'90 KACC 1990. 10. 26~27

Model-Based Velocity Measurement Using Image Processing

Kohtaro OHBA, Tadashi ISHIHARA and Hikaru INOOKA

Department of Mechanical Engineering, Tohoku University,
Aramaki aza Aoba, Aoba-ku, Sendai, 980, Japan

Abstract

In this paper, we propose a model-based method of estimating the velocity of a moving object from a series of images. The proposed method utilizes Kalman filtering technique. Assuming that the motion is described by an affine transformation, we construct a discrete-time state variable model of the motion based on the dynamic motion imagery modeling technique proposed by Schalkoff. Using this state variable model, we derive a Kalman filter algorithm. Some simulation results are presented to show that the proposed Kalman filter algorithm is superior to a simple least square method without a model.

1. Introduction

Recently, several methods have been proposed to estimate the velocity of a moving object from a series of images [1] [2] [3]. Using these methods, we can measure the velocities of numerous points at the same time without special sensors. These methods have been used in several fields especially in flow visualization and motion analysis. However, most of the existing methods are susceptible to observation noise and require much computation time because no model for the motion is incorporated in the estimation algorithms.

In this paper, we propose a method of estimating the velocity utilizing a model of the motion. The proposed model-based estimation utilizes Kalman filter technique [4]. To derive a Kalman filter algorithm, we construct a discrete-time state variable model of the motion based on the dynamic motion imagery modeling technique proposed by Schalkoff [3]. This modeling technique describes the motion as a distributed parameter system. Assuming that the motion is described by an affine transformation, we can obtain simple relationship between the parameters in the affine transformation and the derivatives of the image intensity with respect to the location and time. This relationship can be used as a model of the observation process on the assumption that the derivatives are obtained by difference approximation. As a dynamic model of a signal to be estimated, we introduce a stochastic model for the variation of the parameters in the affine transformation. Based on this modeling of the image sequence, we can derive a Kalman filter algorithm to estimate the parameters in the affine transformation. We present some simulation results comparing the performance of the proposed algorithm with that of a simple least square method without a model. For a rigid object, we estimate translational and rotational motions with constant and time-varying rates. The results show that the proposed Kalman filter algorithm is superior to the least square method.

2. Modeling of Image Sequence

To utilize a Kalman filter algorithm, we must construct a state variable model consisting of a dynamical model for a signal to be estimated and a model for an observation process. As a preliminary of constructing the discrete-time state variable model, we derive a fundamental equation based on the dynamic imagery modeling proposed by Schalkoff [3]. First, we briefly review Schalkoff's modeling which describes the dynamic behavior of an image as a distributed parameter system (DPS).

We consider a 2D image of a moving object. Let $f(X,t)$ denote the image intensity of the location $X(x,y)$ at time t as a distributed parameter system. We assume that the image intensity of the moving object does not depend on time t. Therefore, the image intensity of the location $X(x,y)$ at time t is equal to that of a location $X_0(x_0,y_0)$ at the initial time t_0 as shown in Fig.1. Let $\xi(X,t,t_0)$ denote a time-varying geometric

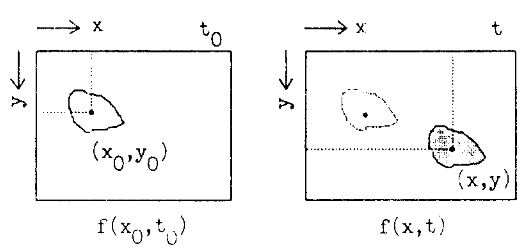


Figure 1: Time-varying spatial image.

transformation satisfying

$$
\begin{aligned}
f(X,t) &= f(X_0,t_0) \\
&= f(\xi(X,t,t_0),t_0).
\end{aligned} \tag{1}
$$

We assume that $\xi(X,t,t_0)$ is an affine transformation expressed as

$$\xi(X,t,t_0) = A(t,t_0)X + B(t,t_0), \tag{2}$$

where A is a 2×2 matrix consisting of rotational and translational parameters of the image, and B is 2×1 vector denoting translation. A wide class of image motions can be described by the affine transformation. We assume that the parameters contained in the matrices A and B in (2) are unknown.

Taking the partial derivatives of (1) with respect to the location X and time t, we get

$$\frac{\partial f(X,t)}{\partial t} = \frac{\partial f(X,t)}{\partial X}^T J_x^{-1} \frac{\xi(X,t,t_0)}{t}, \tag{3}$$

where J_x is time-varying Jacobian of transform function ξ. Using (2), the equation (3) can be written as

$$\frac{\partial f(X,t)}{\partial t} = \frac{\partial f(X,t)}{\partial X}^T [A^{-1}\dot{A}X + A^{-1}\dot{B}], \tag{4}$$

where $\dot{A}$ and $\dot{B}$ denote the partial derivatives of A and B with respect to t, respectively. This is a model of the image as a distributed parameter system.

For simplicity, we define the following matrices.

$$
\begin{aligned}
\bar{A} &= A^{-1}\dot{A} = \begin{pmatrix} \bar{a}_{11} & \bar{a}_{12} \\ \bar{a}_{21} & \bar{a}_{22} \end{pmatrix}, \\
\bar{B} &= A^{-1}\dot{B} = \begin{pmatrix} \bar{b}_1 \\ \bar{b}_2 \end{pmatrix}.
\end{aligned} \tag{5}
$$

Using (5), we can rewrite (4) as

$$f_t = \begin{pmatrix} xf_x & yf_x & xf_y & yf_y & f_x & f_y \end{pmatrix} \begin{pmatrix} \bar{a}_{11} \\ \bar{a}_{12} \\ \bar{a}_{21} \\ \bar{a}_{22} \\ \bar{b}_1 \\ \bar{b}_2 \end{pmatrix}, \tag{6}$$

where f_x, f_y and f_t denote the partial derivatives of the image intensity f with respect to x, y and t, respectively. In principle, the derivatives f_x, f_y and f_t are obtainable from the image data. For actual image data, the partial derivatives f_x and f_y are obtained from the difference of image intensities and f_t is obtained from the difference at the same point in two consecutive images by difference approximation.

We use the relation (6) as a fundamental relation to estimate the vector $(\bar{a}_{11}, \bar{a}_{12}, \bar{a}_{21}, \bar{a}_{22}, \bar{b}_1, \bar{b}_2)$ from the image data. We set an $n \times n$ pixels window in the image plane and assume that the parameter vector $(\bar{a}_{11}, \bar{a}_{12}, \bar{a}_{21}, \bar{a}_{22}, \bar{b}_1, \bar{b}_2)$ is constant within the window. More than six points in the window are required to determine the unknown parameters. Then, we can express the simultaneous equation as

$$Y_t = H_t \theta_t, \tag{7}$$

where

$$
\begin{aligned}
\theta_t &= \begin{pmatrix} \bar{a}_{11} & \bar{a}_{12} & \bar{a}_{21} & \bar{a}_{22} & \bar{b}_1 & \bar{b}_2 \end{pmatrix}^T, \\
Y_t &= \begin{pmatrix} f_t^1 & f_t^2 & \cdots & f_t^{n^2} \end{pmatrix}^T,
\end{aligned} \tag{8}
$$

$$H_t = \begin{pmatrix} x^1 f_x^1 & y^1 f_x^1 & x^1 f_y^1 & y^1 f_y^1 & f_x^1 & f_y^1 \\ x^2 f_x^2 & y^2 f_x^2 & x^2 f_y^2 & y^2 f_y^2 & f_x^2 & f_y^2 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ x^{n^2} f_x^{n^2} & y^{n^2} f_x^{n^2} & x^{n^2} f_y^{n^2} & y^{n^2} f_y^{n^2} & f_x^{n^2} & f_y^{n^2} \end{pmatrix}$$

and the subscript denotes the sampling time.

As a by-product, we show that, for the translational motion, the equation (7) reduced to a fundamental equation derived by a gradient-based method [2]. For the translational case, the transformation function ξ is written as

$$\xi(X,t,t_0) = X + B(t,t_0). \tag{9}$$

For the transformation (9), the equation (4) is reduced to

$$\frac{\partial f(X,t)}{\partial t} = \frac{\partial f(X,t)}{\partial X}^T \dot{B}. \tag{10}$$

Denoting $\dot{B} = [\dot{b}_1 \; \dot{b}_2]^T$, we can write the equation (10) as

$$f_t = \begin{pmatrix} f_x & f_y \end{pmatrix} \begin{pmatrix} \dot{b}_1 \\ \dot{b}_2 \end{pmatrix}. \tag{11}$$

For this motion, the relation (7) is reduced to

$$\begin{pmatrix} f_t^1 \\ f_t^2 \\ \vdots \\ f_t^{n^2} \end{pmatrix} = \begin{pmatrix} f_x^1 & f_y^1 \\ f_x^2 & f_y^2 \\ \vdots & \vdots \\ f_x^{n^2} & f_y^{n^2} \end{pmatrix} \begin{pmatrix} \dot{b}_1 \\ \dot{b}_2 \end{pmatrix}, \tag{12}$$

which is equivalent to the relation used in [2].

3. Kalman Filter Algorithm

We can solve the equation (7) by a least square method. The least square estimate is given by

$$\hat{\theta}_t = (H_t^T H_t)^{-1} H_t^T Y_t, \tag{13}$$

where H_t and Y_t are given in (8). It should be noted that this estimate is instantaneous in the sense that it does not depend on the whole past data although two consecutive images are required to calculate the time derivative f_t. Therefore this estimate does not utilize all the available data as well as a model for the parameter variation. To use the past data and incorporate a model for the parameter variation, we give a Kalman filter algorithm for the estimation of the unknown parameter θ_t. Using the Kalman filter algorithm, we can also incorporate *a priori* information about the observation noise. The sequential least square algorithm can be regarded as a special case of the Kalman filter algorithm.

We assume that the variation of the parameter θ is modeled by a linear stochastic difference equation

$$\theta_{t+1} = F_t \theta_t + W_t, \tag{14}$$

W_t is a zero-mean white noise sequence with a known covariance matrix Q_t. The transition matrix F_t and the covariance matrix of Q_t in (14) are determined by *a priori* information about the variation of the parameters.

Adding an observation noise to the equation (7), we obtain the observation equation

$$Y_t = H_t\theta_t + V_t, \tag{15}$$

where the observation noise V_t is a zero-mean white noise sequence with a known covariance matrix R_t. For the model of the motion described by the state variable model given by (14) and (15), we can obtain the following Kalman filter algorithm to estimate the parameter θ_t.

$$\begin{aligned}
\hat{\theta}_{t+1/t} &= F_t\hat{\theta}_{t/t}, \\
\hat{\theta}_{t/t} &= \hat{\theta}_{t/t-1}, + K_t\left(Y_t - H_t\hat{\theta}_{t/t-1}\right), \\
K_t &= P_{t/t-1}H_t^T, \left(H_t P_{t/t-1} H_t^T + R_t\right)^T, \\
P_{t+1/t} &= F_t P_{t/t} F_t^T + Q_t, \\
P_{t/t} &= P_{t/t-1} - K_t H_t P_{t/t-1}.
\end{aligned} \tag{16}$$

In the above algorithm, K_t is the Kalman gain matrix, and P is the covariance matrix of the estimation error. The subscript $t + 1/t$ denotes the estimate at time $t + 1$ based on the data available at time t.

The conventional sequential least square algorithm is obtained by setting $F_t = I$ and $Q_t = 0$. However, this choice cannot be recommended for the time-varying parameters. An *ad hoc* method to track a slowly varying parameter is to set $F_t = I$ and to choose sufficiently small Q_t.

Once the state estimate $\hat{\theta}_{t+1/t}$ is calculated by the algorithm, then the estimates of the motion parameters in (2) are easily obtained from (5). For the translational motion, the matrix A is identity and $\dot{B} = \bar{B}$ in (2). Then we can calculate the state value by (5). For the rotational motion centered in the window, the matrix B is zero and A is described as

$$A = \begin{pmatrix} \cos\omega & -\sin\omega \\ \sin\omega & \cos\omega \end{pmatrix}, \tag{17}$$

where ω is rotation angle in one sampling time. Using (5), we can calculate the rotation angle ω as

$$\omega = \sqrt{\det \bar{A}}. \tag{18}$$

4. Simulation Results

In this section, we give some simulation results to show the effectiveness of proposed algorithm. We consider a 2D image with the intensity

$$f(X,t) = max \times 0.25 \times \left\{\sin(\frac{2\pi x}{T}) + \sin(\frac{2\pi y}{T}) + 2\right\}, \tag{19}$$

where max denotes the maximum intensity of this image, and T denotes the time cycle of image intensity as shown in Figure 2. We set an observation point and a window around the point on the image plane. Then we translate and rotate this image around the window at constant rate.

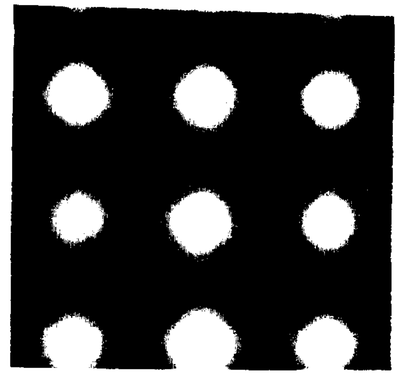


Figure 2: The simulation image data.

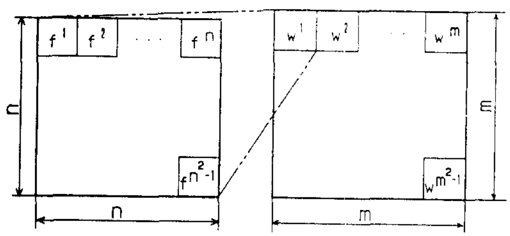


Figure 3: The window in image plane.

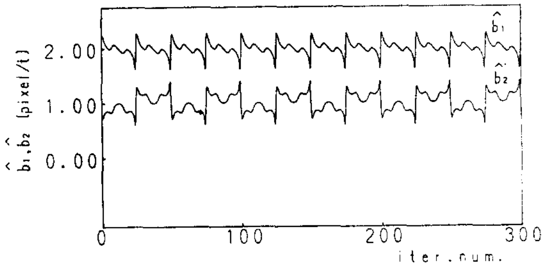
In actual image processing, analog image data is digitized by an A/D converter. In our simulation, we set max in (19) as 63 or 255, and the intensity is quantized into 64 or 256 grades (2^6 or 2^8 bits), respectively. The number of grades depends on the accuracy of A/D converter. The quantization the image data may cause some error in the estimation result.

To correct the estimation error caused by the quantization data, it is reasonable to average the quantized data f_t, f_x, f_y in the window. Then an $m \times m$ set of the windows consisting of $n \times n$ pixels is placed on the image as shown in Fig.3. In each window, the data f_t, f_x, f_y are averaged. We calculate the estimate using the $m \times m$ set of averaged data in each window instead of the quantized data.

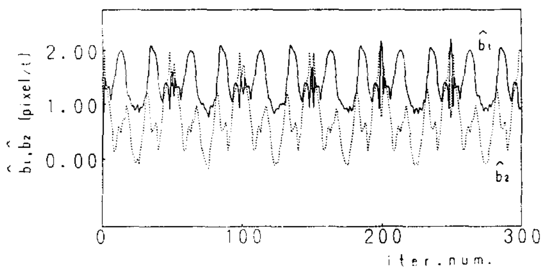
4.1. Least Square Estimation

The least square method is the simplest one to solve the equation like (7) or (12). We can get an estimate by calculating (13). In this method, the estimate depends on the image and location data at each time but does not utilize the whole data and a model for the the parameter variation. As is pointed out before, this method is equivalent to the gradient-based method proposed in [2]. To compare the estimation result with Kalman filter algorithm, we perform some simulations for translational motion.

Figure 4(a) shows the estimation result for the image

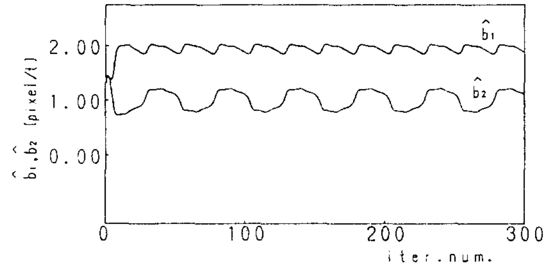


(a) Original data.

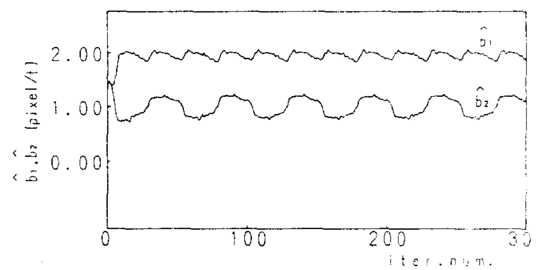


(b) Quantized data.

Figure 4: The estimation result by the least square method (m=1,n=7).



(a) Original data.



(b) Quantized data.

Figure 5: The estimation result by the Kalman filter algorithm (m=1,n=7).

data without quantization, where the movement of object is translation ($b_1 = 2$ [$pixel/t$], $b_2 = 1$ [$pixel/t$]), the window size is $n = 7$, the number of windows is $m = 1$, and $T = 50$ and $max = 63$ in equation (19). This unit [$pixel/t$] denotes the moving pitch of pixel in the sampling time of serial images. We use one window without averaging of the data. The axial line denotes the estimation result and horizontal line denotes the iteration number of estimation.

The estimation result shows an oscillation around the true value to be estimated. It is caused by the smallness of the derivatives (f_x, f_y) in the image data around the top or bottom of the image intensity distribution curve (19). In other words, we can not estimate the motion parameter by the least square method at the point where the gradient of intensity is small. This drawback of the least square method has already been pointed out in [2].

The result for a quantized data is shown in Fig.4(b). The estimates are very sensitive to the image data. This is because the least square method provides only an instantaneous estimate.

4.2. Kalman Filter Estimation

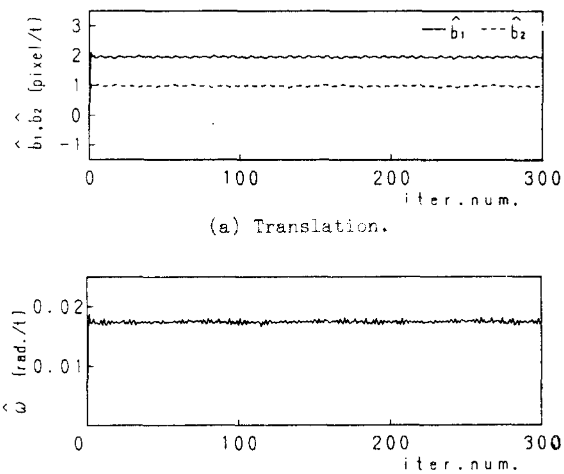
We give simulation results for the Kalman filter algorithm. In this simulation, we set $F_t = I$ in (14) because we move the simulation image data at constant rate. In the equation (16), we set each condition as follows

$$\begin{aligned} \hat{\theta}_{0/-1} &= 0, \\ P_{0/-1} &= \alpha_p I, \\ E[W_t\, W_t^T] &= Q_t = \alpha_q I, \\ E[V_t\, V_t^T] &= R_t = \alpha_r I. \end{aligned} \tag{20}$$

Then $\hat{\theta}_{0/-1}$ and $P_{0/-1}$ denote the initial condition for the estimation. Considering of the observation noise covariance caused by rounded data in our simulation, we set $\alpha_r = 0.25$ in our simulation. The coefficients α_p and α_q are determined by trail and error method as $\alpha_p = \alpha_q = 0.25$.

Figure 5 shows the estimation result without the quantization corresponding to Fig.4. We set the situation of estimation the same as previous one. Then the estimation result is calculated by (16) with (12). The estimation result shows an oscillation around the real value the same as the previous result. But remarkable reduction of the effect of the quantization is achieved compared with the least square method in Fig.4(b).

Furthermore, we put an $m \times m$ set of windows on the image to reduce the estimation error. Therefore the data is averaged in each $n \times n$ window. The result for $m = 7$ and $n = 7$ is shown in Figure 6. This estimation result is calculated by (16) with (7). In this result, we use the image data that has 256 grades of the intensity and time cycle of intensity is 50. Figures (a) and (b) show results for a translational case ($b_1 = 2$ [$pixel/t$], $b_2 = 1$ [$pixel/t$]) and for a rotational case ($\omega = 0.017$ [$rad./t$]). The estimates track the true values rapidly, and do not show large oscillation around the true value. This result is clearly better than those without averaging given in Fig.5.



(b) Rotation.

Figure 6: The estimation result by the Kalman filter algorithm (m=7,n=7).

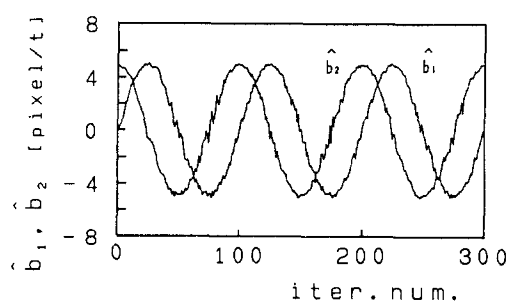


Figure 7: The estimation result for the moving parameter (m=7,n=7).

In the above simulation, we assume that the motion parameter is constant. The proposed algorithm can be applied for time varying motion parameters if they vary slowly compared with the width of the window. Figure 7 shows a simulation result for sinusoidal-like parameter variations

$$\begin{aligned} b_1 &= 5 \times \sin(\frac{2\pi k}{100}), \\ b_2 &= 5 \times \cos(\frac{2\pi k}{100}), \end{aligned} \tag{21}$$

where k denotes the iteration number. The estimates track the true value effectively. This shows that the proposed method is useful even if the motion parameter is not constant.

5. Conclusions

We have proposed a model-based estimation of the motion parameters from a series of images. This method utilizes Kalman filter algorithm which incorporates the model for the motion as well as various *a priori* information.

The simulation results for a constant translational motion show that the estimation results by the proposed Kalman filter algorithm are superior to those by the instantaneous least square method without a model. It is also shown by the simulation, the propose method can effectively be used to track slowly varying parameters.

In our future research, we plan to apply the proposed algorithm to a real dynamic model such as water flow.

References

[1] I.Kimura, T.Takamori and T.Inoue, *"Image Processing Instrumentation of Flow Velocity Vector Distribution by Using Correlation Technique", Trans.SICE. Vol.23, No.2, pp.101-107, 1987.*

[2] S.Ando, *"Gradient-Based Feature Extraction Operators for the Classification of Dynamical Images", Trans.SICE. Vol.20, No.12, pp.1330-1336, 1987.*

[3] R.J.Schalkoff, *"Dynamic Imagery Modeling and Motion Estimation Using Weak Formulation", IEEE Trans. Pattern Analysis and Machine Intelligence. Vol.PAMI-9, No.4, pp.578-583, 1987.*

[4] R.E.Kalman and R.S.Bucy, *"New Results in Linear Filtering and Prediction Theory", Trans. ASME, J. Basic Eng., Vol.82D, No.1, pp.95-108, 1961.*