

単眼視連続画像からの奥行きと運動パラメータの反復計算手法

6D-6

ウィワット ウォンワライパット 石塚 満

東京大学生産技術研究所

1. はじめに

時間的に連続した画像をとり、物体の運動パラメータを検出する研究が、ロボットの分野などで、多く行なわれている。本稿は一つのカメラからの連続画像からオプティカルフロー検出でのあいまいさを考慮して、カメラの運動パラメータ検出と同時に奥行き情報の検出法を検討し、その結果を報告する。計算された運動はカメラを固定とすれば物体の運動に変換できる。

2. オプティカルフロー検出

時刻 t に撮った画像Aと時刻 $t+\Delta t$ に撮った画像Bから、画像の中の点がどのように移動したかを検出する方法はマッチング法などの色々な方法がある。ここで示す方法は移動量が微小で輝度分布も連続的であると仮定し、その輝度の変化が線形化できるとすると、輝度の変化と移動量の関係は次のように近似できるといものである^[1]。

$$I_t + I_x U_x + I_y U_y = 0 \tag{1}$$

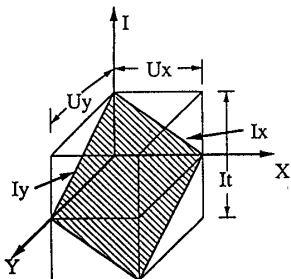


図1. 移動量と輝度の変化の関係

I_t は二つの画像の輝度の差で、 I_x, I_y は x, y 方向の輝度の画像空間勾配で、 U_x, U_y は画像中の移動量である。各点に式が1つしかないのに対して未知数は2つあるので周辺の画素も同じ動きをすると仮定し、最小自乗法で求める方法がある。ここで注意すべきなのは、検出されたオプティカルフローは実際の対象物の移動ではなく、画像上での移動にすぎず、実世界でのカメラの運動パラメータを検出するには工夫が必要である。

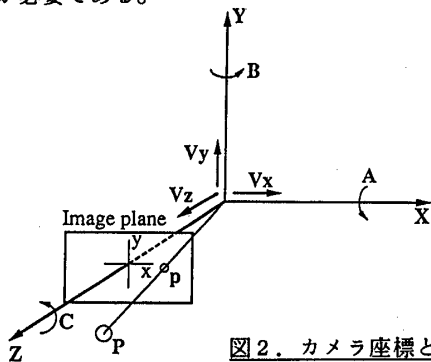


図2. カメラ座標と実世界座標

3. カメラの運動解析

図2に示すように実世界座標での対象物の位置 $P=(X, Y, Z)^t$ 、カメラの平行移動 $T=(V_x, V_y, V_z)^t$ 、回転 $R=(A, B, C)^t$ の対象物が2次元画像面 ($Z=1$ のところにおいてZ軸と垂直である)の $p=(x, y)^t$ に投影されるとする。カメラの中心は原点にあるとする。

点 p の移動量は次のように近似できる^[2]。

$$U_x = V_x/Z + B - yC - xV_z/Z \tag{2}$$

$$U_y = V_y/Z - A + xC - yV_z/Z \tag{3}$$

カメラが静止した環境の中で移動すると限定すると、画像に投影された対象物はカメラに対して同じ動きをするから、 T, R は一意に決まるので、 Z がわかれば最小自乗法を使ってすべての特徴点 (I_x, I_y が0でない点) から計算できる。多くの研究はオプティカルフローを計算した後にカメラの運動パラメータを計算するが、オプティカルフロー検出する段階での誤差があると、運動パラメータも誤差が大きくなる。その誤差は雑音や測定誤差以外に局所的な処理に起こるアパチャー問題^[3]という根本的な問題がある。

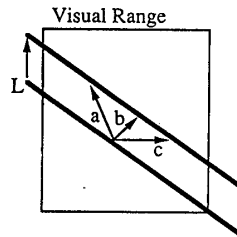


図3. アパチャー問題

図3のように窓をとって線Lが見えるとする。今度線Lが真上に移動したにも拘らず、観測した人間はB方向に移動したと勘違いする。実際にはA, C方向に移動することもあり得るので、解決するには大局的な処理が必要である。しかし、A, B, Cいずれの方向に移動しても、ある制約の下にあるので、その制約をつけ加えることによって、もっと正確な処理結果が得られる。ここではその U_x, U_y の関係の制約は式(1)であるため、途中の変数 U_x, U_y を求めないで直接に式(2), (3)を(1)に代入して計算を行なう。そして(1)の右辺を0ではなく、 e (誤差)と定義し、 e の自乗の合計(全画像の画素に対して)が最小になるような T, R を計算する。

4. 反復法による奥行き検出

上記から Z が未知の場合には結果が計算できないが、ここではまず各画素の Z を適当な値に初期化し、カメラの運動パラメータを計算する。その運動パラメータの元に全体の誤差が減少する方向に各画素の Z を書換えにいく。図4のように反復して計算して、求めた運動パラメータの変化があるしきい値以内になると、収束したとみなし、処理を終了する。 Z を書き換えるときに、式(2), (3)から逆算する以外に、

1. スムージングを行なって結果の安定化をはかる。またエッジ保存のために原画像の輝度差も使って、周辺との輝度差が大きいところではZの差も大きい可能性があるのでスムージングの係数を小さくする。

2. 反復する度にZが発散しないように、前の値も使って極端な変化を抑える。

新しいZは次のように定義する。

$$Z(n+1) = \alpha Z(n) + \beta Z_0(n) + (1-\alpha-\beta)Z_i \quad (4)$$

α, β はスムージングパラメータで、 $0 \leq \alpha, \beta, 0 \leq \alpha + \beta \leq 1$ 、 Z_i は運動パラメータから逆算した値、 $Z(n)$ は前回の値、 $Z_0(n)$ は周辺のZの平均値である。

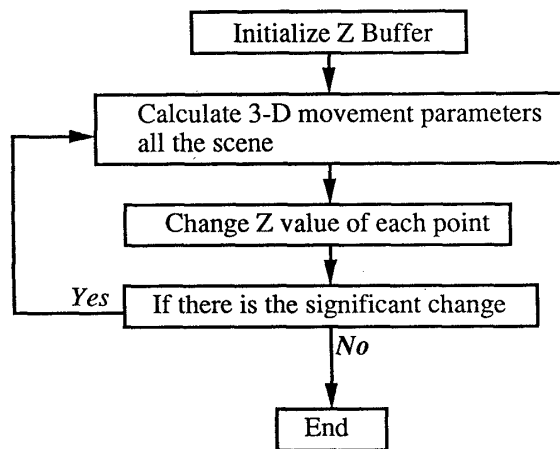


図4. 反復処理の流れ

5. 実験結果

画像を撮った後(図5に示す)にカメラを動かして、もう1枚の画像を撮って処理を行なった。画像の解像度は256×256画素で8bitsである。8台のコンピュータで並列に処理を行なって高速化した。処理は100回ぐらいで集束した。一回の処理は約2.5秒であるから、収束するまで約4分かかる。収束後の奥行き情報の結果は図6に示す。

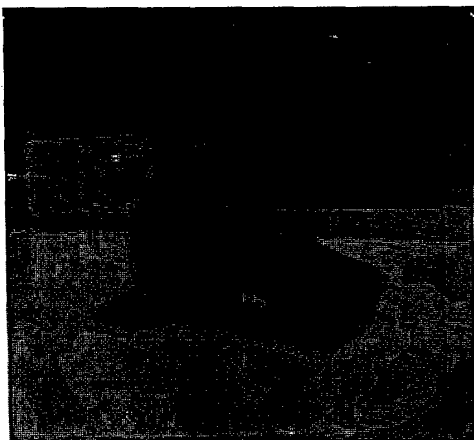


図5. 原画像

奥行き逆数を輝度に置き換えて表示する。近いものは明るく表示し、遠いものは暗く表示する。注意しなければならないのは、運動パラメータの回転Rの絶対値は得られるが、平行移動Tの値はZによって変化するので、絶対値は得られない。Tの中の1つの要素の絶対値がわかれば、スケール係数もわかるので、あと2つの絶対値もわかることからステレオ視のような場合には絶対値が簡単に得られる。



図6. 奥行き情報の結果

6. おわりに

以上、カメラの運動パラメータ検出と同時に奥行き情報も検出する方法を検討して、実験した結果を報告した。本手法は全画像に対して最小自乗誤差法による計算を用いているため雑音に強いといえる。また一般的に、各画素の持っている情報の重要度は同じではないから、確信度などの何らかの尺度が必要になる。ここでは輝度勾配を使ってその役割とすることによって、より正確な結果が得られると期待できる。

この方法は単純な処理が大部分なので、並列処理以外に部分的にハードウェア化することによっての高速化も可能である。一つの平行移動のパラメータの絶対値または特定の画素の奥行きがわかれば全部のパラメータの絶対値も得られるので、有用であると考えられる。今後の問題としては次の事項について検討している。

1. 収束が早くしかも結果が安定になるようになるスムージングのパラメータの決め方。

2. 画像中に他の移動物体の混合がある場合、移動量領域で分割^[4]した後、各々の領域の中で3次元運動パラメータと奥行き情報を検出する。

参考文献

(1) HANS-HELLMUT NAGEL, WILFRIED ENKELMANN : 'An Investigation of Smoothness Constraints for Estimation of Displacement Vector Fields from Image Sequence', IEEE, PAMI-8, No. 5, 1986, pp565-593

(2) GUY L. SCOTT : 'Local and Global Interpretation of Moving Images', PITMAN PUBLISHING (1988)

(3) MURALIDHARA SUBBARAO : 'Interpretation of Visual Motion : A Computational Study', PITMAN PUBLISHING (1988)

(4) ウィワット、石塚 : 'コンピュータによる並列移動物体検出'、画像理解の高度化と高速化シンポジウム講演論文集、電子情報通信学会 情報システム研究グループ パターン認識・理解研究会 (1989, 4)