

点列の連結性を考慮したアウトライアを含む データへの楕円当てはめ

益崎智成^{1,a)} 菅谷保之^{1,b)}

概要: 本論文では画像から抽出した点列に楕円弧以外の点列が含まれていても正しく楕円を当てはめることができる新しい方法を提案する。提案手法は、入力点列が連結した点列であることを仮定して各点の楕円への当てはめ誤差を調べることで、楕円当てはめに使用する点列を選別して楕円当てはめを繰り返す方法である。シミュレーション実験および実画像実験により、アウトライアを含む幾何学的当てはめによく用いられる RANSAC と比べて、計算量と当てはめ精度の面で提案手法が優れていることを示す。

Connectivity-based error evaluation for ellipse fitting

TOMONARI MASUZAKI^{1,a)} YASUYUKI SUGAYA^{1,b)}

Abstract: We propose a new method for fitting an ellipse to a point sequence extracted from an image. This method can fit an ellipse if the point sequence consists of points on elliptic arc and the other points. Assuming that an input points are spatially connected, we iteratively select inlier points and fit an ellipse to them by checking variations of the fitting errors for the connected image points. By using simulated data and real images, we compare the performance of our method with the existing methods, for example the RANSAC and show that their accuracy and computation time of the proposed method is superior to the existing methods.

1. まえがき

シーン中の円形物体を撮影すると画像中では楕円となり、その投影像からその物体の3次元位置が解析できる [6]。そのため、画像から楕円を抽出することは視覚ロボットを含むさまざまな応用の基本的な処理の一つであり、画像中より楕円弧を抽出し楕円を当てはめるさまざまな研究が行われている [18], [19], [23]。

画像から抽出した点列に楕円を当てはめる最も代表的な方法は、最小二乗法であり、最も単純な方法として知られている。より精度の高い方法に Taubin 法 [21] がある。さらに高次の誤差項を除いて偏差が存在しない超精度最小二乗法 [5], [11] や超精度くりこみ法 [12], [13] も提案されている。

また、当てはめる楕円と点列との距離の二乗和（これを再投影誤差 [4] と呼ぶ）を最小にする最尤推定がある。再投影誤差はサンプル誤差 [4] と呼ばれる関数でよく近似さ

れ、サンプル誤差を最小化する反復手法として Chojnacki ら [1] の FNS 法, Leedan ら [14] や Matei ら [16] の HEIV 法, 著者らの射影ガウスニュートン法 [7], [22] がある。また、これらを反復的に適用することによって厳密な最尤推定解が計算できる [9], [10], [17]。

しかし、これらの手法はすべて点列に x, y の2次式（円錐曲線）を当てはめるものであり、ノイズが大きいと楕円以外（双曲線や放物線など）が当てはまることがある。これを防いで楕円のみが当てはまる方法を最初に提案したのは Fitzgibbon ら [3] である。これは代数的距離の最小化であって計算は単純であるが精度が低い。最近 Szpak ら [20] はより精度の高い方法を発表した。また筆者らは超精度くりこみ法を利用して、Szpak ら [20] よりも精度の高い方法を提案した [15]。

これまで紹介した手法は、基本的には入力点列に楕円弧以外の点（これ以降アウトライアと呼ぶ）が含まれることを考慮しておらず、入力点列にアウトライアが含まれている場合には得られる解の精度が低下する。実画像から楕円を検出するには、画像からエッジを抽出してそのエッジ点列を入力として楕円当てはめ、もしくは楕円検出を行う。このとき、楕円弧上の点列とそれ以外の点列が連結して抽出

¹ 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
a) masuzaki@iim.cs.tut.ac.jp
b) sugaya@iim.cs.tut.ac.jp

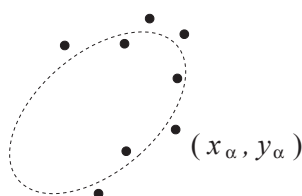


図 1 点列に楕円を当てはめる。

されることがある。点列の曲率をもとにした方法や直線検出などにより楕円弧上の点とそれ以外の点を分離することで、事前にアウトライアを除去する試みもあるが、必ずしも完全にアウトライアが除去できるわけではない。

アウトライアを含むデータに対する幾何学的当てはめで最もよく用いられるのは RANSAC[2] である。RANSAC は、入力データからランダムに選び出したデータからパラメータ推定することを繰り返し、当てはまり具合のよいデータ数が最大になるようなパラメータを求めるため、十分に解が収束するまでに多くの反復が必要であったり、アウトライアの割合が大きいと正しい解に収束しない問題がある。Yu ら [23] は、連結した点列に対して楕円の当てはめ誤差をもとにアウトライアらしい点列を反復して除去することで正しい楕円を当てはめる方法を提案している。Yu らの方法の特徴は一度にアウトライアをすべて除去するのではなく、誤差の大きさをもとにアウトライアと判定された点のうち、最大誤差の点を含む連結点列のみをアウトライアとして除去することを繰り返すことである。これにより、反復の過程で少しずつアウトライアを除去して最終的に正しい楕円を当てはめようとする。しかし、反復の途中でアウトライアと判定された点列は、それ以降の処理で楕円当てはめには使用しないので、除去された点列に楕円弧上の点が含まれていた場合には、正しい楕円を当てはめる点列が少なくなり、得られる楕円の精度が低下する可能性がある。

我々は、入力点列が連結した点列であることを仮定して、入力点列を楕円の当てはめ誤差をもとに部分弧に分割して、部分弧ごとにインライアとアウトライアを判別する新しい方法を提案する。提案方法では、連結した点列に見られる楕円への当てはめ誤差の変化に着目して、インライア点列を選別するため、アウトライアの検出性能が高く、それを繰り返すことでアウトライアの検出性能がさらに向上する。点列の選別はすべての入力点列から毎行うため、Yu らの方法のように反復の過程で対象となる点列が減少することがない。また、RANSAC のようにランダムサンプリングを行わないので、収束までの反復回数も少ない。

2. 楕円当てはめ

x, y の 2 次式

$$Ax^2 + 2Bxy + Cy^2 + 2f_0(Dx + Ey) + f_0^2 F = 0 \quad (1)$$

は円錐曲線と総称され、楕円、放物線、双曲線、およびそ

の退化 (2 直線など) を表す [8]。これが楕円を表すのは

$$AC - B^2 > 0 \quad (2)$$

の場合である [8]。式 (1) 中の f_0 はスケールを調節する固定した定数である*1。ノイズ (以下、データの誤差をノイズと呼ぶ) のある点列 $(x_1, y_1), \dots, (x_N, y_N)$ に楕円を当てはめることは $\alpha = 1, \dots, N$ に対して

$$Ax_\alpha^2 + 2Bx_\alpha y_\alpha + Cy_\alpha^2 + 2f_0(Dx_\alpha + Ey_\alpha) + f_0^2 F \approx 0 \quad (3)$$

となる A, B, C, D, E, F を計算することである (図 1)。6 次元ベクトル

$$\begin{aligned} \xi_\alpha &= (x_\alpha^2, 2x_\alpha y_\alpha, y_\alpha^2, 2f_0 x_\alpha, 2f_0 y_\alpha, f_0^2)^\top, \\ \theta &= (A, B, C, D, E, F)^\top \end{aligned} \quad (4)$$

を定義し、ベクトル \mathbf{a}, \mathbf{b} の内積を (\mathbf{a}, \mathbf{b}) と書けば式 (3) は次のように書ける。

$$(\xi_\alpha, \theta) \approx 0, \quad \alpha = 1, \dots, N \quad (5)$$

θ には定数倍の不定性があるので、 $\|\theta\| = 1$ と正規化する。データ点 (x_α, y_α) は真の位置 $(\bar{x}_\alpha, \bar{y}_\alpha)$ に期待値 0、標準偏差 σ の独立な正規分布に従うノイズ $\Delta x_\alpha, \Delta y_\alpha$ が加わったものであるとみなすと ξ_α の誤差は第 1 近似において次のようになる。

$$\begin{aligned} \Delta \xi_\alpha &= (2x_\alpha \Delta x_\alpha, 2\Delta x_\alpha y_\alpha + 2x_\alpha \Delta y_\alpha, 2y_\alpha \Delta y_\alpha, \\ &\quad 2f_0 \Delta x_\alpha, 2f_0 \Delta y_\alpha, 0)^\top \end{aligned} \quad (6)$$

仮定より $E[\Delta x] = E[\Delta y] = 0, E[\Delta x^2] = E[\Delta y^2] = \sigma^2, E[\Delta x \Delta y] = 0$ であるから、 ξ_α の共分散行列は次のようになる。

$$V[\xi_\alpha] \equiv E[\Delta \xi_\alpha \Delta \xi_\alpha^\top] = \sigma^2 V_0[\xi_\alpha] \quad (7)$$

ただし、 $V_0[\xi_\alpha]$ を次のように置き、正規化共分散行列と呼ぶ。

$$V_0[\xi_\alpha] = 4 \begin{pmatrix} x_\alpha^2 & x_\alpha y_\alpha & 0 & f_0 x_\alpha & 0 & 0 \\ x_\alpha y_\alpha & x_\alpha^2 + y_\alpha^2 & x_\alpha y_\alpha & f_0 y_\alpha & f_0 x_\alpha & 0 \\ 0 & x_\alpha y_\alpha & y_\alpha^2 & 0 & f_0 y_\alpha & 0 \\ f_0 x_\alpha & f_0 y_\alpha & 0 & f_0^2 & 0 & 0 \\ 0 & f_0 x_\alpha & f_0 y_\alpha & 0 & f_0^2 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix} \quad (8)$$

この正規化共分散行列は、提案手法の中で当てはめた楕円と入力点との誤差を計算する際に使用する。

*1 実験では $f_0 = 600$ とした。

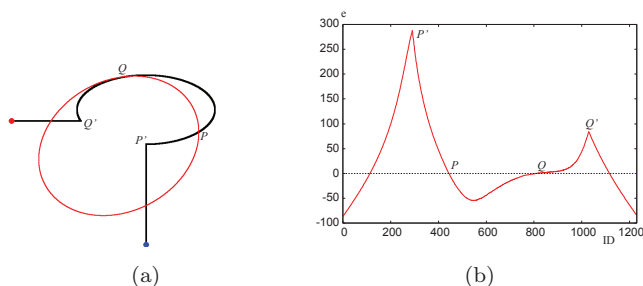


図 2 アウトライアを含む点列への楕円当てはめと誤差のグラフ。(a) 入力点列(黒点)と当てはめた楕円(赤). 入力点列は青丸を始点, 赤丸を終点とする順番に連結性を保って並んでいる。(b) 当てはめた楕円に対する入力点列との誤差. 横軸は(a)の青丸を始点とする点の番号. 縦軸は式(1)の左辺の値によって符号づけした楕円の当てはめ誤差.

3. Fitzgibbon らの方法

式(2)を必ず満たし楕円を当てはめる方法を初めて示したのは Fitzgibbon ら [3] である. これは式(5)の左辺の二乗和

$$J_{LS} = \sum_{\alpha=1}^N (\xi_{\alpha}, \theta)^2 \quad (9)$$

を $\|\theta\| = 1$ ではなく, 次の正規化条件のもとで最小にするものである.

$$AC - B^2 = 1 \quad (10)$$

具体的な手順は以下ようになる.

(1) 次の行列 M_{LS} , N_F を計算する.

$$M_{LS} = \frac{1}{N} \sum_{\alpha=1}^N \xi_{\alpha} \xi_{\alpha}^T, \quad N_F = \begin{pmatrix} 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & -2 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix} \quad (11)$$

(2) 一般固有値問題

$$N_F \theta = \mu M_{LS} \theta, \quad (12)$$

の最大一般固有値 μ に対する単位一般固有ベクトル θ を計算する.

提案手法では, 楕円当てはめにより楕円のみが得られるように, Fitzgibbon らの手法を用いることにする.

4. 提案手法

本研究では, 図 2(a) に示すように, 入力点列は連結しており, 楕円弧上の点列とそれ以外の点列が結合しているものとし, このような入力点列に楕円を当てはめると, 入力点列と交差するような楕円が得られると仮定する. そして,

当てはめた楕円と交わる点で連結点列を部分弧に分割して, 部分弧ごとに当てはめた楕円との誤差の変化を調べ, インライアとアウトライアを判定する. この誤差の変化は連結点列の曲率の変化を擬似的に表しており, 点列が滑らかな曲線であればその誤差変化も滑らかなため, この誤差の変化をもとにしたインライア検出は, 入力点列をその曲率で分割することと同じ発想である. しかし, 提案手法では, すべての点を調べるのではなく, 分割した部分弧ごとに誤差がピークとなる点の誤差グラフの曲率を調べて, その曲率が大きい部分弧をアウトライア, それ以外の部分弧をインライアと判定するので効率的である.

図 2 の例では, 当てはめた楕円により入力点列が 5 つの部分弧に分割される. そして, インライアのみを含む部分弧のグラフでは, 誤差の変化が滑らかなのに対して, アウトライア点を含む部分弧のグラフでは, 誤差がピークの周辺で急激に変化していることが確認できる.

また, インライアである部分弧 PQ の両端の弧 PP' と弧 QQ' もインライアであることがわかる. 点 P' , Q' はインライアと判定した部分弧 PQ に隣接した部分弧の誤差のピーク点である. 従って, インライアと判定された部分弧のみではなく, その両端の部分弧の誤差のピーク点まで含めた部分弧(図 2 の例では部分弧 $P'Q'$) をインライアとして, 次の楕円当てはめに使用する.

提案手法の流れを以下にまとめる.

- (1) 点列に Fitzgibbon らの方法で楕円を当てはめる.
- (2) 当てはめた楕円のパラメータを用いて各点ごとに式(1)の左辺の値を計算し, その符号が変化する点で点列を部分弧に分割する.
- (3) 分割した部分弧ごとに, その中で最大誤差をもつ点とその周辺の点とのなす角 ϕ を計算する.
- (4) 1 回目の楕円当てはめであればステップ(a), そうでなければステップ(b)を行う.
 - (a) 部分弧の長さがしきい値*2よりも長い部分弧の中で計算した角度が最大の部分弧とそれに隣接する部分弧の誤差のピークまでの弧をインライアとして楕円を当てはめる.
 - (b) 計算した角度がしきい値 $\hat{\phi}$ *3以上の部分弧をインライアとして楕円を当てはめる.
- (5) 得られるインライア点数が変化しなくなるまでステップ 2 からステップ 4 の処理を繰り返す.

提案手法では, インライアとして検出する部分弧の選択基準が, 1 回目とそれ以降で異なっている. 前述のようにインライアと判定された部分弧が本当にインライアであれば, それに隣接する部分弧の一部もインライアであるから, こ

*2 実験では入力点数の 5% をしきい値とした.

*3 実験では $\hat{\phi} = 100^\circ$ とした.

れらをすべて使用して楕円を当てはめると効率的に正しい楕円が得られる。しかし、本来アウトライアである部分弧を誤ってインライアと判定した場合には、それに隣接するアウトライアをさらに追加してしまうため、正しい楕円が得られにくくなる。

そのため、1回目のインライア選択では、十分な長さの部分弧であり誤差のピーク周辺での誤差の変化が一番滑らかなものを信頼できるインライアとし、それに隣接する部分弧の一部をさらにインライアとして追加する。2回目以降では、インライアと判定された部分弧をすべてインライアとして採用するが、インライアの部分弧に隣接した部分弧の一部はインライアとしては追加しない。これは、入力点列のうちインライアのみ、もしくはほとんどがインライアである点に楕円を当てはめた場合には、当てはめた楕円と点列の交点インライアとアウトライアの境界となるので、アウトライア点を誤ってインライア点として追加しないためである。

以降の節で各処理の詳細について説明する。

5. 点列の分割

式(1)の左辺の x, y に任意の点 $p(x, y)$ の座標を代入すると、その点が楕円弧上の点であれば0、楕円の外側もしくは内側であれば正の値か負の値となる(符号は反対の場合もある)。従って、当てはめた楕円と入力点列の交点は、式(1)の左辺の符号が変化する点として検出することが可能である。

ただし、点列が楕円に接する場合には、式(1)の左辺の値が厳密に0になることはないので、隣接点間で符号の変化が激しく起こり、非常に短い部分弧が多数生成されることがある。これらの部分弧は楕円の当てはめ誤差が非常に小さくなるため、部分弧内の最大誤差がしきい値以下となる部分弧は当てはめた楕円に接している部分弧として、ピーク点の誤差変化を調べることなくインライアとする。以降の説明ではこの部分弧を「接部分弧」と呼ぶことにする。

6. 部分弧の判別

分割した部分弧ごとに誤差が最大となる点を検出し、その点と前後の点を結ぶ線分のなす角を計算する。その角度がしきい値以下であれば、その部分弧をアウトライア、しきい値以上である部分弧をインライアとする。

最大誤差を持つ点を点とする2直線のなす角は誤差のグラフの横軸のスケールに依存する。例えば、同じ形状の点列でも、その大きさが異なる場合、大きさが大きい方、すなわち、点列の数が多い方が対応する部分の角度が大きくなる。そこで、グラフの横軸をすべての入力点中の最大誤差を基準に正規化して、次の2次元ベクトルで表現する。

$$Q_\alpha = \left(\frac{\lambda e_{max} \alpha}{N}, e_\alpha \right)^T \quad (13)$$

ここで、 e_{max} はすべての入力点中の楕円の当てはめ誤差の

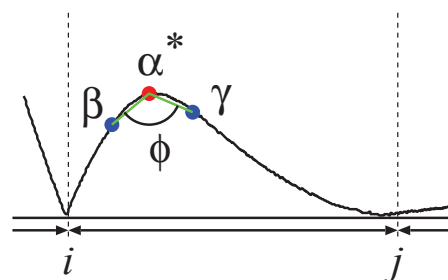


図3 分割した点列の誤差変化の指標

最大値、 λ は正規化のための定数*4、 e_α は式(14)で計算される点 α の当てはめた楕円に対する誤差である。

$$e_\alpha = \sqrt{\frac{(\xi_\alpha, \theta)}{(\theta, V_0[\xi_\alpha] \theta)}} \quad (14)$$

また、当てはめ誤差の最大値が極端に大きい場合には式(13)の正規化がうまく働かないため、誤差の最大値のしきい値 E_{max} を定めて*5、 e_{max} がしきい値 E_{max} より大きい場合には、 e_{max} を E_{max} に置き換えて正規化する。

M 個の部分弧 $R_\kappa(i, j) = \{Q_\beta, \beta = i, \dots, j\}, \kappa = 1, \dots, M$ に分割した点 $Q_\alpha, \alpha = 1, \dots, N$ に対して、次の手順でインライアとアウトライアの判定を行う。アルゴリズム中で説明する記号を図示したものを図3に示す。

- (1) 部分弧 R_κ 内の誤差の最大値を探索し、その点番号を α^* とする。ここで、この部分弧内の誤差の最大値がしきい値 E_{min} *6より小さい場合には、その部分弧を接部分弧としてインライアに分類して終了する。
- (2) 点 α^* から d だけ離れた点を点 β, γ とし、次のように計算する。

$$\beta = \alpha^* - d, \gamma = \alpha^* + d, d = (j - i) / r \quad (15)$$

ここで、 r は最大誤差を取る点と隣接点との距離を定めるパラメータである*7。もし、 β と γ がともにその部分弧内になければ、どちらかが部分弧内となるまで、 $d \leftarrow d - 1$ とする。

- (3) 次の三つの条件に従い、二つのベクトル $x^{(1)}, x^{(2)}$ を計算する。ここで、 $x_b^{(a)}$ はベクトル $x^{(a)}$ の第 b 成分を示す。

(a) Q_β と Q_γ がともに部分弧内の点の場合:

$$x^{(1)} = Q_{\alpha^*} - Q_\beta, x^{(2)} = Q_{\alpha^*} - Q_\gamma \quad (16)$$

(b) Q_β のみが部分弧内の場合:

$$x^{(1)} = Q_{\alpha^*} - Q_\beta, x^{(2)} = (-x_1^{(1)}, x_2^{(1)})^T \quad (17)$$

(c) Q_γ のみが部分弧内の場合:

*4 実験では $\lambda = 2.0$ とした。

*5 実験では(入力点の座標の最大値/3)とした。

*6 実験では $E_{min} = 1.5$ とした。

*7 実験では $r = 8$ とした。

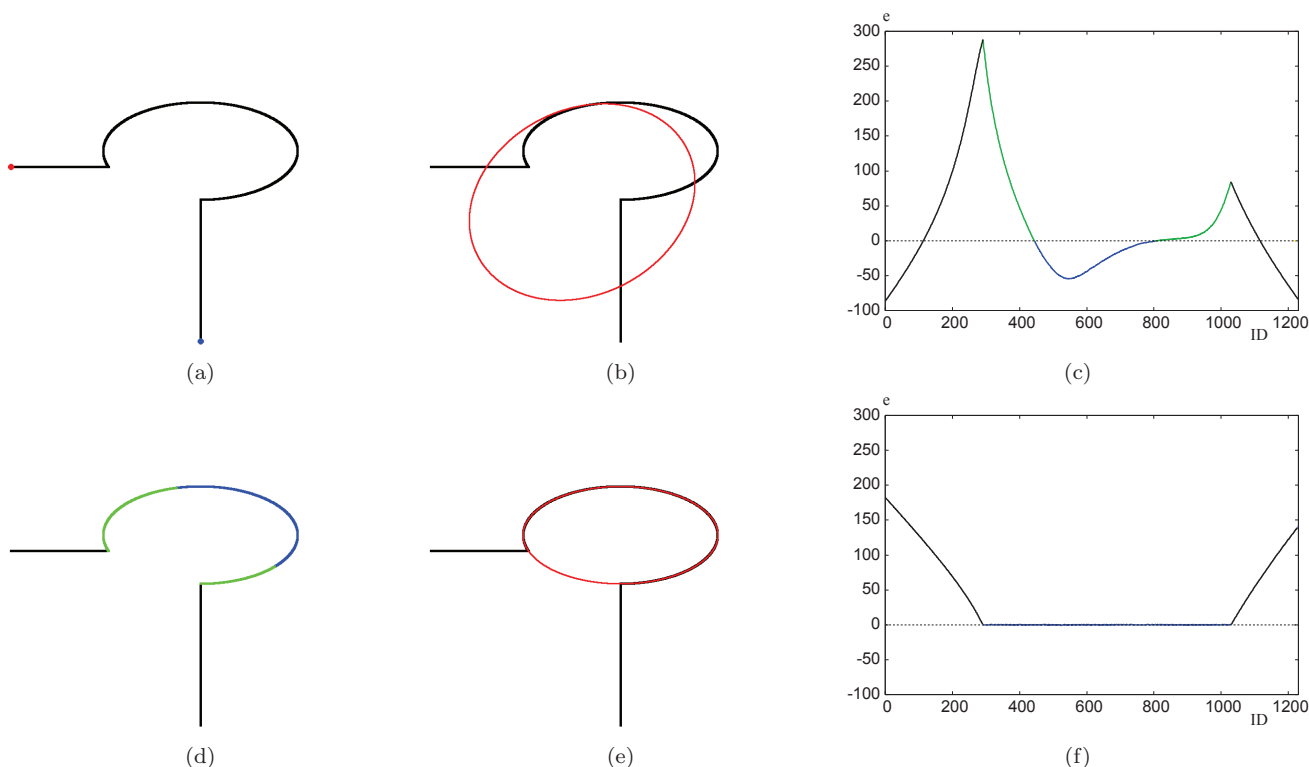


図 4 アウトライアを含む点列への提案手法の適用結果. (a) 入力点列. (b) 入力点列に当てはめた楕円. (c) 当てはめた楕円 (b) に対する各点の誤差のグラフ. (d) 検出したインライア弧. (e) インライア弧 (d) に当てはめた楕円. (f) 当てはめた楕円 (e) に対する各点の誤差のグラフ.

$$\mathbf{x}^{(2)} = \mathbf{Q}_{\alpha^*} - \mathbf{Q}_\gamma, \mathbf{x}^{(1)} = (-x_1^{(2)}, x_2^{(2)})^\top \quad (18)$$

(4) 次の式でベクトル $\mathbf{x}^{(1)}, \mathbf{x}^{(2)}$ のなす角 ϕ を計算する.

$$\phi = \cos^{-1} \left(\frac{(\mathbf{x}^{(1)}, \mathbf{x}^{(2)})}{\|\mathbf{x}^{(1)}\| \|\mathbf{x}^{(2)}\|} \right) \quad (19)$$

(5) ϕ がしきい値 $\hat{\phi}$ よりも小さい場合, その部分弧をアウトライア, そうでない場合にはインライアとする.

提案手法の一番最初のインライア検出では, 上記のアルゴリズムで検出された接部分弧と, それ以外で長さがしきい値よりも大きいインライア弧の中で誤差のピークの角度が最大となる部分弧, その部分弧に隣接する部分弧のうち誤差のピーク点から隣接するインライア弧の端点までを次の楕円当てはめに使用するインライアとする. ただし, インライア弧に隣接する部分弧が接部分弧であった場合は, さらにそれに隣接する部分弧を調べて接部分弧ではない部分弧のピーク点から隣接する接部分弧の端点までをインライアとする.

7. 実験

7.1 提案手法による楕円当てはめの確認

図 4(a) の左図に示す楕円弧以外の点を含む点列に提案手法によって楕円を当てはめる. 図 4(a) のすべての点列に対して Fitzgibbon らの方法で楕円を当てはめた結果が,

図 4(b) である. また, その当てはめた楕円に対する各点の誤差をプロットしたのが図 4(c) である. このグラフでは式 (14) で計算される当てはめた楕円との誤差の値を式 (1) の左辺に当てはめた楕円のパラメータと各点の座標を代入して得られる符号で符号づけしたグラフをプロットしている. インライアとして判定された部分弧を青点で示している. また, このインライア弧の端点から隣接する部分弧のピークまでを検出したインライアとして緑点で示している. アウトライアは黒線で示している. これを入力点列で表したものが図 4(d) である. 1 回目の検出で楕円弧上の点がすべて得られていることがわかる.

2 回目の楕円当てはめでは, 図 4(d) の青点と緑点を入力して楕円を当てはめる. その結果が図 4(e) であり正しい楕円が得られていることがわかる. そして, その当てはめ誤差のグラフが図 4(f) であり, 楕円弧上の点でほぼ誤差が 0 となり, このインライア部分弧とアウトライア部分弧の境界が入力点列の楕円上の弧とアウトライア点列の境界になっている.

7.2 比較実験

提案手法の有効性を確認するために, RANSAC による楕円当てはめと Yu らの方法との比較実験を行った. 図 5 に実験結果を示す. 図 5(a), (b) は提案手法で当てはまった楕円と, その楕円を当てはめるために使用した点列 (青点) で

表 1 計算時間と反復回数の比較. 計算時間 [秒](反復回数)

	提案手法	RANSAC	Yu らの手法
データ (1)	0.004 (5)	0.041 (111)	0.080 (20)
データ (2)	0.008 (5)	0.061 (113)	0.148 (48)
データ (3)	0.012 (7)	0.079 (137)	0.208 (17)
データ (4)	0.004 (3)	0.049 (152)	0.288 (106)
データ (5)	0.028 (6)	0.059 (139)	0.096 (26)
データ (6)	0.008 (4)	0.054 (161)	0.632 (161)
データ (7)	0.004 (4)	0.059 (218)	0.216 (72)
データ (8)	0.004 (6)	0.042 (138)	0.136 (76)
データ (9)	0.008 (9)	0.042 (148)	0.600 (178)
データ (10)	0.004 (3)	0.064 (143)	0.424 (157)
データ (11)	0.008 (8)	0.068 (134)	0.048 (10)

ある. 図 5(c), (d) は RANSAC で当てはまった楕円と, その楕円に対してインライア*⁸と判定された点列である. また, 図 5(e), (f) は Yu らの手法で当てはまった楕円と, その楕円を当てはめるために使用した点列である.

この結果より, RANSAC と提案手法ではほとんどのデータに対して正しい楕円が得られていることがわかる. Yu らの方法もほぼ正しい楕円が得られているが, 楕円当てはめに使用する点が少ないために, 正しい楕円とは若干ずれた楕円が得られているものもある. また, インライアとアウトライアが滑らかに接続している (8), (9) のデータに対しても提案手法で正しい楕円が得られ, インライアとアウトライアを選別できていることが確認できる.

しかし (10) のデータでは, 入力データに対するアウトライアの割合が多いため, RANSAC では正しい楕円が得られていない. それに対して提案手法ではアウトライアの割合が多い場合でも正しい楕円が得られている. また, (11) のデータは提案手法と RANSAC で正しい楕円が得られているが, Yu らの方法では正しい楕円が得られなかった例である.

表 1 はそれぞれの手法の反復回数と計算時間をまとめたものである. 実験環境は, CPU は Intel Core 2Duo 3.00GHz×2, 主メモリ 4GB, OS Ubuntu 12.04 を用いた. RANSAC については, 50 回続けて解が変化しなかったときに収束したと判定して, 収束までの総反復回数を求めた. 乱数により試行のたびに反復回数が変わるため, 同じデータに対して 10 回の試行を行った平均を計算した. この結果より, 提案手法は数回の反復で収束し, 計算時間も高速であることがわかる. それに対して RANSAC では解が収束するまでに多くの反復が必要なことから提案手法と比べて多くの計算時間がかかっている. また Yu らの方法は多くの反復と計算時間がかかることがわかる.

7.3 実画像実験

最後に実画像から抽出したエッジ点列に対して提案手法と RANSAC, Yu らの方法で楕円当てはめした結果を比較

*⁸ 式 (14) の値が 1.5 以下の点をインライアとした.

した. 図 6(a) は入力画像, 図 6(b) は canny オペレータによって抽出したエッジ点列のうち連結した点列の長さが 50 点以上のもをを描画した結果である. その中から赤色で描画した連結点列を選んで, それぞれの手法の入力とした. 得られた楕円当てはめの結果が図 6(c), (d), (e) であり, それぞれ, 提案手法, RANSAC, Yu らの手法の出力である. RANSAC や Yu らの方法では正しい楕円が得られていないのに対して, 提案手法では正しい楕円が得られていることがわかる. また図 6(f) から図 6(j) は提案手法の初期当てはめから反復の過程で当てはまった楕円を示したものである. 徐々に正しい楕円に近づいて最終的に正しい楕円に収束したことが確認できる.

8. まとめ

本論文では画像から抽出した点列に楕円弧以外の点列が含まれていても正しく楕円を当てはめることができる新しい方法を提案した. 提案手法は, 入力点列が連結した点列であることを仮定して, 当てはめた楕円と入力点列の交点で点列を部分弧に分割し, 部分弧ごとにその誤差の変化量によってインライアとアウトライアを判別するものである. 誤差変化の大きさを測る指標として分割した部分弧内の最大誤差を持つ点と, その周辺の点とのなす角を用いた.

アウトライアを含むデータへの楕円当てはめには RANSAC を用いた方法がよく用いられるが, 提案手法は RANSAC と比べて収束までの反復回数が少ないという利点がある. また, RANSAC では入力データに対するアウトライアの割合が大きい場合に, 正しい結果が得られないことがあるが, 提案手法では, アウトライアの割合に影響されずに正しい楕円を当てはめることができる. このことをシミュレーションデータや実画像を用いた実験によって確認した.

謝辞: 本研究の一部は日本学術振興会科学研究費 (基盤研究 (C) No. 26330192) の助成によった. 岡山大学金谷健一教授には有益なご意見を頂きましたことに感謝します.

参考文献

- [1] W. Chojnacki, M. J. Brooks, A. van den Hengel and D. Gawley, On the fitting of surfaces to data with covariances, *IEEE Trans. Patt. Anal. Mach. Intell.*, **22**-11, pp. 1294-1303, 2000.
- [2] M. A. Fischler and R. C. Bolles, Random sample consensus: A paradigm for model fitting with applications to image analysis and automated cartography, *Comm. ACM*, **24**-6, pp. 381-395, 1981.
- [3] A. Fitzgibbon, M. Pilu, and R. B. Fisher, Direct least squares fitting of ellipses, *IEEE Trans. Patt. Anal. Mach. Intell.*, **21**-5 pp. 476-480, 1999.
- [4] R. Hartley and A. Zisserman, *Multiple View Geometry in Computer Vision*, 2nd ed., Cambridge University Press, Cambridge, U.K., 2004.
- [5] 岩元祐輝, プラサンナ・ランガラヤン, 金谷 健一, 楕円当てはめの超精度最小二乗法, 情報処理学会研究報告,

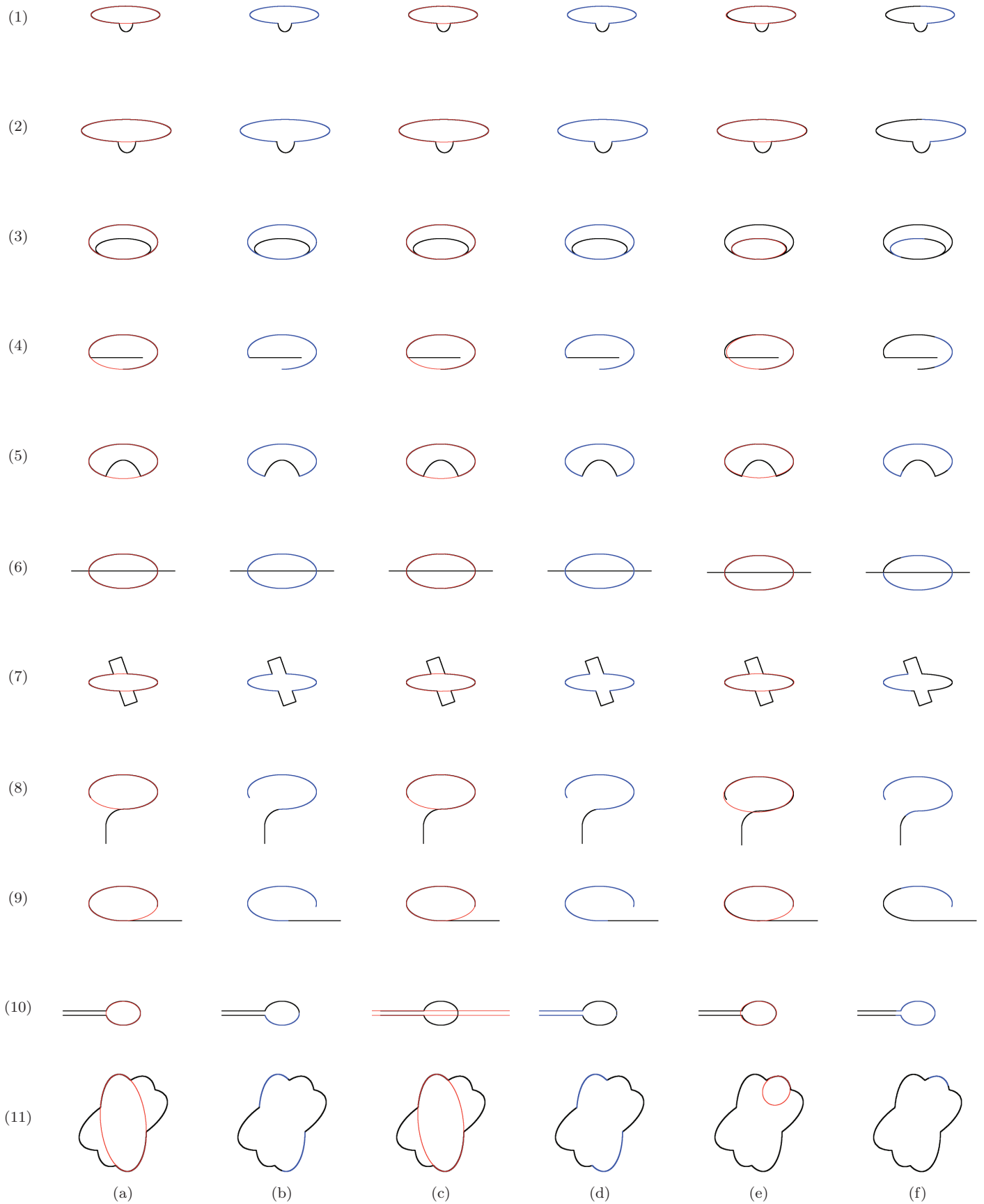


図 5 さまざまなデータへの楕円当てはめの結果の比較. (a) 提案手法で得られた楕円. (b) (a) の楕円当てはめに使用した点列. (c) RANSAC で得られた楕円. (d) RANSAC でインライアと判定された点列. (e) Yu らの方法で得られた楕円. (f) (e) の楕円当てはめに使用した点列.

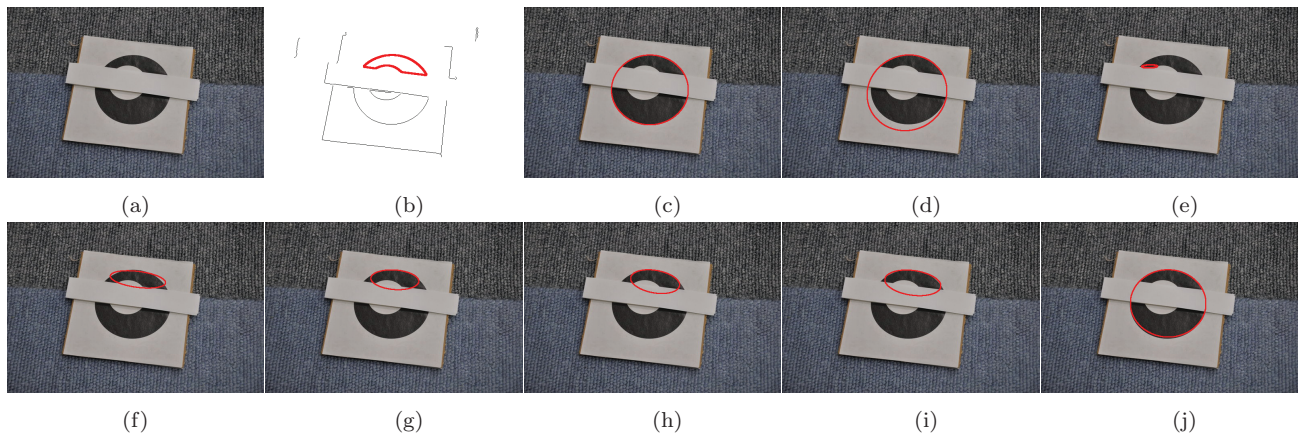


図 6 実画像から抽出したエッジ点列への楕円当てはめ. (a) 入力画像 (b) エッジ画像 (c) 提案手法 (d) RANSAC (e) Yu らの方法 (f)–(j) 提案手法で初期当てはめから各反復で当てはまった楕円.

- 2009-CVIM-168-14, pp. 1–8, 2009.
- [6] K. Kanatani, *Geometric Computation for Machine Vision*, Oxford University Press, Oxford, U.K., 1993.
- [7] K. Kanatani and Y. Sugaya, Performance evaluation of iterative geometric fitting algorithms, *Comput. Stat. Data Anal.*, **52**-2, pp. 1208–1222, 2007.
- [8] 金谷健一, 形状CADと図形の数学, 共立出版, 1998.
- [9] 金谷健一, 菅谷保之, 幾何学的当てはめの厳密な最尤推定の統一的計算法, *情報処理学会論文誌: コンピュータビジョンとイメージメディア*, **2**-1, pp. 53–62, 2009.
- [10] K. Kanatani and Y. Sugaya, Unified computation of strict maximum likelihood for geometric fitting, *J. Math. Imaging Vis.*, **38**-1, pp. 1–13, 2010.
- [11] K. Kanatani, P. Rangarajan, Y. Sugaya and H. Niitsuma, HyperLS and its applications, *IPSJ Trans. Comput. Vis. Appl.*, **3**, pp. 80–94, 2011.
- [12] 金谷健一, アリ・アルシャラドカー, ニコライ・チェルノフ, 菅谷保之, 超精度くりこみ法, *情報処理学会研究報告*, 2012-CVIM-180-23, pp. 1–8, 2012.
- [13] K. Kanatani, A. Al-Sharadqah, N. Chernov, and Y. Sugaya, Renormalization Returns: Hyper-renormalization and Its Applications, *Proc. 12th Euro. Conf. Comput. Vis.*, **3**, pp. 385–398, 2012.
- [14] Y. Leedan and P. Meer, Heteroscedastic regression in computer vision: Problems with bilinear constraint, *Int. J. Comput. Vision*, **37**-2, p. 127–150, 2000.
- [15] T. Masuzaki, Y. Sugaya, and K. Kanatani, High accuracy ellipse-specific fitting, *Image and Video Technology*, pp. 314–324, 2014.
- [16] J. Matei and P. Meer, Estimation of nonlinear errors-in-variables models for computer vision applications, *IEEE Trans. Patt. Anal. Mach. Intell.*, **28**-10, pp. 1537–1552, 2006.
- [17] 中川祐介, 金谷健一, 菅谷保之, 高ノイズレベルにおける最尤楕円当てはめ, *情報処理学会研究報告*, 2008-CVIM-162-10, pp. 53–60, 2008.
- [18] 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*, J85-D-II-12, pp. 1823–1831, 2002.
- [19] Y. Sugaya, Ellipse detection by combining division and model selection based integration of edge points, *4th Pacific-Rim Symposium on Image and Video Technology (PSIVT)*, pp. 64–69, 2010.
- [20] Z. L. Szpak, W. Chojnacki, and A. van den Hengel, Guaranteed ellipse fitting with Sampson distance, *Proc. 12th Euro. Conf. Comput. Vis.*, **5**, pp. 87–100, 2012.
- [21] G. Taubin, 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.*, **13**-11, pp. 1115–1138, 1991.
- [22] 山田 純平, 金谷 健一, 菅谷 保之, 楕円当てはめの高精度計算法とその性能比較, *情報処理学会研究報告*, 2006-CVIM-154-36, pp. 339–346, 2006.
- [23] Yu Qiao, and S. H. Ong, Arc-based evaluation and detection of ellipses, *Journal of Pattern recognition*, **40**-7, pp. 1990–2003, 2007.