制御しているのかを一意に決定できねばならない。

【0055】これの決定は簡単である。以上の設計により、"Eメニューバー(+ツールバー)"とペアを組んでいる"Eコンテンツ表示部X"画面は常に一意決定されている。("Eメニューバー(+ツールバー)"の直前か直後にある。)この画面を制御するように設計すればよい。

【0056】第五機能⑧(分割表示性)に関して、別画面への移動はペン等で操作するのではない点を強調しておきたい。つまり、第五機能⑧とは、各応用ソフトを開いた時点で、"メニューバー(+ツールバー)"が"コンテンツ表示部"を表示しているディスプレイとは別のディスプレイ上に表示されるという意味である。

【0057】また、上の⑨で、集合S(A)を規定した際、"理論上"という言葉を用いておいた。これは、第五機能⑧を使用する場合、本質的に生きてくる。つまり、⑧により、"Aメニューバー(+ツールバー)"と"Aコンテンツ表示部"は別々のディスプレイ上に表示されることになる。そのような場合でも、上述の議論

【0039】~

【0055】は、対応する適当な変形を施せば、そのまま成立することは明白である。(例えば、片方のディスプレイで(タッチにより)"Aコンテンツ表示部"を最前部に移行させると、他方のディスプレイで"Aメニューバー(+ツールバー)"が、自動的に最前部に移るのである。)

【0058】さて、以上は、何らかのソフト(、例えば、インターネットブラウザ)を固定した時の"メニューバー(+ツールバー)"の表示システムに関する話であったが、同様の機能は、その他の("メニューバー(+ツールバー)"を有するOS内の、または、応用)ソフトについても発現できる。例えば、ワープロソフトに関しても、"メニューバー"と"ツールバー"は存在するし、OS内のコントロールパネルにも"メニューバー"は存在する。この場合にも"メニューバー(+ツールバー)"を独立させるように機能設計をするのである。

【0059】この時、当然のことながら、一目で、インターネット用の"メニューバー(+ツールバー)"とワープロ用の"メニューバー(+ツールバー)"が区別できなければ使い物にはならない。この区別付けは簡単である。上述のように、各ソフトを開いた場合、基本的にはセットになって表示されるという事実を考えれば、混同の心配はない。

【0060】但し、原理上の話と、GUIとしての"見易さ"の話題は別である。このため、例えば、インターネット用の"メニューバー(+ツールバー)"にはインターネット印を、ワープロ用の"メニューバー(+ツールバー)"にはワープロ印を押しておけばよい。

【0061】インターネット用の"メニューバー(+ツールバー)"とワープロ用の"メニューバー(+ツールバー)"をさらに統一して、一つの"メニューバー(+ツールバー)"にまとめるようなまねはしない。これは、両者の基本機能が異なるからである。なるほど、両者で重なり合っている機能も幾つかはある。しかし、合体させることで生じる、全体的な煩わしさ、使用上の不便さは、スペース節約のメリットを台無しにするからである。

【0062】このような、"メニューバー(+ツールバー)"の独立が、我々が提唱する、コンパクトGUI向けの、新OSの改変だけで可能になるのか、それとも各種応用ソフト自身をいじらねばならないのかに関しては、従来のOSと従来の各種応用ソフトの相互関係に依存する。(最近、どこまでがOSで、どこから応用ソフトなのかを区別することが難しくなっているが。)しかしながら、我々は従来のOSを改変して、携帯情報端末に特化したコンパクトな新OSを設計するのである。この新OSに乗るように、従来の応用ソフトを改変するのが自然な姿であろう。

【0063】従来のパソコン用の応用ソフトを、そのまま携帯情報端末用に使用することは、まずない。これは、容量及び使い勝手の問題が絡むからである。(「携帯」は「簡易性が命」である。ウィンドウズCE上の「ポケットワード」を見よ。) どうせ改変するなら、その際、我々の新機能を盛り込もうという思想である。

【0064】実は、本特許は、折り畳み式携帯情報端末向けの分割表示性⑧の機能を申請するのが本来の目的であった。しかしながら、⑧のみを独立させて申請するよりも、⑧へ至るまでの途中の機能④~⑦も、一般の(ディスプレイ画面の小さな)携帯情報端末向けに、独立させて同時に申請しておくのが、我々の特許戦略上、有利と判定した。

【0065】さらに言えば、④~⑦の新機能は、それぞれ汎用性のある、「GUIコンパクト化特許」として申請できるコンセプトである。つまり、それぞれが、通常のキーボード使用パソコンのソフトにも、スペース節約\*

「メニューショウ」を初期画面上で左上隅に表示する。」 10

という、究極の最小化を我々のコンパクトGUIとして提案する。

【0069】この「メニューショウ」とは、一メニュー分の文字数(せいぜい、4~5文字)を記すに足るだけの表示面積を有する電子掲示板のようなものであり、この掲示板の上を、各(区切られて、一ユニットになっている)メニューが右から左へと、一定の速度で流れながら表示される仕組みになっている。(町の電子広告を想起してほしい。)つまり、各メニューは、ある一定の(短い)時間間隔で、このメニューショウ上に(瞬間的に入れ替わるのではなく、流れるように)繰り返し登場する。

【0070】各メニューを開きたい時は、従来と同様、目的のメニューがメニューショウ上に登場した時に、そのメニューをタッチすればよい。

【0071】このメニューショウに対して、上記④~⑦+⑧の機能が設定されることになる。つまり、メニューショウの自由な移動、拡大・縮小、他の画面との重ね合わせ、統一化、別画面表示が可能になるように設計する。

【0072】メニューショウを拡大した時、当然、一度に表示できるメニュー数は増加する。それらの、複数のメニューが、やはり、その拡大枠の中で、右から左へ一定速度で流れながら、表示される。当然、左へ消えたメニューは、一定時間後、右から登場する。(丁度、複数メニューの乗ったベルトコンベヤーのような感じである。)

【0073】最終的に、総てのメニューが同時にメニューショウ上に表示できる程、メニューショウの枠が拡大された時点で、この流れは自動的に停止する。

【0074】メニューショウ枠の拡大方向は自由にとれる。つまり、メニューショウの枠組みが、横一列である

\* 機能として適用できる。

【0066】これに関連して、コンパクトGUIの視点から、「メニューバー(+ツールバー)」のスマートで知的な表示法の問題が生じてくる。いかにして、スペースを小さく、しかも、見やすく、使いやすくするという点は、別の特許になる。以下の実施例で、具体例を一つだけ挙げておこう。

【0067】

【実施例】実施例について図面を参照して説明すると、図1は我々が提唱するコンパクトGUIを携帯情報端末上で実現した一例である。この図では、何らかの応用ソフト(例えば、簡易ワープロ)をキーボード付きの携帯情報端末で開いた直後の表示画面の状態を表わしている。

【0068】ここでは、従来のような、「メニューバー(+ツールバー)形式」が初期画面に表示されるのではなく、

必要はない。複数列に開かれた場合は、各列毎に、右から左への独立した流れが生じるように設計する。(各列のメニュー配分は自動とする。全メニューを、ほぼ、その列数に分ければ良い。)

【0075】各メニューを開いた時のコマンド一覧表や、各コマンドを開いた時のダイアログボックスについても、同様の流れを生じさせる。つまり、一度に全情報が表示出来ない(程度の枠組みの大きさの)時には、その枠内で流れを作る。

【0076】なお、この流れの速度は各ユーザが自由に設定できるようにする。

【0077】また、場合によっては、(ユーザが)流れの方向を、左右ではなく、上下(下から上)に変更できるように設計する。

【0078】このアイデアの新規性・進歩性について一言。この機能は、従来の、例えば、電子メールの自動スクロール機能とは、本質的に異なる点に注意せよ。そもそも、電子メール画面はコンテンツ表示部の話である。それに、なによりも、メニューショウでは、流れている各メニューにタッチすると、その命令が実行される。つまり、流れている実体は、単なる情報ではなく、命令なのだ。

【0079】図2は、メニューショウの枠組みを、もとの場所で、最大の大きさ(全メニューが同時に表示されて、流れが生じない大きさ)に拡大した際の様子である。このためには、先頭のアイコンをタッチすればよい。(アイコン部分は流れない。)このアイコン(ペン模様)は、このメニューショウが、今開かれているワープロソフトのメニューショウであることを示す。

【0080】図3は、新機能④(独立移動性)、⑤(相互重合性)、⑥(独立縮小・拡大性)を用いて、このメニューショウを、ディスプレイ内で自由に移動し、且

つ、縮めた結果を表わす。この図では画面右下部に移動して、縮小されている。

【0081】図4は、新機能⑤（相互重合性）により、メニューショウがコンテンツ表示部の背後に隠された場合のディスプレイ画面を表わしている。この図では、メニューショウはコンテンツ表示部で完全に隠されておらず、その一部が、コンテンツ表示部の背後から顔を出している状態を表わしている。

【0082】図5は、新機能⑦（統合表示性）により、一つのメニューショウに対し、二つのコンテンツ表示部が開かれている状態を示している。二つのコンテンツ表示部はディスプレイ上で、左右に並んで表示されている。この両者の前面にメニューショウが開かれている。

【0083】図6は、我々が提唱する"モデルI"という両開きのディスプレイを有する新携帯情報端末上で、機能⑧（分割表示性）を実施した例である。図中、上側のディスプレイにコンテンツ部が表示され、下側のディスプレイにメニューショウが表示されている。

【0084】この実施例では、メニューショウが表示されたディスプレイ上には、それ以外に、キーボードのシミュレート機能画面と、その他のアイコン（このアイコンの具体的な機能に関しては、ここでは話題にしない。）が二つ表示されている。一方、上側のディスプレイでは、開いたソフトのコンテンツ部が画面一杯に表示されている。

【0085】

【発明の効果】本発明は、以上説明したように構成されているので、以下に記載されるような効果を奏する。

【0086】本システムを採用したコンパクトGUIモデルは、携帯情報端末のディスプレイ画面が（、従来の画面と比較して）小さな時、感性上も実用上も、メリットを有する。具体的には

【0087】第一に、我々の新GUIでは、"メニューバー（+ツールバー）"の位置を（、開いたコンテンツ表示部を動かすことなく、）ディスプレイ上で、独立、且つ、自由に移動させることができる。（独立移動性）

【0088】さらに、各メニューを開いた時の"コマンド一覧表"や各項目を開いた時の"ダイアログボックス"の位置も（、開いたコンテンツ表示部を動かすことなく、）自由に移動できる。これにより、コンテンツ部の内容が観やすくなり、新GUIとしてのメリットが増す。

【0089】第二に、"コンテンツ表示部"やその他の（ソフトを開いた）画面を"メニューバー（+ツールバー）"や"コマンド一覧表"や"ダイアログボックス"の上に重ねて表示できる。また、その逆も可能になる。（相互重合性）これにより、コンテンツの表示面積という観点から見た場合、メニューバー（+ツールバー）"部分のスペースが節約できる。つまり、新GUIとしてのメリットが増す。

【0090】第三に、"メニューバー（+ツールバー）"や"コマンド一覧表"や"ダイアログボックス"をそれぞれ、自由に縮小、拡大できるようにする。（独立縮小・拡大性）これにより、コンテンツ部の内容が観やすくなり、新GUIとしてのメリットが増す。

【0091】第四に、同一ソフト（、例えばインターネットのブラウザソフト、）で異なる二つの画面を開いた場合、それぞれに対応する同一の"メニューバー（+ツールバー）"は、それぞれ二つが別々に表示されるのではなく、一つの合体"メニューバー（+ツールバー）"のみが表示される。（統合表示性）言い換えれば、同じソフトで、別々の画面を二つ（以上）同時に開いても、同じ一つの"メニューバー（+ツールバー）"が両者（それら）を共通に制御できるようにする。これにより、実質的に何らの不便を被ることなく、コンテンツ部の内容がさらに観やすくなり、新GUIとしてのメリットが増す。

【0092】第五の機能は、対象とする携帯情報端末が、二つの独立したディスプレイ部を持つ、折畳式のモデルの場合に実現する。このようなモデルの携帯情報端末では、"メニューバー（+ツールバー）"部分を、"コンテンツ表示部"画面が開かれているディスプレイとは別のディスプレイの方へ分割して表示させる機能を実現できる。（分割表示性）

（特願平9-279300"新携帯情報端末：モデル1"を参照せよ。）

【0093】これにより得られるコンパクトGUIとしてのメリットは大きい。"メニューバー（+ツールバー）"部分が、別の画面に表示されるという事実は、当然、それに付随した"コマンド一覧表"や"ダイアログボックス"も別画面のほうに表示されるという事実を包含する。つまり、"コマンド一覧表"や"ダイアログボックス"が、どんなに長く大きくなっても、その影響は、"コンテンツ表示部"には出ない。（上述の③を思い起こしてほしい。）

【0094】勿論、"メニューバー（+ツールバー）"を表示するディスプレイ側に、別のソフト（例えば、キーボードのシミュレーション機能）が開かれている場合もある。このような場合、このディスプレイ側で、必要に応じて④~⑦を実行する。このように、別ディスプレイ側でも、必要に応じて④~⑦を実行するのならば、実質的に何が有り難いのか？それは次の点である。

【0095】つまり、一枚パネルのディスプレイの場合、"メニューバー（+ツールバー）"や"コマンド一覧表"や"ダイアログボックス"で隠される部分として、別ソフトの画面のみならず、今、一番見たい、当の、"コンテンツ表示部"までも影響が及ぶ。一方、2枚パネルの折り畳み式ディスプレイの場合、"コンテンツ表示部"に影響は及ばない。この結果、両者では、

50 "必要に応じて"の度合い・頻度が本質的に違ってくる

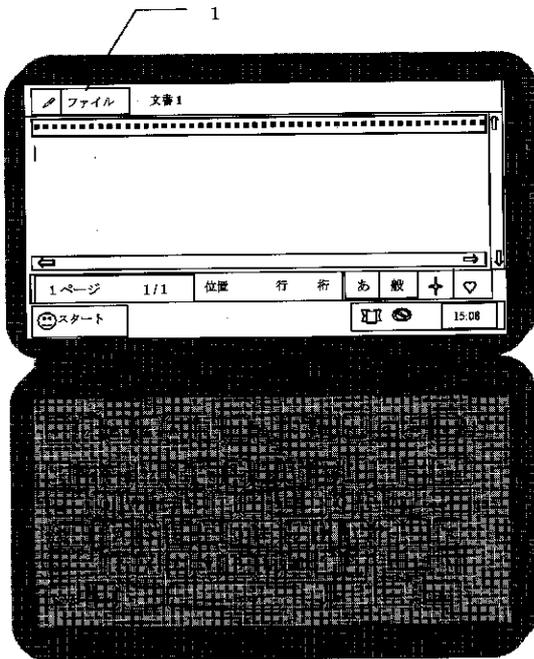
のだ。

【0096】④~⑥は、ユーザが携帯情報端末を使用中、たとえ余分な操作をしても、隠れた部分を見たいという場合に実行されるに決まっている。具体的に、どのような場面で、そのような必要性を感じるかは、各ユーザの感性と組み込みソフト、及び、携帯情報端末の使用状況に依存する。要は、「いざ必要となった時に、ちゃんと用意が整っている」というのが④~⑥のメリットである。一方、⑦は「無駄を省こう」という単純性の思想を实践したものである。また、⑧のメリットは上で述べた。これ、らこそ、「知的インターフェイス」の真髄であろう。

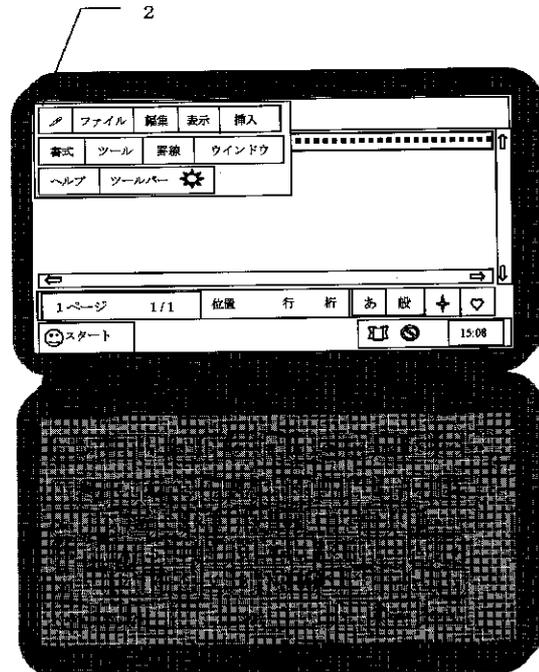
【0097】我々のコンパクトGUIでは、以上のような、従来の常識では考えられないような機能を獲得できる。逆に言えば、そのような新機能をコンパクトGUIとして装備するのである。狭い画面上で、上記④~⑦+⑧を統合して得られる融合機能から得られるメリットは計り知れない。

【図面の簡単な説明】

【図1】



【図2】



\* 【図1】何らかの応用ソフト（例えば、簡易ワープロ）を開いた直後の表示画面の状態。

【図2】メニューショウを拡大表示した一例。

【図3】メニューショウを、ディスプレイ内で自由に移動し、且つ、縮めた結果を表わす。

【図4】メニューショウがコンテンツ表示部の背後に隠された場合。

【図5】一つのメニューショウに対し、二つのコンテンツ表示部が開かれている状態。

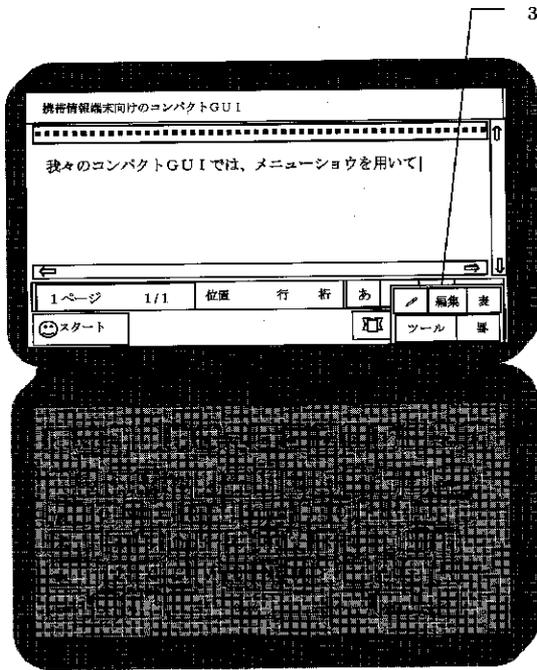
10 【図6】両開きのディスプレイを有する新携帯情報端末上で、機能⑧（分割表示性）を実施した例。

【符号の説明】

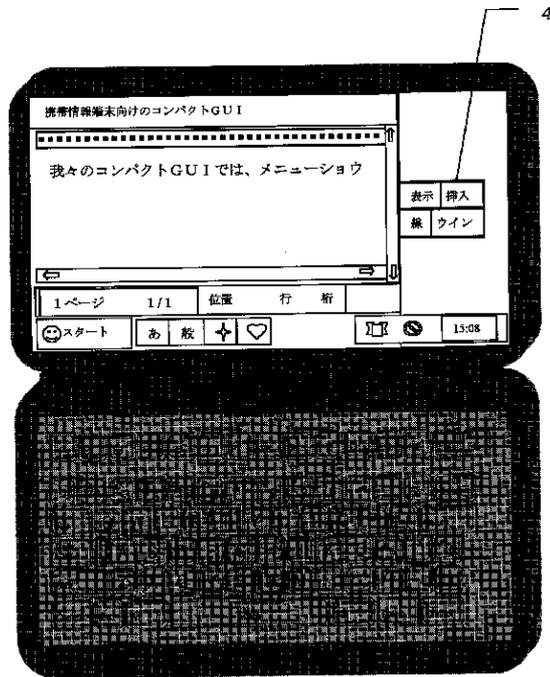
- 1 初期状態のメニューショウ
- 2 最大画面のメニューショウ
- 3 移動・縮小したメニューショウ
- 4 一部顔を出したメニューショウ
- 5 二つの画面が共有する統合メニューショウ
- 6 別のディスプレイ上に表示されたメニューショウ

\*

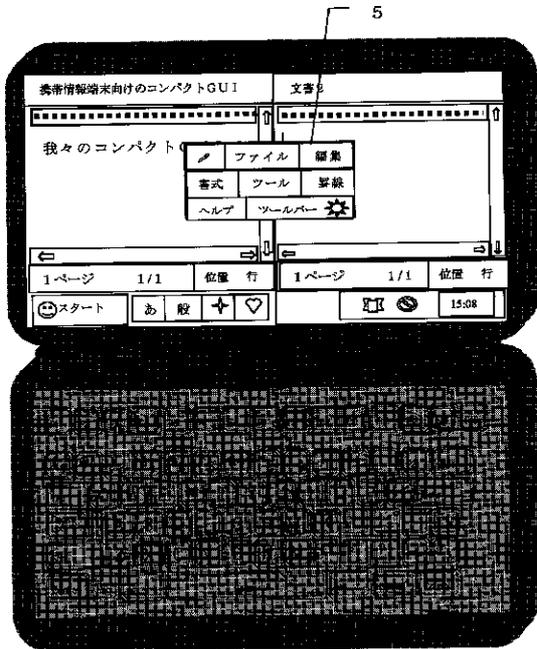
【図3】



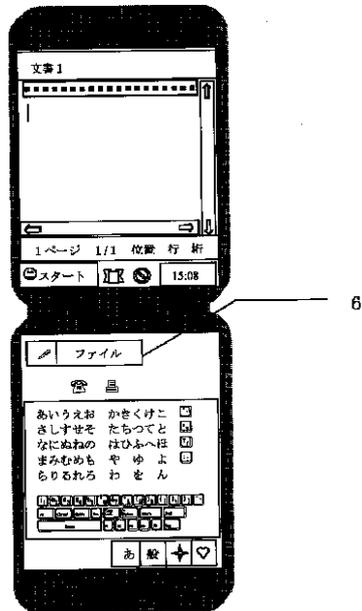
【図4】



【図5】



【図6】



## フロントページの続き

F ターム(参考) 5C080 AA10 BB05 DD01 DD13 EE01  
EE17 FF09 GG02 GG12 JJ01  
JJ06 KK07  
5C082 AA01 AA22 BA02 BA12 BB42  
CA52 CA59 CB05 DA42 DA87  
MM09 MM10  
5E501 AA04 AA14 BA03 CA02 CB05  
EA05 EA10 EA11 EA15 EB01  
EB05 FA05 FA08 FB04 FB22