

メニュー選択に基づく子音による日本語入力手法 Popie の候補選択インタフェースの検討

佐藤 大介* 志築 文太郎* 田中 二郎*

我々はペンによるメニュー選択に基づく日本語入力手法 Popie の開発を行い、ペンによる日本語入力の速度向上を実現してきた。Popie では、FlowMenu を用いて子音列を素早く入力し、それに対応する候補から目的の語を選択することで日本語入力を行う。しかし、これまでの Popie には、目的の語を選択するインタフェースが操作しづらいという問題があった。我々は、いくつかのインタフェースによる比較実験を行い、リストからの候補の選択には、FlowMenu のパラダイムではなく、ダイレクト操作による選択をすることが最も有効であるという結果を得た。

The Study of Candidate Selection Interface for a Menu-selection-based Japanese Input Method Popie

Daisuke Sato[†] Buntarou Shizuki[†] Jiro Tanaka[†]

We have developed Popie: a menu-selection-based method for inputting Japanese text. Since FlowMenu is used in Popie, a user can input the sequence of consonants smoothly. Then, the user selects the candidate from the list inferred by the system. However, the candidate selection method was not efficient in our system. We have designed several methods to solve the problem. We carried out an experiment and we obtained the result that the direct manipulation is the most efficient method.

1 はじめに

近年、PDA やタブレット PC のようなペンで操作するコンピュータが注目されている。ユビキタス社会では、ペンで操作するコンピュータがますます増え、従来のマウスとキーボードで行われた操作が、指やペン、音声による操作に代わり、ペンで操作するためのインタフェースは、重要な役割を持つようになって考えられる。

しかし、コンピュータをペンで操作するためのインタフェースは、従来のキーボードとマウスのために設計されたものを、単に拡張しただけで、ペンに適したものではない。マウスのような相対位置による操作では、画面が広くてもマウスの移動量は変わらない。しかし、ペンでは絶対位置を操作するため、インタフェースが散在していると、ペンの移動量が増え、結果的に使いづらくなってしまふ。

*筑波大学 システム情報工学研究科 コンピュータサイエンス専攻

[†]Department of Computer Science, Graduate School of Systems & Information Engineering, University of Tsukuba

この問題を改善するインタフェースとして、Pie Menu[1] や FlowMenu[2][3] のような、ペンの周囲だけで操作できるメニューが有望である。FlowMenu は、同じパラダイムで動作する文字入力システム Quikwriting[4] を実装し、アルファベットや記号の入力ができるが、日本語のような変換を伴う入力ができなかった。

そこで、我々は FlowMenu(以降フローメニューとする)のパラダイムで、日本語入力が行えるシステム Popie を開発し、評価を行ってきた。Popie は、ユーザの入力した子音列に対して候補を提示するシステムであり、ユーザは目的とする単語を候補から選択することで、日本語を入力することができる。Popie による文字入力実験では、文字認識による入力よりも速い結果を示した [5]。

[5]の実験では、入力全体にかかる時間のうち、子音入力の時間が 62%、候補選択の時間の 30% であった。そして、6割を占める子音入力のインタフェースについて検討した結果、子音を入力するキーの配置を最適化しても、高々 5.5% 程度の速度向上しか望めないという結論を得た [6]。

一方、3割の時間を占める候補選択のインタフェースは、実験後のインタビューで「使いづらい」という答えが多かった。我々は、この問題を解決するため、いくつかの候補選択インタフェースによる比較実験を行い、最適な候補選択インタフェースについて検討を行った。

2章ではフローメニューと Popie のインタフェースについて述べ、3,4章では比較するインタフェースと、比較実験について述べ、最後にまとめる。

2 フローメニューと Popie

フローメニューはドーナツ型のインタフェースを持ち、ドーナツ型の内側の部分をレストエリア、ドーナツ型の外側の 8つの部分をオクタントと呼んでいる。そして、レストエリアやオクタントの境界線をペンで横切ること、操作を行う。フローメニューの利点は、一連の操作が連続的なストロークとして扱えることと、ユーザがメニュー項目の位置を覚えることにより、ジェスチャを使うように素早くメニューを選択できるようになることである。Popie ではこの利点を活かし、子音の選択を素早く行うことができる。

図 1 に Popie を使って子音を 1つ入力する例を示す。まず、Popie システムの初期状態から (a)、“位 AK” とラベルの付いた上側のオクタントにペンを移動し、サブメニューを表示させる (b)。続けて、サブメニューの右上側のオクタント ‘K’ の位置にペンを移動させ (c)、最後に中心部のレストエリアにペンを戻すことで ‘K’ を入力する (d)。

このとき、中心部、上側、右上側、中心部という順番で (d) のような三角形を描くようにすることが、子音 ‘K’ の入力であると覚えれば、メニュー項目を見ずに子音の入力ができる。

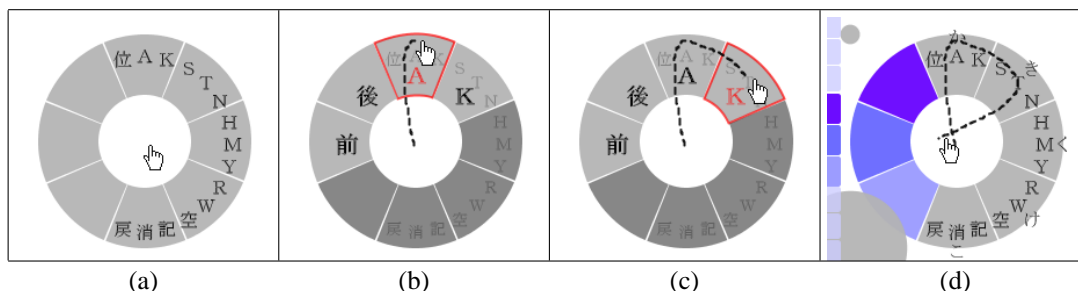


図 1: Popie を使って、子音 ‘K’ を入力する例

2.1 Popie における候補選択

Popie は、子音による日本語入力システム [7][8] であり、入力したい単語の読みに対応する子音の入力に対して候補を提示する。例えば、“AWTHAWS”を入力すると、“インタフェース”などの候補が表示される。便宜的に行は‘A’、‘ん’や‘ー’は‘W’を用いて入力する。Popie は、子音列に対して最適な単語が上位にくるように候補をソートしている。ユーザは提示された候補から、目的の単語を選択することで日本語入力を行う。

Popie のインタフェースを図 2 に示す。Popie は入力選択部と、候補表示部から構成される。入力選択部では子音入力、母音選択、候補の選択など全ての操作を行う。候補表示部には、ユーザの入力に対する候補を表示する。上部の文字列はユーザが入力した子音列である。図 2 は、「インタフェース」と入力しようとして“AWTHAWS”を入力したところである。

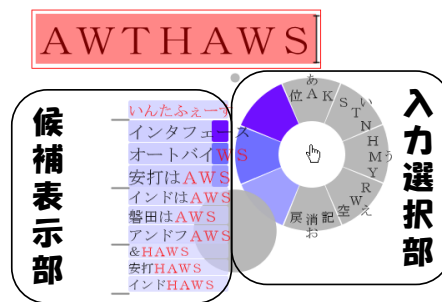


図 2: Popie のインタフェース

次に、候補から目的の単語を選択する操作を図 3 に示す。もし、目的の単語が上位 3 候補に入っていれば、中央の対応するオクタントにペンを移動させ、レストエリアに戻すことで選択できる。一方、目的の単語が上位 3 候補に入っていない場合は、リストをスクロールし、中央の対応する位置に目的の単語を移動させてから選択を行う。

リストのスクロールは、左下側と下側、または左上側と上側のオクタントの間でペンを行き来させることで行う。例えば第 4 位の候補“商社”を選択する場合、まず左下側のオクタントにペンを移動させ (a)、次に下側のオクタントにペンを移動させるが (b)、この時点では変化はない。続いて、ペンを左下側のオクタントに戻すと (c)、リストが一つスクロールされる。最後に、対応するオクタントからレストエリアにペンを戻すと“商社”が選択される (d)。

このとき、(b), (c) の操作を繰り返し行えば、任意の候補を選択できるようになる。

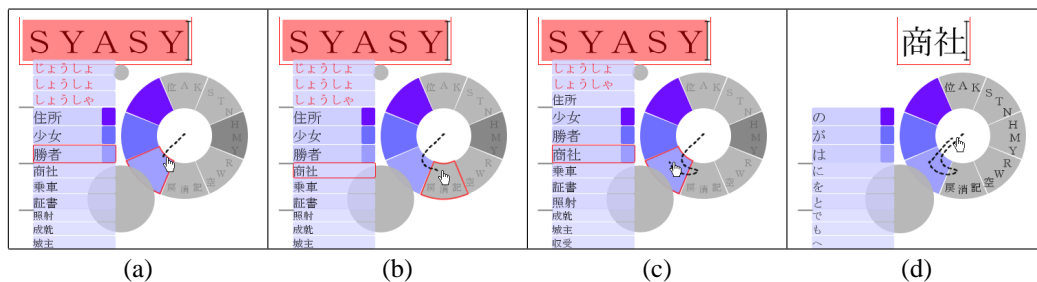


図 3: “住所”と“商社”を選択する例

2.2 候補選択インタフェースの問題点

子音のみの入力では、組み合わせによって爆発的に候補の数が増えるため、システムによる候補のソートが有効でなかった場合、選択操作の負担が大きくなる。我々は、子音列の先頭から母音を決定できるインタフェースを導入し、ユーザが適宜候補を絞り込むことで、選択操作の負担を軽減させてきた。しかし、同音多義語の多い日本語では、母音を確定しても候補数を減らせない場合も多く、さらに全体の入力時間の3割を占めるため、候補選択のインタフェースは重要な役割を持つ。

しかし、従来の候補選択のインタフェースは、以下の3点の問題があり、「使いづらい」というアンケートの結果につながったと考えられる。

1. 候補の上位3つに候補がない場合スクロールが必要になるが、候補のリストを一つずつしかスクロールできないため、目的の単語が候補の下位になるほど、選択にかかる時間がより多くなってしまふ点
2. 候補がリストの下部に表示されていても、中央の3つのオクタントに対応する位置まで、間接的な操作によるスクロールを続けなければならない点

3 候補選択インタフェースの比較

我々は、前述の候補選択インタフェースの問題を解決するため、いくつかのインタフェースの比較実験を行った。比較するインタフェースを設計するため、以下の3つの方法についての検討を行い、従来手法を含めた4つのインタフェースを用いて実験を行った。

スクロール量を変える

1つずつスクロールさせるのではなく、一度に複数個スクロールさせる方法を考える。候補選択に使用できるオクタントは3つであり、1度に3つより多くスクロールさせると、選択できない候補ができてしまうので、従来手法で最大の3つずつスクロールする方法を最初の比較対象とした (Indirect3)。

ダイレクトにスクロール

従来は、フローメニューの枠組みで間接的にリストをスクロールしていたが、リストの上でペンを動かすことでリストをダイレクトにスクロールする方法を考える。リストの中央より下側の部分では上方向にペンを動かしたときだけスクロールし、リストの中央より上側の部分では下方向にペンを動かしたときだけスクロールする (DirectScroll)。図4に、「数値」をリストの中央までスクロールする例を示す。まず、リストの上にペンを移動し (a)、リスト中の「数値」の部分までペンを移動する (b)。このときは何も変化はない。次に、リストの中央までペンを戻すと (c)、「数値」が中央までスクロールされる。スクロール後の選択方法は従来と同じように、対応するオクタントからレストエリアにペンを移動することで行う。

ダイレクトに選択

中央3つのオクタントに対応する位置までスクロールさせて、レストエリアにペンを戻すという動作をしなくても、目的の候補の上でペンを離すことで選択が行えるようにする方法を考える (DirectSelect)。スクロールは、リストの上端、下端を横切ることで、それぞれ3つずつスクロールするようにした。この場合は一度に12個分までスクロールすることが可能であるが、従来の方法の最大値に合わせた。この方法で、図4において「数値」を選択する場合は、最

初に、リストの上にペンを移動し (a), リスト中の「数値」の部分までペンを移動する (b). そして、決定を実行するためにペンを画面から離す.

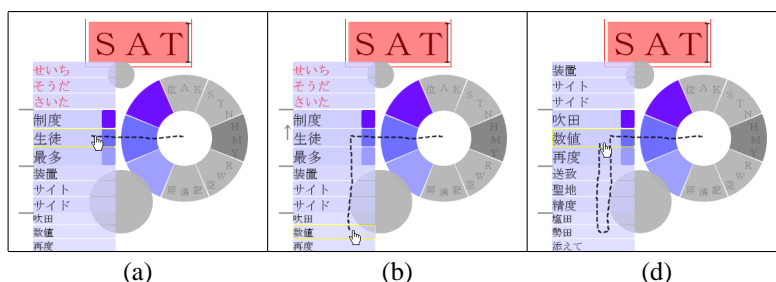


図 4: 「数値」を中央までスクロールする例

4 評価

表 1 に比較するインタフェースの特徴を示す. これらのインタフェースを用い, 候補リストから指定された単語を選択する実験を行った. 被験者は男性 7 名, 女性 1 名の計 8 名, 年齢は 20~32 才であった. 被験者の中で Popie を使用したことがある人は 4 名いた. 被験者には, 10 分程度のインタフェースの説明と練習の後, 各インタフェースで単語を 30 回ずつ選択してもらい, 最後にアンケートに答えてもらった. 実験に使うインタフェースの順番は, 順序による影響を考慮し 8 人とも異なるようにした.

各試行では, 被験者が開始の意思を示した時点で, 選択すべき単語と候補のリストを表示させた. そして, 候補の表示から候補を選択し終えるまでの時間を計測した. ただし, DirectSelect では, ペンを離して候補を選択した後に, フローメニューの中心までペンを戻すまでを計測の対象とした. これは, 実際に文章を入力する際は, 選択に続いて次の子音の入力を行うからである.

選択すべき単語の候補リスト中での順位は, 4~9 位, 10~15 位, 25~34 位, 45~54 位, 75~84 位の 5 つのグループから, それぞれのグループで同じ回数選ばれるようにランダムに選択した. 1~3 位の候補は対応するオクタントを用いてすぐに選択できるので, 今回の実験では対象外とした. 4~9 位は候補表示の最初の段階で, スクロールせずに視認できる範囲の候補群であり, 10~15 位は視認するのに若干スクロールが必要な候補群である. 残りは, 30 位, 50 位, 80 位前後の候補群である.

実験にはタッチパネル付液晶ディスプレイ EIZO FlexScapL661P を使用した (図 5 参照). 使用した計算機の実環境は, WindowsXP Professional, Pentium4 3.0GHz, 1.5GBRAM であり, Popie を実行するための Java のバージョンは J2SDK1.4.2_02 であった.

表 1: 比較するインタフェースの特徴

識別子	スクロールの仕方	選択方法
Indirect1	メニュー上で間接的に 1 つずつスクロール	メニューの中心に戻って選択
Indirect3	メニュー上で間接的に 3 つずつスクロール	メニューの中心に戻って選択
DirectScroll	リストをなでるようにスクロール	メニューの中心に戻って選択
DirectSelect	リストの上端, 下端で 3 つずつスクロール	候補の上でペンを離して選択

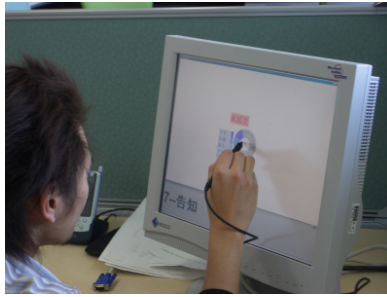


図 5: 実験の様子, タッチパネル付液晶ディスプレイ

4.1 実験結果

図 6 は, 候補の順位グループごとに選択するのに要した時間の平均値をグラフにしたものである. 平均値を計算するにあたり, 間違っ候補を選択したり, 選択すべき候補を見逃したために著しく時間がかかったと思われるデータは, ミスをしたと判断してデータから除外した. ここで, 著しく時間がかかったと判断するのに用いた基準は, 間違っ候補を選択した場合のデータを除いたものの平均値に, その標準偏差の 3 倍を加えた時間より遅かったものとした. また, 図 7 は, ミス率を示したグラフである. ミス率は間違っ候補を選択したものと, ミスと判断したものを合わせた割合として計算した.

次に, 各順位のグループごとに, インタフェース間での優位差を見るため, $** p < 0.01$ と $* p < 0.05$ の T 検定を行った. その結果は表 2 に示す. また, 実験後に行ったアンケートの結果を表 3 に示す.

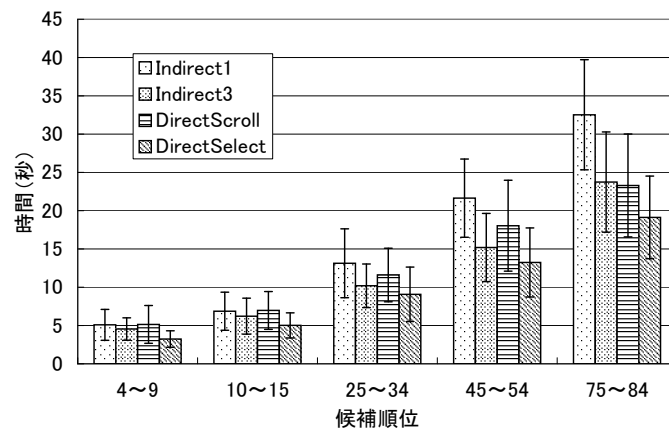


図 6: 選択に要した平均時間

表 2: それぞれのインタフェース間での T 検定 ($** p < 0.01$, $* p < 0.05$)

	4~9 位	10~15 位	25~34 位	45~54 位	75~84 位
Indirect1-Indirect3	×	×	非等分散**	等分散**	等分散**
Indirect1-DirectScroll	×	×	×	等分散**	等分散**
Indirect1-DirectSelect	非等分散**	非等分散**	等分散**	等分散**	等分散**
Indirect3-DirectScroll	×	×	等分散*	等分散*	×
Indirect3-DirectSelect	非等分散**	非等分散**	×	等分散*	等分散**
DirectScroll-DirectSelect	非等分散**	非等分散**	等分散**	等分散**	等分散**

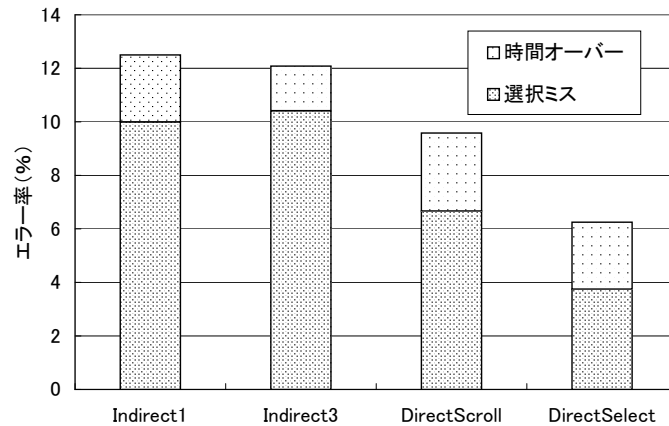


図 7: 各インタフェースでのミス率

表 3: 実験後のアンケートの結果

	Indirect1	Indirect3	DirectScroll	DirectSelect
最も使いやすいと思ったインタフェース			3人	5人
最も使いにくかったインタフェース	4人	1人	3人	
最も疲れなかったインタフェース	1人		2人	5人
最も疲れたインタフェース	5人	3人		
各インタフェースの評価平均(10点満点)	3.38	5.50	6.63	8.50

4.2 実験の考察

実験から、DirectSelect が最も選択に要した時間が短く、他のインタフェースに比べても優位差があることが分かる。またエラー率も最も少ない。アンケートにおいても、使いやすい、疲れないと答えたのが 8 人中 5 人であり、10 点満点の主観評価でも平均 8.5 点と高い値を示した。この結果は、従来のフローメニューのパラダイムによる候補選択インタフェースよりも、ダイレクト操作を用いた候補選択インタフェースの方が効果的な候補選択ができることを示している。

Indirect3 と DirectScroll では若干 Indirect3 が良い結果になっており、30 位前後と、50 位前後では優位差が認められているが、アンケートの評価やエラー率の点から言うと、DirectScroll の方が良い結果となっている。

Indirect1 は候補の順位が低くなるほどに、他のインタフェースに比べて明らかに遅く、エラー率も高くなっている事が分かる。アンケートでも、「使いにくい」、「疲れる」と半分以上の人が回答しており、主観評価も平均 3.38 点と最も低くなっている。これが、従来の Popie の候補インタフェースが使いづらいとされる原因である。

その他、3 つずつスクロールする際に、一度にいったん動いてしまっているため、アニメーションで中間を補完すれば、表示されている候補を把握しやすく、使いやすいのではないかという意見や、DirectScroll や Indirect1 の方が目が疲れにくいという意見があった。このことから、スクロールさせる際のアニメーション等の表示に関しても、操作のしやすさに影響を与えていると考えられる。

4.3 最適な候補選択インタフェースの検討

今回の実験から、DirectSelectのように、候補選択の際にペンを離すという操作により、従来手法に比べて、スクロールが必要な候補選択にかかる時間が優位に減ることが分かった。また、マウスをある程度使いこなすフローメニューの初心者にも、操作感覚という点で、このインタフェースは使いやすいと考えられる。

また、スクロールに関していくつか検討の余地があるが、Indirect1のように、1つずつのスクロールでは使い勝手は良くない。今回は3つずつのスクロールを採用したが、表示できる最大の12ずつスクロールする方法も考えられる。しかしどちらにしても、新たに現れる候補を識別するまでに時間がかかるので、3つ以上のスクロールの場合、あまり大きな差がないと予想される。

むしろ、スクロール中のアニメーション表示を追加し、リストを滑らかに移動させることで、どこから、どこまでが新しく表示されたのかが分かるようにする、候補の認識の助ける方法が必要だと考えられる。

5 まとめ

ペンによるメニュー選択に基づく日本語入力手法 Popie の、最適な候補選択のインタフェースについて検討するため、いくつかのインタフェースを実装し、ユーザ実験により評価を行った。

評価により、選択すべき候補の表示の上でペンを離して決定する、ダイレクト操作による候補選択のインタフェースが、フローメニューを使った間接的な操作による候補選択のインタフェースよりも、スクロールが必要な候補の選択にかかる時間が優位に少ないことが分かった。

今後はスクロール時のアニメーションの有無による、候補選択への効果についても調査したいと考えている。

参考文献

- [1] Don Hopkins. The design and implementation of pie menus. *Dr. Dobb's Journal*, pp. 16–26, 1991.
- [2] François Guimbretière and Terry Winograd. FlowMenu: Combining command, text, and data entry. In *Proceedings of ACM User Interface Software and Technology 2000 (UIST 2000)*, pp. 213–216, May 2000.
- [3] François Guimbretière, Andrew Martin, and Terry Winograd. Measuring flowmenu performance. Technical report, Stanford CS, 2001. CS-TR-2001-02.
- [4] Ken Perlin. Quikwriting: Continuous stylus-based text entry. In *Technical Note of ACM User Interface Software and Technology 1998 (UIST 1998)*, November 1998.
- [5] 佐藤大介, 三浦元喜, 志築文太郎, 田中二郎. ペンによるメニュー選択に基づく日本語入力手法. 日本ソフトウェア科学会第20回大会論文集, 2003.
- [6] Daisuke Sato, Motoki Miura, Buntarou Shizuki, and Jiro Tanaka. Menu-selection-based japanese input method with consonants for pen-based computers. APCHI, 2004 to appear.
- [7] K. Tanaka-Ishii, Y.Inutsuka, and M.Takeichi. Japanese text input system with digits –can japanese text be estimated only from consonants?-. In *Proceedings of Human Language Technology Conference 2001 (HLT 2001)*, March 2001.
- [8] T9. T9 text input home page. <http://www.t9.com>.