

画面角と画面端のターゲット配置が操作時間に与える影響

大場洋介¹ 薄羽大樹¹ 山中祥太² 宮下芳明¹

概要：Windows 10 では画面の左下角にスタートボタンがあったが、Windows 11 では下端中央に変更された。ターゲットが画面角や画面端に接している場合、誤選択が起きづらく、操作時間は短縮される。ポインティングの操作時間を予測できるモデルは従来から提案されており、画面角、画面端に接する配置それぞれにおけるポインティングのパフォーマンスが調査されている。一方で、画面角に接するターゲットと画面端に接するターゲットのどちらが良い配置となるか、その条件は明らかにされていない。たとえば横長のディスプレイでは、画面角に接するターゲットは画面端に接するターゲットよりも画面内の各位置からの平均距離が長く、条件によっては画面端に接するターゲットの方が平均的に速く選択できるかもしれない。本研究では、画面角に接するターゲットと画面端に接するターゲットの配置の差が操作時間に与える影響を調査した。カーソルの存在確率を一様分布と仮定して、操作時間予測モデルから平均操作時間を求めた。結果、操作時間のみを考慮すれば、ターゲットの大きさがある程度以上（17 インチディスプレイで 0.94 inches 以上）大きい場合は画面端に接するターゲットに、それ以下では画面角に接するターゲットにすべきであることがわかった。

1. はじめに

Windows 10 では画面の左下角にスタートボタンがあったが、Windows 11 では下端中央に変更された（図 1）。ポインティング（ボタンやアイコンなどのターゲットを選択する操作）では、速く正確に選択できることが良しとされるが、画面角や画面端にあるターゲットは誤選択が起きづらく [1]、より速く選択できることが知られている [2]。操作時間に影響を与える主な要因はターゲットまでの距離とターゲットの大きさであり [3]、距離が長く小さいターゲットは操作時間が長くなる。また、ターゲットの配置も操作時間に影響し、Windows 11 のスタートボタンのような画面端に接するターゲット（以下、画面の 1 辺に接するターゲットを *Edge*）はより速く選択できることが知られている [1]。カーソルが画面外に移動せず画面端にとどまるため、微細な制御せずに操作が完了できるからである。また、Windows 10 のスタートボタンのような画面角に接するターゲット（以下、画面の 2 辺に接するターゲットを *Corner*）も操作時間の短縮が望める。画面角、画面端に接する配置それぞれにおけるポインティングのパフォーマンスはこれまでも調査されている [1, 2]。一方で、どのような条件で *Corner* と *Edge* のどちらが良い配置であるかは明らかにされていない。たとえば横長のディスプレイでは、

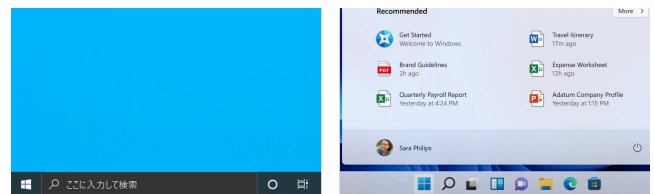


図 1 （左）Windows 10 ではスタートボタンが画面の左下角に接しており、（右）Windows 11 では画面の下端中央に接して配置されている。

Corner は *Edge* よりも画面の各位置からターゲットまでの平均距離が長く、条件によっては *Edge* が平均的に速く選択できるかもしれない。

本研究の目的は、*Corner* と *Edge* の差を明らかにし、たとえば、上述のスタートボタンのようなターゲットをデザインする際の適切なターゲット配置を示すことである。実験参加者がさまざまな条件下で *Corner* と *Edge* を選択するポインティング実験を行い、配置などの要因が操作時間とエラー率に与える影響を調査した。実験の結果、どちらのターゲット配置でもエラー率は低く、*Corner* が *Edge* より操作時間が短いことが示された。また、カーソルの存在確率を一様分布と仮定したとき、ターゲットサイズがある程度以上（17 インチディスプレイで 0.94 inches 以上）大きい場合に *Corner* より *Edge* のほうが短い平均操作時間が観測された。そのターゲットサイズは、いずれも大きめであり、それ以上のサイズのターゲットは *Corner* でも *Edge* でも容易にポインティングできると考えられた。しかし、

¹ 明治大学

² ヤフー株式会社

一般的に画面角は4つあるのみなので、操作時間のみを考慮すれば、そのサイズ以上のターゲットは *Edge* に、そのサイズ以下は *Corner* にすべきであることがわかった。

2. 関連研究

2.1 画面角・画面端に接するターゲットの影響

ターゲットを画面端に接して配置することで操作時間を短縮できることが知られている [1, 2]. 通常のポインティングでは、ターゲット内でカーソルを止める必要があるが、*Edge* の場合では、カーソルが画面端で止まるため、ユーザはカーソルを画面端に衝突させるだけでポインティングを成功させられる。Dizmen らは画面端から 1 pixel だけ離れているターゲットは、接するターゲットに比べて操作時間が遅くなることを示した [1]. また、*Edge* の操作時間はフィットの法則にしたがうことを示した。Appert らは、異なる高さの矩形の *Edge* を比較した [2]. 実験の結果、ターゲットの高さが大きいほど速く選択できることがわかった。また、Appert らは、カーソルのターゲットへの進入角度の影響も調査し、ターゲットが接する画面端に対して垂直方向のときには速く、水平方向のときには遅くなることを示した。また、一般的な矢印のカーソルでは、ターゲットが画面下端より上端に接するほうが速いことも示した。

2.2 ポインティングの時間予測モデル

フィットの法則 [3] は、ターゲットまでの距離 A とターゲットサイズ W から決定される難易度指標 ID (Index of difficulty) から操作時間 MT を予測できる (式 1).

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

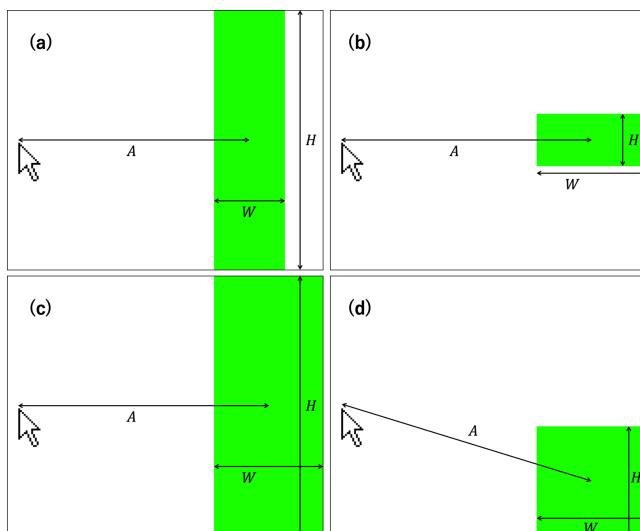


図 2 (a) W が規定され、 H が無限大とみなせるターゲット. (b) W が無限大とみなせ、 H が規定されるターゲット (*Edge*). (c) (a) のターゲットが画面端に接し、 W と H がともに無限大とみなせるターゲット. (d) 画面角に接し、 W と H がともに無限大とみなせるターゲット (*Corner*).

ここで、 a と b は回帰定数である。以降、 a, b, c, d は回帰定数である。フィットの法則は、図 2 (a) や図 2 (b) (この場合は式 1 の W が H) に適用できる [1].

フィットの法則はターゲットに対するカーソルの進入角度を考慮していない。図 2 (b) のようなターゲットに対して、左から右に進入する場合、進入角度に対するターゲットサイズは W と H である。このとき、カーソルは画面右端で静止するため、 W は無限大であるとみなせる。一方で、左上からターゲットに進入する場合、進入角度に平行なターゲットサイズは W ではなく、無限大でもない。そのため、左から右に進入する場合に比べ、左上からの方がポインティングの難易度が高くなると考えられる。Appert らはターゲットへの進入角度 $Angle$ を考慮できる ID を提案した [2] (式 2).

$$ID = \log_2 \left(\frac{A}{W} + \frac{A}{H} + c \cdot \sin(|Angle|) \frac{A}{\min(W, H)} + 1 \right) \quad (2)$$

ここで $Angle$ は、図 2 (b) のようにターゲットが接している画面端に対して垂直に進入するとき 0° である。式 2 の $A/W + A/H$ は Accot と Zhai の矩形ターゲットのモデル [4] から導出されている。そのため、 W と H が等しい場合には、 $A/W + A/H$ をフィットの法則の A/W に置き換えるのが適切であると考えられる。つまり、 W と H が等しい場合、式 2 は次のように表せる。

$$ID = \log_2 \left(\frac{A}{W} + c \cdot \sin(|Angle|) \frac{A}{W} + 1 \right) \quad (3)$$

式 1 における ID が一定以下の場合、 MT は A にのみ依存することが知られている [5]. このとき、式 1 の ID は次のように修正される。

$$ID = \sqrt{A} \quad (4)$$

図 2 (a) のターゲットが画面端に接する場合 (図 2 (c)), W と H がともに無限大とみなせ、式 1 で計算される ID よりも操作が簡単であり、適切に MT を予測できない [6]. また、図 2 (b) のターゲットが画面角に接する場合 (図 2 (d)) も W と H がともに無限大とみなせ、同様に MT を予測できないと考えられる。

3. 実験

3.1 機材

PC (mouse, G-Tune, P870DM, Intel Core i7-6700, 3.4 GHz, 16.0 GB RAM, Windows 10 Home), 外部ディスプレイ (LG Electronics, 43UD79-B, 42.5 inches, 2560×1440 pixels, 平面), 有線マウス (Logicool G502WL, 400 DPI) を使用した。参加者の普段の設定とあわせるため、「ポインタの精度を高める」をオンにし、カーソル速度はコントロールパネルにおけるスライダーの中間とした。システムは Processing で実装され、フルスクリーンで表示された。計測値は 60 Hz で記録された。

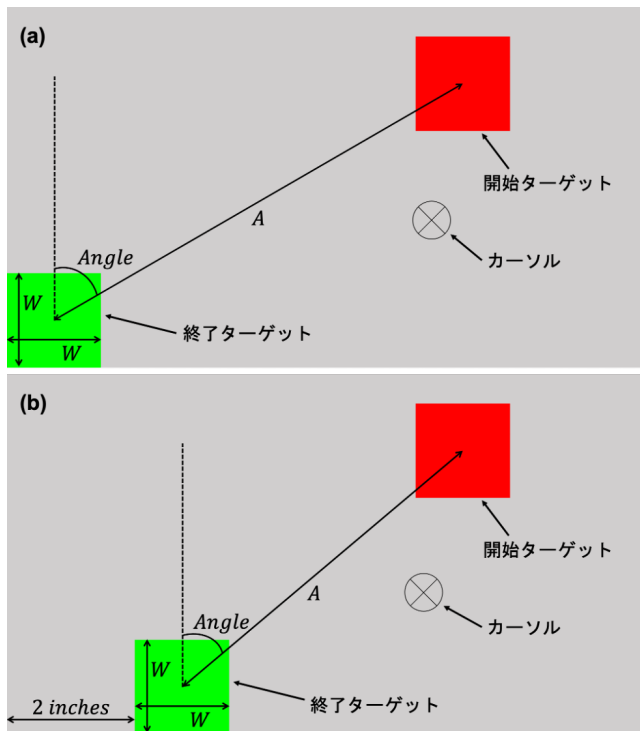


図 3 タスクの概要図。(a) 終了ターゲットが画面左下角に接する配置 (Corner 条件)。(b) 終了ターゲットが画面下端に接する配置 (Edge 条件)。

3.2 参加者

参加者は情報系の大学生 12 名 (平均 22.6 歳, 標準偏差 1.56 歳) であった。参加者全員がマウス操作に習熟しており, 利き手の右手で操作を行った。

3.3 タスク

灰色の背景に赤色の開始ターゲットと緑色の終了ターゲットが表示された (図 3)。参加者は開始ターゲットをクリックすることで試行を開始させ, また, 開始音と開始ターゲットの消滅が試行開始のフィードバックとして提示された。終了ターゲットをできるかぎり速く正確に目指し, 次のクリックによってその試行を終了した。クリック位置が終了ターゲット内であった場合, その試行を成功とし, それ以外は失敗 (エラー) とした。また, 試行の成否に対応した音が再生された。

開始ターゲットか終了ターゲット内にカーソルがある場合, そのターゲットの色は黄色に変化した。これは, 実際の GUI (Graphical User Interface) 環境において, カーソルがターゲット上にあるとき, ターゲットの色が変化することから, 本実験においても変化させるようにした。

3.4 デザイン

開始ターゲットと終了ターゲットはどちらも正方形であり, その大きさ $W (= H)$ は 0.25, 0.50, 0.75, 1.0 inches (17, 35, 52, 69 pixels) の 4 条件とした。 W は, Windows

10 のディスプレイ設定の「拡大縮小とレイアウト」のうち, 「表示スケール」と「ディスプレイの解像度」を「推奨」にしている場合の実寸のターゲットサイズを参考にした。Windows 10 のタスクバーの右下の音量ボタンの幅が約 0.25 inches, 左下のスタートボタンの幅が約 0.5 inches, デスクトップのショートカットアイコンの幅が約 1.0 inches であったため, 0.75 inches を追加した 4 条件を W とした。

開始ターゲットの中心から終了ターゲットの中心までの距離 A は 3, 10, 17 inches (207, 690, 1173 pixels) の 3 条件であった。 A は, 式 1 における ID の間隔がおおよそ一定になるように設定した。また, 本実験において 42.5 inches の大型ディスプレイを用いたのは, A の条件の最大値である 17 inches を x 軸方向と y 軸方向に確保するためである。

終了ターゲットの配置 $Position$ は, 画面左下角に接する配置 (Corner 条件), 画面下端に接する配置 (Edge 条件) の 2 条件とした。Edge では, 終了ターゲットは画面左端から 2 inches (138 pixels) 離れていた。ターゲットが画面の一辺から 1 pixel 以上離れていれば接するターゲットと比べポインティングの性能が下がることが知られているが [1], 参加者が Edge 条件であることを認識しやすくするために, また, ディスプレイ内にターゲットを収めるために, 2 inches とした。

終了ターゲットへの進入角度 $Angle$ は, y 軸の上方向から時計回りに $0^\circ, 30^\circ, 60^\circ, 90^\circ$ の 4 条件とした。 $Angle$ に負の角度を加えなかったのは, 基準の軸に対して線対称な進入角度で同様の影響がある [2] ためである。

一般的な矢印のカーソルでは, 画面下端のターゲットを狙う場合, 環状型のカーソルに比べ, 操作時間が増加することが知られている [2]。これは, 画面下端に矢印のカーソルのホットスポット (クリックが行われる場所) が位置した場合, カーソルのほとんどが画面外に出てしまい, カーソルの位置が認識しづらくなるためである。本実験では, 先行研究 [7] を参考に, 長さ 0.125 inches の線を X 字に交差させそれを円で囲ったカーソルを用いた。このカーソルは, ホットスポットやカーソルの位置を認識しやすく, 操作時間の増加を防げると考えられる。

3.5 手順

参加者は 1 セット ($3A \times 4W \times 2Position \times 4Angle = 96$ 試行) の練習を行い, 1 分間の休憩後, 10 セットの本番を行った。本番 5 セット後に 2 分間の休憩を設けた。1 セット内における条件はランダムに提示された。実験には参加者 1 名あたり約 40 分を要した。参加者には, (1) できるかぎり速く正確にタスクを行うこと, (2) クラッチ (操作の途中でマウスを浮かす行為) をできるかぎり行わないこと, (3) 開始のフィードバックを確認してからカーソルの操作を行うこと, (4) 終了ターゲット内にカーソルが位置

していることを確認してから選択を行うことを指示した。クラッチはフィットの法則のモデル適合度を下げることが知られている [8]。モデル適合度への影響を実験条件のみにするため、本実験ではクラッチをできるかぎり行わないように指示した。ただ、本実験の A 条件はクラッチを行わずに十分カーソルを移動させられる距離であったため、クラッチを制限した影響はほとんどないと考えられる。

3.6 計測値

開始ターゲットをクリックしてから終了ターゲットをクリックするまでの操作時間 MT 、終了ターゲット外をクリックが行われた割合（エラー率）を計測した。

4. 結果

11,520 回 ($3A \times 4W \times 2Position \times 4Angle \times 10$ セット $\times 12$ 名) の試行が計測された。分析には繰り返しのある分散分析、多重比較には Bonferroni の方法を用いた。独立変数は A , W , $Position$, $Angle$ 、従属変数は操作時間 MT 、エラー率 ER であった。グラフ中のエラーバーは標準誤差、***, **, * はそれぞれ $p < 0.001$, $p < 0.01$, $p < 0.05$ を示す。分析には IBM SPSS Statistics 27 を用いた。

4.1 エラー率 ER

エラーとなった試行は 256 回 (2.22%) であった。主効果が見られたのは W ($F_{3,33} = 5.27$, $p < 0.01$, $\eta_p^2 = 0.324$), $Position$ ($F_{1,11} = 10.8$, $p < 0.01$, $\eta_p^2 = 0.497$, 図 4 (a)) であった。 A ($F_{2,22} = 1.71$, $p = 0.204$, $\eta_p^2 = 0.134$), $Angle$ ($F_{3,33} = 0.170$, $p = 0.916$, $\eta_p^2 = 0.0152$) では見られなかった。 $Position$ において有意差が見られたが、どちらも低い ER であった。

4.2 操作時間 MT

エラーを除いた試行は 11,264 回であった。分散分析の結果を表 1、多重比較の結果を図 4 (b) に示す。 $Position$ 間で差が見られ、 $Corner$ では、 $Edge$ よりも小さい MT が観測された。また、交互作用から $Corner$, $Edge$ それぞれにおいて、 A , W , $Angle$ が MT に対して有意に影響を与えることがわかった。交互作用はみられたものの、各条件が MT に与える影響は、主効果とほぼ同じ傾向であった (たとえば、 $Position$ 条件では、常に $Corner$ が $Edge$ より小さかった)。

4.3 画面全体に対する平均操作時間

図 4 に示されるように、 ER はどちらのターゲット配置条件でも低かったが、 $Corner$ では $Edge$ よりも短い MT が観測された。一方で、横長のディスプレイでは、 $Corner$ よりも $Edge$ の方がターゲットまでの距離は平均的に短いため、画面の各位置からターゲットまでの操作時間を調べ

表 1 操作時間 MT の分散分析の結果.

Factors	DF	DFDen	F	p	η_p^2
A	2	22	330	< 0.001	0.968
W	3	33	179	< 0.001	0.942
Position	1	11	36.9	< 0.001	0.770
Angle	3	33	15.2	< 0.001	0.580
A × W	6	66	0.471	0.828	0.0410
A × Position	2	22	62.5	< 0.001	0.850
A × Angle	6	66	4.85	< 0.001	0.306
W × Position	3	33	132	< 0.001	0.923
W × Angle	9	99	2.39	< 0.05	0.178
Position × Angle	3	33	7.90	< 0.001	0.418
A × W × Position	6	66	1.04	0.406	0.0866
A × W × Angle	18	198	0.704	0.804	0.0602
A × Position × Angle	6	66	2.30	< 0.05	0.173
W × Position × Angle	9	99	2.02	< 0.05	0.155
A × W × Position × Angle	18	198	1.42	0.124	0.144

た場合、 $Corner$ よりも $Edge$ の方が短い平均 MT を観測する可能性がある。また、ターゲットサイズがある程度大きい場合には、どちらの $Position$ 条件においてもポインティングが容易であり、どちらの条件においても同等の MT が観測されるだろう。

この節では、カーソルの存在確率を一様分布と仮定して、操作時間予測モデルから平均操作時間を求めた。これにより、実際の UI デザインを行う際に $Corner$ にすべきか、 $Edge$ にすべきかを判断できると考えられる。

4.3.1 モデル選択

$Position$ において有意差が見られたので、 $Corner$ と $Edge$ それぞれ、32 (= $3A \times 4W \times 4Angle$) 点から MT を予測できるモデルを構築する。

$Edge$ は、式 3 で MT を予測できることが知られている [2]。適合度を調べたところ、 $R^2 = 0.975$ で高い値を示した (表 3 (行 1))。そのため、 $Edge$ のモデルには式 3 を用いる。

$Corner$ では、 A , W , $Angle$ が MT に影響を与えることがわかった (図 5)。しかし、 $Corner$ は W が無限大とみなせるため、式 1 や式 3 では適切に MT を予測できないと考えられる。適合度はそれぞれ $R^2 = 0.817$, $R^2 = 0.821$ であり、Soukoreff の閾値 ($R^2 < 0.9$ [9]) より低い値を示した (表 3 (行 2, 3))。 W が無限大とみなせる場合、式 1 における ID が極端に小さくなるため、 MT が A のみに依存する式 4 で予測できると考えた。適合度を調べたところ、 $R^2 = 0.872$ であった (表 3 (行 4))。また、 $Angle$ ごとに式 4 に適合度を調べたところ、 $Angle = 90^\circ$ のとき $R^2 = 0.882$ であり、他条件では $R^2 > 0.9$ であった (表 2)。

表 2 $a + b \cdot \sqrt{A}$ に対する $Corner$ の $Angle$ ごとの適合度.

Angle	a	b	R^2
0°	389	138	0.920
30°	407	113	0.964
60°	429	108	0.913
90°	395	133	0.882

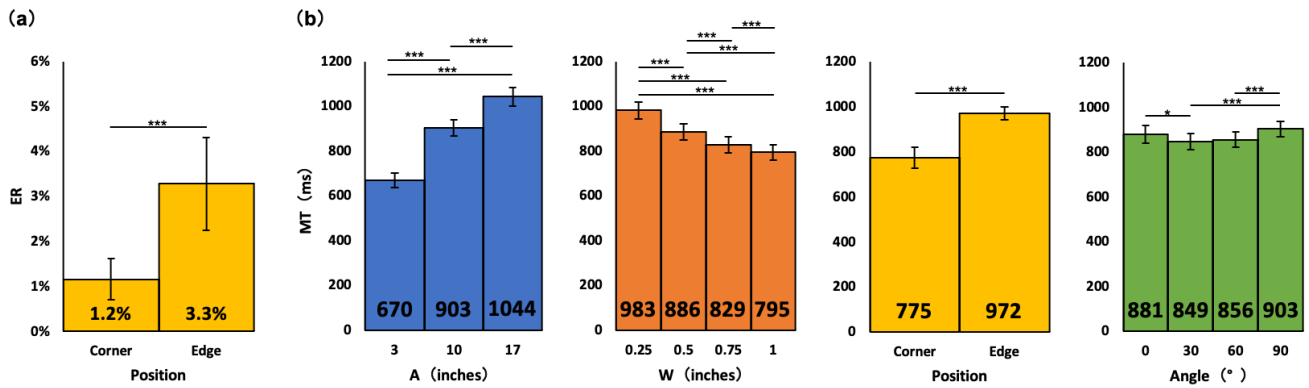


図 4 (a) エラー率 ER に対する $Position$ の影響. (b) 操作時間 MT に対する A , W , $Position$, $Angle$ の影響.

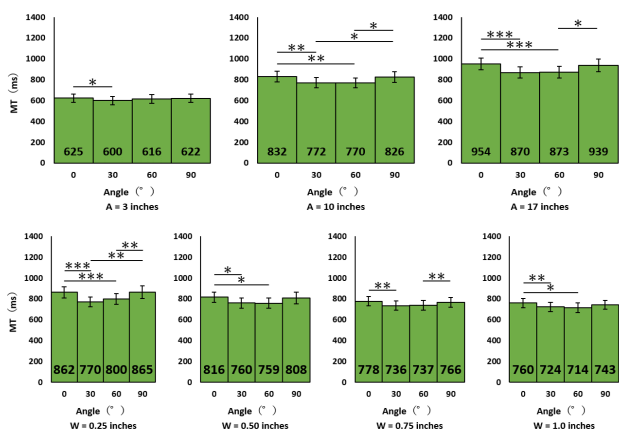


図 5 操作時間 MT に対する $Corner$ 条件における $A \times Angle$ (上) $W \times Angle$ (下) の影響.

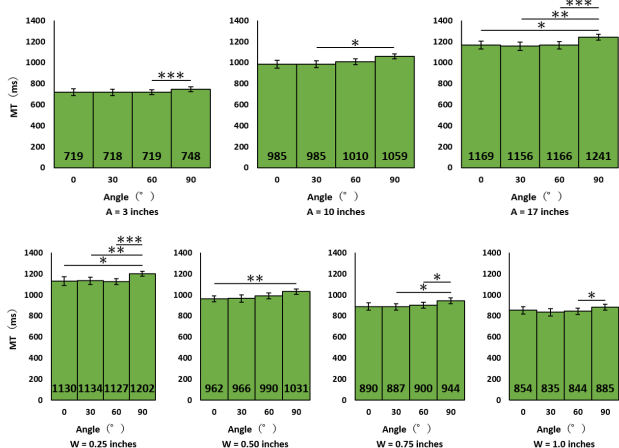


図 6 操作時間 MT に対する $Edge$ 条件における $A \times Angle$ (上), $W \times Angle$ (下) の影響.

W も MT に対して影響を与えていたが、式 4 がほとんどの $Angle$ 条件で MT を予測できたため、式 4 をもとに $Angle$ を考慮できるモデルを構築した。Edge の $Angle$ を考慮できるモデル (式 3) は、式 1 をもととし、 $Angle$ の項が追加されたものである。式 4 に対して式 2 と同様に

$Angle$ の項を追加すると、モデルは次のようになる。

$$ID = \sqrt{A + c \cdot \sin(|Angle|)} \cdot A \quad (5)$$

このモデルは $R^2 = 0.877$ を示した (表 3 (行 5))。 MT が最小である $Angle$ が A と W の条件によって異なるため (図 5 (左)), 式 5 の $Angle$ に始点を追加した (式 6)。

$$ID = \sqrt{A + c \cdot \sin(d + |Angle|)} \cdot A \quad (6)$$

適合度を調べたところ、 $R^2 = 0.918$ であった (表 3 (行 6))。モデル適合度は定数が多いほど適合度が高くなるため、定数の数を考慮できる赤池情報量基準 (AIC) [10] もモデルの選択基準に加えた (表 3 (列 8))。 AIC が低く、 R^2 が高いモデルが良いモデルとされる。式 6 が最も高い R^2 と最も低い AIC を示した。そのため、 $Corner$ のモデルには式 6 を用いる。

4.3.2 平均操作時間の予測

$Corner$ は左下角に終了ターゲットがあるとして、また、 $Edge$ は画面下端中央に終了ターゲットがあるとして MT を予測した。Windows 11 のスタートボタンの配置 (図 1) を参考に、 $Edge$ は画面下端中央とした。実験では、 $Edge$ は画面左端から 2 inches 離れた位置であったが、モデル (式 3) より画面端からターゲットまでの距離は MT に影響しないと考えられるため、画面下端中央の予測にも問題なく適用できると考えられる。開始ターゲットは各ピクセルであり、各ピクセルから終了ターゲットの中心が A である。ポインティングの MT は反応時間を下回らないと考えられるため、また、あまりにも距離が短い場合に誤った予測 MT が計算される可能性があるため、モデルから予測された MT と参加者の反応時間を比較し、大きいほうをその条件での MT とした。参加者の反応時間は、本実験のタスクにおいて、最後に終了ターゲット内にカーソルが入ってからクリックを行うまでの時間とした。

全参加者の反応時間の平均を求めたところ、252 ms であった。平均 MT は 0.01 inches 刻み (W) で予測された。本実験の A の最大値が 17 inches であり、モデルから予測

表 3 モデルの定数の推定値と適合度.

行	Position	モデル	a	b	c	d	R^2	AIC
1	Edge	$a + b \cdot \log_2 \left(\frac{A}{W} + c \cdot \sin(Angle) \cdot \frac{A}{W} + 1 \right)$	217	183	0.196		0.975	
2	Corner	$a + b \cdot \log_2 \left(\frac{A}{W} + 1 \right)$	379	99.2			0.817	262
3	Corner	$a + b \cdot \log_2 \left(\frac{A}{W} + c \cdot \sin(Angle) \cdot \frac{A}{W} + 1 \right)$	379	99.2	0.000		0.817	264
4	Corner	$a + b \cdot \sqrt{A}$	405	123			0.872	254
5	Corner	$a + b \cdot \sqrt{A + c \cdot \sin(Angle) \cdot A}$	405	127	-0.107		0.877	255
6	Corner	$a + b \cdot \sqrt{A + c \cdot \sin(d + Angle) \cdot A}$	405	171	-0.570	43.7	0.918	249

される MT の正確さを保証するため, 13, 15, 17 インチのディスプレイを想定して予測を行った. ディスプレイの解像度は計算の便宜上, フル HD (1920 × 1080) とした. 予測された *Corner* と *Edge* の平均 MT , *Corner* より *Edge* の平均 MT が小さくなる W , その W における *Edge* の平均 MT を図 7 に示す.

5. 議論

5.1 画面全体に対する平均操作時間

確率分布と操作時間予測モデルから画面全体に対する平均 MT を求めた結果, W がある程度大きい場合に画面角 (*Corner*) よりも画面端 (*Edge*) のほうが短い平均 MT が観測された. 予測においてディスプレイの解像度は, あまりに解像度が低い場合を除いて, 予測の結果に著しく影響しないと考えられ, 本予測の結果で議論できると考えられる. *Corner* より *Edge* の平均 MT が短くなると予測された W は, いずれも Windows のデスクトップのアイコンのサイズに近く, ターゲットサイズとしては大きめである. つまり, その W 以上のターゲットはどちらの *Position* 条件でも容易に選択できるため, *Corner* でも *Edge* でも良いだろう. しかし, 一般的に画面角は 4 つのみであるので, 操作時間を考慮すると, その W 以上のターゲット (たとえば, 上述したデスクトップのアイコンなど) は *Edge* に, その W 以下は *Corner* にすると良いと考えられる.

Windows のスタートボタンは約 0.5 inches である. ディスプレイサイズが小さくなるほど予測される W は小さくなるが (図 7), Windows のスタートボタンよりも予測される W が小さいディスプレイサイズは 4.0 インチ以下であるので, ほとんどの場合において, *Corner* であるべきと考えられる.

5.2 画面角に接するターゲットの操作時間予測モデル

提案した *Corner* の操作時間予測モデル (式 6) は, W を考慮していない. 本研究の目的は, *Corner* と *Edge* の差を明らかにすることであるため, モデルの精度が多少不十分であっても達成されただろう. しかし, *Corner* の操作時間をより正確に予測したい場合には, W などのターゲットの大きさを考慮したモデルを構築すべきである.

6. 制約・展望

6.1 カーソルの確率分布

本研究では, カーソルの存在確率が一様分布であると仮定して分析を行った. しかし, 実際には, 行っている作業 (アプリケーション) によってカーソルの確率分布は変化すると考えられる [11]. たとえば, カーソルの確率分布を調べ, *Edge* 付近にカーソルがよく存在する場合, *Corner* よりも *Edge* のほうが短い平均 MT になるだろう. また, 実際のカーソルの確率分布から調査する場合, 主要なターゲット (たとえば, Windows のスタートボタン) が, *Corner* であるか *Edge* であるかも影響すると考えられる. そのため, 主要なターゲットを *Corner* にした場合と, *Edge* にした場合のそれぞれで十分な時間で計測する必要があるだろう. 今後は, このような方法で実際のカーソルの確率分布を調べ, より現実的な状況を想定した分析を行っていきたい.

6.2 接する画面端の影響

画面全体に対する平均 MT を求めるにあたって, *Edge* は画面下端中央に接する配置とした. 一般的に用いられる矢印のカーソルでは画面上端のほうが下端に接するターゲットより速く選択できること, また, 画面端に接しないターゲットのポインティングは垂直方向より水平方向のほうが速いことが示されている [2]. そのため, ターゲットの接する画面端によっても画面全体に対する平均 MT に影響があると考えられる. たとえば, 画面上端に接する場合, *Corner* より *Edge* の平均 MT が短くなると予測される W が小さくなるだろう. また, *Edge* が画面角に近い場所にある場合, 画面内の各位置からの平均距離が長くなるため, 予測される W が大きくなるだろう.

6.3 ターゲットが矩形である場合

本実験においてターゲットは正方形であったが, 実際の GUI 環境におけるターゲットは矩形であることが多い. 提案した *Corner* の操作時間予測モデル (式 6) は, ターゲットの大きさを考慮していないため, 矩形のターゲットにおける平均 MT の予測もできると考えられる. しかし, *Edge* は高さが大きいほど速く選択できることが示されて

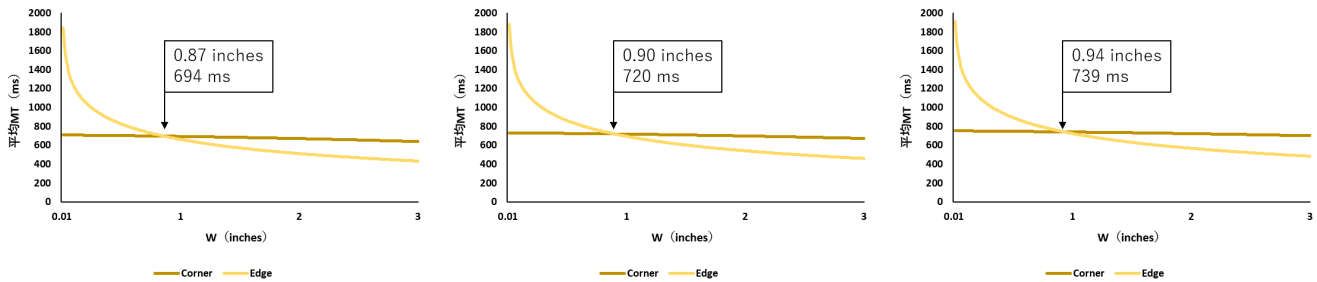


図 7 カーソルの確率分布が一様分布と仮定したときの平均 MT に対する W の影響。(左) 13 インチ, (中央) 15 インチ, (右) 17 インチのフル HD ディスプレイで予測。グラフ中の数値は Corner より Edge の平均 MT が小さくなる W, その W における Edge の平均 MT を示す。

いる [2]。Edge の W と H が平均 MT に影響するため、より現実に近い予測を行う場合、矩形のターゲットを考慮し、式 2 を用いて予測を行うべきである。

6.4 操作時間以外を考慮した場合

Edge が Corner より有意に ER が高いことが示された。これは、Edge はターゲットの高さを無限大とみなせるのに対して、Corner はターゲットの幅と高さがともに無限大とみなせるため、Edge よりもカーソルの微細な制御なしで正確にポインティングできたためであると考えられる。本研究では、操作時間のみからどちらのターゲット配置が良いのかの分析を行ったが、もし、ポインティングのパフォーマンスを統合的に予測できるモデルがあれば、予測される W の値は変化する可能性がある。

Microsoft の Baca らは、Windows が巨大ディスプレイでも使用されるようになり、そのようなディスプレイ上では画面角のターゲットは平均的に距離が遠いため、Windows 11 のメニューは画面端の中央に配置したと述べている [12]。Microsoft はカーソルを長距離移動させることによる疲労や、画面角のターゲットの認知のしづらさなどを考慮したのでだろう。考慮する項目を増やした場合、本研究と異なる結果が得られる可能性があり、操作時間のみを考慮したことが本研究の制約といえる。

6.5 画面角のターゲットの操作時間

本実験で得られた Corner の操作時間は Gan らが得た操作時間よりも平均的に長い傾向にあった [5]。本研究の予備実験において、終了ターゲット内にカーソルが入っているかどうかを確認しない操作が多くみられた。そのため、本実験では「終了ターゲット内にカーソルが入っていることを確認してから選択を行うこと」を教示した。参加者がこの教示に従い、カーソルが終了ターゲット内に入っていることを慎重に確認したため、本実験の Corner の操作時間は平均的に長くなったと考えられる。しかし、この教示による影響は操作時間に含まれる参加者の反応時間が増大

するだけに留まるだろう。そのため、参加者がタスクに熟達し、操作時間が短縮された場合でも、式 6 で Corner の操作時間を予測できると考えられる。

7. 結論

本研究では、画面角、画面端に接するターゲットのどちらが、どのような条件下において良い配置となるかを明らかにするため、画面角に接するターゲット (Corner) と画面端に接するターゲット (Edge) の配置の差が、画面全体に対する平均操作時間に与える影響を調査した。実験の結果、エラー率はどちらのターゲット配置条件でも低かったが、Corner が Edge よりも操作時間が短いことが示された。また、カーソルの存在確率を一様分布と仮定して、操作時間予測モデルから平均操作時間を求めた結果、ターゲットサイズがある程度以上 (17 インチディスプレイで 0.94 inches 以上) 大きい場合に Corner より Edge のほうが短かった。そのターゲットサイズは、いずれも大きめであり、それ以上のサイズのターゲットは Corner でも Edge でも容易にポインティングできると考えられた。しかし、一般的に画面角は 4 つあるのみなので、操作時間のみを考慮すれば、そのサイズ以上のターゲットは Edge に、そのサイズ以下は Corner にすべきであることがわかった。

参考文献

- [1] Dizmen, C., Hoffmann, E. R. and Chan, A. H.: Movement time to edge and non-edge targets, *Ergonomics*, DOI: 10.1080/00140139.2013.855824 (2014).
- [2] Appert, C., Chapuis, O. and Beaudouin-Lafon, M.: Evaluation of Pointing Performance on Screen Edges, *Proceedings of the Working Conference on Advanced Visual Interfaces, AVI '08*, New York, NY, USA, DOI: 10.1145/1385569.1385590 (2008).
- [3] Fitts, P.: The information capacity of the human motor system in controlling the amplitude of movement., *Journal of experimental psychology*, Vol. 47 6, pp. 381–91 (1954).
- [4] Accot, J. and Zhai, S.: Refining Fitts' Law Models for Bivariate Pointing, *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI

- '03, New York, NY, USA, DOI: 10.1145/642611.642646 (2003).
- [5] GAN, K.-C. and HOFFMANN, E. R.: Geometrical conditions for ballistic and visually controlled movements, *Ergonomics*, DOI: 10.1080/00140138808966724 (1988).
- [6] 山中祥太, 宮下芳明: 無限大のサイズを持つターゲットのポインティングに関する調査, *情報処理学会論文誌*, Vol. 57, No. 4, pp. 1319–1329 (2016).
- [7] Yamanaka, S.: Mouse Cursor Movements towards Targets on the Same Screen Edge, *Proceedings of the 44th Graphics Interface Conference, GI '18*, Waterloo, CAN, DOI: 10.20380/GI2018.16 (2018).
- [8] Casiez, G., Vogel, D., Pan, Q. and Chaillou, C.: RubberEdge: Reducing Clutching by Combining Position and Rate Control with Elastic Feedback, *Proceedings of the 20th Annual ACM Symposium on User Interface Software and Technology, UIST '07*, New York, NY, USA, DOI: 10.1145/1294211.1294234 (2007).
- [9] 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 (オンライン), DOI: <https://doi.org/10.1016/j.ijhcs.2004.09.001> (2004).
- [10] Akaike, H.: A new look at the statistical model identification, *IEEE Transactions on Automatic Control*, DOI: 10.1109/TAC.1974.1100705 (1974).
- [11] Chapuis, O., Blanch, R. and Beaudouin-Lafon, M.: Fitts' Law in the Wild: A Field Study of Aimed Movements, Technical report (2007).
- [12] JOURNAL, T. W. S.: Why Microsoft Centered the Start Button in Windows 11, <https://www.wsj.com/articles/why-microsoft-centered-the-start-button-in-windows-11-11627038002> (2021).