

# ポインティング動作を用いたスクロール技術 DuH の提案

高嶋 和毅<sup>†</sup> スリラム サブラマニアン<sup>††</sup>, 築谷 喬之<sup>†</sup>  
北村 喜文<sup>†</sup> 岸野 文郎<sup>†</sup>

画面外のオブジェクトを効率よく獲得できるインタラクション技術 DuH (Delphian Desktop using Halo) を提案する。DuH では、画面外のオブジェクトまでの距離を把握するために距離の手がかりとして Halo を使い、自動的にスクロールするために Delphian Desktop の予測アルゴリズムを用いる。まず、Halo の距離の手がかりとしての効果をポインティング運動特性によって評価を行った。その結果、Halo によって推定されるターゲットへのポインティングの場合も、通常と同様にピーク速度とターゲット距離との間に線形の関係があることを見だし、ピーク速度による予測スクロール技術の可能性を確認した。次に DuH の評価実験では、DuH と Hop と Pan with Halo の三種類のインタラクション技術の比較を行い、タスク完了時間において、DuH が Hop や Pan with Halo より有効な地図ナビゲーション技術であることを確認した。

## Proposal of a Scroll Technique DuH Using Pointing Interaction

KAZUKI TAKASHIMA,<sup>†</sup> SRIRAM SUBRAMANIAN,<sup>††</sup>  
TAKAYUKI TSUKITANI,<sup>†</sup> YOSHIFUMI KITAMURA<sup>†</sup>  
and FUMIO KISHINO<sup>†</sup>

We conducted experiment to investigate the Halo effect as a cue of target distance by measuring the kinematics features during pointing movement toward off-screen target. The result shows that linear relationship between off-screen target distance and peak velocity. We propose the DuH (Delphian Desktop using Halo) based on the result for map navigation. Curvature of Halo is used to estimate the off-screen target distance and prediction algorithm of Delphian Desktop is used to scroll to target location automatically. We also conducted experiment to evaluate DuH by comparing with Hop and Pan with Halo. As a result of user study, DuH is more effective technique than others for reducing task completion time.

### 1. はじめに

GPS システムが発展し、Google Earth のような地図ナビゲーションを使う機会が増えている。このような地図は、ほとんど無限に近い広がりを持つが、ディスプレイやモバイル端末など、小さい画面を用いて観察することが多く、大局と詳細を同時に把握することは困難である。また、一般的に、スクロールバーやパン、そしてズームといったインタラクションが利用されているが、画面内のオブジェクトへのアクセスに比べて、画面外のオブジェクトを得るための操作は困難で、時間もかかる。それを改善するために、画面外の

オブジェクトの位置などの情報を画面内に可視化し、地図の詳細と同時に画面外の様子も把握できる技術が提案されている<sup>3),9)</sup>。また、効率よく画面外オブジェクトを獲得するために、スクロールを支援する技術の研究も盛んで、ズームとスクロールを組み合わせた技術<sup>8)</sup>などが提案されている。

一方、スクロールのタスクは、到達運動を行ってターゲットを獲得するという点で、ポインティングと似たタスクと考えることができる<sup>7)</sup>。したがって、スクロール距離が短い場合には、スクロール用に特化されたインタラクションに切り替えるのではなく、ポインティングという使いなれたインタラクションを用いて、画面外オブジェクトを獲得することができれば、より効率的に画面外の領域を扱うことができると考えられる。

そこで我々は、ユーザに画面外のターゲットまでの距離を把握させる Halo<sup>3)</sup> という可視化技術と、Delphian Desktop<sup>2)</sup> というポインティング中のピーク速度を用いたターゲット距離の予測技術を組み合わせ

<sup>†</sup> 大阪大学大学院情報科学研究科  
Graduate School of Information Science and Technology, Osaka University

<sup>††</sup> サスカチュワン大学  
University of Saskatchewan  
現在、フィリップス リサーチ  
Presently with Philips Research

た、予測型スクロール技術 DuH (Delphian Desktop using Halo) を提案する。DuH は、Halo によって推定される画面外オブジェクトへのポインティング中のピーク速度を用いてターゲット距離を予測し、予測地点まで自動的にスクロールを行うことで、馴染み深いインタラクション動作による画面外オブジェクトの獲得を実現する技術である。本論文では、DuH の予測モデルとなる Halo を用いた場合のポインティング運動特性について実験した結果を述べ、その後、提案する DuH の概要とその評価結果について述べる。

## 2. 関連研究

### 2.1 ポインティング

GUI 環境におけるポインティング動作は古くから研究されており、ポインティングに要する時間はフィッツの法則に従うことや<sup>5)</sup>、ポインティング中の速度波形は、図 1 に示すようなベル型波形を描き、計画時間と調整時間の二つに分けられることが知られている<sup>4)</sup>。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、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二つの運動の境界については多くの議論があるが<sup>4),11)</sup>、ピーク速度以前の運動を特徴づける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るという結果が報告されている<sup>10)</sup>。さらに、速度波形中のピーク速度以前の運動は、ポインティング動作を始める前に計画されていることを示した研究例もあり、ピーク速度が計画運動の代表値であることは明らかになっている<sup>6)</sup>。また、これらを基にした研究では、ピーク速度とターゲット距離は線形的な関係にあるという知見が得られており、この関係を用いて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し、カーソルを自動的に予測地点にジャンプさせる予測インタフェース Delphian Desktop が提案されている<sup>2)</sup>。

### 2.2 可視化

Baudisch らは、画面外のオブジェクトの情報を画面内に提示する手法として、Halo を提案している<sup>3)</sup>。この技術では、画面外オブジェクトはその周囲に円を持ち、その半径はそのオブジェクトから現在のウィンドウ画面端までの距離である。したがって、作業ウィンドウの端にはその円の一部である弧が架かって見えることになり、ユーザはその弧の形状から画面外オブジェクトの距離を見積もることができる。また、Halo の弧の位置と画面外オブジェクトの方向もある程度一

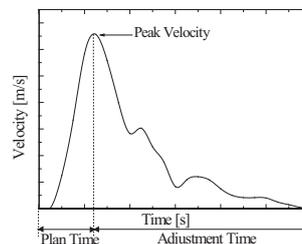


図 1 ポインティング中の速度波形

Fig. 1 Velocity profile during pointing task

致しており、空間把握にも優れているため、Pan 動作を補助するだけでなく、新たに提案されている地図ナビゲーションでも用いられている<sup>9)</sup>。しかし、絶対的な距離の手がかりとしては詳細に検討されていなかったため、その効果は不明である。

### 2.3 ナビゲーション・スクロール

Igarashi らは、高速なスクロール時に、画面が不鮮明になることを防ぐ、Speed-dependent automatic zooming (SDAZ) という技術を提案している<sup>8)</sup>。この技術の基本アルゴリズムはスクロールスピード × スケール = constant の関係を用いており、スクロールスピードに応じて画面のスケールが変化し、高速なスクロール時には自動的にズームアウトを行い、地図やドキュメントの全体像を把握しやすくする。Appert らは、1次元のドキュメントスクロールに関して、カーソルを水平方向に動かすとズームを行い、画面上下方向に動かすことでスクロールを行う技術を提案している。この技術は非常に効率がよく、単純なターゲット獲得タスクでは、SDAZ より高速になると報告されている<sup>1)</sup>。また、カーソルを円形に巡回運動させることでドキュメントをスクロールする技術も提案されている<sup>12)</sup>。これは商用的にも広く利用されており、スクロール距離が長いほど、手やマウスを置き換えることなく効率よくスクロールを行えることが利点である。

ズームングを基にした技術も提案されており、Zone-zoom は、地図を 9 つの領域に格子状に分割し、携帯電話などで用いられる 1-9 のテンキーをそれぞれの領域に対応づけ、1-9 のボタンを on/off することで、対応する領域が自動的にズームイン/アウトされる。これは、PC のキーボードだけでなく、多くのモバイル端末においても利用可能な技術である<sup>13)</sup>。

Irani は、Halo とマウス操作によるレーザービームとプロキシ、そしてテレポートを組み合わせた地図ナビゲーション技術 Hop を提案している<sup>9)</sup>。初期画面として、画面外オブジェクトが画面内に Halo によって可視化されている状況がユーザに与えられる。ユーザは

画面上の任意の点を中心とする円状のカーソルの動きを用いてレーザービームを操作し、画面外のオブジェクトのコピーであるプロキシをカーソル付近に生成する。次に、カーソル付近に集まったプロキシ群の中から目的地のプロキシをクリックすると、アニメーションとともに目的地へ自動的にスクロールされる。Iraniらの実験では、地図ナビゲーションにおいて、ターゲット探索という動作を含んだタスクを設定しており、単純にパンやズームを繰り返すより、Halo とプロキシという可視化技術を利用し、画面外の状況を把握させることで、ターゲット獲得時間は短縮されることが示されている。

### 3. 実験 1: Halo を用いたポインティング特性

#### 3.1 概要

本章では、Halo の絶対的な距離の手がかりの効果をポインティング中の運動特性を用いて評価する。通常のポインティングでは、ポインティング中のピーク速度はターゲット距離との間に線形的な関係が見られるため<sup>2)</sup>、Halo によって推定した画面外オブジェクトへ向けたポインティングにおいても、同様の関係が見られた場合、人は Halo によってオブジェクトまでの距離を推定できるとみなすことができ、Delphian Desktop の予測アルゴリズムの適用が可能となる。

#### 3.2 実験装置

図 2 に実験装置を示す。ディスプレイは 50 インチのプラズマディスプレイ (110×62 cm, 1280×768 pixels) を用い、サウンドフィールドバック用にディスプレイ両脇にステレオスピーカを設置する。カーソルはマウスパッド上 (Power Support's AirPad Pro 3 AK-07, 408×306 mm) で、光学式マウス (Microsoft IntelliMouse1.1A) を用いて操作する。被験者は机に向かって正対し、肘の高さが机の高さと同じになるように椅子の高さを調節して座る。マウスの C-D 比は実験を通して一定とする (Control:Display = 1:3.3)。

図 2 に示すように、PDP 内に表示された灰色の部分は全て画面外を意味しており、中央の白色の四角形は画面内を意味する。実際の PDP の解像度は 1280×768 pixels であるが、5 章で述べる DuH の評価で用いる実験環境との統制のため、三倍の解像度の 3840×2304 pixels と仮定し、本実験でのターゲット距離の値はこの仮定に基づいたものを用いる。

#### 3.3 タスク

被験者は、ディスプレイに表示されたターゲットへ向けてマウスを用いてカーソルを操作し、できるだけ早くかつ正確にターゲットをクリックする。カーソル



図 2 実験装置  
Fig. 2 Experimental setup

は画面内 (白色領域) の中心に固定されており、一度クリックをするとカーソルは動作可能となる。次節で述べるように、実験の条件によってはターゲットやカーソルが不可視の場合も存在するが、ターゲットは常に一つである。被験者がターゲットを正確にクリックした場合は正解音出力され、外れた場合でも 100 pixels 以内の距離であれば「許容範囲」を意味する音出力される。それ以外の場合は失敗音出力される。被験者が精度のみに注力することを防ぐため、正解音と不正解音は、報酬の意味が薄いものである。一つのタスクが終われば、実験順序に従って新しいターゲットが自動的に表示され、カーソルは自動的にスタート位置に固定される。

#### 3.4 実験計画

実験要因として、ポインティング条件 (Normal, Full Halo, Partial Halo)、ターゲット距離 (500, 700, 900, 1100 pixels)、ターゲット角度 (0, 45, 90, 135, 180, 225, 270, 315 deg) を用いる。角度は水平右向きを 0 deg とし、半時計回りを正方向とする。全組み合わせは 96 通りになり、被験者は各条件を 10 回反復を行う。192 試行ごとに 1 分間の休憩をはさみ、計 960 回の試行を行う。タスク順序は完全なランダムを用い、本実験前の 100 回程度の練習も含めて、40 分程度を要する実験である。被験者は、右利きで GUI を頻繁に利用する 8 人 (男: 6, 女: 2, 22-36 歳) である。

##### 3.4.1 ポインティング条件

図 3 は、三種類のポインティング条件において、右斜め上に表示されたターゲットへのポインティング動作を示している。

**Normal:** ターゲットは常に可視で、画面外 (灰色) にカーソルが達してもカーソルは可視 (図 3(a))。

**Full Halo:** ターゲットは常に可視で、ターゲットが持つ Halo の全体の形状も可視。画面 (灰色) にカーソルが達した以降はカーソルは不可視となる (図 3(b))。Halo とターゲットまでの距離をユーザに学習させるために用いた条件。

**Partial Halo:** ターゲットは常に不可視で、ターゲッ

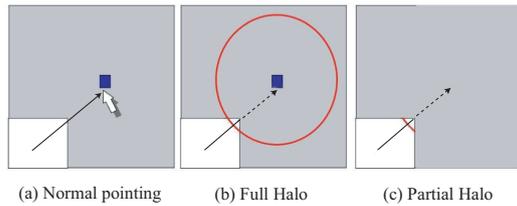


図3 ポインティング条件  
Fig. 3 Pointing condition

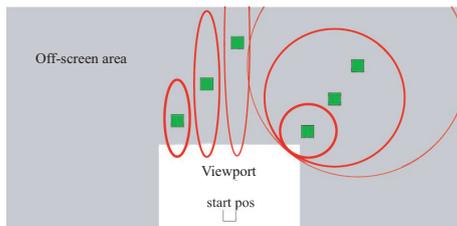


図4 Halo の設定  
Fig. 4 Halo setting

トが持つ Halo は、画面内（白色）の端にだけ表示され、弧のみの可視。画面外（灰色）にカーソルが達した以降はカーソルは不可視となる（図 3(c)）。実際の Halo を用いる場面に近い条件。

### 3.4.2 Halo 設定

図 4 に示すように、水平・鉛直方向へ向けてポインティングを行うタスクでは楕円の Halo を用い、対角へ向けて（コーナー）のタスク時は円形の Halo を用いる。この設定は Irani らが用いたものと同様である<sup>9)</sup>。また、Halo の線の太さは、ターゲット距離が遠くなる（円の大きさが大きくなる）につれて細くし、ターゲット距離の手がかりになるように設定する。

### 3.5 結果

3 要因分散分析の結果、ターゲット距離 ( $F(3, 21) = 37.667, p < .001$ ) と、ターゲット角度 ( $F(7, 49) = 9.608, p < .001$ ) に主効果が見られた。しかし、ポインティング条件 ( $F(2, 14) = 1.017, p > .05$ ) には有意差は見られなかった。多重比較の結果、ターゲット距離に関しては全ての水準間 (500, 700, 900, 1100 pixels) に有意差が見られた。また、ターゲット角度に関しては、水平、垂直、対角のタスク間で有意差が見られた。図 5 にピーク速度の平均値とターゲット距離の関係を示す。ターゲットが不可視で、Halo のみが可視の Partial Halo の条件でも、Normal 条件と同様にターゲット距離とピーク速度の関係はほぼ線形であることがわかる。図 6 には、ターゲット角度とピーク速度の関係を示す。この結果からも、ピーク速度はポインティング条件には依存せず、ターゲット角度に

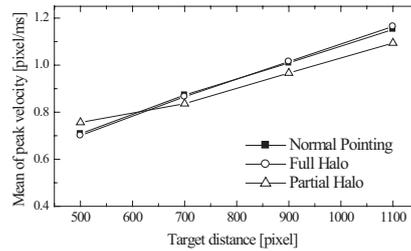


図5 ターゲット距離 vs. ピーク速度  
Fig. 5 Target distance vs. peak velocity

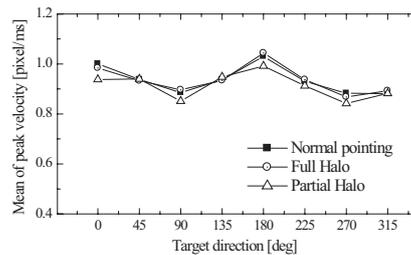


図6 ターゲット角度 vs. ピーク速度  
Fig. 6 Target direction vs. peak velocity

よって大きく変動し、水平タスクほど大きくなること がわかる。これらは先行研究と同様の結果である<sup>2)</sup>。

### 3.6 議論

ターゲット距離とピーク速度の線形的な関係が見られたという結果から、Halo はユーザに有効なターゲット距離に関する手がかりを与えることができると考えられる。しかし、ブロック間の差（学習の効果）は見られなかったものの、Halo の理解には、かなりの練習が必要である被験者も見られた。また、ターゲット距離が最も短く、Halo の形状から距離が推定しやすい 500 pixels と、最も距離が長く、Halo の線も細い 1100 pixels では、Partial Halo の結果とその他二条件の結果の間に差があり、ピーク速度は、ターゲット距離推定の難易度に依存する様子も見られている。しかし、全体の傾向として、Delphian Desktop で用いているピーク速度-ターゲット距離の線形性が、Halo を用いたポインティング環境でも得られたことで、画面外へのポインティングにおいてもピーク速度を用いた予測アルゴリズムが適用可能であると考えられる。

### 4. DuH

画面外オブジェクトをポインティング動作によって獲得するために、Delphian Desktop の技術をスクロール動作に適合させた DuH (Delphian Desktop using Halo) を提案する。Delphian Desktop と同様に、システムを利用する前にキャリブレーションが必要であり、前章の実験 1 で述べたように、8 方向に関して Par-

tial Halo の条件で得られた予測モデルを用意しておく。DuH では、ポインティング中に検出されたピーク速度を用いてターゲットまでの距離を予測し、滑らかなアニメーションで目的値まで自動スクロール（ジャンプ）を行う。図 7 は、ユーザに与えられる初期画面（Halo 内の小さなアイコンは次章の実験 2 で用いるプロキシ）で、画面外の情報が Halo によって画面内に可視化されている。ユーザは、その弧の形状からターゲット位置を推定し、推定したターゲットに向けてポインティング動作を開始する。カーソル速度がピークに達した場合、その時のカーソル移動方向をユーザの意図するポインティング方向とし、キャリブレーションにより得られている 8 方向の回帰式の中から最も適切な式を選び、ターゲット距離を算出する。予測完了後、自動的に予測距離に応じてスクロールが行われ、カーソルも図 8 に示すようなアニメーションとともに予測地点に移動される。

実験 1 で得られた結果の分析において、ピーク速度は、ターゲットとスタート位置のほぼ中間で検出されることが多かった。DuH の予測アルゴリズムはカーソルが画面端に達する前にピーク速度を検出する必要があるため、8 方向を考慮すると、作業ウィンドウの大きさの三倍程度の大きさの地図への適用が限界である。現在の実装では、ピーク速度を検出する前に、カーソルが作業画面端に達してしまった場合は、その地点のカーソル速度をピーク速度と見なして予測する。

## 5. 実験 2: DuH の評価実験

### 5.1 概要

前章で提案した DuH と、関連研究で述べた Hop、そして一般的によく用いられる Pan に Halo の可視化技術を適用したものの、3 種類のインタラクション技術を比較する。被験者には、地図ナビゲーションの環境での二種類のタスクを課す。タスク 1 は、インタラクション技術を用いて、画面外のオブジェクト群の中からターゲットを一つ探して獲得し、クリックするものである。その後に行うタスク 2 では、別のタブレット PC に表示された全体の地図の中から、タスク 1 でクリックしたオブジェクトをスタイラスを用いて選択する。DuH や Hop では自動スクロールが行われるため、ユーザの地図全体に対するターゲット位置把握に影響を与える恐れがあり、このタスク 2 にてその調査を行う。画面外のオブジェクトの距離や位置などの情報は画面内に可視化された Halo によって示され、そのオブジェクトの属性、たとえばレストランや駅などの情報はプロキシ（オブジェクトのコピー）によって

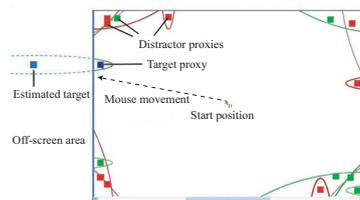


図 7 初期画面

Fig. 7 Initial display

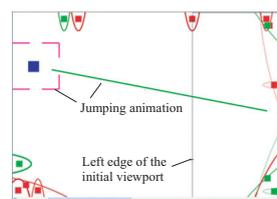


図 8 DuH のスクロールアニメーションの様子

Fig. 8 Scrolling animation of DuH

示される。このプロキシは基本的に Halo 内に表示されるが、オブジェクト数が多い時などは、できるだけ重ならないように配置される。

### 5.2 実験計画

実験要因は、インタラクション技術（DuH, Hop, Pan with Halo）と、ターゲット距離（700, 900, 1100 pixels, 全て画面外）、オブジェクト密度（16, 32, 48 objects）を用いる。全組み合わせは 27 通りで、1 タスクあたり 16 回反復する（8 方向 × 2 反復）。被験者は、途中で 2 分間の休憩を 3 回挟み、計 432 回の試行を行う。実験時間は、本実験前の練習を含めて 90 分程度を要する。練習は、各インタラクション技術の扱いに慣れるまでで、50 回を限度に行う。ターゲット距離の条件は 3 章の結果に基づいて決定し、ターゲット以外のオブジェクトは、地図を 9 つの格子に分割し、中心を除いた 8 つの領域それぞれに均等にランダムに配置させる。例えば、オブジェクト密度 16 の場合は、それぞれの領域に 2 つのターゲットではないオブジェクトが配置されることになる。タスク順序は、インタラクション技術の条件は被験者間でランダムとし、その他は完全にランダムとする。

### 5.3 インタラクション技術

簡単のため、地図上のランドマークは全て赤、緑、青のいずれかで塗りつぶされており、ターゲットは青でそれ以外は赤と緑である。Halo 内に表示されるプロキシもターゲットと同様の表示で、Halo の形状は 3 章の実験環境と同様のものを用いる。また、各インタラクション技術を対等に比較するために、ターゲット獲得の難易度  $ID$  をフィッツの法則<sup>5)</sup> と Hinckley

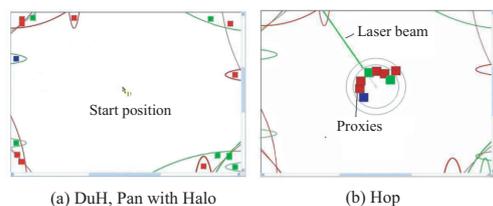


図 9 Halo 内のプロキシと Hop によるプロキシと  
Fig.9 Proxy in Halo and proxy by Hop

のスクロール時間のモデル<sup>7)</sup>を用いて統制している。ターゲットが可視状態でのポインティング時の  $ID$  は、フィッツの法則に従い、ターゲットの距離と幅によって算出する。また、スクロールを行ってターゲットを獲得するタスクでは、ターゲット距離と、画面内のどの範囲にターゲットを含めればタスクを完了とするかの許容範囲の大きさをターゲット幅として用い、難易度  $ID$  を算出する。

**DuH** 初期画面を図 9(a) に示す。画面端に表示されたプロキシの中からターゲットを探索し、そのプロキシの Halo の弧の状況からターゲットまでの距離を見積もる。その後、前章で述べた DuH のアルゴリズムを用いて自動スクロールを行う。予測地点を算出した後、予測地点とターゲットとの距離が 150 pixels 以内 (300 pixels の四角形が許容範囲) であれば、ターゲットを予測地点と見なす調整を行い、カーソルをターゲット上に移動させる。それ以外の場合は調整は行わない。スクロール後、ターゲットがウィンドウ内に表示されていれば、DuH を作動させずにターゲットをクリックし、タスクを完了させる。ターゲットが画面外にある場合は、再度 DuH を作動させてスクロールを行う。タスクの難易度  $ID$  は Hinckley のモデルを用いて、1.72-2.20 bits に設定している<sup>7)</sup>。

**Hop** 初期画面は画面端に表示された Halo の弧のみであり、ユーザは任意の点で左クリックを押した状態で円状にカーソルを移動させる。カーソルの動きに合わせてレーザービームが動作し、レーザーの先端と Halo が交差した場合に、図 9(b) に示すように、その Halo を持つオブジェクトのプロキシがカーソル付近に生成される。この作業を続けてプロキシを集めた後、カーソル付近のプロキシ群からターゲットを探索し、目的のプロキシをクリックする。その後、自動的にターゲット位置までスクロールが行われ、カーソルがターゲット上に固定される。プロキシをポインティングする際の難易度  $ID$  はフィッツの法則に従い、1.80-2.20 bits になるように設定している<sup>5)</sup>。

**Pan with Halo** 初期画面は DuH と同様で、画面端のプロキシと Halo からターゲットを探索した後、

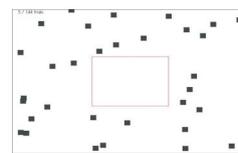


図 10 空間把握テストの表示  
Fig.10 Display of spatial recognition test

マウスのドラッグ-シフト-リリース (パン) を繰り返す。ターゲットがウィンドウの中心付近に来るまでパン動作を繰り返す。この条件では、Hinckley のモデルによる難易度  $ID$  を DuH や Hop と同じにするため、一辺 300 pixels の点線で描かれた四角形がウィンドウ中心に表示されており、この四角形を許容範囲 (ターゲット幅) とし、この中にターゲットが入った状態でターゲットをクリックするとタスクが完了する。

DuH と Hop の自動スクロールの時間は 1000 ms と常に一定値を用いている。

#### 5.4 実験装置と被験者

実験装置は 3 章のものと同様であるが、ディスプレイは 15 インチ LCD (1024×768 pixels) に変更し、タスク 2 の空間把握テストのためにタブレット PC を正面のディスプレイ脇に配置する。マウスの C-D 比も同様である。被験者は 12 人 (男: 6, 女: 4, 22-36 歳) で、実験 1 に参加した 8 人を含み、その他の 4 人は、予測の回帰モデルを得るために、実験 1 の反復回数を減らした簡易的な DuH のキャリブレーション後に本実験に入る。全ての被験者は GUI 環境でのマップナビゲーションの経験者であった。

#### 5.5 手続き

この実験は二つのタスクから構成され、タスク 1 は上で述べたインタラクション技術を用いて提示されたターゲットを獲得するもので、主にタスク完了時間を測定する。タスク 2 では、スクロール後の空間的な位置把握を測定する。タスク 1 が終わると画面がクリアされ、その後、手元にあるタブレット PC に表示されている全体の地図 (図 10) から、タスク 1 でクリックしたオブジェクトをスタイラスを用いて選択する。これが終了し、マウスの左ボタンを押すと 15 インチ LCD に新たな実験画面が表示され、カーソルが画面の中心に置かれた状態で次のタスクが開始される。タスク 1 ではポインティングの正否に関して、音のフィードバックが出力されるが、タスク 2 では出力されない。

#### 5.6 結果

##### 5.6.1 タスク完了時間

三要因分散分析の結果、インタラクション技術 ( $F(2,22) = 5.8583, p < .001$ )、ターゲット距離

## ポインティング動作を用いたスクロール技術 DuH の提案

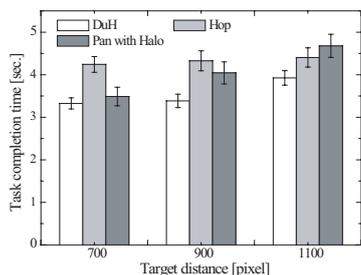


図 11 ターゲット距離 vs. タスク完了時間

Fig. 11 Target distance vs. task completion time

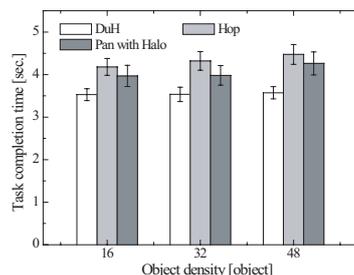


図 12 オブジェクト密度 vs. タスク完了時間

Fig. 12 Object density vs. task completion time

( $F(2,22) = 130.311, p < .001$ ), オブジェクト密度 ( $F(2,22) = 11.389, p < .001$ ), インタラクション技術 × ターゲット距離 ( $F(4,44) = 21.721, p < .001$ ), インタラクション技術 × オブジェクト密度 ( $F(4,44) = 2.958, p < .001$ ) の交互作用に関して有意差が認められた．図 11 にターゲット距離とタスク完了時間を示し，図 12 にオブジェクト密度に対するタスク完了時間を示す．図 11 に関する多重比較の結果，DuH が最も早い結果であったが，ターゲット距離 700 pixels では，DuH と Hop の間には差があり，DuH と Pan with Halo (以下 PwH) との間には差は見られなかった．しかし，900 と 1100 pixels の遠いターゲット距離の条件では，DuH と PwH の間に差は見られた．1100 pixels の距離条件は，ディスプレイサイズの制約から，DuH が画面内で正確にピーク速度を検出できる限界値に近く，多くのアンダーショットが見られており，その結果，Hop との差はあまり見られなくなっている．しかし，PwH の操作時間はターゲット距離に大きく依存するため，遠い距離 (DuH の限界に近い) でも DuH より遅い結果となっている．

図 12 では，オブジェクト密度 16 の場合には，DuH と Hop に差があり，DuH と PwH の差はわずかであることが分かる．しかし，Hop と PwH はオブジェクト密度が多くなるにつれて操作時間は大きくなる傾向にあり，32, 48 objects の条件では，DuH が Hop や PwH より有意に短い時間でタスクを完了できている．

### 5.6.2 空間把握テスト

三要因分散分析の結果，インタラクション技術 ( $F(2,22) = 1.868, p > .05$ ) には主効果は見られず，ターゲット距離 ( $F(2,22) = 5.052, p < .05$ ) とオブジェクト密度 ( $F(2,22) = 18.102, p < .001$ ) に主効果が見られ，ターゲット距離とオブジェクト密度の交互作用 ( $F(4,44) = 3.150, p < .05$ ) も見られた．オブジェクト密度に対する各インタラクション技術の正解率を図 13 に示す．インタラクション技術の主効果

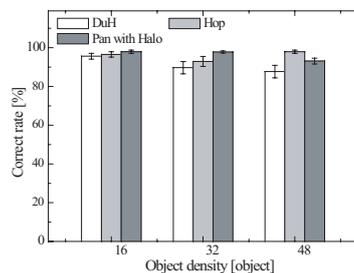


図 13 オブジェクト密度 vs. 正解率

Fig. 13 Object density vs. correct rate

は見られていないが，DuH の精度は他よりも低いものとなっている．特に二回，三回ジャンプした場合や，予測の精度が極端に悪かった場合にミスが生じることが多く，空間把握テストに大きく影響している．また，全インタラクション技術において，オブジェクト密度が増加するにつれて正解率は下がる傾向にあった．

## 6. 議 論

3 章の実験では，Delphian Desktop でも用いているターゲット距離とピーク速度間の線形性を，Halo を用いた画面外へのポインティングに対しても確認することができた．Partial Halo の条件でも線形の関係であったが，図 5 に示しているように不安定な部分は見られる．500 pixels の短いターゲット距離においては，Halo の弧の形状から比較的ターゲット距離を推定しやすく，その結果，通常のポインティングより強い計画運動を行い，大きいピーク速度が検出されやすかった．逆に，1100 pixels の条件では，弧の形状からターゲット距離の推定は難しいため，通常のポインティングよりも弱い計画運動がなされることが多く，ピーク速度は若干通常のポインティングより小さなものになっている．このように，Halo の形状や線の太さなど，ターゲット距離を推定するための手がかりの状態によってピーク速度は影響を受けると考えられ，より安定した DuH の提案をするためには，最適な Halo の条件など

をより詳細な実験で明らかにする必要がある。

DuH は、ピーク速度の検出できる範囲の限界などの問題から、空間把握テストにおいてよい結果は得られていない。また、今回用いた地図ナビゲーションという実験環境では、Halo の中にプロキシを表示する必要があり、これが原因で Halo の弧の形状が判断しにくかったという意見や、プロキシをターゲットであると誤認識してしまったという意見も多くあった。最適なプロキシサイズや形状なども考慮する必要があると考えられるが、タスクの性質とプロキシに表示させる情報量、さらにはオブジェクト密度や最適な地図サイズとの関連など、画面外オブジェクトの可視化技術としては、依然大きい課題があると考えられる。

Hop では、特徴的な操作を用いて画面外ターゲットを獲得するが、レーザービームなどの操作は一般の GUI ではあまり馴染み深い動作とは言えず、また、プロキシの位置が完璧にターゲットの方向と一致していないなど、直感性に欠ける部分があると考えられる。我々の実験の結果、タスク 2 の空間把握テストに備えるために、タスク 1 では、多くのユーザは動作を慎重に行ったため、DuH の方が Hop より、タスク完了時間などについては有効となったと考えられる。しかし、プロキシ+テレポートの概念は多くの場面で有効であると考えられるので、プロキシを Halo の中に表示させ、そのプロキシをクリックすることで自動的にターゲット位置までスクロールするという技術と DuH との比較なども調査していきたい。

今回、空間把握テストは、インタラクション技術の差を検出できる評価指標にはなりえなかったが、PwH や Hop の動作が単純な反復動作になることを防ぎ、タスクをより現実的にすることができたと考えられる。速さと精度の両方を被験者に要求する場合、今回用いた空間把握テストのようなタスクを課すことは有効で、今後、経路計画などより現実的で複雑なタスクの設定を行う際に考慮していきたい。

## 7. ま と め

より効率的に画面外のオブジェクトを現在のウィンドウに獲得するための新たなスクロール技術として、Delphian Desktop というポインティング中のピーク速度を用いたターゲット距離を予測する技術と、ユーザにターゲットまでの距離を把握させる Halo という可視化技術を組み合わせ、DuH (Delphian Desktop using Halo) を提案した。実験 1 では、Halo を用いた環境でもポインティングピーク速度とターゲット距離の間には線形の関係があり、Delphian Desktop の

予測アルゴリズムが適用可能であることを確認した。DuH の評価実験では、DuH が、Hop, Pan with Halo より早く画面外オブジェクトを獲得できることを確認した。今後は Halo によるポインティング難易度の検討や、より現実的なタスクによる評価を考えている。

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) の研究助成による。

## 参 考 文 献

- 1) Appert, C. and Fekete, J.: OrthoZoom scroller: 1D multiscale navigation, *Proc. CHI '06*, pp.21-30 (2006).
- 2) Asano, T., Sharlin, E., Kitamura, Y., Takashima, K. and Kishino, F.: Predictive interaction using Delphian Desktop, *Proc. UIST '05*, pp.133-141 (2005).
- 3) Baudisch, P. and Rosenholtz, R.: Halo: a technique for visualizing off-screen objects, *Proc. CHI '03*, pp.481-488 (2003).
- 4) Elliot, 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).
- 5) Fitts, P.M.: The information capacity of the human motor system in controlling the amplitude of movement, *J. Experimental Psychology*, 47, pp.381-391 (1954).
- 6) Heath, M., Hodges, N., Chua, R. and Elliot, D.: On-line control of rapid aiming movements: unexpected target perturbations and movement kinematics, *Canadian J. Experimental Psychology*, No.52:4, pp.163-173 (1998).
- 7) Hinckley, K., Cutrell, E., Batiche, S. and Muss, T.: Quantitative analysis of scrolling techniques, *Proc. UIST '02*, pp.65-72 (2002).
- 8) Igarashi, T. and Hinckley, K.: Speed-dependent automatic zooming for browsing large documents, *Proc. UIST '00*, pp.139-148 (2000).
- 9) Irani, P., Gutwing, C. and Yang, X.D.: Improving selection of off-screen targets with hopping, *Proc. CHI '06*, pp.299-308 (2006).
- 10) MacKenzie, C.L., Marteniuk, R.G., Dugas, C. and Eickmeier, B.: Three-dimensional movement trajectories in Fitts' task: implications for motor control, *Q. J. Experimental Psychology*, No.39A, pp.629-647 (1987).
- 11) 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).
- 12) Moscovich, T. and Hughes, J.F.: Navigating documents with the virtual scroll ring, *Proc. UIST '04*, pp.57-60 (2004).
- 13) Robbins, D., Cutrell, E., Sarin, R. and Horvits, E.: ZoneZom: map navigation for smartphones with recursive view segmentation, *Proc. AVI '04*, pp.231-234 (2004).