

(19) 日本国特許庁 (JP)

(12) 特 許 公 報 (B2)

(11) 特許番号

特許第4920154号  
(P4920154)

(45) 発行日 平成24年4月18日 (2012. 4. 18)

(24) 登録日 平成24年2月10日 (2012. 2. 10)

(51) Int. Cl.

F I

G 0 6 F 17/22 (2006. 01)

G 0 6 F 17/22 5 1 0 N

G 0 6 F 17/21 (2006. 01)

G 0 6 F 17/21 5 9 2 J

請求項の数 35 (全 34 頁)

(21) 出願番号 特願2001-535148 (P2001-535148)  
 (86) (22) 出願日 平成12年11月3日 (2000. 11. 3)  
 (65) 公表番号 特表2003-513389 (P2003-513389A)  
 (43) 公表日 平成15年4月8日 (2003. 4. 8)  
 (86) 国際出願番号 PCT/US2000/030465  
 (87) 国際公開番号 W02001/033324  
 (87) 国際公開日 平成13年5月10日 (2001. 5. 10)  
 審査請求日 平成19年11月5日 (2007. 11. 5)  
 (31) 優先権主張番号 60/163, 588  
 (32) 優先日 平成11年11月5日 (1999. 11. 5)  
 (33) 優先権主張国 米国 (US)  
 (31) 優先権主張番号 09/606, 811  
 (32) 優先日 平成12年6月28日 (2000. 6. 28)  
 (33) 優先権主張国 米国 (US)

(73) 特許権者 500046438  
 マイクロソフト コーポレーション  
 アメリカ合衆国 ワシントン州 9805  
 2-6399 レッドモンド ワン マイ  
 クロソフト ウェイ  
 (74) 代理人 100077481  
 弁理士 谷 義一  
 (74) 代理人 100088915  
 弁理士 阿部 和夫  
 (72) 発明者 ジアン ワン  
 中華人民共和国 100081 北京 ハ  
 ンゾウ ズエイアン プロビンス ハン  
 ダ シン チュン (番地なし) ビルデ  
 イング 21 アパートメント 40

最終頁に続く

(54) 【発明の名称】 言語入力ユーザインタフェース

(57) 【特許請求の範囲】

【請求項 1】

ユーザにより入力された入力テキストを入力デバイスにより受け取るステップ、  
 前記入力テキストを、該入力テキストに対応する出力テキストにプロセッサにより変換  
 するステップ、および

前記入力テキストと前記出力テキストを共通の入力行内に表示装置の表示画面上で表示  
 するステップ

を有し、前記変換するステップは、

第 1 の測定基準に基づいて第 1 の候補リスト内の出力テキスト候補を前記プロセッサに  
 より順序付けるステップであって、前記第 1 の候補リスト内の複数の出力テキスト候補は  
 当該複数の出力テキスト候補内の 1 つをポインティングデバイスにより前記出力テキスト  
 として選択することが可能な出力テキストの候補である、順序付けるステップと、

前記第 1 の測定基準と異なる第 2 の測定基準に基づいて第 2 の候補リスト内の出力テキ  
 スト候補を前記プロセッサにより配列するステップであって、前記第 2 の候補リスト内の  
 複数の出力テキスト候補は当該複数の出力テキスト候補内の 1 つをポインティングデバイ  
 スにより前記出力テキストとして選択することが可能な出力テキストの候補である、配列  
 するステップと、

前記第 1 の候補リストおよび前記第 2 の候補リストを前記表示装置の表示画面に表示す  
 るステップ、および

前記表示装置の表示画面に表示されている第 1 の候補リストおよび第 2 の候補リストの

10

20

内の 1 つの出力テキスト候補を前記入力テキストから変換される出力テキストとしてポインティングデバイスにより選択するステップ

を有することを特徴とする方法。

【請求項 2】

前記入力テキストが表音文字テキストを含み、前記出力テキストが文字ベースの言語テキストを含むことを特徴とする請求項 1 に記載の方法。

【請求項 3】

前記入力テキストが中国語ピンインを含み、前記出力テキストが漢字を含むことを特徴とする請求項 1 に記載の方法。

【請求項 4】

前記入力テキストおよび前記出力テキストを表示するステップでは前記入力テキストと前記出力テキストを共通の水平行内に共に表示することを特徴とする請求項 1 に記載の方法。

【請求項 5】

前記入力テキストおよび前記出力テキストを表示するステップでは、出力テキストへ変換する入力テキストの代わりに前記出力テキストを示すことを特徴とする請求項 1 に記載の方法。

【請求項 6】

さらに追加入力テキストが前記入力デバイスにより入力された時に前記出力テキストを別の出力テキストに前記プロセッサにより修正し、前記別の出力テキストは以前に入力された入力テキストおよび追加入力テキストから変換された出力テキストであることを特徴とする請求項 1 に記載の方法。

【請求項 7】

さらにユーザが句読点を前記入力デバイスにより入力したことに応答して追加入力テキストを入力するとき前記出力テキストをさらに修正するのを前記プロセッサにより中止することを特徴とする請求項 6 に記載の方法。

【請求項 8】

さらに以前に修正された前記出力テキストを前記入力デバイスにより確認応答して追加入力テキストを入力するとき前記出力テキストをさらに修正するのを前記プロセッサにより中止することを特徴とする請求項 6 に記載の方法。

【請求項 9】

さらに、変換されていない入力テキストを修正するためにアクティブのまま残しながら、ユーザが前記出力テキストを前記入力デバイスにより確認応答して前記出力テキストをさらに修正するのを前記プロセッサにより中止することを特徴とする請求項 6 に記載の方法。

【請求項 10】

追加入力テキストを入力するとき前記出力テキストを選択的に前記プロセッサにより修正するステップであって、当該修正は予め用意された複数の候補の 1 つを、候補について予め定めた変換の正確度に基づいて選択することにより行われ、予め定めた高い変換の正確度が得られない場合には修正を行わないようにすることを特徴とする請求項 1 に記載の方法。

【請求項 11】

入力モードと編集モードを切り替えずに、前記共通の入力行内で前記出力テキストを前記プロセッサにより編集可能とすることを特徴とする請求項 1 に記載の方法。

【請求項 12】

さらに編集対象の出力テキストをユーザが前記ポインティングデバイスにより選択したことに応答して、前記入力行内の前記選択した出力テキストの隣りの編集ウィンドウを前記表示デバイスの表示画面上で示すことを特徴とする請求項 1 に記載の方法。

【請求項 13】

前記入力行が第 1 の方向を向いており、さらに前記編集ウィンドウは第 1 の方向と直交

10

20

30

40

50

する第 2 の方向に向いていることを特徴とする請求項 1 2 に記載の方法。

【請求項 1 4】

さらに編集対象の出力テキストをユーザが前記ポインティングデバイスにより選択したことに応答して、前記入力行内の前記選択した出力テキストの隣りに入力テキストヒントウィンドウを前記表示デバイスの表示画面上で表示し、前記入力テキストヒントウィンドウに前記選択した出力テキストが変換された前記入力テキストが含まれることを特徴とする請求項 1 に記載の方法。

【請求項 1 5】

さらに編集対象の出力テキストをユーザが選択したことに応答して、前記入力行内の前記選択した出力テキストの隣りに第 1 の候補リストを前記表示デバイスの表示画面上で表示し、前記第 1 の候補リストに前記選択したテキストの代わりにできる 1 つまたは複数の代替え出力テキストが含まれることを特徴とする請求項 1 に記載の方法。

10

【請求項 1 6】

さらにその候補が出力テキストであることの確率に基づき前記第 1 の候補リスト内で前記出力テキスト候補を前記プロセッサにより順序づけることを特徴とする請求項 1 5 に記載の方法。

【請求項 1 7】

前記第 1 の候補リストが前記表示デバイスの表示画面上でスクロール可能であり、さらに前記リストがスクロールされるときに前記出力テキスト候補の移動を前記表示デバイスの表示画面上でアニメーション化することを特徴とする請求項 1 5 に記載の方法。

20

【請求項 1 8】

前記入力行が第 1 の方向を向いており、さらに編集のためユーザが出力テキストを前記入力デバイスにより選択したことに応答して、

前記第 1 の方向に直交する第 2 の方向で前記選択した出力テキストの上にある入力テキストヒントウィンドウを前記表示デバイスの表示画面上に表示し、前記入力テキストヒントウィンドウに前記選択した出力テキストが前記プロセッサにより変換された前記入力テキストが含まれ、および

第 2 の方向に前記選択した出力テキストの下に第 1 の候補ウィンドウを前記表示デバイスの表示画面上に表示し、前記第 1 の候補ウィンドウに前記選択した出力テキストの代わりにできる 1 つまたは複数の代替え出力テキスト候補が含まれることを特徴とする請求項 1 に記載の方法。

30

【請求項 1 9】

前記入力テキストが表音文字テキストと非表音文字テキストを含み、さらに、  
前記表音文字テキストを言語テキストに前記プロセッサにより変換し、および  
言語テキスト、非表音文字テキスト、および新規入力した表音文字テキストを共通の入力行内に前記表示装置の表示画面上で表示することを特徴とする請求項 1 に記載の方法。

【請求項 2 0】

さらにユーザが第 1 の言語の第 1 の入力モードと第 2 の言語の第 2 の入力モードを切り替えずに、複数の言語が含まれる入力テキストを前記入力デバイスにより入力できるようにすることを特徴とする請求項 1 に記載の方法。

40

【請求項 2 1】

前記入力テキストが個々の入力文字を含み、さらに、1 個から 6 個までの入力文字が前記表示デバイスの表示画面上に表示されるときに、前記 1 個から 6 個までの入力文字を前記出力テキストに前記プロセッサにより変換することを特徴とする請求項 1 に記載の方法。

【請求項 2 2】

前記入力テキストが個々の入力文字を含み、さらに、  
文字照合のため複数の変換候補を前記プロセッサにより評価し、  
両方の変換候補の 1 つまたは複数の文字が一致する場合に、1 つまたは複数の入力文字を一致する文字に前記プロセッサにより変換することを特徴とする請求項 1 に記載の方法

50

。

## 【請求項 23】

前記入力テキストが個々の入力文字を含み、さらに、一番最近入力した入力文字を常に前記表示デバイスの表示画面上に表示することを特徴とする請求項 1 に記載の方法。

## 【請求項 24】

前記入力テキストが個々の入力文字を含み、さらに、第 1 の最も可能性の高い変換候補のスコアが 2 番目に可能性が高い変換候補よりも著しく大きい場合に、1 つまたは複数の入力文字を前記第 1 の最も可能性の高い変換候補の前記出力テキストに前記プロセッサにより変換することを特徴とする請求項 1 に記載の方法。

## 【請求項 25】

プロセッサに請求項 1 に記載の前記方法をコンピュータに実行させるための実行可能命令を記録したことを特徴とする 1 つまたは複数のコンピュータ読み取り可能媒体。

## 【請求項 26】

表音文字テキストをユーザが入力デバイスにより表音文字テキストを入力したとおり前記表示デバイスの表示画面上に表示するステップ、

表音文字テキストから対応する言語テキストにプロセッサにより変換した後当該変換された言語テキストを前記表示デバイスの表示画面上に表示するステップ、

前記言語テキストを、変換される表音文字テキストの代わりに前記表示デバイスの表示画面上に表示するステップ、および

前記言語テキストおよび未変換の表音文字テキストをいっしょに前記表示デバイスの表示画面上に表示するステップ

を有し、前記対応する言語テキストに変換するステップは、

第 1 の測定基準に基づいて第 1 の候補リスト内の言語テキスト候補を前記プロセッサにより順序付けるステップであって、前記第 1 の候補リスト内の複数の言語テキスト候補は当該複数の言語テキスト候補内の 1 つをポインティングデバイスにより前記言語テキストとして選択することが可能な言語テキストの候補である、順序付けるステップと、

前記第 1 の測定基準と異なる第 2 の測定基準に基づいて第 2 の候補リスト内の言語テキスト候補を前記プロセッサにより配列するステップであって、前記第 2 の候補リスト内の複数の言語テキスト候補は当該複数の言語テキスト候補内の 1 つをポインティングデバイスにより前記言語テキストとして選択することが可能な言語テキストの候補である、配列するステップと、

前記第 1 の候補リストおよび前記第 2 の候補リストを前記表示装置の表示画面に表示するステップ、および

前記表示装置の表示画面に表示されている第 1 の候補リストおよび第 2 の候補リストの内の 1 つの言語テキスト候補を前記表音文字テキストから変換される言語テキストとしてポインティングデバイスにより選択するステップ

を有することを特徴とする方法。

## 【請求項 27】

前記表音文字テキストが中国語ピンインを含み、前記言語テキストが漢字を含むことを特徴とする請求項 26 に記載の方法。

## 【請求項 28】

さらに前記未変換の表音文字テキストと前記言語テキストを共通の水平行内にいっしょに前記表示装置の表示画面に表示することを特徴とする請求項 26 に記載の方法。

## 【請求項 29】

さらに追加表音文字テキストが前記入力デバイスにより入力された時に前記言語テキストを別の言語テキストに前記プロセッサにより修正し、前記別の言語テキストは以前に入力された表音文字テキストおよび追加表音文字テキストから変換された言語テキストであることを特徴とする請求項 26 に記載の方法。

## 【請求項 30】

さらにユーザが句読点を前記入力デバイスにより入力したことに応答して追加表音文字

10

20

30

40

50

テキストを入力するとき前記言語テキストをさらに修正するのを前記プロセッサにより中止することを特徴とする請求項 2 9 に記載の方法。

【請求項 3 1】

さらにユーザが前記言語テキストを入力デバイスにより確認応答して追加表音文字テキストを入力するとき前記言語テキストをさらに修正するのを前記プロセッサにより中止することを特徴とする請求項 2 9 に記載の方法。

【請求項 3 2】

さらに、変換されていない表音文字テキストを修正するためにアクティブのまま残しながら、ユーザが前記言語テキストを前記入力デバイスにより確認応答して前記言語テキストをさらに修正するのを前記プロセッサにより中止することを特徴とする請求項 2 9 に記載の方法。

10

【請求項 3 3】

さらに入力モードと編集モードを切り替えずに、前記言語テキストを前記プロセッサにより編集できるようにすることを特徴とする請求項 2 6 に記載の方法。

【請求項 3 4】

さらに編集対象の言語テキストをユーザが選択したことに応答して、前記選択した言語テキストの隣りに編集ウィンドウを表示することを特徴とする請求項 2 6 に記載の方法。

【請求項 3 5】

プロセッサ上で実行すると請求項 2 4 に記載の前記方法をコンピュータに実行させるためのコンピュータ実行可能命令を備えることを特徴とする 1 つまたは複数のコンピュータ読み取り可能媒体。

20

【発明の詳細な説明】

【0001】

(関連事例)

本出願は、1999年11月5日出願の米国仮特許出願第60/163,588号の特典を主張する。

【0002】

本特許出願はさらに、「Language Input Architecture For Converting One Text Form to Another Text Form With Tolerance To Spelling, Typographical, And Conversion Errors」という表題の\_\_\_\_年\_\_\_\_月\_\_\_\_日に同時に出願された米国特許出願第\_\_\_\_号および「Language Input Architecture For Converting One Text Form to Another Text Form With Modèles Entry」という表題の\_\_\_\_年\_\_\_\_月\_\_\_\_日に同時に出願された米国特許出願第\_\_\_\_号との同時係属出願でもある。これらの同時係属出願は両方とも、引用により本出願に取り込まれている。

30

【0003】

(技術分野)

本発明は、言語入力ユーザインタフェースに関する。より具体的には、本発明は、言語固有または多言語文書処理システム(word processing systems)、電子メールシステム、ブラウザなどで使用できる、表音文字テキスト(phonetic text)を入力し、言語テキスト(language text)に変換する言語入力ユーザインタフェースに関する。

40

【0004】

(発明の背景)

英数字キーボード(たとえば、英語QWERTYキーボード)を利用する言語固有文書処理システムは何年も前から存在している。英数字キーボードは、欧文文字セットなどの小さなアルファベットを採用している言語ではうまく機能する。しかし残念なことに、すべての言語が小さなキャラクタベース(character base)を持つわけではな

50

い。たとえば、中国語、日本語、韓国語などのキャラクタベース言語（記号言語（symbol languages）ともいう）は、数千文字で構成される。多くの異なる文字に対し別々のキーをサポートするキーボードを組み立てることは実際上不可能であるため、キャラクタベース言語には言語固有キーボードが存在していない。

#### 【0005】

費用のかかる言語および方言固有のキーボードを設計するのではなく、言語固有の文書処理システムではむしろ、ユーザが小さな文字セットキーボード（たとえば、QWERTYキーボード）から表音文字テキストを入力し、その表音文字テキストをキャラクタベース言語の言語テキストに変換するようにできる。「表音文字テキスト」は、所定の言語を発話したときに発生する音声を表すが、「言語テキスト」はテキストに表示されるとおり実際に書かれた文字を表す。たとえば、中国語では、ピンインは表音文字テキストの一例であり、漢字は言語テキストの一例である。通常、表音文字テキストを表すのに必要な文字セットは、言語テキストを表すのに使用される文字セットに比べてかなり小さい。表音文字テキストを言語テキストに変換することにより、多くの異なる言語を、従来のコンピュータおよび標準QWERTYキーボードを使用する言語固有のワードプロセッサにより処理できる。

#### 【0006】

表音文字テキストのユーザ入力を容易にするため、言語固有文書処理システムは言語入力ユーザインタフェース（UI）を採用することが多い。しかし、既存の言語入力UIは、学習が簡単でなく、高速タイピングに適していないためあまり使い勝手がよいとはいえない。このような使い勝手の悪さの例として、一部の従来型言語入力ユーザインタフェースでの表音文字テキスト入力と変換済み言語テキスト出力との分離が挙げられる。たとえば、ユーザが表示画面上のある場所に表音文字テキストを入力すると、言語テキストの変換済み文字が画面上の別のはっきりと区別される場所に表示される。2つの場所は、その場所に固有のローカルカーソルを備えている場合さえある。この二重表示方式は、入力を実際にどこで行っているかということに関してユーザを混乱させることがある。さらに、ユーザは画面上の場所から場所へ連続的に見て行く必要がある。

#### 【0007】

そのため、既存の言語入力UIは、多くの場合、日々のパーソナルコンピュータ（PC）ユーザではなくプロのタイピストでしか使用されない。キャラクタベース言語の国々では、これらの問題は、PCの普及に大きな影響を及ぼす。

#### 【0008】

一般に、言語入力ユーザインタフェースには、（１）コードベースのユーザインタフェースと（２）モードベースのユーザインタフェースの２種類がある。コードベースのユーザインタフェースでは、ユーザが言語の単語に関係するコードを記憶する。コードは、入力デバイスを使用して入力され、目的の言語テキストに変換される。この種のユーザインタフェースでは、ユーザはいったんコードを記憶してしまうとテキストを非常に高速に入力できる。しかし、これらのコードは、記憶するのは簡単でなく、忘れるのは簡単であるという場合が多い。

#### 【0009】

モードベースのユーザインタフェースでは、表音文字テキストが入力され、目的の言語テキストに変換される。モードベースのユーザインタフェースでは、ユーザがコードを記憶する必要はないが、通常、ユーザが言語テキストを入力するモードと編集するモードとを切り替える必要がある。モードベースのユーザインタフェースの一例は、中国語など表音文字テキスト・言語テキスト変換機能を利用して外国語用に改造されたMicrosoft社の「ワード」ブランドの文書処理プログラムがある。表音文字を「ワード」プログラムに入力するときに、ユーザが表音文字（たとえば、中国語ピンイン）を入力する入力モードとユーザが認識および変換プロセスの結果ときおり発生する不可避な誤りを修正するための編集モードとを切り替えるためのローカライズされたツールバーがユーザに対して表示される。

10

20

30

40

50

## 【 0 0 1 0 】

このような従来のインタフェースの短所の一つは、現在のモード - 入力または編集 - を意識し、テキスト入力とは無関係な追加ステップ（つまり、ツールバーコントロールボタンをクリックすること）を実行してモードを切り替える必要があるという点である。このインタフェースのせいで、ユーザに余分な仕事が生じ、ユーザの注意がテキスト入力から他の周辺の制御面にそらされ、入力速度が著しく低下する。

## 【 0 0 1 1 】

モードベースのユーザインタフェースと関連する他の問題として、ユーザインタフェースの観点から、不可避な変換エラーの処理法法に関するものがある。変換エラーは、認識および変換エンジンが表音文字テキストを不正な言語文字に変換するときが発生する。これは、指定された言語の性質と、表音文字テキストを使用して意図した文字を予測する精度から、きわめてよくあることである。ユーザが編集モードに切り替えた後、ユーザインタフェースは通常、ユーザが文字を修正するための何らかの手段を用意する。たとえば、中国語用の *Microsoft* の「ワード」ブランドの文書処理プログラムは、可能な代替文字が表示されているボックスをユーザに対し提示する。この文字のリストが長い場合、ボックスには、可能な文字のリスト内をスクロールするためのコントロールが表示される。

10

## 【 0 0 1 2 】

従来のモードベースのユーザインタフェースの他の短所として、異なる言語を入力するためにはモード切替をする必要があるという点が挙げられる。表音文字テキストを入力して、第 2 の言語のテキストを入力したくなった場合、ユーザは第 2 の言語を入力するためにモードを切り替える必要がある。たとえば、*Microsoft* の「ワード」の文脈では、ローカライズされたツールバーに、ユーザが第 1 の言語（たとえば中国語ピンイン）の入力と第 2 の言語（たとえば英語）の入力を切り替えるためのコントロールボタンが用意されている。ユーザは、意図した言語を単語認識エンジンに知らせるために意識的にコントロールをアクティブにする必要がある。

20

## 【 0 0 1 3 】

特にプロでないタイピストの観点からの言語入力 UI に関する問題として他に、タイピング誤りがある。表音文字テキスト入力 UI の平均的ユーザは、特にタイポを入力しがちである。タイピングの誤りを犯す理由の 1 つは、ユーザの地域が異なれば使用するキャラクターベース言語の方言も異なることである。ユーザは、ユーザが使用している方言のせいで表音文字テキストのスペルを誤る。表音文字テキストが少しでも違うと、文字テキストはまったく違ったものになることがある。

30

## 【 0 0 1 4 】

したがって、言語入力ユーザインタフェースを改善する必要がある。

## 【 0 0 1 5 】

（発明の概要）

本発明は、ユーザが入力した表音文字テキストと、表音文字テキストから同じ画面領域に変換された言語テキストとを使いやすく統合する言語入力ユーザインタフェースに関するものである。このユーザインタフェースは、ユーザが入力モードと編集モードを切り替える必要がないという点でモードレスである。このモードレスのユーザインタフェースはさらに、言語間の明示的なモード切替なしで複数の言語の入力に対応できる。そのため、ユーザインタフェースは、ユーザにとって直観的であり、学習しやすく、また使い勝手もよい。

40

## 【 0 0 1 6 】

一実装では、言語入力ユーザインタフェース（UI）は、表音文字テキストと変換された言語テキストとを統合するインライン入力機能を備える。特に、ユーザが入力中の表音文字テキストはすでに入力されている表音文字テキストおよびすでに変換されている言語テキストと並行して、同じ行に表示される。入力された表音文字テキストを同じ行にすでに変換された言語テキストとともに表示することで、ユーザは自分の目の焦点を同じ行に合

50

わせることができ、より直観的で自然なユーザインタフェースとなる。

【0017】

言語入力UIでは、1) 言語テキストを追加する、2) 言語テキストを削除する、3) 選択した言語テキストを1つまたは複数の置換言語テキスト候補に置き換えるなどの言語テキスト編集操作機能をサポートしている。ユーザインタフェースを使用することで、ユーザは言語テキストを選択し、手動で新規表音文字テキストをタイプしてこれを新規言語テキストに変換することにより置き換えることができる。それとは別に、このユーザインタフェースは言語テキスト候補の1つまたは複数のリストを備える。まずフローティングリストが、選択した変更する言語テキストと関連して表示される。この方法で、言語テキスト候補は、シーケンス構造内の適所に表示され、ユーザは文法上の文脈で修正を視覚化することができる。候補のリストは、選択したものが実際にユーザが元々意図したものである可能性についてのランクまたはスコアに応じた並べ替え順序で表示される。階層は、確率、文字ストローク、またはその他の測定基準に基づく。上位の候補は、そのセンテンスに最高スコアが付けられた候補であり、それに続いて、2番目の候補は、センテンスに次に高いスコアを付け、という具合に進む。

10

【0018】

ユーザがリストをスクロールすると、リストはコンテキストメニュー内で更新される。さらに、現在表示されている選択項目が、スクロール動作の方向にアニメーションで動くように表示される。このアニメーションにより、ユーザはリストのスクロール量またはスクロール速度を確認できる。ユーザが置換テキストを選択した後、そのテキストがセンテンス内の言語テキストの代わりに挿入され、ユーザは編集している単一行に注目することができる。

20

【0019】

言語入力UIの他の機能として、ユーザが編集している言語テキストのすでに入力された表音文字テキストを表示する機能がある。ユーザは、すでに入力されている表音文字テキストを選択し、選択した後、すでに入力されている表音文字テキストが言語テキストの代わりに表示される。続いて、表音文字テキストを編集し、新規言語テキストに変換することができる。

【0020】

言語入力ユーザインタフェースの他の機能として、センテンスベースの自動変換機能がある。センテンスベースの自動変換では、後続の表音文字テキストを入力した後センテンス内のすでに変換されている言語テキストをさらに自動的に異なる言語テキストに変換できる。センテンスがピリオドで完了したことが示されると、そのセンテンス内の言語テキストが固定され、後のセンテンス内に入力テキストを入力した結果、異なる言語テキストにさらに自動的に変換されることはない。語句ベースまたは類似の自動変換を他の実施形態でも使用できることは明白であろう。

30

【0021】

言語入力ユーザインタフェースの他の機能として、言語テキスト確認のあるセンテンスベースの自動変換機能がある。表音文字テキストが言語テキストに変換された後、ユーザは変換されたばかりの言語テキストを確認し、センテンスの文脈に照らして、変換されたばかりの言語テキストがさらに自動的に変換されることをしないようにすることができる。

40

【0022】

言語入力ユーザインタフェースの他の機能として、モード切替なしで複数の言語を取り扱える機能がある。第2の言語の単語または記号は、表音文字テキストと混ぜて使用した場合、特別な言語入力テキストとして取り扱われ、第2の言語テキストとして表示される。したがって、ユーザは、異なる言語を入力するときにモードを切り替える必要がない。

【0023】

これらの機能やその他のさまざまな機能とさらに、本発明を特徴付ける利点は、以下の詳細な説明を読み、関連する図面を参照することにより明白になるであろう。

【0024】

50



図全体を通して類似の構成要素および特徴を参照するのに同じ番号を使用している。

【 0 0 2 5 】

( 好ましい実施形態の詳細な説明 )

本発明は、表音文字テキスト入力および言語テキストへの変換を容易にする言語入力ユーザインタフェースに関する。説明のため、汎用コンピュータで実行される文書処理プログラムの一般的文脈で本発明を説明する。ただし、本発明は、文書処理以外の異なる多くの環境（たとえば、電子メールシステム、ブラウザなど）に実装することができ、またさまざまな種類のデバイスで実施することができる。

【 0 0 2 6 】

システムのアーキテクチャ

図 1 は、中央処理装置（CPU）102、メモリ104、および入出力（I/O）インタフェース106を備えるコンピュータシステム実施例100を示している。CPU 102は、メモリ104およびI/Oインタフェース106と通信する。メモリ 104は、揮発性メモリ（たとえば、RAM）および不揮発性メモリ（たとえば、ROM、ハードディスクなど）を表す。

【 0 0 2 7 】

コンピュータシステム100は、I/Oインタフェース106を介して接続された1つまたは複数の周辺装置を備える。周辺装置実施例は、マウス110、キーボード112（たとえば、英数字QWERTYキーボード、表音文字キーボードなど）、ディスプレイモニタ114、プリンタ116、周辺記憶装置118、およびマイクロホン120を備える。たとえば、コンピュータシステムは、汎用コンピュータで実装できる。したがって、コンピュータシステム100は、メモリ104に格納され、CPU 102で実行されるコンピュータのオペレーティングシステム（図に示されていない）を実装する。オペレーティングシステムは、ウィンドウ操作環境をサポートするマルチタスクオペレーティングシステムであるのが好ましい。適当なオペレーティングシステムの例として、Microsoft Corporation社のWindows（登録商標）ブランドのオペレーティングシステムがある。

【 0 0 2 8 】

ハンドヘルドデバイス、マルチプロセッサシステム、マイクロプロセッサベースのまたはプログラム可能な家電製品、ネットワークPC、ミニコンピュータ、メインフレームコンピュータなど、他のコンピュータシステム構成を使用できることに注意されたい。さらに、図1にはスタンドアローンのコンピュータが示されているが、通信ネットワーク（たとえば、LAN、インターネットなど）を介してリンクされているリモート処理デバイスによってタスクが実行される分散コンピューティング環境で言語入力UIを実用することもできる。分散コンピューティング環境では、プログラムモジュールをローカルとリモートの両方のメモリ記憶デバイスに配置できる。

【 0 0 2 9 】

データまたは文書処理プログラム130は、メモリ104に格納され、CPU 102で実行される。他のプログラム、データ、ファイルなども、メモリ104に格納できるが、説明を簡単にするため示していない。文書処理プログラム130は、表音文字テキストを受け取り、自動的に言語テキストに変換するように構成されている。より具体的には、文書処理プログラム130は、説明のためメモリ内に格納されプロセッサによって実行されるコンピュータソフトウェアとして実装されている言語入力アーキテクチャを実装する。文書処理プログラム130は、アーキテクチャ131に加えて他のコンポーネントも備えることができるが、そのようなコンポーネントは、文書処理プログラムに標準のものであると考えられるため、図に詳細に示したり、詳細に説明していない。

【 0 0 3 0 】

文書処理プログラム130の言語入力アーキテクチャ131は、ユーザインタフェース（UI）132、検索エンジン134、言語モデル136、およびタイピングモデル137を備える。アーキテクチャ131は、言語独立である。UI 132および検索エンジン

10

20

30

40

50

134は、汎用であり、どの言語でも使用できる。アーキテクチャ131は、言語モデル136およびタイピングモデル137を変更することにより特定の言語に合わせる。このアーキテクチャの詳細は、本発明に引用により取り込まれている「Language Input Architecture For Converting One Text Form to Another Text Form With Tolerance To Spelling, Typographical, And Conversion Errors」という表題の米国同時係属出願第\_\_\_\_\_号および「Language Input Architecture For Converting One Text Form to Another Text Form With Modelless Entry」という表題の米国同時係属出願第\_\_\_\_\_号で説明されている。

10

#### 【0031】

検索エンジン134、言語モジュール136、およびタイピングモデル137はいっしょに使用することで、表音文字テキスト - 言語テキスト間のコンバータ138を形成する。この開示の目的のために、「テキスト」は、1つまたは複数の文字および/または文字以外の記号とする。「表音文字テキスト」は一般に、所定の言語を話すときに生じる音声を表す英数字テキストである。「言語テキスト」は、書き言葉を表す文字と非文字記号である。「非表音文字テキスト」は、所定の言語を話すときに生じる音声を表さない英数字テキストである。非表音文字テキストは、言語テキスト以外の書き言葉を表す句読点、特殊文字、および英数字テキストを含む場合がある。

20

#### 【0032】

図27は、表音文字テキスト、変換された言語テキスト、および非表音文字テキストの一例を示している。この例では、表音文字テキストは中国語ピンインテキストであり、「hello」と翻訳される。この文字テキスト実施例は、漢字テキストであり、これも「hello」と翻訳される。非表音文字テキスト実施例は、英数字記号テキスト文字列「@3m」である。説明の目的のために、ワードプロセッサ130は、中国語ベースのワードプロセッサの文脈で説明し、言語入力アーキテクチャ131はピンインを漢字に変換するように構成されている。つまり、表音文字テキストはピンインであり、言語テキストは漢字である。

30

#### 【0033】

しかし、言語入力アーキテクチャは、言語独立であり、他の言語にも使用できる。たとえば、表音文字テキストは日本語の話し言葉の形態でよいが、言語テキストは漢字などの日本語の書き言葉を表す。それだけには限らないが、アラビア語、韓国語、インド語、その他のアジア言語など他の例も多数存在する。

#### 【0034】

たぶん、より一般的に述べると、表音文字テキストは、書いたときに欧文文字ベースの文字セットを使用しない所定の言語を話したときに出る音声を表す欧文文字セット（たとえば、英語のアルファベット）で表された英数字でよい。言語テキストは、所定の言語に対応する書かれた記号である。

40

#### 【0035】

表音文字テキストは、マウス110、キーボード112、またはマイクロホン120などの1つまたは複数の周辺入力デバイスを介して入力する。この方法で、ユーザは、キー入力または音声による表音文字テキスト入力が可能である。音声入力の場合、コンピュータシステムはさらに、話し言葉を受け取る音声認識モジュール（図に示されていない）を実装し、表音文字テキストに変換することができる。以下の説明では、キーボード112によるテキストの入力をフルサイズの標準英数字QWERTYキーボードで実行すると想定している。

#### 【0036】

UI 132では、表音文字テキストを入力と同時に表示する。このUIは、グラフィカルユーザインタフェースであるのが好ましい。ユーザインタフェース132は、表音文字

50

テキスト (P) を検索エンジン 134 に渡し、さらに、これは表音文字テキストをタイピングモデル 137 に渡す。タイピングモデル 137 は、表音文字テキストに誤りが含まれていると思われる場合にユーザが意図した表音文字テキストの適当な編集結果と考えられるさまざまなタイピング候補 ( $TC_1, \dots, TC_N$ ) を生成する。タイピングモデル 137 は、タイピング候補を検索エンジン 13 に渡し、さらに、これはタイピング候補を言語モデル 136 に渡す。言語モデル 136 は、ユーザが意図した表音文字テキストの変換された形式を表すと考えられる言語テキストで書かれているさまざまな変換候補 ( $CC_1, \dots, CC_N$ ) を生成する。変換候補は、タイピング候補と関連付けられている。表音文字テキストから言語テキストへの変換は 1 対 1 変換ではない。同じあるいは類似した表音文字テキストが言語テキスト内の多数の文字または記号を表すことがある。したがって、表音文字テキストの文脈は、言語テキストへの変換前に解釈される。他方、非表音文字テキストの変換は、通常、表示される英数字テキストが英数字入力と同じである直接的な 1 対 1 変換となる。

#### 【0037】

変換候補 ( $CC_1, \dots, CC_N$ ) が検索エンジン 134 に戻され、このエンジンにより、タイピング候補および変換候補のうちどれがユーザが意図するものである確率が最も高いかを判別する統計分析が実行される。確率が計算されると、検索エンジン 134 により、確率が最も高い候補が選択され、変換候補の言語テキストが UI 132 に返される。UI 132 では、表音文字テキストを変換候補の言語テキストに置き換え、同じ行に表示する。一方で、新規入力した表音文字テキストが新規挿入された言語テキストの前の行に表示され続ける。

#### 【0038】

ユーザが言語テキストを検索エンジン 134 で選択したものから変更したい場合、ユーザインタフェース 132 に、選択が実際に意図した回答である可能性の高さの順序でランク付けられた他の高確率の候補の第 1 のリストが表示される。ユーザがまだ可能な候補に満足しない場合、UI 132 は可能なすべての選択肢を与える第 2 のリストを表示する。第 2 のリストは、確率またはその他の測定基準 (たとえば、ストロークカウントまたは中国語文字の複雑さ) に関してランク付けることができる。

#### 【0039】

言語入力ユーザインタフェース

これ以降では、特にユーザインタフェース 132 の機能を説明する。とりわけ、ユーザインタフェース 132 では、入力された表音文字テキストと、変換された言語テキストとを統合して画面上の同じ行に視覚的に表示する。これらの機能の多くは、ウィンドウやメニューまたはカーソルの存在および場所など、画面に視覚的に表示する方法について説明されている。このような機能は、ユーザインタフェース 132 単独で、またはオペレーティングシステムでサポートされていることに留意されたい。

#### 【0040】

図 2 ~ 10 は、言語入力ユーザインタフェース 132 の実施例のさまざまな画面表示を示している。UI で入力され表示されているが、まだ言語テキストには変換されていない表音文字テキストを表すために記号「P」を図 2 ~ 10 全体をとおして使用している。記号「C」は、入力表音文字テキスト P から変換された変換された言語テキストを表す。下付きは、表音文字テキスト P、たとえば、 $P_1, P_2, \dots, P_N$  および変換された言語テキスト C、たとえば、 $C_1, C_2, \dots, C_N$  と、それぞれで使用し、表音文字テキストと変換された言語テキストの個々のものを表す。

#### 【0041】

統合されたインラインテキスト入出力

図 2 は、言語入力 UI 132 単独で、またはオペレーティングシステムで表示される画面表示 200 を示している。この図では、画面表示 200 は、Microsoft の Windows (登録商標) ブランドのオペレーティングシステムで生成されたものなど、カスタムグラフィックウィンドウと似ている。グラフィックウィンドウは、言語入力の文脈

10

20

30

40

50

での使用に合わせて手直しされており、表音文字テキストを入力しその後言語テキストに変換するインライン入力領域 202 を表す。インライン領域 202 は、平行な波線で絵として表されている。

#### 【0042】

入力カーソル 204 は、次の表音文字テキスト入力が行われる現在の位置をマークする。グラフィックUI はさらに、ツールバー 206、208、210、212、またはその他の機能などの複数のツールバーを、アプリケーション、たとえば、ワードプロセッサ、データプロセッサ、表計算ソフト、インターネットブラウザ、電子メール、オペレーティングシステムなどに応じて備えることができる。ツールバーは、一般に、文書処理技術またはデータ処理技術として知られており、ここでは詳述しない。

10

#### 【0043】

インライン入力領域 202 は、表音文字テキスト P の入力と変換された言語テキスト C の出力を統合したものである。このため、ユーザが画面の単一の領域注意を集中できて便利である。ユーザが表音文字を（キー入力または音声で）入力すると、表音文字テキスト P は第 1 の方向（たとえば、画面上で水平方向に）インラインで表示される。入力カーソル 204 は、変換された言語テキスト  $C_1 C_2$  および入力表音文字テキスト  $P_1 P_2 P_3$  により、またはそれと合わせて配置される。図 2 では、入力シーケンスは左から右であり、入力カーソル 204 は、すでに入力された表音文字テキスト  $P_1 P_2 P_3$  の右側に配置される。テキストを所定の言語を読み取るのと同じ方向に入力することは本発明の範囲内にあること、また本実装で説明している「左から右への」入力シーケンスは一例にすぎないことは明白であろう。さらに、言語入力 UI は、それだけには限らないが、垂直、対角線方向など実質的にどのような方向でもインライン入力することができることは明白であろう。インライン入力機能がユーザから見て離れて行くまたは向かって来るように見えるさまざまな 3 次元形式などその他のインライン形式を考えることができる。

20

#### 【0044】

##### 自動変換

ユーザが表音文字テキスト P を入力すると、コンバータ 138 は自動的に表音文字テキストを変換済み言語テキスト C に変換する。通常、少数の表音文字テキストエレメント P（たとえば、1 から 6 個の表音文字テキストエレメント P）を入力してから、表音文字テキスト P を言語テキスト C に変換する。

30

#### 【0045】

変換を行うときに、変換済み言語テキスト C がインライン領域 202 によって示されているように、表音文字テキスト P と同じ行に表示される。ユーザが表音文字テキストを入力し続けると、一番最近の入力表音文字テキスト P がすでに変換された言語テキスト C とインラインで表示される。たとえば、図 2 で、表音文字テキスト  $P_1 P_2 P_3$  は、一番最近に変換された言語テキスト  $C_1 C_2$  とインラインで表示される。入力された表音文字テキスト P を同じ行にすでに変換された言語テキスト C とともに表示することで、ユーザは自分の目の焦点を同じ行に合わせることができ、入力プロセスがより直観的で自然になるだけでなく入力が高速化される。

#### 【0046】

ユーザが表音文字テキスト P を入力し続けると、ユーザインタフェースは自動的に表音文字テキスト P をリアルタイムで言語テキスト C に変換し、ユーザはモードを切り替える必要がない。図 3 の例に示されているように、ユーザが表音文字テキスト  $P_4$  を入力するとすぐに、表音文字テキスト  $P_1 P_2 P_3$  が自動的に言語テキスト  $C_3$  に変換される。ユーザは、モードを切り替えたり、躊躇することなく、表音文字テキスト  $P_4 P_5 P_6 P_7$  を入力し続ける。

40

#### 【0047】

表音文字テキストから言語テキストへの変換は、言語モデル 136 によって制御される自動プロセスである。言語テキスト  $C_3$  は、可能なすべての言語テキストのうち最高の確率を持つものとして選択され、したがって、自動変換で使用される。ただし、ユーザのタイ

50

ピングが多いほど、考慮する文脈は大きくなる。したがって、言語テキスト  $C_3$  は、 $P_4 P_5 P_6 P_7$  などの表音文字テキストをさらに入力した後異なる言語テキストに変更される場合がある。

#### 【0048】

言語入力アーキテクチャ 131 は、追加入力テキストの入力への応答として変換済み言語テキストを変更する頻度が最小になるように構成することができる。文脈によっては、変換された言語テキストを入力テキストのそれぞれの入力文字とともに変更し、本質的に、所定の文脈でユーザが意図しているおおよそ等しい確率の 2 つまたはそれ以上の可能な解釈を切り替えることが可能である。言語テキストの一定の切り替えは、ユーザは視覚的に気を散らされる可能性がある。

10

#### 【0049】

テキスト切り替え作業を最小限に抑えるために、コンバータ 138 は、他の文脈が意図される確率が有意でない限り現在の言語テキストを維持することを規定する 1 つまたは複数の確率ベースの規則を実装することができる。このようにして、コンバータ 138 は、第 2 の言語テキストが統計的な観点からわずかしきよくない場合に、変換された言語テキストを第 2 の言語テキストに変更しにくい。有意性の度合いは文脈とともに変化する。たとえば、コンバータ 138 は、修正された言語テキストの可能性が置き換える言語テキストよりも少なくとも 5 パーセントポイント以上高いときのみ言語テキストを修正するように構成できる。

#### 【0050】

20

センテンススペースおよび確認済み自動変換

ユーザは、非常に長いテキスト文字列（たとえば、テキストの段落）が変換の対象となる場合に快適に感じられないことがある。ユーザインタフェースの一実装では、表音文字テキスト  $P$  から言語テキスト  $C$  への自動変換はセンテンススペースの自動変換である。つまり、センテンスが完了すると、そのセンテンス内の言語テキスト  $C$  は、後のセンテンス内に表音文字テキスト  $P$  を入力したときに異なる言語テキスト  $C$  にさらに自動的に変換されることはない。センテンススペースの自動変換機能を使用すると、ユーザのタイピング誤りが減るだけでなく、前のセンテンスが連続的に自動変換されることもなくなる。

#### 【0051】

センテンスは他にもいろいろ多くの方法で定義できることは明白である。たとえば、センテンスは、2 つのピリオドに挟まれたテキスト文字列、さまざまな定義済み句読点に挟まれたテキスト文字列、いくつかのテキストエレメントを含むテキスト文字列などある種の定義済み句読点で挟まれたテキスト文字列として定義することができる。ユーザが句読点を入力すると、句読点と前の句読点との間に入力したテキスト文字列は、もしあれば、センテンスとして処理することができる。そのセンテンス内の変換済み言語テキスト文字列  $C$  は、ユーザが表音文字テキストを後のセンテンスに入力した際にさらに自動的に変換されることはない。当業者であれば、必要ならば自動変換を 2 つまたはそれ以上のセンテンスに基づいて行うようにできることは理解できるであろう。

30

#### 【0052】

図 4 は、センテンスが句読点で確認される表示画面 200 の位置を示している。センテンスを確認して、句読点を入力すると、通常、センテンスの終わりにある表音文字テキスト  $P$  が言語テキスト  $C$  に自動的に変換される。たとえば、図 4 に示されているように、カンマ 400 を入力した後、表音文字テキスト  $P_4 P_5 P_6 P_7$  が言語テキスト  $C_4$  に変換される。すると、言語テキスト文字列  $C_1 C_2 C_3 C_4$  はセンテンスとして取り扱われる。変換済み言語テキスト  $C_1 C_2 C_3 C_4$  は、さらに自動的に変換されることはなくなる。

40

#### 【0053】

センテンススペースの自動変換に加えて、ユーザは入力した表音文字テキスト  $P$  から変換した後、変換済み言語テキスト  $C$  のうち 1 つまたは複数を確認することができる。ユーザはキーボードでユーザコマンドを入力して（たとえば、スペースバーを押して）変換されたばかりの言語テキスト  $C$  を確認し、センテンスの文脈に照らして、変換されたば

50

かりの言語テキストCがさらに自動的に変換されることのないようにすることができる。  
この機能の詳細については、図20および図24を参照して後述する。

#### 【0054】

##### 遅延変換

多くの言語では、ユーザは通常、表音文字テキストよりも言語テキストのほうが、読み修正する作業に慣れている。表音文字テキストを入力するときに、ユーザは、ふつう、変換が行われるのを待ってそれから、入力されたテキストが正確かどうかを判別しようとする。これは、特に、ピンイン文字よりも、漢字のほうの読み、修正する作業を好む中国語ユーザの場合にあてはまる。

#### 【0055】

このユーザ特性に照らして、言語入力アーキテクチャ131は表音文字テキストを言語テキストに変換する時期を考慮するように設計されている。一般に、変換は、変換済み言語テキストがユーザの意図したものであることが十分確実であるとコンバータ側で判別したときに行われる。UIの文脈では、最終的な変換によりユーザ入力により表音文字的なテキストとなったときに修正されそうにない確度の高い言語テキストが得られるように表音文字テキストのうち何文字を一度に表示すべきかという問題が特徴的なものとなっている。変換の開始時期が早すぎると、変換された言語テキスト内の誤りが増えて、ユーザが変換された言語テキストの強制的修正を行う頻度が高まる。変換開始時期が遅すぎると、ユーザに対し目的の言語テキストではなく長い表音文字テキスト文字列が表示されることになり使いにくい。

#### 【0056】

変換開始が早すぎるのと変換開始が遅すぎるのとの妥協として、高い変換精度が保証される最適な数の表音文字が入力されるまで変換を遅らせるように言語入力アーキテクチャを構成することができる。実際、このアーキテクチャは、最小の文字数分を入力してから最大の文字数分を入力する前までに表音文字テキストの代わりに変換された言語テキストを選択して表示する操作を遅らせるように設計されている。一実施例として、少なくとも1つのピンイン文字と高々6個のピンイン文字が入力されUIに表示されるときにピンインテキストを漢字テキストに変換するように構成した中国語用に手直しされた言語入力アーキテクチャが示されている。

#### 【0057】

一実装では、言語入力アーキテクチャは、与えられた文脈について、変換された言語テキストを選択して表示するまでに入力可能な表音文字の最適な個数を決定する一組の規則を実装している。規則は以下のようにまとめられる。

#### 【0058】

規則1：最後の（つまり、一番最近に入力した）入力文字を常に表示する。

#### 【0059】

規則2：複数の入力文字を入力し表示した後、一致すると思われる候補内の1つまたは複数の文字について上位N個の変換候補を評価する。少なくとも1つの変換された文字がN個の変換候補すべてについて同じであれば、入力テキストの一部を形成する少なくとも1つの入力文字を出力テキスト内の一致する変換済み文字に変換する。

#### 【0060】

規則3：第1の最も確度の高い変換候補のスコアが第2の最も確度の高い候補よりも著しく大きい場合、少なくとも1つの入力文字を第1の変換候補の文字に変換する。

#### 【0061】

##### モードレス編集

図5～9は、このアーキテクチャでサポートしているモードレス編集機能の実施例を示している。このユーザインタフェースを使用することにより、ユーザは、明示的なモード切替操作を行わずに、入力モードから編集モードへの継ぎ目のない遷移を行うことが可能である。さらに、編集モードは、言語テキストの追加、削除、および置換などの従来の編集機能をサポートしている。本発明では、新規表音文字テキストを入力するか、または少な

10

20

30

40

50

くとも1つの置換言語テキスト候補のリストから置換言語テキストを選択することにより、言語テキストを置換できる。

#### 【0062】

##### インプレース誤り修正

図5は、さまざまな編集機能のある画面表示200を示している。説明のため、ユーザは句読点400を入力して言語テキスト $C_1C_2C_3C_4$ （前には図4の）を確認しており、また確認された言語テキスト $C_1C_2C_3C_4$ を編集したいと仮定する。ユーザは、カーソル204の位置を確認済み言語テキスト $C_1C_2C_3C_4$ 内の目的の位置に変更する。カーソルの位置決めは、それだけには限らないが、矢印キー、マウスのクリック、または音声コマンドなどさまざまな方法で行える。図5は、編集用にこの文字を選択するため言語文字 $C_3$ の前に位置が変更されたカーソル204を示している。

10

#### 【0063】

カーソル204の位置を言語文字 $C_3$ の前に変更した後、ユーザは1つまたは複数のユーザコマンドを入力して、編集する文字を含むテキスト内の位置にあるインライン領域202上またはそのあたりにスーパーインポーズされる編集ウィンドウまたはボックス500を呼び出す。ユーザコマンドは、それだけには限らないが、キーボード112のエスケープキー「ESC」を押すなどの、当業でよく知られているいくつかの方法で実行できる。

#### 【0064】

図解されている実装では、編集ウィンドウまたはボックス500は、インラインテキストの第1の方向（たとえば、水平方向）に直交する第2の方向（たとえば、垂直方向）で言語文字 $C_3$ の隣りにポップアップする。ポップアップ編集ウィンドウ500は、入力テキストヒントウィンドウ502とスクロール可能な候補ウィンドウ504の2つの部分に分かれる。これら2つの部分は、共通のユーザコマンドで同時に呼び出されるのが好ましい。文字 $C_3$ の対応する表音文字テキスト $P_1P_2P_3$ は、すでにユーザによって入力されており、編集している言語文字 $C_3$ の真上に、縦に揃えられて、入力テキストヒントウィンドウ502内に表示される。入力表音文字テキスト $P_1P_2P_3$ を表示することで、ユーザは言語テキスト $C_3$ に対しすでに何を入力しているかを確認し、必要ならば編集することができる。入力テキストヒントウィンドウ502は、スクロールアップバー506が上部に配置されている。このスクロールアップバー506を起動することで、表音文字テキスト $P_1P_2P_3$ がセンテンス内のスライドし、言語テキスト文字 $C_3$ を置き換える。

20

30

#### 【0065】

候補ウィンドウ504は、少なくとも1つの置換言語テキスト候補 $C_3^a$ 、 $C_3^b$ 、 $C_3^c$ 、 $C_3^d$ のスクロール可能なリストが配置され、言語テキスト $C_3$ と同じまたは類似した表音文字テキストがある。候補ウィンドウ504は、言語テキスト $C_1C_2C_3C_4$ を含み、言語文字 $C_3$ の真下に縦方向に揃えられた、インライン入力領域202に直交する形で配列されている。上付きを使用して、 $C_3^a$ 、 $C_3^b$ 、 $C_3^c$ 、 $C_3^d$ など異なる言語テキスト文字を表す。候補ウィンドウ504に表示できる以上の候補がある場合、スクロールダウンバー508が候補ウィンドウ504の下部に表示される。ユーザはスクロールダウンバー508を選択（たとえば、クリック）して、追加置換言語テキストを表示できる。インプレースウィンドウ502および504の機能の1つでは、スクロール操作をアニメーション化し、候補またはテキストが上下に移動する状態を見せることができる。これにより、リストが一度に1項目ずつスクロールする視覚的フィードバックがユーザに返される。

40

#### 【0066】

入力テキストヒントウィンドウ502内の表音文字テキスト $P_1P_2P_3$ および候補ウィンドウ504内の置換言語テキスト候補 $C_3^a$ 、 $C_3^b$ 、 $C_3^c$ 、 $C_3^d$ は、さらに、図に示されているように、番号0、1、2、3、4で追加参照される。置換言語テキストの番号付け方法と候補ウィンドウ504のサイズは、いろいろな方法で設定できる。一実装では、候補ウィンドウ504はサイズが制限されており、置換言語テキストの上位4つの最高確率のみをリスト表示する。

#### 【0067】

50

候補ウィンドウ 504 内の言語テキスト候補  $C_3^a$ 、 $C_3^b$ 、 $C_3^c$ 、 $C_3^d$  は、何らかの順序またはランク付けで配列するのが好ましい。たとえば、順序は、候補が実際にユーザが最初に意図したものである確率または可能性に基づくことができる。この確率は、言語モデル 136 によって返される候補とともに、検索エンジン 134 によって計算される。与えられた文脈でのある置換言語テキストの確率が与えられた文脈での他の置換言語テキストの確率よりも高い場合、確率が高いほうの置換言語テキストが編集対象の言語テキストの近くに、低い参照番号で表示される。

#### 【0068】

ユーザは、オプションにより、適切な参照番号を入力して文字テキスト  $C_3$  を置き換えるか、または他の一般的な手法（選択したオプションをポイントしてクリックする）により、表音文字テキスト  $P_1P_2P_3$  を選択するか、または置換言語テキスト  $C_3^a$ 、 $C_3^b$ 、 $C_3^c$ 、 $C_3^d$  のうちから 1 つ選択することができる。選択した置換が、インラインテキスト内で文字  $C_3$  の代わりになる。ユーザが候補を選択した後、ポップアップウィンドウ 500 が自動的に閉じるが、修正したテキストを残すように構成できる。それとは別に、ユーザは、ウィンドウ 502 および 504 の外でマウスをクリックするなどの従来の方法を使用して、テキストヒントウィンドウ 502 と候補ウィンドウ 504 を明示的に閉じること

10

#### 【0069】

インプレースウィンドウ 502 および 504 で実装されるテキスト置換機能は、インプレース誤り修正機能（*in-place error correction feature*）と呼ばれる。選択した表音文字テキスト  $P_1P_2P_3$  または置換言語テキスト  $C_3^a$ 、 $C_3^b$ 、 $C_3^c$ 、 $C_3^d$  のうちから選択した 1 つが置換対象の言語テキスト  $C_3$  のある場所にインプレースで表示される。インプレース誤り修正機能を使用すると、ユーザは編集する言語テキストが含まれる言語テキスト文字列を一般的に近似することに集中できる。

20

#### 【0070】

##### 第 2 候補リスト

図 6 は、図 5 に示されているのと似た画面表示 200 を示しているが、第 1 の候補ウィンドウ 504 から分離され、隣接する第 2 の候補ウィンドウ 600 をさらに示している。第 2 の候補ウィンドウ 600 には、編集する文字テキスト  $C_3$  の対応する表音文字テキスト  $P_1P_2P_3$  と同じまたは類似する表音文字テキストを持つ置換言語テキストのより大きなまたはたぶん完全なリストが表示される。入力テキストヒントウィンドウ 502 内の表音文字テキスト  $P_1P_2P_3$  および候補ウィンドウ 504 内の置換言語テキスト候補  $C_3^a$ 、 $C_3^b$ 、 $C_3^c$ 、 $C_3^d$  はさらに第 2 の候補ウィンドウ 600 内に表示される。他の実施形態では、追加置換候補のみ、第 2 の候補ウィンドウ 600 内にリスト表示される。

30

#### 【0071】

第 2 の候補ウィンドウ 600 を開くために、ユーザは、候補ウィンドウ 504 内でアクティブになっている状態で、キーボードの右矢印キーを押すなど、コマンドを入力する。ユーザは、その後、マウスのクリックやキー入力などの適当なコマンドで目的の置換言語テキストを選択することができる。ユーザは、テキスト文字列からテキスト文字ヘフォーカス 602 を移すことができる。

40

#### 【0072】

第 2 の候補ウィンドウ 600 内の候補はさらに、何らかの順序で配列することもできるが、必ずしも、第 1 の候補ウィンドウ 504 で使用したのと同じランク付け方法に従うわけではない。通常、第 1 の候補ウィンドウ 504 の候補でも行っているような確率スコアで並べ替える操作は、多数の候補の違いが小さく、いくぶん意味がないため、完全な候補ウィンドウ 600 には役立たない場合がある。ユーザは、この設定で特定の候補を見つけることについて直観的感覚を持たない場合がある。したがって、第 2 の候補ウィンドウ 600 では、目的の候補の直観的発見を可能にする他の何らかの方法で候補のランク付けを行うおとする。

#### 【0073】

50



第2の候補ウィンドウ600内の候補をランク付けるために使用できる基準の1つは、特に日本語および中国語の文脈において、文字または記号の複雑さの測定基準となっている。たとえば、中国語テキスト候補のリストについては、候補を形成するのに必要なストローク数に応じて候補をリスト表示することができる。ストロークの順序により、目的の言語テキストを探索しているユーザはある種の明確な感覚をつかめる。ユーザは、一見似た複雑さを持つ文字が納められているウィンドウ600の特定の領域をざっと見ることができる。このランク付け基準は、ユーザが正確なストローク数を数えたり、それを知るようにすることを意図したものではなく、強い、首尾一貫した、視覚的に認識可能な並べ替え順序を与えることのみを意図している。

【0074】

ウィンドウ600を閉じるために、ユーザは、ウィンドウ600の外でキーボードでキー入力したり、マウスをクリックするなどコマンドを入力する。ウィンドウの開閉、ウィンドウ内の上下および左右のスクロール、ウィンドウ内の上下のスクロールは、当業で知られており、ここでは詳述しないことは理解されるであろう。

【0075】

インプレース表音文字テキスト修正

図7~9は、図5に示されている表音文字テキスト $P_1P_2P_3$ のインプレース表音文字テキスト修正を説明するさまざまな事例での一連の画面表示200を示している。この例では、ユーザは、入力テキストヒントウィンドウ502内の表音文字テキスト $P_1P_2P_3$ が正しくないと判断している。正しい表音文字テキストは $P_1^aP_2P_3$ であるべきである。表音文字テキストを修正するために、ユーザはまず、入力テキストヒントウィンドウ502から表音文字テキスト $P_1P_2P_3$ を選択する。

【0076】

図7は、選択した表音文字テキスト $P_1P_2P_3$ が、編集しているテキスト文字 $C_3$ の場所にインプレースで表示されることを示している。ユーザは、 $P_1$ を $P_1^a$ に変更して表音文字テキストを編集することができる。

【0077】

図8は、表音文字テキストが $P_1^a$ に変更された後のUIを示している。テキストヒントウィンドウ502がさらに更新され、変更を反映する。編集操作の結果、同じまたは類似の編集された表音文字テキスト $P_1^aP_2P_3$ を持つ少なくとも1つの新規置換言語テキスト $C_3^j$ が候補ウィンドウ504に表示される。その後、ユーザは、候補ウィンドウ504内の置換言語テキスト（たとえば、 $C_3^j$ ）を選択できる。

【0078】

図9は、編集された表音文字テキスト $P_1^aP_2P_3$ と置き換わる選択した置換テキスト $C_3^j$ を示している。他の実施形態では、編集された表音文字テキストは、最も有望な新規置換言語テキストに自動的に変換できる。

【0079】

混合言語入力

言語入力アーキテクチャは、さらに、2つまたはそれ以上の言語を区別するように構成できる。第1の言語を表音文字テキストとして検出して言語テキストに変換し、第2の言語を非表音文字テキストとして検出してそのまま保持する。UI 132は、この2つの言語をユーザがテキストを入力したときと同じ行に同時に表示する。この手法は、多言語テキストを入力するときに2つの入力モードを切り替えずに済むという利点がある。ユーザに関する限り、ユーザインタフェースはモードレスである。

【0080】

図10は、ユーザインタフェースの画面表示200を示しており、2つの異なる言語の混合テキストの統合された取り扱いと表示を示している。記号「A」は、第2の言語テキストの文字を表している。第2の言語Aは、第2の言語テキストAがユーザによる入力時に表示される非表音文字言語である。たとえば、第1の言語は漢字であり、第2の言語は英語である。多言語を任意個数の異なる言語とすることができることは明白であろう。

10

20

30

40

50

## 【0081】

一実装では、ユーザは混合言語テキストを入力することができ、そのうちの1つは言語テキストC（たとえば、漢字）に変換可能な表音文字テキストP（たとえば、ピンイン）である。文字ベース言語の表音文字テキストPは、表音文字テキストPが言語テキストCに自動変換されるまで言語テキストAとともにインラインに表示され、言語テキストCは第2の言語の言語テキストAとインラインに表示される。図10は、入力表音文字テキストP、変換された言語テキストC、および同じインライン領域202内にある第2の言語テキストAを示している。

## 【0082】

異なるフォントまたは色を使用して、表音文字テキストPと非表音文字テキストAとを区別することができる。たとえば、表音文字テキストPは、第1のフォントまたは色で表示され、非表音文字テキストAは、第1のフォントまたは色と異なる第2のフォントまたは色で表示される。フォントおよび色の他に、表音文字テキストPと非表音文字テキストAとを視覚的に区別する他の手法を使用することもできる。

## 【0083】

一般的なUI操作

図11～19は、言語入力アーキテクチャによって実行される方法を示している。これらの方法を言語入力ユーザインタフェースの一部として実行し表音文字テキストの入力および編集を簡単に行えるようにする他に、変換された言語テキストの編集も行えるようにする。図11は、一般的なプロセスを示しているが、図12～19は操作のいくつかをより詳細に説明している。これらの方法については、図2～10の画面表示をさらに参照して説明している。

## 【0084】

図11は、言語入力ユーザインタフェースを介してテキストを入力する方法100を示している。操作1102で、ユーザインタフェースにより、ユーザは共通のインライン領域202内にテキストを入力できる。説明している実装では、入力テキストは、中国語ピンインなどであることを特徴とする表音文字テキストである。入力テキストは、漢字などの文字ベースの言語の言語テキストに自動的に変換される（操作1104）。この変換の一実施例については、図1を参照して上で説明している。読者がこれに関心がある場合、このアーキテクチャの詳細について、本発明に取り込まれている「Language Input Architecture For Converting One Text Form to Another Text Form With Tolerance To Spelling, Typographical, And Conversion Errors」という表題の米国同時係属出願第\_\_\_\_\_号および「Language Input Architecture For Converting One Text Form to Another Text Form With Modelless Entry」という表題の米国同時係属出願第\_\_\_\_\_号を参照されたい。

## 【0085】

操作1106では、カーソルの位置変更または明示的コマンドにより示されるとおり、変換後言語テキストをユーザが編集するかどうかを判別する。編集する場合（つまり、操作1106からの「はい」の経路）、UIは編集する文字に近いカーソルのユーザの位置変更を受け取る（操作1108）。図5に示されているように、カーソルは言語テキスト文字の前に位置変更できる。

## 【0086】

操作1110で、UIは図5に示されているようにユーザコマンドへの応答として編集ウィンドウ500を開く。編集ウィンドウ500は、言語テキストを置換するための第1の候補リスト504を含む。適当な置換候補が候補リスト504に表示されない場合、ユーザは図6に示されているような第2の候補リストウィンドウ600を呼び出すことを決めることができる。操作1112により、ユーザが第2の候補ウィンドウ600を要求した

かどうかを判別する。適当な候補が第1の候補リスト504にあり、ユーザが候補リストウィンドウを開かないことに決めた場合（つまり、操作1112からの「いいえ」の分岐）、ユーザは第1の候補リストウィンドウから置換言語テキストを選択して、編集する言語テキストを置換することができる（操作1114）。

【0087】

他方、ユーザが第2の候補ウィンドウを呼び出した場合（つまり、操作1112からの「はい」の分岐）、UIは第2の候補リストウィンドウを開き、ユーザは編集中の言語テキストを置換する置換言語テキストを選択できる（操作1116）。第1の候補リストウィンドウ504または第2の候補リストウィンドウ600から選択した置換言語テキストが、インライン領域202内の言語テキストの場所にインプレースで表示される（操作1118）。操作の流れは操作1106で続く。

10

【0088】

ユーザがテキストを編集したくない場合（つまり、操作1106からの「いいえ」経路）、ユーザがテキスト入力続けるかどうかをUIで判別するが、これは、ユーザがカーソル位置を変更して文字を入力し続けることで示される（操作1120）。ユーザの動作はテキスト入力続けることを示唆する傾向がある場合は、カーソルは現在のセクションの終わりのところの入力位置戻り（操作1122）、操作の流れは入力インライン操作1102で続く。ユーザが続行したくない場合、プロセスは終了する。

【0089】

インライン入力：操作1102および1104

20

図12は、インライン入力サブプロセス1200を示しているが、これは、図11の操作1102および1104の一実施例である。このサブプロセスを示す画面表示例が図2と図3に示されている。

【0090】

操作1202で、UIは、入力デバイス（たとえば、キーボード、音声認識）から表音文字テキスト（たとえば、ピンイン）の入力文字列を受け取る。言語入力UIは、すでに変換されている言語テキストと同じインライン領域202内に表音文字テキストを表示する（操作1204）。表音文字テキストから言語テキストへのコンバータ138は、操作1206で表音文字テキスト文字列を言語テキスト（たとえば、漢字）に変換する。言語入力UIは、表音文字テキストを変換済み言語テキスト文字列に置き換え、インライン領域202内にその言語テキストを表示する（操作1208）。そこでサブプロセス1200は終了する。

30

【0091】

センテンススペースの変換：操作1104

図13は、自動変換サブプロセス1300を示しているが、これは、操作1104の他の実施例である。このサブプロセスを示す画面表示例が図3と図4に示されている。

【0092】

操作1302で、言語入力アーキテクチャは、入力デバイスを介してユーザによる表音文字テキスト入力文字列を受け取る。言語入力UIは、インライン領域202内に入力表音文字テキストを表示する（操作1304）。操作1306で、言語入力アーキテクチャは、表音文字が既存のセンテンスに属するのかそれとも新規センテンスに属するのかを判別する。この判別は、ピリオドやカンマなどの何らかの句読点をユーザが入力したかどうかに基づいて行うことができる。

40

【0093】

入力表音文字テキストが新規センテンスに属している場合（つまり、操作1306からの「新規」経路）、入力された表音文字テキストは、前のセンテンスがあればその中の前のテキストの内容を考慮することなく自動的に言語テキストに変換される（操作1308）。逆に、入力表音文字テキストが新規センテンスに属していない場合（つまり、操作1306からの「既存」経路）、センテンス内の表音文字テキストは、そのセンテンスの文脈内で自動的に変換される（操作1310）。この変換の一部として、追加テキストがセン

50

テンス全体の意図した意味を変更し続けるのと同時に、すでに変換されている言語テキストをさらに修正することができる。操作の流れは、変換操作 1 3 0 8 および 1 3 1 0 に続いて終了する。

【 0 0 9 4 】

確認済み変換：操作 1 1 0 4

図 1 4 は、自動変換サブプロセス 1 4 0 0 を示しているが、ここではユーザが変換済み言語テキストを確認する。サブプロセス 1 4 0 0 は、操作 1 1 0 4 の他の実施例である。

【 0 0 9 5 】

操作 1 4 0 2 で、言語入力アーキテクチャは、入力デバイスを介してユーザによる表音文字テキスト入力文字列を受け取る。言語入力 UI は、インライン領域 2 0 2 内に入力表音文字テキストを表示する（操作 1 4 0 4）。対応する未確認言語テキストの表音文字テキストは、文字ベースの言語の言語テキストに自動的に変換される（操作 1 4 0 6）。

【 0 0 9 6 】

操作 1 4 0 8 で、言語入力 UI は、ユーザが変換済み言語テキストを確認したかどうかを判別する。確認されていない場合、サブプロセスは終了する。そうでなく、ユーザが言語テキストを確認している場合（つまり、操作 1 4 0 8 からの「はい」経路）、UI では、変換された言語テキストを確認し、追加表音文字テキストが入力されるとともに他の文脈上の考察からその言語テキストを削除する（操作 1 4 1 0）。そこで操作の流れが終了する。

【 0 0 9 7 】

インプレース誤り修正：操作 1 1 0 8 ~ 1 1 1 8

図 1 5 ~ 1 8 は、インプレース誤り修正サブプロセスの異なる実装を示しているが、これは、図 1 1 の操作 1 1 0 8 ~ 1 1 1 8 の一実施例である。図 1 5 および 1 6 のサブプロセスは、言語テキストを修正するための第 1 と第 2 の候補リストの使用に関するものである。図 1 7 および 1 8 のサブプロセスは、表音文字テキストヒントウィンドウを使用して表音文字テキストを修正することに関するものである。

【 0 0 9 8 】

図 1 5 は、他の言語テキストをポップアップ候補ウィンドウで提供することにより変換済み言語テキストを修正するインプレース誤り修正サブプロセス 1 5 0 0 を示している。このサブプロセス 1 5 0 0 を示す画面表示例が図 5 に示されている。

【 0 0 9 9 】

操作 1 5 0 2 で、ユーザがカーソルをすでに入力している言語テキストの近くに移動することに応じて（たとえば、文字の前）、言語入力 UI では、編集する言語テキストを選択または識別する。この UI は、編集する言語テキストの真下にある第 1 の候補ウィンドウを含む編集ウィンドウ 5 0 0 を開き、選択した言語テキストの置換候補のリストを表示する（操作 1 5 0 4）。

【 0 1 0 0 】

操作 1 5 0 6 で、UI は第 1 の候補ウィンドウ 5 0 4 から置換候補のユーザによる選択を受け取る。言語入力 UI は、表音文字テキストを同じインライン領域 2 0 2 内で、選択された言語テキストの代わりに、選択した置換言語テキスト候補を表示する（操作 1 5 0 8）。そこで操作の流れが終了する。

【 0 1 0 1 】

図 1 6 は、他の言語テキストを他の言語テキストの完全なリストを二次の大きなポップアップ候補ウィンドウで提供することにより変換済み言語テキストを修正するインプレース誤り修正サブプロセス 1 6 0 0 を示している。このサブプロセス 1 6 0 0 を示す画面表示例が図 6 に示されている。

【 0 1 0 2 】

操作 1 6 0 2 で、ユーザがカーソルをすでに入力している言語テキストの近くに移動することに応じて（たとえば、文字の前）、言語入力 UI では、編集する言語テキストを選択または識別する。この UI は、編集する言語テキストの真下にある第 1 の候補ウィンドウ

10

20

30

40

50

504を含む編集ウィンドウ500を開き、選択した言語テキストの置換候補の短いリストを表示する(操作1604)。ユーザが適切な置換候補を見つけられない場合、ユーザは置換言語テキスト候補の第2の候補ウィンドウ600を呼び出すことができる(操作1606)。第2の候補リストは、第1の候補ウィンドウよりも大きなまたはより完全な置換言語テキスト候補のリストを含む。

【0103】

操作1608で、UIは第2の候補ウィンドウ600から置換候補のユーザによる選択を受け取る。言語入力UIは、表音文字テキストを同じインライン領域202内に選択された言語テキストの代わりに選択した置換言語テキスト候補を表示する(操作1610)。そこで操作の流れが終了する。

10

【0104】

図17は、ポップアップヒントウィンドウを介してすでに入力されている表音文字テキストを編集することにより変換済み言語テキストを修正するインプレース誤り修正サブプロセス1700を示している。このサブプロセス1700を示す画面表示例が図7に示されている。

【0105】

操作1702で、ユーザがカーソルをすでに入力している言語テキストの近くに移動することに応じて(たとえば、文字の前)、言語入力UIでは、編集する言語テキストを選択または識別する。このUIは、編集する言語テキストの真上にある表音文字テキストヒントウィンドウ502を含む編集ウィンドウ500を開き、ユーザが入力したとおりに表音文字テキストを表示する(操作1704)。

20

【0106】

ヒントウィンドウ502で表音文字テキストをユーザが選択した後(つまり、操作1706からの「はい」経路)、UIは編集している言語テキストの代わりに表音文字テキストを表示する(操作1708)。これにより、ユーザはインライン領域202内で表音文字テキストを修正できる。そこで操作の流れが終了する。

【0107】

図18は、すでに入力されている表音文字テキストを編集し、編集後の候補の新規セットを表示することにより変換済み言語テキストを修正するインプレース誤りサブプロセス1800を示している。このサブプロセス1800を示す画面表示例が図8と図9に示されている。

30

【0108】

操作1802で、ユーザがカーソルをすでに入力している言語テキストの近くに移動することに応じて(たとえば、文字の前)、言語入力UIでは、編集する言語テキストを選択または識別する。このUIは、選択した言語テキストの真上にある表音文字テキストヒントウィンドウ502および言語テキストの真下にある第1の候補ウィンドウ504を含む、編集ウィンドウ500を開く(操作1804)。

【0109】

ヒントウィンドウ502で表音文字テキストをユーザが選択した後(つまり、操作1806からの「はい」経路)、UIは編集している言語テキストの代わりに表音文字テキストを表示する(操作1808)。UIは、表音文字テキストのユーザ編集結果を受け取ってインライン編集領域202内に表示する(操作1810)。編集への応答として、UIは第1の候補ウィンドウ504に置換言語テキスト候補の新規リストを表示する(操作1812)。ユーザは、さらに、必要な場合に第2の候補ウィンドウ600を呼び出すことができる。

40

【0110】

操作1814で、UIは第1の候補ウィンドウ504の新規リストから置換候補のユーザによる選択を受け取る。言語入力UIは、表音文字テキストを同じインライン領域202内に選択された言語テキストの代わりに選択した置換言語テキスト候補を表示する(操作1610)。そこで操作の流れが終了する。

50

## 【0111】

## 多言語入力

図19は、2つまたはそれ以上の異なる言語をインライン入力UIを使用して入力する多言語入力サブプロセス1900を示している。このサブプロセス1900を示す画面表示例が図10に示されている。

## 【0112】

操作1902で、言語入力アーキテクチャは、入力デバイスを介してユーザによる表音文字テキストと非表音文字テキストの混合入力文字列を受け取る。言語入力UIは、すでに変換されている言語テキストと同じインライン領域202内に混合テキストを表示する(操作1904)。

10

## 【0113】

操作1906で、言語入力アーキテクチャは、入力テキストが非表音文字テキスト(たとえば、英語)に対して表音文字テキスト(たとえば、ピンイン)かどうかを判別する。入力テキストが表音文字テキストである場合(つまり、操作1906からの「はい」経路)、言語入力アーキテクチャにより、表音文字テキストは言語テキストに変換される(操作1908)。UIは、入力された表音文字テキストの代わりに、また前のテキストとインラインに言語テキストを表示する(操作1910)。他方、入力テキストが表音文字テキストである場合(つまり、操作1906からの「いいえ」経路)、言語入力アーキテクチャでは変換せず、UIは前のテキストとインラインに非表音文字テキストを表示する(操作1912)。そこで操作の流れが終了する。

20

## 【0114】

## 中国語ベースの実施例

図20~26は、中国語の文脈における言語入力アーキテクチャとUIの実施例を示している。この文脈では、表音文字テキストは中国語ピンインであり、言語テキストは漢字である。

## 【0115】

図20は、インライン入力機能の一例を示す、中国語入力ユーザインタフェースの実装である。表2000は、インライン入力領域に表示されるとおりのユーザによるピンインテキスト2002および2004入力および対応する変換された漢字テキスト2006および2008の2つの文字列を含む。表示画面例2010は、表2000の下に示されており、変換された漢字テキスト2008を含む。カーソルバー2012のところで入力しているピンインテキストは変換された中国語テキストとインラインに表示されることに注意されたい。表示画面2010に示される他の特性は、文書処理技術の分野では知られている。

30

## 【0116】

図21は、変換された漢字テキストが現在インライン入力領域2102に表示されている中国語UI画面2100を示している。ユーザは、編集のためカーソルを移動して中国語文字テキスト2104を選択し、ピンインテキストヒントウィンドウ2108と第1の漢字テキスト候補ウィンドウ2110からなるポップアップ編集ウィンドウ2106を呼び出している。選択した中国語文字テキスト2104と関連するピンインテキスト2112が、ピンインテキストヒントウィンドウ2108に表示される。

40

## 【0117】

図22は、インブレース誤り修正機能の一例を示す、中国語入力ユーザインタフェースの一実装である。表2200では、表音文字ヒントおよび候補リストを含む編集ウィンドウを開くアクション2202および候補リストから項目「1」を選択するアクション2204の2つのユーザアクションを左の列に示している。左の列のユーザアクションへの応答として、表2200の右の列は、対応するスクリーンショット例2206および2208を示している。

## 【0118】

スクリーンショット2206に関して、ユーザは、カーソルを文字テキスト2210の前

50

に移動し、編集対象の中国語文字テキスト 2 2 1 0 を選択する。ユーザは、ピンインテキストヒントウィンドウ 2 2 1 2 および第 1 の候補リストウィンドウ 2 2 1 4 を含む編集ウィンドウを開くコマンドを入力する。次に、ユーザは候補リスト 2 2 1 4 から項目「1」を選択し、項目「1」と関連する第 1 の候補 2 2 1 6 が元の選択したテキスト 2 2 1 0 と置き換わる。さらに、リスト 2 2 0 8 内の候補が更新され（つまり、上へ 1 項目分スクロール）、選択した候補 2 2 1 6 がインライン入力領域内に移動することにも注意されたい。更新作業がアニメーション化され、選択された候補 2 2 1 6 がインライン領域に移動することが視覚的に示される。

#### 【 0 1 1 9 】

図 2 3 は、ピンインテキストのインプレース修正を示す、中国語入力ユーザインタフェースの他の実装である。表 2 3 0 0 の左の列は、5 つのユーザアクション列 2 3 0 2 ~ 2 3 1 0 を含み、右の列は、ユーザアクションで得られる対応するスクリーンショット例 2 3 1 2 ~ 2 3 2 0 を示している。

10

#### 【 0 1 2 0 】

ユーザが文字テキストを編集することに決めた場合、ユーザはカーソルを編集する文字テキストの前に移動する（アクション 2 3 0 2）。ユーザが編集する中国語文字テキスト 2 3 3 0 を選択するとする（UI スクリーンショット 2 3 1 2）。文字テキスト 2 3 3 0 の前にカーソルを移動した後、ユーザはコマンドを入力し（たとえば、「ESC」キーを押す）、編集ウィンドウを呼び出す（アクション 2 3 0 4）。その結果、ピンインテキストヒントウィンドウ 2 3 3 2 および第 1 の候補リストウィンドウ 2 3 3 4 が、UI スクリーンショット 2 3 1 4 に示されているように開かれる。

20

#### 【 0 1 2 1 】

次に、ユーザは「0」（アクション 2 3 0 6）を入力して、ピンインテキストヒントウィンドウ 2 3 3 2 でピンインテキスト 2 3 3 6 を選択する。選択したピンインテキスト 2 3 3 6 が、UI スクリーンショット 2 3 1 6 で示されているように選択した文字 2 3 3 0 に取って代わる。このときに、ユーザは元のピンインテキストを自由に編集できる。

#### 【 0 1 2 2 】

ユーザがピンインテキスト 2 3 3 6 内にアポストロフィを追加し（アクション 2 3 0 8）、UI スクリーンショット 2 3 1 8 に示されているようなテキスト 2 3 3 6 を出力すると仮定する。編集されたピンインテキスト 2 3 3 6 が、インライン領域と、ピンインテキストヒントウィンドウ 2 3 3 2 の両方に表示される。編集の後、第 1 の候補ウィンドウ 2 3 3 4 が文字テキスト候補の新規リストで更新される。この例では、編集されたピンインテキスト 2 3 3 6 に対応する新規文字テキスト候補 2 3 3 8 が、第 1 の候補リストウィンドウ 2 3 3 4 に表示される。

30

#### 【 0 1 2 3 】

最後に、ユーザは、たとえば「1」を入力して第 1 の候補リストウィンドウ 2 3 3 4 で目的の文字テキスト 2 3 3 8 を選択する（アクション 2 3 1 0）。その結果、選択した文字テキスト 2 3 3 8 が、UI スクリーンショット 2 3 2 0 に示されているように、編集されたピンインテキスト 2 3 3 6 の場所にインプレースで表示される。この方法で、新規文字テキスト 2 3 3 8 が実際に、元の言語テキスト 2 3 3 0 の代わりに表示される。

40

#### 【 0 1 2 4 】

図 2 4 は、中国語と英語などの混合言語の入力を示す、中国語入力ユーザインタフェースの他の実装である。表 2 4 0 0 の左の列は、2 つのユーザアクション列 2 4 0 2 および 2 4 0 4 を含み、右の列は、ユーザアクションで得られる対応するスクリーンショット例 2 4 0 6 および 2 4 0 8 を示している。

#### 【 0 1 2 5 】

ユーザがアクション 2 4 0 2 で示されているようにピンインテキスト 2 4 1 0 と英語テキスト 2 4 1 2 の混合テキストを入力すると仮定する。ユーザは、中国語入力と英語入力のモード切替を行うことなく混合テキストを言語入力 UI に入力できる。つまり、ユーザは、停止することなく、同じ行にピンインテキストと英語テキストを簡単に入力できる。ピ

50

ンインテキスト2410は、UIスクリーンショット2406に示されているように、中国語テキスト2414に変換され、同じインライン領域内に表示される。英語テキスト2412は、言語入力アーキテクチャで変換されず、入力したとおりに表示される。

【0126】

その後、ユーザはモードを切り替えずに、ピンインテキスト2416、英語テキスト2418、およびピンインテキスト2420の混合テキストを入力する(アクション2404)。ピンインテキスト2416および2420は、UIスクリーンショット2408に示されているように、中国語テキスト2422および2424に変換される。英語テキスト2418は、変更されずそのままであり、変換された中国語テキストとインラインに表示される。

10

【0127】

一実装により、表音文字テキストと非表音文字テキストが異なった形で表され、区別できる。たとえば、図20の表2000および図24の表2400内の混合テキストを比較する。ピンインテキスト(たとえば、図20の2612)は狭い太字フォントで表示されるが、英語テキスト(たとえば、図24の2412または2418)は細いc o u r i e rタイプのフォントで表示される。

【0128】

図25は、第1と第2の候補リストのインプレース編集を示す、中国語入力ユーザインタフェースの他の実装である。表2500の左の列は、2つのユーザアクション列2502および2504を含み、右の列は、ユーザアクションで得られる対応するスクリーンショット例2506および2508を示している。

20

【0129】

アクション2502で、ユーザは編集する中国語テキストを選択し、ピンインテキストヒントウィンドウ2510と第1の文字テキスト候補リスト2512を開くコマンドを入力する。ウィンドウ2510および2512は、UIスクリーンショット2506に示されているように、それぞれ、インライン入力領域の上および下に表示される。

【0130】

次に、アクション2504で、ユーザは、第2の文字テキスト候補リストを開くコマンドを入力する。第2の文字テキスト候補リストウィンドウ2514が、UIスクリーンショット2508に示されているように、第1の候補リスト2512の隣りにポップアップで開く。その後、ユーザは第2の文字テキスト候補リストウィンドウ2514から文字テキスト候補を選択することができる。

30

【0131】

図26は、確認済み文字テキストのあるセンテンススペースの自動変換を示す、中国語入力ユーザインタフェースの他の実装である。表2600の左の列は、5つのユーザアクション列2602~2610を含み、右の列は、ユーザアクションで得られる対応するスクリーンショット例2612~2620を示している。

【0132】

アクション2602で、ユーザはピンインテキスト2622および2624を入力する。UIスクリーンショット2612に示されているように、ピンインテキスト2622は自動的に文字テキスト2626に変換され、ピンインテキスト2624はさらにユーザ入力があるまで変換されずそのままである。アクション2604で、ユーザは続いてピンインテキスト2628を入力する。すると、すでに変換されている文字テキスト2626は、ピンインテキスト2628を追加することにより文脈が変更された結果、異なる中国語文字テキスト2630に自動的に変換される。変換された文字テキストのこのような修正は、UIスクリーンショット2614に示される。ピンインテキスト2624および2628は、この時点では変換されずそのままであり、修正された言語テキストとインラインに表示され続ける。

40

【0133】

次に、アクション2606で、ユーザは、変換したばかりの文字テキスト2630を確認

50



する確認コマンド（たとえば、スペースバーを押す）を入力する。一方、ピンインテキスト 2624 および 2628 は、ここまでのセンテンスの文脈に基づいてそれぞれ自動的に中国語テキスト 2632 および 2634 に変換される。これは、スクリーンショット 2616 に示されている。

【0134】

その後、アクション 2608 で、ユーザが追加ピンインテキスト（図に示されていない）を同じセンテンス内に入力し、ピンインテキストが、UI スクリーンショット 2618 に示されているように、文字テキスト 2636 に変換される。確認済みの文字テキスト 2630 はピンインテキストを後から入力しても変更されないことに注意されたい。

【0135】

比較のため、文字テキスト 2630 はユーザアクション 2606 によって確認されないと仮定する（たとえば、ユーザがスペースバーを押さない）。その代わりに、ユーザは、文字テキスト 2630 の確認をせずに、追加ピンインテキストを入力する。この場合、UI スクリーンショット 2620 に示されているように、文字テキスト 2626 は変更されずそのままであり、テキスト 2630 に修正されない。これは、ピンインテキストから文字テキストへの自動変換がセンテンススペースであり、文字テキスト 2626 がセンテンスの一部であるためである。センテンスがアクティブである限り（つまり、句読点でセンテンスが終わっていないか、または新しいセンテンスがまだ開始していない）、現在のセンテンス内のすでに変換された文字テキストは、ユーザが変換済み文字テキストを確認しない限り、さらに修正を受ける。

【0136】

結論

上の説明では、構造機能および／または方法論的動作に固有の言語を使用しているが、付属の特許請求の範囲で定義されている本発明は説明した特定の機能または動作に限られるわけではない。むしろ、特定の機能および動作は、本発明を実装する実施例として開示されている。

【図面の簡単な説明】

【図 1】 言語入力アーキテクチャを実装する言語固有のワードプロセッサを備えるコンピュータシステムのブロック図である。言語入力アーキテクチャは、言語入力ユーザインタフェース（UI）を備える。

【図 2】 言語入力ユーザインタフェースの一実装の画面表示を示す図である。図 2 は、言語入力 UI のインライン入力機能を示している。

【図 3】 自動変換機能を示す言語入力 UI の画面表示を示す図である。

【図 4】 センテンススペースの自動変換機能を示す言語入力 UI の画面表示を示す図である。

【図 5】 インプレースエラー修正機能および表音文字ヒント機能を示す、言語入力 UI の画面表示を示す図である。

【図 6】 第 2 の候補リスト機能を示す、言語入力 UI の画面表示を示す図である。

【図 7】 インプレース表音文字テキスト修正機能を示す言語入力 UI の画面表示を示す図である。

【図 8】 図 7 のインプレース表音文字テキスト修正機能の後続の画面を示す言語 UI の画面表示を示す図である。

【図 9】 図 7 および図 8 のインプレース表音文字テキスト修正機能の後続の画面を示す言語 UI の画面表示を示す図である。

【図 10】 複数の異なる言語を含む混合テキストの入力を示す言語入力 UI の画面表示を示す図である

【図 11】 言語入力ユーザインタフェースを使用してテキストを入力する方法の流れ図である。

【図 12】 インライン入力サブプロセスの流れ図である。

【図 13】 自動変換サブプロセスの流れ図である。

【図 1 4】 文字テキスト確認を行う自動変換サブプロセスの流れ図である。

【図 1 5】 インプレースエラー修正サブプロセスの流れ図である。

【図 1 6】 第 2 の候補リストがあるインプレースエラー修正サブプロセスの流れ図である。

【図 1 7】 表音文字テキストヒントサブプロセスの流れ図である。

【図 1 8】 インプレース表音文字テキスト修正サブプロセスの流れ図である。

【図 1 9】 インライン入力混合言語テキストサブプロセスの流れ図である。

【図 2 0】 インライン入力機能の一例を示す、ユーザ入力の実施例と中国語入力ユーザインタフェース実施例の実行結果のスクリーンショットの図である。

【図 2 1】 ピンインテキストヒント機能の一例を示す中国語入力ユーザインタフェース実施例の画面表示例である。

10

【図 2 2】 インプレースエラー修正機能の一例を示す、ユーザ入力の実施例と中国語入力ユーザインタフェース実施例の実行結果のスクリーンショットの図である。

【図 2 3】 インプレースピンインテキスト修正機能の一例を示す、ユーザ入力の実施例と中国語入力ユーザインタフェース実施例の実行結果のスクリーンショットの図である。

【図 2 4】 英語 / 中国語混合入力機能の一例を示す、ユーザ入力の実施例と中国語入力ユーザインタフェース実施例の実行結果のスクリーンショットの図である。

【図 2 5】 第 2 の候補リスト機能の一例を示す、ユーザ入力の実施例と中国語入力ユーザインタフェース実施例の実行結果のスクリーンショットの図である。

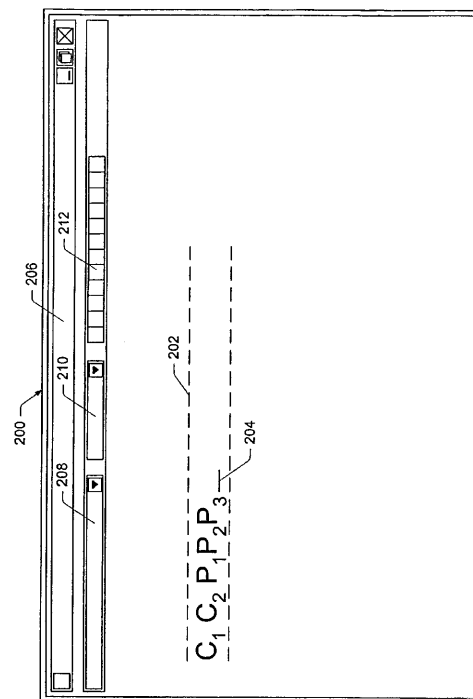
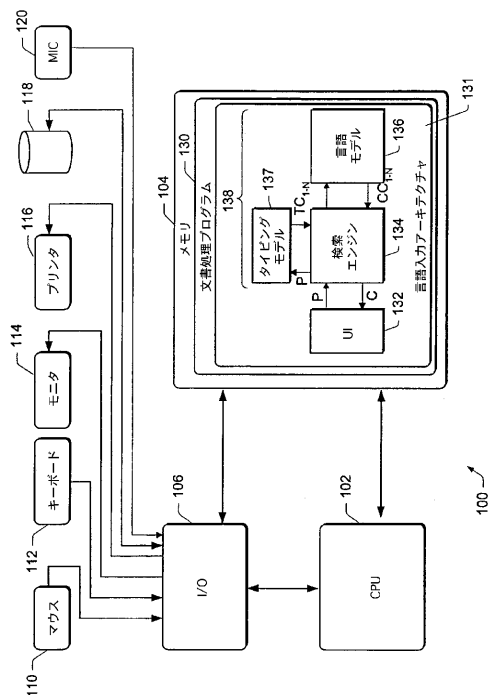
【図 2 6】 文字確認機能のあるセンテンススペースの自動変換一例を示す、ユーザ入力の実施例と中国語入力ユーザインタフェース実施例の実行結果のスクリーンショットの図である。

20

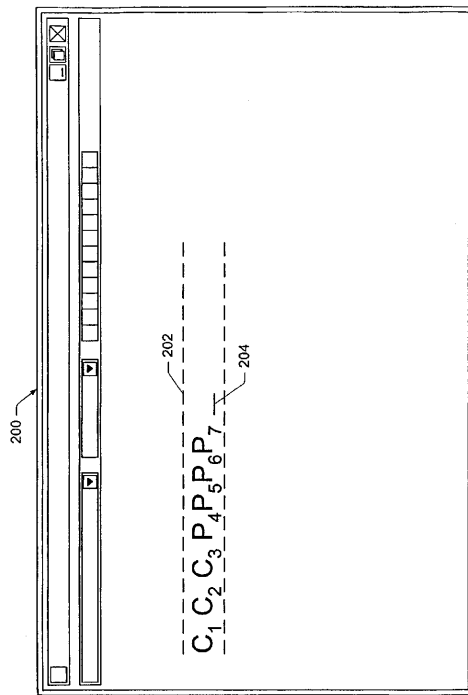
【図 2 7】 表音文字テキスト（たとえば、中国語ピンインテキスト）とその対応する文字テキスト（たとえば、中国語文字テキスト）の定義と、非表音文字テキスト（たとえば、英数字テキスト）の定義を示す図である。

【図 1】

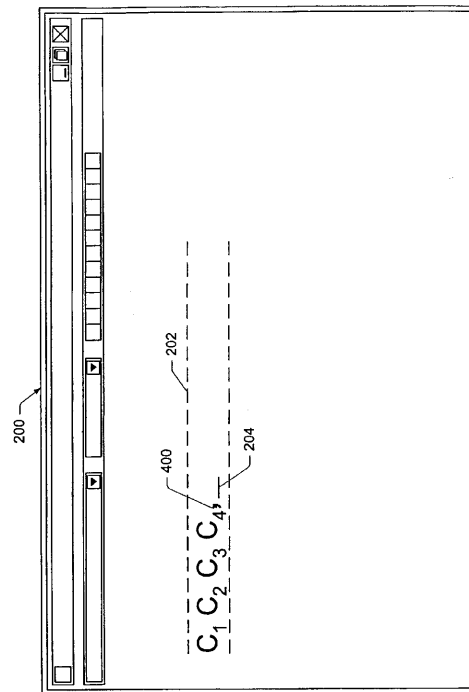
【図 2】



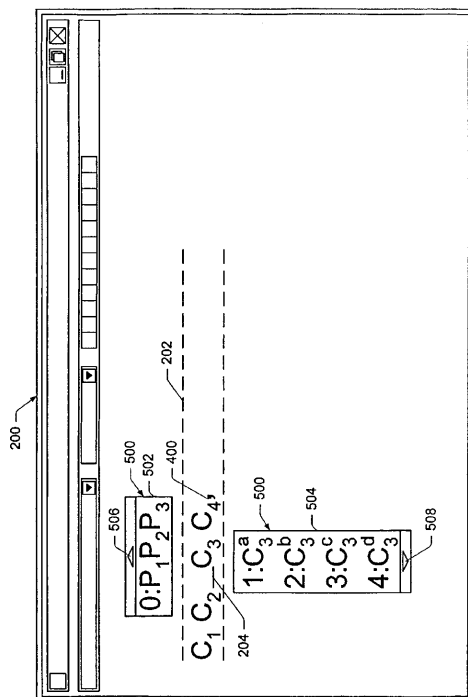
【図 3】



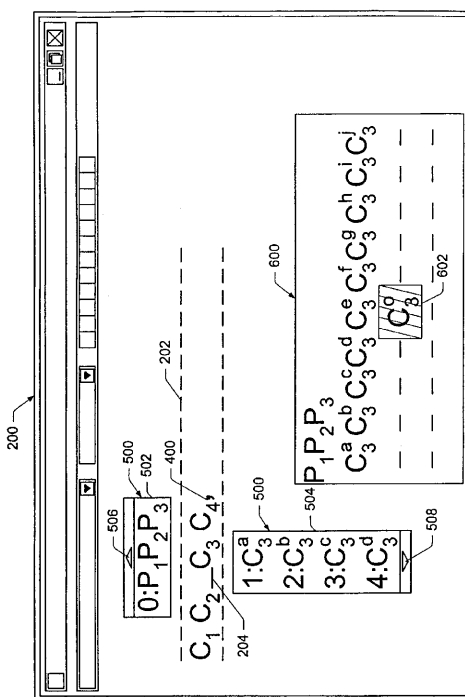
【図 4】



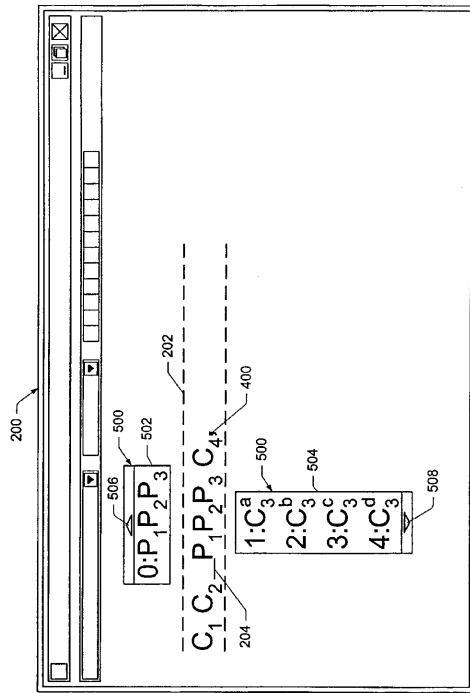
【図 5】



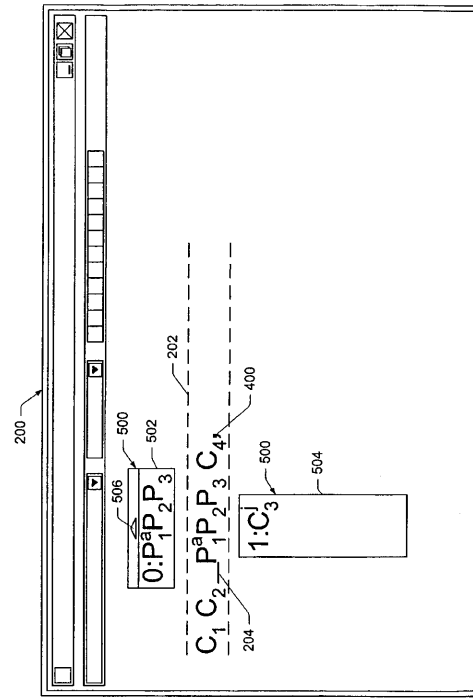
【図 6】



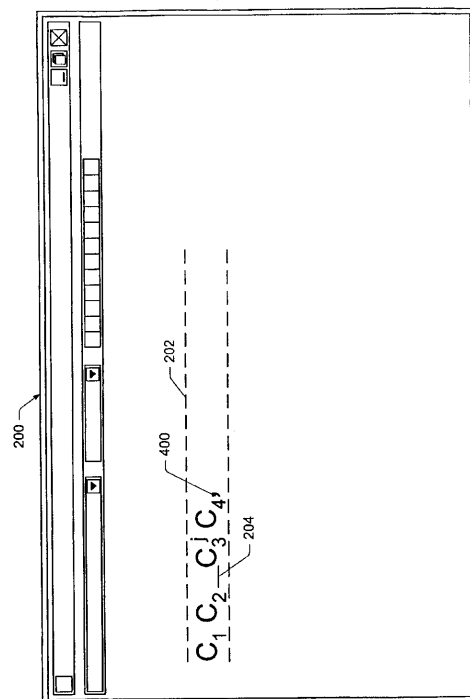
【図 7】



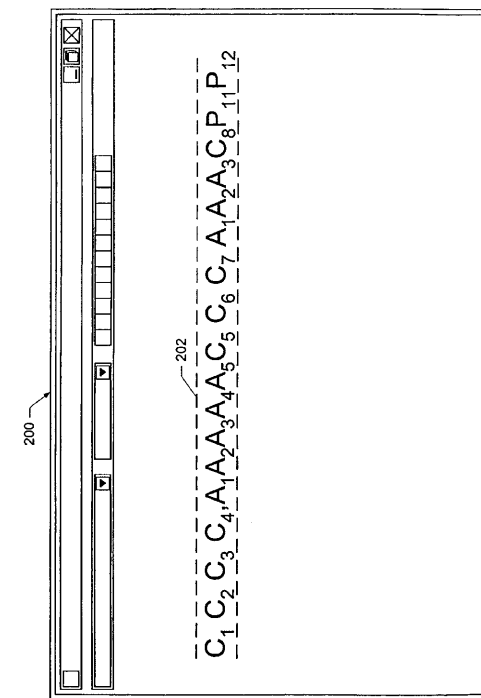
【図 8】



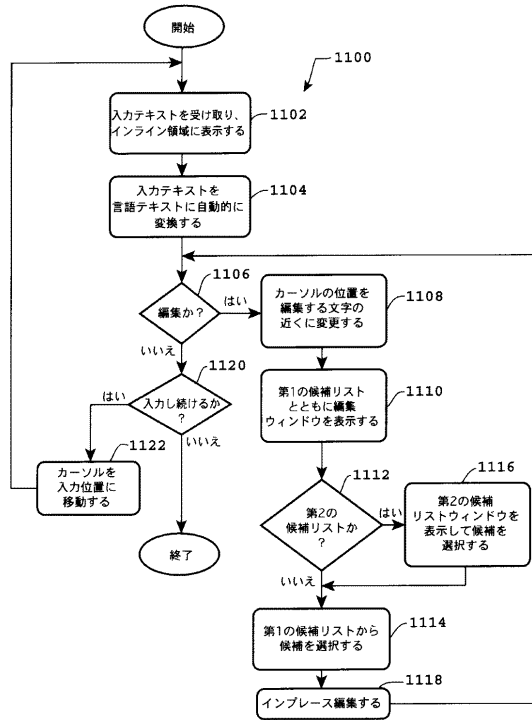
【図 9】



【図 10】

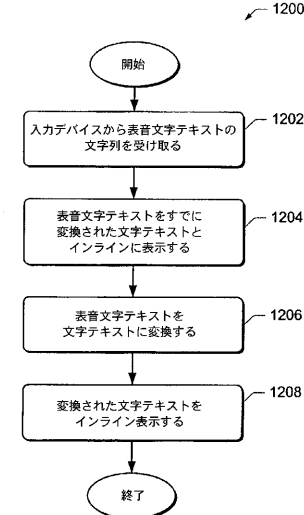


【図 11】



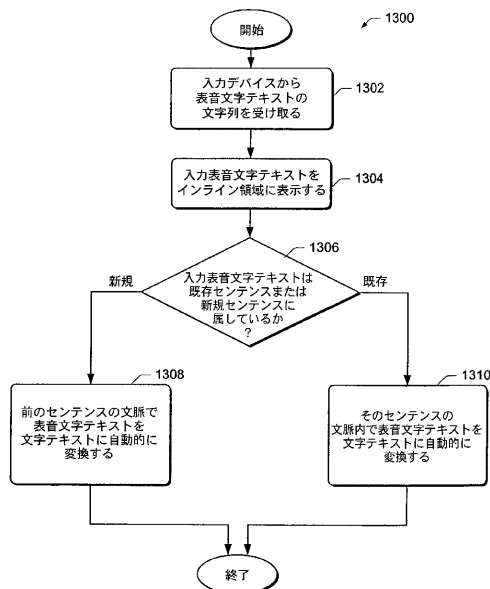
【図 12】

操作1102および1104：インライン入力および表示



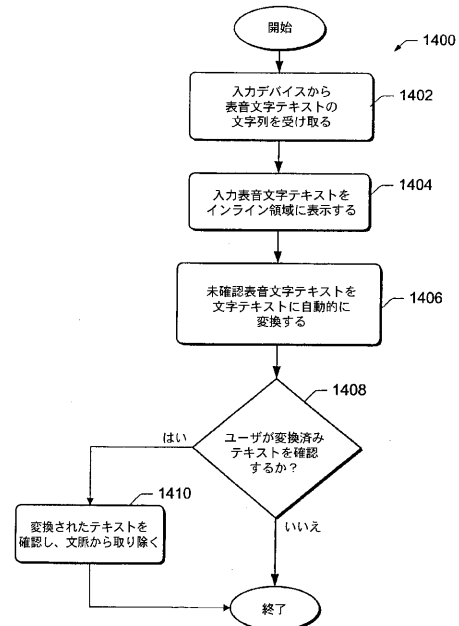
【図 13】

操作1104：センテンスベースの自動変換



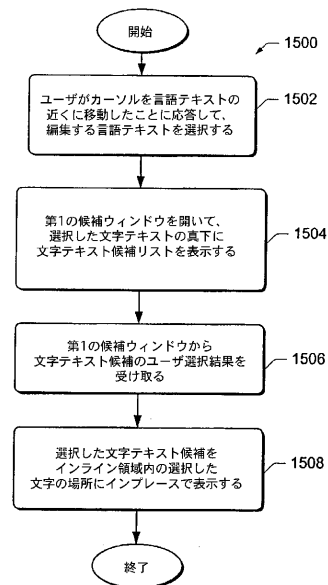
【図 14】

操作1104：確認がある自動変換



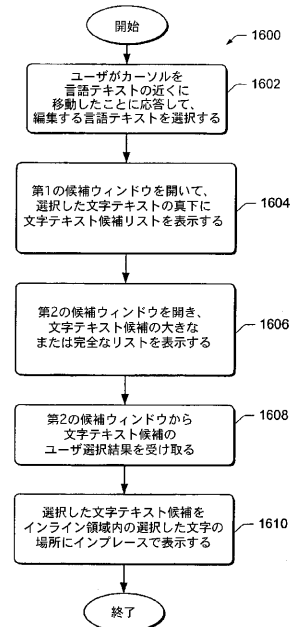
【図 15】

操作1108~1118：第1の候補ウィンドウを  
介したインプレース誤り修正



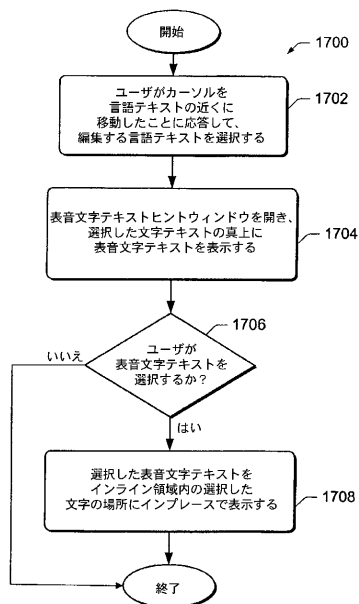
【図 16】

操作1108~1118：第2の候補ウィンドウを  
介したインプレース誤り修正



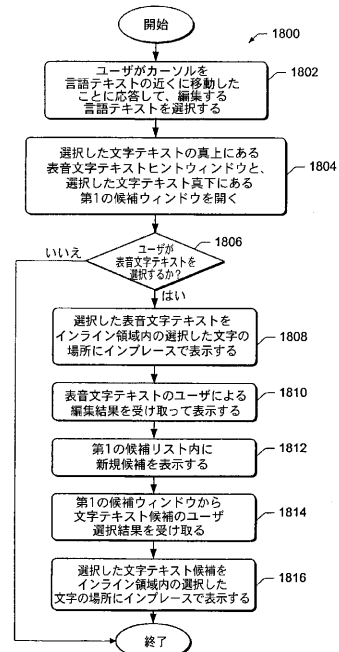
【図 17】

操作1108~1118：表音文字テキスト  
ヒントウィンドウを介した  
インプレース誤り修正

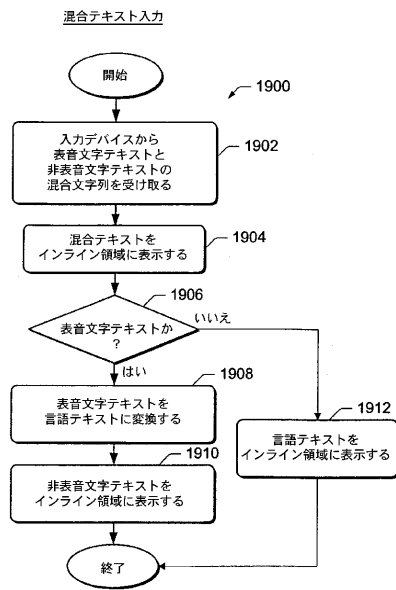


【図 18】

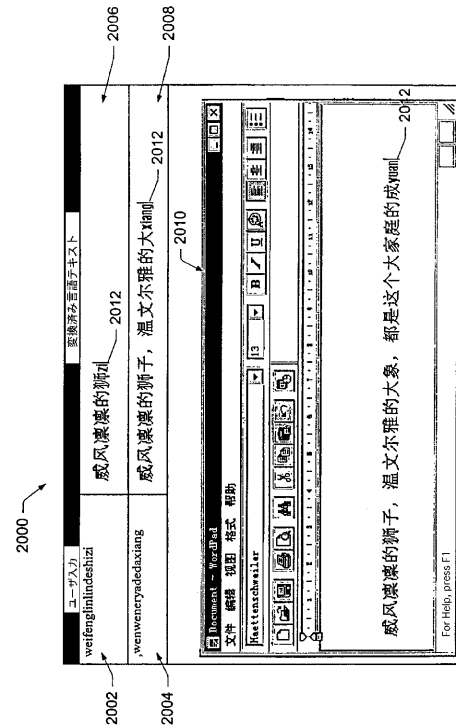
操作1108~1118：表音文字テキストヒント  
および候補ウィンドウを介した  
インプレース誤り修正



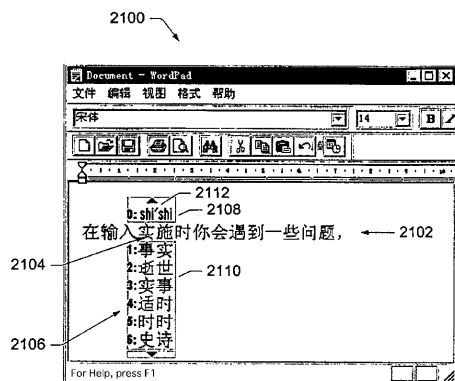
【図 19】



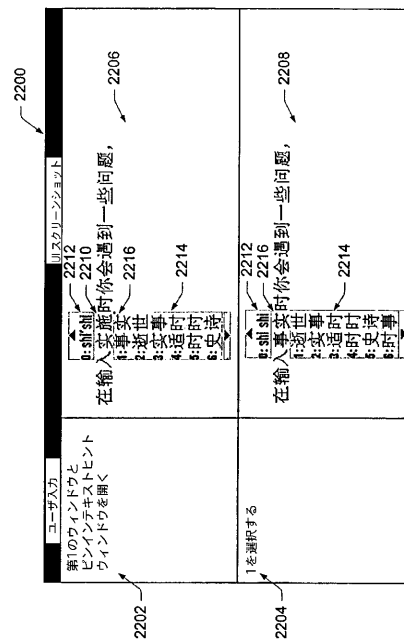
【図 20】



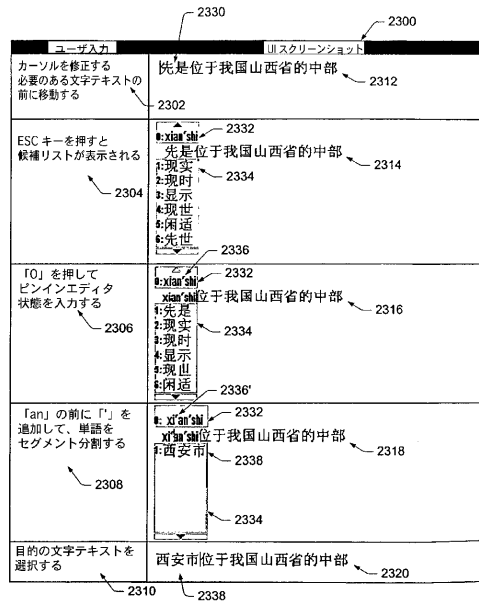
【図 21】



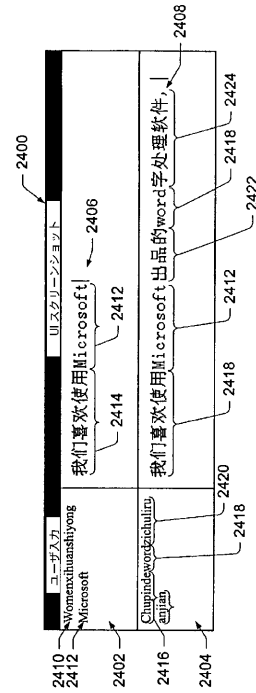
【図 22】



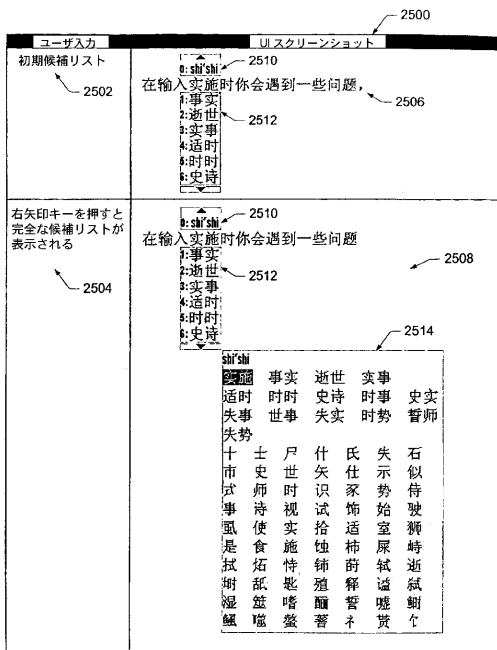
【図 23】



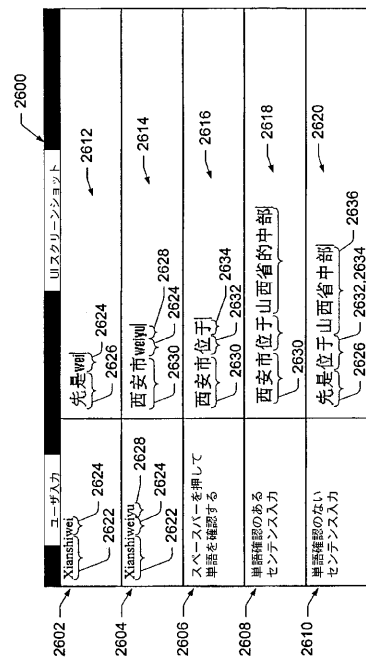
【図 24】



【図 25】



【図 26】





【図 27】

2700 ↙

nihao -----> 表音文字テキスト（たとえば、一般的に  
「hello」に翻訳される中国語ピンインテキスト）

    |          |

你好 -----> 文字テキスト（たとえば、一般的に「hello」に  
翻訳される中国語文字テキスト）

@3M -----> 非表音文字テキスト（一般的に英数字記号を  
含む）

## フロントページの続き

- (72)発明者 ガオ ザン  
中華人民共和国 100080 北京 ハイダン ディストリクト ズイチュン ロード ナンバ  
ー 49
- (72)発明者 ジアン ハン  
中華人民共和国 100080 北京 ハイダン ディストリクト ドンワンズアン コミュニテ  
ィー (番地なし) ビルディング 5 ルーム 702
- (72)発明者 ズエン チェン  
中華人民共和国 100080 北京 ゾン グアン ユアン (番地なし) ビルディング 5  
04 ルーム 105
- (72)発明者 シャノニン リン  
アメリカ合衆国 98007 ワシントン州 ベルビュー 142 プレイス ノースイースト  
3609
- (72)発明者 カイ・フ リー  
アメリカ合衆国 98072 ワシントン州 ウッディンビル 214 ウェイ ノースイースト  
14233

審査官 成瀬 博之

- (56)参考文献 国際公開第95/017729(WO, A1)  
特開昭61-009758(JP, A)  
特開昭63-155262(JP, A)  
特開平10-232863(JP, A)  
Mark Kernighan, 外2名, A Spelling Correction Program Based on a Noisy Channel Model, P  
roc. of the 13th Conference on Computational Linguistics, 米国, Association for Comput  
ational Linguistics, 1990年, Vol. 2, pp. 205-210

- (58)調査した分野(Int.Cl., DB名)  
G06F 17/20-17/26