

ポインティングタスクにおけるターゲット予測を 利用したインタフェースの提案と検討

浅野 岳史^{*1} Ehud Sharlin^{*1, *2} 高嶋 和毅^{*1} 北村 喜文^{*1} 岸野 文郎^{*1}

^{*1} 大阪大学大学院情報科学研究科

^{*2} University of Calgary

ポインティングタスクにおいて、カーソルのピーク速度とターゲット距離に相関があるという特性がある。この特性を利用してポインティングタスク中のターゲット位置を予測するインタフェースを提案する。まず、ユーザが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、取得したデータに対して回帰分析を行うことで回帰式を獲得する。その後、タスク中に検出したピーク速度を獲得した回帰式に代入することで、そのタスクにおけるターゲットの位置を予測する。これを利用して、GUI環境においてポインティング時間を短縮し、ユーザの負担を軽減するためのインタフェースを試作し、実験によって有効性を評価、検討する。

Proposing an interface that predict a target from a peak velocity in a pointing task

Takeshi Asano^{*a}, Ehud Sharlin^{*b, *a}, Kazuki Takashima^{*a}, Yoshifumi Kitamura^{*a} and Fumio Kishino^{*a}

^{*a} Graduate School of Information Science and Technology, Osaka University

^{*b} University of Calgary

We propose an interface which introduces spatial prediction to current desktop interaction reducing dramatically the time required to point to a faraway target. The predictions of the user spatial intentions are based on tracking user actions manifested through cursor movements. The interface is constantly sampling the cursor location and velocity and attempts to predict the user goal location using a peak velocity. In this paper we present the design and implementation principles. We detail the system evaluation methodology and some results from user study.

1 はじめに

大画面で高解像度のディスプレイが普及してきている。それに伴い、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り、この傾向は今後更に顕著になると考えられる。そのため、ポインティング時間を短縮し利用者の作業労力軽減を図ることが求められている。

ポインティング時間の短縮を狙った研究として、ターゲットのアイコンなどのオブジェクトの大きさを拡大する手法 [1, 2] や、カーソルの移動方向にある最も近いオブジェクトへカーソルをジャンプさせる手法 [3]、デバイス操作量とポインタ移動量の比である Control-Display (C-D) 比をオブジェクトとそれ以外の場所で変化させる手法 [3] などがあるが、これらの手法は主にカーソルとターゲットの間の距離や方向といった静的な関係を用いているため、カー

ソルの周囲や進行方向にあるターゲット以外のオブジェクトもすべてターゲットであるかのように扱ってしまう。

ポインティング作業中のカーソルの移動速度の変化といった運動特性を利用してターゲットの位置を予測することができれば、高精度で柔軟性に富んだインタフェースを構築できる可能性がある。そこで本稿では、GUI環境のポインティングタスク中のピーク速度を検出し、それを利用してターゲットを予測する手法を提案し検討する。

2 関連研究

ポインティングタスクはGUIを利用する上で最も多く行われる基本的なタスクであるため、このタスクに必要な時間を短縮し、利用者の作業労力を軽減できれば非常に有効である。そのため、多くの研究がなされてきた。McGuffinらとZhaiらは、カーソルがターゲットに近づいたときにターゲットのサイズを拡大する手法を提案し、有効であることを示した[1, 2]。しかし、カーソルがターゲットの近くに移動するまでの動きに対しては影響がないため、ターゲット距離が長くなるほど、ポインティング時間を短縮できる割合が小さくなる。

Guiardらはカーソルをオブジェクトからオブジェクトへジャンプさせるオブジェクトポインティングと呼ぶ手法を提案した[3]。カーソルはオブジェクト上では通常通りに動き、オブジェクトの上からカーソルが出ようとした場合はカーソルの移動方向にある最も近いオブジェクトへカーソルをジャンプさせるというものである。しかし、距離を考慮せずに方向のみで移動先のオブジェクトを決定しているため、カーソルに対して同方向に存在するオブジェクト数が増えると、ターゲットとは別のオブジェクトに間違っ

て移動しやすい。Blanchらは、オブジェクトとそれ以外の場所でC-D比を変化させるセマンティックポインティングと呼ぶ手法を提案した[4]。オブジェクト上ではC-D比を大きくしてカーソルの動きを遅くし、それ以外の場所ではC-D比を小さくしてカーソルの動きを速くするという方法で、ポインティング時間の短縮を図っている。しかし、カーソルとターゲットの間にオブジェクトが複数存在した場合、それらのオブジェクト上でもカーソルの動きが遅くなるため、無駄にポ

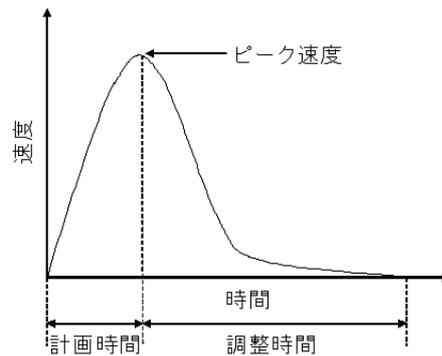


図 1: ポインティングタスク時の速度変化

インティング時間を延ばしてしまう傾向がある。

これら研究では、いずれも、操作中のある時刻のポインタの位置のみに基づいて予測しているため、効果が不十分である。移動速度の変化など、人間の運動特性を利用してターゲットが予測できれば、より精度良く、柔軟にターゲットを予測し、ポインティング時間の短縮と作業労力の軽減を図ることができると考えられる。

3 運動特性の分析

人間の知覚運動に関する研究は昔から広く研究されており、あるスタート地点からある目標地点まで手を動かすポインティング動作は、手の速度変化の特徴から、計画時間と調整時間の2つのフェーズに分けられることが知られている[5]。計画時間は、視覚フィードバック情報を利用せずに目標地点に向かって手の動きを加速させていくフェーズであり、調整時間は、視覚フィードバック情報を利用して手の速度を調整しながら目標地点まで動かすフェーズである。

また、ポインティングタスクにおける運動特性の研究では、マウスなどのポインティングデバイスを用いた際の運動特性の研究も行われており[6]、マウスを使用した際のポインティングタスクにおける一般的なカーソルの速度変化は図1のようになることがわかっている。そこで、このようなカーソルのピーク速度とターゲット距離の関係を確認する実験を行った[7, 8]。

実験では、仮想物体操作時の運動特性を計測し分析を行った。被験者は男性6名であり、6自由度センサの取り付けられた一辺5cmの紙製立方体のコントローラを机の上に設置して使用し、被験者の前方に設置されたディスプレイに表示される仮想空間平面

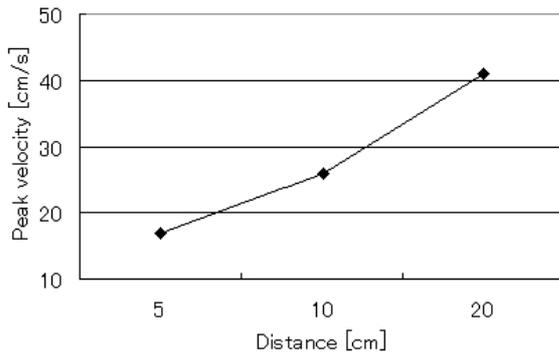


図 2: ターゲット距離に対するピーク速度の変化

上の立方体カーソルを操作する。また、コントローラの移動量はカーソルの移動量に 1 対 1 で対応付けており、被験者は仮想平面上に表示されたターゲットにカーソルを合わせるといったタスクを繰り返す。タスク条件として、移動方向は被験者の前方を 0 度とし、時計回り方向に -90, -45, 0, 45, 90 度方向の 5 つ、ターゲット距離は 5cm, 10cm, 20cm の 3 つを用意する。実験後、6 自由度センサから得られた位置データに対して差分をとることにより、カーソルの速度を算出する。その後、これらのデータについて、ターゲット距離と移動方向による影響を検討した。

実験結果として、ターゲット距離の変化に対するピーク速度への影響を図 2 に示し、ターゲット距離と移動方向の変化に対するピーク速度への影響を図 3 に示す。

図 2 から、ピーク速度はターゲット距離と相関があり、ターゲット距離が長くなるほどピーク速度は一次関数的に大きくなっていることがわかる。また、図 3 から、ピーク速度は移動方向によって大きさが異なり、左右方向ほどその速度が大きくなることから、ピーク速度からターゲット距離を予測できる可能性があると考えられる。

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、ポインティングタスクにおけるカーソル移動中のピーク速度 (PV) とターゲット距離 (D) に相関があるため、両者の関係は式 (1) のように一次式で表すことができる。

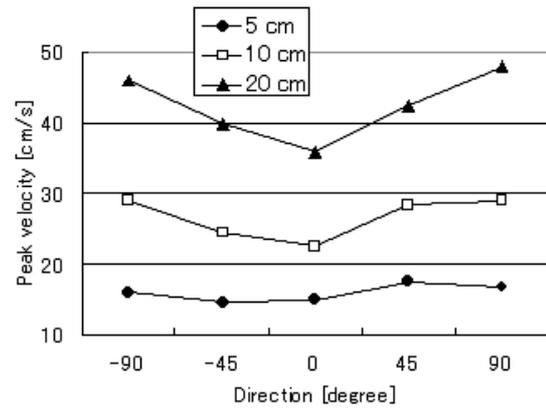


図 3: ターゲット距離と移動方向に対するピーク速度の変化

$$D = a \times PV + b \quad (1)$$

この式を利用して、ピーク速度からターゲットを予測する手法を提案する。なお、 a , b は実験によって求まる定数である。まず、ユーザがポインティングタスクを行う時のピーク速度とターゲット距離をサンプリングする。それらのサンプルを線形回帰分析することで、式 (1) に相当する回帰式を獲得する。なお、図 3 に示したように、ターゲット距離が同じであっても移動方向によってピーク速度が異なるため、回帰式はポインティングタスクの方向別にそれぞれ獲得する。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、方向別に獲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ものを選択する。この時、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であるとする。そして、選択した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。

ピーク速度に関しては、予測に使うことを考える場合、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時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利用する。

ターゲット距離を予測した後は、そのタスクのスタート位置から移動方向へ予測した距離だけ延長した点を算出し、その点に最も近いオブジェクトをターゲットと予測する。

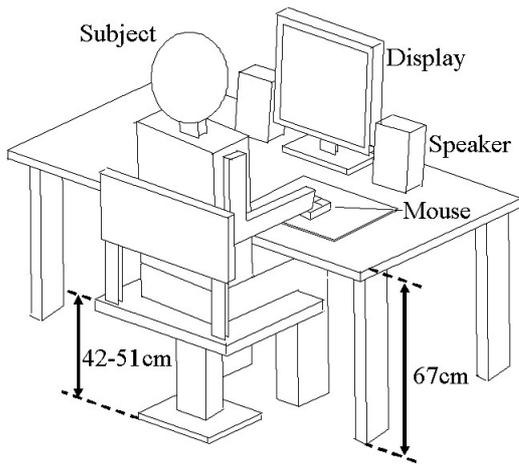


図 4: 実験環境

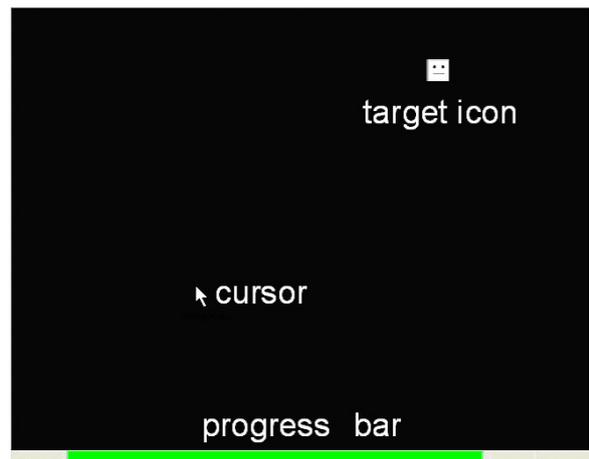


図 5: 画面表示

5 実験

5.1 実験環境

実験環境の概観を図 4 に示す．計算機は DELL 社の Dimension 8300 (CPU : Pentium 4, 3.2GHz, メモリ : DDR-SDRAM 1GB, OS : Windows XP Professional), ディスプレイは 17 インチの TFT 液晶ディスプレイを 1280×1024 ピクセルの解像度で利用する．また, ステレオスピーカ (Harman/Kardon HK206) をディスプレイの両脇に設置する．入力デバイスは Logitech 社の Wheel Mouse (M-BJ58) [9], マウスパッドは Power Support 社の Airpad Pro III (AK-07, 408×306mm) [10] を使用する．被験者が座る椅子は座席の高さが 42~51cm の可変式のものを使用し, 机は 67cm の高さのものを使用する．各被験者には, 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ってもらう．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, ひじを軽く曲げた楽な姿勢を取れる場所が作業領域の中心となるようにマウスパッドを配置する．

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 比は実験全体を通して 0.5 とする．

5.2 概要

実験は 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 フェーズの各タスクではカーソルの速度を記録し, それらの記録から各タスクのピーク速度を算出する．被験者による一連のタスクの後, 方向別にピーク速度とターゲット距離のデータを用いて線形回帰分析を行い, 一次式で表した回帰式を得る． Prediction フェーズでは, Calibration フェーズで得られた回帰式を用いてターゲットを予測する．

5.3 画面表示とタスク

Calibration と Prediction の両フェーズにおいて, 一つのタスクは図 5 のようにディスプレイ上に表示されたターゲットアイコンを, マウスを動かしてできる限りすばやく正確にクリックしてもらうことで完了する．被験者にはこのタスクを何度も繰り返してもらう．また, 実験の間, 被験者が退屈せずに集中力を保てるようにするため, ターゲットアイコンには図 6 のような顔を描画する．通常時は図 6(a) の顔が表示されており, 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図 6(b) に変化して成功音が鳴り, 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 6(c) に変化して失敗音が鳴る．図 5 に示された画面下部のプログレスバーはタスクの進行度を示しており, タスクが一つ完了するごとに短くなっていく．また, ある程度タスクが進むと, アイコンに描画される顔, 音, バーの色を変化させる．

両フェーズにおいて, タスクの条件は 40 通り (45 度毎に 8 方向 × 5 条件の距離) である．ターゲットアイ



図 6: ターゲットアイコン

コンのサイズは 50×50 ピクセルであり、 1280×1024 ピクセルのディスプレイ領域に 100 ピクセル間隔で 13×10 の格子状に配置してあるターゲット候補から、タスク毎に 1 つ選ばれ表示される。ディスプレイの上方向を 0 度とし、 $0, 90, 180, 270$ 度方向のタスクでは、 $500, 600, 700, 800, 900$ ピクセルの距離条件を用意し、 $45, 135, 225, 315$ 度方向のタスクでは、絶対距離が等しくなるように、 $707, 848, 989, 1131, 1272$ ピクセルの距離条件を用意する。被験者がターゲットアイコンを 1 回クリックすると、一つのタスクが完了する。被験者がターゲットをクリックすると、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。次のタスクでは、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新たなスタート位置となるようにする。この順を考慮し、かつ被験者に規則性を悟られずにランダム順にしたのと同じ条件となるよう、また各タスク条件を等しい回数実行したのと同じ条件となるように、図 7 に示す順序で行う。図 7 では、一連の順序を理解しやすいように (a) ~ (f) の 6 つの図に分けて示してある。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行くことを示す。また、図 7(a) ~ (f) の図のつながりが理解しやすいように、一つ手前の図を破線で示してある。図 7(a) ~ (f) に示すタスクを一巡すれば、 40 通りのタスク条件が 1 回ずつ実行されたことになる。実際には図 7 に示した実行順序で各タスク条件を一巡したあと、次の一巡は逆順で行い、これを交互に繰り返す。

5.4 予備実験

平均年齢 28.2 歳の男性 5 名と 27 歳の女性 1 名に対して予備実験を行った。Calibration フェーズと Prediction フェーズをそれぞれ練習パートと本番パートにわけ、2 つのフェーズの間には 5 分間の休憩を設けた。各パートのタスク数は Calibration フェーズでは練習パート 40 回、本番パート 640 回、Prediction フェーズでは練習パート 120 回、本番パート 300 回

である。

予備実験を行った結果、連続で行うには 1 つのフェーズが長いという意見が被験者から出たため、本実験では本番パートを等しい回数の 2 つに分け、各パート間に 3 分間の休憩時間を設けることとした。また、予備実験を行う前は Calibration フェーズと別に Non-prediction フェーズを設け、Prediction フェーズと同回数のタスクを行うことを考えていたが、被験者の負担が大きくなると考えられたため、Calibration フェーズを Non-prediction フェーズとして兼用することとし、Prediction フェーズのタスク数を Calibration フェーズと同回数に揃えることとした。ただし、Calibration フェーズでは、ターゲット以外の場所を誤ってクリックするか、移動の途中で 100ms 以上カーソルが停止した場合に、そのタスクのスタート位置にカーソルを戻し、同じタスクをやり直しさせるという条件が加わり、その点が Prediction フェーズと異なる。また、Calibration フェーズを先に行うため、Prediction フェーズを行う時に被験者に疲れが残る可能性もある。しかし、これらは Non-prediction フェーズで得られるデータにとって有利な条件となるため、この条件の下で Prediction フェーズから得られたデータに優位性が示されれば、より有効な結果になると考えられる。

5.5 本実験

平均年齢 23.0 歳の男性 8 名と平均年齢 25.8 歳の女性 8 名に対して本実験を行った。全員、コンピュータを使用した経験はあるが、その程度にはばらつきがあった。本実験は、Calibration フェーズと Prediction フェーズをそれぞれ練習パート、本番パート 1 、本番パート 2 に分けて行った。各パートのタスク回数はそれぞれ 120 回、 320 回、 320 回とし、各パート間には 3 分間の休憩を設け、Calibration フェーズと Prediction フェーズの間には 5 分間の休憩を設けた。 40 通りの各タスク条件に対して、Calibration フェーズと Prediction フェーズのそれぞれの本番パートで 16 回 ($320/40 + 320/40$) 行わせた。

6 実験結果

予測の有無と、ターゲット距離を要因とする二元配置の ANOVA を行ったところ、予測の有無 ($F(1,$

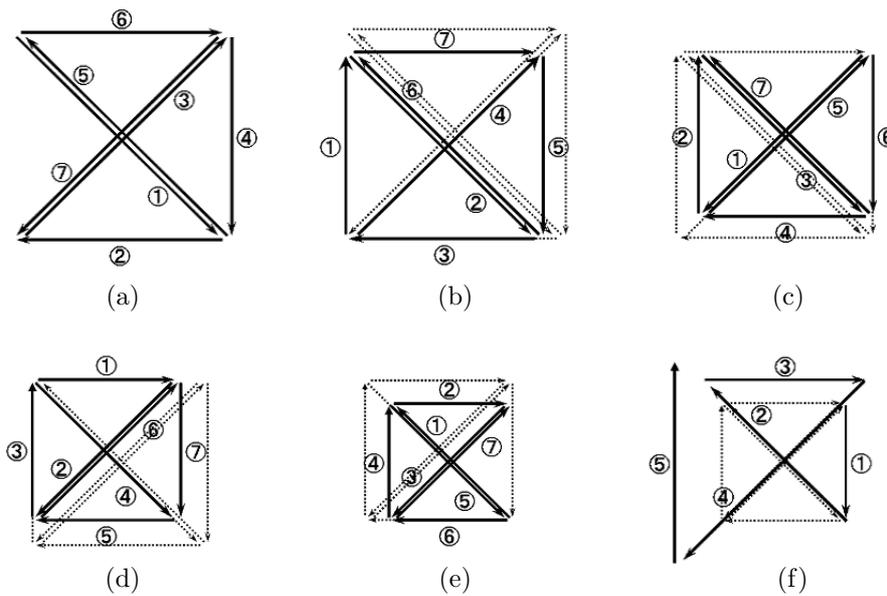


図 7: タスクの実行順序

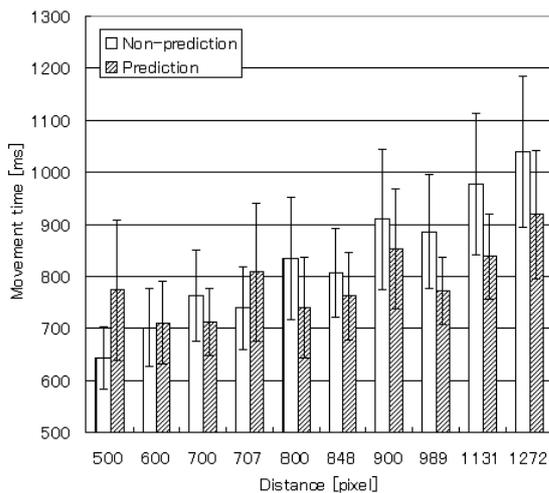


図 8: ターゲット距離に対する移動時間

300) = 12.593, $p < .001$), ターゲット距離 ($F(9, 300) = 23.893, p < .001$) 及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移動時間を図 8 に示す. 図では標準偏差を重ねて表示している. 図 8 から, 500, 600, 707 ピクセルといった短いターゲット距離においては, Prediction フェーズの移動時間が Non-prediction フェーズの移動時間を上回っていることがわかる.

予測が有効に働いていると言える距離のしきい値

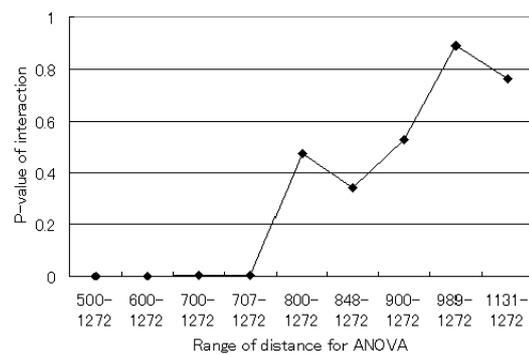


図 9: 予測の有無とターゲット距離の交互作用に対する p 値

を調べるために, 予測の有無とターゲット距離を要因とする二元配置の ANOVA を, ターゲット距離の範囲を変更して行った. ターゲット距離の範囲を変更して行った ANOVA ごとく予測の有無とターゲット距離の交互作用に対する p 値を図 9 に示す. 図 9 からわかるとおり, ターゲット距離を 800 ピクセル以上の範囲に限定した時, 予測の有無とターゲット距離の交互作用 ($p > .30$) で有意差が認められなくなっており, この範囲では予測の有無のみにより有意差があることがわかる.

本実験での各タスク条件に対応する難易度 ID を表 1 に示し, ID に対する移動時間の関係を図 10 に示す. ただし, W, H, D はそれぞれターゲット幅, ターゲット高さ, ターゲット距離であり, ID は値が大きいほ

表 1: 各タスク条件に対する ID

D(pixel)	W, H(pixel)	ID(bit)
500	50	3.46
600	50	3.70
700	50	3.91
707	50	3.92
800	50	4.09
848	50	4.17
900	50	4.25
989	50	4.38
1131	50	4.56
1272	50	4.72

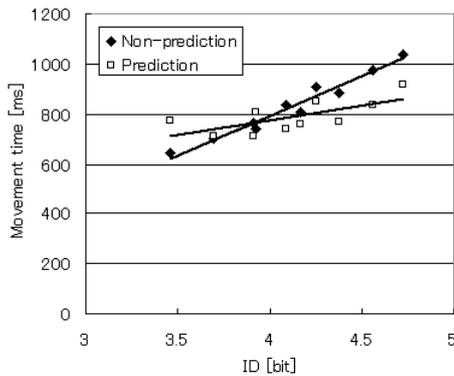


図 10: ID に対する移動時間

どタスクが難しいことを示す。これらから、3.92 bits よりも高い ID のタスクにおいて、Non-prediction よりも Prediction の方が有利であることがわかる。なお、ID は二次元のポインティングタスクのモデルとして提案されている式 (2)[11] を用いて算出した。

$$ID = \log_2 \left(\frac{D}{\min(W, H)} + 1 \right) \quad (2)$$

Prediction フェーズにおいて、タスクのスタート位置からジャンプ元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を算出すると、これはそのタスクにおいて、実際にカーソル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 Gain として図 11 に示す。なお、Gain の平均は 1.21 であった。

Prediction フェーズにおいて、ターゲット方向と、ピーク速度を検出した時点で予測した移動方向との誤差による移動時間の違いを図 12 に示す。この図から、誤差が大きくなるほど移動時間が増加すること

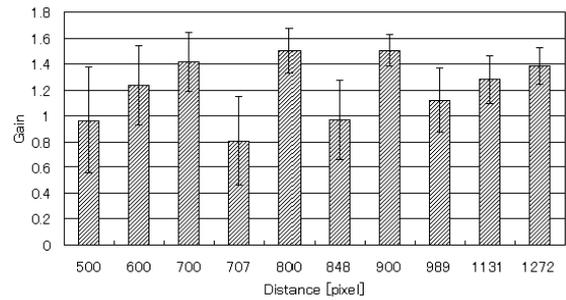


図 11: ターゲット距離に対する Gain

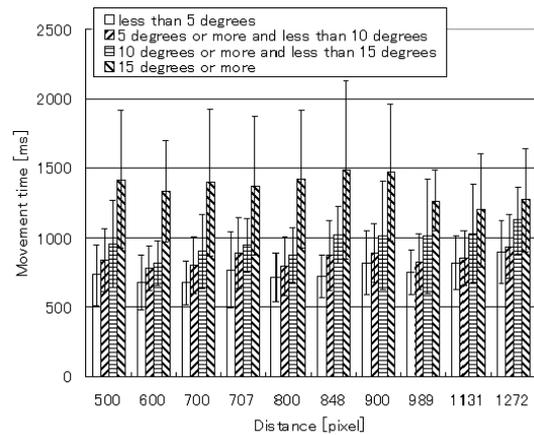


図 12: ターゲット方向と予測した移動方向の誤差による移動時間の違い

がわかる。当該誤差が 5 度未満、5 度以上 10 度未満、10 度以上 15 度未満、15 度以上であったタスクの割合はそれぞれ 78.5%、17.6%、2.2%、1.7%であった。

7 考察

実験より得られた結果について検討する。図 8 と図 9 より、提案手法を用いて予測したターゲットへカーソルをジャンプさせる手法は、ターゲット距離が 800 ピクセル以上の場合に有効である。800 ピクセルよりも短い距離で本手法が有効に働いていないのは、カーソルをジャンプさせる際に、ジャンプそのものの時間とその後被験者がジャンプしたことを認識する時間が必要であり、短い距離の場合はその時間の割合の影響が大きくなるためと考えられる。また、図 10 において、ID に対する移動時間の変化の傾きが本手法を用いた方が緩やかになっていることから、ターゲット距離が長くなるほど、本手法はより有効になると考えられる。

図 11 において、500, 600, 700, 800, 900 ピクセル

の垂直・水平方向のタスクと, 707, 848, 989, 1131, 1272 ピクセルの斜め方向のタスクに分けて考えると, 斜め方向のタスクの方が Gain が小さいが, これはターゲット方向と実際にジャンプした方向との誤差が斜め方向の方が大きくなり, それによって合計のカーソル移動距離が伸びたためと思われる. また, 垂直・水平方向と斜め方向のそれぞれのタスクにおいて, ターゲット距離が長くなるほど Gain が増加しているが, これはターゲット距離が長くなるほどジャンプによってカバーする距離が増加し, ターゲット方向とジャンプした方向との誤差によって増加する移動距離の影響が小さくなったためと思われる. また, Gain の平均が 1.21 と 1 より大きいため, 本手法によって被験者の操作の負担が軽減できていると考えられる.

本手法はポインティング時間を短縮する手法として, ターゲットを拡大する手法 [1, 2] や, C-D 比を変更する手法 [4] と競合するものではないと思われるため, 本手法でターゲットを予測した後に, 予測したターゲット周辺でこれらの手法を利用すれば, さらにポインティング時間を短縮できる可能性がある.

8 おわりに

本稿では, GUI 環境でカーソル移動中のピーク速度からターゲットを予測する手法を提案し, 実験用に本手法を用いたインタフェースを試作し, 有効性を評価, 検討した. 実験結果から, 本手法はターゲット距離が 800 ピクセル以上であれば有効であり, ポインティング時間を短縮できることを確認した. また, 本手法によってポインティング時に必要な移動距離が短縮され, ユーザの操作の負担を軽減できることを確認した.

今後の課題としては, ユーザの移動方向を予測する手法としてより優れた手法を提案することや, 異なる方向に並んだオブジェクトからターゲットを予測する手法 [12] などを利用することが考えられる. また, 実際に普段使用しているデスクトップ環境に実装しての評価や, マウスドライバへの搭載なども考えられる.

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) の研究助成と科学研究費補助金基盤研究 (B)(2)13480104 による.

参考文献

- [1] M. McGuffin and R. Balakrishnan. "Acquisition of expanding targets," In Proc. CHI 2002, pp. 57-64.
- [2] S. Zhai, S. Conversy, M. Beaudouin-Lafon and Y. Guiard. "Human on-line response to target expansion," In Proc. of CHI 2003, pp. 177-284.
- [3] Y. Guiard, R. Blanch and M. Beaudouin-Lafon. "Object pointing: a complement to bitmap pointing in GUIs," In Proc. of Graphics Interface, pp. 9-16, 2004.
- [4] R. Blanch, Y. Guiard and M. Beaudouin-Lafon. "Semantic pointing: improving target acquisition with control-display ratio adaptation," In Proc. of CHI 2004, pp. 519-526.
- [5] R. Woodworth. "The accuracy of voluntary movement," Psychological Review Monograph Supplement, vol. 3, no. 13, pp. 1-114, 1899.
- [6] N. Walker, D.E. Meyer, and J. B. Smelcer. "Spatial and temporal characteristics of rapid cursor-positioning movements with electromechanical mice in human-computer interaction," Human Factors, 35-3, pp. 431-458, 1993.
- [7] 高木 龍, 北村 喜文, 正城敏博, 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.
- [8] 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 2001 年総合大会講演論文集, A-15-17, p. 316, 2001.
- [9] Logitech: "<http://www.logitech.com/>"
- [10] Power Support: "<http://www.pawasapo.co.jp/>"
- [11] I. S. MacKenzie and W. Buxton. "Extending Fitts' law to two-dimensional tasks," In Proc. of SIGCHI, pp. 219-226, 1992.
- [12] A. Murata. "Improvement of performance by method for predicting targets in pointing by mouse," IEICE Transactions Fundamentals, vol. E78-A, no. 11, pp. 1537-1541, 1995.