

顔の向きに頑健なパーティクルフィルタと
エッジ方向特徴量を用いた目検出法
Eye Detection Robust to Face Orientation Changes Using
Particle Filter and Edge Directional Features

浅野 誠之[†]
Masayuki Asano

高野 博史[†]
Hironobu Takano

中村 清実[†]
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]~[3]。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、ALS（筋萎縮性側索硬化症）や脳卒中の患者あるいは手足が不自由な障害者に需要が見込まれる。このような障害者が用いる目入力装置では、カメラで撮像して得られた顔画像から目の位置を検出し、検出した目の位置情報をディスプレイ上のカーソルの位置に対応させることで操作を行う。ここで、画像中における目の位置を推定する一般的な手法として、顔画像を用いた画像処理が挙げられる。この手法では、目の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、目の位置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、テンプレートマッチングを用いるもの[4]、AAM（Active Appearance Model）を用いるもの[5][6]、SSR（Six-Segmented Rectangular）フィルタを用いるもの[7]、エッジ特徴量を用いるもの[8]、目の形状をした楕円モデルを用いるものなどがある[9]。

Yuilleらの手法は、まぶたや目から構成される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[4]。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ために多くの計算量を必要とするため、リアルタイム処理が困難である。Kobayashiら[5]とXuら[6]の手法は、顔器官を表現するための特徴点の位置や輝度値を用いてモデルを構築することにより目を検出する。これらの手法は、学習した人物でないと高精度な目検出が困難である。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[7]。Pengらの手法は、エッジ特徴量を用いることで目検出している[8]。上述した手法は、顔の向きが変化すると目検出精度が低下する問題がある。Hansenらの手法は、EMアルゴリズムと虹彩の形状をした楕円モデルを用いることでこの問題を克服している[9]。しかし、この手法は一度誤検出が生じると再び目を検出することが困難である。

本研究では、顔の向き（3次元的变化）に頑健であり、誤検出が生じて再び目を高速に検出できるようにパーティクルフィルタとエッジ方向特徴量を用いた目検出法を提案する。本論文では、顔のroll, yaw, pitch方向の変化に対する目検出精度を調査し、提案手法が顔向きの変化に頑健であることを示す。

2. パーティクルフィルタ

SIFTは対象物体のスケール変化やノイズ、照明変化に対して頑健であることが知られている[10]。しかし、この手法は計算量が多いためリアルタイム処理が難しい。本研究では、ノイズやオクルージョン、顔向きの変化に頑健かつ安定して目を検出するために、パーティクルフィルタを用いる[11]。次に、パーティクルフィルタの概要について述べる。

2.1 尤度計算

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 $p(\mathbf{x}_t | \mathbf{x}_{t-1})$ で表される。本研究では、状態ベクトルである \mathbf{x}_t を目中心位置とその移動速度、観測結果である \mathbf{y}_t を画像から得られる目の特徴量としている。

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

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対象物体（本研究では目）を追跡する。

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

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

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 $s_{00} \sim p_0(\mathbf{x})$ を生成. ただし, $p_0(\mathbf{x})$ 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. l 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

$$s_{t|t-1}^{(n)} = F_t(s_{t-1|t-1}^{(n)}, \mathbf{v}_t^{(n)}) \quad (5)$$

を用いて次の状態 $s_{t|t-1}^{(n)}$ を算出.

- (b) 事前分布に従うパーティクル $s_{t|t-1}^{(n)}$ について, 重み $\pi_t^{(n)}$ を下式を用いて算出.

$$\pi_t^{(n)} = \frac{p(\mathbf{y}_t | \mathbf{x}_t = s_{t|t-1}^{(n)})}{\sum_{n=1}^N p(\mathbf{y}_t | \mathbf{x}_t = s_{t|t-1}^{(n)})} \quad (6)$$

ここで, $p(\mathbf{y}_t | \mathbf{x}_t = s_{t|t-1}^{(n)})$ は, 状態 \mathbf{x}_t が $s_{t|t-1}^{(n)}$ であるときに観測値 \mathbf{y}_t を得る確率である.

- (c) $s_{t|t-1}^{(n)}$ 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. エッジ方向特徴量を用いた目検出法

3.1 目検出の流れ

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. 次に, 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 (輝度値チェック). ここで, 条件を満たしたパーティクルに対して, さらに 4 領域の勾配方向のチェックを行う (4 領域勾配チェック). その後, それぞれの条件に合った尤度計算を行い, パーティクルのリサンプリングを行う. 目の位置は, 尤度が高い (重みが高い) パーティクルの上位 c 個の平均位置とする. 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目の検出候補点を削減するために行う.

3.2 輝度値チェック

輝度値チェックでは, 目より低い輝度値を示す領域が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目の上端と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. 式(7)を満たしたパーティクルについては 4 領域勾配チェックを行い, 満たさないパーティクルについては, 0 に近い一定の値 z を与える.

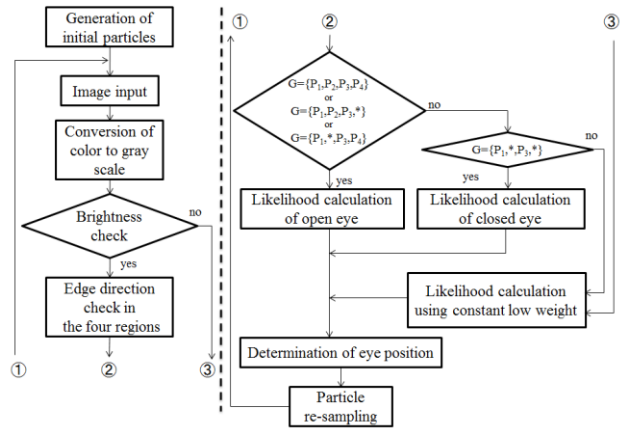


図 1 目検出の流れ

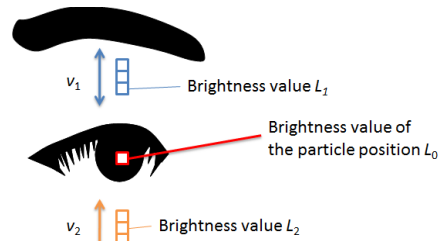


図 2 輝度値チェック領域

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

3.3 尤度計算

本研究では, 左右の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. 目の位置 (X_t, Y_t) は, 各パーティクル群に対して重み ($\pi_t^{(n)}$) が大きい上位 c 個のパーティクルの位置座標を用いて式(8), (9)より決定する.

$$X_t = \frac{1}{c} \sum_{i=1}^c P_{x,t}(i) \quad (8)$$

$$Y_t = \frac{1}{c} \sum_{i=1}^c P_{y,t}(i) \quad (9)$$

ここで, $P_{x,t}(i)$, $P_{y,t}(i) (i=1, \dots, c)$ は, $\pi_t^{(n)}$ を大きい順に選んだ上位 c 個のパーティクルの x , y 座標値である. 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目検出位置を基準にして, 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.

3.3.1 開眼時の尤度計算

尤度を計算するために, 図 3 に示す 4 種類の Prewitt オペレータ $\{P_1$:下から上, P_2 :右から左, P_3 :上から下, P_4 :左から右) を使用する. なお, 算出したエッジ強度に定数 amp を乗じることで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(10)に示すように amp を乗じてエッジ強度 $f'_j(x, y)$ を求める.

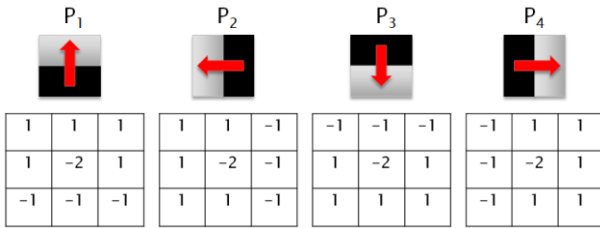


図3 4種類のPrewittオペレータ

$$f_j'(x, y) = amp \times f_j(x, y) \quad (10)$$

図4に勾配強度を計算する領域を示す。ここで、勾配強度とは、ある一定領域内におけるエッジ強度の累積値とする。輝度値チェックの条件を満たしたパーティクルについては、図4に示す $R_1 \sim R_4$ の4領域について勾配強度が最大となる方向を用いて4領域勾配チェックを行う。その後、パーティクルの位置を基準とした領域 $R_1 \sim R_6$ 内の勾配強度を計算することで目を検出する。ここで、領域 $R_1 \sim R_6$ は本目検出システムが許容する目(顔)の大きさからあらかじめ決定した。尤度計算には、各領域 $R_j (j=1, \dots, 4)$ において求めたエッジ強度 $f_j'(x, y)$ の上位 k 個の値と領域 R_5, R_6 で求めた $f_1'(x, y)$ の上位 k 個の値を用いる。式(11)および(12)に示すように、 R_j における $f_j'(x, y)$ の上位 k 個の値を $h_j(x, y)$ 、領域 R_5, R_6 における $f_1'(x, y)$ の上位 k 個の値を $h_l(x, y)$ とする。

$$h_j(x, y) = \begin{cases} f_j'(x, y)[\text{上位 } k \text{ 個}] & (j=1, \dots, 4) \\ 0 & [\text{その他}] \end{cases} \quad (11)$$

$$h_l(x, y) = \begin{cases} f_1'(x, y)[\text{上位 } k \text{ 個}] & (l=5, 6) \\ 0 & [\text{その他}] \end{cases} \quad (12)$$

また、上位 k 個の値の平均値を $\gamma_m (m=1, \dots, 6)$ とし、式(13)によって求める。

$$\gamma_m = \frac{1}{k} \sum_{x, y \in R_m} h_m(x, y) \quad (m=1, \dots, 6) \quad (13)$$

時刻 t におけるパーティクルの重み $\pi_i^{(n)} (n=1, \dots, N)$ は、式(14)に示すように γ_m を用いて算出される。

$$\pi_i^{(n)} = \alpha \left(q \frac{\gamma_1 + \gamma_2 + \gamma_3 + \gamma_4}{4 \times 255} + (1-q) \frac{\gamma_5 \gamma_6}{255^2} \right) \quad (14)$$

ここで、 $\alpha (\leq 1)$ は4領域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、 α の値は4領域勾配チェックで得られる領域 $R_1 \sim R_4$ における勾配強度が最大となる組み合わせによって決定する。次に、4領域勾配チェックにより α を求める方法について述べる。

式(15)に示すように、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 g_{ij} を下式で求める。

$$g_{ij} = \sum_{x, y \in R_i} f_j'(x, y) \quad (15)$$

式(16), (17)のように、領域 R_i 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き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、勾配強度の組み合わせが $G = \{P_1, P_2, P_3, *\}$ (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き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。本研究では、条件1 ($G = \{P_1, P_2, P_3, P_4\}$) の時 $\alpha = 1.0$ 、条件2 ($G = \{P_1, P_2, P_3, *\}$)、条件3 ($G = \{P_1, *, P_3, P_4\}$) では $\alpha = 0.6$ とする。

$$\varepsilon = \arg \max_j g_{ij} \quad (16)$$

$$D_i = P_\varepsilon \quad (17)$$

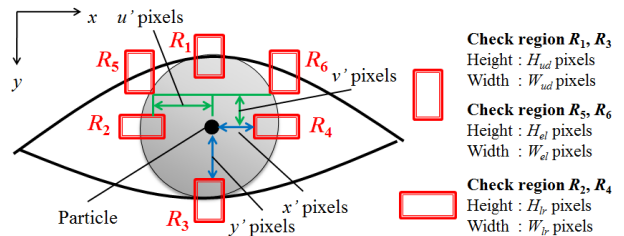


図4 勾配強度を計算する領域

3.3.2 閉眼時の尤度計算

検出対象の目が閉眼状態の場合、領域 R_2, R_4 において、最大となる勾配強度がそれぞれ P_2 や P_4 とは限らない。4領域勾配チェックにおいて、勾配強度が最大となる組み合わせが $G = \{P_1, *, P_3, *\}$ (条件4) を満たした場合、そのパーティクルは閉眼の目近傍に存在すると推定される。この条件の場合、テンプレートマッチングを用いて尤度計算を行う。ここで、テンプレートは閉眼画像の平均画像を用いる。尤度計算では、マッチング度 M を用いて $\pi_i^{(n)} = \beta M$ とした。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。なお、 β は条件1~3より得られる尤度より低くするための重み係数である。

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 ε を与える。

3.3.3 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、パーティクルによって検出した目が正検出か誤検出であるかを推定し、誤検出をしている場合は、そのパーティクル群を再生成し、検出されていない目を再度探索する機能である。本目検出システムでは、顔の回転(roll角度)に対して、 $\pm 40^\circ$ 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5に誤検出除去の流れを示す。

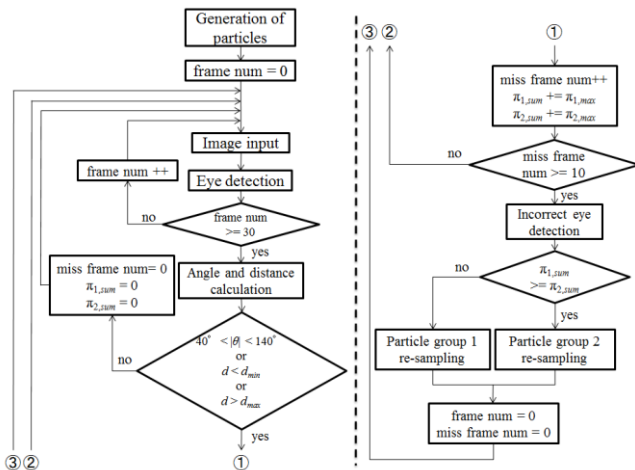


図5 誤検出除去の流れ

まず、生成したパーティクル群の中心位置（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、各パーティクル群が検出した目の位置関係から顔の回転角度 θ [°]と両目間距離 d [pixels]を求める。図6に両目の角度と距離の定義を示す。

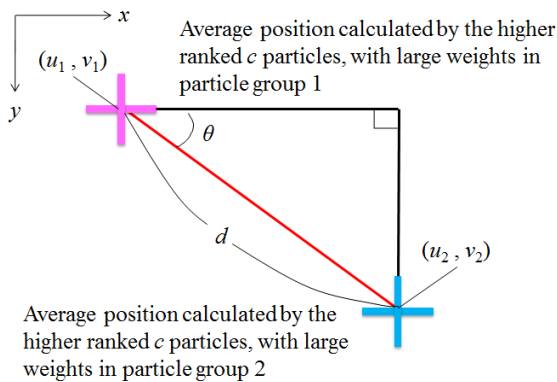


図6 両目の角度と距離の定義

検出した両目の角度と両目間距離が式(18)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。

$$40^\circ < |\theta| < 140^\circ \text{ or } d < d_{min} \text{ or } d > d_{max} \quad (18)$$

ここで、 d_{min} [pixels]は両目間距離の最小値であり、 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(18)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によって検出される目の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フレーム間は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再生成され、画面全体にばら撒かれる。誤検出と判断されたパーティクル群がもう一方のパーティクル群の目検出位置を中心とした半径 r [pixels]内に侵入した場合、侵入したパーティクル群の尤度には0に近い一定の値 z を与える。この範囲を図7にRegion 1として示す。誤検出と判断されたパーティクル群が誤検出していた位置を再び目として検出しないように、誤検出した位置を中心として、半径 r [pixels]内に侵入したパーティクルの尤度にも z を与える。この範囲を図7にRegion 2として示す。Region 1, 2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。この機能を用いることによって、検出されていない目を再度探索できるようにした。

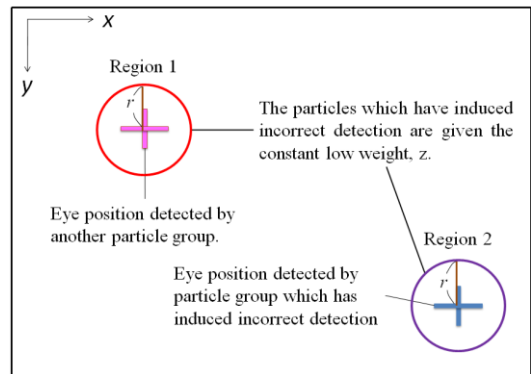


図7 誤検出したパーティクル群の尤度に定数zを与える領域

4. 目検出実験

本研究で提案した目検出法の性能を評価するために、財団法人ソフトピアジャパン顔画像データベース[12]を用いて顔のroll, yaw, pitch方向への回転変化に対する目検出精度を調べた。なお、本研究では検出した目の位置が瞳孔もしくは虹彩内にある場合を正検出とする。実験に用いた顔画像はそれぞれ300枚であり、開眼のものを用いている。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った。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、150フレームを用いて目検出率を求めた。実験では、 $c=20$ 、 $k=20$ 、 $q=0.3$ 、 $\beta=0.05$ とし、各群のパーティクル数は500個とした。図8に各回転角度において目を検出している様子を示す。ここで、十字は検出した目の位置を示している。図8より、十字が目の中に存在していることがわかる。

次に、各回転角度における目検出率を図9~11に示す。表1~3はそれぞれ全回転角度に対する平均目検出率、平均誤検出率とそれらの標準偏差を示している。ここで、図9~11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9~11より、roll, yaw, pitchにおける回転変化において、誤検出除去機能を用いた場合の平均両目検出率は90%以上であることがわかる。roll方向については顔の回転変化に対して目検出率に影響はないが、yaw方向に対しては $\pm 30^\circ$ 、pitch方向に対しては -15° において両目検出率が低下した。

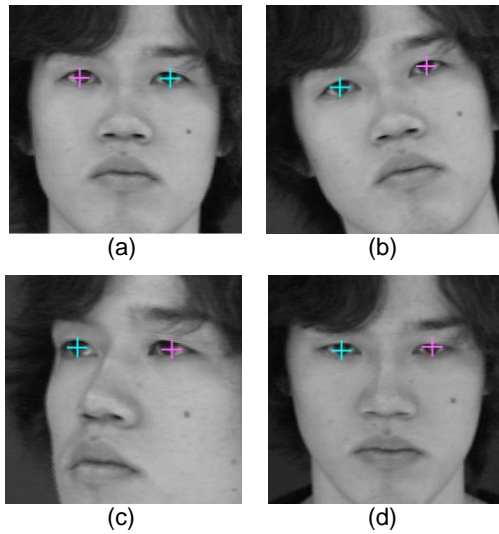


図8 各回転角度において目を検出している様子
(a) roll 0°, (b) roll -15°, (c) yaw -30°, (d) pitch 15°

これは、両目とも誤検出する割合は変化せず、片目検出率が上昇していることから、顔の向きが変化することによりどちらか一方の目を誤検出しやすくなっていることを示している。誤検出している箇所は主に、yaw 方向については眉毛、pitch 方向については鼻の穴であった。次に、表 1~3 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。誤検出除去機能を用いることによって、顔の回転変化 roll, yaw, pitch における平均両目検出率がそれぞれ 2.8%, 4.3%, 5.3% 上昇した。このことから、誤検出除去機能が有効であることがわかる。

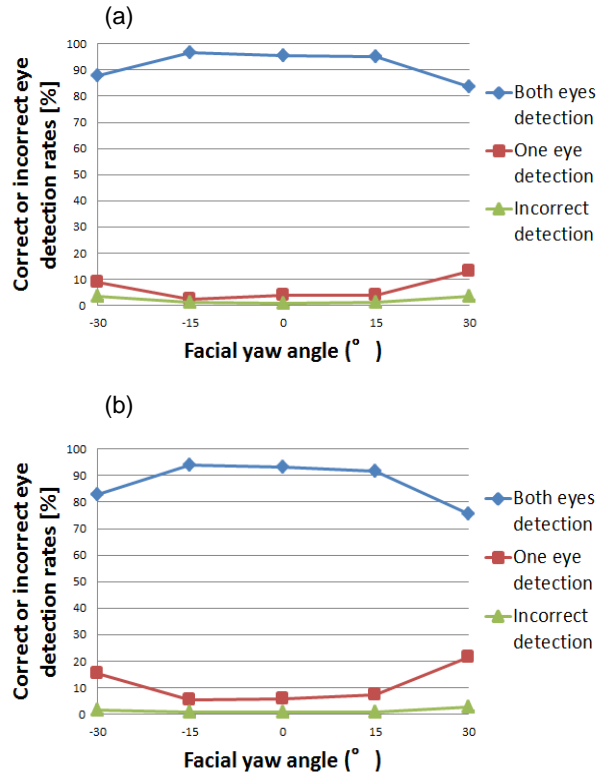


図10 yaw角における目検出率と誤検出率
(a) 誤検出除去機能あり, (b) 誤検出除去機能なし

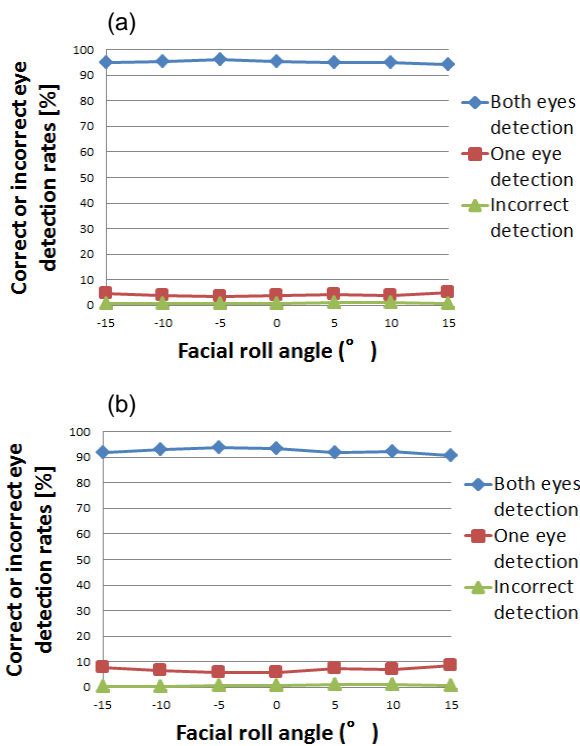


図9 roll角における目検出率と誤検出率
(a) 誤検出除去機能あり, (b) 誤検出除去機能なし

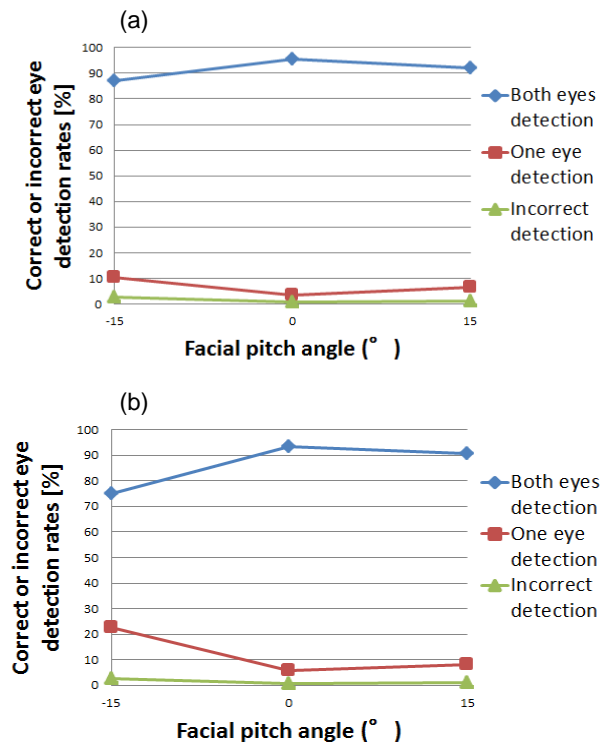


図11 pitch角における目検出率と誤検出率
(a) 誤検出除去機能あり, (b) 誤検出除去機能なし

表1 全 roll 角に対する平均目検出率, 平均誤検出率とそれらの標準偏差

	誤検出除去 機能あり [%]	誤検出除去 機能なし [%]
両目検出率	95.2±0.5	92.4±1.0
片目検出率	4.0±0.5	6.9±1.0
誤検出率	0.8±0.1	0.7±0.2

表2 全 yaw 角に対する平均目検出率, 平均誤検出率とそれらの標準偏差

	誤検出除去 機能あり [%]	誤検出除去 機能なし [%]
両目検出率	91.8±5.2	87.5±7.1
片目検出率	6.3±4.1	11.1±6.4
誤検出率	1.9±1.2	1.4±0.8

表2 全 pitch 角に対する平均目検出率, 平均誤検出率とそれらの標準偏差

	誤検出除去 機能あり [%]	誤検出除去 機能なし [%]
両目検出率	91.6±3.6	86.3±8.2
片目検出率	6.9±2.8	12.3±7.4
誤検出率	1.5±0.8	1.4±0.7

5. まとめ

本研究ではパーティクルフィルタとエッジ勾配強度を用いた顔の向きに頑健な目検出法を提案した。実験結果より、誤検出除去機能を用いることによって各回転角度に対して 90%以上の平均両目検出率が得られた。また、平均両目検出率が 2.8%~5.3%上昇し、この機能が有効であることが示された。さらに、本手法が顔の向きに影響されることなく安定した目検出率が得られることが示された。

今後の課題としては、動画像を用いて性能評価する必要がある。また、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]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).
- [2]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).
- [3]Y. Ebisawa, “Improved video-based eye-gaze detection method”, IEEE Trans. Instrumentation and Measurement, Vol.47, No.4, pp.948-955 (1998).
- [4]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).
- [5]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).

- [6]G. Xu, Y. Wang, J. Li, X. Zhou, “Real time detection of eye corners and iris center from images acquired by usual camera”, Int. J. Intelligent Engineering & Systems, Vol.3, No.1, pp.25-32 (2010).
- [7]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).
- [8]K. Peng, L. Che, S. Ruan, and G. Kukharev, “A robust algorithm for eye detection on gray intensity face without spectacles”, J. Computer Science & Technology, Vol.5, No.3, pp.127-132 (2005).
- [9]D. W. Hansen and A.E.C. Pece, “Eye tracking in the wild”, Computer Vision and Image Understanding, Vol.98, No.1, pp.182-210 (2005).
- [10] D. G. Lowe, “Distinctive image features from scale-invariant keypoints”, Int. J. Computer Vision, Vol.60, No.2, pp.91-110 (2004).
- [11] M. Isard and A. Blake, “CONDENSATION-Conditional density propagation for visual tracking”, Int. J. Computer Vision, Vol.29, No.1, pp.5-28 (1998).
- [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>.