

K-075

## 頭部の自由な動きを許容する視線測定システム A Gaze Measurement System Permitting a Head Motion

澤島 康仁†  
Yasuhiro Sawahata

小峯 一晃†  
Kazuteru Komine

浦谷 則好†  
Noriyoshi Uratani

### 1. まえがき

視線と音声を用いて操作する、テレビインタフェースを開発している [1]。視線情報を利用することにより、「これ」「それ」などの指示語を発話した際、その言葉が何を指しているのかを解決することが可能になる。すなわち、指示語のあいまいさを視線情報で補完することにより、より円滑なテレビ操作の実現が見込める。

本稿では特に、試作したテレビインタフェースの構成要素のうち、頭部の動きを許容しつつ視線測定を行う部分について述べる。テレビ視聴時の視線測定に求められるのは、測定装置を身につけないこと、測定時でもユーザは自由に動けることである。そこで、従来の視線測定手法と、顔の特徴点抽出に基づく眼球追跡手法とを組み合わせ、ユーザの頭部の自由な動きを許容する視線測定システムを試作した。また、視線を用いて画面上のオブジェクトを選択する方法も合わせて実装した。本発表では、これらについての詳細とその評価について述べる。

### 2. 視線測定システム

#### 2.1 概要

テレビ視聴のための視線測定は、ユーザは測定装置を装着せず、かつ自由な動きができることが望ましい。本システムでは、2台のカメラを用いて視線測定を行う。第1のカメラは、眼球のアップを撮影する。第2のカメラは、ユーザの周辺領域を撮影する。第2のカメラの情報に基づき、第1のカメラを制御することで、ユーザの頭部の自由な動きを許容する視線測定環境を実現した(図1)。

また、測定された視線データ(座標値)は、生理現象によるデータのばらつきや、測定時のノイズが含まれており、視線測定装置の出力をそのままポイントとして利用するのは適切ではない。本システムでは、注目オブジェクトの検出アルゴリズムを実装することで、「これ」が何を指しているかを解決している。

以後、これらについての詳細を述べる。

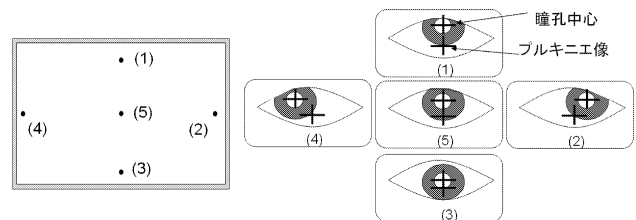
#### 2.2 視線測定の原理

非拘束性を実現した視線測定手法として、眼球に近赤外光を当て、角膜表面からの反射光(プルキニエ像)と瞳孔中心位置を画像処理により求め、それらの相対位置の変化により視線方向を得る手法が知られている。視線位置が変わると、瞳孔中心とプルキニエ像の位置は図2のように変化する。詳しい方法はここでは省略するが、これらの位置関係の変化を利用することで、視線位置を測定することができる。しかしながら、この測定原理では、キャリブレーション位置と実際の測定位置が異なる場合は、系統的な誤差が生ずる。この問題に対しては、大野らの方法 [2] を利用することで対応が可能である。

†NHK 放送技術研究所, NHK Science and Technical Research Laboratories



図 1: 視線測定と眼球追跡



(a) 画面上の点 (b) 画面上の点を見た際の眼球の様子

図 2: 瞳孔中心とプルキニエ像の関係

視線方向を精度良く測定するためには、眼球の画像を高解像度で得る必要がある。解像度を得るためにズームレンズを搭載したカメラを利用すれば良いが、ユーザが動くとその視野角から外れてしまい、眼球の画像を得ることができなくなる。眼球画像を常に捉えるために、カメラのパン・チルトを行う必要がある。本システムでは、ユーザの動きに対応するために、回転軸が直交する2つのミラーを第1のカメラの前に設置し、それらの角度を制御することで、カメラをパン・チルトさせることと同様の効果を得ている。第1のカメラで取得した眼球画像がその視野角から外れようとする際に、ミラーを適切に制御することで、ユーザが動いても常に眼球の画像を得ることができる(第1のカメラ単体による眼球追跡)。

#### 2.3 顔の特徴点を用いた眼球追跡

最初に眼球の位置を見つける際や、ミラーの制御が追いつかない程の早い動きがある場合は、ユーザの周辺領域を捉える第2のカメラにより目の位置を検出し、眼球の追跡を開始/再開することができる(第2のカメラを併用した眼球追跡)。

顔の特徴点から適切なミラー位置を計算する際に、事前のキャリブレーションが必要である。図3(b)のような、3点が描かれた板を第2のカメラで撮影し、それぞれの点を第1のカメラが捉えられるようにミラーを調節する。3点の座標とミラーの位置の組の関係が、線形であると仮定することで、第1のカメラが第2のカメラ内の任意の点を捉えるミラー位置を計算することができる。

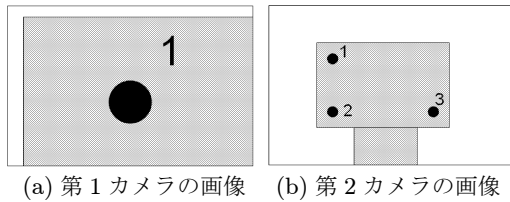


図 3: キャリブレーション

キャリブレーションの手順は次のとおりである。ミラーを動かす、図 3(a) のように、点 1 が画像の中心に来るようにミラーを動かす。このときのミラーの角度を、 $(m_{1h}, m_{1v})$  とし、図 3(b) 内の点 1 の座標値を  $(x_1, y_1)$  とする。同様の作業から、点 2, 点 3 から、 $(m_{2h}, m_{2v}), (m_{3h}, m_{3v}), (x_2, y_2), (x_3, y_3)$  が得られる。これらの点を  $x$  方向、 $y$  方向について線形補間することにより、第 2 のカメラで捉えた画像内の任意の点  $(x, y)$  を第 1 のカメラで捉えるためのミラーの角度  $(m_h, m_v)$  は、次式で与えられる。

$$m_h = \frac{m_{3h}x_2 - m_{2h}x_3}{x_2 - x_3} + \frac{m_{2h} - m_{3h}}{x_2 - x_3}x \quad (1)$$

$$m_v = \frac{m_{1v}y_2 - m_{2v}y_1}{y_2 - y_1} + \frac{m_{2v} - m_{1v}}{y_2 - y_1}y \quad (2)$$

但し、ユーザがカメラからの距離方向に移動するような状況では、式 (1)(2) は不適切になる。

ところで、ミラーが動いている時は、撮影している眼球画像が乱れるため、視線の測定ができない。ユーザの動きに対応しつつ、ミラーを動かす頻度を最小限に抑える必要がある。

第 1 のカメラ単体による追跡は、眼球画像を捉えているか否かに基づきミラーを動かすため、ミラーを動かす頻度は最小であるが、ユーザの早い動きには対応できない。一方で、第 2 のカメラを併用した場合の追跡は、ユーザの早い動きには対応可能だが、眼球画像を捉えているかどうかを考慮していない。これらを鑑み、できるかぎり第 1 のカメラ単体による追跡方法を利用し、その追跡方法では対応しきれない動きがあるときのみ、第 2 のカメラを併用する方法を採用する。

具体的には、次のようなタイミングで切り替える。まず、単位時間あたりの視線データの出力数 (取得レート) をチェックする。視線の測定が適切に行われている場合は、取得レートは約 30[sample/sec] 取れるが、そうでないときは、取得レートが著しく低下するという特徴がある。本システムでは、しきい値を 10 とし、取得レートがこの値を下回ったとき、眼球追跡の方法を、第 2 のカメラを併用する追跡方法に切り替えている。これにより、ユーザの早い動きに対応しつつ、最小頻度のミラー操作で視線測定が可能になる。

## 2.4 注目画面の決定方法

視線測定装置により得た測定値 (座標値) は、測定時におこるノイズや、人間の生理現象に基づく不随意的な運動により、ユーザが一点を凝視しているつもりでも、ばらつきをもって測定される。そのため、測定値を直接ポインタとして利用すると、システムの使い勝手を低下させてしまう。そこで、測定データのスムージングと、注目画面を決定する領域のサイズを動的に変更する処理により、使い勝手の向上を図った。

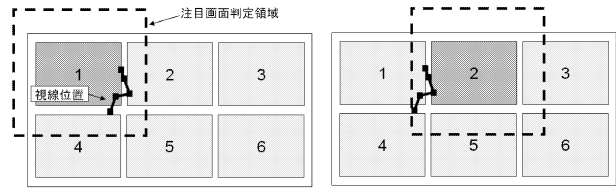


図 4: 注目画面判定領域

(a) 子画面 1 を注目時 (b) 子画面 2 を注目時

測定された  $x, y$  の座標値をバッファし、ソートを行い、バッファ内の中央値 (メジアン) を現在の視線の位置とすることで、スムージングをしている。バッファの長さは、経験的に 10 としている。新しい測定値が到着したときは、バッファ内でもっとも古い測定値が破棄され、新測定値はバッファ内の適切な位置に挿入される。これにより、ノイズにより視線位置が大きく外れる現象が防げる。

それでもなお、主に不随意的な運動による測定値のばらつきは残る。これに対応するため、注目画面判定領域を図 4 のようにとった。図中で破線で示された注視画面判定領域を脱したときに、注視画面が変わったと判定する。これにより、不随意的な運動による視線の細かい変動により、注視画面が頻繁に切り替わるという現象を除くことができる。

## 3. 実装と評価

視線測定装置として、竹井機器工業 (株) の FreeView DTS を利用した。30[sample/sec] の視線データの出力ができる。この装置には、赤外線光源とカメラ (第 1 のカメラ) およびミラーが搭載されている。第 2 のカメラとしては、Canon VC-C4 (雲台一体型カメラ) を利用した。顔の特徴点を抽出するソフトウェアは、N-Vision (株) の SDK を利用した。顔の特徴点は、30[fps] で追跡が可能である。また、これらの機器を制御するソフトウェアは、Visual C++により実装した。

ユーザが動き、眼球追跡が外れたときから、再度眼球の追跡を復帰するまでにかかる時間は、約 1[sec] であった。また、ユーザの報告によると、注視している子画面を決定するアルゴリズムは、おおむね好評を得ている。これについては今後、客観的な評価を得るための実験を行う予定である。

## 4. おわりに

視線と音声で操作するテレビインタフェースのための視線測定システムについて述べた。眼球を撮影するカメラと、ユーザの様子を撮影するカメラの 2 つのカメラを利用することで、ユーザの頭部の動きを許容する視線測定が可能になった。

今後は、視線測定の原理も含め、システムに関して再考し、テレビ視聴環境に適した視線測定システムについて検討する予定である。

## 参考文献

- [1] 小峯他, “視線情報を利用したテレビ用ユーザインタフェースの開発”, FIT2004, 2004
- [2] 大野他, “2点補正による簡易キャリブレーションを実現した視線測定システム”, 情報処理学会論文誌, Vol.44, No.4, pp.1136-1149, 2003