

(19) 日本国特許庁(JP)

(12) 公開特許公報(A)

(11) 特許出願公開番号

特開2006-167256
(P2006-167256A)

(43) 公開日 平成18年6月29日(2006.6.29)

(51) Int. Cl.	F I	テーマコード (参考)
A 6 1 B 3/113 (2006.01)	A 6 1 B 3/10 B	5 B 0 8 7
G 0 6 F 3/033 (2006.01)	G 0 6 F 3/033 3 1 0 A	

審査請求 未請求 請求項の数 6 O L (全 18 頁)

(21) 出願番号 特願2004-365546 (P2004-365546)
(22) 出願日 平成16年12月17日 (2004.12.17)

(71) 出願人 304023318
国立大学法人静岡大学
静岡県静岡市駿河区大谷836
(74) 代理人 100122219
弁理士 梅村 勁樹
(72) 発明者 海老澤 嘉伸
静岡県浜松市大平台3丁目16番10号
Fターム(参考) 5B087 AA09 AC09 BC05 BC12 BC13
BC26 BC32

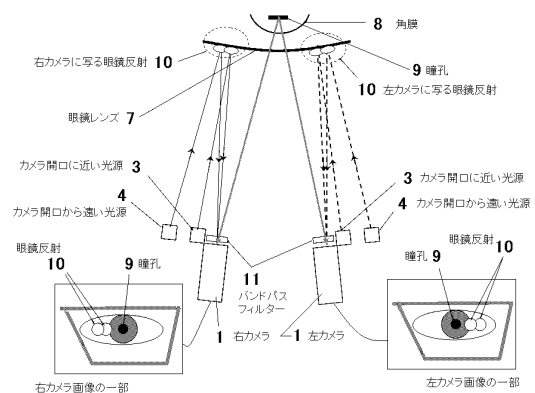
(54) 【発明の名称】 瞳孔検出装置

(57) 【要約】

【課題】入力装置としてのマウスに代替できる瞳孔位置検出によるポインティング装置において、眼鏡をかけている被験者に対して、光源の眼鏡反射に邪魔されことなく瞳孔を検出する。

【解決手段】左カメラの近傍に取り付ける遠近1対の光源と、右カメラの近傍に取り付ける遠近1対の光源とで、波長の異なる光源を使用する。例えば、一方には中心波長が810nmの光源を使用し、もう一方は850nmの光源を使用する。そして、各カメラの開口部には、それぞれの光源と同じ中心波長を持つ、十分に狭帯域のバンドパスフィルターを取り付け、互いの画像に干渉しあわないようにする。各光源の波長域がオーバーラップする場合には、光源の前にもバンドパスフィルターを取り付け、光源の波長域を狭くしてから、頭部に近赤外線を照射する。これにより、左右の光学系は独立し、互いに干渉しなくなる。

【選択図】 図8



【特許請求の範囲】**【請求項 1】**

1 台のビデオカメラ(1)と、該ビデオカメラの開口部に近い第1の光源(3)と、該開口部から遠い第2の光源(4)と、前記第1及び第2の光源をビデオカメラのフィールドまたはフレームに同期させて交互に点灯させる光源制御手段と、各点灯時期に得られる画像の差分から瞳孔を検出する検出手段とからなる撮像システムをN(但しNは $N > 1$ の整数)セット設け、これらのセット間において光源の波長を異ならせるとともに、それぞれのビデオカメラに各光源の波長に対応したバンドパスフィルタを設けてなる瞳孔検出装置。

【請求項 2】

1 台のビデオカメラ(1)と、該ビデオカメラの開口部に近い第1の光源(3)と、該開口部から遠い第2の光源(4)と、前記第1及び第2の光源をビデオカメラのフィールドまたはフレームに同期させて交互に点灯させる光源制御手段と、各点灯時期に得られる画像の差分から瞳孔を検出する検出手段とからなる撮像システムをN(但しNは $N > 1$ の整数)セット設け、これらのセット間において光源の発光時期を異ならせるとともに、それぞれのビデオカメラに各光源の発光時期に対応したシャッタ機能を設けてなる瞳孔検出装置。

10

【請求項 3】

眼鏡やサングラス等を装着している被験者の撮影画像において、どちらか一方のカメラの画像において光源の眼鏡反射のために瞳孔が検出できない場合であって、他方のカメラの画像において瞳孔が検出できるときに、その画像を利用して、瞳孔検出を行う検出制御手段を備えてなる請求項1または請求項2記載の瞳孔検出装置。

20

【請求項 4】

前記ビデオカメラは被験者の両瞳孔を撮影し得る画角とし、両カメラ画像からの左右の瞳孔の有無の情報が一致しないときは、検出ができていないほうのカメラ画像の瞳孔の移動に合わせて、瞳孔が検出されるようになるまで、瞳孔位置を予測しながら瞳孔抽出用のウィンドウを与え続け、両方の画像において、左右のうち同一の瞳孔が検出できなくなったとき初めて、それに相当する側の目が閉じたと判断する判断手段を備えてなる請求項3記載の瞳孔検出装置。

【請求項 5】

前記検出制御手段は、左右の目の開閉をポインティングデバイスの左右の押しボタンスイッチの押下状態情報に対応させる機能を有する請求項3記載の瞳孔検出装置。

30

【請求項 6】

前記検出制御手段は、得られた瞳孔位置からディスプレイ上のカーソル位置を制御する信号を作成する機能を有する請求項3記載の瞳孔検出装置。

【発明の詳細な説明】**【技術分野】****【0001】**

本発明は、入力装置としてのマウスに代替できる瞳孔検出装置及び該装置を用いるポインティング装置に関する。特に、眼鏡をかけている被験者に対して、光源の眼鏡反射に邪魔されることなく瞳孔を検出するための装置に関するものである。

【背景技術】

40

【0002】

これまでに視点(ユーザの視線方向)によるポインティングが種々提案されている。

しかし、視点によって、ポインティングする場合は、表示装置上のポインティングしたい場所へ視点を動かさなければならないが、見るべき位置を探索している間も、カーソル(ポインタ)が視点の位置に常に現れるため、カーソルを移動させたい位置にカーソルを移動させにくいという問題がある。また、視点位置にいつもカーソルが提示されることにより、それが視覚刺激となり、見たい対象が見にくいなどの問題がある。

頭部の運動によれば、見ているところとカーソルの位置は異なり、しかも、前方の表示装置上の必要なところに視線を向けながらも、周辺視でカーソルを見ながら目的の位置にカーソルを動かすことができる。

50

従来の頭部の動きによるポインティングシステムの例を次に示す。

非特許文献1に記載された技術(Eagle eyeと名づけられている)は、パソコンの横にビデオカメラを設置し、顔の画像を撮影しながら、画像処理により、予め指定する顔の一部をテンプレートマッチングにより追尾し、その動きに応じて、パソコン画面上のカーソルを動かす方式である。追尾対象としては、鼻の先、あごの先、黒眼などをためしている。しかし、テンプレートを更新しても、急減な周囲の明るさの変化などに対応しにくい可能性があり、また、追尾すべき対象が徐々にずれていくなどの問題も発生しやすい。

【0003】

本発明者は、これらの欠点を解消すべく瞳孔位置検出によるポインティング装置を提案している(特願2003-419110:2003年12月17日出願)。この技術は、不可視な赤外光源によって顔を照らし、ビデオカメラによって、顔の横幅全体が楽に入る拡大率で撮影し、ビデオカメラによって撮影される画像内の2つもしくは1つの瞳孔を直接検出する方式である。そして頭の動きによって生じる画像内の瞳孔の動きに合わせて、パソコン等のモニター上のカーソルを動かしたり、モニター上に表示されているメニューの選択をしたり、キーボードソフトを利用して文字を入力したり、複数のウインドウを切り替えたりすることができる。また、装置単独でマウスの左右ボタンの押下に相当する信号を発生することができ、他のクリック操作用入力補助装置の必要がない。

しかしながら、被験者(ユーザ、オペレータ)が眼鏡を使用しているとき、その眼鏡反射により瞳孔検出が妨害される場合がある。

【非特許文献1】"The Camera Mouse: Visual Tracking of Body Features to Provide Computer Access for People With Severe Disabilities", (Margrit Betke, James Gips, and Peter Fleming), IEEE Transactions on Neural Systems and Rehabilitation Engineering, Vol.10, No. 1, March 2002

【発明の開示】

【発明が解決しようとする課題】

【0004】

本発明者は、視野の広い1台のカメラで、フレーム内の2個の瞳孔のそれぞれの位置を検出して、被験者が頭部運動を行うときに画像内を移動する瞳孔の移動に応じて、カーソル(ポインタ)を動かす手法を先の出願で提案した。この方法では、左右の瞳孔を区別して、片目を瞬間的に閉じた場合はマウスのクリック、また、片目を閉じた状態で頭部を動かした場合は、もう片方の瞳孔移動によってカーソルを動かす、それによりドラッグをすることを可能にした。この手法を実現するためには、瞳孔を検出しやすくする必要があった。そのために、カメラ(1)の開口(2)に近い光源(3)と遠い光源(4)をビデオのフィールドに同期させて交互に点灯させ(図2、3)、そのとき得られる明瞳孔画像(図1-a)と暗瞳孔画像(図1-b)を実時間で差分を行い、背景部をおよそ相殺させ、瞳孔部を浮き立たせた後に瞳孔を検出した(図1-c)。光源の配置例としてLED(発光ダイオード;5)を用いたものを図3に示している。この瞳孔検出法は、カメラの開口に近い光源のみを設けて明瞳孔画像を得て瞳孔を画像処理により検出する方法や、カメラの開口から遠くの光源のみを設けて暗瞳孔画像を得て画像処理により瞳孔を検出する方法に比較して、周囲の明るさなどに影響を受けにくく、格段に瞳孔が検出しやすい。さらに、画像が単純なため、画像処理量が大きく軽減できるなどにおいて有利である。

【0005】

さらにこの瞳孔検出法では、被験者が視力矯正用の眼鏡やサングラスをしている場合にも、眼鏡反射の輝度は画像中において飽和するくらいに明るく他の部分と区別しやすいため、眼鏡反射の部分の輝度を下げてから画像差分をすることにより、眼鏡反射を画像処理の段階で除去することができる(図4)。図4(a)は、眼鏡反射が写っている明瞳孔画像、図4(b)は、眼鏡反射が写っている暗瞳孔画像、図4(c)は高輝度部分の輝度を落としてから差分をした画像である。しかし、眼鏡反射像が瞳孔に完全にかぶってしまったときには、眼鏡反射により瞳孔がかき消されてしまうため、画像処理の段階で瞳孔を検出するのは不可能である。このような場合にも、光源の前に偏光フィルターを設置し、カメラの

10

20

30

40

50

開口の前に前述の偏光フィルターとは偏光方向を垂直にした偏光フィルターを設置することによって、眼鏡反射をある程度除去できる。しかし、この際に眼鏡反射を完全に除去することは難しく、一部が残ると眼鏡反射像はかえって瞳孔に類似した画像となり、区別しにくくなる。

【0006】

また、顔に一切のマーカ―などを取り付けず、視線の検出を行う場合は、光源の角膜反射像を参照点として、瞳孔の位置により視線方向を検出することがある。すなわち、カメラ画像中の瞳孔と角膜反射の相対座標から視線方向を推定する。角膜反射は眼鏡反射とサイズは異なるが性質は似ているため、偏光フィルターを用いると、眼鏡反射を除去すると角膜反射まで除去されてしまうため、使用できない。

10

もし、カメラ開口の近くに光源を配置することがなければ、たとえば、図5のようにカメラ(1)の開口から離れた左右の位置に1個ずつ光源(6)を設置しそれら交互に点灯させる方法がある。図5の7は眼鏡レンズ、8は角膜である。この場合、どちらかの光源の一方の眼鏡反射(10)が瞳孔(9)を隠したとしても、もう一方の光源の眼鏡反射が瞳孔を隠さない場合が多いので、眼鏡反射が瞳孔を隠していない方の画像を利用して瞳孔を検出することができる。しかし、前述のように、光源の一方をカメラの開口の近くにどうしても設置しなければならない場合は、この方法は使用できない。

【0007】

いま、図6のように、カメラ(1)を2台使用し、それぞれのカメラの開口付近に光源(6)を用意する。そして、これらの光源をビデオフレームもしくはフィールドに同期させて交互に点灯させる。右の光源が点灯しているときには、右のカメラでは明瞳孔画像(図6-d)が得られ、左のカメラでは暗瞳孔画像(図6-b)が得られる。逆に左の光源が点灯しているときには、右のカメラでは暗瞳孔画像(図6-e)が得られ左のカメラには明瞳孔画像(図6-c)が得られる。各カメラにおいて得られる明瞳孔画像と暗瞳孔画像を差分することによって、一般には前述のように差分画像において瞳孔が簡単に検出できる。しかし、眼鏡を装着している際には、図6のように、眼鏡レンズ(7)がある場合、眼鏡反射(10)が瞳孔(9)にかぶさることがある。その場合には、瞳孔は検出できない。この例では、右光源が点灯しているときには、左右両方のカメラ画像において瞳孔に眼鏡反射が重なっている。このような例では、左のカメラから得られる差分画像でも、右のカメラから得られる差分画像においても、眼鏡反射が瞳孔に重なっているために、両方のカメラにおいて瞳孔が検出できない。

20

30

次に、図7のように、2台のカメラを利用し、それぞれのカメラ(1)に開口に近い光源(3)と遠い光源(4)を取り付け、それらの光源を交互に点灯させることを考える。すると、1台のカメラについて、4個の眼鏡反射(10)が現れることになり、かえって瞳孔(9)が検出しにくくなる。

【発明を実施するための最良の形態】

【0008】

そこで、図8、図9に示すように、図7に示した方法と同様に、同期させて駆動する2台のカメラ(1)のそれぞれの開口部に近い位置と遠い位置に近赤外光源(3, 4)を設置し、それらの光源をビデオのフィールド(フレーム)に同期させて交互に点灯させ、それぞれのカメから得られる明瞳孔画像と暗瞳孔画像から、画像差分を基本とした画像処理により瞳孔を検出する。ここまでは図7に示した方法と同じである。本発明で提案する1つ目の解決法は、左カメラの近傍に取り付ける一対の光源と、右カメラの近傍に取り付ける一対の光源とで、波長の異なる光源を使用することによる。2つ目の解決法は、左の2個の光源がどちらかが点灯している間は、左のカメラで撮影し(左カメラの電子シャッターを開く)、右の2個の光源のどちらかが点灯している間に右のカメラで撮影する(右カメラの電子シャッターを開く)、というように、時分割の方法による。

40

【実施例1】

【0009】

図8、図9に波長の異なる光源を用いる方法を示す。図7の方法と異なり、左カメラの

50

近傍に取り付ける一対の光源と、右カメラの近傍に取り付ける一対の光源とで、波長の異なる光源を使用する。例えば、一方には中心波長が810nmの光源を使用し、もう一方は850nmの光源を使用する。そして、各カメラの開口部には、それぞれの光源と同じ中心波長を持つ、十分に狭帯域のバンドパスフィルター(11)を取り付け、互いの画像に干渉しあわないようにする。各光源の波長域がオーバーラップする場合には、光源の前にもバンドパスフィルターを取り付け、光源の波長域を狭くしてから、頭部に近赤外線を照射する。これにより、左右の光学系は独立し、互いに干渉しなくなる。なお、図8、図9においては、カメラ画像には、開口に近い光源に眼鏡反射像と開口から遠い光による眼鏡反射像が一枚の画像に同時に表現されている。光源の点滅を制御する光源制御手段は図示されていない。

10

このほかに、異なる波長の光源に代えて、時分割駆動による光源制御も考えられる。

左の2個の光源のどちらかが点灯している間は、左のカメラで撮影し(左カメラの電子シャッターを開く)、右の2個の光源のどちらかが点灯している間に右のカメラで撮影する(右カメラの電子シャッターを開く)、というように、時分割の光源制御方法を採用することにより、波長制限とほぼ同様の効果が得られる。

【0010】

これまで述べた方法では、基本的に、眼鏡反射像が瞳孔像に重なってしまった場合は、瞳孔が検出できないことは、これまでの方法とかわらない(図9の右カメラの画像)。ただし、確率的に、両方のカメラの画像において、同時に瞳孔が隠れてしまうことは、稀である。すなわち、瞳孔は、常に、2台のカメラの画像のうち、少なくともどちらか一方から検出できる。これを利用して、瞳孔マウスのロバスト性を高める。ここで、図8は、両方のカメラで同時に瞳孔が検出できる場合を示しており、図9は、被験者から向かって左のカメラでは瞳孔が検出でき、右のカメラでは検出できない例を示している。

20

【0011】

ここまで、2台のカメラによるものを述べたが、カメラを3台以上とすることにより、さらに眼鏡反射と瞳孔とが重なることを避け得ることは、自明である。

また、2台のカメラで捉えた顔画像から三角測量により、2つの瞳孔の3次元位置を検出することができる。さらに2つの顔画像のそれぞれにおいて検出される2つの瞳孔のそれぞれの周りを詳細に解析し、2つの正確な瞳孔中心と2つの角膜反射中心を検出し、2つの眼の視線を求めることができる。2本の視線の交点が視点であり、瞳孔の3次元位置と視線から三次元の視点の計測ができる。これにより求めた視点位置からカーソルの位置を制御する用途において、瞳孔や角膜反射を捉え、その3次元位置を測定する場合には、少なくとも2台のカメラで対象物を撮影することが必要であり、この場合には3台以上のカメラを配置する。

30

他に、被験者が広角度に頭を回転させる、もしくは頭を広範囲に動かす場合においても、カメラを多数台配置することが有効である。

【0012】

瞳孔の移動を検知し移動量情報を作成するに当り、(1)瞳孔の位置をカーソル位置とする(絶対量による制御と名づける)、(2)瞳孔の単位時間あたりの移動量を、カーソルの現在位置からの単位時間あたりの移動量とする(相対量による制御と名づける)、(3)瞳孔の原点からのずれに比例して、カーソルの単位時間あたりの移動量を生成する(積分量による制御と名づける)、の3つが考えられる。上記(1)は、旧来のジョイスティックに対応し、レバーの変位量に見合った量で、カーソル位置が決定される(絶対位置の制御)。上記(2)は現在のマウスやトラックボールに対応し、回転量に見合った量で、カーソルが移動する(相対位置の制御)。上記(3)は、現在のノートパソコンに見られるスティック状のコントローラに対応し、レバーの傾きが存在する間、カーソルは移動しつつける(積分による位置決定)。カーソルキーによるカーソルコントロールも、上記(3)に相当すると言える。

40

【0013】

さらに用途によっては(4)位置情報の微分によりカーソルを移動させることも考えられ

50

る。実際の利用にあたっては、不感帯を設けたり、移動が大きいときには加速機能として多目の移動量を作成することもあるため、これらの混合となるものである。

瞳孔の移動を検知し移動量情報を作成する場合には、マウス・インタフェースを用いて他のシステムと接続するのか、ジョイスティック・インタフェースあるいは他のスティック・インタフェースを用いるのかによって、出力信号に対する考慮が必要であり、多種多様の変換形態があることが理解されよう。

【0014】

接続インタフェースは、制御手段をP C Iスロットに挿入する場合にはP C Iバス・インタフェースによるものとし、外部アダプタとする場合にはU S Bインタフェースによるものが好ましい。

さらには、パソコンそのものを制御手段として用いることも可能であり、その場合には、光源の照射(点滅)制御部分を分離して外部に設けることも考えられる。

所望により制御手段の制御機能を分割し、ソフトウェアで実行する部分、ハードウェアで実行する部分、パソコン自体で実行する部分、P C Iボードで実行する部分など、適宜機能分散することが可能であることは、当業者であれば理解できよう。

【0015】

瞳孔検出によりマウス代替とするには、左右の押下ボタンとの対応づけも重要である。

顔画像から検出される瞳孔の検出状態は、(a)両目が開いており、両瞳孔とも検出されている状態(右瞳孔検出、左瞳孔検出)、(b)右目のみが開いており、右瞳孔のみが検出されているとき(右瞳孔検出、左瞳孔非検出)、(c)左目のみが開いており、左瞳孔のみが検出されているとき、(右瞳孔非検出、左瞳孔検出)、(d)両目とも閉じており、両瞳孔とも検出されないとき(右瞳孔非検出、左瞳孔非検出)の4通りである。

汎用の2ボタンマウス信号から出力される情報は、マウスボタン情報(右ボタンのDOWN(押した状態)・UP(離れた状態)と左ボタンのDOWN・UP)とカーソル移動情報(水平方向、垂直方向)である。

例えば、片目を1度閉じてから開けばクリック、また、片目を閉じながら頭部を動かせば、ドラッグが開始され、その後閉じていた目を開けば、ドラッグが終了する。

そのために、右目の開閉と左目の開閉を、汎用2ボタンマウスの右ボタンのUP, DOWNと左ボタンのUP, DOWNに対応させる処理を行うものとする。この対応づけは左右入れ換えることも可能である。

ただし、瞬目(まばたき)により両目が閉じる場合は、それは被験者の意図ではないために例外的に扱う。通常、汎用マウスの両ボタンとも押している状態は利用されないため、両目が閉じている場合は、両ボタンとも逆にUPになっている状態とみなす。ただし、瞳孔移動量は零であるとする。これにより、両目が閉じられている場合は、マウスを何も操作していない状態(手を放している状態)に相当させることになる。このような状態は、両目が閉じている場合だけでなく、顔がカメラに映っていない場合も同様の扱いとする。

【0016】

(1)瞬きを行う場合、両瞳孔が非検出となるが、右と左の眼を閉じる際に、個人によって1~2フレーム程ずれがある場合がある。これは、瞬き後の眼を開くときも同様である。このような場合、単に、両瞳孔の検出状態変化を汎用マウスの左右のボタンに対応させると、ちょうど左右のボタンを、少しだけ時間をずらして押したのと同じ状況となり、汎用のマウスでこれが起こると多くのアプリケーションソフトウェアは使用者の意図とは異なった動作をすることになる。

(2)また、速く頭が動くと、それに伴って瞳孔も速く動く、その際に瞳孔の画像がボケてピーク輝度値が低下するなどして、結果的に瞳孔が検出できないことがある。これは、両目を開いているとき(カーソルを単に動かしているとき)において、両瞳孔に関して、もしくは、片方の瞳孔に関してありうる。その場合、瞳孔の検出状態が変化したと判断すると、被験者の意図とは異なったボタン情報を出力してしまう。

上の(1)と(2)のような問題を防ぐために、次のような対処を行う。

現在の画像フレームから得られた瞳孔の検出状態(検出、非検出)が、前のフレームから

10

20

30

40

50

別の状態に変化しても、変化後に同じ状態が数フレーム（例えば、3フレーム、100ms）以上連続に続いたときに、はじめて変化後の瞳孔状態に一致するボタン情報を出力するものとする。

【0017】

顔画像からの瞳孔検出においては、両瞳孔が検出されているとき、各瞳孔のX座標から左右の瞳孔を判定する。したがって、両瞳孔が開いている状態から、どちらか一方の目が閉眼したときは、座標の比較から、左右どちらの瞳孔が検出状態から非検出状態に変化したかを判定するものとする。また、両目が閉眼した状態から片目から先に開く場合は、両目が開かれるまでは先に開いた目がどちらの目か分からないため、このような状態変化の場合には、両瞳孔が検出されるまで、瞳孔が左右どちらか判定せず、両瞳孔が検出されてから初めて両瞳孔の左右を区別する。したがって、以前に両瞳孔が非検出状態であった場合は、両瞳孔が検出されたらと判定されるまでは、ボタン動作は左右ともUPのままの状態とし（両瞳孔が検出された後も、両瞳孔ともUPのまま）、カーソル移動は片目の瞳孔移動量に従うものとする。

10

一方で、片側の瞳孔のみ検出の状態から、他方の瞳孔のみの検出の状態に変移する場合がある。これは、ドラッグ中に片目を閉じていて、ドラッグ終了のときに、閉じていた目がちょうど開いたと判断されたときに、何らかの理由で開いていたほうの瞳孔が検出されなくなったときに生じる。しかし、片眼のみを閉じた状態から、反対側の片眼のみを閉じた状態に使用者が意図的にすることはありえないと仮定し、本来検出されるはずのもう一方の瞳孔が何らかの理由で検出されそくなって片目が開いていた状態から両瞳孔検出状態に変化した、と判断する。その後は、検出されている瞳孔のみの移動に対応させてカーソル移動をさせる。

20

【0018】

両瞳孔が検出され続けているときは（単なるカーソル移動に相当）、連続するフレーム間における両瞳孔の重心移動量を瞳孔移動量とし、それをカーソル移動量へ対応させることが望ましい。片方の瞳孔のみが検出され続けているときは、連続するフレーム間における、検出されている瞳孔の移動量を、瞳孔移動量として、カーソル移動量に対応させる。なお、使用者（被験者）が頭部運動中に（必ずしも運動していなくてもよいが）、意図とは関係なく、両目を開いているつもりが、指定した数フレーム間より短い時間に片方の瞳孔しか検出できなかつたときには（眼鏡反射が瞳孔像に重なったときなどにも起こる）、その間は、片方の瞳孔の移動量をカーソル移動量に対応させる。また、使用者が頭部運動中に（必ずしも運動していなくてもよいが）、意図とは関係なく、片目を開いているつもりが、指定した数フレーム間より短い時間に両瞳孔とも検出できなかつたときには、その間は、瞳孔の移動はないものとし、カーソルは移動しないものとする。

30

【0019】

本来は、使用者がカーソルを動かそうとしているにもかかわらず、瞳孔が検出できなかつた数フレーム間は瞳孔が移動しないことになる。この場合、カーソルが動かない状態で頭が回りきってしまうこともあり、使用者のその後の対処が面倒となる。そこで、瞳孔が検出できなかつた数フレームの直前の（1フレーム前の）瞳孔位置と、直後の瞳孔位置の差を、瞳孔移動量として、カーソル移動量に対応させるものとする。

40

両瞳孔が連続して検出されていたときに、使用者の意図とは関係なく、その瞳孔が指定した数フレーム間より短い時間に両瞳孔が検出できなかつた場合（頭部移動中に瞬目が生じた場合などが考えられる）も同様に対処する。ただし、片方の瞳孔が検出され続けている場合には、適用されず、その場合は、片方の瞳孔の移動量を、瞳孔移動量として、カーソル移動量に対応させる。

【0020】

被験者が緊張したりすると、頭部が振動し、それに合わせて瞳孔も振動し、カーソルが振動してしまう。（これは、瞳孔検出に問題があるのではなく、頭部の動きでカーソルを動かそうとすること自体の問題）。これを、後で述べる方法によって解決する。はじめに、一般的に考えられる方法を述べる。

50

通常は、得られる瞳孔移動量とカーソル移動量の対応付けを

$$C_x = k_x \cdot P_x \quad ,$$

$$C_y = k_y \cdot P_y$$

として対応付けることが考えられる。ここで、 P_x と P_y は、それぞれ画像中の水平方向、垂直方向の瞳孔移動成分（正負あり）、 C_x 、 C_y はパソコン画面中の水平方向、垂直方向のカーソル移動量である。また、 k_x と k_y は被験者によって好みの値を選ぶための係数である。

【0021】

頭部を回転もしくは移動させて、カメラから見た瞳孔を移動させることにおいて、 k_x 、 k_y を大きく設定すると、小さな頭部の動きでカーソルは大きく動くが、その代わりに、頭部を被験者が止めているつもりでも、頭部の振動が原因でカーソルが振動してしまう。逆に、 k_x 、 k_y を小さく設定すると、大きな頭部の運動を行わないと、カーソルを大きく動かすことができなくなる。できるだけ適当な値を選択しても、カーソルを希望の位置に合わせようとしたときに、カーソルが振動して位置合わせが難しい。

解決法としては、

$$C_x = k_x \cdot P_x \cdot |P_x|^{b-1} \quad ,$$

$$C_y = k_y \cdot P_y \cdot |P_y|^{b-1}$$

と対応させることである。 $b > 1$ のとき、前記の単純比例の場合より、 P_x 、 P_y の次数が高くなり、 $b = 2$ のとき瞳孔移動量の二乗をカーソル移動量に対応させることになる。このような対応付けにより、瞳孔の振動はほぼ零に抑えられ、頭部の運動による瞳孔移動は強調される。ここでも、 k_x 、 k_y は被験者の好みに合わせるものとする。このように、次数を高くすると、比較的すばやく頭部を動かした場合は、小さな頭部運動によってもカーソルは大きく移動し、逆に頭部をゆっくり動かす場合は、ほんの少しだけカーソルが移動できるようになる。したがって、カーソルを大きく動かしたいときも、希望の位置からカーソルがほんの少しずれているときに、それを希望の位置に合わせたいときにも、両方に適している。

【0022】

しかし、 $b = 2$ にしても、頭部の振動がカーソル移動に悪影響を与える場合は、いわゆる不感帯（dead zone）を設けて対処する。すなわち、 P_x 、 P_y が指定した閾値以下のときは、 $P_x = 0$ 、 $P_y = 0$ （もしくは、 $C_x = 0$ 、 $C_y = 0$ ）とする。このようにすると不感帯の範囲の小さな頭部振動では、カーソルがまったく動かなくなる。

または、不感帯を設けた上で、不感帯の範囲内の小さな瞳孔移動の間は、すべての瞳孔移動の積分値を記録しておき、不感帯を超える大きな瞳孔移動があったときは、その積分値を使用して、カーソル移動をさせる。顔の向く方向によりカーソル制御を行う場合には、このようにすることによって、ほぼ顔の向いた方向にカーソルを瞬間的に向けることが可能である。

さらに、不感帯と、不感帯の境界を原点とする指数関数を用いれば、頭部の小さな振動ではカーソルが全く動かず、不感帯よりも頭部の移動が大きくなると徐々に動くようにすることができる。

【0023】

クリックしたりドラッグを開始したりするとき、眼を開いた状態から、閉じる。また、ドラッグを終了するとき、眼を閉じた状態から開く。このように眼の開閉の短い時間（例えば、2フレーム間）、一般に、人間は、注意しないと眼が動いてしまう。特別な処理をしないと、眼の動きに合わせてカーソルが移動してしまい、使用者の意図とは違う位置にカーソルが動くために、クリックやドラッグをしたい場所とは異なった場所をクリックやドラッグをしてしまうことになる。この問題を防ぐために、ちょうど使用者が目的の位置にカーソルを合わせて、眼を閉じる、もしくは、開こうとする時点でのカーソル位置の座標が入力情報となることが望ましい。そこで、例えば、常に最近の数フレームにわたるカーソル移動量を記録しておき、クリックもしくはドラッグの開始、終了が起こったと判断されたフレーム（これをフレームBとする）から、時間を数フレーム分だけ遡ったフ

10

20

30

40

50

レーム（フレーム E とする）におけるカーソル位置にカーソルを戻すための処理を施す。すなわち、フレーム B からフレーム E までの間の瞳孔移動量の積分値を、フレーム B における瞳孔移動量から引いた値を、新たな瞳孔移動量に置き換えてカーソル移動に使用する。

【 0 0 2 4 】

片目の開閉の苦手な使用者もいる。そのような使用者についても、瞳孔マウスのボタン操作に相当する部分を他のボタン押し装置（汎用マウス、キーボードなどを含む）で代行させたり、併用させたりすることができる。例えば、大きなディスプレイ上のカーソルを制御する場合に有用である。

また、瞳孔マウスと汎用の手動マウスを併用することも有用である。

10

汎用の手動マウスは細かいカーソル移動は得意であるが、大きなカーソル移動をするためには、手を大きく動かすか、何度か空に浮かせて位置を戻す必要があり、使用しにくい面がある。それに対して、瞳孔マウスにおいては、大きな頭の動き（回転）に対して、カーソルは素早く動く。そして、細かい作業については、瞳孔マウスが作用しないようにしておき、手動マウスによりカーソルを動かす。

【 0 0 2 5 】

さらに、大きなディスプレイ上においてカーソルを見失うことがよくあり、使いにくい。瞳孔マウスを調整して、カーソルを頭の回転に合わせて大きく移動させるようにすれば、ほぼ頭の向いている方向にカーソルが移動させるようにできるので、カーソルを見つけやすくもなる。以上のことをするためには、不感帯を大きめに設定し、小さな瞳孔移動ではカーソルがまったく動かないように設定しておく。その上で、すべての瞳孔移動の積分値を記録しておき、再度、不感帯を超える大きな瞳孔移動があったときは、その積分値を使用して、瞳孔移動をさせる。このようにすることによって、ほぼ顔の向いた方向にカーソルを瞬間的に向けることが可能である。

20

【 0 0 2 6 】

ここまで、眼の開閉状態をマウスボタン押下に直接対応づけるものとして、説明した。この場合には、片目の 2 度のまばたきにより OS（あるいはアプリケーションソフト）は、ダブルクリックとの判定を行い、それに対応する動作を実行する。一方で、片目を開いたまま頭部を移動し、ドラッグに相当する操作を行いがたいユーザがいる。これに対処するために、片目の 2 度のまばたきを検出したときには、マウスボタン押下状態でロックしたものと判定すれば、両目開眼状態でドラッグ操作が行える。

30

左右の眼の開閉情報をポインティングデバイスの左右の押しボタンスイッチの押下状態情報に対応づける手段における対応づけを、使用状況に応じて切換え可能に構成することにより、有用性は高まる。

また、他の形態として、片目まばたきの繰返し周期の弁別を行うことにより、たとえば早い周期での 2 度のまばたきはダブルクリックとし、遅い周期での 2 度のまばたきは押下ロックとすることも考えられる。さらには、3 度のまばたきで押下ロックとするなど、ユーザの操作しやすい条件により、ロック状態を含めて種々のボタン押下信号を発生する変形が考えられる。

ここまで述べた瞳孔位置検出、瞳孔の有無によるまばたきの検出、瞳孔移動量からカーソル移動量への変換など画像処理及びマウス代替信号への変換機能を行う検出制御手段は図示されていない。

40

【 実施例 2 】

【 0 0 2 7 】

これまで、狭角のカメラで瞳孔を検出するものとして説明したが、狭角のカメラに代えて広角のカメラを使用することができる。

広角のカメラを使用した場合には、両目を撮影することにより、1 台のカメラ画像から 2 個の瞳孔が検出される。各カメラ画像からは、瞳孔が容易に検出でき、閉眼も明確にわかり、しかも、どちらの目を閉じたかも分かる。しかし、眼鏡を装着している場合、眼鏡反射が原因で、2 台のカメラ画像のどちらか一方で目を開いているにもかかわらず瞳孔が

50

検出できない場合がある。

この場合の判定手法について以下に述べる。

1. 2台のカメラの両方において、2つの瞳孔が検出されているときには、計4個の瞳孔のフレーム間の移動量の平均を、カーソルの移動量に反映させる。

2. 2台のカメラの両方において、左瞳孔のみ、もしくは右瞳孔のみが検出された場合は、左目もしくは右目が開いており、右目もしくは左目は閉じたと判断する。

【0028】

3. 2台のカメラで、両瞳孔とも検出できない場合は、両目が閉じたと理解するが、これは、瞬目(まばたき)か、被験者がカメラの前にいない場合は、何も起こらなかったと判断する。

4. 1台のカメラでは両瞳孔が検出され、もう1台のカメラでは、片方の瞳孔のみが検出された場合は、両目は開いていると判断して、計3個の瞳孔もしくは2台のカメラの両方で検出された瞳孔(計2個)の平均のフレーム間移動量を、カーソル移動量に反映させる。

5. 1台のカメラでは、右瞳孔のみが検出され、もう一台のカメラでは、左瞳孔のみが検出された場合、両目は開いていると解釈し、これら計2個の瞳孔のフレーム間移動量の平均を、カーソル移動量に反映させる。

6. 1台のカメラでは、右瞳孔もしくは左瞳孔が検出され、もう一台のカメラでは、両瞳孔とも検出されなかったとき、右目もしくは左目だけが開かれており、左目もしくは右目は閉じていると判断し、計1個の瞳孔のフレーム間移動量を、カーソル移動量に反映させる。

カメラ台数を増やすと、瞳孔の移動量も数多く得られる。この際に、瞳孔移動量の単純平均をカーソル移動に反映させるのではなく、求められた瞳孔移動量のうち最大と最小を除去して平均を求める、または奇数個のデータにあっては中央の値、偶数個にあっては中央の2つの平均を瞳孔移動量とするなど、統計的手法による考え方を導入することも考えられる。

【実施例3】

【0029】

[眼球回転に伴うカーソル移動の除去に関して]

次に、眼球回転による影響の除去について述べる。

瞳孔マウスは、本来、頭部の移動に伴ってビデオカメラ画像中を移動する瞳孔の座標の時間的变化とカーソル移動を対応づけるものである。瞳孔移動とカーソル移動の対応づけのための係数(式は、別途述べている)を変えることにより、同じ瞳孔移動に対するカーソル移動の度合いが変化するのはもちろんであるが、ビデオカメラに顔を写す際の拡大率も上記係数と同様にカーソル移動に大きく影響を与える。

もし、かなり広い範囲で(たとえば、30cmx30cmx30cm)、頭部が動いても使用できるようにするためには、ビデオカメラの拡大率を小さくして、頭部が大きく動いても常に両瞳孔が検出できなければならない。それに対して、小さな首の動きで大きなカーソル移動をさせたい場合、たとえば首の動きも思いのままならない患者が使用するためには、小さな頭部運動でカーソルを滑らかに動作させなければならないために、前記係数を大きくするよりは、ビデオカメラの拡大率を大きくして使用するのが理想的であり、たとえば顔の横幅がカメラ画像の横幅に一致する程度に写す場合が考えられる。その場合、空間内で、拡大率が小さいときと同じ大きさの瞳孔移動が起こったとしても(同じ頭部運動をしたとしても)、カメラ画像中での瞳孔移動は、拡大率が小さい場合に対して大きいので、結果的に、小さな頭部運動でも、十分に大きなカーソル移動が得られる。

【0030】

しかし、拡大率を大きくすると、頭部の運動が無くとも、眼球運動によりカーソルが移動する現象が発生する。その理由は、頭部が静止していても、眼球回転に伴い瞳孔が動き、それがビデオカメラ画像中でも移動するためである。拡大率を小さくして使用する場合は、頭部を大きく動かすのが一般的であるため、頭部運動に起因する瞳孔移動が眼球回転に起因する瞳孔移動に比較して十分に小さく後者を無視できても、拡大率を大きくして使

10

20

30

40

50

用する場合は、無視できなくなる。結果的に、被験者がパソコン画面上のいろいろな場所を観察するために頭部がまったく動かずとも、目を動かすたびにカーソルは小刻みに動くことになる。場合によって、これが被験者にとって不快に感じたり、誤操作の原因になる可能性もある。

そこで、その問題を解決する方法を以下に述べる。

図10に示すように、眼球(20)は中心のずれた2重球からなり、その一方を角膜球(23)とする目のモデルを考えることができる。今、通常、眼球の大きさは、カメラから眼球までの距離に比べ十分に小さいので、便利のためにカメラから無限遠に眼球があるとする。また、光源(6)がカメラ(1)の開口の中心にあるとする。図10(b)のように、視線(25)がカメラの方を向いていると、光源の角膜反射(24)は瞳孔(9)の中心付近に現れる(図10(b)右)。さらに角膜反射(24)と瞳孔中心(22)の延長線上には眼球回転中心(21)が存在する。ここで、図10(a)のように、視線がカメラからずれた位置に向いたとすると、角膜反射(24)と瞳孔中心(22)はずれる。さらに眼球回転中心(21)もずれる。

10

【0031】

ここで、頭部が動かない限り、眼球回転中心は、眼球が回転してもほとんど動かない。したがって、瞳孔の代わりに眼球回転中心の移動をカーソル移動に対応させれば、目が動いても頭部が動かない限りカーソルは動かないことになる。

しかし、眼球中心は画像中に写るわけではないので、位置を推定する必要がある。それについて、以下に述べる。まず、図10では、瞳孔はひとつしか表現していないが、一般には2個の瞳孔とも同時に写る。640×480画素程度や320×240画素程度の分解能のビデオカメラでは、顔の横幅が画像の横幅に一致する程度の拡大率、もしくはそれ以下の拡大率では、光源をカメラの開口部の周りに並べた場合にも、図11のように光源は、ひとつの光点として写る。

20

【0032】

今、図11のように、眼球回転中心から角膜中心までの距離を r 、眼球回転中心から瞳孔中心までの距離を R として、視線方向が、カメラの存在する方向からだけずれているとする。ここで、視線は眼球回転中心と角膜球中心と瞳孔中心を通る直線であり、視線の先が被験者の見ている視点である。

このとき、カメラの方向から見たときの眼球回転中心から瞳孔中心までの距離を p 、眼球回転中心から瞳孔中心までの距離を g とすると、

30

【数1】

$$g = r \sin\theta \quad (1)$$

【数2】

$$p = R \sin\theta \quad (2)$$

である。よって、

【数3】

$$g/p = r/R = k \quad (3)$$

とおくと、 $0 < k = r/R < 1$ で表せる。ここで、人間の標準的な寸法は、

$r = 5.9\text{mm}$ 、 $R = 9.7\text{mm}$ であることが知られているので、それから $k = 0.6$ と求まる。

40

【0033】

画像中では、推定する眼球回転中心を $E(X_e, Y_e)$ と置き、画像解析によって求まる瞳孔中心と角膜反射をそれぞれ $P(X_p, Y_p)$ 、 $G(X_g, Y_g)$ とすると、

【数4】

$$\overrightarrow{EG} = (X_g - X_e, Y_g - Y_e) \quad (4)$$

【数5】

$$\overrightarrow{EP} = (X_p - X_e, Y_p - Y_e) \quad (5)$$

50

と表せ、ベクトル \vec{EG} とベクトル \vec{EP} との間には、(3)式と同様に

$$\text{【数 6】} \quad \frac{\vec{EG}}{\vec{EP}} = k \quad (6)$$

の関係が成り立つ。

【0034】

よって、以下のように変形して、

$$\text{【数 7】} \quad k \cdot \vec{EP} = \vec{EG}$$

$$k \cdot (X_p - X_e, Y_p - Y_e) = (X_g - X_e, Y_g - Y_e)$$

$$k \cdot (X_p, Y_p) - k \cdot (X_e, Y_e) = (X_g, Y_g) - (X_e, Y_e)$$

$$(1-k)(X_e, Y_e) = (X_g, Y_g) - k \cdot (X_p, Y_p)$$

$$(X_e, Y_e) = ((X_g, Y_g) - k \cdot (X_p, Y_p)) / (1-k) \quad (7)$$

を得る。(7)式を使用することにより、画像中の角膜反射座標 (X_g, Y_g) と瞳孔中心座標 (X_p, Y_p) から、眼球回転中心 (X_e, Y_e) を推定することができる。ここで、 k の値は個人で異なるが、大きな違いがあるわけではないので、 $k = 0.6$ を与える。瞳孔中心 (X_p, Y_p) のフレーム間移動をカーソル移動に対応させるのに比較すれば、眼球回転中心座標 (X_e, Y_e) を対応させたほうが、明らかに眼球運動(眼球回転)のカーソル移動への影響は小さくなる。もちろん、頭部を固定して、故意に眼球運動を起こさせることにより、次に述べる方法により k を推定することは可能である。

【0035】

(3)式より、

$$\text{【数 8】} \quad \Delta g / \Delta p = k \quad (8)$$

であるため、頭部を固定するか、できるだけ静止させて、目を動かして複数の位置を被験者に見させ、そのときの角膜反射の位置の変化 g と瞳孔中心の変化 p をプロットし、その傾きから k を求める。

以上のように、 k を被験者ごとに実験的に求めても、はじめから適当な値(たとえば、 0.6)を与えても良い。(7)式により眼球回転中心を求めて、瞳孔中心の代わりに眼球回転中心の移動に合わせてカーソルを動かす方法が、目の回転に依存せず、頭部の回転のみ依存するカーソル制御ができる。

【0036】

しかし、頭部を大きく回転させた場合には、角膜反射がまぶたやまつげによって隠れたり、カメラに対する角度が大き過ぎ角膜反射が現れず、代わりに白目が光ることがある。このような場合、カメラから角膜反射が観察できず、眼球回転中心を推定することができない。

そのような場合の処理について以下に述べる。

(i)連続する2枚のビデオフレーム画像において、両画像において角膜反射が写っている場合は、(8)式より

$$\text{【数 9】} \quad (\Delta X_e, \Delta Y_e) = ((\Delta X_g, \Delta Y_g) - k \cdot (\Delta X_p, \Delta Y_p)) / (1-k) \quad (9)$$

と表すことができ、この (X_e, Y_e) を瞳孔移動の代わりにカーソル移動に使用する。ここで、 (X_e, Y_e) 、 (X_g, Y_g) 、 (X_p, Y_p) は、連続するフレーム間でのそれぞれ眼球回転中心移動、瞳孔中心移動、角膜反射移動である。

(ii)連続する2枚のビデオフレーム画像のうち、どちらか一方もしくは両方において、角膜反射が検出されなかったときは、2フレーム間の瞳孔中心移動 (X_p, Y_p) をカーソ

10

20

30

40

50

ル移動に使用する。

【実施例 4】

【0037】

[瞳孔中心を高精度に検出するための画像処理に関して]

次に、瞳孔中心の検出精度の向上について述べる。

瞳孔の動きに対してカーソル移動を制御する場合、瞳孔中心を高分解能で検出することが望ましい。たとえば、一般に使用されるテンプレートマッチング法では、座標が1画素毎にしか得られないため、滑らかなカーソル移動は難しい。そこで、後で述べるような方法に瞳孔中心を高精度に検出するが、その前の画像処理における前提について述べる。

基本的には、全画像に対して同一の画像処理をして、瞳孔中心の高精度検出は難しい。しかも、全画像中には、瞳孔と特徴が類似した画像が現れることもあるため、それを誤認識しないようにしなければならない。

そこで、差分画像より輝度が明るい部分として瞳孔を一度検出したら(そのときの中心座標の高い精度は必要ない)、そこにウィンドウを与え、次のフレームからはウィンドウ内だけを瞳孔探索するようにする。それによって、ウィンドウ以外の画像処理が必要なくなるため、ウィンドウ内の画像処理演算に多くの時間をかけることができる。ウィンドウサイズとしては、顔の縦全体がちょうど画像に入るくらいのサイズであれば、 320×240 画素の分解能の画像で、 21×21 画素程度の方形ウィンドウを与えればよいし、顔の横全体がちょうど画像に入る程度であれば、 25×25 画素程度でもよい。いずれにしても、このウィンドウのサイズは、頭部や瞳孔の動くスピードを考慮しなければならない。通常の1秒間に30フレームのビデオカメラ(N T S C方式)では、1フレームの時間に瞳孔が大きく動くため、次のフレームでの瞳孔位置をカルマンフィルターなどの予測モデルを利用して、予測した瞳孔位置を中心にした上記のウィンドウを設定する必要がある。

【0038】

さて、高精度の瞳孔中心の算出法について少し考察する。瞳孔中心を求める際に、差分画像の輝度を考慮した重心法により、瞳孔中心を求める方法が考えられる。重心法では、高分解能が期待できる。しかし、差分画像においても、図12(a)で示す例のように、瞳孔が中心にあり、その周囲にまぶたが観察されるような場合がある(差分画像を使用するため、まぶたに動きがあると、現れる)。このとき、単なる輝度重心を瞳孔中心として求めようとしても、どの範囲の画像の重心計算をすればよいか問題となり、範囲の設定によっては大きな誤差に繋がったり、不安定な結果となりうる。したがって、単なる重心法は使用しにくい。さらに、図12(a)のような画像を2値化してから重心を求める方法も考えられるが、さらに精度に期待が持てない。

最近、円形に近い黒目の検出のために、分離度フィルタ(K.Fukui: "Edge extraction method based on separability of image features", IEICE Trans. Inf. & Syst., Vol.E78-D, No.12, pp.1533-1538(1995)) が用いられることがある(M.Yuasa, O.Yamaguchi, K.Fukui: "Precise pupil contour detection based on minimizing the energy of pattern and edge", IEICE Trans. Inf. & Syst., Vol.E87-D, No.1, pp.105-112 (2004),

川口 剛、モハメッド リゾン、日高大輔: "ハフ変換と分離度フィルタによる人物顔からの両目の検出", 信学論D-II, Vol.J84-DII, No.10, pp.2190-2200 (2001))。

この場合、図13のように、2重円状のマスクを用いて、内側の領域を領域1、外側の領域を領域2とし、画像中で分離度が最大になる座標を求める。しかし、分離度が最大になる座標を求めるという方法では画素単位の分解能でしか座標が検出できず、滑らかなカーソル移動に支障を来す。そこで、最大値を示す画素を中心に狭い範囲の重心を求める方法が考えられるが、対象が数ピクセルであることから、あまり分解能等に期待が持てない。範囲を広げると、重心に大きく影響する分離度値を示す画素が範囲内に存在する可能性があり、やはりあまり精度が期待できない。

【0039】

さらに、分離度フィルタは一般に多くの演算時間を必要とするなどの問題がある。そこで、以下のような手法をとる。

10

20

30

40

50

基本的に、差分画像において瞳孔が円形で半径が可変であり、周囲より輝度が高いという特徴を持つということを利用する。

差分画像の座標 (x, y) における瞳孔特徴量 $P(x, y)$ を次式のように定義し、ウインドウ内について求める。

【数 1 0】

$$P(x, y) = I(x, y) \cdot S(x, y) \quad (10)$$

ここで、 $I(x, y)$ は座標 (x, y) における差分画像の輝度(Intensity)である。 $S(x, y)$ については次に述べる。計算時間の短縮化のために、図 1 4 のような形状をした 8 方向に 5 画素分伸ばした放射状マスク上の演算のみを行う(計算時間が長くなってよければ 5 画素より多くても良い。また、8 方向とは限らない)。すなわち、解析対象である差分画像の注目画素 (x, y) に図 1 4 のマスクパターンの中央の画素が位置するように重さね、マスク下の差分画像上の画素のみについて演算を行う。図 1 4 に示したマスクの中央の画素から L 画素分の領域(破線で示した方形内の領域; 領域 1)とその外側の領域(領域 2)に分ける。 L を変化させるたびに(1 から 5)分離度 s を計算し(後述)、 s 値が最大となるときの L (L_m) を決定する。また、そのときの s 値を各座標 (x, y) における最大を示す分離度とし、 $S(x, y)$ に置き換え、(1)式に使用した。ここで、注目すべき点は、ウインドウ内の差分画像が図 1 2 (a) のように目蓋(まぶた)が写っている場合でも、 $P(x, y)$ を表す画像は図 1 2 (b) のようになり、目蓋が消えてほぼ瞳孔の中心部だけが高い値を持つ画像になることである。(なお、図 1 4 に描いた円は、瞳孔の輪郭を模擬的に示している。瞳孔は通常円であるのに対して、領域 1 と 2 との境界は方形になるように設定してある。実際は、できるだけ円になるように放射状マスクの 1 本、1 本の長さを変えたほうが良いことは言うまでもないが、この例では簡略化して示してある。)

【0 0 4 0】

さらに、ウインドウ内で $P(x, y)$ の最大値を求め、それが閾値(経験値)を超えなかったときは瞳孔が存在しないと判断し、超えたときは瞳孔が存在すると判断する。この方法により、以前のフレームで存在した瞳孔が現在にフレームで消失した場合に、それを判断する。さらに、ウインドウを解除するきっかけ(トリガー)に利用する。瞳孔が存在すると判断されたら、最大値を示した座標を中心に一辺 $2 L_m \sim 4 L_m$ の程度の方形領域内における $P(x, y)$ の重心を計算し、最終的な瞳孔中心とする。

しかし、[眼球回転に伴うカーソル移動の除去に関して]の項でも述べたように、ビデオカメラの拡大率が大きい場合には、放射状マスクの各ラインの長さは、中央を含めると 5 画素である。大きく写る瞳孔中心を正確に求めるためには、このラインの長さを長くすれば解決できるが、それでは、計算時間が長くなってしまふ。それを解決するためには、図 1 5 のように、マスクの中心範囲に近いほうの画素の代わりに遠いほうの画素を増やし、結果的にマスクの画素数を変えないようにする。このような工夫によって、演算時間を増やさずにビデオカメラの拡大率に対応させることができる。

【0 0 4 1】

なお、分離度 s は、画像中の 2 つの領域 1 と領域 2 があるとき、式(1 1)から(1 3)における領域 1 と領域 2 からの情報を使用して線形判別法によって計算できる。

【数 1 1】

$$\eta = \sigma_b^2 / \sigma_T^2 \quad (11)$$

【数 1 2】

$$\sigma_b^2 = n_1(\bar{P}_1 - \bar{P})^2 + n_2(\bar{P}_2 - \bar{P})^2 \quad (12)$$

【数 1 3】

$$\sigma_T^2 = \sum_{i=1}^{n_1+n_2} (P_i - \bar{P})^2 \quad (13)$$

ここで、 P_i は 1 つの画素 i における画像の特徴(たとえば、輝度)であり、アッパーライン P_1 とアッパーライン P_2 はそれぞれ領域 1 と 2 の平均値である。アッパーライン P

10

20

30

40

50

は両領域全体における平均であり、 n_1 と n_2 はそれぞれ領域1と2の画素数である。(K .Fukui: "Edge extraction method based on separability of image features", IEICE Trans. Inf. & Syst., Vol. E78-D, No.12, pp.1533-1538 (1995)を参照)

【産業上の利用可能性】

【0042】

(1) 本方法では、2台のカメラを用いているが、ステレオカメラというわけではないので、頭を動かす範囲に、2台のカメラを適当に向ければよく、カメラ較正も必要ない点で使用しやすい。

(2) 1台のカメラでは最多で2個までの瞳孔の移動の平均をカーソルに対応させていたが、2台のカメラでは最多で4個まで対応させることができるため、カーソル移動により高い安定性が得られる。とくに、ドラッグをさせるときには、もともと片方の瞳孔移動だけを検出するので効果が大きい。

(3) 眼鏡反射がなく、目を開いていても、何らかの理由で瞳孔が検出できないときにも、それを補うことができる。

(4) 瞳孔の検出率を向上させる場合に、一度瞳孔を検出したら、その周囲の小さなウィンドウを与え、次のフレームでは、そのウィンドウの内部だけを探索し瞳孔を検出する方法が有効である。すばやい瞳孔の動きに対応するためには、前のフレームとさらにその前のフレームの瞳孔位置から、現在のフレームにおける瞳孔位置を予測し、ウィンドウを与えることによって、瞳孔の検出率は飛躍的に向上する。一台のカメラのみの場合には、目が開いているにもかかわらず瞳孔が検出できなくなった場合、ウィンドウを解除するか、ウィンドウを大きくして瞳孔を検出するしかない。しかし、ウィンドウを大きくすれば、その分、誤って瞳孔以外の部分を瞳孔として検出する可能性が高くなり、正しい瞳孔検出率が低下する。2台のカメラを用い、両者の距離をあまり大きくしなければ、両者のカメラによって左右の瞳孔の位置関係はほぼ同じであるため、片方のカメラ画像での瞳孔のフレーム間移動量は、もう片方のカメラのそれとあまり変わらない。そこで、瞳孔が検出されているほうのカメラ画像内での瞳孔移動と同等の瞳孔移動が、瞳孔が検出されなかったカメラ画像内においても起こっているとして、その移動量にあわせて動的ウィンドウをしばらくの間(数フレーム間)与え続ける。その間に、眼鏡反射像が瞳孔像を通り抜けて、再度、そのウィンドウ内で検出できれば、カーソルは何事もなかったかのように正常に移動することになる。このように、2台のカメラから得られる左右の瞳孔の対応付け(マッチング)が容易であるので、眼鏡反射などが理由で、目を開いているにもかかわらず瞳孔が検出できなかつたときに、有効である。

【図面の簡単な説明】

【0043】

【図1】明瞳孔画像と暗瞳孔画像との差分による瞳孔検出の例を示す図。(a)は明瞳孔画像、(b)は暗瞳孔画像、(c)は差分画像を示している。

【図2】カメラにおける光源の配置例を示す図

【図3】カメラにおける光源の別の配置例を示す図

【図4】眼鏡反射を伴う画像による差分を示す図。(a)は眼鏡反射像が写っている明瞳孔画像、(b)は眼鏡反射像が写っている暗瞳孔画像、(c)は高輝度部分の輝度を落としてから差分をした画像を示す図。

【図5】カメラの左右に光源を設けた例を示す図。(a)はカメラと光源と瞳孔との位置関係、(b)は左側の光源が点灯したときのカメラ画像の一部、(c)は右側の光源が点灯したときのカメラ画像の一部を示している。

【図6】カメラ2台を設けた例を示す図。(a)はカメラと光源と瞳孔との位置関係、(b)は右光源が点灯したときの左カメラ画像の一部(暗瞳孔)、(c)は左光源が点灯したときの左カメラ画像の一部(明瞳孔)、(d)は右光源が点灯したときの右カメラ画像の一部(明瞳孔)、(e)は左光源が点灯したときの右カメラ画像の一部(暗瞳孔)を示している。

【図7】2台のカメラを設けて、それぞれに2つの光源を設けた例を示す図

- 【図 8】バンドパスフィルタを設けた例を示す図
- 【図 9】眼鏡反射が瞳孔と一致する場合を示す図
- 【図 10】眼球中心を求めるための概念を示す図。(a)は視線がカメラからずれた位置に向いている状態、(b)は視線がカメラの方を向いている状態を示す。
- 【図 11】眼球中心を求めるための詳細を示す図
- 【図 12】まぶたと瞳孔とが抽出された例を示す図。(a)はまぶたが写っている状態、(b)は演算処理によりまぶた像が除去された状態を示す。
- 【図 13】分離度フィルタの概念を示す図
- 【図 14】分離度フィルタにおける輝度抽出点を示す図
- 【図 15】分離度フィルタにおける輝度抽出点の変形例を示す図

10

【符号の説明】

【0044】

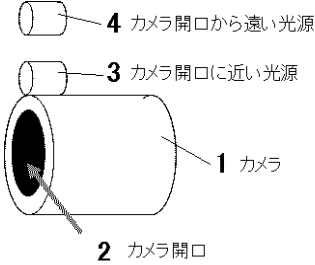
- 1 カメラ
- 2 カメラ開口
- 3 カメラ開口に近い光源
- 4 カメラ開口から遠い光源
- 5 発光ダイオード(LED)
- 6 光源
- 7 眼鏡レンズ
- 8 角膜
- 9 瞳孔
- 10 眼鏡反射
- 11 バンドパスフィルタ

20

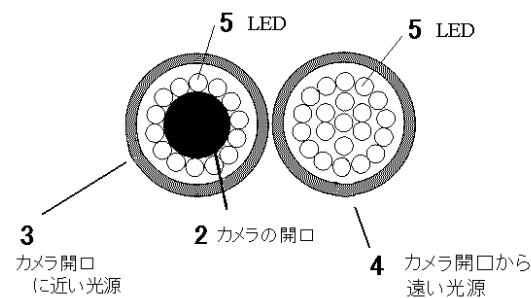
【図 1】



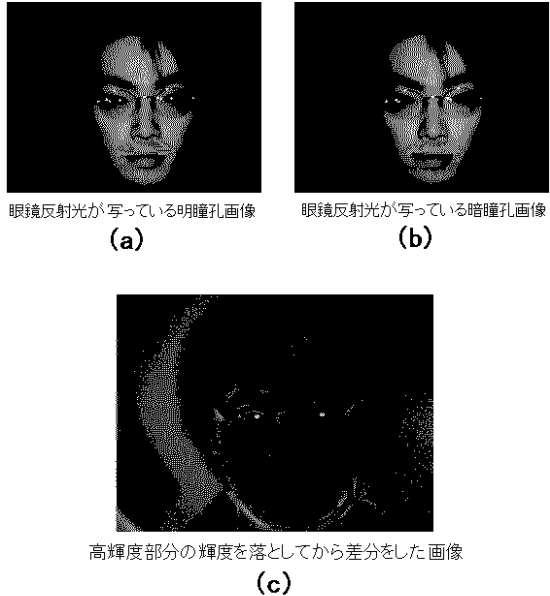
【図 2】



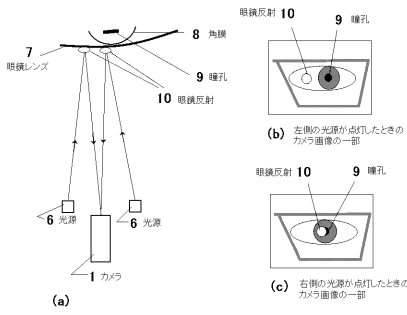
【図 3】



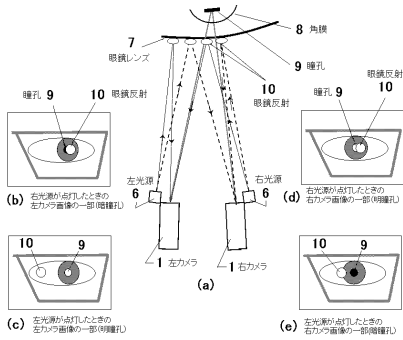
【図 4】



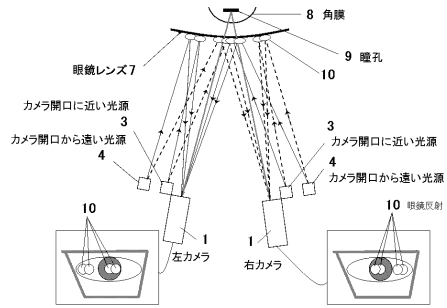
【図5】



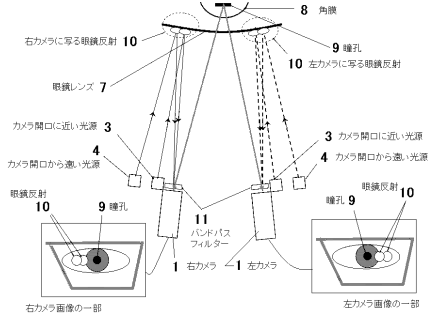
【図6】



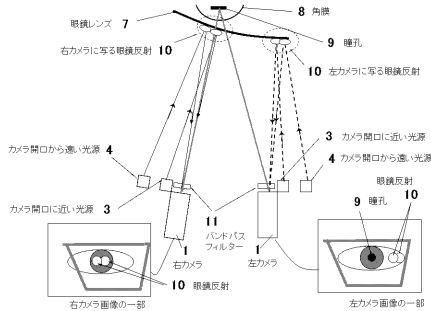
【図7】



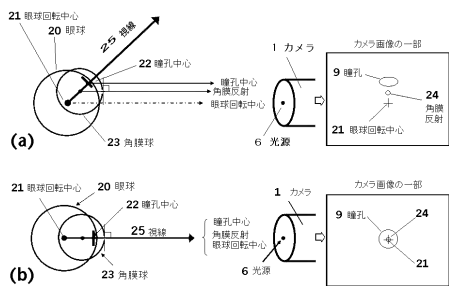
【図8】



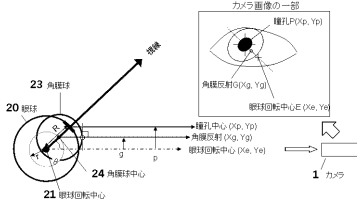
【図9】



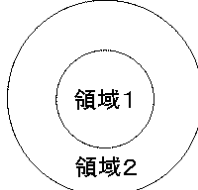
【図10】



【図11】



【図13】



【図14】

