

# Motor Width と Visual Width の差を考慮した ポインティングのモデル化

薄羽 大樹<sup>1,a)</sup> 山中 祥太<sup>2,b)</sup> 宮下 芳明<sup>1,c)</sup>

受付日 2020年4月16日, 採録日 2020年11月5日

**概要:** GUI 上では, ナビゲーションバーのようにクリック可能領域の大きさ (Motor Width) とディスプレイ上に表示される見た目の大きさ (Visual Width) が異なるターゲットが存在する. そのようなインタフェースでは, カーソルがアイテムの中心からアイテムがハイライトされるまで, ユーザは Motor Width を正確に認識できないことがある. そのため, 必要以上に慎重に操作したり, すでにクリック可能であるにもかかわらず無駄にカーソルを動かしてしまうことが考えられる. 本研究の実験では, Motor Width と Visual Width に差があり, かつターゲットと非ターゲットが存在する状況でポインティングを行う. 実験の結果, 操作時間は Motor Width に強く依存し, Motor Width と Visual Width の差によってわずかに操作時間が増加することが分かった. また, 実験結果を元に, Motor Width と Visual Width の差を考慮したポインティングのモデルを構築し, 高精度に操作時間を予測できることを示した. そして, 実験結果と提案モデルを元に既存のナビゲーションバーの改善を提案した.

**キーワード:** パフォーマンスモデリング, フィッツの法則, ポインティング, GUI

## Modeling Users' Pointing Performance with Different Motor and Visual Target Widths

HIROKI USUBA<sup>1,a)</sup> SHOTA YAMANAKA<sup>2,b)</sup> HOMEI MIYASHITA<sup>1,c)</sup>

Received: April 16, 2020, Accepted: November 5, 2020

**Abstract:** In GUIs, users must capture targets that have different motor and visual widths, such as items in navigation bars. In such objects, users do not understand that the motor width is larger than the visual width before hovering on the item, and thus, it takes them time to notice that they can already click on the item. In this study, we conducted an experiment where there is a difference between the motor and the visual widths of a target located between nontargets. We found that participants' movement time strongly depends on the motor width and is slightly affected by the difference. Based on our results, we build a model for capturing this difference, and our results show a better fit than the baseline model. We also discuss existing navigation bars and show the refinement achieved by using our model.

**Keywords:** performance modeling, Fitts' law, pointing, graphical user interface

### 1. はじめに

GUI (Graphical User Interface) 上のクリック可能なオ

ブジェクトは, 異なる *Motor Width* と *Visual Width* を所持している. 本論文では, ユーザが実際にクリック可能な領域を *Motor Width*, ディスプレイ上に表示されるオブジェクトの大きさを *Visual Width* と定義する. たとえば, Windows 8 のウィンドウ枠は明確に描画されており, *Motor Width* と *Visual Width* は等しい. 一方で, macOS Sierra のウィンドウ枠はウィンドウと背景の境界によって示されるため, *Visual Width* が非常に小さくなっているが, 実際にはそれよりも大きい *Motor Width* が設定され

<sup>1</sup> 明治大学  
Meiji University, Nakano, Tokyo 164-8525, Japan

<sup>2</sup> ヤフー株式会社  
Yahoo Japan Corporation, Chiyoda, Tokyo 102-8282, Japan

a) m@mimorisuzu.co

b) syamanak@yahoo-corp.jp

c) homei@homei.com



図 1 ナビゲーションバーの例。アイテムにカーソルがのったとき、アイテムの背景がハイライトされる。背景がハイライトされるまで Motor Width の正確な大きさを把握できないが、左右のアイテムの並びからおおよその Motor Width を推測できる。このとき、アイテムのラベルが Visual Width である

Fig. 1 Example of navigation bar. When a cursor hovers over an item, the background of the item is highlighted. Although users do not understand the motor width of the item until they highlight it, they can predict the motor width based on the arrangement of other items. The item's label is the visual width.

ている。ウィンドウ枠のような小さいオブジェクトにおいては、その見た目は操作時間にあまり影響を与えないことが知られている [1], [2]。

図 1 はインタラクション 2019 のサイト上のナビゲーションバーである\*1。このナビゲーションバーでは、カーソルがアイテムの上ののったとき、アイテムの背景がハイライトされる。そして、ハイライトされた背景が Motor Width である。このようなオブジェクトでは、カーソルをアイテムにのせる前は (図 1 左)、アイテムのラベル部分を目指してカーソルを動かすはずである (Visual Width にカーソルが入るようにする)。そして、カーソルがアイテムののったとき (図 1 右)、背景のハイライトによって Visual Width よりも Motor Width の方が大きかったことが明示される。つまり、操作を行う前は Motor Width を小さく見積もっているため、必要以上に慎重に操作してしまうことが推測される。ウィンドウ枠よりも大きなオブジェクトに関しても、Motor Width と Visual Width の差の影響の調査はすでに行われており、ユーザの動きは Motor Width に強く依存することが知られている [2], [3]。

図 1 に示されるように、ユーザがクリックしたいオブジェクト (ターゲット) はそれ以外のオブジェクト (非ターゲット) に囲まれている。しかし、先行研究 [2], [3] で行われた実験のタスクには、開始領域とターゲットしか存在しない (図 2)。基本的に、それぞれのオブジェクトは、Motor Width が干渉しないように配置されているため、非ターゲットとターゲットの Visual Width からターゲットの Motor Width をおおよそ推測できると考えられる。本研究では、この前提から、ターゲットと非ターゲットが並ぶ状況において (図 3)、Motor Width と Visual Width の差がポインティング性能にどのような影響を与えるのかを調査する。加えて、先行研究 [1], [2], [3] で行われていない

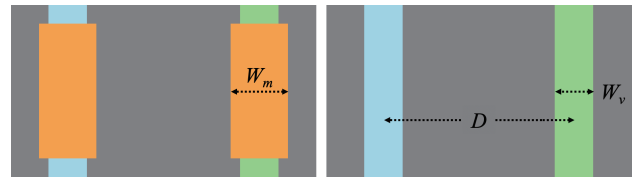


図 2 先行研究のタスクの概要 (文献 [3] の Figure 2 より引用)。青が開始領域、緑がターゲット、橙がハイライトされた Motor Width

Fig. 2 Experimental task outline in Ref. [3] (Fig. 2 in Ref. [3]). Blue object is the start area, green is the target, and orange is the highlighted motor width.

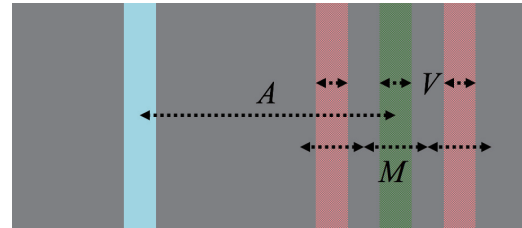


図 3 タスクの概要、ターゲット (緑) が非ターゲット (赤) に囲まれている

Fig. 3 Experimental task outline. Green object is the target and red objects are nontargets.

Motor Width と Visual Width の差を考慮したポインティングのモデルについても提案する。

本論文の貢献は以下のとおりである。

- ターゲットと非ターゲットが存在するポインティングタスクを用いて実験を行った。実験の結果、操作時間は Motor Width に強く依存し、Motor Width と Visual Width の差はわずかに影響を与えることが分かった。この結果は先行研究と整合していた [1], [2], [3]。
- Motor Width と Visual Width の差を考慮したモデルを提案し (式 (1))、そのモデルは既存モデルよりも高い適合度を示した (adjusted  $R^2 = 0.984$ )。

$$MT = a + b \log_2 \left( \frac{A}{M - c|M - V|} + 1 \right) \quad (1)$$

ここで  $MT$  は操作時間、 $A$  はターゲットまでの距離、 $M$  は Motor Width、 $V$  は Visual Width である。また、 $a$ ,  $b$ ,  $c$  は回帰分析によって与えられる定数である (以降の  $a$ ,  $b$ ,  $c$ ,  $d$  も同様である)。

本論文は国内会議 [4] で口頭発表した内容をまとめたものである。

## 2. 関連研究

### 2.1 ポインティングのモデル化

フィッツの法則 (式 (2)) [5] は、ターゲットまでの距離 ( $A$ , いくつかの研究では  $D$ ) とターゲット幅 ( $W$ ) を用いてポインティングの操作時間 ( $MT$ ) を予測するモデルである。式 (2) の対数項は Index of Difficulty ( $ID$ ) と呼ばれ、そのタスクの難易度を示し、つまり、 $ID$  が高くなる

\*1 <https://www.interaction-ipsj.org/2019/>

とそのポインティングを達成するのにより長い時間を要することが予測される。

$$MT = a + b \log_2 \left( \frac{A}{W} + 1 \right) \quad (2)$$

ポインティングタスクを課すユーザスタディでは、実験者は参加者に対してできる限り速く正確にポインティングを完了させるように指示する。また、ポインティングタスクには速さと正確性のトレードオフが存在し、参加者が速くポインティング操作を実行すると、エラー率が高くなり、逆もまた然りである [6]。もし、そのトレードオフが釣り合っていれば、エラー率はおおよそ4%になることが知られている [7], [8]。そして、エラー率が4%から明らかに離れている場合には、フィッツの法則は *Effective Width* [8], [9] (式 (3)) を用いて修正される。Effective Width はクリック座標の標準偏差 ( $\sigma$ ) から計算され、その値をフィッツの法則の  $W$  と置き換える。いくつかの研究 [6], [10], [11] では、入力デバイスの比較をする場合になどに限って Effective Width を使うべきであると指摘されている。

$$MT = a + b \log_2 \left( \frac{A}{\sqrt{2\pi e \sigma}} + 1 \right) \quad (3)$$

フィッツの法則は様々な入力デバイスや操作に適用できる [8], [12], [13], [14]。また、フィッツの法則の変形はいくつか存在し、タッチポインティング [15]、2次元 (式 (4)) [8], [16]、3次元 [17] のポインティングにも適用できる。

$$MT = a + b \log_2 \left( \sqrt{\left( \frac{A}{W} \right)^2 + \eta \left( \frac{A}{H} \right)^2} + 1 \right) \quad (4)$$

ここで、 $\eta$  は回帰によって求まる定数であり (式 (7) でも同様)、 $H$  はターゲットの高さである。

## 2.2 Motor Width と Visual Width の効果

我々はこれまで Motor Width と Visual Width の影響をウィンドウ枠のような小さいターゲット [1], [2]、そしてそれよりも大きいターゲット [2], [3] について調査している。これらの研究では、Motor Width が Visual Width よりも小さい、等しい、大きい条件が存在した。実験の結果、ユーザの動きは Visual Width ではなく Motor Width に強く依存することが分かった。たとえば、ポインティング中の平均速度は Visual Width (ターゲットの外見) によって変化せず、また、クリック座標の分布は Motor Width に依存する。加えて、Effective Width を使うことで高精度に操作時間を予測できることを示した。また、指を用いたタッチポインティングにおける Motor Width と Visual Width の影響も調査し、操作時間は Motor Width に強く依存することが分かっている [18]。

## 2.3 動的な Motor Space や Motor Width の変更

マウスの動きとスクリーン上のカーソルの動きの関係は Control-Display (C-D) ゲインと呼ばれる。たとえば、C-D ゲインが高いと、ユーザは少しのマウスの動きでカーソルを速く操作できる。Semantic Pointing [19] では、カーソルがターゲットに近づいたときに C-D ゲインを下げ、カーソルの速度を遅くしている。C-D ゲインを小さくすることは、実質的にターゲットサイズが大きくなることと同等の効果がある。Sticky Icons [20] も同様のアプローチを取っており、カーソルがターゲット上にある場合に C-D ゲインを減少させている。Chapuis と Dragicevic は *Visual Space* と *Motor Space* の効果を調査した [10]。なお、ここでの Visual Space は本論文の Visual Width と同義だが、Motor Space は Motor Width でなく C-D ゲインを指している。

macOS の Dock のような動的なターゲットの拡大 [21], [22], [23] や、動的なターゲットの縮小についても調査されている [24], [25], [26]。動的なターゲットの拡大に関しては、視覚的にのみターゲットを拡大しても [23]、またターゲットの拡大を推測できなくとも [22] ポインティング操作を容易にすることが知られている。また、動的なターゲットの縮小に関しては、縮小する速度が速いほど Time Pressure が生まれ、速く操作できることが知られている [25]。加えて、Bubble Cursor [27] のようないくつかのエリアカーソル [28], [29] では、動的にターゲットの Motor Width を変更している (実際にはカーソルのアクティベーションエリアを変更している)。

上述のように、動的な Motor Space や Motor Width の変更についての研究はこれまでもなされている。本研究では、静的に Motor Width と Visual Width が異なる状況を対象とする。

## 3. 実験

### 3.1 機材

PC は Apple MacBook Pro (Intel Core i5, 2.4 GHz, 2 cores, Intel Iris 1536 MB, 8 GB RAM, macOS Sierra)、解像度は 1,680 × 1,050 pixels であり、ディスプレイは 13.3 インチ (286.47 × 179.04 mm, 0.17 mm/pixel) であった。入力デバイスは Logitech M100R (1,000 dpi) の光学マウスであった。カーソルの速度は macOS のデフォルトであり、実験システムはつねにフルスクリーンで表示された。

本研究では、デスクトップインタフェースを操作する際によく用いられるマウスを入力デバイスとして使用した。

### 3.2 参加者

12 名が参加した。1 名は女性、平均年齢は 22.58 歳、標準偏差は 1.51 歳であった。すべての参加者が普段から右手でマウスを操作し、実験も右手で行った。実験後、それ

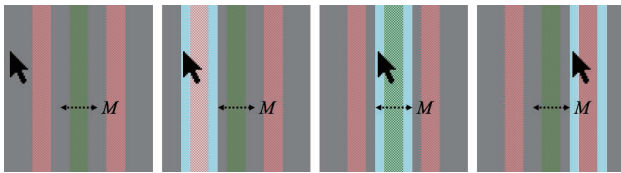


図 4 タスクでは、カーソルがターゲットか非ターゲットにのったとき、その Motor Width がハイライトされた

Fig. 4 In the task, when the cursor hovers over the target or nontargets, the motor width of the object is highlighted.

それぞれの参加者に 2,000 円が報酬として支払われた。

### 3.3 タスク

図 3 にタスクの概要を示す。タスクには、開始領域 (青)、ターゲット (緑)、2 つの非ターゲット (赤) が存在した。それぞれのオブジェクトがディスプレイ上に表示される大きさは Visual Width ( $V$ ) によって制御された。ターゲットと非ターゲットは同じ Motor Width ( $M$ ) を所持し、Motor Width に隙間がないように配置された。

参加者が開始領域をクリックすると、試行の開始を伝える音が再生された。次に、参加者のクリックがターゲットの範囲 ( $M$ ) 内であれば、その試行を成功とし、ベル音が再生された。一方で、クリックが  $M$  の範囲外であった場合は、試行はエラーとしてカウントされ、失敗を伝える音が再生された。参加者は「できる限り速く正確にターゲットを狙わなければならない」と指示された。

GUI 上に実在するナビゲーションバーと同様、参加者には背景の変化によるフィードバックを与えた (図 4)。たとえば、インタラクション 2019 のナビゲーションバーでは、カーソルがアイテムにのったとき、背景が黒から緑に変化し、ユーザにアイテムの Motor Width を知覚させる (図 1)。我々の実験でも同様、図 4 に示すように、カーソルが Motor Width にのったとき、Motor Width をハイライトした。このハイライトは、ターゲットだけでなく、非ターゲットでも行った。また、ターゲットと非ターゲットが同じ色である場合、参加者はどのオブジェクトがターゲットであるか判断できないため、異なる色を使用した。

### 3.4 デザインと手順

開始領域の中心からターゲットの中心までの距離 ( $A$ ) は 2 種類 (600, 800 pixels; それぞれ, 102.31, 136.41 mm), Motor Width ( $M$ ) と Visual Width ( $V$ ) の組合せ ( $Pattern$ ) は 10 種類 ( $(M, V) = (20, 20), (40, 20), (40, 40), (70, 20), (70, 40), (70, 70), (120, 20), (120, 40), (120, 70), (120, 120)$ ; それぞれ, 20, 40, 70, 120 pixels は 2.35, 4.70, 8.22, 14.09 mm) であった。つまり、Motor Width は Visual Width と等しいか、それよりも大きかった。これらの値は先行研究 [2], [3] を参考にした。非ターゲットが存在する場合では、図 1 のように Motor Width

が Visual Width よりも大きい状況か、もしくは、アイテム間に区切り線が引いてある (つまり、Motor Width と Visual Width が等しい) 状況が多い。そのため、本研究では、Motor Width が Visual Width よりも大きい状況、もしくはそれらが等しい状況を実験条件とした。

1 セットは  $2A \times 10 Pattern = 20$  試行であり、セット内の  $A$  と  $Pattern$  の出現順序はランダムであった。実験の開始前、それぞれの参加者には簡単な実験の説明を行った。参加者は 1 セットの練習を行い、10 セットの本番を行った。本番セットを行ったのち、参加者に「それぞれの条件でどのような戦略を取り、試行を完了させたか」を尋ねた。全試行は 2,400 回 (つまり、 $2A \times 10 Pattern \times 10$  セット  $\times$  12 名) であり、1 名あたり 10 分を要した。

### 3.5 計測値

操作時間 ( $MT$ ; エラーを除く、開始領域をクリックしてからターゲットをクリックするまでの時間)、反応時間 ( $RT$ ; エラーを除く、カーソルがターゲットにのってからターゲットをクリックするまでの時間)、エラー率であった。

### 3.6 結果

2395 試行のうち (5 試行を外れ値とした\*2), エラーは 33 試行であった (1.38%)。このエラー率は標準 (4% [7], [8]) よりも低かった。繰返しのある分散分析で分析を行った。多重比較には Bonferroni 法を用いた。独立変数は、 $A$  と  $Pattern$  であり、従属変数は、 $MT$ ,  $RT$ , エラー率であった。グラフ中のエラーバーは標準誤差を示し, \*\*\*, \*\*, \* はそれぞれ,  $p < 0.001$ ,  $p < 0.01$ ,  $p < 0.05$  を示す。

#### 3.6.1 エラー率

$A$  ( $F_{1,11} = 0.88$ ,  $p = 0.37$ ,  $\eta_p^2 = 0.074$ ),  $Pattern$  ( $F_{9,99} = 0.99$ ,  $p = 0.45$ ,  $\eta_p^2 = 0.083$ ) であり、主効果は見られなかった。Appert らは、背景のハイライトによってより多くのエラーが観測されると述べている [30]。一方で、Akamatsu らは、ハイライトの有無によるエラー率の差はないと述べている [31]。本実験では、5 名の参加者は「左の非ターゲットのハイライトを確認してから少し右にカーソルを移動させ、ポインティングを成功させた」と述べていた。つまり、本実験においては、Motor Width のハイライトがポインティングを支援していたと考えられる。

本実験では、先行研究 [2], [3] と同じ  $A$ ,  $M$ ,  $V$  を使用した。そして、先行研究で観測されたエラー率は 3.21% であり、そもそも、この実験条件 ( $A$ ,  $M$ ,  $V$ ) で観測されるエラー率は、標準よりも低いことが予測される。以上 2 つの理由により、本実験のエラー率は標準よりも低くなった

\*2 クリック地点が  $\frac{A}{2}$  よりも手前だった場合、その試行を外れ値として扱った [7]。また、「クリック位置がターゲットの中心から  $2W$  以上離れている場合」の基準は使用しなかった。これは、我々の実験では Motor Width と Visual Width が異なるため、そのような試行もありうると考えたからである。

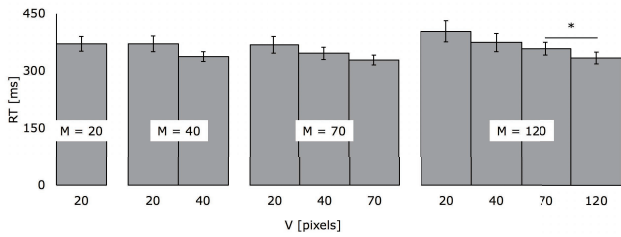


図 5 各 Motor Width ( $M$ ) における反応時間 ( $RT$ ) に対する Visual Width ( $V$ ) の影響

Fig. 5  $RT$  versus  $V$  for each  $M$ .

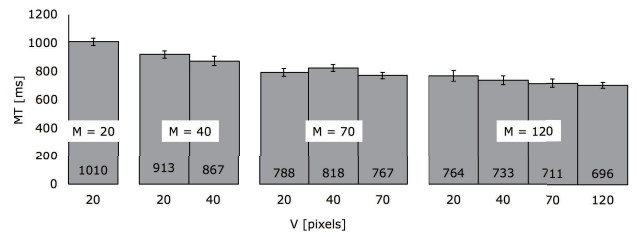


図 7 各 Motor Width ( $M$ ) における操作時間 ( $MT$ ) に対する Visual Width ( $V$ ) の影響

Fig. 7  $MT$  versus  $V$  for each  $M$ .

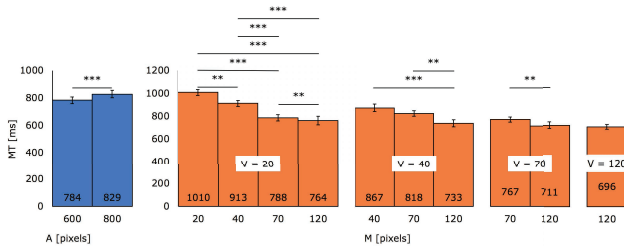


図 6 各 Visual Width ( $V$ ) における操作時間 ( $MT$ ) に対する距離 ( $A$ ) と Motor Width ( $M$ ) の影響

Fig. 6  $MT$  versus  $A$  and versus  $M$  for each  $V$ .

と考えられる。

### 3.6.2 反応時間 ( $RT$ )

主効果が見られたのは *Pattern* ( $F_{9,99} = 7.64, p < 0.001, \eta_p^2 = 0.41$ ) であった。図 5 に多重比較の結果を示す。ほぼすべての  $V$  ペア間では、差が見られなかった。また、 $M = V$  のときに  $RT$  は最速であり、 $M$  と  $V$  の差が増加すると  $RT$  は増加した。Motor Width と Visual Width に差がある場合に反応時間が減少することは、先行研究 [1], [2], [3] と整合した結果であった。交互作用が見られたのは  $A \times Pattern$  ( $F_{9,99} = 2.73, p < 0.01, \eta_p^2 = 0.20$ ) であった。

### 3.6.3 操作時間 ( $MT$ )

主効果が見られたのは  $A$  ( $F_{1,11} = 70.97, p < 0.001, \eta_p^2 = 0.87$ ), *Pattern* ( $F_{9,99} = 81.61, p < 0.001, \eta_p^2 = 0.88$ ) であった。多重比較の結果、 $A$  が減少するほど  $MT$  が減少することが分かった。また、各  $V$  ごとにみても、 $M$  が増加するほど  $MT$  が減少していた (図 6)。そして、各  $M$  ごとにみても、すべての  $V$  ペア間では差は見られなかったが、 $M = V$  のときに  $MT$  が最速であり、また、 $M$  と  $V$  の差が増加するほど  $MT$  が増加していた (図 7)。Motor Width と Visual Width に差がある場合に操作時間が減少することは、先行研究 [1], [2], [3] と整合した結果であった。

### 3.6.4 モデルの適合度

前述のとおり、操作時間 ( $MT$ ) は Motor Width ( $M$ ) に強く影響を受けていた。また、 $M$  と  $V$  の差が増加するほど  $MT$  がわずかに増加していた。我々はフィッツの法則の  $W$  を  $M$  に置き換えたモデル (式 (5), 以後、 $ID_m$  モデル) の適合度を検証した。一般的に用いられる適合度の

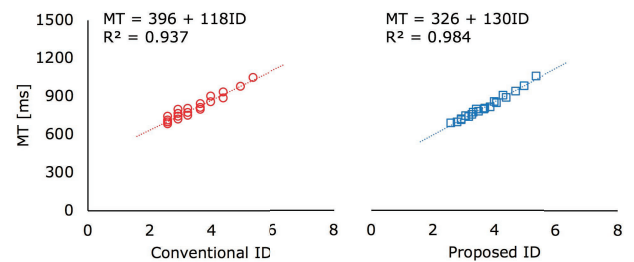


図 8 提案モデルと  $ID_m$  モデルの適合度 ( $N = 20$ )

Fig. 8 Model fitness for our results obtained using the candidate models ( $N = 20$ ).

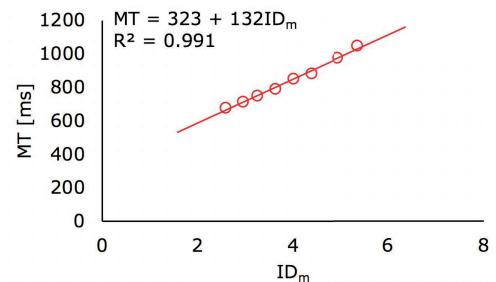


図 9 Motor Width と Visual Width が等しい条件 ( $M = V$ ) のみにおける  $ID_m$  モデルの適合度 ( $N = 8$ )

Fig. 9 Model fitness for our results obtained when  $M = V$  ( $N = 8$ ).

基準値 ( $R^2 > 0.90$  [7], [32]) を考慮すれば、 $ID_m$  モデルは高い適合度を示していた (adj.  $R^2 > 0.937$ , 図 8 左, 表 1 の上段)。また、 $M = V$  のみのデータでは、 $ID_m$  モデルはより高い適合度を示した (adj.  $R^2 > 0.991$ , 図 9)。

$$MT = a + b \log_2 \left( \frac{A}{M} + 1 \right) \quad (5)$$

## 4. モデルの改善

上述のモデルの適合度によれば、 $ID_m$  モデルは全体の結果よりも  $M = V$  のみの場合により高い適合度を示していた。つまり、適合度は  $V$  による  $MT$  のバラつきによって減少すると考えられる。本実験では、 $M \neq V$  は 6 条件のみであったが、 $M \neq V$  の条件が増えると、 $V$  によるバラつきが増加し、適合度が減少し、基準値を下回ることが予測される (つまり  $R^2 < 0.90$  になる)。そのため、より複雑な条件であっても高精度に操作時間 ( $MT$ ) を予測で

表 1 提案モデルと  $ID_m$  モデルの適合度 ( $N = 20$ ).  $\text{adj. } R^2$  が高く,  $AIC$  が低いほど良いモデルである.  $a, b, c$  は各種定数の推定値, および 95%信頼区間の値を [下限値, 上限値] で表記している

Table 1 Model fitting results for the candidate models ( $N = 20$ ). Higher is better for  $\text{adj. } R^2$ , and lower is better for  $AIC$ .  $a, b$ , and  $c$  are estimated constants with 95% CIs [lower, upper].

Eq.	$ID$	$a$	$b$	$c$	$\text{adj. } R^2$	$AIC$
5	$\log_2 \left( \frac{A}{M} + 1 \right)$	396 [342, 450]	118 [103, 133]		0.937	192
6	$\log_2 \left( \frac{A}{M-c(M-V)} + 1 \right)$	326 [282, 361]	130 [121, 139]	0.380 [0.296, 0.465]	0.984	166

きるモデルが必要であると考えられる.

先行研究によれば, Visual Width の水準数が少ない [1], [2], もしくは, Motor Width と Visual Width が似た値であれば [2], [3],  $ID_m$  モデルは高い適合度を示していた. また, Motor Width と Visual Width が様々な値であっても Effective Width (式 (3)) を用いることで高精度に操作時間を予測できるとされている [2], [3]. しかし, Effective Width は, たとえば, 入力デバイスの比較のときに用いるべきであり, GUI におけるオブジェクトサイズなどを設定する場合には Effective Width を使用しないモデルが求められる [6], [10], [11]. そのため, Effective Width がたとえ高い適合度を示しても, ターゲットのパラメータ ( $A, M, V$ ) から  $MT$  を予測できるモデルはデザイナーにとって有用である.

本実験の結果より,  $MT$  は  $A$  と  $M$  に強く依存していた. そのため, 我々は  $A$  と  $M$  が含まれる  $ID_m$  モデルを元に新しいモデルを構築する. また,  $M = V$  のときに, 各条件において  $MT$  が最速であることを観測した. つまり, Motor Width と Visual Width の差によって  $MT$  が僅かに増加することが分かった. 以上のことをふまえると, 式 (6) が Motor Width と Visual Width の差を考慮したポインティングのモデルであると考えられる.

$$MT = a + b \log_2 \left( \frac{A}{M - c(M - V)} + 1 \right) \quad (6)$$

式 (6) は,  $ID_m$  の  $M$  の項に Motor Width と Visual Width の差 ( $M - V$ ) を加えたものである. これにより, Motor Width と Visual Width の差による  $MT$  の増加を予測できるだろう. 式 (6) では,  $M = V$  のとき,  $c(M - V)$  の項は 0 になり,  $ID_m$  モデルと等しくなる. つまり, 見たとくクリック可能領域の大きさが合致している一般的なポインティングタスクでは, 提案モデルは従来のモデルと整合している. また,  $M \neq V$  のときには, 対数項の分母が減少し, 予測される  $MT$  が増加する. つまり, 式 (6) によって予測される  $MT$  は本実験で観測された  $MT$  (図 7) と整合しているといえる.

表 1 はそれぞれのモデルの適合度であり, 図 8 はそれぞれの  $ID$  においてデータをプロットしたものである. 提案モデルは 3 つの定数が必要であり,  $ID_m$  モデルの定数は 2 つだけである. そのため,  $\text{adj. } R^2$  に加えて, 赤池情報量規準 (Akaike Information Criterion,  $AIC$ ) [33], [34] を用いてモデルを比較する.  $AIC$  が低く,  $\text{adj. } R^2$  が高いモデルがより良いモデルである. 特に,  $AIC$  の差が 10 以上あれば統計的に十分であるとされている [35]. HCI 分野では, これまでも  $AIC$  を用いたモデルの比較が行われている [34], [36], [37]. 表 1 に示されるように, 提案モデルは  $ID_m$  モデルよりも高い適合度を示した.

## 5. 議論

### 5.1 戦略の違い

先行研究 [1], [3] によれば, 反応時間と操作時間はどちらも Motor Width と Visual Width が等しいときに最速であり, それらに差がある場合には増加していた. 本実験では, Motor Width を事前に通知しない, 非ターゲットが存在するなどの先行研究と比べて条件が違う点がいくつか存在した. そのため, 先行研究では見られなかった「ハイライトを見ながらポインティング操作を行う戦略」が見られた. 先行研究と本研究を比較してみると, いくつか実験条件や戦略の違いは見られたが, 本研究の実験結果は先行研究と整合していた. つまり, Motor Width の通知や非ターゲットの存在 (また, それによる戦略の違い) はあまりパフォーマンスには影響を与えないと考えられる.

### 5.2 ナビゲーションバーの分析と改善

実験結果と提案モデルの適合度を考慮すれば, デザイナーは Motor Width を大きくとり, そして Motor Width と Visual Width の差を除去するために Motor Width をはっきりと描画すべきである. たとえば, 図 1 のナビゲーションバーでは, デザイナーはアイテム間の区切りを描画し, Motor Width を明確にすべきである (図 10). そうすることで, Motor Width と Visual Width の差をなくすことができ, ユーザの操作時間を減少させられるだろう. 実際に



図 10 インタクション 2019 のナビゲーションバーの改善（上が改善前，下が改善後）

Fig. 10 Interaction 2019 navigation refinement from top to bottom.

は、図 1 のナビゲーションバーでは、アイテム間にマージンがあり（本章の改善ではこのマージンも除去している）、また、実験ではターゲットと非ターゲットのサイズが同じであったが、ナビゲーションバー中のアイテムのサイズは様々である。また、本研究の実験では、ターゲットの高さは画面の高さと同一であったが（つまり、ターゲットの高さは考慮していない実験であったが）、図 1 は Motor Width と Visual Width に加えて異なる Motor Height（ユーザがクリックできるターゲットの高さ）と Visual Height（ディスプレイに表示される高さ）を所持している。そのため、実験結果や提案モデルをそのまま適用できるかについてはさらなる検証が必要であると考えられる。

## 6. 制約と展望

図 1 に示されるナビゲーションバーは、ターゲット幅に加え、異なる Motor Height と Visual Height を所持している。提案モデルは 1D の動きを元に設計されているため、ユーザがナビゲーションバー内で水平にカーソルを操作する場合に適用可能である。そのため、2D ポインティングに対しては、モデルをいくつか修正する必要があると考えられる。図 11 に 2D ポインティングのパラメタを示す。テキストの長さや高さ（Visual Width ( $V_w$ ), Visual Height ( $V_h$ )) は緑のターゲットで示され、橙色の点線はユーザが実際にクリックできる領域（Motor Width ( $M_w$ ), Motor Height ( $M_h$ )) である（実際には不可視である）。 $A$  はターゲットまでの距離である。2D ポインティングのモデル（式 (4)）[8], [16] を考慮すれば、Motor Size と Visual Size を考慮した 2D ポインティングのモデルは式 (7) で示される。式 (7) は、単に 2D のモデルと本研究の 1D の提案モデルを統合しただけであり、実際には定数の適切な個数などさらなる実験によって検証すべきことは多い。さらに、1D の提案モデル（式 (1)）は Motor Width を推測可能であった場合（本実験では、ターゲットと非ターゲットの並びからターゲットの Motor Width をおおよそ推測できた）に適用可能であり、式 (7) に関しても  $M_h$  がある程度推測可能な場合に限られるだろう（たとえば、 $M_h$  が  $V_h$  が近い場合）。

本研究では、ターゲットと非ターゲットは Motor Width においてマージンのない条件のみで実験を行った。しかし、たとえば、図 1 のようなナビゲーションバーでは、アイテム間にマージンがある。そのような場合では、本実験で参

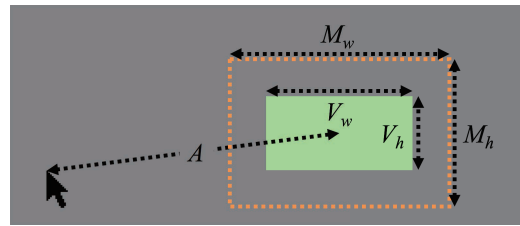


図 11 Motor Size と Visual Size が異なる 2D ポインティング。橙の点線は Motor Size であり、実際には不可視である

Fig. 11 2D pointing with a difference between the motor and the visual sizes. Orange dotted line indicates the motor size that is actually invisible.

加者が行っていた「左の非ターゲットのハイライトを確認してから、少し右にカーソルを移動させる」操作は、カーソルがターゲットではなくマージンにのってしまうため、ポインティングを成功させる操作にならない。そのため、マージンがある場合には本実験と異なる戦略でポインティングを行うことが考えられ、モデルもその戦略にあわせてマージンを考慮する必要があるだろう。2D モデルの構築のために、今後もさらなる検証をしていく予定である。

$$MT = a$$

$$+ b \log_2 \left( \sqrt{\left( \frac{A}{M_w - c |M_w - V_w|} \right)^2 + \eta \left( \frac{A}{M_h - d |M_h - V_h|} \right)^2} + 1 \right) \quad (7)$$

## 7. 結論

本研究では、Motor Width と Visual Width に差があり、ターゲットと非ターゲットが存在するポインティングの実験を行った。実験の結果、参加者の操作時間は Motor Width に強く依存しており、また、Motor Width と Visual Width の差がわずかに操作時間を増加させることが分かった。反応時間や操作時間などの実験結果については、先行研究 [1], [2], [3] と整合していた。また、我々は Motor Width と Visual Width の差を考慮した新しいモデル（式 (1)）を構築した。提案モデルは、Motor Width と Visual Width が異なる状況であっても高精度に操作時間を予測でき、また、Effective Width を含まないため、デザイナーは求める操作時間に対する実際の距離やターゲットのサイズを知ることが可能である。

## 参考文献

- [1] Usuba, H., Yamanaka, S. and Miyashita, H.: User Performance by the Difference Between Motor and Visual Widths for Small Target Pointing, *Proc. 10th Nordic Conference on Human-Computer Interaction, NordiCHI '18*, pp.161–169, ACM (online), DOI: 10.1145/3240167.3240171 (2018).
- [2] 薄羽大樹, 山中祥太, 宮下芳明: Motor Width と Visual Width が異なる状況下でのポインティング性能, 情報処理学会論文誌, Vol.60, No.4, pp.1184–1199 (2019).

- [3] Usuba, H., Yamanaka, S. and Miyashita, H.: Pointing to Targets with Difference between Motor and Visual Widths, *Proc. 30th Australian Conference on Computer-Human Interaction, OzCHI '18* (2018).
- [4] 薄羽大樹, 山中祥太, 宮下芳明: Motor Width と Visual Width の差を考慮したポインティングのモデル化, *インタラクティブ 2019 論文集*, pp.122–130 (2019).
- [5] Fitts, P.M.: The information capacity of the human motor system in controlling the amplitude of movement, *Journal of Experimental Psychology*, Vol.47, No.6, p.381 (1954).
- [6] Zhai, S., Kong, J. and Ren, X.: Speed-accuracy trade-off in Fitts' law tasks: On the equivalency of actual and nominal pointing precision, *International Journal of Human-Computer Studies*, Vol.61, No.6, pp.823–856 (online), DOI: 10.1016/j.ijhcs.2004.09.007 (2004).
- [7] Soukoreff, R.W. and MacKenzie, I.S.: Towards a standard for pointing device evaluation, perspectives on 27 years of Fitts' law research in HCI, *International Journal of Human-computer Studies*, Vol.61, No.6, pp.751–789 (2004).
- [8] MacKenzie, I.S.: Fitts' Law as a Research and Design Tool in Human-Computer Interaction, *Human-Computer Interaction*, Vol.7, No.1, pp.91–139 (online), DOI: 10.1207/s15327051hci0701.3 (1992).
- [9] Crossman, E.R.F.W. and Goodeve, P.J.: Feedback control of hand-movement and Fitts' law, *The Quarterly Journal of Experimental Psychology Section A*, Vol.35, No.2, pp.251–278 (online), DOI: 10.1080/14640748308402133 (1983).
- [10] Chapuis, O. and Dragicovic, P.: Effects of Motor Scale, Visual Scale, and Quantization on Small Target Acquisition Difficulty, *ACM Trans. Comput.-Hum. Interact.*, Vol.18, No.3, pp.13:1–13:32 (online), DOI: 10.1145/1993060.1993063 (2011).
- [11] Wright, C.E. and Lee, F.: Issues Related to HCI Application of Fitts's Law, *Human-Computer Interaction*, Vol.28, No.6, pp.548–578 (online), DOI: 10.1080/07370024.2013.803873 (2013).
- [12] Card, S.K., English, W.K. and Burr, B.J.: Evaluation of Mouse, Rate-Controlled Isometric Joystick, Step Keys, and Text Keys for Text Selection on a CRT, *Ergonomics*, Vol.21, No.8, pp.601–613 (online), DOI: 10.1080/00140137808931762 (1978).
- [13] MacKenzie, I.S., Sellen, A. and Buxton, W.A.S.: A Comparison of Input Devices in Element Pointing and Dragging Tasks, *Proc. SIGCHI Conference on Human Factors in Computing Systems, CHI '91*, pp.161–166, ACM (online), DOI: 10.1145/108844.108868 (1991).
- [14] Gillan, D.J., Holden, K., Adam, S., Rudisill, M. and Magee, L.: How Does Fitts' Law Fit Pointing and Dragging?, *Proc. SIGCHI Conference on Human Factors in Computing Systems, CHI '90*, pp.227–234, ACM (online), DOI: 10.1145/97243.97278 (1990).
- [15] Bi, X., Li, Y. and Zhai, S.: FFitts Law: Modeling Finger Touch with Fitts' Law, *Proc. SIGCHI Conference on Human Factors in Computing Systems, CHI '13*, pp.1363–1372, ACM (online), DOI: 10.1145/2470654.2466180 (2013).
- [16] Accot, J. and Zhai, S.: Refining Fitts' Law Models for Bivariate Pointing, *Proc. SIGCHI Conference on Human Factors in Computing Systems, CHI '03*, pp.193–200, ACM (online), DOI: 10.1145/642611.642646 (2003).
- [17] Grossman, T. and Balakrishnan, R.: Pointing at Trivariate Targets in 3D Environments, *Proc. SIGCHI Conference on Human Factors in Computing Systems, CHI '04*, pp.447–454, ACM (online), DOI: 10.1145/985692.985749 (2004).
- [18] Usuba, H., Yamanaka, S. and Miyashita, H.: Touch Pointing Performance for Uncertain Touchable Sizes of 1D Targets, *Proc. 21st International Conference on Human-Computer Interaction with Mobile Devices and Services*, pp.1–8, Association for Computing Machinery (online), DOI: 10.1145/3338286.3340131 (2019).
- [19] Blanch, R., Guiard, Y. and Beaudouin-Lafon, M.: Semantic Pointing: Improving Target Acquisition with Control-display Ratio Adaptation, *Proc. SIGCHI Conference on Human Factors in Computing Systems, CHI '04*, pp.519–526, ACM (online), DOI: 10.1145/985692.985758 (2004).
- [20] Worden, A., Walker, N., Bharat, K. and Hudson, S.: Making Computers Easier for Older Adults to Use: Area Cursors and Sticky Icons, *Proc. ACM SIGCHI Conference on Human Factors in Computing Systems, CHI '97*, pp.266–271, ACM (online), DOI: 10.1145/258549.258724 (1997).
- [21] McGuffin, M. and Balakrishnan, R.: Acquisition of Expanding Targets, *Proc. SIGCHI Conference on Human Factors in Computing Systems, CHI '02*, pp.57–64, ACM (online), DOI: 10.1145/503376.503388 (2002).
- [22] Zhai, S., Conversy, S., Beaudouin-Lafon, M. and Guiard, Y.: Human On-line Response to Target Expansion, *Proc. SIGCHI Conference on Human Factors in Computing Systems, CHI '03*, pp.177–184, ACM (online), DOI: 10.1145/642611.642644 (2003).
- [23] Cockburn, A. and Brock, P.: Human On-line Response to Visual and Motor Target Expansion, *Proc. Graphics Interface 2006, GI '06*, pp.81–87, Canadian Information Processing Society (2006), available from <http://dl.acm.org/citation.cfm?id=1143079.1143093>.
- [24] Johnson, W.W. and Hart, S.G.: Step Tracking Shrinking Targets, *Proc. Human Factors Society Annual Meeting*, Vol.31, No.2, pp.248–252 (online), DOI: 10.1177/154193128703100225 (1987).
- [25] Hoffmann, E.R.: Capture of shrinking targets, *Ergonomics*, Vol.54, No.6, pp.519–530 (online), DOI: 10.1080/00140139.2011.576775 (2011).
- [26] Hoffmann, E.R., Chan, A.H. and Dizmen, C.: Capture of shrinking targets with realistic shrink patterns, *Ergonomics*, Vol.56, No.11, pp.1766–1776 (online), DOI: 10.1080/00140139.2013.829924 (2013).
- [27] Grossman, T. and Balakrishnan, R.: The Bubble Cursor: Enhancing Target Acquisition by Dynamic Resizing of the Cursor's Activation Area, *Proc. SIGCHI Conference on Human Factors in Computing Systems, CHI '05*, pp.281–290, ACM (online), DOI: 10.1145/1054972.1055012 (2005).
- [28] Chapuis, O., Labrune, J.-B. and Pietriga, E.: DynaSpot: Speed-dependent Area Cursor, *Proc. SIGCHI Conference on Human Factors in Computing Systems, CHI '09*, pp.1391–1400, ACM (online), DOI: 10.1145/1518701.1518911 (2009).
- [29] Su, X., Au, O.K.-C. and Lau, R.W.: The Implicit Fan Cursor: A Velocity Dependent Area Cursor, *Proc. SIGCHI Conference on Human Factors in Computing Systems, CHI '14*, pp.753–762, ACM (online), DOI: 10.1145/2556288.2557095 (2014).
- [30] Appert, C., Chapuis, O. and Beaudouin-Lafon, M.: Evaluation of Pointing Performance on Screen Edges, *Proc. Working Conference on Advanced Visual In-*



- terfaces, *AVI '08*, pp.119–126, ACM (online), DOI: 10.1145/1385569.1385590 (2008).
- [31] Akamatsu, M., Mackenzie, I.S. and Hasbroucq, T.: A comparison of tactile, auditory, and visual feedback in a pointing task using a mouse-type device, *Ergonomics*, Vol.38, No.4, pp.816–827 (online), DOI: 10.1080/00140139508925152 (1995).
- [32] Gori, J., Rioul, O., Guiard, Y. and Beaudouin-Lafon, M.: The Perils of Confounding Factors: How Fitts' Law Experiments Can Lead to False Conclusions, *Proc. 2018 CHI Conference on Human Factors in Computing Systems, CHI '18*, pp.196:1–196:10, ACM (online), DOI: 10.1145/3173574.3173770 (2018).
- [33] Akaike, H.: A new look at the statistical model identification, *IEEE Trans. Automatic Control*, Vol.19, No.6, pp.716–723 (online), DOI: 10.1109/TAC.1974.1100705 (1974).
- [34] Ren, X., Kong, J. and Jiang, X.-Q.: SH-Model: A Model Based on Both System and Human Effects for Pointing Task Evaluation, *IPSJ Digital Courier*, Vol.1, pp.193–203 (online), DOI: 10.2197/ipsjdc.1.193 (2005).
- [35] Burnham, K.P. and Anderson, D.R.: *Model selection and multimodel inference: A practical information-theoretic approach*, Springer Science & Business Media (2003).
- [36] Yamanaka, S., Stuerzlinger, W. and Miyashita, H.: Steering Through Sequential Linear Path Segments, *Proc. 2017 CHI Conference on Human Factors in Computing Systems, CHI '17*, pp.232–243, ACM (online), DOI: 10.1145/3025453.3025836 (2017).
- [37] Yamanaka, S., Stuerzlinger, W. and Miyashita, H.: Steering Through Successive Objects, *Proc. 2018 CHI Conference on Human Factors in Computing Systems, CHI '18*, pp.603:1–603:13, ACM (online), DOI: 10.1145/3173574.3174177 (2018).



宮下 芳明 (正会員)

千葉大学工学部卒業 (画像工学), 富山大学大学院で音楽教育を専攻, 北陸先端科学技術大学院大学で博士号 (知識科学) 取得, 優秀修了者賞, 2007 年度明治大学理工学部に着任. 2009 年度准教授. 2013 年同大学総合数理学部先端メディアサイエンス学科所属. 2014 年より教授, 現在に至る. 日本ソフトウェア科学会, VR 学会, ヒューマンインタフェース学会, ACM 各会員.



薄羽 大樹 (学生会員)

2018 年度明治大学大学院で修士号 (理学) を取得. 2019 年度から同大学院博士後期課程に所属, 現在に至る.



山中 祥太 (正会員)

2016 年明治大学大学院で博士号を取得 (工学). 2015 年日本学術振興会特別研究員 DC2, 2016 年日本学術振興会特別研究員 PD および明治大学総合数理学部客員研究員を勤めたのち, 2017 年ヤフー株式会社に入社 (Yahoo!

JAPAN 研究所, 主任研究員), 2018 年より同社上席研究員, 現在に至る. ユーザインタフェース研究, 特に運動性能のモデル化に興味を持つ.