

効率的な楕円弧選択と幾何学的MDLによる楕円弧統合を組み合わせた楕円検出

益崎 智成^{1,a)} 関 正貴² 菅谷 保之²

受付日 2017年6月11日, 採録日 2017年12月8日

概要: 本論文では, 楕円あてはめに基づく楕円検出手法の枠組みとして, エッジ点列からの効率的な楕円弧選択技術と同一楕円上の分離したエッジ点列を統合する技術を組み合わせた楕円検出法を提案する. 提案手法は, これまでに我々が提案した手法を組み合わせたものであるが, それぞれの手法の問題点を修正することで, 1) 楕円弧の抽出性能を向上, 2) 楕円弧の統合効率の向上を実現し, 高精度な楕円検出を可能とした. さまざまな実験により, 提案手法で使用した手法について, 従来の手法からの性能向上を示すとともに, その他の楕円検出手法との比較を行い, 提案手法の有効性を示す.

キーワード: 楕円検出, 楕円あてはめ, 楕円弧選択, 幾何学的 MDL

Ellipse Detection by Effective Elliptic Arc Extraction and Integration Using Geometric MDL

TOMONARI MASUZAKI^{1,a)} MASATAKA SEKI² YASUYUKI SUGAYA²

Received: June 11, 2017, Accepted: December 8, 2017

Abstract: We propose a new ellipse detection method based on ellipse fitting. This method detects ellipses in two stages. In the first stage, we extract elliptic arcs from an input image by the effective elliptic arc extraction method. Then we integrate elliptic arcs which lie on the same ellipse. The two methods are already proposed by us, however, we point out the problems of those methods and propose new techniques for improving the performance of them. By many experiments, we confirmed that the performance of the improved methods are superior to the original methods. Then, we evaluated the effectiveness of our ellipse detection method by comparing with existing methods.

Keywords: ellipse detection, ellipse fitting, elliptic arc selection, geometric-MDL

1. まえがき

実世界の物体は直線や円などの基本図形で構成されることが多く, 画像認識において基本図形の検出は重要な要素の1つである. 基本図形の中でも円形の物体は画像に楕円として投影されるため, 画像中の楕円を検出して円形物体

を認識することができる. また, 検出した楕円から円形物体の3次元位置の解析が可能である [5]. そのため, 画像から楕円を抽出することで視覚ロボットを含むさまざまな応用手法に適用でき, 画像中に含まれる複数の楕円を検出するさまざまな研究が行われている [9], [12], [15].

画像から抽出したエッジ点列から楕円を検出する手法は, ハフ変換などによる投票に基づいたパラメータの推定手法と, 入力点列から直接的に楕円パラメータを推定する楕円あてはめ手法 [1], [6], [8], [13] の2種類に大別できる.

近年は楕円あてはめ手法に基づいた楕円検出が注目されてきているが, 楕円あてはめ手法は, 入力として与えられる点列は楕円弧上の点にエッジ検出時の誤差が加わったよ

¹ 弓削商船高等専門学校情報工学科
Information Science and Technology Department, National Institute of Technology, Yuge College, Ochi-gun, Ehime 794-2593, Japan

² 豊橋技術科学大学情報・知能工学系
Department of Computer Science and Engineering, Toyohashi University of Technology, Toyohashi, Aichi 441-8580, Japan

a) t.masuzaki@yuge.ac.jp

うな点列を仮定しており，そのような入力に対して，いかに高精度に楕円パラメータを推定するかに重点がおかれている．したがって，入力点列が楕円弧上の点列であれば，高精度に楕円パラメータを推定することが可能である一方，入力点列が直線などの楕円弧上の点列ではなかったり，楕円弧とそれ以外の形状が組み合わさったりした点列に対しては，正しい楕円を検出することが困難である．

以上より，楕円あてはめに基づく楕円検出では，抽出したエッジ点列からいかに効率的に正しい楕円弧を見つけ出せるかがポイントとなる．Sugaya は連結したエッジ点列を対象として，RANSAC を用いて楕円弧以外の点列（以下ではアウトライアと呼ぶことにする）を含んだ点列から楕円あてはめを行う手法を提案した [12]．RANSAC はアウトライアを含む入力データから正しいパラメータを推定するためのフレームワークであり，さまざまなあてはめ問題に対して有効的に使用されている一方で，解が十分に収束するまでに多くの反復計算が必要であるという問題や，アウトライアの割合が大きいと正しい解に収束しないという問題がある．

Qiao らは，連結したエッジ点列に Fitzgibbon らの楕円限定あてはめ [3] を適用し，あてはめ誤差の大きい点を除去することを繰り返して，アウトライアを含んだ点列から楕円を検出する手法 [15] を提案した．Qiao らの手法では，反復処理の中であてはめ誤差が大きく楕円弧候補から除外された点はその後の処理の対象とならないため，反復回数が多くと最終的に得られる点列が短くなり，正しい楕円よりも小さな楕円が検出される傾向がある．

Shao らは，連結した点列を一定の長さや条件により分割し，楕円のあてはめ誤差が小さくなる点列の組合せを総当たりで探索する手法を提案した [11]．Wang らは，点列を短い線分ごとに直線セグメントとして分割し，楕円がよくあてはまる点列の組合せを選択する手法 [14] を提案した．しかし，これらの手法は，経験的なしきい値などを用いて入力点列を細かな部分弧に分割するため，必要以上に多くの部分弧に分割してしまったり，それにより，最適な点列の組合せの探索に時間がかかったりすることがある．

これに対し，Masuzaki らは連結した点列にあてはめた楕円と入力点列との交点によって点列を分割し，分割した点列のあてはめ誤差を考慮して楕円弧を効率的に選択する手法を提案した [7]．この手法では，入力点列をあてはめた楕円との交点によって自動的に部分弧に分割するため，経験的なしきい値などを必要としない．また，選択した楕円弧に対してあてはめた楕円を用いて入力点列を再分割することを繰り返すことで，初期の部分弧分割が正しくなくても，最終的に楕円弧上の部分弧が得られるという利点がある．

本研究では，Masuzaki らのエッジ点列からの楕円弧選択手法 [7] と Sugaya の幾何学的 MDL (Geometric Mini-

mum Description Length) による同一楕円上の楕円弧統合手法 [12] を組み合わせさせた新たな楕円検出の枠組みを提案する．上記の 2 つの技術には新規性はないが，本論文ではこれらの技術の問題点を示し，それらを解決する手法を提案することで，1) 楕円弧の抽出性能を向上，2) 楕円弧の統合効率の向上を実現する．また，改良した 2 つの技術を組み合わせることで，既存の楕円検出手法よりも高精度な楕円検出が可能なることを示す．

2. 楕円弧選択手法の改良

2.1 楕円弧選択の問題点

Masuzaki らの手法では，入力点列に対してあてはめた楕円との交点によって，点列を部分弧に分割した．そして，分割した部分弧ごとに，それに対応する楕円のあてはめ誤差グラフの最大値前後での誤差変化を調べて，楕円に属する部分弧では誤差グラフの最大値周辺での誤差の変化が小さいと仮定して部分弧を選択していた．

誤差グラフの最大値周辺の誤差変化のみに注目したのは処理の効率化のためであり，楕円弧に属する部分弧に対してはこの仮定が成立するが，部分弧の形状によっては，楕円上の点列でなくても誤差グラフの最大値周辺の誤差変化が小さくなることもある．また，誤差変化の大きさをしきい値で判断しているため，このしきい値の設定が楕円弧選択の性能に影響する．図 1 (a) は入力点列とそれにあてはめた楕円を表し，その楕円との交点で入力点列を部分弧に分割し，色分けして表示している．図 1 (b) は入力点列に対する楕円あてはめの誤差をプロットしたグラフであり，グラフの色は図 1 (a) の部分弧の色と対応している．青色の部分弧は楕円上の点列であり，対応する誤差グラフでは，最大値周辺での誤差変化が緩やかであることが分かる．一方，緑色の部分弧は楕円弧と直線が結合した点列であるが，

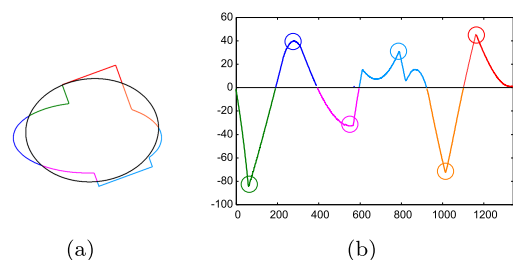


図 1 楕円弧選択の失敗例．(a) 入力点列とそれにあてはめた楕円．(b) 楕円のあてはめ誤差のグラフ．点列が楕円の外部にある場合には正の値，内部にある場合には負の値で示してある．グラフ中の丸印は部分弧単位で見たグラフの最大値である

Fig. 1 Example of elliptic arc selection. (a) An input point sequence and the fitted ellipse. (b) The signed residual graph. If the points are located at inside of the fitted ellipse, we show the corresponding residual values as negative values. Circular symbols indicate the maximum residuals for each partial arc.

これに対応する誤差グラフの最大値（図 1 のグラフではあてはめた楕円の内部の点列に対応する誤差には負の符号を与えているために最小値）周辺の誤差変化が大きく尖った形状をしている。しかし、ピンク色の点列では、対応する誤差グラフの最大値周辺での誤差変化が緩やかであるため、この点列が楕円上の点列として間違えて選択される可能性がある。

2.2 楕円弧選択の改良

図 1 の例でも示したように、誤差グラフの最大値周辺の誤差変化では正しく楕円上の部分弧を選択できない可能性がある。そこで我々は、誤差グラフの最大値周辺の誤差変化を調べる代わりに、分割した部分弧に楕円をあてはめ、1 点あたりの平均あてはめ誤差が小さい点列を選択する。これは Shao らや Wang らの手法と考え方は同じであるが、すべての部分弧の組合せに対して楕円あてはめを行うのではなく、あくまでも単一の部分弧ごとに楕円あてはめを行い、1 つの部分弧を選択する。これは、以下のような理由からである。

- あてはめた楕円との交点によって分割した部分弧は、楕円上の点列とそれ以外の点列が組み合わさったものである可能性があるため、Shao らの手法のような組合せ探索では正しい楕円弧点列を検出できない。
- Masuzaki らの手法は、楕円弧選択と楕円あてはめを繰り返して、その過程で得られる部分弧も変化し、最終的に入力点列は楕円上の点列とそれ以外に分割される。そのための初期選択では、楕円上の点列が一部でも得られればよい。

2.3 改良後の楕円弧選択アルゴリズム

本手法でも、Masuzaki らの手法の考え方を引き継ぎ、1 回目の点列選択と 2 回目以降の点列選択では、異なる手法で部分弧選択を行う。

2.3.1 1 回目の部分弧選択

1 回目の部分弧選択の結果はそれ以降の結果に大きな影響を与えるため、より信頼度の高い楕円弧点列を選択したい。そのため、楕円との交点で分離した部分弧だけではなく、隣接する部分弧の誤差の最大値までの点列を追加した点列を用いた楕円あてはめと平均誤差の計算を行う [7]。これは対象とする部分弧が楕円上の点列である場合には、それに隣接する追加対象とした点列も楕円上の点列であることが多いという事実に基づいている。具体的な手順を以下にまとめる。

- (1) 入力点列にあてはめた楕円との交点により入力点列を部分弧に分割する。
- (2) 各部分弧とそれに隣接する部分弧の誤差の最大値までの点列に楕円をあてはめる。ただし、隣接する部分弧

が接部分弧*1であった場合は、さらに隣接した部分弧の誤差の最大値までの点列を追加する。

- (3) あてはめた楕円に対する 1 点あたりの平均誤差を計算し、平均誤差が最小となる部分弧を楕円弧として選択する。楕円とのあてはめ誤差の計算方法は Masuzaki らの手法と同様である。

2.3.2 2 回目以降の部分弧選択

2 回目以降の点列選択では、1 つ前のステップで選択した部分弧とそれ以外の部分弧を 1 つずつ組み合わせたペアに対して、楕円あてはめを行い、平均あてはめ誤差が最小なペアを選択する。以下に、2 回目以降の部分弧選択の手順を示す。

- (1) 1 つ前のステップで選択した部分弧と、その点列にあてはめた楕円によって分割される部分弧の組合せに対して、楕円をあてはめる。
- (2) 平均あてはめ誤差を計算し、最小となる部分弧を選択する。
- (3) 選択した部分弧に含まれる点数が、1 つ前のステップで選択したものより少なければ、前のステップで選択した部分弧とその楕円パラメータを出力して終了する。また、他に選択できる部分弧がない場合は、現在選択している部分弧と楕円パラメータを出力して終了する。そうでなければ、手順 (1) に戻る。

3. 幾何学的 MDL を用いた楕円弧統合の改良

Sugaya [12] は 2 つの点列が同一楕円上の点列であるかを、2 つの点列に別々に楕円をあてはめた残差と 1 つの楕円をあてはめた残差の比較にモデルの自由度を考慮した幾何学的 MDL によるモデル選択を導入することで、しきい値を用いない楕円弧統合判定を提案した。

3.1 楕円弧統合の問題点

この手法では、1) 与えられた楕円弧のすべての組合せに対して楕円をあてはめ、2) そのあてはめ誤差の小さい順に楕円弧の組を並べ替える。そして、あてはめ誤差の小さい順に幾何学的 MDL による統合判定を行い、楕円弧の統合が行われた場合には、統合した楕円弧と残りの楕円弧に対して上記の処理を行い、これを統合される楕円弧がなくなるまで繰り返す。そのため、入力楕円弧の数が多いと楕円弧の組合せ数が大きくなり、非常に処理時間がかかってしまう。

3.2 楕円弧統合の改良

本研究では、すべての組合せに対して楕円あてはめを行い、あてはめ誤差計算を行うのではなく、楕円弧の形状や

*1 点列にあてはめた楕円と点列が接しており、部分弧内の最大のあてはめ誤差が閾値（実験では 1.5 以下のあてはめ誤差とした）以下の部分弧を接部分弧と定義している。

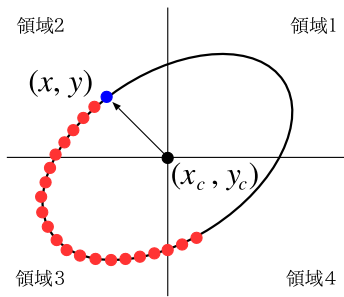


図 2 楕円弧点列の分類. 楕円を楕円中心を原点とした軸により 4 つの領域に分割する

Fig. 2 Classification of an elliptic arc. We divide an ellipse into four regions by the horizontal and the vertical axis whose center is located at the ellipse center.

位置などの幾何学的な情報を用いて統合判定を行うべき組合せを選別することで、統合処理の効率化を目指す。

楕円弧統合を行う組合せを選別するために、以下の特徴を考慮する。

- (1) あてはめた楕円の弧長に対する楕円弧の占める割合
たとえば、あてはめた楕円の弧長に対して楕円弧点列の数がその大部分を占めていれば、入力点列に対してほぼ正しい楕円が得られていると考えられる。そのため、この割合が大きい楕円弧は統合判定から除外する。
- (2) あてはめた楕円に対する楕円弧の位置
楕円弧どうしの位置があてはめた楕円から見て離れた位置にある場合には、それらは別々の楕円上の点列である可能性が高い。また、楕円弧がある程度の長さを持つ場合には、あてはめた楕円の右上に位置する弧どうしのように、楕円上で似た位置関係にある楕円弧が同一の楕円上の点列である可能性は低い。これらを考慮して、同一楕円上にない可能性がある楕円弧どうしの統合判定は行わない。

以上を考慮した楕円弧の選別を行うために、あてはめた楕円に対する入力楕円弧の位置に基づく楕円弧選別手法を提案する。

3.2.1 点列の分類

あてはめた楕円の中心を原点として、水平方向および垂直方向に軸をとり、その軸によって楕円を図 2 に示す 4 つの領域に分割する。そして、楕円あてはめに使用した楕円弧点列と楕円中心の座標を比較して、その大小関係により、楕円弧点列を 4 つの領域に分類する。

図 2 の青点 (x, y) の例では、楕円中心 (x_c, y_c) との大小関係を見ると、 $x - x_c < 0$ 、 $y - y_c > 0$ であるから、領域 2 に分類される。

3.2.2 点列の分類を利用した楕円弧の選別

前項の手法により楕円弧点列ごとに各領域に含まれる点の数を調べ、その情報を用いて、1) 楕円弧統合判定の対象から除外する楕円弧の検出と、2) 統合判定を行う組合せの

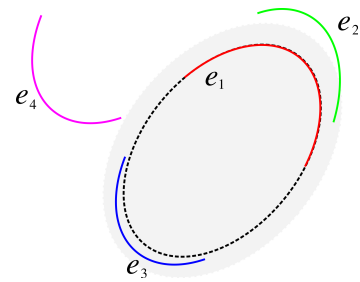


図 3 統合判定を行う組合せの判定
Fig. 3 Selection of elliptic arcs for integration.

選別の 2 段階の処理により、楕円弧統合判定を行う楕円弧を絞り込む。

楕円弧統合判定の対象から除外する楕円弧の検出

以下の条件を満たす楕円弧を統合判定の対象から除外する。

- (1) 3 つ以上の領域に点が存在する場合
正しい楕円を得るのに十分な点が存在するものとして、楕円弧統合判定の対象から除外する。
- (2) 楕円の長軸方向の 2 領域に点が存在する場合
楕円の両側の点列から楕円をあてはめると正しい楕円が得られる可能性が高いため、楕円弧統合判定の対象から除外する。ただし、楕円の短軸方向の 2 領域に点が存在する場合は、長軸方向に楕円がずれる可能性があるため、楕円弧統合判定の対象からは除外しない。楕円の長軸方向の 2 領域の選択については付録に記載した。

統合判定を行う組合せの選別

残った楕円弧に対して、4 つの領域の中で含まれる点の数が最大な領域をその楕円弧の代表領域として定め、それぞれ代表領域が異なる楕円弧どうしのみを統合判定の対象とする。また、楕円弧どうしの画像中の位置関係を考慮するために、上記の条件を満たす組合せの中で、一方の楕円弧にあてはめた楕円の領域内*2にもう一方の楕円弧が存在する場合に、その組合せを統合判定の対象とする。

図 3 を例にして楕円弧 e_1 との統合判定対象となる組合せを考える。楕円弧 e_2 は楕円弧 e_1 と代表領域が同じため、 $\{e_1, e_2\}$ は統合判定の対象にしない。次に楕円弧 e_3 は代表領域が異なり、さらに楕円弧 e_1 にあてはめた楕円 (図中の点線で示した楕円) 領域を拡大した領域 (図中の網掛けの楕円領域) に楕円弧 e_2 の点が含まれるため、 $\{e_1, e_3\}$ は統合判定の対象とする。最後に、 e_4 は代表領域は楕円弧 e_1 と異なるが、網掛けの楕円領域内に点が含まれないため、 $\{e_1, e_4\}$ は統合判定の対象にしない。

以上の手法で選別した楕円弧組に対して、幾何学的 MDL による統合判定を行い、楕円弧の統合を行う。

*2 実験では、楕円弧に楕円をあてはめた楕円に対して長軸と短軸をそれぞれ $\delta = 20$ 画素だけ拡張した楕円領域内を対象とした [9]。

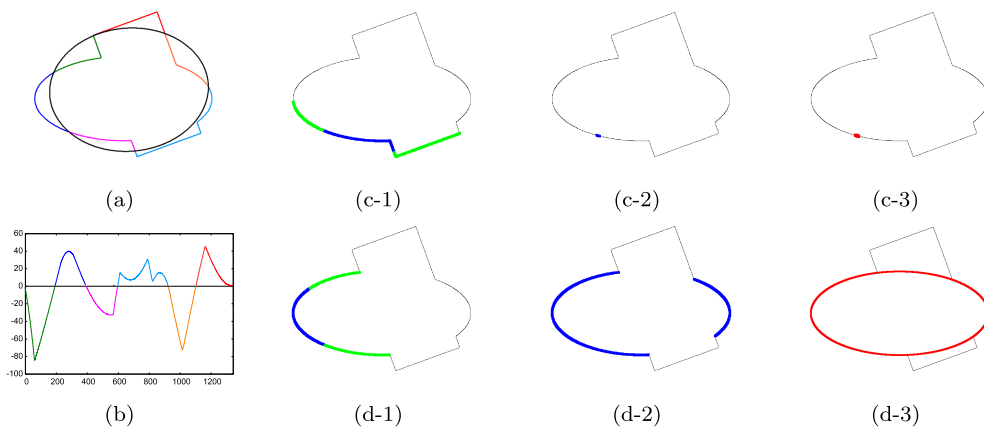


図 4 楕円弧選択性能の比較. (a) 入力点列とそれにあてはめた初期楕円, (b) 初期楕円のあてはめ誤差グラフ, (c-1), (d-1) 初期選択楕円弧 (改良前の手法, 改良後の手法), 青の点列は選択した部分弧, 緑の点列は拡張した隣の部分弧, (c-2), (d-2) 最終的に得られた楕円弧 (改良前の手法, 改良後の手法), (c-3), (d-3) 得られた楕円弧にあてはめた楕円 (改良前の手法, 改良後の手法)

Fig. 4 Comparison of elliptic arc extraction. (a) Input points and the fitted ellipse. (b) The signed residual graph. (c-1), (d-1) The extracted arc in the first stage by before/after our improvement. The blue arc is selected partial arc. The green arcs are the extended arcs from the selected partial arc. (c-2), (d-2) The final extracted arcs. (c-3), (d-3) The fitted ellipse to the extracted arcs.

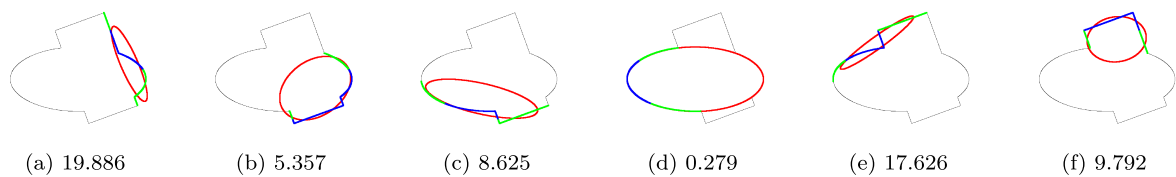


図 5 提案手法における初期楕円弧の選択過程. 青の点列は選択した部分弧, 緑の点列は拡張した隣の部分弧, 赤の楕円は青の点列と緑の点列にあてはめた楕円. 数値はあてはめた楕円に対する入力点列の平均誤差

Fig. 5 The average residuals for each partial arc in the first stage. The blue arc is a target arc. The green arcs are the extended arcs from the target. The red ellipse is the fitted ellipse to the blue and the green arcs. The value indicates the average residual.

4. 実験

提案手法の有効性を示すために, 次に示す 4 つの実験を行った. 実験 1 から実験 3 の実験では, 検証事項が明確に判断できるように簡単な図形を描画した入力画像を用い, 実験 4 では実画像を用いてそれぞれの実験を行った.

- (1) 楕円弧選択性能の向上確認
- (2) Shao らの手法との楕円弧選択性能の比較
- (3) 楕円弧統合効率の向上確認
- (4) 楕円検出性能の比較

4.1 楕円弧選択性能の向上確認

まず, Masuzaki らの手法 [7] と提案手法による改良手法との楕円弧選択の性能比較を行った. 図 4(a) は入力点列とそれにあてはめた初期楕円である. また, 図 4(b) は, 初

期楕円に対する入力点列のあてはめ誤差をプロットしたグラフである. 図 4(c-1) は, Masuzaki らのオリジナルの手法で 1 回目の反復で得た楕円弧であるが, 誤差の変化に基づく判定では楕円弧ではない点列が選択されていることが分かる. 一方, 図 4(d-1) は, 改良手法によって選択した楕円弧である. 楕円弧ごとの楕円のあてはめ誤差に基づいて判定することにより, 楕円上の点列が得られている. また, 図 4(c-2), (c-3) は, Masuzaki らの手法で最終的に抽出した楕円弧とそれにあてはめた楕円である. 1 回目の反復で間違っただけの点列を選択した影響で, 最終的に正しい楕円弧を抽出できず, 正しい楕円が得られていない. しかし, 改良手法では正しい楕円弧を抽出し, それにより正しい楕円が得られている (図 4(d-2), (d-3)).

図 5 は, 改良手法による 1 回目の楕円弧選択での選択対象点列とそれにあてはめた楕円およびその平均あてはめ誤

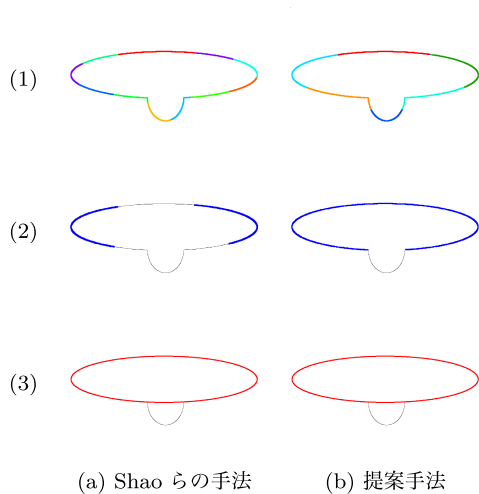


図 6 楕円弧選択性能の比較 1. (1) 分割した部分弧, (2) 最終的に抽出した点列, (3) (2) の点列にあてはめた楕円

Fig. 6 Comparison of elliptic arc extraction vs. Shao's method. (1) Divided partial arcs. (2) Extracted arcs. (3) Fitted ellipse to the extracted arcs. (a) Shao's method. (b) Our method.

差である. 図 5(c) の点列はオリジナルの手法で選択された点列であるが, 楕円のあてはめ誤差は大きいことが分かる. そして, 楕円あてはめの平均誤差に基づいて図 5(d) の点列が正しく選択されている.

4.2 楕円弧選択性能の比較

次に Shao らの手法との比較を行い, 楕円弧抽出の効率と性能の面で提案手法が優れていることを示す. 図 6(1)(a) は Shao らの手法によって分割した入力点列を分割した点列ごとに色分けして表示したものである. Shao らの手法では, 分割したこれらの点列の組合せに対して楕円あてはめ誤差が最小になる組合せを総あたりで計算する. その結果, 得られた点列が図 6(2)(a) の青色の点列である. この例では 12 個の部分弧が得られ, 楕円あてはめは $\sum_{i=6}^{12} 12C_i = 2,510$ 通りの部分弧の組合せに対して行った. また, 最終的に抽出した点列は楕円上の点列の一部のみであったが, 正しい楕円が得られている.

一方, 図 6(1)(b) は入力点列を提案手法によって分割した点列であり, 入力点列にあてはめた楕円との交点により分割を行っている. 提案手法では, あてはめた楕円との交点により機械的に点列を分割しているが, 点列選択と楕円あてはめを繰り返すたびに点列の分割を行うため, 最終的に図 6(2)(b) のように楕円上の点列をもれなく抽出することができており, それにあてはめた楕円も正しい結果となっている. また, 提案手法では, 全体で 25 回の楕円あてはめが行われ, Shao らの手法と比べて点列の分割の処理コストと, 楕円あてはめの回数の点で効率的であることが分かる.

図 7 は別の入力点列に対する実験結果である. Shao ら

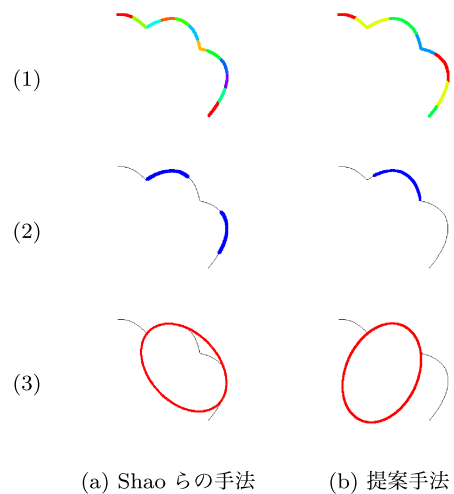


図 7 楕円弧選択性能の比較 2. (1) 分割した部分弧, (2) 最終的に抽出した点列, (3) (2) の点列にあてはめた楕円

Fig. 7 Comparison of elliptic arc extraction vs. Shao's method. (1) Divided partial arcs. (2) Extracted arcs. (3) Fitted ellipse to the extracted arcs. (a) Shao's method. (b) Our method.

の手法では, 異なる楕円上の点列を抽出しており, 正しい楕円が得られていない. 一方, 提案手法では, 正しい楕円弧を抽出しており, その結果, 正しい楕円が得られている. 図 7(1) の初期の点列分割結果を見ると, 2 つの手法でおおよそ似た分割結果を得ている. Shao らの手法は, この点列分割結果に完全に最終結果が依存するが, 提案手法では, 反復処理のたびに点列の分割を繰り返すため, 反復のたびに得られる分割結果が実際の楕円弧に一致したものに自動的に近づいていくという利点がある.

4.3 楕円弧統合の効率向上の確認

図 8(a) の入力点列を用いて, 提案手法によって楕円弧統合の効率が向上することを確認する実験を行った. 図 8(b) は入力楕円弧に楕円をあてはめた結果である. 従来のアルゴリズムでは, この 8 個の点列のすべての組合せに対して楕円あてはめを行い, あてはめ誤差の小さい順に統合判定処理を行っていく. 図 8(c) は, 提案手法の第 1 段階で統合判定対象から除外した点列である.

次に残った点列に対して, 図 2 に示した 4 つの領域の中から代表領域を割り当てた結果を代表領域ごとに色分けして図 8(d) に示す. この結果をもとに統合判定を行う点列の組合せを求めたところ, 5 組の点列が統合判定の対象となり, オリジナルの方法よりも統合判定対象となる点列を効率的に削減することができている. 最後に図 8(e) が楕円弧の統合結果を示す. 同じ色の点列が同一楕円上の点列として判定されたものである. そして, 図 8(f) がそれにあてはめた楕円である. なお, オリジナルの方法でも最終的には同じ結果を得たが, 提案手法により処理の効率を上げながら, 間違った統合結果を減らす効果も期待できる.

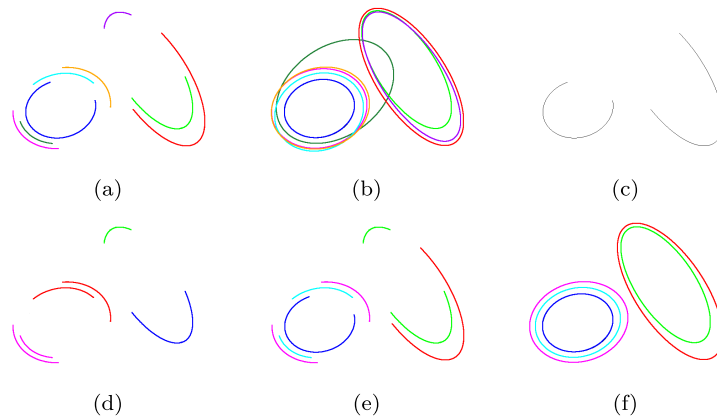


図 8 楕円弧統合の効率向上の確認。(a) 実験に使用したエッジ点列, (b) (a) にあてはめた楕円, (c) 統合対象から除外した点列, (d) 統合対象点列の領域割当て結果, (e) 楕円弧の統合結果, (f) 最終的に得られた楕円

Fig. 8 Improvement of elliptic arc integration. (a) Input arcs. (b) Fitted ellipses to each input arc. (c) Excluded arcs from integration. (d) Grouping result. (e) Integration result. (f) Fitted ellipses to the integrated arcs.

4.4 楕円検出性能の比較

最後に、実画像を入力*3としてさまざまな楕円検出手法との楕円検出性能の比較を行った。

楕円検出手法には提案手法のほかに、Shao らの手法 [11], Qiao らの手法 [15], Fornaciari らの手法 [2] を用いた。Qiao らの手法のパラメータである、楕円と判定するためのあてはめた楕円に対する使用した点列が占める楕円弧角度を 60 度以上とした。また実験は、CPU Intel Core i5-4690 3.50 GHz, 主メモリ 16 GB, OS Ubuntu 12.04 LTS の環境で行った。

4.4.1 楕円の検出結果

図 9 に実験結果を示す。図 9 (a) は入力画像、図 9 (b) はそれぞれの楕円検出手法に入力した点列、図 9 (c), (d), (e), (f) はそれぞれ、提案手法、Shao らの手法、Qiao らの手法、Fornaciari らの手法によって検出した楕円である。ただし、すべての手法で、短軸と長軸の長さの比が 0.3 よりも小さい細長い楕円と、短軸の長さが 10 ピクセルより短い小さな楕円は誤検出と判断して描画していない。図の結果より、すべての手法でおおむね良好な楕円検出結果が得られたが、提案手法では図 9 (2) のお盆の縁のように短い点列がばらばらに得られた部分に対しても正しい楕円が検出されていることが分かる。また、図 9 (3) では、同一楕円上の点列が複数に分離されていたり、楕円上の点列以外の点列と連結していたりする場合でも、提案手法による点列の分割と複数楕円の統合によって正しい楕円が検出できている。一方、Fornaciari らの手法では、楕円の統合が行われない影響で 1 つの円形部分に複数の楕円が検出されていることが分かる。

4.4.2 楕円の精度評価

次に、検出した楕円の精度評価を Prasad らの論文 [10] で用いられている式 (1) を用いて行った。

$$D = 1 - \frac{\text{count}(\text{XOR}(\bar{\mathcal{L}}, \mathcal{L}_\alpha))}{\text{count}(\text{OR}(\bar{\mathcal{L}}, \mathcal{L}_\alpha))} \quad (1)$$

ここで、 $\bar{\mathcal{L}}$ は正解の楕円内に含まれる画素の集合、 \mathcal{L}_α は検出した楕円 α 内に含まれる画素の集合を表し、 $\text{count}(\text{XOR}(\bar{\mathcal{L}}, \mathcal{L}_\alpha))$ は $\bar{\mathcal{L}}$ と \mathcal{L}_α のどちらか一方の領域のみ含まれる点の数、 $\text{count}(\text{OR}(\bar{\mathcal{L}}, \mathcal{L}_\alpha))$ は、 $\bar{\mathcal{L}}$ と \mathcal{L}_α のいずれかの領域に含まれる点の数を表す。したがって、 D は正解の楕円と検出した楕円の重なり具合を表し、値が 1 に近いほど正しい楕円が得られたことを意味する。

図 9 (a) の入力画像上に描画した赤い楕円は、検出した楕円の精度評価を行うために指定した楕円である。これらは、シーン中で円形をしており、その円弧上に全体のおよそ 50% 以上の点列が得られていると判断したものである。そして、楕円パラメータの正解値は、円弧上に得られたエッジに基づいて最尤推定法により算出した。その際、検出したエッジ点のみでは正しい楕円形状が得られないものに関しては目視で円弧上の点を追加した上で楕円パラメータを推定した。

精度評価の結果を表 1 に示す。表中の列の番号は図 9 に示した画像番号である。また、各手法について上段の数値は、図 9 (a) に示した評価対象の楕円に対する検出した楕円の平均評価値を表す。このとき、評価対象の楕円が検出されなかった場合にはその評価値を 0 として扱った。下段の数値は、評価値が 0 となったものを除いた平均値を表し、検出した楕円数と評価対象の楕円数も示した。また、画像例ごとに最高評価値を太文字で強調した。

表 1 の結果より、多くの画像例で提案手法が精度良く

*3 入力画像の一部は、Caltech256 [4] データセットを利用した。

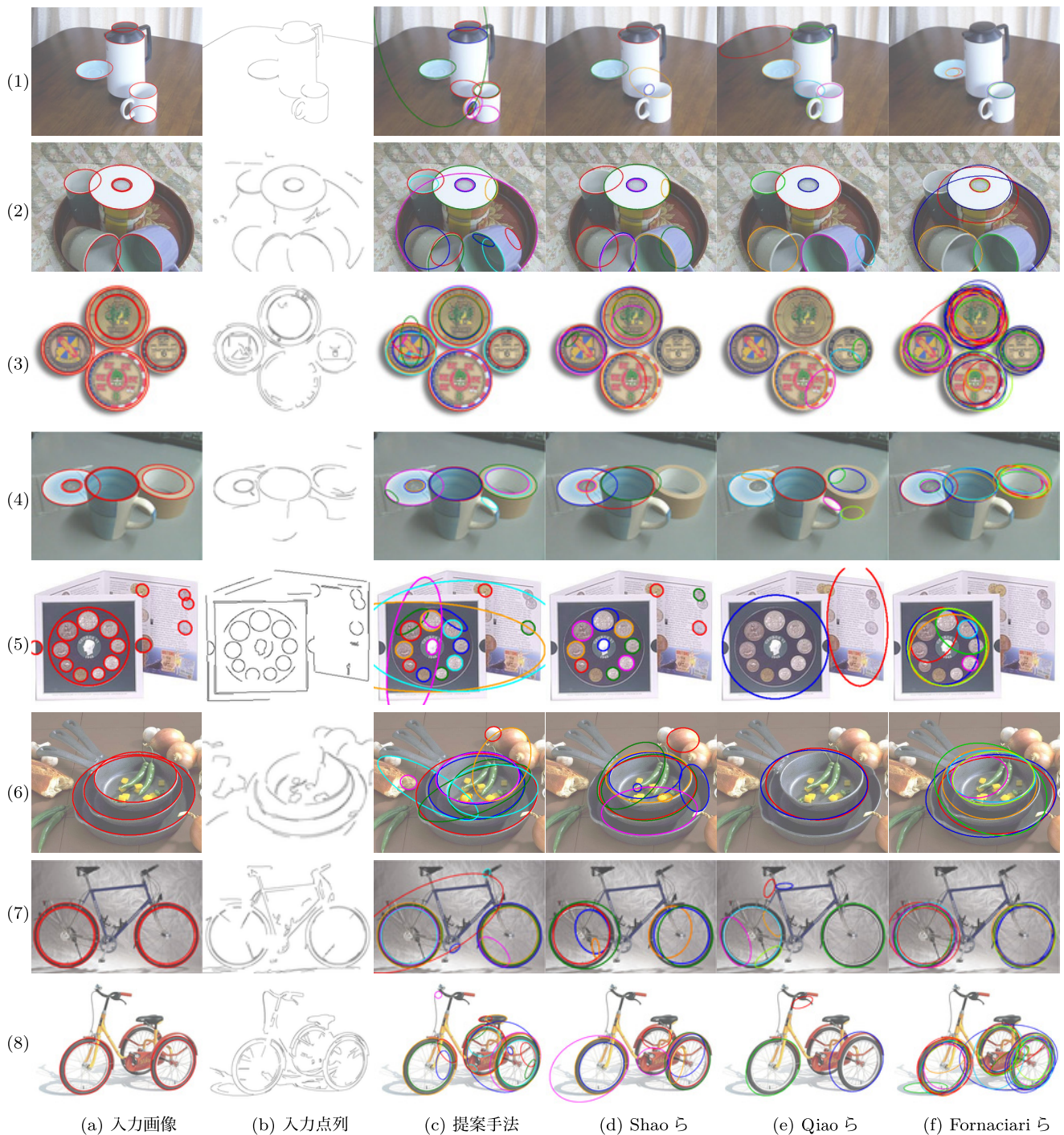


図 9 楕円の検出結果の比較

Fig. 9 Ellipse detection for real images. (a) Input image. (b) Our method. (c) Shao's method. (d) Qiao's method. (e) Fornaciari.

楕円を検出できており、検出漏れも少ないことが分かる。また、すべての画像例での平均評価値は、提案手法 0.889, Shao らの手法 0.717, Qiao らの手法 0.376, Fornaciari らの手法 0.767 となり、この値からも提案手法の有効性が確認できる。

4.4.3 計算時間

それぞれの手法について計算時間を計測した結果*4を

*4 Shao らの手法のみ Matlab を用いて実験を行っている。

表 2 に示す。Fornaciari らの手法は非常に高速である。Qiao らの手法と提案手法はほぼ同程度の計算時間である。Qiao らの手法も我々の手法と同様に楕円あてはめを繰り返して楕円弧を選択する枠組みであるが、提案手法は楕円弧の統合処理が加わるため、提案手法の方が若干ではあるが計算時間が長い傾向にある。提案手法を C 言語で実装したのに対し、Shao らの手法は Matlab を用いて実行をしているため、単純には計算時間を比較できないが、入力点列

表 1 検出した楕円の精度比較. 上段: 検出されなかった楕円も含めて評価対象とした楕円の評価値平均. 下段: 検出できた楕円のみでの評価値平均と検出した楕円数/評価対象の楕円数

Table 1 Accuracy comparison of the detected ellipses. Upper: Average accuracy for all targets. Lower: Average accuracy for detected ellipses. Num. of detected ellipses/num. of target ellipses.

	(1)	(2)	(3)	(4)	(5)	(6)	(7)	(8)
提案手法	0.979	0.772	0.931	0.986	0.651 (0.837, 12/16)	0.970	0.857	0.964
Shao ら	0.410 (0.614, 3/5)	0.812	0.719 (0.862, 8/10)	0.622 (0.871, 5/7)	0.647 (0.940, 11/16)	0.714	0.655 (0.765 5/6)	0.970
Qiao ら	0.532 (0.886, 3/5)	0.631 (0.946, 4/6)	0.376 (0.627, 6/10)	0.310 (0.559, 3/7)	0	0.414 (0.827, 1/3)	0.334 (0.468, 4/6)	0.413 (0.826, 2/4)
Fornaciari ら	0.306 (0.919, 1/5)	0.657 (0.876, 4/6)	0.802 (0.824, 9/10)	0.867	0.266 (0.761, 3/16)	0.821	0.914	0.886

表 2 楕円検出時間の比較 (単位は秒)

Table 2 Comparison of computation time (sec).

	提案手法 (効率化処理なし)	Shao ら	Qiao ら	Fornaciari ら
(1)	0.214 (1.016)	336.560	0.292	0.022
(2)	0.236 (1.844)	19.304	0.292	0.013
(3)	1.493 (10.418)	22.907	0.624	0.111
(4)	0.193 (0.981)	16.468	0.156	0.035
(5)	0.868 (10.415)	86.830	0.136	0.011
(6)	0.406 (5.092)	42.337	0.268	0.010
(7)	0.387 (22.786)	174.236	2.228	0.035
(8)	1.342 (26.163)	43.622	2.532	0.094

の部分弧分割に計算時間がかかっているものと思われる。

また, 提案手法について, 楕円弧の統合対象を選別せずに検出したすべての楕円弧を統合対象にする場合 (表 2 中では「効率化処理なし」と記述した) との計算時間の比較を行うと, 検出した楕円弧の数が多いほど提案手法の効果が現れていることが分かる。ただし, 統合対象の点列数に応じて計算時間が長くなるという問題は完全には解決できていないため, 今後の検討課題となる。

5. まとめ

本論文では, 楕円あてはめに基づく楕円検出手法の枠組みとして, エッジ点列からの効率的な楕円弧選択技術と同一楕円上の分離したエッジ点列を統合する技術を組み合わせた楕円検出法を提案した。その中で, 従来手法の問題点を解決し, 楕円弧抽出性能を向上させるとともに, 楕円弧の統合効率の向上を実現し, 高精度な楕円検出を可能とした。そして, シミュレーションデータや実画像による実験を行い, 提案手法の有効性を評価した。

謝辞 本研究の一部は JSPS 科研費 (基盤研究 (C) No. 26330192) の助成による。

参考文献

- [1] Chojnacki, W., Brooks, M.J., van den Hengel, A. and Gawley, D.: On the fitting of surfaces to data with covariances, *IEEE Trans. Patt. Anal. Mach. Intell.*, Vol.22, No.11, pp.1294–1303 (2000).
- [2] Fornaciari, M., Prati, A. and Cucchiara, R.: A fast and effective ellipse detector for embedded vision applications, *Pattern Recognition*, Vol.47, No.11, pp.3683–3708 (2014).
- [3] Fitzgibbon, A., Pilu, M. and Fisher, R.B.: Direct least squares fitting of ellipses, *IEEE Trans. Patt. Anal. Mach. Intell.*, Vol.21, No.5, pp.476–480 (1999).
- [4] Griffin, G., Holub, A. and Perona, P.: Caltech-256 object category dataset. Technical Report UCB/CSD-04-1366, California Institute of Technology (2007).
- [5] Kanatani, K.: *Geometric Computation for Machine Vision*, Oxford University Press, Oxford, U.K. (1993).
- [6] Leedan, Y. and Meer, P.: Heteroscedastic regression in computer vision: Problems with bilinear constraint, *Int. J. Comput. Vis.*, Vol.37, No.2, pp.127–150 (2000).
- [7] Masuzaki, T. and Sugaya, Y.: Connectivity-based error evaluation for ellipse fitting, *Proc. IAPR Conf. on Machine Vision Applications (MVA2015)*, pp.118–121 (2015).
- [8] Matei, J. and Meer, P.: Estimation of nonlinear errors-in-variables models for computer vision applications, *IEEE Trans. Patt. Anal. Mach. Intell.*, Vol.28, No.10, pp.1537–1552 (2006).
- [9] 岡部光生, 金谷健一, 太田直哉: 楕円成長法による円形物体の自動検出, *電子情報通信学会論文誌 D-II*, Vol.J85-D-II-12, pp.1823–1831 (2002).
- [10] Prasad, D.K., Leung, M.K.H. and Cho, S.-Y.: Edge curvature and convexity based ellipse detection method, *Pattern Recognition*, Vol.45, No.9, pp.3204–3221 (2012).
- [11] Shao, M., Ijiri, Y. and Hattori, K.: Grouped outlier removal for robust ellipse fitting, *14th IAPR International Conference on Machine Vision Applications (MVA2015)*, pp.138–141 (2015).
- [12] Sugaya, Y.: Ellipse Detection by combining division and model selection based integration of edge points, *4th Pacific-Rim Symposium on Image and Video Technology (PSIVT2010)*, pp.64–69 (2010).
- [13] Taubin, G.: Estimation of planar curves, surfaces, and non-planar space curves defined by implicit equations with applications to edge and range image segmentation, *IEEE Trans. Patt. Anal. Mach. Intell.*, Vol.13, No.11,

pp.1115–1138 (1991).

- [14] Wang, Y., He, Z., Liu, X., Tang, Z. and Li, L.: A fast and robust ellipse detector based on top-down least-square fitting, *Computers and Electrical Engineering*, Vol.40, No.4, pp.1415–1428 (2014).
- [15] Qiao, Y. and Ong, S.H.: Arc-based evaluation and detection of ellipses, *Journal of Pattern Recognition*, Vol.40, No.7, pp.1990–2003 (2007).

付 録

楕円の長軸方向の2領域の計算手法

入力： $(\mathbf{x}, Q\mathbf{x}) = 0$ を満たす 3×3 対称行列 Q 。ただし， \mathbf{x} は点の座標から成る3次元ベクトルで， $\mathbf{x} = (x, y, f_0)^T$ と定義する． f_0 はスケール定数。

出力：楕円の長軸方向の2領域番号

- (1) 行列 Q の 2×2 小行列を S とし，行列 S の最大固有値に対する単位固有ベクトル \mathbf{v} を計算する．このベクトル \mathbf{v} が楕円の長軸方向を表す。
- (2) ベクトル \mathbf{v} の第1成分と第2成分の符号が同符号であれば，図2の領域1と領域3を長軸方向の2領域とし，異符号であれば，図2の領域2と領域4を長軸方向の2領域とする。

ただし，ベクトル \mathbf{v} のどちらかの成分が微小定数 ϵ (実験では $\epsilon = 10^{-4}$ とした) よりも小さい場合は，楕円の長軸が X 軸もしくは Y 軸と平行になり，長軸方向の2領域が定められないため，含まれる点数が最大となる領域とその対角の領域を選択する。



菅谷 保之 (正会員)

1996年筑波大学第三学群情報学類卒業。2001年筑波大学大学院博士課程修了。工学博士。2001年岡山大学工学部助手。2006年豊橋技術科学大学情報工学系講師。2009年豊橋技術科学大学情報工学系准教授。コンピュー

タビジョン，主に複合現実感の研究に従事。



益崎 智成

2014年豊橋技術科学大学情報・知能工学課程卒業。2016年弓削商船高等専門学校助教。コンピュータビジョン，楕円検出，複合現実感の研究に従事。



関 正貴

2016年豊橋技術科学大学情報・知能工学課程卒業。2016年豊橋技術科学大学大学院情報・知能工学専攻在学中。コンピュータビジョン，楕円検出の研究に従事。