

映像解析における射影変換処理の実用化とその課題

佐々木 敏, 角 田 和 彦*

Problems in and Practical Use of
Projective Transformation in Image Analysis

TSUTOMU SASAKI, KAZUHIKO TSUNODA

Abstract

This study had two purposes. The first one was to use coordinate transformation practically and applicably for dealing with image data. The other was to examine possible problems in those transformations. We examined some equations to find the optimum method for computing coordinate transformations. We chose a projective transformation which was the optimum equation for image analysis. Then we applied two equations to find the coefficients of that transformation. There were two methods of doing this the least squares method and the dimensional simultaneous equations method.

Our findings were as follows:

- 1) The projective transformation fit the theoretical points better than the affine transformation. The projective transformation made less errors than the affine transformation when modifying the declination.
- 2) Using more than four control points, it was necessary to use the least squares method for the transformation equation. Then the errors decreased.
- 3) Using projective transformation, the errors increased in proportion to the squares of the distance from the origin.
- 4) Large control points on the display made accidental errors decrease.

Keywords: projective transformation, method of least square, image analysis, practical use.

緒 言

映像による物体の運動解析は、運動を離散化させる方法として良く知られるところである。映像により最初に運動解析を行ったのは、米国の MYBRIDGE と言われている¹⁵⁾。このころの撮影速度は16fps から24fps であった。現在では、離散化の精度も高くなり、市販のビデオカメラを用いても、60Hz でのサンプリングが可能となっている¹⁶⁾。

運動の映像による記録は、その表現が具体的であることから、理解し易い面が多いと考えられるが、問題点が多いのもまた事実である^{2) 4) 5) 6)}。まず第一に、離散化の精度、第二に入力誤差の混入、第三として誤差の平滑化処理などがよく指摘されている^{2) 12) 13)}。また、さらに挙げるなら、実際の運動が3次元空間で行われるのに対し、一つの映像は常に2次元データとして提供されることである。

しかし、運動の方向が2次元平面上にて転換可能なものであれば相当な精度で解析できるようになってきている。今日多くの研究者がこの方法により運動解析を行っている^{1) 3) 7) 8) 9)}。ここで、身体モデルを考え、単純化したモデルが2次元平面上に転換されるような運動であるという条件下であれば射影変換技術が適用できると考えられる^{4) 10)}。また、この技術を、パーソナルコンピュータ上で実現することは、今後の運動解析に大きな影響を与えるものと考えられる。したがって射影変換技術の応用による運動解析システムの開発とこれの実用化に向けての問題点を探ることを本研究の目的とした。

研究方法

座標変換方程式の種類と特徴

本研究の目的は座標変換システムを映像解析に応用し、これを実用化することである。ここでどの座標変換システムを用いるかは、幾何学的補正に用いられている変換式を検討することで与えられるものと考えた。これは、映像が2次元平面上の幾何学的図形として理解されると言う考えに基づくものである¹⁰⁾。

幾何学的補正の代表的な変換式

- 1) ヘルマート変換
- 2) アフィン変換
- 3) 疑似アフィン変換
- 4) 射影変換
- 5) 高次多項式による変換

ここで、歪みのある座標から歪みの無い座標系に変換することを幾何学的補正と言う。偏位修正とは撮影位置をそのままにし、撮影方向の傾きを変えた画像に変換することである。偏位修正を行なうとき基準点を用いるが、この基準点を含む平面上に奥行きを含む全ての座標点が射影されるかたちで変換される。このことから運動の方向に対してカメラの光軸が直交する場所での撮影が不可能なところでは、このような変換技術の応用が必要であると考えられる^{2) 10)}。それぞれの変換式の特徴を挙げると、ヘルマート変換はアフィン変換の、アフィン変換は射影変換の特殊な形である。アフィン変換の特徴は、1、「直線は直線に変換され、直線上の点の比は保存される。」2、「線分の平行性は保存される。」という2点である。また、射影変換の特徴は「直線は直線に変換されるが、直線上の各点における相互の距離の比は保存されない。また、これらの複比は保存されるが、線分の平行性は保存されない。」というものである。加えて、射影変換は透視変換を含むと言われている¹⁰⁾。つまり、偏位修正を行うためには、平面上で1点ないし2点の消失点をもつ透視画像を透視

をもたない画面に変換する射影変換が適当と考えられる。一方、多項式による変換も考えられるが、この場合変換式を解くために必要な校正点数は、3次多項式を用いた変換でも最小限10点が必要である。撮影条件の限られる場所においてこの条件を満たすことは不可能に近い。したがって3次多項式の応用は実用的ではないと考えられる。また、変換式を決定する係数に誤差が混入する可能性もある。その結果座標軸が歪むと指摘されている¹⁰⁾。このようなことから運動解析という物理的な数量化を目指すために座標変換式として射影変換を適用することの妥当性は高いと考えられる。

座標変換方程式の選択

本研究において、他の座標変換式をも検討したが、直線的な変換を得られることと、その他の特性から射影変換式の適応の妥当性を導くことができた。用いた射影変換式を以下に示す。

歪みのある座標系 (x, Y) から歪みの無い座標系 (u, v) への変換を $F(x, y)=u, G(x, y)=v$ とすると変換式は以下の式

$$u = \frac{a_1x + a_2y + a_3}{a_7x + a_8y + 1} \tag{1-1}$$

$$v = \frac{a_4x + a_5y + a_6}{a_7x + a_8y + 1} \tag{2-1}$$

で与えられる¹⁰⁾。

結果及び考察

射影変換を実行するにあたっては、未知数を決定しなければならない。未知数を決定するには、2次元空間における全ての点について変換しなければならない。したがって、射影平面上の点を基に方程式の係数を求めるためには、最小限4点の校正点が必要であることは以上の方程式より明らかである。また係数を求めるための方法について若干の検討を試みた。これには、最小校正点数(4点以上)を用いる場合と、より多くの校正点数(5点以上)を用いる場合とがある。したがって、それぞれについて検討することとする。

4点の校正点を用いる場合〔歪のある座標系の座標値が既知の点 $p_i(x_i, y_i)$ と歪の無い座標系の点 $p_i(u_i, v_i)$ ($i=1,2,3,4$)〕, 上記の座標値を変換式に代入し8元連立方程式を立て一意的に方程式を解くことができる¹¹⁾ 一方、5点以上の校正点を用いる場合〔歪のある座標系の既知の座標点 $p_i(x_i, y_i)$ と歪のない座標系の点 $p_i(u_i, v_i)$ ($i=1, \dots, n$)〕, 最小2乗法を用いて係数を求め、変換方程式を導かなければならない¹⁰⁾。

連立方程式による係数の決定

4点の校正点を基に変換方程式を求めるとき、8元連立方程式により係数を求めることができる。一方、8元連立方程式を立て一意的に解く場合、歪のある座標系の既知の点 $P_i(X_i, Y_i)$ が歪の無い座標系の点 $p_i(u_i, v_i)$ に写される。このとき、変換式の係数はガウスの消去法を用いて求めることができる¹¹⁾。この計算を以下に示す。

$p_i(x_i, y_i)$, $p_i(u_i, v_i)$, ($i=1,2,3,4$) を (1-1), (2-1) 式に代入すると

$$u_i = \frac{a_1x_i + a_2y_i + a_3}{a_7x_i + a_8y_i + 1} \tag{1-2}$$

$$v_i = \frac{a_4 x_i + a_5 y_i + a_6}{a_7 x_i + a_8 y_i + 1} \quad (1-2)$$

(1-2), (2-2) を変形すると

$$u_i = a_1 x_i + a_2 y_i + a_3 - a_7 x_i u_i - a_8 y_i u_i \quad (1-3)$$

$$v_i = a_4 x_i + a_5 y_i + a_6 - a_7 x_i v_i - a_8 y_i v_i \quad (2-3)$$

式 (1-3), (2-3) に $p_i(x_i, y_i)$, $p_i(u_i, v_i)$, ($i=1,2,3,4$) を代入すると

$$u_1 = a_1 x_1 + a_2 y_1 + a_3 - a_7 x_1 u_1 - a_8 y_1 u_1$$

$$v_1 = a_4 x_1 + a_5 y_1 + a_6 - a_7 x_1 v_1 - a_8 y_1 v_1$$

$$u_2 = a_1 x_2 + a_2 y_2 + a_3 - a_7 x_2 u_2 - a_8 y_2 u_2$$

$$v_2 = a_4 x_2 + a_5 y_2 + a_6 - a_7 x_2 v_2 - a_8 y_2 v_2$$

$$u_3 = a_1 x_3 + a_2 y_3 + a_3 - a_7 x_3 u_3 - a_8 y_3 u_3$$

$$v_3 = a_4 x_3 + a_5 y_3 + a_6 - a_7 x_3 v_3 - a_8 y_3 v_3$$

$$u_4 = a_1 x_4 + a_2 y_4 + a_3 - a_7 x_4 u_4 - a_8 y_4 u_4$$

$$v_4 = a_4 x_4 + a_5 y_4 + a_6 - a_7 x_4 v_4 - a_8 y_4 v_4$$

この 8 元連立方程式を解いて係数 $a_1 \sim a_8$ を求めればよい。計算機上でガウスの消去法を用いたプログラムを資料 1 に示す。

最小 2 乗法を用いた係数の決定

校正点が 5 点以上になる時、係数は一意的に求まらない。このため未知数の決定には最小 2 乗法を用いることが一般的である¹⁰⁾。すなわち最小 2 乗法は、基準点の座標の組 (u_i, v_i) , (x_i, y_i) , $i=1,2,\dots, n$, を与え、この時の残差 Wx_i, Wy_i の 2 乗和 Q を最小にするような係数を求めると、

$$Wx_i = u_i - f(x_i, y_i)$$

$$Wy_i = v_i - g(x_i, y_i)$$

$$Q = \sum (Wx_i^2 + Wy_i^2) \rightarrow \min \text{ である。}$$

この式は、

$$QWx = \sum Wx_i^2 \rightarrow \min$$

$$QWy = \sum Wy_i^2 \rightarrow \min$$

と同値である。

したがって

$$QWx = \sum_{i=1}^n \{u_i - (a_1 x_i + a_2 y_i + a_3 - a_7 x_i u_i - a_8 y_i u_i)\}^2 \rightarrow \min$$

$$QWy = \sum_{i=1}^n \{v_i - (a_4 x_i + a_5 y_i + a_6 - a_7 x_i v_i - a_8 y_i v_i)\}^2 \rightarrow \min$$

となる。

未知の係数 a_1, a_2, \dots は以下に示す偏微分により得られる方程式を解くことで得られる。

$$\frac{\partial QW_x}{\partial a_i} = 0 (i=1, 2, 3, 7, 8)$$

$$\frac{\partial QW_y}{\partial a_i} = 0 (i=4, 5, 6, 7, 8)$$

これを解くと

$$\frac{\partial QW_y}{\partial a_1} = \sum_{i=1}^n 2\{u_i - (a_1 x_i + a_2 y_i + a_3 - a_7 x_i u_i - a_8 y_i u_i)\}(-x_i) = 0$$

a_2, a_3, a_7, a_8 については同様の方法で求めることができる。

a_5, a_6 については下記の式を順次用いることが得られる。

$$\frac{\partial QW_y}{\partial a_4} = \sum_{i=1}^n 2\{v_i - (a_4 x_i + a_5 y_i + a_6 - a_7 x_i v_i - a_8 y_i v_i)\}(-x_i) = 0$$

これを展開すると

$$a_1 \sum_{i=1}^n x_i^2 + a_2 \sum x_i y_i + a_3 \sum x_i - a_7 \sum x_i^2 u_i - a_8 \sum y_i x_i u_i = \sum u_i x_i$$

$a_1 \sim a_8$ までについて同様に展開式を求め得られた8元連立方程式を解くことで射影変換式の係数を求める。

これにより得られた係数を射影変換式に代入して用いる。

計算機上ではQR分解法を用いた。

上記の変換式を基に作成したプログラムを資料2に示す。

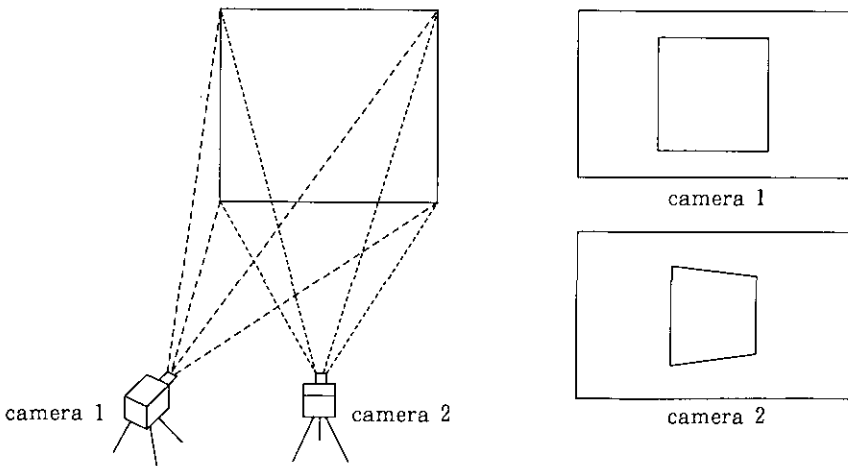


Figure. 1 Diagram of an experimental plane and camera position.
The real image was modified by projective transformation.

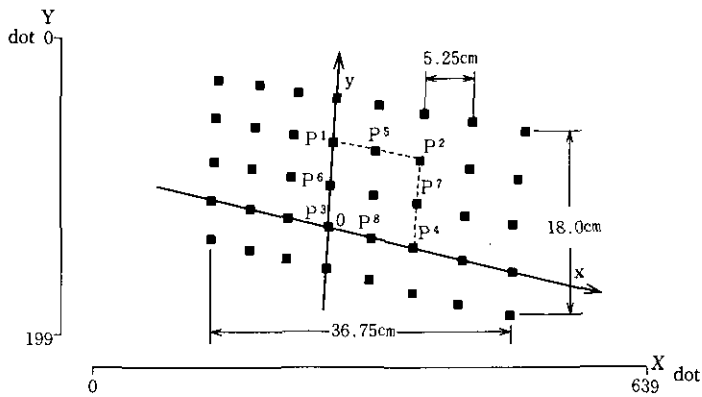


Figure. 2 The projected picture shows actual data points on computer display screen.

Legends : X,Y axis show the numbers of dot on the display screen.

x,y axis shows on the picture of actual data points.

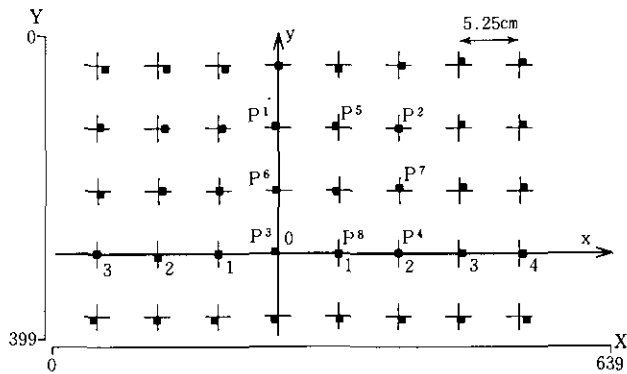


Figure. 3 Marks show the theoretical points and transformed data which were taken by projective transformation when using eight dimensional simultaneous equations.

Legends: The marks (+) are theoretical points.

The marks (■) are transformed data.

以上の方法で求めた係数を用い、射影変換式を完成させることができる。本研究で検討し、論議の中心となるのはこれら2種類の係数決定方法から導かれた射影変換式を使用した結果についてである。したがって、座標値が既知である一平面上の点をビデオカメラで撮影し、これを2つの方法により射影変換した。これを図1に示す。これらの具体的な例について資料を示し考察することとした。ここでは先ず、画面上に写し込まれた点を射影変換し、その適合のぐあいと変換誤差の程度を論理的に推定した。これには、座標値が既知の点を斜め右上方からビデオカメラで撮影し、写された点について画面上での座標値を読み取った。実平面上の点は縦横5.25cmの間隔の格子であった。

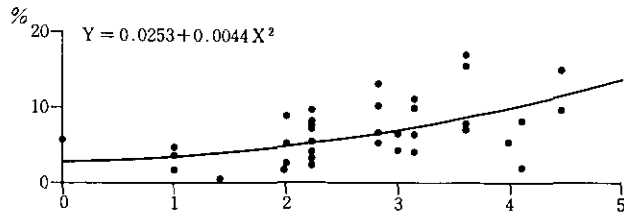


Figure. 4 Marks (●) show errors. The line is a regression curve showing the data of projective transformation when using eight dimensional simultaneous equations.

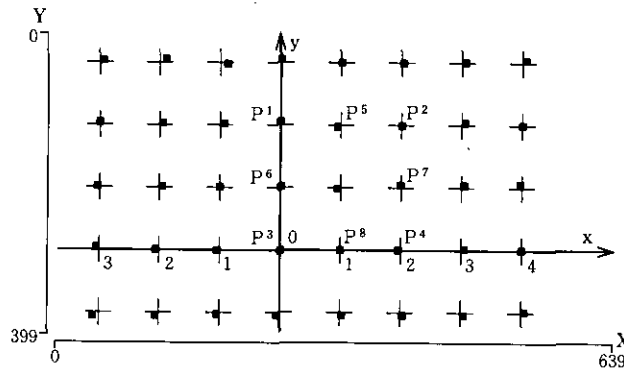


Figure. 5 Mark show the theoretical points and transformed data which was taken by projective transformation when using the least square method.

Legends: The marks (+) are theoretical points.
The marks (■) are transformed data.

格子については縦5行、横8列 ($p(i, j)$, $i=1\sim 5$, $j=1\sim 8$, の計40点) を使用した。この格子が画面に写った様子を図2に示した。また、格子上の原点を、便宜的に $p(4, 4)$ とした。画面の座標系は x の範囲 $0 \leq x \leq 639$, y の範囲 $0 \leq y \leq 199$ で左上を原点とする。このときの画面上の点の座標を図2に示した。これからこの図が透視図として得られていることが理解される。

連立方程式からの変換

4つの校正点に $p1(4, 2)$ $p2(6, 2)$ $p3(4, 4)$ $p4(6, 4)$ を使用して射影変換をおこなった結果 (図3) について考えることとする。図中の (+) 印は透視図中の各点が誤差なく変換された場合の論理的な座標点である。また、同時に示されている (■) 印は実際の変換点である。ここで、理論上の座標点と射影変換後の座標点とを比較する。図3において、原点から最も離れた座標点では、理論点と変換点のずれが顕著に観察される。このずれを変換誤差と考え、誤差の発現について、原点からの位置と、誤差の大きさについて図4に示した。図中の x 軸の数値は、理論上の各点につい

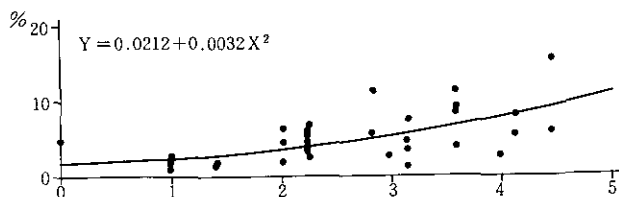


Figure. 6 Marks (●) show errors. And the line is a regression curve showing the data of affine transformation when using eight dimensional simultaneous equations.

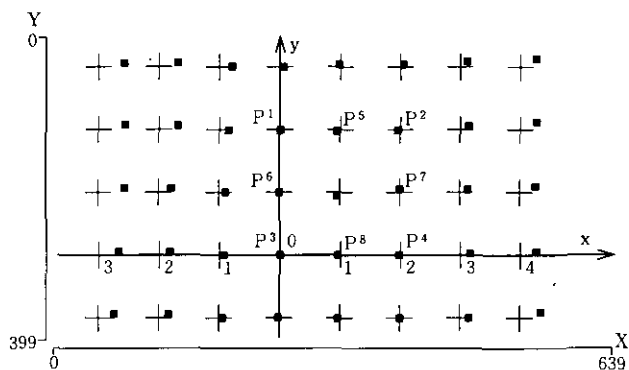


Figure. 7 Marks show the theoretical points and transformed data which was taken by affine transformation when using the least square method.

Legends: The marks (+) are theoretical points.

The marks (■) are transformed data.

て、隣接する点の距離を単位とし単位の倍数で示した。また、変換誤差については、それぞれの誤差を1単位を基準とし100分率で表わした。これを誤差率と呼ぶこととする。また、図中に与えられた線は2次の回帰式で得られた回帰曲線である。これによると、誤差の大きさは、原点から遠ざかるほど大きくなるとう理解される。しかもその大きさは2次関数に従う傾向が明らかである。また、その傾きは大きいものではないと考えられる。一方、各単位毎のデータに関して5%から7%程度のばらつきが認められる部分がある。これらは、原点付近の座標点ではあまり認められないが、2単位を越えた付近から急激に増加する傾向を示している。従って、この方法でデータを処理しようとするときは、校正点をあまり越えないところでデータを採取し処理するべきと考える。

最小自乗法からの変換

図5は、図3と同様にして示したものである。この図と図3とを比較し検討する。図3において、原点から離れた部分での変換点は理論点に対して明確にずれを認めた。しかし、図5においてはそのずれの大きさは少ないと理解される。この傾向は、図6を観察することでより明らかである。回帰曲線の変化率について検討すると、8元連立方程式を用いた場合0.0044であった。これに対して、

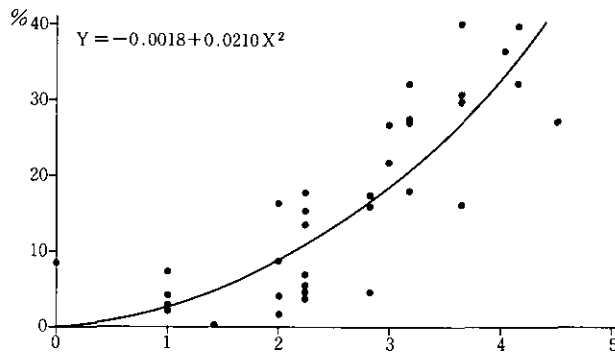


Figure. 8 Marks (●) show accidental errors. The line is regression curve showing the data of affine transformation when using eight dimensional simultaneous equations.

最小自乗法を用いた例ではその係数は、0.0032であった。したがって、最小自乗法を用いた場合、原点から離れるにしたがいその誤差が大きくなる傾向は同じであった。しかし、回帰係数でみる限り、誤差の程度は小さく、手法の相違による改善傾向は明らかと考えられる。また、各単位毎の誤差のばらつきも小さい傾向が認められる。これらのことから、現段階においてデータ採取に射影変換を用いる必要がある場合は、変換式の係数は最小自乗法により決定すべきと結論づけられる。しかし、採取するデータが写し込む校正点の範囲内であると言う条件下であればその限りではないと考えられる。

最後に射影変換に対してアフィン変換よりも、その性質が類似している擬似アフィン変換について検討した。この結果を図7に示す。しかし、この変換式が原データに対して適合性がよくないことはこの図によっても理解されるところである。擬似アフィン変換はアフィン変換と異なり、消失点を持つ変換となることから射影変換との類似性が伺える。しかし、変換データに関しては非常に異なる結果であった。したがって、画像データを採取するための変換式としての適正はないと判断される。このことは、図8の回帰式の係数が0.021と非常に大きな値を示していることでも明らかである。特に、偏位修正のための変換式としての適正は低いと判断される。

まとめ

本研究の目的は射影変換式を画像処理に応用し実用化することであった。そのために、その係数の決定方法を吟味し2つの方法について比較検討した。これらの結果を以下に示す。

- 1) 射影変換は擬似アフィン変換と比較して誤差が少なく偏位修正を行うには、適合がよいと考えられる。
- 2) 校正点の数を4点以上撮影し最小2乗法を使用して変換式を立てた方が誤差が減少する。
- 3) 射影変換を用いた座標変換は校正点からの距離の2乗に比例して誤差が増大する傾向があると考えられる。
- 4) 校正点を画面に対して広範囲にとった方が全体の誤差が減少すると予測される。射影変換による座標変換をおこなうことによって底周波の誤差が減少と言われているが、カメラのレンズ系の画面周差による誤差に対しては高次多項式等の変換を考える必要があると考えられる。

引用参考文献

- 1) Ae M., Sakatani Y., Yokoi T., Hashihara Y., and Shibukawa K. "Biomechanical analysis of preparatory motion for takeoff in the fosbury flop", *International journal of sport biomechanics*, Vol. 2:66-77, 1986.
- 2) 安藤幸司, 「高速度写真撮影手法による写真計測法」 *J. J. sports sci.* 2-3:200-11, 1983.
- 3) Feltner M. and Dapena J. "Dynamics of the shoulder and elbow joints of the throwing arm during a baseball pitch", *International journal of sport biomechanics*, Vol. 2:235-59, 1986.
- 4) 小林一敏・大島義晴「映像解析におけるフィルタリング」 *J. J. sports sci.* 2-3:172-81, 1983.
- 5) 小林一敏「射影変換による画像ノイズの除去」 *日本体育学会第32回大会号*, 401, 1981.
- 6) 小林一敏「画像雑音のfiltering」 *日本体育学会第32回大会号*, : 87, 1981.
- 7) Luethi S. M., Frederick E. C., Hawes M. R. and Nigg B. M. "Influence of shoe construction on lower extremity kinematics and load during lateral movements in tennis", *International journal of sport biomechanics*, Vol. 2:166-74, 1986.
- 8) McGinnis P. M. and Bergman L. A. "An inverse dynamic analysis of the pole vault", *International journal of sport biomechanics*, Vol. 2:186-201, 1986.
- 9) Miller J. A. Jr. and Hay J. G. "Kinematics of a world record and other world-class performances in the triple jump", *International journal of sport biomechanics*, Vol. 2:272-88, 1986.
- 10) 南茂夫「科学計測のための波形データ処理」 CQ出版, 1986, pp. 84-110.
- 11) 中川徹・小柳義夫「最小自乗法による実験データ解析」 *東京大学出版会*, 1984, pp. 28-39.
- 12) 大島義晴・小林一敏「映画分析におけるノイズの周波数特性」 *日本体育学会第32回大会号*, 399, 1981.
- 13) 大道等「写真解析の読み取り誤差と微分演算」 *J. J. sports sci.* Vol. 2:182-99, 1983.
- 14) 坂田俊文「コンピュータイメージング」 *写真工業出版社*, 1983, pp98-115.
- 15) Winter D. A. "Biomchanics of human movement", A wiley-interscience publication, 1979, pp9-38.

資料1

```

1000      renum 1000
1010
1020      save "mat2GAU",a
1030
1040      射影変換      ガウスの消去法
1050
1060      u=F(x,y)
1070      v=G(x,y)
1080
1090      N=8
1100      DIM A(N,N+1)
1110
1120      校正点の座標の入力
1130
1140      PRINT " control-point"
1150      PRINT
1160      FOR I=0 TO 3
1170          PRINT "no ";I
1180          PRINT
1190              INPUT "screen x= ", X
1200              INPUT "screen y= ", Y
1210              INPUT "real   u= ", U
1220              INPUT "real   v= ", V
1230
1240      8行9列の行列を作る
1250
1260          K=I*2+1
1270          A(K,1)=X
1280          A(K,2)=Y
1290          A(K,3)=1
1300          A(K,4)=0
1310          A(K,5)=0
1320          A(K,6)=0
1330          A(K,7)=-U*X
1340          A(K,8)=-U*Y
1350          A(K,9)=U
1360
1370          K=I*2+2
1380          A(K,1)=0
1390          A(K,2)=0
1400          A(K,3)=0
1410          A(K,4)=X
1420          A(K,5)=Y
1430          A(K,6)=1
1440          A(K,7)=-V*X
1450          A(K,8)=-V*Y
1460          A(K,9)=V
1470      PRINT
1480      NEXT I
1490
1500      ガウスの消去法を用いて方程式を解く
1510
1520      FOR K=1 TO N
1530          FOR I=K TO N
1540              IF A(I,K)<>0 THEN 1580
1550          NEXT I
1560          GOTO 1980
1570
1580          P=A(I,K)
1590          FOR J=K TO N+1
1600              A1=A(I,J)
1610              A(I,J)=A(K,J)

```

・ 疑似アフィン変換時の右辺
 ・ X*Y
 ・ X
 ・ Y
 ・ 1
 ・ 0
 ・ 0
 ・ 0
 ・ 0
 ・ 0
 ・ U
 ・ 0
 ・ 0
 ・ 0
 ・ 0
 ・ 0
 ・ X*Y
 ・ X
 ・ Y
 ・ 1
 ・ V

```

1620     A(K,J)=A1/P
1630     NEXT J
1640 *
1650     FOR I=1 TO N
1660     IF I=K THEN 1710
1670     Q=A(I,K)
1680     FOR J=K TO N+1
1690     A(I,J)=A(I,J)-Q*A(K,J)
1700     NEXT J
1710     NEXT I
1720     NEXT K
1730 *
1740 *   求められた係数の表示
1750 *
1760     FOR I=1 TO N
1770     PRINT "x";I;"=",A(I,N+1)
1780     NEXT I
1790 *
1800 *   変換を行う
1810 *
1820     INPUT "x=",X
1830     INPUT "y=",Y
1840 *
1850 *   射影変換の変換式
1860     U=A(1,9)*X+A(2,9)*Y+A(3,9)/(A(7,9)*X+A(8,9)*Y+1)
1870     V=A(4,9)*X+A(5,9)*Y+A(6,9)/(A(7,9)*X+A(8,9)*Y+1)
1880 *
1890 *   疑似アフィン変換時の変換式
1900 *   u=A(1,9)*X+Y+A(2,9)*X+A(3,9)*Y+A(4,9)
1910 *   v=A(5,9)*X+Y+A(6,9)*X+A(7,9)*Y+A(8,9)
1920 *
1930     PRINT "u=";U
1940     PRINT "v=";V
1950     GOTO 1790
1960 END
1970 *   消去法の式が求まらない時の処理
1980     PRINT "解が求まりません"
1990 END

```

資料2

```

1000 '      renum 1000
1010 '
1020 '      save "b:mat2QR.bas",a
1030 '
1040 '
1050 '      射影変換  Q R 法による解法
1060 '
1070 '      u=f(x,y)
1080 '      v=g(x,y)
1090 '
1100 '      校正点の数の入力
1110 '
1120 CLS
1130 INPUT "number of control-points < ",NP
1140 IF NP<4 THEN GOTO 1130
1150 '
1160 '      初期設定  配列
1170 '
1180 N=2*NP
1190 M=8
1200 '                                y(i)=x(i,j)*b(j)
1210 DEFDBL X,Y,B,Q,R,C,V,S,T
1220 '
1230 DIM X(N,M)      'x(N,M)=q(N,M)*r(M,M)
1240 DIM Y(N)
1250 DIM B(M)
1260 DIM Q(N,M)
1270 DIM R(M,M)
1280 DIM C(M)
1290 DIM V(N)
1300 '
1310 '
1320 '      校正点の座標の入力
1330 '
1340 PRINT
1350 FOR I=1 TO NP
1360 PRINT "no ";I
1370 PRINT
1380 INPUT "screen  x= ", X
1390 INPUT "screen  y= ", Y
1400 INPUT "real    u= ", U
1410 INPUT "real    v= ", V
1420 PRINT
1430 '
1440 K=I*2-1                                '疑似アフィン変換時の右辺
1450 X(K, 1)=X                                'x*y
1460 X(K, 2)=Y                                'x
1470 X(K, 3)=1                                'y
1480 X(K, 4)=0                                '1
1490 X(K, 5)=0                                '0
1500 X(K, 6)=0                                '0
1510 X(K, 7)=-U*X                              '0
1520 X(K, 8)=-U*Y                              '0
1530 Y(K) =U                                    'u
1540 '
1550 K=I*2
1560 X(K, 1)=0                                '0
1570 X(K, 2)=0                                '0
1580 X(K, 3)=0                                '0
1590 X(K, 4)=X                                '0
1600 X(K, 5)=Y                                'x*y
1610 X(K, 6)=1                                'x

```

```

1620          X(K, 7)=-V*X          'y
1630          X(K, 8)=-V*Y          '1
1640          Y(K)   =V             'v
1650      NEXT I
1660
1670      配列 X(N,M) Y(N) の表示
1680
1690      FOR I=1 TO N
1700          FOR J=1 TO M
1710              PRINT X(I,J);
1720          NEXT J
1730          PRINT Y(I)
1740      NEXT I
1750
1760      FOR J=1 TO M
1770          S=0
1780          FOR I=1 TO N
1790              S=S+X(I,J)*Y(I)
1800          NEXT I
1810          C(J)=S
1820      NEXT J
1830
1840
1850      S=0
1860      FOR I=1 TO N
1870          S=S+X(I,1)^2
1880      NEXT I
1890      T=SQR(S)
1900      R(1,1)=T
1910      FOR I=1 TO N
1920          Q(I,1)=X(I,1)/T
1930      NEXT I
1940
1950      FOR K=2 TO M
1960          KM1=K-1
1970          FOR I=1 TO N
1980              V(I)=X(I,K)
1990          NEXT I
2000          FOR J=1 TO KM1
2010              S=0
2020              FOR I=1 TO N
2030                  S=S+V(I)*Q(I,J)
2040              NEXT I
2050              R(J,K)=S
2060              FOR I=1 TO N
2070                  V(I)=V(I)-S*Q(I,J)
2080              NEXT I
2090          NEXT J
2100
2110
2120          S=0
2130          FOR I=1 TO N
2140              S=S+V(I)^2
2150          NEXT I
2160          T=SQR(S)
2170          FOR I=1 TO N
2180              Q(I,K)=V(I)/T
2190          NEXT I
2200          R(K,K)=T
2210      NEXT K
2220
2230      配列 Q(N,M) の表示
2240
2250      FOR I=1 TO N
2260          FOR J=1 TO M
2270              PRINT Q(I,J),

```

```
2280     NEXT J
2290     PRINT
2300     NEXT I
2310 '
2320     FOR I=1 TO M
2330         S=0
2340         FOR J= 1 TO N
2350             S=S+Q(J,I)*Y(J)
2360         NEXT J
2370         C(I)=S
2380     NEXT I
2390 '
2400     B(M)=C(M)/R(M,M)
2410     FOR K=2 TO M
2420         I=M-K+1
2430         IP1=I+1
2440         S=C(I)
2450         FOR J=IP1 TO M
2460             S=S-R(I,J)*B(J)
2470         NEXT J
2480         B(I)=S/R(I,I)
2490     NEXT K
2500 '
2510         求められた係数 B(I) の表示
2520 '
2530     FOR I=1 TO M
2540         PRINT B(I)
2550     NEXT I
2560 '
2570 '
2580         変換を行なう
2590 '
2600     INPUT "x=",X
2610     INPUT "y=",Y
2620 '
2630     射影変換の変換式
2640     U=(X*B( 1)+Y*B( 2)+B( 3))/(X*B( 7)+Y*B( 8)+1)
2650     V=(X*B( 4)+Y*B( 5)+B( 6))/(X*B( 7)+Y*B( 8)+1)
2660 '
2670     疑似アフィン変換時の変換式
2680     U= X*Y*B( 1)+X*B( 2)+Y*B( 3)+B( 4)
2690     V= X*Y*B( 5)+X*B( 6)+Y*B( 7)+B( 8)
2700 '
2710     PRINT "u=";U
2720     PRINT "v=";V
2730     GOTO 2560
```