

Birdlime Icon: 動的にターゲットを変形するポインティング支援手法の提案

The Birdlime icon: Facilitating Pointing by Dynamically Stretching Target Shape

築谷 喬之 高嶋 和毅 朝日 元生 伊藤 雄一 北村 喜文 岸野 文郎*

Summary. 本稿では、ターゲット上を行き過ぎようとするカーソルを捕らえるようにターゲットを変形することで、ポインティングを支援する手法「Birdlime icon」を提案する。この手法は、カーソルを減速することなくユーザへ粘性感覚を与えるようにターゲットを変形し、カーソルがターゲット以外のオブジェクト上を通過しなければならない状況でも、その影響を受けずにポインティング時間を短縮できる。この提案手法の有効性を評価するために、ターゲット上でカーソル速度を遅くする支援手法である Sticky と、二つの比較実験を行った。実験 1 では一つのターゲットを配置した環境で実験を行い、Sticky と同程度にポインティング時間を短縮できることを確認した。実験 2 では、ターゲットの周囲に妨害刺激を配置した環境で実験を行い、Sticky とは異なり妨害刺激があっても提案手法はパフォーマンスが落ちることがなく有効であるという知見を得た。

1 はじめに

マウスによるポインティングは GUI 環境上で最も基本的な動作であるが、長時間作業による回数の増加やディスプレイの高解像度化によって、ポインティング中の運動負荷は増加する傾向にある。さらに、通常 GUI 環境には操作対象となるオブジェクトが多数配置されているため、それを考慮した上で複数のオブジェクトの中から効率よくターゲットを選択できる手法が求められている。

これまでにもポインティング中の運動負荷を軽減するための手法は様々な提案されているが [2]、画面上に複数のオブジェクトが配置された環境では十分な効果が得られないものが多い。また、複数オブジェクトの環境でもある程度効果が得られていても、ユーザの意図しない動作によって認知的負荷を増大させてしまう手法もある。

そこで我々は、複数オブジェクトによる影響を受けにくく、認知的負荷も少ない手法である「Birdlime icon」を提案する。この手法では、ターゲット上を行き過ぎようとするカーソルを捕らえるようにターゲットを変形し、ユーザに粘性感覚を与えることでポインティングを支援する。

本稿では、まずこの提案手法である Birdlime icon の設計について述べる。また、複数オブジェクトの影響を明らかにするために、Birdlime icon によるターゲット拡大とカーソル減速によるマウス操作領域だけのターゲット拡大との違いに着目し、二つの環境で行った評価実験の結果について報告する。

Copyright is held by the author(s).

* Takayuki Tsukitani, Kazuki Takashima, Yuichi Itoh, Yoshifumi Kitamura and Fumio Kishino, 大阪大学 大学院情報科学研究科, Motoo Asahi, 株式会社 日本総合研究所

2 関連研究

2.1 ポインティング基礎検討

ポインティング動作は、最もカーソル速度が速くなるピーク速度が現れる時点を境界に、前半を計画時間、後半を調整時間として二つに分けられるというモデルが提案されている [6]。このモデルでは、計画時間中の特徴量はターゲット距離のみに依存し、調整運動中はターゲット幅に依存する運動となることが示されており、運動負荷の半分以上が調整時間で引き起こされているという結果も得られている。ポインティング時のエラーに関しては、カーソルがターゲットを行き過ぎる動作であるオーバシュートが多いことが報告されている [11]。

2.2 ポインティング支援手法

ポインティング時間の短縮は、運動負荷の軽減や作業効率の向上に繋がる。ポインティングの評価指標であるフィッツの法則によると、ターゲット距離が短いほど、またはターゲットサイズが大きいほど、ポインティング時間は短くなり、これを基に様々な支援技術が提案されている [2]。

2.2.1 ターゲット距離を短くする手法

Asano らはピーク速度とターゲット距離の関係を利用してターゲットを予測し、カーソルを移動する Delphian Desktop を提案している [1]。しかし、ターゲットが多いときの予測精度は明らかにされておらず、予測の誤りが増大する可能性がある。Kobayashi らの提案する Ninja cursor は、画面上に複数配置されたカーソルを同時に操作できる手法である [8]。この手法には、画面上のオブジェクト密度が高いとパフォーマンスが悪くなり、選択状態にあるカーソルが分かりにくくなるという問題がある。

2.2.2 ターゲットサイズを大きくする手法

ターゲットサイズを大きくする手法としては、ターゲットそのもののサイズを拡大する手法、カーソルを減速することでマウス操作領域でのサイズを大きくする手法、カーソルサイズを大きくして相対的なターゲットサイズを大きくする手法の三つのアプローチが考えられる。

一つ目のアプローチは数多くの検討がなされており、カーソルがターゲット距離の 90% に達した後ターゲットを拡大しても効果があることや [10], ユーザがそのターゲットが拡大することを事前に知らなくても効果があることが知られている [12]. しかし、これらの検討では、ターゲットサイズが単純に拡大するだけであり、より効果的なターゲット拡大や変形手法の検討が必要であると言える。

二つ目のアプローチとしては、Sticky icons [5, 9] や Semantic pointing [3] という手法がある。また、Mandryk らはユーザに知覚できないほどのカーソルの減速であっても、ポインティング時間の短縮に効果があることを示している [9]. しかし、これらの手法では、画面上にオブジェクトが複数ある場合に、不必要なカーソルの減速が起こり十分な効果が得られないという問題がある。

三つ目のアプローチは、エリアカーソルと呼ばれているが [4, 7], 複数のオブジェクトが近接するとき、ターゲットを一意に決定できないという問題がある。これを解決するために、常に一つのオブジェクトを選択するようにカーソルサイズを動的に変更する Bubble cursor [7] や、カーソル速度によってカーソルサイズを変化させる DynaSpot [4] などが提案されている。しかし、いずれの手法においても、意図しないカーソルサイズの変化が認知的負荷を増大させるという問題が考えられる。

3 提案手法

3.1 概要

関連研究で述べたように、ターゲット距離を短くする手法は、予測の誤りによるリスクが大きく、カーソルを見失う危険性も伴う。今回は、できるだけリスクや認知的負荷が小さい手法で、ポインティング時間を短縮しエラーを少なくすることを目的とし、ターゲットサイズを大きくする手法について検討する。この手法は、ポインティング時の負荷の半分以上がポインティング後半で起こるといった点からも [6], 有効であると期待できる。

ターゲットサイズを大きくする三つのアプローチの中では、Sticky icons に代表されるカーソル速度を遅くする手法は、複数オブジェクトの環境では効果が大きく落ちるといった問題が指摘されているものの、単一ターゲットでは非常に有効である。不必要な視覚フィードバックがなく、突入してくるカーソルに対してのみ寛容であるという点は、ポインティ

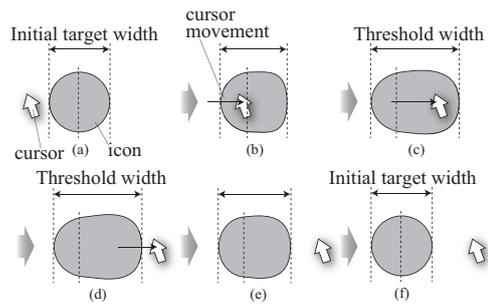


図 1. Birdlime icon の様子

ング時にはオーバーシュートによるエラーが多いことから考えて、非常に理にかなっていない。そこで本研究では、Sticky icons のように少ない視覚フィードバックでオーバーシュートに対して寛容な手法を目指す。ただし、Sticky icons のようにカーソルを減速するのではなく、粘性感覚を与えるようにターゲットを変形することで同様の効果を与え、複数オブジェクトの環境でも効果が落ちにくいようにする。

3.2 設計

まず、カーソルがターゲット内に突入するまではターゲットの形状は変化しない (図 1(a)). カーソルがターゲット内に突入した瞬間からターゲットの変形がはじまり (図 1(b)), オーバシュートしようとするカーソルを捕らえるようにさらに変形する (図 1(c)). ターゲットの変形する大きさには閾値を設けており、それ以上は変形しない (図 1(d)). カーソルがターゲットから脱出するか、ターゲット内で一定時間停止していると、ターゲットの形状は自動的に元に戻る (図 1(e) および (f)). これらの、ターゲットの変形はスムーズに行われ、カーソルがターゲットの中心に達するまでは変形速度は小さく、中心を越えると大きくなる。なお、本稿の実験においては、これらの変形に関するパラメータは予備実験での結果を基に決定している。

4 実験 1: 単一ターゲット環境での実験

4.1 実験概要

提案手法である Birdlime icon によるターゲットの変形は、カーソルがターゲットに突入した瞬間に行われる。そのため、ターゲットサイズの変化する大きさとタイミングは、マウスの操作領域で見ると Sticky icons のようなカーソル速度を遅くする手法と同等となる。そこで実験 1 では、提案手法のような見た目の拡大も伴うターゲット拡大と、Sticky icons のような操作領域のみでのターゲット拡大の違いを明らかにし、提案手法の有効性を評価するための比較実験を行う。さらに、ターゲットサイズの拡大と操作領域での拡大を同時に適用したときに得られる効果も調査するために、両手法を組み合わせた手法についても比較対象とする。

4.2 実験環境

実験プログラムは Microsoft 社の Visual C++ 2005 で実装し、計算機は DELL 社の Inspiron 530 (CPU: Core 2 Duo, 3 GHz, メモリ: RAM 2GB, OS: Windows XP Professional), ディスプレイは DELL 社の 30 インチディスプレイを 2,560×1,600 ピクセル (40 dpi) の解像度で使用する。入力デバイスは Logitech 社の Wheel Mouse (M-BJ58) を用いて 60 Hz でマウスカーソル位置を取得し、C-D 比は約 0.23 (Control : Display = 1 : 4.3) とする。マウスパッドは Power Support 社の Airpad Pro III (AK-07, 408×306 mm) を使用する。

4.3 実験参加者

実験参加者は男性 8 名と女性 4 名の合計 12 名で、平均年齢は 23.1 歳の情報系の大学生または大学院生である。全員日頃からマウスを用いた計算機操作に慣れており、右利きである。実験においても全員右手でマウスを操作する。

4.4 実験計画

今回は、提案手法 (Birdlime), カーソル速度を遅くする手法 (Sticky), Birdlime と Sticky を同時に適用した手法 (Combination) および通常の動作 (Normal) の 4 手法を比較する。このうち Normal 以外の 3 手法に対して拡大率を定義する。拡大率は、Birdlime においては最大でどれだけターゲットを変形するかを表し、Sticky においてはカーソル速度を減速する度合いを表す。例えば、拡大率 2.0 のときは、Birdlime は最大でターゲットの幅が 2 倍となり、Sticky においてはカーソル速度が 2 分の 1 となる。すなわち、Birdlime と Sticky は拡大率が同じ場合、マウス操作領域では同じようにターゲットサイズが拡大されることになる。Combination では、ターゲット拡大とともにカーソル速度も遅くなるため、Birdlime や Sticky と比べてさらに 2 倍ターゲットサイズが拡大されることになる。本実験では、拡大率は Mandryk らの研究に基づいて、実用的に使える範囲を考慮して設定する。

実験はターゲット距離 (4, 16, 32, 48 cm), ターゲットサイズ (0.5, 1.0, 2.0 cm), 手法 (Birdlime, Sticky, Combination, Normal), 拡大率 (1.6, 2.0, 4.0) の 4 要因反復実験である。これらが、ポインティング時間と、全タスク中でオーバシュートが起こった回数の割合を表すオーバシュート率に与える影響を調査する。

4.5 実験手順

参加者は楽な姿勢になるように椅子の高さを調整して座り、操作しやすい位置にマウスパッドを設置する。このとき参加者の視点とディスプレイの距離は約 60 cm である。実験はまずカーソルとターゲットが画面に表示され、参加者がマウスボタンを

クリックすることでタスクが開始される。参加者はターゲットまでできるだけ早くかつ正確にカーソルを動かしてクリックする。

ターゲット距離とサイズはランダムな順番で提示され、これを 10 回反復する。したがって、 $4 \times 3 \times 10 = 120$ タスクが各手法および拡大率ごとに試行され、これを 1 ブロックとする。手法と拡大率は 1 ブロックの間は同じであり、学習効果を減らすように、参加者間で提示する順番を変える。したがって、1 参加者あたり、3 手法 (Birdlime, Sticky, Combination) \times 3 拡大率 \times 120 タスク (1 ブロック) + 1 手法 (Normal) \times 120 タスク (1 ブロック) = 1,200 タスクを行う。実験中は 1 ブロックごとに 5 分の休憩を挟むとともに、慣れの影響を減らすために全体の半分である 5 ブロック完了時には 1 時間の休憩を挟む。また、実験後には各手法の主観評価についてアンケートを行う。

4.6 実験結果

参加者 12 名 \times 1,200 タスク = 14,400 試行のうち、169 試行 (1.174%) でエラーが観察されたが、大半が単純なクリックミスによるものであり、以後の分析ではこのエラーであったタスクも含めて分析する。各測定項目について、4 要因分散分析で解析し、多重比較には Tukey's HSD を用いた。

図 2 に各実験条件におけるポインティング時間を示す。ポインティング時間に関して、ターゲット距離 ($F(3, 33) = 388, p < .001$) とターゲットサイズ ($F(2, 22) = 563, p < .001$) による主効果が見られた。手法 ($F(3, 33) = 7.48, p < .01$) による主効果も見られ、Combination が最も早く ($p < .01$), Sticky と Birdlime の間に有意差は見られなかった ($p > .05$)。拡大率による主効果は見られず ($F(2, 22) = .269, p > .05$)、ポインティング時間に影響を与えにくいことが分かった。また、ターゲット距離とターゲットサイズによる交互作用が認められ ($F(6, 66) = 1.14, p < .05$)、ターゲット距離が 4 cm と 16cm のときに Normal がその他の手法に比べて遅かった ($p < .01$)。さらに、ターゲットサイズと手法の間にも交互作用が見られ ($F(6, 66) = 2.96, p < .05$)、ターゲットサイズが 0.5 cm のときに、Normal が Sticky および Combination よりも遅いことが分かった ($p < .01$)。

オーバシュート率に関しては、ターゲット距離 ($F(3, 33) = 13.80, p < .001$) とターゲットサイズ ($F(2, 22) = 43.88, p < .001$) については主効果が見られたが、手法 ($F(3, 33) = .007, p > .05$) と拡大率 ($F(2, 22) = 1.218, p > .05$) には主効果が認められず、手法間や拡大率の違いはオーバシュート率には影響を与えにくいことが分かった。

実験後に 4 段階 (4:良い~1:悪い) で参加者の手法ごとの主観評価を調査した。その結果、評価が高い順に、Combination (3.67 ポイント), Sticky (2.83

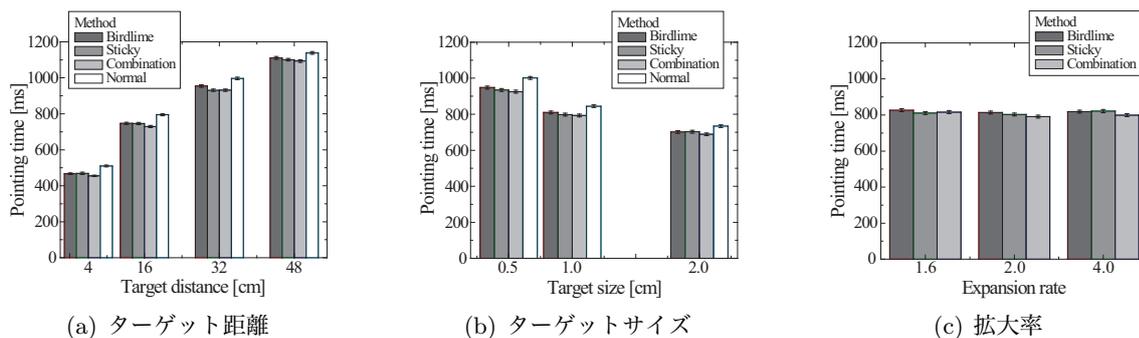


図 2. 各実験条件におけるポインティング時間の平均と標準誤差

ポイント), Birdlime (2.42 ポイント), Normal (1.08 ポイント) となった. また, 各手法使用時にターゲットから粘性感覚が得られたかを調査したところ, Sticky と Combination では全ての, Birdlime でも 12 人中 4 人の参加者が粘性を感じる と答えた.

5 実験 2: 複数オブジェクト環境での実験

5.1 実験概要

実験 1 から単一ターゲット環境においては, Combination が最もポインティング時間を短縮でき, Birdlime は Sticky と同様にポインティング時間の短縮に有効であることが分かった. 実験 2 では実際の GUI 環境を想定し, カーソルとターゲットの間にターゲット以外のオブジェクト (妨害刺激) がある条件を設定する. この条件で, Sticky や Combination といった操作領域を拡大する手法と比較し, Birdlime の複数オブジェクト環境における有効性を調査する. なお, 実験環境は実験 1 と同様である.

5.2 実験参加者

実験参加者は男性 12 名と女性 6 名の合計 18 名で, 平均年齢は 24.9 歳の情報系の大学生または大学院生である. このうち 8 名は実験 1 にも参加している. 実験 1 と同様に, 全員右手でマウスを操作する.

5.3 実験計画

この実験では妨害刺激の影響を調査するが, 関連研究では妨害刺激の密度を要因とする実験がよく行われている [4, 7, 8]. しかし, 今回比較する各手法は妨害刺激上を通過するかどうかの影響が大きく, 参加者の操作によって通過する妨害刺激の数が増える密度というパラメータは適切ではない. そこで, 通過する妨害刺激の数を正確にコントロールするために, 図 3 のように妨害刺激をターゲットを囲むように配置し, カーソルが必ず妨害刺激を通過するようにする. 妨害刺激は全てターゲットと同じように振る舞い, ループの間隔は実際のデスクトップ環境を想定しターゲット幅と同じに設定する.

実験は, ターゲット距離 (16, 32 cm), ターゲットサイズ (0.5, 1.0, 2.0 cm), 手法 (Birdlime,

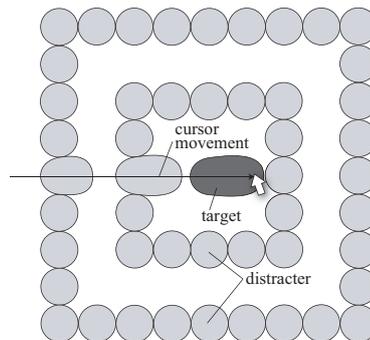


図 3. 妨害刺激の配置 (2 周のとき)

Sticky, Combination, Normal) と, 妨害刺激の配置 (1, 2 周) の 4 要因反復実験である. 実験 1 でポインティングに影響を及ぼさなかった拡大率は, あまり現実的でない 4.0 を除き比較的パフォーマンスの良かった 2.0 を使用する.

5.4 実験手順

実験タスクは実験 1 のものとほぼ同様であり, 10 回反復するため各手法ごとに $2 \times 3 \times 2 \times 10 = 120$ 回の試行が行われる. 手法は 120 試行が終わるまで変更しないが, 手法の提示順は参加者間で学習効果の影響を少なくするためにバランスをとり, 手法が切り替わるごとに参加者は 3 分程度休憩する. したがって, 1 参加者あたり, $4 \text{ 手法} \times 120 = 480$ 試行を行い, 実験後には主観評価を調査する.

5.5 実験結果

参加者 18 名 \times 480 タスク = 8,640 試行のうち, 287 試行 (3.322 %) でエラーが観察されたが, 以降ではこれらも含めて実験 1 と同様に分析する.

図 4 に各実験条件でのポインティング時間を示す. ポインティング時間に関して, ターゲット距離 ($F(1, 17) = 436, p < .001$) とターゲットサイズ ($F(2, 34) = 473.2, p < .001$), 手法 ($F(3, 51) = 32.9, p < .001$), 妨害刺激の配置 ($F(1, 17) = 36.6, p < .001$) による主効果が見られ, Combination がその他の手法に比べて遅かった ($p < .01$). また, ターゲット距離と手法 ($F(3, 51) = 5, p < .01$), ターゲットサイズと手法 ($F(6, 102) = 11.7, p < .001$), 手法と妨害刺激の配置 ($F(3, 51) = 9.38, p < .001$) の間にそれぞれ交互作用が見られ,

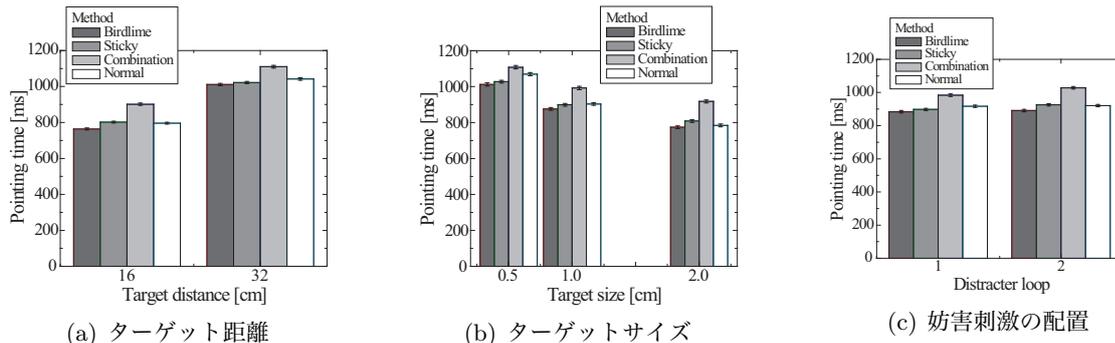


図 4. 各実験条件におけるポインティング時間の平均と標準誤差

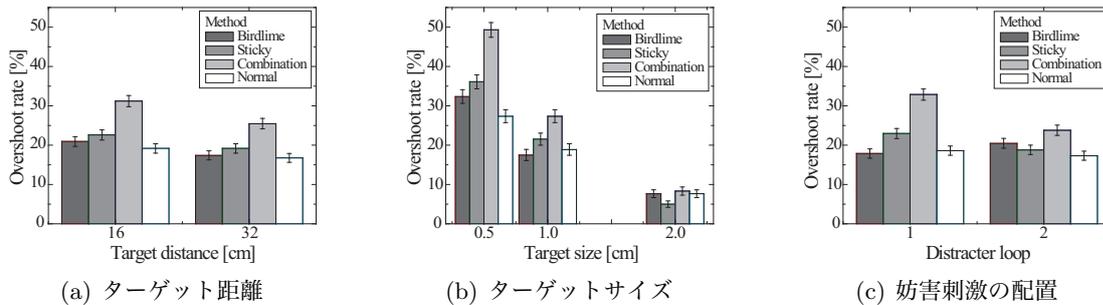


図 5. 各実験条件におけるオーバシュート率の平均と標準誤差

ターゲット距離が 16 cm のとき ($p < .01$) とターゲットサイズが 2.0 cm のとき ($p < .05$), 妨害刺激の配置が 2 周のとき ($p < .05$) に, Sticky よりも Birdlime の方がポインティング時間が短かった。

図 5 に各実験条件におけるオーバシュート率を示す。オーバシュート率に関して, ターゲット距離 ($F(1, 17) = 8.02, p < .05$), ターゲットサイズ ($F(1, 17) = 155, p < .01$), 手法 ($F(3, 51) = 7.84, p < .01$) および妨害刺激の配置 ($F(1, 17) = 29.97, p < .01$) による主効果が見られ, Combination が最もオーバシュートが多いことが分かった ($p < .05$)。また, ターゲットサイズと手法の間に交互作用が見られ ($F(6, 102) = 6.45, p < .01$), ターゲットサイズが 0.5 cm のときに Combination が他の手法よりオーバシュートが多いことが分かった ($p < .05$)。Birdlime と Sticky の間には有意差はなかったものの ($p > .05$), 全体として Sticky の方がオーバシュート率が高くなる傾向が見られた。

実験後のアンケートでは, 最も好まれた順番に, Birdlime (3.44 ポイント), Normal (2.61 ポイント), Sticky (2.44 ポイント), Combination (1.50 ポイント) となった。

6 考察

実験 1 では, 提案手法である Birdlime は Sticky と同程度にポインティング時間を短縮でき, Birdlime によるターゲット拡大は Sticky による操作領域の拡大と同等の効果を持つことが分かった。一方で, 実験 2 においては, ターゲット距離が短いとき (16 cm) とターゲットサイズが大きいとき (2.0 cm) ときに, Birdlime は Sticky よりも良いパフォーマンス

となった。ターゲット距離に関しては, 短いターゲット距離のときはカーソル速度が遅く, Sticky によるカーソル減速による影響が大きくなったためだと考えられる。ターゲットサイズに関しては, このサイズは一般的なデスクトップ環境でのアイコンの大きさに相当するため, 実用的な環境での Birdlime の有効性は示せたと言える。

実験 1 においては, 手法間にオーバシュート率に差はなかったが, 実験 2 においては手法間に差が現れ, Birdlime に比べて Sticky や Combination のオーバシュート率が高くなった。また, 実験 2 では, Sticky や Combination の主観評価の点数が低く, 実験中には参加者がカーソルをうまく制御できていない様子が見られた。この原因として, 妨害刺激による意図しないカーソルの減速が参加者の運動計画を狂わせ, 不自然な運動を強いていたことが考えられる。特に, Combination は Sticky の極端な例となっており, カーソルの減速が運動計画に強い影響を与えている様子が観察された。

Birdlime のターゲット変形は, ターゲットを獲得しやすくする利点を持つが, 複数オブジェクト環境では妨害刺激の拡大が見た目の煩わしさを与え, ポインティングに悪影響を与える可能性がある。しかし, 今回の実験では, 参加者からそのような感想は聞かれず, ターゲットを獲得しやすくなる利点に比べて, 妨害刺激の見た目の煩わしさによる影響は微々たるものであったと言える。それに対して, Sticky や Combination のような操作領域での拡大では, 妨害刺激による不必要なカーソルの減速が大きな影響を持ち, 場合によっては Normal よりもパフォーマンスが悪くなることが分かった。

今回の実験では拡大率がパフォーマンスの改善に大きな影響を持たなかったため、参加者は Birdlime icon の拡大領域を十分に活かせていなかったと考えられる。Cockburn らは、ターゲット拡大はクリックできる領域を大きくする効果もあるが、拡大自体がターゲット領域内にカーソルがあるという視覚的な手がかりとなり、ユーザの反応時間を短縮する効果が大きいことを示している [5]。Birdlime icon でもこの反応時間の短縮がある程度の役割を果たしている可能性があるため、今後は見た目のターゲット拡大とマウス操作領域でのターゲット拡大それぞれの影響を調査する必要がある。

Birdlime によるターゲット拡大はカーソル速度は変化させないが、何人かの参加者はターゲットに捕まえられるような粘性を感じると答えた。このことから、カーソル変形のさせ方によっては、カーソル速度を変化させずに粘性感覚を与えられる可能性が示唆された。しかし、今回の変形方法では十分に粘性を感じられなかった参加者も多く、Birdlime はポインティング時間を短縮できるものの Normal との差が顕著でなかった原因の一つとも考えられる。ターゲットを変形する形状や速度によっては、さらに効果的に粘性感覚を与えられると予想されるため、今後はユーザが粘性を感じやすい条件や、粘性感覚がポインティングに与える影響について検討していく予定である。また今回、Combination はカーソルの減速による影響が大きすぎて有効に働かなかったが、Mandryk らによるユーザが感じないほどのカーソル速度変化でも効果があるという結果もあるため [9]、ターゲット拡大とカーソル速度変化の有効な組み合わせ方についても検討を進めていきたい。

今回はターゲットサイズを大きくする手法に注目したが、これらの中で既存のターゲット拡大やエリアカーソルには、オブジェクト同士が近いときに領域選択動作のようなオブジェクト間の空白を利用する操作が困難になるという問題も指摘できる。これに対して Birdlime icon では、ターゲット領域にカーソルが突入してから変形するため、空白を利用した動作を阻害することがない。ただし、オブジェクト同士が近接している場合には、提案手法によるターゲット変形では変形領域が他のオブジェクトに重なる可能性がある。そこで、他のオブジェクトに重ならないように変形の範囲を限定することを考えている。これによって、少なくとも通常よりもポインティングパフォーマンスは悪くならず、可能な範囲でターゲット変形による恩恵を受けられる。これらから、単純なポインティング時間だけの評価では Birdlime icon よりもターゲット拡大やエリアカーソルの方が良い結果となる可能性もあるが、オブジェクト再配置などの実際の環境を想定した評価では Birdlime icon は他の手法よりも有効であると考えられ、このような観点からも改善し評価していく予定である。

7 おわりに

本稿では、Birdlime icon を提案し、その評価実験の結果を報告した。Birdlime icon は、オーバシュートしようとするカーソルを捕えるようにターゲットを変形することで、ポインティング動作を支援する。この手法は評価実験において、カーソルの動く途中に妨害刺激が存在していたとしても、Sticky icons などのカーソル速度を制御する手法と違い、パフォーマンスが落ちることがなく有効であることを示した。さらに、妨害刺激や拡大率の影響について議論した。今後は、オブジェクト同士が隣り合うなど、より複雑な状況においても適用可能なターゲットの変形手法を検討していく予定である。

謝辞

本研究の一部は、文部科学省グローバル COE プログラム (研究拠点形成費) の補助によるものである。

参考文献

- [1] T. Asano, et al. Predictive interaction using the Delphian Desktop. In *Proc. of UIST '05*, pp. 133-141, 2005.
- [2] R. Balakrishnan. "Beating" Fitts' law: virtual enhancements for pointing facilitation. *Int'l J. of Human-Computer Studies*, 61(6):857-874, 2004.
- [3] R. Blanch, Y. Guiard, and M. Beaudouin-Lafon. Semantic pointing: improving target acquisition with control-display ratio adaptation. In *Proc. of CHI '04*, pp. 519-526, 2004.
- [4] O. Chapuis, J.-B. Labrune, and E. Pietriga. DynaSpot: speed-dependent area cursor. In *Proc. of CHI '09*, pp. 1391-1400, 2009.
- [5] A. Cockburn and P. Brock. Human on-line response to visual and motor target expansion. In *Proc. of GI '06*, pp. 81-87, 2006.
- [6] D. Elliot, R. Chua, and W. F. Helsen. A century later: Woodworth's (1899) two-component model of goal-directed aiming. *Psychological Bulletin*, 127(3):342-357, 2001.
- [7] T. Grossman and R. Balakrishnan. The bubble cursor: enhancing target acquisition by dynamic resizing of the cursor's activation area. In *Proc. of CHI '05*, pp. 281-290, 2005.
- [8] M. Kobayashi and T. Igarashi. Ninja cursors: using multiple cursors to assist target acquisition on large screens. In *Proc. of CHI '08*, pp. 949-958, 2008.
- [9] R. L. Mandryk and C. Gutwin. Perceptibility and utility of sticky targets. In *Proc. of GI '08*, pp. 65-72, 2008.
- [10] M. J. McGuffin and R. Balakrishnan. Fitts' law and expanding targets: experimental studies and designs for user interfaces. *ACM Trans. Comput.-Hum. Interact.*, 12(4):388-422, 2005.
- [11] J. G. Phillips and T. J. Triggs. Characteristics of cursor trajectories controlled by the computer mouse. *Ergonomics*, 44(5):527-536, 2001.
- [12] S. Zhai, et al. Human on-line response to target expansion. In *Proc. of CHI '03*, pp. 177-184, 2003.