

複数枚写真を用いた3次元顔モデルの構成

竹之内 航*・山下 英生**

(平成21年10月29日受理)

Reconstruction of 3D Human Face Model Using Plural Photographs

Wataru TAKENOUCHI and Hideo YAMASHITA

(Received Oct. 29, 2009)

Abstract

If it is easy to construct the three-dimensional model of each person's face, we can realize the face database of person, and many application fields using human faces like animation will spread.

In this paper, a method for making a three-dimensional model by using two face pictures taken from two different directions and using the epipolar geometry is described. In order to reconstruct a free-form surface like the shape of human face, a circle distribution image composed of a lot of circles is irradiated on a face, and we reconstruct the three-dimensional face shape model by using the center points of circles as characteristic points. We propose a method for extracting central coordinates of a lot of circles automatically. Next, the restoration accuracy of the proposed technique is verified, and the result of application to a face model is described.

Key Words: human face reconstruction, epipolar geometry, circle distribution image

1 はじめに

近年、コンピュータビジョンの分野では、3次元対象物(ビルなどの建造物や樹木など)の写真から、その形状の復元を行う研究が行われている。また、映画やゲームなどの映像製作においては、人間の顔を表現した作品が数多く作られており、3次元顔モデルの需要も高まってきている。しかし、3次元モデルを作る作業は一般的に簡単ではなく、膨大な時間やコストを必要とする。これらのことを踏まえ、もし異なる2つの視点から撮影された個人の顔写真からその個人の3次元顔モデルを生成することができれば、膨大な時間をかけずに、コストも抑えて、比較的簡単に各個人の顔モデルを作成することが可能となる。これにより、各個人の顔のデータベース化が実現でき、犯罪者データベースの3次元化、顔の骨格に関する種々の研究、アニメーション

の作成など数多くの応用分野への活用が広がる。

複数枚の画像から物体までの距離を算出するためには、カメラと物体との3次元空間での位置関係を求める必要がある。コンピュータビジョンの分野において、複数枚の画像から物体の3次元情報を取得する3次元復元手法に関する研究が数多く行われている^{1)~3)}。3次元復元手法の代表的な手法には、ステレオ視を用いた手法²⁾、エピポーラ幾何を用いた手法¹⁾などがある。本研究では、カメラ位置、姿勢、カメラの内部パラメータと複数枚の画像間の対応を示す特徴点から、特徴点の3次元空間中での点の位置を算出できるエピポーラ幾何を用いて顔の3次元形状の復元を行う。

本稿では、まずエピポーラ幾何を用いた複数枚写真から3次元復元する方法について説明する。次に、顔面の形状のような自由曲面を復元するために多数の小円群から構成

* 広島工業大学大学院工学系研究科情報システム科学専攻

** 広島工業大学情報学部知的情報システム学科

される小円分布映像を照射し、その円の中心を特徴点とし、その中心座標を自動抽出する方法について説明する。次に提案手法の復元精度について検証し、最後に、顔モデルに適用した結果について述べる。

2 エピポーラ幾何¹⁾を用いた3次元復元手法

エピポーラ幾何とは、図1に示すように3次元空間中の点Pを2つの視点C1, C2から見ているときの相対的なカメラ位置や姿勢の情報を記述する幾何である。図1のように、2つのカメラから3次元空間中の同じ点Pを見る場合、以下に述べるエピポーラ平面、エピポーラ線、エピポールが構成される。エピポーラ平面は、3次元空間中の点Pと2つのカメラ位置C1, C2から構成される平面である。エピポーラ線は、エピポーラ平面とそれぞれのカメラの投影面1, 2との交線である。また、エピポールは、2つのカメラ位置C1, C2を結ぶ直線と2つの投影面1, 2との交点であり、また一方の投影面に他方のカメラ位置を投影した点でもある。また、エピポーラ線はエピポールと特徴点を通る直線となる。ここで、特徴点とは3次元空間中の点を投影した点である。

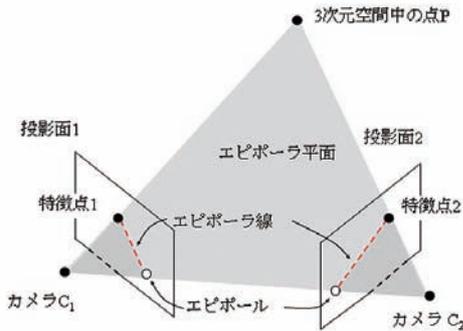


図1 エピポーラ幾何

エピポーラ幾何は以下のような関係を持つ。

一方のカメラに対する他方のカメラの位置と姿勢に関する運動は、回転行列 \mathbf{R} と並進ベクトル \mathbf{t} によって表現される。ここで同じ3次元空間中の点を投影したカメラ座標系での特徴点1の座標値 \mathbf{x} と特徴点2の座標値 \mathbf{x}' の間には(1)式の関係が成り立つ。なお、 \mathbf{x} や \mathbf{x}' の上付き \sim 記号は同次座標を表す。

$$\tilde{\mathbf{x}}'^T \mathbf{E} \tilde{\mathbf{x}} = 0 \quad (1)$$

ここで \mathbf{E} は基本行列と呼ばれ、(2)式のようにカメラの回転行列 \mathbf{R} と並進ベクトル \mathbf{t} から構成される行列となる。

$$\mathbf{E} = [\mathbf{t}]_x \mathbf{R} \quad (2)$$

ここで $[\mathbf{t}]_x$ は次のような歪対称行列で表される。

$$[\mathbf{t}]_x = \begin{bmatrix} 0 & -t_3 & t_2 \\ t_3 & 0 & -t_1 \\ -t_2 & t_1 & 0 \end{bmatrix}$$

ここで t_1, t_2, t_3 は並進ベクトル \mathbf{t} の x, y, z 成分を表している。

同じ3次元空間中の点を投影している画像座標系における特徴点1の座標値 \mathbf{m} と特徴点2の座標値 \mathbf{m}' は、次の(3)式を満足する。

$$\tilde{\mathbf{m}}^T \mathbf{F} \tilde{\mathbf{m}}' = 0 \quad (3)$$

上記の \mathbf{F} は基礎行列 \mathbf{F} と呼ばれる。基礎行列 \mathbf{F} と基本行列 \mathbf{E} の間には、焦点距離や画像中心の座標などのカメラの内部パラメータから構成されるカメラの内部パラメータ行列 \mathbf{A} を用いて(4)式のような関係が成り立つ。

$$\mathbf{F} = \mathbf{A}^{-T} \mathbf{E} \mathbf{A}^{-1} \quad (4)$$

したがって、基礎行列 \mathbf{F} は、カメラの内部パラメータとカメラの運動に関する情報から構成される行列となる。基礎行列 \mathbf{F} は 3×3 の行列で要素が9個あるが、(3)式で示すように定数倍の不定性がある。したがって、画像座標系での特徴点1の座標値 \mathbf{m} と特徴点2の座標値 \mathbf{m}' の対応する組が8組以上与えられれば、(3)式より基礎行列 \mathbf{F} を求めることができる。求めた基礎行列 \mathbf{F} からカメラの運動および内部パラメータが算出でき、3次元空間中の点 X と2つのカメラ位置 $\mathbf{C}_1, \mathbf{C}_2$ から三角形が構成されることを利用して、3次元空間中の点の3次元座標を算出できる。

なお、復元する際にはあらかじめ使用するカメラに対してカメラ校正を行い、カメラの内部パラメータを算出する。その後、本節で述べた方法で3次元座標を求める。

3 本手法の処理手順

本手法では、復元対象をカメラで2方向から撮影し、その2枚の画像を用いて3次元復元を行う。その際、2節で述べたように、2枚の画像に共に存在する特徴点が必要となる。また、顔面形状のように自由曲面を復元するには、曲面上に多数の特徴点を配置する必要がある。ここでは、多数の小円を規則正しく配置した映像を顔面に照射し、その小円の中心を特徴点とすることにした。2枚の画像に撮影された小円群の中心座標値を自動抽出する手法について以下に述べる。

3.1 特徴点

前述したように、本手法ではあらかじめ小円が規則正しく分布した映像を復元対象に照射する。照射する映像は図2に示すような映像である。

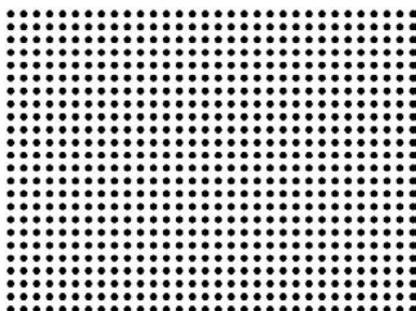


図2 照射する映像

このような小円が分布した映像を復元対象に照射し、撮影を行う。円筒状のモデルに映像を照射した様子を図3に示す。

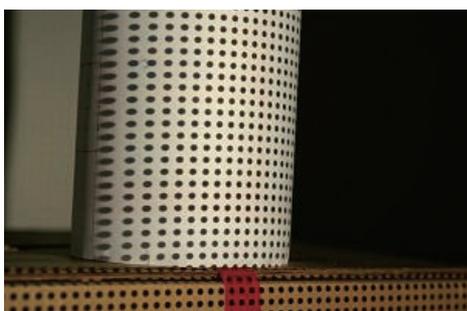


図3 映像を照射した円筒モデル

図3に映っている小円の中心を自動抽出し、これを特徴点（復元点）として復元を行う。

3.2 小円の中心座標値を抽出するアルゴリズム

次に小円を自動抽出するアルゴリズムについて説明する。

アルゴリズムの全体イメージと簡単な流れを以下に示す。ここでは画像の横軸をx座標（右方向を正）、縦軸をy座標（下方向を正）として説明する。また、座標軸の原点は画像の左上とする。

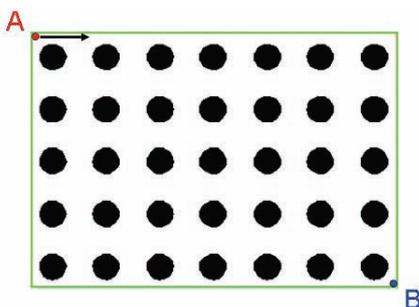


図4 復元対象画像のイメージ図

処理①：小円が分布した映像を照射した復元対象を撮影し、その画像に2値化処理を施した画像（たとえば図4）を読み込む

処理②：対象の復元する領域（図4の緑枠）を指定するため、図4に赤と青で示すA点、B点を指定する

処理③：図4の矢印のようにA点から右方向へ探索を開始する

処理④：現在の画素と右隣の画素の色を比較し、白から黒に変化する画素を探索する

もし、その画素が見つからず、緑枠領域の右端に達すると、y座標値を1増やして再度左端から右方向へと探索する

処理⑤：画素が白から黒に変化する点が見つかったら、黒の小円の外周を追跡し、一周する

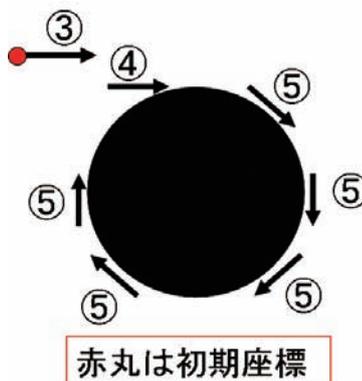
処理⑥：外周のxおよびy座標の最大値、最小値から小円の中心座標（X, Y）を次式により求める

$$X = (x \text{ の最大値} + x \text{ の最小値}) / 2$$

$$Y = (y \text{ の最大値} + y \text{ の最小値}) / 2$$

処理⑦：処理④、⑤、⑥を繰り返し行い、指定したB点まで探索し、終了する

処理④は、処理③のように走査しながら画素の色が白から黒に変わる画素を探す。その画素が見つかったら処理⑤の円の外周を見つける処理に移行する。この様子を図5に示す。



赤丸は初期座標

図5 処理③、④、⑤のイメージ図

処理④で円の上端部分が見つかるので、そこから時計回りに外周を調べていき、一周して処理④で見つけた画素まで戻ると、処理⑤は終了とする。処理⑤の詳しい内容は付録に示す。

次に、処理⑤で見つけた外周の座標の最大値、最小値から円の中心座標を求める。これが処理⑥になる（図6参照）。

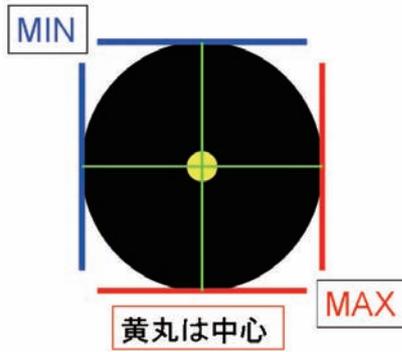


図6 処理⑥のイメージ

以上の処理を繰り返し行い、画像上の指定範囲内にある全ての小円を抽出する。

しかし、このアルゴリズムでは同じ円を多重抽出することになる。これを防止する処理を次節で説明する。

3.3 円の多重抽出を防止する処理

上記のアルゴリズムでは、y座標が小さいときに見つけた小円を再度抽出することになるため、一度見つけた小円はスキップする処理が必要となる。以下に、この処理について説明する。

まず、抽出した各小円には番号を付け、その外周座標を全て配列に格納する。

[小円スキップ終了端画素抽出処理]

探索中に白から黒い画素に変わる画素を見つけると、すでに抽出した小円の外周座標と一致するかどうかの判定を行う。一致する場合、外周座標の中でy座標値が等しいものの中からx座標値が最大の画素を探し、その画素までスキップする(図7に示すようにCからDにスキップする)。

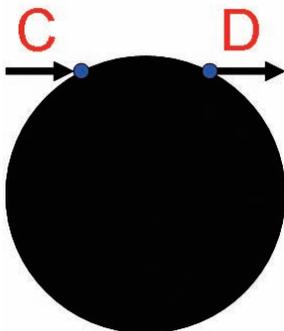


図7 小円スキップ終了端画素抽出処理

なお、上述の小円スキップ終了端画素抽出処理では、すでに抽出した全ての小円の外周座標とのチェックを行うことになり、処理時間が膨大なものになる。そこで、無駄な処理を省くために次のような処理を追加する(図8参照)。

処理①：抽出した全ての小円にフラグを持たせる(フラグ

の初期値は0とする)

処理②：抽出した小円の外周座標のy座標とx座標のそれぞれの最大値と最小値(x_{max} , x_{min} , y_{max} , y_{min})を格納する

処理③：現在、探索している画素のy座標値よりも y_{max} が小さな小円はflag=1とし、flag=1の円は上述した小円スキップ終了端画素抽出処理を行わない

処理④：現在、探索している画素のx座標値と比べて x_{min} が小さく、かつ x_{max} が大きな円を見つけ、その円番号を取得し、小円スキップ終了端画素抽出処理を行う

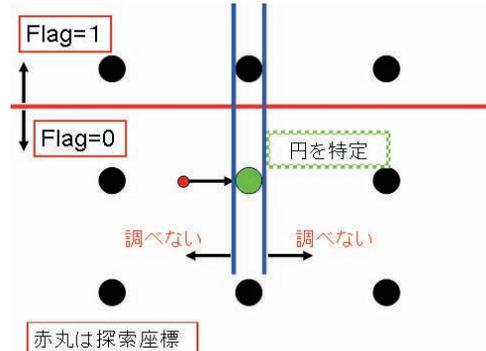


図8 処理時間短縮のための処理

図8において、赤の丸が現在探索している画素とする。矢印方向に探索していき、緑の小円に到達したとき、赤の線より上の小円はy座標の最大値が現在探索しているy座標値よりも小さいのでflag=1の小円となる。次にflag=0の小円で、青の線に挟まれる円(探索している画素のx座標値より、 x_{min} が小さく、かつ x_{max} が大きな小円)を見つけることにより、スキップする円を特定できる。これにより、無駄なくスキップする小円を見つけることができる。このようにして対象の小円を見つけた後に、図7のようにスキップし、探索を行う。

4 球体モデルを用いた誤差検証

次に、本手法による曲面の復元精度について検討する。検証するために、球体モデルを使用し、その曲面の復元を行った。復元するモデルを図9に示す。

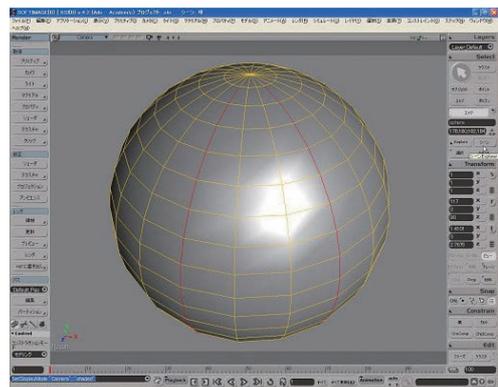
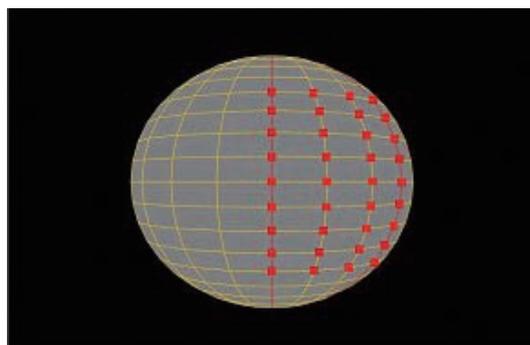
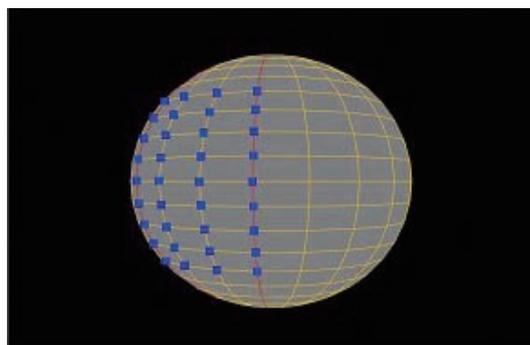


図9 球体モデル

図10が2つのカメラから撮影した画像であり、赤と青の点で示す点が特徴点である。



左画像



右画像

図10 2枚の画像の特徴点

ここでは、2つのカメラ間の角度が復元にどのように影響するかを調べるため、カメラ間の角度を10度、30度、50度と変化させ、それぞれ復元前の球体の半径と復元したモデルの半径の誤差を計算した。その結果を表1に示す。

表1 誤差検証結果

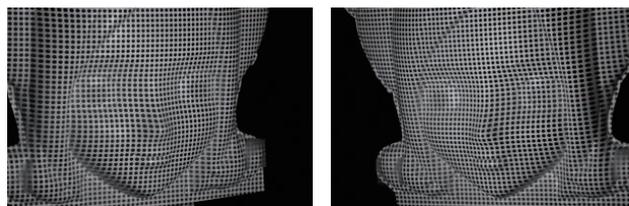
	半径 (真値5)		相対誤差
	最大	最小	
10度	最大	5.046	0.009
	最小	4.976	0.005
30度	最大	5.043	0.008
	最小	4.928	0.014
50度	最大	5.081	0.016
	最小	4.924	0.015

表1に、特徴点36点全ての半径を求め、その中の誤差が最大、最小のものを示している。元のモデルの半径は5であり、表1に、復元したモデルの半径と相対誤差を示している。

この結果から、最大の誤差が1.6%であり、カメラ間の角度はあまり影響しないということがわかった。また、この程度の誤差であれば人が目で見てわからない程度の誤差であるため、本手法が顔モデルに適用可能であると判断した。

5 顔モデルへの適用

次に、実際にお面を使用して顔モデルの復元を行った。復元に使用する2枚の画像を図11に示す。

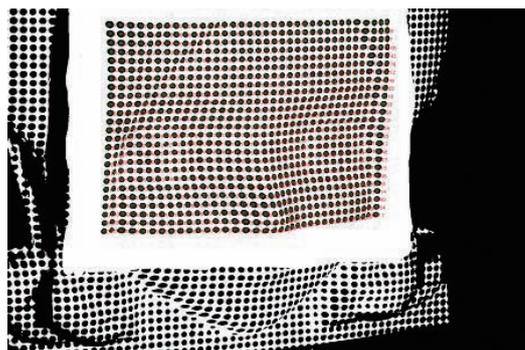


左画像

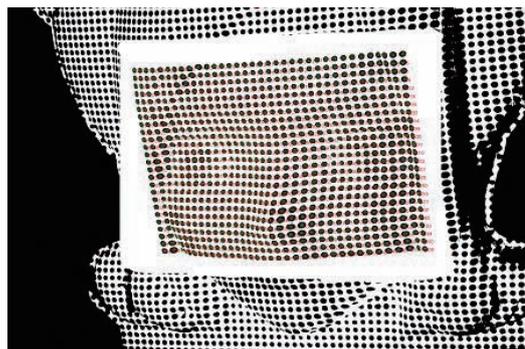
右画像

図11 復元するお面の顔の2枚の写真

次に復元する875点の特徴点を赤点で示したものを図12に示す。



左画像



左画像

図12 復元する特徴点

復元した結果を図13に、図13をレンダリングした結果を図14に示す。この結果から、額や目、鼻が正しく復元できていることが確認できる。このことから本手法が顔モデルに適用可能だと言える。

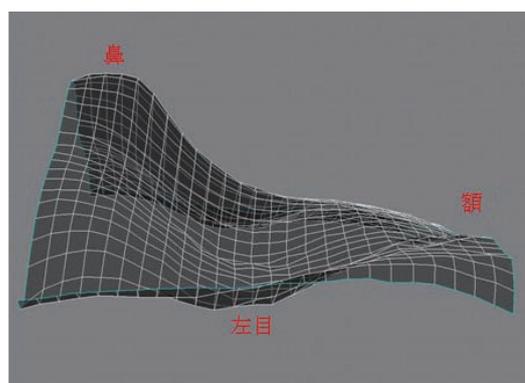


図13 復元結果



図 14 レンダリング結果

6 おわりに

本論文では、エピポーラ幾何を用いて、複数枚写真から3次元復元する方法について述べた。また、顔のような自由曲面の形状を復元する際に、必要となる特徴点の与え方と抽出方法を提案し、その検証と顔モデルへの適用を行った。その結果、カメラ間の角度は復元精度にあまり影響しないということがわかった。また、人が目で見てもわからない程度の誤差であるため、本手法が顔モデルに適用可能であると言える。

今後の課題として顔全体の復元、いろいろなモデルへの適応などがあげられる。

文 献

- 1) 徐 剛 『写真から作る3次元CG』 近代科学社
- 2) デジタル画像処理編集委員会 『デジタル画像処理』 CG-ARTS 協会
- 3) 竹之内航, 山下英生; 「エピポーラ幾何を用いた3次元顔モデルの構成に関する基礎実験」, 平成20年度電気・情報関連学会中国支部第59回連合大会講演論文集, No.26-4, p.484, 2008

付 録

A. 円抽出アルゴリズム

3.3節において、円の中心座標を求めるアルゴリズムを述べた。そのアルゴリズムの処理⑤の詳細について述べる。

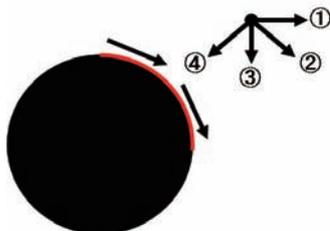


図 A-1 アルゴリズム 1

[アルゴリズム 1]

処理④で円の上端画素が見つかる。まずは図 A-1 の赤線部に沿って黒の画素を探索する。その際、次に探索する画素の

方向に優先順位を付け、その順に探索する。その優先順位を図 A-1 の右上部分に矢印と番号で示す。すなわち、まず①番の方向の画素が黒かどうかを調べる。もし白であれば、②, ③, ④番の順に、その方向の画素を調べる。④番の方向の画素まで調べ、黒の画素が見つからないときは次の処理に移る。

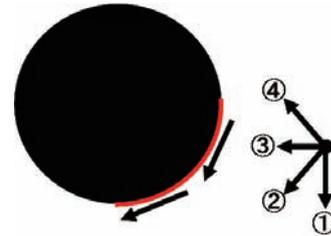


図 A-2 アルゴリズム 2

[アルゴリズム 2]

図 A-2 のように、円周右下部分の黒の画素を同図に示す優先順位に従って追跡する。アルゴリズム 1 と同様に④番の方向の画素まで調べ、黒の画素が見つからないときは次の処理に移る。

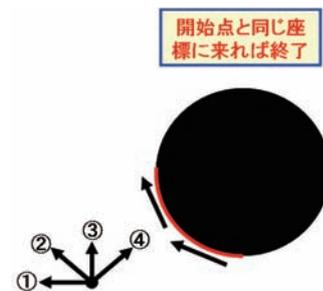


図 A-3 アルゴリズム 3

[アルゴリズム 3]

図 A-3 に示すような優先順位で円周左下部分の黒の画素を追跡する。左方向を一番優先とし、先ほど同様に処理を行う。一周して最初の画素と同じ座標まで戻ると、この処理⑤は終了とする。④番の方向の画素まで調べ、黒の画素が見つからないときは次の処理に移る。

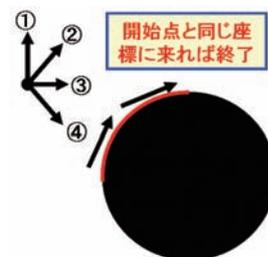


図 A-4 アルゴリズム 4

[アルゴリズム 4]

これまで同様に、図 A-4 に示すような優先順位で円周左上部分の黒の画素を追跡する。一周して最初の画素と同じ画素を見つけたら終了する。この処理の場合、④番の方向の画素まで調べ、黒の画素が見つからないときは「これは円ではない」と判断する。