

目入力装置のための顔の回転変化に頑健な黒目検出法 Iris Detection Method Robust to the Face Rotation Changes For Eye Input Device

浅野 誠之[†] 高野 博史[†] 中村 清実[†]
Masayuki Asano Hironobu Takano Kiyomi Nakamura

1. はじめに

近年、コンピュータの発展とともにマウスやキーボードのような手を使ったインタフェースだけでなく、非接触で入力可能なインタフェースの開発が盛んに行われている[1]~[4]。目を用いた入力装置は、手が使えない状況下においても入力操作が可能であり、手足が不自由な障害者あるいはALS(筋萎縮性側索硬化症)や脳卒中の患者に需要が見込まれる。ここで、入力装置に用いる目検出法には、目の誤検出や検出見逃しを極力抑えることが求められている。例えば、入力操作中に誤検出や検出見逃しが発生した場合、使用者の意図に反した誤操作が行われる可能性がある。ゆえに、操作性の優れた入力装置を開発するためには、目の状態変化を正確に検出する必要がある。

目検出の従来手法としては、アイモデルを用いるもの[5][6]、AAM(Active Appearance Model)を用いるもの[7][8]、SSR(Six-Segmented Rectangular)フィルタを用いるものなどがある[9]。

Yuilleらの手法は、まぶたや黒目から構成されるDeformable Templateを画像中の目にフィッティングさせることで目を検出する。この手法は、目形状とテンプレートのマッチング度から求まる最適なパラメータを求めるのに時間がかかる問題がある。Wuらによる手法は、黒目の輪郭とまぶたの形状から構成されるアイモデルを画像中の黒目にフィッティングさせることで黒目を検出する。この手法は、Condensation[10]を用いることによって瞬きなどのオクルージョンに対応しているが、3次元の姿勢変化に対応していない。Kobayashiらと高谷らによる手法は、AAMを用いることで目を検出する。これらの手法は、学習した人物と同一人物でなければ正検出率が低下する問題がある。Sawetanusornらによる手法は、6分割した領域の平均濃度の明暗関係から目を検出する。この手法は、顔向きが変化した場合十分な検出率が得られない。

本研究では、これらの手法の問題点の一つである顔向きに対応するために、エッジ強度とパーティクルフィルタを用いた黒目検出法を提案する。本論文では、カメラの光軸に垂直な平面における顔の回転変化や頭部を上下左右に振る動作のような3次元的な顔向きのうち、顔の回転変化に対する目検出精度を調べた。

2. パーティクルフィルタ

本研究では、オクルージョンやノイズに対して頑健かつ安定して黒目を検出するために、パーティクルフィルタを用いる[11]。これは、時系列フィルタの一種であり、状態の確率密度関数にガウス性や単峰性といった制約がない。よって、パーティクルフィルタでは任意の確率密度関数を扱うことが可能である。以下にパーティクルフィルタの概要を述べる。

2.1 尤度推定

時刻 t における状態ベクトルを \mathbf{x}_t 、画像から得られる観測結果を \mathbf{y}_t 、時刻 t までに得られた観測結果を $\mathbf{Y}_t = (\mathbf{y}_1, \dots, \mathbf{y}_t)$ とする。このとき、時刻 t における状態ベクトル \mathbf{x}_t の事後分布 $p(\mathbf{x}_t | \mathbf{Y}_t)$ は、ベイズの定理を用いて式(1)のように求められる。

$$p(\mathbf{x}_t | \mathbf{Y}_t) \propto p(\mathbf{y}_t | \mathbf{x}_t) p(\mathbf{x}_t | \mathbf{Y}_{t-1}) \quad (1)$$

ここで、 $p(\mathbf{y}_t | \mathbf{x}_t)$ は \mathbf{y}_t の尤度であり、 $p(\mathbf{x}_t | \mathbf{Y}_{t-1})$ は事前分布である。ここで、 $p(\mathbf{x}_t | \mathbf{Y}_{t-1})$ は \mathbf{x}_t のマルコフ性を考慮すると、式(2)に示すように時刻 $t-1$ の事後分布 $p(\mathbf{x}_{t-1} | \mathbf{Y}_{t-1})$ と状態推移確率 $p(\mathbf{x}_t | \mathbf{x}_{t-1})$ で表わされる。

$$p(\mathbf{x}_t | \mathbf{Y}_{t-1}) = \int p(\mathbf{x}_t | \mathbf{x}_{t-1}) p(\mathbf{x}_{t-1} | \mathbf{Y}_{t-1}) d\mathbf{x}_{t-1} \quad (2)$$

2.2 アルゴリズム

パーティクルフィルタは、状態量と尤度を持つ多数のパーティクルによって確率分布を離散的に近似し、確率モデルを伝播させることで追跡を行う。

事前分布 $p(\mathbf{x}_t | \mathbf{Y}_{t-1})$ に従うパーティクル群を $\mathbf{S}_{t|t-1} = \{\mathbf{s}_{t|t-1}^{(n)}\}_{n=1}^N$ 、事後分布 $p(\mathbf{x}_t | \mathbf{Y}_t)$ に従うパーティクル群を $\mathbf{S}_{t|t} = \{\mathbf{s}_{t|t}^{(n)}\}_{n=1}^N$ とすると、事前分布と事後分布は下式のように表わすことができる。

$$p(\mathbf{x}_t | \mathbf{Y}_{t-1}) \approx \frac{1}{N} \sum_{n=1}^N \delta(\mathbf{x}_t - \mathbf{s}_{t|t-1}^{(n)}) \quad (3)$$

$$p(\mathbf{x}_t | \mathbf{Y}_t) \approx \frac{1}{N} \sum_{n=1}^N \delta(\mathbf{x}_t - \mathbf{s}_{t|t}^{(n)}) \quad (4)$$

ただし、 N はパーティクル数であり、 δ はディラックのデルタ関数である。これらを踏まえて、パーティクルフィルタは以下に示すアルゴリズムを繰り返すことによって対象物の追跡を行う。

- (1) $n=1, \dots, N$ について $\mathbf{S}_{0|0} \sim p(\mathbf{x}_0)$ を生成。ただし、 $p(\mathbf{x}_0)$ は状態 \mathbf{x}_t の初期分布である。
- (2) 時刻 $t=1, \dots, T$ について以下のステップ(a), (b), (c) を実行する。
 - (a) それぞれのパーティクルに対して以下のステップ i, ii を実行する。
 - i. 1次元のシステムノイズを乱数として生成 ($\mathbf{v}_t^{(n)} \sim q(\mathbf{v})$)。
 - ii. システムモデル

$$\mathbf{s}_{t|t-1}^{(n)} = \mathbf{F}_t(\mathbf{s}_{t-1|t-1}^{(n)}, \mathbf{v}_t^{(n)}) \quad (5)$$

- (b) 事前分布に従うパーティクル $\mathbf{s}_{t|t-1}^{(n)}$ について、重み $\pi_t^{(n)}$ を下式を用いて推定。

[†] 富山県立大学大学院 工学研究科
Graduate School of Engineering, Toyama Prefectural University

$$\pi_t^{(n)} = \frac{p(y_t | x_t = s_{t|t-1}^{(n)})}{\sum_{n=1}^N p(y_t | x_t = s_{t|t-1}^{(n)})} \quad (6)$$

$$L_0 \leq L_1 \text{ and } L_0 \leq L_2 \quad (7)$$

ここで、 $p(y_t | x_t = s_{t|t-1}^{(n)})$ は、状態 x_t が $s_{t|t-1}^{(n)}$ であるときに観測値 y_t を得る確率である。

(c) $s_{t|t-1} = \{s_{t|t-1}^{(n)}\}_{n=1}^N$ から $s_{t|t-1}^{(n)}$ を $\pi_t^{(n)}$ の確率でリサンプリングし、 $s_{t|t} = \{s_{t|t}^{(n)}\}_{n=1}^N$ を生成。

3. エッジ強度を用いた目検出法

3.1 黒目検出の流れ

黒目検出のフローチャートを図1に示す。始めにパーティクルを生成する。カメラから入力された256階調のカラー画像(640×480 pixels)に対してグレイスケール化を行い、その後、パーティクルの上下の一定領域に対して輝度値をチェックする。ここで、条件を満たしたパーティクルに対して、さらに4領域勾配チェックを行う。その後、それぞれの条件に合った尤度推定を行い、パーティクルのリサンプリングを行う。なお、輝度値チェックは誤検出率と計算量を減少させるために、4領域勾配チェックは黒目の検出候補点を削減するためにを行う。

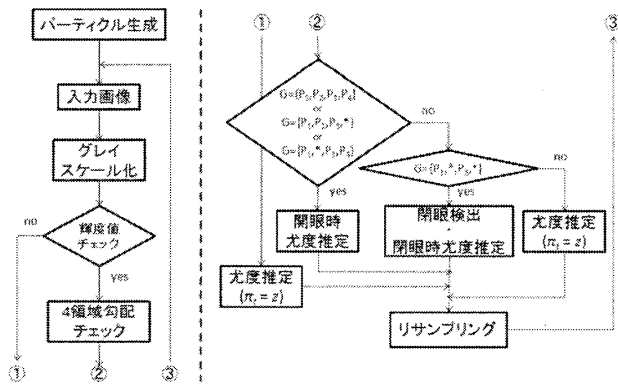


図1 黒目検出のフローチャート

3.2 輝度値チェック

輝度値チェックでは、黒目より低い輝度値を示す領域が黒目の上下にないと仮定している。ゆえに、式(7)を満たせばパーティクルが黒目の領域内に存在する可能性があるかと推定できる。図2に輝度値チェック領域を示す。図2において、 L_0 はパーティクルの位置の輝度値、 L_1 と L_2 はパーティクルの位置から上下に一定距離離れた領域 v_1 、 v_2 における平均輝度値であり、領域 v_1 、 v_2 はそれぞれ眉毛の下端から黒目の上端と黒目の下端から下の領域を想定した一定サイズの領域である。

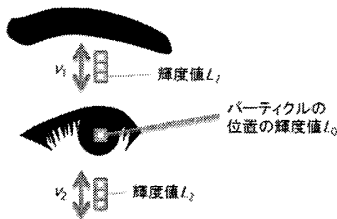


図2 輝度値チェック領域

3.3 4領域勾配チェック

3.2節の輝度値チェックだけでは、黒目の検出候補点は多数存在する。ゆえに、黒目の4つの領域の勾配方向をチェックすることで、さらに黒目検出候補点を削減する。

3.3.1 エッジ方向特徴量

本研究では、黒目を検出するために4種類のPrewittオペレータ $\{P_1: \text{下から上}, P_2: \text{右から左}, P_3: \text{上から下}, P_4: \text{左から右}\}$ を使用する(図3)。これらのオペレータを使用して、パーティクル近傍の一定領域内における勾配方向をチェックする。なお、算出したエッジ強度に定数 amp を乗じることで黒目のエッジを強調した。位置 (x, y) を基準としてオペレータ $P_j (j=1, \dots, 4)$ をかけて得られるエッジ強度を $f_j(x, y)$ とし、式(8)に示すように amp を乗じてエッジ強度 $f_j'(x, y)$ を求める。

$$f_j'(x, y) = amp \times f_j(x, y) \quad (8)$$

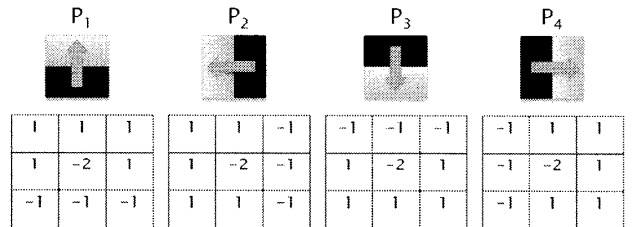


図3 4種類のPrewittオペレータ

3.3.2 勾配方向チェック

黒目の検出候補点をさらに削減するために、勾配方向チェックを行う。勾配方向チェックを行う4つの領域を図4に示す。ここで、領域 R_1, R_3 は縦幅 H_{ud} pixels, 横幅 W_{ud} pixels の大きさであり、領域 R_2, R_4 は縦幅 H_{lr} pixels, 横幅 W_{lr} pixels の大きさである。また、パーティクルの位置から領域 R_1, R_3 は $\pm y'$ pixels, 領域 R_2, R_4 は $\pm x'$ pixels 離れている。領域 $R_i (i=1, \dots, 4)$ において、Prewittオペレータ $P_j (j=1, \dots, 4)$ をかけて得られる勾配強度の和を g_{ij} とする(式(9))。式(10), (11)のように、領域 $R_i (i=1, \dots, 4)$ において g_{ij} が最大となるオペレータを D_i とすると、4領域における勾配強度が最大となる組み合わせ $G = \{D_i | i=1, \dots, 4\}$ が $G = \{P_1, P_2, P_3, P_4\}$ (条件1) となるとき最も目らしいとして尤度を計算する。しかし、視線や顔向きが変化すると黒目の形状が変化するため、上記のチェックだけでは対応できない。よって、勾配チェックが $G = \{P_1, P_2, P_3, *\}$ (条件2) もしくは $G = \{P_1, *, P_3, P_4\}$ (条件3) である場合も目らしいと推定し尤度を計算する。ここで、“*”は任意のPrewittオペレータを示す。しかし、条件2, 3を満たす領域は目以外にも多数存在するので、条件1を満たすときに比べて尤度を低くするために、算出した尤度に重み係数 α をかける。なお、尤度の算出方法は3.4節で詳しく述べる。また、閉眼時においては、領域 R_2, R_4 の勾配方向は必ずしも P_2, P_4 とはならない。よって、条件1~3を満たさず、かつ $G = \{P_1, *, P_3, *\}$ (条件4) を満たせば閉眼と推定する。閉眼時における尤度の算出方法も3.4節で詳しく述べる。

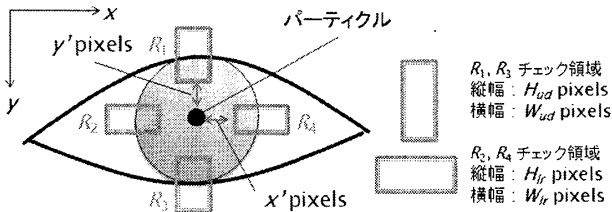


図4 勾配をチェックする領域

$$g_{ij} = \sum_{x,y \in R_j} f_j'(x,y) \quad (9)$$

$$k = \arg \max_j g_{ij} \quad (10)$$

$$D_i = P_k \quad (11)$$

3.4 尤度推定

本研究では、左右の黒目を独立に検出するためにパーティクル群を2種類用いる。黒目の位置は尤度が高いパーティクルの上位 C 個の重心位置とする。また、2つのパーティクル群が同じ目を追跡することを防ぐため、各パーティクル群の重心位置を基準にして、ある一定領域内にもう一方のパーティクルが侵入した場合、そのパーティクルの尤度には0に近い一定の値 z を与える。

3.4.1 開眼時の尤度推定

勾配方向チェックの条件1~3を満たせば、そのパーティクルは開眼の目近傍付近に存在していると推測される。ゆえに、開眼状態と推定されたパーティクルに対して、さらにまぶたを想定した領域における勾配強度を計算し尤度を求める。図5にまぶたの勾配強度を算出する領域を示す。ここで、領域 R_5, R_6 は縦幅 H_{el} pixels, 横幅 W_{el} pixels の大きさであり、パーティクルの位置から x 方向に $\pm u$ pixels, y 方向に $-v$ pixels 離れている。なお、領域 R_5, R_6 では Prewitt オペレータの P_1 のみを使用する。

尤度計算では、各領域 $R_j (j=1, \dots, 4)$ から得られる勾配強度 $f_j'(x, y)$ の上位 k 個の値を用い、それらを $h_j(x, y)$ とする。また、領域 R_5, R_6 から得られる $f_1'(x, y)$ の上位 k 個の値を $h_l(x, y)$ とする。上位 k 個の値の平均値を $\gamma_m (m=1, \dots, 6)$ とすると、 γ_m は式(12)より求められる。また、時刻 t におけるパーティクルの重み $\pi_i^{(n)} (n=1, \dots, N)$ を式(13)より求める。ここで、 $\alpha (\leq 1)$ は勾配チェックで得られた条件に対する重み係数であり、 q は γ_m の比率係数である。なお、条件1 ($G = \{P_1, P_2, P_3, P_4\}$) を満たせば $\alpha = 1.0$, 条件2 ($G = \{P_1, P_2, P_3, *\}$), 条件3 ($G = \{P_1, *, P_3, P_4\}$) を満たせば $\alpha = 0.6$ とする。

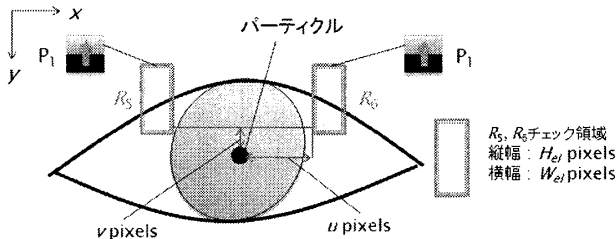


図5 まぶたの勾配強度を計算する領域

$$h_j(x, y) = \begin{cases} f_j'(x, y) [\text{上位 } k \text{ 個}] \\ 0 \quad [\text{その他}] \end{cases} \quad (j=1, \dots, 4)$$

$$h_l(x, y) = \begin{cases} f_1'(x, y) [\text{上位 } k \text{ 個}] \\ 0 \quad [\text{その他}] \end{cases} \quad (l=5, 6)$$

$$\gamma_m = \frac{1}{k} \sum_{x,y \in R_m} h_m(x, y) \quad (m=1, \dots, 6) \quad (12)$$

$$\pi_i^{(n)} = \alpha \left(q \frac{\gamma_1 + \gamma_2 + \gamma_3 + \gamma_4}{4 \times 255} + (1-q) \frac{\gamma_5 \gamma_6}{255^2} \right) \quad (13)$$

3.4.2 閉眼時の尤度推定

閉眼と推定された場合、尤度推定にはテンプレートマッチングを行う。使用するテンプレートは閉眼時における目周辺の平均画像とした。尤度計算では、マッチング度 M に条件1~3より得られる尤度より低くするための重み係数 β を乗じることで求めた ($\pi_i^{(n)} = \beta M$)。ここで、 M はピクセル毎に SAD (Sum of Absolute Difference) を求めることにより得た。

一方、輝度値チェックの条件や勾配方向チェックの条件1~4のいずれも満たさないパーティクルの尤度には0に近い一定の値 z を与える。

3.5 誤検出除去機能

本研究では、誤検出率を減少させるために誤検出除去機能を付加した。これは、誤検出をしているパーティクル群を再生成し、検出されていない黒目を再度探索する機能である。本目検出システムでは、顔をカメラの光軸に対して垂直な平面内で回転する動作において、 $\pm 40^\circ$ まで対応できるようにした。また、検出した目の位置が正検出か誤検出かの判断基準として、顔の回転角度 θ と両目間距離 d を用いて設定した。ここで、両目間距離は検出する顔サイズによって決まる。図6に誤検出除去の流れを示す。まず、生成したパーティクル群の重心位置(目の位置)が安定するまでに約30フレームを要するので、最初の30フレーム間は誤検出除去機能を無効とする。30フレームが経過した後、各パーティクル群の重心位置関係から顔の回転角度 $\theta [^\circ]$ と両目間距離 d [pixels] を求める。図7に顔の回転角度と両目間距離を示す。各パーティクル群の重心位置から θ, d を求める。検出した両目の角度と両目間距離が式(14)の状態を10フレーム連続して満たしたならば、どちらか一方のパーティクル群が誤検出していると判断する。

$$40^\circ < |\theta| < 140^\circ \text{ or } d < d_{min} \text{ or } d > d_{max} \quad (14)$$

ここで、 d_{min} [pixels] は両目間距離の最小値を、 d_{max} [pixels] は両目間距離の最大値である。なお、 d_{min} と d_{max} は複数の正面顔を用いて、許容する顔サイズから求めた。式(14)で示す判断基準を用いて誤検出と判断されたパーティクル群は再生成される。どちらのパーティクル群を再生成するかは、誤検出と判断されるまでの10フレーム間において、それぞれのパーティクル群の尤度の最大値 $\pi_{1,max}, \pi_{2,max}$ の累積値 $\pi_{1,sum}, \pi_{2,sum}$ の大小で判断する。 $\pi_{1,sum} < \pi_{2,sum}$ ならばパーティクル群1を再生成する。一方、 $\pi_{1,sum} > \pi_{2,sum}$ ならばパーティクル群2

を再生成する。再生成したパーティクル群の重心位置が再び安定するまでに約 30 フレームを要するので、パーティクル群を再生成した後の 30 フレーム間も誤検出除去機能は無効とする。

目を誤検出していると判断されたパーティクル群は、もう一方のパーティクル群の重心位置を中心として、図 8 の領域① (緑色) に示す縦幅 $2h$ pixels, 横幅 $2d_{max}$ pixels の領域内にばら撒くようにした。ここで、 W_I , H_I はそれぞれ入力画像の横幅と縦幅であり、単位は pixels である。また、もう一方のパーティクル群の重心位置を中心として、上下左右に $\pm r$ pixels の領域に侵入したパーティクルの尤度には 0 に近い一定の値 z を与える。この範囲を領域② (赤色) で示す。さらに、誤検出と判断されたパーティクル群が再び誤検出していた位置を黒目として検出しないように、誤検出と判断されたパーティクル群が検出していた重心位置を中心として、上下左右に $\pm r$ pixels の領域に侵入したパーティクルの尤度にも z を与える。この範囲を領域③ (紫色) で示す。なお、尤度に z を与える領域②, ③は誤検出と判断されたパーティクル群のみに適応し、誤検出除去機能が再び有効となると同時に消滅する。

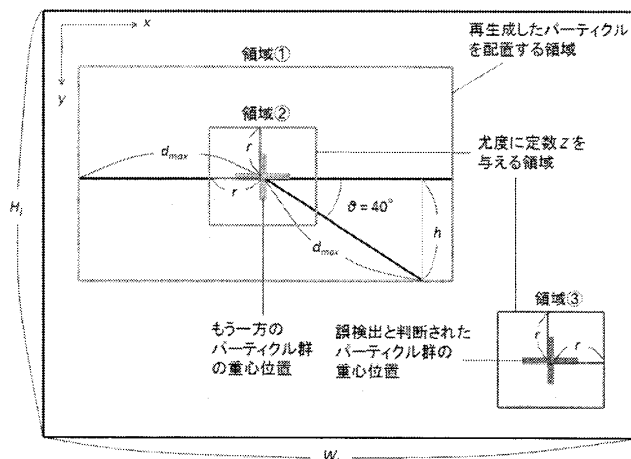


図 8 誤検出したパーティクル群を再生する領域 (緑) と尤度に定数 z を与える領域 (赤, 紫)

4. 目検出実験

顔の回転角度に対する目検出率を求めるために、財団法人ソフピアジャパン顔画像データベース[12]を用いて実験を行った。実験に使用した画像は 120 枚 (1 人あたり 1 枚) であり、開眼のものを使用した。顔画像を画像中心を基準として -15° から 15° の範囲を 5° ずつ回転 (カメラの光軸に対して垂直な平面内での回転) させて目検出評価のための顔画像を作成した。各回転角度において、顔画像を 1 人につき 180 フレーム目検出を行い目検出率を求めた。なお、パーティクルが収束するまでに約 30 フレームを要するので、最初の 30 フレームは評価対象外とした。実験では、 $c = k = 20$, $q = 0.3$, $\beta = 0.05$ とし、各群のパーティクル数は 500 個とした。図 9 に回転角度 0° と -15° における両目検出の様子を示す。ここで、ピンク色と青色の十字は検出した目の位置を示している。図 9 より、十字が黒目の中に存在していることがわかる。

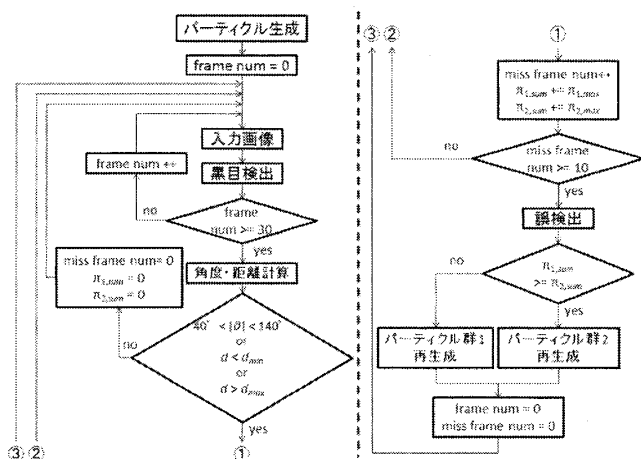


図 6 誤検出除去の流れ

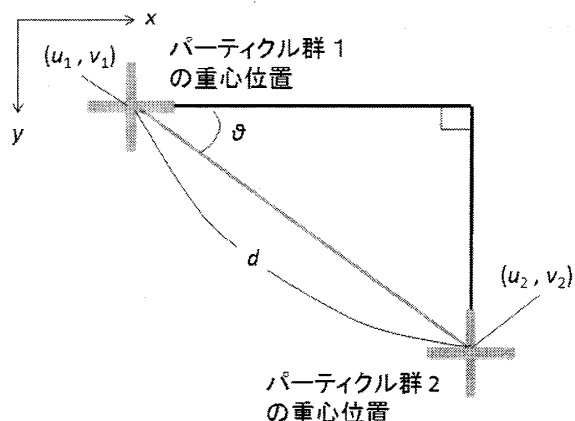


図 7 顔の回転角度と両目間距離



(a) 回転角度 0° (b) 回転角度 -15°
図 9 顔が回転した時の両目検出の様子

次に、誤検出除去機能を用いない場合の目検出率を図 10 に、誤検出除去機能を用いた場合の目検出率を図 11 に示す。また、全回転角度に対する目検出率 (平均 \pm 標準偏差) を表 1 に示す。ここで、図 10, 11 の縦軸は両目、片目の正検出率もしくは誤検出率、横軸は顔の回転角度である。◆は両目とも正検出できた割合、■は片目のみを正検出できた割合、▲は両目とも誤検出した割合を示す。図 10, 11 より、両目検出率がどの回転角度においても 90% 以上であることがわかる。また、表 1 より、顔の回転角度に対する目検出率のばらつきが少ないことがわかる。このことから、本手法が顔の回転角度に影響されることがなく、安定して黒目を検出できることが示された。また、誤検出除去機能を付加することで、付加しない場

合に比べて平均両目検出率が 92.9%から 95.3%となり、2.4%上昇したことがわかる。これより、誤検出除去機能が有効であることも示された。

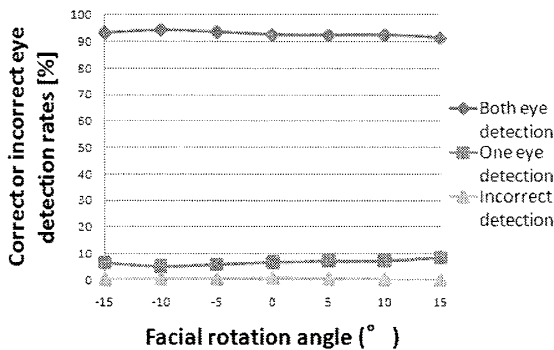


図 10 顔の回転角度に対する両目検出率と誤検出率 (誤検出除去機能なし)

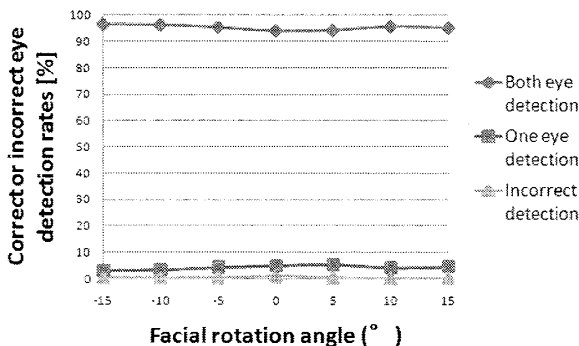


図 11 顔の回転角度に対する黒目検出率と誤検出率 (誤検出除去機能あり)

表 1 全回転角度に対する目検出率

	誤検出除去機能なし [%]	誤検出除去機能あり [%]
両目検出率	92.9±0.9	95.3±0.9
片目検出率	6.8±1.0	4.3±0.7
誤検出率	0.3±0.2	0.4±0.3

次に、誤検出除去機能を付加した場合において、眼鏡なし (96 人) と眼鏡あり (24 人) に対する目検出率を表 2 に示す。表 2 より、「眼鏡あり」に対する両目検出率が「眼鏡なし」に対する両目検出率より低いことがわかる。この原因として、眼鏡のフレームを誤検出する割合が多いことが挙げられる。

表 2 全回転角度において眼鏡なし (96 人) と眼鏡あり (24 人) に対する目検出率

	誤検出除去機能あり	
	眼鏡なし [%]	眼鏡あり [%]
両目検出率	97.4±1.1	86.9±2.0
片目検出率	2.4±1.0	12.1±2.3
誤検出率	0.2±0.2	1.0±0.7

5. まとめ

本研究ではエッジ強度とパーティクルフィルタを用いた黒目検出法を提案した。提案した黒目検出法は、顔の回転角度変化に対して安定した高い目検出率を維持しており、顔の回転変化に頑健であることが示された。

今後の課題としては、首を左右に振るなどの顔向きと照明変化に対する性能評価を行う必要がある。また、眼鏡のフレームに影響されないような工夫が必要である。さらに、計算量を削減するために、パーティクル数を少なくし、かつ高い目検出率を維持できるような工夫が必要である。

謝辞

本論文で使用した顔画像データは、財団法人ソフトピアジャパンから使用承諾を受けたものであり、権利者に無断で複写、利用、配布等を行うことは禁じられています。

参考文献

- [1]比嘉広樹, 国吉真史, 堂上高司, 西原賢, “眼球運動を用いたヒューマンインタフェースの検討”, 信学技報, MBE2006-67, pp.5-8, 2006.
- [2]大野健彦, 武川直樹, 吉川厚, “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”, 情処学論, Vol.44, No.4, pp.1136-1149, 2003.
- [3]T. Nishimura, M. Nakashige, T. Akashi, Y. Wakasa, and K. Tanaka, “Eye interface for physically impaired people by genetic eye tracking”, Proc. SICE Annual Conference, pp.828-833, 2007.
- [4]Y. Ebisawa, “Improved video-based eye-gaze detection method”, IEEE Trans. Instrumentation and Measurement, Vol.47, No.4, pp.948-955, 1998.
- [5]A. L. Yuille, P.W. Hallinan, and D. S. Cohen, “Feature extraction from faces using deformable templates”, IJCV, Vol.8, No.2, pp.99-111, 1992.
- [6]H. Wu, Y. Kitagawa, T. Wada, T. Kato, and Q. Chen, “Tracking iris contour with a 3D eye-model for gaze estimation”, Asian Conf. Computer Vision, Part I, LNCS 4843, pp.688-697, 2007.
- [7]A. Kobayashi, J. Satake, T. Hirayama, H. Kawashima, and T. Matsuyama, “Person-independent face tracking based on dynamic AAM selection”, Proc. FG2008, pp.1-8, 2008.
- [8]高谷学, 滝口哲也, 有木康雄, “過学習を考慮した AAM パラメータの選択と回帰分析による顔・視線方向同時推定”, 画像の理解・認識シンポジウム MIRU2009, IS1-60, pp.769-776, 2009.
- [9]O. Sawetanusorn, Y. Senda, S. Kawato, N. Kawashima, and T. Matsuyama, “Detection of face representative using newly proposed filter”, J. Signal Processing, Vol.8, No.2, pp.137-145, 2004.
- [10] M. Isard and A. Blake, “CONDENSATION-Conditional density propagation for visual tracking”, Int. J. Computer Vision, Vol.29, No.1, pp.5-28, 1998.
- [11] 樋口知之, “粒子フィルタ”, 電子情報通信学会誌, Vol.88, No.12, pp.989-994, 2005.
- [12] 財団法人ソフトピアジャパン, “ソフトピアジャパン 顔画像データベースの貸し出し案内”, <http://www.softopia.or.jp/rd/facedb.html>.