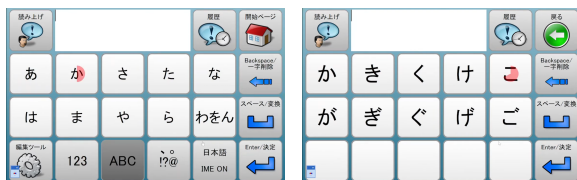


FixedPointStart: 視線操作を用いた文字入力手法の提案と評価 FixedPointStart: Text Entry Method using Eye Gestures

倉知 慎太郎¹⁾ 郷 健太郎²⁾
Shintaro Kurachi Kentaro Go

1 はじめに

視線操作による文字入力は、発話や手指入力が困難な人々がコミュニケーションをとる上で有効な手段である。例えば筋萎縮性側索硬化症 (ALS) [1] の患者がコミュニケーション時に使用している。ALS 患者が利用する文字入力装置の多くはソフトウェアキーボードのボタン上を一定時間注視することで文字を決定する。実際に扱われている重度障害者用意思伝達装置 TC スキャン [2] では、あ段 (あ～わ) の 10 個のボタンのどれかを選択すると、その行が展開され最終的な文字の選択が可能となる。表示可能なボタン数に限りがあるため、1 つの文字を入力するために 2 回の選択が必要となる。1 回ボタンを選択するには 2 秒かかるため、1 文字を入力するために最低 4 秒かかることとなる。



(a) 「か行」の選択

(b) 「こ」の選択

図 1 TC スキャンによる「こ」の入力 [2]

この問題を解決する手法の一つが EyeWrite [3] である。EyeWrite とは視線操作を用いたシングルストロークジェスチャによる文字入力手法である。この手法は矩形の入力領域の角の通過順序で文字を表現する。固定された入力領域で多くの文字や特殊キーを表現可能であり、画面の遷移を挟む必要はない。

一方で、EyeWrite には実用上の課題が 2 つある。1 つ目は入力開始点の複数あり誤入力を誘発する点である。各入力文字の開始点が矩形の入力領域の 4 つの角のどれかに割り当てられており、開始点を間違えた場合その時点で正しい文字の入力を行うことはできない。2 つ目は画数の多い文字の入力難度が高くなる点である。EyeWrite で採用されるシングルストロークジェスチャは、より複雑な文字形状をもつ言語や特殊文字の表現には適していないと考えられる。

これらの課題を解決するために、本研究では FixedPointStart という文字入力手法を提案する。本手法では入力開始点の統一し、さらにマルチストロークジェスチャに対応する。既存手法 EyeWrite と提案手法 FixedPointStart とを実装し、実験を行ってその効果を明らかにする。

- 1) 山梨大学大学院医工農学総合教育部 Integrated Graduate School of Medicine, Engineering, and Agricultural Sciences, University of Yamanashi
- 2) 山梨大学大学院総合研究部 Interdisciplinary Graduate School, University of Yamanashi

2 関連研究

2.1 EdgeWrite [3]

EdgeWrite は運動障害をもつ人々のために設計された小型機器向けの一筆書き文字入力手法である。矩形の入力領域の縁 (辺) と対角線に沿うようにして文字を入力する。これにより文字の形状ではなく角の通過順序で文字が認識され、高い精度での入力が可能である。

図 2 は EdgeWrite で扱われる入力ジェスチャの一部である。それぞれの英語アルファベット (以下、アルファベット) の大文字もしくは小文字に似た形状で一筆書きが表現されている。

この手法を用いる利点として、小さい入力フィールドでも活用可能であることや、手を上手く動かせない人でもペン先を縁に押し当てて導ることで入力が可能となることが挙げられる。この技術は入力の安定性と視覚に頼らず触覚のみでの操作を可能にする。

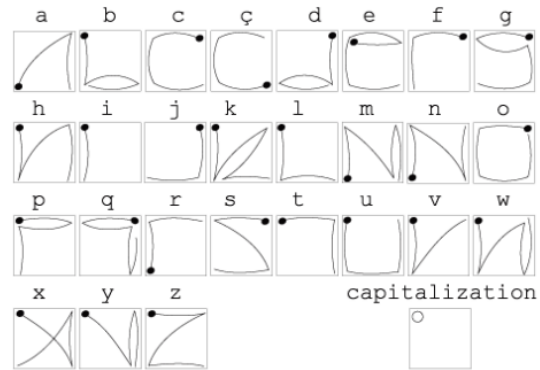
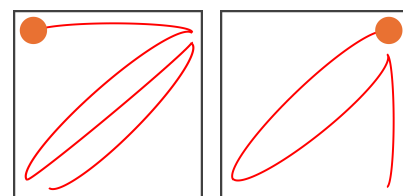


図 2 EdgeWrite アルファベットの入力ジェスチャ [3]

2.2 Katakana EdgeWrite [4]

アルファベットに対する EdgeWrite の概念を拡張し、日本語に対して設計したものが Katakana EdgeWrite である。本手法では、矩形の入力領域の縁 (辺) と対角線に沿うようにカタカナ文字を入力する。

図 3 はカタカナの入力ジェスチャの一例である。入力開始点は文字の形状に従って定まる。複数画ある文字は間を補完するジェスチャが含まれることもあり、図 3(a) 及び図 3(b) では 1 画目の終了地点 (左下) から 2 画目の開始地点 (右上) に補完を行っている。



(a) 「ア」の入力

(b) 「イ」の入力

図 3 Katakana EdgeWrite の入力ジェスチャ [4]

2.3 EyeWrite [5]

Wobbrock らは視線操作によって EdgeWrite を実現する EyeWrite を実装して評価を行った。EyeWrite を実装する上で EdgeWrite と仕組みが異なる点は、文字の終了の指定方法である。既存の EdgeWrite はペン入力を想定していたため、ペンが入力領域から離れた瞬間を入力ジェスチャの終了とみなして文字を認識していた。一方で、視線は途切れのない連続的なジェスチャ入力であるため、シングルストロークでの入力ジェスチャの終了を何らかの手段で指定しなければならない。そのため、矩形の入力領域の中央部にある判定領域を一定時間注視することで入力の終了を指定するように実装した (図 4)。

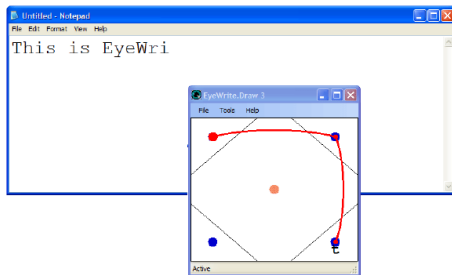


図 4 EyeWrite: 「T」の入力 [5]

Wobbrock らは EyeWrite の入力性能を評価するため、視線操作によるオンスクリーンキーボード入力と EyeWrite 入力とを比較する実験を行った。20 代の 8 名の実験参加者に対して 14 セッションの入力を行ってもらい、入力速度とエラー率、主観評価値の変化を計測した。その結果、セッション全体での平均入力速度は、オンスクリーンキーボード入力が 7.03 WPM (SD=2.65) であり、EyeWrite が 4.87 WPM (SD=1.43) であった。EyeWrite の初期のセッションでは約 2.5WPM であり、いずれの入力手法もセッションが進むにつれて参加者の入力性能は向上して、第 14 セッションの EyeWrite の入力速度は、第 2 セッションのオンスクリーンキーボードの入力速度に匹敵する値 (約 6WPM) を示した。

3 FixedPointStart

本研究では、EyeWrite と Katakana EdgeWrite の特徴と課題を分析してこれを改善する入力手法を提案する。

3.1 従来手法の特徴と課題

3.1.1 入力ジェスチャの開始位置

EyeWrite でのアルファベットの入力には図 2 の入力パターンを用いる。文字によって入力ジェスチャの開始位置の角が異なる。例えば「A」は左下の角から書き出しを始めるが、「B」は左上、「C」は右上となっている。

すなわち、入力開始位置を誤ってしまった場合、その時点で目的の文字を入力することができずに入力誤りになってしまう。このことがエラー率を上げてしまっていると考えられる。

3.1.2 シングルストローク用の入力領域

EdgeWrite の入力パターンは元の文字の形状を再現するように作成される。アルファベットは全体的に単純な形状が多いため、単純な入力パターンで表現が可能であり、ストローク数の少ない入力ジェスチャ、具体的にはシングルストロークジェスチャを設計することができた。

一方でカタカナの場合には、EdgeWrite の矩形領域で表現するには複雑な文字が多い。特に画数が多い文字は、入力ジェスチャのデザインに多数のストロークが必要となる。さらに、EdgeWrite の矩形領域で複数の画数の文字を表現するためには、各ストロークの終了位置から開始位置への補完ジェスチャが増えることになる。これは 3.1.1 で述べた入力ジェスチャの開始位置の課題をもたらすことになり、入力速度の低下やエラーを誘発する要因になりうる。

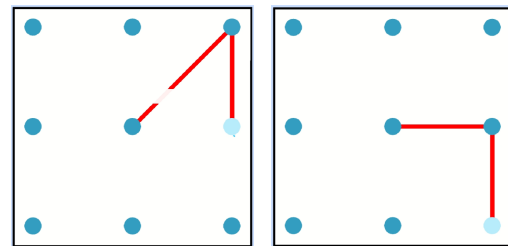
3.2 提案手法

前述の課題を克服する手法として FixedPointStart を提案する。

3.2.1 入力開始点の統一

3.1.1 の課題を解決するために、入力開始位置を一箇所に統一する。

FixedPointStart では入力領域となる矩形内に合計 9 箇所の判定位置を設置する。具体的には、矩形領域の中心と中心から放射方向にのびる 8 つの位置である (図 5)。これらの 9 箇所のうち、入力開始点を中心 1 つに統一する。これによって、誤入力や補完ジェスチャの減少が期待でき、視線によるより効率的なジェスチャ入力を実現できると考えられる。



(a) 「A」の入力

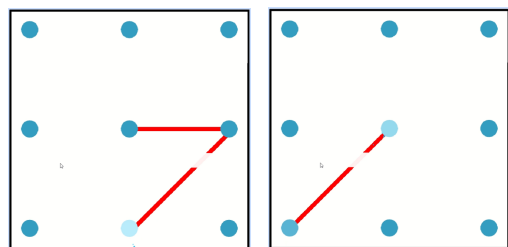
(b) 「T」の入力

図 5 入力開始点の統一

3.2.2 マルチストロークの実装

3.1.2 の課題を解決するために、マルチストローク入力機能を実装した。

マルチストローク入力機能とは、筆順を分割して入力できる機能である。例えば、「ア」を記述したいときは、図 6(a) のように中心から 1 画目のストロークを描いた後に再び中心に視線を戻し、図 6(b) のように 2 画目のストロークを記述することで入力ができる。2.2 で説明したシングルストロークジェスチャより実際のカタカナに近い書き方ができるだけでなく、補完ジェスチャの長さも短縮でき、ストロークの開始点の誤りも減るため、入力効率の向上や誤入力の減少が期待できる。



(a) 「ア」の一画目の入力

(b) 「ア」の二画目の入力

図 6 カタカナ入力に対するストロークの分割表現

4 評価実験

従来手法 EyeWrite と提案手法 FixedPointStart の評価実験を行う。

4.1 入力フレーズと視線ジェスチャ

本実験では、実験参加者は無作為に表示されるアルファベットもしくはカタカナのフレーズ（提示文）を、視線入力によって入力フィールドに転記する。アルファベットのフレーズには Mckenzie らのフレーズセット [6] を用いた。また、カタカナのフレーズには、ChatGPT で生成した 120 個のパングラムを用いた。提示されたフレーズと全く同じ文字列を入力することが実験課題の終了条件である。入力文字が誤った場合には、消去ジェスチャを入力した上で、文字を再入力する。

アルファベット入力用の視線ジェスチャには、両手法ともに図 2 の入力ジェスチャを採用した。カタカナ入力用の視線ジェスチャには、両手法ともに [4] の入力ジェスチャに基づき、複数文字に対して重複したジェスチャは再設計を行った（例：マ、ヌ、スに対するジェスチャ）。また [3] に基づき、文字以外を示すジェスチャとしてスペースとバックスペースの 2 種類を用意した。スペースは左から右へのシングルストローク、バックスペースは右から左へのシングルストロークである。

4.2 実験用システム

図 7 に実験用システムの入力画面を示す。本システムでは、各手法の入力フィールドの中央上部に提示文を表示し、中央下部に実験参加者が現在入力している文字列が表示される。

本システムには 2 つの入力支援機能を実装した。1 つ目は、入力フレーズにおいて次に入力すべき文字が赤く表示される機能である。この機能はユーザが次に入力すべき文字を見失うことを防ぐ。2 つ目は入力ジェスチャの一覧表の提示機能である。この機能はユーザが入力ジェスチャを忘れた場合に使用するヘルプ機能である。

これらに加え、本システムではジェスチャの終了を指定する方法として瞬きを採用した。本機能によりジェスチャ間の待ち時間を減少させることが期待できる。

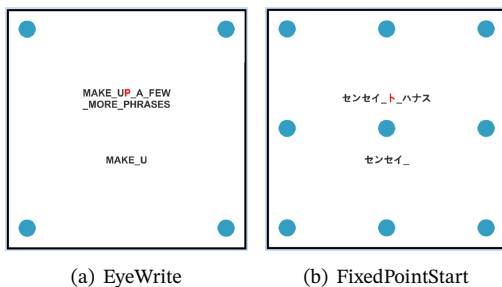


図 7 実験用システムの入力画面

4.3 実験手順

まず、計測時のディスプレイと実験参加者の顔の距離を 60 cm に統制した。次に視線に関して Tobii Experience を使用してキャリブレーションを行った。実験途中でも、必要に応じてキャリブレーションを実施した。

実験には 8 名（平均年齢 21.75, SD=0.83, 男性 7 名, 女性 1 名）の大学生・大学院生に参加してもらった。アルファベット入力実験とカタカナ入力実験にそれぞれ 4

人ずつを割り当てた。実験参加者は、それぞれの入力実験で EyeWrite と FixedPointStart の 2 手法で文字入力を行った。手法の提示順序は均等になるようにバランスをとった。各手法の実験終了後には質問紙に回答してもらった。質問紙は 5 段階のリッカート尺度（1：全くそう思わない, 5：とてもそう思う）を用いて評価した。

4.4 実験結果

実験参加者からは、アルファベットとして EyeWrite が 1,416 文字、FixedPointStart が 1,538 文字の計測データを取得することができた。同様に、カタカナとして EyeWrite が 1,120 文字、FixedPointStart が 1,189 文字の計測データを取得することができた。

文字入力速度の指標として、アルファベットには 1 分間あたりの入力単語数 (Words Per Minute, WPM) を、カタカナには 1 分間あたりの入力文字数 (Characters Per Minute, CPM) を使用する。

4.4.1 アルファベット入力

実験参加者あたりのアルファベットの入力速度の違いを図 8 に示す。4 名全員に対して、EyeWrite と比較して FixedPointStart の方が入力効率が高い傾向がみられた。

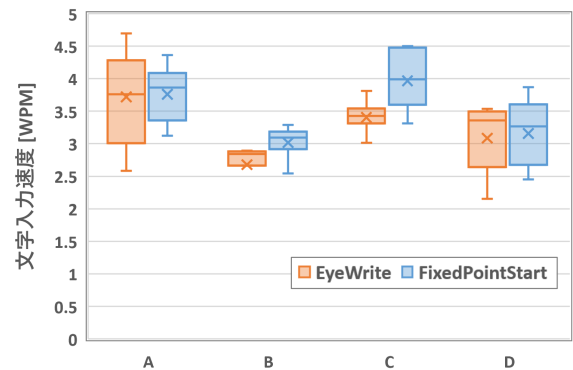


図 8 実験参加者毎のアルファベット入力速度の違い

アルファベット入力のエラー率を図 9 に示す。実験参加者 4 人ともに FixedPointStart の方がエラー率が低いという結果がみられた。t 検定の結果、有意水準 5% として参加者 A と C に有意差がみられた。A は EyeWrite が平均 10.7% (SD=9.3), FixedPointStart が平均 4.5% (SD=3.6), $t(18) = 2.299, p = 0.033$ であり, C は EyeWrite が平均 6.0% (SD=3.8), FixedPointStart が平均 1.7% (SD=2.6), $t(20) = 3.100, p = 0.006$ であった。

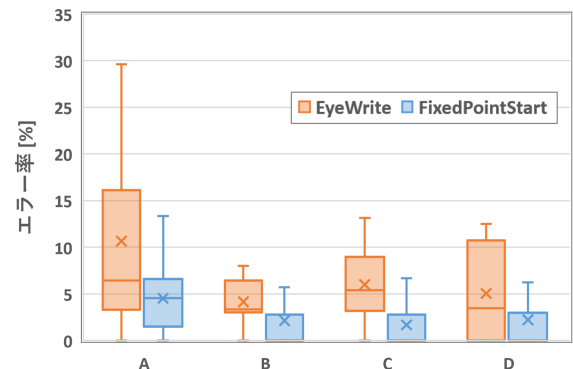


図 9 実験参加者毎のアルファベット入力エラー率の違い

4.4.2 カタカナ入力

実験参加者あたりのカタカナの入力速度の違いを図 10 に示す。実験参加者 E と G は FixedPointStart の方が速く、F と H は EyeWrite の方が速いという結果となった。t 検定の結果、有意水準 5%として参加者 F だけに有意差がみられた。EyeWrite が平均 18.1CPM (SD=3.6), FixedPointStart が平均 14.8CPM (SD=2.5) で $t(46) = 3.633, p = 0.001$ となった。

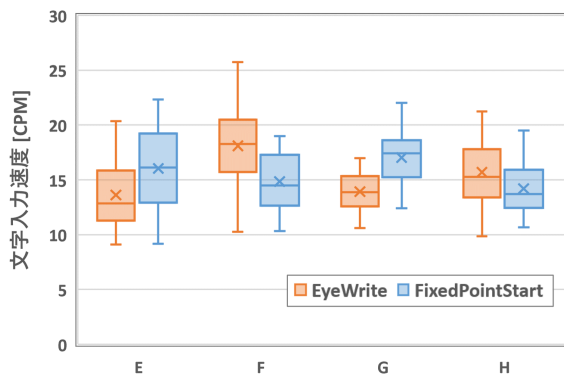


図 10 実験参加者毎のカタカナ入力速度の違い

カタカナ入力のエラー率を図 11 に示す。実験参加者 4 名全員に対して、FixedPointStart のエラー率の方が、EyeWrite のエラー率より低い傾向が示された。

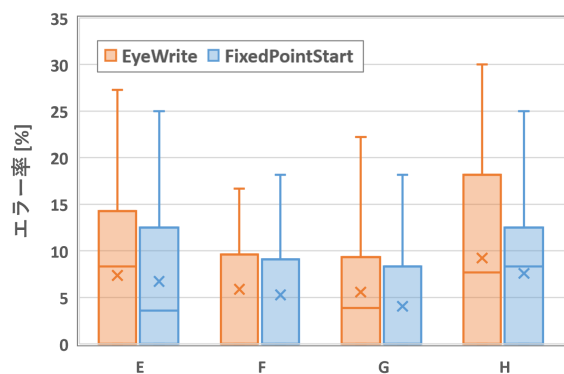


図 11 実験参加者毎のカタカナ入力エラー率の違い

4.4.3 エラーの内訳

入力エラーの内訳は次の 3 つに分類できる：(1) 入力の開始点異なる、(2) 入力途中でジェスチャを間違える、(3) 入力中に誤って瞬きを行う。FixedPointStart では (1) のエラーは仕様上発生しないことから、EyeWrite と比べエラー率が低くなっているか考えられる。また、(3) のエラーはジェスチャが複雑であるほど高い傾向にある。実際に本実験では、カタカナ入力がアルファベット入力と比べて全体的にエラー率が高い傾向を示した。

4.5 質問紙

質問紙による主観評価によると、「思った文字を的確に入力できる」という設問に対して、アルファベット入力では、EyeWrite が平均 3.00, FixedPointStart が平均 4.00 となった。同様に、カタカナ入力では、EyeWrite が平均 3.50, FixedPointStart が平均 4.25 となった。この結果から EyeWrite よりも FixedPointStart の方が、誤入力防止の仕組みや使用性に優れた印象をもつことが示された。

「文字ごとの入力ジェスチャは覚えやすい」という設問に対しては、カタカナ入力の実験では、EyeWrite が平均 2.25, FixedPointStart が平均 4.50 となった。この結果から FixedPointStart の方が、カタカナの入力に適した視線ジェスチャを表現可能であるといえる。

4.6 考察

アルファベット入力の実験の結果、入力速度とエラー率に関して FixedPointStart の方が EyeWrite よりも性能が高い傾向にあることから、入力開始位置の統一には効果があると考えられる。一方で、カタカナ入力に関しては実験順序が交絡変数となっているようにみえる。具体的には、実験参加者 E と G は先に EyeWrite を、後に FixedPointStart の実験を行った。これに対して実験参加者 F と H はその逆の順序で実験を行った。すなわち図 10 から、後に実験を行った手法の方が入力速度が速いという解釈を否定することができない。この傾向がより一般的なのかどうかについては、被験者数を増やして追加検証を試みたい。

全体をとおして、FixedPointStart のエラー率が EyeWrite のエラー率より低い傾向が示された。このことは、入力開始点のエラーを減らすという FixedPointStart のデザインの効果を表していると考えられる。また実験者の観察によると、EyeWrite では瞬きの後に直前に見ていた点が始点が決まってしまうという問題が生じていた。このような特徴もエラー率に影響を与えたようであり、EyeWrite への瞬き入力の導入にはさらなる工夫が求められる。

5 おわりに

本研究では、視線操作を用いた文字入力について、既存手法と提案手法を実装して評価を行った。実験結果から、提案手法である FixedPointStart には一定の改善効果が認められた。今後の課題として、より多くの参加者や熟練者による実験を行った上で、さらなる手法の改善を行う予定である。

参考文献

- [1] ALS ステーション, 2020, 「ALS コミュニケーションツール体験レポート」, <https://als-station.jp/alsaction/communication/> (accessed 2023-12-13).
- [2] Creact, 2022, 「重度障害者用意思伝達装置 TC スキャン」, <https://www.creact.co.jp/welfare/communication-device/tcscan/index.html> (accessed 2024-01-12).
- [3] Wobbrock, J. O., Myers, B. A., and Kembel, J. A., 2003, “EdgeWrite: A Stylus-based Text Entry Method Designed for High Accuracy and Stability of Motion,” In *User Interface Software and Technology (UIST)*, ACM, New York, NY, pp. 61–70.
- [4] Go, K. and Kinoshita, Y., 2014, “Katakana EdgeWrite: An EdgeWrite Version for Japanese Text Entry,” *IEICE Transactions Online*, August 1, 2014.
- [5] Wobbrock, J. O., Rubinstein, J., Sawyer, M. W., and Duchowski, A. T., 2008, “Longitudinal Evaluation of Discrete Consecutive Gaze Gestures for Text Entry,” *Proceedings of the 2008 Symposium on Eye Tracking Research & Applications*, pp. 11–18.
- [6] MacKenzie, I. S., and Soukoreff, R. W., 2003, “Phrase Sets for Evaluating Text Entry Techniques,” *Extended Abstracts of the ACM Conference on Human Factors in Computing Systems (CHI 2003)*, ACM, New York, pp. 754–755.