

Bubble Gaze Cursor : バブルカーソル法の視線操作への適用

崔 明根^{1,a)} 坂本 大介¹ 小野 哲雄¹

受付日 2019年4月15日, 採録日 2019年11月7日

概要: 視線入力インタフェースにおいて小さなターゲットを選択することは非常に困難であり, 容易にするためにはインタフェースデザインの変更か動作時間の増加を強いられる. 本稿では, 動作時間やデザインの汎用性を維持した状態で小さなターゲットを選択することを容易にすることを目的として, エリアカーソルの一種であるバブルカーソルを視線操作時のカーソルとして用いる手法を提案する. バブルカーソルが視線入力インタフェースにおいてどのような挙動をするのか確認するため, マウスのバブルカーソルと視線のポイントカーソル, そして視線のバブルカーソルに対してポインティングタスクを行った. その結果, 視線カーソルをバブルカーソルに変更したものはポイントカーソルよりもつねに高速に動作したうえ, ユーザビリティ評価指標もポイントカーソルよりも有意に高かった. したがって視線カーソルにバブルカーソルを用いる提案手法はポインティングを容易に, かつ高速にするうえで有効な手法だといえる.

キーワード: 視線インタフェース, バブルカーソル, ポインティングタスク

Bubble Gaze Cursor: Applying Bubble Cursor Technique to Eye-gaze Interaction

MYUNGGUEN CHOI^{1,a)} DAISUKE SAKAMOTO¹ TETSUO ONO¹

Received: April 15, 2019, Accepted: November 7, 2019

Abstract: Selecting a small target with an eye-gaze interface is difficult. Redesigning interface and/or increasing operation time are required for making eye-gaze interface easy to use. In this paper, we present a method of using an idea of bubble cursor, which is a kind of area cursor, for the eye-gaze interface in order to make it easy to select a small target while maintaining operation time and generality of interface design. We performed an experiment to validate our concept by comparing three interfaces, standard bubble cursor technique with a mouse, a standard eye-gaze interface with a point cursor, and the bubble cursor as an area cursor with eye-gaze interface in order to understand how the bubble cursor contributes to eye-gaze input interface. Results indicated that the bubble cursor with the eye-gaze interface always faster than the standard point cursor-based eye-gaze interface, and the usability score was also significantly higher than the standard eye-gaze interface. From those results, the bubble gaze cursor technique is an effective method to make eye-gaze pointing easier and faster.

Keywords: eye-gaze interface, bubble cursor, pointing task

1. はじめに

従来の人の視線の動きを計測するアイトラッキングデバイスやシステムは大きく高価であった. しかし, アイトラッキング技術の発展によってデバイスの小型化が進み, かつ安価に手に入れることができるようになったことから,

ヒューマンコンピュータインタラクション (HCI) の分野においてアイトラッキング技術を活用した研究が活発に実施されるようになった. 特によく研究されているのはユーザの注視点のポイントとして操作するような「注視操作システム」である. このシステムはユーザの視線のみを入力として受け取るため, ALS 患者など, 手足を動かすことができない人々のマウスに代わる入力インタフェースとして活用されている. また, マウス操作や音声入力, タッチ操

¹ 北海道大学
Hokkaido University, Sapporo, Hokkaido 060-0814, Japan
^{a)} myunguen.choi@complex.ist.hokudai.ac.jp

作などに対してユーザの視線データを活用することで、通常のマウスよりも高速なポインティングの実現方法 [1], [2] や効率的で容易な操作方法 [3], [4] など提案されている。

しかし、現在の視線操作にはいくつかの問題がある。たとえば視線情報を処理するだけでは、ターゲットを注視によって選択する際にユーザが入力したいのか、単に眺めているだけなのかを判断できないため、意図しない入力が行われてしまうという問題があげられる。この問題はギリシャ神話の触れるものすべてを黄金に変える力を持つミダスになぞらえ、Midas Touch Problem [5] と呼ばれており、注視型入力を採用する際には避けては通れない問題である。また特に顕著な問題として、小さなターゲットの選択が困難であるという点がある。これは固視微動（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）や安価なアイトラッカの低いトラッキング精度によって、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。固視微動に関しては人間の生理的な反応でありユーザが意図して取り除くことができないため、視線入力インタフェースとして利用しやすいように視線データを整形する必要がある。しかし、いまだに完全な視線データの制御は達成できていない。したがって、「小さなターゲットを選択することが難しい」という問題を解決するには、ズームなどの手法を用いて選択を容易にするよう工夫することや [6], [7], [8], [9], そもそも大きなターゲットのみが配置されるようなインタフェースデザインを採用すること [10] が必要になる。しかしこれらの手法は、ポインティングを容易にする代わりに、ポインティング速度やインタフェースデザインの汎用性を著しく損なってしまう。

そこで本稿では、視線入力インタフェースを用いる際に画面デザインを変更せずに、カーソルの機能や動作を変更することで小さなターゲットの選択を容易にするための手法を提案する。具体的には、ユーザのディスプレイ上の注視点を表す視線カーソルにバブルカーソル [11] を応用することで、小さなターゲットに対する容易なポインティングを達成する。現在、一般的な視線入力インタフェースにおいて、視線カーソルにはポイントカーソルが用いられている。この視線カーソルを、ターゲットを必ず1つだけ含む大きさにカーソルを拡大する手法であるバブルカーソルに変更することで、ターゲットの幅を実質的に拡大し、ポインティングを容易にすることを試みる。本稿では従来手法であるバブルカーソルおよび視線カーソルと、提案手法である Bubble Gaze Cursor を比較する実験を行い、この結果を報告する。

2. 関連研究

2.1 視線における動作時間予測式

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 [12], [13] である。

Fitts' Law は手を用いたポインティングタスクの動作時間に対して、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 [14] は、

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

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対して成り立つと考えられており、主にマウスによるポインティングタスクに対して用いられることが多く、視線による入力に対してもこの法則が成り立つと結論付ける研究 [15], [16], [17], [18] も存在する。しかし、視線を用いたポインティングタスクの動作時間が Fitts' Law と一致しないという研究が存在する [1], [6], [19] ほか、Fitts' Law が成り立つとする論文に対して取り扱っている変数の値が不適切であるといった指摘 [20], [21] や、そもそも視線は弾道的な動きをするため Fitts' Law は視線の動作時間に対する予測式としては不適切であるといった指摘 [22] があり、視線入力および視線情報を利用したポインティングタスクに対して Fitts' Law が成り立つとはいえない。

ゆえに、Zhang ら [20] は視線によるターゲットングにおいて、ターゲット間の距離よりもターゲットの幅の方が大きく動作時間に対して影響を与えるという考えから、視線によるポインティングタスクの動作時間の予測式として以下の式を提案した。

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

ここで λ 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、 μ 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、 AR (Average Radius) はターゲット内に視線が入ってから軌跡の中心からの平均の距離を、 \overline{AR} はすべての独立変数に対して得られた AR の平均を表している。

また、同論文内で $\lambda = 0.0005$, $\mu = 2\overline{AR}$ とおくことで適切な予測式を得ることができるうえに、 A や W の単位を考慮せずに用いることができるようになる主張している。

通常のディスプレイで視線を用いてポインティングタスクを行う際、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。ゆえに、動作時間に対してターゲット間の距離の増減が大きく影響を与えることはないと考えられる。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の幅を大きくする手法が効果的であると考

えられる。

2.2 ジッタを減少させる手法

視線はジッタや高速な挙動などの特徴を有しているため、アイトラッキングデバイス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入力インタフェースに用いることは難しい。ゆえに、ジッタノイズを抑えるために、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 [23], [24]。

最も単純な平滑化アルゴリズム [25] は式 (3) のように視点位置 pos_t を、得られた生の視線データ $gaze_t$ とその 1 つ前の視点位置 pos_{t-1} を用いて平滑化するアルゴリズムである。

$$pos_t = \alpha gaze_t + (1 - \alpha) pos_{t-1} \quad (3)$$

また、最後の視点位置から N 点の視点位置の平均を現在の視点位置とするようなアルゴリズム [26] や、N 点の視点位置ごとに重みをかけてから平均をとるようなアルゴリズム [27] など平滑化のために用いられている。また、これらの平滑化アルゴリズムの欠点であるサッケードの遅延を改善するアルゴリズム [28], [29], [30] も提案されている。Zhang ら [30] はターゲット内に視線が入ったときのみターゲット中心から離れていく視線の動きに対して式 (3) のような遅延フィルタをかける手法 iSR (improved Speed Reduction) を導入することで、視線の高速性を保持した状態でジッタを大幅に減少させた。

上記より、ジッタを減少させる取り組みは成功しており、これらの手法を用いることで視線によるポインティングを容易にすることができる。しかし、ジッタの幅 μ は 0 より小さくなることはないため ($W - \mu$) は W より大きくなることはない。ゆえに、ターゲットの幅 W がそもそも小さい場合、ジッタを減少させることは動作時間減少のための対策としては不十分だと考えられる。したがって、小さなターゲットに対するポインティングを根本的に改善するためには、ターゲットの幅 W を増大させるような手法を用いる必要があるといえる。

2.3 ターゲットの幅を増大させる手法

ターゲットの幅を増大させる手法としてまず考えられる手法はズームによるターゲットサイズの拡大である。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 [31], [32] ことが確認されている。

Skovsgaard ら [7] は、注視点に対して離散的なズームを複数回行うことでターゲットを拡張する手法を提案した。しかし、この手法を視線操作インタフェースに組み込む場合、ズームによって画面を拡大している場合にズーム領域外の情報が損失してしまうことや、ターゲット選択にかか

る時間のほかにズーム操作のための時間が必要になることによるタスク達成時間の増加などが起こると考えられる。Ashmore ら [8] はズーム領域外の情報の損失を防ぐために魚眼レンズによる局所的なズームによってターゲット拡張を行う手法を提案した。この手法では、魚眼レンズ内の文字などの情報の損失を防ぐために、視線がある程度固定された後に魚眼レンズによる拡大が行われる。しかし、この手法も Skovsgaard ら [7] の手法と同様にズームが発生するまでの時間がターゲット選択のほかに必要となるため、操作時間が大きくなってしまふ。

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。Špakov ら [9] 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。Miniotas ら [6] 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できないため、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。ゆえに、本研究では動作時間やデザインの汎用性を維持し、かつ小さなターゲットのポインティングを容易にする手法を提案する。

2.4 視線操作インタフェースにおいて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場合、網膜の中心に存在する中心窩でオブジェクトをとらえる必要がある。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 度 (眼球中心からディスプレイまでの角度) である。したがって、ディスプレイ上のターゲットが中心窩の視角に内包されたとき、人はターゲットをはっきりと見ており、その視覚以上に正確に目の位置を調整する必要はない。ゆえに、中心窩内に複数のオブジェクトが含まれるような状況において唯一の注視オブジェクトを決定することは困難である [33]。したがって、視線操作インタフェースにおけるオブジェクトの大きさは最低でも中心窩の視角以上の大きさは必要であると考えられる。また、ターゲッ

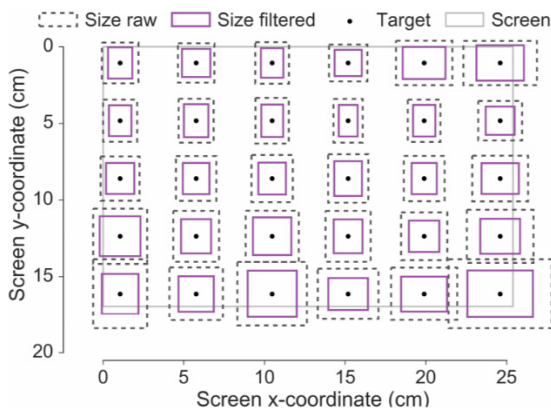


図 1 ディスプレイ位置ごとの適切なターゲットサイズ [24]

Fig. 1 Appropriate target sizes for different screen positions [24].

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、すべての視点位置を内包する領域は非常に大きくなる。Feitら [24] 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 3.28×3.78 cm (2.90×3.33 度)、適切な平滑化アルゴリズムを使用する場合は 1.90×2.35 cm (1.67×2.07 度)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 [11]。したがって、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ような手法が必要であるといえる。図 1 にディスプレイの位置ごとのターゲットの適切なターゲットサイズの図を示す。

2.5 視線カーソル

視線カーソルとは、ディスプレイ上におけるユーザの視点を可視化したものであり、様々な形状が用いられている。一般的な視線カーソルの形状は点状のカーソルである。この形状は従来の視線入力によるポインティング手法に多く用いられるほか、カーソル自体が小さいことから、アイタッピング [34] のように密集したターゲット群を選択するインタフェースなどに対して用いられている。他の形状としては、半径の大きな円形のカーソル [3], [4] や四角形のカーソル [27] などが用いられている。近年商業的に最も用いられている Tobii 社のアイトラッカでは、視線カーソルとしてカーソルの動作によって形状が変わるような泡状のカーソルを採用している。この視線カーソルはカーソル自体がある程度の大きさを持つことでアイトラッカの低い精度を保証している。しかし、カーソルの大きさよりも小さなターゲットが密集している場合、視線カーソル内に複数のターゲットを含んでしまう可能性があるほか、ターゲットが選択されているかどうかを視線カーソルの状態から判

別することができないため、視線カーソル内にターゲットが内包されている状態でもターゲットが選択されないことが起こりうる。

また、単純にディスプレイ上の視線の位置を表すほかに、視線カーソル自体に情報を付加する研究や、インタフェースデザインの1つとする研究も存在する。EyeDraw [27] は視線カーソルを絵筆にすることによって絵を描くためのインタフェースを提案し、その中で視線カーソルの色を、ただ見ている場合は緑、線を引いている場合は赤色に変更することで現在の視線カーソルの状態を表現した。ceCursor [35] は GUI を視線で操作するために視線カーソルの形状を上下左右の4方向を指し示すような形状にし、アイコンなどの選択を、視線カーソルを離散的に移動させることによって容易にした。

上記のように視線カーソルは様々な実験で用いられているが、視線入力によって達成したい目的がターゲット選択ではないことが多く、視線カーソル自体に焦点を当てるような研究は少ない。

2.6 バブルカーソル

ターゲット選択において、点のカーソルを用いて面のターゲットを選択することと、面のカーソル（エリアカーソル） [36], [37] を用いて点のターゲットを選択することは動作時間の観点から同義である。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、カーソルが複数のターゲットを含んでしまう状況が起こりうる。この問題を解消するために提案された手法がバブルカーソル [11] である。バブルカーソルはつねにターゲットを1つだけ含むような大きさにカーソルの大きさを動的に変更し続ける円形のエリアカーソルである。したがって、エリアカーソルの特性より、バブルカーソルを用いることでターゲットの幅をバブルカーソルの幅に拡大することと同等の効果を得ることができる。

バブルカーソルによって拡大されるターゲットの実質的な幅 EW (Effective Width) はボロノイ図で表現することができる。ボロノイ図とは平面内の各母点の再近隣領域（ボロノイ領域）を分割した図である。つまり図 2 におけるボロノイ領域がバブルカーソルによって拡張されたターゲットの実質的な大きさとなる。

3. 提案手法：Bubble Gaze Cursor

本稿では小さなターゲットに対するポインティングを容易にすることを目的として、視線カーソルにバブルカーソルを用いる手法（Bubble Gaze Cursor）を提案する。

Bubble Gaze Cursor を用いる利点として、ターゲットの幅を増大させる既存手法の難点であった動作時間の増大とインタフェースデザインの制限を回避しつつ、小さなターゲットに対するポインティングを容易にすることがあげら

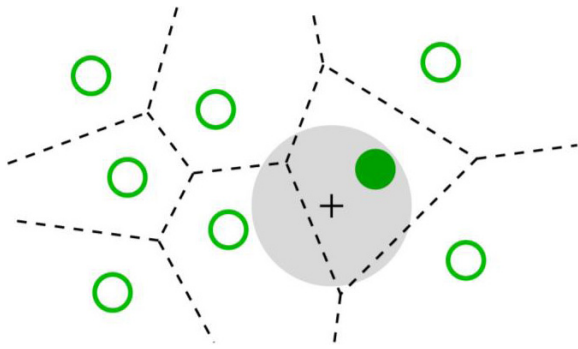


図 2 Grossman ら [11] によるバブルカーソルとポロノイ図
 Fig. 2 Bubble cursor and Voronoi diagram by Grossman et al. [11].

れる。バブルカーソルはターゲットを必ず1つだけ含む大きさに変形するカーソルであるため、カーソルの移動とともにカーソルはつねに大きさを変える。したがって、ターゲットの幅を増大させる処理がつねに行われている状態でありターゲット拡大の処理に時間がかからないため、動作時間を維持してポインティングを行うことができると考えられる。また、カーソルによる処理であるため、Bubble Gaze Cursor を用いることで視線入力インタフェースに小さなターゲットの配置を可能にすると考えられる。

4. 実験

マウス操作時においてバブルカーソルはポインティングタスクの動作時間を減少させ、かつカーソルの幅を用いて Fitts' Law で定式化することが可能であることが確認されている [11]。しかし、視線操作時において、バブルカーソルが正常に動作するか、また動作時間を予測できるかどうかについては十分に研究されていない。ゆえに、視線カーソルをバブルカーソルに変更することによってどのような影響を及ぼすか検証するために、ポイントカーソル（視線）とバブルカーソル（視線とマウス）を用いてポインティングタスクを行う。

4.1 使用機器と実験参加者

本実験は、2.30 Ghz の Intel (R) Core (TM) 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 (1,920 × 1,080 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。アイトラッカには Tobii EyeX を使用し、実験に用いたソフトウェアは C# で作成した。実験システムを図 3 に示す。

実験には主に大学生 (男性 14 人, 女性 4 人, 平均年齢 23.9 歳) の 18 人が参加した。すべての実験参加者はコンピュータやマウス操作を日常的に用いていたが、実験参加者の中でアイトラッキングを実際に使用したことがある者は 4 人で、それ以外はアイトラッキングを使ったことがなかった。また報酬として、実験参加者にそれぞれ 1,000 円

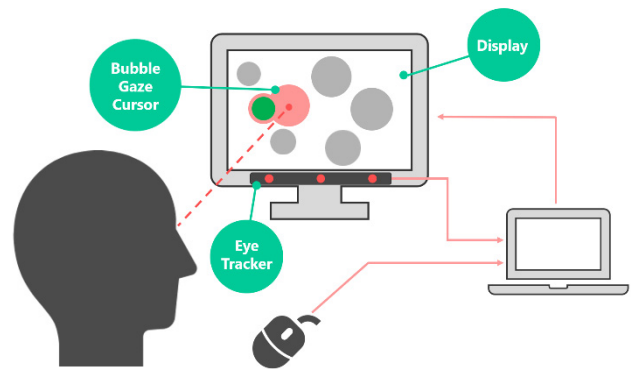


図 3 実験システム
 Fig. 3 Experimental system.

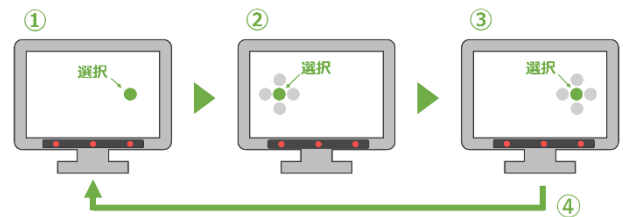


図 4 実験タスクの手順
 Fig. 4 Experimental task procedure.

分の Amazon ギフトカードを渡した。

4.2 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、視線によるタスクを行う場合はアイトラッカのキャリブレーション (ディスプレイ上の視点位置を個人に合わせて調整すること) を行った。実験開始前には視線入力やタスクの内容に慣れるために、これから行うタスクと同様のポインティングタスクを、パラメータを実際のタスクとは異なった値に設定し、実験参加者が挙動を理解したと宣言するまで練習させた。そして練習が終了した後に実験タスクを開始した。

すべての操作手法において、まず “START” と印字されたボタンがディスプレイ上に提示される。それをマウスでクリックすることでタスクを参加者自らが開始する。その後、図 4 のように以下の試行を一定回数繰り返す。

- ① 画面上に緑色の円形のターゲットのみが画面に提示されるので、このターゲットを選択する。選択はマウスの場合にはクリック、視線の場合にはターゲットを注視することによって選択される。すべての操作手法において、このターゲットを選択する時間は計測されておらず、位置合わせを目的としてこの試行を行う。
- ② 試行①直後、試行①の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の対称な位置にターゲット群が生成される。ターゲット群は緑色の円形ター

ゲットを四方に囲むように灰色の円形ターゲットが配置されたものである。灰色の円形ターゲットは緑色と同じ大きさで、緑色のターゲットに対応するボロノイ領域の幅がバブルカーソルの最大幅と一致するように配置されている。実験参加者はこのターゲット群の中から緑色のターゲットを試行①と同様の方法で選択する。

③ 反対側に同様のターゲット群が生成されるので同様に緑色のターゲットを選択する。

④ 試行①に戻る。

緑色のターゲットの中心には 10 ピクセルの濃緑色の点が描画されている。これは注視すべき凝視点が存在しない場合ターゲットを注視し続けることが難しい [38] ため、配置した。実験参加者にはポインティングタスクの際に視線カーソルではなくこの濃緑色の点を見るように促した。また、バブルカーソルの色は薄い赤色で描画し、マウスのバブルカーソルの中心にはクロスカーソルを、視線のバブルカーソルの中心には 6 ピクセルの濃赤色の点を描画した。ポイントカーソル（視線）は視線のバブルカーソルの中心に描画した濃赤色の点をカーソルとして用いた。

視線入力の際の滞留時間は、今回は単純なポインティングタスクを行うことから、Zhang ら [39] の実験より 600 ms とした。また、試行①で表示されるターゲットに対しては、実験の高速化のために滞留時間を 450 ms とした。

4.3 実験デザイン

実験は参加者内実験計画を用いて行った。独立変数はカーソルの種類 CT (Cursor Type) (BubbleMouse: マウスカーソルにバブルカーソルを用いた手法, PointGaze: 視線カーソルにポイントカーソルを用いた手法, BubbleGaze: 視線カーソルにバブルカーソルを用いた手法), ターゲット間の距離 A (350, 550, 800 ピクセル), ターゲットの幅 W (50, 65, 80 ピクセル), バブルカーソルの最大の幅 EW (100, 120, 140 ピクセル) とした。 CT の図解を図 5 に示す。独立変数 A , W , EW の値は式 (2) の難易度を表す項である ID_{EYE} の値がばらつくように設定した。 ID_{EYE} の値は $2000e^{A/2000}/(W - 2\overline{AR})$ である。特にターゲットの幅 W は、適切な平滑化アルゴリズムを用いた場合において十分なターゲットサイズである 1.67×2.07 度 [24] を参考とし、それよりも小さなターゲットの幅 W を設定した (ターゲットの幅 W はそれぞれ 1.27 度, 1.65 度, 2.03 度)。ターゲットの幅 W のうち、最も大きな幅である 2.03 度は十分なターゲットサイズの横幅よりも大きい、生成ターゲットが円形であることやトラッキングの精度のブレを考慮すると、本実験で設定したターゲットの幅は十分に小さいといえる。

また、従属変数としてすべてのカーソルタイプに対して動作時間 MT (Movement Time) と精度を求め、視線に

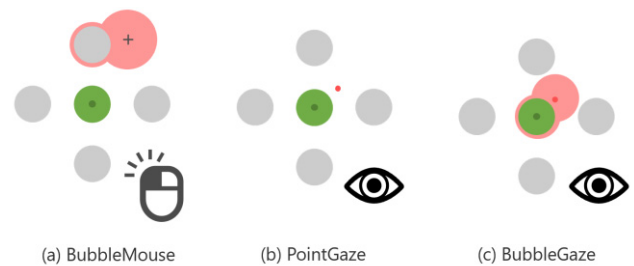


図 5 CT の図解
Fig. 5 Graphic of CT .

よる入力を行う CT に対してはターゲットにカーソルが触れるまでの時間 EMT (Eye Movement Time) と、Zhang ら [30] のジッタの幅を求める方法を参考にジッタの半径 AR (Average Radius) を求めた。精度は灰色のターゲットを選択した場合と、試行時間に 5 秒以上かかった場合を失敗とし、計算した。

実験全体の拘束時間はおよそ 90 分であり、 CT 条件ごとに 9 ブロックのタスクを行ってもらった。それぞれのブロックごとに、実験参加者はランダムな順に提示される A , W , EW の 27 の組合せに対して実験タスクを行った。休憩は図 4 の試行①のタイミングかブロック完了ごとに実験参加者の自己判断で行った。

また、それぞれの CT 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 NASA Task Load Index (NASA-TLX) [40] と System Usability Scale (SUS) [41] によるアンケートを実施し、実験終了後に各手法に関して気付いた点をアンケートおよびインタビューで尋ねた。

4.4 実装

バブルカーソルを、滞留時間を用いた視線入力インタフェースに導入するにあたって変更すべき点がある。バブルカーソルはターゲットを必ず 1 つ含むような大きさに変形するカーソルである。ゆえに、視線カーソルに対してそのままバブルカーソルを移植する場合、つねにいずれかのターゲットを注視している状態になるため、誤入力が頻発することとなる。この問題の解決のためにバブルカーソルに最大半径を導入し、どのターゲットも選択されていない状態を作った。今回の実験ではバブルカーソルの最大半径を EW とした。

また、サッケード時にバブルカーソルの描画が滑らかに行われず、離散的に描画されたため、式 (3) の平滑化アルゴリズムを導入した。さらにジッタの抑制と過剰なエラーを抑えるために Zhang ら [30] の iSR を採用した。

4.5 結果

それぞれの従属変数に対して 4 要因分散分析を行った。多重比較には Bonferroni の手法を用いた。

4.5.1 動作時間 (MT)

MT に対して CT ($F_{2,34} = 1590.92, p < 0.001$), A ($F_{2,34} = 721.37, p < 0.001$), W ($F_{2,34} = 101.86, p < 0.001$), EW ($F_{2,34} = 36.76, p < 0.001$) がそれぞれ有意であった。また, 1 次の交互作用として CT × A ($F_{4,68} = 8.20, p < 0.001$), CT × W ($F_{4,68} = 81.31, p < 0.001$), CT × EW ($F_{4,68} = 9.19, p < 0.001$) が, 2 次の交互作用として CT × A × EW ($F_{8,136} = 2.30, p < 0.05$) が有意であった。すべての CT は A に有意な差が確認された (BubbleMouse ($F_{2,34} = 926.70, p < 0.001$), PointGaze ($F_{2,34} = 142.80, p < 0.001$), BubbleGaze ($F_{2,34} = 359.81, p < 0.001$)) が, W は PointGaze のみに有意な差が (BubbleMouse ($F_{2,34} = 0.33, p = 0.72$), PointGaze ($F_{2,34} = 101.47, p < 0.001$), BubbleGaze ($F_{2,34} = 0.71, p = 0.50$)), EW は BubbleMouse と BubbleGaze にのみ有意な差が確認された (BubbleMouse ($F_{2,34} = 45.03, p < 0.001$), PointGaze ($F_{2,34} = 0.27, p = 0.77$), BubbleGaze ($F_{2,34} = 49.58, p < 0.001$)). MT の平均は BubbleMouse が 793.92 ms, PointGaze が 1,529.90 ms, BubbleGaze が 1,269.81 ms であった。W と EW の組合せすべてに対する CT ごとの MT を図 6 に示す。

図 7 には PointGaze と BubbleGaze に対して, 式 (2) の動作時間予測式に則って, MT の値から線形回帰分析した結果を示した。図 7 の横軸 ID_{EYE} (Index of Difficulty EYE) は式 (2) より, $2000e^{A/2000}/(W-2\bar{A}R)$ とした。PointGaze のモデルの適合度 R^2 の値は 0.89 であり, BubbleGaze は 0.54, PointGaze と BubbleGaze のデータをあわせて線形回帰した場合は 0.91 であった。

4.5.2 ターゲットに触れるまでの時間 (EMT)

EMT に対して CT ($F_{1,17} = 922.99, p < 0.001$), A ($F_{2,34} = 600.17, p < 0.001$), W ($F_{2,34} = 146.33, p < 0.001$), EW ($F_{2,34} = 13.86, p < 0.001$) が有意であった。また, 1 次の交互作用として CT × A ($F_{2,34} = 6.70, p < 0.01$), CT × W ($F_{2,34} = 141.76, p < 0.001$), CT × EW ($F_{2,34} = 16.46, p < 0.001$), A × EW ($F_{4,68} = 2.90, p < 0.05$) が有意であった。両方の CT は A に有意な差があった (PointGaze ($F_{2,34} = 220.27, p < 0.001$), BubbleGaze ($F_{2,34} = 933.97, p < 0.001$)) が, W は PointGaze のみに有意な差が (PointGaze ($F_{2,34} = 162.56, p < 0.001$), BubbleGaze ($F_{2,34} = 0.80, p = 0.46$)), EW は BubbleGaze にのみ有意な差が確認された (PointGaze ($F_{2,34} = 0.24, p = 0.79$), BubbleGaze ($F_{2,34} = 94.45, p < 0.001$)). EMT の平均は PointGaze が 819.67 ms, BubbleGaze が 612.15 ms であった。W と EW の組合せすべてに対する CT ごとの EMT を図 8 に示す。

図 9 には PointGaze と BubbleGaze に対して, 式 (2) の動作時間予測式に則って ($\mu = 0$ とすることで EMT に対

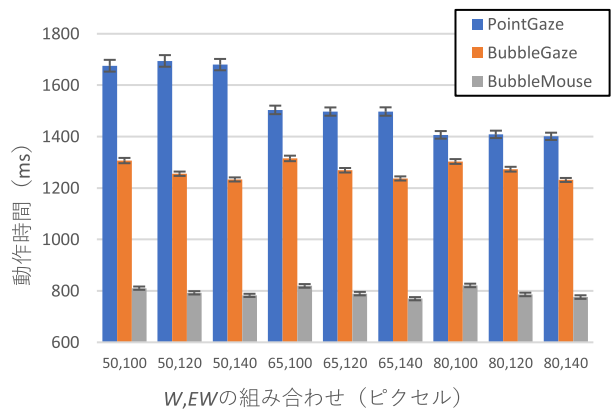


図 6 W と EW の組合せに対する MT
Fig. 6 MT by combination of W and EW values.

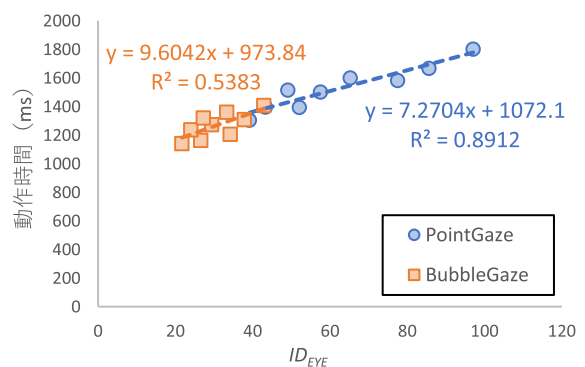


図 7 ID_{EYE} に対する MT
Fig. 7 MT by ID_{EYE} .

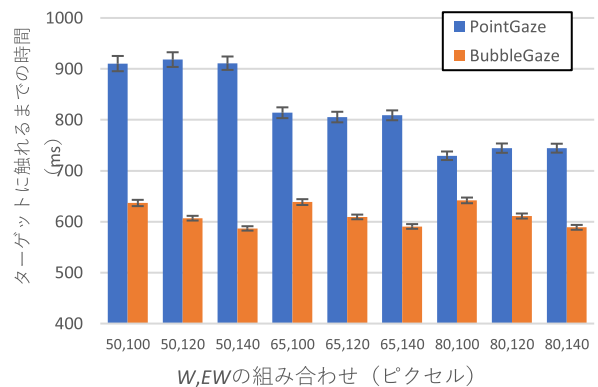


図 8 W と EW の組合せに対する EMT
Fig. 8 EMT by combination of W and EW values.

する動作時間予測式を計算することができる [20]), EMT の値から線形回帰分析した結果を示した。PointGaze のモデルの適合度 R^2 の値は 0.85 であり, BubbleGaze は 0.59, PointGaze と BubbleGaze のデータをあわせて線形回帰した場合は 0.90 であった。

4.5.3 ジッタの半径 (AR)

AR に対して CT ($F_{1,17} = 206.73, p < 0.001$), A ($F_{2,34} = 35.57, p < 0.001$), W ($F_{2,34} = 3.57, p < 0.05$), EW ($F_{2,34} = 5.29, p < 0.01$) が有意であった。また, 1 次の交互作用として CT × A ($F_{2,34} = 6.98, p < 0.01$),

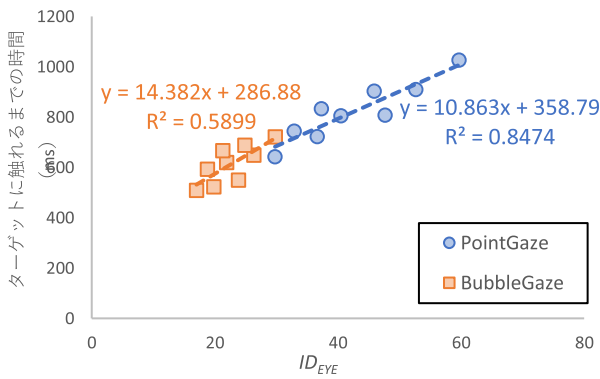


図 9 ID_{EYE} に対する EMT
Fig. 9 EMT by ID_{EYE} .

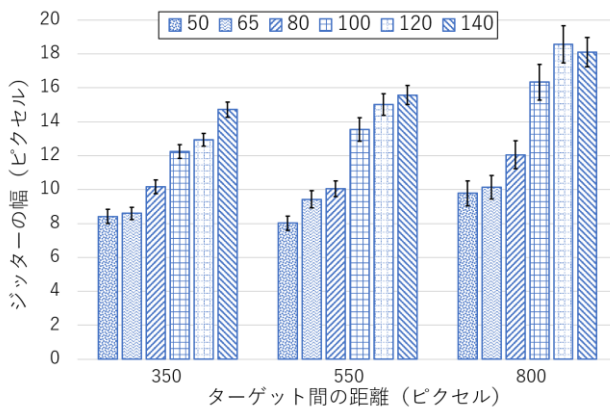


図 10 ターゲットの幅 (W, EW) とターゲット間の距離 A に対する AR
Fig. 10 AR by target width (W, EW) and the distance A between targets.

$CT \times W$ ($F_{2,34} = 3.46, p < 0.05$), $CT \times EW$ ($F_{2,34} = 8.91, p < 0.001$) が有意であった。両方の CT は A に有意な差があった (PointGaze ($F_{2,34} = 9.52, p < 0.001$), BubbleGaze ($F_{2,34} = 26.33, p < 0.001$)) が, W は PointGaze のみに有意な差が (PointGaze ($F_{2,34} = 7.76, p < 0.01$), BubbleGaze ($F_{2,34} = 0.23, p = 0.79$)), EW は BubbleGaze にのみ有意な差が確認された (PointGaze ($F_{2,34} = 1.82, p = 0.18$), BubbleGaze ($F_{2,34} = 5.98, p < 0.01$)). AR の平均は PointGaze が 9.63 ピクセル, BubbleGaze が 15.22 ピクセルであった。ターゲットの幅 (PointGaze は W , BubbleGaze は EW) とターゲット間の距離 A に対する AR を図 10 に示す

4.5.4 精度

精度に対して W ($F_{2,34} = 7.48, p < 0.01$), EW ($F_{2,34} = 10.08, p < 0.001$) が有意であった。また, 1 次の交互作用として $CT \times A$ ($F_{4,68} = 9.13, p < 0.001$), $CT \times W$ ($F_{4,68} = 23.99, p < 0.001$), $CT \times EW$ ($F_{4,68} = 4.70, p < 0.01$), $A \times EW$ ($F_{4,68} = 2.86, p < 0.05$) が, 2 次の交互作用として $CT \times A \times W$ ($F_{8,136} = 3.41, p < 0.01$), $CT \times A \times EW$ ($F_{4,68} = 2.87, p < 0.01$) が, 3 次の交互作用として $CT \times A \times W \times EW$ ($F_{16,272} = 2.26, p < 0.01$)

が有意であった。精度の平均は BubbleMouse が 98.67%, PointGaze が 98.15%, BubbleGaze が 98.27% であった。

4.6 NASA-TLX および SUS

NASA-TLX の平均スコアは BubbleMouse が 25.56, PointGaze が 64.15, BubbleGaze が 39.44 であった。これらの値には有意な差が存在し ($F_{2,34} = 25.43, p < 0.001$), それぞれの CT 間にも有意な差が存在した (BubbleMouse-PointGaze ($p < 0.001$), PointGaze-BubbleGaze ($p < 0.001$), BubbleGaze-BubbleMouse ($p < 0.05$)).

SUS の平均スコアはそれぞれ, BubbleMouse が 78.19, PointGaze が 55.56, BubbleGaze が 69.44 であった。これらの値には有意な差が存在し ($F_{2,34} = 12.24, p < 0.001$), PointGaze と他の CT に有意な差は存在したが (BubbleMouse-PointGaze ($p < 0.01$), PointGaze-BubbleGaze ($p < 0.01$)), BubbleMouse と BubbleGaze の間に有意な差は存在しなかった ($p = 0.062$).

5. 議論

本稿では, 小さなターゲットのポインティングを容易にする手法として視線カーソルにバブルカーソルを用いる手法を提案した。バブルカーソルが視線入力インタフェースにおいて正常に動作するか検証するため, ポイントカーソル (視線) とバブルカーソル (視線とマウス) を用いてポインティングタスクを行った。その結果, 以下の点が明らかになった。

1. 動作時間 (MT) はすべての条件の組合せに対して, BubbleMouse, BubbleGaze, PointGaze の順に速く, それぞれの MT の平均は, 793.92 ms, 1,269.81 ms, 1,529.90 ms であった。
2. ターゲットに触れるまでの時間 (EMT) (PointGaze, BubbleGaze に対して計測) はすべての条件の組合せに対して PointGaze よりも BubbleGaze の方が速く, EMT の平均は PointGaze が 819.67 ms, BubbleGaze が 612.15 ms であった。
3. MT に対する PointGaze のモデルの適合度 R^2 の値は 0.89 であり, BubbleGaze の R^2 の値は 0.54, PointGaze と BubbleGaze のデータをあわせて線形回帰した場合の R^2 は 0.91 であった。また, EMT に対する PointGaze の R^2 の値は 0.85 であり, BubbleGaze の R^2 の値は 0.59, PointGaze と BubbleGaze のデータをあわせて線形回帰した場合の R^2 は 0.90 であった。計算結果より PointGaze の適合率は高かったが, BubbleGaze の適合率は低かった。
4. ジッタの半径 (AR) の平均は PointGaze が 9.63 ピクセル, BubbleGaze が 15.22 ピクセルであった。また, 精度の平均は BubbleMouse が 98.67%, PointGaze が 98.15%, BubbleGaze が 98.27% であった。

5. NASA-TLX の平均スコアは BubbleMouse が 25.56, PointGaze が 64.15, BubbleGaze が 39.44 であった. SUS の平均スコアは BubbleMouse が 78.19, PointGaze が 55.56, BubbleGaze が 69.44 であった.

5.1 得られた従属変数に対する考察

5.1.1 動作時間 (MT), ターゲットに触れるまでの時間 (EMT)

実験の結果より, すべての実験の組合せにおいて動作時間 MT は BubbleMouse, BubbleGaze, PointGaze の順に速く, ターゲットに触れるまでの時間 EMT は BubbleGaze, PointGaze の順で速かった. したがって視線操作において BubbleGaze は PointGaze よりもつねに高速に動作するといえる. また BubbleGaze の MT , EMT は EW に対する有意差が存在したが, W に対する有意差は存在しなかった. これは BubbleMouse の MT に対しても同様であることから, 視線カーソルにバブルカーソルを用いることによってターゲットの幅が拡大されていることが分かる.

一方で BubbleGaze は BubbleMouse よりも動作時間が大きかった. MT の平均で比較すると BubbleMouse は BubbleGaze よりもおよそ 470 ms 高速に動作した. この差は, 今回設定した滞留時間 600 ms よりも小さい. したがって滞留時間以外の入力手法 (筋電 [2], ボタンクリック [15] など) の手法を用いることでマウス操作の動作時間に迫ることができると考えられる. また, 視線はマウスよりも非常に高速であるためターゲット間の距離が開けば開くほど, より視線操作が高速に動作する. ゆえに, 本実験で用いた変数よりもターゲット間の距離が大きい場合, BubbleGaze と BubbleMouse との差は小さくなると考えられる.

5.1.2 ジッタの半径 (AR)

実験の結果より, ジッタの半径 AR の平均は PointGaze が 9.63 ピクセル, BubbleGaze が 15.22 ピクセルであり, BubbleGaze の方が PointGaze よりも AR が大きいという結果となった. これは図 10 から読み取れるとおり, ターゲットの幅 (PointGaze の場合は W , BubbleGaze の場合は EW) が大きくなるほど, AR も同様に大きくなるからである. AR がターゲットの幅に比例して大きくなるという特性は先行研究 [20] でも指摘されており, ターゲットが大きくなるほどターゲットに視線を固定しようという注意が散漫になり, AR が大きくなると考察している. しかし BubbleGaze の場合, ターゲットの実質的な幅が可視化されていないが, AR と EW は有意な相関が存在し, EW が大きくなるほど AR が大きくなっている. これはユーザがターゲット群の配置からターゲットの実質的な幅を推測しているためだと考えられる.

5.2 Bubble Gaze Cursor における動作時間予測式

実験より, PointGaze と BubbleGaze のどちらも式 (2)

の MT , EMT に対する予測式に対する適合度が高くなかった. 特に BubbleGaze の R^2 は低く, MT は 0.54, EMT は 0.59 であった. 予測式との適合率が低い理由として, 式 (3) の平滑化アルゴリズムを導入したことが考えられる. 式 (3) の平滑化アルゴリズムは高速な視線データを遅延させるアルゴリズムであり, この遅延によってジッタを減少させるが, 同時にサッケードなどの視線の高速移動も遅延させてしまう. ゆえに視線の速度が遅くなり, ターゲット間の距離 A が通常の視線操作よりも大きく影響を及ぼすようになったため, 式 (2) に適合しなくなったと考えられる. 実際, 式 (2) を提唱した Zhang らの論文 [20] では, ターゲット内部でのみ機能し, ジッタを減少させるアルゴリズムである iSR [30] を除いて, 視線データに対して平滑化アルゴリズムを利用することはいっさいしていない. したがって, 今後実験デザインを変更して追実験を行う必要があるといえる.

一方で, PointGaze と BubbleGaze の両方のデータをあわせて線形回帰した場合の R^2 は, MT は 0.91, EMT は 0.90 であった. この値は PointGaze や BubbleGaze 単体の適合度よりも高い. したがって, 平滑化アルゴリズム適用時においても, PointGaze から得られる動作時間予測式から BubbleGaze の動作時間のある程度予測することが可能であると考えられる.

5.3 提案手法の利用可能性

現在デスクトップにおける視線操作は Windows10 で公式にサポートされており, Web ブラウジングを視線のみで行うことができる [42]. しかし, 用いられるインタフェースデザインはマウス操作を前提としたデザインであり, 視線操作用にデザインが再生成されることはない. マウス操作を前提としたターゲットサイズは視線で操作するには不十分 [10] である. Windows10 においてはこの不十分なターゲットサイズをズームとカーソル位置の微調整によってフォローしているが, 代償として動作時間の増大を引き起こしている. したがって提案手法を利用することで, 容易にかつ高速に Web ブラウジングが可能になると考えられる.

同様に, 提案手法はスマートフォンやタブレットなどのモバイルデバイスに対する視線操作を容易にする可能性がある. モバイルデバイスと視線操作を組み合わせた研究はすでいくつか行われている [3], [4], [43] ゆえに, 深度カメラがモバイルデバイスに組み込まれることが一般的になり, 視線をモバイルデバイスだけで検出することができるようになった [44] ことから, より活発に研究されることが考えられる. 一方で, モバイルデバイスの視線推定の精度はデスクトップ用のアイトラッカの精度 [45] と比較すると低い [46], [47]. これは限られた計算リソースや内蔵 RGB カメラの性能, 頭部の頻繁な移動が原因である. したがって

十分な大きさを持ったオブジェクトを画面上に配置する必要があるが、モバイルデバイスの小さな画面に配置する場合、わずかなオブジェクトで画面が占有されてしまう可能性がある。ゆえに提案手法を用いることで、ターゲットサイズを実質的に拡張しながら画面に対する情報量を保つことができると考えられる。

また、他の利用法としてEyeDraw [27] のようにカーソルの色を変更することによる情報提示なども考えられる。視線カーソルはつねに視界内に存在するため、視線カーソルの色の変更を自然に知覚できるほか、バブル自体が大きいいため、色の変化が分かりやすい。情報提示の例としては、現在のターゲット選択時のアクションの視覚化（左クリック、右クリック、ダブルクリックなど）などが考えられる。

5.4 限界と今後の課題

バブルカーソルは視線操作においても正常に動作するか検証することを目的として、本稿では水平方向に限定されたポインティングタスクを行った。しかし、ターゲットの出現方向を制限しないポインティングや実環境における挙動を検証していないほか、Midas Touch Problem [5] の回避のために設定した、バブルカーソルの最大半径の最適な値の検証も必要であると考えられる。したがって、提案手法に対する検討はまだ必要であり、提案手法が実用に耐えるか、引き続き実験していくべきだといえる。

また、本研究においては視線カーソルにバブルカーソルを用いることで視線カーソルをエリアカーソルに拡張することができるか確認したが、バブルカーソルのほかにも視線カーソルに拡張することができるか検証すべきエリアカーソルは様々に存在する [48], [49]。したがって、それぞれのエリアカーソルに対しても検証を続け、最適な視線カーソルを探求していく必要があるといえる。

6. おわりに

本稿では視線入力インタフェースでの小さなターゲットに対するポインティングを容易にすることを目的として、視線カーソルを、既存のポイントカーソルからバブルカーソルに変更する提案を行った。視線ポイントカーソルと視線バブルカーソルを比較するポインティングタスクを行ったところ、視線バブルカーソルは視線ポイントカーソルよりもつねに高速で動作するうえ、ユーザビリティ評価指標やメンタルワークロード指標も視線ポイントカーソルより優れていることが確認された。

謝辞 本研究はノーステック財団「研究開発助成事業」の助成を受けたものです。

参考文献

[1] Zhai, S., Morimoto, C. and Ihde, S.: Manual and gaze input cascaded (MAGIC) pointing, *Proc. SIGCHI con-*

- ference on Human Factors in Computing Systems (CHI '99)*, pp.246–253, DOI: 10.1145/302979.303053 (1999).
- [2] Mateo, J.C., San Agustin, J. and Hansen, J.P.: Gaze beats mouse: Hands-free selection by combining gaze and emg, *CHI '08 Extended Abstracts on Human Factors in Computing Systems (CHI EA '08)*, pp.3039–3044, DOI: 10.1145/1358628.1358804 (2008).
- [3] Pfeuffer, K., Alexander, J., Chong, M.K. and Gellersen, H.: Gaze-touch: Combining gaze with multi-touch for interaction on the same surface, *Proc. 27th Annual ACM Symposium on User Interface Software and Technology (UIST '14)*, pp.509–518, DOI: 10.1145/2642918.2647397 (2014).
- [4] Pfeuffer, K., Alexander, J., Chong, M.K., Zhang, Y. and Gellersen, H.: Gaze-Shifting: Direct-Indirect Input with Pen and Touch Modulated by Gaze, *Proc. 28th Annual ACM Symposium on User Interface Software & Technology (UIST '15)*, pp.373–383, DOI: 10.1145/2807442.2807460 (2015).
- [5] Jacob, R.J.K.: 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, DOI: 10.1145/123078.128728 (1991).
- [6] Miniotos, D., Špakov, O. and MacKenzie, I.S.: Eye gaze interaction with expanding targets, *CHI '04 Extended Abstracts on Human Factors in Computing Systems (CHI EA '04)*, pp.1255–1258, DOI: 10.1145/985921.986037 (2004).
- [7] Skovsgaard, H., Mateo, J.C., Flach, J.M. and Hansen, J.P.: Small-target selection with gaze alone, *Proc. 2010 Symposium on Eye-Tracking Research & Applications (ETRA '10)*, pp.145–148, DOI: 10.1145/1743666.1743702 (2010).
- [8] Ashmore, M., Duchowski, A.T. and Shoemaker, G.: Efficient eye pointing with a fisheye lens, *Proc. Graphics Interface 2005 (GI '05)*, pp.203–210 (2005).
- [9] Špakov, O. and Miniotos, D.: Gaze-based selection of standard-size menu items, *Proc. 7th International Conference on Multimodal Interfaces (ICMI '05)*, pp.124–128, DOI: 10.1145/1088463.1088486 (2005).
- [10] Raudsandmoen, H. and Rødsjø, B.: Empirically Based Design Guidelines for Gaze Interaction in Windows 7, Master thesis, Norwegian University of Science and Technology, Trondheim, Norway (2012), available from <http://hdl.handle.net/11250/253070>.
- [11] Grossman, T. and Balakrishnan, R.: The bubble cursor: Enhancing target acquisition by dynamic resizing of the cursor's activation area, *Proc. SIGCHI Conference on Human Factors in Computing Systems (CHI '05)*, pp.281–290, DOI: 10.1145/1054972.1055012 (2005).
- [12] Fitts, P.M.: The information capacity of the human motor system in controlling the amplitude of movement, *Journal of Experimental Psychology*, Vol.47, No.6, pp.381–391, DOI: 10.1037/h0055392 (1954).
- [13] MacKenzie, I.S.: Fitts' law as a research and design tool in human-computer interaction, *Human Computer Interaction*, Vol.7, No.1, pp.91–139, DOI: 10.1207/s15327051hci0701_3 (1992).
- [14] MacKenzie, I.S., Sellen, A. and Buxton, W.A.S.: A comparison of input devices in element pointing and dragging tasks, *Proc. SIGCHI Conference on Human Factors in Computing Systems (CHI '91)*, Robertson, S.P., Olson, G.M. and Olson, J.S. (Eds.), pp.161–166, DOI: 10.1145/108844.108868 (1991).
- [15] Ware, C. and Mikaelian, H.H.: An evaluation of an

- eye tracker as a device for computer input, *Proc. SIGCHI/GI Conference on Human Factors in Computing Systems and Graphics Interface (CHI '87)*, Carroll, J.M. and Tanner, P.P. (Eds.), pp.183–188, DOI: 10.1145/29933.275627 (1986).
- [16] Miniotos, D.: Application of Fitts' law to eye gaze interaction, *CHI '00 Extended Abstracts on Human Factors in Computing Systems (CHI EA '00)*, pp.339–340, DOI: 10.1145/633292.633496 (2000).
- [17] Sibert, L.E. and Jacob, R.J.K.: Evaluation of eye gaze interaction, *Proc. SIGCHI Conference on Human Factors in Computing Systems (CHI '00)*, pp.281–288, DOI: 10.1145/332040.332445 (2000).
- [18] Vertegaal, R.: A Fitts Law comparison of eye tracking and manual input in the selection of visual targets, *Proc. 10th International Conference on Multimodal Interfaces (ICMI '08)*, pp.241–248, DOI: 10.1145/1452392.1452443 (2008).
- [19] Surakka, V., Illi, M. and Isokoski, P.: Gazing and frowning as a new human-computer interaction technique, *ACM Trans. Appl. Percept.*, Vol.1, No.1, pp.40–56, DOI: 10.1145/1008722.1008726 (2004).
- [20] Zhang, X., Ren, X. and Zha, H.: Modeling dwell-based eye pointing target acquisition, *Proc. SIGCHI Conference on Human Factors in Computing Systems (CHI '10)*, pp.2083–2092, DOI: 10.1145/1753326.1753645 (2010).
- [21] Gori, J., Rioul, O., Guiard, Y. and Beaudouin-Lafon, M.: The Perils of Confounding Factors: How Fitts' Law Experiments can Lead to False Conclusions, *Proc. 2018 CHI Conference on Human Factors in Computing Systems (CHI '18)*, pp.1–10, DOI: 10.1145/3173574.3173770 (2018).
- [22] Drewes, H.: A Lecture on Fitts' law (2013), available from ([http://www.cip.ifi.lmu.de/~drewes/science/fitts/A Lecture on Fitts Law.pdf](http://www.cip.ifi.lmu.de/~drewes/science/fitts/A%20Lecture%20on%20Fitts%20Law.pdf)) (accessed 2019-11-15).
- [23] Špakov, O.: Comparison of eye movement filters used in HCI, *Proc. Symposium on Eye Tracking Research and Applications (ETRA '12)*, Spencer, S.N. (Ed.), pp.281–284, DOI: 10.1145/2168556.2168616 (2012).
- [24] Feit, A.M., Williams, S., Toledo, A., Paradiso, A., Kulkarni, H., Kane, S. and Morris, M.R.: Toward Everyday Gaze Input: Accuracy and Precision of Eye Tracking and Implications for Design, *Proc. 2017 CHI Conference on Human Factors in Computing Systems (CHI '17)*, pp.1118–1130, DOI: 10.1145/3025453.3025599 (2017).
- [25] Schenk, S., Dreiser, M., Rigoll, G. and Dorr, M.: GazeEverywhere: Enabling Gaze-only User Interaction on an Unmodified Desktop PC in Everyday Scenarios, *Proc. 2017 CHI Conference on Human Factors in Computing Systems (CHI '17)*, pp.3034–3044, DOI: 10.1145/3025453.3025455 (2017).
- [26] Jacob, R.J.K.: Eye movement-based human-computer interaction techniques: Toward non-command interfaces, *Advances in Human-Computer Interaction*, Vol.4, pp.151–190 (1993).
- [27] Hornof, A.J. and Cavender, A.: EyeDraw: Enabling children with severe motor impairments to draw with their eyes, *Proc. SIGCHI Conference on Human Factors in Computing Systems (CHI '05)*, pp.161–170, DOI: 10.1145/1054972.1054995 (2005).
- [28] Kumar, M., Klingner, J., Puranik, R., Winograd, T. and Paepcke, A.: Improving the accuracy of gaze input for interaction, *Proc. 2008 Symposium on Eye Tracking Research & Applications (ETRA '08)*, pp.65–68, DOI: 10.1145/1344471.1344488 (2008).
- [29] Casiez, G., Roussel, N. and Vogel, D.: 1 € filter: A simple speed-based low-pass filter for noisy input in interactive systems, *Proc. SIGCHI Conference on Human Factors in Computing Systems (CHI '12)*, pp.2527–2530, DOI: 10.1145/2207676.2208639 (2012).
- [30] Zhang, X., Ren, X. and Zha, H.: Improving eye cursor's stability for eye pointing tasks, *Proc. SIGCHI Conference on Human Factors in Computing Systems (CHI '08)*, pp.525–534, DOI: 10.1145/1357054.1357139 (2008).
- [31] McGuffin, M. and Balakrishnan, R.: Acquisition of expanding targets, *Proc. SIGCHI Conference on Human Factors in Computing Systems (CHI '02)*, pp.57–64, DOI: 10.1145/503376.503388 (2002).
- [32] Zhai, S., Conversy, S., Beaudouin-Lafon, M. and Guiard, Y.: Human on-line response to target expansion, *Proc. SIGCHI Conference on Human Factors in Computing Systems (CHI '03)*, pp.177–184, DOI: 10.1145/642611.642644 (2003).
- [33] MacKenzie, I.S.: An eye on input: Research challenges in using the eye for computer input control, *Proc. 2010 Symposium on Eye-Tracking Research & Applications (ETRA '10)*, pp.11–12, DOI: 10.1145/1743666.1743668 (2010).
- [34] Majaranta, P. and Riih a, K.-J.: Twenty years of eye typing: Systems and design issues, *Proc. 2002 Symposium on Eye Tracking Research & Applications (ETRA '02)*, pp.15–22, DOI: 10.1145/507072.507076 (2002).
- [35] Porta, M., Ravarelli, A. and Spagnoli, G.: ceCursor, a contextual eye cursor for general pointing in windows environments, *Proc. 2010 Symposium on Eye-Tracking Research & Applications (ETRA '10)*, pp.331–337, DOI: 10.1145/1743666.1743741 (2010).
- [36] Kabbash, P. and Buxton, W.A.S.: The “prince” technique: Fitts' law and selection using area cursors, *Proc. SIGCHI Conference on Human Factors in Computing Systems (CHI '95)*, Katz, I.R., Mack, R., Marks, L., Rosson, M.B. and Nielsen, J. (Eds.), pp.273–279, DOI: 10.1145/223904.223939 (1995).
- [37] Worden, A., Walker, N., Bharat, K. and Hudson, S.: Making computers easier for older adults to use: Area cursors and sticky icons, *Proc. ACM SIGCHI Conference on Human Factors in Computing Systems (CHI '97)*, pp.266–271, DOI: 10.1145/258549.258724 (1997).
- [38] Penkar, A.M., Lutteroth, C. and Weber, G.: Designing for the eye: Design parameters for dwell in gaze interaction, *Proc. 24th Australian Computer-Human Interaction Conference (OzCHI '12)*, Farrell, V., Farrell, G., Chua, C., Huang, W., Vasa, R. and Woodward, C. (Eds.), pp.479–488, DOI: 10.1145/2414536.241460 (2012).
- [39] Zhang, X., Xu, P., Zhang, Q. and Zha, H.: Speed-accuracy trade-off in dwell-based eye pointing tasks at different cognitive levels, *Proc. 1st International Workshop on Pervasive Eye Tracking & Mobile Eye-based Interaction (PETMEI '11)*, pp.37–42, DOI: 10.1145/2029956.2029967 (2011).
- [40] Hart, S.G. and Staveland, L.E.: Development of NASA-TLX (Task Load Index): Results of Empirical and Theoretical Research, Human Mental Workload, *Advances in Psychology*, Vol.52, pp.139–183, DOI: 10.1016/S0166-4115(08)62386-9 (1988).
- [41] Brooke, J.: SUS—A quick and dirty usability scale, *Usability Evaluation in Industry*, pp.189–194 (1966).

- [42] Microsoft: Get started with eye control in Windows 10, available from (<https://support.microsoft.com/en-ca/help/4043921/windows-10-get-started-eye-control>) (accessed 2019-11-15).
- [43] Nagamatsu, T., Yamamoto, M. and Sato, H.: MobiGaze: Development of a gaze interface for handheld mobile devices, *CHI '10 Extended Abstracts on Human Factors in Computing Systems (CHI EA '10)*, pp.3349–3354, DOI: 10.1145/1753846.1753983 (2010).
- [44] Khamis, M., Alt, F. and Bulling, A.: The past, present, and future of gaze-enabled handheld mobile devices: Survey and lessons learned, *Proc. 20th International Conference on Human-Computer Interaction with Mobile Devices and Services (MobileHCI '18)*, Article 38, pp.1–17, DOI: 10.1145/3229434.3229452 (2018).
- [45] Gibaldi, A., Vanegas, M., Bex, P.J. and Maiello, G.: Evaluation of the Tobii EyeX Eye tracking controller and Matlab toolkit for research, *Behavior Research Methods*, Vol.49, No.3, p.923–946, DOI: 10.3758/s13428-016-0762-9 (2017).
- [46] Huang, M.X., Li, J., Ngai, G. and Va Leong, H.: ScreenGlint: Practical, In-situ Gaze Estimation on Smartphones, *Proc. 2017 CHI Conference on Human Factors in Computing Systems (CHI '17)*, pp.2546–2557, DOI: 10.1145/3025453.3025794 (2017).
- [47] Huang, Q., Veeraraghavan, A. and Sabharwal, A.: TabletGaze: Dataset and analysis for unconstrained appearance-based gaze estimation in mobile tablets, *Mach. Vision Appl.*, Vol.28, No.5–6, pp.445–461, DOI: 10.1007/s00138-017-0852-4 (2017).
- [48] Mott, M.E. and Wobbrock, J.O.: Beating the bubble: Using kinematic triggering in the bubble lens for acquiring small, dense targets, *Proc. SIGCHI Conference on Human Factors in Computing Systems (CHI '14)*, pp.733–742, DOI: 10.1145/2556288.2557410 (2014).
- [49] Chapuis, O., Labrune, J.-B. and Pietriga, E.: DynaSpot: Speed-dependent area cursor, *Proc. SIGCHI Conference on Human Factors in Computing Systems (CHI '09)*, pp.1391–1400, DOI: 10.1145/1518701.1518911 (2009).



崔明根

2019年北海道大学工学部情報エレクトロニクス学科卒業。現在、北海道大学大学院情報科学院修士課程在学中。現在、視線を用いたインタラクションに関する研究に従事。



坂本 大介 (正会員)

2008年公立ほこだて未来大学大学院システム情報科学研究科博士(後期)課程修了。博士(システム情報科学)。国際電気通信基礎技術研究所(ATR)でインターン、東京大学で日本学術振興会特別研究員PD, JST ERATO 五十嵐デザインインタフェースプロジェクト研究員, 東京大学大学院情報理工学系研究科コンピュータ科学専攻助教, 特任講師を経て, 現在, 北海道大学大学院情報科学研究科准教授。人とロボットを含む情報環境とのインタラクション設計に関する研究に従事。



小野 哲雄 (正会員)

1997年北陸先端科学技術大学院大学情報科学研究科博士後期課程修了。同年(株)ATR 知能映像通信研究所客員研究員。2001年公立ほこだて未来大学情報アーキテクチャ学科助教授, 2005年同学科教授。2009年北海道大学大学院情報科学研究科教授, 現在に至る。博士(情報科学)。ヒューマンエージェント/ロボットインタラクション(HAI/HRI), インタラクティブシステムに関する研究に従事。電子情報通信学会, ヒューマンインタフェース学会, 認知科学会, ACM 各会員。