

ポインティングタスク中のピーク速度を用いた ターゲット予測インタフェースの提案

高嶋 和毅[†] 浅野 岳史[†], エフッド シャーリン^{††}
北村 喜文[†] 岸野 文郎[†]

GUI環境において増加する傾向にあるポインティング負荷を軽減することを目的とし、ポインティング中の運動特性に基づいたターゲット予測インタフェース「Delphian Desktop」を提案する。ユーザのポインティング動作中のピーク速度はターゲット距離に依存するという特性があり、この特性は回帰式の形で表すことができる。提案するインタフェースでは、ポインティング動作の前半に検出されたピーク速度を得られた回帰式に適用することでタスク完了前にターゲット距離を予測し、カーソルを予測地点まで自動的にジャンプさせる。評価実験によって、提案インタフェースが800 pixel以上の長いターゲット距離においてポインティング時間の短縮に有効であり、マウス移動距離の軽減にも有効であることを確認した。

Proposing a Target Prediction Interface Using a Peak Velocity during Pointing Tasks

KAZUKI TAKASHIMA,[†] TAKESHI ASANO,[†] EHUD SHARLIN,^{††}
YOSHIFUMI KITAMURA[†] and FUMIO KISHINO[†]

We propose the Delphian Desktop which enables to predict spatial user intentions to reduce the time required to point to faraway targets in GUIs. Predictions are calculated by tracking cursor movement and estimating the user goal location using inexpensive online algorithm based on the linear peak cursor velocity-distance to target relationship. Once the system detects the peak velocity during pointing movement, the automatic cursor jump toward predicted point is performed. In testing the Delphian Desktop effectively shortened pointing time for 800 pixel or longer target distance and reduced the physical mouse movement.

1. はじめに

ディスプレイが大画面高解像度化するのにもとない、グラフィカルユーザインタフェース (GUI) 環境において、ポインティングタスクに必要な時間が増加する傾向にある。この傾向は今後も続く予想されるため、ポインティング時間を短縮し、利用者の作業労力の軽減が求められている。

これまででも、ターゲットの拡大²⁶⁾ やカーソルをターゲットへ向けてジャンプさせる¹⁰⁾ などのポインティング時間の短縮を狙った様々なポインティング支援技術

が提案されてきた²⁾。しかし、これらの多くは、カーソルの位置とターゲットの位置の静的な関係のみを使用してきたため、ユーザの意向を反映できない場合や、ポインティング時の環境によってはその効果が薄くなりがちであった。

そこで本研究では、カーソルの位置だけでなく動作中の人の運動特性に着目し、ポインティング過程の情報をターゲットの予測に活用することを試みる。本論文では、ポインティング中のピーク速度からターゲット距離を予測する手法を提案し、提案手法を用いた予測型インタフェース「Delphian Desktop」の概要と評価結果を報告する。

2. 関連研究

2.1 ポインティングタスクにおける運動特性

GUI環境におけるポインティング動作は古くから広く研究されており、ポインティングに要する時間はターゲットの幅が大きくなるほど減少し、ターゲット

[†] 大阪大学大学院情報科学研究科
Graduate School of Information Science and Technology,
Osaka University

^{††} カルガリー大学コンピュータサイエンス学部
Department of Computer Science, University of
Calgary
現在、野村総合研究所
Presently with Nomura Research Institute

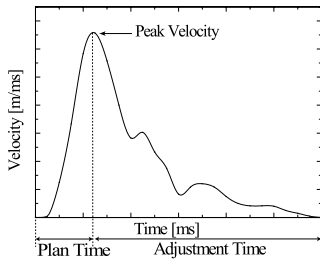


図1 ポインティング中の速度波形

Fig. 1 Velocity profile during pointing task.

までの距離が長くなるほど増加するというフィッツの法則などがよく知られている^{1),7)}。また、マウスポインティングや腕のリーチング中の速度波形は、図1に示すようなベル型波形を描くことが定説であり^{11),22)}、人の運動計画の解明を目的にした分野で広く調査され続けている。Woodworthは、ポインティング動作は、計画時間と調整時間の2つに分けられるというモデルを提案し²⁴⁾、このモデルはそれ以降行われた多くの研究に支持されている^{6),18),22),23)}。計画時間では、視覚フィードバック情報を利用しないフィードフォワード制御が行われ、調整時間では、視覚フィードバックループによって、手の速度を調整しながら目標地点まで正確に動かそうとする制御が行われる。速度波形中におけるこの2つの運動の境界については多くの議論があるが^{6),19)}、MacKenzieらはこの境界を速度がピークに達する地点と仮定してポインティング実験を行い、ピーク速度以前の運動を特徴づける特徴量はターゲット距離にのみ依存すること、およびピーク速度以後は、ターゲット幅に依存する調整時間であり、その間に視覚フィードバック制御が行われることを示している¹⁵⁾。Heathらの研究では、速度波形中のピーク速度以前の特徴量は、ポインティング運動を始める前に計画されていることを明らかにし、ピーク速度が運動計画の情報を持つ非常に重要な特徴量であると考察されている^{6),12)}。さらに、これらの研究を基に、速度波形中のピーク速度とそのタスクにおけるターゲット距離までの関係を調査した研究もあり、ピーク速度とターゲットまでの距離には相関があることが報告されている^{20),21)}。

2.2 ポインティング支援技術

フィッツの法則を指標としてポインティング時間を短縮するためには、ターゲット距離の短縮、またはターゲット幅の増加、およびその両方を実現するといった3通りの方法が考えられ、それぞれの考えに基づいて様々なポインティング支援手法が提案されている²⁾。

2.2.1 ターゲット距離の短縮

Baudischらは、Drag-and-drop操作におけるターゲット距離を短縮するためにDrag-and-popと呼ばれる手法を提案している³⁾。この手法は、システムがユーザの初期のカーソル移動方向と把持しているオブジェクトの属性からターゲットの候補となりうるフォルダなどを予測し、すべてのターゲット候補をカーソル(ドラッグの対象オブジェクト)に向けて近づけることでターゲット距離を短縮している。また、この手法の改良として、Collombらは、ターゲット候補をカーソル付近に作成したミニチュアデスクトップ内に表示することでターゲット距離を短縮するPush-and-popを提案している⁵⁾。これらの手法は、Drag-and-dropのみに適用されるものであり、大画面でターゲット距離が長い場合に効果を発揮するが、逆にターゲット距離が短い場合や、ターゲット候補が多すぎる場合は、ユーザの動作を妨害したり、ユーザを惑わせたりする恐れもある。これらとは逆に、カーソルをターゲットへ向けて移動させることで、ターゲットまでの距離を短縮する手法も提案されている。Guiardらは、スタートアイコンからのカーソルの脱出方向の延長線上にあるオブジェクトをターゲットとし、瞬間的にカーソルをターゲット上に移動させるObject pointingを提案している¹⁰⁾。この手法では、ポインティング時間は短縮されるものの、瞬間的にジャンプするために直感性に欠け、ターゲット周辺に他のオブジェクトが密に存在する場合は、エラーが増加する可能性が高いと指摘されている。

2.2.2 ターゲット幅の増加

視覚的にターゲット幅を増加させることによってポインティング時間を短縮しようとする手法がZhaiらによって提案されている^{17),26)}。これは、カーソルがターゲットに近づいたときにターゲットサイズを拡大する手法で、ポインティング終盤でターゲットを拡大したとしてもポインティング時間の短縮に有効であることが示されている。しかし、カーソルがターゲットに近づくまでの動きに対しては効果がないため、ターゲット距離が長い場合は、短縮できるポインティング時間の割合が小さくなってしまふ。これとは逆に、カーソルの幅を増加させ、カーソルがターゲットにアクセスできる領域を増やすことで、マウス移動スペースにおけるターゲット幅を増加させる試みもなされている¹³⁾。この手法はエリアカーソルと呼ばれ、古くからポインティング支援の代表的な手法であったが、その効果はターゲット群の密度とエリアカーソルの大きさの関係に大きく依存することから、有効な場面が限ら

れるといわれてきた²⁵⁾。最近では、その問題を改良する研究もさかんで、複数のオブジェクトをエリアカーソルでとらえ、エリアカーソル内にとらえたオブジェクトの内容を表示させる Zoom selector などが提案されている⁹⁾。また、カーソル周囲のオブジェクトの位置や密度によってエリアカーソルの大きさを動的に変化させる Bubble cursor が Grossman らによって提案されている⁸⁾。この手法では、ポインティング時間の短縮にもつながり、スタート地点とターゲット間にオブジェクトが密に存在してもエリアカーソルがつねに最適な大きさを保つことでその効果を維持することができている。ただし、ユーザの意図とは関係なく、ポインティング動作中にもエリアカーソルの大きさが変化し続けるため、直感性が失われる可能性もある。

2.2.3 ターゲット距離の短縮とターゲット幅の増加

Blanch らは、オブジェクト付近では C-D 比 (control-display ratio) を減少させ、それ以外の場所で C-D 比を増加させる手法を提案している⁴⁾。この手法では、ターゲット付近までは高ゲインで高速にカーソルを動かし、ターゲット付近では低ゲインに設定することで、低速で精度の高いカーソル移動を実現する。これらの考え方は、それぞれ、マウスの移動スペースにおいて、ターゲット距離を減らし、またターゲットの幅を増加することに相当する。しかし、スタート地点とターゲット間に他のオブジェクトが存在すれば、そのオブジェクトに対しても無駄にカーソルの C-D 比を減少させてしまうなど、ターゲット以外のオブジェクトの影響を受けてしまう傾向がある。

2.2.4 まとめ

フィッツの法則に基づいた関連研究における問題点としては、スタート地点とターゲット間やターゲット周辺に、他のオブジェクトが存在する場合に十分対処できないという点があげられる。また、カーソルとマウスという間接指示環境においては両者の論理的な一致が重要であるため、C-D 比やカーソルサイズの動的な変更など、ポインティング中にシステムの介入をとまう手法は、直感性に若干欠けるものになる可能性も指摘できる。これらは、カーソルの位置とターゲットの位置の静的な関係のみを使用してきたために生じる問題であるといえ、柔軟なポインティングの支援を行うためには、カーソルの運動特性などを用いてターゲットを予測するなど、ユーザの意図をシステムに反映させる必要があると考えられる。

3. ポインティングの運動特性に関する実験

3.1 概要

ポインティング中の速度波形を用い、タスク完了前にターゲット位置をシステムが予測できれば、より柔軟なポインティング支援が可能である。Keuning らによって、ポインティング中の速度波形の対称性を用いたターゲット距離の予測が試みられているが¹⁴⁾、150、250、350 pixel (約 10 cm) の短いターゲット距離での検討であり、速度波形の詳細な分析や予測精度の評価には至っていない。本章では、マウスによるポインティング実験を行い、速度波形の特徴量を分析することでターゲット予測手法の検討を行う。主に、ポインティングタスクの前半に現れる特徴量で、ターゲット距離に依存するとされるピーク速度に着目し、そのタスクにおけるターゲット距離との関係を検証する。我々の過去の研究でも、その関係を論じたものもあるが^{20),21)}、ここでは、一般的なマウスという入力デバイスと、PC のディスプレイという出力デバイスを用いて検討を行う。

3.2 実験環境

実験環境の概観を図 2 に示す。計算機は DELL 社の Dimension 8300 (CPU : Pentium 4 3.2 GHz, メモリ : DDR-SDRAM 2GB, OS : Windows XP Professional), ディスプレイは 19 インチの TFT 液晶ディスプレイを 1,280 × 1,024 pixel の解像度で利用する。入力デバイスは Logitech 社の Wheel Mouse (M-BJ58), マウスパッドは Power Support 社の Airpad Pro III (AK-07, 408 × 306 mm) を使用し、ステレオスピーカをディスプレイの両脇に設置する。図 2 に示すように被験者は、机に向かって正対し、ひじの高さが机の高さと同じになるように椅子の高さを調整して座る。マウスを操作する右手は身体正面方向にまっすぐ伸ばし、マウスパッドはひじを軽く曲げた楽な姿勢をとれる場所に配置する。実験用アプリケーションは Microsoft Visual C++ 6.0 を用いて実装し、C-D

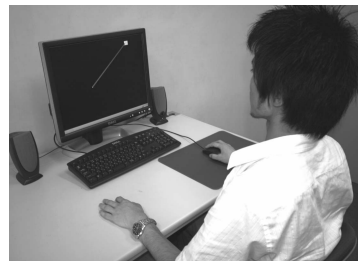


図 2 実験環境

Fig. 2 Experimental setup.



図3 ターゲット
Fig. 3 Target.

比は実験全体を通して 0.5 で一定とする．図中のディスプレイ表示は 5 章で述べる試作インタフェースが動作している様子を示している．

3.3 手続き

被験者は図 3 (a) のように表示されたターゲットアイコンにむけてマウスを用いてカーソルを動かして、できるだけ早く、正確にクリックする．被験者がターゲットアイコンを正しくクリックした場合は、ターゲット表示を図 3 (b) に変化させ成功音を出力する．被験者が誤ってターゲットアイコン以外の場所をクリックした場合は、図 3 (c) に変化させ失敗音を出力する．また、ある程度タスクが進むと、アイコンに描画される顔と音を変化させる．これらは、被験者を飽きさせず、モチベーションを保って実験を遂行させるための工夫である．

3.4 実験計画

ターゲット距離（縦横 4 方向：500, 600, 700, 800, 900 pixel, 斜め 4 方向：707, 848, 989, 1,131, 1,272 pixel）に関する反復実験を行う．ターゲットに関するタスク条件は、45 度ごとに 8 方向 × 5 距離条件で 40 通りとなり、この条件に従って、50 × 50 pixel のサイズのターゲットが、1,280 × 1,024 pixel のディスプレイ領域に 100 pixel 間隔で 13 × 10 の格子状に配置してあるターゲット候補から、1 つ選ばれ表示される．ここでは、ディスプレイ上方向を 0 度として扱い、ターゲットサイズ (50 × 50 pixel) は、一般的なディスプレイサイズと解像度で使用されているアイコンのサイズを基に決定している．被験者がターゲットアイコンを 1 回クリックすると 1 タスクが完了し、次のタスク条件に対応した位置に新たなターゲットを表示する．次のタスクは、被験者のカーソル位置の確認を容易にし、また不必要なマウス移動を避けるために、直前にクリックしたターゲット位置がそのタスクのスタート位置となるように設定する．これを考慮し、単純な反復動作によって被験者にタスク順を完全に把握されるのを避け、かつ各タスク条件を等しい回数実行できるようにするため、40 通りのタスク条件を図 4 (a) ~ (f) に示す順序で行う．図 4 中の で囲まれた数字は昇順にその図中のタスクが実行されていくことを示している．図 4 (a) ~ (e) に示すように、同一ターゲット距離

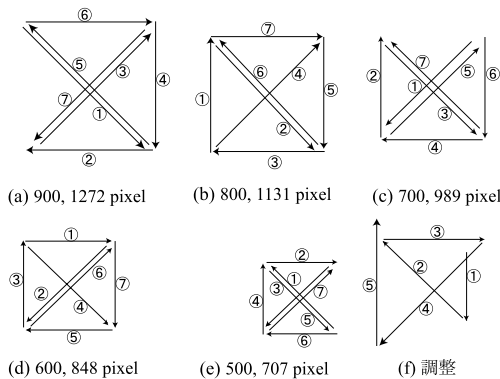


図4 タスク順序
Fig. 4 Task order.

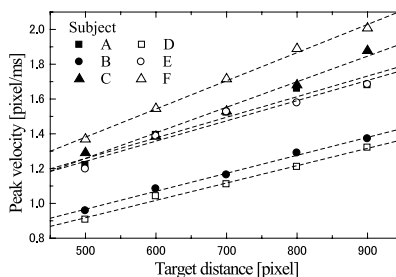


図5 ターゲット距離に対するピーク速度
Fig. 5 Target distance vs. peak velocity.

を 7 試行ずつ続け、図 4 (f) に示すように最後の 5 試行で 40 タスク条件すべてを提示し終える順になっている．一巡したあと、次の一巡は逆順で行い、これを繰り返す．被験者は、120 回の練習後、途中 1 分間の休憩を挟み、1 タスクあたり 16 回反復し、計 640 回 (40 task × 16 repeat = 640) の試行を行う．実験時間は約 30 分程度を要する．

被験者 6 人 (平均年齢 23.3 歳の男性) に対して実験を行い、マウスから得られた位置データを基に、ピーク速度、ターゲット距離と移動角度の関係を分析する．

3.5 実験結果

縦横 4 方向におけるターゲット距離に対するピーク速度を図 5 に示し、ターゲット距離と移動角度に対するピーク速度を図 6 に示す．図 5 は 6 人の被験者それぞれのピーク速度の平均値と線形近似直線を示しており、図 6 は 6 人の被験者全員の平均値をターゲット距離ごとに分けて示している．実験後の被験者アンケートの結果、ターゲット距離も含めて次のタスクを完全に予測できた被験者は見られなかった．図 5 から、ピーク速度はターゲット距離との間に強い相関があり、被験者ごとに係数は異なるものの、一次関数的に近似が可能であることが分かる．この傾向は、斜めの 4 方向においても見られた．また、図 6 から、同

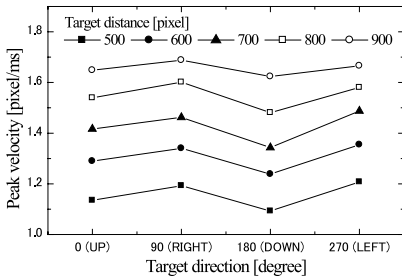


図 6 ターゲット距離およびターゲット角度に対するピーク速度
Fig. 6 Target distance and target direction vs. peak velocity.

じターゲット距離であってもポインティング移動角度によってピーク速度の大きさが異なり、ディスプレイに対して水平（左右）方向ほど大きくなることが確認できる。したがって、ターゲット角度を考慮したうえで、ポインティングタスク中の前半に現れるピーク速度からターゲット距離を予測できると考えられる。

4. ピーク速度からターゲットを予測する手法

3章で述べたとおり、カーソル移動中のピーク速度 V_p とターゲット距離 D の関係は、式 (1) のように一次式で近似することができる。

$$D = a \times V_p + b \tag{1}$$

ここで、 a 、 b は実験によって求まる定数である。またその関係は図 5 に示したように、被験者ごとに異なっているため、その被験者の運動特性に適した関係式を用意する必要がある。提案手法では、まず、ユーザが通常のポインティングタスクを行う際のピーク速度とターゲット距離をサンプリングし、線形回帰分析によって、ポインティングタスクの方向別に、式 (1) に相当する回帰式を取得する。その後、ユーザがポインティングタスクを行う際にピーク速度を検出し、方向別に取得した回帰式からその時点での移動方向に最も近い方向の回帰式にピーク速度を代入することで、そのタスクにおけるターゲット距離を予測する。このとき、そのタスクのスタート位置からその時点でのカーソル位置までのベクトルの平均を、ユーザが意図する移動方向と推定する。ピーク速度は、タスクが完了する前に検出する必要があるため、カーソルの速度をリアルタイムで監視し、最初に速度が減少したときの速度がある閾値以上であった場合にピーク速度として用いる。

5. 実験

5.1 ターゲット予測インタフェースの試作

4章で述べた予測手法を用いて、ポインティング動

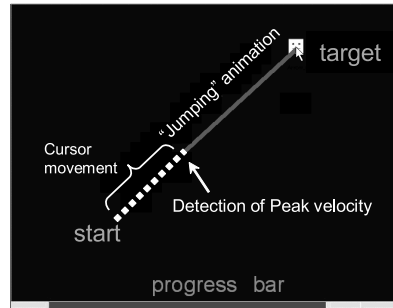


図 7 Delphian Desktop の動作
Fig. 7 Delphian Desktop.

作中のピーク速度を用いて予測地点を算出し、予測地点までカーソルをジャンプさせるインタフェース「Delphian Desktop」を試作した。図 7 に示すように、ユーザがジャンプ先を容易に認識できるようにするため、効果音とともにジャンプの軌跡をアニメーション表示する。この試作インタフェースでは、カーソルジャンプは瞬間的に行うが、ジャンプ後はユーザのマウス操作における腕の慣性の影響を吸収するために、カーソルを 100 ms の間予測地点に固定し、そのタイミングに軌跡アニメーションを出力する。軌跡アニメーションは、ピーク速度検出地点と予測地点を軌跡で瞬時に結び、その後、100 ms の間に、ピーク速度検出地点側の線端が予測地点へ向けて移動し、予測地点に軌跡が吸い込まれるような動作になっている。そのため、ユーザのポインティング動作における調整運動はアニメーションによる視覚的なフィードバックによって示される。

5.2 概要

試作インタフェースの評価実験は、3章で述べたポインティングの運動特性に関する実験と同様の実験環境、ターゲット表示、手続きを用いる。ただし、本実験では、Calibration フェーズと Prediction フェーズの 2つのフェーズで構成する。いずれのフェーズでも被験者は与えられた開始点からマウスを用いてターゲットのアイコンをクリックするタスクを繰り返し実行する。Calibration フェーズでは、得られた被験者のポインティング特性からユーザ個々に一次式で表した回帰式を取得し、Prediction フェーズでは、前フェーズで得られた回帰式を用いてターゲット予測を行ったポインティング実験を行う。ただし、試作インタフェースを通常のポインティング環境と比較するため、Calibration フェーズにおけるポインティング実験の結果を予測なしの Non-Prediction フェーズの結果として扱うことにする。

3章の実験とは異なる右利きの被験者 16 人(平均年齢 23.0 歳の男性 8 人と平均年齢 25.8 歳の女性 8 人)に対して,ポインティング実験を行い,ポインティング時間,エラー,ユーザの負荷の 3 つの項目について分析し,試作インタフェースを評価する.

5.3 実験計画

予測の有無(Non-Prediction, Prediction),ターゲット距離(縦横 4 方向:500, 600, 700, 800, 900 pixel, 斜め 4 方向:707, 848, 989, 1,131, 1,272 pixel)の 2 要因反復実験を行う.タスク順も 3 章で述べた実験と同様で,図 4 に示す順である.

Calibration フェーズ(Non-Prediction フェーズ)は,それぞれ練習パート,本番パート 1,本番パート 2 と,休憩を挟んで 3 パートに分割する.練習パートでは 120 回行い,各本番パートでは 1 タスクあたり 8 回の試行を行い,それぞれ計 320 回ずつ行う(40 task \times 8 repeat \times 2 part = 640).また,5 分間の休憩の後,Prediction フェーズを前フェーズと同構成で行う.したがって,被験者あたり計 1,280 回(640 \times 2 prediction style)の試行となり,実験時間は約 1 時間程度を要する.

6. 実験結果

6.1 ポインティング時間

予測の有無とターゲット距離を要因とする二元配置の分散分析の結果,予測の有無($F(1, 300) = 12.593$, $p < .001$),ターゲット距離($F(9, 300) = 23.893$, $p < .001$)および予測の有無とターゲット距離の交互作用($F(9, 300) = 5.626$, $p < .001$)で有意差が認められた. Non-prediction フェーズとしての Calibration フェーズ(以降, Non-prediction フェーズ)と Prediction フェーズのターゲット距離に対するポインティング時間を図 8 に示す. 図中の棒グラフは全被験者の平均値であり,エラーバーは標準偏差である. 予測が有効に働いているといえる距離の閾値を調べるために,ターゲット距離に関する事後検定を行った結果,ターゲット距離が 800 pixel 以上の範囲で提案手法による予測が有効であることが分かった($p < .05$). しかし,逆に 800 pixel よりも短い距離では,本手法によってポインティング時間が増加する傾向が見られた. 2 章で述べたフィッツの法則は,1 次元での実験によるものであり,GUI の評価などの実用上を考え,Accot らによってポインティングにおける難易度 ID は,式 (2) で表されるように,ターゲット高さ H も含めた二次元の形に拡張されている. フィッツの法則では,この難易度 ID とポインティング時間

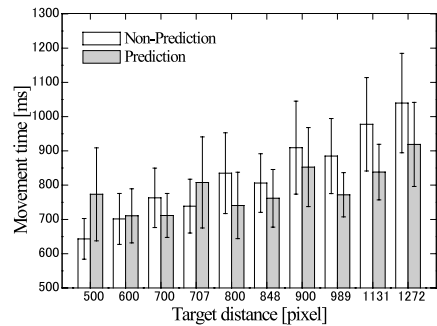


図 8 ターゲット距離に対するポインティング時間
Fig. 8 Target distance vs. movement time.

表 1 各タスク条件に対する ID

Table 1 ID for task condition.

D [pixel]	W, H [pixel]	ID [bit]
500	50	3.92
600	50	4.16
700	50	4.38
707	50	4.39
800	50	4.56
848	50	4.64
900	50	4.72
989	50	4.85
1,131	50	5.04
1,272	50	5.21

は線形の関係にあるとされており,この法則の観点からポインティング時間を分析する.

$$ID = \log_2 \left(\sqrt{\left(\frac{D}{W}\right)^2 + \eta \left(\frac{D}{H}\right)^2} + 1 \right) \quad (2)$$

式 (2) 中の D はターゲットまでの距離で, W, H はそれぞれターゲット幅と高さである. η はデバイスに依存する定数であり,ここでは 1 を用いる.本実験での各タスク条件に対応する難易度 ID を算出した結果を表 1 に示し, ID に対するポインティング時間を図 9 に示す. 図 9 から,予測ありの場合は,予測なしの場合と比較して若干適合度が低い結果となっているが, ID に対する変化の傾きが緩やかである. 図 8 と同様に,ターゲット距離が 800 pixel 付近の ID (約 4.5) よりも大きい場合,試作インタフェースにおけるターゲット予測がポインティング時間の短縮に有効であることが確認できる. また逆に,800 pixel 以下($ID < 4.5$)では,ポインティング時間を増加させる負の効果が生じている.

6.2 エラー

本実験においてのミスクリックの割合は 6%程度見られ,過去報告されているポインティング実験での平均エラー約 4%¹⁶⁾ を超える結果であった. しかし,

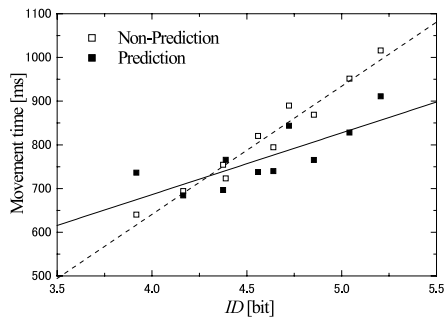


図 9 ID に対するポインティング時間

Fig. 9 ID vs. movement time.

試作インタフェースでは、カーソルジャンプという比較的大きなシステムの介入を試みたものであり、通常のポインティングにおいても4%程度のエラーが生じることを考えると、今回用いたカーソルジャンプは、ユーザのポインティング精度に大きく悪影響を与えるものではなかったと考えられる。予測距離とターゲット距離の誤差率の絶対値平均は15.4%であり、予測角度とターゲット方向の誤差の絶対値平均は3.89度であった。距離に関する誤差が大きくなった理由は、試作インタフェースにおいてリアルタイムに行うピーク速度の検出に問題があったためだと考えられる。実際のポインティングではつねになめらかな速度波形を描くとは限らず、閾値の設定の問題などによりピーク値より小さな速度値を検出し、ターゲット距離を短く予測してしまう場合がいくつか見られた。これらは今後、ピーク速度検出アルゴリズムの改良に加えて、インタフェース内での回帰式の最適化などで改善していくことが必要であると考えられる。角度の誤差については、カーソルを大きな弧を描いて動作させる被験者が数人存在しており、その対応が十分できていなかったために増加したと思われる。また、解析の結果、予測角度の誤差が5度以下に実現できたときはポインティング時間も小さくなる傾向が見られているため、この角度検出アルゴリズムの改良は重要な課題である。

6.3 ユーザ負荷

予測ありの実験において、スタート位置からジャンプ直前の位置までの距離とジャンプ先の位置からターゲット距離を合計した距離は、実際にカーソル(マウス)を動かす必要のあった距離を表す。この距離に対するターゲット距離の割合をゲインとし、ターゲット距離との関係を図10に示す。図中の棒グラフは全被験者の平均値で、エラーバーは標準偏差を表している。ターゲット距離が500, 600, 700といった縦横方向の場合と、707, 848, 989などの斜め方向の場合を分け

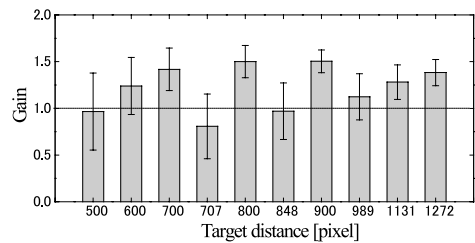


図 10 ターゲット距離に対するゲイン

Fig. 10 Target distance vs. gain.

て考えると、ターゲット距離が長くなるにつれてゲインは増加する傾向が見られる。また全タスクの平均ゲインも1.21と1を超えており、マウス移動距離においても被験者の負荷を軽減できていることが確認できる。また、Calibrationフェーズの後にPredictionフェーズを行ったが、被験者の主観評価の中に、Predictionフェーズでは腕の負担が少ないとの感想が多く見られた。これらは、Gainの増加という見方以外に、試作インタフェースでのピーク速度検出後のジャンプが、ポインティング負荷が集中する調整時間を短縮することにつながったためと考えることができる。今後のポインティング関連の研究において、ユーザの負荷に関してより詳細な議論をするためには、この点についての定量的な評価が重要になると考えられる。

7. 考 察

提案手法および試作インタフェースでは、比較的長いターゲット距離に対するポインティングタスクではポインティング時間を短縮することに成功し、有効性を示すことができた。さらに、関連研究が持つ問題点についても、ユーザが意図するターゲットを予測することで、ある程度対処できると考えられる。たとえば、ターゲットと同一線上に並んだオブジェクトの影響を大きく受けるC-D比の変更技術⁴⁾は、本提案手法を適用し、ターゲット予測後にC-D比の減少を行うことで、ターゲットまでに存在するオブジェクトに対する無駄なC-D比の変更を避けることが可能となる。そのほかにも、カーソルの動きを予測し、ターゲットを特定したうえでターゲットサイズを拡大²⁶⁾させるといった技術の組合せも可能であり、ターゲット予測技術は、より柔軟で有効なポインティング支援技術に発展する可能性を持つといえる。

図8に示したように、800 pixel未滿のターゲット距離条件では、試作インタフェースによって、ポインティング時間が増加してしまっている。今回の実装では、ジャンプ距離にかかわらず、ジャンプアニメーション

の長さはずねに一定としていた．そのため，800 pixel 以下では，このアニメーションの時間が適切ではなかった可能性もあり，カーソルがターゲット上に到達していても，クリックする動作が遅れてしまっていると見受けられる様子が多数観察されている．このクリックまでの時間の短縮を図るためには，予測距離に応じたアニメーション時間や種類の設定が重要になると思われる．また，試作インタフェースを実際の GUI に適用することを考えると，他にもジャンプの機能を作動させるべきでない状況が存在すると考えられる．閾値を設けて，小さなピーク速度検出時にはジャンプの機能を作動させないようにすることや，マウスのいずれかのボタンを押した状態でのポインティング時のみ，予測機能が働くようにするなど，状況に応じたポインティング支援の適切な作動が必要で，そのための判断基準を明らかにすることも今後の重要な課題である．

2章の関連研究では，ポインティング中のカーソルエリアの動的な変更など，システムの介入が直感性の低下を招く恐れがあると述べてきた．試作インタフェースにおいてもカーソルジャンプというシステムの介入は行われるが，ポインティングタスク中のターゲットへ向けて狙いを定める計画時間は，システムの介入なしにユーザによって操作させている点で，他の研究と大きく異なる．ピーク速度まではポインティング運動前にユーザによって計画されているという Heath らの研究報告¹²⁾を考慮すると，本試作インタフェースでピーク速度以後に行われるカーソルジャンプは，運動前の計画に含まれていない時間帯での介入であり，計画を途中で狂わされることによる違和感や，直感性の低下は少なかったと考えられる．カーソルジャンプという比較的大きなシステムの介入ではあったが，被験者の主観評価の結果において，ジャンプアニメーションが有効であったなど，ジャンプに対する違和感は少ないという意見が多数見られている．

通常の PC デスクトップ環境などを考えた場合，ターゲット周辺に他のオブジェクトが密に存在することが多々あるが，この問題に対しては，今後の課題としている移動角度やリアルタイムピーク速度検出アルゴリズムの改善で対応していくことが必要である．しかし，頻繁に利用するフォルダやアプリケーションアイコンなどをまとめて複数の領域に配置しているような広いデスクトップを想定すると，ターゲットを含むターゲット群やその領域の中心を予測地点としてジャンプさせることは現状のシステムでも可能であり，単一ターゲットの特定が困難であっても，ポインティング時間や移動距離の短縮に関して，十分効果を発揮で

きるものと考えられる．

現在広く普及している OS では，マウスやタッチパッドによるポインティングを支援するために加速機能が設けられている．マウスの動作が速ければカーソルは加速するという，ユーザのポインティング速度に応じて C-D 比を変化させるポインティング支援技術である．試作インタフェースでは，ポインティングでの基礎特性に基づいており，評価の第 1 段階として一般的な環境との比較実験を行う必要があると考え，本論文では，加速機能を試作インタフェースの比較対象とせず，キャリブレーションの段階からも加速は用いない C-D 比が一定の環境を設定している．そのため，本提案手法の有効性についての 800 pixel という数字は SXGA の解像度で C-D 比が 0.5 程度の環境に制限されるものであり，加速機能が加わった場合には，キャリブレーションや予測モデルを再検討する必要がある．単純に試作インタフェースと加速機能のポインティング時間の比較を行ったとしても，ポインティング時間の短縮の割合は，加速の設定にもよるが，本論文で示した C-D 比が一定の場合よりも少なくなると予想される．しかし，加速機能は，ターゲット予測手法と競合するものではないため，提案手法によってターゲットを予測した後に C-D 比（加速度）を動的に変更するという形をとることで，提案する予測手法の有効性を加速機能と対等に比較・評価することが可能である．試作インタフェース内で行うジャンプというシステムの介入はずねに最適な手法であるとは必ずしもいえず，状況によって様々な使い分けが必要と考えられ，またそれが可能であることも本提案手法の利点の 1 つである．したがって，この加速機能は，提案手法の拡張を考えるうえでの，共存が可能な関連技術として，予測の表現方法やその使い分けを検討する際に考慮していきたい．

多くのノートパソコンには，入力デバイスとしてタッチパッドやトラックボールが搭載されている．我々の提案手法は，ポインティング中の速度の波形が，図 1 に示すような 1 つのピーク速度を含んだ運動として観測できる場合に適用可能である．トラックボールのようなポインティングデバイスでは，ポインティング中の速度波形の分布がマウスと比べて大きく異なり，本提案手法のままでは適用することはできないと思われるが，タッチパッドについては，マウスと同様の速度波形が得られるため，提案手法および試作インタフェースが適用でき，有効な支援技術となりうると考えられる．

8. おわりに

本論文では、ポインティング中のピーク速度からターゲット距離を予測する手法の提案について述べ、提案手法を用いた予測型インタフェース「Delphian Desktop」の概要と評価結果について述べた。その結果、提案する予測手法および試作インタフェースは、800 pixel 以上のターゲット距離においてポインティング時間短縮に有効（解像度：SXGA，C-D 比：0.5 の環境において）であることを確認し、また、ポインティング時に必要な移動距離を短縮し、ユーザの操作負担を軽減できることを確認した。さらに、関連研究において指摘される問題点などを本手法がおぎなえる可能性を示すことができたと考える。今後は、試作インタフェースの改良や、予測型インタフェースにおける適切なポインティング支援の方法など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 21 世紀 COE プログラム（研究拠点形成費補助金）の研究助成と総務省戦略的情報通信研究開発推進制度（SCOPE）による。

参考文献

- 1) Accot, J. and Zhai, S.: Refining Fitts' law models for bivariate pointing, *Proc. Conference on Human Factors in Computing Systems (CHI '03)*, pp.193–200 (2003).
- 2) Balakrishnan, R.: "Beating" Fitts' law: virtual enhancements for pointing facilitation, *International Journal of Human-Computer Studies*, Vol.61, No.6, pp.811–821 (2004).
- 3) Baudisch, P., Cutrell, E., Robbins, D., Czerwinski, M., Tandler, P., Bederson, B. and Zierlinger, A.: Drag-and-pop and drag-and-pick: techniques for accessing remote screen content on touch-and pen-operated systems, *Proc. INTERACT '03*, pp.57–64 (2003).
- 4) Blanch, R., Guiard, Y. and Beaudouin-Lafon, M.: Semantic pointing: improving target acquisition with control-display ratio adaptation, *Proc. Conference on Human Factors in Computing Systems (CHI '04)*, pp.519–526 (2004).
- 5) Collomb, M., Hascœt, M., Baudisch, P. and Lee, B.: Improving drag-and-pop on wall-size displays, *Proc. Graphics Interface '05*, pp.25–32 (2005).
- 6) Elliott, D., Chua, R. and Helsen, W.F.: A century later: Woodworth's (1899) two-component model of goal-directed aiming, *Psychological Bulletin*, Vol.127, No.3, pp.342–357 (2001).
- 7) Fitts, P.M.: The information capacity of the human motor system in controlling the amplitude of movement, *Journal of Experimental Psychology*, 47, pp.381–391 (1954).
- 8) Grossman, T. and Balakrishnan, R.: The Bubble cursor: enhancing target acquisition by dynamic resizing of the cursor's activation area, *Proc. Conference on Human Factors in Computing Systems (CHI '05)*, pp.281–290 (2005).
- 9) Guan, Z., Ren, X., Li, Y. and Dai, G.: Zoom selector: a pen-based interaction technique for small target selection, *IPSJ Journal*, Vol.45, No.8, pp.2087–2097 (2004).
- 10) Guiard, Y., Blanch, R. and Beaudouin-Lafon, M.: Object pointing: a complement to bitmap pointing in GUIs, *Proc. Graphics Interface '04*, pp.9–16 (2004).
- 11) Harris, C.M. and Wolpert, D.M.: Signal-dependent noise determines motor planning, *Nature*, Vol.394, No.6695, pp.780–784 (1998).
- 12) Heath, M., Hodges, N., Chua, R. and Elliott, D.: On-line control of rapid aiming movements: unexpected target perturbations and movement kinematics, *Canadian Journal of Experimental Psychology*, Vol.52, No.4, pp.163–173 (1998).
- 13) Kabbash, P. and Buxton, W.: The "Prince" technique: Fitts' law and selection using area cursors, *Proc. Conference on Human Factors in Computing Systems (CHI '95)*, pp.273–279 (1995).
- 14) Keuning, H. and Houtsma, A.J.M.: Cursor displacement and velocity profiles for targets in various locations, *Proc. EuroHaptics '01*, pp.108–112 (2001).
- 15) MacKenzie, C.L., Marteniuk, R.G., Dugas, C. and Eickmeier, B.: Three-dimensional movement trajectories in Fitts' task: implications for motor control, *Quarterly Journal of Experimental Psychology*, No.39A, pp.629–647 (1987).
- 16) MacKenzie, I.S.: Fitts' law as a research and design tool in human-computer interaction, *Human-Computer Interaction*, No.7, pp.91–139 (1992).
- 17) McGuffin, M. and Balakrishnan, R.: Acquisition of expanding targets, *Proc. Conference on Human Factors in Computing Systems (CHI '02)*, pp.57–64 (2002).
- 18) Meyer, D.E., Abrams, R.A., Kornblum, S., Wright, C.E. and Smith, J.E.K.: Optimality in human motor performance: ideal control of rapid aimed movements, *Psychological Review*, Vol.95, No.3, pp.340–370 (1988).

- 19) Plamondon, R. and Alimi, A.M.: Speed/accuracy trade-offs in target-directed movements, *Behavioral and Brain Sciences*, No.20, pp.279–349 (1997).
- 20) 高木 龍, 北村喜文, 岸野文郎: ポジショニングタスクにおけるターゲット予測に関する一検討, 電子情報通信学会総合大会講演論文集, A-15-17, p.316 (2001).
- 21) 高木 龍, 北村喜文, 岸野文郎: 並進と回転を伴う仮想物体操作時の運動特性の分析, ヒューマンインタフェース学会研究報告集, Vol.2, No.2, pp.25–30 (2000).
- 22) Uno, Y., Kawato, M. and Suzuki, R.: Formation and control of optimal trajectory in human multijoint arm movement, *Biological Cybernetics*, Vol.61, pp.89–101 (1989).
- 23) Walker, N., Meyer, D.E. and Smelcer, J.B.: Spatial and temporal characteristics of rapid cursor-positioning movements with electromechanical mice in human-computer interaction, *Human Factors*, Vol.35, No.3, pp.431–458 (1993).
- 24) Woodworth, R.S.: The accuracy of voluntary movement, *Psychological Review Monograph Supplement*, Vol.3, No.13, pp.1–119 (1899).
- 25) Worden, A., Walker, N., Bharat, K. and Hudson, S.: Making computers easier for older adults to use: area cursors and sticky icons, *Proc. Conference on Human Factors in Computing Systems (CHI '97)*, pp.266–271 (1997).
- 26) Zhai, S., Conversy, S., Beaudouin-Lafon, M. and Guiard, Y.: Human on-line response to target expansion, *Proc. Conference on Human Factors in Computing Systems (CHI '03)*, pp.117–284 (2003).

(平成 18 年 3 月 6 日受付)

(平成 18 年 11 月 2 日採録)



高嶋 和毅

2006 年大阪大学大学院情報科学研究科マルチメディア工学専攻博士前期課程修了。同年同専攻博士後期課程入学, 現在に至る。ヒューマンインタフェースの研究に従事。電子情報通信学会, 日本バーチャルリアリティ学会学生会員。



浅野 岳史

2005 年大阪大学大学院情報科学研究科マルチメディア工学専攻博士前期課程修了。同年野村総合研究所入社, 現在に至る。在学中ヒューマンインタフェースの研究に従事。



エフッド シャーリン

1997 年ベングリオン大学 (イスラエル) にて修士取得。2003 年アルバータ大学にて博士取得。同年大阪大学大学院情報科学研究科特任助手。2004 年よりカルガリー大学コンピュータサイエンス学部助教授, 現在に至る。ヒューマンロボットインタラクション, ヒューマンコンピュータインタラクションの研究に従事。ACM 等会員。



北村 喜文 (正会員)

1987 年大阪大学大学院基礎工学研究科博士前期課程修了。同年キャノン株式会社情報システム研究所, 1992 年 ATR 通信システム研究所, 1997 年大阪大学大学院工学研究科助教授, 2002 年より同大学大学院情報科学研究科助教授。ACM, 電子情報通信学会, ヒューマンインタフェース学会, 日本バーチャルリアリティ学会等会員。博士 (工学)。



岸野 文郎

1971 年名古屋工業大学大学院電子工学専攻修士課程修了。同年日本電信電話公社 (現 NTT) 電気通信研究所入所。同ヒューマンインタフェース研究所において, 高速・広帯域通信システムの研究・実用化, 画像処理の研究に従事。1989 年 ATR 通信システム研究所知能処理研究室室長。臨場感通信, 画像処理等の研究に従事。1996 年大阪大学大学院工学研究科教授, 2002 年より同大学院情報科学研究科教授。電子情報通信学会, 映像情報メディア学会, ヒューマンインタフェース学会, 日本バーチャルリアリティ学会等会員。博士 (工学)。