

## 動的分割を実現した視線制御インタフェース

松金朗啓† 湯浅将英† 阿部清彦†‡ 大井尚一‡ 大山 実†  
 東京電機大学 大学院 情報環境学研究科† 関東学院大学 工学部 情報ネット・メディア工学科‡ †  
 東京電機大学 工学部 電子工学科 †

### 1. はじめに

これまで広く視線入力インタフェースの研究がなされてきた。視線入力には、自然光を用いた方式<sup>[1]</sup>と赤外線を用いた方式の二つがある。自然光を用いた方式では、赤外線のような特殊光線を眼球に照射する必要がない。従って、赤外線を眼球へ照射することの影響が未知である事を考えれば、その安全性は高い。しかし、自然光方式は赤外線方式と比べ、水平方向に5段階、垂直方向に3段階の視線方向が認識できる程度なので入力精度に欠ける。そのため、自然光を用いた方式では少ない選択肢で多数の選択対象からターゲットを選ぶ必要がある。また、視線でWebページを操作する手法についてこれまで様々な提案がされてきたが、自然光を用いた視線入力においては、選択対象とは別に指標を準備し、指標を直視することによりカーソルをターゲットに移動させる手法<sup>[2][3]</sup>が殆どである。

本研究は、少ない選択肢でデスクトップ上のアイコンやWeb ページ上のターゲットを選ぶ新しい方法の実現を目的とする。

### 2. 視線制御インタフェースの問題点

自然光を用いた視線制御インタフェースの問題点は、その入力精度が低いため利用できる選択肢の数が少ない事である。そのため、画面上のあるポイントという形で細かく座標指定することができない。そこで本研究では、選択対象を絞り込むため、ターゲットが含まれている領域の選択と拡大という手法を繰り返し適用することで、目的の画面上のポイントを指定する方法を提案する。

### 3. Grid-Based Mechanism

Grid-based Mechanism<sup>[4]</sup>は音声インタフェースで考案された考え方で、図1のように画面をグリッドで分割する。ターゲットが含まれているグリッドを選択すると、選択されたグリッドをさらに分割する。これを繰り返し、画面上の特定のターゲットが含まれている位置を絞り込むことでターゲットが選択できる。

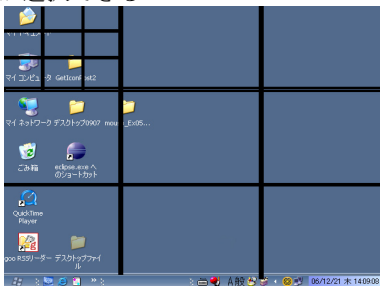


図1 : Grid-Based Mechanism

Gaze-Controlled Interface Using Grid-Based Mechanism  
 † Graduate School of Information Environment,  
 Tokyo Denki University  
 ‡ School of Engineering, Tokyo Denki University  
 † ‡ Department of Network and Multi-Media Engineering,  
 Kanto Gakuin University

### 4. Grid-Based Mechanism の視線入力への適用

文献4で提案された Grid-Based Mechanism は、画面をグリッドに分割する操作を繰り返し、グリッド面積を小さくしていく方式であり、視線入力の精度を考慮すると、そのまま視線入力に適用することはできない。そこで以下の方式を採用した。Grid-based Mechanism を利用して、画面全体をグリッドに分割する。ユーザが、ターゲットが含まれているグリッドを選択すると、選択されたグリッドを画面全体に拡大表示する。選択され拡大されたグリッドの中にまだ複数の選択対象があれば、さらにいくつか分割し選択してもらう。これを繰り返し、グリッド内の選択対象が1つになるまで絞り込んでターゲットの選択をする。この方式は選択対象を直視すれば選択ができるため、これまでの指標を用いてカーソルを移動させる間接的な選択方法より自然な選択ができる。

### 5. 画面分割アルゴリズム

本稿では、選択対象の存在位置を考慮して、画面の分割数を動的に変化させる手法の提案をする。視線制御インタフェースの性質上、選択肢の数が少ない(=画面の分割数が少ない)ほど認識率が上がる<sup>[1]</sup>ので、選択対象の数に応じて9分割より画面分割数を減すこと(8~2分割:図2)で正確に、そして速く選択することができる。

以下に分割アルゴリズムを示す。

1. 全ての分割方法で分割した場合について、各グリッド内に含まれる選択対象の数を降順に並べる
  2. 配置が2.1を満たさない場合2.2を行う
  - 2.1 選択対象が縦または、横方向のみに一列に並んでいる場合は、2または3分割を適用する
  - 2.2 グリッド内に含まれる選択対象の最大値が1になり、かつ最も分割数が少ない分割方法を採用する。
  3. 8分割以下で2に該当する分割方法が無い場合、9分割を行い1からやり直す
- これをグリッドごとに繰り返し、なるべく少ない分割方法を選択する。

### 6. 比較実験

少ない分割数を適用する有効性を示すため、単純に9分割のみを行う場合と、動的分割アルゴリズムを用いて分割数を変化させる手法を比較実験した。

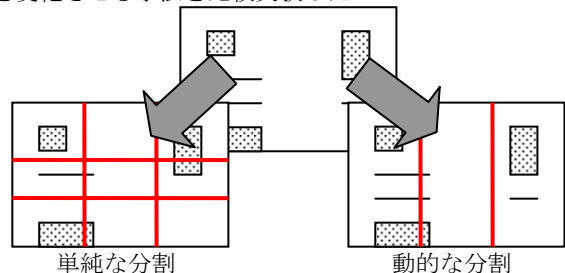


図2 : 二つの分割手法

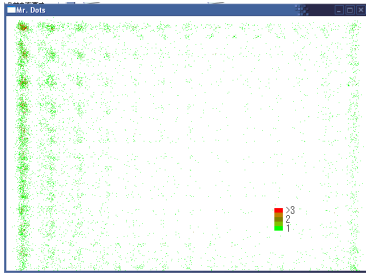
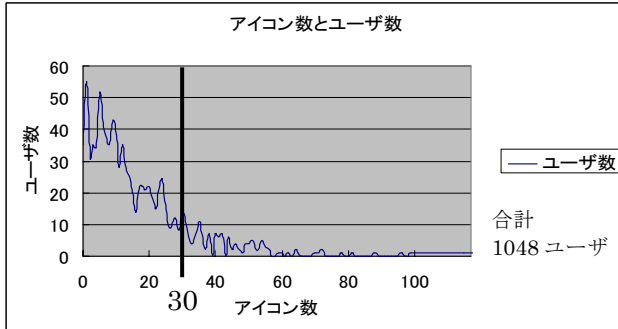


図3：アイコン位置調査のプロット図

表1：一画面に配置するアイコン数のグラフ



選択対象をアイコンとし、アイコンの配置間隔は Windows の標準設定に則った。従って 640x480 の想定解像度では 117 個の配置可能位置からランダムに必要なアイコンを配置することになる。アイコンの配置位置の決定にあたってはアイコンが一樣に分布していると仮定した場合の一樣乱数と、実際のアイコン配置をベースとした乱数（偏りのある乱数）の 2 通りを試す必要がある。偏りのある乱数をつくるため、1048 枚のデスクトップ画像を集めアイコン位置を調査した。調査した合計 19038 個のアイコン座標を元データとして、これから座標を選ぶことで偏りのある乱数が利用できる。各デスクトップ画像に対するアイコンの配置数の関係を調査してみると、図3のようなアイコンの集中が見られた。また、表1のように約 78% のユーザがデスクトップ上に 30 個以内のアイコンを配置している。この結果をもとに、配置アイコン数を 10 個、20 個、30 個、40 個、50 個と増やし、それぞれ一樣乱数で配置した場合、偏りのある乱数による配置の 2 種類について、単純に 9 分割する場合と、動的分割アルゴリズムを用いた分割の場合の 2 種類をそれぞれ実験する。ここでは単純 9 分割を手法 1 とし、動的分割アルゴリズムを手法 2 として実験を行った。

## 7. 実験結果と考察

表 2～4 に示すとおり手法 1 では 9 分割のみしか行わないので、全ての場合で 9 分割しか選ばれていない。それに対して手法 2 では、一樣、偏りのあるランダム配置はともに表 2 のアイコン数 10 個の場合、一樣乱数なら 8 分割以下が約 86% の割合で採用されることがわかる。偏り乱数であるなら約 89% が 8 分割以下である。表 3 のアイコンが 30 個になると約 63% 程度の割合で 8 分割以下が採用されている。表 4 のアイコン配置数を 50 個に増やした場合であっても、約 60%～51% の割合で 8 分割以下が選ばれている。このように、選択対象となるアイコンが多くなっても、9 分割以外で分割できる場合が約 50% 程度あることが実験結果からわかる。

表 2：配置アイコン数 10 個（10000 回試行）

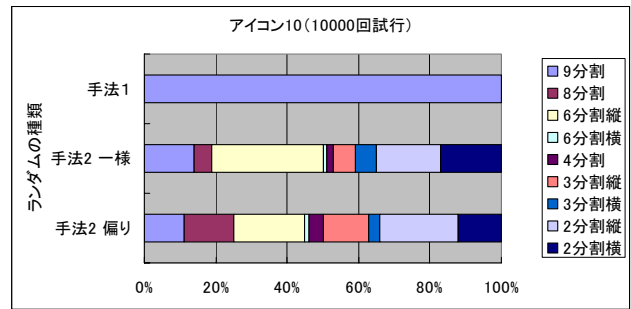


表 3：配置アイコン数 30 個（10000 回試行）

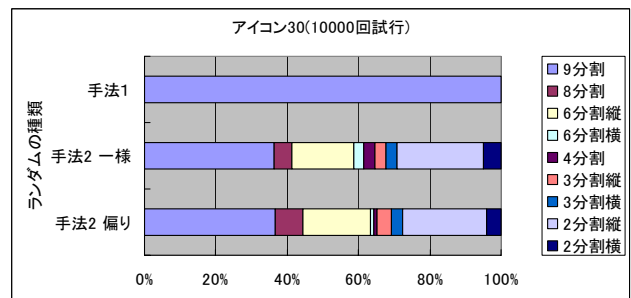
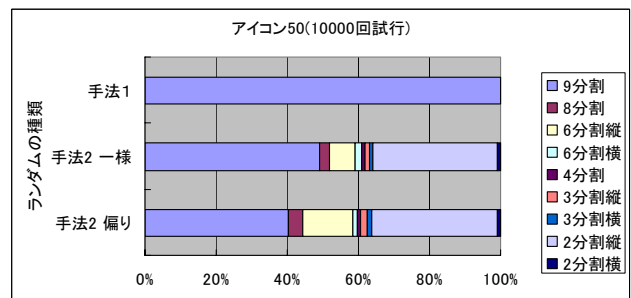


表 4：配置アイコン数 50 個（10000 回試行）



従って、本稿で提案する動的分割アルゴリズムの有効性が明らかになった。

## 8. まとめ

比較実験の結果と、視線制御インターフェースは選択肢の数が少ないほど視線の認識率が高くなる事を鑑みると、視線インターフェースにおいて分割数が減る事は、ターゲットの選択精度、および選択速度の向上に極めて有効である。また、ターゲットが含まれるグリッドを直視するインターフェースのため直感的かつ自然に利用できる。現在のグリッドは長方形の均等な分割にとどまるが、グリッドの形状を可変にすることでさらに最適な分割が可能になる。今後は、グリッドの形状を変化させる分割方法について検討を行う。

## 参考文献

- [1] 阿部, 大山, 大井: "自然光下における画像解析を用いた多指標視線入力システム", 映像メディア学会誌 Vol.58, No.11, pp.1656-1664 (2004)
- [2] 松金, 緑川, 湯浅, 大山: "多指標視線入力を用いた Web 閲覧システム", 情処学, 第 67 回全国大会, 4, pp.147-148(2005)
- [3] 叶世, 阿部, 大山, 大井: "視線入力制御によるパソコンテレビ鑑賞システム", 信学技報, 35, 59, pp.37-42 (2005)
- [4] Liwei, D., Rich, G., Andrew, S., and Jeremy, L.: "Speech-Based Cursor Control :A Study of Grid-Based Solutions", ACM, 77-78, pp.94-101 (2004)