

ステレオカメラのセルフキャリブレーション

高橋 裕信[†] 富田 文明[‡]

[†]三洋電機筑波研究所 [‡]電子技術総合研究所

ステレオでは正しいカメラパラメータが得られていることを前提として距離測定を行なっている。これらのパラメータは既知のテストパターンを用いたカメラキャリブレーションなどにより事前に計算することができる。しかし、キャリブレーションには誤差が存在し、またキャリブレーション後にいくつかのパラメータが変化する場合も考えられる。その誤差が微小であっても、対応点を求める場合や距離を計算する場合に与える影響は大きい。しかし観測する度にテストパターンをカメラの前に設定し、キャリブレーションをやり直すことは、機械的な輻輳運動によって注視点を変えるステレオカメラや移動ロボットに搭載されるステレオカメラにとっては非現実的である。したがって、未知の観測データから現在のカメラパラメータを自律的に計算することのできるセルフキャリブレーションは、現実のステレオシステムには必要不可欠のものである。そこでステレオ画像の境界線表現に基づいてセルフキャリブレーションを行なう方法を提案し、その実験結果を報告する。

Self-Calibration of Stereo Cameras

Hironobu Takahashi[†] Fumiaki Tomita[‡]

[†] Tsukuba Research Center, Sanyo Electric Company, Ltd.

2-1 Koyadai Tsukuba Ibaraki 305 Japan

[‡] Electrotechnical Laboratory

1-1-4 Umezono Tsukuba Ibaraki 305 Japan

All stereo systems assume that the correct camera parameters are obtained. In fact, those parameters can be calibrated in advance using known test patterns. However, not only there is calibration errors but also some parameters are easy to change after the calibration. Even if the errors may be small, the effects on identifying corresponding points and computing distances are very large. It is infeasible to compute camera parameters every time before observation especially for stereo cameras which change its focus of attention by mechanical convergence or those on mobile robots which observe their environment and acquire depth information. The self-calibration of stereo cameras, which autonomously computes its current camera parameters from unknown observed data, is really necessary for real stereo systems. In this paper, we propose a self-calibration method based on boundary representations of a pair of stereo images.

1 はじめに

ステレオはシーンの距離データを得る最も一般的な方法であり、将来のロボットに共通的な眼として期待される。しかし、ステレオは、いわゆる対応問題に加えて、距離の測定精度と認識など後続する高レベル処理が利用できる出力形式の問題を解決する必要がある。実際、ステレオを使って物体認識まで行なった例は少ない。この意味において、ステレオは能動的センサであるレンジファインダに遅れをとっている。

これまででは、対応問題を解決するために、多くのアルゴリズムが開発されてきた[1,2,3]。その多くは対応の単位として、画像の特徴点、特に、画像の2次微分のゼロクロッシングを用いている。しかし、その実験結果としては視差分布図に留まり、距離測定まで行なっているものは少ない。これは、視差を距離に変換すると誤差や雑音が増幅され使用に堪えないからであると思われる。そこで、最近では、エッジセグメント[4,5]や領域の境界線セグメント[6]を対応の単位として用いる場合が多くなってきている。これは、対応の単位が大きく、かつ多くの情報を含んでいる場合には、対応の探索空間は小さく、曖昧な対応が減少するだけでなく、測定精度や出力形式の点でも優れているからである。

一方、距離測定まで行なっている場合には、当然のことながら正しいカメラパラメータが得られていると仮定している。実際、これらのパラメータは既知のテストパターンを用いて事前に計算することができ、これをカメラキャリブレーションという。しかし、事前にキャリブレーションを行なったとしても、距離測定精度の観点からはまだ不十分である。これは、キャリブレーションには必ず誤差が存在するし、また、キャリブレーション後にいくつかのパラメータが変化してしまうからである。たとえその誤差が微小であったとしても、対応点を求める場合にも、それから距離を計算する場合にも、その影響は大きいのである。かと言って、観測する度にテストパターンをカメラの前に設定し、キャリブレーションをやり直すことは、特に、機械的な輻輳運動によって注視点を変えるステレオカメラや移動ロボット[7,8]に搭載されるステレオカメラにとっては非現実的である。したがって、未知の観測データから現在のカメラパラメータを自律的に計算することのできるセルフキャリブレーションは、現実のステレオシ

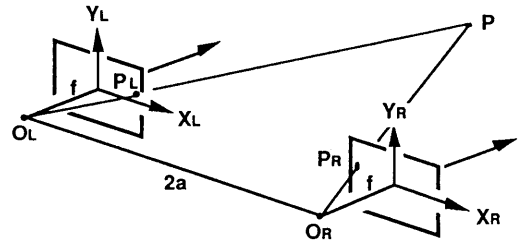


図1 基準ステレオカメラの構成

ステムには必要不可欠のものであると考えられているが、有効な方法は考案されていない。

そこで、本論文では、ステレオ画像の境界線表現に基づいてセルフキャリブレーションを行なう方法を提案し、その実験結果を報告する。

2 カメラキャリブレーション

以下に示すキャリブレーションの方法は、3眼以上の多眼視の場合にも適用することが可能であるが、説明の便宜上、左右に配置された2台のカメラを用いる場合について説明する。

2.1 基準ステレオカメラの構成

基準となるステレオカメラの構成を図1に示す。左右2台のカメラのレンズの中心 O_L と O_R を通る直線をx軸とする。2台のカメラの光軸は、互いに平行であり、x軸に直交している。この光軸の方向をz軸とする。さらに、x軸とz軸に直交する方向をy軸とする。また、カメラの焦点距離を1とし、2台のカメラ間の距離を $2a$ とする。いま、原点を O_L と O_R の中点に置くと、3次元空間上の1点 $P(x, y, z)$ の左右画像への投影像である $P_L(X_L, Y_L)$ 、 $P_R(X_R, Y_R)$ は下記(1)式で与えられる。

$$\begin{cases} \begin{bmatrix} X_L \\ Y_L \end{bmatrix} = \frac{1}{z} \begin{bmatrix} x+a \\ y \end{bmatrix} \\ \begin{bmatrix} X_R \\ Y_R \end{bmatrix} = \frac{1}{z} \begin{bmatrix} x-a \\ y \end{bmatrix} \end{cases} \quad (1)$$

この(1)式から、 $a > 0$ であるので下記(2)式が得られる。

$$\begin{cases} X_L > X_R \\ Y_L = Y_R \end{cases} \quad (2)$$

これは、一方の画像上の1点の他方の画像上での対応点は、エピポラ線である同じ走査線の半直線上に存在することを表す。以下、これをエピポラ条件と呼ぶ。

2. 2 カメラパラメータ

実際のカメラの構成は、基準カメラのようにはなっていないが、そのカメラの特性を表すパラメータがわかれば、そのカメラの画像を基準カメラによって得られる画像に変換することができる。

計測すべきカメラパラメータには、次のように、カメラそのものに依存しない外部パラメータと、個々のカメラに固有の内部パラメータがある。

1) 外部パラメータ

- ・カメラ座標系の原点（レンズ中心）の位置（3自由度）
- ・カメラ座標系の原点まわりの回転角（3自由度）

2) 内部パラメータ

- ・撮像面へのアフィン変換（焦点距離、縦横比、画像の中心、画像面の回転を総合する6自由度）

ただし、アフィン変換のうち、光軸まわりの回転はカメラのz軸まわりの回転と区別できないため、全部の自由度は11となる。

これらのパラメータは、空間中でその位置が既知であるテストパターンをカメラに投影することにより求めることができる（例えば[9]）。

2. 3 パラメータの推定

実際に測定されたパラメータには誤差があり、外部パラメータはキャリブレーション後に変化することは避けられない。特に、回転パラメータの誤差は発生しやすい上に、他のパラメータに比べて、距離の測定精度に非常に大きな影響を与える。そこで、本論文では、カメラの回転パラメータと、カメラにズームレンズを装着する場合を考慮して、その拡大率を未知数として、それらの値を求めることを問題とする。

まず、左右画像の対応点が得られていると仮定した場合（対応点を求める方法については、3章で述べる）に、左右それぞれのカメラに対して、基準カメラからの回転パラメータを推定する方法について述べる。他の方法として、1台のカメラを固定し、他方のカメラの相対的な位置と方向を求める方法[10]もある。しかし、この場合エビボラ線が走査線とは異なるので計算が複雑になる。一方、本論文で述べるように、基準カメラの構成に変換するように両方のカメラの回転角を求める

方法は、走査線をエビボラ線とすることができるので、実際に対応点を求めることが容易になるし、算法も簡略化される利点がある。

いま、基準カメラ座標系での3次元空間上の1点 $p(x, y, z)$ の、実際の左カメラ座標系での座標を $p_L'(x_L', y_L', z_L')$ とし、その画像上での座標値を $P_L'(X_L', Y_L')$ とする。また、実際の左カメラのズームレンズの拡大率を σ_L 、基準カメラからのx、y、z各軸まわりの回転量を $(\theta_L, \phi_L, \alpha_L)$ とする。このとき、点pと点 P_L' の関係は、下記(3)式と(4)式で表される。

$$\begin{bmatrix} x'_L \\ y'_L \\ z'_L \end{bmatrix} = S(\sigma_L) R_x(\theta_L) R_y(\phi_L) R_z(\alpha_L) \begin{bmatrix} x+a \\ y \\ z \end{bmatrix} \quad (3)$$

$$\begin{bmatrix} X'_L \\ Y'_L \end{bmatrix} = \frac{1}{z'_L} \begin{bmatrix} x'_L \\ y'_L \end{bmatrix} \quad (4)$$

ここで、 R_x, R_y, R_z は x、y、z 各軸まわりの回転を表す行列である。

$$\begin{aligned} R_x(\theta_L) &= \begin{pmatrix} 1 & 0 & 0 \\ 0 & \cos \theta_L & -\sin \theta_L \\ 0 & \sin \theta_L & \cos \theta_L \end{pmatrix} \\ R_y(\phi_L) &= \begin{pmatrix} \cos \phi_L & 0 & \sin \phi_L \\ 0 & 1 & 0 \\ -\sin \phi_L & 0 & \cos \phi_L \end{pmatrix} \\ R_z(\alpha_L) &= \begin{pmatrix} \cos \alpha_L & -\sin \alpha_L & 0 \\ \sin \alpha_L & \cos \alpha_L & 0 \\ 0 & 0 & 1 \end{pmatrix} \end{aligned} \quad (5)$$

また、Sは画像の拡大率を表す行列である。

$$S(\sigma_L) = \begin{pmatrix} \sigma_L & 0 & 0 \\ 0 & \sigma_L & 0 \\ 0 & 0 & 1 \end{pmatrix} \quad (6)$$

上記(3)および(4)式から下記(7)式が得られる。

$$\begin{aligned} \begin{bmatrix} x+a \\ y \\ z \end{bmatrix} \\ = S\left(\frac{1}{\sigma_L}\right) R_z(-\alpha_L) R_y(-\phi_L) R_x(-\theta_L) \begin{bmatrix} x'_L \\ y'_L \\ z'_L \end{bmatrix} \end{aligned} \quad (7)$$

この(7)式から両辺を z_L で割って下記(8)式が得られる。

$$\begin{aligned} \frac{1}{z_L} \begin{bmatrix} x+a \\ y \\ z \end{bmatrix} &= S \left(\frac{1}{\alpha_L} \right) R_z(-\alpha_L) R_y(-\phi_L) R_x(-\theta_L) \begin{bmatrix} x'_L/z'_L \\ y'_L/z'_L \\ 1 \end{bmatrix} \\ &= S \left(\frac{1}{\alpha_L} \right) R_z(-\alpha_L) R_y(-\phi_L) R_x(-\theta_L) \begin{bmatrix} X'_L \\ Y'_L \\ 1 \end{bmatrix} \quad (8) \end{aligned}$$

したがって、 y/z を (X'_L, Y'_L) の関数として下記(9)式で表すことができる。

$$\begin{aligned} \frac{y}{z} &= \frac{y/z'_L}{z/z'_L} \\ &= \frac{\begin{bmatrix} S \left(\frac{1}{\alpha_L} \right) R_z(-\alpha_L) R_y(-\phi_L) R_x(-\theta_L) \begin{bmatrix} X'_L \\ Y'_L \\ 1 \end{bmatrix} \end{bmatrix}_y}{\begin{bmatrix} S \left(\frac{1}{\alpha_L} \right) R_z(-\alpha_L) R_y(-\phi_L) R_x(-\theta_L) \begin{bmatrix} X'_L \\ Y'_L \\ 1 \end{bmatrix} \end{bmatrix}_z} \\ &= \frac{\frac{1}{\alpha_L} \begin{bmatrix} R_z(-\alpha_L) R_y(-\phi_L) R_x(-\theta_L) \begin{bmatrix} X'_L \\ Y'_L \\ 1 \end{bmatrix} \end{bmatrix}_y}{\begin{bmatrix} R_z(-\alpha_L) R_y(-\phi_L) R_x(-\theta_L) \begin{bmatrix} X'_L \\ Y'_L \\ 1 \end{bmatrix} \end{bmatrix}_z} \quad (9) \end{aligned}$$

ただし、[式]y と [式]z は式の y 成分と z 成分を表す。

同様に、実際の右カメラについても、添字の L を R に替えることにより下記(10)式が得られる。

$$\begin{aligned} \frac{y}{z} &= \frac{y/z'_R}{z/z'_R} \\ &= \frac{\begin{bmatrix} S \left(\frac{1}{\alpha_R} \right) R_z(-\alpha_R) R_y(-\phi_R) R_x(-\theta_R) \begin{bmatrix} X'_R \\ Y'_R \\ 1 \end{bmatrix} \end{bmatrix}_y}{\begin{bmatrix} S \left(\frac{1}{\alpha_R} \right) R_z(-\alpha_R) R_y(-\phi_R) R_x(-\theta_R) \begin{bmatrix} X'_R \\ Y'_R \\ 1 \end{bmatrix} \end{bmatrix}_z} \\ &= \frac{\frac{1}{\alpha_R} \begin{bmatrix} R_z(-\alpha_R) R_y(-\phi_R) R_x(-\theta_R) \begin{bmatrix} X'_R \\ Y'_R \\ 1 \end{bmatrix} \end{bmatrix}_y}{\begin{bmatrix} R_z(-\alpha_R) R_y(-\phi_R) R_x(-\theta_R) \begin{bmatrix} X'_R \\ Y'_R \\ 1 \end{bmatrix} \end{bmatrix}_z} \quad (10) \end{aligned}$$

ここで、(2)式より

$$Y_L = Y_R = \frac{y}{z}$$

であるので、下記(11)式が成立する。

$$\begin{aligned} &\frac{\frac{1}{\alpha_L} \begin{bmatrix} R_z(-\alpha_L) R_y(-\phi_L) R_x(-\theta_L) \begin{bmatrix} X'_L \\ Y'_L \\ 1 \end{bmatrix} \end{bmatrix}_y}{\begin{bmatrix} R_z(-\alpha_L) R_y(-\phi_L) R_x(-\theta_L) \begin{bmatrix} X'_L \\ Y'_L \\ 1 \end{bmatrix} \end{bmatrix}_z} \\ &= \frac{\frac{1}{\alpha_R} \begin{bmatrix} R_z(-\alpha_R) R_y(-\phi_R) R_x(-\theta_R) \begin{bmatrix} X'_R \\ Y'_R \\ 1 \end{bmatrix} \end{bmatrix}_y}{\begin{bmatrix} R_z(-\alpha_R) R_y(-\phi_R) R_x(-\theta_R) \begin{bmatrix} X'_R \\ Y'_R \\ 1 \end{bmatrix} \end{bmatrix}_z} \quad (11) \end{aligned}$$

その結果、左右の画像間の対応点についてこの式を満たすように、左右カメラの回転 $(\theta_L, \phi_L, \alpha_L)$ 、 $(\theta_R, \phi_R, \alpha_R)$ と拡大率 σ_L 、 σ_R を定めることにより、実際のカメラの基準カメラからの回転量を求めることができる。ただし、 θ_L と θ_R はカメラを結ぶ軸まわりの回転であり、互いに独立ではないため何れか一方の値、あるいはその差のみを求めることができる (カメラが3台以上で一直線上にはない場合には、一意に定めることができる)。また、拡大率 σ_L と σ_R も独立ではないが、両者の比を求めることができる。

ところで、上記(11)式の方程式は解析的に解くことができなため、実際には1次近似式を用いた反復法により解を求める。すなわち、実際のカメラが輻輳によって基準カメラから大きくずれている場合でも、テストパターンを用いる初期のカメラキャリブレーションによって、先に各パラメータを基準カメラに近付けておくことができるので、 $(\theta_L, \phi_L, \alpha_L) \equiv (0, 0, 0)$ 、 $(\theta_R, \phi_R, \alpha_R) \equiv (0, 0, 0)$ と仮定することができる。そこで、(11)式の左辺と右辺の1次近似式の誤差 e を下記(12)式で表すことができる。

$$e = \frac{1}{\alpha_L} \frac{\alpha_L X'_L + Y'_L - \theta_L}{-\phi_L X'_L + \theta_L Y'_L + 1} - \frac{1}{\alpha_R} \frac{\alpha_R X'_R + Y'_R - \theta_R}{-\phi_R X'_R + \theta_R Y'_R + 1} \quad (12)$$

また、 $\lambda = \sigma_R/\sigma_L - 1 \equiv 0$ とし、(12)式の分母を払うことにより誤差 e' を下記(13)式で表す。

$$e' = (\lambda + 1) (\alpha_L X'_L + Y'_L - \theta_L) (-\phi_R X'_R + \theta_R Y'_R + 1) - (\alpha_R X'_R + Y'_R - \theta_R) (-\phi_L X'_L + \theta_L Y'_L + 1) \quad (13)$$

さらに、前述の理由により θ_R を便宜上 0 とし、また 1 次以上の項を削除することにより下記(14)式が得られる。

$$e' \equiv (Y'_L Y'_R + 1)\theta_L + X'_L Y'_R \phi_L + X'_L \alpha_L - X'_R Y'_L \phi_R - X'_R \alpha_R + Y'_L \lambda + (Y'_L - Y'_R) \quad (14)$$

そして、この誤差が最小となるように各パラメータの値を最小 2 乗法によって求めることができる。しかし、その結果得られる解は 1 次近似式の解として得られたものであるため誤差を含んでいる。そこで、左右のカメラのそれぞれについて得られたパラメータの値を用いて再び対応点の誤差を求め、誤差が一定値以下になるまで上記の処理を繰り返す。

2. 4 誤差の収束性

上記のアルゴリズムでは誤差の 1 次近似による反復法を用いているため、その収束性が問題となる。そこで、シミュレーションにより誤差の収束性を調べた。左右のカメラのレンズ中心の座標を $O_L(-20,0,0)$ と $O_R(20,0,0)$ とし、3 次元空間の点として、 $P_1(-50,-50,300)$ 、 $P_2(50,-50,300)$ 、 $P_3(-50,50,300)$ 、 $P_4(50,50,300)$ 、 $P_5(-50,-50,500)$ 、 $P_6(50,50,500)$ の 6 点を用いた。各点の左右画像への投影像を図 2 に示す。そして、一様乱数を用いてカメラの回転パラメータを変化させ、各場合についてパラメータの推定を行なった。その結果を図 3 に示す。水平軸は角度 (ラジアン) の一様分布の最大値を表し、垂直軸は 1000 回の試行のうち正しい値に収束しなかった割合を表す。その結果、角度の変化が 0.35 ラジアン以内の範囲であれば、数回の繰り返しの後、パラメータは常に正しい値に収束することがわかる。また、その誤差は 10^{-6} 以下である。したがって、実際の状況において上記アルゴリズムを用いることには問題がないことがわかる。

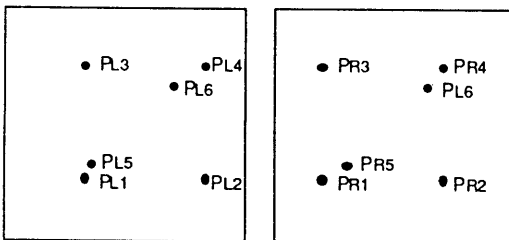


図 2 対応点の合成画像

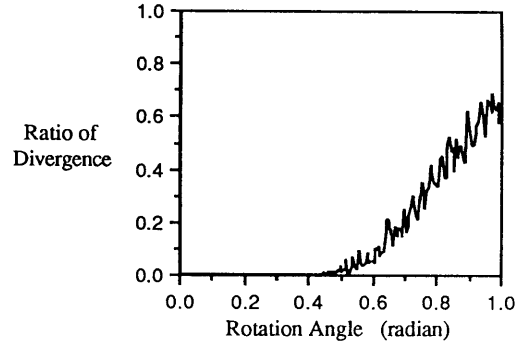


図 3 パラメータの収束

3 ステレオの対応探索

上記のアルゴリズムは、左右の画像で対応点が得られていることを前提としている。したがって、残る問題は、カメラパラメータに誤差が含まれており、エピソード条件が正確には成立していない場合でも、左右の画像で対応点を求めることである。ステレオ画像の境界線表現の境界線セグメントを対応の単位とする方法がこれを可能にする。

3. 1 画像の境界線表現

筆者らは、画像の中間表現として、最も一般的なソリッドモデルである 3 次元の境界線表現 (B-reps) の仕様にしたがって、画像の 2 次元の境界線表現を得る方法を開発した[11]。この表現は、原画像を再構成するのに十分な情報を持つだけでなく、認識を含め後続する各種の高レベル処理によって利用しやすいデータ構造となっている。その概略を述べると、境界線表現では、画像は一様な領域の集合に分割されている。各領域 R は、図 4 のように、外周境界線 B と穴が存在する場合にはその内周境界線 H によって表現される。各境界線は、それを構成する境界線セグメント S の系列によって表現される。各セグメントは、領域を右側に見るように方向付けられており、直線または 2 次曲線によって表現されている。また、各セグ

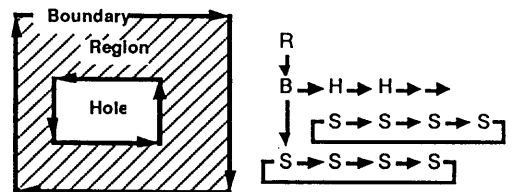


図 4 画像の境界線表現

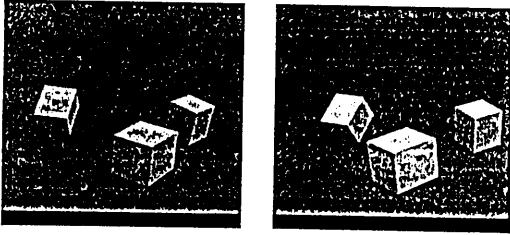


図5 ステレオ画像

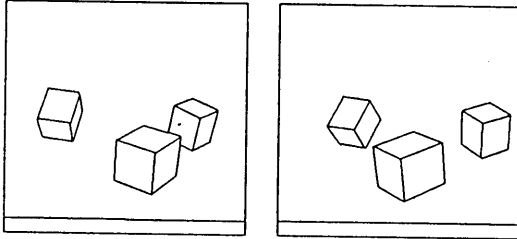


図6 境界線セグメント

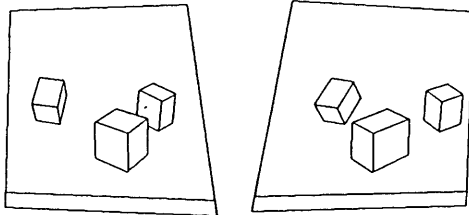


図7 境界線セグメントの座標変換

メントは隣接する領域の境界線を構成するセグメントでもあるので、それを示すポイントを持ち、いわゆるウイングドエッジを形成している。

ステレオの左右画像のそれぞれに対してこの境界線表現を求め、ステレオの対応はこのデータ構造上で行なわれる。例えば、図5のステレオ画像(512×512、8ビット)の境界線表現から得られる境界線セグメントを図6に示す。この場合、ステレオの左右カメラ間の基線の長さは30cmであり、物体はカメラから80cm位の距離にある。

3. 2 カメラの基準化

実際のステレオカメラの構成は、基準のステレオカメラの構成とは異なるのが通常である。そこで、まず、既知のテストパターンを用いて、カメラの外部パラメータと内部パラメータを計算する。そして、得られたカメラパラメータを用いて、左右ステレオ画像の境界線表現のそれぞれを、基準カメラの画像から得られるであろう境界線表現に変換する。この変換は容易で、原画像そのものを変換する必要がない。

例えば、図6の原画像から得られたセグメント

は、図7のセグメントに変換される(その効果がわかるように、画像の枠に対しても同じ変換を施している)。

3. 3 セグメントの対応

セグメントをステレオの対応の単位として用いる。セグメントの対応の条件として、エッジポラ条件だけでなく、領域の明るさ、セグメントの傾き、および、他のセグメントとの連結性を利用する。対応の方法の詳細は文献[6]にあるので、ここではその概略を述べる。

- 1) まず、領域間で対応の候補を求める。対応の必要条件は以下の通りである。
 - ・領域の明るさが類似している。
 - ・領域内にエッジポラ条件を満足する対応部分が存在する。
- 2) 次に、対応する領域の境界線を構成するセグメント間で対応の候補を求める。いま、図8のように、左画像のセグメントを S_j 、右画像のセグメントを S_j' とする。2つのセグメントが対応する必要条件は以下の通りである。
 - ・セグメント上に、 AB と $A'B'$ のように、エッジポラ条件を満足する対応部分が存在する。
 - ・対応部分のベクトル AB と $A'B'$ の方向が類似する。
- 3) 以上の処理は、対応の候補を求める段階であり、多重対応、すなわち、1つのセグメントが同じエッジポラ線上で複数のセグメントと対応する場合もある。そこで、対応するセグメントの空間的な連続性を調べ、その結果得られるセグメントの系列の中で、最も大きくかつその形状が最も類似する系列を選択する。

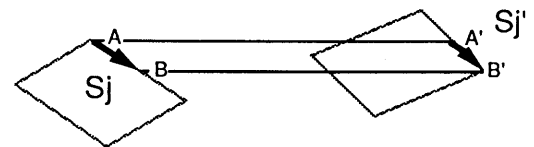


図8 セグメントの対応

以上で本来のステレオの対応は終了するが、セルフキャリブレーションのために、セグメントの対応結果を利用して、さらに点単位の対応を求め

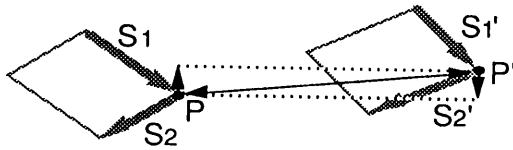


図9 頂点の対応

なければならない。そこで、対応点として、連続する2つのセグメントが形成する頂点を利用する。

3. 4 頂点の対応

いま、図9のように、セグメント S_1 がセグメント S'_1 に対応し、セグメント S_2 がセグメント S'_2 に対応し、 S_1 と S_2 は頂点 P を形成し、 S'_1 と S'_2 は頂点 P' を形成している場合を考える。このとき、点 P と点 P' はエピポーラ条件を満足してはいないが、対応しているものと考えることができる。

以上のように、セグメントを対応の単位として利用する最大の利点は、初期のキャリブレーションに誤差があってエピポーラ条件が厳密には満足されず、その対応点の y 座標が一致しない場合でも、正しいセグメントの対応を得ることができることである。その結果、連続するセグメントが構成する頂点を左右画像の対応点として安定して求めることができる。通常の点単位の対応法では、エピポーラ条件に強く依存するので、それが不正確であると正しい対応を求めることが非常に難しくなる。

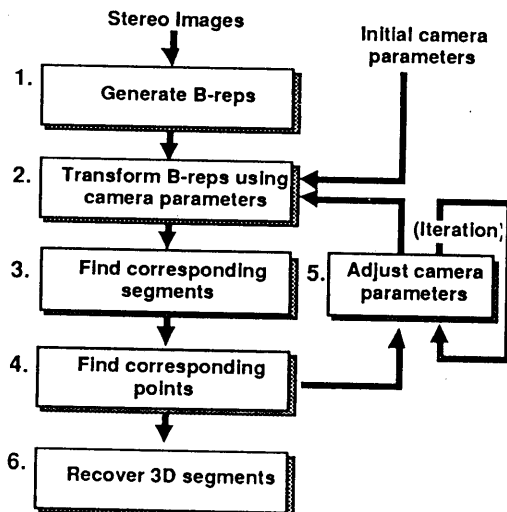


図10 処理の流れ

4 ステレオシステム

セルフキャリブレーションを行なうステレオシステムにおける処理の流れを図10に示す。事前に既知のテストパターンを用いて初期のカメラパラメータが測定されているものとして、以下の処理を行なう。

- 1) まず、ステレオカメラから入力された画像の境界線表現を作成する。
- 2) 現在のカメラパラメータにしたがって、画像の境界線表現を基準カメラの画像の境界線表現に変換する。
- 3) セグメントの対応を求める。
- 4) 頂点の対応を求める。そして、対応する頂点の y 座標の誤差を測定し、誤差が小さい場合は6)に進む。誤差が大きい場合は5)に進む。
- 5) 誤差を最小にするように、反復法によりカメラパラメータの推定を行なう。そして、2)に戻る。
- 6) 対応するセグメントを用いて、3次元のセグメントを構成する。

セルフキャリブレーションは、以上のように、誤差が小さくなるまで繰り返されるが、対応を求める際に利用するエピポーラ条件は、最初は緩く適用し、誤差が小さくなるにしたがって、次第に厳密に適用する。

5 実験結果

セルフキャリブレーションの実験として、初期のカメラキャリブレーションを行なった後、両方のカメラを意図的に任意に回転させた。したがって、画像の境界線表現を、カメラパラメータを用いて基準カメラの画像の境界線表現に変換したと思っても、その対応点の y 座標には誤差が生じることになる。その結果を図11に示す。矢印は対応点の y 座標の誤差ベクトルを表す。平均誤差は20.21画素である。この場合、通常の状態より誤差が大きいため、誤対応も生じている。

これに対してセルフキャリブレーションを行なった結果を図12に示す。誤差を表す矢印は10倍に拡大されている。平均誤差は0.48画素で、勿論誤対応はない。

最後に、対応するセグメントの視差から3次元

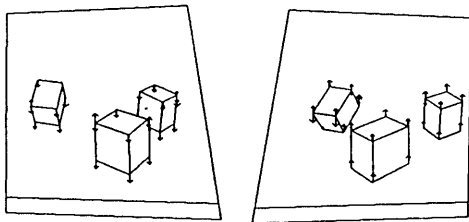


図 1.1 対応点の誤差ベクトル

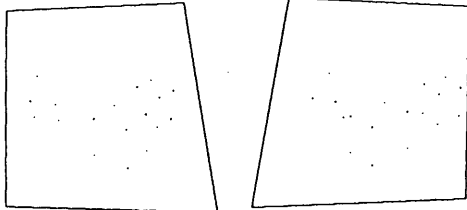


図 1.2 補正後の誤差ベクトル

のセグメントを計算した結果を図 1.3 に示す。

6 おわりに

本論文では、ステレオカメラのキャリブレーションの方法として、従来のように既知のテストパターンを用いることなく、観測している未知のシーンの特徴点を用いて、自律的にカメラパラメータを推定するセルフキャリブレーションの方法を提案した。ステレオの対応の探索には、画像の境界線表現を利用した。その利点は、(1)画像の境界線表現を基準カメラの画像の境界線表現に変換することが容易であること、(2)キャリブレーションの誤差によりエピソード条件が厳密には成立しない場合でも、対応するセグメントを求めることができること、そして、(3)キャリブレーションに必要な対応点を、空間的に連続して対応するセグメントが形成する頂点の対応によって容易かつ正確に求めることができることである。

その結果、これまでのようにカメラパラメータを正確に測定しなければならなかった負担から解放されるだけでなく、以前に比べて飛躍的に（おそらくは、究極的に）正確な距離測定精度を得ることが可能になる。セルフキャリブレーションは、すべてのステレオシステムにとって有益なものとなるだろう。

謝辞

日頃有益な助言と討論をいただき、電子技術総合研究所視覚情報研究室大島正毅室長はじめ研究室の皆様へ感謝致します。

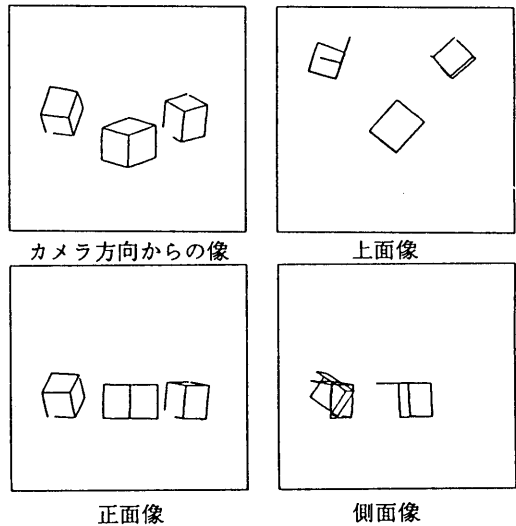


図 1.3 3次元セグメントの正射影像

参考文献

- [1]Barnard,S.T. and Fischler,M.A.: Computational Stereo, Computing Surveys, Vol.14, No.4, pp.553-572 (1982).
- [2]Grimson, W.E.L., Computational Experiments with a Feature Based Stereo Algorithm, IEEE Trans., PAMI-7, 1,pp.17-34, 1985.
- [3]Ohta, Y. and Kanade, T., Stereo by Intra- and Inter-Scanline Search Using Dynamic Programming, IEEE Trans., PAMI-7, 2, pp.139-154, 1985.
- [4]Medioni, G.G. and Nevatia, R., Segment-Based Stereo Matching, Proc. Image Understanding Workshop, pp.128-136, 1983.
- [5]Ayache, N. and Lustman, F., Trinocular Stereovision: Recent Results, Proc. 10th IJCAI, pp.826-828, 1987.
- [6]富田,高橋: ステレオ画像の境界線表現に基づく対応,信学論, Vol.J71-D, No.6, pp.1074-1082 (1988).
- [7]Ayache, N. and Faugeras, O.D., Building a Consistent 3D Representation of a Mobile Robot Environment by Combining Multiple Stereo Views, Proc. 10th IJCAI, pp.808-810, 1987.
- [8]Tsuji, S. and Zheng, J.Y., Visual Path Planning by a Mobile Robot, Proc. 10th IJCAI, pp.1127-1130, 1987.
- [9]Yakimovsky, Y.: A System for Extracting Three-Dimensional Measurements from a Stereo Pair of TV Cameras, Computer Graphics and Image Processing, Vol.7, pp.195-210 (1978).
- [10]Gennery,D.B.: Stereo-Camera Calibration, Proc. Image Understanding Workshop, pp.101-107 (1979).
- [11]富田,高橋: 画像のB-repのためのアルゴリズム, 信学会PRU研資, 86-87 (1987).