

# 肢体不自由者向け視線入力システムの開発

船瀬和記<sup>†</sup> 西岡芳隆<sup>†</sup> 武藤寛<sup>††</sup>

近年、眼球運動機能以外が維持されていない肢体不自由者向けに、視線の挙動によりコミュニケーションを実現できる視線入力システムの開発が行われている。既存のシステムでは赤外線 eyeball 照射や高価な装置を用いることが、ユーザへの経済・肉体的負担となっていた。著者らは当負担を軽減すべく、自然光下においてパーソナルコンピュータ(PC)と家庭用デジタルビデオカメラ1台のみを用いて、カーソル移動・文字入力等のPC操作が行えるシステムを構築した。

## Eye-gaze input system for the physically handicapped people

Kazuki Funase<sup>†</sup>, Yoshitaka Nishioka<sup>†</sup> and Hiroshi Mutoh<sup>††</sup>

The PC operation system by the eye-gaze input was developed as a communication supporting device for the seriously physically handicapped people. Infrared-ray-emitting systems and the expensiveness of the existing devices has been burden for users. We constructed an inexpensive and non-contact system that is composed only of the DV camera and PC. Users can enter characters and move the cursor by using this system with no stress.

### 1. はじめに

近年、重度肢体不自由者向けの支援装置として、視線の挙動によりコミュニケーションを実現できる視線入力システムの研究が行われている[1]-[4]。視線入力システムは、眼球や顔の動きのみを入力信号として使用するので、重度 ALS (筋萎縮性側索硬化症) 患者のように眼球の運動機能以外が維持されていない人たちの意思疎通を容易にし、ストレスを軽減することが期待される。

これまでに研究開発されてきた視線入力システムの多くは、高価な装置をユーザの

<sup>†</sup> 早稲田大学大学院 基幹理工学研究所  
Graduate School of Fundamental Science and Engineering, Waseda University

<sup>††</sup> 早稲田大学 基幹理工学部  
School of Fundamental Science and Engineering, Waseda University

頭部に取り付けたり、赤外線 eyeball に照射するため、ユーザへの経済的・肉体的負担が大きく、実際にはほとんど普及に至っていない。そこで著者らは、自然光の照明のもとに、パーソナルコンピュータ(PC)と家庭用デジタルビデオ(DV)カメラ1台のみを用いて、これらの負担を除去する視線入力システムを構築している。操作性・利便性の向上を研究目的に、様々なユーザが使用できる文字入力・カーソル移動システムを構築する。

### 2. システム概要

システムを構成するハードウェアは、市販の DV カメラと PC, ディスプレイの3要素のみであり、PC で画像処理を行い、眼の開閉状態や視線方向等を検出し、カーソルの移動やクリックなどの、PC 操作を視線入力で可能にする(図1)。なお図中の数値は目安となる値であり、必ずしもこの値通りに配置する必要はない。

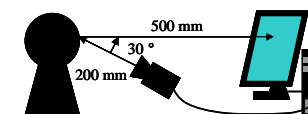


図1 システム全体図  
Figure 1 The whole of system

### 3. 瞳孔中心検出アルゴリズム

本研究では眼の外形的な要素のうち、瞳孔中心位置・眼の輪郭を検出することで、視線方向を判定する。DV カメラから取得する画像は高さ 520[pixel]、幅 728[pixel]の BMP 画像である。以下、眼の輪郭領域・瞳孔中心位置の検出方法について述べる。

#### 3.1 画像の4値化・眼の大まかな範囲の取得

まず、キャプチャされた画像(図2)における、各ピクセルの輝度や色相の値を取得し、その値を基に4値化処理を行い、肌色・黒色・白色と認識されたピクセルに分類する(図3)。次にこの処理結果をもとに、眼の大まかな範囲を取得する。輝度や色相、周囲のピクセルとの相関から、瞳孔位置の可能性のあるピクセルにポイントを加



図2 取得画像  
Figure 2 acquired image

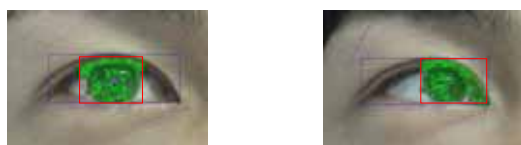


図3 4値化処理とヒストグラム  
Figure 3 Four value classification and histogram

算させる．図3におけるヒストグラムは水平方向の列ごとに加算したポイントを分布させたものである．そして，最もポイントの高い列を瞳孔の中心である可能性が高い列（図3，水平破線）として記憶する．その列周辺からさらに上下左右へ走査線をかけ，色情報から眼の大まかな範囲を取得する．図3の実線枠が取得された範囲であり，以降処理をこの範囲のみに絞る．

### 3.2 瞳孔の抽出

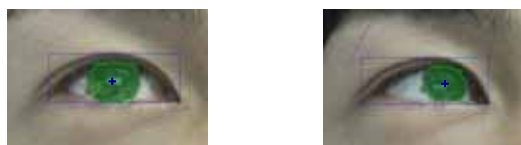
上下左右へ走査線をかけ，黒点および輝度の変動が大きなピクセルが一定以上続く部位を眼の端と認識し，眼の輪郭に外接する四角形を検出する．さらに瞳孔の大まかな範囲，及びその概形を取得する(図4)．図4 サンプル画像2では，上眼瞼の影により白眼の一部が誤検出されている．この部位を除去するために，眼の上部から走査線をかけ，瞳孔と認識しているピクセルのうち上部領域の輝度変動が激しいピクセルを除去する．さらに抽出されているピクセルの平均輝度を閾値に設定し2値化を行い，再度瞳孔の抽出を行う(図5)．図5では前述したノイズをほぼ完全に除去できているのが確認できる．なお，十字線で示した座標は，抽出した瞳孔の重心の座標である．



サンプル画像1                      サンプル画像2

図4 色情報による瞳孔の抽出

Figure .4 Detection of pupil by color information



サンプル画像1                      サンプル画像2

図5 画像処理結果

Figure .5 Result of image processing

## 4. 視線座標の取得

### 4.1 視線座標検出パラメータ

図6に，瞳孔中心と目頭の水平方向の距離  $G_x$  [pixel]，瞳孔中心と目頭の垂直方向の

距離  $G_y$  [pixel]，および輪郭の横幅  $W$  [pixel]を示す．水平方向における視線検出では，式(4.1)で表される  $I_x$ ，すなわちユーザの眼とカメラの距離の変化による影響を受けないパラメータを得るため  $G_x$  を  $W$  で除算して得られる値を，視線検出パラメータとして利用する．同様に垂直方向では，式(4.2)で表される  $I_y$  を定義する．

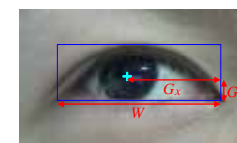


図6 取得値

Figure .6 Acquired value

$$I_x = \frac{G_x}{W} \quad (4.1)$$

$$I_y = \frac{G_y}{W} \quad (4.2)$$

これらのパラメータを用いユーザの視点をディスプレイ上の座標に変換していく．

### 4.2 キャリブレーション

システムではまず，キャリブレーションを行い(図7)，パラメータ  $I_x \cdot I_y$  の値をディスプレイ上の座標に変換するための特性式を定義する．図8の a ~ i 計9点の注視点を設定し，各注視点に対する  $I_x \cdot I_y$  の平均値を記録する．キャリブレーション時に取得した a ~ i における  $I_x \cdot I_y$ ，そして，ディスプレイ左上を(0,0)としたときの各ポイントにおける座標  $(x, y)$  (図7)を基に，最小二乗法より式(4.3)  $A, B, C, D$  の値を算出する．

$$\begin{aligned} I_x &= A \cdot x + B \\ I_y &= C \cdot y + D \end{aligned} \quad (4.3)$$

算出した  $A, B, C, D$  の値から式(5.2)の特性式を完成させる．キャリブレーション終了後，

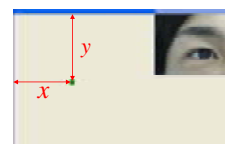


図7 キャリブレーションの様子

Figure .7 calibration

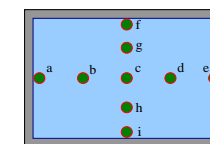


図8 キャリブレーション時の注視点

Figure .8 point of gaze in calibration

取得画像から  $I_x \cdot I_y$  を検出し、(4.4)式に代入することで、求めたいディスプレイ上の座標  $(D_x, D_y)$  (図9)へ変換する。

$$D_x = \frac{I_x - B}{A}$$

$$D_y = \frac{I_y - D}{C} \quad (4.4)$$

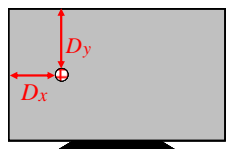


図9 変換後の座標  $(D_x, D_y)$

Figure 9 coordinate after conversion

1	2	3	4	5
6	7	8	9	10
11	12	13	14	15
16	17	18	19	20
21	22	23	24	25

図10 ユーザ試験における注視点

Figure 10 point of gaze in test

#### 4.3 ユーザ試験

(4.4)の変換式によってシステムが導出した座標と、ユーザが実際に注視している座標にどれほどの誤差があるのかを確かめるべく、15人の被験者による試験を行った。システムを起動しキャリブレーションを行った後、ユーザにディスプレイ上の25点(図10)を順に注視させ、その際にシステムが導出した平均座標  $(D_x, D_y)$  を計測した。結果を図11に示す。また実際の注視点の座標  $(x, y)$  と  $(D_x, D_y)$  との水平方向の誤差値  $E_x$  及び垂直方向の誤差値  $E_y$  を式(4.5)で定義する。

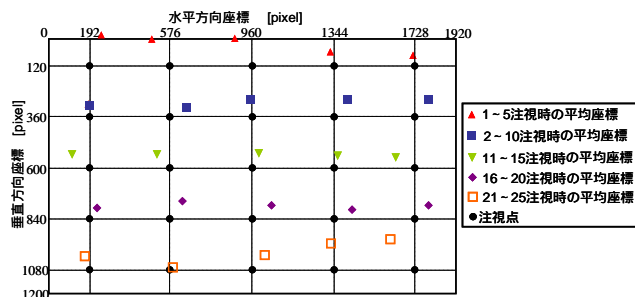


図11 各ポイントの平均座標分布

Figure 11 distribution chart of average coordinate

$$E_x = |x - D_x|$$

$$E_y = |y - D_y| \quad (4.5)$$

15点全ての誤差値の平均値  $\overline{E_x}$ 、 $\overline{E_y}$  はそれぞれ 160[pixel]、158[pixel]となった。図

11からは、左上・右下などの斜め方向視時に特に誤差が大きいことが分かるが、原因は、キャリブレーションでパラメータを取得していない方向を注視している際に、 $I_x \cdot I_y$  の値が視線方向と相関性を持たないこと等が考えられる。今後は斜め方向にも対応できるような新たなパラメータを検討する必要があるが、現段階では上述の誤差を考慮し、操作性の良いカーソル移動・文字入力システムを提案する。

#### 5. システムの操作方法

4章で述べたアルゴリズムより検出した座標を、カーソル移動・文字入力等のシステム操作に利用する。以下に、システムの流れ・操作方法について述べる。

##### 5.1 カーソル移動

システム起動直後は4.2章で述べたキャリブレーションを行い、キャリブレーション終了と同時にマウスカーソルを操作できる状態(マウスモード)へ移行する。

カーソル操作方法において理想となるのは、ユーザの視点を正確に座標に変換し、瞬時にカーソルを座標に移動させるという直接移動式のカーソル移動システムである。しかし、4章で述べたように本システムでは、ユーザの注視点とシステムの検出座標に誤差があり、現段階の検出精度では適当でない。そこで本視線入力システムでは、ユーザの視点到合わせてターゲットが動き、瞬きによってカーソルをターゲットの位置まで徐々に移動させることが出来るシステムを構築した(図12)。この際、再度瞬きを行うことでカーソルを静止させることも可能である。よって目標とする座標とターゲットの座標が正確に一致していなくても、ターゲットとカーソルを結ぶ線分上に目標とする座標があれば、カーソルを任意の位置座標まで移動できる。

また、カーソルがターゲットに移動する速度については、カーソルの位置座標を  $(x, y)$ 、ターゲットの位置座標を  $(D_x, D_y)$  とし、1処理後のカーソル位置  $(x_1, y_1)$  を、

$$x_1 = x + \frac{(D_x - x)}{10}$$

$$y_1 = y + \frac{(D_y - y)}{10} \quad (5.1)$$

と設定することで、カーソルがターゲットに近づくにつれ速度は遅くなり、ユーザは目標とする座標でカーソルを容易に静止できる。

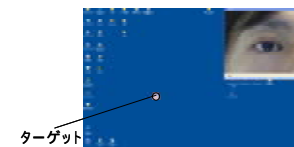


図12 ターゲットを用いたカーソル移動システム

Figure 12 cursor movement system using target input

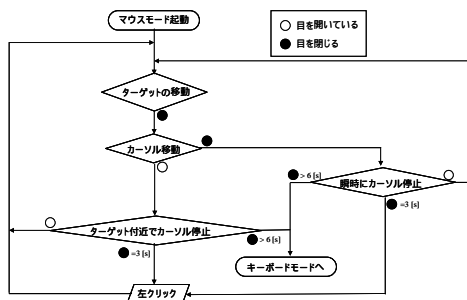


図 13 フローチャート(カーソル移動)

Figure 13 flowchart (cursor movement)

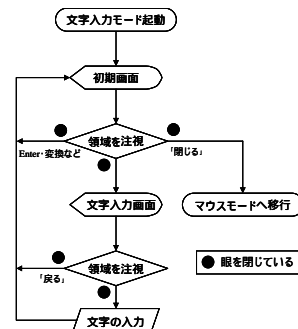


図 14 フローチャート(文字入力)

Figure 14 flowchart (character input)

カーソル操作の流れを簡易的に示したフローチャートを図 13 に示す。

## 5.2 文字入力

文字入力に関しては、ディスプレイ上に分割された入力領域を設け、「ユーザの注視した場所」を直接入力するシステムを構築した。図 15 のようにディスプレイ上に半透明化された入力領域を水平 6×垂直 3 の計 18 領域設け、18 の入力信号と閉眼信号の計 19 の信号で文字入力を可能にする。入力領域を水平 6×垂直 3 に分割した理由だが、本システムで使用しているディスプレイは水平 1920[pixel]、垂直 1200[pixel]であり、分割数  $n$  と水平・垂直方向それぞれに必要な精度  $X_n$  [pixel]、 $Y_n$  [pixel] は表 1 のようになるので、4.3 章で述べた平均誤差値と比較し、最適な分割数を採用した。

表 1 分割数と必要な精度

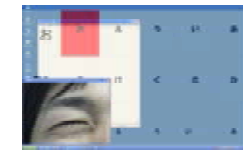
Table1 number of partitions and indispensability accuracy

分割数 $n$	$X_n$ [pixel]	$Y_n$ [pixel]
2	480	300
3	320	200
4	240	150
5	192	120
6	160	100
7	137	86

マウスモード時に 6[s]眼を閉じるにより文字入力モードに移行する。文字入力モードに移行後 18 個の領域に五十音の行や、英数字・記号などが各々半透明で表示される(図 15(a))。ユーザが注視していると認識された領域は青く表示され、瞬きを行うことで領域が赤くなり、文字が入力される(図 15(b))。文字入力システムのフローチャートを図 14 に示す。



(a) 入力文字選択



(b) 文字の入力

図 15 文字入力の様子  
Figure 15 character input

## 6. まとめ

本研究では、肢体不自由者向けの意思疎通支援装置として、視線入力による PC 操作システムを開発した。当システムは自然光下において DV カメラと PC のみを用いてユーザの視線を検出するので、ユーザへの負担が少なく利便性は大きく向上した。

瞳孔重心座標を取得するプログラムでは、睫毛や上眼瞼の影などによるノイズを除去するアルゴリズムを作成し、正確な瞳孔中心検出が可能になった。また、検出した瞳孔中心座標から  $I_x \cdot I_y$  なるパラメータを得て、ディスプレイ上の座標に変換するアルゴリズムを構築した。座標への変換精度に関して、15 人の被験者による試験を行ったが、垂直方向に平均 160[pixel]、水平方向に平均 158[pixel]の誤差が生じる結果となった。試験結果を踏まえ、ターゲットを用いた入力方法によるカーソル移動システム・ディスプレイ上に計 18 区分の入力領域を設けた文字入力システムを搭載した。

これらのカーソル移動・文字入力システムにより操作性は向上し、高価な装置や赤外線を用いた視線入力システムに劣らない意思疎通支援装置が完成した。視点を直接ディスプレイ上の座標に反映させるシステムは、ユーザにとって可視的にカーソル移動や文字入力が把握でき、利便性の高いインターフェースである。今後も本システムを改良し、新たな瞳孔中心検出パラメータを検討することで座標への変換誤差を無くしていく。システム全体としての操作性をより向上させることで、肢体不自由者に有用なシステムの完成を目指したい。

## 参考文献

- 1) 伊藤, 伊福部: ビデオキャプチャ画像処理による視線検出及び意思伝達装置への応用; 電子情報通信学会論文誌, Vol. J88-D-I, No.2, pp.527-535
- 2) 佐藤, 阿部, 他: 視線入力による重度肢体不自由者向けコンピュータ操作支援装置システム; 電子情報通信学会技術研究報告, Vol. 105, No.507, pp.15-20 (2006).
- 3) 西内, 高田: 虹彩輪郭の楕円近似を用いた非接触視線検出法の研究; 社団法人日本機械学会論文集, Vol. 69, No.682, pp.1611-1617 (2003).
- 4) 竹上, 後藤, 大山: 視線方向計測のための高精度瞳孔検出アルゴリズム; 電子情報通信学会論文誌, Vol. J86-D- , No.2, pp.252-261 (2003).