

アナログジョイスティックに適した パイメニュー型インタフェースの設計と実装

宮本雅勝† 村松邦彦† 寺田 努† 塚本昌彦†

†神戸大学大学院工学研究科

近年、スペースが限られた環境での入力インタフェースとしてペンとパイメニューを組み合わせた入力方式が提案されている。しかし、文字入力を行うなど多数の項目を選択する必要がある場合、操作が困難であったり、入力速度が遅いなどさまざまな問題点がある。そこで本研究では、省スペースな入力インタフェースの構築を目的として、アナログジョイスティックと階層型パイメニューを組み合わせた新しい入力インタフェースを提案する。半径の異なる円をいくつか表示することでパイメニューを多層化し、文字の使用頻度に応じた領域分割アルゴリズムを用いることで、メニュー項目が増えた場合にも、高速かつ高精度でわかりやすい入力を実現する。

Design and Implementation on a Pie Menu Interface for Analog Joysticks

Masakatsu Miyamoto† Kunihiro Muramatsu† Tsutomu Terada† Masahiko Tsukamoto†

†Graduate school of engineering, Kobe University

Recently, there are several input methods that combine the pen-based input with the pie menu as an input interface in environments where working space is limited. However, they have various problems that the input operation is difficult to learn and the input speed is too slow in the case where there are many items in the menu. In this research, we propose an input interface that combines an analog joystick and a hierarchical pie menu as a new space-saving input interface. The proposed method employs a hierarchical pie menu to enhance the number of items, and a frequency-based area division algorithm efficiently works to expand the area size for important item, using our method an intuitive input interface is achieved with high-speed and high accuracy.

1 はじめに

近年、コンピュータの小型化、高性能化により、コンビニやスーパーマーケットでの商品管理や、医療現場でのカルテ入力などの様々な環境でパーソナルコンピュータが使用されている。

上記の環境においては、通常のデスクトップ環境とは異なり、狭いスペースや立ちながらの作業など動作に制約があるため、マウスやキーボードのようなインタフェースを用いることは困難である。そこで、スペースが限られた環境での入力インタフェースとして、タッチパネルを用いてペンや指で操作を行う方式が普及している。一方、従来のタッチパネルインタフェースは、キーボードとマウスを単純にペン操作へと拡張しただけのものが多く、例えばポインティングに関して、マウスのように相対的に位置を制御する方式では画面が広くてもマウスの移動量は変わらないが、ペンでは直接タッチすることで位置を決定するため、操作対象が散在しているとペンの移動量が増えて使いづらいという問題がある。

これらの問題点を解決するため、ペンでタッチした部分の周囲だけで操作できるパイメニューを用いたインタフェースに関して多くの研究が行われている。[1, 2]パイメニューとは、選択項目を円周上に配置したメニューである。

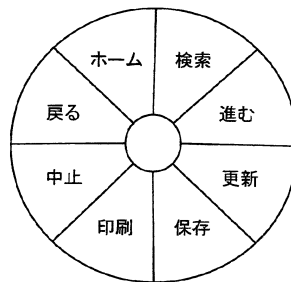


図 1: パイメニューの例

しかし、従来のペンとパイメニューを組み合わせた文字入力インタフェースはメニュー項目の増加が選択のしにくさに直接影響するため、多数の選択肢への対応に適していない。そこで、本研究では、アナログジョイスティックとパイメニューを組み合わせた省スペースで多数の選択肢を高速に入力できるインタフェースを提案する。また、アナログジョイスティックの特性を利用したパイメニュー選択方式と領域分割アルゴリズムを用いることで、高速かつ高精度な入力を実現する。

以下、2章で関連研究について述べ、3章では提

案する適応的パイメニューの設計について述べる。4章では、提案手法の実装について述べ、5章で評価と考察を行う。最後に6章で本論文のまとめを行う。

2 関連研究

ペンとパイメニューを組み合わせた文字入力インタフェースに関して多くの研究が行われている。

Venolia らによる T-Cube[3] は1ストロークで1文字を入力する方式で、最初に表示されているパイメニューのある項目をペンでタッチするとタッチした箇所に新たなパイメニューが表示され、入力したい文字の方向へペンを滑らせることで項目を選択する。しかしこの方式では、2段階の操作を必要とするため入力速度が遅くなる。また、1ストロークで複数文字を入力する手法として FlowMenu がある [5]。FlowMenu はパイメニューの各領域をペンで横切り、元の位置に戻るようにストロークを描くことで複数の文字を入力する。Perlin らによる Quikwriting[6] は、中心領域からペンで領域間を移動し、再び中心に戻ってくることで文字を入力するインタフェースである。しかし、1つの文字を選択する際のストロークが複雑であり、操作の習得に時間がかかるという問題点がある。Mankoff らによる Cirrin[7] は、円周を26分割して、各マスにアルファベットを配置し、ペンが通過した部分のアルファベットを入力することができる。だが、1つ1つの文字が表示されている領域が狭く、隣の領域を通過してしまうなどの選択ミスが起こる可能性がある。

3 適応的パイメニュー方式

パイメニューは選択項目が円状に配置されており、項目数が増加しても一定の時間でメニュー選択ができる。また、項目が円の中心を基準として配置されていることから、操作するデバイスの移動量が少なく済み、十分な場所を確保できない環境でも操作できる。

しかし、パイメニューを多数の選択肢に対応させる場合、それぞれのインタフェースにおいて、1つの項目に割り当てられる領域が細くなり選択ミスが起こりやすい、段階的な操作を導入すると入力速度が落ちる、複雑さのため操作の習得に時間を要するなどの問題点が生じる。

そこで本研究では、狭い環境や立ちながらの作業において簡単に文字入力を行うために、従来手法の問題点を解決する適応的パイメニュー方式を提案する。

3.1 インタフェースの設計

提案するインタフェースは以下の要求事項を満たす必要がある。

- 高速な文字入力
- 正確な文字入力
- 直観的な操作
- 省スペース

高速な文字入力を行うために、デバイスの移動量を少なくすること、1段階の操作で項目を選択できる必要がある。また、意図した項目を正確に選択し入力するために、項目数を増やすと1つの選択項目の領域が狭くなるという問題点を改善し、より単純な入力方式を用いることで、選択項目が増えても視覚的にわかりやすいインタフェースにする。さらに、狭いスペースでの作業や立ちながらの作業など動作に制約がある環境においての使用を想定し、作業スペースが限られた環境での入力インタフェースを実装する。

提案するインタフェースは以下の特徴をもつ。

- 多層のメニュー項目
- 使用頻度に基づく領域分割アルゴリズム
- アナログジョイスティックの採用

パイメニューの角度を細かく分割せずに、半径の異なる円をいくつか重ねることで、パイメニューを多層化して同時に表示する選択項目を増やす。また、選択項目数が増えた場合、全ての項目に対して均等な領域を与えると1つ1つの領域が小さくなるため項目の使用頻度に応じて領域を分割するアルゴリズムを用いる。さらに、アナログジョイスティックをパイメニューと組み合わせることで、ジョイスティックの特性を利用した単純な入力が可能であり、省スペースな入力インタフェースが実現できる。

3.2 領域分割アルゴリズム

文字入力を始めとする多項目入力をスムーズに行うために、提案手法では円を重ねることで項目数を増やす手法を採用している。しかし、重ねる円を増やしすぎると移動距離が長くなるため入力速度が遅くなり、1つ1つの項目を小さくすると選択ミスが起こりやすい。つまり、全ての項目に十分な選択領域を与え同時に表示することは困難である。

そこで、使用頻度の高い項目ほど広い領域を与えるアルゴリズムを導入し、正確で素早い入力を実現

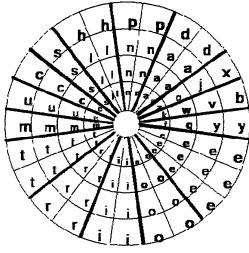


図 2: 縦分割方式

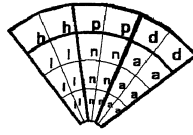


図 3: 重み 6 と 2 に対する文字表示

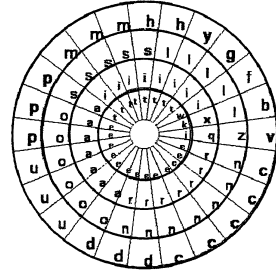


図 4: 環状分割方式

する。領域分割を行う際には、各領域の境界がわかりやすいことが重要である。

多層型のパイメニューにおける境界の決定においては、以下の 2 手法を提案する。

(1) 縦分割方式

ユーザは距離の調節に比べて角度の調節が容易であるという考え方にに基づき、図 2 に示すように縦に領域を分割する。この方式では、12, 8, 4 と 4 の倍数分の領域を与えることによって、丁度よく項目を配置できる。そこで、割り当てる領域数は 4 の倍数として、重みの大きい項目から順に円周状に配置する。ただし、特別な例として 4 の倍数以外に 6, 2, 1 をそれぞれ領域数としてとれるようにしている。領域数が 6 と 2 の項目は図 3 のように丁度よく配置できる。領域数が 4 の倍数の項目を並べ終えた後、領域数が 6 と 2 の項目を図 3 のように配置する。また、使用頻度が非常に低い項目には領域数 1 を与え、パイメニュー内で項目が配置されていない領域に配置する。

(2) 環状分割方式

ユーザは角度の調節に比べて距離の調節が容易であるという考え方にに基づき、図 4 に示すように環状に領域を分割する。この方式では、与える領域数に条件は設定しない。重みの値が大きいものを円の内側から順に並べていき、1 つの領域が横の境界をまたぐことがないように文字を配置する。最後の列の全ての領域に項目を配置し終わったときにまだ配置されてない項目があれば、項目が配置されていない領域に項目を配置する。

重みとは、各項目の使用頻度の割合である。パイメニューの領域数が合計で 96 個のため、各アルファベットがもつ領域数は、合計が 96 となり、かつ各項目の重みの割合と各文字の領域数の割合ができるだけ等しくなるようにする。

以下、各項目に割り当てる領域数を決定するアルゴリズムを説明する。境界を縦にとる場合のアルゴリズムは、

1. 全ての項目を重みの小さな順に並べる。 i 番目の重みを w_i とすると、 $\sum_{i=1}^N w_i = 1$, $w_i < w_{i+1}$ となる。ただし、割り当てる項目の総数を N とする。
2. $i = 1$ とする。また、使用領域数 $a = 0$ とする。使用領域数とは、項目が既に割り当てられている領域の数である。
3. i 番目の項目に対して、最適割り当て領域数 $p_i = \lceil 96 \times \sum_{j=1}^N w_j \rceil - a$ を計算する。ただし、 $i = 96$ のときは、 $p_i = 96 - a$ となる。
4. 1, 2, 4, 6, 8, 12, 16... (以降 4 の倍数) の中で p_i に最も近い数字 n_i を i 番目の項目に対して割り当てられる領域数とする。
5. $i = N$ であれば終了
6. $a = a + n_i$, $i = i + 1$, を行って (3) に戻る。

この手順によって、各項目がもつ重みに応じた領域をもつことになる。一方、境界を横にとる場合のアルゴリズムは、上記の手順 (4) において、 $n_i = p_i$ を i 番目の項目に対して割り当てる領域数とする。

4 システムの実装

4.1 ハードウェア

本研究では、パイメニューに適したインタフェースとして、ペンの代わりにアナログジョイスティックを採用する。アナログジョイスティックは、アナログ値をとるため傾ける方向と傾ける角度によって値を微調整できる。傾けた方向と角度を多層化したパイメニューにマッピングすることで高速に項目を選択できる。また、ジョイスティックを傾けるという動作だけで操作が行えるため、デバイスの移動を必要とせずスペースが限られた環境でも操作が可能となる。さらに、アナログジョイスティックは手を

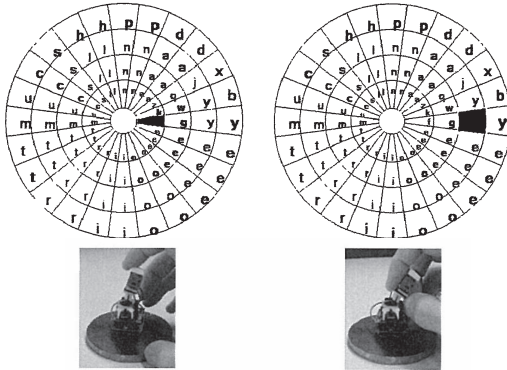


図 5: 文字を選択する例

離すと自動で中央の位置に戻るため連続して素早く入力を行える。

本研究で提案するパイメニュー型文字入力インタフェースのハードウェアとして、無線小型アナログジョイスティックを作成した。アナログジョイスティックは、倒した方向と角度がアナログ値で取得できるジョイスティックと、スティック上部及び側面に配置した2つのボタンで構成されている。無線ICとしてはATMEL社製AT86RF230を採用し、2.4GHz帯でのIEEE802.15.4通信が可能である。装備している2つのボタンのうち1つを押すとマウスカーソルの位置を中心としてパイメニューが表示されるようにした。

4.2 ソフトウェア

多数項目の例として、文字入力を行うシステムを実装した。ジョイスティックに備え付けたボタンを押すと、マウスカーソルの位置を中心としたパイメニューが表示される。パイメニューは、中心を同じ位置として半径の異なる4つの円が重なって表示されている。その4つの円をそれぞれ24分割しており、合計で96個のマスそれぞれに文字が割り当てられている。

パイメニューを表示した状態で、図5のようにジョイスティックを右に少しだけ傾けると中心から右に4つ並んだマスのうち、円の中心から最短距離のマスが赤く光る。その状態から図5のようにジョイスティックを少し深く傾けると、その右2つ隣りのマスが赤く光る。このように、現在ユーザが選択している項目を赤く光らせることで、視覚的にわかりやすい操作を実現している。ユーザがいずれかの文字を選択した状態で手を離すと、その文字が入力される。

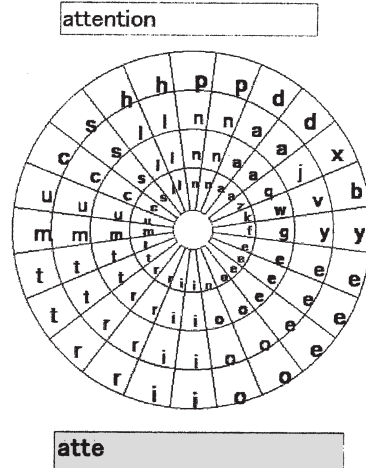


図 6: 評価実験を行う画面

5 評価と考察

提案手法の有効性を評価するため、文字入力速度と精度を評価する。

5.1 実験の手順

実験は、1024 × 768 ピクセルの PC の画面上に実装したパイメニューを表示し、次々と提示されるランダムな英単語をパイメニューを用いて被験者に入力させることで行った。英単語はあらかじめ用意しておいた一般的な単語 2000 語を用いた。

パイメニューの文字レイアウトとしては、縦分割方式と環状分割方式の2種類の他に、文字の領域の境界線を設定せず単にアルファベット順に配置する2種類の並べ方を用意した。ただし、1つは文字に重みを与え使用頻度に応じた領域を与えて配置し、もう1つは各文字に均等な領域しか与えずに配置した。それぞれの方式において重みを計算するためのデータとしては評価のために用意した 2000 語の英単語を用いた。

今回の実験では、4種類の領域分割方式それぞれについて、図6の上方にランダムで表示される単語の入力を20語ずつ行わせ、入力する文字の位置と1文字を入力するまでの平均経過時間、誤った文字を入力した回数を記録した。被験者は研究室に所属する22歳から24歳の男性6名である。

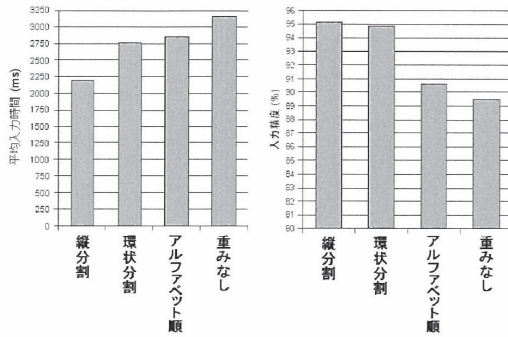


図 7: 入力速度と精度の評価結果

5.2 提案手法の評価結果

提案手法である縦分割方式と環状分割方式、前節で述べた2つの比較手法において、文字入力速度と精度の違いを比較した。実験結果を図7に示す。

この結果より、文字に重みを与える方法の方が重みを与えない方法よりも入力速度が速くて精度も高いことがわかる。さらに、提案手法による文字の並べ方の方が、境界線を設定せずアルファベット順に並べた方法よりも、入力速度が速くて精度が高い。よって、選択領域の境界線をわかりやすく設定することは、文字をアルファベット順に並べて配置することよりも重要だといえる。選択項目の境界が縦と横の場合で比較すると、縦の場合が明らかに入力速度が速くて精度も高い。したがって、アナログジョイスティックは、傾ける方向を調節する方が傾ける角度を調節するよりも容易であるといえる。

次に、最もよい結果が得られた縦分割方式を分析する。表1に縦分割方式における各文字に対する重みと入力頻度の割合、1文字の平均入力時間を示す。この表から e, i, r といった入力頻度が高い3文字の平均入力速度は速く、j, k, w, x などの入力頻度が低い文字は平均入力速度が遅いことがわかる。提案方式は入力頻度が高い文字の入力を高速化し、入力頻度の低い文字の入力の遅さを補うことで全体の平均入力速度が速くなっていることがわかる。

また、表2にパイメニュー上のそれぞれの角度ごとに平均入力時間をまとめたものを示す。表から、ジョイスティックを傾けやすい水平方向と垂直方向である90°、180°、270°に配置されている文字の入力速度が速いことがわかる。0°に配置されている文字は他の3つの角度と比較すると遅いが、0°には重みが1の文字を表示しており、他に重みが1の文字を表示している330°と345°の入力速度と比較すると、約2倍の速度で入力できているため0°の位置の入力速度も十分に速いといえる。また、105°

表 1: 重みと入力頻度と平均入力時間

文字	重み	入力頻度 (%)	平均入力時間 (ms)
e	12	12.1	1464
i	8	8.6	1347
r	8	8.2	1704
o	8	7.9	2628
a	6	7.5	2630
n	6	7.1	2247
t	8	7.0	2448
l	6	6.2	2352
s	4	5.0	2097
p	2	3.7	1722
u	4	3.7	3272
c	4	3.3	2537
h	2	3.0	2957
d	2	2.8	2473
m	4	2.8	1633
g	1	2.3	2205
y	1	2.3	1861
f	1	1.5	2411
v	1	1.3	3523
b	1	1.1	2524
w	1	0.7	4962
x	1	0.6	4054
k	1	0.3	5447
z	1	0.3	2590
j	1	0.1	8270
q	1	0.1	2814

と285°は、90°と270°と同じ文字が割り当てられていたが、被験者は入力が簡単な90°と270°のみを用い、105°と285°の位置は利用されなかった。このことから、水平方向と垂直方向に配置される項目は高速に精度よく入力できるため、これらの方向に配置された項目は他の方向に配置する必要がないと考えられる。この結果を利用すると、水平と垂直の2方向に割り当てる領域の大きさを小さくでき、その分を他の領域に割り当てられるため、より高速で精度の高い領域分割が可能になるといえる。

5.3 既存研究との比較

提案手法のうち、最もいい結果が得られた縦分割方式のパイメニューと、既存のパイメニュー型入力インタフェースであるT-CubeとQuikwritingについて、文字入力の手入力速度と精度の比較を行った。実験結果を図8に示す。

この結果より、提案手法は、Quikwritingと比較して入力速度が速くて精度が高く、優れたインタフェースであるといえる。一方、T-Cubeと比較すると提案手法の精度は低く、文字入力速度もほとんど差がなかった。より高速で精度の高いインタフェースの実現は、今後の課題であるが、前節で述べた角度特性を考慮してインタフェースを改良することで、速度および精度が改善できると考えている。

表 2: 角度と平均入力時間の関係

角度	入力時間 (ms)	角度	入力時間 (ms)
0°	2159	180°	1588
15°	1602	195°	3272
30°	1722	210°	2537
45°	1068	225°	2097
60°	2689	240°	2444
75°	2567	255°	2585
90°	1301	270°	1609
105°	1531	285°	3260
120°	1657	300°	3043
135°	1752	315°	2139
150°	2531	330°	4432
165°	2364	345°	4114

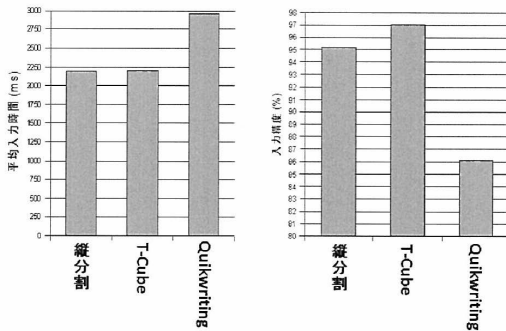


図 8: 既存研究との比較

6 まとめ

本研究では、アナログジョイスティックを用いたパイメニュー方式の文字入力インタフェースを提案し、設計と実装を行った。提案する文字入力インタフェースは、アナログ値を用いてジョイスティックを傾け手を離すという1つの単純な動作で、素早く正確で簡単な文字入力が可能となった。小型のジョイスティックを使用したことで、スペースが限られた環境での入力インタフェースとして有効に利用できると思われる。

今後の課題としては、文字を単に重みが大い順に並べるアルゴリズムだけでなく、ジョイスティックを傾けやすい位置や1番内側の円周での細かい操作の難易度を考慮し、使用頻度に応じた項目を適切な位置に適切な方法で配置するアルゴリズムを考えることで、さらに高速で高精度かつ簡単な入力を実

現することが挙げられる。

参考文献

- [1] D. Hopkins: The design and implementation of pie menus, *Dr. Dobb's Journal*, Vol. 16, No. 12, pp. 16–26, 1991.
- [2] J. Callahan, D. Hopkins, M. Weiser, and B. Shneiderman: An empirical comparison of pie vs. linear menus, *Proceedings of the SIGCHI Conference on Human Factors in computing systems (CHI' 88)*, pp. 95–100, 1988.
- [3] D. Venolia and F. Neiberg: T-Cube: A fast, Self-disclosing pen-based alphabet, *Proceedings of the ACM Conference on Human Factors in Computing Systems (CHI' 94)*, pp. 265–270, 1994.
- [4] G. Kurtenbach and W. Buxton: The limits of expert performance using hierarchic marking menus, *Proceeding of the ACM conference on Human factors in computing systems*, pp. 482–487, 1993.
- [5] F. Guimbretiere and T. Winograd: Flow-Menu: Combining command, text, and data entry, *Proceedings of ACM User Interface Software and Technology (UIST' 00)*, pp. 213–216, 2000.
- [6] K. Perlin: Quikwriting: Continuous stylus-based text entry, *Proceedings of the ACM Symposium on User Interface Software and Technology (UIST' 98)*, pp. 215–216, 1998.
- [7] J. Mankoff and G. Abowd: Cirrin: A word-level unistroke keyboard for pen input, *Proceedings of User Interface Software and Technology (UIST' 98)*, pp. 213–214, 1998.