

# カーソルが進入できないノッチが ポインティングの操作時間に与える影響

大場洋介<sup>1</sup> 宮下芳明<sup>1</sup>

**概要：**MacBook Pro (2021) 画面上部のノッチ領域には、カーソルが進入できる。しかし、カーソルの一部や全部が隠れてしまう。この影響でノッチが操作時間を増加させることを、筆者らは先行研究で明らかにした。そして本稿では、ノッチを「カーソルが進入できない領域」とした方が良いと提言する。ノッチの外縁でカーソルが留まるため、ノッチと隣接したターゲットを高速に選択できると考えられるからである。実験1では、現行のノッチの仕様で、「ノッチに進入する戦略」よりも「ノッチを回避する戦略」が望ましいことを示した。実験2では、ノッチを「カーソルが進入できない領域」に変更した場合、ノッチと隣接するターゲットを速く正確に選択できるようになることを示した。

## 1. はじめに

MacBook Pro (2021) のディスプレイ上端中央に配置されたノッチ（描画が行われない黒い領域）には、カーソルが進入できる（図1）。しかし、カーソルが進入した際に、カーソルの一部や全部がノッチによって隠されてしまう。

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）は、速く正確に行えることが良しとされる。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 [1]、距離が長く小さいターゲットは操作時間が長くなる。著者らの先行研究によって、ノッチがカーソルを隠してしまうことも操作時間を増加させる要因となることが示された [2]。

本稿では、ノッチを「カーソルが進入できない領域」とした方が良いと提言する。ノッチの外縁でカーソルが留まるため、ノッチと隣接したターゲットを高速に選択できると考えられるからである（図2）。

著者らの先行研究において、現行のノッチの仕様（カーソルが進入できる）に対して、参加者は主に2種類の戦略で操作を行った [2]。画面上端にカーソルを沿わせて「ノッチに進入する戦略」と、「ノッチを回避する戦略」であった。そこで本稿の実験1では、現行のノッチの仕様で、上記のどちらの戦略が望ましいかを調査した。参加者がさまざまな条件下でターゲットを選択するポインティング実験を行い、ノッチとターゲットの距離、ターゲットを目指す戦略の要因が操作時間とエラー率に与える影響を調査した。結果、「ノッチに進入する戦略」よりも「ノッチを回避する戦



図1 MacBook Pro (2021) に搭載されているノッチ。  
図の黒い領域では、画面の描画が行われない。  
カーソルが進入できるため、カーソルの一部や全部が隠される。



図2 ノッチを「カーソルが進入できない領域」にした場合に  
予想されるカーソルの移動経路。

略」の方が操作時間が短いことがわかった。

ノッチを回避することが望ましいと考えられたため、ノッチをカーソルが進入できない領域に変更することを検討できる。そこで実験2では、ノッチを「カーソルが進入できない領域」に変更して実験を行った。結果、ノッチを「カーソルが進入できない領域」に変更した場合、ノッチと隣接するターゲットを速く正確に選択できるようになることがわかった。そのため、ノッチをディスプレイに配置する場合、ノッチを「カーソルが進入できない領域」に設計することが望ましいと考えられた。

<sup>1</sup> 明治大学

## 2. 関連研究

### 2.1 ノッチによる影響

著者らは、画面上端のターゲットから同じく画面上端の他のターゲットを選択する場合、ノッチが操作時間を増加させることを示した [2]。また、主にターゲット間にノッチがある条件で、操作時間が増加していた。原因として、参加者がマウス操作で行った、主な2つの戦略が影響したと考えられた。画面上端にカーソルを沿わせる戦略では、ノッチ内でカーソルの位置が分からなくなり、ノッチを回避する戦略では、経路が長くなることが影響したとされた。

### 2.2 ポインティングの操作時間予測モデル

フィッツの法則 [1] は、ターゲットまでの距離  $A$  とターゲット幅  $W$  から決定される難易度指標  $ID$  (Index of difficulty) によって操作時間  $MT$  を予測できる (式 1)。

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

ここで、 $a$  と  $b$  は回帰定数である。以降、 $a$ ,  $b$ ,  $b_1$ ,  $b_2$ ,  $c$  は回帰定数である。

Jax らは、ターゲットまでの経路上に障害物があり、障害物を回避して操作した場合の操作時間を予測するモデルを提案した [3]。Vaughan らは、Jax らのモデルに対して、回帰定数が増えていることを指摘した [4]。また、Yamanaka らが Vaughan らの提案したモデルを式 1 と対応させる修正を行った (式 2) [5]。

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

式 2 における  $B$  は、ターゲットまでの直線経路に対して垂直な、障害物を回避するために必要な最小限の距離を示す。Yamanaka らはオブジェクト間を通過する操作時間は Jax らのモデルや、式 2 で式 1 より大幅に正確に予測できることを示した [5]。また、Yamanaka らの実験で、障害物が経路上にはないが経路付近にある場合、ユーザは障害物を少し避けるように経路を曲げることがあった。しかし、操作時間への影響は小さく、式 1 で十分予測が可能であることを示した。

ターゲットの高さ  $H$  を考慮した 2 次元タスクのモデルを、Accot と Zhai [6], Zhang ら [7] が提案した。Yamanaka は、この 2 次元タスクのモデルが、画面端のターゲットから同じ画面端の他のターゲットを選択する操作時間を予測できることを示した [8]。また、Usuba らは、ターゲットと障害物の間隔  $I$  を考慮したモデルを提案した [9]。

著者らは、画面上端のターゲットから同じく画面上端の他のターゲットをポインティングする時に、ノッチが配置されていることを考慮したモデルを提案した (式 3) [2]。

$$MT = a + b_1 \log_2 \left( \sqrt{c \left( \frac{A}{W} \right)^2 + (1-c) \left( \frac{A}{H} \right)^2} + 1 \right) + b_2 \log_2 \left( \frac{1}{I + 0.0049} + 1 \right) \quad (3)$$

式 3 は、2 次元タスクのモデルと、ターゲットと障害物の間隔  $I$  を考慮したモデルを足し算することで導出された。なお式 3 において  $I$  は、ノッチとターゲットの間隔である。

### 2.3 画面端・画面角による影響

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 [10–12]。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。しかし、画面端に接するターゲットの場合、カーソルが画面端で留まるため、画面端に対して水平方向のみ調整を行えば選択できる。さらに、画面角に接するターゲットの場合、カーソルを画面角に衝突させるだけで選択できるため、高速に選択できる [13]。

### 2.4 Path Efficiency

画面端でのポインティング動作では、カーソルが画面端で留まることを利用して、微細な制御をせずに選択が完了できる [10–12]。しかし、余分に画面端にカーソルを押し込む動作 (pushing-edge) を行うことは、マウスの移動距離を増加させ、操作時間を増加させる可能性がある。Yamanaka は、カーソルの経路の効率性を計算するため、 $PE$  (Path Efficiency) を定義した (式 4) [8]。 $PE$  は、on-screen travel distance (画面上でカーソルが移動した距離) と virtual travel distance (画面端でカーソルが留まることを無視した、仮想のカーソルの移動距離) で計算される。

$$PE = \frac{\text{on-screen travel distance}}{\text{virtual travel distance}} \times 100\% \quad (4)$$

$PE$  が低い値の場合、ユーザは pushing-edge を行っており、カーソルの経路の効率性が低いことを示す。 $PE$  が高いほど、ユーザは画面外への移動を短くすることに注力していることを示すが、画面端でカーソルが留まることを利用していない可能性がある。よって、より慎重なマウス操作が行われているかもしれない。そのため、 $PE$  が 100% に近いほど、操作時間が短縮されているとは限らないことに注意されたい。また、本稿ではカーソルが進入できないノッチに、カーソルを押し込む動作が行われる可能性がある。その場合も  $PE$  が低い値を示すことに注意されたい。

## 3. 実験 1

PC (Intel Core i9-12900KF, GeForce RTX 3070 Ti, 32GB RAM, Windows 10 Home), 外部ディスプレイ (AOPEN, 25XV2QFbmiiprx, 24.5 inches, 1920×1080 pixels), 有線マウス (Logicool, G300s, 1600 dpi) を使用し

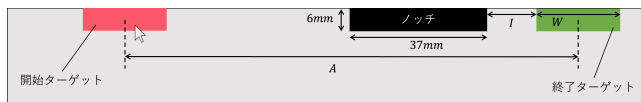


図 3 タスクの概要図。A は開始ターゲットの中心から終了ターゲットの中心までの距離、W は終了ターゲットの幅、I はノッチと終了ターゲットの距離を示す。

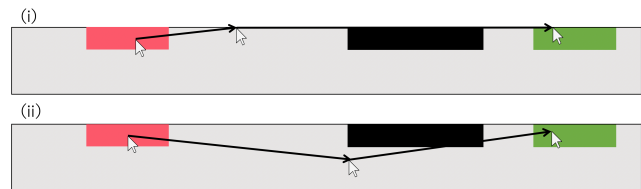


図 4 Strategy の概要図。(i) 画面端にカーソルを沿わせる戦略 *along*。(ii) ノッチを回避する戦略 *avoid*。

た。参加者の普段の設定とあわせるため、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、「ポインターの精度を高める」をオンにした。システムは HSP 3.6 で実装し、フルスクリーンで表示した。ディスプレイのリフレッシュレートは 360 Hz であった。

参加者は情報系の大学生 12 名（平均 22.3 歳，標準偏差 1.67 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。

タスクの概要図を図 3 に示す。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。参加者が開始ターゲットをクリックすると，試行が開始された。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与えられた。参加者は終了ターゲットをできるだけ速く正確に目指し，次のクリックによって試行を終了した。このクリックが終了ターゲット内で行われた場合，その試行を成功とし，それ以外は失敗（エラー）とした。試行の成否に対応した音のフィードバックが提示された。

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  $MT$ ，終了ターゲット外でクリックが行われた割合（エラー率  $ER$ ），タスク中のカーソルの速度を計測した。

### 3.1 デザイン

ノッチは黒色の矩形とし，MacBook Pro (2021) のノッチの実寸を参考に，高さを 6 mm，幅を 37 mm とした。カーソルがノッチと重なった場合，カーソルの重なる部分がノッチによって隠された。終了ターゲットの高さはノッチと同じ 6 mm，幅  $W$  は 6, 23 mm の 2 条件とした。  $W$  は，MacBook Pro (2021) のタスクバーのターゲットの実寸を参考にした。開始ターゲットは常に終了ターゲットと同じサイズであった。開始ターゲットから終了ターゲットまでの距離  $A$  は 100, 200 mm の 2 条件であった。式 1 の  $ID$  の間隔がおおよそ一定になるように，  $W$  を考慮して設定した。ノッチと終了ターゲットの距離  $I$  は，参考文

献 [2,9] を参考に，0, 12,  $\infty$  mm とした。また，  $\infty$  mm はノッチがない条件である。なおノッチは常にターゲット間に配置された。終了ターゲットを目指す戦略 *Strategy* は，画面端にカーソルを沿わせる戦略 *along* (図 4 (i))，ノッチを回避する戦略 *avoid* (図 4 (ii)) の 2 条件とした。

画面上端に接するターゲットに対して，カーソルの進入角度が操作時間に与える影響は，  $y$  軸を基準に線対称な進入角度で同等であることが知られている [11]。そのため，終了ターゲットが開始ターゲットの左側でも右側でも，パフォーマンスは変わらないと考えられる。本実験では開始ターゲットは常に終了ターゲットの左側に配置した。

画面上端のターゲットを狙う場合，一般的な矢印のカーソルと環状型のカーソルで操作時間に有意な差は見られないことが知られている [11]。本実験では一般的な矢印のカーソルを用いた。またカーソルのサイズは，MacBook Pro (2021) のデフォルトのサイズを参考にした。

### 3.2 手順

参加者は *Strategy* ごとにタスクを行った。初めに片方の *Strategy* の説明を受け，24 試行の練習を行った。その後，1 セットを 12 試行 ( $=2A \times 2W \times 3I$ ) とし，20 セット (240 試行) の本番を行った。20 セット終了後，2 分間の休憩を設けた。休憩終了後，もう片方の *Strategy* の説明を受け，同様に練習と本番を行った。1 セット内の条件はランダムに提示された。 *Strategy* の順序は，参加者間でカウンターバランスを取った。参加者 1 名あたり約 20 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) 試行開始前に提示された条件を確認することを指示した。クラッチはフィッツの法則のモデル適合度を下げることが知られている [14]。モデル適合度への影響を実験条件のみにするため，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  $A$  はクラッチせずに十分カーソルを移動でき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だろう。

## 4. 結果 1

5,760 回 ( $2A \times 2W \times 3I \times 2Strategy \times 20$  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  $A$ ,  $W$ ,  $I$ , *Strategy*，従属変数はエラー率  $ER$ ，操作時間  $MT$ ，PE における virtual travel distance 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。

### 4.1 エラー率 $ER$

エラーの試行は 241 回 (4.18%) であった。主効果は  $W$  ( $F_{1,11} = 61.4$ ,  $p < 0.001$ ,  $\eta_p^2 = 0.848$ , 図 5 (i)) で，交互

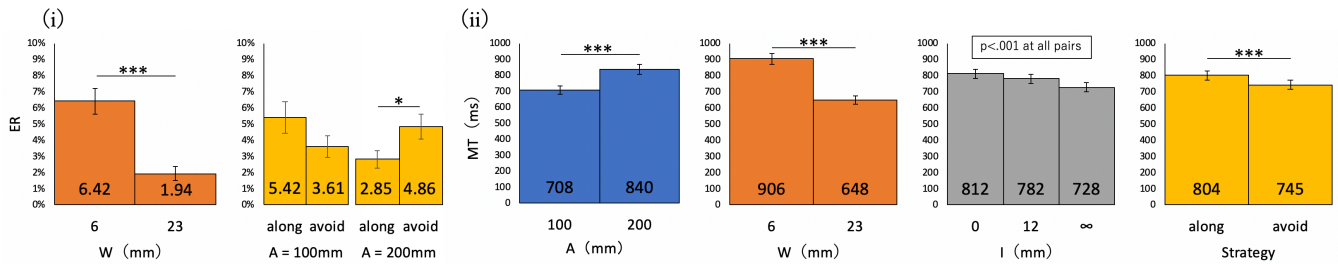


図 5 (i) エラー率  $ER$  への  $W$  の影響,  $A \times Strategy$  の交互作用.  
 (ii) 操作時間  $MT$  への  $A, W, I, Strategy$  の影響.

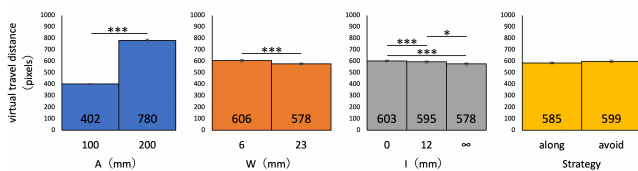


図 6 virtual travel distance に対する  $A, W, I, Strategy$  の影響.

表 1 操作時間  $MT$  の分散分析の結果.

Factors	DF	DFDen	F	p	$\eta_p^2$
A	1	11	330	< 0.001	0.968
W	1	11	404	< 0.001	0.973
I	2	22	87.2	< 0.001	0.888
Strategy	1	11	24.1	< 0.001	0.687
A × W	1	11	0.999	0.339	0.0833
A × I	2	22	0.757	0.571	0.0497
A × Strategy	1	11	3.39	0.0928	0.235
W × I	2	22	2.84	0.0799	0.205
W × Strategy	1	11	69.1	< 0.001	0.863
I × Strategy	2	22	15.5	< 0.001	0.584
A × W × I	2	22	0.427	0.658	0.0373
A × W × Strategy	1	11	8.29	< 0.05	0.430
A × I × Strategy	2	22	5.28	< 0.05	0.324
W × I × Strategy	2	22	9.67	< 0.001	0.468
A × W × I × Strategy	2	22	0.122	0.886	0.0110

作用は  $A \times Strategy$  ( $F_{1,11} = 13.7, p < 0.01, \eta_p^2 = 0.554$ ) で見られた (図 5 (i)).  $A = 100$  では  $Strategy = avoid$  が  $Strategy = along$  より高く,  $A = 200$  では  $Strategy = avoid$  が  $Strategy = along$  より高かった. しかし, どちらも先行研究 [2, 8] のエラー率と近い値であった.

#### 4.2 操作時間 $MT$

エラーの試行を除いた 5,519 回に対して分析を行った. 分散分析の結果を表 1, 多重比較の結果を図 5 (ii) に示す. 主効果において  $Strategy$  間で有意差が見られ,  $along$  では,  $avoid$  より長い  $MT$  が観測された.  $A = 100$  において,  $W = 23$  の時,  $Strategy$  間で有意差が見られなかった. また,  $Strategy = avoid$  において,  $A = 100$  の時,  $I = 0$  と  $I = 12$  で有意差が見られず,  $A = 200$  の時は, 全ての  $I$  間で有意差が見られなかった.  $Strategy = avoid$  においては,  $W = 23$  の時,  $I = 0$  と  $I = 12$  で有意差が見られず,  $W = 6$  の時,  $I = 0$  と  $I = 12, I = 12$  と  $I = \infty$  で有意差が見られなかった. ノッチがない時, もしくは  $W = 23$  において  $I = 12$  の時,  $Strategy$  間で有意差が見られなかつ

た. なお, 交互作用はみられたものの, 各条件が  $MT$  に与える影響は, 主効果と同じ傾向であった (例えば,  $Strategy$  条件では, 常に  $along$  が  $avoid$  より大きかった).

#### 4.3 virtual travel distance

エラーの試行を除いた 5,519 回に対して分析を行った. エラーを除いた 5,519 回の試行に対する多重比較の結果を図 6 に示す. 主効果は  $A$  ( $F_{1,11} = 177 \times 10, p < 0.001, \eta_p^2 = 0.994$ ),  $W$  ( $F_{1,11} = 43.4, p < 0.001, \eta_p^2 = 0.798$ ),  $I$  ( $F_{2,22} = 24.6, p < 0.001, \eta_p^2 = 0.691$ ) で見られた.  $Strategy$  ( $F_{1,11} = 2.49, p = 0.143, \eta_p^2 = 0.185$ ) では見られなかった. 交互作用は  $A \times Strategy$  ( $F_{1,11} = 4.91, p < 0.05, \eta_p^2 = 0.309$ ),  $I \times Strategy$  ( $F_{2,22} = 4.08, p < 0.05, \eta_p^2 = 0.270$ ),  $A \times I \times Strategy$  ( $F_{2,22} = 3.78, p < 0.05, \eta_p^2 = 0.256$ ) で見られた. なお, 交互作用はみられたものの, 各条件が virtual travel distance 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,  $Strategy$  条件では, 常に  $avoid$  が  $along$  より大きかった).

### 5. 議論 1

#### 5.1 操作時間

ノッチを回避する戦略  $Strategy = avoid$  が, 画面端にカーソルを沿わせる戦略  $Strategy = along$  より, virtual travel distance は長い傾向があるものの, 操作時間は短い傾向が見られた. そのため, マウスを操作する距離は長くなるものの, ノッチを回避する戦略をとることが望ましいと考えられる. なお,  $A = 100$  において,  $W = 23$  の時,  $Strategy = avoid$  が  $Strategy = along$  より操作時間が短いものの, 有意差はみられなかったことから, 式 1 の  $ID$  が小さい場合には  $Strategy$  の影響は小さくなると考えられる.

交互作用から,  $Strategy = avoid$  では, 操作時間への  $I$  の影響が小さいことが観察された.  $Strategy = avoid$  では, ノッチを回避後, ターゲットを下側から狙うため, ノッチとターゲットの間隔による影響が小さくなったと考えられる. ノッチ付近にターゲットを配置することは操作時間を増加させてしまうとされていた [2]. そこで, ユーザがノッチを回避して操作すると, ノッチとターゲットの距離が操作時間に与える影響を小さくできる可能性がある.



表 2 モデルの定数の推定値と適合度.

行	Strategy	式	$a$	$b_1$	$c$	$b_2$	$R^2$	$AIC$
1	both		239	46.3			0.878	145
2	along	式 1	199	162			0.899	67.2
3	avoid		279	126			0.972	57.5
4	both		0.500	0.000			0.000	
5	along	式 2	195	162			0.887	71.8
6	avoid		274	126			0.964	58.8
7	both		219	144	1.00	7.47	0.909	131
8	along	式 3	170	162	1.00	11.2	0.955	67.0
9	avoid		229	134	0.982	3.72	0.984	58.6

## 5.2 参加者アンケート

参加者 12 名のうち 10 名が、ノッチを回避する戦略  $Strategy = avoid$  の方が好みであったと回答した。画面上端に沿わせる戦略  $Strategy = along$  では、ノッチ内でカーソルを見失ってしまったり、ノッチを大きく通り過ぎてしまったりしてしまうストレスがあったと回答した。反面、 $Strategy = avoid$  ではそういったストレスがなかったことが、好みと回答した理由として挙げられていた。10 名のうち 1 名、 $Strategy = avoid$  よりも  $Strategy = along$  の方が平均操作時間が短い参加者がいたが、差は 7 ms と小さかった。残り 2 名のうち 1 名は、ノッチと終了ターゲットが隣接している条件では、 $Strategy = avoid$  を好み、隣接していない場合は  $Strategy = along$  を好んだ。もう 1 名は  $Strategy = along$  の方が好みであったと回答した。両名ともに、 $Strategy = along$  では、カーソルの操作を左右方向のみ行えばよかったことを、好みと回答した理由に挙げている。なお、両者ともに  $Strategy = avoid$  の方が、 $Strategy = along$  よりも平均操作時間は短かった。

## 5.3 モデル適合

表 2 にモデル適合の結果を示す。列  $Strategy$  において、 $both$  の行では 24 条件に対するモデル適合を示し、 $along$  と  $avoid$  の行ではそれぞれの  $Strategy$  における 12 条件に対するモデル適合を示す。適合度 ( $R^2$ ) に加えて、赤池情報量基準 ( $AIC$ ) [15] の値も示す。 $R^2$  が高く、 $AIC$  が低いモデルが良いモデルとされる。 $AIC$  の差が 2 以上であれば、考慮に値する差であるとされ、10 以上であれば有意に差があることを示すとされている。

フィッツの法則 (表 2 (行 1-3)) において、Soukoreff の閾値 ( $R^2 > 0.9$  [16]) を超えたのは  $Strategy = avoid$  のみであった。 $Strategy = both$  で低い適合度を示したのは、先行研究 [2, 8] と異なる結果であった。これは、本実験で  $Strategy$  を実験条件としたことが影響したと考えられる。

障害物を回避して操作した場合のモデル (表 2 (行 4-6)) に対する適合を調査した。式 2 における  $B$  は実験で配置したノッチの高さの半分である 3 mm とした。Soukoreff の閾値を超えたのは  $Strategy = avoid$  であった。障害物を回避することを前提とした式であるため、 $Strategy = avoid$  において高い適合度を示したと考えられる。

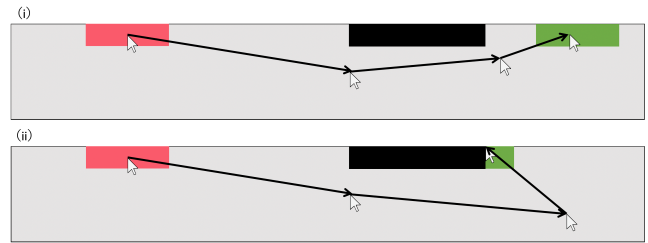


図 7 ノッチとターゲットの距離に応じて指示された戦略の概要図。  
(i) ノッチとターゲットが隣接していない条件。ノッチを回避して操作することが指示された。(ii) ノッチとターゲットが隣接している条件。ノッチの外縁にカーソルが留まることを利用して操作することが指示された。

ノッチとターゲットの距離  $I$  を考慮できるモデル [2] (表 2 (行 7-9)) においては、すべての  $Strategy$  で Soukoreff の閾値を超える適合度を示した。 $Strategy = avoid$  で、フィッツの法則の方が著者らのモデルより低い  $AIC$  を示したが (表 2 (行 3, 9)), その差は 2 未満であった。よって、 $Strategy = avoid$  ではフィッツの法則で十分に操作時間を予測できるが、ターゲットの高さ  $H$  やノッチとターゲットまでの距離  $I$  を考慮できるモデルを用いることで、すべての  $Strategy$  でより高精度に予測できると考えられる。

## 6. 実験 2

実験 1 において、ノッチを回避する戦略が操作時間の観点から望ましいと考えられた。そのため、ノッチをカーソルが進入できない領域に変更することを検討できる。ノッチをカーソルが進入できない領域とした場合、ノッチと隣接するターゲットを、2.3 の画面上端に接するターゲットと同様に捉えられる可能性がある。つまり、ノッチの外縁にカーソルが留まることを利用することで、ノッチと隣接するターゲットを速く選択できるかもしれない。そこで、実験 2 では実験 1 と同様の実験を、ノッチをカーソルが進入できない領域に変更して行った。実験 1 と同様の機材、参加者、タスク、計測値であった。

### 6.1 デザイン

ノッチはカーソルが進入できない領域であった。 $Strategy$  は条件に含まなかった。ノッチの条件  $Notch$  は、カーソルが進入できない条件  $block$ 、カーソルが進入できる条件  $pass$  (実験 1 の  $Strategy = avoid$  のデータを使用) とした。参加者は、(1) ノッチとターゲットが隣接していない条件ではノッチを回避すること (図 7 (i)), (2) ノッチとターゲットが隣接している条件ではノッチの外縁にカーソルが留まることを利用すること (図 7 (ii)) を指示された。その他の条件は実験 1 と同様であった。

### 6.2 手順

参加者は図 7 (ii) の操作に慣れるため、48 試行の練習を

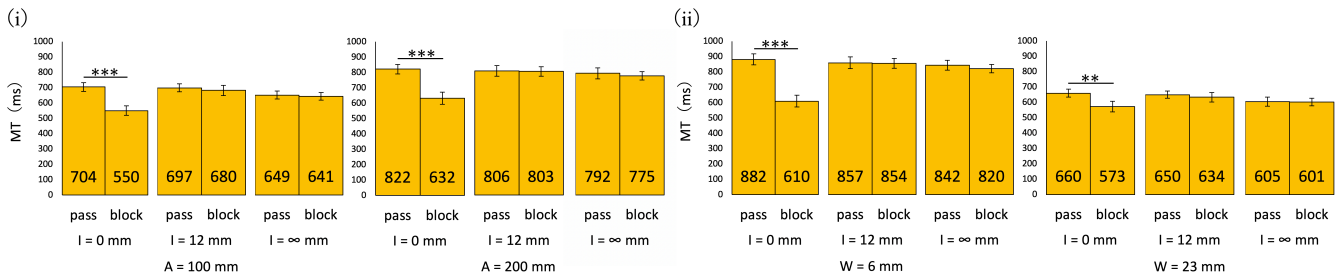


図 8 操作時間  $MT$  に対する (i)  $A \times I \times Notch$  (ii)  $W \times I \times Notch$  の交互作用.

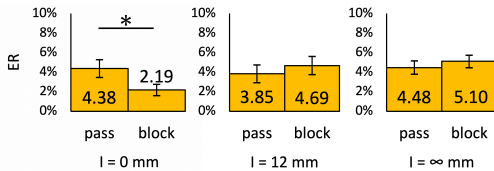


図 9 エラー率  $ER$  に対する  $I \times Notch$  の交互作用.

行った. その後, 1 セットを 12 試行 ( $=2A \times 2W \times 3I$ ) とし, 20 セット (240 試行) の本番を行った. 1 セット内の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 10 分を要した. 参加者に実験 1 と同様の教示を行った.

## 7. 結果 2

2,880 回 ( $2A \times 2W \times 3I \times 20$  セット  $\times 12$  名) の試行が計測され, 実験 1 の  $Strategy = avoid$  を  $Notch = pass$  としてデータに加えた 5,760 回 ( $2A \times 2W \times 3I \times 2Notch \times 20$  セット  $\times 12$  名) の試行を分析の対象とした. 独立変数は  $A, W, I, Notch$ , 従属変数はエラー率  $ER$ , 操作時間  $MT$ ,  $PE$ ,  $PE$  における virtual travel distance であった.

### 7.1 エラー率 $ER$

エラーの試行は 237 回 (4.11%) であった. 主効果は  $W$  ( $F_{1,11} = 72.1, p < 0.001, \eta_p^2 = 0.868$ ) で, 交互作用は  $I \times Notch$  ( $F_{2,22} = 3.91, p < 0.05, \eta_p^2 = 0.262$  (図 9)) で見られた.  $I = 0$  の時,  $Notch = block$  が  $Notch = pass$  より有意に低かった.

### 7.2 操作時間 $MT$

エラーの試行を除いた 5,523 回に対して分析を行った. 分散分析の結果を表 3 に示す.  $A \times I \times Notch$  と  $W \times I \times Notch$  で交互作用が見られ (図 8),  $I = 0$  において  $Notch = block$  が  $Notch = pass$  より有意に短かった.

### 7.3 virtual travel distance

エラーの試行を除いた 5,523 回に対して分析を行った. 主効果は  $A$  ( $F_{1,11} = 578, p < 0.001, \eta_p^2 = 0.981$ ),  $W$  ( $F_{1,11} = 14.7, p < 0.01, \eta_p^2 = 0.573$ ),  $I$  ( $F_{2,22} = 23.4, p < 0.001, \eta_p^2 = 0.681$ ),  $Notch$  ( $F_{1,11} = 22.0, p < 0.001, \eta_p^2 = 0.667$ ) で見られた. 交互作用は  $A \times I$  ( $F_{2,22} = 6.77,$

表 3 操作時間  $MT$  の分散分析の結果.

Factors	DF	DFDen	F	p	$\eta_p^2$
$A$	1	11	443	< 0.001	0.976
$W$	1	11	417	< 0.001	0.974
$I$	2	22	21.8	< 0.001	0.664
$Notch$	1	11	10.8	< 0.01	0.496
$A \times W$	1	11	8.36	< 0.05	0.432
$A \times I$	2	22	9.02	< 0.01	0.450
$A \times Notch$	1	11	1.41	0.260	0.113
$W \times I$	2	22	41.9	< 0.001	0.792
$W \times Notch$	1	11	27.9	< 0.001	0.717
$I \times Notch$	2	22	57.0	< 0.001	0.838
$A \times W \times I$	2	22	0.919	0.414	0.0771
$A \times W \times Notch$	1	11	1.68	0.222	0.132
$A \times I \times Notch$	2	22	3.88	< 0.05	0.261
$W \times I \times Notch$	2	22	37.1	< 0.001	0.771
$A \times W \times I \times Notch$	2	22	0.553	0.583	0.0479

$p < 0.01, \eta_p^2 = 0.381$ ),  $A \times Notch$  ( $F_{1,11} = 8.59, p < 0.05, \eta_p^2 = 0.438$ ),  $W \times I$  ( $F_{2,22} = 4.45, p < 0.05, \eta_p^2 = 0.288$ ),  $I \times Notch$  ( $F_{2,22} = 19.0, p < 0.001, \eta_p^2 = 0.633$ ),  $A \times I \times Notch$  ( $F_{2,22} = 9.43, p < 0.01, \eta_p^2 = 0.462$  (図 10 (上))),  $W \times I \times Notch$  ( $F_{2,22} = 4.53, p < 0.05, \eta_p^2 = 0.292$  (図 10 (下))) で見られた.  $A \times I \times Notch, W \times I \times Notch$  で見られた交互作用において,  $I = \infty$  以外で  $Notch = block$  が  $Notch = pass$  より有意に長かった.

### 7.4 Path Efficiency $PE$

エラーの試行を除いた 5,523 回に対して分析を行った. 主効果は  $A$  ( $F_{1,11} = 21.3, p < 0.001, \eta_p^2 = 0.659$ ),  $W$  ( $F_{1,11} = 5.36, p < 0.05, \eta_p^2 = 0.328$ ),  $I$  ( $F_{2,22} = 23.0, p < 0.001, \eta_p^2 = 0.676$ ),  $Notch$  ( $F_{1,11} = 20.5, p < 0.001, \eta_p^2 = 0.650$ ) で見られた. 交互作用は  $A \times I$  ( $F_{2,22} = 14.1, p < 0.001, \eta_p^2 = 0.562$ ),  $A \times Notch$  ( $F_{1,11} = 13.1, p < 0.01, \eta_p^2 = 0.543$ ),  $W \times I$  ( $F_{2,22} = 3.50, p < 0.05, \eta_p^2 = 0.241$ ),  $I \times Notch$  ( $F_{2,22} = 32.7, p < 0.001, \eta_p^2 = 0.748$ ),  $A \times I \times Notch$  ( $F_{2,22} = 10.8, p < 0.001, \eta_p^2 = 0.496$  (図 11 (上))),  $W \times I \times Notch$  ( $F_{2,22} = 5.54, p < 0.05, \eta_p^2 = 0.335$  (図 11 (下))) で見られた.  $A \times I \times Notch, W \times I \times Notch$  で見られた交互作用において, 主に  $I = 0$  の時,  $Notch = block$  が  $Notch = pass$  より有意に低かった.

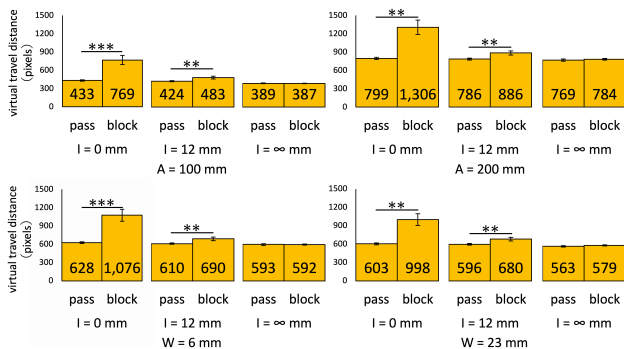


図 10 virtual travel distance に対する (上)  $A \times I \times Notch$  (下)  $W \times I \times Notch$  の交互作用。

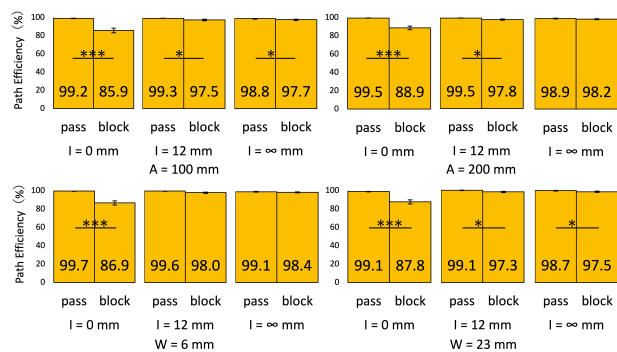


図 11 Path Efficiency  $PE$  に対する (上)  $A \times I \times Notch$  (下)  $W \times I \times Notch$  の交互作用。

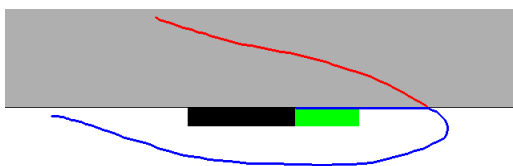


図 12 カーソルの軌跡の一例。赤い線は画面端でカーソルが留まることを無視した、仮想のカーソルの軌跡を示す。青い線は画面上でのカーソルの軌跡を示す。

## 8. 議論 2

### 8.1 操作時間 $MT$

ノッチにカーソルが進入できない条件  $Notch = block$  が、ノッチにカーソルが進入できる条件  $Notch = pass$  より、 $I = 0$  において有意に操作時間が短かった。 $I = 0$  において、 $Notch = block$  の virtual travel distance が長くなっていること、 $PE$  が低くなっていることから、参加者はカーソルがノッチの外縁で留まることを利用して操作したと考えられる (図 12)。そのため、微細な制御をせずに、ターゲットを短い操作時間で選択できたと考えられる。また、操作時間は  $I = 0$  においてのみ有意差があった。ノッチを「カーソルが進入できない領域」にすると、ノッチと隣接しないターゲットの操作時間は変わらず、ノッチと隣接するターゲットの操作時間を主に短縮できると考えられる。

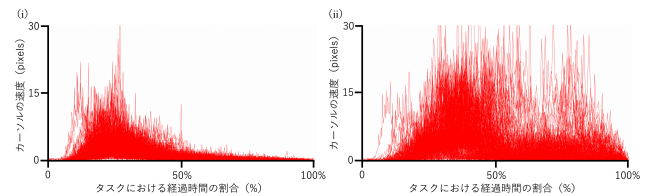


図 13 タスクにおける経過時間の割合に対する (i)  $Notch = pass$  (ii)  $Notch = block$  のカーソルの速度。

### 8.2 エラー率 $ER$

$Notch = block$  が、 $Notch = pass$  より、 $I = 0$  において有意にエラー率が低かった。2.3 の画面端に接するターゲットと同様に、カーソルがノッチの外縁で留まり、微細な制御をせずに選択できたことが影響していると考えられる。

### 8.3 参加者アンケート

参加者全員がノッチにカーソルが進入できないことに対して好印象であった。特にノッチとターゲットが隣接している条件で好まれ、ノッチにカーソルが留まるため操作が容易であったと回答した。また、virtual travel distance は増加しているものの、操作における肉体的負担が増加したと回答した参加者はいなかった。なお参加者のうち 3 名が、不必要にノッチにカーソルがひっかかってしまったことがあり、邪魔と感じたこともあったと回答した。

### 8.4 カーソルの速度

タスクにおける経過時間の割合に対する、カーソルの速度を図 13 に示す。図 13 は  $A = 200$ ,  $W = 6$ ,  $I = 0$  の条件を示す。ポインティングタスクでは、操作時間の前半に速度のピークがあり、その後カーソルをターゲット内に入れるために減速することが知られている [17]。図 13 (i) は  $Notch = pass$  のカーソルの速度を示し、先行研究 [17] と同様の傾向が見られる。対して、 $Notch = block$  のカーソルの速度を示す図 13 (ii) では、後半でもあまり減速していないことが見て取れる。この傾向は、画面端に接するターゲットのカーソル速度を調査した山中らの傾向と同様であった [12]。ノッチの外縁にカーソルが留まることを利用し、カーソルを高速に動かしながらポインティングができたためと考えられる。

### 8.5 モデル適合

$Notch = block$  に対して、式 3 への適合を調査したところ、Soukoreff の閾値を下回る適合度を示した ( $R^2 = 0.861$ )。式 3 はターゲットとノッチの距離  $I$  が小さくなるほど操作時間が増加することを示すが、図 8 より  $I = 0$  で最も操作時間が短いため、適合度が低くなったと考えられる。 $I = 0$  を除外して式 3 への適合を調査したところ、 $R^2 = 0.999$  と高い適合度を示した。そのため、 $I = 0$  に対して、異なる操作時間予測モデルを構築する必要があるだろう。

## 9. 制約・展望

本実験のノッチは、MacBook Pro (2021) のノッチの実寸を参考に設定した。しかし、今後発売される製品ではノッチの大きさが変化する可能性がある。その場合、カーソルを画面上端に沿わせる戦略のほうがノッチを回避する戦略よりも望ましい状況があるかもしれない。例えば、高さが小さいノッチでは、カーソルの全体がノッチに遮蔽されることがなくなり、ノッチ内でカーソルを見失わなくなる。また、iPhone 14 Pro では Dynamic Island (角丸長方形の描画が行われない領域) が配置された。同様に MacBook のノッチの形状も変更されるかもしれない。今後は、ノッチの大きさや形状を条件に加えて実験を行いたい。

本実験では、カーソルを操作するデバイスとしてマウスを用いた。しかし、MacBook Pro (2021) にはトラックパッドが搭載されているため、そちらを用いるユーザも多くいるだろう。トラックパッドでは、本実験で制限したクラッチがマウスよりも多く行われる。また、指を用いて操作するため、マウスよりも高速な操作を行うことが難しいかもしれない。その影響で、トラックパッドを用いた場合では、本実験とは異なる結果が観察される可能性がある。今後は、操作デバイスを条件に加えて実験を行いたい。

障害物を回避して操作した場合の操作時間を予測するモデルは、Bubble Clusters [18] や Attribute Gates [19] といった、障害物を回避しつつ連続的にカーソルを移動させる必要がある手法に適用できる可能性があると示唆されている [5]。本稿で提案した操作時間予測モデル (式 3) は、本実験条件に対して高い適合を示したが、こういった手法の操作時間の予測に応用できる可能性がある。今後、式 3 がノッチが配置された状況における操作時間の予測以外への応用が可能か、調査を行いたいと考えている。

本研究では操作時間とエラー率のみを考慮して評価を行った。しかし、ノッチを配置することは、ユーザの心理的なストレスの増加につながるかもしれない。例えば、ノッチをカーソルが進入できない領域にした場合、不必要にノッチにカーソルがひっかかることを不快に感じるかもしれない。そのため、ノッチが操作時間とエラー率に与える影響のみを調査したことが、本研究の制約といえる。

## 参考文献

- [1] 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, pp. 381–391 (1954).
- [2] 大場洋介, 宮下芳明: ノッチがポインティングの操作時間に与える影響, 研究報告ヒューマンコンピュータインタラクション, Vol. 2022-HCI-199, No. 2, pp. 1–8 (2022).
- [3] Jax, S., Rosenbaum, D. and Vaughan, J.: Extending Fitts' Law to manual obstacle avoidance, *Experimental brain research. Experimentelle Hirnforschung.*

- Expérimentation cérébrale*, DOI: 10.1007/s00221-007-0996-y (2007).
- [4] Vaughan, J., Barany, D. A., Sali, A. W., Jax, S. A. and Rosenbaum, D. A.: Extending Fitts' Law to three-dimensional obstacle-avoidance movements: support for the posture-based motion planning model, *Experimental brain research*, Vol. 207, No. 1, pp. 133–138 (2010).
- [5] Yamanaka, S. and Stuerzlinger, W.: Necessary and Unnecessary Distractor Avoidance Movements Affect User Behaviors in Crossing Operations, *ACM Trans. Comput.-Hum. Interact.*, DOI: 10.1145/3418413 (2020).
- [6] Accot, J. and Zhai, S.: Refining Fitts' Law Models for Bivariate Pointing, CHI '03, New York, NY, USA, Association for Computing Machinery, pp. 193–200 (2003).
- [7] Zhang, X., Zha, H. and Feng, W.: Extending Fitts' Law to Account for the Effects of Movement Direction on 2d Pointing, CHI '12, New York, NY, USA, Association for Computing Machinery, pp. 3185–3194 (2012).
- [8] Yamanaka, S.: Mouse Cursor Movements towards Targets on the Same Screen Edge, GI '18, Waterloo, CAN, Canadian Human-Computer Communications Society, pp. 115–122 (2018).
- [9] Usuba, H., Yamanaka, S. and Miyashita, H.: A Model for Pointing at Targets with Different Clickable and Visual Widths and with Distractors, OzCHI '20, New York, NY, USA, pp. 1–10 (2020).
- [10] Dizmen, C., Hoffmann, E. R. and Chan, A. H.: Movement time to edge and non-edge targets, *Ergonomics*, Vol. 57, No. 1, pp. 130–135 (2014).
- [11] Appert, C., Chapuis, O. and Beaudouin-Lafon, M.: Evaluation of Pointing Performance on Screen Edges, AVI '08, New York, NY, USA, pp. 119–126 (2008).
- [12] 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).
- [13] 大場洋介, 薄羽大樹, 山中祥太, 宮下芳明: 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, 研究報告ヒューマンコンピュータインタラクション, Vol. 2022-HCI-197, No. 55, pp. 1–8 (2022).
- [14] Casiez, G., Vogel, D., Pan, Q. and Chaillou, C.: RubberEdge: Reducing Clutching by Combining Position and Rate Control with Elastic Feedback, UIST '07, New York, NY, USA, pp. 129–138 (2007).
- [15] Akaike, H.: A new look at the statistical model identification, *IEEE Transactions on Automatic Control*, Vol. 19, No. 6, pp. 716–723 (1974).
- [16] 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).
- [17] Asano, T., Sharlin, E., Kitamura, Y., Takashima, K. and Kishino, F.: Predictive Interaction Using the Delphian Desktop, UIST '05, New York, NY, USA, Association for Computing Machinery, p. 133–141 (2005).
- [18] Watanabe, N., Washida, M. and Igarashi, T.: Bubble Clusters: An Interface for Manipulating Spatial Aggregation of Graphical Objects, UIST '07, New York, NY, USA, Association for Computing Machinery, DOI: 10.1145/1294211.1294241 (2007).
- [19] Sulaiman, A. N. and Olivier, P.: Attribute Gates, UIST '08, New York, NY, USA, Association for Computing Machinery, DOI: 10.1145/1449715.1449726 (2008).