

消失点検出における四面体体積最小化法の提案

Proposal of Volume Minimization of Tetrahedrons for Vanishing Point Detection

鈴木 卓[†]
Taku Suzuki海林 健二[†]
Kenji Shoji外山 史[†]
Fubito Toyama宮道 壽一[†]
Juichi Miyamichi

1. まえがき

近年、コンピュータビジョンの分野で三次元形状の計測や復元に関する研究が盛んに行われている。その中の1つとして、消失点に関する研究が挙げられる。一般に、三次元空間中にある平行な直線群が二次元平面に投影されると、それらがある1点で交わる。これを消失点という。しかし、三次元空間中で平行な直線群を二次元平面に投影したとき、二次元平面でも直線が平行で消失点位置が無限遠となる場合がある。本論文ではBarnard[1]によって提案されたガウス球に基づく手法[2][3]と三角形面積最小化法(TAM)のような最小二乗法を用いた手法[4]の利点を組み合わせた四面体体積最小化法を提案する。提案手法では、ガウス球面上で消失点を扱うため無限遠の消失点が存在する場合にも適用が可能で、尚且つ最小二乗法で最適化を行うため高精度の消失点検出が可能である。

消失点検出の検出フローは大きく3つに分けることができる。第1に画像中から線分を検出する工程、第2に検出した線分がどの消失点に収束するのかを判別する工程(線分クラスタリング)、第3に分類したクラスターごとに消失点の位置を計算する工程である。本論文では、線分検出に濃度勾配を用いた手法を適用し、線分クラスタリングでは、RANSAC[5]に基づくクラスタリングアルゴリズムを適用している。

2. 四面体体積最小化による消失点位置計算

本論文では図1に示すような $x-y-z$ 座標系を用い、投影中心 O を原点とし、座標平面を $z = -f$ に置く。また、ガウス球と呼ばれる半径が1の単位球を中心が原点 O に来るように配置し、消失点をガウス球面上の点で表現する。今、消失点が画像平面上で $(x_{vp}, y_{vp}, -f)$ に位置しているとすると、ガウス球面上の消失点の位置 (v_x, v_y, v_z) は以下のように表せる。

$$\begin{aligned} v_x &= x_{vp} / \sqrt{x_{vp}^2 + y_{vp}^2 + f^2} \\ v_y &= y_{vp} / \sqrt{x_{vp}^2 + y_{vp}^2 + f^2} \\ v_z &= -f / \sqrt{x_{vp}^2 + y_{vp}^2 + f^2} \end{aligned} \quad (1)$$

逆に、ガウス球面上の消失点 (v_x, v_y, v_z) は画像平面上では、

$$\begin{aligned} x_{vp} &= -f \cdot u_x / u_z \quad (u_z \neq 0) \\ y_{vp} &= -f \cdot u_y / u_z \quad (u_z \neq 0) \end{aligned} \quad (2)$$

のように表すことができ、 $u_z = 0$ のときは画像平面上の消失点が (u_x, u_y) 方向の無限遠にある。

[†]宇都宮大学大学院 工学研究科

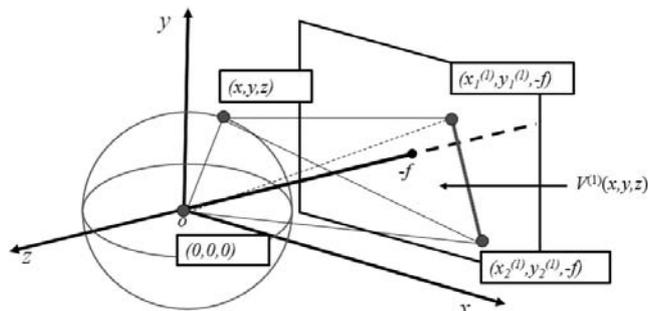


図1: 原点, ガウス球面上の点, 線分の始点と終点の4点で構成される四面体

ある消失点に収束する N 本の線分集合の i 番目の線分の始点と終点を $(x_1^{(i)}, y_1^{(i)}, -f)$ と $(x_2^{(i)}, y_2^{(i)}, -f)$ のように表すと、ガウス球面上の消失点 (v_x, v_y, v_z) は以下の通りに計算される。まず、図1に示すようにガウス球の中心 O 、ガウス球面上の1点 (x, y, z) 、画像平面上の線分の始点と終点 $(x_1^{(i)}, y_1^{(i)}, -f)$ 、 $(x_2^{(i)}, y_2^{(i)}, -f)$ の4点で構成される四面体を考える。

図1の四面体の体積 $V^{(i)}(x, y, z)$ は式(3)のように表される。

$$V^{(i)}(x, y, z) = \frac{1}{6} \begin{vmatrix} x_1^{(i)} & y_1^{(i)} & -f & 1 \\ x_2^{(i)} & y_2^{(i)} & -f & 1 \\ x & y & z & 1 \\ 0 & 0 & 0 & 1 \end{vmatrix} \quad (3)$$

もし、ガウス球面上の1点 (x, y, z) がガウス球の中心と画像平面上の線分の始点と終点の3点を通る平面上に存在する場合、四面体の体積 $V^{(i)}(x, y, z)$ は0となる。理想的には、すべての i について体積 $V^{(i)}(x, y, z)$ が0となるような点 (x, y, z) がガウス球面上の消失点である。しかし、現実にはノイズなど影響によって体積 $V^{(i)}(x, y, z)$ が0となることは少ない。そこで提案手法では、ガウス球面上の点 (x, y, z) の評価関数 $S(x, y, z)$ は、 $V^{(i)}(x, y, z)$ が符号付きの値であるため、すべての i についての体積 $V^{(i)}(x, y, z)$ の2乗の総和とした。

$$S(x, y, z) = \sum_{i=1}^N (V^{(i)}(x, y, z))^2 \quad (4)$$

解くべき問題は、点 (x, y, z) をガウス球面上におくための条件 $x^2 + y^2 + z^2 = 1$ の下で評価関数 $S(x, y, z)$

を $S'(x, y, z)$ として書き直し,

$$\begin{aligned} S'(x, y, z) &= \sum_{i=1}^N (V^{(i)}(x, y, z))^2 \\ &\quad - \lambda(x^2 + y^2 + z^2 - 1) \\ &= \sum_{i=1}^N (a_i x + b_i y + c_i z)^2 \\ &\quad - \lambda(x^2 + y^2 + z^2 - 1) \end{aligned} \quad (5)$$

のように表す. ここで,

$$\begin{aligned} a_i &= (y_2^{(i)} - y_1^{(i)})f/6 \\ b_i &= (x_1^{(i)} - x_2^{(i)})f/6 \\ c_i &= (x_1^{(i)}y_2^{(i)} - x_2^{(i)}y_1^{(i)})/6 \end{aligned} \quad (6)$$

である. この評価関数 $S'(x, y, z)$ を x, y, z についてそれぞれ偏微分し, それらを 0 にすると

$$\begin{bmatrix} \sum_i a_i^2 & \sum_i a_i b_i & \sum_i c_i a_i \\ \sum_i a_i b_i & \sum_i b_i^2 & \sum_i b_i c_i \\ \sum_i c_i a_i & \sum_i b_i c_i & \sum_i c_i^2 \end{bmatrix} \begin{bmatrix} x \\ y \\ z \end{bmatrix} - \lambda \begin{bmatrix} x \\ y \\ z \end{bmatrix} = 0 \quad (7)$$

となり, 3×3 の行列の固有値と固有ベクトルを求める問題となる. 式 (7) を解くと, 3 個の固有ベクトルが得られる. それぞれのベクトルの大きさを 1 に正規化して, それらの中から評価関数 $S(x, y, z)$ が最少になるものを選ぶことによってガウス球面上の消失点を得る. この手法は, 線分でなく直線に対しても適用することが可能である. 直線の場合には, 画像上の各直線とガウス球の中心によって構成される各平面からの距離の 2 乗の和が最小になるようなガウス球面上の点を求めることで, 消失点を得ることになる.

提案する四面体体積最小化法は下記のように解釈することもできる. 原点とある消失点に収束する線分によって定まる平面 (Barnard は Interpretation Plane と呼ぶ.) の集合は, 消失点に向かうある 1 本の直線と交差する. $p_1^{(i)}$ を原点から線分 i の始点 $(x_1^{(i)}, y_1^{(i)}, -f)$ へ向かうベクトル, $p_2^{(i)}$ を原点から線分 i の終点 $(x_2^{(i)}, y_2^{(i)}, -f)$ に向かうベクトルとする. ベクトル $p_1^{(i)}$ と $p_2^{(i)}$ によって定まる平面の法線ベクトルと向きが一致し, 大きさがその線分の長さに比例するようなベクトル $q^{(i)}$ を考える. ベクトル $q^{(i)}$ は

$$q^{(i)} = p_1^{(i)} \times p_2^{(i)} \quad (8)$$

のようにして求めることができる. すべての i についてベクトル $q^{(i)}$ は, 消失点に向かう直線に垂直な平面上に分布する. 従って, すべて i についてのベクトル $q^{(i)}$ の共分散行列から最小固有値に対応する固有ベクトルを求めることによって原点から消失点に向かうベクトルを得ることができる. このベクトル $q^{(i)}$ は式 (6) の (a_i, b_i, c_i) に他ならない. そして, その共分散行列はちょうど式 (7) の 3×3 の行列となる.

3. 線分クラスタリングアルゴリズム

論文で提案する四面体体積最小化法は消失点位置の計算と同時に線分クラスタリングを行う. 消失点位置の計算では消失点 (x, y, z) は評価関数 $S(x, y, z)$ によって評価されるため, その評価値が小さくなるような線

分集合を見つけることによって線分クラスタリングを行う. クラスタリングの対象となる線分集合は, 画像から線分を検出したとき, 互いに同一直線上にある線分には同じラベル, 異なる直線上にある線分には異なるラベルが付けられるものとする. 以下に線分集合から 3 個の消失点を求めるクラスタリングアルゴリズムを説明する.

本クラスタリングアルゴリズムでは RANSAC を用いる. RANSAC とは, ランダムサンプリングに基づくロバストなモデル作成方法であり, 多くの研究に適用されているアルゴリズムである. クラスタリングアルゴリズムでは, まず検出したすべての線分集合を L とし, その中から互いに異なるラベルの (同一直線上でない) 線分をランダムに 3 本選ぶ. 選んだ 3 本の線分について評価関数 $S(x, y, z)$ を最小化して, 消失点 vp_3 と, そのときの線分 1 本あたりの四面体体積の 2 乗 ev_3 を評価値として得る. その評価値 ev_3 が閾値 th_1 以下なら, この 3 本の線分を初期クラスタ C とする. 残りの各線分について, 原点と vp_3 と線分両端点から構成される四面体体積の 2 乗を評価値として, その値が閾値 th_1 以下のとき, その線分をクラスタ C に追加する. ただし, クラスタ中の線分のラベルと一致する線分は追加されないものとする. 残りの各線分についての評価が終了したら, クラスタ C 中の線分数をそのクラスタの評価値とし, 初期クラスタをランダムに生成するところからこれを反復し, 線分数が最も多いクラスタ C_{opt} を求める. そして, そのクラスタ C_{opt} の全線分を用いて評価関数 $S(x, y, z)$ を最小化し, 消失点 vp を得る.

最後に, 別の消失点を求めるための準備として, L から C_{opt} を除外し, さらに, 残った L 中の線分から, その延長線が消失点 vp の近くを通るものを除外する. 具体的には, 原点と vp と線分両端点から構成される四面体体積の 2 乗を評価値として, その値が閾値 th_2 以下となる線分を除外する. ここで, 閾値 th_2 は, $th_2 > th_1$ となるように設定する. 上記の除外後の L を用いて, 再度, 別の消失点位置の計算を行う.

このようにクラスタリングアルゴリズムでは 2 つの閾値 th_1 と th_2 を用いている. th_1 は最初に線分を 3 本選ぶときの閾値と線分をクラスタに追加するときの閾値で, th_2 は最後に不要な線分を除外するときの閾値である. th_1 は, 精度良く消失点の位置を計算するために厳しく設定し, th_2 は, その後の消失点位置の計算に影響が無いように, 既に得た消失点の方向に向かっていると判断できる線分をすべて除外するために大きい閾値を設定している. 閾値は線分 1 本につきどれだけの誤差 (四面体体積の 2 乗) を許容するかで設定した. また, 実験では 500 回連続して最大線分数が更新されないことを終了条件とした. 以下にクラスタリングの手順を示す.

- (1) 濃度勾配法により検出した線分集合を L とする.
- (2) 検出済み消失点数が 3 になるまで, または, L 中の互いに異なるラベルの線分数が 3 未満にな

るまで、(2.1)~(2.5)を繰り返す。

- (2.1) 最良線分集合 C_{opt} を空にする。
- (2.2) 最良解更新がない状態が500回に達するまで(2.2.1)~(2.2.3)を繰り返す。
- (2.2.1) L からランダムに選択した相異なるラベルの3本の線分集合を C とする。 C から四面体体積最小化により消失点 vp_3 を求める。そのときの線分1本あたりの四面体体積の2乗を ev_3 とする。
- (2.2.2) ev_3 が閾値 th_1 より小さいとき、 $\{L - C\}$ の線分集合から1本ずつ線分 i を取り出す。線分 i と vp_3 と原点で定まる四面体体積の2乗が閾値 th_1 より小さく、線分 i のラベルが C 中の線分のラベルと異なるとき、 C に線分 i を追加する。
- (2.2.3) C の線分数が C_{opt} の線分数より大きいとき、 C_{opt} に C を代入することで最良解更新を行う。
- (2.3) C_{opt} が空ならば終了。
- (2.4) C_{opt} から四面体体積最小化により消失点 vp を求める。
- (2.5) L から C_{opt} を除く。残った L 中の各線分 i について、線分 i と vp と原点で定まる四面体体積の2乗が閾値 th_2 より小さいとき、 L から線分 i を除外する。

線分クラスタを評価値ではなく線分数で評価する理由は、あらゆる方向からより多くの線分が収束する点が消失点である可能性が高いという考え方からである。一方、短い線分のほうが体積が小さくなるため線分数が多くなり選ばれやすい。そこで、線分検出の際にある長さ以上の線分だけを検出することでこのような問題を軽減した。

4. 実験と考察

提案手法の有効性を確認するために行った2つの実験について説明する。まず実験1では、消失点が既知であるシミュレーションデータを用いて従来手法と提案手法の消失点検出の精度の比較を行った。従来手法としては、最小二乗法を用いて高精度の消失点検出が可能な三角形面積最小化法に基づく手法(従来手法1)と、画像平面上の直線をガウス球面上の投票空間に投影するガウス球に基づく手法(従来手法2)を用いた。次に実験2では、直方体形状の建築物が写った実画像に対して提案手法を適用し、消失点を検出する実験を行った。焦点距離 f は実験に用いた画像のサイズ 640×480 に対して一般的な値 $f = 700$ とし実験を行った。

4.1. 実験1

実験1では消失点が既知であるシミュレーションデータを用いて従来手法と提案手法の消失点検出の精度の比較をするために、従来手法と提案手法で得られた消失点位置を正解の消失点位置と比較し、原点からガウス球面上の消失点に向かうベクトルの角度のずれで評価した。従来手法では、消失点は画像平面上の座標で

求められるため、これをガウス球面上の座標に変換して比較した。シミュレーションデータは、 640×480 画素の画像内の1点の実数値座標を乱数により生成し、その1点から設定した消失点位置に向かって線分をのばした。線分の長さは平均500、標準偏差50の正規乱数で生成した。線分の本数は各消失点に向かう線分を10本から30本の間でランダムに決定した。シミュレーションデータは線分両端点座標が誤差なしと誤差ありの2種類とした。線分両端点誤差なしのデータは両端点の $x-y$ 座標を実数値で与えた。線分両端誤差ありのデータは両端点 $x-y$ 座標に対して標準偏差2.5の正規乱数でノイズを加えて作成した。さらに、生成する線分の種類で3つに分け、それぞれシミュレーションデータ1、シミュレーションデータ2、シミュレーションデータ3とした。シミュレーションデータ1は各消失点に向かう線分のみを含み、シミュレーションデータ2は各消失点に向かう線分に加えてどの消失点にも向かわないノイズ線分を含むようにした。また、無限遠の消失点に向かう平行な線分を含む線分端点誤差なしのデータをシミュレーションデータ3として生成した。表1に実験結果を示す。表1には、各シミュレーションデータから求めた消失点位置と正解の消失点位置のずれを原点からガウス球面上の消失点に向かう両ベクトル間の角度で評価し、その角度誤差の平均値を示す。表1を見ると従来手法、提案手法とも正解値との角度誤差が 0.1° 程度で、同程度の精度で検出できていることが分かる。しかし、従来手法1は平行な線分を含むシミュレーションデータ3で消失点を検出できなかった。また、従来手法2は他の手法に比べ線分端点誤差ありのシミュレーションデータにおける角度誤差が大きくなる。以上の実験結果より、提案手法は従来手法と同程度の消失点検精度を持ち合わせていること、平行な線分を含むデータであっても例外なく扱えることを確認した。

表1: 計算された消失点位置の角度誤差の平均 ($^\circ$)

	シミュレーションデータ1		シミュレーションデータ2	
	線分端点誤差なし	線分端点誤差あり	線分端点誤差なし	線分端点誤差あり
従来手法1	0.01以下	0.09	0.01以下	0.10
従来手法2	0.01以下	0.11	0.01以下	0.12
提案手法	0.01以下	0.09	0.01以下	0.09
	シミュレーションデータ3			
従来手法1	検出できない			
従来手法2	0.01以下			
提案手法	0.01以下			

4.2. 実験2

実験2では、提案手法の有効性を確認するために直方体形状の建築物が写った実画像40枚に対して提案手法を適用し、40枚中何枚の画像に対して適切に3個の

消失点を検出できるかを、原点から検出した各消失点にのびる線分がそれぞれ直交するかどうかによって評価した。パラメータとして一般的なカメラの焦点距離の値 $f = 700$ を与えている点、線分検出の際にノイズが発生する点を考慮し、本実験では求められた2つの消失点と原点からなる角度が 75° から 105° の場合に正しい消失点が求められたと判断する。また、線分検出は、画像中にある樹木等のエッジの影響をうけにくくするために、濃度勾配により重みを付けて投票を行うハフ変換により直線検出を行い、検出した直線上の画素濃度勾配の値が閾値以上で、一定画素数連続している部分を線分として抽出した。実験では、直線検出の閾値を最大累積値の $1/10$ に設定し、検出した直線に対して、その直線上で濃度勾配の値が32以上の画素が連続して60画素以上続いている部分を線分として抜き出した。閾値設定は、実験に用いた画像内で共通して良好な線分を検出できる値とした。表2に実画像実験の結果を示す。40枚中19枚の画像に対して3個の消失点が求められているが、14枚が消失点を2個までしか検出できていない。これは残り1つの消失点に向かっている線分が少ないために、消失点ではない点に収束してしまい、誤った消失点が検出されてしまうためである。また、7枚の画像に対しては消失点を1個までしか検出できていないが、これは実験に用いた実画像が評価値よりも大きなノイズを含んでしまっていたためである。図2に消失点を検出した画像の例を示す。画像に描かれている色分けされた線分がそれぞれの消失点を構成しているクラスタに属する線分である。以上の結果より実画像に対しても提案手法の有効性を確認できた。

表2: 実画像での実験結果

	画像枚数 (枚)
3個中3個検出	19
3個中2個検出	14
3個中1個検出	7
検出できない	0

5. むすび

本稿では、消失点検出における四面体体積最小化の提案をした。これは線分クラスタリングと消失点位置の計算を同時に行い、高精度な消失点の検出を可能にする手法である。提案手法は消失点をガウス球面上で表現するため、画像平面上で消失点の計算する三角形面積最小化法とは異なり無限遠の消失点が存在する場合でも例外として扱うことなく計算できる。また実験の結果、ガウス球に基づく手法に比べ線分端点誤差がある場合でも線分のずれの影響を受けにくいことがわかり、提案する四面体体積最小化法の有効性を確認できた。実画像においてもほとんどの画像に対して消失点を2個以上検出でき、今後の検出精度の向上によっては3個すべて検出できると考えられることから提案手法の有効性を確認できた。今後は、向きの異なる複

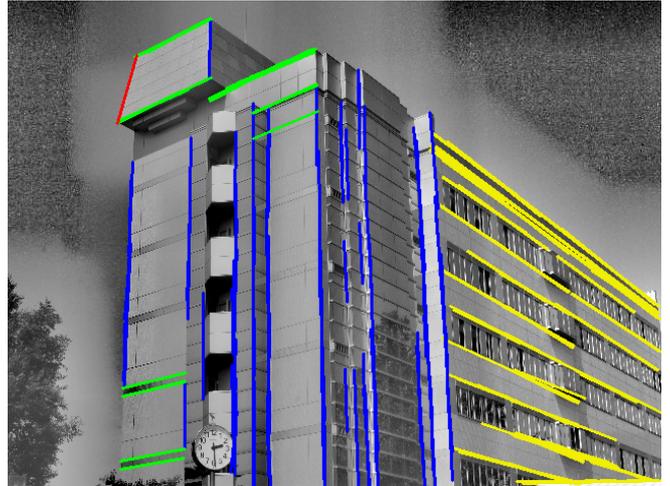


図2: 消失点を検出した画像の例

数の建築物を含む画像に対しても提案手法が適用できるかが課題となる。

参考文献

- [1] S. Barnard, "Interpreting Perspective Images," *Artificial Intelligence*, Vol.21, pp. 435–462, 1983.
- [2] Evelyne Lutton, Henri Maitre, and Jaime Lopez-Krahe, "Contribution to the Determination of Vanishing Points Using Hough Transform," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol.16, No.4, pp. 430–438, 1994.
- [3] Jefferey A. Shufelt, "Performance Evaluation and Analysis of Vanishing Point Detection Techniques," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol.21, No.3, pp. 282–288, 1999.
- [4] Burchardt, C. B., and Voss, K, "Robust vanishing point determination in noisy images," *ICPR*, Vol.1, pp. 559–562, 2000.
- [5] Martin A. Fischler, and Robert C. Bolles, "Random Sample Consensus: A Paradigm for Model Fitting with Applications to Image Analysis and Automated Cartography," *Comm. of the ACM*, Vol.24, pp. 381–395, 1981.