

片手用キーボードを用いた高速な文字入力手法

A High-Speed Text Input Method for One-Handed Keyboard

片山 拓也 村尾 和哉 寺田 努 塚本 昌彦*

Summary. ウェアラブルコンピューティング環境では装着性の観点から小型のデバイスが望まれる。一般に文字入力デバイスとしてはキーボードが広く普及しているが、キーボードを単純に小型化すると、ユーザビリティに影響を与えてしまう。そこで本研究では、ユーザが既に体得しているキーボード入力能力を活かすために、キー配列を維持しながらキーボードを左右に分割し、どちらか一方のみを装着して用いる手法を提案する。システムは、分割された半分のキーボードの打鍵情報に基づいて、単語の切れ目ごとに入力単語を推測し、優先度を付けて提示する。提案手法を用いることで、従来のキーサイズと入力動作を最大限に保ったままキーボードの大きさを半減できる。

1 はじめに

近年、ウェアラブルコンピューティングに関する注目が高まっている。この環境では、各種デバイスは携帯性や装着性の観点から小型であることが求められる。現在、文字入力には主にキーボードが用いられているが、フルサイズのキーボードは持ち歩きに適さず、衣服への装着も難しい。一方、キーボードの単純な小型化は、キータッチのしやすさなどのユーザビリティを低下させ、入力速度を著しく低下させる。それに対して、文字入力デバイスの小型化は、キー数を減らす手段が一般的である。携帯電話で採用されているめくり方式では、1キーに複数文字を割り当て、打鍵回数で入力文字を決定する。CUT方式は、使用頻度が高い文字を少ない打鍵回数で入力できるようにキーを割り当てる。Wearable Half Keyboard[1]は図1(左)に示すようにキーボードの左半分を切り取ったデザインで、右半分にある文字を打つにはスペースキーを押しながら対応するキーを押す。キーの対応は図1(右)のような写像型を採用している[2]。Twiddler[3]は12キーの同時入力の組み合わせでアルファベットを入力する。これらはいずれも片手入力を想定しており、少ないキー数で全入力を賄うために同時入力などの特殊な入力を要求する。そのため、習熟に時間を要する上、両手入力のキーボードに比べて入力速度が大きく劣る。

そこで本研究では、既存のキー配列を維持しながらキーボードを左右に分割し、どちらか一方のみを装着して用いる入力方式を提案する。提案手法では、ユーザは片方の手は従来と同様にキーボードを打鍵して、もう一方の手は机や腿などを空打ちする。そ

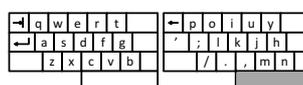


図 1. Wearable Half Keyboard 配列



図 2. 実機プロトタイプ

の際、空打ち動作で発生するキー入力の打鍵間隔を利用して、空打ちした手の打鍵数を推測する。そして、入力された単語を推定し、優先度順に候補を提示する。提案手法では、従来のキーボードの配列をそのまま用いており、従来のキーボードの入力動作を最大限維持しているため、ユーザが既に体得しているキーボード入力能力を活かして大きさを半減できる。本研究では英文入力向けの実機プロトタイプ(図2)を作成し、入力速度を調査した。

2 システム設計

2.1 動作フロー

システムの動作例を図3に示す。図3は右手打鍵時の“keyboard”の入力例を示している。システムは英単語データとユーザの打鍵速度データを保持している。提案手法では、単語探索部内で打鍵手の入力度に英単語の候補を絞り込み、優先度決定部内で打鍵間隔からその間に含まれる非打鍵手の打鍵数を推測し、推測された打鍵数に応じて候補に優先度を付けて提示する。そして、単語の切れ目で単語選択の状態に入り、入力単語を候補の中から選択する。その後、次の単語の入力が開始されると選択した単語が出力される。ここで、入力単語が候補の最上位にある場合は選択動作が必要ないため、従来のキーボードの入力動作が維持される。

2.2 単語探索部

提案システムでは、Web ページをクロールして作成した英単語データをトライ木構造に格納する。

Copyright is held by the author(s).

* Takuya Katayama, Kazuya Murao, and Masahiko Tsukamoto, 神戸大学大学院工学研究科, Tsutomu Terada, 神戸大学大学院工学研究科, 科学技術振興機構さきがけ

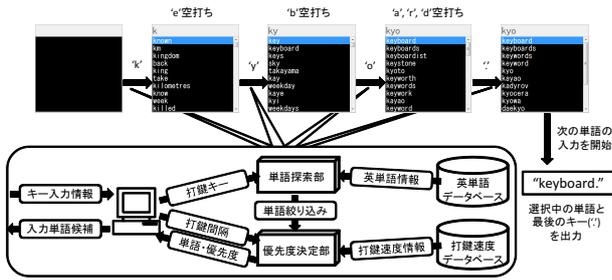


図 3. システムの動作フロー (右手打鍵時)

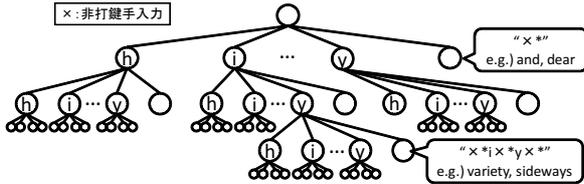


図 4. 右手打鍵時のトライ木

トライ木は、根から自身までのルートがキーに対応している。根には空の文字列が対応し、あるノードの配下にある全ノードは、自身に対応する文字列を接頭部としてキーにもつ。右手を打鍵手にした時のトライ木の構造を図 4 に示す。各ノードは右手入力文字をもつ 10 ノードと文字をもたない 1 ノードを子としてもつ。一方、自身が文字を持たないノードは葉ノードとなり、自身のキーと左手入力文字の組み合わせから成る単語が格納される。

提案システムでは根を始点として、打鍵手の文字入力があった時に、それに対応する子ノードに移動する。その後、単語の切れ目としてアルファベット以外のキーが入力されると、子の葉ノードに格納されている単語が入力単語の候補となる。また、提案システムでは、辞書に収録されていない単語に対応するために、前述した Wearable Half Keyboard と同様の反転入力の機構を備えている。この入力機構は候補の絞り込みとしても使用できる。

2.3 優先度決定部

優先度計算のために、非打鍵手入力数 n に応じた打鍵手の打鍵間隔 t_n を取得する。ここで、 n の上限を 4 に設定する。そして、 n 毎の打鍵間隔の平均 t_{μ_n} と標準偏差 t_{σ_n} をユーザの打鍵速度データとして用いる。非打鍵手入力数の推測には正規分布の確率密度関数の高さを用いる。打鍵手での打鍵間隔 x が与えられた時の非打鍵手入力数が n の確率は、以下のように平均 t_{μ_n} 、分散 $t_{\sigma_n}^2$ の正規分布の確率密度関数 f_n の高さ $f_n(x)$ で表される。

$$f_n(x) = \frac{1}{\sqrt{2\pi t_{\sigma_n}}} \exp\left(-\frac{(x - t_{\mu_n})^2}{2t_{\sigma_n}^2}\right)$$

そして、打鍵間隔 x に対して全ての n について $f_n(x)$ を計算した後、以下に示す式で打鍵間隔 x の時の非打鍵手入力 n の重み $p_n(x)$ を計算する。

$$p_n(x) = \frac{f_n(x)}{\sum_{i=0}^4 f_i(x)} + \alpha$$

ここで、 α を加えることで、打鍵間隔が平均値から大きくずれた場合の影響を抑える。

最後に全打鍵間について $p_n(x)$ を計算した後、単語の優先度 W を計算する。優先度の初期値 W_{base} は辞書作成中の出現回数とする。そして、打鍵手入力が $k_1 k_2 \dots k_n k_{n+1}$ (k_{n+1} は非アルファベット) で k_i と k_{i+1} の打鍵間隔が x_i であった時、“[非打鍵手 k'_0 文字] k_1 [非打鍵手 k'_1 文字] $k_2 \dots k_{n-1}$ [非打鍵手 k'_{n-1} 文字] k_n [非打鍵手 k'_n 文字]” で表される単語の優先度 $W(k'_0, k'_1, \dots, k'_n)$ は以下のように計算される。

$$W(k'_0, k'_1, \dots, k'_n) = W_{base} * \prod_{i=1}^n p_{k'_i}(t_i)$$

また、探索を容易にするために、候補をアルファベット順や文字数順に並び替える機能を実装した。単語選択後に選択された単語を元にユーザの打鍵速度データを更新する。

3 評価

3.1 予備実験

提案手法の有効性を評価するために、提案手法を通常キーボード上で用いた際の入力速度を調べた。3名の被験者の入力をキーボード、提案手法(右, 左), Wearable Half Keyboard 方式, スマートフォンのソフトキーボード, 携帯電話のめくり方式で比較する。結果、全ての被験者がキーボードで最も早い入力を実現した。2名が提案手法(左), 1名がソフトキーボードが次いで高い入力速度を実現した。

提案手法とソフトキーボードの入力速度は、ユーザの好みや入力する単語によって前後することが分かった。しかし、本評価に使用したソフトキーボードでは、ユーザは打鍵時の振動から打鍵感を得るのみで、ホームポジションの概念がないため、指先の感覚では自分の指の位置を把握できない。それに対して、提案システムでは通常のキーボードを用いる時と同様に、ホームポジションを把握してタッチタイピングができる。ウェアラブルコンピューティング環境では、街中で入力を行うので入力デバイスを注視しながらの入力は危険を伴う。よって、タッチタイピングが出来る提案手法はよりウェアラブルコンピューティング環境向きの入力方法だと言える。

3.2 実機プロトタイプを用いた習熟実験

前節で入力速度が高かった左手入力用の実機プロトタイプを作成して、4名の被験者で提案手法の習熟の評価を行った。期間は5日間で1回の実験は10分程度の練習と15分程度のデータ採取から成る。なお、反転入力による絞り込みは用いずに入力させた。

各被験者のキーボードと提案手法の入力速度の推移を表 1 に示す。被験者 A, B は1日目から入力速度が安定して、自身のキーボードの約 30% の入力速度に収まった。一方、被験者 C, D は3日目あたりで

表 1. 習熟実験結果

被験者	キーボード	入力速度 (WPM)				
		1日	2日	3日	4日	5日
A	59.6	17.0	16.3	19.6	18.8	16.2
B	52.2	13.9	13.5	14.5	14.7	18.3
C	20.2	8.4	11.1	12.4	12.6	12.9
D	21.4	8.4	10.7	11.8	12.5	12.4

入力速度が安定し、キーボードの約60%の入力速度を実現した。入力時間の内訳として、打鍵時間（最初の打鍵から候補選択に入るまでの時間）と選択時間（候補選択に入ってから候補を決定するまでの時間）を調べたところ、全ての被験者の打鍵時間は2日目には収束していた。これから提案手法の利用に伴う非打鍵手の空打ち動作はすぐに習得できると言える。一方で、選択時間は安定せず、最終日では平均で入力時間の80.7%を選択時間が占めている。選択時間が安定しない原因を明らかにするために入力単語の提示位置を調べたところ、1日目では入力単語が平均61.9%の割合で最上位に提示されていたにも関わらず5日目では平均39.9%まで下がっていた。そこで、非打鍵手入力数の推測の精度を調べたところ、予備実験時には90%以上の精度が出ていたにも関わらず、本評価時の非打鍵手入力数の推測精度は50%を下回った。結果を細かく解析したところ、過半数の打鍵間で非打鍵手入力数が0と推測されている一方で、約10%が4と推測されていることが分かった。これは、外れ値が打鍵速度データの中に取り込まれて引き起こされていたと考えられる。具体的には、打鍵のもたつきなどが挙げられる。

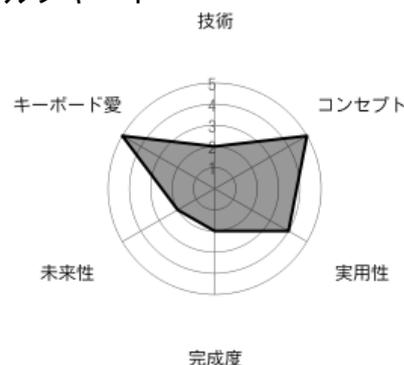
4 おわりに

本研究では、キーボードのキー配列を維持しながら左右に分割し、片側のみを用いる入力方式を提案した。提案手法によってユーザビリティを低下させることなく、ユーザが既に体得している能力を活かして、キーボードの大きさを半減できる。今後は入力速度の改善と日本語入力への対応を予定している。

参考文献

- [1] Matias Wearable Half Keyboard: <http://www.halfkeyboard.com/wearable/index.html>.
- [2] E. Matias, I. S. MacKenzie, and W. Buxton: Half-qwerty: Typing with One Hand using Your Two-Handed Skills, Companion of the CHI 1994 Conference on Human Factors in Computing Systems, pp. 51-52 (1994).
- [3] Twiddler 2: <http://www.handykey.com/>.

アピールチャート



未来ビジョン

提案システムは、ウェアラブルコンピューティング普及のきっかけになる可能性を秘めている。現在、ウェアラブルコンピューティングの適用例として、健康管理や行動記録などのサービスが展開されているが、これらはいずれもバックグラウンドで起動して異常時にメッセージを受けるといふ、受動的なサービスである。一方、ユーザの端末入力が要求されるような能動的なサービスはなかなか普及しておらず、その大きな要因が文字入力デバイスの不整備にあると考えられる。ウェアラブルコンピューティングには、リアルタイムの情報提供、センサを利用したコンテキストウェアサービス、外部コンピュータとの連携サービスなどの画期的なサービスの可能性が秘められているにも関わらず、文字入力を行うのが煩わしく、Yes/No 程度の回答しか素早く行えないという

状況であれば、ユーザは携帯電話、スマートフォンなどの既存のサービスで妥協してしまうだろう。提案システムは、既に体得しているキーボードの入力能力を活かした入力が可能なので、ウェアラブルコンピュータを使い始めるにあたってのハードルを排除する。

さらに、提案システムは片手用文字入力デバイスとして、ウェアラブルコンピュータ以外の適用先も考えられる。例として、車内入力がある。現在、広く流通しているカーナビの操作部分は運転席と助手席の間にあるので両手での操作が困難である。提案システムを従来の操作部分に設置すると、右ハンドル車の場合、運転席からは左手用文字入力デバイス、助手席からは右手用文字入力デバイスとして扱える。他にも、片手を怪我をしてしまった時の入力、お菓子を食べて汚れた手でキーボードを触りたくない時の入力、恋人と繋いだ手を離したくない時の入力などに適用できる。