社団法人 情報処理学会　研究報告
IPSJ SIG Technical Report

2003－CVIM－141　(14)
2003／11／7

２段階特徴点追従アルゴリズム

深尾 隆則[†], 金出 武雄[‡]

摘要　Lucas-Kanade の特徴点追従アルゴリズムは高速であり広く利用されているが，理想的なアフィン変形の場合でも追従に失敗することがある．そこで，実用的には高速かつより高精度に追従可能なアルゴリズムが望まれている．本研究では，追従性能が格段に向上した Lucas-Kanade の特徴点追従アルゴリズムの発展形を提案する．これは平行移動成分とそれ以外を分離して追従を行うものである．また，本手法に適した追従しやすい特徴点の発見法も提案し，追従に用いる特徴領域の大きさもそれぞれに適切なものを決定する．

Two Step Algorithm for Point Feature Tracking

T. Fukao[†] and T. Kanade[‡]

Abstract　Lucas-Kanade algorithm for point feature tracking is widely utilized because of its fastness, but it sometimes fails even if the deformation is occurred by an ideal affine transformation. For practical use, the fast and accurate algorithm to track a point feature is desired. In this paper, an extension of Lucas-Kanade algorithm is proposed to improve the tracking ability. This algorithm consists of two steps: the first step is done only for translation in order to attain accurate tracking, and the second step recovers the change between successive images by rotation and scaling. A detection algorithm which is good for the proposed tiracking algorithm is also provided by considering the separatability of two steps. The window size for a point feature is also decided to be appropriate for each step.

1　はじめに

２枚の画像に関する輝度の２乗偏差和を最小化するアルゴリズムとして画像の輝度勾配を使ったものは，相関を使ったマッチングアルゴリズムとは違い，リアリタイムアプリケーションで利用するのに十分な可能性がある [1,2]．その代表的なものは Lucas-Kanade のアルゴリズムであるが，それに基づいた特徴追従アルゴリズムについて見ていく．まずオリジナルの Lucas-Kanade トラッカ [1] はアフィン変形を前提とし，テンプレート画像を１フレーム目のものに取り，ワープを行う必要がある．次に，KLT トラッカ [2] は平行移動だけを考えている．アフィン変形に拡張可能 [3] だが，時系列画像を考えており，テンプレート画像を最終時刻の画像に取ることにより，平行移動成分以外の変動が小さいために影響があまり出ず，またワープの必要もない．さらに，この方法に適した追従しやすい特徴の発見法も提案している．Lucas-Kanade のアルゴリズムで，ワープの仕方による計算量評価やアルゴリズムによる収束性などを調べているものもある [4]．

本研究では，これらのアルゴリズムよりも高精度に追従可能なものを提案する．このアルゴリズムは２段階に分けられており，まず平行移動成分だけの追従を行い，次に平行移動成分以外の変化を補償するものである．ここで，どの成分を分離するか，またそのためにどのような領域を取れば良いかなどの解析を行うことによりアルゴリズムを構築し，性能向上につなげている．さらにこれらの解析に基づいた特徴領域の自動発見，決定も可能になる．

なお，文献 5) では平行移動・回転の成分と，スケーリング・剪断成分の２段階に分けて行っているが，本研究の解析からは妥当な分離とは言えないものである．

2　Lucas-Kanade のアルゴリズム

画像上の点 $\mathbf{x}$ が次のフレームで点 $A\mathbf{x}+\mathbf{d}$ に移った時，次式のような関係を仮定する．

$$J(A\mathbf{x}+\mathbf{d}) \;=\; I(\mathbf{x}) \tag{1}$$

ここで，I は点 $\mathbf{x}$ における最初の画像の輝度を表し，J は点 $A\mathbf{x}+\mathbf{d}$ における次の画像の輝度を表す．

本節では，まず特徴点の周辺領域が平行移動，回転，対称的なスケーリングだけを伴うと仮定する．そこで，A は次のように表される．

$$A \;=\; \begin{bmatrix} s_1 & 0 \\ 0 & s_1 \end{bmatrix}\begin{bmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{bmatrix}$$

† 京都大学大学院 情報学研究科
Graduate School of Informatics, Kyoto University
fukao@i.kyoto-u.ac.jp
‡ カーネギーメロン大学 ロボット工学研究所
The Robotics Institute, Carnegie Mellon University

$$\begin{aligned}
&= \begin{bmatrix} 1+s_0 & 0 \\ 0 & 1+s_0 \end{bmatrix} \\
&\quad \left(\mathbf{1} + \begin{bmatrix} \cos\theta - 1 & -\sin\theta \\ \sin\theta & \cos\theta - 1 \end{bmatrix} \right) \\
&\approx \mathbf{1} + \begin{bmatrix} s_0 & -\theta \\ \theta & s_0 \end{bmatrix} \\
&\equiv \mathbf{1} + D
\end{aligned} \tag{2}$$

ここで，$\mathbf{1}$ は単位行列，$s_1 = 1 + s_0$ とおき，$s_0 \ll 1, \theta \ll 1$ となる．また，2 次以上の項はここでは無視する．

良く知られているように，(1) 式は一般的には満されず，次の 2 乗誤差を最小化するような変位量を発見する．

$$\epsilon = \int_{\mathcal{W}} \left[J(A\mathbf{x} + \mathbf{d}) - I(\mathbf{x}) \right]^2 w(\mathbf{x}) d\mathbf{x} \tag{3}$$

ここで，$\mathcal{W}$ は考慮する画像領域であり，$w(\mathbf{x})$ はガウス関数や円形化する関数により表される重み関数である．

変位 $D\mathbf{x} + \mathbf{d}$ が $\mathbf{x}$ より十分に小さい時，輝度関数を $\mathbf{x}$ の回りで近似線形化すると，

$$\begin{aligned}
J(A\mathbf{x} + \mathbf{d}) &\approx J(\mathbf{x} + D\mathbf{x} + \mathbf{d}) \\
&\approx J(\mathbf{x}) + \frac{\partial J}{\partial \mathbf{x}} (D\mathbf{x} + \mathbf{d})
\end{aligned} \tag{4}$$

となり，(3) 式は次のように書ける．

$$\epsilon \approx \int_{\mathcal{W}} \left[J(\mathbf{x}) - I(\mathbf{x}) + \frac{\partial J}{\partial \mathbf{x}} (D\mathbf{x} + \mathbf{d}) \right]^2 w(\mathbf{x}) d\mathbf{x} \tag{5}$$

そこで，(5) 式を $\mathbf{z} = [\theta, s_0, d_x, d_y]^T$ で偏微分すると，次式が得られる．

$$G\mathbf{z} = \mathbf{e} \tag{6}$$

ここで，

$$G = \int_{\mathcal{W}} \begin{bmatrix} G_a & G_b \\ G_b^T & G_c \end{bmatrix} w(\mathbf{x}) d\mathbf{x} \tag{7}$$

$$\mathbf{e} = \int_{\mathcal{W}} [I(\mathbf{x}) - J(\mathbf{x})] \begin{bmatrix} -g_x y + g_y x \\ g_x x + g_y y \\ g_x \\ g_y \end{bmatrix} w(\mathbf{x}) d\mathbf{x} \tag{8}$$

$$G_a = \begin{bmatrix} (-g_x y + g_y x)^2 & (-g_x y + g_y x)(g_x x + g_y y) \\ (-g_x y + g_y x)(g_x x + g_y y) & (g_x x + g_y y)^2 \end{bmatrix} \tag{9}$$

$$G_b = \begin{bmatrix} (-g_x y + g_y x) g_x & (-g_x y + g_y x) g_y \\ (g_x x + g_y y) g_x & (g_x x + g_y y) g_y \end{bmatrix} \tag{10}$$

$$G_c = \begin{bmatrix} g_x^2 & g_x g_y \\ g_x g_y & g_y^2 \end{bmatrix} \tag{11}$$

ただし $g_x = \frac{\partial J}{\partial x}$, $g_y = \frac{\partial J}{\partial y}$.

(6) 式を Newton-Raphson 形式で解くことによって，文献 1,6,3,4) に示されているように (5) 式は最小化されていく．

3 G の評価・解析

この節では，(7) 式で定義される G の評価・解析を行い，前節に述べた方法に関して得られる知見を述べる．

文献 6) に述べられている通り，領域 $\mathcal{W}$ が大きくなるとオクルージョンやノイズなどのために追従の精度は悪くなる．しかし，領域を小さく選択すると，回転やスケーリングの影響は十分に現れないために Lucas-Kanade の方法では問題が生じることがある．

3.1 回転とスケーリングの影響

まず回転による変化量を計算する．領域の大きさの半分を k_w 画素とし，回転は中心回りで θ 度とすると，変化量は次式の通りである．

$$p = k_w \tan\theta \tag{12}$$

領域の端にある特徴に関して，回転角とその回転に伴い 1 画素変化するために必要な領域の大きさの関係を表 1 に示す．

θ [deg]	k_w [pixel]
1	57
2	29
3	19
4	14
5	11
10	5.7
15	3.7
20	2.7
30	1.7

表 1: Relation between rotational angle and window size

次に，スケーリングによる変化量を計算する．スケーリングの大きさが中心回りに $1 + s_0$ であるとすると，変化量は次の通りである．

$$p = s_0 k_w \tag{13}$$

この時，領域の端にある特徴に関して，スケーリングの変化分 s_0 と 1 画素変化するために必要な領域の大きさは表 2 の通りである．

また 1 フレーム当りの移動距離 Δz [m/frame] は次の通りである．

$$\Delta z = \frac{1}{1 + s_0} z - z \tag{14}$$

ここで，z は 5 [m] と 1 [m] とした．

さらに，毎フレーム 50 [ms] かかるとして速度 v [km/h] を求め，表に加えた．

s_0	k_w [pixel]	Δz [m/frame]	v [km/h]	Δz [m/frame]	v [km/h]
		$z = 5$ [m]		$z = 1$ [m]	
0.01	100	-0.05	-3.6	-0.01	-0.7
-0.01	100	0.05	3.6	0.01	0.7
0.05	20	-0.24	-17	-0.05	-3.4
-0.05	20	0.26	19	0.05	3.8
0.1	10	-0.45	-32	-0.09	-6.5
-0.1	10	0.55	40	0.11	8
0.2	5	-0.83	-60	-0.17	-12
-0.2	5	1.25	90	0.25	18
0.5	2	-1.67	-120	-0.33	-24
-0.5	2	5.00	360	1.00	72

表 2: Relation between scaling variance and window size

ウィンドウサイズが小さい時は回転とスケーリングの影響が陽にあまり現れないので，(6) 式の次の部分だけを利用すれば良い．

$$\int_{\mathcal{W}} G_c w(\mathbf{x}) d\mathbf{x} \begin{bmatrix} d_x \\ d_y \end{bmatrix} = \int_{\mathcal{W}} [I(\mathbf{x}) - J(\mathbf{x})] \begin{bmatrix} g_x \\ g_y \end{bmatrix} w(\mathbf{x}) d\mathbf{x}. \quad (15)$$

もちろん，この (15) 式だけを解けば良いという訳ではなく，時間とともに回転とスケーリングの影響が現れてくるので，その補償をしなければならない．

3.2 追従に適した特徴

もし G が対角行列なら，$\mathbf{z}$ の要素は完全に別々に解くことが出来る．このような状況であれば，どのような組み合せ，順番でも求めることは出来る．この条件を満たす１つの例は図 1 に示すような長方形である．一方，回転した長方形は満たさないが，これが正方形なら条件を満足する．

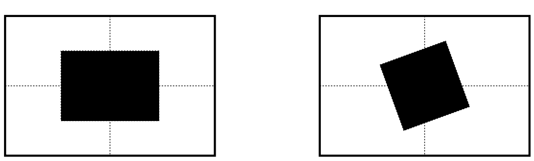

図 1: Features of diagonal matrix G

図 2 の左図のように注目しているウィンドウより大きい場合，$\mathbf{d}$ の変化だけなら良い特徴であるが [6]，$\mathbf{d}$ と s_0 の変化があるときは混同を生じる．これは，以下の行列が正則ではないということである．

$$\begin{bmatrix} G_{22} & G_{23} & G_{24} \\ G_{32} & G_{33} & G_{34} \\ G_{42} & G_{43} & G_{44} \end{bmatrix} \quad (16)$$

$$= \int_{\mathcal{W}} \begin{bmatrix} g_x^2 a^2 + g_y^2 b^2 & g_x^2 a & -g_y^2 b \\ g_x^2 a & g_x^2 & 0 \\ -g_y^2 b & 0 & g_y^2 \end{bmatrix} w(\mathbf{x}) d\mathbf{x}$$

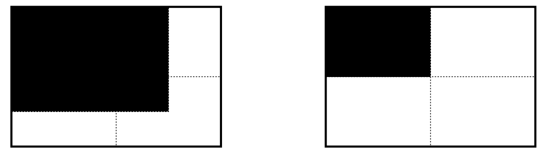

図 2: Aperture problems in tracking features

なお，もし図 2 の右図のように角が選ばれた場合は，$\mathbf{d}$ だけは正しい値を得ることが出来る．

平行移動だけを考えた文献 6) に述べられているのと同様に G は正則でなければならないことが良く分かる．このため，G の固有値は十分に大きくなければならない．

3.3 平行移動，回転，スケーリングの独立性

G は対称行列であるので固有値は実数であり，G は次のように書ける．

$$L^{-1} G L = \Lambda \quad (17)$$

ここで，G は正則であると仮定し，Λ は対角行列，$L = [\mathbf{l}_1\ \mathbf{l}_2\ \mathbf{l}_3\ \mathbf{l}_4]$ は正規直交行列である．

$\mathbf{z}$ を次のように表す．

$$\mathbf{z} = L\mathbf{a} \quad (18)$$

ここで，$\mathbf{a} = [a_1\ a_2\ a_3\ a_4]^T$．この時，(6)，(17) 式より次式が得られる．

$$GL\mathbf{a} = L\Lambda\mathbf{a} = \mathbf{e} \quad (19)$$

つまり，

$$\Lambda\mathbf{a} = L^{-1}\mathbf{e} \quad (20)$$

$\mathbf{l_i}, 1 \le i \le 4$ を次の基本ベクトル（または，その逆ベクトル）の１つと選ぶ．

$$\begin{bmatrix} 1 \\ 0 \\ 0 \\ 0 \end{bmatrix}, \begin{bmatrix} 0 \\ 1 \\ 0 \\ 0 \end{bmatrix}, \begin{bmatrix} 0 \\ 0 \\ 1 \\ 0 \end{bmatrix}, \begin{bmatrix} 0 \\ 0 \\ 0 \\ 1 \end{bmatrix} \quad (21)$$

この時，次のことが容易に確かめられる．

$$G_{ij} = 0, \qquad i \neq j \quad (22)$$

つまり，$\mathbf{z}$ の要素は G が対角行列でなければ，完全に別々に解くことは出来ない．しかし，(θ, s_0) と (d_x, d_y) のような２つのブロックに部分的に分離出来る可能性はある．これは，前述したベクトルが次のように表されることを意味する．

$$\begin{bmatrix} * \\ * \\ 0 \\ 0 \end{bmatrix}, \begin{bmatrix} * \\ * \\ 0 \\ 0 \end{bmatrix}, \begin{bmatrix} 0 \\ 0 \\ * \\ * \end{bmatrix}, \begin{bmatrix} 0 \\ 0 \\ * \\ * \end{bmatrix} \quad (23)$$

ここで，* は零ベクトルにならない任意の値である．

以上をまとめると，以下の通りである．

- G は実対称行列である．
- 固有値は実数である．
- 対応する固有空間は直交し，以下のように書ける．
$$L_1 \oplus \cdots \oplus L_4 = R^4 \qquad (24)$$
ただし，固有値がすべて異なるとする．
- $a = L^{-1}z$ という変換によって得られる空間は完全に分離可能である．
- しかし，物理的な空間とは違う．(23) 式を満足すれば，そのまま分離可能な空間である．

4 2段階特徴点追従アルゴリズム

前節で G の特性を調べたが，回転とスケーリングの影響は特徴領域の大きさが小さいと逐次的に陽には現れて来ず，画像間の回転とスケーリングの変化が小さければ，平行移動成分 $\mathbf{d}$ を得るためには (15) 式を解くだけで十分である．つまり，小さい領域が平行移動のために適しており，回転とスケーリングのためには大きい領域が適している．また，G の対角性と固有値が重要であることを見た．これらを考慮した上で，特徴点追従のための新しいアルゴリズムを提案する．

ステップ 1: 平行移動

注目点の回りの小領域において，以下の方程式を用い，テンプレートと現画像との平行移動量 $\mathbf{d}$ を求める．

$$\int_{\mathcal{W}_1} G_c w(\mathbf{x}) d\mathbf{x} \begin{bmatrix} d_x \\ d_y \end{bmatrix} = \int_{\mathcal{W}_1} [I(\mathbf{x}) - J(\mathbf{x})] \begin{bmatrix} g_x \\ g_y \end{bmatrix} w(\mathbf{x}) d\mathbf{x} \qquad (25)$$

ステップ 2： 回転とスケーリング

ステップ 1 よりも大きい領域を新しいテンプレートとして設定する．ここで，ステップ 1 の追従結果を用いることにより，中心位置を $\mathbf{d}$ だけずらした新しい現在の領域を得る．そして，次の方程式を解き，回転量とスケーリング変化量 θ，s_0 を得る．

$$\int_{\mathcal{W}_2} G_a w(\mathbf{x}) d\mathbf{x} \begin{bmatrix} \theta \\ s_0 \end{bmatrix} = \int_{\mathcal{W}_2} [I(\mathbf{x}) - J(\mathbf{x})] \begin{bmatrix} -g_x y + g_y x \\ g_x x + g_y y \end{bmatrix} w(\mathbf{x}) d\mathbf{x} \qquad (26)$$

テンプレート画像に関して，KLT トラッカ [6] は現時点より 1 つ前の画像をテンプレートとして用いる．そのため，特に小領域を用いることで回転やスケーリングの影響は小さく，平行移動だけを考慮するのは妥当と思われるが，一度追従に失敗するとその回復は本質的に不可能である．また，追従の失敗はしばしば起る．

一方，L-K トラッカは初期フレーム画像をテンプレートとして用いる．このためにワープが必要となり，初期画像か現画像に適用される．初期画像がワープされる場合には，G の計算は現画像に基づくために，3.2 節や 5.1 節で述べた G の対角性や分離性が変化してしまう．そこで，現在の画像をワープすることにより，この特性を保ったままに出来る．また，初期画像をそのまま用いることで，ずっと同じ G を使うことが出来る．つまり，文献 4) に述べられているような計算量の減少も同時に達成出来る．

提案するアルゴリズムでは，以下のような行列を $N+1$ フレームのワープで用いる．

$$W^N = \left[\mathbf{1} + \begin{bmatrix} \sum_{i=2}^{N} s_0^i & 0 \\ 0 & \sum_{i=2}^{N} s_0^i \end{bmatrix} \right] R\left(\sum_{i=2}^{N} \theta^i\right) \qquad (27)$$

または，

$$W^N = \begin{bmatrix} \prod_{i=2}^{N} (1 + s_0^i) & 0 \\ 0 & \prod_{i=2}^{N} (1 + s_0^i) \end{bmatrix} R\left(\sum_{i=2}^{N} \theta^i\right) \qquad (28)$$

ここで，s_0^i と θ^i は第 i フレーム後に計算されるものであり，$R(\cdot)$ は以下により表される回転行列である．

$$R(\alpha) = \begin{bmatrix} \cos\alpha & -\sin\alpha \\ \sin\alpha & \cos\alpha \end{bmatrix} \qquad (29)$$

さらに，領域を四角形で考えるより円形として考える方が回転にとって適している．特に，平行移動と分離して考えているので，その効果はさらに生かされる．

この 2 段階アルゴリズムは Newton-Raphson 形式の解法における局所解の数を低減する．これは特徴点追従問題に対する非線形最適化におけるパラメータ空間の効果的な分離として考えられる．さらに次の節で述べる追従しやすい特徴領域の性質を壊さずに生かすものである．図 3 にアルゴリズムの概要を示す．

5 2段階アルゴリズムのための特徴発見法

文献 6) において，KLT トラッカに適した特徴発見法が提案されている．しかし，回転とスケーリングも含めて 2 段階アルゴリズムにとって十分なものではない．

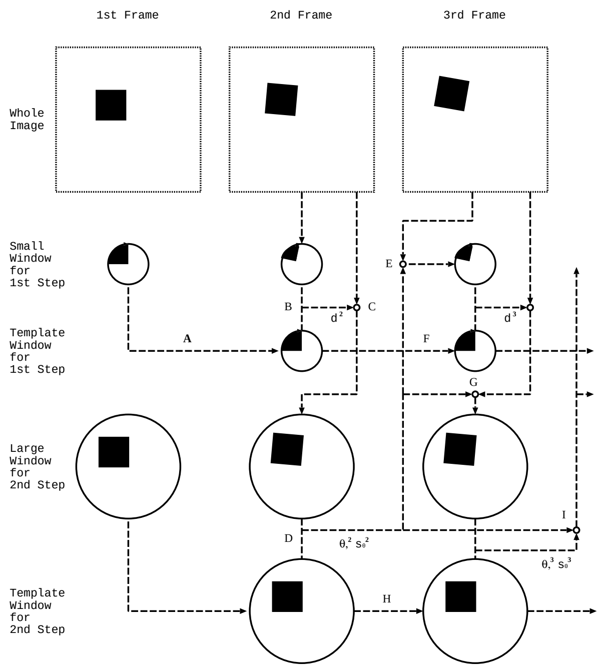

図 3: Two step algorithm. **A**: From the first frame a small size image is taken around an attended point and it is the first template of the first step. **B**: The same size image is chosen from the current image around the attended point, and the translation **d** is got by solving (25). **C**: A large size image is taken around the point which is translated by **d** from the attended point of **B**. **D**: From the large size image and the large size template which is in the first frame, the rotational angle θ and the scaling size s_0 are obtained by solving (26). **E**: The next small size image which is taken from the current frame is warped by rotation and scaling inversely. **F**: The small size template is not rotated and scaled, that is, it is the same image as the previous one. **G**: The large size image from the current frame is also warped by rotation and scaling inversely after the translation **d**. **H**: The large size template of the first image is not rotated and scaled. **I**: The rotational angles and scaling ratios are added or multiplied like (27) or (28).

5.1 2 段階アルゴリズムのための追従に適した特徴点

3.3 節を参考にすると，(23) 式を満たすか，または近い G は 2 段階アルゴリズムにとってより良いものである．正規直交行列を $L = [l_{ij}]$ と書くとすると，G のブロック分離性を示す次の量が小さいことが望まれる．

$$q \;=\; \sum_{i=3}^{4} l_{i1}^2 + \sum_{i=3}^{4} l_{i2}^2 + \sum_{i=1}^{2} l_{i3}^2 + \sum_{i=1}^{2} l_{i4}^2 \quad (30)$$

ここで，最初の 2 列は θ と s_0 のモードであり，最後の 2 列は d_x と d_y のモードである．これらのモードは次の通りである．まず固有値，固有ベクトルを用いて，$Gl_i = \lambda_{Gi}$，つまり，以下のように表す．

$$\begin{bmatrix} G_a & G_b \\ G_b^T & G_c \end{bmatrix} \begin{bmatrix} l_i^{(1)} \\ l_i^{(2)} \end{bmatrix} = \lambda_i \begin{bmatrix} l_i^{(1)} \\ l_i^{(2)} \end{bmatrix} \quad (31)$$

ここで，$l_i = [l_i^{(1)T}\ l_i^{(2)T}]^T$ とした．l_i は単位ベクトルであり，一般的に $\mathrm{eig}(G_a) > \mathrm{eig}(G_b) > \mathrm{eig}(G_c) > 0$ であるので，θ, s_0 のモードは大きな固有値 λ_i を持ち，d_x, d_y のモードは小さな固有値 λ_i を持つ．

ステップ 1 の領域が小さいことを考慮して，G_c の 2 つの固有値 λ_1, λ_2 は $\mathbf{d}$ の精度にもよるが十分大きくなければならない [6)]．また，ここでその小さい方の固有値を λ_m とおく．つまり，$\lambda_m = \min(\lambda_1, \lambda_2)$.

以上より，q と λ_m を良い特徴を選ぶために利用する．q は 2 段階分離のために重要な要素であり，λ_m は平行移動のためだけのものである．そこで，以下のような場合を調べた．紙面の都合上，結果の詳細は省略するが，より先鋭化される V, VI を用いることにする．

I:	$(1-q) \times \lambda_m$,	IV:	$(1-q) \times \lambda_{mw}$
II:	$(1-q)^2 \times \lambda_m$,	V:	$(1-q)^2 \times \lambda_{mw}$
III:	$(1-q)^3 \times \lambda_m$,	VI:	$(1-q)^3 \times \lambda_{mw}$

ただし，q と λ_m はそれぞれの最大値により正規化され，置換されたものである．また，λ_{mw} は λ_m を以下により変換したものである．

$$\lambda_{mw} \;=\; \frac{1}{1 + e^{-10(\lambda_m - 0.5)}} \quad (32)$$

さらに，今後ガウス関数と円形化関数を組み合せた重み関数を用いる．

$$w(\mathbf{x}) \;=\; \frac{e^{-(x^2+y^2)/(2\sigma^2)}}{1 + e^{2(x^2+y^2-(c-1)^2)}} \quad (33)$$

ここで，$\mathbf{x} = [x,\ y]^T$，原点は領域の中心，c は領域の大きさの半分である．

5.2 特徴点追従に適した領域の大きさ

図 4 において，実線の境界の方が点線よりスケーリングに対して適切なものである．もし点線の方が選ばれると，対象である ‘A’ のスケーリング変化は (26) 式を解く際に大きな影響を与える．これは，境界をまたぐ部分の特徴変化が激しいためである．

これを含めて，表 3 に追従領域の大きさに関する最大値，最小値に関する考察を示す．ただし，ここではオクルージョンはないものとする．

平行移動，回転，スケーリングの組み合せ，またはアフィン変換に関しては，次のようにまとめることが出来る．

- 2 段階アルゴリズムは平行移動だけのための領域と，回転・スケーリング，あるいは

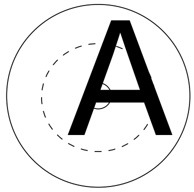

図 4: Adequate size of the window for scaling

変化	平行移動	回転	スケーリング
最大	回転とスケーリングからの影響なし	なし (無限が良い)	なし (無限が良い)
最小	固有値が十分大きいフレーム間の平行移動量が領域の大きさ以内	固有値が十分大きいフレーム間の回転の影響が画素に現れる	固有値が十分大きいフレーム間のスケーリングの影響が画素に現れる
特記	２段階の領域サイズを別々に設定出来るので良い	領域は無限ではないので，円形に取るのが適当である	領域境界付近に変化があまりないことが望まれる．特に逆解法は有効に働く

表 3: Analysis on maximum and minimum size of tracking window

アフィン変換においては平行移動以外の変形のための領域を分けて利用することが出来る．

- ステップ 1 がうまく働くと，ステップ 2 の領域の中心は回転・スケーリングの真の中心に等しいかまたは非常に近くなる．
- 逆解法は追従に適した特徴領域を保つために，特にスケーリングの影響を低減するのに役立つ．

以下では，ステップ 2 の領域の大きさの決定法を提示する．

輝度勾配の情報を十分に持たなければならないので，まず回転とスケーリングに関する固有値の濃度に基づいて基本領域を決定する．ここで，固有値の濃度とは考えている領域の面積でその領域に対して得られる固有値を割ることにより求められるものである．つまり，次の評価規範が最大になる領域の大きさをステップ 2 の基本領域の大きさとする．

$$\bar{F}_j = \frac{\min\{\bar{\lambda}_{j1}, \bar{\lambda}_{j2}\}}{\pi \bar{r}_j^2} \tag{34}$$

ここで，$\bar{\lambda}_{j1}$ と $\bar{\lambda}_{j2}$ は基本領域の j 番目の候補に対する回転とスケーリングに関する固有値であり，$\bar{r}_j$ は j 番目の候補領域の半径である．

選択された領域は，勾配情報が詰め込まれている部分を出来るだけまたがないように調整される．これによって前述した影響は及ぼされない．具体的には，次の規範を最小にするように領域の大きさを決定する．

$$F_k = \frac{\int_{\mathcal{W}_k} (g_x x + g_y y)^2 w(\mathbf{x}) d\mathbf{x} - \int_{\mathcal{W}_{k-1}} (g_x x + g_y y)^2 w(\mathbf{x}) d\mathbf{x}}{\pi r_k^2 - \pi r_{k-1}^2} \tag{35}$$

ここで，$k \geq 1$ とし，r_k は k 番目の円形領域の半径，r_0 は最初に選択された基本領域の半径である．

このアルゴリズムの概要は図 5 に示される通りである．また，種々の画像に適用した結果は省略するが，非常に良好なものである．

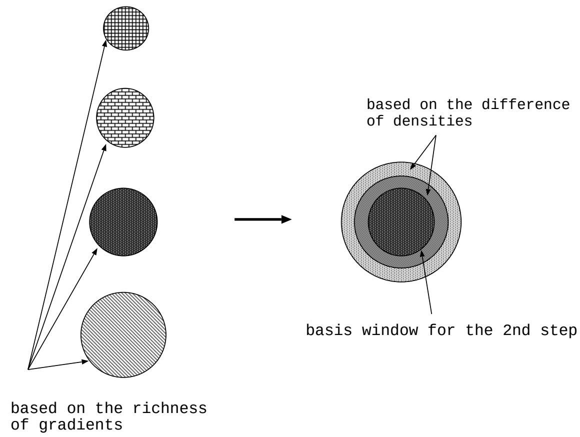

図 5: Algorithm for determination of 2nd step window size

6 アフィン変形への拡張

今までの節では，平行移動と回転，スケーリングのみを考えていた．弱射影環境などではこれらに限って推定を行う方がよりロバストな結果が得られるが，ここではアフィン変換で表される変形を考える．ただし，対称スケーリング，非対称スケーリング，歪みの有無などに応じた方法への拡張も可能であるが，ここでは省略する．

(1) 式の A は次のように表すことが可能である [7]．

$$A = R(-\phi) S R(\phi) R(\theta) \tag{36}$$

ここで，$R(\theta)$ と $R(-\phi)$ はそれぞれ θ，$-\phi$ だけの回転を表し，S は以下のようなスケーリングを表す．

$$S = \begin{bmatrix} s_1 & 0 \\ 0 & s_2 \end{bmatrix} \tag{37}$$

これにより，A は次のように変形出来る．

$$A = R(-\phi) \left[\mathbf{1} + \begin{bmatrix} s_{01} & 0 \\ 0 & s_{02} \end{bmatrix} \right] R(\phi) R(\theta) \tag{38}$$

$$\approx \mathbf{1} + \begin{bmatrix} 0 & -\theta \\ \theta & 0 \end{bmatrix} \tag{39}$$

$$+ \begin{bmatrix} s_{01}\cos^2\phi + s_{02}\sin^2\phi & (-s_{01}+s_{02})\sin\phi\cos\phi \\ (-s_{01}+s_{02})\sin\phi\cos\phi & s_{01}\sin^2\phi + s_{02}\cos^2\phi \end{bmatrix}$$

$$\equiv \mathbf{1} + \begin{bmatrix} 0 & -\theta \\ \theta & 0 \end{bmatrix} + \begin{bmatrix} q_1 & q_2 \\ q_2 & q_3 \end{bmatrix} \tag{40}$$

ここで，$\mathbf{1}$ は単位行列とし，$s_1 = 1 + s_{01}$, $s_2 = 1 + s_{02}$ とおいた．また $s_{01} \ll 1$, $s_{02} \ll 1$, $\theta \ll 1$ と仮定し，2 次以上の項を無視した．

$\mathbf{z} = [\theta, q_1, q_2, q_3, d_x, d_y]^T$ として，(6) 式の G, e は次のように書ける．

$$G = \int_{\mathcal{W}} \begin{bmatrix} G_a & G_b \\ G_b^T & G_c \end{bmatrix} w(\mathbf{x}) d\mathbf{x} \tag{41}$$

$$\mathbf{e} = \int_{\mathcal{W}} [I(\mathbf{x}) - J(\mathbf{x})]\, h w(\mathbf{x}) d\mathbf{x} \tag{42}$$

$$G_a = \begin{bmatrix} (-g_x y + g_y x)^2 & (-g_x y + g_y x) g_x x & (-g_x y + g_y x)(g_x y + g_y x) & (-g_x y + g_y x) g_y y \\ (-g_x y + g_y x) g_x x & (g_x x)^2 & g_x x (g_x y + g_y x) & g_x x g_y y \\ (-g_x y + g_y x)(g_x y + g_y x) & g_x x (g_x y + g_y x) & (g_x y + g_y x)^2 & (g_x y + g_y x) g_y y \\ (-g_x y + g_y x) g_y y & g_x x g_y y & (g_x y + g_y x) g_y y & (g_y y)^2 \end{bmatrix}$$

$$G_b = \begin{bmatrix} (-g_x y + g_y x) g_x & (-g_x y + g_y x) g_y \\ g_x^2 x & g_x x g_y \\ (g_x y + g_y x) g_x & (g_x y + g_y x) g_y \\ g_y y g_x & g_y^2 y \end{bmatrix}$$

$$G_c = \begin{bmatrix} g_x^2 & g_x g_y \\ g_x g_y & g_y^2 \end{bmatrix}$$

$$h = \begin{bmatrix} -g_x y + g_y x \\ g_x x \\ g_x y + g_y x \\ g_y y \\ g_x \\ g_y \end{bmatrix}$$

ワープのために次の行列を用いる．

$$W^N = \left[\mathbf{1} + \begin{bmatrix} \sum_{i=2}^{N} q_1^i & \sum_{i=2}^{N} q_2^i \\ \sum_{i=2}^{N} q_2^i & \sum_{i=2}^{N} q_3^i \end{bmatrix} \right] R(\sum_{i=2}^{N} \theta^i) \tag{43}$$

あるいは，

$$W^N = \begin{bmatrix} \prod_{i=2}^{N} (1 + q_1^i) & \sum_{i=2}^{N} q_2^i \\ \sum_{i=2}^{N} q_2^i & \prod_{i=2}^{N} (1 + q_3^i) \end{bmatrix} R(\sum_{i=2}^{N} \theta^i) \tag{44}$$

さらに，ϕ は次のようにして直接得ることが出来る．

(i) $q_1 \neq q_3$ の時

$$\phi = 0.5 \tan^{-1} \frac{2q_2}{-q_1 + q_3} \tag{45}$$
$$s_1 = \frac{q_1 \cos^2\phi - q_3 \sin^2\phi}{\cos 2\phi}$$
$$s_2 = \frac{-q_1 \sin^2\phi + q_3 \cos^2\phi}{\cos 2\phi}$$

(ii) $q_1 = q_3$,　$q_2 = 0$ の時

$$s_1 = s_2 = q_1 \tag{46}$$

(iii) $q_1 = q_3$，$q_2 \neq 0$ の時

$$\begin{aligned} \phi &= \pi/4 & & \phi = -\pi/4 \\ s_1 &= q_1 - q_2 & \text{または}\quad & s_1 = q_1 + q_2 \\ s_2 &= q_1 + q_2 & & s_2 = q_1 - q_2 \end{aligned} \tag{47}$$

以上より，θ, s_1, s_2, ϕ を直接得ることが出来るため，(38) 式をワープに適用することが可能となる．ただし，未知パラメータ間の関係が非線形となるためか，上述の行列 W^N を用いた結果より良くはない．

6.1　特徴発見アルゴリズムの拡張

アフィン変換の場合の特徴発見アルゴリズムも 5 節と同様であるが，違いは $L = [l_{ij}]$ の次元であり，$L = [\mathbf{l}_1\ \mathbf{l}_2\ \mathbf{l}_3\ \mathbf{l}_4\ \mathbf{l}_5\ \mathbf{l}_6]$ とする．(30) 式と同様に次式が小さくなれば良い．

$$q = \sum_{j=1}^{4} \sum_{i=5}^{6} l_{ij}^2 + \sum_{j=5}^{6} \sum_{i=1}^{4} l_{ij}^2 \tag{48}$$

ここで，最初の 4 つが平行移動を除いたアフィンパラメータのモードであり，最後の 2 つが平行移動のモードである．

7　シミュレーション結果

提案したアルゴリズムの性能を確認するために，いくつかのシミュレーション結果を示す．

まず，図 6 に示すような黒の背景に白の長方形があり，その長方形が回転・縮小しながら右方向に動く場合についての実験を行う．

図 6: Tracking of a basic moving object

背景の左上角を原点とし，水平方向は右向きが正，垂直方向は下向きが正とし，左上角を (60, 40) においた白い長方形は右下の方向に進む．図 7 に 1 フレーム当り 1 pixel の平行移動，1 度の回転，0.005 の割合で左下を中心として縮小した場合の結果を示す．ここで，ステップ 1 の追従領域の大きさは 15×15 pixel，ステップ 2 は 65×65 pixel とした．

次に図 8 にアフィン変換 (40) を顔画像に適用した結果を示す．i 番目のフレームにおいて，1 フレーム当り $10 \sin i$ pixel の平行移動，5 度の回転，$s_1 = 1 + 0.5 \sin i$, $s_2 = 1 - 0.25 \sin i$ で表されるスケーリング，歪み方向を表す ϕ を 45 度とした．図中の小さい画像の上段は初期フレームから取った理想的なもの，下段は推定されたアフィンパラメータを用いてワープした結果得られた画像を示している．

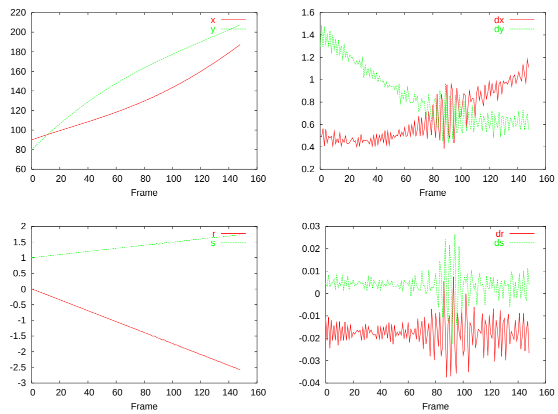

図 7: Tracking results of a moving object

これらの結果から，特徴追従は非常にうまくいっており，アフィンパラメータの推定もかなり良いものである．

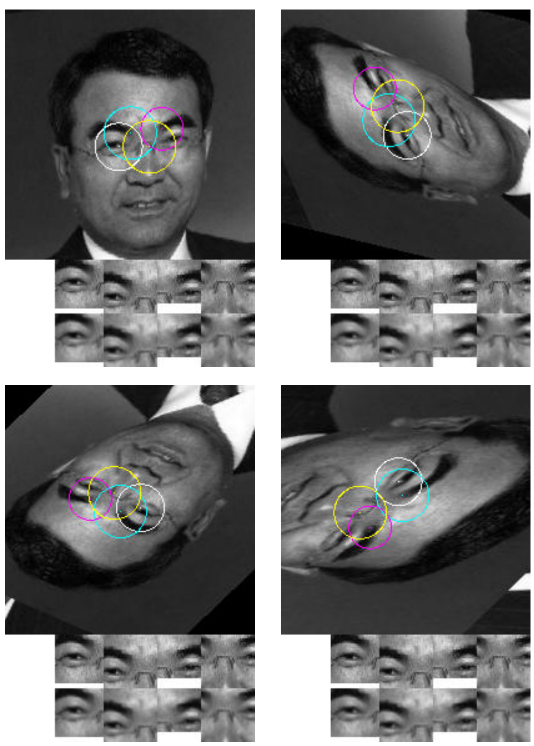

図 8: Tracking of a deformed face

最後に，このアルゴリズムを適用した飛行船ロボットによる監視システム [8,9] で利用した追従画像を示す．これは小型無線カメラで撮影し，弱射影環境と考えられる状況で用いている．

8 おわりに

Lucas-Kanade のアルゴリズムの発展として２段階特徴点追従アルゴリズムを提案した．連

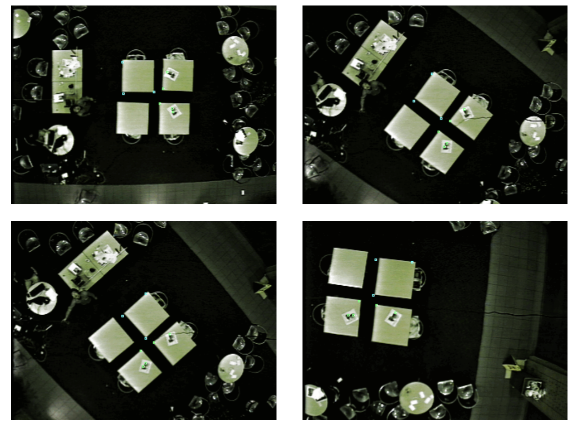

図 9: Autonomous blimp system

続画像が２次元アフィン変換で提示される場合において，非常に良好な追従結果が得られることを確認した．さらに，追従しやすい特徴領域の発見・決定法も提案した．本アルゴリズムはワープを通常のものとは逆に行っているが，これは発見した良い特徴を残すということにおいて非常に重要である．またこのことは，既に指摘されているように計算量の減少をもたらすものでもある．

参考文献

1) B. Lucas and T. Kanade. An iterative image registration technique with an application to stereo vision. In *IJCAI*, 1981.

2) C. Tomasi and T. Kanade. Detection and tracking of point features. Technical report, CMU-CS-91-132, 1991.

3) J. Shi and C. Tomasi. Good features to track. In *IEEE CVPR*, 1994.

4) S. Baker and I. Matthews. Lucas-Kanade 20 years on: A unifying framework: Part 1. Technical Report CMU-RI-TR-02-16, Carnegie Mellon University, 2002.

5) G. Hager and K Toyama. X Vision: A portable substrate for real-time vision applications. *Computer Vision and Image Understanding*, Vol. 69, No. 1, pp. 23–37, 1998.

6) C. Tomasi and T. Kanade. Detection and tracking of point features. Technical Report CMU-CS-91-132, Carnegie Mellon University, 1991.

7) R. I. Hartley and A. Zisserman. *Multiple View Geometry in Computer Vision*. Cambridge University Press, ISBN: 0521623049, 2000.

8) T. Fukao, K. Fujitani, and T. Kanade. An autonomous blimp for a surveillance system. In *Proc. IEEE/RSJ Int. Conf. Intelligent Robots and Systems*, to appear, 2003.

9) T. Fukao, K. Fujitani, and T. Kanade. Image-based tracking control of a blimp. In *Proc. 42th IEEE Conference on Decision and Control*, to appear, 2003.