

ペンを握る動作を用いた離散入力操作の検討

An Exploration of Discrete Input Operation Using Gripping Motions

鈴木 優 三末 和男 田中 二郎*

Summary. 本研究ではペングリップを握る力を利用したインタラクション手法, Gripping を提案する. Gripping とは, ペンを握った状態でさらにペングリップを強く握る操作である. Gripping はペンから指を離すことなく行えるため, 従来のペンの使いやすさを保持したまま入力操作を増やすことができる. さらに, ペンをタッチディスプレイから離れた状態で操作を行うことができるので, タップ操作とは独立した制御が行える. Gripping をコンピュータへの入力として扱う方法として, 連続的な入力として扱う方法と離散的な入力として扱う方法の2種類が考えられる. 既存のペン入力インタフェースはストロークという効果的な連続的な入力手法を既に有している一方で, 離散の入力手法はほとんど有していない. 離散的な入力を可能にすることで, たとえばキーボードでのショートカットキーのような操作が行えるようになり, 操作性向上が期待できる. そこで我々は Gripping を離散的な入力として利用することが有用であると考え, Gripping による離散値入力に必要な実験を行った. 実験では, 人間が実用的に利用することができる離散値の段階数 n , および入力した離散値を選択するためのトリガー操作について検討した. 我々は Gripping を行うためのスタイラス, Pressure-Sensitive Stylus を開発し, これを用いて実験を実施した.

1 はじめに

ペンはコンピュータが発明されるはるか以前から利用されており, ペンの持ち方や書き方などの使用方法もイディオム化されてきた. ペン入力インタフェースはペンを模したデバイスであり, ペン・イディオムを利用するインタフェースである. よって, ペンを使ったことがある人間にとって, ペン入力インタフェースは高いユーザビリティを持つデバイスであるといえる.

しかしながら, 現在のペン入力インタフェースはコンピュータを操作するための十分な操作性を有していない. 我々はペン入力インタフェースの入力操作の少なさがその原因の一つであると考えている. そこで, 我々はペン入力インタフェースの入力操作を増やすことを試みている. 単純にペンに多くのボタンを付加することでも入力操作を増やすことはできるが, そのような方法ではペン・イディオムを維持できないためにペンの使いやすさを失ってしまう.

我々は, ペンの使いやすさを失わずに入力操作を増やすことができるインタラクション手法, Gripping を開発した. Gripping とは, ペンを握った状態でさらにペングリップを強く握る操作である. Gripping では握って使うというペン・イディオムを維持することができるため, ペン型デバイスとしての使いやすさを維持できる. また, 指をペングリップから離すことなく, ペンを握ったまま操作を続行することが

可能になる. さらに, ペンをタッチディスプレイから離れた状態で操作を行うことができるので, タップ操作 (ペン先をディスプレイに接地させる操作) とは独立した制御が可能である.

ペングリップを握る力をコンピュータへの入力として扱う方法として, 連続的な入力として扱う方法と離散的な入力として扱う方法の2種類が考えられる. ペン入力インタフェースには, ストローク操作 (ペン先を接地させたまま動かす操作で, マウスのドラッグ操作に相当) という連続的な入力を行う手法が既に備わっており, ペン入力インタフェースはこれを得意としている. 一方, ペン入力インタフェースには実用的な離散値入力手法がほとんど備わっていない. 離散的な入力を可能にすることで, たとえばキーボードでのショートカットキーのような操作が可能になり, 操作性向上が期待できる. そこで, 我々は Gripping で離散的な入力を行うことを検討する. Gripping を用いた離散値入力を行えるようにするためには, 人間が実用的に利用可能な離散値の段階数 n , および離散値入力を決定するトリガーに適した入力手法を検討する必要がある. 我々はこれらを決定するための実験を実施した.

2 関連研究

ペン入力インタフェースの操作性を改善することを目的とした研究は数多く行われている. そのアプローチは大きく分けて, ハードウェア面の改善とソフトウェア面の改善の2つに分類することができる. ここでは, それぞれのアプローチで行われている研究を紹介し, 我々の研究の位置づけを明確化する.

Copyright is held by the author(s).

* Yu Suzuki, Kazuo Misue and Jiro Tanaka, 筑波大学大学院システム情報工学研究科 コンピュータサイエンス専攻

2.1 ハードウェアの改良

我々は過去にペンの空中での動作を利用するインタラクション手法を開発した [13]. rolling, shaking, swinging という人間が自然に行える 3 つの動作利用して、ペンの使いやすさを維持したまま新しい入力操作を増やした. rolling はペン軸周りにペンを回転させる, shaking と swinging はそれぞれペンをペン軸方向, ペン軸に垂直な方向に振る動作である. Bi らも我々と同様に rolling を利用した入力操作に関する研究を行っている [2].

Miura らはスタイラスをホルダーに挿した状態でのスタイラスの平行移動量と, スタイラスの回転量を利用するインタラクション手法を開発した [6]. Siao らはペンを使うときに手のひらで紙を押さえるという動作をスイッチ入力として利用するインタラクション手法を提案した [11].

ワコムが開発しているタブレットでは, ペン先の XY 座標だけでなく, 筆圧やペンの傾きを検出することができる. その筆圧やペンの傾きを応用した研究も存在する [8, 9, 14].

これらの研究もペン入力インタフェースの入力操作を増加させることを試みている点では我々と共通である. 我々はペングリップを握るというペン入力インタフェースを使う上で必須の動作を利用して新しい入力操作を開発する. 筆圧も必須の動作を利用しているが, タップ操作と干渉するという欠点がある. 一方, 提案する Gripping はペンが空中にある状態でも操作が行えるため, タップ操作とは独立した制御が可能である.

2.2 ソフトウェアの改良

Hopkins はペン入力インタフェースで扱いやすい放射状メニュー, Pie Menu[4] を提案している. さらに, Kurtenbach らは Pie Menu のメニュー表示がなくても, ペンの動きだけでメニュー項目を選択できる手法, Marking Menu[5] を提案している. 階層化された Marking Menu を操作する場合, ジェスチャのような感覚でメニュー項目を選択できる.

メニュー以外のスタイラス向けインタフェースに関する研究もさまざまなものがある. GUI 上でコマンドを実行する操作として, Accot らの提唱したクロッシング [1] という手法がある. クロッシングとはストロークによりターゲットを横切る操作である. クロッシングはタップと比較して正確な操作を行いやすいという特徴がある. Smith らは円を描くジェスチャによるスクロール操作 [12] を提案している. ディスプレイ上をスタイラスでタップするとその位置にガイドが表示され, ガイド上で円を描くジェスチャを行うことでスクロール操作が行える.

このように, 多くの研究では入力操作としてストロークを活用している. ストロークは操作位置と操作対象を直接視認しながら行える連続的な入力であり,

ペン入力インタフェースの特徴的な操作である. そして, これらの研究成果からわかるように, ストロークは使い勝手の良い連続値入力手法である.

一方で, ペン入力インタフェースは離散的入力をあまり得意としていない. ボタンや Pie Menu 等の GUI を利用すれば間接的に離散入力が可能であるが, 我々は GUI を介さずにペン入力インタフェースのみで行える直接的な離散入力の実現を目指す. 直接的な離散入力の実現により, ディスプレイに目ややることなく利用者自身の力感覚のみで入力操作が行えるようになり, ペン入力インタフェースの操作性向上が期待できる.

3 提案するインタラクション手法

3.1 基本アイデア

入力操作を増やす上で, ペンを使う上で人間が行っている動作を利用するアプローチをとった. そのような動作を利用することで, 入力のために新しい動作を学習する必要がなく, 普段の何気ない動作を少し意識して行うだけで新しい入力操作を行うことができるというメリットがある.

人間はペンを使うときに, ペングリップを親指, 人差し指, 中指の 3 本の指で握っているが, 握る強さはほとんど意識していない. しかしながら, 人間の指先には高密度の受容体が存在し, 身体の中でも特に鋭い感覚を持つため, ペンを握る力は細かく調整できると思われる. そこで, 我々はペンを握る強さをインタラクションに利用することを考えた.

本研究では, 上記の考えを基に考案したインタラクション手法, Gripping を提案する. Gripping では, ペンを使うときに必ず行う “ペンを握る” という動作を利用する. Gripping はペンを握った状態でさらにペングリップを強く握るという非常に単純な操作である.

3.2 予想される有効性

Gripping ではペンを握る動作を利用するため, ペンから指を離したり, ペンを持ち替えたりする必要がない. よって, ペン型デバイスの使いやすさを維持できると同時に, 通常のペン操作と Gripping 操作を連続的に行うことができる.

Gripping のように, ペン入力インタフェースの入力操作を増やす手段としては筆圧やバレルボタンがある. ここで, これらの入力操作と Gripping との差異を表 1 に示す. 表 1 では入力値と操作場所の 2 つ観点から比較している. Gripping と筆圧は連続値と離散値の両方を入力することができる一方, バレルボタンは 2 値の離散値しか入力できない. Gripping はペンとディスプレイが接している状態と接していない状態のどちらでも入力を行うことができる一方, 筆圧とバレルボタンはディスプレイに接している状

表 1. Gripping と筆圧、バレルボタンの比較

	入力値	操作場所
Gripping	連続値／離散値	空中／平面上
筆圧	連続値／離散値	平面上
バレルボタン	2 値の離散値	平面上（実装）

態でしか入力操作が行えない。バレルボタンは実装次第で空中での状態でも操作可能であるが、筆圧はペン先とディスプレイ面にかかる力なので操作時に必ずディスプレイに接地する必要がある。

よって、ペンがディスプレイに接していない状態、つまりタップ操作とは独立して連続値と離散値の両方を入力できることが Gripping のアドバンテージである。さらに、Gripping と筆圧を組み合わせることで、2つの値を同時に入力するという今までのペン入力デバイスにはない、自由度の高い操作も可能となる。

3.3 離散値入力の重要性

ペン入力インタフェースでも離散的な入力を可能にすることで、キーボードでのショートカットキーのようにディスプレイ上の GUI 操作が不要な、利用者の力感覚のみで行える入力操作が可能になり、操作性の向上が期待できる。よって、Gripping の離散値入力はペン入力インタフェースの操作性向上に向けて重要であると考えた。

4 Pressure-Sensitive Stylus

我々はペングリップにかかる指の力を感知できる Gripping 用のデバイス、Pressure-Sensitive Stylus(以下、PS Stylus)を開発した(図1)。

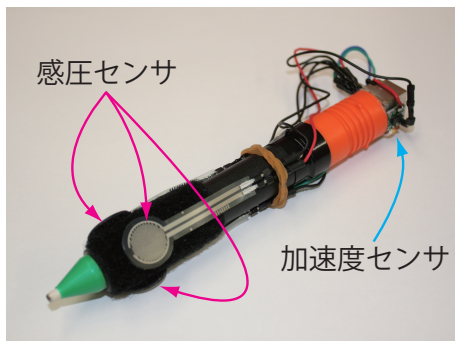


図 1. Pressure-Sensitive Stylus

4.1 デバイスの構成

3本の指にかかる力を検出するために感圧センサ FSR402 を3つ使用した。ペンとセンサはマジックテープを利用して固定した。これは、センサ位置を

自由に変えられるようにするためである。将来的にはペングリップを全て覆うことが可能な感圧センサの利用を想定している。さらに、ペンの上部には加速度センサを配置してある。我々の過去の研究[13]と同様に、ペン自体の姿勢や動きを検出するために利用する。これは実験で用いるが、具体的な利用方法は次章で説明する。両センサにより検出したアナログ値は Arduino マイコンを用いて AD 変換され、PC ヘデジタル信号として送られる。

4.2 デバイスの設計

PS Stylus は 30～500g の力を検出するように設計した。人間は一定の刺激を連続して与えられると、刺激に対する感度が弱くなるという特性を持つ。つまり、指先に強い力を長時間加え続けるとそのセンシング能力は徐々に低下する。よって、Gripping に利用する力の上限として 500g を採用した。また、ペンの重量だけでペンを握る力が検出されることを避けるために下限を 30g とした。

PS Stylus は力の強さを 1024 段階で識別するように設計した。アプリケーションでは 0～1023 の圧力レベルとして取得できる(値が大きいほど力が強い)。センサの特性上、センサの出力値と実際にかかる力は線形ではなく対数特性を持つため、ソフトウェアで特性が線形になるように補正した。これにより、センサの出力値と実際の圧力は概ね比例するようになったが、完全な線形特性には補正できていないため、かかる力が弱いときに出力値がやや大きくなる特性がある。加速度センサでは、3軸の加速度を $\pm 2G$ の範囲で検出できる。PS Stylus の時間分解能は 50Hz であるため、操作に対する遅延を感じずに操作することができる。

5 実験

5.1 目的

本実験は、人間が快適に区別して使い分けことができる握る力の段階数を調査することを目的とする。握る力を区別するというタスクには、フィードバックが重要になると考えられる。そこで、フィードバックの重要性についても調査する。

また、Gripping を利用するためには入力した離散値を選択するためのトリガー操作が必要となる。マウスを使った操作の場合、一般的にはクリックがトリガー操作になっている。よって、Gripping のトリガー操作としてクリックに相当するタップを使用することも考えられるが、空中で操作が完結しなくなるため Gripping の大きな特徴が損なわれてしまう。そこで我々は空中で行うことができる4種類のトリガー操作、*Keeping*、*Quick Release*、*Finger Release*、*Swinging*を用意した。*Keeping*は握る力を一定時間(本実験では 1000ms)維持する操

作, *Quick Release* はペンに加えている力を素早く緩める操作, *Finger Release* はペンから人差し指のみを離す操作, *Swinging* はペンを振る操作である. *Swinging* は我々が [13] で提案したインタラクション手法の 1 つであるが, *Swinging* は空中で行え, かつ手首を回転させるだけで行える簡単な動作で行えるため, 我々は今回これを採用した. *Swinging* は加速度センサの出力値から, その他は感圧センサの出力値から操作を検出する. 我々はこの中から Gripping に適したトリガー操作を選定する.

5.2 被験者

被験者は 22~26 歳の男性 6 名, 女性 2 名の合計 8 名, 7 名は右利き, 1 名は左利きであった. ペンの持ち方は被験者ごとに多少異なっていたが, 全員が 3 本の指を用いていた. 各被験者のペンを握る位置に応じて PS Stylus の 3 つの感圧センサ位置を微調整した. これにより被験者はセンサ位置を意識することなく普段通りにペンを握ることができる.

5.3 タスク

握る力を制御してターゲットを選択するタスクを行った. 被験者にはボックスとカーソルが提示される. ボックスのサイズは 600×800 ピクセルで, カーソルは握る力に応じてボックス内を垂直方向に移動する. 1024 段階の圧力レベルは 800 ピクセルに均一にマッピングされており, 圧力レベルが 0 のときにカーソルはボックスの最下部に, 1023 のときに最上部に表示される. ボックス内にはターゲットが 1 つ提示される. 被験者はそのターゲットにカーソルを合わせ, それを選択するためのトリガー操作を行う. トリガー操作は *Keeping*, *Quick Release*, *Finger Release*, *Swinging* の 4 種類であり, それぞれ, 力を維持し始めたときの力, 力を緩める直前の力, 指を離す直前の力, ペンを振る直前の力を測定した.

視覚的フィードバックが Gripping 操作に与える影響について調査するために, 2 種類のフィードバックを用意した. 完全フィードバック (以下, FF) では, ターゲットとカーソルの動きが提示される. カーソルがターゲット内に入るとターゲットの色が変わるという視覚的フィードバックも備える. 一方, 部分フィードバック (以下, PF) では, 初期状態ではターゲットとカーソルが提示されているが, 試行が始まるとカーソルは非表示になる. カーソルがターゲット内に入ってもターゲットの色は変化しない. つまり, 被験者は自身の記憶と力感覚のみでカーソルをターゲット内に入れることになる. この PF タスクは, Marking Menus[5] のような, コンピュータ熟練者を対象とした eyes-free インタラクションをシミュレートする. さらに, FF と PF を比較することで Gripping におけるフィードバックの重要性の検証も行う.

離散値の段階数 n は 2~12 の 11 段階を用意した. 各段階のターゲットサイズは $800/n$ である. n が大きくなればなるほど微妙な力加減が必要になり, 被験者はより慎重な操作が要求される. 各被験者はトリガー操作ごとに, 11 段階の n の試行をそれぞれ 6 回ずつ行った. ターゲット選択に失敗した場合, 成功するまでその試行を繰り返した. 4 種類のトリガー操作を行う順番は被験者ごとに変え, カウンターバランスをとった. まず FF で全試行を行い, その後 PF で行った. PF は Gripping 熟練者を対象としているため, 初めに FF を行ってもらうことで Gripping の経験を少しでも積んでもらう狙いがある. まとめると, この実験では, 11 段階 $\times$ 6 試行 $\times$ 被験者 8 人 $\times$ 4 トリガー操作 $\times$ 2 フィードバック, 合計 4224 回の正解ターゲット選択が行われた.

5.4 パフォーマンスの測定

一般にコンピュータへの入力操作は素早く, 容易に, 正確に行えることが求められる. そこで本実験では, ターゲット選択のエラー率 (ER), ターゲットをクロスした回数 (NC), ターゲットの選択時間 (ST) の 3 つの観点から評価を行う. ER は 1 回のターゲット選択で発生するエラー率, NC はターゲットにカーソルが入った後にカーソルがターゲットの境界をクロスした数 (たとえば NC = 2 は, カーソルがターゲット内に入った後に一旦ターゲットを出て再度入ったことを示す), ST は被験者が力を入れ始めてからターゲット選択が完了するまでの時間である. ER は正確さ, NC は容易さ, ST は素早さを示す指標として利用する.

5.5 結果

NC と ST は正解ターゲット選択の試行から得たデータである. また, $\pm 2\sigma$ を超えた 24 個の計測値を外れ値としてデータセットから取り除いた.

ER の分析

エラー率 (ER) を図 2 に示す. FF のグラフを見ると, *Keeping* が他の手法と比較して明らかにエラー率が低いことがわかる. また, *Keeping* は n が増加してもエラー率がほぼ 0 で横ばいに推移する一方で, 他の 3 手法は n が増加するごとにエラー率が増加している. 4 つの手法について分散分析を行った結果, 有意差があることがわかった ($p < 0.001$). さらに, 4 手法のペア 6 組に対して t 検定を行った結果, *Keeping* とその他 3 手法全てのペアに有意差があることがわかった ($p < 0.001$).

一方, PF のグラフを見ると, *Keeping* が他の手法と比較してややエラー率が低いことがわかる. また, FF とは異なり, 全手法とも n が増加するごとにエラー率が増加している. FF と同様に分散分析を行った結果, 有意差は確認できなかった ($p = 0.050$).

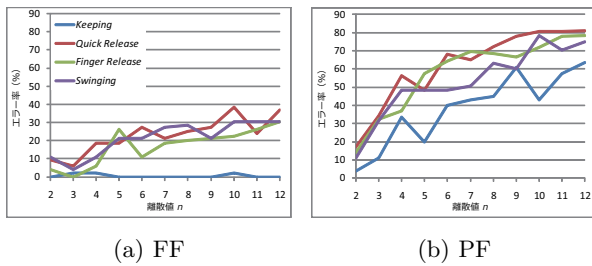


図 2. エラー率

NC の分析

クロス数 (NC) を図 3 に示す。FF のグラフを見ると、*Quick Release* のクロス数が他の手法よりもやや少ないことがわかる。また、4 手法とも n が増加するにつれてクロス数が指数関数的にする傾向が読み取れる。特に n が 6 を超えてから増加傾向が強い。4 つの手法について分散分析を行った結果、有意差は確認できなかった ($p = 0.313$)。

一方、PF のグラフを見ると、*Finger Release* のクロス数が他の手法よりもやや多いことがわかるが各手法には大差がない。また、 n が 10 以上のときは PF の方が FF よりもクロス数が少ない傾向がある。4 つの手法について分散分析を行った結果、FF と同様に有意差は確認できなかった ($p = 0.167$)。

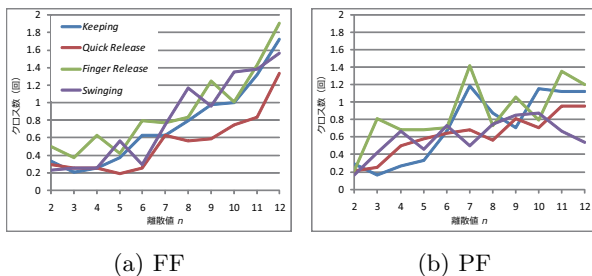


図 3. クロス数

ST の分析

選択時間 (ST) を図 4 に示す。FF のグラフを見ると、*Keeping* が他の手法よりもおよそ 1000ms 選択に時間を費やしていることがわかる。また、各手法とも n が増加するにつれて緩やかに選択時間も増加する傾向がある。4 つの手法について分散分析を行った結果、有意差があることがわかった ($p < 0.001$)。さらに、4 手法のペア 6 組に対して t 検定を行った結果、*Keeping* とその他 3 手法全てのペア、および *Quick Release* と *Finger Release* のペアに有意差があることがわかった ($p < 0.001$)。

PF のグラフを見ると、FF と同様に *Keeping* の選択時間は他の手法よりも長いことがわかる。一方、*Swinging* はほとんどの場合に最も早く、 n の値によらず安定した選択時間を示している。4 つの手法について分散分析を行った結果、有意差があること

がわかった ($p < 0.001$)。さらに、4 手法のペア 6 組に対して t 検定を行った結果、*Quick Release* と *Finger Release* のペアを除いた 5 つのペアに有意差があることがわかった ($p < 0.001$)。

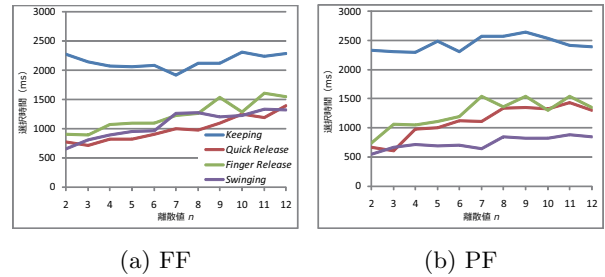


図 4. 選択時間

5.6 考察

トリガー操作の選定

FF のエラー率 (図 2(a)) での *Keeping* は n が増加してもエラー率がほぼ 0 である一方、その他の手法は n が増加するごとにエラー率も上昇した。よって、FF では *Keeping* が適切なトリガー操作であると思われる。選択時間 (図 4(a)) では、*Keeping* は他の手法よりも選択に時間がかかることがわかった。FF ではその差はおよそ 1000ms である。*Keeping* は握る力を 1000ms 間維持することで決定操作とする手法であるので、この差は適切な差であるといえる。しかしながら、1 回の操作時間が 2000ms とやや長い。この操作時間は維持する時間やブレの許容範囲を調整することで短縮できると考えている。

離散値の段階数 n の決定

まず、FF のクロス数 (図 3(a)) の結果から、 n が 6 を超えてからクロス数が大きく増加することがわかった。分散分析に有意差も見られなかったため、どの手法でも同様の傾向があると思われる。また、FF での *Keeping* のエラー率はほぼ 0 であるため、 n には依存しない。これらの結果を考察すると、適切なフィードバックがある状態では、*Keeping* で行う離散値入力として 6 段階が適切であるといえる。

視覚的フィードバック

PF は熟練者向けのタスクとして設計したが、今回の被験者には熟練者はいなかった。FF と PF のエラー率 (図 2) を比較すると、PF は FF の 2~3 倍程度のエラーが発生していることがわかる。熟練者であれば PF のエラー率が低くなるかどうかは確認できないが、ここでいえることは Gripping の初心者にとってフィードバックは非常に重要であるということである。利用者に対して適切な視覚的フィードバックを与えることで、Gripping は有用な入力操作になるといえる。

6 アプリケーション例

ここでは、Grippingによる離散値入力の具体的な応用例を2つ紹介する。Grippingはペン先がディスプレイから離れた状態で操作できるため、タップ操作に干渉しない独立した入力手法を提供できる。これはGrippingだからこそ実現可能なことであり、筆圧やパレルボタンでは実現できない。

Gripping ランチャー

Gripping ランチャーは6つの離散値に対して、アプリケーション固有の機能などを割り当て、キーボードのショートカットキーのような操作を実現するアプリケーションである。ペン入力インタフェースでメニューバーや小さなメニューアイコンなどのメニューインタフェースを操作することは容易ではない。Gripping ランチャーを用いることで、利用頻度の高いアプリケーションの起動や、アプリケーション内で利用頻度の高い機能の呼び出しなどに利用できる。

Gripping ソフトウェアキーボード

ペン入力インタフェースにとって文字入力は大きな課題の一つである。ソフトウェアキーボードでは、キーボードの大文字／小文字、かな／英字などを切り替えには物理的なキーボードと同様にシフトキーやCaps Lockキー、半角全角キーなどをタップする必要がある。Grippingの離散入力にそれらを割り当てることで、ボタンをタップすることによる切り替え操作なしに切り替えが可能になる。また、ストロークを用いた文字入力手法[7, 10]も提案されているが、それらにも切り替え操作が伴う。そのような切り替え操作に対してもGrippingを応用できる。

7 まとめ

ペン入力インタフェースの操作性向上を目指し、我々はペンを握る動作を利用したインタラクション手法、Grippingを提案した。Grippingの利用方法として連続的入力と離散的入力の2種類が考えられるが、ペン入力インタフェースは実用的な離散的入力を備えていない。そこで、我々はGrippingの離散的入力への応用に着目して研究を行っている。本論文では、人間が快適に区別して使い分けことができる握る力の段階数、および入力した離散値を選択するのに適したトリガー操作に関する実験について述べた。実験の結果、Gripping操作に適切なフィードバックがある場合には6段階の離散的入力が適切であることがわかった。また、トリガー操作には、握る力を一定時間、一定範囲内に維持する*Keeping*手法が適切であることもわかった。

今回の実験の結果、熟練度がエラー率に与える影響を調べる必要があることがわかった。今後はそれらの追加調査も行う。また、ペングリップを握る力

と筆圧との間には何からの関係があると思われる。今後は握る力と筆圧の関係を調査し、それら2つを組み合わせたインタラクション手法も開発したい。

参考文献

- [1] J. Accot and S. Zhai. More than dotting the i's - Foundations for crossing-based interfaces. In *Proc. of CHI'02*, pp. 73–80, 2002.
- [2] X. Bi, T. Moscovich, G. Ramos, R. Balakrishnan, and K. Hinckley. An Exploration of Pen Rolling for Pen-based Interaction. In *Proc. of UIST'08*, pp. 191–200, 2008.
- [3] K. Hinckley, P. Baudisch, G. Ramos, and F. Guimbretière. Design and Analysis of Delimiters for Selection-Action Pen Gesture Phrases in Scriboli. In *Proc. of CHI'05*, pp. 451–460, 2005.
- [4] D. Hopkins. The Design and Implementation of Pie Menus. In *Dr. Dobb's Journal*, Vol. 16, pp. 16–26, 1991.
- [5] G. Kurtenbach and W. Buxton. The Limits Of Expert Performance Using Hierarchic Marking Menus. In *Proc. of CHI'93*, pp. 482–487, 1993.
- [6] M. Miura and S. Kunifuji. RodDirect: Two-Dimensional Input with Stylus Knob. In *Proc. of MobileHCI'06*, pp. 113–120, 2006.
- [7] K. Perlin. Quikwriting: Continuous Stylus-based Text Entry. In *Proc. of UIST'98*, pp. 215–216, 1998.
- [8] G. Ramos, M. Boulos, and R. Balakrishnan. Pressure Widgets. In *Proc. of CHI'04*, pp. 487–494, 2004.
- [9] X. Ren, J. Yin, S. Zhao, and Y. Li. The Adaptive Hybrid Cursor: A Pressure-based Target Selection Technique for Pen-based User Interfaces. In *Proc. of INTERACT 2007*, LNCS4662, pp. 310–323, 2008.
- [10] D. Sato, B. Shizuki, M. Miura, and J. Tanaka. Menu-Selection-Based Japanese Input Method with Consonants for Pen-based Computers. In *Proc. of APCHI'04*, LNCS3101, pp. 399–408, 2004.
- [11] I. Siio and H. Tsujita. Mobile Interaction Using Paperweight Metaphor. In *Proc. of UIST'06*, pp. 111–114, 2006.
- [12] G. M. Smith and m. c. schraefel. The Radial Scroll Tool: Scrolling Support for Stylus- or Touch-Based Document Navigation. In *Proc. of UIST'04*, pp. 53–56, 2004.
- [13] Y. Suzuki, K. Misue, and J. Tanaka. Pen-based Interface Using Hand Motions in the Air. In *The IEICE Transactions on Information and Systems*, Vol. E91-D, pp. 2647–2654, 2008.
- [14] F. Tian, L. Xu, H. Wang, X. Zhang, Y. Liu, V. Setlur, and G. Dai. Tilt Menu: Using the 3D Orientation Information of Pen Devices to Extend the Selection Capability of Pen-based User Interfaces. In *Proc. of CHI'08*, pp. 1371–1380, 2008.