PAPER *Special Section of Papers Selected from ITC-CSCC'99*

Estimation of Camera Rotation Using Quasi Moment Features

Hiroyuki SHIMAI[†], **Toshikatsu KAWAMOTO**[†], ***Student Members***, **Takaomi SHIGEHARA**[†], **Taketoshi MISHIMA**[†], **Masaru TANAKA**[††], ***and*** **Takio KURITA**[††], ***Regular Members***

SUMMARY We present two estimation methods for camera rotation from two images obtained by the active camera before and after rotation. Based on the representation of the projected rotation group, quasi moment features are constructed. Camera rotation can be estimated by applying the singular value decomposition (SVD) or Newton's method to tensor quasi moment features. In both cases, we can estimate 3D rotation of the active camera from only two projected images. We also give some experiments for the estimation of the actual active camera rotation to show the effectiveness of these methods.

key words: *quasi moment feature, projected rotation group, spherical harmonics, active camera*

1. Introduction

Recently the active camera has been popular in various fields, because it is getting lower in cost and higher in its performance. It is quite important to estimate the camera rotation from two images obtained by itself before and after rotation for the self-calibration problem of the active camera and the control of a robot with vision system, etc.

Usually the estimation method for the camera rotation is based on the point matching [8] and/or moment features [9]. However, the point matching method has two problems, one is the difficulty of selecting the collateral points and the other is the unstableness under noises. The method based on moment features doesn't have these problems because they are the integral quantities. However the moment features have also two problems. One is that marginal regions of the screen contribute dominantly rather than the center of the screen. It causes the loss of the important image information because the target image is usually located at the center of the screen. The other is that, it is more serious, moment features can only estimate the rotation around Z axis along the light axis of the lens. Thus, they cannot estimate the rotation included pan and/or tilt, this is because pan and tilt can be represented by 3D rotations. Ordinally, moment features don't take the projection from 3D space onto 2D plane into consideration.

Manuscript received October 1, 1999.
Manuscript revised January 7, 2000.

[†]The authors are with Saitama University, Urawa-shi, 338-8570 Japan.
[††]The authors are with Electrotechnical Laboratory, Tsukuba-shi, 305-8568 Japan.

However, quasi moment features, which is presented by Tanaka [6] ([7] also gives the equal features up to the second order), could resolve these four problems. They are invariant or covariant under the projected rotation group, because they are constructed based on the representation of the projected rotation group. The projected rotation group consists of two successive transformations, projection and 3D rotation [3], [4]. Then the features would be quite useful to characterize an object under the projected rotation, like a camera rotation.

In this paper, we give two estimation methods with quasi moment features for the camera rotation. One is the method using by singular value decomposition (SVD) and the other is the method with Newton's method. We also show the effectiveness of two methods through some experiments.

This paper is organized as follows: in Sect. 2, the representation of the projected rotation group is given briefly. In Sect. 3, we consider the quasi moment features based on the representation of the projected rotation group. In Sect. 4, two estimation methods of camera rotation using tensor quasi moment features are considered. In Sect. 5, some experiments are given for the estimation of the camera rotation under both an ideal pin-hole camera and an actual active camera.

2. Representation of the Projected Rotation Group

In this section, we give the representation of the projected rotation group.

The projected rotation group consists of two successive transformations, that is, the projection and 3D rotation. The basis function of the representation of the projected rotation group are known as the spherical harmonics Y_ℓ^m [3], [4], which is well known in the context of Quantum Physics [1].

Assume the image function $F(x, y)$ is obtained through a projection of an object onto 2D plane $Z = f$ (see Fig. 1). In these coordinate system, we obtain the following basis function of the representation of the projected rotation group

$$Y_\ell^m \left(\tan^{-1} \frac{\sqrt{x^2 + y^2}}{f}, \tan^{-1} \frac{y}{x} \right). \tag{1}$$

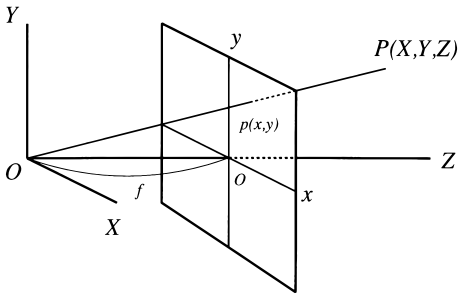


Fig. 1 The world coordinate system (X,Y,Z) and the screen coordinate system (x,y): The origin O is the center of the camera lens, f is a focal length of the camera and a plane $Z = f$ is the screen which is assigned xy-coordinate system. The point $P(X, Y, Z)$ in the 3D space is projected onto the point $p(x, y)$ in the screen.

Note that, if $\ell_1 \neq \ell_2$, the projected rotation group cannot mix the space S_1 and the space S_2, where the space S_1 is constructed by $\{Y_{\ell_1}^{m_1}\}_{m_1=-\ell_1,\ldots,\ell_1}$ and the space S_2 is constructed by $\{Y_{\ell_2}^{m_2}\}_{m_2=-\ell_2,\ldots,\ell_2}$. This representation is called irreducible representation. When the finite transformation is needed, we have to make successive infinitesimal transformations in order to get the finite projected rotation.

3. Quasi Moment Features

In the previous section, we gave the representation of the projected rotation group. The obtained representation is irreducible, that is, the linear spaces associated with $\{Y_\ell^m\}(\ell = 0, 1, 2, \ldots)$ are completely separated each other with respect to ℓ. This means that all invariant and covariant quantities under the projected rotation group can be obtained for every index ℓ. Therefore some quantities constructed with the linear combinations of the representation of the projected rotation group with the same index ℓ are invariant and/or covariant under the projected rotation group, too. In the following, the quasi moment features are defined by the linear combinations of the spherical harmonics with the same index ℓ in order to make invariant and covariant quantities into more familiar form.

First of all, we list the representation of the projected rotation group in our coordinate system, explicitly. For example, if $\ell = 0$, we have

$$Y_0^0 = \frac{1}{\sqrt{4\pi}}, \tag{2}$$

if $\ell = 1$,

$$Y_1^1 = -\frac{1}{4}\sqrt{\frac{6}{\pi}}\left(\frac{x}{k} + i\frac{y}{k}\right), \tag{3}$$

$$Y_1^0 = \frac{1}{2}\sqrt{\frac{3}{\pi}}\frac{f}{k}, \tag{4}$$

and if $\ell = 2$,

$$Y_2^2 = \frac{1}{8}\sqrt{\frac{30}{\pi}}\left\{\left(\frac{x}{k}\right)^2 + 2i\frac{x}{k}\frac{y}{k} - \left(\frac{y}{k}\right)^2\right\}, \tag{5}$$

$$Y_2^1 = -\frac{1}{4}\sqrt{\frac{30}{\pi}}\frac{f}{k}\left(\frac{x}{k} + i\frac{y}{k}\right), \tag{6}$$

$$Y_2^0 = -\frac{1}{4}\sqrt{\frac{5}{\pi}}\left\{-2\left(\frac{f}{k}\right)^2 + \left(\frac{x}{k}\right)^2 + \left(\frac{y}{k}\right)^2\right\}, \tag{7}$$

where $k = \sqrt{x^2 + y^2 + f^2}$. Note that for the negative value of m, we always use the following identity

$$Y_\ell^{-m} = (-1)^m Y_\ell^{m*}. \tag{8}$$

This identity makes it possible for us to consider only the case of the positive m.

In general, the representation of the projected rotation group, spherical harmonics, are known that they can be written in suitable linear combinations of

$$\left(\frac{x}{k}\right)^\alpha \left(\frac{y}{k}\right)^\beta \left(\frac{f}{k}\right)^\gamma, \tag{9}$$

where α, β and γ are non-negative integers. Therefore we can also represent Eq. (9) in terms of the representation of the projected rotation group with noting that

$$\left(\frac{x}{k}\right)^2 + \left(\frac{y}{k}\right)^2 + \left(\frac{f}{k}\right)^2 = 1. \tag{10}$$

For example, if $\alpha + \beta + \gamma = 0$,

$$1 = \sqrt{4\pi}Y_0^0, \tag{11}$$

if $\alpha + \beta + \gamma = 1$,

$$\frac{x}{k} = -2\sqrt{\frac{\pi}{6}}(Y_1^1 - Y_1^{-1}), \tag{12}$$

$$\frac{y}{k} = i2\sqrt{\frac{\pi}{6}}(Y_1^1 + Y_1^{-1}), \tag{13}$$

$$\frac{f}{k} = 2\sqrt{\frac{\pi}{3}}Y_1^0, \tag{14}$$

and if $\alpha + \beta + \gamma = 2$,

$$\left(\frac{x}{k}\right)^2 = \frac{1}{3} + 2\sqrt{\frac{\pi}{30}}\left(Y_2^2 + Y_2^{-2} - \frac{\sqrt{6}}{3}Y_2^0\right), \tag{15}$$

$$\left(\frac{y}{k}\right)^2 = \frac{1}{3} - 2\sqrt{\frac{\pi}{30}}\left(Y_2^2 + Y_2^{-2} + \frac{\sqrt{6}}{3}Y_2^0\right), \tag{16}$$

$$\left(\frac{f}{k}\right)^2 = \frac{1}{3} + \frac{4}{3}\sqrt{\frac{\pi}{5}}Y_2^0, \tag{17}$$

$$\frac{x}{k}\cdot\frac{y}{k} = -i2\sqrt{\frac{\pi}{30}}(Y_2^2 - Y_2^{-2}), \tag{18}$$

$$\frac{x}{k}\cdot\frac{f}{k} = -2\sqrt{\frac{\pi}{30}}(Y_2^1 - Y_2^{-1}), \tag{19}$$

$$\frac{y}{k}\cdot\frac{f}{k} = i2\sqrt{\frac{\pi}{30}}(Y_2^1 + Y_2^{-1}). \tag{20}$$

Because Eq. (9) can be considered more fundamental than original spherical harmonics for its familiar form, we can use Eq. (9) instead of spherical harmonics to extract invariant and covariant features from a given image.

Then the quasi moments of the order ℓ are defined by

$$m_\ell = \int\int \left(\frac{x}{k}\right)^p \left(\frac{y}{k}\right)^q \left(\frac{f}{k}\right)^r F(x,y)\mathrm{d}m(x,y), \tag{21}$$

$$k = \sqrt{x^2+y^2+f^2}, \tag{22}$$

where $\ell = p+q+r$ and p, q, r are non-negative integers, f is the focal length, $F(x,y)$ is an image obtained by the camera and $\mathrm{d}m(x,y)$ is the invariant measure,

$$\mathrm{d}m(x,y) = \frac{f\mathrm{d}x\mathrm{d}y}{\sqrt{(x^2+y^2+f^2)^3}}. \tag{23}$$

The order ℓ stands for the behavior of the spherical harmonics under the projected rotation. Therefore quasi moments are called scalar if $\ell = 0$, vector if $\ell = 1$, and tensor if $\ell = 2$. They can be given as follows

(i) the 0th order quasi moment

$$\boldsymbol{S} = \int\int S(x,y)F(x,y)\mathrm{d}m(x,y), \tag{24}$$

$$S(x,y) = 1, \tag{25}$$

(ii) the 1st order quasi moment

$$\boldsymbol{V} = \int\int V(x,y)F(x,y)\mathrm{d}m(x,y), \tag{26}$$

$$V(x,y) = \frac{1}{k}\begin{pmatrix} x \\ y \\ f \end{pmatrix}, \tag{27}$$

(iii) the 2nd order quasi moment

$$\boldsymbol{T} = \int\int T(x,y)F(x,y)\mathrm{d}m(x,y), \tag{28}$$

$$T(x,y) = \frac{1}{k^2}\begin{pmatrix} x^2 & xy & xf \\ yx & y^2 & yf \\ fx & fy & f^2 \end{pmatrix} \tag{29}$$

$$= V(x,y)V(x,y)^T, \tag{30}$$

and the trace of $T(x,y)$ is 1.

Note that the number of the independent ℓth order quasi moments is $2\ell+1$.

Under the camera rotation matrix $\boldsymbol{R}$ in 3D such as,

$$\boldsymbol{R} = \begin{pmatrix} R_{11} & R_{12} & R_{13} \\ R_{21} & R_{22} & R_{23} \\ R_{31} & R_{32} & R_{33} \end{pmatrix}, \tag{31}$$

these features have the following relations,

$$\boldsymbol{S}_2 = \boldsymbol{S}_1, \tag{32}$$

$$\boldsymbol{V}_2 = \boldsymbol{R}\boldsymbol{V}_1, \tag{33}$$

$$\boldsymbol{T}_2 = \boldsymbol{R}\boldsymbol{T}_1\boldsymbol{R}^T, \tag{34}$$

where $\boldsymbol{S}_1, \boldsymbol{V}_1, \boldsymbol{T}_1$ are the extracted features before the camera rotation, scalar, vector and tensor, respectively and $\boldsymbol{S}_2, \boldsymbol{V}_2, \boldsymbol{T}_2$ represent after the camera rotation.

4. Two Estimation Methods for Camera Rotation

In this section, we consider two kinds of the estimation methods for camera rotation with quasi moment features.

4.1 Estimation through Singular Value Decomposition (SVD)

The relation of tensor features $\boldsymbol{T}_1$ and $\boldsymbol{T}_2$ is

$$\boldsymbol{T}_2 = \boldsymbol{R}\boldsymbol{T}_1\boldsymbol{R}^T. \tag{35}$$

Then we solve the eigen equation with respect to $\boldsymbol{T}_2$, as follows

$$\begin{aligned} \det(\lambda\boldsymbol{E} - \boldsymbol{T}_2) &= \det\left(\lambda\boldsymbol{E} - \boldsymbol{R}\boldsymbol{T}_1\boldsymbol{R}^{\mathrm{T}}\right) \\ &= \det\left(\boldsymbol{R}(\lambda\boldsymbol{E} - \boldsymbol{T}_1)\boldsymbol{R}^{\mathrm{T}}\right) \\ &= \det\boldsymbol{R}\,\det(\lambda\boldsymbol{E} - \boldsymbol{T}_1)\,\det\boldsymbol{R}^{\mathrm{T}} \\ &= \det(\lambda\boldsymbol{E} - \boldsymbol{T}_1), \end{aligned} \tag{36}$$

where $\boldsymbol{R}$ is 3D rotation of the active camera, and $\det\boldsymbol{R} = 1$.

This result shows that $\boldsymbol{T}_1$ and $\boldsymbol{T}_2$ have same eigenvalues. Furthermore, they are real symmetric matrices as $\boldsymbol{T} = \boldsymbol{T}^T$. Therefore they are diagonalizable with orthogonal matrices $\boldsymbol{U}_1$ and $\boldsymbol{U}_2$, respectively. Thus we have following relations

$$\boldsymbol{T}_1 = \boldsymbol{U}_1\boldsymbol{D}_1\boldsymbol{U}_1^T,\ \boldsymbol{T}_2 = \boldsymbol{U}_2\boldsymbol{D}_2\boldsymbol{U}_2^T, \tag{37}$$

where $\boldsymbol{D}_1$ and $\boldsymbol{D}_2$ are diagonal matrices, that is, their diagonal elements are eigenvalues. Then the relation of diagonal matrices is

$$\boldsymbol{D}_1 = \boldsymbol{D}_2. \tag{38}$$

In this case, the relation between $\boldsymbol{T}_1$ and $\boldsymbol{T}_2$ is rewritten as

$$\begin{aligned}\boldsymbol{T}_2 &= \boldsymbol{U}_2\boldsymbol{U}_1^T\boldsymbol{T}_1\boldsymbol{U}_1\boldsymbol{U}_2^T \\ &= (\boldsymbol{U}_2\boldsymbol{U}_1^T)\boldsymbol{T}_1(\boldsymbol{U}_2\boldsymbol{U}_1^T)^T.\end{aligned} \tag{39}$$

Then the rotation matrix $\boldsymbol{R}$ is computed as

$$\boldsymbol{R} = \boldsymbol{U}_2\boldsymbol{U}_1^T. \tag{40}$$

All of the camera rotation in 3D can be represented by one rotation axis $\boldsymbol{n}$ and one rotation angle θ. They can be calculated from the rotation matrix $\boldsymbol{R}$, such that,

$$\cos\theta = \frac{1}{2}(\mathrm{Tr}\boldsymbol{R} - 1), \tag{41}$$

$$\boldsymbol{n} \propto \begin{pmatrix} R_{23} - R_{32} \\ R_{31} - R_{13} \\ R_{12} - R_{21} \end{pmatrix} \tag{42}$$

Note that $\boldsymbol{U}_1$ and $\boldsymbol{U}_2$ are constructed by arranging the eigenvector corresponding to the largest eigenvalue to the first column, the eigenvector corresponding to the second largest eigenvalue to the second column and the eigenvector corresponding to the third largest eigenvalue to the third column.

4.2 Estimation through Newton's Method

We give another estimation method using Newton's method.

SVD is likely to increase the error because it requires larger amount of calculation than Newton's method. Thus, Newton's method seems to give more acculate results. However, it is not so easy to get the procedure of Newton's method under the full 3D rotation. Actually, when we use an actual active camera, the camera doesn't rotate around Z axis along the light axis of its lens. Therefore, it doesn't cause any problem in actual use to restrict the Newton's method to pan and tilt of camera. So Newton's method is used for the subset of the 3D rotation which consisits of the rotations around X axis and/or Y axis.

In this case, the camera rotation $\boldsymbol{R}$ can be represented by

$$\boldsymbol{R} = \boldsymbol{R}_Y\boldsymbol{R}_X, \tag{43}$$

where $\boldsymbol{R}_X$ is the rotation around X axis and $\boldsymbol{R}_Y$ is the rotation around Y axis, such that

$$\boldsymbol{R}_X = \begin{pmatrix} 1 & 0 & 0 \\ 0 & \cos\theta_X & -\sin\theta_X \\ 0 & \sin\theta_X & \cos\theta_X \end{pmatrix}, \tag{44}$$

$$\boldsymbol{R}_Y = \begin{pmatrix} \cos\theta_Y & 0 & \sin\theta_Y \\ 0 & 1 & 0 \\ -\sin\theta_Y & 0 & \cos\theta_Y \end{pmatrix}, \tag{45}$$

where θ_X is the angle of rotation around X axis and θ_Y is the angle of rotation around Y axis. Then we want to estimate θ_X and θ_Y through Newton's method.

From the relation of tensor Eq. (34), we have

$$\begin{aligned}\boldsymbol{T}_2 &= \boldsymbol{R}\boldsymbol{T}_1\boldsymbol{R}^T \\ &= (\boldsymbol{R}_Y\boldsymbol{R}_X)\boldsymbol{T}_1(\boldsymbol{R}_Y\boldsymbol{R}_X)^T \\ &= \boldsymbol{R}_Y\boldsymbol{R}_X\boldsymbol{T}_1\boldsymbol{R}_X^T\boldsymbol{R}_Y^T.\end{aligned} \tag{46}$$

Thus we have

$$\boldsymbol{R}_Y^T\boldsymbol{T}_2\boldsymbol{R}_Y = \boldsymbol{R}_X\boldsymbol{T}_1\boldsymbol{R}_X^T. \tag{47}$$

Let the energy function E be

$$E = ||\boldsymbol{R}_Y^T\boldsymbol{T}_2\boldsymbol{R}_Y - \boldsymbol{R}_X\boldsymbol{T}_1\boldsymbol{R}_X^T||^2. \tag{48}$$

Then the problem becomes the minimization of the energy function E respect to θ_X and θ_Y. For minimizing E, we should solve the following simultaneous equations

$$f(\theta_X, \theta_Y) = \frac{\partial E}{\partial \theta_X} = 0, \tag{49}$$

$$g(\theta_X, \theta_Y) = \frac{\partial E}{\partial \theta_Y} = 0. \tag{50}$$

The Newton's method gives us the following procedures

$$\theta_X^{(n+1)} = \theta_X^{(n)} + \Delta\theta_X, \tag{51}$$

$$\theta_Y^{(n+1)} = \theta_Y^{(n)} + \Delta\theta_Y, \tag{52}$$

where

$$\Delta\theta_X = -\frac{f\frac{\partial g}{\partial \theta_Y} - g\frac{\partial f}{\partial \theta_Y}}{\frac{\partial f}{\partial \theta_X}\frac{\partial g}{\partial \theta_Y} - \frac{\partial f}{\partial \theta_Y}\frac{\partial g}{\partial \theta_X}}, \tag{53}$$

$$\Delta\theta_Y = -\frac{g\frac{\partial f}{\partial \theta_X} - f\frac{\partial g}{\partial \theta_X}}{\frac{\partial f}{\partial \theta_X}\frac{\partial g}{\partial \theta_Y} - \frac{\partial f}{\partial \theta_Y}\frac{\partial g}{\partial \theta_X}}. \tag{54}$$

Then we can estimate the rotation angle by using this recursive procedure until satisfying a following condition

$$|\Delta x| + |\Delta y| < \varepsilon, \tag{55}$$

where ε is a tolerance.

5. Experiments

In this section, we give some experiments for the estimation of the camera rotation, angle and axis. Here we consider both types of estimation methods, SVD and Newton's method.

First, we consider the estimation under the ideal

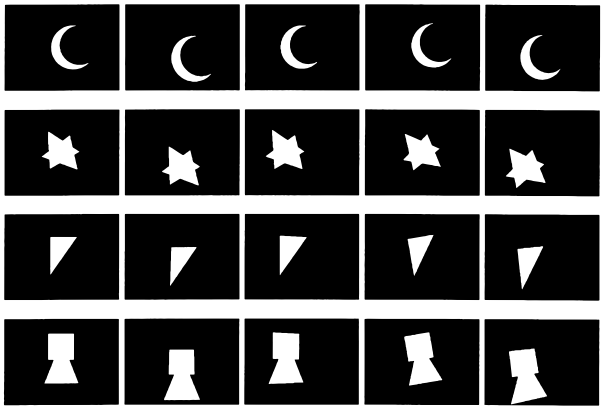


Fig. 2 Artificial images for the experiments: From top to bottom, Moon, Star, Triangle and Tumulus. From left to right, the original image is used as before camera rotation, the actual rotation is 10° around X axis, 10° around Y axis, 10° around Z axis and the successive 10° rotation around X,Y,Z axis in the order.

Table 1 The estimation results using SVD for the artificial images: 10° rotation around X axis.

	θ	X	Y	Z
actual rotation	10.00	1.00	0.00	0.00
Moon	9.96	1.00	0.00	-0.02
Star	9.98	1.00	0.00	-0.01
Triangle	9.91	1.00	0.01	-0.01
Tumulus	10.04	1.00	0.00	-0.03

camera condition such as pin-hole camera. The artificial images used this experiment are made by ray-tracing software POVRAY, and they are binary images such as black and white (see Fig. 2). The considered images are all planar ones so that they have no thickness. Then, it prevent from appearing the another aspect in the image by the rotation. The object should be kept within the screen in order to get the complete image of the object. The coordinate system is that Z axis is perpendicular to the screen, X axis is parallel to the horizontal axis of the screen and Y axis is parallel to the vertical axis of the screen. We made four types of images, Moon, Star, Triangle and Tumulus, and let each image rotate 10° around X axis, 10° around Y axis, 10° around Z axis and rotate successively 10° around X, Y, Z, axis in the order.

The estimation results using SVD are shown in Tables 1–4, where θ is the rotation angle and X,Y and Z are elements of rotation axis. From these tables, we see that the rotations around any axes are well estimated because the errors between the actual rotation angle and the estimated angle are less than 2% of the actual rotation angle.

The estimation results using Newton's method are shown in Table 5 and the corresponding rotation angle and axis are given in Tables 6–8 in order to compare with the results for SVD. From these tables, we see that the errors of rotation angle are small, which less than 3% of the actual rotation angle, and the rotation

Table 2 The estimation results using SVD for the artificial images: 10° rotation around Y axis.

	θ	X	Y	Z
actual rotation	10.00	0.00	1.00	0.00
Moon	10.03	0.00	1.00	0.01
Star	10.01	0.00	1.00	-0.02
Triangle	10.08	0.01	1.00	-0.01
Tumulus	9.95	0.00	1.00	-0.03

Table 3 The estimation results using SVD for the artificial images: 10° rotation around Z axis.

	θ	X	Y	Z
actual rotation	10.00	0.00	0.00	1.00
Moon	9.84	0.00	0.00	1.00
Star	9.93	0.00	0.00	1.00
Triangle	10.03	-0.01	-0.01	1.00
Tumulus	9.82	0.00	0.00	1.00

Table 4 The estimation results using SVD for the artificial images: successive 10° rotation around X,Y,Z axis in the order.

	θ	X	Y	Z
actual rotation	16.79	0.54	0.64	0.54
Moon	16.82	0.54	0.64	0.54
Star	16.69	0.55	0.65	0.53
Triangle	16.83	0.54	0.65	0.54
Tumulus	16.65	0.55	0.65	0.53

Table 5 The estimation results using Newton's method for the artificial images: X10 shows 10° rotation around X axis, Y10 shows 10° rotation around Y axis and XY10 shows successive 10° rotation around X,Y axis in the order.

	X10		Y10		XY10	
actual rotation	θ_X	θ_Y	θ_X	θ_Y	θ_X	θ_Y
	10.00	0.00	0.00	10.00	10.00	10.00
Moon	9.91	0.03	-0.01	10.02	9.76	10.29
Star	9.91	-0.01	0.01	10.00	9.74	10.15
Triangle	9.76	-0.19	0.14	10.07	9.60	9.98
Tumulus	10.03	-0.03	0.00	9.93	9.81	10.06

Table 6 The estimation results using Newton's method for the artificial images: 10° rotation around X axis.

	θ	X	Y	Z
actual rotation	10.00	1.00	0.00	0.00
Moon	9.91	1.00	0.00	0.00
Star	9.91	1.00	0.00	0.00
Triangle	9.76	1.00	-0.02	0.00
Tumulus	10.03	1.00	0.00	0.00

axis is estimated very accurately than that in the case of SVD. One of the considerable reasons of this is that the calculation amount of SVD routine is bigger than that of Newton's method.

Next, we give the estimation results for the actual active camera. The camera used in these experiments is SONY EVI-D30. EVI-D30 can rotate from $-100°$ to 100° for pan and from $-25°$ to 25° for tilt through RS232C interface. The prepared objects are planar Pigeon, Board in sponge, puppet of Cat and bottle of

Table 7 The estimation results using Newton's method for the artificial images: 10° rotation around Y axis.

	θ	X	Y	Z
actual rotation	10.00	0.00	1.00	0.00
Moon	10.02	0.00	1.00	0.00
Star	10.00	0.00	1.00	0.00
Triangle	10.07	0.01	1.00	0.00
Tumulus	9.93	0.00	1.00	0.00

Table 8 The estimation results using Newton's method for the artificial images: successive 10° rotation around X,Y axis in the order.

	θ	X	Y	Z
actual rotation	14.13	0.71	0.71	-0.06
Moon	14.17	0.69	0.72	-0.06
Star	14.06	0.69	0.72	-0.06
Triangle	13.84	0.69	0.72	-0.06
Tumulus	14.04	0.70	0.71	-0.06

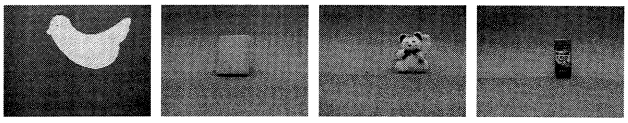


Fig. 3 The original images obtained by an active camera: From left to right, Pigeon, Board, Cat, Coffee.

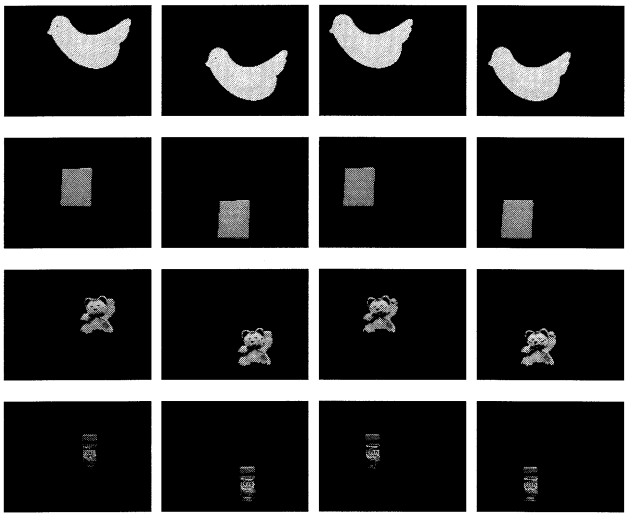


Fig. 4 Images without background for the experiments: From top to bottom, Pigeon, Board, Cat and Coffee. From left to right, original image used as before camera rotation, the actual rotation is 10° around X axis, 10° around Y axis and the successive 10° rotation around X,Y axis.

Coffee (see Fig. 3). They put on the blue background and at 60 centimeter distant from the camera. The original images are taken in color. Then the blue background is removed from each image and the images are transformed into gray scale images. They are rotated 10° around X axis, 10° around Y axis and rotated successively 10° around X, Y in the order. Each image is shown in Fig. 4.

Table 9 The estimation results using SVD for the actual active camera: 10° rotation around X axis.

	θ	X	Y	Z
actual rotation	10.00	1.00	0.00	0.00
Pigeon	9.98	1.00	0.00	0.07
Board	9.89	1.00	-0.01	-0.09
Cat	9.79	0.93	-0.01	-0.37
Coffee	9.76	0.99	-0.01	-0.11

Table 10 The estimation results using SVD for the actual active camera: 10° rotation around Y axis.

	θ	X	Y	Z
actual rotation	10.00	0.00	1.00	0.00
Pigeon	9.88	-0.01	0.97	0.26
Board	9.82	0.01	1.00	-0.02
Cat	9.51	-0.01	1.00	-0.05
Coffee	9.83	0.00	1.00	0.09

Table 11 The estimation results using SVD for the actual active camera: successive 10° rotation around X,Y axis in the order.

	θ	X	Y	Z
actual rotation	14.13	0.71	0.71	-0.06
Pigeon	13.91	0.71	0.69	0.08
Board	13.81	0.69	0.71	0.12
Cat	13.66	0.69	0.68	-0.25
Coffee	13.78	0.71	0.71	0.02

The estimation results for SVD are shown in Tables 9–11. From these tables, we see that the errors are very small and rotations around any axes are well estimated for the actual active camera, because the errors between the actual rotation angle and the estimated angle are less than 5% of the actual rotation angle, though the center of rotation does not coincide with the center of lens. The estimation of Cat's rotation is a little worse against others. It seems that the change of illumination of Cat depending on the rotation angle affects on this result, because Cat is susceptible of the lighting condition as it has a complex form. Generally speaking, the method based on SVD is sensitive against noise because it requires the large amount calculations than Newton's method.

The estimation results for Newton's method are shown in Table 12 and the corresponding rotation angle and axis are given in Tables 13–15 in order to compare with the results for SVD. From these tables, we see that the errors of rotation angle are small and the rotation axis is estimated more accurately than that in the case of SVD.

Now, we consider the effect of noise. Suppose to the case that each pixel of an image is blurred by the additive Gaussian noise $N(0, \sigma)$. In Eq. (21), the image $F(x, y)$ is changed into $\tilde{F}(x, y)$, where $\tilde{F}(x, y) = F(x, y) + N_{x,y}(0, \sigma)$. The change of quasi moment features by noise m_N is represented by

Table 12 The estimation results using Newton's method for the actual active camera: X10 shows 10° rotation around X axis, Y10 shows 10° rotation around Y axis and XY10 shows successive 10° rotation around X,Y axis in the order.

	X10		Y10		XY10	
actual rotation	θ_X	θ_Y	θ_X	θ_Y	θ_X	θ_Y
	10.00	0.00	0.00	10.00	10.00	10.00
Pigeon	10.13	0.02	-0.29	9.78	10.21	9.69
Board	9.79	-0.05	0.09	9.80	9.64	9.76
Cat	10.05	0.00	-0.34	9.48	10.20	9.46
Coffee	10.26	0.01	-0.09	9.81	10.34	9.79

Table 13 The estimation results using Newton's method for the actual active camera: 10° rotation around X axis.

	θ	X	Y	Z
actual rotation	10.00	1.00	0.00	0.00
Pigeon	10.13	1.00	0.00	0.00
Board	9.79	1.00	0.00	0.00
Cat	10.05	1.00	0.00	0.00
Coffee	10.26	1.00	0.00	0.00

Table 14 The estimation results using Newton's method for the actual active camera: 10° rotation around Y axis.

	θ	X	Y	Z
actual rotation	10.00	0.00	1.00	0.00
Pigeon	9.78	-0.03	1.00	0.00
Board	9.80	0.01	1.00	0.00
Cat	9.49	-0.04	1.00	0.00
Coffee	9.81	-0.01	1.00	0.00

Table 15 The estimation results using Newton's method for the actual active camera: successive 10° rotation around X,Y axis in the order.

	θ	X	Y	Z
actual rotation	14.13	0.71	0.71	-0.06
Pigeon	14.07	0.72	0.69	-0.06
Board	13.71	0.70	0.71	-0.06
Cat	13.90	0.73	0.68	-0.06
Coffee	14.23	0.72	0.69	-0.06

$$m_N = \int\int \left(\frac{x}{k}\right)^p \left(\frac{y}{k}\right)^q \left(\frac{f}{k}\right)^r N_{x,y}(0,\sigma)\mathrm{d}m(x,y). \tag{56}$$

Then, the expectation value of m_N is

$$E[m_N] = 0, \tag{57}$$

because each noise is added independently and they are independent of the location. Thus, quasi moment features are not affected by the independent Gaussian noise with the mean 0.

The estimation results with noise are shown in Figs. 5–6, Fig. 5 is the estimation result by SVD and Fig. 6 is the estimation result by Newton's method. The images used in the experiments are Tumulus (an ideal camera condition) and Cat (an actual active camera). The mean of Gaussian noise is 0 and its standard deviation σ is increased from 0 to 10. Each result is given by the mean of 10 trials.

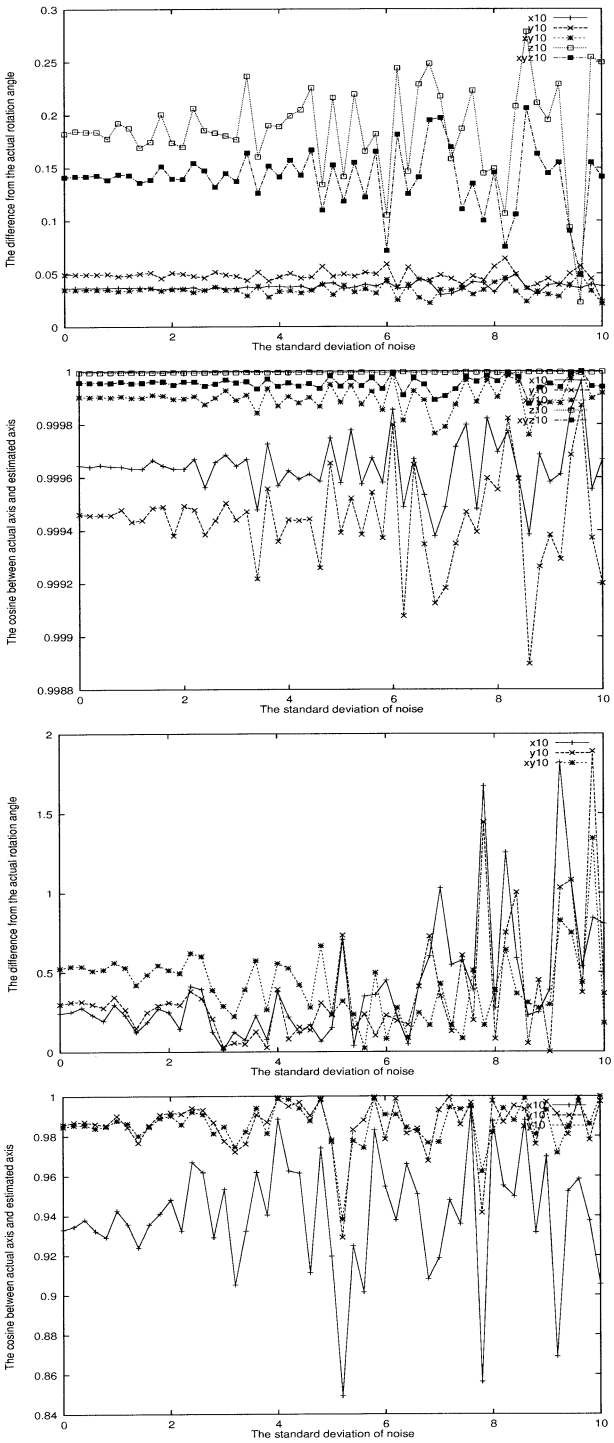


Fig. 5 The estimation results using SVD in the case where the Gaussian noise is added to each image independently: From top to bottom, the error of rotation angle for Tumulus (an ideal camera condition), the cosine between actual rotation axis and the estimated rotation axis for Tumulus, the error of rotation angle for Cat (an actual active camera) and the cosine between actual rotation axis and the estimeted rotation axis for Cat. The holizontal axis is the standard deviation of Gaussian noise and the vertical axis is the difference from the actual rotation angle for the rotation angle estimation, or the cosine between actual axis and the estimated axis for the rotation axis estimation.

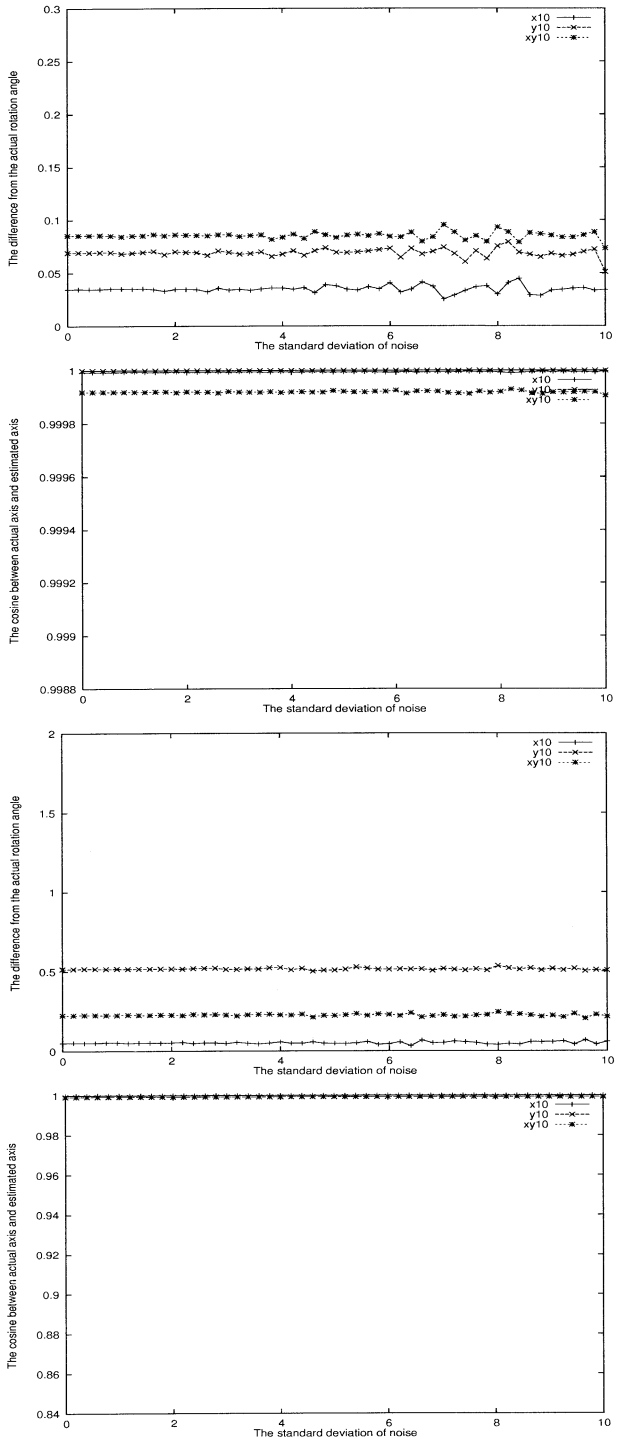


Fig. 6 The estimation results using Newton's method in the case where the Gaussian noise added to each image independently: From top to bottom, the error of rotation angle for Tumulus (an ideal camera condition), the cosine between actual rotation axis and the estimated rotation axis for Tumulus, the error of rotation angle for Cat (an actual active camera) and the cosine between actual rotation axis and the estimeted rotation axis for Cat. The holizontal axis is the standard deviation of Gaussian noise and the vertical axis is the difference from the actual rotation angle for the rotation angle estimation, or the cosine between actual axis and the estimated axis for the rotation axis estimation.

From these results, we see that both methods aren't influenced by the noise seriously.

6. Conclusion

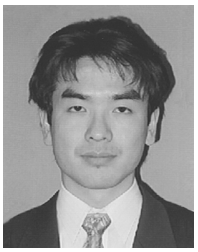
We have presented two kinds of estimation methods for the active camera rotation from only two images obtained before and after camera rotation. Some experiments have been given to show the effectiveness of these methods. First, the artificial images were considered. The estimation results show that the rotations around any axes are well estimated shown in Tables 1–4. Second, the images obtained from an actual active camera were considered. We compared two estimation method, one is based on SVD and the other is based on Newton's method. In both cases, the active camera rotation is well estimated shown in Tables 5–7,9–11.

Then, we considered the effect of noise. From the results in Figs. 5–6, we see that both methods aren't influenced by the noise seriously.

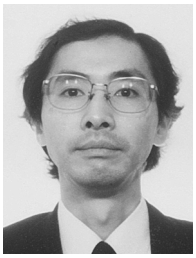
References

[1] G. Arfken, Mathematical methods for physicists, Second edition, Academic Press, 1970.
[2] M. Tanaka, "Image understanding via Galilei group," ETL Technical Report, TR-91-37, 1991.
[3] M. Tanaka, "On the representation of the projected motion group in 2D—Toward the further understandings," ETL Technical Report, TR-91-41, 1991.
[4] M. Tanaka, "On the representation of the projected motion group in 2+1D," Pattern Recognition Letters, vol.14, pp.671–678, 1993.
[5] M. Tanaka, T. Kurita, and S. Umeyama, "Image understanding via representation of the projected motion group," Pattern Recognition Letters, vol.15, pp.993–1001, 1994.
[6] M. Tanaka, "Finite transformation of quasi-moment," SPIE—Vision Geometry VI, vol.3168, 1997.
[7] K. Kanatani, "Camera rotation invariance of image characteristics," Computer Vision, Graphics, and Image Processing, vol.39, pp.328–354, 1987.
[8] C. Tomasi and T. Kanade, "The factorization method for the recovery of shape and motion from image streams," Proc. Image Understanding Workshop, pp.459–472, 1992.
[9] M. Takagi and H. Shimoda, eds., Handbook of Image Analysis, Univ. of Tokyo Press, 1991.

Hiroyuki Shimai received the B.Eng. from Saitama University in 1998. Now he is in Master course of engineering of Saitama University. His current interests are pattern recognition, image understanding, and biomimetic vision. He recieved the Young Engineer Award in 1999 from the IEICE.

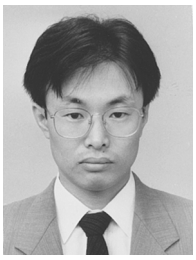


Toshikatsu Kawamoto received the B.Eng. from Saitama University in 1999. Now he is in Master course of engineering of Saitama University. His current interests are signal processing, image understanding, and music signal processing.

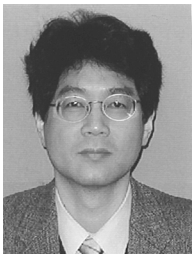


Takaomi Shigehara received the B.S., M.S. and Ph.D. degrees in Physics from the University of Tokyo in 1983, 1985 and 1988, respectively. He is currently an Associate Professor at the Department of Information and Computer Sciences, Saitama University. His research interests are quantum mechanics, quantum chaos, pattern recognition, high-performance computing, and numerical analysis.

Taketoshi Mishima received the B.E., M.E. and Ph.D. degrees in Electrical Engineering from Meiji University, in 1968, 1970 and 1973, respectively. He is currently a Professor at the Department of Information and Computer Sciences, Saitama University. His research interests are foundation of symbolic and algebraic computation, axiomatic logic system, mathematical pattern recognition, quantum chaos, and parallel computation.



Masaru Tanaka received B.Sc. degree in 1986, M.Sc. degree in 1988 and Dr.Sc. degree in 1991 from Kyushu University. He is a senior researcher in Information Science Division, Electrotechnical Laboratory and also an assistant professor of Osaka University since 1995. From 1995 to 1996, he was a visiting research scientist in Canada National Research Council. His current research interests are genetic algorithms, pattern recognition, biomimetic vision and information geometry. He is a member of the IEEE computer society, and the JNNS.



Takio Kurita received the B.Eng. degree from Nagoya Institute of Technology and the Dr. Eng. degree from University of Tsukuba, in 1981 and in 1993, respectively. Since 1981, he has been with the Electrotechnical Laboratory, AIST, MITI, Japan. From 1990 to 1991 he was a visiting research scientist at Institute for Information Technology, NRC, Ottawa, Canada. His current research interests include statistical methods, neural networks and their applications to pattern recognition. He is a member of the IEEE computer society, the IPSJ, the JNNS, and the Behaviormetric Society of Japan.