



Title	視線インタフェースにおける誤選択を低減する手法に関する研究
Author(s)	崔, 明根
Citation	北海道大学. 博士(情報科学) 甲第15997号
Issue Date	2024-03-25
DOI	10.14943/doctoral.k15997
Doc URL	<a href="http://hdl.handle.net/2115/91756">http://hdl.handle.net/2115/91756</a>
Type	theses (doctoral)
File Information	Myunguen_Choi.pdf



[Instructions for use](#)

# 博士論文

## 視線インタフェースにおける 誤選択を低減する手法に関する研究

北海道大学 大学院情報科学院  
情報科学科 情報理工学コース  
ヒューマンコンピュータインタラクション研究室

崔 明根

2024年2月

## Doctoral Dissertation

Study on Eye-Gaze Interface for Reducing False Selection

by

Myunguen CHOI

Human-Computer Interaction Laboratory,  
Research Group of Synergetic Information Engineering,  
Division of Computer Science and Information Technology,  
Graduate School of Information Science and Technology,  
Hokkaido University  
February 2024

# 目次

<b>第1章 序論</b>	<b>1</b>
1.1 本研究の背景	1
1.1.1 「コンピュータ」の普及によるインタラクション対象の拡大	2
1.1.2 本研究の主題: 視線インタフェース	4
1.2 本論文の目的	5
1.3 本論文の構成	6
<b>第2章 関連研究</b>	<b>9</b>
2.1 目の生理的特徴	9
2.1.1 Fixation (固視)	9
2.1.2 Saccade (サッケード)	9
2.1.3 Smooth Pursuit (滑動性眼球運動)	10
2.1.4 Vergence (輻輳開散運動)	10
2.2 視線計測手法と精度	10
2.2.1 角膜反射法	10
2.2.2 顔画像を用いた機械学習視線検出法	11
2.2.3 その他の視線検出法	12
2.3 モノモーダル視線インタフェースの入力手法	13
2.3.1 滞留時間選択	13
2.3.2 視線ジェスチャ	14
2.3.3 Smooth Pursuit	15
2.3.4 視線深度情報を用いた入力手法	15
2.4 マルチモーダル視線インタフェースの入力手法	17
2.4.1 視線と手を組み合わせた入力手法	17
2.4.2 視線と頭部動作を組み合わせた入力手法	18
2.4.3 視線とその他のモダリティを組み合わせた入力手法	19
2.5 視線入力手法の性能比較	19
2.5.1 視線入力手法ごとの比較	19

2.5.2	視線以外のモダリティとの比較 . . . . .	21
2.6	本章のまとめ . . . . .	21
<b>第3章</b>	<b>Bubble Gaze Cursor + Lens: エリアカーソル手法の視線操作への拡張</b>	<b>23</b>
3.1	関連研究 . . . . .	24
3.1.1	視線インタフェースのパフォーマンスに影響する要因 . . . . .	24
3.1.2	必要なターゲットサイズ . . . . .	26
3.1.3	ジッタ・視線精度の改善手法 . . . . .	27
3.1.4	ターゲットサイズを拡大する手法 . . . . .	27
3.1.5	視線カーソル . . . . .	29
3.1.6	エリアカーソル . . . . .	29
3.2	提案手法: Bubble Gaze Cursor + Lens . . . . .	32
3.3	実験1: Bubble Gaze Cursor の性能評価実験 . . . . .	32
3.3.1	実験参加者と使用機器 . . . . .	32
3.3.2	比較手法 . . . . .	33
3.3.3	実験デザイン . . . . .	34
3.3.4	実験手順 . . . . .	35
3.3.5	結果 . . . . .	36
3.3.6	実験結果のまとめ . . . . .	40
3.4	実験1に関する議論 . . . . .	41
3.4.1	視線インタフェースにおける Bubble Cursor の効果 . . . . .	41
3.4.2	視線インタフェースとマウスインタフェースの比較 . . . . .	42
3.5	実験2: Bubble Gaze Lens の性能評価実験 . . . . .	42
3.5.1	Bubble Gaze Lens の実装 . . . . .	43
3.5.2	実験参加者と使用機器 . . . . .	46
3.5.3	実験デザイン . . . . .	46
3.5.4	実験手順 . . . . .	47
3.5.5	評価指標 . . . . .	48
3.5.6	結果 . . . . .	49
3.5.7	実験結果のまとめ . . . . .	53
3.6	実験2に関する議論 . . . . .	54
3.6.1	拡大レンズによるパフォーマンスの改善 . . . . .	54
3.6.2	拡大レンズ起動アルゴリズムの性能 . . . . .	56
3.6.3	拡大レンズの欠点 . . . . .	57

3.6.4	提案手法の制限 . . . . .	58
3.7	視線インタフェースにエリアカーソル法を導入することへの議論 . . . . .	58
3.7.1	一般的な視線入力手法との比較 . . . . .	58
3.7.2	環境ごとの提案手法の有用性 . . . . .	59
3.7.3	他のエリアカーソル手法・拡大アルゴリズムの検討 . . . . .	60
3.8	本章のまとめ . . . . .	61
<b>第4章</b>	<b>Kuiper Belt: VRにおける自然ではない視線角度を用いた視線入力手法の提案</b>	<b>62</b>
4.1	関連研究 . . . . .	63
4.1.1	ターゲットとラベルの分離による Midas Touch 問題の解決手法 . . . . .	63
4.1.2	人の視線の分布と特徴 . . . . .	65
4.1.3	VR・MRにおける視線インタラクション領域 . . . . .	66
4.2	提案手法: Kuiper Belt . . . . .	67
4.3	実験1: Kuiper Belt のデザインスペースの検討 . . . . .	67
4.3.1	実験参加者と使用機器 . . . . .	67
4.3.2	実験デザイン . . . . .	68
4.3.3	実験手順 . . . . .	69
4.3.4	データ収集と前処理 . . . . .	70
4.3.5	結果 . . . . .	70
4.3.6	実験結果のまとめ . . . . .	73
4.4	実験2: Kuiper Belt の性能評価実験 . . . . .	74
4.4.1	実験参加者 . . . . .	75
4.4.2	比較手法 . . . . .	75
4.4.3	実験デザイン . . . . .	77
4.4.4	実験手順 . . . . .	77
4.4.5	結果 . . . . .	79
4.4.6	実験結果のまとめ . . . . .	83
4.5	応用アプリケーション例 . . . . .	84
4.5.1	メニュー選択 . . . . .	84
4.5.2	電子機器操作インタフェース . . . . .	84
4.5.3	密で小さなオブジェクトの選択手法 . . . . .	85
4.5.4	VR ゲームでのアイテム選択 . . . . .	85
4.6	議論 . . . . .	85

4.6.1	Kuiper Belt の長所・短所・適応性 . . . . .	86
4.6.2	Kuiper Belt 使用時のユーザ負荷 . . . . .	87
4.6.3	視線インタフェースにおけるメニューアイテムの配置すべき場所 . . . . .	87
4.6.4	Kuiper Belt に残された Midas Touch 問題 . . . . .	88
4.6.5	提案手法の制限 . . . . .	88
4.7	本章のまとめ . . . . .	88
<b>第 5 章</b>	<b>Asteroid Gazer: VR における視線のみを用いた遮蔽されたオブジェクト の選択手法の提案</b>	<b>90</b>
5.1	関連研究 . . . . .	91
5.1.1	遮蔽されたオブジェクトの選択手法 . . . . .	91
5.1.2	仮想再配置アプローチ . . . . .	94
5.2	提案手法: Asteroid Gazer . . . . .	96
5.2.1	ターゲット候補選択 . . . . .	98
5.2.2	メニューアイテムに対するターゲット候補の仮想再配置 . . . . .	98
5.2.3	メニューアイテム選択によるターゲット選択 . . . . .	99
5.3	実験: Asteroid Gazer の性能評価実験 . . . . .	100
5.3.1	実験参加者と使用機器 . . . . .	101
5.3.2	比較手法 . . . . .	101
5.3.3	実験デザイン . . . . .	103
5.3.4	実験手順 . . . . .	103
5.3.5	評価指標 . . . . .	104
5.3.6	結果 . . . . .	105
5.4	応用アプリケーション例 . . . . .	109
5.4.1	ジグソーパズルアプリケーション . . . . .	110
5.4.2	スマートホームアプリケーション . . . . .	110
5.4.3	ショッピングアプリケーション . . . . .	110
5.5	議論 . . . . .	111
5.5.1	部分的に遮蔽されたオブジェクト選択タスクにおける Asteroid Gazer の性能 . . . . .	111
5.5.2	階層メニューの利点 . . . . .	112
5.5.3	Asteroid Gazer のパラメータ . . . . .	113
5.5.4	制限 . . . . .	113
5.6	本章のまとめ . . . . .	114

<b>第 6 章 総合的な議論</b>	<b>115</b>
6.1 本研究の貢献	116
6.1.1 可視領域が小さなオブジェクトの選択	116
6.1.2 視覚探索中の意図しないオブジェクト注視による誤選択 (Midas Touch) の解決	116
6.1.3 入力多様性の向上	117
6.2 今後の視線インタフェースの展望	117
6.2.1 本研究の展望	117
6.2.2 本研究の限界とそれに応じた展望	119
6.2.3 モノモーダル視線インタフェースの展望	119
6.2.4 視線モダリティの展望	121
6.2.5 長期的な未来における入力インタフェースと視線インタフェース	121
<b>第 7 章 結論</b>	<b>123</b>
<b>付録 A 研究業績リスト</b>	<b>155</b>

# 第1章 序論

## 1.1 本研究の背景

近年の計算機科学の発展は目覚ましい。ディープラーニングを筆頭とした機械学習モデルの性能向上によってコンピュータは人間以上の物体認識機能を獲得し、かつ膨大なデータから規則性を発見することが可能になった。さらに、巨大なデータセットを学習した大規模言語モデル（LLM; Large Language Models）は人間並みの言語能力を獲得した。上記のような計算機科学の発展は今後も続き、コンピュータはより優れた存在へと進化していくだろう。しかし、コンピュータを扱う存在たる人間の性能は今後成長することはない。コンピュータがどれだけ高性能になろうと、人間がコンピュータと適切な対話を行うことが出来ない限り、人間がコンピュータの恩恵を受けることは出来ない。したがって人間とコンピュータとのインタラクションに適したデザインを検討することは重要であり、それを調査する研究領域が Human-Computer Interaction（HCI）である。

「コンピュータとの対話」とは一般的に、ユーザがコンピュータに対して入力を与え、入力に応じた出力をコンピュータが出力し、それをユーザが受け取り再度入力を行う、といった繰り返しを指している。つまり対話を行うためにはコンピュータに対して入力を行う必要があり、その入力を実行するためのシステムが入力インタフェースと呼ばれている。入力インタフェースは入力デバイスや入力操作方法の総称であり、マウスやキーボードが一般的に馴染み深い。マウスにおいては画面上のポインタを対象上に配置しクリック動作を行うことで対象を選択し、キーボードにおいてはボタンを押すことでボタンに対応した入力を実行する。対象の選択（マウス）やコマンド入力（キーボード）は入力インタフェースにおける基本的な入力動作であり、他の入力インタフェースにおいても同様の操作を行うことが可能である。マウスやキーボードが社会で大いに受け入れられている理由は、これらが非常に優れた入力インタフェースだからである。HCI 領域における優れた入力インタフェースの要件は様々だが、大きな要素としては「入力速度が速い」「入力精度が高い」「ユーザビリティが高い」などがある。これらの要素が優れているほど入力情報量が向上し、人間がコンピュータに与えられる情報量が増加する。ゆえに、HCI 領域では入力インタフェースの入力性能向上が研究課題の一つとして捉えられている。入力インタフェースの性能向上による入力情報量の向上量は計算機科学の研究分野から見ると非常に





図 1.1: 左図は Weiser らの Ubiquitous Computing のイメージ [209]. 人々の手元にコンピュータが把持されており、コンピュータの偏在化を予想している. 右図は Mixed Reality のイメージ<sup>1</sup>. 仮想オブジェクトがインタラクション対象となっている.

わずかである. しかし一つの入力に必要な時間の減少は僅かであっても, 何度も入力を行うほど総削減時間は線型的に増加し, コンピュータとインタラクションできる回数・量も増加する. したがって「人間の入力性能向上」につながる入力インタフェースの性能向上は, コンピュータの性能向上と共に試みるべき課題だと言える.

### 1.1.1 「コンピュータ」の普及によるインタラクション対象の拡大

入力インタフェースはインタラクション対象に対して入力を行うシステムである. このインタラクション対象は過去においてはデスクトップ PC に限定されていた. なぜなら過去においてインタラクション可能な対象はコンピュータに限定され, かつ特定の位置に固定されたコンピュータのみが存在していたためである. しかし Weiser が提唱した Ubiquitous Computing [209] の概念が実現され, コンピュータはあらゆる場所に偏在的に存在するようになった (図 1.1 左). コンピュータの小型化により配置の制約が小さくなり, 現在ではモバイル PC やタブレット, スマートフォン, パブリックディスプレイ, IoT など, コンピュータは現在身の回りに溢れるほど存在する. また Virtual Reality (VR) や Mixed Reality (MR) に関する技術が発展し, 現実世界のオブジェクト以外にも仮想環境内のオブジェクトがインタラクション可能な対象となった (図 1.1 右). したがって現在の過去に比べてインタラクション対象が著しく増加しており, 今までの入力インタフェースでは不十分な状況が多く存在すると考えられる.

増加したインタラクション対象の性質から, 現在のインタラクション対象の存在する領域は二次元環境 (ディスプレイ) から三次元環境 (実空間, 仮想空間) に拡大されつつあ

<sup>1</sup><https://learn.microsoft.com/ja-jp/windows/mixed-reality/discover/mixed-reality>

ると言える。三次元環境におけるインタラクションでは、二次元環境と三次元環境に対するインタラクションがそれぞれ実行される。前者はディスプレイや仮想情報として表示された二次元 UI、後者は仮想オブジェクトや実オブジェクトなどが該当する。ゆえに、これらが混合する環境内で様々なオブジェクトに対してシームレスにインタラクションを実行できる手法が求められている。またインタラクション対象の拡張に伴い、ユーザが入力を実行するタイミングはより多様になると考えられる。デスクトップ PC の前に座ることがインタラクション開始の合図だった以前と異なり、起立中や徒歩中のモバイルデバイスとのインタラクション、遠距離の IoT デバイスに対するインタラクション、料理や掃除など日常生活のインタラクションの実行など、インタラクション対象のユーザに対する位置関係やユーザの状態は多岐にわたると考えられる。これは特定の状況やインタフェースに特化した入力インタフェースの限界を示しており、様々な入力インタフェースの併用や統合が必要であることを示している。ゆえに複数の入力インタフェースの併用を想定し、特定のインタラクション対象や状況において高性能な入力が可能な入力インタフェースを開発することが必要だと言える。

平面ディスプレイなどの二次元環境における最適なインタフェースは、基本的に手を用いたインタフェースである。PC においてはマウスやキーボード、モバイルデバイスにおいてはタッチ操作などが相当する。これは手が人間の器官の中で最も精密な操作が可能であるためであり、入力速度や精度、入力の多様性に関しても手は非常に優れたモダリティである。三次元環境においても手を用いた入力が一般に優勢であり、コントローラや空中ハンドジェスチャなどが VR や MR 環境におけるインタラクションに使用されている。しかし、手を用いたインタフェースは三次元環境においては二次元環境ほど支配的な入力インタフェースではない。その理由としてはまず、一部の例外を除き、遠距離のインタラクションに適していない。手を用いた入力インタフェースはインタラクション対象に直接触れるものが多く、対象が手が届く範囲内に存在する必要がある。そのため遠距離のオブジェクトに対してインタラクションを行うことは困難である。この課題に関しては、レーザーポインタのようにレイ（光線）を照射するコントローラを使用することで手で遠距離とインタラクションを行うことは可能である [130]。しかしこれらのデバイスを持ち運ぶことは日常生活において負担が大きく、また日常生活では手が塞がる状況が存在するためコントローラの使用が適切ではない場合がある。次に、頻繁な入力は身体的な負荷が大きい。三次元環境における手を用いたインタラクションは、基本的に手を浮かせて入力を実行する。しかしこの手を浮かせる行為は身体的負荷が大きく [59, 15]、頻繁な入力が実行されるインタフェースには適していない。手を用いた入力インタフェースはこれらの欠点を有しながらも三次元環境におけるファーストオプションではあるが、あらゆる状況で万能な入力インタフェースではないと言える。ゆえに、手を用いた入力インタフェースの欠点を

補うような入力インタフェースの開発が必須だと考えられる。

### 1.1.2 本研究の主題: 視線インタフェース

手は優秀な入力モダリティであるが、万能の入力モダリティではない。ゆえに手が適していない状況を補助する、または特定の状況下においては手よりも優れた入力を達成する入力インタフェースが探求されてきた。そのため、音声 [162] や脳波 [216], 視線 [189, 25], 足 [138, 217], 頭部 [35] など、手以外の様々な箇所を用いた入力インタフェースが検討されてきた。特に音声インタフェースはスマートスピーカなど様々なデバイスで主要な入力方法となり、手よりも高速な入力速度を達成する事例も確認されている [163, 42]。また脳波を用いたインタフェースは、脳から得られる情報量が多いことから、非常に期待が高い入力インタフェースだと言える。しかし、これらの入力インタフェースは高度な入力が可能である一方で、対象を規定しない。そのため対象足り得るオブジェクトが環境に複数存在する場合、どのオブジェクトに対して入力を実行すべきかが分からない。つまり環境に点在する対象とのインタラクションを行うためには、追加の操作による対象規定が必要となり、入力時間の増加が生じる。ゆえに筆者は対象規定を即座に実行可能な入力インタフェースであり、かつそれらの中で最も高速な動作を有する視線インタフェースに着目した。

視線インタフェースとはユーザの視線を検出し、入力とするインタフェースである。視線を目から照射されるレイとして解釈し、見つめたオブジェクトの選択などを実行する。インタラクション対象がディスプレイの場合は視線とディスプレイの交点を注視点と定義し、注視点にポインタを配置しマウスのように選択を行う。視線インタフェースを用いたアプリケーションは多数存在し、二次元環境においては GUI 操作のためのインタフェース [125] やキーボード（アイタイピング） [118], 三次元環境においてはオブジェクト選択 [189, 25] などが実装されている。人間の目は人間の器官の中で最も高速に動作し、視線の最大速度は 1000m/s にまで達する。ゆえに視線インタフェースは高速なポインティングが可能であり、かつ視線をレイとして解釈することで遠距離に対するインタラクションが可能である。事実、ターゲットをポインティング（捕捉）する速度は他のインタフェースよりも速い [178]。また目を動かすことによる身体的な負荷が非常に小さいことから、日常的な使用に適した入力インタフェースだと言える。ゆえに視線インタフェースは手を用いた入力インタフェースとは異なる優位点を有しており、手に代わる有望な入力インタフェースの一種だと言える。現在視線インタフェースが最も使用されている用途は、筋萎縮性側索硬化症（ALS）患者など手足を動かすことが出来ない人々のマウスに代わる入力インタフェースとしての用途である。一般的には視線インタフェースを用いたキーボー

ド入力による発話が主な用途であったが、Microsoft Windows 10 では GUI を視線で操作する機能がベータ版で搭載され始め、視線だけで様々な PC 操作が可能となった。また健常者が視線インタフェースを使用する環境も整備され続けており、発売されている多くの Head Mounted Displays (HMDs) にはアイトラッカ (視線計測器) が搭載されるようになった。さらに RGB カメラを用いた顔画像から視線方向を推定する技術も発展を続けており、人間の視線データを取得できる環境は一般化しつつあると言える。

## 1.2 本論文の目的

視線インタフェースの一般的な入力方法は、ターゲットを一定時間注視することである。視線はマウスなどとは異なり、ボタンやトリガなど明示的な入力機構が存在しない。ゆえに何らかの選択動作を定義する必要がある、それが視線をターゲット上に一定以上滞留させることである。この選択方法を滞留時間入力手法と呼び、選択に必要な注視時間を滞留時間と呼ぶ。世に存在する視線インタフェースのほとんどは滞留時間入力手法を採用しているが、この手法にはいくつか課題が存在する。それぞれの課題を以下に示す。

### 視線インタフェースの課題 1) 小さなオブジェクトの選択が難しい

滞留時間入力手法は閾値を超えるまで視線をオブジェクト内に固定する必要がある。しかし視線入力は注視中の不随意的な視線の揺れ (固視微動) が生じることやアイトラッキングの精度が低いことから、小さなターゲットの内部に視線を留め続けることが困難となる。また一度視線がターゲット外に出ると注視時間のカウンタがリセットされるため、選択時間の大幅な増大につながる。つまり小さなターゲットの選択が困難なことは選択精度に影響する上、選択時間の増大にもつながる。基本的にディスプレイ環境においてはサイズが小さなオブジェクトのみがこの課題の対象となるが、三次元環境においてはオブジェクトが遮蔽されることによって可視領域が減じたオブジェクトも対象となる。つまり環境に応じてターゲットが小さい原因は異なり、それぞれの環境における小さなオブジェクトの選択手法の開発が必要だと言える。

### 視線インタフェースの課題 2) 意図しない誤選択 (Midas Touch) が生じる

滞留時間入力を使用するためにはオブジェクトへの注視時間を計測する必要がある、それはすなわちオブジェクトに目を向けるたびに選択動作が生じる可能性があるということである。そして選択の意思がなくとも、オブジェクトに対する注視時間が滞留時間の閾値を超え、オブジェクトが誤選択されてしまうことがある。この誤選択は Midas Touch と呼

ばれ [78], 確認されてから長い間解決されていない. Midas Touch が存在するという事は, 環境内の選択可能なオブジェクト全てに誤選択の可能性があるということである. ゆえに誤選択による影響が大きい操作に対して滞留時間入力を用いることは適しておらず, Midas Touch の存在が視線インタフェースの使用用途を大きく狭めていると言える.

視線インタフェースが抱える課題はどちらも視線入力使用時の誤選択に関わるものであり, これらのエラーが視線インタフェースの汎用性を大きく損ねている. ゆえに本論文では上記の課題の改善を目指した手法を提案することで視線インタフェースの選択対象のスコープを広げ, 視線インタフェースをより様々な場面で使用できるようにする.

本研究では二次元環境 (研究 1) と三次元環境 (研究 3) における小さなターゲット選択手法を提案し, 三次元環境における Midas Touch を解消する手法 (研究 2) を提案する. 一つ目の研究では, オブジェクトサイズを空間分割によって暗黙的に拡大することで二次元環境における小さなオブジェクトを容易に選択する手法を提案する. この手法のオブジェクト拡大は特定の操作を行わずに実行されるため, 拡大操作による選択時間の増大を招かずに選択精度を向上させることが出来る. 二つ目の研究では, 頭部方向に対して極端な角度の領域で視線インタラクションを行うことで Midas Touch を解消する手法を提案する. 選択の意図の有無に関わらず環境内のオブジェクトに視線が乗ることが Midas Touch の原因である. ゆえに意図的に目を動かさない限り視線が届かない領域にのみ選択可能なオブジェクトを配置し, 選択意図を明確化することで Midas Touch を解消可能だと考えた. 三つ目の研究は, オブジェクトを仮想的に再配置することで三次元環境における小さなオブジェクトを選択する手法を提案する. 本手法ではターゲット候補を大まかに選択し, それらをターゲット候補とは別の領域に選択しやすいように再配置する. このアプローチにはターゲット候補とは明確に異なる領域が必要である. ゆえに研究 2 で検討した領域を活用することでこのアプローチの実装を可能とした. 図 1.2 にそれぞれの研究のイメージを示す.

### 1.3 本論文の構成

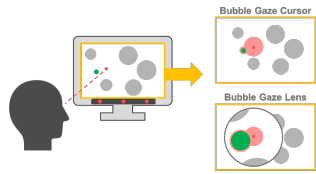
本論文は全 7 章で構成される. 第 2 章以降の構成を以下に示す.

#### 第 2 章 関連研究

第 2 章では目の生理的特徴や視線トラッキング手法について触れ, 今まで提案されてきた既存の視線入力手法の概観や各入力手法の性能の比較を示す.

### 第3章

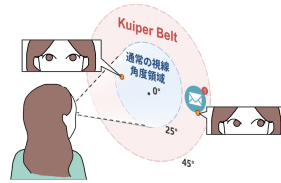
#### 空間分割による暗黙的なターゲット拡大手法



課題1) [二次元環境]  
小さなターゲットの選択が難しい

### 第4章

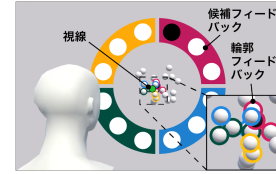
#### 極端な視線角度を活用したMidas Touch低減手法



課題2) [三次元環境]  
意図せぬ誤選択が生じる

### 第5章

#### 仮想的な再配置によるターゲット選択手法



課題1) & 課題2) [三次元環境]  
意図せぬ誤選択を避けつつ、  
小さなターゲットの選択が難しい

図 1.2: 本論文で行なった研究の概要. 3-5 章にてそれぞれの課題の解決を試みる.

### 第3章 Bubble Gaze Cursor + Lens: エリアカーソル手法の視線操作への拡張

第3章では視線インタフェースにエリアカーソルを適用することで二次元環境における小さなオブジェクトの選択を容易にする手法の提案を行った. 本研究ではマウスインタフェースにおいて小さなオブジェクトの選択を容易にする手法であるエリアカーソル手法を視線インタフェースに導入することを検討した. 本研究ではエリアカーソル手法の一種である Bubble Cursor と Bubble Lens を視線インタフェースと組み合わせた手法である Bubble Gaze Cursor, Bubble Gaze Lens を実装した. 提案手法の評価を行う実験の結果, これらの手法は小さなオブジェクトの選択精度を向上させることを確認した.

### 第4章 Kuiper Belt: VR における自然ではない視線角度を用いた視線入力手法の提案

第4章では視覚探索中の Midas Touch が生じない三次元視線メニュー UI のデザインを行なった. 目の水平方向の最大可動範囲は平均  $45^\circ$  である. しかし目が限界まで動くことはほとんどなく, 視線は基本的に視線と頭部方向が成す角度 (視線角度)  $25^\circ$  以内に分布している. 我々はこの  $25^\circ$ – $45^\circ$  の領域を Kuiper Belt と名付けた. 当領域はユーザが意図的に目を動かさない限り視線が移動することはほとんどない. ゆえに Kuiper Belt を活用することで, VR における視覚探索時の意図しない選択操作である Midas Touch が減少すると考えられる. 本稿では Kuiper Belt で視線インタラクションを行うためのパラメータ設計を行う実験と, Kuiper Belt を用いた手法の有用性と負担を検討する実験を行った. 実験結果より, Kuiper Belt を活用することで視覚探索時の選択操作の高速化と誤入力の減少が可能であることが示された.

## 第 5 章 Asteroid Gazer: VR における視線のみを用いた遮蔽されたオブジェクトの選択手法の提案

第 5 章では第 4 章で作成した三次元視線メニューを活用することで遮蔽されたオブジェクトの選択手法 Asteroid Gazer の提案を行った。Asteroid Gazer は視線のみで VR 環境の高密度、部分的に遮蔽されたオブジェクトを選択する手法である。この手法はターゲット候補を Kuiper Belt に配置されたメニューアイテムに仮想的に再配置することで遮蔽を解消する。その後、階層メニューのアイテムを段階的に選択することでターゲットを選択する。この階層メニューの使用によって高密度な状況においても正確な選択を可能とする。Asteroid Gazer と一般的な視線選択手法と比較する実験を行った結果、提案手法は高い遮蔽率においてエラー率を 63.5 %削減した。

## 第 6 章 総合的な議論

第 6 章では本論文で行ってきた研究について総合的な議論を行う。

## 第 7 章 まとめ

第 7 章では本論文のまとめを行う。さらに本研究の発展的な話題についても踏み込み、本研究分野の将来についての議論を行う。

## 第2章 関連研究

本章では本論文の理解に必要な知識や関連研究を示す。なお、別の章の研究内容に近い関連研究は、わかりやすさのために別途その章に追加で示す。なお、本論文ではオブジェクトの大きさの指標として「視角」を用いる。視角とは目に投影される物体がなす角度のことであり、人から見たオブジェクトの大きさを示している。大きさの尺度はオブジェクトに対する実測値が一般的ではあるが、目との距離で視覚的な大きさは変わるため、以後視角を用いてオブジェクトサイズを示す。

### 2.1 目の生理的特徴

人は外界からの入力を目の網膜で捉えることで、外界を見る。網膜は眼球の底に存在しており、網膜内の視細胞で光を受容する。この視細胞は網膜の中心窩中央部に高密度に分布しており、中心窩内にオブジェクトを捉えることでオブジェクトをはっきりと見ることができる。つまり視線とはこの中心窩の方向を指す。しかし中心窩は眼球の底に存在することから、目を外部から観察しても完全な視線方向の推定はできない。ゆえに本研究における視線は目の中心の方向と定義する。

目は人体で最も高速に動作する器官であり、眼球の速度や挙動に応じて眼球運動を分類することができる。以下にそれぞれの眼球運動の特徴を示す。

#### 2.1.1 Fixation (固視)

固視とは外界を目で捉えるために目を静止させるときの視線運動である。一般的に注視とも呼ばれる。人間の視線は注視時も完全に停止することはなく、眼球を特定の位置に固定する外眼筋によって揺れる。したがって視線が完全に固定されることはない。

#### 2.1.2 Saccade (サッケード)

注視位置を横方向、縦方向、あるいは斜め方向に変える際に生じる急速な眼球運動をサッケードと呼ぶ。移動する振幅が大きくなるほど速度は大きくなり、持続時間は振幅に線形的に近似される。サッケードは基本的に一度の弾道運動で完了されるが、最初のサッ



ケードだけではターゲットを捕捉できない場合、さらに第二のサッケードが修正的な動作でターゲットに向かうことがある [13, 208]. これらの副次的に行われるサッケードを修正サッケードと呼び、一度のサッケードでは捕捉し難いターゲット、つまりターゲットが小さい場合やターゲットまでの距離が遠い場合に生成されやすい.

### 2.1.3 Smooth Pursuit (滑動性眼球運動)

この眼球運動は運動する視覚対象を追うときに生じる眼球運動であり、 $30^\circ/s$  まで可能だとされているが、 $100^\circ/s$  まで可能だとする主張も存在する [127]. 文字通り非常に滑らかな視線起動を描き、基本は視覚対象が存在する場合にのみ生じるが、暗闇で自分の手の動きを目で追うことができるなど、脳内のイメージを目で追う時にも生じることがある [45].

### 2.1.4 Vergence (輻輳開散運動)

視対象の奥行き方向の変化に対応して生じる眼球運動である. つまり上記の眼球運動が上下左右に対する視線移動である一方で、輻輳開散運動は奥行き方向の運動である. 視線を遠い位置から近い位置に移動する眼球運動を輻輳運動 (Convergence), 近い位置から遠い位置に移動する眼球運動を開散運動 (Divergence) と呼ぶ.

## 2.2 視線計測手法と精度

視線計測 (アイトラッキング) に関する研究は 100 年以上前から行われてきた. 西暦 1908 年, Huey はコンタクトレンズにポインタを装着することで眼球運動を検出する視線計測手法を考案した [67]. この手法は非常に身体負荷が非常に高く、実用的な手法とは言い難い. ゆえに、高い視線推定精度と低い身体負荷を目指し、様々な視線計測手法が提案されてきた.

視線計測手法の精度は Accuracy と Precision でそれぞれ評価される. Accuracy は視線推定誤差であり、記録された視線データと実際の視線位置の誤差で計算される. Precision は注視中の視線の揺らぎであり、視線データの標準偏差で計算される. つまり Accuracy が大きいと注視対象とは離れた位置に視線が分布し、Precision が大きいと注視対象内に視線データを収め続けることが困難となる. 以下では視線計測手法と、その精度を紹介する.

### 2.2.1 角膜反射法

角膜反射法は角膜に照射された光源の反射像を用いて視線を推定する手法である [18]. 市販品では主に赤外線が光源として使用されている. 光源を角膜に照射すると 4 つの反射

像が現れ、その中で最も明るい反射像である角膜全面の反射像が視線推定時に主に使用される。しかしこの反射像のみを用いる場合、眼球運動と頭部動作を判別することができない。そのため、頭部にカメラを固定するなどの処理 [116] が必要となるが、それは侵襲性が高く、使用用途が限定される。ゆえに、この課題を解消するために、水晶体後面の反射像である第4プルキンエ像を利用する手法が提案された [24]。本手法は眼球運動が回転運動である一方で頭部運動が並行運動である事実を活用しており、約  $1'$  ( $0.017^\circ$ ) という非常に高精度な Accuracy を達成することが可能である。一方で、第4プルキンエ像を検出するために高輝度の光源が必要であるほか、装置が複雑で調整に時間がかかるといった欠点を有しており、市販品ではほとんど用いられていない。

現在最も使用されている視線計測手法は、角膜反射像と瞳孔中心を検出する手法 [126] である。本手法では画像解析で瞳孔中心を検出し、その点を参照して頭部運動を無視する。本手法は頭部の移動が 2–3 cm の立方体内に収まる場合、ディスプレイへの注視の Accuracy は  $0.5^\circ$  程度と十分な精度である。ゆえにユーザへの負荷の低さや装置の簡易さから、アイトラッカの多くは本手法を採用している。ただし、キャリブレーション（視線データを個人に合わせて補正する作業）手法の都合上、キャリブレーション用の視線データを収集した点から離れるほど精度は悪化し、ディスプレイの端に近づくほど Accuracy は悪化する。また、三次元環境を対象とする場合、角膜反射法の Accuracy は大きく低下する。これはキャリブレーションポイントとの三次元的な距離がポイントから離れば離れるほど Accuracy が低下するため、ディスプレイ環境よりもキャリブレーションポイントとの距離が大きくなりがちだからである。そのため VR 用 HMD の VIVE Pro Eye や MR 用 HMD の Microsoft HoloLens 2 の視線推定精度は低く [179, 4]、かつ頭部正面から遠ざかるほど Accuracy は悪化する [179]。図 2.1 に角膜反射法のイメージを示す。

### 2.2.2 顔画像を用いた機械学習視線検出法

顔や目の画像から機械学習を用いて視線を推定する手法が提案されている [8, 145, 229]。これらの手法は大きくモデルベース (Model-based) とアピランスベース (Appearance-based) に二分される。モデルベース手法は瞳孔や眼球中心、目尻などの目の特徴を検出し、3D 眼球モデルに対してそれらに対応させることで視線を推定する手法である [22]。初期の研究では角膜反射法のように赤外光源を用いていた [71] が、最近の研究では RGB 画像から機械学習で目の特徴を高精度に検出することが可能となっている [191]。アピランスベース手法はモデルベース手法とは異なり、眼球モデルなどを使用せず、画像から直接視線を推定する [188]。本手法は目の特徴の検出を行わないため、モデルベースよりも

<sup>1</sup><https://www.tobiipro.com/ja/service-support/learning-center/eye-tracking-essentials/how-do-tobii-eye-trackers-work/>

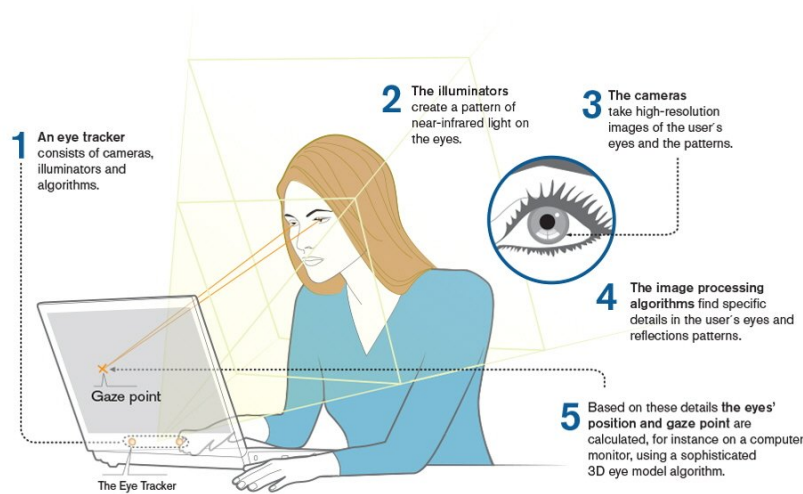


図 2.1: 角膜反射法のイメージ<sup>1</sup>.

低い画質の画像から視線を推定することが可能である。これらの手法は目元の画像のみが必要であるため角膜反射法よりもユーザの姿勢に対して寛容である。また、機械学習技術の向上に伴い視線推定精度が向上すると考えられるため、今後の技術発展が期待される手法だと言える。図 2.2 に機械学習による視線推定手法のイメージを示す。

機械学習による視線推定手法の Accuracy は、基本的に角膜反射法よりも低い。角膜反射法の Accuracy が  $0.5^\circ$  であるのに対し、機械学習手法の視線推定精度は高くとも  $2.0^\circ$  程度である。しかし、角膜反射法を用いる場合はアイトラッカから 50–80 cm 以内に顔を配置する必要があるのに対し、機械学習手法はカメラまでの距離が大きく空いていても視線を推定することが可能である。特にアピアランスベース手法はモデルベース手法よりも距離に対してロバストであり、2.0 m ほど離れた場合でも Accuracy が低下しないことが明らかになっている [235]。

### 2.2.3 その他の視線検出法

角膜反射法や機械学習視線推定手法などが一般的ではあるが、状況によってはこれらの手法よりもその他の視線推定手法が優れている場合がある。ゆえに、上記の 2 手法以外の視線推定手法を以下に示す。

Electro-oculography (EOG) 法は眼球運動による電圧変化を検出することで視線を推定する手法である [190]。本手法は眼球を観察する必要がないため、目を閉じていても視線方向を推定することが可能である。本手法はカメラや光源などを用いずに視線を検出す

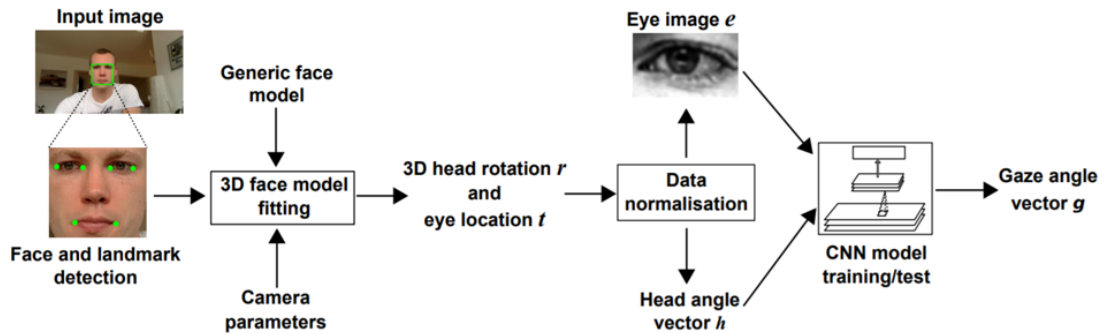


図 2.2: 機械学習による視線推定手法の例 [229].

ることが可能であるが、 $1.0^\circ$  以下の Accuracy を達成することが難しく、また長時間の使用によって精度が下がってしまう。また、電圧変化を検出するためには電極を顔に配置する必要があるため、使用用途は大きく制限されていると言える。

サーチコイル (Search coil) 法はコンタクトレンズにコイルを取り付け、交流磁場内の誘導電流を検出することで視線を推定する手法である [161]。本手法は非常に高精度に視線を検出することができ、Accuracy は  $0.1^\circ$  ほどである。本手法はコンタクトレンズに鏡を取り付ける手法であるオプティカルレバー (Optical lever) 法 [219] よりもコンタクトレンズが軽量であり、比較的体負担は低い。しかしながら未だ装着負担は大きく、装着時に表面麻酔剤などが必要である [86]。また装着者は交流磁場内にいる必要があるため、実験室実験などの用途以外には向いていない。

## 2.3 モノモーダル視線インタフェースの入力手法

視線のみで入力を行うインタフェースをモノモーダル視線インタフェースと呼ぶ。モノモーダル視線インタフェースの入力手法は用いる眼球運動に応じて分類することができる。一般的な視線入力手法を以下に示す。

### 2.3.1 滞留時間選択

滞留時間選択とはモノモーダル視線入力において最も一般的なターゲット選択手法であり、視線を一定時間ターゲット上に滞留させる (注視する) ことによる選択手法である。この選択に必要な注視時間を滞留時間と呼ぶ。滞留時間選択は非常に直感的な入力方法であり、また滞留時間を 150 ms に設定することでマウスよりも高速な入力を可能にする [172]。一方で短すぎる滞留時間は意図せぬ誤選択 (Midas Touch) [78] を頻発させるため、滞留時間は 450 ms から 1000 ms の間の値に設定されることが一般的である [119]。このよう

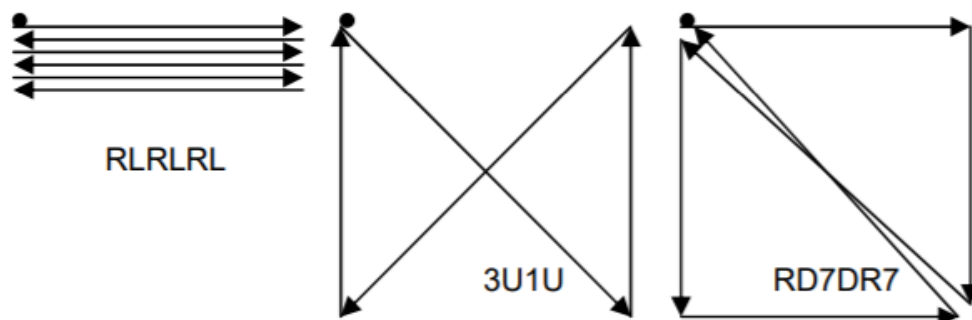


図 2.3: Drews らの提案した視線ジェスチャの例 [31].

に滞留時間を長くすることで Midas Touch を解決することが可能だが、代わりにタスク完了時間が長くなり、マウスよりも入力が遅くなる。ゆえに Midas Touch が起こらない程度にユーザによる滞留時間の手動調整 [117] や入力される可能性に応じたキーごとの滞留時間調整 [137] など、滞留時間を減少させる手法が提案されている。

滞留時間選択手法は非常に直感的な手法であり、視線インタフェースで最も使用されている手法である。ゆえにブラウザ操作 [102] やアイタイピング [118], VR 環境におけるターゲット選択 [189, 25] など、様々な場面において真っ先に滞留時間入力が検討されてきた。そして単純な滞留時間入力では難しい入力の実行や、抱える課題の解決を目的とした手法が提案されてきた。

### 2.3.2 視線ジェスチャ

視線ジェスチャとは視線で特定の軌道を描くことで入力を行う手法であり、サッケードを用いた手法として解釈することができる。Isokoski によるディスプレイ外のターゲットを用いた視線軌道による入力 [72] など視線軌道を用いた入力は以前から行われていたが、Drewes らが特定の視線軌道を入力とする視線ジェスチャを提案したこと [31] がきっかけで様々な視線軌道を用いた手法が盛んに提案されるようになった。図 2.3 に Drewes が提案した視線ジェスチャの例を示す。

視線ジェスチャ手法は単純に視線で軌道を描いて入力を行う手法と、オブジェクトへ視線を移すことで視線軌道を生成する（特定のオブジェクトを順に滞留時間入力で選択する）手法に二分される。前者は入力を行う場所やタイミングに一切の制限がないため、いつでもどこであっても視線ジェスチャを実行することができる [31, 44]。前者の場合、日常的に行う視線動作と視線ジェスチャが一致する場合 Midas Touch が生じるため、複雑な軌道を視線ジェスチャとして採用する必要がある。しかし、複雑な視線ジェスチャを行うことはユーザに対して大きな負荷が生じるほか、複数の視線ジェスチャを用意することは困難で

ある。後者は特定の位置を一定時間凝視することを繰り返すことで、実質的に視線ジェスチャ入力を行う。使用する位置はディスプレイ内に存在することが一般的 [212, 155, 134] ではあるが、ディスプレイ画面外への注視 [72, 75, 85] など様々な位置が検討されている。この手法は視線軌道の他に入力位置も参照しているため、Midas Touch の危険性が低い。Istance らは画面中央から特定の位置を見た後に中央に視線を戻す視線ジェスチャ（二本足ジェスチャ）と、二箇所視線を移した後に中央に視線を戻す視線ジェスチャ（三本足ジェスチャ）を実装した [76]。これらのジェスチャが 15 分間の PC ゲーム中に誤作動することはほとんどなく、特に三本足ジェスチャは一度も実行されなかった。ゆえに多くの視線ジェスチャ手法は後者の入力方法を採用しており、視線ジェスチャを用いたキーボード入力手法 [212, 155] やゲーム操作手法 [75]、車椅子操作手法などが提案されている [134]。

### 2.3.3 Smooth Pursuit

Smooth Pursuit とは中速度で動くオブジェクトを目で追うことで入力を行う手法であり、滑動性眼球運動を用いた入力手法である [195]。この手法では、ターゲット軌道と視線軌道の近似度を計算し、閾値以上の場合にターゲットが選択される。ゆえに、画面上に配置された軌道の異なる動的ターゲットを目で追うことで、ユーザは取得したいターゲットを任意に取得することが出来る。本手法は選択可能なターゲットの量が多くなればなるほど精度は下がるため、初期の研究においては 8 個程度が限界 [34] であったが、様々な近似度計算アルゴリズム [192] が検討された結果最大 160 個のターゲットを判別できるようになった [30]。一方で Smooth Pursuit は高速な入力に適しておらず、滞留時間よりも長い入力時間を必要とする [193, 195]。しかし、キャリブレーションを行う必要がないことから、キャリブレーションが難しいパブリックディスプレイに対する視線インタラクションに広く用いられている [195, 153, 89, 236]。またターゲットサイズに一切影響を受けない手法であることから、スマートウォッチ [34] や VR 環境 [88] の小さなオブジェクトとのインタラクションに Smooth Pursuit を用いる手法も提案されている。図 2.4 に Smooth Pursuit のイメージを示す [195]。

### 2.3.4 視線深度情報を用いた入力手法

上記の入力手法は視線の移動速度ごとの特徴を活かした実装となっており、多くのモノモーダル視線入力は上記のどれかに属する。しかし、上記の手法は  $x$ - $y$  軸（顔に対する上下左右）における移動速度を参照しており、 $z$  軸方向（顔に対する奥行き方向）における視線移動は考慮していない。人間の目は一般的に二つ存在し、それらの目から発射される視線が交差する点を焦点と呼ぶ。この焦点の奥行き（視線深度）を変える眼球運動を輻輳

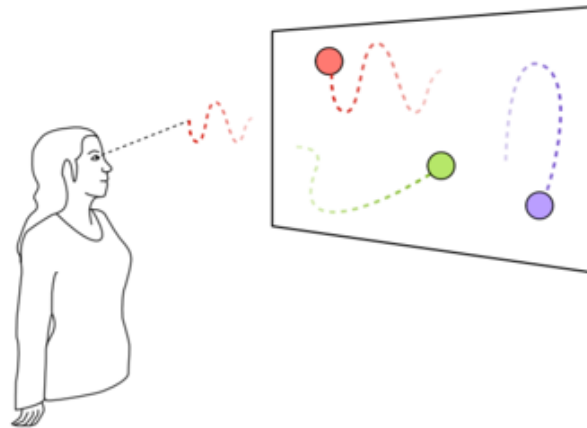


図 2.4: Smooth Pursuit のイメージ [195].

開散運動と呼び、視線深度の値に応じて入力を行う手法が提案されている。

視線深度を入力に用いる手法では、特定の深度を凝視することでオブジェクト選択 [94] やブラウザ操作 [60], スクロール [144], 壁の透過 [205] など様々な手法が提案されている。視線深度を用いた入力手法は、「選択対象の深度」と「入力に用いる視線深度（入力深度）」が一致しているかどうかで大別できる。一致している手法 [173, 120] は入力を行うと、選択対象をそのまま選択する。これらの手法ではオブジェクトを直接見るだけで視線深度を容易に合わせることが可能である。一致していない手法は、入力時の視線方向のオブジェクトに対してインタラクションを行う手法が多い。これらの手法では入力深度に自発的に視線を移動する必要がある。基本的に視線深度は自発的に変更することは難しく、特に輻輳運動（近くを見る）よりも開散運動（遠くを見る）の方が難しい [91]。また、開散運動を行うことが出来たとしても、視線を開散したまま固定し続けることは困難である [205]。ゆえに視線深度の固定を容易にし、かつ視線方向を選択対象に正確に向ける方法として、入力深度に目標物を配置する手法 [61, 94] がある。Kudo らはディスプレイの前に鏡を置くことでディスプレイ内に表示されたターゲットの奥に目標を表示し [94], Wang らはユーザから 6.0 m の位置に固定された視線カーソルを目標とし [205], Zhang らはユーザの視界中心（頭部正面）に目標を固定した [227]。

これらの視線深度を用いる入力手法は単純な入力のみが可能であり、視線深度の遠近による二種類の入力操作か、一定の深度よりも近いか遠い時に入力を行う手法が一般的である。これは視線深度の Accuracy と Precision が注視点が遠距離になるほど低下し、視線深度の予測値は実測値よりも近い深度に偏る [202] ためである。ゆえに視線深度を用いた入力手法に関する研究の数は少なく、実用性には乏しい。しかし視線方向を変えずに入力することが可能である点や、日常では用いない視線深度を活用することで Midas Touch を

回避することが可能な点から、今後の研究による発展が期待される手法である。

## 2.4 マルチモーダル視線インタフェースの入力手法

### 2.4.1 視線と手を組み合わせた入力手法

視線モダリティと他のモダリティを組み合わせて入力を行うインタフェースをマルチモーダル視線インタフェースと呼ぶ。複数のモダリティを組み合わせることで単一のモダリティでは困難な入力や、単一モダリティによる入力よりも高速または高精度な入力を可能とする。視線と組み合わせられることが多いモダリティは手である。GUI などにおいてはマウスなどの手を用いるインタフェースで高速な入力を達成することができるが、視線と組み合わせることでより高速な入力が可能となる。Zhai らはマウス入力と視線を組み合わせた入力手法である MAGIC を提案した [226]。本手法はマウスの移動を視線で一部担う手法であり、マウス移動より視線移動の方が高速であることから、一般的なマウス入力よりも高速な入力が可能である。またフリーハンドによる入力は遠距離に対する入力が困難であることから、視線と併用して遠距離のオブジェクトとインタラクションする手法が提案されている [186, 184, 185, 149, 150, 223, 99]。Stellmach らは手元のデバイスからの入力と視線を組み合わせて遠方のディスプレイとインタラクションする手法を提案した [186, 184, 185]。また Pfeuffer らはタッチサーフェスに対するタッチ入力と視線を組み合わせた入力手法を提案し、通常のタッチ入力と視線と組み合わせたタッチ入力を併用する手法を提案した [149, 150]。これらの手法は視線上のオブジェクトに対して手で入力を行う手法であり、視線を用いることで手を動かさずに遠距離のオブジェクトとインタラクションを行うことが可能である。

また VR や MR 環境などの三次元環境を対象とした入力手法も提案されている。VR においてはコントローラを持っていることが多いため、コントローラと視線を用いた手法が提案されている [223, 21, 69, 99, 151]。コントローラは視線と同様にレイを照射することができるため、コントローラのみで遠距離のオブジェクトを選択することができる。しかしレイ上のどのオブジェクトを選択したいか不明であるため、視線とコントローラのレイの交差点を計算してターゲットを明確化する手法などが提案されている [21]。MR 環境においては日常生活におけるインタラクションを想定しているため、コントローラのような外部デバイスを持ち歩くことが想定されず、フリーハンドで入力を実行することが多い。フリーハンド入力を頻繁に行うことは身体的な負荷が大きいため、手と視線を組み合わせた快適な入力手法が検討されている [111, 110, 200, 152]。これらの手法はハンドジェスチャと視線を組み合わせた場合の入力を複数提案しては比較検討を行っており、MR 環境における最も容易なフリーハンド視線入力手法を探索している。事実 Apple 社が発売す



る予定である HMD の Apple Vision Pro はこの手と視線を組み合わせた入力を標準搭載しており、今後は手と視線を組み合わせた手法が一般的になる可能性が高い。

一方で手と視線を組み合わせる手法は、手を用いないという視線インタフェースの利点を失うことにつながる。そのため料理や掃除といった日常生活における手が塞がった状況では、上記の手法は用いることはできない。ゆえに手以外のモダリティと視線を組み合わせる手法の提案が必要だと言える。

#### 2.4.2 視線と頭部動作を組み合わせた入力手法

マルチモーダル視線入力手法のうち、視線と頭部動作を組み合わせた入力手法は手を使わないうえ、視線と頭部動作の両方を HMD で容易に追跡できる。ゆえにこの入力手法は VR・MR に適した手法であり、他のモダリティを組み合わせた手法よりも多く検討されてきた。頭部動作を用いた入力でも最も直感的な手法は、視線方向のオブジェクトに対して頭部ジェスチャで入力を行う手法である。これらの手法は頭部の前方への傾き（頷き）や横への傾きを行うと、視線方向のオブジェクトに対して入力が行われる [83, 154, 141]。Ju らはピアノ演奏中に、視線の先の楽譜を頭部動作によってめくる操作インタフェースを作成した [83]。本手法はピアノ演奏という首から下が動かせない状況で使える上、音声入力が適していない状況に適した手法であるため、視線と頭部動作がこの状況において最適なモダリティだと言える。

また、その他に頭部動作を入力に用いる場合、頭部動作の変位をカーソルなどに反映する手法や、頭部方向と視線方向の関係性を用いた手法が提案されている。前者は視線で大まかにポインティングを行い、頭部動作で細かくポインタ位置を調整する手法が提案されている [99, 97, 81, 198]。基本的に視線と比べて頭部動作のポインティングは精度が高いため、これらの手法では高速な視線ポインティングを正確な頭部動作で補正することが可能である。後者は頭部方向と視線方向を活用した手法であり、これらの方向の差を入力に活用している [175, 177, 121]。Sidenmark らは滞留時間入力の開始タイミングを視線方向と頭部方向が一致することとする手法を提案した [175]。この手法は正面を見据える時以外は Midas Touch が生じないため、視界内を概ね自由に探索することができる。また目を固定して頭部を動かすことを入力動作とすることで、スライダ操作など視線のみでは難しい操作を実行することが可能である [177]。これらの手法の多くは目と頭の性質に着目している。例えば 20° 以内の視線シフト時には基本的に頭は動かないため、頭を動かすことを注視位置の更新とする手法などが提案されている [175]。ゆえに目と頭の位置関係に関する性質を活用することで、視線のみでは難しかった操作を実行することが可能になると考えられる。

### 2.4.3 視線とその他のモダリティを組み合わせた入力手法

その他のマルチモーダル視線入力手法では、ポインティング動作と選択動作を別々のモダリティに実行させることが多い。特に視線モダリティはその高速性からポインティング動作を担うことが多い。選択動作を他のモダリティで実行することのメリットは Midas Touch を回避することができる点である。ゆえにフットジェスチャ [55] や顔面筋電 [187, 122], 瞬き・ウインク [158] など、様々なモダリティの組み合わせが検討されてきた。

また、発話と視線を組み合わせた手法も多く検討されている。発話の内容が問われない (ノンバーバル) ような手法 [56] も存在するが、基本的に視線方向に対して発話のワードに応じた入力が行われる手法が一般的である。Miniotas らは視線の先の複数のオブジェクトの輪郭に異なる色を表示し、その色を発話することで色に対応したオブジェクトを選択する手法を提案した [133]。また視線の先のオブジェクトに対して発話コマンドを実行する手法も提案されている [33, 204]。以前までは単語のみの単純な発話コマンドのみが使用されていたが、近年では音声認識技術の向上に伴い長文の理解が可能になり、文構造の発話と視線を組み合わせる手法も提案されるようになった。Mayer らは「あれ」や「これ」といった発話中の指示語を視線の先のオブジェクトに置き換える手法を提案した。これにより、ユーザは曖昧な単語を用いながら音声入力が可能となった [123]。また Lee らは Mayer らの手法を改善し、音声認識した文章を ChatGPT に渡すことで、視線の先のオブジェクトを参照しながら ChatGPT と日常生活中にインタラクションが可能となる手法を提案した [104]。

## 2.5 視線入力手法の性能比較

### 2.5.1 視線入力手法ごとの比較

ここでは滞留時間入力と視線ジェスチャ、Smooth Pursuit を比較する。

#### 滞留時間入力

滞留時間入力手法はターゲットを注視することで選択する手法であり、最も直感的に選択を行うことができる手法である。視線で入力を行う際に真っ先に想起する手法であり、他手法よりも習得難易度が非常に低い。また目を動かす必要がないため、他手法よりも負荷が小さな手法だと言える。そのため視線インタフェースの多くは滞留時間入力手法を採用しており、総合的に鑑みて三手法の中で最も優れた入力手法だと言える。しかし、小さなターゲットの選択が難しいことや、Midas Touch が生じることなど欠点は多く、それゆえに特定の状況においては他手法の方が優れている場合も多い。ゆえにこれらの課題を解

決することが重要だと言える。

## 視線ジェスチャ

視線ジェスチャは特定の視線軌道を入力とする手法であり、理想的にはあらゆる種類の入力を実行可能である。Wobbrockらは視線ジェスチャによるキーボード入力手法にて、4点の視線経由点を用いた26種類の視線ジェスチャを作成した[212]。ゆえに視線ジェスチャは他手法よりも入力の設計自由度が高い手法だと言える。また視線ジェスチャは他の手法と異なり、注視対象を必要とせず、何もない空間においても視線ジェスチャを実行することが可能である。しかし、裏を返すと、三手法のうち視線ジェスチャのみが入力対象を規定しない。視線ジェスチャはあくまでコマンド入力手法であり、その入力をどの対象に実行するためには別を選択する操作を行う[73]か、特定の選択方法を用意する[140]必要がある。そのため視線ジェスチャは特定の操作（キーボード、ゲームの自キャラ操作、など）を行う際の入力手法に適しており、全ての入力を担うことは難しい。一方で視線ジェスチャが他の手法に比べて完全に劣っているわけではない。Hyrskykariらはゲームプレイ中におけるコマンド選択において、滞留時間入力よりも視線ジェスチャの方がエラー率や選択速度の点で優れていたことを報告した[68]。ゆえに視線ジェスチャ手法は特定の状況においては他手法よりも適している可能性がある。

## Smooth Pursuit

Smooth Pursuitは動く対象を目で追うことで入力する手法であり、キャリブレーションの必要がない手法である。ゆえにSmooth Pursuitはキャリブレーションの実行が難しい状況（通行人に対するインタラクションなど）においては滞留時間入力手法よりも優先される手法である。またこの手法はターゲットサイズに影響されないため、小さな表示領域、例えばスマートウォッチなどにおける視線インタラクションに適している[34]。この研究ではSmooth Pursuitの適切な軌道直径やオブジェクト個数が検討されており、結果として $2.63^\circ$ の円軌道に8個の選択可能オブジェクトを動作させることができることを確認した。つまり、滞留時間入力においては一つのターゲットのみを配置できる大きさの領域に対して、Smooth Pursuitは8種類のターゲットを配置することができる。したがって入力領域に対する入力多様性はSmooth Pursuitが滞留時間入力手法よりも優れている。しかしSmooth Pursuitは視線軌道とオブジェクト軌道のマッチングを常に行なっているため、選択可能なオブジェクトの配置可能な数は限定される。つまりインタラクション領域内に9個以上のオブジェクトを配置することが推奨されない。また入力方法の性質上、動いていないオブジェクトの選択は困難である。ゆえにSmooth Pursuitを使用するには

専用の UI が必要であり、この入力手法を利用できる状況は限定的である。

### 2.5.2 視線以外のモダリティとの比較

視線入力手法と他モダリティの入力手法を比較する場合、Fitts' Law Task などのターゲット選択タスクを実施し、選択速度や選択精度、それらを総合的に鑑みたスループットなどを計算することで評価する。これらの研究では滞留時間入力手法か視線ポインティングとデバイス入力を組み合わせた手法が視線モダリティの手法として採用されることが一般的である。これらの比較実験は多数行われており、見解が一致する結果や矛盾する結果が得られている。

まず、視線は最も高速にターゲットに到達するモダリティである。選択動作はターゲットのポインティング動作とトリガ入力などの選択実行動作に二分され、このポインティング動作が視線は高速であることが明らかになっている。Sidenmark らは視線と頭部方向、コントローラのポインティング性能の比較実験を行った [178]。その結果、ターゲットサイズやターゲット距離、ターゲット生成位置の事前知識の有無などの全ての条件において、視線は最も高速にターゲットに到達する手法であった。

一方で視線入力による選択速度はポインティング速度ほど速くはない。まず手を用いた入力方法（マウス、コントローラなど）は視線よりも高速な選択が可能である [25, 178, 129, 39, 135]。そして頭部方向を用いた入力方法も視線より高速な選択時間を達成する場合があった [129, 157]。これは視線が頭部方向よりも Accuracy や Precision が低いことに起因しており、視線がターゲット外に出ることで選択時間が増加していると考えられる。事実、どの研究においても視線入力は頭部方向入力よりもエラー率が高く、それが視線入力手法のスループットの低さに直結している。一方で頭部方向よりも視線の方が選択時間が速く [14, 178, 39, 135]、スループットとしても頭部方向より視線の方が高い [39, 135] という研究報告も確認されている。ゆえに頭部方向と視線は現状では同程度の入力性能を有していると考えられる。ただし、視線推定精度が向上するほど視線インタフェースの入力性能は向上するため、将来的には視線インタフェースの方が頭部方向よりも優れた入力手法になると考えられる。

## 2.6 本章のまとめ

本章では視線インタフェースによる入力手法について概観し、それらの手法の比較や他の入力インタフェースとの比較を行なった。本章にて紹介した研究は視線インタフェースに関する主要な研究であり、今後の研究の理解に必要な基本的な概念の理解を目的として紹介を行なった。視線インタフェースが抱える課題の解決を目指した手法や研究に関連す

る具体的な内容は、研究に関する章で関連研究として追加で紹介する。  
次章からは具体的な研究を段階を追って述べていく。

## 第3章 Bubble Gaze Cursor + Lens: エリアカーソル手法の視線操作への拡張

滞留時間選択には小さなターゲットの選択が困難であるという課題がある。この課題は固視微動（人間の意図とは関係なく不随意に起こる非常に微小な眼球運動）やアイトラッカーの精度の限界 [114] などが原因で、ターゲット内に視線を固定し続けることが難しいため起こる。この問題はオブジェクトサイズが十分に大きい専用の UI を用いることで回避することが可能である。しかし、VR や MR などの三次元環境におけるターゲット選択や、ディスプレイが遠い位置に存在する時の視線入力、マウス用の UI で視線選択を行う場合など、小さなターゲットを選択しなければならない状況は多数存在し、単純な UI デザインで回避することは難しい。

小さなターゲットを選択する一般的な方法は、ターゲットをズームなどの操作によって拡大した後に選択することである [102, 3, 52, 125]。これらのターゲット拡大手法は拡大操作を行った後、その拡大されたターゲットの選択を行う。しかしこの選択動作は拡大操作の分だけ選択時間が増大するため、頻繁に行う入力手法としては不適切である。ゆえに選択時間を増大させず、かつターゲットを拡大して小さなターゲットを選択する手法として、視線インタフェースにエリアカーソル手法 [84, 214] を導入することを検討する。エリアカーソルは暗黙的にターゲットを拡大する手法であり、ユーザが拡大操作を行わずともターゲットが拡大される。ゆえに通常の滞留時間入力と同様の入力方法で小さなオブジェクトを選択することができる。本論文では既存のエリアカーソル手法である Bubble Cursor [49] と Bubble Lens [137] を視線インタフェースに導入した手法、Bubble Gaze Cursor と Bubble Gaze Lens を提案する（図 3.1）。

Bubble Cursor はカーソルに最も近いオブジェクトを含むようにカーソルの大きさが変化する手法であり、マウスにおける小さなオブジェクトの選択性能を向上させることができる。Bubble Lens は Bubble Cursor で選択が困難な、密集した小さなターゲットの選択を容易にするための手法である。Bubble Lens は十分な大きさのターゲットに対しては Bubble Cursor として動作するが、閾値以上に小さいターゲットを選択する場合はターゲット付近で拡大レンズが自動的に開く。この拡大レンズによってターゲットの幅を拡大し、容易なポインティングを達成している。これらの手法はマウス入力に対する手法であるが、視線インタフェースにおいても小さなオブジェクトの選択を容易にすることができ

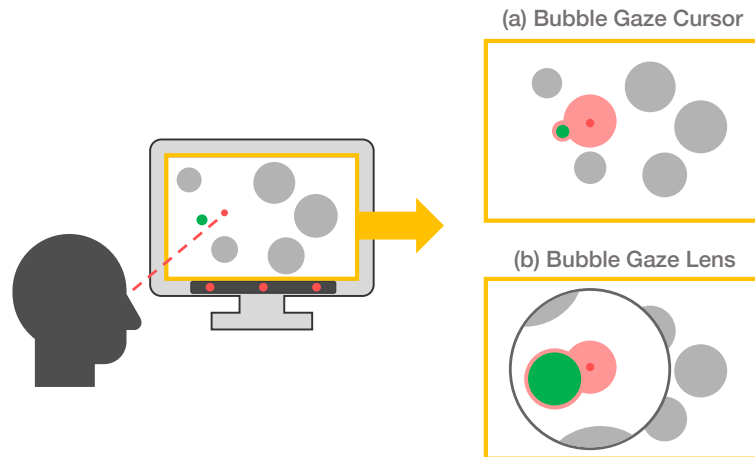


図 3.1: (a) Bubble Gaze Cursor, (b) Bubble Gaze Lens のイメージ.

ると考えた. 一方で, Bubble Cursor や Bubble Lens を視線インタフェースに実装するためには自明ではない解決すべき問題が存在する. ゆえにこれらの課題を解決し, 提案手法が一般的な視線インタフェースでは選択が困難なほど小さなオブジェクトを選択できることを確認する. 本章の貢献は以下である.

- エリアカーソル手法の一種である Bubble Cursor と Bubble Lens を視線入力に適したパラメータ調整を行い視線入力インタフェースに実装した.
- Bubble Cursor を視線インタフェースに実装した手法が視線入力手法よりも小さなオブジェクトを選択できることを確認した.
- Bubble Lens を視線インタフェースに実装した手法が Bubble Cursor を視線インタフェースに実装した手法よりも密集した小さなオブジェクトを選択できることを確認した.

## 3.1 関連研究

### 3.1.1 視線インタフェースのパフォーマンスに影響する要因

HCI の分野において最も有名なポインティングタスクの動作時間を求める予測式は Fitts' Law [41, 113] である. Fitts' Law とは, 手を用いたポインティングタスクの動作時間において, ターゲット間の距離  $A$  とターゲットの幅  $W$  の間に相反的な関係が存在することを示したものである. 最も一般的に用いられている動作時間  $MT$  の予測式 [115] は,

$$MT = a + b \log_2\left(\frac{A}{W} + 1\right) \quad (3.1)$$

であり、 $a$ 、 $b$ は定数である。

この予測式は手を用いたポインティングタスクにおいて成り立つと考えられており、視線による入力においてもこの法則が成り立つと結論付ける研究 [172, 206, 131, 194] も存在する。そして、ターゲット内部に 50 ms 滞留するまでの時間を動作時間とする場合、 $3^\circ$  以下のターゲットに対する動作時間に Fitts' Law が成り立つという主張も存在する [169]。一方で、視線を用いたポインティングタスクの動作時間が Fitts' Law と一致しない研究も存在する [226, 132, 187] ほか、Fitts' Law が成り立つとしている論文の取り扱っている変数が不適切であるという指摘 [46] や、そもそも視線は弾道的な動きをするため Fitts' Law は視線の動作時間の予測式としては不適切であるといった指摘 [29] がある。そのため現状正しい結論を得ることは出来ていない。

以上に挙げた論文は Fitts' Law が視線ポインティングタスクの動作時間予測に適しているかが議論の焦点であった。それとは異なり、Zhang らは Fitts' Law よりも視線ポインティングタスクの動作時間を高精度で予測する予測式を以下のように提案した [232]。

$$MT = a + b\left(\frac{e^{\lambda A}}{W - \mu}\right) \approx a + b \times 2000\left(\frac{e^{\frac{A}{2000}}}{W - 2\overline{AR}}\right) \quad (3.2)$$

ここで  $\lambda$  はサッケード（ターゲットに視線を向ける際の高速な眼球運動）を表現する定数であり、 $\mu$  はジッタ（固視微動によって生じる視線の不意の揺らぎ・マイクロサッケードなど）を、 $\overline{AR}$  (Average Radius) はターゲット内に視線が入ってからの軌跡の中心からの平均距離を、 $\overline{AR}$  は全ての独立変数に対して得られた  $\overline{AR}$  の平均を表している。また、同論文内で  $\lambda = 0.0005$ 、 $\mu = 2\overline{AR}$  と置くことで適切な予測式を得ることが出来るうえに、ターゲット間距離  $A$  やターゲット幅  $W$  の単位を考慮せずに用いることが出来る。

Zhang らの式では、ターゲット間の距離の増減の影響は小さい。なぜならノートパソコンやデスクトップ用のディスプレイで視線ポインティングを行う場合、ターゲット間の最大距離はディスプレイの大きさに依存するため長大な距離になることはないためである。したがって、視線におけるポインティングタスクの動作時間を減少させるための取り組みとして、ターゲット間の距離を減少させるような手法は効果的ではなく、ジッタを減少させるような手法やターゲットサイズを大きくする手法など、ターゲットの幅  $W$  を大きくするような手法が効果的であると考えられる。



### 3.1.2 必要なターゲットサイズ

人はターゲットをはっきりと見る時、網膜の中心に存在する中心窩でオブジェクトを捉える必要がある。中心窩は人の高精細な中心視野に寄与している部位であり、視角は約  $1-2^\circ$  である。視線インタフェースにおけるオブジェクトの大きさは、この視角以上の大きさが必要である [114]。また、ターゲット注視時のディスプレイ上の視点位置は固視微動によって分散が大きく、全ての視点位置を内包する領域は非常に大きくなる。Feit ら [38] はターゲットに対して注視しているときの視線データの 95 % がターゲット内に内包されるようなターゲットサイズを調査した。その結果、75 % のユーザにとって十分なターゲットサイズは、生の視線データを扱う場合は  $3.28 \text{ cm} \times 3.78 \text{ cm}$  ( $2.91^\circ \times 3.33^\circ$ )、適切な平滑化アルゴリズムを使用する場合は  $1.90 \text{ cm} \times 2.35 \text{ cm}$  ( $1.67^\circ \times 2.07^\circ$ ) であった。この大きさはマウスやタッチ操作に必要なターゲットの大きさと比較すると非常に大きい [159]。また Schuetz ら [169] は、動作時間の減少を目的として、修正サッケード（メインサッケードの後に起こるサッケード）がほとんど発生しない  $3^\circ$  以上のターゲットサイズを推奨している。Orquin ら [100] はターゲットに対する注視をターゲット内に高い割合で内包するためには、優れたアイトラッカーであっても  $5^\circ$  以上のターゲットサイズが必要だとした。ゆえに、既存のマウスやタッチ操作に適したインタフェースで小さなターゲットを視線操作で選択する場合、ターゲットサイズを拡大する手法が必要だといえる。図 3.2 にディスプレイの位置ごとの適切なターゲットサイズを示す。

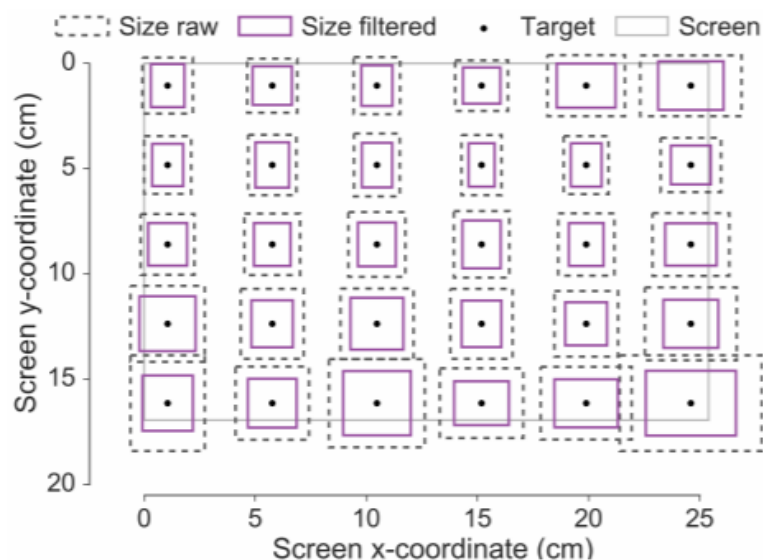


図 3.2: ディスプレイの位置ごとの適切なターゲットサイズ [38].

### 3.1.3 ジッタ・視線精度の改善手法

固視微動や視線の高速な挙動などが原因でアイトラッカから得られる生の視線データはノイズが多く、データを加工せずに視線インタフェースに用いることは難しい。ゆえに、ジッタなどのノイズを排除し、視線の精度を向上させることで容易なポインティングが達成されると考えられる。

視線の精度を検討する際に用いられる指標に Accuracy と Precision がある。Accuracy は実際の注視位置と検出された注視位置の平均誤差であり、Precision は連続した注視点の二乗平均平方根によるデータのばらつきを示している。この Accuracy と Precision が大きいほど必要なターゲットサイズは大きくなる。そこで Precision を小さくするための手法として、視線データに対する平滑化アルゴリズムが数多くの研究で提案されている [38, 196]。最も単純な平滑化フィルタは直近の注視位置  $n$  点の平均を現在の注視位置とするようなアルゴリズム [63, 79] である。また  $n$  点の注視位置ごとに重みをかけてから平均を取るようなアルゴリズム [38, 95, 82, 213] なども提案されている。平滑化フィルタを用いない手法として、ターゲットの中心に小さなドットを描画し、ターゲットではなくそのドットをユーザに凝視させる手法 [230] も提案されている。一方で、Accuracy はハードウェアやキャリブレーションの影響が大きく、視線データの後処理で Accuracy を改善することは難しい [38]。ゆえに、これらの手法では Accuracy に応じたサイズのオブジェクトのみが選択可能であり、より小さなオブジェクトを選択するためには、Accuracy と Precision の改善以外のアプローチが必要だといえる。

### 3.1.4 ターゲットサイズを拡大する手法

ターゲットサイズを増大させる手法として考えられる手法にズームによるターゲットサイズの拡大が挙げられる。ズームによるターゲット拡張はマウス操作において動作時間を有意に減少させ、かつズーム後のターゲットの幅に対して Fitts' Law が成り立つ [225, 124] ことが確認されている。このズーム機能を視線操作インタフェースに実装した研究はいくつか存在する。まず、ズーム開始のトリガーを視線以外の入力を用いる手法が提案されている [96, 10, 150, 92]。EyePoint [96] はキーボード、Zooming interfaces [10] はスイッチによる入力をトリガーとしてユーザの視点位置をズームする手法である。一方で視線のみを用いたズーム手法は少ない。なぜなら、視線が単純な入力しか行うことが出来ないためである。Lankford [102] は二段階の滞留時間を用いたズーム手法を提案した。この手法は一定の滞留時間の後に凝視していた領域の拡大領域がポップアップされ、その拡大画面からターゲットを再度凝視し選択するという手法である。類似手法として、Ashmore ら [3] はポップアップの代わりに魚眼レンズを用いた二段階ズーム手法を提案している。図 3.3

に Ashmore らの手法のイメージを示す。一方で、これらの手法を用いる場合、拡大領域によって画面内の情報損失が発生する。ゆえに、ほかのズーム手法として視線を向けた領域に対して連続的にズームを行う手法が提案されている [52, 125, 181]。これらの手法は関心領域以外が画面外に移動していくようなズームであるため、関心領域に関する情報損失が少ない。これらの2つのズーム手法を比較すると、ポップアップ式のズーム手法はヒット率が高く、逆に連続的なズーム手法は高速であった [180]。しかし、どちらも余分な滞留時間が発生することからズーム手法を用いない手法よりも遅い。

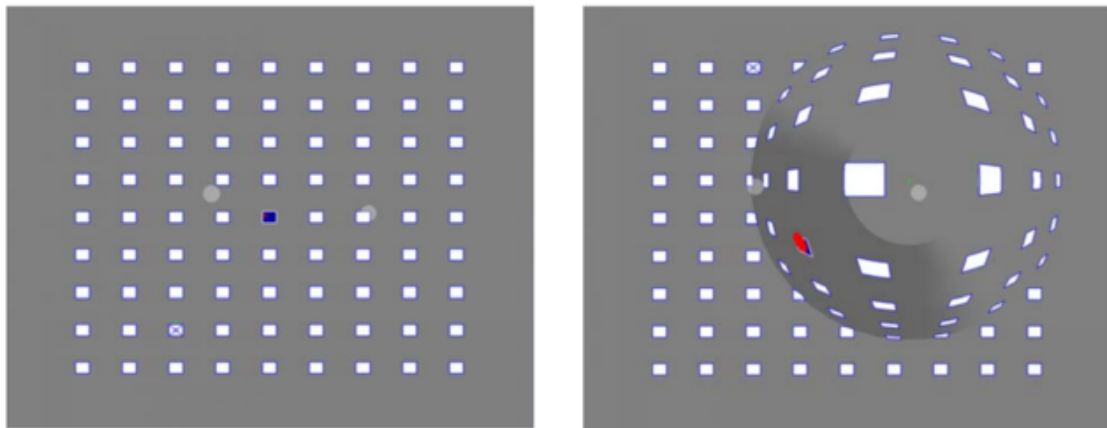


図 3.3: Ashmore らの魚眼レンズによるターゲット拡大手法 [3].

また、ズームを用いずにターゲットの幅を実質的に増大させるような手法も存在する [199, 136]。Špakov ら [199] は視線によってメニューを選択する際に、一定時間凝視したあとにメニューの大きさが拡大されるようなインタフェースデザインを提案した。この手法を用いることでエラーレートを  $1/6$  に改善することに成功した。しかし、同時に 39%の動作時間の増加が引き起こされた。図 3.4 に Špakov らの手法のイメージを示す。また、ターゲットの視覚的な大きさは変えずに判定領域だけ大きくするような手法も提案されている [132, 9]。Miniotas ら [132] は視覚的なターゲットの大きさよりも大きな判定領域を用意することで実質的なターゲット拡張を行う手法を提案した。この手法によってポインティングタスクの作業時間は有意に減少したが、この手法は静的な処理であるため、インタフェースのデザインの変更のたびに判定領域を手動で設定する必要がある。また、判定領域内に入力判定のあるオブジェクトを複数配置することが出来ないため、インタフェースデザインに制限がかかってしまう。上記で触れた手法は、ターゲットサイズを拡大しポインティングを容易にするが、動作時間の増大やデザインの汎用性の損失につながる。特に動作時間に関しては、視線が単一的な入力しか行えないという欠点から、容易なポインティングとトレードオフの関係になっている。

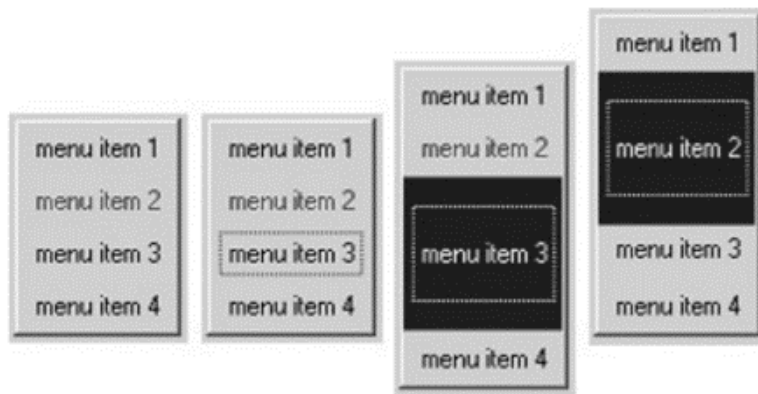


図 3.4: Špakov らのメニューの拡大手法 [199].

### 3.1.5 視線カーソル

視線カーソルとはユーザの視線位置を示すポインタであり、ディスプレイ環境においては視線とディスプレイの交点にカーソルが表示される。三次元環境においては視線がオブジェクトと交わった場合は視線とオブジェクトの交点、それ以外は特定の遠距離にカーソルが表示されることが多い。

視線カーソルの表示に関して肯定的側面と否定的側面、中立的側面が存在する。まず否定的側面としては、視線カーソルを表示すると動的ターゲットを追跡する際の精度が低下する [16] ほか、視線カーソルの存在で気が散る [47] などが挙げられる。また視線推定誤差によって実際の視線位置と視線カーソルの位置が異なることがあり、視線カーソルを見ると視線カーソルが移動し、その視線カーソルを見るとさらに視線カーソルが移動する（滑る）現象が報告されている [80]。肯定的側面としては、視線カーソルを表示すると視線方向のオブジェクトのデバイス入力選択手法の精度が向上する [2] ほか、視線カーソルの色や形でユーザに情報フィードバックを与えることが可能である [75, 64] ことが挙げられる。最後に中立的側面として、視線カーソルの有無は滞留時間を用いた視線入力のパフォーマンスに影響を及ぼさないことが確認されている [230, 2]。したがって、視線カーソルの表示による影響は功罪が存在するため、視線カーソルの表示は目的に合わせて検討されるべきである。

### 3.1.6 エリアカーソル

小さなターゲットを選択する手法として、面状のカーソル（エリアカーソル）でターゲットを選択する手法が提案された [84, 214]。一般的なカーソルは点でありターゲットの上にカーソルを正確に乗せる必要があるが、エリアカーソルは面状のカーソルのどこかがター

ゲット上に乗っていると選択可能であるため、小さなターゲットも容易に選択することができる。これらのカーソルに対してポインティングタスクを行なった結果、ポイントカーソルを用いて面のターゲットを選択することと、エリアカーソル [84, 214] を用いて点のターゲットを選択することは、選択時間の観点から同義であることが明らかになった。つまりエリアカーソルを用いることでターゲットをエリアカーソルの大きさにまで拡大していると言える。しかしエリアカーソルはカーソル自体が面積を持つため、カーソルが複数のオブジェクトを含んでしまう状況が起こり得るという課題を抱えていた。

### Bubble Cursor

エリアカーソルが有する、複数のオブジェクトをカーソルで選択してしまう課題を解決する手法として、Bubble Cursor が提案された [49]。Bubble Cursor はカーソルを常にオブジェクトを一つだけ含むような大きさに動的に変更し続けるエリアカーソル手法である。カーソルの大きさはオブジェクトを母点とするボロノイ領域を計算することで決定される。ボロノイ領域とは各母点の最近隣領域であり、Bubble Cursor はカーソルが存在する領域に対応するオブジェクトを含む大きさに変更する。つまり Bubble Cursor はカーソルから最も近いオブジェクトをカーソル内に包含する手法であり、オブジェクトの大きさをボロノイ領域の大きさに拡大する手法だと言える。実験でポイントカーソルと Bubble Cursor を比較するポインティングタスクを実施した結果、Bubble Cursor はポイントカーソルよりも有意に小さなオブジェクトを高速に選択することができた。図 3.5 に Bubble Cursor のイメージを示す。

Bubble Cursor は非常に簡易なアルゴリズムでありながら、適切なターゲット拡大を達成する。ゆえにマウスインタフェース以外にもあらゆるインタフェースに拡張されており、タッチインタフェース [201] やコントローラのレイ入力 [108] などに Bubble Cursor を適用する手法が提案されている。

### Bubble Lens

Bubble Cursor はオブジェクトをボロノイ領域の大きさに拡大する手法である。ゆえにオブジェクトのボロノイ領域がそもそも十分に大きくない場合、オブジェクトは選択を容易にするほど拡大されない。ボロノイ領域はオブジェクトが密集しているほど小さくなるため、Bubble Cursor は密集した小さなターゲットの選択が難しいと言える。ゆえに Mott らはこの問題を解消する手法として、Bubble Lens 法を提案した [137]。この手法は Bubble Cursor によって十分に拡大されないようなターゲットの付近で拡大レンズを自動的に起動することでターゲットを拡大し、十分なターゲットサイズを確保する手法である。拡大

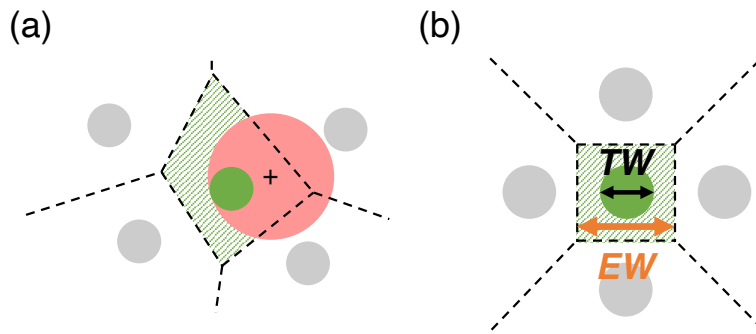


図 3.5: (a) ボロノイ領域によるターゲット拡張と Bubble Cursor. オブジェクトを一つだけ含むように、つまりカーソルに最も近いオブジェクトのみを含むように、エリアカーソルの大きさが設定されている. (b) ターゲットの幅 (Target Width; TW) と Bubble Cursor によって拡大されるターゲットの実質的な幅 (Effective Width; EW).

レンズは、カーソルの動作が弾道運動と修正運動から成り立つという Meyer らの Initial impulse model [128] に則った手法であるキネマティックトリガーを用いており、閾値以下の大きさのターゲット付近にてカーソルの速度プロファイルに2回以上極小値が検出されたときに起動する. Bubble Lens 法を Bubble Cursor 法と比較したところ、動作時間は 10.2 %減少し、エラー率は 37.9 %減少した. 図 3.6 に Bubble Lens 法での拡大レンズが起動する様子を示す.

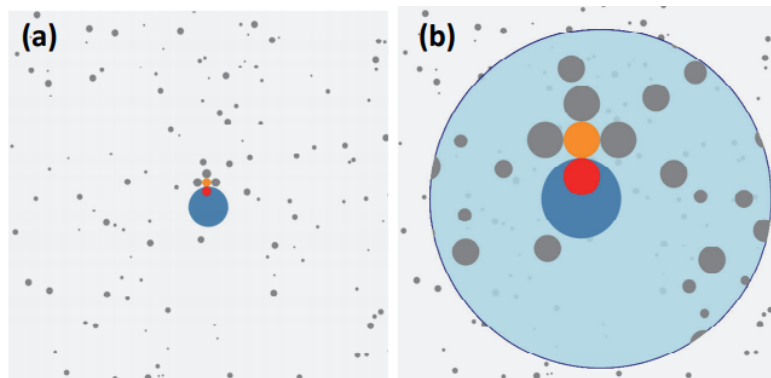


図 3.6: (a) Bubble Lens の拡大レンズが起動する直前の様子, (b) 拡大レンズが起動した画面状態 [137].

### その他のエリアカーソル手法

Bubble Cursor や Bubble Lens 以外にもエリアカーソル手法は存在し、基本的に Bubble Cursor の欠点を解消する手法が提案されている. Bubble Cursor の欠点として挙げられるものは (1) 小さくて密集したオブジェクトの選択が難しい, (2) ターゲットのポインティ

ング動作が最適な動作ではない, (3) バブルの頻繁な膨張と縮小が視覚的に煩わしい, などが挙げられる. (1) の改善手法は Bubble Lens などが該当し, その他に加重重み付けボロノイ領域を用いることでターゲットをより拡大する Additive Voronoi Cursor などが提案されている [23]. (2) は Bubble Cursor の視覚的フィードフォワードではターゲット領域への最短経路が不明であるために生じる. ゆえに, 最短経路の位置にターゲットのゴーストを配置する手法 [98] や, 拡大される領域をそのまま可視化する手法 [51] などが検討されている. (3) は Bubble Cursor の元論文でも触れられており, バブルの最大幅を調整することで過度なバブルの膨張を防ぐことを推奨している. またその他にもカーソル速度に応じてバブルの膨張を防ぐ手法 [20] などが提案されている

## 3.2 提案手法: Bubble Gaze Cursor + Lens

視線インタフェースにおける小さなオブジェクトの選択を容易にする手法として, 視線インタフェースにエリアカーソルを適用する手法を提案する. エリアカーソル手法は複数存在するが, 本研究においてはエリアカーソル手法の中でも一般的な手法である Bubble Cursor と, その改善手法である Bubble Lens を視線インタフェースに導入した手法 Bubble Gaze Cursor と Bubble Gaze Lens を提案する. これらの手法を用いると, ターゲットが常時拡大されることとなる. つまり既存手法とは異なり拡大動作なしでターゲットを拡大することができるため, 小さなターゲットを高速に選択することが可能だと考えられる.

本研究では提案手法が小さなオブジェクトの選択を容易にするか検討する2つの実験を行う. 実験1では Bubble Gaze Cursor とポイントカーソルを用いた一般的な視線選択手法の比較を行い, Bubble Cursor が視線インタフェースにおいてもポインティングの性能向上に寄与するか検討する. 次に, 実験2で Bubble Gaze Cursor と Bubble Gaze Lens を比較し, Bubble Lens が視線インタフェースにおいても密集したオブジェクトの選択を改善するか検討する.

## 3.3 実験1: Bubble Gaze Cursor の性能評価実験

提案手法である Bubble Gaze Cursor がターゲットの選択パフォーマンスを向上させるか検討するための実験を行った.

### 3.3.1 実験参加者と使用機器

実験参加者として18名(男性14名, 女性4名, 平均年齢23.9歳, 標準偏差3.7歳)が実験に参加した. 4名は実験以前にVRにおける視線計測の経験があった. 実験報酬は1,000

円であった。実験時間は約 90 分であった。

本実験は 2.30 Ghz の Intel Core i5-6200U の CPU を搭載したパソコンで行った。ディスプレイはフル HD の 27 インチ (1920 × 1080 ピクセル; 1 ピクセルあたり約 0.31 mm) のディスプレイを使用した。実験に用いたソフトウェアは C# で作成した。アイトラッカ は Tobii EyeX (サンプリングレート: 60 Hz) を用いた。

### 3.3.2 比較手法

提案手法が有用であるか検討するために Bubble Cursor を用いたマウス入力 (Bubble Cursor [49]), ポイントカーソルを用いた視線入力 (一般的な視線入力手法), Bubble Cursor を用いた視線入力 (提案手法) の比較実験を行う。比較手法の図示を図 3.7 に示し, 詳細を以下に示す。

#### Bubble Cursor with Mouse (BM)

BM は Bubble Cursor を用いたマウス選択手法であり, Bubble Cursor におけるオリジナル手法である [49]。本手法はカーソルの形状が円形であり, カーソルに最も近いオブジェクトを含む (オブジェクト上にカーソルが乗る) ようにカーソルの形状が変化する。そしていずれかのオブジェクトがカーソルに含まれた状態で左クリックを行うと, そのオブジェクトが選択された。

#### Point Cursor with Gaze (PG)

PG はポイントカーソルを用いた視線選択手法であり, 一般的な視線入力手法である。カーソルは点状の形状であり, ユーザの視線とディスプレイの交点に常に位置する。ユーザがオブジェクトを注視 (カーソルがオブジェクト上に乗っている) して 600 ms が経過した時, そのオブジェクトが選択された。

#### Bubble Cursor with Gaze (BG)

BG は Bubble Cursor を用いた視線選択手法であり, 提案手法である。カーソルは BM と同じ Bubble Cursor であり, ユーザの視線とディスプレイの交点に常に位置する。ユーザがオブジェクトを注視 (カーソルがオブジェクト上に乗っている) して 600 ms が経過した時, そのオブジェクトが選択された。BG では Bubble Cursor のカーソルの大きさを制御するパラメータとして, バブルの最大幅が設定されている。元手法ではバブルの最大幅は設定されていなかったが, Bubble Cursor を視線インタフェースに実装する際に必要



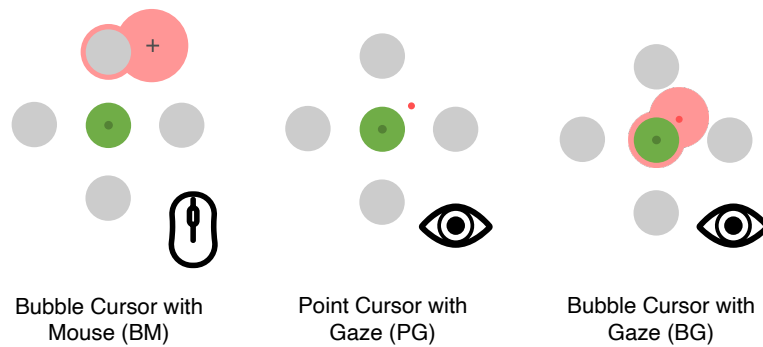


図 3.7: 比較手法の図示.

であるため設定した。バブルの最大幅とは Bubble Cursor がこの値以上に大きくならない指標であり、カーソルと最も近いオブジェクトがこの値の半分よりも遠い時、カーソルはどのオブジェクトも含まなかった。ゆえに「バブルの最大幅」は「オブジェクトが拡大され得る最大幅」と捉えることが出来る。BG にバブルの最大幅を採用した理由は、バブルの最大幅を設定しないと BM 使用時に Midas Touch (意図しない誤選択) が頻発するためである。視線インタフェースに Bubble Cursor をそのまま実装する (バブルの最大幅が無限) と、ユーザはどこを見てもいづれかのオブジェクトを注視していることとなる。仮にディスプレイ上に一つのオブジェクトのみが存在する場合、ディスプレイのどこを見てもオブジェクトを注視することになり、Midas Touch が生じる。ゆえにバブルの最大幅を設定し、オブジェクトが拡大され得るサイズに制限を加えた。

### 3.3.3 実験デザイン

実験は参加者内実験計画を用いて行った。独立変数は 4 つ (図 3.8) 存在し、それぞれ以下のように設定されている。

- 手法 (*Method*; Bubble Cursor with Mouse (BM), Point Cursor with Gaze (PG), Bubble Cursor with Gaze (BG))
- ターゲット間の距離 (*Amplitude*; 350, 550, 800 ピクセル (8.89°, 13.94°, 20.16°))
- ターゲットの幅 (*TW*; 50, 65, 80 ピクセル (1.27°, 1.66°, 2.04°))
- ターゲットの実質的な幅 (*EW*; 100, 120, 140 ピクセル (2.55°, 3.06°, 3.56°))

*Method* には提案手法 BG との比較を目的として、Bubble Cursor を用いない通常の視線選択手法である PG を採用した。また、本来の Bubble Cursor 手法と視線入力に Bubble Cursor を適用した提案手法を比較するために、マウス入力における Bubble Cursor 手法

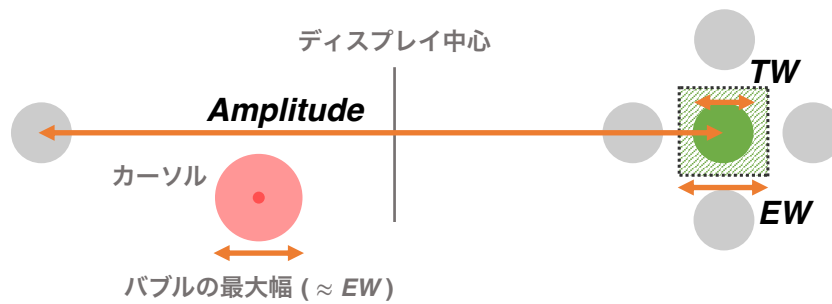


図 3.8: 独立変数  $Amplitude$ ,  $TW$ ,  $EW$  の図示.

である BM を採用した。 *Method* はラテン方格法を考慮した順番で提示された。 実験参加者は *Method* ごとに 9 セッション行った。 各セッションではランダムな順序で表示される  $Amplitude$ ,  $TW$ ,  $EW$  の 27 の組み合わせを 2 回ずつ行う。 つまり実験参加者はランダムに提示される特定の組み合わせに対して連続して 2 回の試行を行う必要があった。 総じて、実験参加者は  $3 \text{ Method} \times 9 \text{ セッション} \times 3 \text{ Amplitude} \times 3 \text{ TW} \times 3 \text{ EW} \times 2 = 1,458$  回のターゲット選択を行った。 また実験参加者は 18 名であるため、合計 26,244 データが収集された。

従属変数は選択時間、エラー率である。 選択時間はターゲットが生成されてからターゲットを選択するまでの時間であり、ターゲットを正確に選択した時のみ計測された。 また、間違っただけのオブジェクトが選択されたか 3 秒間何も選択されなかった場合、エラーとした。 実験参加者は *Method* ごとに、ユーザビリティと精神的負荷を計測することを目的とした System Usability Scale (SUS) [17] と NASA Task Load Index (NASA-TLX) [54] のアンケートに回答した。

### 3.3.4 実験手順

はじめに、実験参加者をディスプレイから 70 cm ほど離れた位置に座らせ、目がディスプレイの中心に位置する高さに椅子を調整させた。 その後、これから行うタスクの内容を紙に印刷された図などを用いて説明した後、Tobii EyeX で提供されているアイトラッキングの 6 点キャリブレーションを行った。 そして実験参加者が実験タスクの挙動を理解するまで、実験タスクからパラメータを変更したタスクを練習として行わせた。

実験タスクは水平方向に対して交互に提示されるターゲットを選択することである。 まず、画面上に緑色の円形のオブジェクトが表示された。 このオブジェクトがターゲットであり、全ての試行において同じ外観であった。 ターゲットは画面中心から水平方向に対して  $Amplitude$  の値の半分だけ離れた位置に配置された。 ターゲットの中心には直径 10 ピクセル (0.25°) の濃緑色の点が凝視対象として描画されている。 この点を凝視することで

注視時の視線の分散が小さくなることが確認されており [230], 実験参加者はこの点を凝視することが求められた. ターゲットは, BM ではマウスのクリックで, PG と BG では 450 ms の間注視することで選択できる. このターゲット選択はカーソルの位置合わせを目的としており, データ収集は行わなかった. ターゲットを選択するとターゲットが消え, ターゲットの位置からディスプレイ中心を通る垂直二等分線に線対称な位置に新しいターゲットとそれを四方に囲む灰色のオブジェクトが生成された. オブジェクトはターゲット同じ形状と大きさであり,  $EW$  を制御する位置に配置された. これらのオブジェクトは BM ではマウスのクリックで, PG と BG では 600 ms の間注視することで選択できる. いずれかのオブジェクトが選択された時これらのオブジェクト群が消え, これらのオブジェクト群と同じものが, ターゲットの位置からディスプレイ中心を通る垂直二等分線に線対称な位置 (水平方向に  $Amplitude$  の値だけ離れた位置) に生成された. これらのオブジェクトは, 以前に生成されたオブジェクト群に対する選択方法と同じ方法で選択できる. そしていずれかのオブジェクトを選択すると 1 試行が完了し, 次の試行が開始された. なお, これらのターゲットの生成パラメータである独立変数  $Amplitude$ ,  $TW$ ,  $EW$  は 1 試行の間同じ値であるため, 実験参加者は特定の独立変数の組み合わせを連続して 2 回体験する. つまり 1 試行ごとに 2 回の選択が行われる. そしてこの 2 回の選択において, ターゲット以外のオブジェクトが選択された場合と 3 秒以内に選択を完了できなかった場合をエラーとした.

タスクの実行中は  $Method$  に応じたカーソルが常に表示された. BM ではクロスカーソルを中心とした赤色の Bubble Cursor が, PG では直径 6 ピクセル ( $0.15^\circ$ ) の濃赤色のポイントカーソルが, BG では PG と同じポイントカーソルを中心とした赤色の Bubble Cursor が表示された. バブルの最大幅は  $EW$  と同じ値であった. 視線データは Exponential Decay Filter [168] と iSR [231] で平滑化された.

### 3.3.5 結果

選択時間とエラー率に対してノンパラメトリックな分散分析手法である整列ランク変換 (ART) [58, 166, 211] を行い, 混合モデル REML を用いてデータを評価した. 多重比較にはウィルコクソンの符号順位検定を用い, Holm 法 [62] による補正を用いた. SUS と NASA-TLX によって得られた値はフリードマン検定とウィルコクソンの符号順位検定を用い, Holm 法による補正を用いた.

## 選択時間

BM, PG, BG の選択時間の中央値はそれぞれ 0.77 秒, 1.33 秒, 1.23 秒であった。手法 *Method*, ターゲット間の距離 *Amplitude*, ターゲットの幅 *TW*, ターゲットの有効幅 *EW* を独立変数とし, 選択時間を従属変数として分析を行った。選択時間に対して *Method* ( $F_{2,25409} = 24480, p < .01$ ), *Amplitude* ( $F_{2,25409} = 1810.5, p < .01$ ), *TW* ( $F_{2,25409} = 116.79, p < .01$ ), *EW* ( $F_{2,25409} = 63.47, p < .01$ ) に有意な差が確認された。さらに 1 次の交互作用として *Method*×*Amplitude* ( $F_{4,25409} = 13.98, p < .01$ ), *Method*×*TW* ( $F_{4,25409} = 132.70, p < .01$ ), *Method*×*EW* ( $F_{4,25409} = 22.17, p < .01$ ) が確認された。図 3.9 に *Method* ごとの選択時間を示す。

## エラー率

BM, PG, BG のエラー率はそれぞれ 1.33 %, 5.22 %, 1.87 %であった。手法 *Method*, ターゲット間の距離 *Amplitude*, ターゲットの幅 *TW*, ターゲットの有効幅 *EW* を独立変数とし, エラー率を従属変数として分析を行った。エラー率に対して *Method* ( $F_{2,26146} = 4299.59, p < .01$ ), *Amplitude* ( $F_{2,26146} = 1400.85, p < .01$ ), *TW* ( $F_{2,26146} = 6414.97, p < .01$ ), *EW* ( $F_{2,26146} = 100.82, p < .01$ ) に有意な差が確認された。さらに 1 次の交互作用として *Method*×*Amplitude* ( $F_{4,26146} = 1290.78, p < .01$ ), *Method*×*TW* ( $F_{4,26146} = 1101.09, p < .01$ ), *Amplitude*×*TW* ( $F_{4,26146} = 1717.99, p < .01$ ), *Method*×*EW* ( $F_{4,26146} = 111.68, p < .01$ ), *Amplitude*×*EW* ( $F_{4,26146} = 428.28, p < .01$ ), *TW*×*EW* ( $F_{4,26146} = 43.31, p < .01$ ) が, 2 次の交互作用として *Method*×*Amplitude*×*TW* ( $F_{8,26146} = 900.99, p < .01$ ), *Method*×*Amplitude*×*EW* ( $F_{8,26146} = 211.55, p < .01$ ), *Method*×*TW*×*EW* ( $F_{8,26146} = 103.93, p < .01$ ), *Amplitude*×*TW*×*EW* ( $F_{8,26146} = 237.89, p < .01$ ) が, 3 次の交互作用として *Method*×*Amplitude*×*TW*×*EW* ( $F_{16,26146} = 235.42, p < .01$ ) 確認された。図 3.10 に *Method* ごとのエラー率を示す。

## System Usability Scale (SUS)

BM, PG, BG の SUS スコアの中央値はそれぞれ 80.00, 58.75, 70.00 であった (高いほど良い)。フリードマン検定より, *Method* ( $\chi^2_{2,N=18} = 12.33, p < .01$ ) に有意差が確認された。また事後検定より, BM と PG ( $Z = -3.36, p < .01$ ), BM と BG ( $Z = -2.29, p < .05$ ), PG と BG ( $Z = -2.69, p < .05$ ) に有意差が確認された (図 3.11a)。

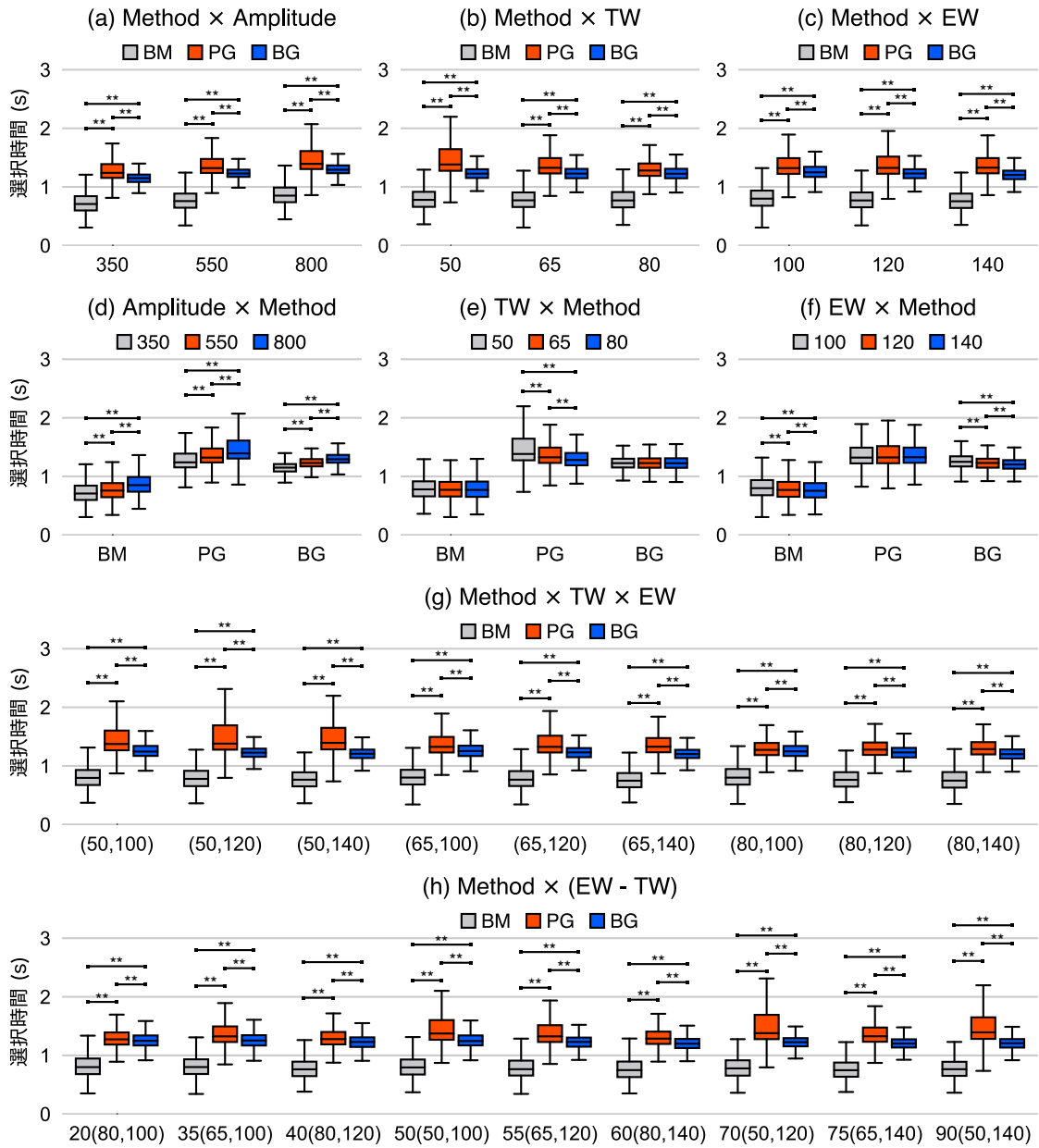


図 3.9: (a)  $Method \times Amplitude$ , (b)  $Method \times TW$ , (c)  $Method \times EW$ , (d)  $Amplitude \times Method$ , (e)  $TW \times Method$ , (f)  $EW \times Method$ , (g)  $Method \times TW \times EW$ , (h)  $Method \times (EW - TW)$  の選択時間. (g) の x 軸ラベルは  $(TW, EW)$  であり, (h) の x 軸ラベルは  $EW - TW$  ( $TW, EW$ ) である. 有意差は  $**p < .01$ .

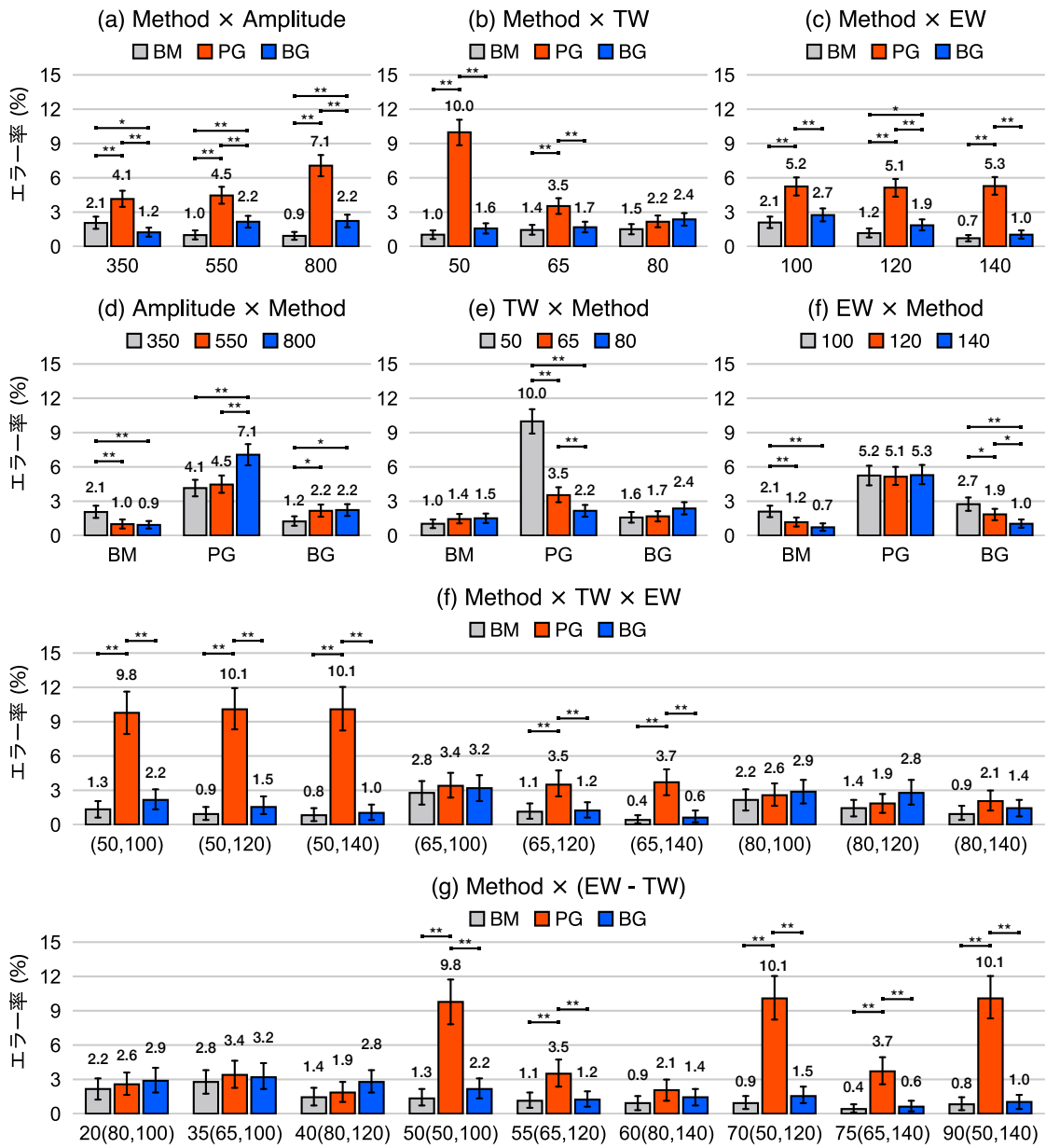


図 3.10: (a)  $Method \times Amplitude$ , (b)  $Method \times TW$ , (c)  $Method \times EW$ , (d)  $Amplitude \times Method$ , (e)  $TW \times Method$ , (f)  $EW \times Method$ , (g)  $Method \times (EW - TW)$  のエラー率. (g) の x 軸ラベルは  $(TW, EW)$  であり, (h) の x 軸ラベルは  $EW - TW$  ( $TW, EW$ ) である. エラーバーは 95%信頼区間. 有意差は  $*p < .05$ ,  $**p < .01$ .

### NASA Task Load Index (NASA-TLX)

BM, PG, BG の総合ワークロードスコアの中央値はそれぞれ 22.33, 64.17, 39.00 であった (低いほど良い). NASA-TLX によって得られた 6 つの項目に対するスコアと総

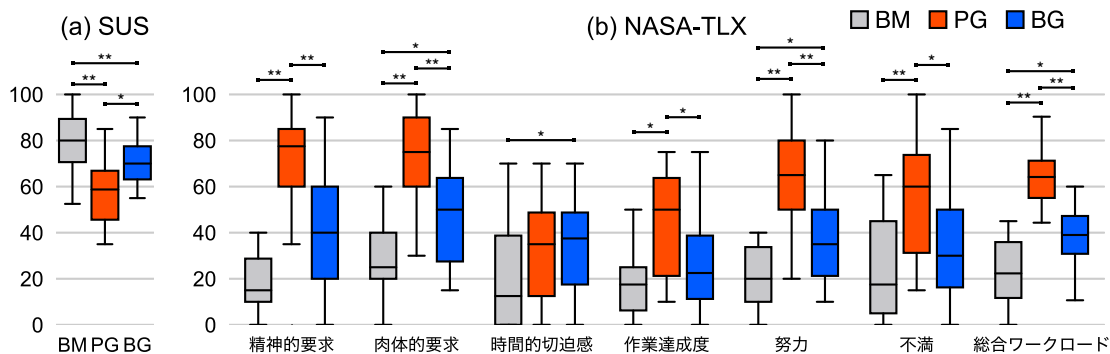


図 3.11: Method ごとの (a) SUS, (b) NASA-TLX の結果. 有意差は\* $p < .05$ , \*\* $p < .01$ .

合ワークロードスコアに対してフリードマン検定を用いて分析した結果, Method に対して精神的要求 ( $\chi^2_{2,N=18} = 24.80, p < .01$ ), 肉体的要求 ( $\chi^2_{2,N=18} = 20.99, p < .01$ ), 時間的切迫感 ( $\chi^2_{2,N=18} = 16.29, p < .05$ ), 作業達成度 ( $\chi^2_{2,N=18} = 14.00, p < .01$ ), 努力 ( $\chi^2_{2,N=18} = 19.97, p < .01$ ), 不満 ( $\chi^2_{2,N=18} = 15.72, p < .01$ ), 総合ワークロード ( $\chi^2_{2,N=18} = 23.11, p < .01$ ) に有意差が確認された. また事後検定より総合ワークロードに対して BM と PG ( $Z = -3.53, p < .01$ ), BM と BG ( $Z = -2.44, p < .05$ ), PG と BG ( $Z = -3.44, p < .01$ ) に有意差が確認された (図 3.11b).

### 3.3.6 実験結果のまとめ

Bubble Gaze Cursor (BG) は選択時間, ユーザビリティ, 精神的負荷の観点でベースライン手法である PG よりも有意に優れていた. そのうえ NASA-TLX の項目である精神的要求, 肉体的要求, 作業達成度, 努力, 不満において, BG は PG よりも有意に優れていた. したがって Bubble Cursor はマウスだけではなく視線インタフェースのポインティングも容易にするとと言える. 一方で, BG は選択時間, ユーザビリティ, 精神的負荷の観点で元手法である BM よりも劣っていた. そのうえ NASA-TLX の項目である精神的要求, 肉体的要求, 時間的切迫感, 努力において BG は BM よりも有意に劣っていた. したがって Bubble Cursor を視線インタフェースに活用したとしてもマウスインタフェースには性能は及ばないと言える.

## 3.4 実験1に関する議論

### 3.4.1 視線インタフェースにおける Bubble Cursor の効果

#### 選択時間

実験結果より、全ての条件で BG は PG よりも有意に高速な選択時間を記録した。特に最もターゲットが小さい ( $TW=50$ ) とき、BM は PG に対して最も大きく選択時間の中央値を改善した。さらに図 3.9, BM は選択時間の分布の幅が BM や PG よりも狭かった。これは BG が他手法よりも安定した速度で入力を完了している証左だと言える。また、BG は  $EW$  が大きくなるほど選択時間が有意に高速になった。これは  $EW$  が大きくなるほど視線カーソルがターゲットに触れるまでの時間が少なくなるためだと考えられる。したがって Bubble Cursor を使用することで視線インタフェースのポインティング時間を改善することが出来ると言える。

#### エラー率

実験結果より、 $TW$  が小さい ( $TW=50, 65$ ) 時 BG は PG よりもエラー率が有意に低かった。特に  $TW=50$  の時の PG のエラー率が 10.0 % であるのに対し BG のエラー率は 1.6 % であり、顕著なエラー率の改善が確認された。したがって Bubble Cursor は十分なターゲット拡大が可能な手法だと言える。

一方で、ターゲットサイズが大きいほどエラー率は低いはずだが、特定の  $TW$  と  $EW$  の組み合わせ ( $TW-EW: 80-100, 80-120$ ) では PG よりも BG の方が僅かながらエラー率が高い場合があった。また BG では  $EW$  が大きいほどエラー率が低いはずだが、 $EW$  が同じ値でも  $TW$  が大きい時エラー率が高くなった。これは Bubble Cursor によってターゲット以外の偽ターゲットのサイズも拡大されたことが原因だと考えられる。Bubble Cursor を用いると領域全てがどれかのターゲットのボロノイ領域になるため、全ての試行が密集した状況におけるターゲット選択となる。したがってターゲット間のスペースが少ないほど偽ターゲットを選択する割合が増え、 $TW$  が増えるほど偽ターゲットを選択するようなエラーが増加したと考えられる。実際、図 3.10g より、スペース ( $EW-TW$ ) が大きくなるほど BM と BW のエラーが減少していることが読み取れる。ただし、このエラー率増大の影響はターゲットサイズ拡大によるエラー率減少の影響よりも小さいため、提案手法の価値を損なうものではないと考えられる。



### 3.4.2 視線インタフェースとマウスインタフェースの比較

本実験では Bubble Cursor を使用した視線インタフェース (BG) とマウスインタフェース (BM) を用いてポインティングタスクを実施した。その結果、BM は BG よりも選択時間やユーザビリティ、精神的負荷の指標においてそれぞれ優れていた。特に選択時間に関しては BM は BG よりも非常に優れており、BM は BG よりも約 500 ms 程度高速に動作した。この選択時間の差は BG のターゲット選択に必要な注視時間 600 ms が影響を及ぼしていると考えられる。滞留時間選択手法はマウスカーソルよりも選択時間が遅い結果は既存研究で示されており [129]、本研究によって Bubble Cursor を用いたとしても同様の結果が得られることが明らかになった。

一方で、Bubble Cursor が視線インタフェースに及ぼす影響はマウスインタフェースよりも大きいと考えられる。視線インタフェースにおける適切なターゲットサイズはマウスに比べると非常に大きく [38]、小さなターゲットサイズに相当するオブジェクトはマウスよりも多い。ゆえに Bubble Cursor によるターゲット拡大の恩恵を得る機会は視線インタフェースの方が多いと考えられる。また、選択時間の観点からも視線インタフェースの方が Bubble Cursor の影響が大きい可能性がある。滞留時間入力においてはターゲットから視線データが外れるたびに滞留時間のカウントがリセットされるため、ターゲットサイズが小さくなるほど選択時間は大きくなる。事実、PG のエラーの多くは 3 秒以内にターゲットを選択できなかったためのエラーが大半を占めている。ゆえに、Bubble Cursor を視線インタフェースに導入することで、より高速な選択と安定した選択時間を達成することが可能となっている。そしてこれはマウスには無い特色だと言える。

## 3.5 実験 2: Bubble Gaze Lens の性能評価実験

実験 1 ではエリアカーソル法の一つである Bubble Cursor を視線インタフェースに導入した Bubble Gaze Cursor の検討を行った。Bubble Gaze Cursor は小さなターゲットを暗黙的に拡大することで高精度で選択することが可能であるが、密集した環境における小さなターゲットは十分に拡大することが出来ず、選択を容易にすることが出来ない。ゆえに Bubble Cursor の改善手法である Bubble Lens を視線インタフェースに用いる Bubble Gaze Lens を提案する。本手法ではターゲット付近にユーザの視線が近づいたときに、ターゲットを内包する拡大レンズを自動的に起動する。この拡大レンズによって Bubble Gaze Cursor で選択が困難であった密集した小さなターゲットの選択を容易にする。また拡大レンズは自動的に起動するため、拡大操作を実行する時間を削減することが出来る。ゆえに Bubble Gaze Lens が有用であるか検討するために、Bubble Gaze Cursor と Bubble Gaze Lens との比較実験を行い、提案手法のパフォーマンスを測定する。

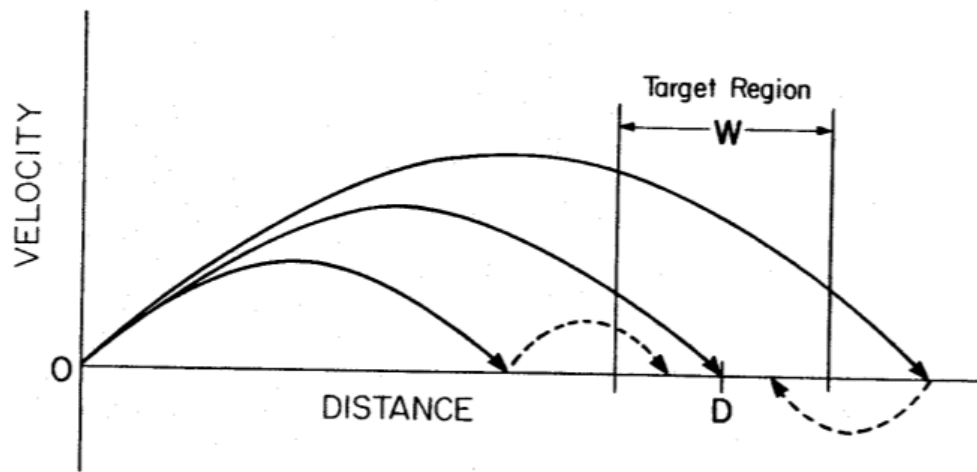


図 3.12: Meyer らのポインティング動作のモデル [128].

### 3.5.1 Bubble Gaze Lens の実装

Bubble Lens の拡大レンズの起動アルゴリズムは、マウスカーソルの移動動作が単一の弾道運動と複数の修正運動によって成り立つという Meyer らのモデル [128] (図 3.12) に則っている。この弾道運動とはユーザが大まかにターゲットに向かってマウスカーソルを移動させる運動を指し、修正運動とは弾道運動後にターゲットに対して修正的にマウスカーソルを近づけていく動作を指す。マウスポインティングでは弾道運動のみで概ねターゲットの付近に着陸する。ゆえに、弾道運動後の修正運動を検知し、そこで拡大レンズを起動することで、拡大操作なしで拡大レンズを適切なタイミングで起動することが出来る。

マウスカーソルの移動動作と同様に、視線の高速な移動動作であるサッケードも単一の弾道運動と修正運動を有している場合がある。サッケードは基本的に単一の弾道運動のみで構成されているが、ターゲット指向のサッケードはターゲットの離心率の 5-10% ほど離れた位置に着地することがあり [57, 156], それを追加の短いサッケード (二次サッケード) [13, 208] が修正することがある。この二次サッケードを修正運動と考えることで、Bubble Lens を視線操作インタフェースに導入することが可能であると考えた。

Bubble Gaze Lens における主要な選択手順は以下である。

1. ターゲットに向かって視線を移動させる。
2. 拡大レンズ起動アルゴリズムに則り、拡大レンズの起動が検討される。拡大レンズの起動の有無に応じて以下の操作を行う。
  - (a) 拡大レンズが起動した場合、拡大レンズ内の拡大されたターゲットを注視する

ことで選択する。なお、拡大されていない領域からターゲットを選択することはできない。

- (b) 拡大レンズが起動しなかった場合、そのままターゲットを注視することで選択する。

### Bubble Gaze Lens の拡大レンズ起動アルゴリズム

Bubble Gaze Lens のアルゴリズムは Bubble Lens のキネマティックトリガーを参考にしている。キネマティックトリガーはマウスカーソルの速度プロファイルに極大値が2ヶ所検出されたときに拡大レンズを起動するアルゴリズムである。具体的なキネマティックトリガーの仕組みは以下である。まず、マウスカーソルの速度データを 100 Hz でリサンプリングした後にガウシアンカーネルフィルタで平滑化する。そして極大値が2ヶ所検出され、かつマウスカーソルの直近ターゲットが既定の有効サイズよりも小さい場合に拡大レンズを起動する。

提案手法ではキネマティックトリガーを視線操作用に拡張した。まず、Bubble Lens においては速度データをリサンプリングした後に平滑化を行ったが、Bubble Gaze Lens においては使用するアイトラッカーのサンプリングレートが低い (90 Hz) ことからリサンプリング・平滑化を行わなかった。さらに、視線はマウスと異なり常に動き続けているため、判定をリセットするためにウインドウサイズを定める必要がある。このウインドウサイズは潜時 (Saccade Latency; メインサッケードの前に生じる 150 ms–200 ms の視線の固定) と一連のサッケードを十分に含む大きさである必要があり、それを考慮した値であるウインドウサイズ 50 データ (約 560 ms) に定めた。Bubble Gaze Lens のアルゴリズムを以下に示す (図 3.13)。

1. ウインドウの初めに視線が 150 ms の間  $8.8^\circ/\text{s}$  (固視閾値) 以内に固定されている (潜時の推定)。
2.  $100^\circ/\text{s}$  以上の極大値 (メインサッケード) と  $30^\circ/\text{s}$  以上の極大値 (2次サッケード) を検出。
3. 2ヶ所の極大値が 50 ms–250 ms 離れている。
4. 1, 2, 3 が成り立ち、ウインドウの最後の 40 ms の視線が  $8.8^\circ/\text{s}$  (固視閾値) 以内に固定されているときに拡大レンズを起動する。

Bubble Gaze Lens のアルゴリズムの説明、およびパラメタの採用理由に関して順に述べる。まず、1. における固視検出は、メインサッケードを検出するために潜時を推定して

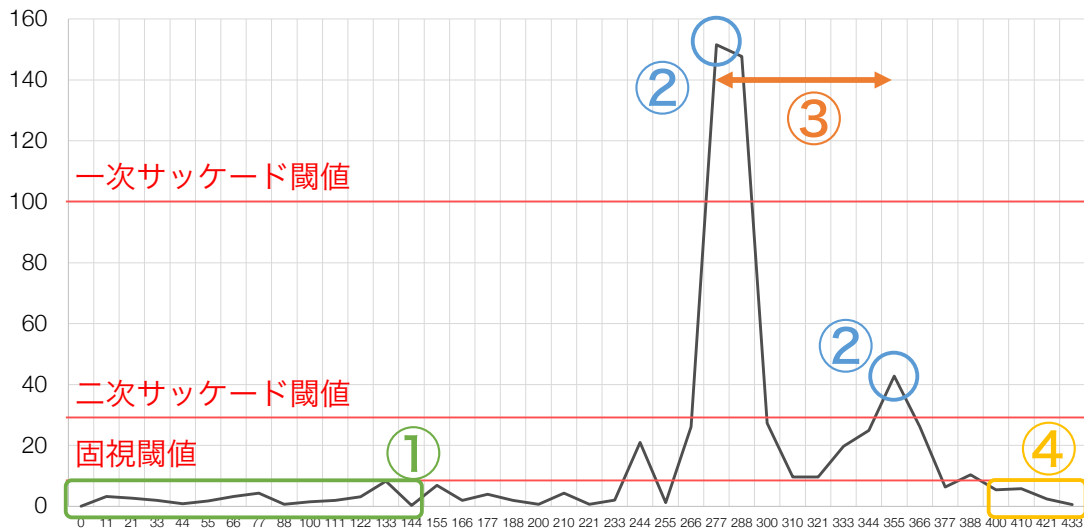


図 3.13: 視線の速度プロファイルに対する Bubble Gaze Lens のアルゴリズムの概略.

いる。その際に固視閾値  $8.8^{\circ}/s$  を用いた。このパラメタは他の研究で用いられたパラメタ [170, 143] よりも小さく、固視の検出に十分だといえる。次に 2. はメインサッケードと二次サッケードを検出している。メインサッケードの検出に用いた速度閾値  $100^{\circ}/s$  は、60 Hz のアイトラッカーを用いた研究 [36, 65] におけるサッケード閾値  $^{\circ}/s$  よりも 75 大きいので、偽陽性率の低い、適切な値だと考えられる。二次サッケード閾値  $30^{\circ}/s$  に関しては、メインサッケードよりも二次サッケードの方がピーク速度が小さいことから、検討の結果  $30^{\circ}/s$  に設定した。次に 3. は二次サッケードと連続したサッケード [12] を分けて検出することを目的としている。サッケード間隔が短すぎる場合、複数のサッケードが連続して行われていると考えられ、サッケード間隔が長すぎる場合、サッケードが終了した後にまたサッケードが開始していると考えられる。そのため、二次サッケードの潜時が 100 ms–150 ms であること [6] や二次サッケードの一種である Dynamic Overshoot [5] を考慮し、サッケード間隔が 50 ms–250 ms 離れていることを二次サッケード検出の条件に組み込んだ。最後に 4. はサッケードが終了したことを検出している。

#### 拡大レンズ起動後の選択動作

拡大レンズ起動アルゴリズムによって拡大レンズの起動が決定されると、アルゴリズムで使用した視線データの最も直近の視線位置を中心とした拡大レンズが起動される。拡大レンズの起動時、滞留時間がリセットされる。ユーザは拡大レンズ内のターゲットを注視することでそのターゲットを選択することができた。なお、拡大レンズ内のオブジェクトのみ選択が可能であり、拡大レンズ外に視線を向けても視線カーソルは表示されなかった

(滞留時間は計測されなかった).

### 3.5.2 実験参加者と使用機器

24名の実験参加者を募ったが、2名はキャリブレーションのエラーで、2名は不規則な視線運動が原因で除外した。ゆえに実験参加者として20名(男性16名, 女性4名, 平均年齢23.0歳, 標準偏差2.1歳)が実験に参加した。1名はアイトラッキングを用いた経験があった。また, 報酬として実験参加者にそれぞれ1,000円分のAmazonギフトカードを渡した。実験時間は約60分であった。

本実験は3.40GHzのIntel(R) Core(TM) i7-6700UのCPUを搭載したパソコンで行った。ディスプレイはフルHDの23インチ(1920 × 1080ピクセル; 1ピクセルあたり約0.27 mm)のディスプレイを使用した。アイトラッカーはTobii Eye Tracker 4C(サンプリングレート: 90 Hz)を使用し, 実験に用いたソフトウェアはC#で作成した。

### 3.5.3 実験デザイン

実験は参加者内実験計画を用いて行った。独立変数は4つ存在し, それぞれ以下のように設定されている。

- 手法 (*Method*; Bubble Gaze Cursor (BC), Bubble Gaze Lens (BL))
- 視線からターゲット中心までの距離 (図 3.14a) (*Distance*; 350, 550, 800ピクセル (7.72°, 12.11°, 17.54°))
- ターゲットの幅 (*TW*; 20, 32, 52ピクセル (0.44°, 0.71°, 1.15°))
- スペース (図 3.14b) (*Spacing*; スペースなし: 0ピクセル, ターゲット半径の半分:  $TW/4$ ピクセル, ターゲット半径:  $TW/2$ ピクセル))

*Method* はラテン方格法を考慮した順番で提示された。実験参加者は *Method* ごとに15セッション行った。各セッションはランダムな順序で表示される *Distance*, *TW*, *Spacing* の27の組み合わせで構成されている。ゆえに実験参加者は2 *Method* × 15セッション × 3 *Distance* × 3 *TW* × 3 *Spacing* = 810回のターゲット選択を行った。また実験参加者は20名であるため, 合計16,200データが収集された。

*Method* に対する従属変数は選択時間とエラー率を求めた。エラーは, 間違っただターゲット選択したとき, 起動した拡大レンズ内にターゲットが内包されていないとき, そして5秒以上ターゲットが選択されなかったときに計上された。またBLに対してのみ, キネマ

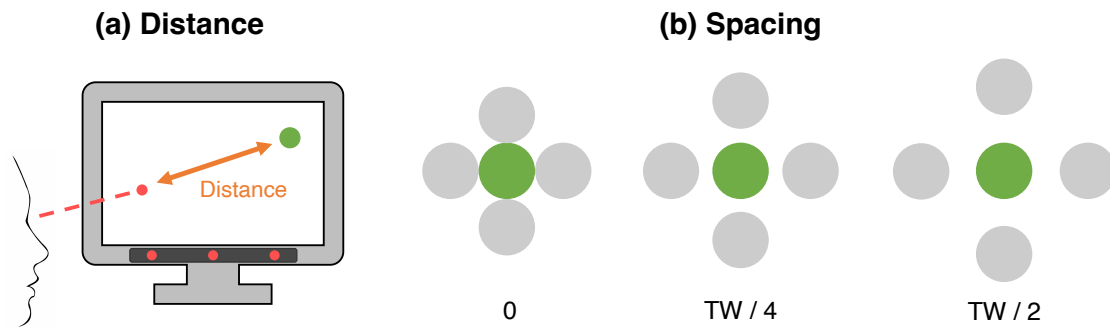


図 3.14: (a) *Distance*, (b) *Spacing*, の図示.

ティックトリガーが失敗した割合を従属変数として扱った。そして、それぞれの *Method* 条件完了後にメンタルワークロードとユーザビリティを計測するために System Usability Scale (SUS) [17] と NASA Task Load Index (NASA-TLX) [54] によるアンケートを実施した。

### 3.5.4 実験手順

はじめに、実験参加者をディスプレイから 70 cm ほど離れた位置に座らせ、目がディスプレイの中心に位置する高さに椅子を調整させた。その後、これから行うタスクの内容を紙に印刷された図などを用いて説明した後、Tobii eye tracker 4C で提供されているアイトラッキングの 3 点キャリブレーションを行った。そして実験参加者が実験タスクの挙動を理解するまで、実験タスクからパラメータを変更したタスクを練習として行わせた。

実験タスクはオブジェクト群の中からターゲットを選択することである。ターゲットは緑色の円形オブジェクトであり、中心には 10 ピクセル ( $0.22^\circ$ ) の濃緑色の点が描画されている。この点を凝視することで注視時の視線の分散が小さくなることが確認されており [230], 実験参加者はこの点を凝視することが求められた。ターゲットの周りには、ターゲットの有効幅を制御することを目的とした円形オブジェクトが四方に表示された。このオブジェクトは灰色であり、ターゲットと同じ大きさで、独立変数 *Spacing* の値と一致する位置に配置された。さらに 16–40 ピクセル ( $0.35^\circ$ – $0.88^\circ$ ) の大きさの灰色の円形オブジェクトが、ターゲットの有効幅を変更しないランダムな位置にそれぞれ重ならないように 350 個生成された。実験参加者はオブジェクトを 600 ms の間注視することで、そのオブジェクトを選択することができた。BC ではオブジェクトを直接注視することで選択が可能であった。BL では拡大レンズが起動していない場合はオブジェクトを直接、拡大レンズが起動している場合は拡大レンズ内の拡大されたオブジェクトを注視することで選択ができた。この拡大レンズが起動しているとき、拡大レンズ外のオブジェクトを選択する

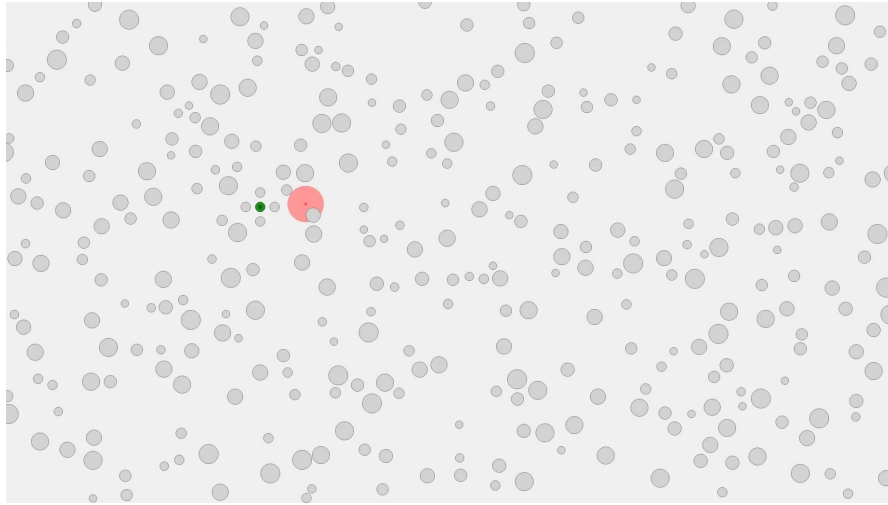


図 3.15: 実験時の様子. ターゲットは緑色の円であり, 偽ターゲットが灰色の円である.

ことはできなかった. 実験参加者がオブジェクトを選択するか, 5 秒間何も選択しなかったとき, オブジェクト群は再生成された. 本実験ではターゲットではないオブジェクトを選択した試行や, 5 秒間何も選択されなかった試行, 拡大レンズ起動時に拡大レンズ内にターゲットが含まれていない試行をエラーと定義した. また拡大レンズ内にターゲットが含まれていなかった場合, 1000 ms のポーズが生じた. オブジェクト群が再生成される際, 以前のターゲットが生成された位置から  $Distance$  の値だけ離れ, かつ画面横端から 200 ピクセルと縦端 150 ピクセルに含まれないランダムな位置にターゲットが生成された. これは拡大レンズに表示されるターゲットが画面外に表示されないように設定した. 上記が 1 試行に該当し, 実験参加者はこの試行を繰り返す必要があった. 図 3.15 に実際に実験に使用したアプリケーションの画面を示す.

タスクの実行中は濃赤色のポイントカーソルと, ポイントカーソルを中心とした赤色の Bubble Cursor が表示された視線データは Saccade detection [38] with a triangular kernel filter [82] によって平滑化された. なお, 拡大レンズ起動アルゴリズムの計算のために用いる視線データは平滑化されなかった. また BC と BL におけるバブルの最大幅は 100 ピクセル ( $2.21^\circ$ ) であった. この最大幅は拡大レンズ内でも維持される. 拡大レンズによる拡大率は 4 であり, 拡大レンズの幅は 560 ピクセル ( $12.33^\circ$ ) であった. したがって, 拡大レンズ起動位置から 70 ピクセル ( $1.55^\circ$ ) を半径とする円の領域が拡大された.

### 3.5.5 評価指標

本実験では Bubble Gaze Cursor (BC) と Bubble Gaze Lens (BL) の両方に対する評価指標と, BL に対してのみの評価指標が存在する. 後者は拡大レンズの起動に関する評

価指標である。

BC と BL の両方に対する従属変数は「選択時間」「エラー率」である。「選択時間」は試行の開始からターゲットの選択に成功するまでの時間である。つまり、エラーが生じた場合、その試行において選択時間は計測されなかった。「エラー率」は全試行に対する (a) ターゲットではないオブジェクトを選択した試行, (b) 5 秒以内に選択が行われなかった試行, (c) 起動した拡大レンズ内にターゲットが存在しなかった試行, の割合である。なお (c) は BL でのみ考慮されたため, BC では (a) と (b) の全試行に対する割合, BL では (a) と (b) と (c) の全試行に対する割合をエラー率と定義した。

BL のみに対する従属変数は「拡大レンズ起動率」「拡大レンズにおけるターゲット非包括率」である。「拡大レンズ起動率」は全試行に対する拡大レンズが起動した割合である。つまりキネマティックトリガーが選択するまでに起動した割合を指す。「拡大レンズにおけるターゲット非包括率」は拡大レンズが起動した試行に対して, 拡大レンズ内にターゲットが描画されなかった割合である。拡大レンズ起動時に, 拡大レンズの中心とターゲット中心間の距離が  $70 + TW/2$  ピクセル以上離れていた時, 拡大レンズ内にターゲットが包括されなかった。

### 3.5.6 結果

選択時間とエラー率に対してノンパラメトリックな分散分析手法である整列ランク変換 (ART) [58, 166, 211] を行い, 混合モデル REML を用いてデータを評価した。多重比較にはウィルコクソンの符号順位検定を用い, Holm 法 [62] による補正を用いた。SUS と NASA-TLX によって得られた値はウィルコクソンの符号順位検定を用いた。また, データの可視化のためにターゲットの有効幅 ( $EW$ ) [49] を用いた。ターゲットの有効幅は Bubble Cursor によって実質的に拡大されるターゲット幅を指し,  $TW$  と  $Spacing$  から計算できる。ゆえに 9 つの  $EW$  (表 3.1: 20, 25, 30, 32, 40, 48, 52, 65, 78 ピクセル (それぞれ  $0.44^\circ$ ,  $0.55^\circ$ ,  $0.66^\circ$ ,  $0.71^\circ$ ,  $0.88^\circ$ ,  $1.06^\circ$ ,  $1.15^\circ$ ,  $1.44^\circ$ ,  $1.72^\circ$ )) が得られる。

表 3.1:  $TW$  と  $Spacing$  の組み合わせ。

	Effective Width ( $EW$ )								
	20	25	30	32	40	48	52	65	78
$TW$	20	20	20	32	32	32	52	52	52
$Spacing$	0	5	10	0	8	16	0	13	26



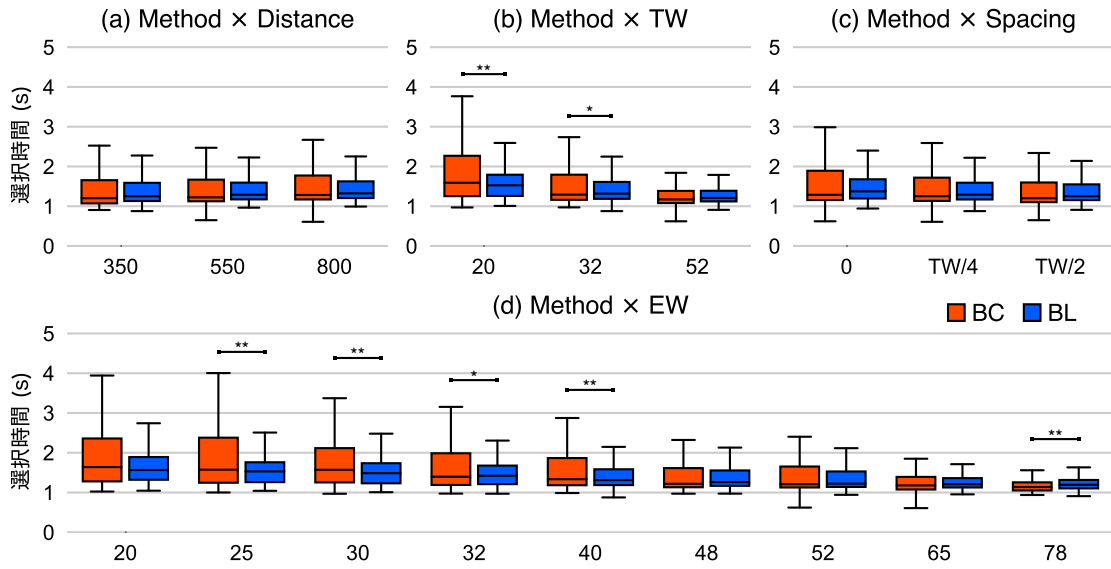


図 3.16: (a)  $Method \times Distance$ , (b)  $Method \times TW$ , (c)  $Method \times Spacing$ , (d)  $Method \times EW$  の選択時間. 有意差は $*p < .05$ ,  $**p < .01$ .

### 選択時間

手法  $Method$ , 視線からターゲットまでの距離  $Distance$ , ターゲットの幅  $TW$ , スペース  $Spacing$  を独立変数とし, 選択時間を従属変数として分析を行った. 選択時間に対して  $Method$  ( $F_{1,10956} = 13.18, p < .01$ ),  $Distance$  ( $F_{2,10972} = 44.75, p < .01$ ),  $TW$  ( $F_{2,10954} = 609.57, p < .01$ ),  $Spacing$  ( $F_{2,10953} = 59.09, p < .01$ ) に有意な差が確認された. さらに 1 次の交互作用として  $Method \times TW$  ( $F_{2,10954} = 36.72, p < .01$ ),  $TW \times Spacing$  ( $F_{4,10953} = 5.34, p < .01$ ) が確認された. 図 3.16 に  $Method$  ごとの選択時間を示す.

### エラー率

手法  $Method$ , 視線からターゲットまでの距離  $Distance$ , ターゲットの幅  $TW$ , スペース  $Spacing$  を独立変数とし, エラー率を従属変数として分析を行った. エラー率に対して  $Method$  ( $F_{1,16127} = 1980.39, p < .01$ ),  $TW$  ( $F_{2,16127} = 2039.44, p < .01$ ),  $Spacing$  ( $F_{2,16127} = 304.56, p < .01$ ) に有意な差が確認された. さらに 1 次の交互作用として  $Method \times Distance$  ( $F_{2,16127} = 45.92, p < .01$ ),  $Method \times TW$  ( $F_{2,16127} = 540.60, p < .01$ ),  $Distance \times TW$  ( $F_{4,16127} = 6.68, p < .01$ ),  $Method \times Spacing$  ( $F_{2,16127} = 181.49, p < .01$ ),  $TW \times Spacing$  ( $F_{4,16127} = 84.69, p < .01$ ) が, 2 次の交互作用として  $Method \times Distance \times TW$  ( $F_{4,16127} = 19.20, p < .01$ ),  $Method \times Distance \times Spacing$  ( $F_{4,16127} = 5.08, p < .01$ ),  $Method \times TW \times Spacing$  ( $F_{4,16127} = 29.17, p < .01$ ),  $Distance \times TW \times Spacing$  ( $F_{8,16127} =$

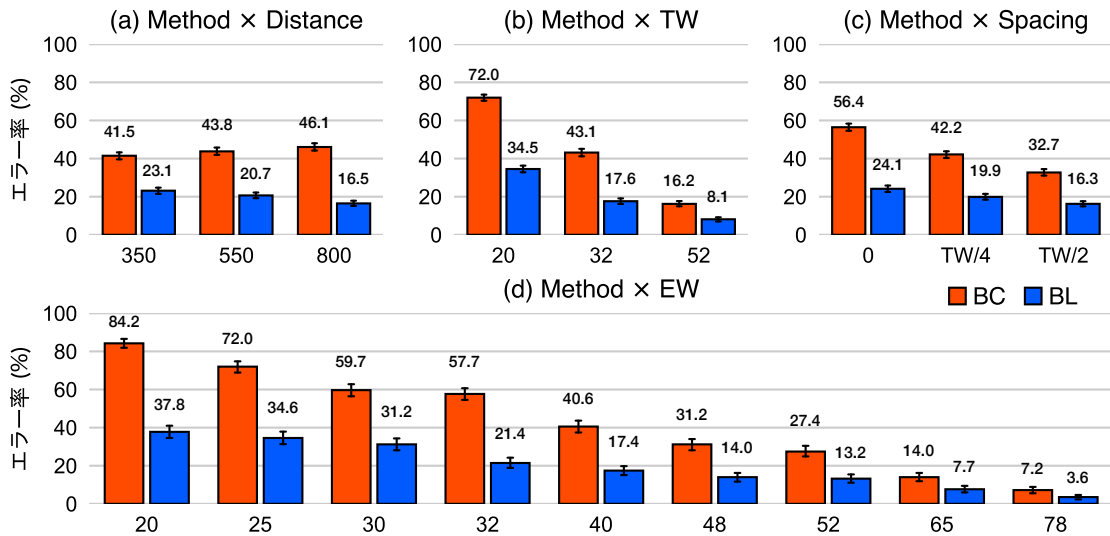


図 3.17: (a) *Method* × *Distance*, (b) *Method* × *TW*, (c) *Method* × *Spacing*, (d) *Method* × *EW* のエラー率. エラーバーは 95%信頼区間. 全ての組み合わせに対して  $p < .01$  の有意差が確認された.

2.38,  $p < .05$ ) が確認された. 図 3.17 に *Method* ごとのエラー率を示す.

### Bubble Gaze Lens: 拡大レンズ起動率

視線からターゲットまでの距離 *Distance*, ターゲットの幅 *TW*, スペース *Spacing* を独立変数とし, 拡大レンズ起動率を従属変数として分析を行った. 拡大レンズ起動率に対して *Distance* ( $F_{2,8054} = 41.57, p < .01$ ), *TW* ( $F_{2,8054} = 8.34, p < .01$ ), *Spacing* ( $F_{2,8054} = 30.28, p < .01$ ) に有意な差が確認された. さらに 1 次の交互作用として *Distance* × *TW* ( $F_{4,8054} = 5.11, p < .01$ ), *Distance* × *Spacing* ( $F_{4,8054} = 5.74, p < .01$ ), *TW* × *Spacing* ( $F_{4,8054} = 5.39, p < .01$ ) が, 2 次の交互作用として *Distance* × *TW* × *Spacing* ( $F_{8,8054} = 5.25, p < .01$ ) が確認された. 図 3.18 に Bubble Gaze Lens の拡大レンズ起動率を示す.

### Bubble Gaze Lens: 拡大レンズにおけるターゲット非包括率

全体の試行に対するターゲット非包括によるエラー率は 0.95 % であり, 拡大レンズが起動した試行に対するターゲット非包括によるエラー率は 1.32 % であった. 以下の統計分析は後者のターゲット非包括率に対する分析である. 視線からターゲットまでの距離 *Distance*, ターゲットの幅 *TW*, スペース *Spacing* を独立変数とし, ターゲット非包括率を従属変数として分析を行った. ターゲット非包括率に対して *Distance* ( $F_{2,5793.5} = 116.40, p < .01$ ), *TW* ( $F_{2,5783.9} = 42.64, p < .01$ ), *Spacing* ( $F_{2,5784.4} = 293.49, p < .01$ ) に有意な差が

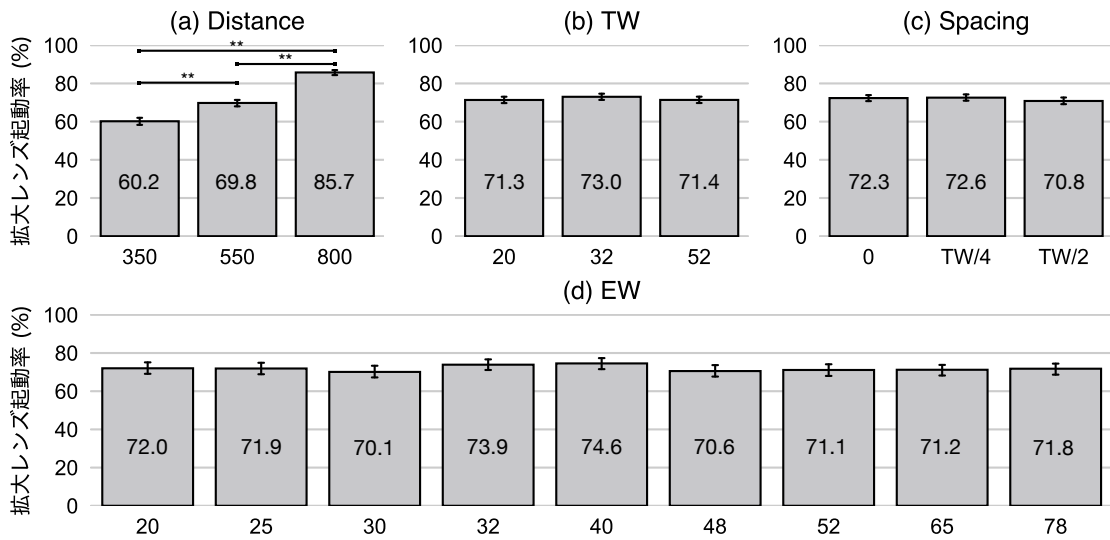


図 3.18: (a)  $Method \times Distance$ , (b)  $Method \times TW$ , (c)  $Method \times Spacing$ , (d)  $Method \times EW$  の拡大レンズ起動率。エラーバーは 95%信頼区間。有意差は\*\* $p < .01$ 。

確認された。さらに 1 次の交互作用として  $Distance \times TW$  ( $F_{4,5786.2} = 88.58, p < .01$ ),  $Distance \times Spacing$  ( $F_{4,5785.1} = 110.42, p < .01$ ),  $TW \times Spacing$  ( $F_{4,5784.8} = 53.06, p < .01$ ) が、2 次の交互作用として  $Distance \times TW \times Spacing$  ( $F_{8,5784.8} = 165.63, p < .01$ ) が確認された。図 3.19 に Bubble Gaze Lens のターゲット非包括率を示す。

### Bubble Gaze Lens: 拡大レンズ起動の有無ごとのエラー率

BL (Bubble Gaze Lens) におけるエラーがいつ発生したか調べたところ、全体のエラーの 72.79%が拡大レンズが起動しなかった場合に発生していた。ゆえに BL のエラーの多くはキネマティックトリガーが正しく動作しなかったことに起因しているといえる。また、レンズ内にターゲットを含まない位置で拡大レンズが開いた割合は全試行の 0.95%であり、27.21%のエラー試行のうち 3.49%であった。拡大レンズの起動の有無ごとのエラー率を図 3.20 と 3.21 に示す。

### System Usability Scale (SUS)

得られたスコアに対してウィルコクソンの符号順位検定を用いて分析した。BC (Bubble Gaze Cursor) と BL (Bubble Gaze Lens) の平均 SUS スコアはそれぞれ 60.88 と 70.13 であり、有意な差が確認された ( $Z = 3.11, p < .01$ )。SUS の結果を図 3.22a に示す。

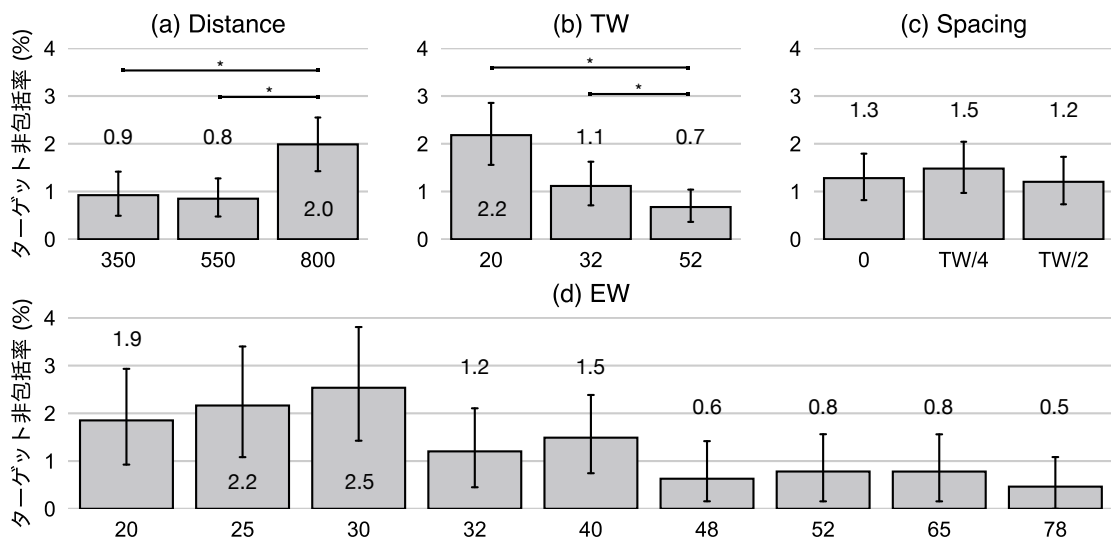


図 3.19: (a)  $Method \times Distance$ , (b)  $Method \times TW$ , (c)  $Method \times Spacing$ , (d)  $Method \times EW$  のターゲット非包括率。エラーバーは 95%信頼区間。有意差は  $*p < .05$ 。

### NASA Task Load Index (NASA-TLX)

NASA-TLX によって得られた 6 つの項目に対するスコアと総合ワークロードスコアに対して、ウィルコクソンの符号順位検定を用いて分析した。BC (Bubble Gaze Cursor) と BL (Bubble Gaze Lens) の総合ワークロードスコアはそれぞれ 46.10 と 34.57 であった。また  $Method$  に対して精神的要求 ( $Z = 3.04, p < .01$ ), 肉体的要求 ( $Z = 2.47, p < .05$ ), 総合ワークロードスコア ( $Z = 2.92, p < .01$ ) に有意な差が確認された。NASA-TLX の結果を図 3.22b に示す。

#### 3.5.7 実験結果のまとめ

BL (Bubble Gaze Lens) は BC (Bubble Gaze Cursor) よりも実質的な幅  $EW$  が小さなオブジェクトの選択が有意に高速であった。またエラー率は全ての  $EW$  において BL が有意に優れており、最も小さな  $EW$  の値 ( $EW=20$ ) の時、BL は BC のエラー率を 46.4% 改善した。その他にもユーザビリティや精神的負荷の指標が BC よりも BL の方が有意に優れていることから、視線インタフェースにおいても Bubble Lens は Bubble Cursor よりも小さくて密集したオブジェクトの選択に適した手法だと言える。

一方で、BL の拡大レンズは常に起動したわけではなく、起動率は試行全体の 71.9% であった。BL のエラーの多くは拡大レンズが起動しなかった場合の試行で起こっており、BL のエラー率 20.0% のうち 13.7% が拡大レンズの非起動が原因のエラー率である。したがって拡大レンズの起動成功率は 7 割程度であり、アルゴリズムの更なる改善が必要だと

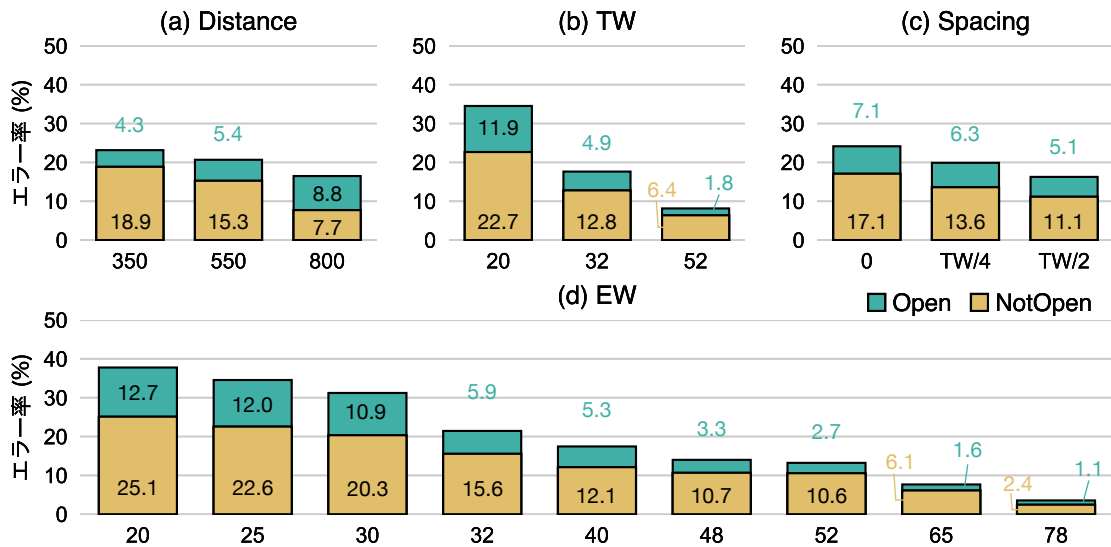


図 3.20: (a) *Method*×*Distance*, (b) *Method*×*TW*, (c) *Method*×*Spacing*, (d) *Method*×*EW* の拡大レンズが起動した際のエラー (Open) と拡大レンズが起動しなかった際のエラー (NotOpen) の積み上げ縦棒グラフ。

言える。

## 3.6 実験 2 に関する議論

### 3.6.1 拡大レンズによるパフォーマンスの改善

#### 選択時間

実験結果より, 多くの *EW* に対して Bubble Gaze Lens (BL) の方が Bubble Gaze Cursor (BC) よりも高速に動作した。特にターゲットサイズが小さい時 ( $TW=20, 32$ ), BL は BC よりも有意に低い選択時間であった。ゆえに Bubble Gaze Lens は Bubble Gaze Cursor よりも密集した小さなターゲットを高速に選択可能な手法だと言える。

一方で *EW* が大きくなるほど BL と BC の選択時間の差が小さくなり,  $EW=78$  においては BL の方が選択時間が大きくなった。これは拡大レンズ起動によってターゲット位置が変化し, それを目で追うことによって選択時間が増加したためだと考えられる。大きな視線移動 (サッケード) を行う場合, サッケードの着陸位置の決定のために 200 ms 程度視線の動きが止まる (潜時)。この潜時とサッケードによって通常の Bubble Cursor よりも選択に時間がかかっていると考えられる。Bubble Lens においてもターゲットが十分に大きくなると Bubble Cursor よりも選択時間が遅くなることが確認されており, 起動される拡大レンズ内に一定以下のサイズのオブジェクトが存在しない限り拡大レンズを起動しない (Bubble Cursor と同様の動作をする) ことでこの問題を解消している。本実験にお

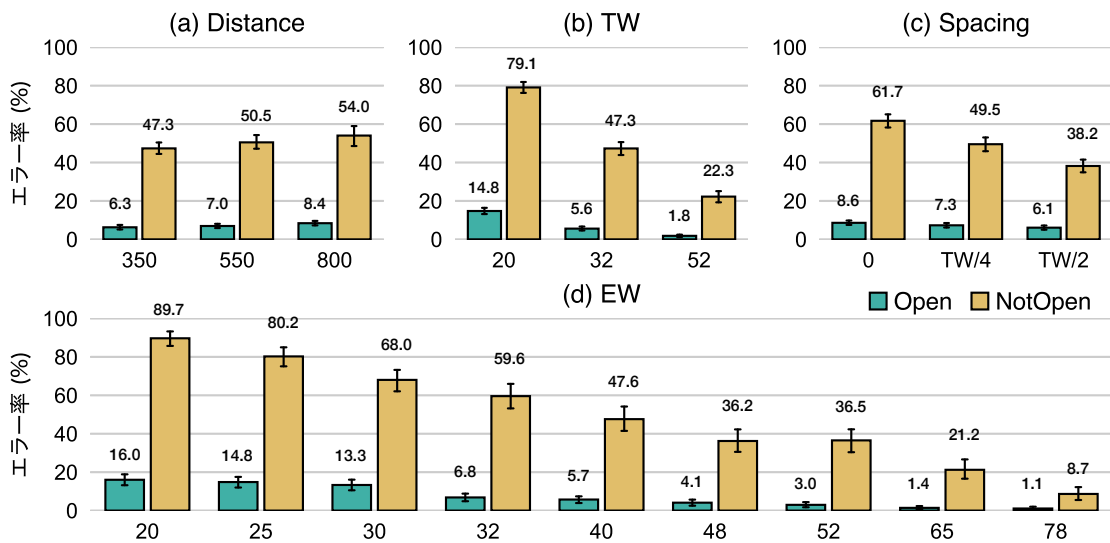


図 3.21: (a) *Method* × *Distance*, (b) *Method* × *TW*, (c) *Method* × *Spacing*, (d) *Method* × *EW* の拡大レンズが起動した際のエラー (Open) と拡大レンズが起動しなかった際のエラー (NotOpen) の割合. エラーバーは 95%信頼区間. なお, このエラー率は条件ごとの試行に対するエラーの割合である.

いてはこの拡大レンズの起動を決定するターゲットサイズの適切な閾値が不明であったため, この処理を実装しなかった. ただし, 今回の実験より, この閾値は  $EW=78$  付近に存在すると考えられる. ゆえに拡大レンズの起動の閾値にこの値を採用することで, 選択時間の増大は防ぐことが可能だと考えられる.

### エラー率

実験結果より BL は BC に対して約半分以上エラー率を改善し, 全ての従属変数の組み合わせにおいて BL は BC よりも有意にエラー率が低かった. したがって拡大レンズを用

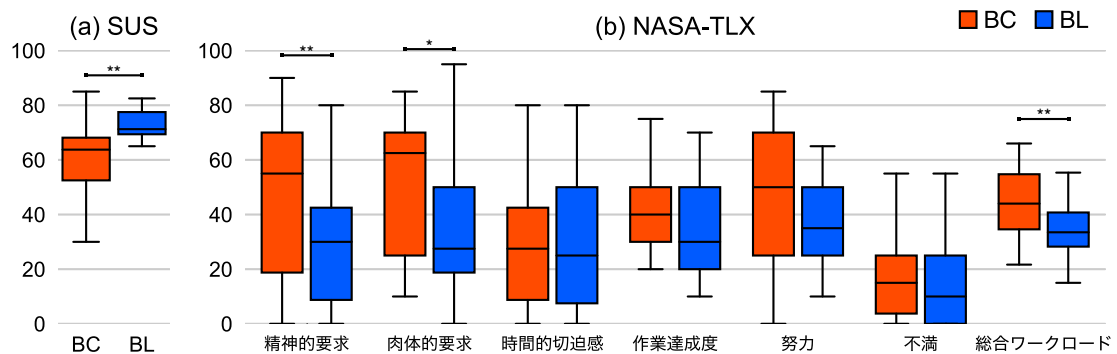


図 3.22: *Method* ごとの (a) SUS, (b) NASA-TLX の結果. 有意差は  $*p < .05$ ,  $**p < .01$ .

いることで密集した小さなオブジェクトの選択が容易になることが明らかになり、視線インタフェースにおいても Bubble Lens は Bubble Cursor のパフォーマンスを改善することが出来た。

しかし BL の選択精度は低く、小さなターゲットの選択は未だ難しいままである。BL のエラーの主な原因は拡大レンズが 3 割弱起動しなかったことである。BL のエラー率 20.0 % のうち 6.3 % が拡大レンズ起動時のエラー率であった。このエラー率は  $EW$  が大きくなるほど低くなり、 $EW=32$  の時、全体の試行に対する拡大レンズ起動時のエラー率は 5.9 %、拡大レンズが起動した試行に対するエラー率は 6.8 % であった。ゆえに、拡大レンズの起動率を 100 % にすることが出来れば、 $EW=32$  のオブジェクトまでは高精度で選択することが可能になると言える。一方でターゲットサイズが  $EW=20, 25, 30$  の時、拡大レンズが起動したとしてもエラー率は 10 % 以上であった。拡大倍率 4 倍によってターゲットが拡大されてもエラー率が高い確たる原因は不明だが、一つに潜時による拡大レンズ起動時の視線の固定が考えられる。拡大レンズが起動すると、視線座標の直近のターゲットが選ばれている状態で潜時が発生することがある。ここで選択されているオブジェクトがターゲットではないとすると、200 ms 程度、間違ったオブジェクトに視線を固定した状態から視線の移動が始まる。そしてそのオブジェクトに対する滞留時間が閾値を超え、エラー (Midas Touch) が生じている可能性がある。また BL のエラー率は  $Distance$  が大きくなるほど減少した。これは  $Distance$  が大きくなるほど拡大レンズ起動率が大きくなる (図 3.18) ためだと考えられる。

Bubble Gaze Lens でも選択が難しいような小さなターゲットを選択するための容易な対処法は、拡大レンズの倍率を上げることである。本実験では倍率を 4 倍とし、拡大レンズのサイズを 560 ピクセルとしたが、これらの値を上げれば十分なターゲットサイズは容易に確保することが出来る。ただし、拡大倍率を上げすぎると元々大きかったオブジェクトが不必要な大きさにまで拡大され低しまう現象が生じてしまう。これは拡大レンズ内のオブジェクトの大きさに合わせて動的に拡大倍率を変えることで対処できると考えられる。

### 3.6.2 拡大レンズ起動アルゴリズムの性能

#### 拡大レンズ起動率

拡大レンズ起動率は  $Distance$  に対して大きな主効果を有しており、 $Distance$  が大きくなるほど拡大レンズ起動率は大きくなった。本実装における拡大レンズ起動アルゴリズムは、二次サッケードの検出を行うアルゴリズムである。二次サッケードは視線とターゲット中心間の距離が大きくなればなるほど発生する割合が大きくなることが以前の研究で明らかになっており [207]、本実験の結果とも一致する。しかし、全てのサッケードに二次

サッケードが含まれることはない。これは二次サッケードをトリガーとすることの限界が表れている。特に近距離に対するトリガーとしては、現状の拡大レンズ起動アルゴリズムは不十分だといえる。また、拡大レンズ起動アルゴリズムの改善が必要だと考えられる。現在のアルゴリズムの二次サッケード検出は速度閾値を用いる手法 [167] であり、検出精度は低い。二次サッケードの検出手法は様々提案されているが、基本的にオフラインにおけるデータ分析を前提とした手法である [142]。ゆえに動的な二次サッケードの検出手法を検討していく必要がある。

### ターゲット非包括率

拡大レンズによってターゲットが拡大されなかった割合（ターゲット非包括率）は、図 3.19 より高くても 2.0 % 程度であり非常に低いと言える。ターゲット非包括率が比較的高かった条件は、*Distance* が遠い時と、*TW* が小さい時であった。前者はターゲットまでの距離が遠いことによって視線がアンダーシュートし、ターゲットに到達する前に拡大レンズが起動したと考えられる。後者はターゲットが小さいことによって拡大レンズ内にターゲットが包含されなかった回数が増えたためだと考えられる。まとめると、本章で提案した拡大レンズ起動アルゴリズムの起動タイミングは概ね適切であり、ターゲットへ視線が到達するタイミングを適切に計測することに成功していると言える。

### 3.6.3 拡大レンズの欠点

本実験より、拡大レンズの起動による選択時間の増加は最低限に抑えられ、また拡大レンズの起動によるターゲット拡大は小さなオブジェクトの選択を容易にすることを確認した。一方で現状の拡大レンズの実装は何点か制限を有している。まず、拡大レンズの起動取り消し操作が存在しない。実験においてはターゲットが含まれない位置で拡大レンズが起動すると、その試行をエラーとした。しかし実際のタスクにおいては拡大レンズ起動の取り消し後にターゲットを再度選択するような動作を取るが、現状の実装ではそれができない。ゆえに拡大レンズの起動取り消し方法の検討が必要だと言える。この機能の最も素朴な実装は拡大レンズに取り消しボタンを実装することであり、これは十分に動作すると考えられる。次に視覚探索タスクなど、何度も視線を移動させる必要があるタスクには不向きである。本実験のタスクはターゲットが明示的であり、一度の視線移動でそのターゲットを選択することが出来た。ゆえに拡大レンズはターゲット付近で起動したが、複数の視線移動が必要な複雑なタスクにおいては視線が移動するたびに拡大レンズが起動する恐れがある。これは情報を見るための視線移動と拡大レンズが起動する視線移動が一致することがあるためである。したがって Bubble Gaze Lens はアイタイピングなど、ター



ゲットが明示的なタスクに適しており、それ以外のタスクにおいてはポイントカーソルを使用した方が良い可能性がある。

#### 3.6.4 提案手法の制限

本研究で用いたアイトラッカは非常に安価（約2万円）であり、サンプリングレートは90 Hzである。このサンプリングレートはマイクロサッケード（短いサッケード）を取得するには非常に小さい値である [210, 112] ため、二次サッケードの検出に適していない。二次サッケードのピーク速度域の挙動を正確に検出するためには240 Hz以上のサンプリングレートを有したアイトラッカを用いる必要がある [210]。しかしこれらのアイトラッカは非常に高価であるため使用しなかった。ゆえに、より高いサンプリングレートを有したアイトラッカを使用することで、さらに二次サッケードの検出率を上げることが出来る可能性がある。また、サンプリングレートを補う手法として視線データにアップサンプリングを行う手法が提案されている [112]。この手法によって低いサンプリングレートであっても二次サッケードの検出精度を向上させることが出来るが、この手法はオフラインを前提とした手法であり、リアルタイムでの適用は難しい。ゆえに本研究では用いなかった。

### 3.7 視線インタフェースにエリアカーソル法を導入することへの議論

本章では Bubble Cursor を視線インタフェースに導入した Bubble Gaze Cursor の評価実験と、Bubble Lens を視線インタフェースに導入した Bubble Gaze Lens の評価実験を行った。以下ではこの二手法に対する総合的な議論を行う。

#### 3.7.1 一般的な視線入力手法との比較

Bubble Gaze Cursor の評価実験にて、Bubble Gaze Cursor はポイントカーソルを用いた滞留時間入力手法よりも高速かつ正確に小さなオブジェクトを選択することができた。そして Bubble Gaze Lens の評価実験にて、Bubble Gaze Lens は Bubble Gaze Cursor よりも正確に実質的な幅が小さなオブジェクトを選択することができた。したがって実験タスクのようなポインティングタスクにおけるこれらの視線入力手法の性能は、Bubble Gaze Lens (Bubble Lens) > Bubble Gaze Cursor (Bubble Cursor) > 通常の視線インタフェース (Point Cursor) だと言える。ただし、全てのタスクにおいてこれらの性能関係が成立するわけではなく、ポイントカーソルに対するエリアカーソルのデメリットは存在する。一つにオブジェクトを見ていないのにオブジェクトが選択される可能性がある。

通常の視線インタフェースでは Midas Touch を避けるために選択の意図がない限りオブジェクトに視線を向けないことが、エリアカーソルを用いる場合はオブジェクトに視線を向けずとも選択が実行されることがある。元々のエリアカーソル手法はマウスのクリック操作によって選択を行うため、カーソルがいつ、どのオブジェクトに対してフォーカスしているかは一切影響しなかった。しかし視線インタフェースではカーソルのフォーカスが生じた時点で滞留時間のカウントが開始されるため、Midas Touch が生じうる。この問題を解消するためにはカーソルが現在どのオブジェクトを選択しているかを明示的にする必要があり、バブルカーソルのフィードフォワードはどのターゲットにフォーカスしているか比較的分かりやすいが、より分かりやすくするためにボロノイ領域を描画するなどの工夫が必要だと言える。

### 3.7.2 環境ごとの提案手法の有用性

提案手法の評価実験ではデスクトップ PC の操作を前提とした実験デザインを設定しており、提案手法がデスクトップ PC におけるポインティング性能を向上させることは確認した。しかし提案手法はデスクトップ PC 以外にも様々な環境で小さなオブジェクトの選択を容易にすると考えられる。

視線入力を用いて小さなオブジェクトを選択しなければならない状況として、以下が考えられる。一つは視線推定精度の低さによる選択に必要なターゲットサイズの増加、である。視線推定精度は日光の影響やキャリブレーション制度に大きく影響を受ける [38]。また CPU の性能やディスプレイに対する顔の位置 [234] にも影響を受けるため、視線推定精度は時間によって大きく変動する。ゆえに十分なターゲットサイズのみで構成された UI においても、小さなターゲットに対する選択動作は生じ得る。これが特に生じるのはスマートフォンで視線入力を行う場合である。スマートフォンの視線推定深度は CPU の都合上、PC による視線推定よりも低い [48]。その上スマートフォン使用時に顔が完全にカメラに映り込まない割合は約 24 % と高く [87]、実用時にはより視線推定精度が下がることが予想される。したがって、デスクトップ PC 以外にもスマートフォンなどのモバイルデバイスに対する視線入力において提案手法は有用だと考えられる。

次に、距離の増加や遮蔽によるターゲットサイズの減少、が考えられる。三次元環境ではオブジェクトは世界に固定されるため、距離が離れるほど小さくなる。またターゲットが何かで遮蔽され、ターゲットの可視領域が減る場合においてもターゲットサイズは減少する。ゆえに VR や MR 環境で表示される仮想オブジェクト選択 [189, 25] や、IoT デバイスの選択 [90] などで小さなオブジェクトの選択動作は必ず生じる。実際、VR 環境におけるコントローラレイにバブルカーソルを適用する手法は小さなオブジェクトや遮蔽され

たオブジェクトの選択を容易にする [108] ことから、視線インタフェースにおいてもエリアカーソルの適用は効果的だと考えられる。

### マウスと視線の違い

Bubble Cursor はターゲット拡大手法であり、マウス・視線の両方においてもターゲット拡大によって小さなターゲットの選択を容易にしている。しかし、これらのインタフェースごとの Bubble Cursor の選択手順は厳密には異なる。まずマウスインタフェースでは、カーソルをターゲットに近づけ、バブルがターゲットを含んだとき、クリック動作を行う。つまりターゲットが拡大されていることによってマウス移動距離が減り、それによって選択時間が減少している。そのため、ボロノイ領域をフィードフォワードとして与えるなどターゲット領域を明示的にする処理を行うと、ポインタを動かす軌跡が最適化され選択時間が減少することが明らかになっている [51]。一方、視線インタフェースではポイントカーソルと同様に、ターゲットを注視することで選択を行う。つまり拡大された領域に応じた選択動作はなく、ターゲットサイズが大きいほど注視データがターゲット内に収まりやすいがゆえに、選択が容易になっていると言える。したがってマウスでは重視されなかった要素が視線インタフェースにエリアカーソルを導入する際に重要になる可能性がある。

本論文では検討していないが視線インタフェース + エリアカーソルにおいて重要である可能性の高い要素は「領域中心とターゲットの位置関係」である。視線インタフェースにおいてターゲットを選択する際の最適戦略は、ターゲットの中心を見ることである。なぜなら視線は固視微動によって分散が大きく、それを内包するためにはターゲット中心に分散の中心を配置することが適切だからである。しかし、ボロノイ領域の中心はターゲット中心と一致しない場合がほとんどであり、その場合はこの最適戦略は実行されない。領域中心とターゲット中心が一致しない例としては、円上に等間隔で配置されたオブジェクト群によるボロノイ図が挙げられる。このボロノイ図では、ボロノイ領域がそれぞれオブジェクトを末端とした放射状に広がっている。この領域における適切な選択方法はターゲットを一切見ずに領域の中心に目を向けることだが、注視対象がない中の注視動作は難しい。ゆえに、Bubble Cursor を視線インタフェースに導入するにあたり、これらの影響、および対処法は調査の必要があると考えられる。

### 3.7.3 他のエリアカーソル手法・拡大アルゴリズムの検討

本研究ではエリアカーソル手法のうち、Bubble Cursor と Bubble Lens を視線インタフェースに導入した。しかしこれら以外にもエリアカーソル手法は多数存在し [20, 23, 108, 40, 51]、それらを視線インタフェースに導入することでより視線入力性能を向上させるこ

とができる可能性がある。大半のエリアカーソル手法に関する研究は Bubble Cursor を元手法としており、Bubble Cursor の改善を行うアプローチを採用することで Bubble Cursor よりも優れたエリアカーソル手法の開発を行っている。その中でもターゲット領域の更なる拡大を検討する研究は、視線インタフェースにおいても役立つ可能性が高い [20, 51].

また本研究で用いたエリアカーソル手法はボロノイ分割アルゴリズムを使用したがる、ボロノイ分割以外にも空間分割アルゴリズムは存在し [11, 20, 51], それらの中にボロノイ分割よりも視線インタフェースに適したアルゴリズムが存在する可能性がある。Baudisch らは整列されていないオブジェクトの拡大手法として、放射状にオブジェクト領域を拡大する Starburst を提案した [11]. ボロノイ拡張の場合は密度が高いオブジェクトサイズを十分に拡大できない場合があるが、この手法はオブジェクト群の周りに空間がある時、密度が高くとも十分にターゲットサイズを拡張することができる。Guillon らはユークリッド距離による空間分割（ボロノイ分割）の他にマンハッタン距離による空間分割を検討した [51]. ゆえにこれらの分割アルゴリズムも検討の余地があると言える。また、基本的に注視データは横方向よりも縦方向の方が分散が大きいいため、領域分割も縦方向に長い分割の方が良い可能性がある。ゆえに視線インタフェースように領域分割アルゴリズムを作成する研究が必要だと言える。

### 3.8 本章のまとめ

本章ではエリアカーソル手法を視線インタフェースに導入することで、小さなオブジェクトを選択する手法を提案した。本研究ではエリアカーソル手法のうち Bubble Cursor と Bubble Lens に焦点を当て、それらを視線インタフェースに適用した Bubble Gaze Cursor と Bubble Gaze Lens を実装した。Bubble Gaze Cursor は速度、精度、ユーザビリティ、作業負荷の点で通常の視線インタフェースよりも優れ、かつ小さなオブジェクトの選択が容易であった。また Bubble Gaze Lens は Bubble Gaze Cursor よりも密集した小さなオブジェクトの選択に優れていた。

## 第4章 Kuiper Belt: VRにおける自然ではない視線角度を用いた視線入力手法の提案

第3章ではターゲットを単一選択動作のみで視線選択する手法を提案し、検証を行った。そこで提案した Bubble Gaze Cursor, Bubble Gaze Lens は小さなターゲットを、選択時間を増加させずに高い精度で選択することができた。一方で密集した極端に小さなターゲットの選択精度は未だ高いままであり、単一選択動作による選択の限界があり、複数段階の選択動作による複雑な入力の実装が必要だと言える。

また、これらの手法、特に Bubble Gaze Lens は視覚的な探索を伴うタスクに不向きである。そしてこの課題は滞留時間を用いて直接ターゲットを選択するような手法、最も使用されている視線入力手法の全てに対して共通する課題である。滞留時間入力はオブジェクト内に一定時間視線を滞留させると注視したオブジェクトが選択される手法である。つまりターゲットの選択にはターゲットに対して視線を向ける必要がある。一方でこの視線動作は日常生活において頻繁に行われており、何かの情報を探す、参照する、無意識に視線を漂わす、などの視線動作によって特定のオブジェクトに対して視線が滞留することは選択の意図なしに起こり得る。この「選択に必要な視線動作」と「日常的に行われる視線動作」が一致していることが原因でオブジェクトの誤選択が生じてしまう。これを視線インタフェース研究領域ではギリシャ神話になぞらえ、Midas Touch と呼ぶ [78]。

Midas Touch を解消する最も容易な方法は、滞留時間を長くすることである。日常的には行われなほど長い時間を滞留時間に設定することで、日常的に行われな選択動作を実現する方法である。滞留時間の増大は選択時間の増大に繋がるため、研究者は適切なトレードオフを達成する滞留時間を調査した。その結果、多くの視線インタフェースに関する研究では 500–600 ms 程度の滞留時間が、エラーの発生に対してより慎重になる場合は 1000 ms 程度の滞留時間が採用されている [119]。しかしこれらの滞留時間が適切であるタスクは、アイコン選択やキーボード入力といったターゲットが明確なタスクに限る。視覚的情報を参考にターゲットを探す行為（視覚探索）など視覚的情報を知覚するタスクにおいては、オブジェクトが持つ視覚的情報（テキスト、外観など）が大きくなればなるほど、そのオブジェクトを注視する時間は長くなる。つまり前述した滞留時間では不十分と

なり、より長い滞留時間 (2000 ms 以上 [148]) を設定する必要がある。これは非常に長い選択時間が必要になることと同義であり、単純な滞留時間の増加アプローチのみでは不十分であることを示している。

本章では「視線と頭部方向が成す角度」(以後、視線角度) が極端に大きな領域を活用することで Midas Touch 問題の解決を目指す。本研究においては視線角度が  $25^{\circ}$ – $45^{\circ}$  の領域を Kuiper Belt と呼ぶ。人間の目の水平方向の可動範囲は平均で約  $\pm 45^{\circ}$  である [183, 182]。一方で、視線は基本的に視線角度  $25^{\circ}$  以内に分布しており [65, 43], 視線シフト後の視線角度は平均で  $20^{\circ}$  以内である [183, 182]。すなわち人は意図しない限り  $25^{\circ}$  を超える視線角度に目を固定することはほとんどない。つまり、 $25^{\circ}$  以上の視線角度で行われる視線インタラクションは Midas Touch が生じにくいと考えられる。ゆえに Kuiper Belt にメニューを配置するメニュー UI は視覚的探索による Midas Touch が生じず、また Kuiper Belt に対する視線インタラクションは他の領域に対する入力動作と明確に区別することができる。

本章の目的は VR・MR 環境での視覚探索タスクにおける Midas Touch 問題を解決することである。そのため Kuiper Belt (図 4.1, 4.2) におけるメニューアイテム配置の設計原理を検討し、視線角度が大きい領域内のどこにメニューアイテムを配置すべきか調査する。実験 1 では「自然ではない視線角度」の領域を活用したメニュー選択手法のデザイン原理を検討するために、Kuiper Belt のどこにメニューアイテムを配置すべきか調査した。実験 2 ではユーザの負担を考慮したうえで、Midas Touch 問題を解決するために Kuiper Belt をどのように使用すべきか検討した。具体的には他の手法と比較することによって Kuiper Belt を用いたメニューアイテム選択手法の有用性と利用負荷を計測した。

本研究の貢献を以下に要約する。

- 視線角度が大きい領域である Kuiper Belt のどこにメニューを配置すべきか、という設計原理を検討した。
- Kuiper Belt を用いたメニューアイテム選択手法の有用性をユーザビリティや負荷の側面から検討した。
- Kuiper Belt を用いた視線インタフェースが Midas Touch を減少させることを確認した。

## 4.1 関連研究

### 4.1.1 ターゲットとラベルの分離による Midas Touch 問題の解決手法

一般的な視線入力研究の実験タスクではターゲットが明確に強調表示されているか、キーボードのようにターゲットが特定の位置に固定されている。しかし実用的で複雑なタス

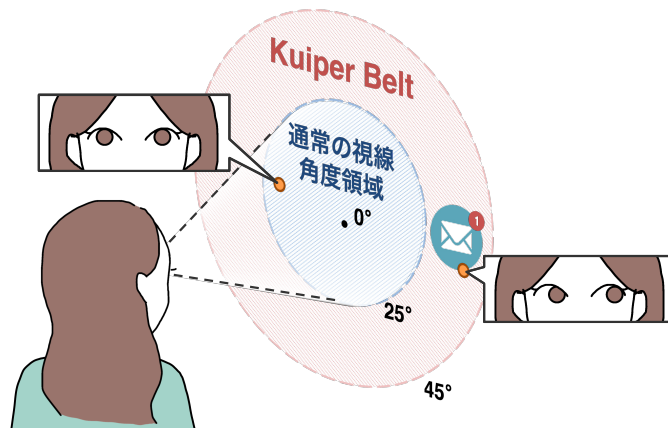


図 4.1: VR における Kuiper Belt. 通常行われなほど大きな視線シフトを用いることで Midas Touch 問題を軽減する.

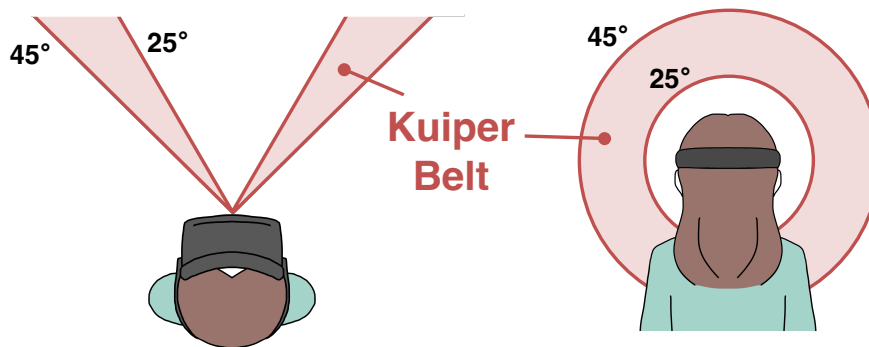


図 4.2: Kuiper Belt の定義領域.

クの場合、ユーザは様々なオブジェクトの中から位置が不明なターゲットオブジェクトを視覚的に検索する必要がある（例：GUI 操作）。このようなタスクを視覚探索（Visual Search）タスクと呼ぶ。Zhang らは複雑な視覚探索が必要なタスクでは Midas Touch を十分に減らすために必要な滞留時間が長くなることを確認した [233]。Zhang らの実験タスクは、中央にラベルが書かれた 7 つの候補ターゲットから、指定されたラベルを持つターゲットを選択することである。実験の結果、滞留時間が 1100 ms（一般的な滞留時間よりも長い [119]）のとき、高難度の視覚探索タスクのエラー率は 16.9 %であった。ゆえに視覚探索の難易度が高い場合、長い滞留時間の設定のみでは Midas Touch 問題を解決することはできない。

Zhang らの実験タスクで高いエラー率が記録された理由の一つに、ターゲット内にラベルが書かれていることが挙げられる。二次元環境において、Penkar らはターゲットとラベルを分離することによって Midas Touch を減らすことが可能であり、かつ短い滞留時間（200 ms）を用いても Midas Touch が生じないことを確認した [146]。Penkar らによる

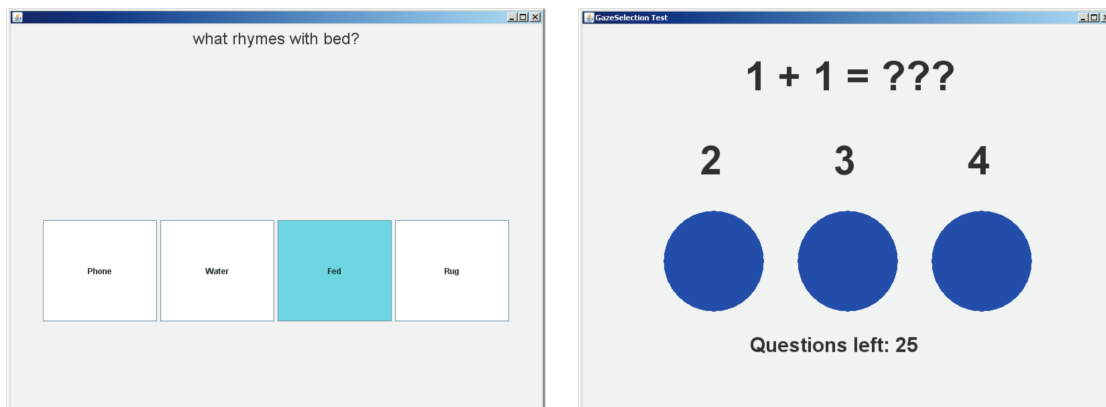


図 4.3: Penkar らの実験で用いたアプリケーションの図 [146]. 左図ではボタン内にラベルが配置され、右図ではボタン外にラベルが配置されている。

ターゲットとラベルの分離の様子を図 4.3 に示す. 同様に, このターゲットとラベルの分離を視線入力に活用した事例として Lutteroth らの Actigaze [109] が挙げられる. Actigaze はウェブブラウザ上で視線のみでハイパーテキストのリンクを選択する手法である. この手法はハイパーリンクを含んだ文章 (ラベル) と選択ボタン (ターゲット) を分離し, 選択するハイパーテキストの決定とリンクのクリックを離散的に行うことでハイパーリンクを選択する. そのため, 所与のハイパーリンクを探す (視覚探索) 最中に Midas Touch が生じない. Actigaze の滞留時間は 200 ms と非常に短い値であったが, 4 % 以下の低いエラー率を達成した.

#### 4.1.2 人の視線の分布と特徴

日常生活での視線と頭部方向が成す角度 (視線角度) は基本的に  $25^\circ$  以内に収まる.  $25^\circ$  以内の視線角度は人間の目にとって快適な領域であり, 自然な目の動きでこの角度外に移動することはほとんどない. Hu らは VR シーン視聴時, 98.7 % の視線データが中央から  $17.5^\circ$  の位置に分布していることを報告し [65], Foulsham らは歩行中の人の視線データのほとんどが  $25^\circ$  以内に分布する (図 4.4) ことを報告した [43].

Stahl らは, 目の離心率が大きくなるほどに眼球運動学 (Ocular Kinematics) が悪化するのために, 極端な視線角度が用いられないと仮定した [183, 182]. 例えば, サッケード着地点の離心率が大きくなるほどサッケード精度は低くなり [218], また  $30^\circ$  を超える視線角度に目を固定し続けると終末位眼振 (end-point nystagmus; 振幅  $0.2^\circ$ – $2.5^\circ$  の眼球の揺れ) が健常者の 50 % に生じる [1]. 加えて視線角度が大きくなるほどアイトラッカの精度は悪化し, 視線インタラクションに必要なターゲットサイズは大きくなる [179]. ゆえに,  $30^\circ$  を超える視線角度の視線特徴は通常の視線角度 ( $< 25^\circ$ ) の視線特徴と異なる. 本章で



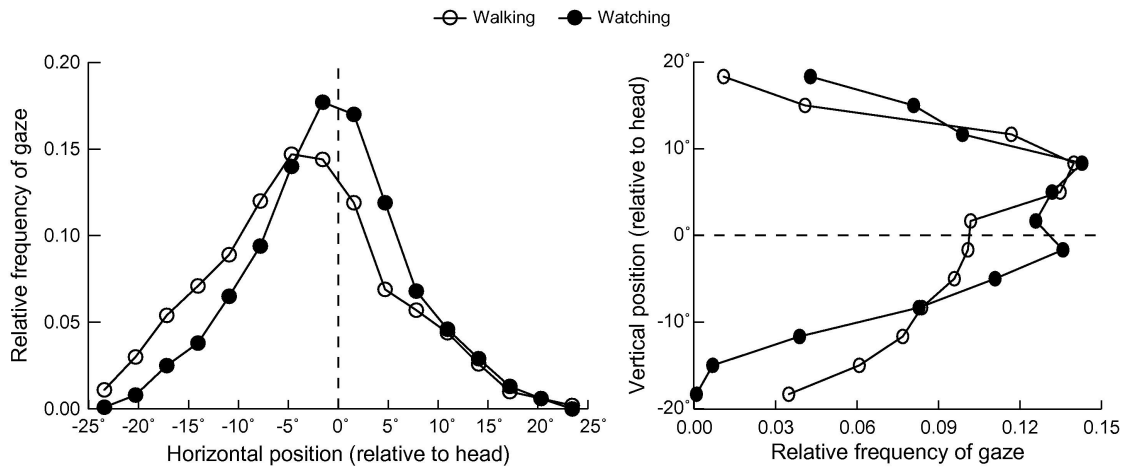


図 4.4: 水平方向（左図）と垂直方向（右図）の頭部に対する視線の分布 [43].

は極端な視線角度における視線インタラクションの特徴を理解し、VR 環境における視線インタラクションの設計原理の発見を目的としている。

#### 4.1.3 VR・MR における視線インタラクション領域

VR・MR では、選択可能なターゲットは通常 (1) HMD, (2) 周辺情報, (3) 三次元の座標やオブジェクト, に固定されている [37]. 視線ポインティング手法の多くは固定されたディスプレイとのインタラクションを前提としているため, 周辺情報に固定されたターゲット (つまり 2 と 3) を対象としている. 一方で HMD に固定されたターゲットを選択する視線ポインティング手法もわずかではあるが存在する. これらの手法 [70, 107, 121] におけるターゲットはユーザの HMD に固定され, ユーザの Field of View (FOV) 内の特定の場所に常に表示される. そのため, ユーザは即座にターゲットを見ることが可能である. しかし, 既存の HMD にターゲットを固定する手法ではターゲットが頭部方向に対して  $20^\circ$  以下の位置に固定されている ([70] は  $18.5^\circ$  未満, [107] は  $20^\circ$  未満, [121] は  $20^\circ$ ). さらにターゲット自体の大きさも考慮すると, 選択に必要な視線移動はより小さくなる. したがって人間の自然な視線角度は  $25^\circ$  以内であるため, これらの手法は自然な視線移動に伴って Midas Touch が生じることがある [107]. 一方で人間の目の最大可動範囲は約  $45^\circ$  である. ゆえに, 視線角度が  $25^\circ$ – $45^\circ$  の領域を利用することで Midas Touch 問題を解決することができると考えた.

## 4.2 提案手法: Kuiper Belt

本章では、極端な視線角度の領域である Kuiper Belt を活用することによって誤選択 (Midas Touch) を減らすことを目指す。本研究では Kuiper Belt を視線角度  $25^{\circ}$ – $45^{\circ}$  の領域と定義する。なぜなら、基本的に視線角度が  $25^{\circ}$  を超えることはほとんど無く [65, 43], かつ目の最大可動範囲は約  $45^{\circ}$  である [183, 182] ためである。この領域における視線インタラクションはほとんど検討されていないが、視線インタラクション研究において有望な研究分野だと考えられる。Saidi らは、頭部正面との角度が Microsoft HoloLens 1 の FOV (垂直方向に  $17.5^{\circ}$ ) から  $30^{\circ}$  離れた下方 (約  $38.75^{\circ}$ ) に存在するスマートフォンを、頭を動かさずに快適に見ることが出来ることを確認した [165]。この研究は視線インタフェースに関する研究ではないが、視線角度が大きい領域における視線インタラクションが有用である可能性を示唆している。

本研究では、Kuiper Belt 領域での選択操作が Midas Touch を低減させるかどうかを調査する 2 つの実験を行う。実験 1 では Kuiper Belt のデザインスペースの検討を行う。具体的には、Kuiper Belt 内のメニューアイテムをどこ (距離, 方向) に配置すべきか検討する。次に、実験 2 で Kuiper Belt が視覚探索における Midas Touch やユーザの負担 (生理的・心理的な作業負荷) をどの程度減少するか調査する。実験 2 では Kuiper Belt を用いたメニュー選択手法と通常の視線角度領域 ( $< 25^{\circ}$ ) を用いたメニュー選択手法、ベースライン手法の 3 つを比較する。

## 4.3 実験 1: Kuiper Belt のデザインスペースの検討

Kuiper Belt 内に配置されたメニューアイテムと視線インタラクションを行う際の適切なメニューアイテム配置を検討することを目的として、メニューアイテム選択タスクを行った。この実験は COVID-19 の感染拡大前に行われた。

### 4.3.1 実験参加者と使用機器

実験参加者として 10 名 (男性 9 名, 女性 1 名, 平均年齢 22.7 歳, 標準偏差 1.4 歳) が実験に参加した。全実験参加者の視力は正常範囲内であり、かつアイトラッキングを用いたユーザビリティ実験の参加経験があった。また本実験は極端な視線角度に視線を向ける必要があるため、眼鏡をかけている場合は視線が眼鏡のフレームに妨げられてアイトラッキングが失敗する。ゆえに眼鏡を使用している人は実験に参加しなかった。ただし、5 名はコンタクトレンズを使用していた。実験報酬は 1,000 円分の Amazon ギフトカードであった。実験時間は約 30 分であった。

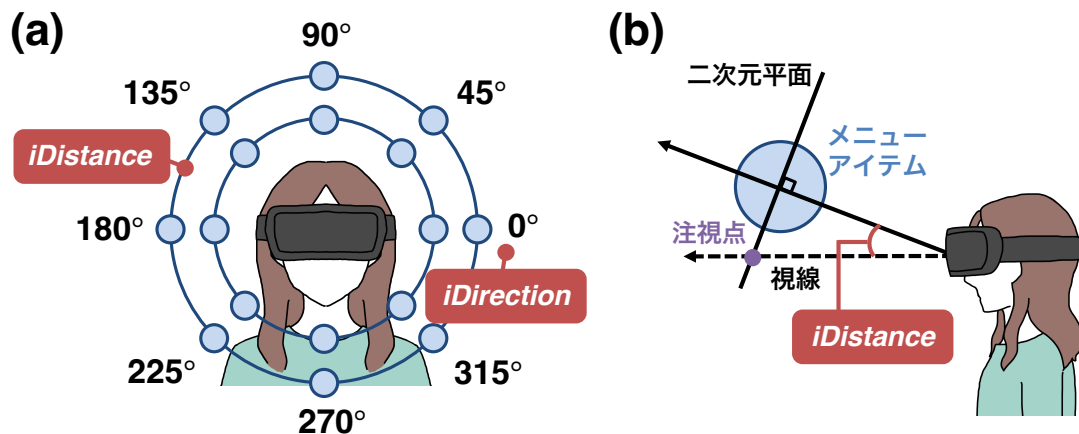


図 4.5: (a) 独立変数の定義, (b) 注視点の定義.

本実験は 2.60 GHz の Intel Core i9-9750H の CPU と NVIDIA GeForce RTX 2070 を搭載したパソコンで行った。実験に用いたソフトウェアは Unity ver. 2018.3.0f2 で作成した。HMD は HTC VIVE Pro Eye を用いた。HTC VIVE PRO Eye の FOV は水平方向に  $100^\circ$ 、垂直方向に  $110^\circ$  であり、搭載されているアイトラッカは 120 Hz のサンプリングレートであった。ただし、HMD のフレームレートが 90 Hz であったため視線データは 90 Hz で収集された。

#### 4.3.2 実験デザイン

実験は参加者内実験計画を用いて行った。独立変数は 2 つ存在し (図 4.5a)、それぞれ以下のように設定されている。

- メニューアイテムの距離 ( $iDistance$ ;  $12^\circ$ ,  $22^\circ$ ,  $32^\circ$ ,  $37^\circ$ ,  $42^\circ$ )
- メニューアイテムの方向 ( $iDirection$ ;  $0^\circ$ ,  $45^\circ$ ,  $90^\circ$ ,  $135^\circ$ ,  $180^\circ$ ,  $225^\circ$ ,  $270^\circ$ ,  $315^\circ$ )

$iDistance$  において、 $12^\circ$  と  $22^\circ$  は通常の視線角度であり、 $32^\circ$  と  $37^\circ$ ,  $42^\circ$  は Kuiper Belt (極端な視線角度) である。これらの値は通常の視線領域と Kuiper Belt を比較するために選択した。また、本実験の目的は Kuiper Belt のデザイン原理の検討であるため、通常の視線角度の  $iDistance$  ( $12^\circ$ ,  $22^\circ$ ) の間隔よりも Kuiper Belt の  $iDistance$  ( $32^\circ$ ,  $37^\circ$ ,  $42^\circ$ ) の間隔を狭くした。

実験で行う全試行は 15 のセッションで構成されており、それぞれのセッションは固定された  $iDistance$  と全ての  $iDirection$  の値の組み合わせにおける試行で構成されている。つまり、セッションは 8 つの異なる  $iDirection$  の値に応じた試行で構成され、かつ実験参

加者は5つの  $iDistance$  に対応するセッションと同じ内容を3回体験する。 $iDistance$  はラテン方格法を用いた順で、 $iDirection$  はセッションごとにランダムに提示された。ゆえに実験参加者は  $5 iDistance \times 3 \text{回} \times 8 iDirection = 120$  回のメニューアイテム選択を行った。実験参加者は10名であるため、合計1,200データが収集された。また、Kuiper Beltで安定した選択操作に必要なメニューアイテムの大きさを調査するために、ターゲット選択時の視線データを収集した。

### 4.3.3 実験手順

実験参加者が実験室に到着後、椅子に座らせ、実験の概要およびタスクの内容を説明した。その後HMDを装着させ、HTC VIVE Pro Eyeで提供されているアイトラッカの5点キャリブレーションを行った。そして実験参加者が実験タスクの挙動を理解するまで、実験タスクからパラメータを変更したタスク ( $iDistance=10^\circ$ ) を練習として行わせた。

実験タスクはHMDに固定されたメニューアイテムを繰り返し選択し続けるタスクである。実験タスクはまず、頭部正面に生成された半径  $1.0^\circ$  (判定領域は半径  $1.5^\circ$ ) の白色の球状メニューアイテムを視線滞留によって選択する(滞留時間は400 ms)。これは視線を頭部正面に移動させることを目的としており、この選択操作におけるデータの収集は行わない選択後、頭部正面のメニューアイテムが消え、それと同時に新しいメニューアイテムが生成される。また、頭部正面のメニューアイテムが存在していた位置には新しいメニューアイテムの方向を指す矢印が表示される。新しいメニューアイテムは半径  $1.0^\circ$  の白色の球であり、HMDから2.0 m離れた位置に配置される。このメニューアイテムは  $iDistance$  と  $iDirection$  の値に応じた位置に生成される。実験参加者がこのメニューアイテムの判定領域を注視すると、フィードバックとしてメニューアイテムが白から赤に変わり、ターゲットの周囲に現在の滞留時間に応じた円形スライダのフィードバック [103] が描かれた(図4.6)。実験参加者はできる限り速く、かつ正確にメニューアイテムを注視するよう指示された。メニューアイテムの判定領域は半径  $7.0^\circ$  の球であり、視線が判定領域に1秒間滞留(滞留時間は1000 ms)した(選択成功)か、滞留できずに5秒間経過した(選択失敗)ときにメニューアイテムは消え、再度頭部正面にメニューアイテムが生成された。このメニューアイテム選択を繰り返すことが実験タスクであり、頭部正面に生成されたメニューアイテムと新しく生成されたメニューアイテムの選択を行うことが1試行である。

実験参加者は合計15セッション行い、同じ  $iDistance$  の値のセッションを連続して3回繰り返した。3セッション完了後、実験参加者は生理快適度(Physical Comfort)と心理快適度(Mental Comfort)を計測するアンケートに回答した。このアンケートは  $iDistance$  (5回)ごとに繰り返し行った。全15回のセッションを終えた後、実験参加者は研究全体

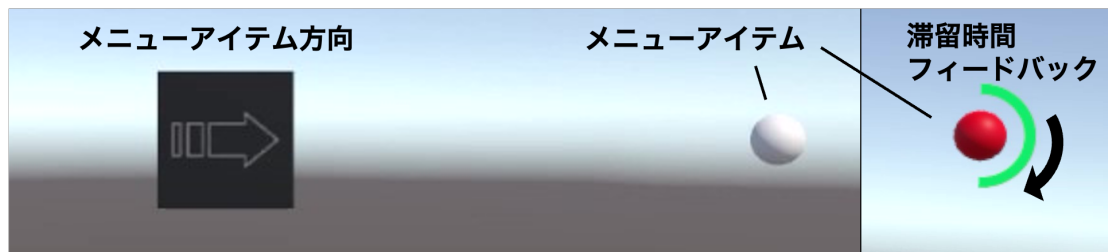


図 4.6: 実験セットアップとフィードバック.

に関するアンケートに答えた。

#### 4.3.4 データ収集と前処理

視線データとして視線ベクトルが記録された。また、以下の2つの方法でデータを抽出した [38]。(1) 選択に成功した場合、注視データ 1000 ms の最初の 200 ms を除いた 800 ms の視線データを抽出した。(2) 選択に失敗した場合、記録していた 5 秒間の注視データのうち最もメニューアイテムに接近した 800 ms の視線データを抽出した。後者の処理は全体の 7.7% の注視データに対して行われた。また注視時に、瞬きや不十分なアイトラッキングによって視線データが得られないことがある。これが 800 ms のうち 100 ms 以上含まれていた場合、その注視データを分析から除外した。この処理によって 1.1% の注視データが除外された。最終的に分析に用いたデータセットには 1,178 の注視データが含まれていた。視線ベクトルは、文献 [222] を参考に HMD とメニューアイテム中心を結ぶ直線に垂直な二次元平面に転写された (図 4.5b)。

生理および心理快適度に関するアンケートは Xu らのアンケート [217] を参考にしており、ユーザの生理および心理の快適度が  $iDistance$  と  $iDirection$  の組み合わせごとにどう変化するか調査するために行った。アンケートは 1-5 の五段階尺度 (1: タスク後に疲労を強く感じた, 5: タスク後に疲労をまったく感じなかった) が用いられた。そして精神的負荷を計測することを目的として、実験参加者は NASA Task Load Index (NASA-TLX) のアンケート [54] に回答した。

#### 4.3.5 結果

それぞれの従属変数 (エラー率, 生理快適度, 心理快適度) に対してノンパラメトリックな分散分析手法である整列ランク変換 (ART) [58, 166, 211] を行い、混合モデル REML を用いてデータを評価した。多重比較にはウィルコクソンの符号順位検定を用い、Holm 法 [62] による補正を用いた。NASA-TLX によって得られた値はフリードマン検定とウィ

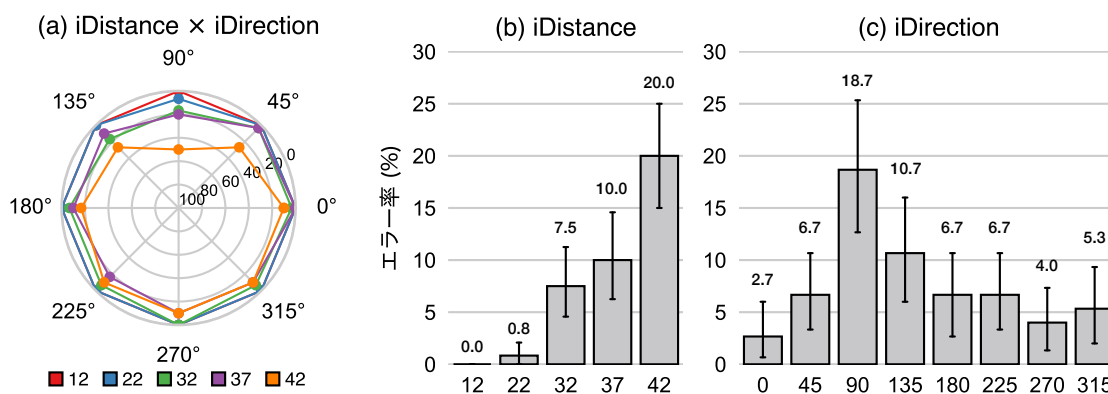


図 4.7: (a)  $iDistance \times iDirection$ , (b)  $iDistance$ , (c)  $iDirection$  ごとのエラー率. エラーバーは 95%信頼区間.

表 4.1:  $iDistance \times iDirection$  ごとのエラー率の値.

$iDistance$	$iDirection$							
	0°	45°	90°	135°	180°	225°	270°	315°
12°	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
22°	0.0	0.0	6.7	0.0	0.0	0.0	0.0	0.0
32°	3.3	3.3	16.7	16.7	6.7	6.7	0.0	6.7
37°	0.0	3.3	20.0	10.0	10.0	16.7	10.0	10.0
42°	10.0	26.7	50.0	26.7	16.7	10.0	10.0	10.0

ルコクソンの符号順位検定を用い, Holm 法による補正を用いた.

### エラー率

メニューアイテムの距離 ( $iDistance$ ), メニューアイテムの方向 ( $iDirection$ ) を独立変数とし, エラー率を従属変数として分析を行った. エラー率に対して  $iDistance$  ( $F_{4,1151} = 165.36, p < .01$ ),  $iDirection$  ( $F_{7,1151} = 31.23, p < .01$ ) に有意な差が確認された. さらに 1 次の交互作用として  $iDistance \times iDirection$  ( $F_{28,1151} = 11.59, p < .01$ ) が確認された. 図 4.7 と表 4.1 に  $iDistance$ ,  $iDirection$  のエラー率を示す.

### 生理快適度

メニューアイテムの距離 ( $iDistance$ ), メニューアイテムの方向 ( $iDirection$ ) を独立変数とし, 生理快適度を従属変数として分析を行った. 生理快適度に対して  $iDistance$  ( $F_{4,351} = 111.94, p < .01$ ),  $iDirection$  ( $F_{7,351} = 20.18, p < .01$ ) に有意な差が確認された. さらに 1 次の交互作用として  $iDistance \times iDirection$  ( $F_{28,351} = 2.84, p < .01$ ) が確

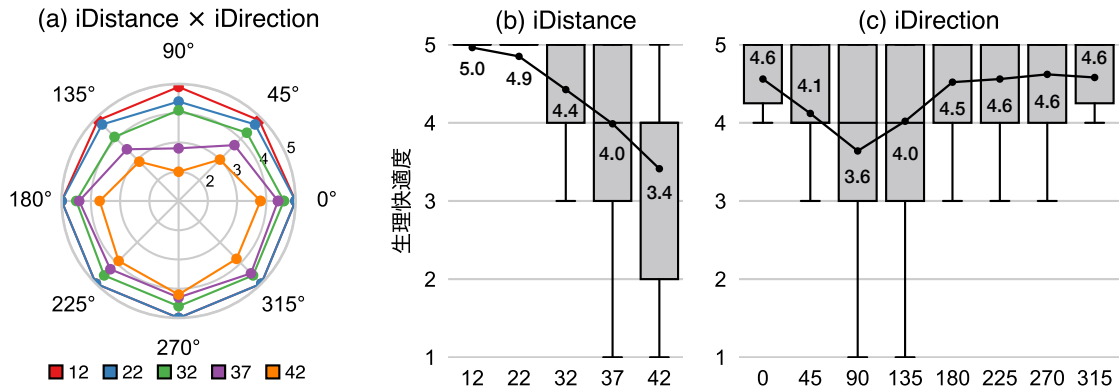


図 4.8: (a)  $iDistance \times iDirection$ , (b)  $iDistance$ , (c)  $iDirection$  ごとの生理快適度. ポイントプロットは平均値.

表 4.2:  $iDistance \times iDirection$  ごとの生理快適度の値.

$iDistance$	$iDirection$							
	0°	45°	90°	135°	180°	225°	270°	315°
12°	5.0	4.9	4.9	4.9	5.0	5.0	5.0	5.0
22°	5.0	4.7	4.4	4.7	5.0	5.0	5.0	5.0
32°	4.6	4.3	4.1	4.1	4.5	4.6	4.6	4.6
37°	4.4	3.7	2.8	3.5	4.4	4.3	4.3	4.5
42°	3.8	3.0	2.0	2.9	3.7	3.9	4.2	3.8

認められた. 図 4.8 と表 4.2 に  $iDistance$ ,  $iDirection$  ごとの生理快適度を示す.

### 心理快適度

メニューアイテムの距離 ( $iDistance$ ), メニューアイテムの方向 ( $iDirection$ ) を独立変数とし, 心理快適度を従属変数として分析を行った. 心理快適度に対して  $iDistance$  ( $F_{4,351} = 59.95, p < .01$ ),  $iDirection$  ( $F_{7,351} = 10.77, p < .01$ ) に有意な差が確認された. さらに 1 次の交互作用として  $iDistance \times iDirection$  ( $F_{28,351} = 2.11, p < .01$ ) が確認された. 図 4.9 と表 4.3 に  $iDistance$ ,  $iDirection$  ごとの心理快適度を示す.

### NASA Task Load Index (NASA-TLX)

$iDistance$  ごと ( $12^\circ, 22^\circ, 32^\circ, 37^\circ, 42^\circ$ ) の総合ワークロードスコアはそれぞれ 6.00, 10.90, 21.83, 35.23, 41.87 であった (低いほど良い). NASA-TLX によって得られた 6 つの項目に対するスコアと総合ワークロードスコアに対してフリードマン検定を用いて分析した結果,  $iDistance$  に対して精神的要求 ( $\chi^2_{4,N=10} = 23.33, p < .01$ ), 肉体的要求

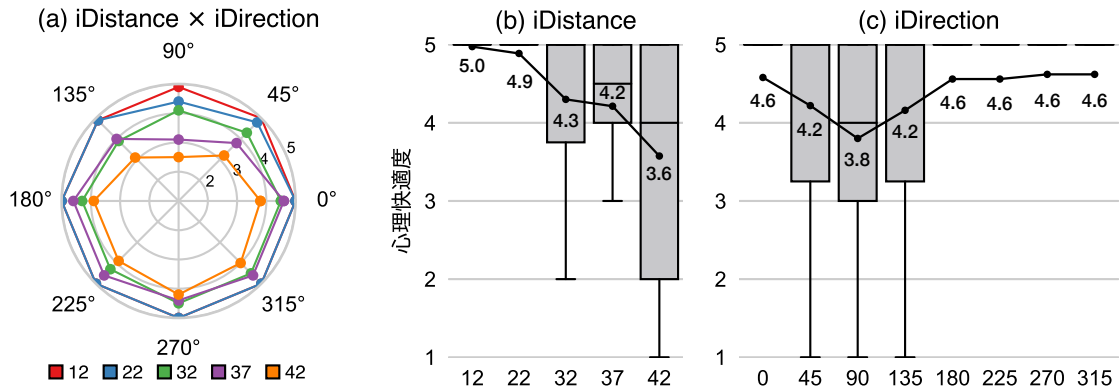


図 4.9: (a) *iDistance* × *iDirection*, (b) *iDistance*, (c) *iDirection* ごとの心理快適度. ポイントプロットは平均値.

表 4.3: *iDistance* × *iDirection* ごとの心理快適度の値.

		<i>iDirection</i>							
		0°	45°	90°	135°	180°	225°	270°	315°
<i>iDistance</i>									
12°		5.0	5.0	4.9	4.9	5.0	5.0	5.0	5.0
22°		5.0	4.8	4.4	4.9	5.0	5.0	5.0	5.0
32°		4.5	4.3	4.1	3.9	4.3	4.3	4.5	4.5
37°		4.6	3.8	3.1	4.0	4.6	4.6	4.4	4.6
42°		3.8	3.2	2.5	3.1	3.9	4.2	4.2	4.0

( $\chi^2_{4,N=10} = 31.01, p < .01$ ), 時間的切迫感 ( $\chi^2_{4,N=10} = 16.98, p < .05$ ), 作業達成度 ( $\chi^2_{4,N=10} = 22.88, p < .01$ ), 努力 ( $\chi^2_{4,N=10} = 29.80, p < .01$ ), 不満 ( $\chi^2_{4,N=10} = 18.62, p < .01$ ), 総合ワークロード ( $\chi^2_{4,N=10} = 30.67, p < .01$ ) に有意差が確認された (図 4.10).

### Kuiper Belt における安定した選択操作に必要なメニューアイテムの大きさ

メニューアイテムの *iDistance* が大きくなるほど、視線精度の悪化によってメニューアイテム選択のエラー率が高くなる。ゆえに Kuiper Belt で通常の見線角度と同じエラー率を維持しつつ安定した選択操作を実現するためには、メニューアイテムをより大きくする必要がある。そのため Kuiper Belt で安定した選択操作を達成するために必要な球直径を注視点の包含率から計算した (表 4.4)。このパラメータは実験 2 で用いた。

### 4.3.6 実験結果のまとめ

*iDistance* = 32°, 37° のエラー率は 10 % 以下だった。一方、*iDistance* = 42° では全体のエラー率が 20 % となり、選択が困難な方向が存在することから、Kuiper Belt における



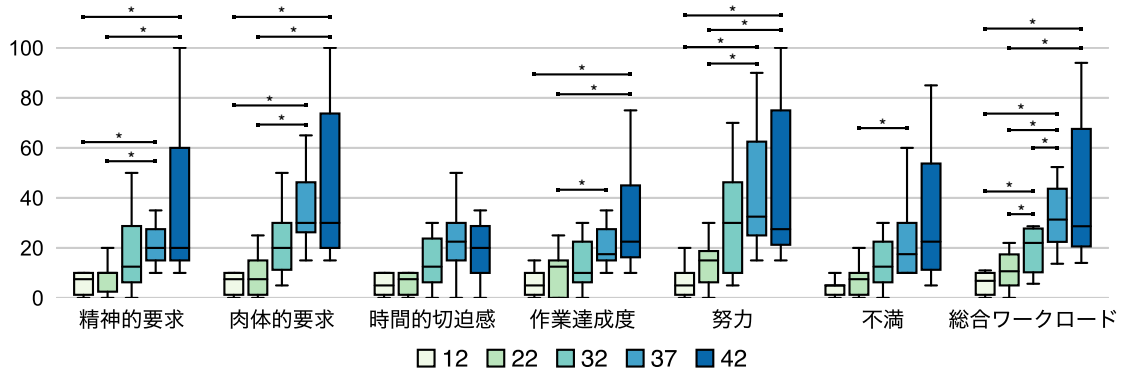


図 4.10: *iDistance* ごとの NASA-TLX の結果. 有意差は  $*p < .05$ .

メニューアイテム配置領域の候補は  $iDistance = 32^\circ, 37^\circ$  である. また,  $iDistance$  が大きくなるほどエラー率, 生理快適度, 心理快適度の指標は悪化した. 特に上方向のメニューアイテム選択負荷は大きく,  $iDistance = 42^\circ$  かつ  $iDirection = 90^\circ$  のときエラー率は 50%であった.

$iDistance$  が大きくなるほど安定した選択操作に必要なメニューアイテムは大きくなった. この結果は以前の研究結果 [179] と一致している. ゆえに Kuiper Belt で安定した選択操作を行うためには, 通常の視線角度に配置する場合よりメニューアイテムを大きくする必要がある. また, 滞留時間が短くなったときエラー率が減少した. これは長時間の Kuiper Belt への連続した注視が難しく, 短い滞留時間を採用することが好ましいことを示している.

実験結果より  $iDistance$  が大きくなるほどエラー率, 生理快適度, 心理快適度の指標は悪化した. これは, メニューアイテムの位置が人間の眼球可動域の限界に近く, 眼球を固定するための努力が必要になるためだと考えられる. 特にメニューアイテムの配置が上向き (上, 右上, 左上) になるほど指標の悪化は顕著であった. これは, 上方向の方が下方向よりも目の可動域が狭いことに起因すると考えられる. ゆえに, メニューアイテムは横向きか下向きに配置するのが好ましい. 結果として,  $iDirection = 45^\circ, 90^\circ, 135^\circ$  のとき  $iDistance = 32^\circ, 37^\circ$  が,  $iDirection = 180^\circ-360^\circ$  のとき  $iDistance = 32^\circ, 37^\circ, 42^\circ$  が好ましいインタラクション領域だといえる.

#### 4.4 実験 2: Kuiper Belt の性能評価実験

Midas Touch が頻発する視覚探索時のメニューアイテム選択タスクにおいて, Kuiper Belt が Midas Touch 問題の解決に有効であるか調査した. 本実験では, Kuiper Belt を用いたメニュー選択手法と他の 2 つの手法 (Two-step selection, Head-gaze) を比較した.

表 4.4: *iDistance* と *iDirection* ごとの, 安定した選択操作のために推奨されるメニューアイテム (球) の直径 (°).

	包含率			
	25 %	50 %	75 %	90 %
<b><i>iDistance</i></b>				
12°	1.40	2.33	3.72	5.20
22°	2.58	3.89	5.84	7.82
32°	2.90	4.83	7.45	11.68
37°	3.09	5.25	8.54	12.02
42°	4.15	7.17	11.12	16.27
<b><i>iDirection</i></b>				
0°	2.03	3.69	6.13	9.12
45°	2.90	4.75	7.60	11.27
90°	2.78	5.22	10.12	14.71
135°	3.37	5.36	8.39	11.70
180°	2.46	4.89	7.75	11.79
225°	2.57	4.24	7.05	11.75
270°	1.87	3.18	5.37	8.25
315°	2.30	3.71	6.26	8.77

実験の使用機器は実験 1 と共通である.

#### 4.4.1 実験参加者

19 名の実験参加者を募ったが, 1 名はキャリブレーションのエラーで除外した. そのため実験参加者として 18 名 (男性 15 名, 女性 3 名, 平均年齢 22.7 歳, 標準偏差 1.7 歳) が実験に参加した. すべての実験参加者は正常な視力の範囲内であり, 1 名のみアイトラッキングを用いた経験があった. また実験 1 と同様の理由で, 実験参加者は全員眼鏡を着用していなかった. 実験参加者は全員実験 1 を経験していなかった. 実験報酬は 1,000 円分の Amazon ギフトカードであった. 実験時間は約 60 分であった.

#### 4.4.2 比較手法

Two-step selection (2SS) と Head-gaze (HG), Head-gaze with Kuiper Belt (KB) の比較を行った. 図 4.11 にそれぞれの手法の概観と主観的な光景を示す. 本実験のタスクは視覚探索タスクである. 具体的にはプレートの中からターゲットパネルを探索し, メニューアイテムを選択するタスクである. メニューアイテムは実験 1 の *iDirection* と同じ 8 方向に配置された.

## Two-step selection method (2SS)

2SSはベースラインとなる視線メニュー選択手法である。ベースラインとして満たすべき要素は(A)2段階入力、(B)滞留時間選択、(C)メニューアイテムが視覚探索領域に存在、(D)HGやKBと同様の円形のメニューアイテム配置、である。2SSは以下の操作手順よりこれらの要素を満たしており、ベースラインとして妥当だといえる。

2SSによる選択は以下の2段階の操作で構成される。

1. ターゲットパネルを  $X$  ms ( $X$ は独立変数の値)の間注視することで選択する。
2. ターゲットパネル選択後、メニューアイテムがユーザの視線の先に空間に固定された状態で表示される。このメニューアイテムを  $X$  ms ( $X$ は独立変数の値)の間注視することで選択する。

ターゲットパネルとメニューアイテムは滞留時間を用いた選択手法で選択される。滞留時間は独立変数であり、ターゲットパネルとメニューアイテムの選択に必要な滞留時間は同じである。メニューアイテムは  $iDistance=12^\circ$  に配置されている。メニューアイテムの直径は  $5.2^\circ$  である。この直径は  $iDistance=12^\circ$  における安定した選択操作に必要なメニューアイテムの直径(表4.4)である。

## Head-Gaze method (HG)

本実験ではメニューアイテムを通常の視線角度領域に配置した手法HGと、メニューアイテムを Kuiper Belt に配置した手法KBの2つのHead-Gaze手法を設計した。HGはターゲットパネルとメニューアイテムの選択を同時に行う手法である。HGによる選択は以下の2段階の操作で構成される。

1. 頭部をターゲットパネルに向ける。
2. ターゲットパネルに頭部を向けたまま、メニューアイテムを  $X$  ms ( $X$ は独立変数の値)の間注視することで選択する。

2SSと異なり、HGのターゲットパネル選択は頭部方向によって行われる。HGのメニューアイテムはHMDに固定されており、かつ自然な視線角度である  $iDistance=12^\circ$  に配置されている。メニューアイテムの直径は2SSと同様に  $5.2^\circ$  である。

## Head-Gaze method with Kuiper Belt (KB)

KBにおけるメニューアイテムの選択方法はHGと同じである。しかし、メニューアイテムの配置位置と大きさは異なる。KBのメニューアイテムは実験1の結果を参考に、上部

( $iDirection=45^\circ, 90^\circ, 135^\circ$ ) は  $iDistance=32^\circ$ , そのほかの方向 ( $iDirection=0^\circ, 180^\circ, 225^\circ, 270^\circ, 315^\circ$ ) は  $iDistance=37^\circ$  に配置されている. メニューアイテムの直径は  $12^\circ$  である. この直径は  $iDistance=37^\circ$  における安定した選択操作に必要なメニューアイテムの直径 (表 4.4) である. ゆえに, メニューアイテムは 2SS や HG よりも KB の方が大きい, 選択の難易度は KB とその他で同じだといえる.

#### 4.4.3 実験デザイン

実験は参加者内実験計画で行った. 独立変数は 2 つ存在し, それぞれ以下のように設定されている.

- 手法 (*Method*; 2SS, HG, KB)
- 滞留時間 (*DT*; 200 ms, 400 ms, 600 ms)

視線インタフェースにおいて 600 ms の滞留時間は一般的であり, 多くの研究でこの値が使用されている [119]. そして独立変数 *DT* においては, 一般的なものよりも短い滞留時間 (200 ms, 400 ms) を選択した. これらの値は視覚探索中に Midas Touch が生じるように選択した. メニューアイテムの数は 8 であり, 実験 1 の *iDirection* と同様の方向に配置されている. メニューアイテムの色は 8 方向ごと ( $0^\circ, 45^\circ, 90^\circ, 135^\circ, 180^\circ, 225^\circ, 270^\circ, 315^\circ$ ) にそれぞれ 8 色 (緑, 青, 赤, ピンク, 灰, 黄, 橙, 紫) の異なる色が割り当てられている.

*Method* はラテン方格法を考慮した順番で提示された. 実験参加者は *Method* ごとに  $DT \times 2$  セッション  $\times 16$  試行 (実験 1 の 8 方向  $\times 2$ ) の選択タスクを行った. *DT* の提示順はラテン方格法に準じている. ゆえに実験参加者は  $3 \text{ Method} \times 3 \text{ DT} \times 2 \text{ セッション} \times 16 \text{ 試行} = 288$  回のメニューアイテム選択を行った. また実験参加者は 18 名であるため, 合計 5,184 データが収集された.

従属変数は試行時間とエラー率である. 間違っただターゲットパネルかメニューアイテムが選択された場合, エラーと扱った. 実験参加者は *Method* ごとに, ユーザビリティと精神的負荷を計測することを目的とした System Usability Scale (SUS) [17] と NASA Task Load Index (NASA-TLX) [54] のアンケートに回答した. さらに実験参加者は *Method* と *DT* の好みのランキングを回答した.

#### 4.4.4 実験手順

実験開始前に使用機器 (実験 1 と共通) や実験監督者の手指を消毒した. 実験参加者が実験室に到着後, 椅子に座らせ, 実験の概要およびタスクの内容を説明した. その後 HMD

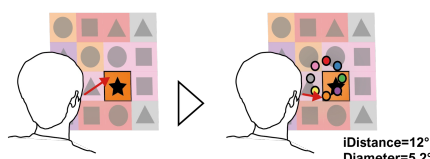

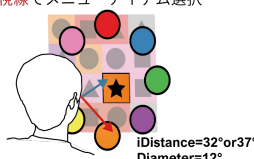
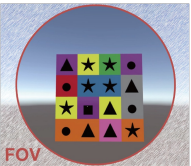
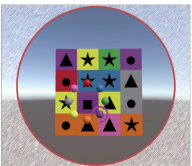
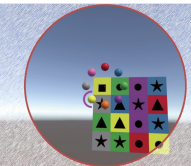
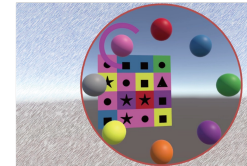
	Two-step Selection (2SS)	Head-Gaze (HG)	Head-Gaze with Kuiper Belt (KB)	
Method	1. 視線でパネル選択    2. 視線でメニューアイテム選択  <i>iDistance=12°</i> <i>Diameter=5.2°</i>	頭部方向でパネル選択し、 視線でメニューアイテム選択  <i>iDistance=12°</i> <i>Diameter=5.2°</i>	頭部方向でパネル選択し、 視線でメニューアイテム選択  <i>iDistance=32°or37°</i> <i>Diameter=12°</i>	
Camera View	 FOV			

図 4.11: 3つの手法の概要と、対応する FOV. カメラビューの赤い円が FOV を表している.

を装着させ、HTC VIVE Pro Eye で提供されているアイトラッカの5点キャリブレーションを行った。そして実験参加者が実験タスクの挙動を理解するまで、実験タスクからパラメータを変更したタスクを練習として行わせた。

実験タスクは視覚探索タスクであり、VR環境に存在するパネル(ターゲット)と色(メニューアイテム)を探索し、視線インタフェースで選択するタスクである。VR環境には、実験参加者から2.0 m離れた位置に16枚のパネルで構成されたプレートが存在しており、プレートの大きさは縦横 $50.0^\circ$ であった。実験参加者はプレートの中からパネルと、メニューの中からそのパネルと同じ色を選択することが求められた。パネルは中央に記号(円、四角、三角、星のいずれか)が描かれており、メニューアイテムの色からランダムに選ばれた色をしていた。ターゲットパネルは他のパネルに描かれていない記号を有しており、その他の3つの記号は残りの15枚のパネルに均等に割り振られていた。本実験のタスクはプレートの中で唯一描かれた記号を有したパネルの選択と、3D環境に配置された8つのメニューアイテムから対象パネルと同じ色のメニューアイテムを選択することである。2SSのメニューはパネル選択後に表示され、HGとKBのメニューは常に表示されていた。メニューはHMDから1.8 mの位置に配置されていた。選択が行われるとプレートは再生成された。ただし間違った選択が行われると500 msのポーズが生じ、その後プレートが再生成された。さらに、選択の正誤に応じた音声フィードバックが与えられた。また、プレート再生成に伴う状況認知時に選択が行われないう、プレート生成後の200 msは滞留時間に含めなかった。現在選択しているパネルはハイライトされ、滞留時間のフィードバックとして実験1と同様の円形スライダーがメニューアイテムの周囲に描かれた。HGかKBを使用している場合は頭部方向を示す頭部カーソルが、2SSを使用している場合は視線カーソルが表示された。これらのカーソルは直径 $1.0^\circ$ である。視線データは

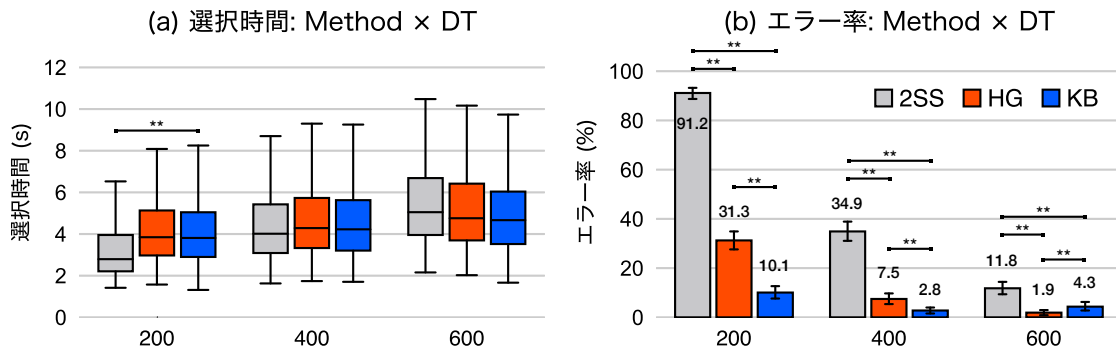


図 4.12: *Method* × *DT* ごとの (a) 選択時間, (b) エラー率 (エラーバーは 95%信頼区間). 有意差は\*\* $p < .01$ .

Outlier filter [95] with a triangular kernel filter [82] によって平滑化された.

実験参加者は全部で 18 セッション行った. 1 セッションは 16 試行 (8 実験 1 の *iDirection* × 2) で構成されている. つまり実験参加者は *Method* × *DT* の組み合わせを 2 回経験する. 実験参加者は *Method* ごとに 3 つの *DT* を経験し, その後 SUS と NASA-TLX, 滞留時間の好みのアンケートに回答した. この手順を 3 つの *Method* に対して繰り返す. 18 セッション完了した後, 手法の好みのアンケートとインタビューを行った. 実験時間は約 60 分であった.

#### 4.4.5 結果

それぞれの従属変数 (試行時間, エラー率) に対してノンパラメトリックな分散分析手法である整列ランク変換 (ART) [58, 166, 211] を行い, 混合モデル REML を用いてデータを評価した. 多重比較にはウィルコクソンの符号順位検定を用い, Holm 法 [62] による補正を用いた. SUS と NASA-TLX によって得られた値はフリードマン検定とウィルコクソンの符号順位検定を用い, Holm 法による補正を用いた.

#### 選択時間

手法 (*Method*) と滞留時間 (*DT*) を独立変数とし, 選択時間を従属変数として分析を行った. 選択時間に対して *Method* ( $F_{2,4032.4} = 9.70, p < .01$ ), *DT* ( $F_{2,4032.5} = 75.04, p < .01$ ) に有意な差が確認された. さらに 1 次の交互作用として *Method* × *DT* ( $F_{4,4032.0} = 11.61, p < .01$ ) が確認された. 図 4.12a に *Method* × *DT* の選択時間を示す.

## エラー率

手法(*Method*)と滞留時間(*DT*)を独立変数とし、エラー率を従属変数として分析を行った。エラー率に対して *Method* ( $F_{2,5158} = 9.70, p < .01$ ), *DT* ( $F_{2,5158} = 75.04, p < .01$ ) に有意な差が確認された。さらに1次の交互作用として *Method*×*DT* ( $F_{4,5158} = 571.56, p < .01$ ) が確認された。図 4.12b に *Method*×*DT* のエラー率を示す。

## 2SS の Midas Touch

2SS のエラーには、パネル選択ミス (視覚探索中の Midas Touch) とメニューアイテム選択ミス (メニューアイテム選択時の Midas Touch) の2つがある。メニューアイテム選択時のエラー率は、*DT* = 200 ms のとき 1.39 %, *DT* = 400 ms のとき 0.52 %, *DT* = 600 ms のとき 0.17 %であった。これらのエラー率は2SSの全体のエラー率と比べると非常に小さい。ゆえに、2SSの大半のエラーは視覚探索中の Midas Touch であると判断できる。

## HG と KB の Midas Touch

HG と KB も同様に2種類のエラーが存在するが、2SSとは異なる分類となる。実験2は視覚探索によって正しいパネルを探すことである。ゆえに、間違っただけのパネルを正しいと勘違いしたまま選択するエラーが存在する。このエラーは意図にそぐわない選択ではないため、Midas Touch ではない。そのため間違っただけのパネルを選択し、かつそのパネルと同じ色のメニューアイテムを選択した場合のエラーを視覚探索エラー、それ以外のエラーを Midas Touch と定義した。HG と KB の Midas Touch の定義は2SSには適用できない。なぜなら2SSは視覚探索時のエラーをHGとKBのように識別することが難しいからである。図 4.13 にHGとKBの視覚探索エラーと Midas Touch 率を *DT* ごとに示す。

## System Usability Scale (SUS)

2SS, HG, KB の SUS スコアの中央値はそれぞれ 50.00, 73.75, 75.00 であった (高いほど良い)。フリードマン検定より、*Method* ( $\chi^2_{2,N=18} = 17.77, p < .01$ ) に有意差が確認された。また事後検定より、2SSとHG ( $Z = -3.34, p < .01$ ), 2SSとKB ( $Z = -3.60, p < .01$ ) に有意差が確認された (図 4.14a)。

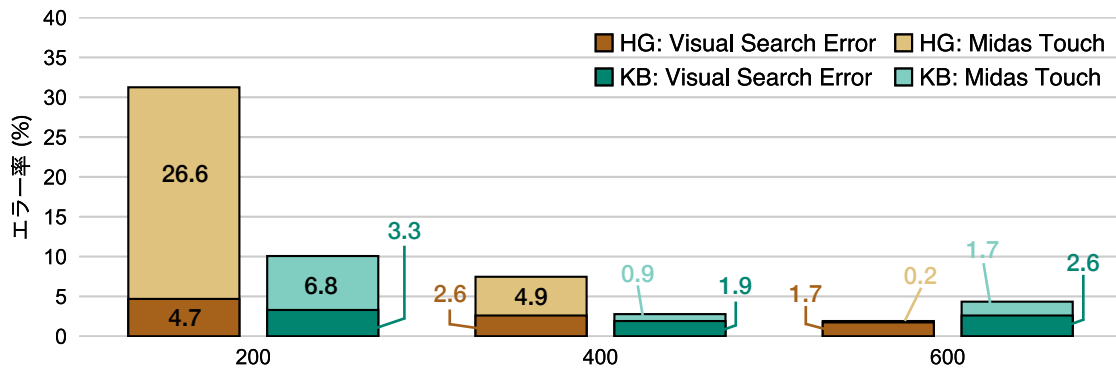


図 4.13: HG と KB の DT ごとの Visual Search Error (視覚探索エラー) と Midas Touch の割合。

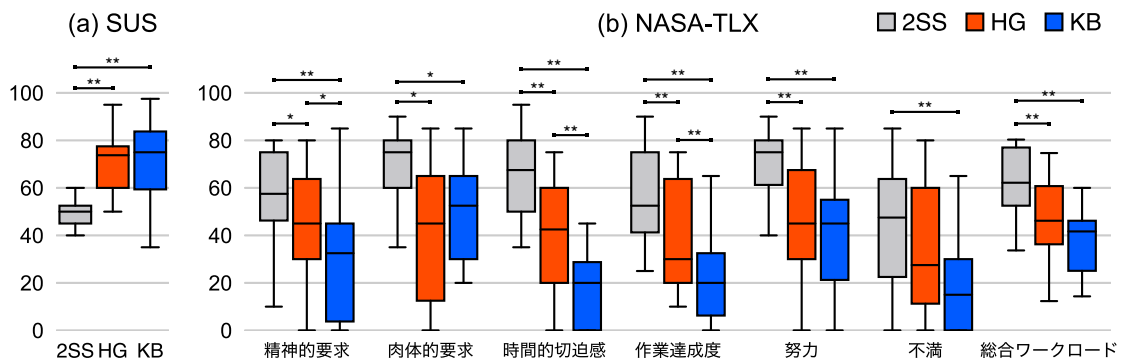


図 4.14: Method ごとの (a) SUS, (b) NASA-TLX の結果. 有意差は\* $p < .05$ , \*\* $p < .01$ .

### NASA Task Load Index (NASA-TLX)

2SS, HG, KB の総合ワークロードスコアの中央値はそれぞれ 62.17, 46.17, 41.67 であった (低いほど良い). NASA-TLX によって得られた 6 つの項目に対するスコアと総合ワークロードスコアに対してフリーマン検定を用いて分析した結果, Method に対して精神的要求 ( $\chi^2_{2,N=18} = 16.60, p < .01$ ), 時間的切迫感 ( $\chi^2_{2,N=18} = 23.43, p < .01$ ), 作業達成度 ( $\chi^2_{2,N=18} = 23.91, p < .01$ ), 努力 ( $\chi^2_{2,N=18} = 10.64, p < .01$ ), 不満 ( $\chi^2_{2,N=18} = 9.76, p < .01$ ), 総合ワークロード ( $\chi^2_{2,N=18} = 12.76, p < .01$ ) に有意差が確認された. また事後検定より総合ワークロードに対して 2SS と HG ( $Z = -2.85, p < .01$ ), 2SS と KB ( $Z = -3.34, p < .01$ ) に有意差が確認された (図 4.14b).

### 実験参加者の各手法 Method の好み

図 4.15a に Method の好みを示す. 図より KB, HG, 2SS の順に好まれた. KB は 9 人, HG は 5 人, 2SS は 4 人の実験参加者に最も好まれた. KB, HG, 2SS の平均ランクはそ



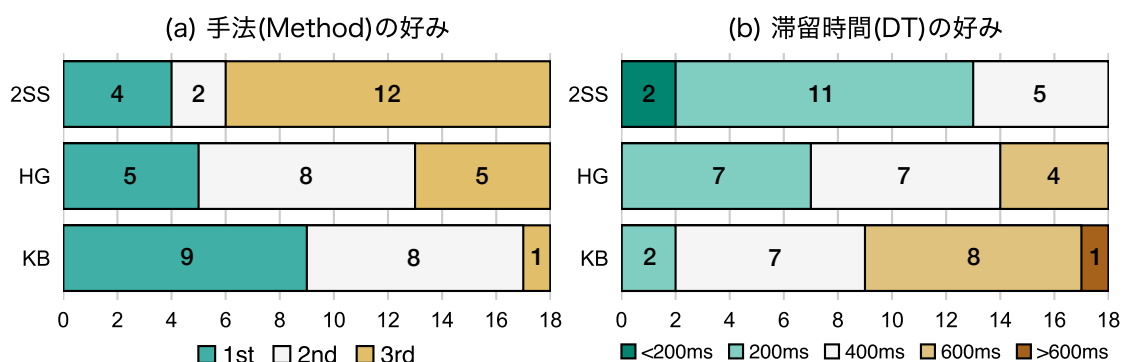


図 4.15: (a) 手法 *Method*, (b) 滞留時間 *DT* の好み.

それぞれ 1.44, 2.00, 2.44 であった.

KB は最も好まれた手法であり, 17 人の実験参加者が KB を 1 位か 2 位に選んだ. P13 は「頭の向きとメニューアイテムに物理的な距離があり, パネル選択とメニューアイテム選択の間に時間的猶予があったので, 滞留時間に関わらず時間的圧迫感は一切なかった」とコメントした. 一方で, Kuiper Belt における色認識が比較的難しいことを実験参加者は報告した. P10 は「メニューアイテムが視野の端にあると色が分かり辛いことがあったので, その点は少し気になった」とコメントした.

HG は二番目に好まれた手法であり, 13 人の実験参加者が HG を 1 位か 2 位に選んだ. P13 は「HG が, 思考時間の確保とメニューアイテム選択のやりやすさのバランスが一番良かったと思った」とコメントした. 一方で, 実験参加者はメニューアイテムが中心視野付近にあることへの不満を報告した. P10 は「メニューアイテムがほぼ中心付近にあったので対象を探すのに若干探しづらいと感じた」, P14 は「選択ミスを防ぐために, 視線をメニューアイテムから逸らすように心がけた」とコメントした.

2SS は最も好まれなかった手法であり, 12 人の実験参加者が 2SS を 3 位に選んだ. この順位は, 短い滞留時間の時の高いエラー率が原因である. P17 は「滞留時間が短いと, パネルを認識する前に選択がされてしまう」とコメントした. 一方で頭の動きがないという理由から, 何人かの実験参加者は 2SS を最も好んだ手法に挙げた. P9 は「視線だけで選択するのは身体的に楽であると感じたから一番よいと思った」とコメントした.

#### 実験参加者の各滞留時間 *DT* の好み

図 4.15b に *Method* ごとの *DT* の好みを示す. 図より, 2SS で最も好まれている滞留時間は 600 ms, HG は 400 ms か 600 ms, KB は 200 ms か 400 ms であった.

KB において 200 ms かそれ以下の滞留時間は, 一般的な視線インタフェースでは短すぎる滞留時間であるにも関わらず, 9 人の実験参加者に好まれた. P12 は「目標を探すと

きに意図しない入力をしてしまう心配がほとんどないので滞留時間が短くても良いのかなと感じました」とコメントした。

HGにおいて400 msか600 msの滞留時間が14人の実験参加者に好まれた。2SSと異なり、実験参加者は比較的短い滞留時間(400 ms)を好んだ。P18は「600 msは長すぎるようにも感じた」とコメントした。また実験参加者は短すぎる滞留時間をあまり好まなかった。P6は「短い滞留時間だと目のやり場に困った。ひたすら目線を動かすか真ん中を見つめ、周りを見たりした」とコメントした。

2SSにおいて600 msかそれ以上の滞留時間が13人の実験参加者に好まれた。実験参加者は短すぎる滞留時間に不満を感じていた。P13は「最長の滞留時間でないと、思考時間が強制的に排除されてしまって、使いにくい」とコメントした。

#### 4.4.6 実験結果のまとめ

KBは視覚探索タスクにおいてエラー(Midas Touch)が最も少ない手法であった。2SSは最も試行時間の短い手法であったが、エラー率は非常に大きかった(91.15% @200 ms, 34.90% @400 ms)。HGは $DT = 400$  ms, 600 msのときは正確であったが、 $DT = 200$  msのときのエラー率は大きかった(31.25%)。また、KB, HG, 2SSの順でユーザビリティや精神的負荷の平均スコアが良く、KBとHGは2SSよりも有意にユーザビリティや精神的負荷の面で優れていた。全体として、Kuiper Beltを用いた手法(KB)は実際の使用に際した改良が必要であるが、研究用のプロトタイプとしては十分に低いエラー率だった。したがってKuiper Beltは誤入力(Midas Touch)を減らすことができたと言える。

KBは最も正確な手法であり、他の手法よりも時間的切迫感の少ない手法であった。KBは2SSよりもエラー率、ユーザビリティ、精神的負荷の点で優れていた。加えて、KBはHGよりも $DT = 200$  msのときのエラー率が有意に低く、NASA-TLX項目のうち精神的要求、時間的切迫感、作業達成度の項目が有意に優れていた。ゆえにKBは視覚探索タスクにおいて他の2手法よりも快適に用いることができる手法だといえる。HGは2SSよりもエラー率、ユーザビリティ、精神的負荷の点で優れていた。さらに、HGは $DT = 400$  msまではMidas Touchを減らすことが可能であった。ゆえにHGは2SSよりも視覚探索タスクに適した手法だといえる。最後に2SSのエラー率は非常に高く、600 msを超える滞留時間が必要であった。これは高難度の視覚探索タスクでは必要な滞留時間が長くなるという以前の研究結果と一致している[233]。滞留時間を長くして誤選択を減らすと選択時間が長くなるため、2SSは視覚探索タスクよりも短くて単純な操作に適しているといえる。

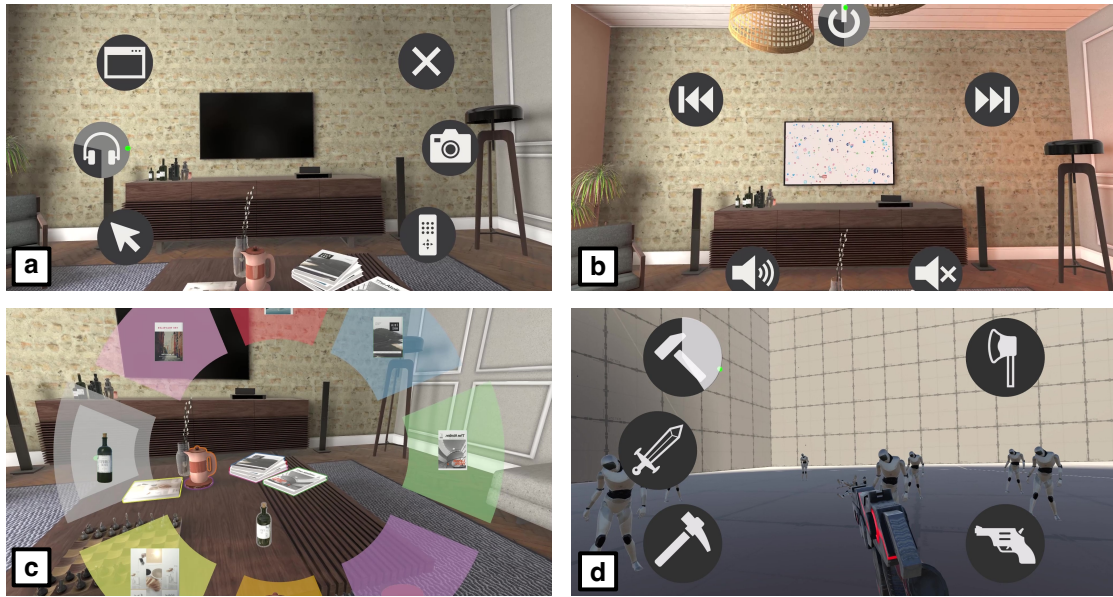


図 4.16: Kuiper Belt を用いたアプリケーションの例示. 緑色の点がユーザの視点. (a) メニュー, (b) TV を消そうとしている様子, (c) 瓶を選択しようとしている様子, (d) 銃から斧へ把持オブジェクトを変更しようとしている様子.

## 4.5 応用アプリケーション例

本章では, 2つの実験結果を受けて開発したアプリケーション例を4つ示す.

### 4.5.1 メニュー選択

Kuiper Belt を主にメニューアイテム選択手法として活用することができる. アプリケーション例として即時起動可能なメニューアプリケーションを作成した(図 4.16a). 本アプリケーション例では, ユーザの視線が自然な視線角度領域に存在するときは, メニューアイテムは視界に干渉しないよう FOV の外側に配置される. ユーザの視線が Kuiper Belt に入ると, メニューアイテムは FOV 外から Kuiper Belt 内に移動する. この設計は Lu らの Gaze-Summon [107] を参考にした. メニューアイテムの配置は頻繁に変更されることはないため, ユーザはアイコンの場所や意味を容易に覚えることができ, Kuiper Belt 内を視覚探索せずにメニューアイテムを選択することが出来ると考えている.

### 4.5.2 電子機器操作インタフェース

二つ目の例は VR 内での電子機器のリモート操作アプリケーション(図 4.16b)である. 操作方法は実験 2 の KB 条件と同様で, 頭部方向で対象の電子機器を決定し, 操作に対応

したメニューアイテムを視線で選択する。メニューアイテムは Midas Touch を減少させるために Kuiper Belt に配置されている。現状このアプリは VR で実装されているが原理的には MR での実装も可能であるため、IoT の操作などにも有用だと考えられる。

#### 4.5.3 密で小さなオブジェクトの選択手法

三つ目の例は小さなオブジェクトが密集した状況から希望するオブジェクトを選択する手法である (図 4.16c)。三次元環境において、他のオブジェクトによって遮蔽されて可視領域が小さくなったオブジェクトを選択することは難しい。この問題を解決するために、ActiGaze [109] を VR に拡張する手法を作成した。ActiGaze は GUI (2D) を対象とした手法であり、視線のみでオブジェクトの密な集合から小さいオブジェクトを選択する手法である。この手法では、まずユーザがターゲットを含むオブジェクトの集合を見ると、それぞれのオブジェクトの輪郭に異なる色が描画される。それらの色は Kuiper Belt に配置されたメニューアイテム上の色である。その後、視線で選択したいターゲットの輪郭と同じ色を視線で選択することで、ターゲットを選択する。この手法によって、視線のみでは選択が難しいオブジェクトの選択を容易にしている。

#### 4.5.4 VR ゲームでのアイテム選択

Kuiper Belt は VR ゲームにも活用することが出来る。例えばゲーム中に、視線インタフェースのみで把持しているアイテムを変更することが考えられる (図 4.16d)。視線インタフェースは他のインタフェースと同時に使うことができるため、ユーザがコントローラを手を持っていても、コントローラの操作と競合することなく使用することができる。Kuiper Belt を用いたインタフェースは他の視線インタフェースと同様に、ユーザがメニューアイテムを見るだけで選択するものである。このような自然なユーザの振る舞いに準じているため、コントローラやジェスチャインタフェースと容易に組み合わせることが可能だと考えられる。

### 4.6 議論

本研究では Kuiper Belt にメニューアイテムを配置する手法 (KB) で Midas Touch 問題を減らすことが出来るかどうか調査するために、2つの実験を行った。以下に実施した2つの実験についてまとめる。

- 実験1: Kuiper Belt のどこにメニューアイテムを配置すべきか調査した。その結果、上方向は  $iDistance = 32^\circ, 37^\circ$  が、それ以外の方向は  $iDistance = 32^\circ, 37^\circ, 42^\circ$  が

好ましいことが明らかとなった。

- 実験2: KBは最も正確かつ短い滞留時間(400 ms)でも機能する手法であった。KBの精神的負荷は三つの手法の中で最も低く、特に精神的要求、時間的切迫感、作業達成度が有意に低かった。Kuiper Beltを用いたメニュー選択手法は、通常の視線領域における視線インタフェースと同様の精神的負荷レベルだといえる。

実験2のエラー率より、KBは視覚探索中のMidas Touchを減らすことが出来ることが示された。一方で、HGよりもKBの方が $DT = 600$  msのエラー率が高かった。集められたデータを実験参加者ごとに精査したところ、1人の実験参加者が最初のMethodであったKBで頻繁にミスをしていたことが分かった。この実験参加者がタスクを正しく理解していたかどうかは定かではないが、これを排除すべき異常値とはせず、データセットに残した。他の17人の実験参加者はKBを容易に使用することが出来た。以上の結果から、KBはMidas Touch問題を解決するための有望な手法であるといえる。

またKBにおける肉体的要求やその他の負荷指標は通常の視線領域のインタフェースと同様の傾向を示した。よってKuiper Beltを用いたメニュー選択手法は視覚探索タスクに適していると考えられる。しかし、Midas Touch問題を解決する唯一の方法がKuiper Beltであるとは考えていない。Kuiper Beltは視線のインタラクションにおいて自然でない視線角度領域を利用しているため、他のインタラクション技術との連携が可能である。ゆえに、Kuiper Beltは他のインタラクション技術を補完するものである。

#### 4.6.1 Kuiper Beltの長所・短所・適応性

本研究では視覚探索タスクにおけるKuiper Beltの有用性を調査したが、アプリケーション例で示したようにKuiper Beltは他のVRタスクにも使用可能である。Kuiper Beltはメニュー選択に用いることが出来る(図4.16a)ほか、オブジェクトとメニューアイテムの同時選択(図4.16b)、通常の視線インタフェースで選択することが難しいような小さくて密集したオブジェクトの選択(図4.16c)などに用いることが出来る。一方で、Kuiper Beltは全てのVRタスクに適した手法ではない。まず地図アプリケーションなどにおける拡大・縮小操作といった、インタラクション領域の変化の確認が重要なタスクにおいては、インタラクション領域とKuiper Beltを頻繁に視線移動させる必要があるためKuiper Beltの使用は適していない。またKuiper Belt内では正確な視線移動が難しくメニュー選択のみが可能であるため、色変更やオブジェクトサイズなどのスライダー操作を前提とした連続値入力も難しい。

また、本研究における実験はシンプルで動きのない背景環境で行われたため、複雑な環境においてKuiper Beltのパフォーマンスは低下する可能性がある。例えばKuiper Belt

のメニューアイテムの色情報が環境に溶け込んでしまうことや、メニューアイテム深度と環境オブジェクト深度が一致することによるメニューアイテムの誤認などが発生する可能性がある。

#### 4.6.2 Kuiper Belt 使用時のユーザ負荷

実験 2 では「極端な視線角度」に対する視線移動に起因するユーザ負荷を調査した。Kuiper Belt を用いた手法がユーザに負担をかける要因として、極端な視線角度への視線移動が考えられる。そこで極端な視線角度 (KB) と一般的な視線角度 (HG) の領域にメニューアイテムを配置した手法を比較することで、極端な視線角度がどの程度ユーザに負担をかけるのかを調査した。実験 2 の結果、NASA-TLX の肉体的要求では、HG と KB のスコアが 2SS のスコアよりも有意に優れており、HG と KB には有意な差がなかった。したがって、Kuiper Belt による肉体的疲労は短期的な使用には問題ないといえる。しかし本研究では Kuiper Belt 手法を長期的に使用した際のユーザ負荷は調査していない。Sidenmark らは平均的なユーザの視線領域範囲外に UI 要素を配置することは可能であるが、ユーザに長期的な負担を与える可能性があると指摘した [176]。実験 2 より、ユーザが Kuiper Belt 領域を使用した視線インタフェースを約 15 分間 (実験 2 の 96 タスク) 使用できることを確認したが、さらに長時間使用した場合の影響については調査していない。ゆえに長時間の Kuiper Belt 領域の使用がユーザに与える影響を調査するには更なる研究が必要だといえる。

#### 4.6.3 視線インタフェースにおけるメニューアイテムの配置すべき場所

実験 2 の結果より、HG と KB は視覚探索タスクに適した手法であることが明らかになった。しかし、HG を使用する特定のユーザは、誤選択を考慮し不自然な視覚的検索行動を行っていた (P12: 「中心以外を見ると誤ってメニューアイテム選択がされてしまう可能性があったので、出来るだけ中心を見ながら首と体を動かすようにした」)。これは仮想コンテンツ (実験 2 においてはメニューとそのアイテム) が、3D 環境の他のオブジェクトを遮ったことに起因していると考えられる。そのため、メニューアイテムを中心視野に配置することは推奨されない [101]。ゆえに、HG は短期的な操作に適した手法であり、常に表示されるインタフェースには適していない可能性がある。

対照的に KB は中心視野外にメニューアイテムが配置されているため、ユーザが見たいと思っている 3D 環境の情報やオブジェクトを遮ることはない。ゆえに、中心視野内にメニューを表示する場合に比べてメニューを長時間表示しやすい。また Gaze-Summon [107] のように、視界の外からメニューが入ってくるようにメニューを表示することもできる。

このように、視線インタラクションにおける「自然ではない視線角度の領域」の新しい使い方を示したことは、今後の視線インタフェースやインタラクションの研究において有望な知見だといえる。

#### 4.6.4 Kuiper Beltに残されたMidas Touch問題

本実験より、Kuiper Beltは視覚探索中のMidas Touchを減らすことが出来ることが明らかになった。しかし、これでMidas Touch問題が完全に解決したわけではない。例えばKuiper Belt内でメニューアイテムを探索する場合、視界探索領域とインタラクション領域が一致しMidas Touchが生じうる。このMidas Touch問題を軽減するためには、Kuiper Beltに配置するメニューやアイテムのデザインを慎重に行う必要がある。例えばユーザーがよく知っているメニューアイテムだけを配置することで、ユーザーはKuiper Belt上のメニューアイテムを探ることなく選択を完了することができる。そのため、ショートカットのように繰り返し使うアイテムをKuiper Beltに配置することが好ましい。

#### 4.6.5 提案手法の制限

本研究で実装したKuiper Beltは研究用のプロトタイプであり、実用面でいくつか制限がある。まず眼鏡着用者はKuiper Belt領域を使用できない。なぜならKuiper Belt領域を見つめる際の視線は眼鏡のフレームに阻まれるため、視線計測が難しいからである。したがって現状のKuiper Belt手法の利用は、コンタクトレンズを除いた視覚補助を必要としないユーザーに制限される。しかし眼鏡の内側にアイトラッカを配置した場合この問題は解消されるため、スマートグラスにアイトラッカが搭載されることで眼鏡着用者でもKuiper Beltを用いることが出来ると考えられる。第二にKuiper Beltは汎用的な設計原理ではなく、全ての視線インタフェースに使用することはできない。例えばアイタイピング [119] は、Kuiper Beltに配置出来るアイテム数に限りがあるため、アイタイピングに必要なだけのメニューアイテムをKuiper Beltに全て配置することは難しい。

### 4.7 本章のまとめ

本章ではVR環境での視線入力においてKuiper Beltと呼ばれる「極端な視線角度の領域」にどのようにメニューアイテムを配置するべきか、という設計原理を検討した。本研究の目的はKuiper Belt領域を用いることで視線入力におけるMidas Touch問題（誤入力）を減少させることができるかどうかを調査することであった。調査にあたってはKuiper Beltを利用したメニューアイテム選択タスクと視覚探索タスクのユーザービリティと作業負

荷を把握するための2つの実験を行った。その結果、Kuiper Belt におけるメニューアイテムを配置すべき位置（実験1）とVR環境でのKuiper Belt を利用したメニュー項目選択の有効性（実験2）が明らかになり、Kuiper Belt が誤入力（Midas touch）を低減できることが示された。実用化に向けては更なる検討が必要であるが、本研究はKuiper Belt 領域を用いたメニュー選択の可能性を示すことができたと考えている。



## 第5章 Asteroid Gazer: VRにおける視線のみを用いた遮蔽されたオブジェクトの選択手法の提案

視線インタフェースが抱える課題の一つに「小さなオブジェクトの選択が難しい」というものがある。第3章ではディスプレイ環境における小さなオブジェクトの選択手法を提案した。ただしディスプレイ環境は二次元環境であり、三次元環境では二次元環境には存在しない小さなオブジェクトが存在する。それは「遮蔽されたオブジェクト」である。三次元環境では奥行きが存在するため、ターゲットの前に遮蔽物が存在する状況が頻出する。遮蔽物が存在する場合ターゲットの可視領域は減少し、ターゲット内に視線を留め続けることは困難となる。そして完全にターゲットが遮蔽されると、第3章で提案した手法では全く選択できなくなる。一般的な遮蔽物の視線選択手法はターゲット候補の再配置操作などで遮蔽物を除去して遮蔽された状況を解消するアプローチを行う [28, 203, 105, 154]。一方でこれらの手法は手を用いており、視線インターフェースのメリットである「手を用いなくても良い」という利点を失っている。これらの手法が手を用いている理由は Midas Touch を避けるためである。つまりターゲット候補の再配置時に Midas Touch が生じなければ、手を用いずに視線のみで遮蔽されたオブジェクトを選択可能だと考えられる。

ゆえに本章ではVR環境における遮蔽されたオブジェクトを視線のみで選択する手法として Asteroid Gazer (図5.1) を提案する。Asteroid Gazer は Midas Touch を回避しながら、高密度でかつ小さく、かつ部分的に遮蔽されたオブジェクトを選択することができる手法である。この手法はオブジェクト仮想再配置アプローチ [109, 171, 139] を活用している。仮想再配置アプローチとは、メニューアイテムにオブジェクトをそれぞれ対応させ、メニューアイテムを選択することで対応しているオブジェクトを選択する手法である。つまりこのアプローチを使用する手法はターゲットサイズや遮蔽状況に影響を受けずにターゲットを選択することができる。その上、単純な再配置アプローチとは異なり、仮想再配置アプローチはターゲットを見ても選択が実行されないため Midas Touch が生じない。しかし、仮想再配置アプローチによって Midas Touch を完全に回避するためには、ターゲット領域がメニューアイテムと完全に分離されている必要がある。ゆえに第4章で提案した Kuiper Belt を活用することで、ターゲット領域をメニュー領域を完全に分離することが

できると考えられる。

Asteroid Gazer における仮想再配置可能なオブジェクトはメニューアイテム数と一致する。ゆえにメニューアイテムが多いほど仮想再配置可能なオブジェクト数は増加し、高密度な状況においてもターゲットを仮想再配置対象にする可能性が上がる。一方で第4章より、Kuiper Belt 内に配置できるメニューアイテム数は Kuiper Belt 内の視線推定精度より限定的である [179]。ゆえに Asteroid Gazer ではメニューアイテム数の増加のために階層メニューを Kuiper Belt に配置することを試みる。本手法では階層メニューを段階的に絞り込むようにメニューアイテムを選択することで、ターゲットを段階的に選択することができる。実験では Asteroid Gazer が遮蔽されたオブジェクトを容易に選択できるか検討した。実験の結果、提案手法は密集した状況においても遮蔽されたオブジェクトを一般的な視線選択手法よりも正確に選択することができた。本章の貢献は以下である。

- 仮想再配置アプローチを活用することで、視線のみで遮蔽されたオブジェクトを選択する手法である Asteroid Gazer を提案した。
- ユーザビリティや精神的負荷の観点から Asteroid Gazer を評価する実験を行い、提案手法が密集した状況における遮蔽されたオブジェクトの選択を容易にすることを確認した。
- Asteroid Gazer を用いた3つのアプリケーションを実装し、提案手法の持つ有用性を示した。

## 5.1 関連研究

### 5.1.1 遮蔽されたオブジェクトの選択手法

#### 手のみを持ちいる手法

三次元環境において、レイキャスティング（光線を用いた選択手法）[130]が一般的なオブジェクト選択手法として用いられている。レイキャスティングは手元のコントローラや指先から光線を発射し、トリガーを入力すると光線が当たっているオブジェクトの中で最もレイの発射点に近いオブジェクトを選択する手法である。光線の距離が実質無限大であることからレイキャスティングは遠方のオブジェクトを選択することができるが、本手法は遮蔽されたオブジェクトの選択が難しいという課題を有している。遮蔽されたオブジェクトを選択するためのアプローチは大きく二つに分けられる。一つは直接遮蔽されたターゲットを選択するアプローチ（直接選択アプローチ）、もう一つはターゲットの遮蔽を解消した後にターゲットを選択するアプローチ（遮蔽解消アプローチ）である。直接選択ア

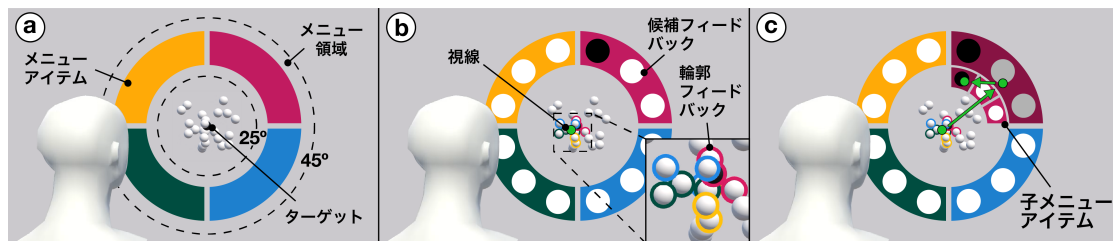


図 5.1: Asteroid Gazer は部分的に遮蔽されたオブジェクトの選択手法であり、ターゲット候補とメニューアイテムを連続的に選択することでターゲットを選択する。(a) メニューアイテムはユーザの頭部から  $25^{\circ}$ – $45^{\circ}$  の位置に固定されている。(b) オブジェクト群を一定時間注視することで、ユーザの視線 (緑の点) に近いオブジェクトがターゲットとなる。ターゲット候補は一層目のメニューアイテムにそれぞれ対応する。フィードバックとして、ターゲット候補の輪郭に対応するメニューアイテムの色が表示され、メニューアイテムの中に対応するターゲット候補の外観が表示される。(c) ユーザがメニューアイテムを注視すると、注視したメニューアイテムの子階層のメニューが表示される。その後、注視したメニューアイテムに対応している複数のターゲット候補が子階層のメニューアイテムにそれぞれ分配される。最後に、ユーザが子階層のメニューアイテムを注視すると、注視したメニューアイテムに対応するターゲット (黒いオブジェクト) が選択される。

プローチでは、ターゲットサイズの拡大や [108], 3D ポイント (レイ上に配置されたマーカ [50, 7], 二本のレイの交点 [215, 228]) を活用した遮蔽物を無視した選択手法が提案されている。

遮蔽解消アプローチでは、一般的にオブジェクト再配置手法が用いられている。この手法はターゲット候補をだまかに選択した後、それらを選択しやすい位置 (グリッド配置 [26, 19, 224], 円形配置 [50, 160]) に再配置する手法である。これらの手法では「再配置するターゲット候補の決定」と、「ターゲット候補からターゲットを選択」の二度の操作が必要となる。Kopper らは再配置手法をターゲット候補が多い状態 (高密度な状況) においてより正確な選択を行う手法として、「ターゲット候補からターゲットを選択」の工程を複数回行う SQUAD を提案した [93]。SQUAD は選択したターゲット候補をクアッドメニュー (4つのアイテムを有するメニュー) にそれぞれ同数ずつ再配置し、メニューアイテムを選択するごとにそのメニューアイテムに再配置されたターゲット候補を新しいクアッドメニューに再配置し続ける。そして再配置されたターゲット候補が一つであるメニューアイテムを選択すると、そのメニューアイテムに再配置されたターゲット候補が選択される。この手法はターゲット候補が膨大であろうと、メニューアイテム選択回数の増加による選択時間の増加とトレードオフとして、ターゲットを正確に選択することができる。SQUAD のイメージを図 5.2 に示す。

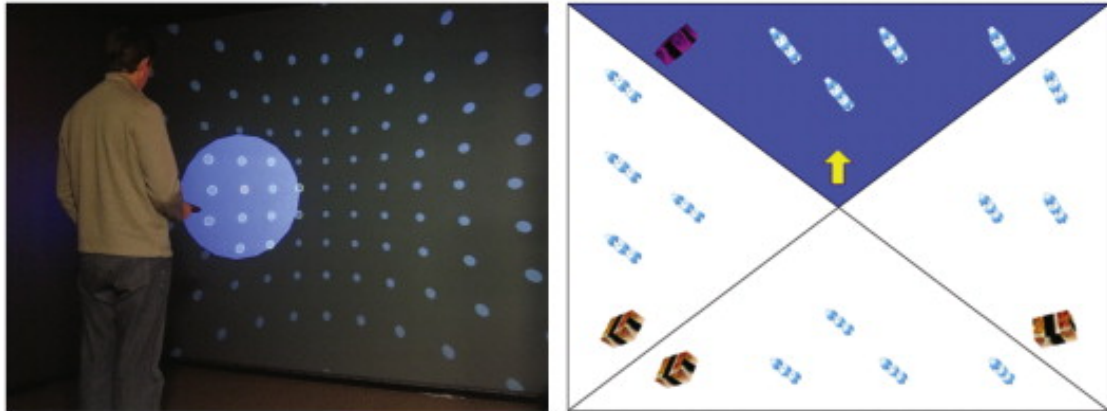


図 5.2: SQUAD のイメージ [93]. 大まかに選択されたターゲット候補（左図）がクアドメニューに再配置される（右図）.

### 視線のみを持ちいる手法

視線による遮蔽されたオブジェクトの選択手法の大半は直接選択アプローチを採用している。視線は右目と左目から照射されているため、これらの交点を 3D ポイントとして三次元的に遮蔽物を無視して選択する手法が提案されている [147, 120]。しかし視線深度（三次元凝視点の奥行き方向の距離）の推定精度は低く、遮蔽されたオブジェクトを正確に選択することは難しい。他の手法として注視データの分布から注視対象を推定する手法 [197, 221] なども提案されているが、これもまた正確に遮蔽されたオブジェクトを選択することは難しい。直接選択アプローチによる視線選択手法が多く提案されている一方で、遮蔽解消アプローチを用いた手法は非常に少ない。Piumsomboon らは、一定時間の注視後に注視点に近いオブジェクトが放射状にゆっくり移動し、移動するターゲットを目で追うことで選択を行う手法 Radial Pursuit を提案した [154]。しかし、この手法は実用的なタスクには向いておらず、実用的なタスクに用いる場合には視覚探索中の Midas Touch を避けるために長い滞留時間（約 2000 ms [148]）を採用する必要がある。図 5.3 に Radial Pursuit のイメージを示す。

### 視線と他のモダリティを組み合わせた手法

視覚探索中の Midas Touch と長い滞留時間を回避するために、視線と手を用いた遮蔽されたオブジェクトの選択手法が提案されている [28, 203, 105, 174, 69, 164]。これらの手法は基本的に再配置手法による遮蔽解消アプローチを採用している [28, 203, 105]。またこれらの手法は手を用いた入力（ハンドジェスチャ [28], コントローラトリガー [203]) によって再配置操作を実行することで、Midas Touch を回避している。しかし、手を用い

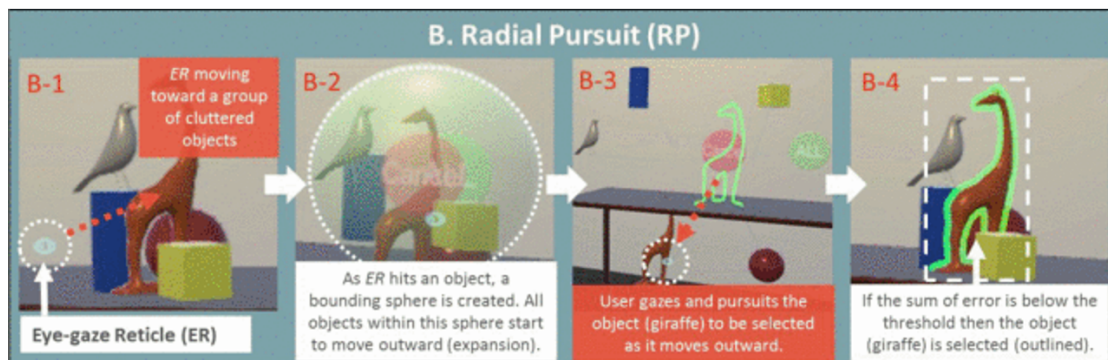


図 5.3: Radial Pursuit のイメージ [154].

これらの手法は視線インタフェースが有する「フリーハンドで入力可能」という優位性を損なっている。

手以外のモダリティを視線と組み合わせた研究は少ないが、いくつか存在する [174, 221]. Sidenmark らは視線と頭部ポインティングを組み合わせた手法である Outline Pursuits を提案した [174]. この手法は、まず頭部レイかコントローラレイに近いオブジェクトの輪郭が色づき、その輪郭をなぞるように動く点を目で追う (Smooth Pursuit) ことで選択を行う手法である。Outline Pursuits のうちコントローラレイを使用する手法の遮蔽オブジェクトの選択エラー率は 13 % であり、通常の視線選択手法 (21 %) よりも低いエラー率を達成した。一方で頭部レイを使用する手法のエラー率は 30 % であり、遮蔽されたオブジェクトの選択精度を改善することができなかった。図 5.4 に Outline Pursuits のイメージを示す。Yi らはまぶたの開ける度合いを調整することで遮蔽物を透過する手法である DEEP を提案した [221]. この手法は目を見開く/細めると Visual Depth が増加/減少し、Visual Depth よりも近いオブジェクトは非表示になる。この操作によって遮蔽されたオブジェクトを非表示にし、ターゲットの遮蔽を解消している。DEEP を用いて遮蔽されたオブジェクトの選択実験を行ったところ、DEEP は遮蔽されたオブジェクトを非常に低いエラー率 (< 6.2 %) で選択することが可能であることがわかった。これらの手法は視線以外のモダリティを視線と組み合わせることで遮蔽されたオブジェクトの選択を可能としている。ただし本章では視線のみで遮蔽されたオブジェクトを選択する手法の検討を目的としているため、提案手法とこれらの手法の比較は行わなかった。

### 5.1.2 仮想再配置アプローチ

情報を探索する行為 (視覚探索) を行うとオブジェクト内の情報にユーザの視線は固定され、その結果 Midas Touch が生じる。つまり、選択対象たるオブジェクトと、オブジェクト内の情報を分離することで Midas Touch を回避することができる [146].

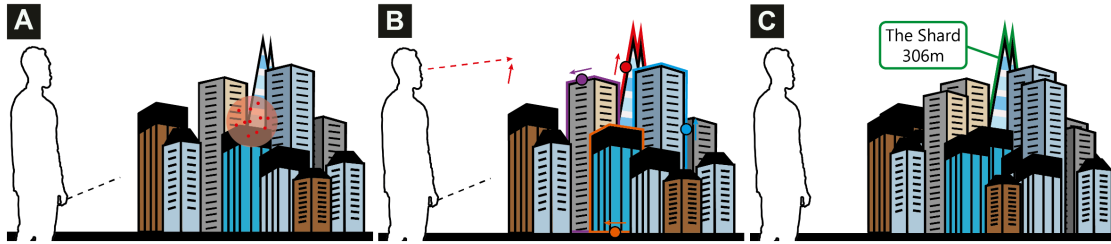


図 5.4: Outline Pursuits のイメージ [174].

ターゲットとラベルの分離アプローチを使用した手法に ActiGaze [109] がある (図 5.5). ActiGaze は視線のみでは選択が難しいほど小さいオブジェクトであるハイパーリンクを、視線のみで選択する手法である. この手法はハイパーリンクテキストとハイパーリンク選択ボタンを分離し、ボタンを GUI の外に配置することでターゲットとラベルを分離する. そして以下の手順で選択を実行する. (1) ユーザがオブジェクト (ハイパーリンクテキスト) を 80 ms 注視すると、注視点に近いオブジェクトがメニューアイテム (ハイパーリンク選択ボタン) と同じ色にそれぞれ色付く. (2) メニューアイテムを 200 ms 注視すると、注視したメニューアイテムと同じ色で色づいたオブジェクトが選択される. この手法はオブジェクトとメニューアイテムをそれぞれ対応させ、メニューアイテムを選択することで対応したオブジェクトを選択している. 言い換えると、ActiGaze はオブジェクトを GUI 外のメニューアイテムに「仮想的に」再配置していると言える. 仮想再配置アプローチはそれほど密集していないオブジェクトを、Midas Touch を生じずに短い滞留時間 (200 ms) で選択することができる. ActiGaze は二次元環境を対象とする密集したオブジェクトを仮想再配置アプローチで選択する手法であるが、三次元環境における密集した遮蔽されたオブジェクトの選択に活用することができる可能性がある.

しかし、ActiGaze を三次元環境に実装することは困難である. 二次元 GUI インタラクションでは GUI 内をターゲットが存在する領域、GUI 外をターゲットが存在しない領域と規定し、GUI 外にラベルを配置することでターゲットとラベルを分離することを可能とした. しかし、三次元 VR インタラクションでは全環境にターゲットが存在するため、ターゲットが存在しない領域は存在せず、二次元環境における分離原則をそのまま活用することはできない. Mutasim らは ActiGaze を VR 内に実装した [139] が、選択対象たるオブジェクトは二次元平面に固定されており、三次元環境に対して ActiGaze を適用した手法とは言い難い. したがって ActiGaze を三次元環境に実装するためにはターゲットが存在する領域とメニューアイテムが存在する領域を分離する必要がある. この分離は Kuiper Belt を活用することで達成することが可能だと考えられる.

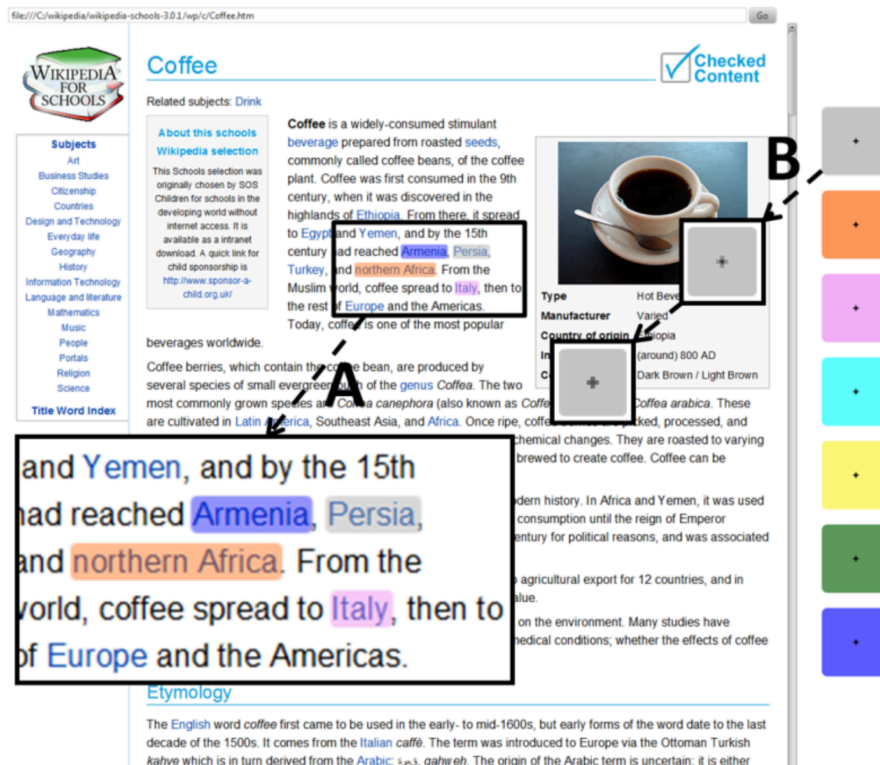


図 5.5: ActiGaze のイメージ [109].

## 5.2 提案手法: Asteroid Gazer

本章では VR 環境における遮蔽されたオブジェクトの選択手法である Asteroid Gazer を提案する。本手法は遮蔽されたオブジェクトをメニューアイテムに「仮想的に再配置」することで遮蔽を取り除き、容易な選択を可能としている。また本手法は仮想再配置アプローチを用いており、直接ターゲットを注視しても選択操作が行われなため、難度が高い視覚探索時においても Midas Touch が生じることはない。仮想再配置アプローチを三次元環境に用いるためにはターゲット領域とメニュー領域を分離する必要がある。ゆえに本手法では Kuiper Belt による領域分離の原理を活用し、ターゲット領域を頭部方向から  $25^\circ$  以内の領域、メニューアイテム領域を  $25^\circ$  から  $45^\circ$  の領域にそれぞれ定義することで領域の分離を行った。

Asteroid Gazer はオブジェクト群からターゲット候補を選択した後に、メニューアイテムに仮想再配置されたターゲット候補の中からターゲットを選択する手法である。選択可能なターゲット候補の数は Kuiper Belt に配置されたメニューアイテムの数と一致する。したがって選択されたターゲット候補の中にターゲットが含まれている可能性を上げるためには、メニューアイテムをできる限り多くするべきだと言える。しかし、頭部中心から

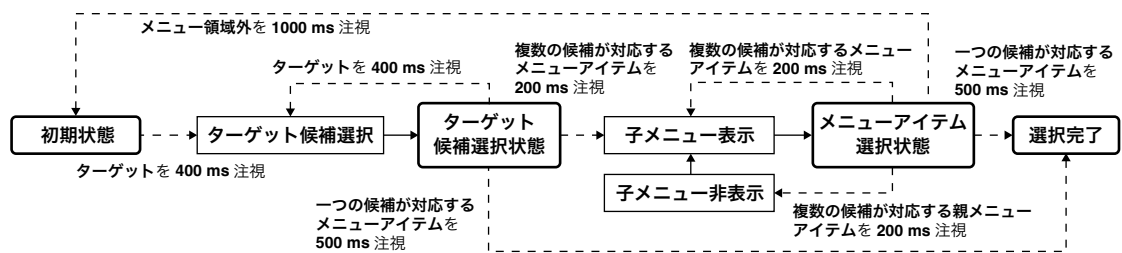


図 5.6: Asteroid Gazer の選択手順の遷移図.

離れれば離れるほどアイトラッカの精度は落ちるため、Kuiper Belt に配置することができるメニューアイテム数は限られている [179]. Yi らは実用性を考慮した際の周辺領域に配置できるアイテムの最大数は 8 であることを明らかにした [220]. ゆえに提案手法では Kuiper Belt に配置するメニューを階層メニューとすることで実質的なメニューアイテム数を増やし、仮想再配置可能なオブジェクト数を増やした.

Asteroid Gazer における主要な選択手順は以下である (図 5.6).

1. ターゲット候補選択: ターゲットが存在する領域を一定時間注視すると、注視点に近いオブジェクト群がターゲット候補となる.
2. メニューアイテムに対するターゲット候補の仮想再配置: ターゲット候補がメニューアイテムに対して仮想的に再配置, つまりターゲット候補とメニューアイテムがそれぞれ対応する.
3. メニューアイテム選択によるターゲット選択: メニューアイテムを一定時間注視すると、注視対象のメニューアイテムが選択される. 選択されたメニューアイテムに対応している選択候補の数に応じて、以下の操作を行う.
  - (a) メニューアイテムに対応するターゲット候補が複数存在する場合、子メニューが展開され、子メニューのメニューアイテムこれらのターゲット候補が分配される.
  - (b) メニューアイテムに対応するターゲット候補が一つだけ存在する場合、メニューアイテムに対応しているターゲット候補が選択される.
4. 単一のターゲット候補が対応したメニューアイテムを選択するまで手順 (3) を繰り返す.

本章における Asteroid Gazer のパラメータは以下を使用した. Asteroid-Gazer-1 (AG1): メニュー層数は 1 であり、メニューアイテム数は 8 である. Asteroid-Gazer-3 (AG3): メ



ニュー層数は3であり、一層目のメニューのアイテム数は8、二層目と三層目は3である。Asteroid Gazerの具体的な選択手順と、AG1とAG3の場合における具体例を以下に示す。

### 5.2.1 ターゲット候補選択

まず、ユーザはオブジェクト群を注視することでターゲット候補を選択する必要がある。ユーザの注視が検出されると、ユーザの視線レイから $2.5^\circ$ 以内のオブジェクトがターゲット候補として選択される(図5.7a)。その後、視線レイとターゲット候補の距離がそれぞれ計算された。この視線レイとターゲット候補の距離は、視線方向ベクトルとターゲット候補ベクトルが為す角度である。ターゲット候補ベクトルの始点は視線ベクトルの始点(両目の中点)である。距離の計算に使用される視線レイはターゲット候補選択時の注視時の視線データの平均ベクトルである。ターゲット候補として選択されたオブジェクトの数がターゲット候補の最大数を超過している場合、視線レイとの距離が近いオブジェクトが順にターゲット候補として選択された。ターゲット候補の最大数は最も深いメニュー層のすべてのメニューアイテム数と一致する。つまり、AG1のターゲット候補最大数は8(一層目のメニューアイテム数)、AG3は72(8一層目のメニューアイテム数 $\times$ 3二層目のメニューアイテム数 $\times$ 3三層目のメニューアイテム数)となる。ユーザの注視を検出するアルゴリズムは、デスクトップ用の注視検出アルゴリズムであるI-DT(Dispersion Threshold Identification)[167]をVR環境用に拡張したアルゴリズム[106]を使用した。アルゴリズムの角度閾値は $1.5^\circ$ 、ウィンドウサイズ(滞留時間)は400msとした。なお、この注視検出アルゴリズムは頭部方向から $25^\circ$ 以内における視線データのみを対象として検出を行った。つまり、ターゲット領域に対する注視のみを対象とし、メニュー領域(頭部方向から $25^\circ$ 以上)に視線が存在する時はターゲット候補の選択を行わなかった。

### 5.2.2 メニューアイテムに対するターゲット候補の仮想再配置

ターゲット候補が選択されると、ターゲット候補がそれぞれメニューアイテムに仮想的に再配置された(ターゲット候補とメニューアイテムが対応する)。一般的に、視線に近いターゲット候補ほど高速かつ容易に選択することが可能であるべきだと考えられる。ゆえに上記の目的を達成する用にターゲット候補の対応規則を設定した。まず、第4章より、上方向よりも下方向に配置されたメニューアイテムの方が容易に選択することが可能である。ゆえに、ターゲット候補とメニューアイテムの対応は右下のメニューアイテムから時計回りに行う。次に、メニューアイテムに対応するターゲット候補が少なければ少ないほど、そのメニューアイテムに対応したターゲット候補を高速に選択することができる。ゆえに、ターゲット候補の対応がメニューアイテムを一周した時、次からの対応は最後に対

応を行ったメニューアイテムから逆回りで一周するまで対応を行う。この操作をターゲット候補すべての対応が終了するまで行う。図 5.7b にターゲット候補とメニューアイテムの対応の流れの例を示す。

この仮想再配置は注視が検出される度に行われる。そのため、ターゲット候補とメニューアイテムの対応関係が頻繁に変わる。これによって、選択しようとしていたメニューアイテムにターゲットが対応していない、という状況が生じることが確認されている [109]。この問題を解消するために、ターゲット候補とメニューアイテムの対応の更新規則を以下のように設定した。新しく選ばれたターゲット候補のうち以前もターゲット候補として選択されていたターゲット候補は、以前と同じメニューアイテムに対応する。一方、以前はターゲット候補として選択されていなかったターゲット候補は、視線から近い順で、右下から時計回りに、対応しているターゲット候補が最も少ないメニューアイテムに対応する。つまり、ターゲット候補の対応が最少ではないメニューアイテムはスキップされる。この対応が一周すると、更なる対応は最後に対応を行ったメニューアイテムから逆回りで一周するまで対応を行う。この操作をターゲット候補すべての対応が終了するまで行う。図 5.7c にターゲット候補とメニューアイテムの対応の更新の流れの例を示す。一層目のメニューアイテムは、それぞれが有する最下層のメニューアイテム数だけターゲット候補と対応することができる。ゆえに AG1 は 1、AG3 は 9 (3 二層目のメニューアイテム数 × 3 三層目のメニューアイテム数) である。

ターゲット候補とメニューアイテムの対応を示すフィードバックとして、メニューアイテムの内部に対応したターゲット候補が描画された。またターゲット候補側には、対応したメニューアイテムの色の輪郭が描画された (図 5.7d)。この輪郭はユーザから見える領域にのみ描画された。

### 5.2.3 メニューアイテム選択によるターゲット選択

メニューアイテムを一定時間注視するとそのメニューアイテムが選択され、メニューアイテムに対応するターゲット候補の数に応じて特定の操作が実行される。ターゲット候補が複数対応しているメニューアイテムを 200 ms 注視すると、子階層のメニューが表示され、選択されたメニューアイテムに対応するターゲット候補が子階層のメニューアイテムに分配される。これらのターゲット候補は視線から近い順に時計回りで子階層のメニューアイテムに分配される。分配が一周すると、最後に分配されたメニューアイテムから逆回りに分配が行われる。この操作をターゲット候補すべての分配が終了するまで行う。図 5.7e にターゲット候補の分配の流れの例を示す。ターゲット候補が一つだけ対応しているメニューアイテムを 500 ms 注視すると、その対応しているターゲット候補が選択される。

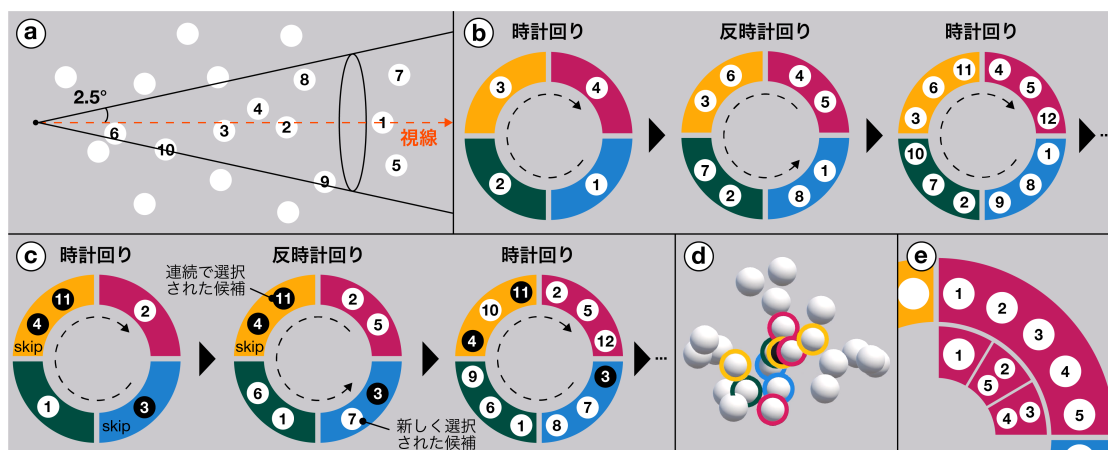


図 5.7: (a) ターゲット候補を選択する際のイメージ. 視線から 2.5° 以内のオブジェクトがターゲット候補として選択されている. オブジェクト内の数字は視線からの近さの順を示している. (b) ターゲット候補とメニューアイテムの対応の例. (c) ターゲット候補とメニューアイテムの対応の更新の例. (d) ターゲット候補の輪郭に表示される, 対応しているメニューアイテムを示すフィードバック. (e) 子階層のメニューアイテムへのターゲット候補の分配の例.

Midas Touch を避けることを目的として, 最終的な選択に必要な滞留時間を 500 ms とし, それ以外を 200 ms に設定した.

現在選択されておらず, かつ複数のターゲット候補が対応している親階層のメニューアイテムを 200 ms 注視すると, 現在展開されている子階層のメニューが閉じ, 新しく選択されたメニューアイテムに対応する子階層のメニューアイテムが表示される. 一方で現在選択されておらず, かつターゲット候補が一つだけ対応している親階層のメニューアイテムを 500 ms 注視すると, その対応しているターゲット候補が選択される. 子階層のメニューが表示されている時にメニュー領域以外の領域を 1000 ms 以上注視すると, すべての子階層のメニューが閉じ, 初期状態に戻る. 子階層のメニューが開いている時はターゲット候補は再選択されることはなく, ターゲット候補を再選択するためには一度初期状態に戻す必要があった.

### 5.3 実験: Asteroid Gazer の性能評価実験

以下の 2 点の項目の調査を目的として実験を行った. (1) Asteroid Gazer は一般的な視線選択手法と比べて遮蔽されたオブジェクトをどれだけ正確に選択することが出来るか. (2) オブジェクトが密集した状況において, Asteroid Gazer の階層メニューが選択の精度向上に寄与するか.

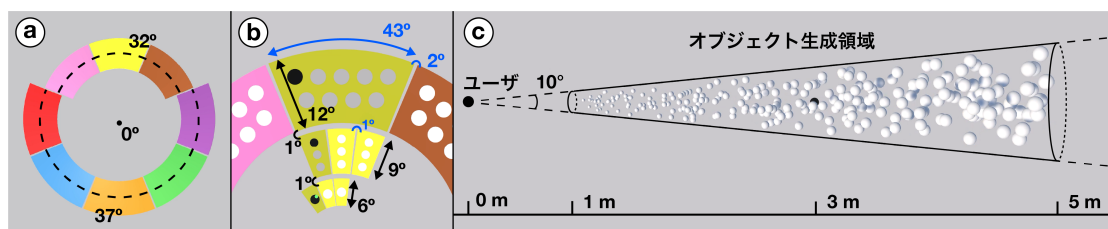


図 5.8: (a) 一層目のメニューの外観と配置. (b) メニューアイテムのパラメータ. 黒字はユーザから見た視角であり, 青字はメニュー中心からの角度である. (c) 生成されたオブジェクトの俯瞰図.

### 5.3.1 実験参加者と使用機器

実験参加者として 18 名 (男性 9 名, 女性 9 名, 平均年齢 21.6 歳, 標準偏差 1.1 歳) が実験に参加した. 実験時 9 名が裸眼であり, 9 名がコンタクトレンズを用いて視力矯正していた. 5 名は実験以前に VR における視線計測の経験があった. 実験報酬は 2,000 円であった. 実験時間は約 1 時間であった.

本実験は 4.90Ghz の Intel Core i9-11900H の CPU と NVIDIA GeForce RTX 3080 を搭載したパソコンで行った. 実験に用いたソフトウェアは Unity ver. 2019.4.29f1 で作成した. HMD は HTC VIVE Pro Eye を用いた. HTC VIVE PRO Eye の FoV は水平方向に 100°, 垂直方向に 110° であり, 搭載されているアイトラッカのサンプリングレートは 120 Hz であった. ただし, HMD のフレームレートが 90 Hz であったため視線データは 90 Hz で収集された.

### 5.3.2 比較手法

視線を用いたオブジェクト選択手法である Dwell-based selection method (Dwell), Asteroid-Gazer-1 (AG1), Asteroid-Gazer-3 (AG3) の比較を行った. Dwell は提案手法が遮蔽されたオブジェクトの選択を容易にするか確認するために, ベースライン手法として採用した. AG1 と AG3 は階層メニューが高密度な状況における選択を容易にするか確認するために採用した. また本研究は視線のみで遮蔽されたオブジェクトを選択する手法の検討が目的であるため, 視線以外のモダリティと組み合わせた手法 (e.g. Outline Pursuits [174]) との比較は行わなかった.

#### Dwell-based selection method (Dwell)

Dwell は視線選択手法におけるベースライン手法である. この手法はユーザが 600 ms の間凝視したオブジェクトを選択する手法である. 視線上に複数のオブジェクトが存在す

る場合、それらの中からユーザに最も近いオブジェクト（ユーザから見えるオブジェクト）が選択される。

### Asteroid-Gazer-1 (AG1)

AG1 は非階層メニュー（単一メニュー）を有した Asteroid Gazer である。AG1 はメニューアイテムごとに一つのターゲット候補のみが対応するため、一度のメニューアイテム選択でターゲット候補を選択することができる。メニューアイテムの数は8である。ゆえに AG1 は最大8つのオブジェクトをターゲット候補とすることができる。

メニューアイテムは8方位の方向にそれぞれ放射状に配置される。これらのメニューアイテムはユーザの頭部から 0.8 m の位置に常に表示されている。メニューアイテムの形状は環状扇形 (Annular Sector) である。上方向 (右上, 上, 左上) のメニューアイテムは頭部方向から  $32^\circ$ 、それ以外の方向のメニューアイテムは  $37^\circ$  の位置に配置した。メニューアイテムの長さは視角で  $12^\circ$  であり、メニューアイテムが成す角度は  $43^\circ$  であった。メニューアイテムの色は真右から時計回りに紫, 緑, 橙, 青, 赤, 桃, 黄, 茶色であった。AG1 のメニューの外観を図 5.8a に示す。

### Asteroid-Gazer-3 (AG3)

AG3 は三層の階層メニューを有した Asteroid Gazer である。一層目のメニューアイテムの数は8であり、二層目と三層目のメニューアイテムの最大数は3である。つまり AG3 は最大で 72 (8 一層目のメニューアイテム  $\times$  3 二層目のメニューアイテム数  $\times$  3 三層目のメニューアイテム数) のオブジェクトをターゲット候補とすることができる。一層目のメニューアイテムに対応するターゲット候補の最大数は9、二層目は3、三層目は1である。親メニューアイテムにターゲット候補が二つ対応している場合は二つの子メニューアイテムが表示され、三つ対応している場合は三つのメニューアイテムが表示された。

AG3 の一層目のメニューは AG1 と同一である。二・三層目のメニューアイテムは一層目のメニューアイテムと同様に環状扇形である。二層目のメニューは、親となる一層目のメニューの位置から  $11.5^\circ$  内側に表示され、メニューアイテムの長さは視角で  $9^\circ$  であった。三層目のメニューは、親となる二層目のメニューの位置から  $8.5^\circ$  内側に表示され、メニューアイテムの長さは視角で  $6^\circ$  であった。つまりメニューは層ごとに視角で  $1^\circ$  離れて表示された。二・三層目のメニューアイテムが二つの場合、メニューアイテムが成す角度はそれぞれ  $21^\circ$  であり、三つの場合はそれぞれ  $13.7^\circ$  であった。つまりこれらの層のメニューアイテムは  $1^\circ$  離れて表示された。また二・三層目のメニューはユーザの頭部から 0.8 m の位置に表示された。AG3 のメニューの外観を図 5.8b に示す。

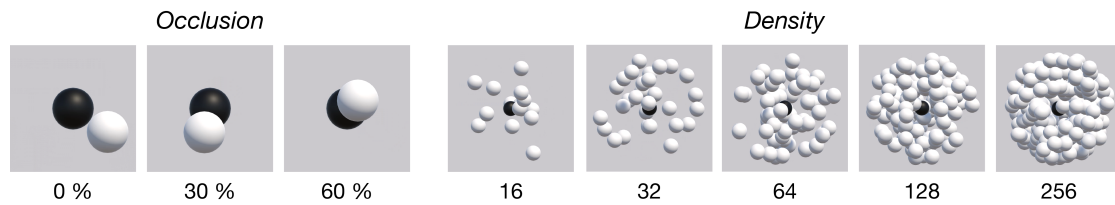


図 5.9: 独立変数 *Density* と *Occlusion*. 黒いオブジェクトがターゲット.

### 5.3.3 実験デザイン

実験は参加者内実験計画を用いて行った. 独立変数は3つ存在し (図 5.9), それぞれ以下のように設定されている.

- 手法 (*Method*; Dwell-based Selection (Dwell), Asteroid-Gazer-1 (AG1), Asteroid-Gazer-3 (AG3))
- 遮蔽率 (*Occlusion*; 0 %, 30 %, 60 %)
- 密度 (*Density*; 16, 32, 64, 128, 256 オブジェクト)

*Method* は先で説明した三つの手法である. *Occlusion* はターゲットオブジェクトの可視領域面積が遮蔽される割合であり, 1つの遮蔽オブジェクトで制御される. *Occlusion* が0%のときはターゲットは遮蔽されず, 遮蔽オブジェクトはターゲットオブジェクトに接するよう見える位置に配置される. この *Occlusion* の値は遮蔽時と非遮蔽時の比較を行うために採用した. *Density* は画面に表示されるオブジェクトの数である.

*Method* はラテン方格法を考慮した順番で提示された. 実験参加者は *Method* ごとに6セッション行った. 各セッションはランダムな順序で提示される *Density*, *Occlusion* の15の組み合わせによって構成されている. ゆえに実験参加者は3 *Method* × 6 セッション × 5 *Density* × 3 *Occlusion* = 270回のターゲット選択を行った. また実験参加者は18名であるため, 合計4,860データが収集された.

### 5.3.4 実験手順

実験参加者が実験室に到着後, 椅子に座らせ, 実験の概要およびタスクの内容を説明した. その後 HMD を装着させ, HTC VIVE Pro Eye で提供されているアイトラッカの5点キャリブレーションを行った. そして実験参加者が実験タスクの挙動を理解するまで, 実験タスクからパラメータを変更したタスクを練習として行わせた.

実験タスクは VR 環境における遮蔽されたターゲットを視線で選択するタスクである. 選択対象オブジェクトは独立変数 *Density* と同じ数だけ配置されており, 全て直径  $1.5^\circ$  の

球（ターゲットは黒色，それ以外は白色）である．オブジェクトの生成領域は円錐台の形状をしており，円錐の頂点が実験参加者の頭部に位置している．円錐台の角度は  $10^\circ$  であり，この生成領域内にオブジェクトが生成される．生成領域の上部の底面は実験参加者の頭部から 1.0 m の位置であり，下部の底面は 5.0 m であった．ターゲットは生成領域の中心（実験参加者から 3.0 m）の位置に生成された．遮蔽オブジェクトは *Occlusion* の値だけ実験参加者の視界からターゲットを遮蔽し，かつ実験参加者から 2.9 m の位置に生成された．それ以外のオブジェクトはターゲットを視覚的に遮蔽しない位置にランダムに配置された（図 5.8c）．タスク中，視線カーソル（直径  $1.0^\circ$  の緑の円）が常に表示された．Dwell では選択中のフィードバックとして，視線上のオブジェクトが黄色の半透明の球で囲まれるフィードバックが与えられた．AG1 と AG3 では選択中のフィードバックとして，視線上のメニューアイテムの彩度が低くなった．視線データは Saccade detection [38] with a triangular kernel filter [82] によって平滑化された．実験参加者がオブジェクトを選択するか，10 秒間何も選択しなかったとき，オブジェクト群は再生成された．本実験ではターゲットではないオブジェクトを選択した試行や，10 秒間何も選択されなかった試行をエラーと定義した．またエラーが生じた場合，実験参加者にその旨を伝えるために 500 ms のポーズが生じた．さらに試行の成否に応じた音声のフィードバックが与えられた．オブジェクト群が再生成される際，以前のターゲットが生成された位置から  $15^\circ$  離れ，かつ胸部正面方向から  $25^\circ$  離れたランダムな位置にオブジェクト群が再生成された．上記が 1 試行に該当し，実験参加者はこの試行を繰り返す必要があった．

実験参加者は全部で 18 セッション行った．1 セッションは 15 試行で構成されており，この 15 試行は *Occlusion* と *Density* の組み合わせである．実験参加者は *Method* ごとに 6 セッションを経験し，その後 System Usability Scale (SUS) [17] と Raw NASA Task Load Index (Raw NASA-TLX) [54, 53] に回答した．この手順を 3 つの *Method* に対して繰り返した．そして 18 セッション完了した後，手法の好みのアンケートとインタビューを行った．実験時間は約 60 分であった．

### 5.3.5 評価指標

本実験の従属変数は「選択時間」「エラー率」「候補選択時間」である．「選択時間」は試行の開始からターゲットの選択に成功するまでの時間である．つまり，エラーが生じた場合，その試行において選択時間は計測されなかった．Dwell においてはターゲットを 600 ms 注視し終えた瞬間，AG1 と AG3 においてはターゲットが唯一対応したメニューアイテムを 500 ms 注視し終えた瞬間に計測を止める．「エラー率」はすべての試行のうちのエラーが生じた割合である．ターゲットではないオブジェクトを選択した試行と，10 秒間選

択が行われなかった試行をエラーと定義した。「候補選択時間」は試行の開始からターゲット候補の選択が完了するまでの時間である。ここでの「候補選択」は一般的に試行中に何度も起こる事象であるが、候補選択時間ではターゲット選択に成功した際の最後に行った候補選択までの時間とする。つまり、エラーが生じた場合、その試行において候補選択時間は計測されなかった。また候補選択時間は AG1 と AG3 においてのみ計測し、Dwell においては計測しなかった。

実験参加者は *Method* ごとに、ユーザビリティと精神的負荷を計測することを目的とした System Usability Scale (SUS) と Raw NASA Task Load Index (Raw NASA-TLX) のアンケートに回答した。さらに実験参加者は *Method* の好みのランキングを回答した。

### 5.3.6 結果

それぞれの従属変数（選択時間、エラー率、候補選択時間）に対してノンパラメトリックな分散分析手法である整列ランク変換 (ART) [58, 166, 211] を行い、混合モデル REML を用いてデータを評価した。多重比較にはウィルコクソンの符号順位検定を用い、Holm 法 [62] による補正を用いた。SUS と Raw NASA-TLX によって得られた値はフリードマン検定とウィルコクソンの符号順位検定を用い、Holm 法による補正を用いた。

#### 選択時間

手法 (*Method*)、遮蔽率 (*Occlusion*)、密度 (*Density*) を独立変数とし、選択時間を従属変数として分析を行った。選択時間に対して *Method* ( $F_{2,3728.3} = 18.35, p < .01$ )、*Occlusion* ( $F_{2,3726.7} = 17.49, p < .01$ )、*Density* ( $F_{4,3726.3} = 65.63, p < .01$ ) に有意な差が確認された。さらに 1 次の交互作用として *Method* × *Occlusion* ( $F_{4,3726.7} = 14.15, p < .01$ )、*Method* × *Density* ( $F_{8,3726.2} = 7.65, p < .01$ )、*Occlusion* × *Density* ( $F_{8,3726.3} = 2.27, p < .05$ ) が、2 次の交互作用として *Method* × *Occlusion* × *Density* ( $F_{16,3726.2} = 1.94, p < .05$ ) が確認された。図 5.10 に *Method* × *Occlusion* と *Method* × *Density* の選択時間を示す。

#### エラー率

手法 (*Method*)、遮蔽率 (*Occlusion*)、密度 (*Density*) を独立変数とし、エラー率を従属変数として分析を行った。エラー率に対して *Method* ( $F_{2,4798} = 1033.66, p < .01$ )、*Occlusion* ( $F_{2,4798} = 346.89, p < .01$ )、*Density* ( $F_{4,4798} = 21.31, p < .01$ ) に有意な差が確認された。さらに 1 次の交互作用として *Method* × *Occlusion* ( $F_{4,4798} = 143.81, p < .01$ )、*Method* × *Density* ( $F_{8,4798} = 30.21, p < .01$ )、*Occlusion* × *Density* ( $F_{8,4798} = 8.14, p < .01$ ) が、



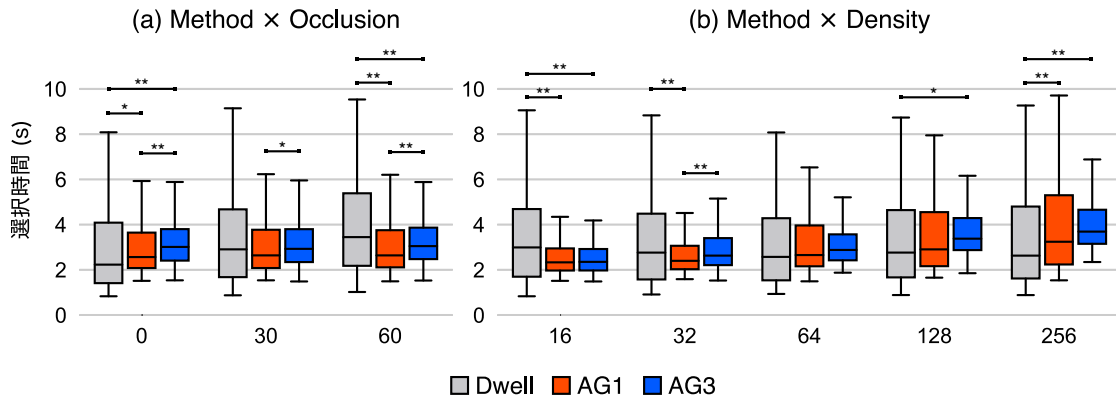


図 5.10: (a) *Method* × *Density* と (b) *Method* × *Occlusion* の選択時間. 有意差は\* $p < .05$ , \*\* $p < .01$ .

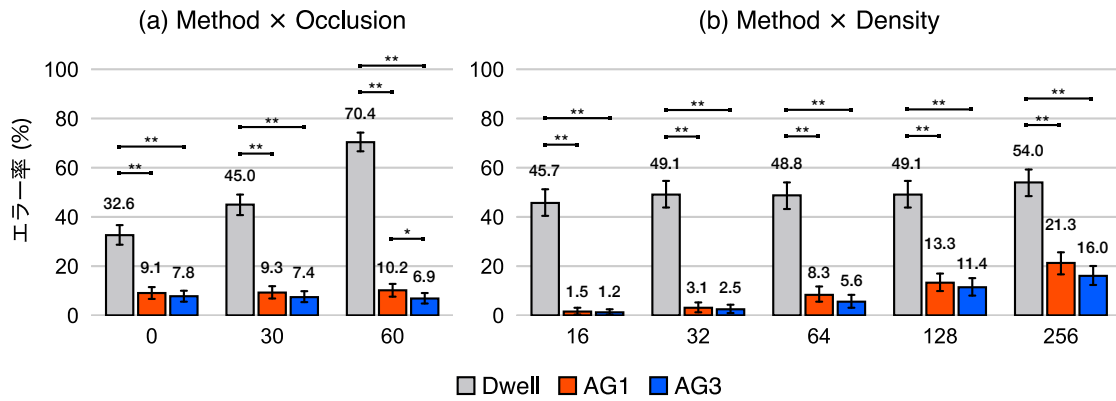


図 5.11: (a) *Method* × *Density* と (b) *Method* × *Occlusion* のエラー率. エラーバーは95%信頼区間. 有意差は\* $p < .05$ , \*\* $p < .01$ .

2 次の交互作用として *Method* × *Occlusion* × *Density* ( $F_{16,4798} = 3.43, p < .01$ ) が確認された. 図 5.11 に *Method* × *Density* と *Method* × *Occlusion* のエラー率を示す.

### 候補選択時間

手法 (*Method*), 密度 (*Density*), 遮蔽率 (*Occlusion*) を独立変数とし, 候補選択時間を従属変数として分析を行った. 分析に用いるデータは AG1 と AG3 のデータのみを使用した. 候補選択時間に対して *Method* ( $F_{1,2920.7} = 272.41, p < .01$ ), *Density* ( $F_{4,2920.2} = 65.50, p < .01$ ), *Occlusion* ( $F_{2,2920.1} = 3.54, p < .05$ ) に有意な差が確認された. さらに 1 次の交互作用として *Method* × *Density* ( $F_{4,2920.2} = 47.52, p < .01$ ), *Density* × *Occlusion* ( $F_{8,2920.1} = 4.23, p < .01$ ) が, 2 次の交互作用として *Method* × *Density* × *Occlusion* ( $F_{8,2920.1} = 5.25, p < .01$ ) が確認された. 図 5.12 に *Method* × *Density* の候補選択時間を

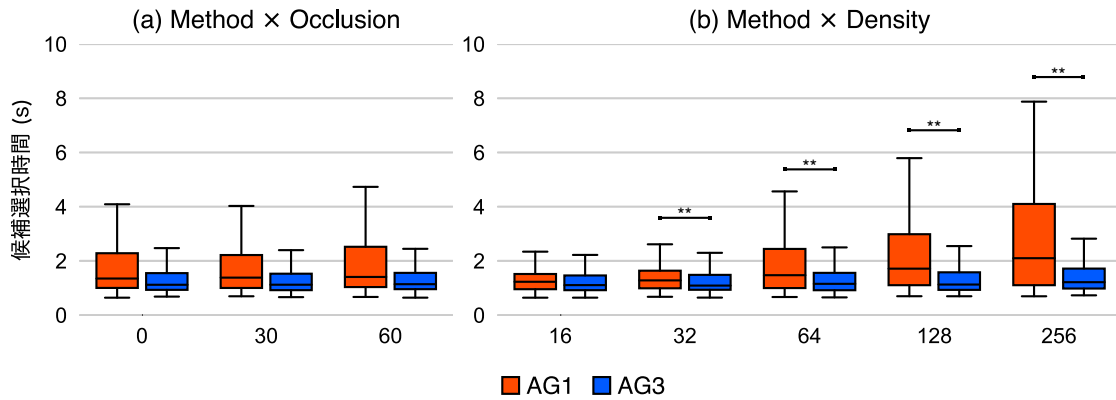


図 5.12: *Method* × *Density* の候補選択時間. 有意差は\*\* $p < .01$ .

示す.

### System Usability Scale (SUS)

Dwell, AG1, AG3 の平均 SUS スコアはそれぞれ 50.42, 78.61, 74.44 であった (高いほど良い). フリードマン検定より, *Method* ( $\chi^2_{2,N=18} = 22.29, p < .01$ ) に有意差が確認された. また事後検定より, Dwell と AG1 ( $Z = -3.64, p < .01$ ), Dwell と AG3 ( $Z = -3.64, p < .01$ ) に有意差が確認された (図 5.13a).

### Raw NASA Task Load Index (Raw NASA-TLX)

Dwell, AG1, AG3 の平均ワークロードスコアはそれぞれ 58.01, 27.41, 26.85 であった (低いほど良い). NASA-TLX によって得られた 6 つの項目に対するスコアと平均ワークロードスコアに対してフリードマン検定を用いて分析した結果, *Method* に対して精神的要求 ( $\chi^2_{2,N=18} = 25.21, p < .01$ ), 肉体的要求 ( $\chi^2_{2,N=18} = 17.82, p < .01$ ), 時間的切迫感 ( $\chi^2_{2,N=18} = 7.62, p < .05$ ), 作業達成度 ( $\chi^2_{2,N=18} = 23.68, p < .01$ ), 努力 ( $\chi^2_{2,N=18} = 23.91, p < .01$ ), 不満 ( $\chi^2_{2,N=18} = 24.03, p < .01$ ), 平均ワークロード ( $\chi^2_{2,N=18} = 27.00, p < .01$ ) に有意差が確認された. また事後検定より平均ワークロードに対して Dwell と AG1 ( $Z = -3.70, p < .01$ ), Dwell と AG3 ( $Z = -3.70, p < .01$ ) に有意差が確認された (図 5.13b).

### 実験参加者の各手法 *Method* の好み

図 5.14 に *Method* の好みを示す. 図より AG1, AG3, Dwell の順に好まれた. AG1 は 10 人, AG3 は 7 人, Dwell は 1 人の実験参加者に最も好まれた. AG1, AG3, Dwell の平

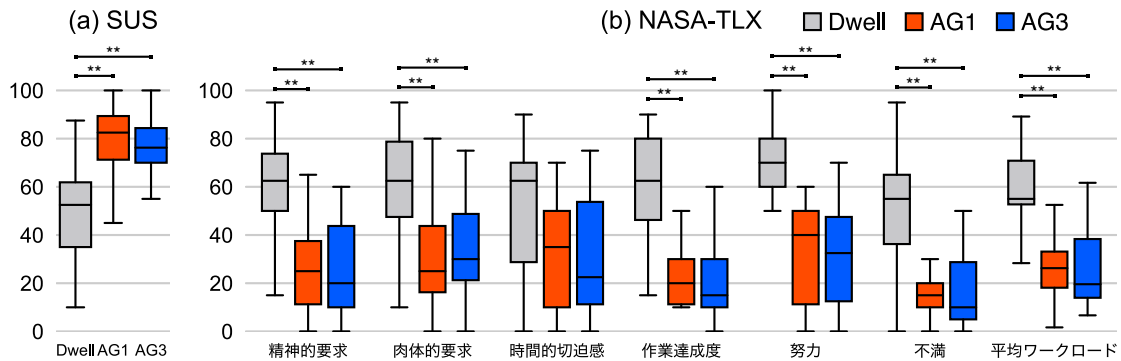


図 5.13: Method ごとの (a) SUS, (b) Raw NASA-TLX の結果. 有意差は\*\* $p < .01$ .

均ランクはそれぞれ 1.56, 1.67, 2.78 であった。

AG1 は最も好まれた手法であり, 10 人の実験参加者が AG1 を 1 位に選んだ. P6 は「AG1 では最初のターゲット凝視が比較的簡単かつメニューに載っているかがわかりやすく, 選択も 1 度のみでスムーズにできました」とコメントした. 一方で, *Density* が高い時の選択が難しいことを実験参加者は報告した. P2 は「選択対象物が非常に多くない場合では, メニューアイテムが 8 つの AG1 で十分だと感じたが, どうしても候補が 8 つでは不十分な時はあった」とコメントした.

AG3 は 2 番目に好まれた手法であり, 17 人の実験参加者が AG3 を 1 位か 2 位に選んだ. P1 は「オブジェクトの数が多い時には, 大部分が隠れているターゲットもメニューに表示されるという点でとても使いやすい手法だと感じた」とコメントした. 一方で階層メニューによる選択手順の増加と, 三段目のメニューアイテムの小ささに対する不満を実験参加者は報告した. P5 は「AG3 は二段目くらいまでならちょうどいいと思ったが, 三段目まで選択肢が出るときに操作が多いため, メニュー選択が面倒だと感じる時もあった」とコメントした. P2 は「AG3 では最も子の階層の選択エリアが小さくて難しかった」とコメントした.

Dwell は最も好まれなかった手法であり, 15 人の実験参加者が Dwell を 3 位に選んだ. この順位は, 小さくかつ遮蔽されたターゲットの選択が困難であることが原因である. P6 は「的が小さいため, ほかの球に視線が分散してしまい一定時間の凝視が難しかった」とコメントし, P3 は「Dwell の場合, 遮蔽された部分によって, 誤ったオブジェクトが選択されてしまうことがあった」とコメントした.

Asteroid Gaze (AG1, AG3) はエラー率, ユーザビリティ, 精神的負荷の観点でベースライン手法である Dwell よりも有意に優れていた. そのうえ NASA-TLX の項目である精神的要求, 肉体的要求, 作業達成度, 努力, 不満において, AG1 と AG3 は Dwell よりも有意に優れていた. 一方で AG1 と AG3 は *Density* の値が大きくなるほど選択時間が長く

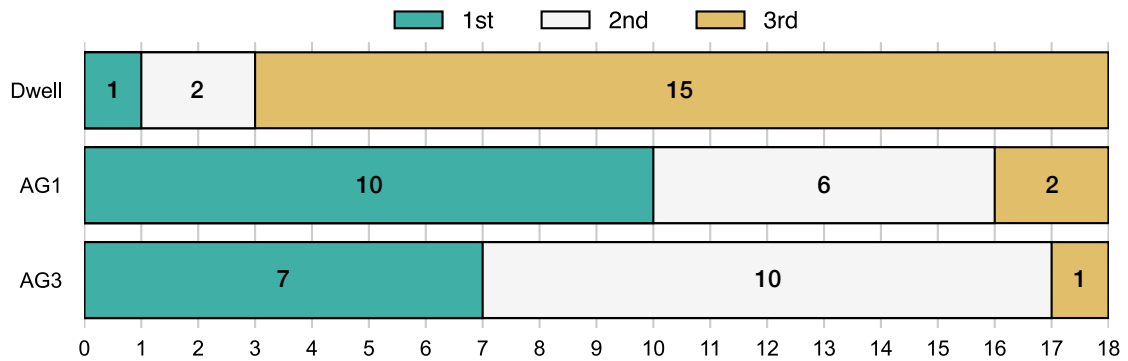


図 5.14: Method の好み.



図 5.15: Asteroid Gazer を活用したアプリケーションの図示. (a) 積まれたジグソーパズルの山からパズルピースを選択する様子. (b) テレビのチャンネルを変えようとしている様子. (c) 三つの商品を買おうとしている様子.

なり,  $Density$  が 256 の時, AG1 と AG3 は Dwell よりも有意に選択時間が長かった. したがって, Asteroid Gazer は高密度になるほど選択時間が長くなるが, 一般的な視線選択手法よりも遮蔽されたオブジェクトを正確に選択することが可能な手法だと言える.

AG3 は  $Occlusion$  が 60 % の時 AG1 よりも有意に低いエラー率を達成し, また  $Density$  が 32-256 の時 AG1 よりも短い候補選択時間を達成した. 一方で  $Occlusion$  が 0 % と 30 % の時と  $Density$  が 32 の時, AG3 は AG1 よりも有意に長い選択時間を記録した. ゆえに AG1 は AG3 よりも高速に選択が可能である一方で, AG3 よりも高いエラーが生じる手法だと言える. したがって, 階層メニューの使用は, 選択時間を増加させる代わりに遮蔽されたオブジェクトを正確に選択することを可能とし, かつ高速なターゲット候補選択を可能とするとと言える.

## 5.4 応用アプリケーション例

本章では, 実験結果を受けて開発したアプリケーション例を 3 つ示す.

#### 5.4.1 ジグソーパズルアプリケーション

Asteroid Gazer は積まれたジグソーパズルなどの、高密度で遮蔽されたオブジェクト群からターゲットを選択することが可能である (図 5.15a)。ジグソーパズルを遊ぶ際、ユーザは使用していないジグソーパズルが積まれた山から所望のジグソーパズルのピースを取り出し、ジグソーパズルを組み立てる空間にそれを持ってくる操作を行う。この操作は手で行うことが一般的であるが、ジグソーパズルの山が遠くにある場合、遠い場所に手を伸ばすことによる肉体的負担が生じる。一方でジグソーパズルの山がユーザに近い場合、パズルを組み立てる領域を侵害し、組み立て作業の阻害になる可能性がある。Asteroid Gazer は視線によって積まれたジグソーパズルの山から所望のピースを取り出すことが出来るため、ジグソーパズルの山がユーザから離れた位置にある場合においても低い肉体的負荷で取り出す操作を行うことが出来る。

#### 5.4.2 スマートホームアプリケーション

ターゲットが明示的である場合、つまりターゲットの大きさが十分に大きい場合、Asteroid Gazer は Radi-Eye [177] のようなターゲットに対する円形インタフェースとして動作させることが出来る。ゆえに円形インタフェースの例として、VR 環境内で動作するスマートホームアプリケーションを実装した (図 5.15b)。本アプリケーションでは、対象の電子機器に対して短い時間の注視を行うと、その電子機器に応じた円形インタフェースが表示される。この円形インタフェースは階層メニューを有したインタフェースであり、Kuiper Belt に表示されている。したがって、電子機器の選択中に Midas Touch は生じず、電子機器に対する複雑な操作を視線のみで達成することが出来る。一方で操作の決定のためにはインタフェース内を視覚探索する必要がある、その際には Midas Touch が生じる。ゆえに操作を確定するための確認ボタンを階層メニューの最も内側に配置し、インタフェース内の自由な視覚探索を可能とした。

#### 5.4.3 ショッピングアプリケーション

前述したアプリケーションより、Asteroid Gazer を用いることで高密度な状況におけるオブジェクトの選択や、十分に大きなサイズに対する複雑な操作が可能である。これに加えて、Asteroid Gazer は中程度の密度のオブジェクトに対して中程度に複雑な操作を行うことが出来る。ゆえにこの操作方法の例示として、VR 環境におけるショッピングアプリケーションを実装した (図 5.15c)。ユーザはまずターゲットを一層目のメニュー選択で選択し、それ以降のメニューアイテム選択で様々な操作 (商品情報の確認、買う個数の決定)

を決定した。

## 5.5 議論

### 5.5.1 部分的に遮蔽されたオブジェクト選択タスクにおける Asteroid Gazer の性能

実験結果よりすべての *Occlusion* と *Density* の組み合わせにおいて、Asteroid Gazer (AG1, AG3) は Dwell よりも正確にターゲットを選択することができた。 *Occlusion* の値が 0 % の時、つまりターゲットが遮蔽されていない時、 *Method* ごとのエラー率は Dwell が 32.6 %、AG1 が 9.1 %、AG3 が 7.8 % であった。 その上 *Occlusion* の値が 60 % の時、つまり本実験において最もターゲットが遮蔽されている時、エラー率は Dwell が 70.4 %、AG1 が 10.2 %、AG3 が 6.9 % であった (図 5.11)。 ゆえに Asteroid Gazer は一般的な視線入力手法 (Dwell) では選択が難しいような、小さいまたは遮蔽されたオブジェクトの選択を容易にすることが出来ると言える。 一方で、AG1 と AG3 のエラー率は *Density* の値が大きくなるほど悪化し、 *Density* の値が 128 (AG1:13.3%, AG3:11.4%) や 256 (AG1:21.3%, AG3:16.0%) の時は特に悪化した。 ゆえに Asteroid Gazer は一般的な視線インタフェースよりも高密度な状況で動作するが、超高密度な状況においては選択精度が悪化するため、改善が必要だと言える。

また Asteroid Gazer は既存の手を用いない視線マルチモーダル手法よりも優れている可能性がある。 Outline Pursuits [174] は視線と頭部方向を組み合わせた手法であり、遮蔽率が 50 % 以下でかつ画面内のオブジェクト数が 40 の時に約 30 % のエラー率を記録した。 DEEP [221] は視線と瞬きを組み合わせた手法であり、遮蔽率が 50 % が 100 % でかつ画面内のオブジェクト数が 50 の時に 6.2 % 以下のエラー率を記録した。 タスクのデザインが異なるとはいえ、Asteroid Gazer はこれらの手法と比較しても、同程度の環境 (*Occlusion*=60%, *Density*=16-256) において十分な精度 (AG3; 6.9 %) を達成している。 そのうえ、Asteroid Gazer は他手法ではできないタスクに用いることが出来る。 Asteroid Gazer の仮想ターゲット再配置アプローチは視覚探索を含むオブジェクト選択タスクに適している。 他手法はオブジェクトを見つめることは Midas Touch の懸念を生じさせるが、本手法は現実環境のような複雑な視覚探索が必要な環境においても用いることが可能だと考えられる。 さらに、本手法はターゲットと遮蔽物の距離が非常に近い場合においても正確にターゲットを選択することが出来る。 DEEP は視覚深度を調整して深度以前のオブジェクトを非表示にすることで遮蔽物を非表示にするアプローチを採用しているが、ターゲットを遮蔽物の距離が近いとこの視覚深度調整が非常に困難となることが想定できる。 つまり、本章で行ったような実験タスクは DEEP にとって不適切なタスクだと言える。 一

方で DEEP は完全に遮蔽されたオブジェクトを選択することができるが、Asteroid Gazer はそれが難しい。したがって、タスクや状況に応じて適切な手法を選択することが肝要だと言える。

### 5.5.2 階層メニューの利点

実験結果より、AG1 と AG3 は遮蔽されたオブジェクトの選択に適した手法だと言える。しかし同じ Asteroid Gazer の一種でありながら、AG1 と AG3 の選択時間やエラーの原因などは異なる。まず AG1 と AG3 の選択時間の中央値は、密度が低い時はほとんど同じであったが、密度が高い時は AG3 は AG1 よりも大きくなった (図 5.10)。これは AG3 のメニューアイテム選択回数が密度の高低に応じて変化することが原因である。密度が低い場合、AG3 はほとんど AG1 と同様に動作し、一度のメニューアイテム選択でターゲットを選択することができる。しかし密度が高い場合、AG3 は最大 3 回のメニューアイテム選択が必要になる場合がある。このメニューアイテム選択回数の違いが AG1 と AG3 の選択時間の違いに寄与していると考えられる。ゆえに AG1 は AG3 よりも高速にターゲットを選択することができるが、一方で AG1 の候補選択時間の中央値 AG3 よりも大きく、また分布も AG3 よりも広がった (図 5.12)。これは AG1 が AG3 よりもターゲットをターゲット候補として選択することが困難であったためだと考えられる。つまり AG1 の最大ターゲット候補数 8 は高密度においては不十分であり、AG1 の選択時間の多くがターゲット候補選択に費やされていると言える。したがって、階層メニューは高密度における選択時間の増加を招く代わりに高速で安定したターゲット候補選択を可能にすると言える。

次に *Occlusion* と *Density* の組み合わせにおいて、AG3 は AG1 よりも低いエラー率を記録した。AG1 のエラーの原因の多くは高密度な状況におけるターゲット候補選択の困難さに起因すると考えられる。AG1 の候補選択時間の分布は AG3 と比較すると非常に広く、箱ひげ図の最大値が制限時間の 10 秒に近い条件 (AG1, *Density*=256: 最大値 (第三四分位数 +  $1.5 \times$  四分位範囲) = 7.88 秒) も存在する。ゆえに AG1 はターゲット候補選択の困難さによって制限時間以内にターゲットを選択できず、エラー率が大きくなったと考えられる。一方で AG3 は AG1 と異なりターゲット候補選択が容易であるため、AG3 のエラーの原因は AG1 と異なると考えられる。AG3 のエラーの原因として、(1) メニューアイテム選択回数の増加によるエラー抽選回数の増加、(2) 三層目の上方向のメニューアイテムの大きさが小さい、が考えられる。三層目上方向のメニューアイテムは、短辺が  $2.13^\circ$ 、長辺が  $3.57^\circ$ 、高さが  $6.00^\circ$  であった。これは PC を対象としたアイトラッカを使用する場合においては十分なサイズである [38] が、VIVE Pro Eye のアイトラッカの精度では不十分である可能性がある [179]。したがって、より高精度なアイトラッカを使用す

ることで AG3 の選択精度をより向上させることが出来ると考えられる。よって上記より、階層メニューを用いる Asteroid Gazer は選択時間を増大させるが、高速で安定したターゲット候補選択やエラー率の削減を実現すると言える。

### 5.5.3 Asteroid Gazer のパラメータ

実験で用いた Asteroid Gazer は十分な精度を達成したが、パラメータを改善することでより性能を向上させることが出来ると考えられる。まず、メニューアイテムサイズを調整することでエラー率を低減することが出来ると考えられる。前述したように、AG3 のエラーの原因の一つは三層目上方向のメニューアイテムサイズが小さいことに起因している。ゆえにこれらのメニューアイテムサイズを選択に十分なサイズに調整することで、選択精度を向上させることが可能だと言える。また代替案としては、子階層のメニューアイテムサイズは親階層のメニューアイテムサイズに依存しているため、親階層のメニューアイテムサイズを調整する（上方向の親階層メニューアイテムサイズを他方向よりも大きくする、親階層のメニューアイテム数を減らす）ことで間接的に三層目上方向のメニューアイテムサイズを大きくすることが出来る。次に、ターゲット候補数をより大きくすることで更なる高密度においても十分に動作するように出来る。実験で用いた AG3 の最大ターゲット候補数は 72 であり、この最大ターゲット候補数は最下層のメニューのアイテム数（8 一層目のメニューアイテム数  $\times$  3 二層目のメニューアイテム数  $\times$  3 三層目のメニューアイテム数）と一致する。つまりメニュー構成を工夫することでより多くの最下層メニューアイテム数を確保することが出来る。例えばメニュー層数を 4 層とすることや、 $6 \times 4 \times 4$  (=96) のメニュー構成を採用することなどが考えられる。特に  $6 \times 4 \times 4$  のメニュー構成は  $8 \times 3 \times 3$  のメニュー構成における二・三層目のメニューアイテムサイズを維持しながら最下層メニューアイテム数を増加させている。ゆえにターゲット候補選択においては  $6 \times 4 \times 4$  のメニュー構成は優れているが、一方で一層目のメニューアイテム数が減っているため低密度における選択時間が増加してしまうデメリットがある。したがって目的に適したメニュー構成を採用するべきだと言える。

### 5.5.4 制限

Asteroid Gazer には実用面に関していくつか制限がある。まず、ターゲット候補の外観が一致している場合、これらを見メニューアイテム内のフィードバックのみでは見分けることはできないため、誤選択が生じることがある。この誤選択はオブジェクト再配置アプローチが有する欠点であり、再配置時にオブジェクトの位置情報が損失することが原因である。一方で Asteroid Gazer は一般的なオブジェクト再配置手法とは異なり、ターゲッ



ト候補の輪郭の色を見ることで、一層目のメニューの数と同じだけの外観が一致したオブジェクトを区別することが出来る。ゆえに、AG1 や AG3 においては8つの外観が一致したオブジェクトを判別し、正確に選択することが可能である。

次に、Asteroid Gazer は完全に遮蔽されたオブジェクトの選択が困難である。Asteroid Gazer はターゲット候補の輪郭に対応するメニューアイテムの色を表示するフィードバックを行なっているが、これはターゲット候補の可視領域にのみ行なっており、完全にターゲット候補が遮蔽される場合このフィードバックは見えなくなる。ゆえに部分的に遮蔽されたオブジェクトと同様に完全に遮蔽されたオブジェクトを選択することはできないが、以下の二種類の方法で完全に遮蔽されたオブジェクトを選択することは出来る。一つ目の方法はユーザから可視か否かを問わず、すべてのターゲット候補の領域に応じて輪郭を描画する方法 [174] である。この方法は密度が高い状況では、輪郭の描画によって視覚的に煩わしくなる可能性があるため、低密度に限定した対処法だと言える。二つ目の方法は、現状の実装から一切変更せず、メニュー内に表示された対応するターゲット候補のフィードバックを参照してターゲットを選択する方法である。この方法は一つ目の方法と異なり、高密度においても動作する。一方でこの方法は輪郭のフィードバックを参照できないため、外観が同じオブジェクトを一切区別することができない。またメニューアイテム内の視覚探索を必須とするため、選択時間の大幅な増加と Midas Touch が見込まれる。ただしこの Midas Touch は、メニューの最下層に確認ボタンを配置することで回避することが出来る。したがって、Asteroid Gazer は完全に遮蔽されたオブジェクトを選択することは難しいが、不可能ではないと言える。

## 5.6 本章のまとめ

本章では VR 環境における視線のみで遮蔽されたターゲットを選択する手法として Asteroid Gazer を提案した。単層メニューと階層メニューの Asteroid Gazer をそれぞれ実装し、それらと一般的な視線インタフェースとの比較実験を行った。実験の結果、Asteroid Gazer は高密度な状況における遮蔽されたオブジェクトの選択時のエラー率を低減できることが明らかになった。三次元環境における遮蔽されたオブジェクトを視線のみで選択することは非常に難しいことであり、今まで達成することは出来なかった課題であった。本提案手法はこの課題を解決し、視線のみで遮蔽されたオブジェクトを選択することが可能である。それはつまり、今までの視線インタフェースが適用し得なかった領域に対しても視線インタフェースがアプローチし得るということであり、今後の視線入力研究の可能性を大きく押し広げた研究だと言える。

## 第6章 総合的な議論

本章では本論文でこれまでに述べた研究について全体を通した議論を行う。まず本論文で行なった研究を以下にまとめる。

今までの視線インタフェースは選択精度や誤選択などの観点で難を抱えており、具体的には「可視領域が小さなオブジェクトの選択が難しい」「意図しないオブジェクト注視による誤選択 (Midas Touch)」が挙げられる。本研究では前者の課題の解決を第3章と第5章で、後者の課題の解決を第4章で試みた。

「可視領域が小さなオブジェクトの選択が難しい」という課題は、二次元環境においてはターゲットサイズが小さなオブジェクトの選択が難しいことを指す。ゆえに第3章では二次元環境における小さなオブジェクトの選択を容易にする手法として、エリアカーソル手法を視線インタフェースに導入する手法を提案した。本章では Bubble Gaze Cursor と Bubble Gaze Lens の二種類のエリアカーソル視線入力手法を実装し、それぞれがポイントカーソルを用いた一般的な視線入力手法よりも小さなオブジェクトの選択を容易にすることを明らかにした。第4章では視覚探索中の「意図しないオブジェクト注視による誤選択 (Midas Touch)」を解消することを目的とし、三次元環境における視覚探索時にメニューの Midas Touch が生じないような三次元メニュー UI を開発した。本メニューは頭部方向から  $25^{\circ}$ – $45^{\circ}$  の領域 Kuiper Belt に配置されており、日常的に目を向ける領域 (頭部方向から  $< 25^{\circ}$ ) における視覚探索時にメニューに対する誤入力が生じることはない。実験で Kuiper Belt を活用することで Midas Touch を減少させることは可能か検討した結果、短い滞留時間においても Kuiper Belt で Midas Touch が生じることは少ないことを確認した。第5章では三次元環境における「可視領域が小さなオブジェクトの選択が難しい」という課題、つまり遮蔽されたオブジェクトの選択が難しい、という課題を解決することを目的として、高密度な状況でも動作する遮蔽されたオブジェクト選択手法 Asteroid Gazer を提案した。本手法は Kuiper Belt に配置したメニューアイテムに対してターゲット候補を仮想的に再配置することで遮蔽状況を解消する。そして第4章で行った研究では Kuiper Belt に配置したメニューは単層メニューであったが、本章では階層メニューの配置を試みた。これにより扱うことのできるターゲット候補数が増加し、高密度な状況においてもターゲットを正確に選択することが可能となった。

## 6.1 本研究の貢献

### 6.1.1 可視領域が小さなオブジェクトの選択

本研究では第3章にて二次元環境におけるターゲットサイズが小さなオブジェクトの選択手法 Bubble Gaze Cursor + Lens を提案し、第5章にて三次元環境における遮蔽によってターゲットサイズが小さくなったオブジェクトの選択手法 Asteroid Gazer を提案した。これらの手法で選択できるターゲットサイズは通常の視線インタフェースで選択できるターゲットサイズよりも非常に小さく、どちらの手法もターゲットの周りに十分な空間がある場合は、ターゲットサイズがどれだけ小さくても選択することが可能である。特に Asteroid Gazer はターゲットサイズや遮蔽状況に依存しない選択方法であり、かつ視覚探索時の Midas Touch も生じないことから、Bubble Gaze Cursor + Lens よりも正確なターゲット選択が期待できる手法だと言える。しかし、Asteroid Gazer があらゆる状況で Bubble Gaze Cursor + Lens よりも優れているわけではない。Asteroid Gazer は多段階の選択動作を採用している一方で、Bubble Gaze Cursor + Lens は一度の選択動作（ターゲット注視）のみで選択を完了することができる。つまり選択時間に大きな差が生じており、Asteroid Gazer はどれだけ選択しやすいターゲットでも 2-3 秒程度の選択時間が必要であるのに対し、Bubble Gaze Cursor + Lens は 1.5 秒以下でターゲットの選択を完了することができる。頻繁に行う動作は単純であればあるほどユーザーに負担がかからないため、Bubble Gaze Cursor + Lens で十分な精度で選択が可能な状況においては、Asteroid Gazer よりも Bubble Gaze Cursor + Lens を使用するべきだと言える。つまり本研究で提案したこれらの手法は、即座に選択を実行したい場合と、視覚探索などが必要な複雑な環境で正確に選択を実行したい場合の両方の状況を適切に支援する手法だと言える。

### 6.1.2 視覚探索中の意図しないオブジェクト注視による誤選択 (Midas Touch) の解決

本研究では第4章にて、三次元環境における視覚探索時の Midas Touch が生じない領域 Kuiper Belt を提案した。Kuiper Belt は頭部方向から 25°-45° の領域であり、意図しない限り人の視線が向くことはほとんどない。この領域を視線インタラクションに活用した研究は本研究を行うまで存在せず、視線インタラクションの新しい可能性を切り開くものだと言える。Kuiper Belt でのみ視線インタラクションを行う場合、日常的な視線動作の中で誤選択が生じる可能性は著しく減る。今までの視線入力手法のほぼ全ては Midas Touch が生じる可能性を常時抱えており、また Midas Touch を避けるためには多大な選択時間の増大を受け入れる必要があった。二次元環境においては画面外が実質的なデッドスペース（入力が行われない領域）となるため誤選択を減らす処理（滞留時間の増加、複雑な入力

方法の採用など)を行わずとも Midas Touch の可能性はある程度低かった。しかし三次元環境にはデッドスペースが存在しないため、三次元環境における商用視線インタフェースではターゲット選択といった重要な選択動作は視線ではほとんど実行されない実装が多い。しかし Kuiper Belt を活用するとこの Midas Touch は生じなくなる。つまり本研究は三次元環境における視線インタフェースの社会実装を大きく推し進める研究であり、今後も研究が必要な領域だと言える。

### 6.1.3 入力多様性の向上

本研究は上記の二つの課題の解決を目的とした研究である。しかし副次的な貢献も存在し、それが「複雑な入力」を実行することが可能となったことである。滞留時間入力手法は基本的にターゲットに対して一種類の入力のみを実行することができる。これはさしづめマウスの左クリックのみが実装されている状態と等しく、右クリックが視線インタフェースには実装されていないのである。ゆえに視線のみでターゲットに対して複雑な入力を実行する手法として、滞留時間入力後に視線ジェスチャ [73] や Smooth Pursuit [27] を実行することで複雑な入力を行う手法が提案されているが、小さなターゲットの選択が困難であることや、使用環境が限定的などの課題が残っていた。しかし、本研究で提案した Kuiper Belt を活用することで、今まで難しかった複雑な入力を視線のみで行うことができる。第5章のアプリケーション例として実装したスマートホームアプリケーション (5.4.2 節) やショッピングアプリケーション (5.4.3 節) より、ターゲット選択領域とメニューアイテム選択領域を Kuiper Belt を用いて分離することで複雑な入力の実行が可能である。また本手法はターゲットサイズや密集度、遮蔽度にほとんど影響されないため、既存手法とは異なり様々な場面で使用することができる。ゆえに本研究はターゲット選択の精度向上の他に、入力多様性の向上という課題に対しても貢献していると言える。

## 6.2 今後の視線インタフェースの展望

### 6.2.1 本研究の展望

#### 個人最適化

本研究で提案した手法は、視線インタフェースが抱える課題の解決に効果的であることが実験で明らかになった。一方でこれらの研究には未だ改善の余地が存在し、その改善により、より高い精度や速度で入力を完了することが可能になると考えられる。そのうちの一つはユーザに対するパラメータの個人最適化である。入力インタフェースのユーザビリティに関する研究では、ユーザに関するあらゆる指標やパラメータを平均値や中央値に選

元し、その値を参考・使用することで手法を設計する。例えば Kuiper Belt における領域の定義に用いたデータは 10 人の視線データから計算したものである。ゆえに入力手法に使用しているパラメータは個人に対して適した数値ではなく、ユーザごとに最適化することでユーザにとってよりユーザビリティの高い手法になる可能性が高い。また、パラメータ以外に視線推定精度との相性もある。視線推定精度は人によって大きくばらつくため、人によって容易な入力可能なターゲットサイズは異なる。また外環境の光など外部のパラメータによって視線推定精度は変化するため、入力手法におけるターゲットサイズをはじめとした UI デザインを状況やユーザに応じて変化させることは必須だと考えられる。

入力手法の個人最適化を行うためにはユーザごとの大量の視線データが必要となる。つまり非常に長い視線データ取得時間が必要となるため、個人最適化は現実的ではなかった。しかし HMD の常時装着や RGB カメラからの視線データの取得頻度の増加など、ユーザの視線データ取得タイミングが今後増加することから、視線データは日常的に大量に取得することが可能である。ゆえにその視線データを活用し、状況に応じて UI デザインやパラメータをユーザに対して自動的に調整可能な手法へと提案手法を改善することで、より日常生活での使用に適した手法になると考えられる。

### それぞれの手法の統合

三次元環境におけるインタラクションは、環境に固定されたオブジェクトとのインタラクションと、二次元平面に対するインタラクションに二分される。前者は三次元環境特有のインタラクションであり、空間に固定された仮想オブジェクトや IoT などの実オブジェクトとのインタラクションである。後者はディスプレイ環境における UI と同等のものを指し、ブラウザやキーボードといった平面上の UI とのインタラクションを指す。理想の三次元視線 UI はこれらのインタラクションを自然に使い分けることが出来るべきである。つまり三次元環境に対するインタラクションが可能な状態を基本としながら、自由なタイミングで二次元平面に対するインタラクションを実行可能であるべきだと考えられる。現状の VR ではコントローラにメニュー起動用のボタンを配置することで、これを実現している。しかし今までの視線インターフェースはメニュー起動用のボタンに相当する操作が存在せず、これらの実装を視線のみで達成することが困難であった。つまり理想的な三次元視線 UI では、(1) 二次元 UI への入力、(2) 三次元オブジェクトへの入力、(3) 二次元 UI を起動する入力、のそれぞれを容易かつ正確に実行できる必要がある。

これらの入力の容易な実行は、本論文にて提案した入力手法によって達成することができる。(1) は Bubble Gaze Cursor (第 3 章)、(2) は Asteroid Gazer (第 5)、(3) は Kuiper Belt (第 4) で達成することができる。つまり、これらの手法を統合することによって、二

次元環境と三次元環境におけるインタラクションをシームレスに実行することが可能になると考えられる。しかしこれらの手法を単純に統合することは難しく、統合のために新たに研究を行う必要があると考えられる。

### 6.2.2 本研究の限界とそれに応じた展望

本論文では視線入力のパフォーマンスの向上を目的とした手法を提案し、実験で提案手法のパフォーマンスを調査した。しかし本研究における実験参加者は全員健常者であり、ALS患者など手足が動かない方々に対して調査は行っていない。現在視線インタフェースを最も使用している人々は肉体的に障害を抱えた人々であり、マウスの代替の入力インタフェースとして視線インタフェースを活用している。健常者と身体障害者の視線入力のパフォーマンスは異なることが確認されている [77] ことから、本研究の調査は健常者に対する貢献に限定されていると言える。

本研究における提案手法は基本的に健常者を対象としている。特に Kuiper Belt と Asteroid Gazer は VR 環境を対象とした手法であり、パラメータは健常者の視線データや実験結果を参考に決定されている。また顔の方向を制御できない以上、身体障害者の方々のインタラクション対象は正面方向のディスプレイ、もしくは、HMD を装着した際の正面方向に対する VR・MR オブジェクトに対するインタラクションに限定される。ゆえにこれらの手法のパラメータや使用する状況は身体障害者の方々を想定しておらず、提案手法をそのまま身体障害者の方々が使うことは難しい。しかし、これらの手法のパラメータを再調節することで、身体障害者の方々に適した手法として再設計することは可能だと考えられる。Kuiper Belt は視線角度が  $25^{\circ}$ – $45^{\circ}$  の領域を指し、視線がほとんど到達しない閾値 ( $25^{\circ}$ ) と目が動く限界の閾値 ( $45^{\circ}$ ) の間の領域である。これらの値は健常者のデータを参考に決定されているため、これらの値を再設定することで身体障害者の方々も Kuiper Belt を使用することが可能になると考えられる。Kuiper Belt は VR 環境など、広い環境を探索する場合に適した手法であり、ディスプレイに対して使用する際には効力を強く発揮できない可能性が高い。しかし頭部が固定されていることからディスプレイ内のどこが Kuiper Belt か定義可能であるため、Asteroid Gazer のような Kuiper Belt を用いた複雑な入力をディスプレイ上でも実装できる可能性がある。ゆえに実験参加者の対象を健常者以外にも拡張し、調査を行うことが必要だと言える。

### 6.2.3 モノモーダル視線インタフェースの展望

本稿は視線のみで入力を完了するモノモーダルインタフェースのパフォーマンス向上を目的とした研究であり、あらゆる人間が視線のみで入力を実行することが一般的となる世界を志向し

ている。この未来はVR・MRの普及やスマートグラスが普及し、人間の視線が容易にトラッキング出来る未来においては有り得るシナリオの一つだと考えられる。人間の手は非常に高性能で有り、日常生活ではあらゆる行為を手で実行している。ゆえに手は様々なタスクによって塞がりがちであり、その他の作業を手以外の優秀なモダリティ、視線が担うことは十分にあり得る話である。ゆえに視線インタフェースだけであらゆる入力を実行可能にすることは重要な研究課題であり、本論文はこの目的の達成に対して大きく歩を進めた論文だと言える。

以前までのモノモーダル視線インタフェースでは、視線の速度ごとの視線運動（固視、サッケード、滑動性眼球運動）に応じた手法の開発がなされてきた。これは二次元環境におけるインタラクションが一般的な視線インタフェースにおいて、視線の位置情報のみがアクセス可能な視線データであったためである。そのためあらゆる視線運動が視線インタラクションに検討された結果、新しい視線入力手法はほとんど提案されなくなった。最後に注目された視線運動である滑動性眼球運動を用いた視線入力手法 [195] が2013年の提案であることを考慮すると、本論文が書かれた2023年度まで10年ほど革新的な視線入力手法は現れなかったと言える。ゆえに視線速度ごとの視線運動以外に着目することが、モノモーダル視線インタフェースに関する研究における活路だと筆者は考える。

近年視線でインタラクションを実行する環境が二次元環境から三次元環境に移りつつある。筆者はこの三次元環境でしか得られない視線データを活用することが、次の視線入力手法の開発につながると考えている。三次元環境における二次元環境とは異なる目に関する特徴的な性質は(1) 頭部の特定の位置に存在する、(2) 目が二つある、(3) 選択対象の種類が異なる、などが挙げられる。まず(1)はHMDによって頭部方向を推定することが可能になったことによって得られた性質である。Kuiper BeltやAsteroid Gazerは頭部方向と視線方向を考慮した手法であり、三次元環境だからこそ使用することができる手法だと言える。(2)は右目と左目の視線データを用いることで視線の奥行きを推定可能であるという性質である。これを用いることでユーザの三次元的な焦点を明らかにし、より複雑な入力を実行できる可能性がある。(3)はGUIのアイコンなど選択対象が限られる二次元環境とは異なり、三次元環境では実オブジェクトや仮想オブジェクト、形状が大きく異なるということを指す。幾何学的な単純形状以外の形状を選択することが想定されるため、それに応じた視線選択の工夫が必要となる。また遮蔽によるターゲットサイズの減少、という二次元環境とは異なる現象が確認されるため、それに応じた対応も必要となる。これらの要素を代表とした、三次元的なモノモーダル視線入力はまだまだ研究余地が大きく、本研究の一部はこの要素に則った研究である。本研究で取り組んだ研究課題に加え、これらの要素に関する研究を行うことは視線入力のポテンシャルを引き出す上で必須だと言える。

#### 6.2.4 視線モダリティの展望

本稿ではモノモーダル視線インタフェースに関して研究を行ってきた。一方で視線以外のモダリティと視線を組み合わせるマルチモーダル視線インタフェースに関する研究が行われており、現状はモノモーダル視線インタフェースよりも検討されている。なぜならモノモーダル視線インタフェースは現状手を用いたインタフェースよりも劣った性能であるが、マルチモーダル視線インタフェースは手を用いたインタフェースよりも状況によっては高性能である場合が観測されているためである。これはモノモーダル視線インタフェースに研究の価値がないことを示しているわけではないが、モノモーダル視線インタフェースが手に次ぐ「第二のインタフェース」とならない場合においても、視線は三次元環境における重要なモダリティである可能性が高いことを示している。現在多くの視線入力研究は、視線モダリティとその他のモダリティを組み合わせた手法の提案を行なっている。手 [99] や頭 [175]、筋電 [187] など様々なモダリティを用いた手法が提案されており、それらの手法では概ね視線がポインティングの動作を担っている。視線は人体において最速の挙動を行うことができるモダリティであり、ターゲットまで到達する速度は手よりも速い。ゆえに視線はポインティング操作においては非常に優秀なモダリティであるため、モノモーダル視線インタフェースが定着しなくとも視線モダリティは今後広く扱われることだろう。そして本研究で検討した手法の一つである Bubble Gaze Cursor は暗黙的にターゲットの大きさを拡大するため、マルチモーダル視線インタフェースにおける視線ポインティングの性能を向上させることが可能だと言える。したがって本研究は視線インタフェースの社会実装を大きく推進する研究だと言える。

#### 6.2.5 長期的な未来における入力インタフェースと視線インタフェース

上記ではスマートフォンと同程度に HMD が普及するような中程度先の未来を想定し、現状の HCI 領域における入力手法に関する研究を参考とした展望の予想を行った。しかし長期的な未来予測をするならば、脳波を使用したブレインマシンインタフェース (Brain Machine Interface; BMI) や、汎用人工知能 (Artificial General Intelligence; AGI) による圧倒的な表現力を有した入力手法の出現も想定され得る。特に AGI は LLM の登場によって現実的に登場し得る存在となりつつあり、HCI の人とコンピュータのインタラクションで閉じていたループに AGI が挿入される可能性が高いと考えられる。つまり人とコンピュータの間に AGI が挿入され、人と AGI が対話を行い、AGI が解釈した操作をコンピュータに与えるようなループになる可能性がある。AGI は多くの人間よりも賢く、より適切な判断を下すことができるとすると、人間が厳密な意思決定を行うことは不合理だと考えられる。ゆえに環境の情報とユーザの大まかな意図を AGI に即座に伝えることが、



今後のユーザとのインタラクションの形になることが予想される。

AGI に対する情報提示として、視線データは非常に有益な情報だと考えられる。人間の目は脳に直結した器官であり、様々な情報を目から推察することができる。ユーザが何を見ているか、現在の精神的負荷の大きさは [32]、集中しているか [66]、視線の先のオブジェクトを選択したいのか [74]、など、様々な情報を視線データから得ることができる。また AGI とインタラクションを行う入力インタフェースの未来を予想させるような研究が発表されている。Lee らは音声認識した文章を LLM に渡すことで、視線の先のオブジェクトを参照しながら LLM と日常生活中に音声で対話する手法を提案した [104]。つまり LLM に対してユーザの視界情報と発話内容を受け渡すインタラクションを行っており、AGI とのインタラクションの形の一種を想像させる。現在は音声入力による対話が検討されているが、これが BMI を用いた対話が可能になると、人間は発話せずに念じるだけであらゆる入力が可能になると考えられる。そしてこれらの入力方法において、視線データは重要な役割を果たす可能性が高い。

上記のような未来におけるポインティング動作は、(1) インタラクションを行いたい対象の決定、(2) 対象に対する操作の決定、で構成されると考えられる。視線データはこのうち (1) を担い、表現力の高い音声や脳波入力が (2) を担うと考えられる。その際、視線インタフェースに要求されることは「いかに高速にインタラクション対象を決定できるか」である。対象がオブジェクトであるならば、オブジェクトサイズや距離などを問わずに推定できるかが問われる。またインタラクション対象はオブジェクトに限らず、オブジェクトの塊 (e.g. サンドウィッチの中のハムではなくサンドウィッチ) や概念 (e.g. 服ではなく、服の色やブランド情報) なども対象になり得るため、それを高速に推定することが要求される。ゆえに単純にターゲットを選択するツールであった視線インタフェースを、より高次元な情報を推定できるよう拡張することが求められると筆者は考えている。

## 第7章 結論

本論文ではモノモーダル視線インタフェースが抱える課題を解決し、視線入力の入力精度を向上させる手法を提案した。視線インタフェースが抱える入力精度に関わる課題として「小さなオブジェクトの選択が難しい」「意図しない入力 (Midas Touch) が実行される」が挙げられる。本研究ではこれらの課題を解決することを目的とし、3章では二次元環境におけるオブジェクトサイズの小さなオブジェクトの選択手法を、4章では三次元環境における視覚探索時の Midas Touch の回避手法を、5章では三次元環境における遮蔽によって可視領域が小さなオブジェクトの選択手法を提案した。

第3章ではボロノイ領域と同等にオブジェクトを拡大する手法を視線インタフェースに導入し、オブジェクトサイズが小さなオブジェクトを容易に選択する手法を提案した。本章ではボロノイ領域にオブジェクトを拡大する手法である Bubble Gaze Cursor と、視線がオブジェクトに近づくと拡大レンズが起動される Bubble Gaze Lens を実装した。実験でこれらの手法の性能を確認した結果、Bubble Gaze Cursor は通常の視線入力手法よりも小さなオブジェクトを正確に選択することができ、Bubble Gaze Lens は Bubble Gaze Cursor よりも密集した状況における小さなオブジェクトを正確に選択することができた。

第4章では日常では視線が向かないほど極端な視線角度（頭部方向と視線の角度）にメニューアイテムを配置することで、視覚探索時の Midas Touch を解消する手法を提案した。日常的に視線が存在する最大視線角度と目の最大稼働範囲にはバッファがあり、この領域には意識しない限り視線を向けることはない。本研究ではこの領域を Kuiper Belt と呼び、この領域にメニューアイテムを配置することで視覚探索時にメニューアイテムを誤選択しないと考えた。実験で視覚探索タスクを行った結果、Kuiper Belt にメニューアイテムを配置することで Midas Touch が大幅に削減されることが明らかになった。

第5章では Kuiper Belt に配置したメニューアイテムを活用することで、部分的に遮蔽されたオブジェクトを容易に選択する手法 Asteroid Gazer を提案した。本手法はメニューアイテムにターゲット候補を仮想的に再配置し、メニューアイテムを選択することで対応するターゲット候補を選択する手法である。さらに Kuiper Belt に階層メニューを配置することで、高密度な状況においても正確なターゲット選択を可能とした。遮蔽されたオブジェクトを選択するタスクを行った結果、提案手法は高密度な状況においても遮蔽されたオブジェクトを正確に選択することができた。

これらの研究から視線インタフェースの選択精度が向上し、より様々な場面で視線インタフェースを用いることが可能になった。ゆえに本論文は視線インタフェースの研究領域に対して大きく貢献することのできる知見が含まれていると考えている。

## 謝辞

この博士論文の研究を進めるにあたり、坂本大介准教授には長年にわたり多大な助言とご指導を賜りました。研究案の相談、実験計画の精査、論文の執筆に至るまで、研究の各段階において深い洞察と支援をいただき、心から感謝しております。また、小野哲雄教授からは、多角的な観点から貴重な指摘をいただき、研究の質の向上に大きく貢献していただきました。心より感謝申し上げます。

お二方が運営された研究室に6年以上にわたり在籍し、その間、研究室生活を大変楽しく、かつ研究に集中できる環境の中で過ごすことができました。これは、指導教員の先生方、そして研究室に所属していた学生の皆様の支えがあったからこそです。皆様に深く感謝の意を表します。最後に、学生生活および研究生活を支えてくれた、家族に最大の感謝を述べます。家族のサポート無しでは研究生活を営むことは出来ませんでした。心より感謝申し上げます。

## 参考文献

- [1] Richard Abadi and Columba Scallan. Ocular oscillation on eccentric gaze. *Vision research*, Vol. 41, pp. 2895–907, 11 2001.
- [2] Roland Alonso, Mickael Causse, François Vachon, Robert Parise, Frédéric Dehais, and Patrice Terrier. Evaluation of head-free eye tracking as an input device for air traffic control. *Ergonomics*, Vol. 56, pp. 246–255, 02 2013.
- [3] Michael Ashmore, Andrew T. Duchowski, and Garth Shoemaker. Efficient eye pointing with a fisheye lens. In *Proceedings of Graphics Interface 2005*, GI '05, pp. 203–210, School of Computer Science, University of Waterloo, Waterloo, Ontario, Canada, 2005. Canadian Human-Computer Communications Society.
- [4] Samantha Aziz and Oleg Komogortsev. An assessment of the eye tracking signal quality captured in the hololens 2. In *2022 Symposium on Eye Tracking Research and Applications*, ETRA '22, New York, NY, USA, 2022. Association for Computing Machinery.
- [5] A.Terry Bahill, Michael R. Clark, and Lawrence Stark. Dynamic overshoot in saccadic eye movements is caused by neurological control signal reversals. *Experimental Neurology*, Vol. 48, No. 1, pp. 107 – 122, 1975.
- [6] A.Terry Bahill and Lawrence Stark. Overlapping saccades and glissades are produced by fatigue in the saccadic eye movement system. *Experimental Neurology*, Vol. 48, No. 1, pp. 95 – 106, 1975.
- [7] Marc Baloup, Thomas Pietrzak, and Géry Casiez. Raycursor: A 3d pointing facilitation technique based on raycasting. In *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems*, CHI '19, p. 1–12, New York, NY, USA, 2019. Association for Computing Machinery.

- [8] T. Baltrusaitis, A. Zadeh, Y. C. Lim, and L. Morency. Openface 2.0: Facial behavior analysis toolkit. In *2018 13th IEEE International Conference on Automatic Face Gesture Recognition (FG 2018)*, pp. 59–66, 2018.
- [9] Michael Barz, Florian Daiber, Daniel Sonntag, and Andreas Bulling. Error-aware gaze-based interfaces for robust mobile gaze interaction. In *Proceedings of the 2018 ACM Symposium on Eye Tracking Research & Applications, ETRA '18*, pp. 24:1–24:10, New York, NY, USA, 2018. ACM.
- [10] Richard Bates and Howell Istance. Zooming interfaces!: Enhancing the performance of eye controlled pointing devices. In *Proceedings of the Fifth International ACM Conference on Assistive Technologies, Assets '02*, pp. 119–126, New York, NY, USA, 2002. ACM.
- [11] Patrick Baudisch, Alexander Zotov, Edward Cutrell, and Ken Hinckley. Starburst: A target expansion algorithm for non-uniform target distributions. In *Proceedings of the Working Conference on Advanced Visual Interfaces, AVI '08*, p. 129–137, New York, NY, USA, 2008. Association for Computing Machinery.
- [12] W. Becker and R. Jürgens. An analysis of the saccadic system by means of double step stimuli. *Vision Research*, Vol. 19, No. 9, pp. 967 – 983, 1979.
- [13] Wolfgang Becker and Albert F. Fuchs. Further properties of the human saccadic system: Eye movements and correction saccades with and without visual fixation points. *Vision Research*, Vol. 9, No. 10, pp. 1247 – 1258, 1969.
- [14] Jonas Blattgerste, Patrick Renner, and Thies Pfeiffer. Advantages of eye-gaze over head-gaze-based selection in virtual and augmented reality under varying field of views. In *Proceedings of the Workshop on Communication by Gaze Interaction, COGAIN '18*, New York, NY, USA, 2018. Association for Computing Machinery.
- [15] Sebastian Boring, Marko Jurmu, and Andreas Butz. Scroll, tilt or move it: Using mobile phones to continuously control pointers on large public displays. In *Proceedings of the 21st Annual Conference of the Australian Computer-Human Interaction Special Interest Group: Design: Open 24/7, OZCHI '09*, p. 161–168, New York, NY, USA, 2009. Association for Computing Machinery.

- [16] Assaf Botzer, Tomer Elbaum, and Michael Wagner. User tracking accuracy during smooth pursuit of a target with and without a crosshair. *ECCE '19*, p. 185–187, New York, NY, USA, 2019. Association for Computing Machinery.
- [17] John Brooke. Sus : A quick and dirty usability scale. *Usability Evaluation in Industry*, pp. 189–194, 1996.
- [18] Guy Thomas Buswell. How people look at pictures: a study of the psychology and perception in art. 1935.
- [19] Jeffrey Cashion, Chadwick Wingrave, and Joseph J. LaViola Jr. Dense and dynamic 3d selection for game-based virtual environments. *IEEE Transactions on Visualization and Computer Graphics*, Vol. 18, No. 4, pp. 634–642, 2012.
- [20] Olivier Chapuis, Jean-Baptiste Labrune, and Emmanuel Pietriga. Dynaspot: Speed-dependent area cursor. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '09, p. 1391–1400, New York, NY, USA, 2009. Association for Computing Machinery.
- [21] Di Laura Chen, Marcello Giordano, Hrvoje Benko, Tovi Grossman, and Stephanie Santosa. Gazeraycursor: Facilitating virtual reality target selection by blending gaze and controller raycasting. In *Proceedings of the 29th ACM Symposium on Virtual Reality Software and Technology*, VRST '23, New York, NY, USA, 2023. Association for Computing Machinery.
- [22] Jixu Chen and Qiang Ji. 3d gaze estimation with a single camera without ir illumination. In *2008 19th International Conference on Pattern Recognition*, pp. 1–4, 2008.
- [23] Jacky Cheung, Oscar Au, and Zhu Kening. *Additive Voronoi Cursor: Dynamic Effective Areas Using Additively Weighted Voronoi Diagrams*, pp. 273–292. 08 2019.
- [24] Tom N Cornsweet and Hewitt D Crane. Accurate two-dimensional eye tracker using first and fourth purkinje images. *JOSA*, Vol. 63, No. 8, pp. 921–928, 1973.
- [25] Nathan Cournia, John D. Smith, and Andrew T. Duchowski. Gaze- vs. hand-based pointing in virtual environments. In *CHI '03 Extended Abstracts on Human*

- Factors in Computing Systems*, CHI EA '03, p. 772–773, New York, NY, USA, 2003. Association for Computing Machinery.
- [26] Henrique Debarba, Jeronimo Grandi, Anderson Maciel, Luciana Nedel, and Ronan Boulic. Disambiguation canvas: A precise selection technique for virtual environments. In *Human-Computer Interaction – INTERACT 2013*, pp. 388–405, Berlin, Heidelberg, 2013. Springer Berlin Heidelberg.
- [27] William Delamare, Teng Han, and Pourang Irani. Designing a gaze gesture guiding system. In *Proceedings of the 19th International Conference on Human-Computer Interaction with Mobile Devices and Services*, MobileHCI '17, New York, NY, USA, 2017. Association for Computing Machinery.
- [28] Shujie Deng, Jian Chang, Shi-Min Hu, and Jian Jun Zhang. Gaze modulated disambiguation technique for gesture control in 3d virtual objects selection. In *2017 3rd IEEE International Conference on Cybernetics (CYBCONF)*, pp. 1–8, USA, 2017. IEEE.
- [29] Heiko Drewes. Only one fitts' law formula please! In *CHI '10 Extended Abstracts on Human Factors in Computing Systems*, CHI EA '10, p. 2813–2822, New York, NY, USA, 2010. Association for Computing Machinery.
- [30] Heiko Drewes, Mohamed Khamis, and Florian Alt. Dialplates: Enabling pursuits-based user interfaces with large target numbers. In *Proceedings of the 18th International Conference on Mobile and Ubiquitous Multimedia*, MUM '19, New York, NY, USA, 2019. Association for Computing Machinery.
- [31] Heiko Drewes and Albrecht Schmidt. Interacting with the computer using gaze gestures. In *Proceedings of the 11th IFIP TC 13 International Conference on Human-Computer Interaction - Volume Part II*, INTERACT'07, p. 475–488, Berlin, Heidelberg, 2007. Springer-Verlag.
- [32] Andrew T. Duchowski, Krzysztof Krejtz, Nina A. Gehrer, Tanya Bafna, and Per Bækgaard. The low/high index of pupillary activity. In *Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems*, CHI '20, p. 1–12, New York, NY, USA, 2020. Association for Computing Machinery.
- [33] Monika Elepfandt and Martin Grund. Move it there, or not? the design of voice commands for gaze with speech. In *Proceedings of the 4th Workshop on Eye Gaze*



- in Intelligent Human Machine Interaction, Gaze-In '12*, New York, NY, USA, 2012. Association for Computing Machinery.
- [34] Augusto Esteves, Eduardo Velloso, Andreas Bulling, and Hans Gellersen. Orbits: Gaze interaction for smart watches using smooth pursuit eye movements. In *Proceedings of the 28th Annual ACM Symposium on User Interface Software Technology*, UIST '15, p. 457–466, New York, NY, USA, 2015. Association for Computing Machinery.
- [35] Augusto Esteves, David Verweij, Liza Suraiya, Rasel Islam, Youryang Lee, and Ian Oakley. Smoothmoves: Smooth pursuits head movements for augmented reality. In *Proceedings of the 30th Annual ACM Symposium on User Interface Software and Technology*, UIST '17, p. 167–178, New York, NY, USA, 2017. Association for Computing Machinery.
- [36] Yu Fang, Ryoichi Nakashima, Kazumichi Matsumiya, Ichiro Kuriki, and Satoshi Shioiri. Eye-head coordination for visual cognitive processing. *PLOS ONE*, Vol. 10, p. 17, 03 2015.
- [37] Steven Feiner, Blair MacIntyre, Marcus Haupt, and Eliot Solomon. Windows on the world: 2d windows for 3d augmented reality. In *Proceedings of the 6th Annual ACM Symposium on User Interface Software and Technology*, UIST '93, p. 145–155, New York, NY, USA, 1993. Association for Computing Machinery.
- [38] Anna Maria Feit, Shane Williams, Arturo Toledo, Ann Paradiso, Harish Kulkarni, Shaun Kane, and Meredith Ringel Morris. Toward everyday gaze input: Accuracy and precision of eye tracking and implications for design. In *Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems*, CHI '17, pp. 1118–1130, New York, NY, USA, 2017. ACM.
- [39] Ajoy S. Fernandes, T. Scott Murdison, and Michael J. Proulx. Leveling the playing field: A comparative reevaluation of unmodified eye tracking as an input and interaction modality for vr. *IEEE Transactions on Visualization and Computer Graphics*, Vol. 29, No. 5, pp. 2269–2279, 2023.
- [40] Leah Findlater, Alex Jansen, Kristen Shinohara, Morgan Dixon, Peter Kamb, Joshua Rakita, and Jacob O. Wobbrock. Enhanced area cursors: Reducing fine pointing demands for people with motor impairments. In *Proceedings of the 23rd*

- Annual ACM Symposium on User Interface Software and Technology*, UIST '10, p. 153–162, New York, NY, USA, 2010. Association for Computing Machinery.
- [41] Paul M. Fitts. 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.
- [42] Margaret Foley, Géry Casiez, and Daniel Vogel. Comparing smartphone speech recognition and touchscreen typing for composition and transcription. In *Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems*, CHI '20, p. 1–11, New York, NY, USA, 2020. Association for Computing Machinery.
- [43] Tom Foulsham, Esther Walker, and Alan Kingstone. To where, what and when of gaze allocation in the lab and the natural environment. *Vision research*, Vol. 51, pp. 1920–31, 07 2011.
- [44] Eira Friström, Elias Lius, Niki Ulmanen, Paavo Hietala, Pauliina Kärkkäinen, Tommi Mäkinen, Stephan Sigg, and Rainhard Dieter Findling. Free-form gaze passwords from cameras embedded in smart glasses. In *Proceedings of the 17th International Conference on Advances in Mobile Computing & Multimedia*, MoMM2019, p. 136–144, New York, NY, USA, 2020. Association for Computing Machinery.
- [45] Hofferer J.M. Gauthier, G.M. Eye tracking of self-moved targets in the absence of vision. *Experimental Brain Research*, Vol. 26, pp. 121–139, 1976.
- [46] Julien Gori, Olivier Rioul, Yves Guiard, and Michel Beaudouin-Lafon. The perils of confounding factors: How fitts' law experiments can lead to false conclusions. In *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems*, CHI '18, p. 1–10, New York, NY, USA, 2018. Association for Computing Machinery.
- [47] Sven-Thomas Graupner and Sebastian Pannasch. Continuous gaze cursor feedback in various tasks: Influence on eye movement behavior, task performance and subjective distraction. Vol. 434, pp. 323–329, 06 2014.
- [48] Robert Greinacher and Jan-Niklas Voigt-Antons. Accuracy assessment of arkit 2 based gaze estimation. In *Human-Computer Interaction. Design and User Experi-*

- ence: Thematic Area, HCI 2020, Held as Part of the 22nd International Conference, HCII 2020, Copenhagen, Denmark, July 19–24, 2020, Proceedings, Part I*, p. 439–449, Berlin, Heidelberg, 2020. Springer-Verlag.
- [49] Tovi Grossman and Ravin Balakrishnan. The bubble cursor: Enhancing target acquisition by dynamic resizing of the cursor’s activation area. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI ’05, pp. 281–290, New York, NY, USA, 2005. ACM.
- [50] Tovi Grossman and Ravin Balakrishnan. The design and evaluation of selection techniques for 3d volumetric displays. In *Proceedings of the 19th Annual ACM Symposium on User Interface Software and Technology*, UIST ’06, p. 3–12, New York, NY, USA, 2006. Association for Computing Machinery.
- [51] Maxime Guillon, François Leitner, and Laurence Nigay. Investigating visual feed-forward for target expansion techniques. In *Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems*, CHI ’15, p. 2777–2786, New York, NY, USA, 2015. Association for Computing Machinery.
- [52] Dan Witzner Hansen, Henrik H. T. Skovsgaard, John Paulin Hansen, and Emilie Møllenbach. Noise tolerant selection by gaze-controlled pan and zoom in 3d. In *Proceedings of the 2008 Symposium on Eye Tracking Research & Applications*, ETRA ’08, pp. 205–212, New York, NY, USA, 2008. ACM.
- [53] Sandra Hart. Nasa-task load index (nasa-tlx); 20 years later. *Proceedings of the Human Factors and Ergonomics Society Annual Meeting*, Vol. 50, , 10 2006.
- [54] Sandra G. Hart and Lowell E. Staveland. Development of nasa-tlx (task load index): Results of empirical and theoretical research. In Peter A. Hancock and Najmedin Meshkati, editors, *Human Mental Workload*, Vol. 52 of *Advances in Psychology*, pp. 139 – 183. North-Holland, 1988.
- [55] Benjamin Hatscher, Maria Luz, Lennart E. Nacke, Norbert Elkmann, Veit Müller, and Christian Hansen. Gazetap: Towards hands-free interaction in the operating room. ICMI ’17, p. 243–251, New York, NY, USA, 2017. Association for Computing Machinery.
- [56] Ramin Hedesly, Chandan Kumar, Raphael Menges, and Steffen Staab. Hummer: Text entry by gaze and hum. In *Proceedings of the 2021 CHI Conference on Human*

- Factors in Computing Systems*, CHI '21, New York, NY, USA, 2021. Association for Computing Machinery.
- [57] David B. Henson. Corrective saccades: Effects of altering visual feedback. *Vision Research*, Vol. 18, No. 1, pp. 63 – 67, 1978.
- [58] James J. Higgins and Suleiman Tashtoush. An aligned rank transform test for interaction. *Nonlinear World*, Vol. 1, No. 2, pp. 201 – 211, 1994.
- [59] Juan David Hincapié-Ramos, Xiang Guo, Paymahn Moghadasian, and Pourang Irani. Consumed endurance: A metric to quantify arm fatigue of mid-air interactions. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI ' 14, p. 1063–1072, New York, NY, USA, 2014. Association for Computing Machinery.
- [60] Yuki Hirata, Hiroki Soma, Munehiro Takimoto, and Yasushi Kambayashi. Virtual space pointing based on vergence. In Masaaki Kurosu, editor, *Human-Computer Interaction. Recognition and Interaction Technologies*, pp. 259–269, Cham, 2019. Springer International Publishing.
- [61] Teresa Hirzle, Jan Gugenheimer, Florian Geiselhart, Andreas Bulling, and Enrico Rukzio. A design space for gaze interaction on head-mounted displays. CHI '19, p. 1–12, New York, NY, USA, 2019. Association for Computing Machinery.
- [62] Sture Holm. A simple sequentially rejective multiple test procedure. *Scandinavian Journal of Statistics*, Vol. 6, No. 2, pp. 65 – 70, 1979.
- [63] Anthony J. Hornof and Anna Cavender. Eyedraw: Enabling children with severe motor impairments to draw with their eyes. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '05, pp. 161–170, New York, NY, USA, 2005. ACM.
- [64] Anthony J. Hornof and Anna Cavender. Eyedraw: Enabling children with severe motor impairments to draw with their eyes. CHI '05, p. 161–170, New York, NY, USA, 2005. Association for Computing Machinery.
- [65] Zhiming. Hu, Sheng. Li, Congyi. Zhang, Kangrui. Yi, Guoping. Wang, and Dinesh. Manocha. Dgaze: Cnn-based gaze prediction in dynamic scenes. *IEEE Transactions on Visualization and Computer Graphics*, Vol. 26, No. 5, pp. 1902–1911, 2020.

- [66] Michael Xuelin Huang, Jiajia Li, Grace Ngai, Hong Va Leong, and Andreas Bulling. Moment-to-moment detection of internal thought during video viewing from eye vergence behavior. In *Proceedings of the 27th ACM International Conference on Multimedia*, MM '19, p. 2254–2262, New York, NY, USA, 2019. Association for Computing Machinery.
- [67] Edmund Burke Huey. The psychology and pedagogy of reading: With a review of the history of reading and writing and of methods, texts, and hygiene in reading. 1908.
- [68] Aulikki Hyrskykari, Howell Istance, and Stephen Vickers. Gaze gestures or dwell-based interaction? In *Proceedings of the Symposium on Eye Tracking Research and Applications*, ETRA '12, p. 229–232, New York, NY, USA, 2012. Association for Computing Machinery.
- [69] Ryo Ishibashi and Ikkaku Kawaguchi. Object manipulation method using eye gaze and hand-held controller in ar space. In *Proceedings of the 28th ACM Symposium on Virtual Reality Software and Technology*, VRST '22, New York, NY, USA, 2022. Association for Computing Machinery.
- [70] Yoshio Ishiguro and Jun Rekimoto. Peripheral vision annotation: Noninterference information presentation method for mobile augmented reality. In *Proceedings of the 2nd Augmented Human International Conference*, AH '11, New York, NY, USA, 2011. Association for Computing Machinery.
- [71] Takahiro Ishikawa, Simon Baker, Iain Matthews, and Takeo Kanade. Passive driver gaze tracking with active appearance models. *IEEE Transactions on Intelligent Transportation Systems - TITS*, 01 2004.
- [72] Poika Isokoski. Text input methods for eye trackers using off-screen targets. In *Proceedings of the 2000 Symposium on Eye Tracking Research Applications*, ETRA '00, p. 15–21, New York, NY, USA, 2000. Association for Computing Machinery.
- [73] Toshiya Isomoto, Shota Yamanaka, and Buntarou Shizuki. Gaze-based command activation technique robust against unintentional activation using dwell-then-gesture. In *Graphics Interface 2020*, 2020.

- [74] Toshiya Isomoto, Shota Yamanaka, and Buntarou Shizuki. Dwell selection with ml-based intent prediction using only gaze data. *Proc. ACM Interact. Mob. Wearable Ubiquitous Technol.*, Vol. 6, No. 3, sep 2022.
- [75] Howell Istance, Richard Bates, Aulikki Hyrskykari, and Stephen Vickers. Snap clutch, a moded approach to solving the midas touch problem. In *Proceedings of the 2008 Symposium on Eye Tracking Research & Applications*, ETRA '08, pp. 221–228, New York, NY, USA, 2008. ACM.
- [76] Howell Istance, Aulikki Hyrskykari, Lauri Immonen, Santtu Mansikkamaa, and Stephen Vickers. Designing gaze gestures for gaming: An investigation of performance. In *Proceedings of the 2010 Symposium on Eye-Tracking Research Applications*, ETRA '10, p. 323–330, New York, NY, USA, 2010. Association for Computing Machinery.
- [77] Howell Istance, Stephen Vickers, and Aulikki Hyrskykari. The validity of using non-representative users in gaze communication research. In *Proceedings of the Symposium on Eye Tracking Research and Applications*, ETRA '12, p. 233–236, New York, NY, USA, 2012. Association for Computing Machinery.
- [78] Robert J. K. Jacob. The use of eye movements in human-computer interaction techniques: What you look at is what you get. *ACM Trans. Inf. Syst.*, Vol. 9, No. 2, pp. 152–169, April 1991.
- [79] Robert J. K. Jacob. Eye movement-based human-computer interaction techniques: Toward non-command interfaces. In *IN ADVANCES IN HUMAN-COMPUTER INTERACTION*, pp. 151–190. Ablex Publishing Co, 1993.
- [80] Robert J. K. Jacob. *Eye Tracking in Advanced Interface Design*, p. 258–288. Oxford University Press, Inc., USA, 1995.
- [81] Shahram Jalaliniya, Diako Mardanbegi, and Thomas Pederson. Magic pointing for eyewear computers. In *Proceedings of the 2015 ACM International Symposium on Wearable Computers*, ISWC '15, p. 155–158, New York, NY, USA, 2015. Association for Computing Machinery.
- [82] Jorge Jimenez, Diego Gutiérrez, and Pedro Latorre. Gaze-based interaction for virtual environments. *Journal of Universal Computer Science*, Vol. 14, No. 19, pp. 3085–3098, nov 2008.

- [83] Qinjie Ju, René Chalon, and Stéphane Derrode. Assisted music score reading using fixed-gaze head movement: Empirical experiment and design implications. *Proc. ACM Hum.-Comput. Interact.*, Vol. 3, No. EICS, jun 2019.
- [84] Paul Kabbash and William A. S. Buxton. The "prince" technique: Fitts' law and selection using area cursors. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '95, pp. 273–279, New York, NY, USA, 1995. ACM Press/Addison-Wesley Publishing Co.
- [85] Jari Kangas, Deepak Akkil, Jussi Rantala, Poika Isokoski, Päivi Majaranta, and Roope Raisamo. Gaze gestures and haptic feedback in mobile devices. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '14, p. 435–438, New York, NY, USA, 2014. Association for Computing Machinery.
- [86] Robert V Kenyon. A soft contact lens search coil for measuring eye movements. *Vision research*, Vol. 25, No. 11, pp. 1629–1633, 1985.
- [87] Mohamed Khamis, Anita Baier, Niels Henze, Florian Alt, and Andreas Bulling. Understanding face and eye visibility in front-facing cameras of smartphones used in the wild. In *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems*, CHI '18, p. 1–12, New York, NY, USA, 2018. Association for Computing Machinery.
- [88] Mohamed Khamis, Carl Oechsner, Florian Alt, and Andreas Bulling. Vrpursuits: Interaction in virtual reality using smooth pursuit eye movements. In *Proceedings of the 2018 International Conference on Advanced Visual Interfaces*, AVI '18, New York, NY, USA, 2018. Association for Computing Machinery.
- [89] Mohamed Khamis, Ozan Saltuk, Alina Hang, Katharina Stolz, Andreas Bulling, and Florian Alt. Textpursuits: Using text for pursuits-based interaction and calibration on public displays. In *Proceedings of the 2016 ACM International Joint Conference on Pervasive and Ubiquitous Computing*, UbiComp '16, p. 274–285, New York, NY, USA, 2016. Association for Computing Machinery.
- [90] Jung-Hwa Kim, Seung-June Choi, and Jin-Woo Jeong. Watch & do: A smart iot interaction system with object detection and gaze estimation. *IEEE Trans. on Consum. Electron.*, Vol. 65, No. 2, p. 195–204, may 2019.

- [91] Dominik Kirst and Andreas Bulling. On the verge: Voluntary convergences for accurate and precise timing of gaze input. CHI EA '16, p. 1519–1525, New York, NY, USA, 2016. Association for Computing Machinery.
- [92] Konstantin Klamka, Andreas Siegel, Stefan Vogt, Fabian Göbel, Sophie Stellmach, and Raimund Dachselt. Look & pedal: Hands-free navigation in zoomable information spaces through gaze-supported foot input. In *Proceedings of the 2015 ACM on International Conference on Multimodal Interaction, ICMI '15*, pp. 123–130, New York, NY, USA, 2015. ACM.
- [93] Regis Kopper, Felipe Bacim, and Doug Bowman. Rapid and accurate 3d selection by progressive refinement. In *2011 IEEE Symposium on 3D User Interfaces (3DUI)*, pp. 67–74, USA, 03 2011. IEEE.
- [94] Shinya Kudo, Hiroyuki Okabe, Taku Hachisu, Michi Sato, Shogo Fukushima, and Hiroyuki Kajimoto. Input method using divergence eye movement. CHI EA '13, p. 1335–1340, New York, NY, USA, 2013. Association for Computing Machinery.
- [95] Manu Kumar, Jeff Klingner, Rohan Puranik, Terry Winograd, and Andreas Paepcke. Improving the accuracy of gaze input for interaction. In *Proceedings of the 2008 Symposium on Eye Tracking Research & Applications, ETRA '08*, pp. 65–68, New York, NY, USA, 2008. ACM.
- [96] Manu Kumar, Andreas Paepcke, Terry Winograd, and Terry Winograd. Eyepoint: Practical pointing and selection using gaze and keyboard. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, CHI '07*, pp. 421–430, New York, NY, USA, 2007. ACM.
- [97] Andrew Kurauchi, Wenxin Feng, Carlos Morimoto, and Margrit Betke. Hmagic: Head movement and gaze input cascaded pointing. In *Proceedings of the 8th ACM International Conference on PErvasive Technologies Related to Assistive Environments, PETRA '15*, New York, NY, USA, 2015. Association for Computing Machinery.
- [98] Chihiro Kuwabara, Keiko Yamamoto, Itaru Kuramoto, Yoshihiro Tsujino, and Mitsuru Minakuchi. Ghost-hunting: A cursor-based pointing technique with picture guide indication of the shortest path. In *Proceedings of the Companion Publication of the 2013 International Conference on Intelligent User Interfaces Com-*



- panion*, IUI '13 Companion, p. 85–86, New York, NY, USA, 2013. Association for Computing Machinery.
- [99] Mikko Kytö, Barrett Ens, Thammathip Piumsomboon, Gun A. Lee, and Mark Billingham. Pinpointing: Precise head- and eye-based target selection for augmented reality. In *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems*, CHI ' 18, New York, NY, USA, 2018. Association for Computing Machinery.
- [100] Jacob L. Orquin and Kenneth Holmqvist. Threats to the validity of eye-movement research in psychology. *Behavior Research Methods*, Vol. 50, , 10 2017.
- [101] Wallace Lages and Doug Bowman. Adjustable adaptation for spatial augmented reality workspaces. In *Symposium on Spatial User Interaction*, SUI '19, New York, NY, USA, 2019. Association for Computing Machinery.
- [102] Chris Lankford. Effective eye-gaze input into windows. In *Proceedings of the 2000 Symposium on Eye Tracking Research & Applications*, ETRA '00, pp. 23–27, New York, NY, USA, 2000. ACM.
- [103] Jae-Young Lee, Hyung-Min Park, Seok-Han Lee, Tae-Eun Kim, and Jong-Soo Choi. Design and implementation of an augmented reality system using gaze interaction. *Multimedia Tools and Applications*, Vol. 68, , 04 2011.
- [104] Jaewook Lee, Jun Wang, Elizabeth Brown, Liam Chu, Sebastian S. Rodriguez, and Jon E. Froehlich. Towards designing a context-aware multimodal voice assistant for pronoun disambiguation: A demonstration of gazepointer. In *Adjunct Proceedings of the 36th Annual ACM Symposium on User Interface Software and Technology*, UIST '23 Adjunct, New York, NY, USA, 2023. Association for Computing Machinery.
- [105] Joong-Jae Lee and Jung-Min Park. 3d mirrored object selection for occluded objects in virtual environments. *IEEE Access*, Vol. 8, pp. 200259–200274, 2020.
- [106] Jose Llanes-Jurado, Javier Marín-Morales, Jaime Guixeres, and Mariano Alcañiz. Development and calibration of an eye-tracking fixation identification algorithm for immersive virtual reality. *Sensors*, Vol. 20, No. 17, 2020.

- [107] Feiyu Lu, Shakiba Davari, Lee Lisle, Yuan Li, and Doug A. Bowman. Glanceable ar: Evaluating information access methods for head-worn augmented reality. In *2020 IEEE Symposium on 3D User Interfaces (3DUI)*, pp. 930–938, 2020.
- [108] Y. Lu, C. Yu, and Y. Shi. Investigating bubble mechanism for ray-casting to improve 3d target acquisition in virtual reality. In *2020 IEEE Conference on Virtual Reality and 3D User Interfaces (VR)*, pp. 35–43, 2020.
- [109] Christof Lutteroth, Moiz Penkar, and Gerald Weber. Gaze vs. mouse: A fast and accurate gaze-only click alternative. In *Proceedings of the 28th Annual ACM Symposium on User Interface Software & Technology, UIST '15*, p. 385–394, New York, NY, USA, 2015. Association for Computing Machinery.
- [110] Mathias N. Lystbæk, Ken Pfeuffer, Jens Emil Sloth Grønbæk, and Hans Gellersen. Exploring gaze for assisting freehand selection-based text entry in ar. *Proc. ACM Hum.-Comput. Interact.*, Vol. 6, No. ETRA, may 2022.
- [111] Mathias N. Lystbæk, Peter Rosenberg, Ken Pfeuffer, Jens Emil Grønbæk, and Hans Gellersen. Gaze-hand alignment: Combining eye gaze and mid-air pointing for interacting with menus in augmented reality. *Proc. ACM Hum.-Comput. Interact.*, Vol. 6, No. ETRA, may 2022.
- [112] David J. Mack, Sandro Belfanti, and Urs Schwarz. The effect of sampling rate and lowpass filters on saccades – a modeling approach. *Behavior Research Methods*, Vol. 49, No. 6, pp. 2146–2162, Dec 2017.
- [113] I. Scott MacKenzie. Fitts’ law as a research and design tool in human-computer interaction. *Hum.-Comput. Interact.*, Vol. 7, No. 1, pp. 91–139, March 1992.
- [114] I. Scott MacKenzie. An eye on input: Research challenges in using the eye for computer input control. In *Proceedings of the 2010 Symposium on Eye-Tracking Research & Applications, ETRA '10*, pp. 11–12, New York, NY, USA, 2010. ACM.
- [115] I. Scott MacKenzie, Abigail Sellen, and William A. S. Buxton. A comparison of input devices in element pointing and dragging tasks. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, CHI '91*, p. 161–166, New York, NY, USA, 1991. Association for Computing Machinery.

- [116] J. F. Mackworth and N. H. Mackworth. Eye fixations recorded on changing visual scenes by the television eye-marker. *J. Opt. Soc. Am.*, Vol. 48, No. 7, pp. 439–445, Jul 1958.
- [117] Päivi Majaranta, Ulla-Kaija Ahola, and Oleg Špakov. Fast gaze typing with an adjustable dwell time. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '09, p. 357–360, New York, NY, USA, 2009. Association for Computing Machinery.
- [118] Päivi Majaranta and Kari-Jouko Räihä. Twenty years of eye typing: Systems and design issues. In *Proceedings of the 2002 Symposium on Eye Tracking Research & Applications*, ETRA '02, pp. 15–22, New York, NY, USA, 2002. ACM.
- [119] Päivi Majaranta and Kari-Jouko Räihä. *Text Entry by Gaze: Utilizing Eye-Tracking*, pp. 175–187. 2007.
- [120] Diako Mardanbegi, Tobias Langlotz, and Hans Gellersen. Resolving target ambiguity in 3d gaze interaction through vor depth estimation. In *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems*, CHI '19, p. 1–12, New York, NY, USA, 2019. Association for Computing Machinery.
- [121] Diako Mardanbegi, Benedikt Mayer, Ken Pfeuffer, Shahram Jalaliniya, Hans Gellersen, and Alexander Perzl. Eyesethrough: Unifying tool selection and application in virtual environments. In *2019 IEEE Conference on Virtual Reality and 3D User Interfaces (VR)*, pp. 474–483, 2019.
- [122] Julio C. Mateo, Javier San Agustin, and John Paulin Hansen. Gaze beats mouse: Hands-free selection by combining gaze and emg. In *CHI '08 Extended Abstracts on Human Factors in Computing Systems*, CHI EA '08, pp. 3039–3044, New York, NY, USA, 2008. ACM.
- [123] Sven Mayer, Gierad Laput, and Chris Harrison. Enhancing mobile voice assistants with worldgaze. In *Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems*, CHI '20, p. 1–10, New York, NY, USA, 2020. Association for Computing Machinery.
- [124] Michael McGuffin and Ravin Balakrishnan. Acquisition of expanding targets. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '02, pp. 57–64, New York, NY, USA, 2002. ACM.

- [125] Raphael Menges, Chandan Kumar, Daniel Müller, and Korok Sengupta. Gazetheweb: A gaze-controlled web browser. In *Proceedings of the 14th Web for All Conference on The Future of Accessible Work, W4A '17*, pp. 25:1–25:2, New York, NY, USA, 2017. ACM.
- [126] Porterfield JL Merchant J, Morrissette R. Remote measurement of eye direction allowing subject motion over one cubic foot of space. In *IEEE Trans Biomed Eng*, Vol. 21, pp. 309–317, New York, NY, USA, 1974. Association for Computing Machinery.
- [127] Craig H. Meyer, Adrian G. Lasker, and David A. Robinson. The upper limit of human smooth pursuit velocity. *Vision Research*, Vol. 25, No. 4, pp. 561–563, 1985.
- [128] David E. Meyer, J. E. Keith-Smith, Sylvan Kornblum, Richard A. Abrams, and Charles E. Wright. Speed-accuracy tradeoffs in aimed movements: Toward a theory of rapid voluntary action. pp. 73–226, 1990.
- [129] Katsumi Minakata, John Paulin Hansen, I. Scott MacKenzie, Per Bækgaard, and Vijay Rajanna. Pointing by gaze, head, and foot in a head-mounted display. In *Proceedings of the 11th ACM Symposium on Eye Tracking Research & Applications, ETRA '19*, New York, NY, USA, 2019. Association for Computing Machinery.
- [130] Mark R. Mine. Virtual environment interaction techniques. Technical report, University of North Carolina at Chapel Hill, USA, 1995.
- [131] Darius Miniotos. Application of fitts' law to eye gaze interaction. In *CHI '00 Extended Abstracts on Human Factors in Computing Systems, CHI EA '00*, p. 339–340, New York, NY, USA, 2000. Association for Computing Machinery.
- [132] Darius Miniotos, Oleg Špakov, and I. Scott MacKenzie. Eye gaze interaction with expanding targets. In *CHI '04 Extended Abstracts on Human Factors in Computing Systems, CHI EA '04*, pp. 1255–1258, New York, NY, USA, 2004. ACM.
- [133] Darius Miniotos, Oleg Špakov, Ivan Tugoy, and I. Scott MacKenzie. Speech-augmented eye gaze interaction with small closely spaced targets. In *Proceedings of the 2006 Symposium on Eye Tracking Research & Applications, ETRA '06*, p. 67–72, New York, NY, USA, 2006. Association for Computing Machinery.

- [134] Emilie Mollenbach, John Paulin Hansen, Martin Lillholm, and Alastair G. Gale. Single stroke gaze gestures. In *CHI '09 Extended Abstracts on Human Factors in Computing Systems*, CHI EA '09, p. 4555–4560, New York, NY, USA, 2009. Association for Computing Machinery.
- [135] Pedro Monteiro, Guilherme Gonçalves, Bruno Peixoto, Miguel Melo, and Maximino Bessa. Evaluation of hands-free vr interaction methods during a fitts' task: Efficiency and effectiveness. *IEEE Access*, Vol. 11, pp. 70898–70911, 2023.
- [136] Carlos H. Morimoto, Jose A. T. Leyva, and Antonio Diaz-Tula. Context switching eye typing using dynamic expanding targets. In *Proceedings of the Workshop on Communication by Gaze Interaction*, COGAIN '18, pp. 6:1–6:9, New York, NY, USA, 2018. ACM.
- [137] Martez E. Mott and Jacob O. Wobbrock. Beating the bubble: Using kinematic triggering in the bubble lens for acquiring small, dense targets. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '14, pp. 733–742, New York, NY, USA, 2014. ACM.
- [138] Florian Müller, Joshua McManus, Sebastian Günther, Martin Schmitz, Max Mühlhäuser, and Markus Funk. Mind the tap: Assessing foot-taps for interacting with head-mounted displays. In *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems*, CHI '19, New York, NY, USA, 2019. Association for Computing Machinery.
- [139] Aunnoy Mutasim, Anil Ufuk Batmaz, Moaaz Hudhud Mughrabi, and Wolfgang Stuerzlinger. Performance analysis of saccades for primary and confirmatory target selection. In *Proceedings of the 28th ACM Symposium on Virtual Reality Software and Technology*, VRST '22, New York, NY, USA, 2022. Association for Computing Machinery.
- [140] Omar Namnakani, Yasmeen Abdrabou, Jonathan Grizou, Augusto Esteves, and Mohamed Khamis. Comparing dwell time, pursuits and gaze gestures for gaze interaction on handheld mobile devices. In *Proceedings of the 2023 CHI Conference on Human Factors in Computing Systems*, CHI '23, New York, NY, USA, 2023. Association for Computing Machinery.

- [141] Tomi Nukarinen, Jari Kangas, Oleg Špakov, Poika Isokoski, Deepak Akkil, Jussi Rantala, and Roope Raisamo. Evaluation of headturn: An interaction technique using the gaze and head turns. In *Proceedings of the 9th Nordic Conference on Human-Computer Interaction, NordiCHI '16*, New York, NY, USA, 2016. Association for Computing Machinery.
- [142] Marcus Nyström and Kenneth Holmqvist. An adaptive algorithm for fixation, saccade, and glissade detection in eyetracking data. *Behavior research methods*, Vol. 42, pp. 188–204, 02 2010.
- [143] Anneli Olsen and Ricardo Matos. Identifying parameter values for an i-vt fixation filter suitable for handling data sampled with various sampling frequencies. In *Proceedings of the Symposium on Eye Tracking Research and Applications, ETRA '12*, p. 317–320, New York, NY, USA, 2012. Association for Computing Machinery.
- [144] Yun Suen Pai, Benjamin Outram, Noriyasu Vontin, and Kai Kunze. Transparent reality: Using eye gaze focus depth as interaction modality. *UIST '16 Adjunct*, p. 171–172, New York, NY, USA, 2016. Association for Computing Machinery.
- [145] Seonwook Park, Xucong Zhang, Andreas Bulling, and Otmar Hilliges. Learning to find eye region landmarks for remote gaze estimation in unconstrained settings. In *Proceedings of the 2018 ACM Symposium on Eye Tracking Research Applications, ETRA '18*, New York, NY, USA, 2018. Association for Computing Machinery.
- [146] Abdul Moiz Penkar, Christof Lutteroth, and Gerald Weber. Designing for the eye: Design parameters for dwell in gaze interaction. In *Proceedings of the 24th Australian Computer-Human Interaction Conference, OzCHI '12*, p. 479–488, New York, NY, USA, 2012. Association for Computing Machinery.
- [147] Thies Pfeiffer, Marc Latoschik, and Wachsmuth Ipke. Evaluation of binocular eye trackers and algorithms for 3d gaze interaction in virtual reality environments. *Journal of Virtual Reality and Broadcasting*, Vol. 5, No. 16, p. 14, 03 2008.
- [148] Ken Pfeuffer, Yasmeeen Abdrabou, Augusto Esteves, Radiah Rivu, Yomna Abdelrahman, Stefanie Meitner, Amr Saadi, and Florian Alt. Attention: A design space for gaze-adaptive user interfaces in augmented reality. *Computers & Graphics*, Vol. 95, pp. 1–12, 2021.

- [149] Ken Pfeuffer, Jason Alexander, Ming Ki Chong, and Hans Gellersen. Gaze-touch: Combining gaze with multi-touch for interaction on the same surface. In *Proceedings of the 27th Annual ACM Symposium on User Interface Software and Technology*, UIST '14, pp. 509–518, New York, NY, USA, 2014. ACM.
- [150] Ken Pfeuffer and Hans Gellersen. Gaze and touch interaction on tablets. In *Proceedings of the 29th Annual Symposium on User Interface Software and Technology*, UIST '16, p. 301–311, New York, NY, USA, 2016. Association for Computing Machinery.
- [151] Ken Pfeuffer, Benedikt Mayer, Diako Mardanbegi, and Hans Gellersen. Gaze + pinch interaction in virtual reality. In *Proceedings of the 5th Symposium on Spatial User Interaction*, SUI '17, p. 99–108, New York, NY, USA, 2017. Association for Computing Machinery.
- [152] Ken Pfeuffer, Jan Obernolte, Felix Dietz, Ville Mäkelä, Ludwig Sidenmark, Pavel Manakhov, Minna Pakanen, and Florian Alt. Palmgazer: Unimanual eye-hand menus in augmented reality. In *Proceedings of the 2023 ACM Symposium on Spatial User Interaction*, SUI '23, New York, NY, USA, 2023. Association for Computing Machinery.
- [153] Ken Pfeuffer, Melodie Vidal, Jayson Turner, Andreas Bulling, and Hans Gellersen. Pursuit calibration: Making gaze calibration less tedious and more flexible. In *Proceedings of the 26th Annual ACM Symposium on User Interface Software and Technology*, UIST '13, p. 261–270, New York, NY, USA, 2013. Association for Computing Machinery.
- [154] Thammathip Piumsomboon, Gun Lee, Robert W. Lindeman, and Mark Billingham. Exploring natural eye-gaze-based interaction for immersive virtual reality. In *2017 IEEE Symposium on 3D User Interfaces (3DUI)*, 3DUI '17, pp. 36–39, USA, 2017. IEEE.
- [155] Marco Porta and Matteo Turina. Eye-s: A full-screen input modality for pure eye-based communication. New York, NY, USA, 2008. Association for Computing Machinery.
- [156] Claude Prablanc, Daniel Massé, and Jean F. Echallier. Error-correcting mechanisms in large saccades. *Vision Research*, Vol. 18, No. 5, pp. 557 – 560, 1978.

- [157] Yuan Yuan Qian and Robert J. Teather. The eyes don't have it: An empirical comparison of head-based and eye-based selection in virtual reality. In *Proceedings of the 5th Symposium on Spatial User Interaction, SUI '17*, p. 91–98, New York, NY, USA, 2017. Association for Computing Machinery.
- [158] Argenis Ramirez Ramirez Gomez, Christopher Clarke, Ludwig Sidenmark, and Hans Gellersen. Gaze+hold: Eyes-only direct manipulation with continuous gaze modulated by closure of one eye. In *ACM Symposium on Eye Tracking Research and Applications, ETRA '21 Full Papers*, New York, NY, USA, 2021. Association for Computing Machinery.
- [159] Håkon Raudsandmoen and Børge Rødsjø. Empirically based design guidelines for gaze interaction in windows 7. Master's thesis, Norwegian University of Science and Technology, 2012.
- [160] Gang Ren and Eamonn O'Neill. 3d selection with freehand gesture. *Computers & Graphics*, Vol. 37, No. 3, pp. 101–120, 2012.
- [161] D. A. Robinson. A method of measuring eye movement using a scleral search coil in a magnetic field. *IEEE Transactions on Bio-medical Electronics*, Vol. 10, No. 4, pp. 137–145, 1963.
- [162] Ronald Rosenfeld, Dan Olsen, and Alex Rudnicky. Universal speech interfaces. *Interactions*, Vol. 8, No. 6, p. 34–44, oct 2001.
- [163] Sherry Ruan, Jacob O. Wobbrock, Kenny Liou, Andrew Ng, and James A. Landay. Comparing speech and keyboard text entry for short messages in two languages on touchscreen phones. *Proc. ACM Interact. Mob. Wearable Ubiquitous Technol.*, Vol. 1, No. 4, jan 2018.
- [164] Kunhee Ryu, Joong-Jae Lee, and Jung-Min Park. Gg interaction: a gaze–grasp pose interaction for 3d virtual object selection. *Journal on Multimodal User Interfaces*, Vol. 13, No. 1, p. 383–393, 2019.
- [165] Houssein Saidi, Emmanuel Dubois, and Marcos Serrano. *HoloBar: Rapid Command Execution for Head-Worn AR Exploiting Around the Field-of-View Interaction*. Association for Computing Machinery, New York, NY, USA, 2021.



- [166] K. C. Salter and R. F Fawcett. The art test of interaction: a robust and powerful rank test of interaction in factorial models. *Communications in Statistics - Simulation and Computation*, Vol. 22, No. 1, pp. 137–153, 1993.
- [167] Dario D. Salvucci and Joseph H. Goldberg. Identifying fixations and saccades in eye-tracking protocols. In *Proceedings of the 2000 Symposium on Eye Tracking Research Applications*, ETRA '00, p. 71–78, New York, NY, USA, 2000. Association for Computing Machinery.
- [168] Simon Schenk, Marc Dreiser, Gerhard Rigoll, and Michael Dorr. Gazeeverywhere: Enabling gaze-only user interaction on an unmodified desktop pc in everyday scenarios. In *Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems*, CHI '17, pp. 3034–3044, New York, NY, USA, 2017. ACM.
- [169] Immo Schuetz, T. Scott Murdison, Kevin J. MacKenzie, and Marina Zannoli. An explanation of fitts' law-like performance in gaze-based selection tasks using a psychophysics approach. In *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems*, CHI '19, pp. 535:1–535:13, New York, NY, USA, 2019. ACM.
- [170] Tayyar Sen and Ted Megaw. The effects of task variables and prolonged performance on saccadic eye movement parameters. In Alastair G. Gale and Frank Johnson, editors, *Theoretical and Applied Aspects of Eye Movement Research*, Vol. 22 of *Advances in Psychology*, pp. 103 – 111. North-Holland, 1984.
- [171] Asma Shakil, Christof Lutteroth, and Gerald Weber. Codegazer: Making code navigation easy and natural with gaze input. In *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems*, CHI '19, p. 1–12, New York, NY, USA, 2019. Association for Computing Machinery.
- [172] Linda E. Sibert and Robert J. K. Jacob. Evaluation of eye gaze interaction. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '00, p. 281–288, New York, NY, USA, 2000. Association for Computing Machinery.
- [173] Ludwig Sidenmark, Christopher Clarke, Joshua Newn, Mathias N. Lystbæk, Ken Pfeufer, and Hans Gellersen. Vergence matching: Inferring attention to objects

- in 3d environments for gaze-assisted selection. CHI '23, p. 1–15, New York, NY, USA, 2023. Association for Computing Machinery.
- [174] Ludwig Sidenmark, Christopher Clarke, Xuesong Zhang, Jenny Phu, and Hans Gellersen. Outline pursuits: Gaze-assisted selection of occluded objects in virtual reality. In *Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems*, CHI '20, p. 1–13, New York, NY, USA, 2020. Association for Computing Machinery.
- [175] Ludwig Sidenmark and Hans Gellersen. Eye & head: Synergetic eye and head movement for gaze pointing and selection. In *Proceedings of the 32nd Annual ACM Symposium on User Interface Software and Technology*, UIST '19, p. 1161–1174, New York, NY, USA, 2019. Association for Computing Machinery.
- [176] Ludwig Sidenmark and Hans Gellersen. Eye, head and torso coordination during gaze shifts in virtual reality. *ACM Trans. Comput.-Hum. Interact.*, Vol. 27, No. 1, December 2019.
- [177] Ludwig Sidenmark, Dominic Potts, Bill Bapisch, and Hans Gellersen. Radi-eye: Hands-free radial interfaces for 3d interaction using gaze-activated head-crossing. In *Proceedings of the 2021 CHI Conference on Human Factors in Computing Systems*, CHI '21, New York, NY, USA, 2021. Association for Computing Machinery.
- [178] Ludwig Sidenmark, Franziska Prummer, Joshua Newn, and Hans Gellersen. Comparing gaze, head and controller selection of dynamically revealed targets in head-mounted displays. *IEEE Transactions on Visualization and Computer Graphics*, Vol. 29, No. 11, pp. 4740–4750, 2023.
- [179] Alexandra Sipatchin, Siegfried Wahl, and Katharina Rifai. Eye-tracking for clinical ophthalmology with virtual reality (vr): A case study of the htc vive pro eye's usability. *Healthcare*, Vol. 9, No. 2, 2021.
- [180] Henrik H.T. Skovsgaard, John Paulin Hansen, and Julio C. Mateo. How can tiny buttons be hit using gaze only? In *Proceedings of COGAIN 2008*, COGAIN '08, pp. 38–42, 2008.
- [181] Henrik Skovsgaard, Julio C. Mateo, John M. Flach, and John Paulin Hansen. Small-target selection with gaze alone. In *Proceedings of the 2010 Symposium on*

- Eye-Tracking Research & Applications*, ETRA '10, pp. 145–148, New York, NY, USA, 2010. ACM.
- [182] John Stahl. Eye-head coordination and the variation of eye-movement accuracy with orbital eccentricity. *Experimental brain research. Experimentelle Hirnforschung. Expérimentation cérébrale*, Vol. 136, pp. 200–10, 02 2001.
- [183] John S. Stahl. Amplitude of human head movements associated with horizontal saccades. *Experimental Brain Research*, Vol. 126, pp. 41–54, 1999.
- [184] Sophie Stellmach and Raimund Dachzelt. Look & touch: Gaze-supported target acquisition. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '12, p. 2981–2990, New York, NY, USA, 2012. Association for Computing Machinery.
- [185] Sophie Stellmach and Raimund Dachzelt. Still looking: Investigating seamless gaze-supported selection, positioning, and manipulation of distant targets. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '13, p. 285–294, New York, NY, USA, 2013. Association for Computing Machinery.
- [186] Sophie Stellmach, Sebastian Stober, Andreas Nürnberger, and Raimund Dachzelt. Designing gaze-supported multimodal interactions for the exploration of large image collections. In *Proceedings of the 1st Conference on Novel Gaze-Controlled Applications*, NGCA '11, New York, NY, USA, 2011. Association for Computing Machinery.
- [187] Veikko Surakka, Marko Illi, and Poika Isokoski. Gazing and frowning as a new human–computer interaction technique. *ACM Trans. Appl. Percept.*, Vol. 1, No. 1, p. 40–56, July 2004.
- [188] Kar-Han Tan, D.J. Kriegman, and N. Ahuja. Appearance-based eye gaze estimation. In *Sixth IEEE Workshop on Applications of Computer Vision, 2002. (WACV 2002). Proceedings.*, pp. 191–195, 2002.
- [189] Vildan Tanriverdi and Robert J. K. Jacob. Interacting with eye movements in virtual environments. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '00, p. 265–272, New York, NY, USA, 2000. Association for Computing Machinery.

- [190] Bernard Tursky. Recording of human eye movements. *Bioelectric Recording Techniques Part C*, 1974.
- [191] Roberto Valenti, Nicu Sebe, and Theo Gevers. Combining head pose and eye location information for gaze estimation. *IEEE Transactions on Image Processing*, Vol. 21, No. 2, pp. 802–815, 2012.
- [192] Eduardo Velloso, Flavio Luiz Coutinho, Andrew Kurauchi, and Carlos H Morimoto. Circular orbits detection for gaze interaction using 2d correlation and profile matching algorithms. In *Proceedings of the 2018 ACM Symposium on Eye Tracking Research Applications*, ETRA '18, New York, NY, USA, 2018. Association for Computing Machinery.
- [193] Eduardo Velloso, Markus Wirth, Christian Weichel, Augusto Esteves, and Hans Gellersen. Ambigaze: Direct control of ambient devices by gaze. In *Proceedings of the 2016 ACM Conference on Designing Interactive Systems*, DIS '16, p. 812–817, New York, NY, USA, 2016. Association for Computing Machinery.
- [194] Roel Vertegaal. A fitts law comparison of eye tracking and manual input in the selection of visual targets. In *Proceedings of the 10th International Conference on Multimodal Interfaces*, ICMI '08, p. 241–248, New York, NY, USA, 2008. Association for Computing Machinery.
- [195] Mélodie Vidal, Andreas Bulling, and Hans Gellersen. Pursuits: Spontaneous interaction with displays based on smooth pursuit eye movement and moving targets. In *Proceedings of the 2013 ACM International Joint Conference on Pervasive and Ubiquitous Computing*, UbiComp '13, p. 439–448, New York, NY, USA, 2013. Association for Computing Machinery.
- [196] Oleg Špakov. Comparison of eye movement filters used in hci. In *Proceedings of the Symposium on Eye Tracking Research and Applications*, ETRA '12, pp. 281–284, New York, NY, USA, 2012. ACM.
- [197] Oleg Špakov and Yulia Gizatdinova. Real-time hidden gaze point correction. In *Proceedings of the Symposium on Eye Tracking Research and Applications*, ETRA '14, p. 291–294, New York, NY, USA, 2014. Association for Computing Machinery.
- [198] Oleg Špakov, Poika Isokoski, and Päivi Majaranta. Look and lean: Accurate head-assisted eye pointing. In *Proceedings of the Symposium on Eye Tracking Research*

- and Applications*, ETRA '14, p. 35–42, New York, NY, USA, 2014. Association for Computing Machinery.
- [199] Oleg Špakov and Darius Miniotas. Gaze-based selection of standard-size menu items. In *Proceedings of the 7th International Conference on Multimodal Interfaces*, ICMI '05, pp. 124–128, New York, NY, USA, 2005. ACM.
- [200] Uta Wagner, Mathias N. Lystbæk, Pavel Manakhov, Jens Emil Sloth Grønbaek, Ken Pfeuffer, and Hans Gellersen. A fitts' law study of gaze-hand alignment for selection in 3d user interfaces. In *Proceedings of the 2023 CHI Conference on Human Factors in Computing Systems*, CHI '23, New York, NY, USA, 2023. Association for Computing Machinery.
- [201] Feng Wang, Xiang Cao, Xiangshi Ren, and Pourang Irani. Detecting and leveraging finger orientation for interaction with direct-touch surfaces. In *Proceedings of the 22nd Annual ACM Symposium on User Interface Software and Technology*, UIST '09, p. 23–32, New York, NY, USA, 2009. Association for Computing Machinery.
- [202] Xi Wang, Kenneth Holmqvist, and Marc Alexa. The mean point of vergence is biased under projection. *Journal of Eye Movement Research*, Vol. 12, , 11 2019.
- [203] Yunhan Wang and Regis Kopper. Efficient and accurate object 3d selection with eye tracking-based progressive refinement. *Frontiers in Virtual Reality*, Vol. 2, pp. 1–14, 2021.
- [204] Zhimin Wang, Haofei Wang, Huangyue Yu, and Feng Lu. Interaction with gaze, gesture, and speech in a flexibly configurable augmented reality system. *IEEE Transactions on Human-Machine Systems*, Vol. 51, No. 5, pp. 524–534, 2021.
- [205] Zhimin Wang, Yuxin Zhao, and Feng Lu. Gaze-vergence-controlled see-through vision in augmented reality. *IEEE Transactions on Visualization and Computer Graphics*, Vol. 28, No. 11, pp. 3843–3853, nov 2022.
- [206] Colin Ware and Harutune H. Mikaelian. An evaluation of an eye tracker as a device for computer input. *SIGCHI Bull.*, Vol. 17, No. SI, p. 183–188, May 1986.
- [207] Ronald B. Weber and Robert B. Daroff. The metrics of horizontal saccadic eye movements in normal humans. *Vision Research*, Vol. 11, No. 9, pp. 921 – IN2, 1971.

- [208] Ronald B. Weber and Robert B. Daroff. Corrective movements following refixation saccades: Type and control system analysis. *Vision Research*, Vol. 12, No. 3, pp. 467 – 475, 1972.
- [209] Mark Weiser. The computer for the 21 st century. *Scientific American*, Vol. 265, No. 3, pp. 94–105, 1991.
- [210] Roel Wierds, Maurice J. A. Janssen, and Herman Kingma. Measuring saccade peak velocity using a low-frequency sampling rate of 50 hz. *IEEE Transactions on Biomedical Engineering*, Vol. 55, No. 12, pp. 2840–2842, Dec 2008.
- [211] Jacob O. Wobbrock, Leah Findlater, Darren Gergle, and James J. Higgins. The aligned rank transform for nonparametric factorial analyses using only anova procedures. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '11, pp. 143–146, New York, NY, USA, 2011. ACM.
- [212] Jacob O. Wobbrock, James Rubinstein, Michael W. Sawyer, and Andrew T. Duchowski. Longitudinal evaluation of discrete consecutive gaze gestures for text entry. In *Proceedings of the 2008 Symposium on Eye Tracking Research Applications*, ETRA '08, p. 11–18, New York, NY, USA, 2008. Association for Computing Machinery.
- [213] Erroll Wood and Andreas Bulling. Eyetab: Model-based gaze estimation on unmodified tablet computers. In *Proceedings of the Symposium on Eye Tracking Research and Applications*, ETRA '14, pp. 207–210, New York, NY, USA, 2014. ACM.
- [214] Aileen Worden, Nef Walker, Krishna Bharat, and Scott Hudson. Making computers easier for older adults to use: Area cursors and sticky icons. In *Proceedings of the ACM SIGCHI Conference on Human Factors in Computing Systems*, CHI '97, pp. 266–271, New York, NY, USA, 1997. ACM.
- [215] Hans Peter Wyss, Roland Blach, and Matthias Bues. Isith - intersection-based spatial interaction for two hands. In *Proceedings of the IEEE Conference on Virtual Reality*, VR '06, p. 123, USA, 2006. IEEE Computer Society.
- [216] Minpeng Xu, Jin Han, Yijun Wang, Tzyy-Ping Jung, and Dong Ming. Implementing over 100 command codes for a high-speed hybrid brain-computer inter-

- face using concurrent p300 and ssvp features. *IEEE Transactions on Biomedical Engineering*, Vol. 67, No. 11, pp. 3073–3082, 2020.
- [217] Wenge Xu, Hai-Ning Liang, Yuxuan Zhao, Difeng Yu, and Diego Monteiro. Dmove: Directional motion-based interaction for augmented reality head-mounted displays. In *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems*, CHI '19, New York, NY, USA, 2019. Association for Computing Machinery.
- [218] Lijing Yao and Caror K. Peck. *Saccadic eye movements to visual and auditory targets*, pp. 25–34. No. 115. 1997.
- [219] AL Yarbus. Eye movements and vision. 1967. *New York*, Vol. 101, , 1967.
- [220] Xin Yi, Yiqin Lu, Ziyin Cai, Zihan Wu, Yuntao Wang, and Yuanchun Shi. Gaze-dock: Gaze-only menu selection in virtual reality using auto-triggering peripheral menu. In *2022 IEEE Conference on Virtual Reality and 3D User Interfaces (VR)*, pp. 832–842, USA, 2022. IEEE.
- [221] Xin Yi, Leping Qiu, Wenjing Tang, Yehan Fan, Hewu Li, and Yuanchun Shi. Deep: 3d gaze pointing in virtual reality leveraging eyelid movement. In *Proceedings of the 35th Annual ACM Symposium on User Interface Software and Technology*, UIST '22, New York, NY, USA, 2022. Association for Computing Machinery.
- [222] Difeng Yu, Hai-Ning Liang, Xueshi Lu, Kaixuan Fan, and Barrett Ens. Modeling endpoint distribution of pointing selection tasks in virtual reality environments. *ACM Trans. Graph.*, Vol. 38, No. 6, November 2019.
- [223] Difeng Yu, Xueshi Lu, Rongkai Shi, Hai-Ning Liang, Tilman Dingler, Eduardo Velloso, and Jorge Goncalves. Gaze-supported 3d object manipulation in virtual reality. In *Proceedings of the 2021 CHI Conference on Human Factors in Computing Systems*, CHI '21, New York, NY, USA, 2021. Association for Computing Machinery.
- [224] Difeng Yu, Qiushi Zhou, Joshua Newn, Tilman Dingler, Eduardo Velloso, and Jorge Goncalves. Fully-occluded target selection in virtual reality. *IEEE Transactions on Visualization and Computer Graphics*, Vol. 26, No. 12, pp. 3402–3413, 2020.

- [225] Shumin Zhai, Stéphane Conversy, Michel Beaudouin-Lafon, and Yves Guiard. Human on-line response to target expansion. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '03, pp. 177–184, New York, NY, USA, 2003. ACM.
- [226] Shumin Zhai, Carlos Morimoto, and Steven Ihde. Manual and gaze input cascaded (magic) pointing. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '99, pp. 246–253, New York, NY, USA, 1999. ACM.
- [227] Chenyang Zhang, Tiansu Chen, Rohan Russel Nedungadi, Eric Shaffer, and Elahe Soltanaghai. Focusflow: Leveraging focal depth for gaze interaction in virtual reality. In *Adjunct Proceedings of the 36th Annual ACM Symposium on User Interface Software and Technology*, UIST '23 Adjunct, New York, NY, USA, 2023. Association for Computing Machinery.
- [228] Futian Zhang, Keiko Katsuragawa, and Edward Lank. Conductor: Intersection-based bimanual pointing in augmented and virtual reality. *Proc. ACM Hum.-Comput. Interact.*, Vol. 6, No. ISS, nov 2022.
- [229] X. Zhang, Y. Sugano, M. Fritz, and A. Bulling. Mpiigaze: Real-world dataset and deep appearance-based gaze estimation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 41, No. 1, pp. 162–175, 2019.
- [230] Xinyong Zhang, Wenxin Feng, and Hongbin Zha. Effects of different visual feedback forms on eye cursor's stabilities. In *Proceedings of the 4th International Conference on Internationalization, Design and Global Development*, IDGD'11, pp. 273–282, Berlin, Heidelberg, 2011. Springer-Verlag.
- [231] Xinyong Zhang, Xiangshi Ren, and Hongbin Zha. Improving eye cursor's stability for eye pointing tasks. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '08, pp. 525–534, New York, NY, USA, 2008. ACM.
- [232] Xinyong Zhang, Xiangshi Ren, and Hongbin Zha. Modeling dwell-based eye pointing target acquisition. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '10, p. 2083–2092, New York, NY, USA, 2010. Association for Computing Machinery.
- [233] Xinyong Zhang, Pianpian Xu, Qing Zhang, and Hongbin Zha. Speed-accuracy trade-off in dwell-based eye pointing tasks at different cognitive levels. In *Pro-*



- ceedings of the 1st International Workshop on Pervasive Eye Tracking & Mobile Eye-based Interaction*, PETMEI '11, pp. 37–42, New York, NY, USA, 2011. ACM.
- [234] Xucong Zhang, Seonwook Park, Thabo Beeler, Derek Bradley, Siyu Tang, and Otmar Hilliges. Eth-xgaze: A large scale dataset for gaze estimation under extreme head pose and gaze variation. In Andrea Vedaldi, Horst Bischof, Thomas Brox, and Jan-Michael Frahm, editors, *Computer Vision – ECCV 2020*, pp. 365–381, Cham, 2020. Springer International Publishing.
- [235] Xucong Zhang, Yusuke Sugano, and Andreas Bulling. Evaluation of appearance-based methods and implications for gaze-based applications. In *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems*, CHI '19, p. 1–13, New York, NY, USA, 2019. Association for Computing Machinery.
- [236] Yanxia Zhang, Jörg Müller, Ming Ki Chong, Andreas Bulling, and Hans Gellersen. Gazehorizon: Enabling passers-by to interact with public displays by gaze. In *Proceedings of the 2014 ACM International Joint Conference on Pervasive and Ubiquitous Computing*, UbiComp '14, p. 559–563, New York, NY, USA, 2014. Association for Computing Machinery.

## 付録 A 研究業績リスト

### 学術論文誌（査読あり）

- [1] 崔 明根, 坂本 大介, 小野 哲雄. Kuiper Belt: VRにおける自然ではない視線角度を用いた視線入力手法の提案. 情報処理学会論文誌, 情報処理学会, Vol.64 (2), pp.400-416. 2023. (推薦論文)
- [2] 阿部 優樹, 崔 明根, 坂本 大介, 小野 哲雄. ストリームライブチャット入力を想定した半透明ダブルフリックキーボードの検証. 情報処理学会論文誌, 情報処理学会, Vol.64 (2), pp.352-365. 2023.
- [3] 秋葉 翔太, 崔 明根, 坂本 大介, 小野 哲雄. 選択ターゲット候補の半円状再配置によるスマートフォンでの片手選択操作手法の提案. 情報処理学会論文誌, 情報処理学会, Vol.62 (2), pp.689-700. 2021.
- [4] 崔 明根, 坂本 大介, 小野 哲雄. Bubble Gaze Lens: バブルレンズ法の視線操作への適用. 情報処理学会論文誌, 情報処理学会, Vol.62 (2), pp.667-679. 2021.
- [5] 崔 明根, 坂本 大介, 小野 哲雄. Bubble Gaze Cursor: バブルカーソル法の視線操作への適用. 情報処理学会論文誌, 情報処理学会, Vol.61 (2), pp.221-232. 2020.

### 国際会議における口頭発表（査読あり）

- [1] Myunguen Choi, Daisuke Sakamoto, and Tetsuo Ono. Kuiper Belt: Utilizing the “Out-of-natural Angle” Region in the Eye-gaze Interaction for Virtual Reality. In CHI Conference on Human Factors in Computing Systems (CHI '22). Association for Computing Machinery, New York, NY, USA, Article 357, pp. 1–17, May, 2022.
- [2] Myunguen Choi, Daisuke Sakamoto, and Tetsuo Ono. Bubble Gaze Cursor + Bubble Gaze Lens: Applying Area Cursor Technique to Eye-Gaze Interface. In Symposium on Eye Tracking Research and Applications (ETRA '20). Association for Computing Machinery, Article 11, pp.1–10, June, 2020.

## 国内会議における口頭発表（査読あり）

- [1] 岩井 望, 崔 明根, 坂本 大介, 小野 哲雄. AR 入力デバイスとしての身体装着型トラックボールの検討. インタラクシオン 2024, 情報処理学会. (to appear)
- [2] 阿部 優樹, 崔 明根, 坂本 大介, 小野 哲雄. ストリームライブチャット入力を想定した半透明ダブルフリックキーボードの入力性能の実験的検証. インタラクシオン 2022, 情報処理学会, INT22014, pp.125-134, 2022 年 2 月.
- [3] 崔 明根, 坂本 大介, 小野 哲雄. Kuiper Belt: バーチャルリアリティにおける極端な視線角度を用いた視線入力手法の検討. インタラクシオン 2022, 情報処理学会, INT22001, pp.1-10, 2022 年 3 月.
- [4] 崔 明根, 坂本 大介, 小野 哲雄. Bubble Gaze Cursor. インタラクシオン 2019, 情報処理学会, INT19003, pp.20-29, 2019 年 3 月.

## 国内会議における口頭発表（査読なし）

- [1] 崔 明根, 坂本 大介, 小野 哲雄. Asteroid Gazer: VR における遮蔽されたオブジェクトの視線選択手法の提案. 情報処理学会研究報告ヒューマンコンピュータインタラクシオン, 2023-HCI-203 (32), pp.1-8, 2023 年 5 月.
- [2] 崔 明根, 坂本 大介, 小野 哲雄. Depth Gaze Cursor: 特定の深度に固定された視線カーソルへの注視入力手法の検討. 情報処理学会研究報告ヒューマンコンピュータインタラクシオン, 2023-HCI-203 (20), pp.1-8, 2023 年 5 月.
- [3] 青木 美春, 崔 明根, 坂本 大介, 小野 哲雄. アルゴリズム嫌悪を引き起こすバーチャルエージェントの要因に関する調査. 情報処理学会研究報告ヒューマンコンピュータインタラクシオン, 2022-HCI-200 (26), pp.1-8, 2022 年 11 月.
- [4] 小柳 元志郎, 崔 明根, 坂本 大介, 小野 哲雄. バーチャルジョイスティックを用いたハンドヘルド AR でのレイ操作手法のユーザビリティ調査. 情報処理学会研究報告ヒューマンコンピュータインタラクシオン, 2022-HCI-200 (7), pp.1-8, 2022 年 11 月.
- [5] 若杉 直生, 崔 明根, 坂本 大介, 小野 哲雄. VR を用いた低照度環境における MR 支援の有効性の調査. 情報処理学会研究報告ヒューマンコンピュータインタラクシオン, 2022-HCI-200 (5), pp.1-8, 2022 年 11 月.

- [6] 高松 大悟, 崔 明根, 坂本 大介, 小野 哲雄. バーチャルジョイスティックを用いたハンドヘルド AR でのレイ操作手法の提案. 情報処理学会研究報告ヒューマンコンピュータインタラクション, 2022-HCI-197 (49), pp.1-8, 2022 年 3 月.
- [7] 日下部 完, 崔 明根, 坂本 大介, 小野 哲雄. 無段階調整インタフェースのためのハンドジェスチャによる操作手法の探索的研究. 情報処理学会研究報告ヒューマンコンピュータインタラクション, 2022-HCI-197 (26), pp.1-8, 2022 年 3 月.
- [8] 小柳 元志郎, 崔 明根, 坂本 大介, 小野 哲雄. スマートフォンの左右回転操作によるスライダ操作手法の検討. 情報処理学会研究報告ヒューマンコンピュータインタラクション, 2022-HCI-197 (23), pp.1-7, 2022 年 3 月.
- [9] 崔 明根, 坂本 大介, 小野 哲雄. Bubble Gaze Lens: バブルレンズ法の視線操作への拡張. 情報処理学会研究報告ヒューマンコンピュータインタラクション, 2019-HCI-185 (24), pp.1-8, 2019 年 12 月.

## その他

- [1] 小林 広夢, 崔 明根, 坂本 大介, 小野 哲雄. VR リダイレクションを用いた座位姿勢改善手法における閾値調査. インタラクション 2024, 情報処理学会. プレミアム発表. (to appear)
- [2] 崔 明根. CHI2022 採択への軌跡. 情報処理学会研究報告ヒューマンコンピュータインタラクション, 2022-HCI-199 (41), pp.1-1, 2022 年 8 月. 学生招待講演.
- [3] 秋葉 翔太, 崔 明根, 坂本 大介, 小野 哲雄. 選択ターゲット候補の半円状再配置によるスマートフォンでの片手選択操作手法の提案. インタラクション 2021, 情報処理学会, 2021 年 3 月. 招待発表.
- [4] 秋葉 翔太, 崔 明根, 坂本 大介, 小野 哲雄. 選択ターゲット候補の半円状再配置によるスマートフォンでの片手選択操作手法の提案. インタラクション 2020, 情報処理学会, 3B35, pp.914-919, 2020 年 3 月. プレミアム発表.

## 受賞等

- [1] 2022 年度情報処理学会論文誌ジャーナル/JIP 特選論文
- [2] 2022 年度 (令和 4 年度) 情報処理学会山下記念研究賞

- [3] 第 26 回一般社団法人情報処理学会シンポジウム インタラクシヨ ン 2022 論文賞
- [4] 2020 年度 北海道大学大学院情報科学院情報理工学専攻修士課程 学院長賞
- [5] 第 185 回情報処理学会ヒューマンコンピュータインタラクシヨ ン研究会 学生奨励賞

# 目 次

1.1	左図は Weiser らの Ubiquitous Computing のイメージ [209]. 人々の手元にコンピュータが把持されており, コンピュータの偏在化を予想している. 右図は Mixed Reality のイメージ <sup>1</sup> . 仮想オブジェクトがインタラクション対象となっている. . . . .	2
1.2	本論文で行なった研究の概要. 3-5 章にてそれぞれの課題の解決を試みる.	7
2.1	角膜反射法のイメージ <sup>2</sup> . . . . .	12
2.2	機械学習による視線推定手法の例 [229]. . . . .	13
2.3	Drews らの提案した視線ジェスチャの例 [31]. . . . .	14
2.4	Smooth Pursuit のイメージ [195]. . . . .	16
3.1	(a) Bubble Gaze Cursor, (b) Bubble Gaze Lens のイメージ. . . . .	24
3.2	ディスプレイの位置ごとの適切なターゲットサイズ [38]. . . . .	26
3.3	Ashmore らの魚眼レンズによるターゲット拡大手法 [3]. . . . .	28
3.4	Špakov らのメニューの拡大手法 [199]. . . . .	29
3.5	(a) ボロノイ領域によるターゲット拡張と Bubble Cursor. オブジェクトを一つだけ含むように, つまりカーソルに最も近いオブジェクトのみを含むように, エリアカーソルの大きさが設定されている. (b) ターゲットの幅 (Target Width; TW) と Bubble Cursor によって拡大されるターゲットの実質的な幅 (Effective Width; EW). . . . .	31
3.6	(a) Bubble Lens の拡大レンズが起動する直前の様子, (b) 拡大レンズが起動した画面状態 [137]. . . . .	31
3.7	比較手法の図示. . . . .	34
3.8	独立変数 <i>Amplitude</i> , <i>TW</i> , <i>EW</i> の図示. . . . .	35
3.9	(a) <i>Method</i> × <i>Amplitude</i> , (b) <i>Method</i> × <i>TW</i> , (c) <i>Method</i> × <i>EW</i> , (d) <i>Amplitude</i> × <i>Method</i> , (e) <i>TW</i> × <i>Method</i> , (f) <i>EW</i> × <i>Method</i> , (g) <i>Method</i> × <i>TW</i> × <i>EW</i> , (h) <i>Method</i> × ( <i>EW</i> - <i>TW</i> ) の選択時間. (g) の x 軸ラベルは ( <i>TW</i> , <i>EW</i> ) であり, (h) の x 軸ラベルは <i>EW</i> - <i>TW</i> ( <i>TW</i> , <i>EW</i> ) である. 有意差は ** <i>p</i> < .01.	38

3.10	(a) <i>Method</i> × <i>Amplitude</i> , (b) <i>Method</i> × <i>TW</i> , (c) <i>Method</i> × <i>EW</i> , (d) <i>Amplitude</i> × <i>Method</i> , (e) <i>TW</i> × <i>Method</i> , (f) <i>EW</i> × <i>Method</i> , (g) <i>Method</i> × <i>TW</i> × <i>EW</i> , (h) <i>Method</i> × ( <i>EW</i> - <i>TW</i> ) のエラー率. (g) の x 軸ラベルは ( <i>TW</i> , <i>EW</i> ) であり, (h) の x 軸ラベルは <i>EW</i> - <i>TW</i> ( <i>TW</i> , <i>EW</i> ) である. エラーバーは 95% 信頼区間. 有意差は * <i>p</i> < .05, ** <i>p</i> < .01. . . . .	39
3.11	<i>Method</i> ごとの (a) SUS, (b) NASA-TLX の結果. 有意差は * <i>p</i> < .05, ** <i>p</i> < .01. . . . .	40
3.12	Meyer らのポインティング動作のモデル [128]. . . . .	43
3.13	視線の速度プロファイルに対する Bubble Gaze Lens のアルゴリズムの概略. . . . .	45
3.14	(a) <i>Distance</i> , (b) <i>Spacing</i> , の図示. . . . .	47
3.15	実験時の様子. ターゲットは緑色の円であり, 偽ターゲットが灰色の円である. . . . .	48
3.16	(a) <i>Method</i> × <i>Distance</i> , (b) <i>Method</i> × <i>TW</i> , (c) <i>Method</i> × <i>Spacing</i> , (d) <i>Method</i> × <i>EW</i> の選択時間. 有意差は * <i>p</i> < .05, ** <i>p</i> < .01. . . . .	50
3.17	(a) <i>Method</i> × <i>Distance</i> , (b) <i>Method</i> × <i>TW</i> , (c) <i>Method</i> × <i>Spacing</i> , (d) <i>Method</i> × <i>EW</i> のエラー率. エラーバーは 95% 信頼区間. 全ての組み合わせに対して <i>p</i> < .01 の有意差が確認された. . . . .	51
3.18	(a) <i>Method</i> × <i>Distance</i> , (b) <i>Method</i> × <i>TW</i> , (c) <i>Method</i> × <i>Spacing</i> , (d) <i>Method</i> × <i>EW</i> の拡大レンズ起動率. エラーバーは 95% 信頼区間. 有意差は ** <i>p</i> < .01. . . . .	52
3.19	(a) <i>Method</i> × <i>Distance</i> , (b) <i>Method</i> × <i>TW</i> , (c) <i>Method</i> × <i>Spacing</i> , (d) <i>Method</i> × <i>EW</i> のターゲット非包括率. エラーバーは 95% 信頼区間. 有意差は * <i>p</i> < .05. . . . .	53
3.20	(a) <i>Method</i> × <i>Distance</i> , (b) <i>Method</i> × <i>TW</i> , (c) <i>Method</i> × <i>Spacing</i> , (d) <i>Method</i> × <i>EW</i> の拡大レンズが起動した際のエラー (Open) と拡大レンズが起動しなかった際のエラー (NotOpen) の積み上げ縦棒グラフ. . . . .	54
3.21	(a) <i>Method</i> × <i>Distance</i> , (b) <i>Method</i> × <i>TW</i> , (c) <i>Method</i> × <i>Spacing</i> , (d) <i>Method</i> × <i>EW</i> の拡大レンズが起動した際のエラー (Open) と拡大レンズが起動しなかった際のエラー (NotOpen) の割合. エラーバーは 95% 信頼区間. なお, このエラー率は条件ごとの試行に対するエラーの割合である. . . . .	55
3.22	<i>Method</i> ごとの (a) SUS, (b) NASA-TLX の結果. 有意差は * <i>p</i> < .05, ** <i>p</i> < .01. . . . .	55

4.1	VRにおける Kuiper Belt. 通常行われなほど大きな視線シフトを用いることで Midas Touch 問題を軽減する. . . . .	64
4.2	Kuiper Belt の定義領域. . . . .	64
4.3	Penkar らの実験で用いたアプリケーションの図 [146]. 左図ではボタン内にラベルが配置され, 右図ではボタン外にラベルが配置されている. . . . .	65
4.4	水平方向 (左図) と垂直方向 (右図) の頭部に対する視線の分布 [43]. . . . .	66
4.5	(a) 独立変数の定義, (b) 注視点の定義. . . . .	68
4.6	実験セットアップとフィードバック. . . . .	70
4.7	(a) $iDistance \times iDirection$ , (b) $iDistance$ , (c) $iDirection$ ごとのエラー率. エラーバーは 95%信頼区間. . . . .	71
4.8	(a) $iDistance \times iDirection$ , (b) $iDistance$ , (c) $iDirection$ ごとの生理快適度. ポイントプロットは平均値. . . . .	72
4.9	(a) $iDistance \times iDirection$ , (b) $iDistance$ , (c) $iDirection$ ごとの心理快適度. ポイントプロットは平均値. . . . .	73
4.10	$iDistance$ ごとの NASA-TLX の結果. 有意差は $*p < .05$ . . . . .	74
4.11	3つの手法の概要と, 対応する FOV. カメラビューの赤い円が FOV を表している. . . . .	78
4.12	$Method \times DT$ ごとの (a) 選択時間, (b) エラー率 (エラーバーは 95%信頼区間). 有意差は $**p < .01$ . . . . .	79
4.13	HG と KB の DT ごとの Visual Search Error (視覚探索エラー) と Midas Touch の割合. . . . .	81
4.14	$Method$ ごとの (a) SUS, (b) NASA-TLX の結果. 有意差は $*p < .05$ , $**p < .01$ . . . . .	81
4.15	(a) 手法 $Method$ , (b) 滞留時間 $DT$ の好み. . . . .	82
4.16	Kuiper Belt を用いたアプリケーションの例示. 緑色の点がユーザの視点. (a) メニュー, (b) TV を消そうとしている様子, (c) 瓶を選択しようとしている様子, (d) 銃から斧へ把持オブジェクトを変更しようとしている様子. . . . .	84



5.1	Asteroid Gazer は部分的に遮蔽されたオブジェクトの選択手法であり、ターゲット候補とメニューアイテムを連続的に選択することでターゲットを選択する。(a) メニューアイテムはユーザの頭部から 25°–45° の位置に固定されている。(b) オブジェクト群を一定時間注視することで、ユーザの視線(緑の点)に近いオブジェクトがターゲットとなる。ターゲット候補は一層目のメニューアイテムにそれぞれ対応する。フィードバックとして、ターゲット候補の輪郭に対応するメニューアイテムの色が表示され、メニューアイテムの中に対応するターゲット候補の外観が表示される。(c) ユーザがメニューアイテムを注視すると、注視したメニューアイテムの子階層のメニューが表示される。その後、注視したメニューアイテムに対応している複数のターゲット候補が子階層のメニューアイテムにそれぞれ分配される。最後に、ユーザが子階層のメニューアイテムを注視すると、注視したメニューアイテムに対応するターゲット(黒いオブジェクト)が選択される。	92
5.2	SQUAD のイメージ [93]。大まかに選択されたターゲット候補(左図)がクアドメニューに再配置される(右図)。	93
5.3	Radial Pursuit のイメージ [154]。	94
5.4	Outline Pursuits のイメージ [174]。	95
5.5	ActiGaze のイメージ [109]。	96
5.6	Asteroid Gazer の選択手順の遷移図。	97
5.7	(a) ターゲット候補を選択する際のイメージ。視線から 2.5° 以内のオブジェクトがターゲット候補として選択されている。オブジェクト内の数字は視線からの近さの順を示している。(b) ターゲット候補とメニューアイテムの対応の例。(c) ターゲット候補とメニューアイテムの対応の更新の例。(d) ターゲット候補の輪郭に表示される、対応しているメニューアイテムを示すフィードバック。(e) 子階層のメニューアイテムへのターゲット候補の分配の例。	100
5.8	(a) 一層目のメニューの外観と配置。(b) メニューアイテムのパラメータ。黒字はユーザから見た視角であり、青字はメニュー中心からの角度である。(c) 生成されたオブジェクトの俯瞰図。	101
5.9	独立変数 <i>Density</i> と <i>Occlusion</i> 。黒いオブジェクトがターゲット。	103
5.10	(a) <i>Method</i> × <i>Density</i> と (b) <i>Method</i> × <i>Occlusion</i> の選択時間。有意差は $*p < .05$ , $**p < .01$ 。	106
5.11	(a) <i>Method</i> × <i>Density</i> と (b) <i>Method</i> × <i>Occlusion</i> のエラー率。エラーバーは 95%信頼区間。有意差は $*p < .05$ , $**p < .01$ 。	106

5.12	<i>Method</i> × <i>Density</i> の候補選択時間. 有意差は** $p < .01$ .	107
5.13	<i>Method</i> ごとの (a) SUS, (b) Raw NASA-TLX の結果. 有意差は** $p < .01$ .	108
5.14	<i>Method</i> の好み.	109
5.15	Asteroid Gazer を活用したアプリケーションの図示. (a) 積まれたジグソーパズルの山からパズルピースを選択する様子. (b) テレビのチャンネルを変えようとしている様子. (c) 三つの商品を買おうとしている様子.	109

# 表 目 次

3.1	<i>TW</i> と <i>Spacing</i> の組み合わせ. . . . .	49
4.1	<i>iDistance</i> × <i>iDirection</i> ごとのエラー率の値. . . . .	71
4.2	<i>iDistance</i> × <i>iDirection</i> ごとの生理快適度の値. . . . .	72
4.3	<i>iDistance</i> × <i>iDirection</i> ごとの心理快適度の値. . . . .	73
4.4	<i>iDistance</i> と <i>iDirection</i> ごとの, 安定した選択操作のために推奨されるメ ニューアイテム (球) の直径 (°). . . . .	75