

# 標定点自動探索のためのコーナの検出

大西 芳幸\* 服部 進\*\*

Corner Detection for Automatic Finding of Orientation Points

Yoshiyuki OHNISI\* Susumu HATTORI\*\*

## ABSTRACT

For automatic orientation of a pair of stereo aerial images, corner points are preferably used. A corner is defined as an intersection of two linear edges. This paper discusses the features of two methods for corner detection. One is the Canny's algorithm for an edge detection procedure and the other is the Forstner's operator for direct corner detection. The Canny's algorithm is said to be the best algorithm to extract edges under general conditions, and therefore corners must be identified by the second operation. The Forstner's operator is a method to detect corners developed in the field of photogrammetry. They were in this paper briefly outlined from algorithmic point of view and compared through an experiment on some typical segments of a pair of stereo aerial image. The result shows that in complicated natural scene, the Forstner's operator is likely to better extract corners suitable for matching of homogeneous points in stereo images.

キーワード：ステレオ画像，標定点，エッジ検出，コーナ検出，Cannyの方法，Förstnerの演算子，空中写真画像

Keyword: stereo images, orientation point, edge extraction, corner detection, Canny, Förstner's operator, aerial images

## 1. はじめに

空中写真画像や衛星画像から地図を描くこと，すなわち図化はステレオ画像から対象空間を再構成することが原理になっている。図化の最初は1対のステレオ画像の標定から始まる。標定は2枚の画像の相対位置を見出し，実空間の相似モデルをつくる相互標定と，モデルを実空間に一致させる絶対標定（対地標定）にわけられる。ここでは前者を扱う。相互標定のためには，画像上の対応する点を5点以上見出す必要がある。アナログ式の図化機では，オペレータが1モデルあたり10点程度を計測するのが実用上の限界であった。1対の画像のみを処理するのであれば，多数の点が取れるが，地図作成には通常100-1000枚の画像を処理する

からである。一方デジタル画像を使った標定では，1点ごとの計測精度が人間の計測に比べて落ちるかわり，大量の対応点を見つけることが可能である。

計算で対応点を求めるには，画像のコーナを見出して対応付けるのが自然である。コーナは2本以上の直線が交わってできる折れ線部と定義する。したがって，コーナの検出は比較的小さな窓ごとに行なう。

コーナの検出には，エッジ検出演算子を使う方法とコーナ検出演算子を使う方法がある。この論文では両者の代表としてCannyの方法とFörstnerの演算子を比較し，実用性を考察する。

\*大学院情報処理工学専攻

\*\*情報処理工学科

## 2. Canny の方法によるエッジの検出

### \*\*情報処理工学

優れたエッジ検出には、次の3つの条件を満たさねばならない。

(1) エッジのないところではエッジを検出しない、すなわち雑音に鈍感であること、逆にエッジのあるところでは正しく検出すること、すなわち信号に敏感であること。

(2) 検知されたエッジの位置と実際のエッジの位置がずれないこと。

(3) 1つのエッジに対しては1つの反応しかないこと。

Canny の方法では以上の3つの基準が達成されるので優れたエッジ検出法であるとされている。Canny のエッジ検出法は次の順序で行なう。ただし実装法は人によりわずかず異なるようである。下記のアルゴリズムは参考文献[2]による。アルゴリズムの一つ一つの要素は必ずしも新しい考えではない。

[STEP1] 雑音除去を含めたエッジ検出のため、画像全体に Gauss フィルタの微分を掛ける。Gauss フィルタの  $x, y$  方向の微分をそれぞれ

$$G_x(x, y) = \frac{\partial G(x, y)}{\partial x} \quad (1)$$

$$G_y(x, y) = \frac{\partial G(x, y)}{\partial y}$$

とする。ただし

$$G(x, y) = \exp\left(-\frac{x^2 + y^2}{\mu^2}\right) \quad (2)$$

であり、 $\mu$  は Gauss 関数の広がりパラメータである。フィルタ操作の出力、すなわち画素値の勾配をそれぞれ、 $l_x(x, y), l_y(x, y)$  とする。

[STEP2] 画素値の勾配の大きさとその方向を求める。すなわち

$$NVI(x, y) = \sqrt{\{l_x(x, y)\}^2 + \{l_y(x, y)\}^2} \quad (3)$$

$$\cos \theta = l_x / NVI, \sin \theta = l_y / NVI \quad (4)$$

[STEP3] 勾配  $NVI(x, y)$  は画素値の勾配の強度を表しているが、エッジを検出するには細線化が必要になる。通常 Canny の方法では2つのパラメータ

$T1, T2(T1 < T2)$  を設定して、勾配  $NVI(x, y)$  が  $T1$

より小さいものは雑音として除去し、 $T2$  より大きなものは無条件でエッジとする。中間のものはエッジの連結性を調べて判断する。ここでは  $T1$  は  $NVI$  の最小値  $l_{min}$  とし、 $T2 = \infty$  とする。すなわち  $NVI < T1$  となる勾配はエッジではないとして、この勾配の値を  $l_{min}$  に置き換える。この処理の出力を  $lbw(x, y)$  とする。

[STEP4]  $lbw(x, y) > l_{min}$  を満たす各画素で周囲  $3 \times 3$

画素  $Q_1, Q_2, Q_3, Q_4$  を取る。図1と式(4)を参照して、

$P_1(l_x / NVI, l_y / NVI), P_2(-l_x / NVI, -l_y / NVI)$  の2点の勾配の値を周囲4画素の勾配の値から線形内挿して求める。それを  $Z(1), Z(2)$  とする。

$lbw(x, y) > \max(Z(1), Z(2))$  であれば、その画素をエッジと判断する。

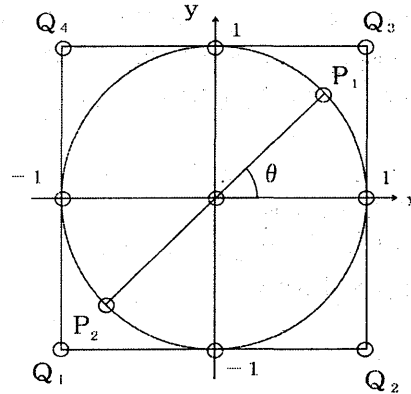


図1 勾配値の内挿によるエッジの判断

Fig. Judgment of a Edge by Interpolation of Intensity Gradients

## 3. Förstner 演算子によるコーナの検出

Förstner 演算子は最初から立体画像の標定のために考え出された方法である[1]。画像を小さく分けたセルごとに処理し、その中にコーナが高々1つあると仮定する。コーナ  $P_0(x_0, y_0)$  があるとすれば、それは2つ以上のエッジの交点である。

各画素P (x, y) で勾配

$$\nabla g = \left( \frac{\partial g}{\partial x}, \frac{\partial g}{\partial y} \right)^T = (g_x, g_y)^T \quad (5)$$

を計算し、この点で引いたエッジがコーナを通過しているとすると、

$$(x - x_0)g_x + (y - y_0)g_y = 0 \quad (6)$$

が成り立つ。

$$g_x = |\nabla g| \cos \theta, \quad g_y = |\nabla g| \sin \theta$$

として式(6)を観測式の形に書くと、

$$v + \cos \theta x_0 + \sin \theta y_0 = \cos \theta x + \sin \theta y \quad (7)$$

を得る。vは観測誤差である。セルの全画素(n個とする)に対して式(7)を立てると

$$v + Ax = e \quad (8)$$

の形の観測式を得る。観測の重みは画像関数の勾配の強さに比例すると考えるのが自然である。すなわち画像関数の勾配が大きい画素はエッジに載っている可能性が高い。そこで重み行列を

$$P = \text{diag}(|\nabla g_i|^2)$$

とすると、結局正規方程式

$$A^T P A x = A^T P e \quad (9)$$

あるいは、

$$\begin{bmatrix} \sum_j g_{xy}^2 & \sum_j g_{xy} g_{yy} \\ \sum_j g_{xy} g_{yy} & \sum_j g_{yy}^2 \end{bmatrix} \begin{bmatrix} x_0 \\ y_0 \end{bmatrix} = \begin{bmatrix} \sum_j (g_{xy}^2 x_j + g_{xy} g_{yy} y_j) \\ \sum_j (g_{xy} g_{yy} x_j + g_{yy}^2 y_j) \end{bmatrix} \quad (10)$$

を得る。式(10)を解くと、コーナ座標の最確値  $\hat{x}$  およ

びその分散共分散行列  $\Sigma_{\hat{x}}$  は次のようになる。

$$\hat{x} = (A^T P A)^{-1} A^T P e \quad (11)$$

$$\left. \begin{aligned} \Sigma_{\hat{x}} &= \hat{\sigma}_0^2 (A^T P A)^{-1} \\ \hat{\sigma}_0^2 &= \frac{1}{n-2} \hat{v}^T P \hat{v} \end{aligned} \right\} \quad (12)$$

$\hat{\sigma}_0^2$  は式(5)の観測誤差の分散の事後推定量である。求めたコーナ座標の分散と共分散は、式(12)の共分散行列

から求められる。その大きさと形を図示するため誤差楕円で表す。すなわち

$$(\mathbf{x} - \hat{\mathbf{x}})^T \Sigma_{\hat{x}}^{-1} (\mathbf{x} - \hat{\mathbf{x}}) = 1 \quad (13)$$

求められたコーナを採用する基準は、誤差楕円の形と大きさである。式(13)から、誤差楕円を正規化するため  $\Sigma_{\hat{x}}^{-1}$  を固有値分解して

$$\Sigma_{\hat{x}}^{-1} = U^T \Lambda^T \Lambda U \quad (14)$$

とする。U は直行行列、 $\Lambda$  は正の特異値を対角要素におく対角行列である。

$$y = \Lambda U (x - x_0) \quad (15)$$

とすると

$$y^T y = 1 \quad (16)$$

となって単位円を得る。 $\Lambda$  を

$$\Lambda = \begin{bmatrix} \lambda_1 & 0 \\ 0 & \lambda_2 \end{bmatrix} \quad (17)$$

で表す。ここで  $\lambda_1 > \lambda_2 > 0$  とする。 $\lambda$  は誤差の標準偏差の逆数に対応するから大きい程良い。したがって、コーナの採用条件は、ある定数  $a$  に対して、

(1)  $\lambda_2 > a$  であること。 $a$  は 0.1 程度になる。

および、直線の交点が鈍角になると誤差楕円が細長くなってコーナは不確定になるので、ある  $r$  に対して

(2)  $(1/\lambda_1)/(1/\lambda_2) = \lambda_2/\lambda_1 = r$  として、誤差楕円

が円に近いためには  $0.5 \leq r \leq 1.0$  が適当である。

## 4. 実験と考察

### 4.1 実験画像

実験に使った画像は 1:10,000 モノクロ空中写真画像の一部である。1画素 1バイト、25 $\mu$ m 幅でスキャンされている。1画素は地上 0.5m に相当する。図 2 に(1)左画像(346画素\*352画素)、(2)右画像(399画素\*361画素)を示す。左右の画像の大きさが違うのは、適当な場所を切り取ったためである。周囲の灌木と芝生、裸地に囲まれた 2 つのビルがコーナ検出の対象である。ビルの屋上と地上との間には最大で約 50 画素の視差があり、標高で 25m に相当する。

### 4.2 Canny の方法によるエッジの検出

図 3 に Canny の方法で検出したエッジを示す。雑音

は抑制され、エッジの検出効果は高い。ただし図 3 からコーナを検出しようとするときの問題がある。

○エッジを追跡してコーナを検出する方法を取るようになるが、細線化の結果エッジはコーナ部で必ずしも連続していない。

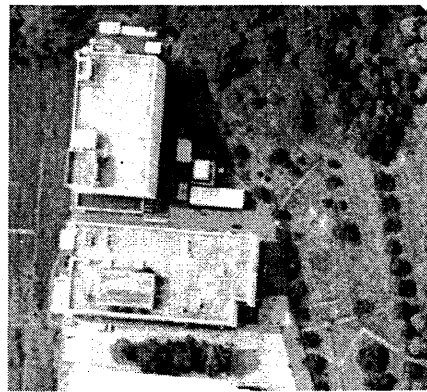
○左右画像とも抽出されるべきコーナが極めて多い。とくに灌木の場所では曲線エッジが大量に出るため、コーナの検出が面倒な上、左右画像のコーナは対応を取るのが困難に見える。例として図 4, 図 5 に灌木の対応部分のエッジを示した。

#### 4. 3 Förstner 演算子によるコーナの検出

図 6 は図 2 から Förstner 演算子でコーナを検出した結果である。  $a = 0.1, r = 0.5$  とした。図 7-9 は図 6 の丸で囲まれた部分を拡大して、検出したエッジをしめた。大きな視差があり、建物の屋上の角は左右画像で対応が取れるが、それ以外には正しい対応が無い。  $a = 0.1$  で固定、図 7-図 9 それぞれで、  $r = 0.3, r = 0.5, r = 0.7$  としたときのコーナを示した。  $r$  が大きくなるにつれ検出されるコーナは当然減少するが、より安定することがわかる。



(1)左画像(346画素 x352画素)  
(1) Left Image(346pixels x352 pixels)



(2)右画像(399画素 x361画素)  
(2) Right Image(399pixels x361pixels)

図 2. 実験に使ったステレオ画像対  
Fig.2 A Pair of Stereo Images

## 5. 結論

Canny の方法では、コーナの検出が難しく、検出できてもそのコーナが本当に正しいのか判断することが難しい。その点 Förstner の演算子なら、正しいコーナかどうか判断することができる。以上の点から Förstner の演算子の方がコーナ検出に適していると考えられる。これからの問題としては、左右像のコーナの対応を取っていくことである。この問題点としては、左右画像には必ずしも同じコーナが現れないことであり、建物の影にコーナが隠れている可能性がある。

## 参考文献

- [1] Förstner W. and E.Guelch: A Fast Operator for Detection and Precise Location of Distinct Points, Corners and Centres of Circular Features, ISPRS Intercommission Workshop, Interlaken, pp.281-305, June 1987
- [2]<http://robotics.eecs.berkeley.edu/~mayi/imgproc/cacode.html>

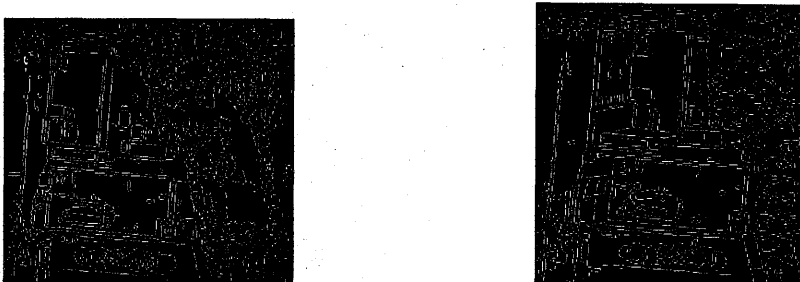


図 3. Canny の方法で検出したエッジ  
Fig.3. Edges detected by the Canny's algorithm

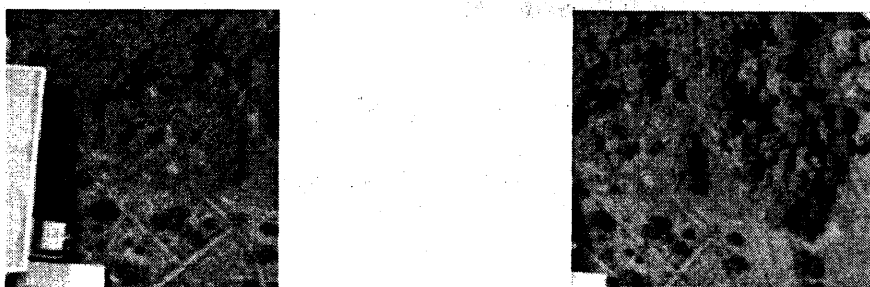


図 4. 対応するステレオ画像の一部 (180 画素 x 180 画素)  
Fig.4. Corresponding portions of the stereo Image (180pixels x 180pixels)



図 5. 図 4 から Canny の方法で検出したエッジ  
Fig.5. Edges detected from Fig.4 by the Canny's Algorithm

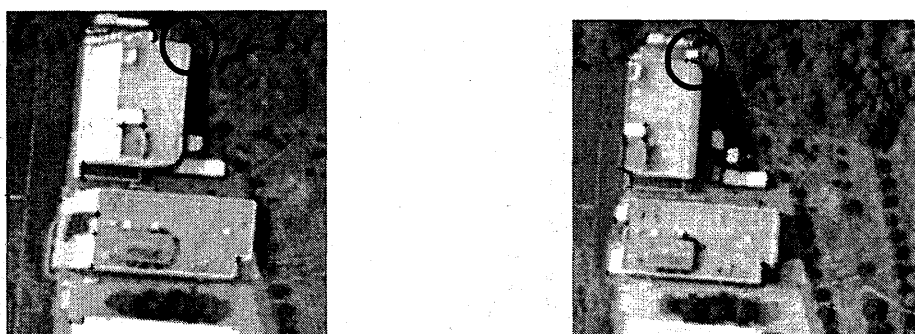


図 6. Förstner 演算子で検出したコーナ ( $a = 0.1, r > 0.5$ )  
Fig.6 Corners detected by the Förstner Operator ( $a = 0.1, r > 0.5$ )

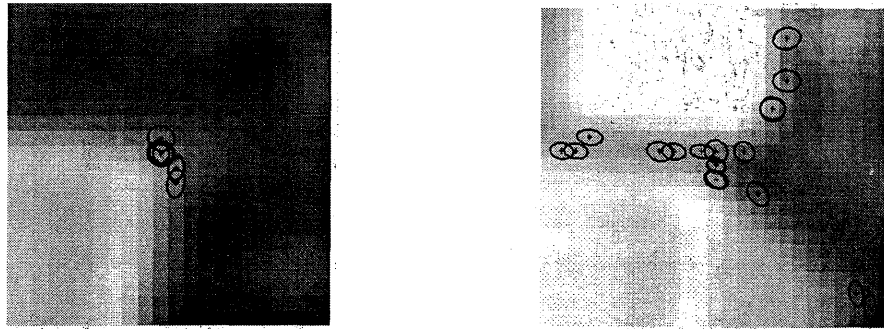


図 7. Förstner 演算子で検出したコーナ (部分拡大  $a = 0.1, r > 0.3$ )

Fig.7. Corners detected by the Förstner operator (partially enlarged:  $a = 0.1, r > 0.3$ )

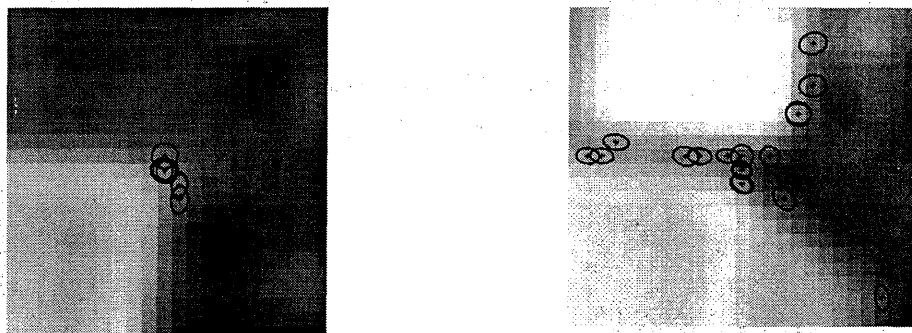


図 8. Förstner 演算子で検出したコーナ (部分拡大  $a = 0.1, r > 0.5$ )

Fig.8. Corners detected by the Förstner operator (partially enlarged:  $a = 0.1, r > 0.5$ )

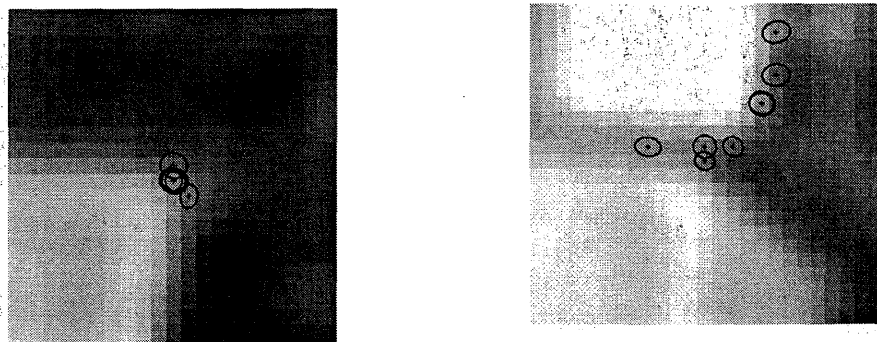


図 9. Förstner 演算子で検出したコーナ (部分拡大  $a = 0.1, r > 0.7$ )

Fig.9. Corners detected by the Förstner operator (partially enlarged:  $a = 0.1, r > 0.7$ )