Recursive Wiener Filter for Motion Parameter Estimation in Three-Parameter Motion Model

Pei-Chuan Liu and Wen-Thong Chang

***Abstract*—Motion compensation is used to reduce the displaced frame difference (DFD) during the video coding. To increase the accuracy of the point correspondence during the compensation, a three-parameter motion model is considered. The matching error can be significantly reduced as compared with that of the two-parameter block matching. To derive the parameters, a partial full search method is used. The full search is used when the zoom value is set at one. Otherwise, the gradient-based algorithm is used. Since the DFD is a nonlinear function of the image gradients and the motion parameters, a linearized model is considered. To eliminate the linearization error, Wiener filtering is used to smoothen the DFD to improve the convergence condition of the iterative gradient search. To make the gradient-based search more robust to the gradient variation, several gradient estimation methods are also compared.**

I. Introduction

In video coding, temporal redundancy is removed with motion compensation. The motion phenomenon between image objects in the image sequence is usually described with motion parameters that relate the coordinates of the same objects in different frames. The two-parameter motion model for video coding is based on the assumption of two-dimensional (2-D) translation motion. In many cases, it is not sufficient to describe the motion phenomenon due to the three-dimensional (3-D) motion of objects or the zoom and pan of the camera. Thus the advantage of motion compensation cannot be fully utilized. For this, a three-parameter motion model is investigated in this paper to show how the use of one extra parameter can improve the motion compensation to reduce the bit rate.

The three-parameter motion model was proposed in [1] with one parameter to describe the ratio of focal lengths before and after zooming and two parameters to describe the pan or translation motion of the camera. In [2], a four-parameter motion model is discussed. With four parameters, rotation effect is also considered. Since motion is a relative movement of objects and the camera, these parameters can also be used to describe the motion phenomenon of objects between two consecutive frames due to the 3-D motion of the moving objects. The purpose of this paper is to discuss methods to derive the parameters in a three-parameter model. Similar to the two-parameter case [3]–[12], point correspondence based on the luminance comparison is used as the basic mechanism to iteratively derive the motion parameters.

For block-based coding, common parameters are derived for a block of pixels with the aim to minimize the total displaced frame differences. With two parameters, the full search and other simplified searches such as three-step, conjugate gradient, etc., are widely used to derive the parameter values. To extend these search methods to the three-parameter case, however, extensive computation for interpolation of pixels corresponding to all the possible zoom values is needed. Thus, to efficiently derive the values of these parameters, a partial full search is used. The full search is applied when the zoom value is set at one. Otherwise, the gradient-based search is used. The gradient-based method is based on a signal representation model that describes the signal luminance as a function of the motion parameters and the signal gradient. In [1], a first-order gradient-based algorithm was considered.

With first-order approximation, the displaced frame difference (DFD) is described as the product of the image gradient and the parameter estimation error. This parameter estimation error is defined as the difference between the true parameter value and the current estimated one. By iteratively minimizing the DFD, the parameter estimation error can be gradually reduced to derive the desired motion parameters. The performance with this first-order gradient search, however, is not very satisfactory [1]. Variation of the parameters during the search process is commonly seen and sometimes the algorithm diverges. Thus, methods to improve the convergence condition of the gradient method are of importance for the three-parameter case.

The problems with the gradient-based method can be due to many factors such as the signal model, the image gradient, and the linearization process, etc. For this reason, it is not commonly used in the two-parameter case. To improve the performance of [1], solutions to the above three factors are considered in this paper. To deal with the high-order expansion terms during the linearization, the Wiener filtering is used to smoothen the DFD when linearization of the model is used. The purpose is to make the observed DFD better fits the linearized model that describes the relation between the DFD and the motion parameters.

The Wiener filtering [5]–[9] is based on the use of the second-order statistics. Thus, the correlation functions of the parameter estimation errors and the random process that models the high-order expansion terms are considered. In the two-parameter model, the motion parameters are usually assumed uncorrelated and a constant diagonal correlation matrix is used.

Manuscript received December 12, 1995; revised June 10, 1997 and July 20, 1998. This work was supported in part by the National Science Council, R.O.C., under Contract NSC87-2213-E009-096.

The authors are with the Department of Computer Engineering, National Chiao-Tung University, Shinchu, 300 Taiwan.

Publisher Item Identifier S 1051-8215(98)09290-8.

1051–8215/98$10.00 © 1998 IEEE

The motion parameters in the three-parameter motion model, however, are correlated. Therefore, the correlation matrix of the parameter errors is considered. Also, to make the gradient-based method more robust to the variation of the gradient estimation, another consideration is the estimation of the pixel gradients. The effect of the gradient accuracy on the parameter estimation is profound. Several gradient estimation methods are compared to make the algorithm more stable to the gradient variation.

For these regions containing edges and object boundaries, divergence of the gradient search still cannot be avoided. One of the reasons is that, in these areas, the assumed signal model does not fit the actual data, since one extra zoom factor is still not enough to describe the point movement. To deal with such situations, we degenerate the three-parameter model to the two-parameter model. Thus a full search with zoom factor set at one is used. With the partial full search method, the minimum DFD from the two searches, the gradient-based search and the full search, is used during the coding. With zoom factor set at one, the result is the same as that of the two-parameter block matching. Thus, the three-parameter model is used as another extra computation seeking a zoom value to improve the coding performance of the two-parameter full search.

To consider the use of the three-parameter motion model, the rest of this paper is organized as follows. In Section II we first briefly review the three-parameter motion model. The signal representation model that describes the relation between the motion parameters and the displaced frame difference is derived. Then, in Section III, the Wiener-based algorithm for parameter estimation is analyzed. Estimation of the related correlation matrices is also discussed. The effect of gradient on the estimation accuracy is discussed in Section IV; the combination of the gradient-based method and the degenerate two-parameter full search and its use for video coding are also shown in this section. Finally, a conclusion is made in Section V.

II. The Signal Representation Model

Based on the Taylor series expansion, a signal representation model is derived. The first-order signal representation model describes the displaced frame difference as a function of the gradient and the parameter estimation error plus an expansion error.

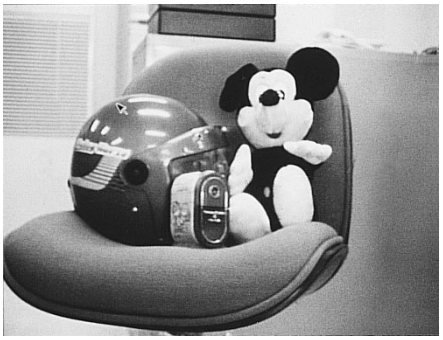
A. The Three-Parameter Motion Model

The three-parameter motion model can be described as the transformation of the coordinate

$$(x_2, y_2) = A(x_1, y_1) = (a_1 x_1 + a_2, a_1 y_1 + a_3) \qquad (1)$$

where (x_1, y_1) and (x_2, y_2) are the coordinates of an image point before and after movement in the image space. The parameter a_1 describes zooming effect and a_2 and a_3 describe panning phenomenon. When the parameter $a_1 = 1$, the three-parameter model will degenerate to the traditional two-parameter motion model with $a_2 = dx$ and $a_3 = dy$. To show the relation between (x_1, y_1) and (x_2, y_2), assuming that

(a)



(b)

Fig. 1. The simulated image (a) Laboratory and (b) Mickey.

a particular point $(X_1, Y_1 Z_1)$ has moved to a new position (X_2, Y_2, Z_2) with

$$\begin{aligned} X_2 &= X_1 + dX \\ Y_2 &= Y_1 + dY \\ Z_2 &= Z_1 + dZ \end{aligned} \qquad (2)$$

where (dX, dY, dZ) is the 3-D motion vector in the object space. According to the camera projection rule, in the image space, we have

$$x_1 = F \cdot X_1 / Z_1 \quad y_1 = F \cdot Y_1 / Z_1$$

and

$$x_2 = F \cdot X_2 / Z_2 \quad y_2 = F \cdot Y_2 / Z_2. \qquad (3)$$

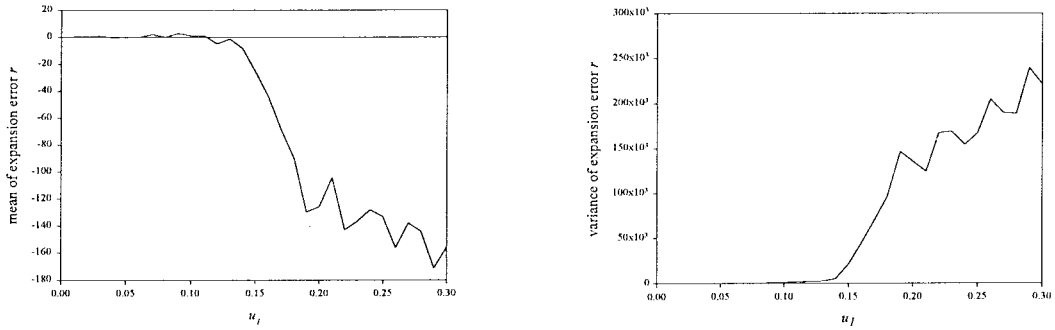
From (2) and (3), the relation between (x_1, y_1) and (x_2, y_2) is

$$\begin{aligned} x_2 &= a_1 x_1 + a_2 \\ y_2 &= a_1 y_1 + a_3 \end{aligned} \qquad (4)$$

with

$$\begin{aligned} k &= dz / z_1 \\ a_1 &= 1/(1+k) \\ a_2 &= F/(1+k) \cdot dX/Z_1 \\ a_3 &= F/(1+k) \cdot dY/Z_1. \end{aligned}$$

Equation (4) describes the relation between the coordinates of image points and the three parameters to be estimated.



(a) (b)

Fig. 2. The relationship of the mean and variance of the expansion error r versus the parameter estimation error u_1 for image "Laboratory." (a) The distribution of mean. (b) The distribution of variance (where u_2 and u_3 are set to be zero).

B. The Signal Representation Model

Based on the above motion model, we then discuss the signal that is used to describe the relationship between the luminance at two image points and the motion parameters. Based on this signal representation model, many algorithms can be derived to estimate the motion parameters from the luminance signal. Let the luminance at position (x, y) in frame k be denoted as $S_k(x, y)$. If the motion model is assumed to be $A(x, y) = (a_1x + a_2, a_1y + a_3)$, then the relationship between $S_k(x, y)$ and $S_{k+1}(x, y)$ can be described as

$$S_{k+1}(x, y) = S_k(A(x, y)). \tag{5}$$

The DFD is relative to $\hat{a}_1, \hat{a}_2$, and $\hat{a}_3$ as

$$\begin{aligned}\mathrm{DFD}(x, y) &= S_{k+1}(x, y) - S_k\left(\hat{A}(x, y)\right) \\ &= S_{k+1}(x, y) - S_k(\hat{a}_x + \hat{a}_2, \hat{a}_1y + \hat{a}_3)\end{aligned} \tag{6}$$

where $\hat{a}_1, \hat{a}_2$, and $\hat{a}_3$ are the current estimate of the true motion parameters a_1, a_2, and a_3. If $\hat{a}_1, \hat{a}_2$, and $\hat{a}_3$ approach a_1, a_2, and a_3, respectively, then DFD will approach to zero. That is the DFD is seen as the result of the estimation error of the motion parameters. Using the first-order Taylor series expansion, the luminance function at point $p_o = A(x, y)$ with estimated $e = (\hat{a}_1, \hat{a}_2, \hat{a}_3)$ can be evaluated with respect to the true motion parameter a_i as

$$S_k(A(x, y)) = S_k(\hat{A}(x, y)) + \sum_{i=1}^{3} \left.\frac{\partial S_k}{\partial a_i}\right|_{p_o,e} (a_i - \hat{a}_i) + r(x, y) \tag{7}$$

where $r(x, y)$ denotes the sum of the high-order expansion terms in this Taylor series expansion and is called as the expansion error in the following.

Substitute (7) into (5) and (6), we have

$$\begin{aligned}\mathrm{DFD}(x, y) &= G_{a1}(x, y)(a_1 - \hat{a}_1) + G_{a2}(x, y)(a_2 - \hat{a}_2) \\ &\quad + G_{a3}(x, y)(a_3 - \hat{a}_3) + r(x, y) \\ G_{a_i}(x, y) &= \left.\frac{\partial S_k}{\partial a_i}\right|_{p_o,e}.\end{aligned} \tag{8}$$

To describe (8) directly from the image signal, the gradients with respect to (w.r.t.) the motion parameters can be replaced with the gradients w.r.t. the coordinate x and y. By defining $(x', y') = A(x, y)$, we have

$$\left.\frac{\partial S_k}{\partial a_i}\right|_{p_o,e} = \left.\frac{\partial S_k}{\partial x'}\frac{\partial x'}{\partial a_i}\right|_{p_o,e} + \left.\frac{\partial S_k}{\partial y'}\frac{\partial y'}{\partial a_i}\right|_{p_o,e} \tag{9}$$

$$\begin{aligned}&\frac{\partial x'}{\partial a_1} = x \quad \frac{\partial x'}{\partial a_2} = 1 \quad \frac{\partial x'}{\partial a_3} = 0 \\ &\frac{\partial y'}{\partial a_1} = y \quad \frac{\partial y'}{\partial a_2} = 0 \quad \frac{\partial y'}{\partial a_3} = 1.\end{aligned} \tag{10}$$

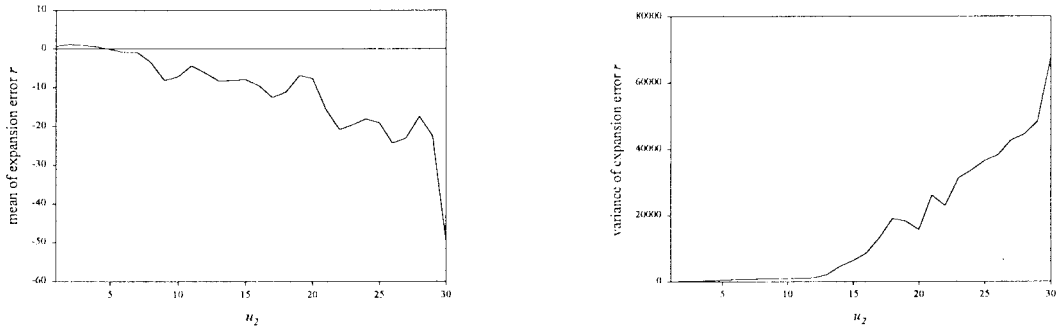
Thus

$$\begin{aligned}\mathrm{DFD}(x, y) &= S_k(A(x, y)) - S_k\left(\hat{A}(x, y)\right) \\ &= (G_x x + G_y y)(a_1 - \hat{a}_1) + G_x(a_2 - \hat{a}_2) \\ &\quad + G_y(a_3 - \hat{a}_3) + r(x, y) \\ &= (G_x x + G_y y)u_1 + G_x u_2 + G_y u_3 + r(x, y)\end{aligned} \tag{11}$$

where $(G_x, G_y) = (\partial S_k/\partial x', \partial S_k/\partial y')$ are directional gradient functions and u_i is the estimation error defined as the difference between a_i and $\hat{a}_i$.

Equation (11) describes the DFD as a function of the parameter error u_1, u_2, and u_3. This equation is valid only when these parameter errors are within a certain range. Under this condition, the expansion error r can be treated as a random process with zero mean. To see this phenomenon, the mean and variance of this expansion error as a function of u_1 are shown in Fig. 2 for the real image "laboratory" for 900 pixels with u_2 and u_3 set to zero. From Fig. 2, it can be seen that for $|u_1| < 0.13, r$ can be treated as a random process with zero mean. Fig. 3 shows this phenomenon as a function of u_2. The range of u_2 (or u_3) can be within about 13 pixels. Out of these ranges, the expansion error r will not be a random process and this signal representation model will not be feasible.

The expansion error in this signal model is also affected by the accuracy of the gradient. This effect is especially large for pixels with large coordinates. The product term $(G_x x + G_y y)u_1 + G_x u_2 + G_y u_3$ can be treated as the first-order estimate of the DFD value. Therefore, a small estimation error in gradient will generate large estimation error in DFD. When compared with the true DFD, the result is a large expansion error. So the expansion error can be seen as the error in the



(a) (b)

Fig. 3. The relationship of the mean and variance of the expansion error r versus the parameter estimation error u_2 for image "Laboratory." (a) The distribution of mean. (b) The distribution of variance (where u_1 and u_3 are set to be zero).

estimation of the DFD with gradient and parameter estimation error u_1, u_2, and u_3.

Actually, from (11), it is found that the three-parameter signal representation model is similar to the two-parameter model. As derived in [5], the two-parameter signal representation model is

$$\mathrm{DFD}(x,y) = G_x u_x + G_y u_y + r'(x,y) \tag{12}$$

with u_x and u_y as the x and y directional estimation error. Since $u_x = u_1 x + u_2$ and $u_y = u_1 y + u_3$, it can be seen that (11) and (12) are similar. The difference between (11) and (12) is the description of the motion phenomenon. The underlying search procedure for point correspondence is the same. Both equations indicate that DFD due to the discrepancy in point correspondence is the product of gradient and parameter estimation errors. Therefore, the characteristics of the expansion error $r(x,y)$ and $r'(x,y)$ are the same. Based on this fact, those features and assumptions proposed in [5] and [6] for the two-parameter case are still valid for the three-parameter case.

For N points of signals, (11) can be written in a matrix form as

$$\boldsymbol{D} = \boldsymbol{G}\left(\boldsymbol{A} - \hat{\boldsymbol{A}}^{(p)}\right) + \boldsymbol{R} = \boldsymbol{G}\boldsymbol{u} + \boldsymbol{R} \tag{13}$$

with

$$\boldsymbol{D} = \begin{bmatrix} \mathrm{DFD}\left(x_1^{(p)}, y_1^{(p)}\right) \\ \mathrm{DFD}\left(x_2^{(p)}, y_2^{(p)}\right) \\ \vdots \\ \mathrm{DFD}\left(x_N^{(p)}, y_N^{(p)}\right) \end{bmatrix} \quad \boldsymbol{R} = \begin{bmatrix} r\left(x_1^{(p)}, y_1^{(p)}\right) \\ r\left(x_2^{(p)}, y_2^{(p)}\right) \\ \vdots \\ r\left(x_N^{(p)}, y_N^{(p)}\right) \end{bmatrix}$$

and

$$\boldsymbol{G} = \begin{bmatrix} G_{x_1^{(p)}} x_1 + G_{y_1^{(p)}} y_1 & G_{x_1^{(p)}} & G_{y_1^{(p)}} \\ G_{x_2^{(p)}} x_2 + G_{y_2^{(p)}} y_2 & G_{x_2^{(p)}} & G_{y_2^{(p)}} \\ \vdots & \vdots & \vdots \\ G_{x_N^{(p)}} x_N + G_{y_N^{(p)}} y_N & G_{x_N^{(p)}} & G_{y_N^{(p)}} \end{bmatrix}$$

where

- $\boldsymbol{A} = [a_1, a_2, a_3]^T$ is the true motion parameter vector;
- $\hat{\boldsymbol{A}}(p) = [\hat{a}_1^{(p)}, \hat{a}_2^{(p)}, \hat{a}_3^{(p)}]^T$ is the estimated motion parameter vector at iteration p;
- $\boldsymbol{u} = [u_1, u_2, u_3]^T$ is the estimation error vector.

From (13), it can be seen that the estimation error $\boldsymbol{u}$ between the true value and the estimated one is embedded in the displaced frame difference and is not explicitly calculatable due to the error process $\boldsymbol{R}$. In [1], a direct pseudo-inverse is used to calculate u by neglecting the high-order expansion error $\boldsymbol{R}$. In this paper, we consider a more rigorous approach by modeling these expansion errors as random variables.

III. The Wiener-Based Estimation Method

In the previous section, a signal model has been described. If the expansion error is neglected, the estimation process becomes a simple pseudo-inverse computation as shown in [1]. To obtain better estimation, however, this random error should be considered. To recover signal from random noise, the Wiener filter has been shown to be very effective. From the signal model, the desired signal is $\boldsymbol{G}\boldsymbol{u}$ and the observed signal is $\boldsymbol{D}$. The purpose is to recover $\boldsymbol{G}\boldsymbol{u}$ from $\boldsymbol{D}$, that is, to calculate the estimated DFD from the true DFD. Therefore, in the following, we first derive the Wiener filtering process to show that $\boldsymbol{G}\boldsymbol{u}$ can indeed be recovered from $\boldsymbol{D}$. To estimate $\boldsymbol{u}$ from $\boldsymbol{G}\boldsymbol{u}$, the knowledge of $\boldsymbol{G}$ is essential. If $\boldsymbol{G}$ is known, then, the two processes can be combined into one. For most applications, however, $\boldsymbol{G}$ is usually unknown and an additional estimation procedure is needed.

A. The Wiener-Based Algorithm

Based on the representation model derived in Section II-B, (13) may be rewritten as

$$\boldsymbol{D} = \boldsymbol{Q}\boldsymbol{e} + \boldsymbol{R} \tag{14}$$

where $\boldsymbol{e} = \boldsymbol{G}\boldsymbol{u}$ is an $N \times 1$ matrix and $\boldsymbol{Q}$ is an $N \times N$ identity matrix. The Wiener-based algorithm is to find a linear estimator to operate on the displaced frame difference $\boldsymbol{D}$ to estimate the term $\boldsymbol{e}$ with the criterion that the mean square value $E\{\|\boldsymbol{e} - \boldsymbol{e}^{(p)}\|^2\}$ is minimized. If $\boldsymbol{G}$ is known, minimization of $E\{\|\boldsymbol{e} - \boldsymbol{e}^{(p)}\|^2\}$ is essentially equivalent to minimization of $E\{\|\boldsymbol{u} - \boldsymbol{u}^{(p)}\|^2\}$. Let the linear estimator be denoted as $\boldsymbol{L}$, according to the Wiener-based process [5], [6], the linear estimator can be described as

$$\boldsymbol{L} = \left(\boldsymbol{P}_{\boldsymbol{e}}\boldsymbol{Q}^T + \boldsymbol{P}_{\boldsymbol{eR}}\right)\left(\boldsymbol{Q}\boldsymbol{P}_{\boldsymbol{e}}\boldsymbol{Q}^T + \boldsymbol{P}_{\boldsymbol{R}} + \boldsymbol{Q}\boldsymbol{P}_{\boldsymbol{eR}} + \boldsymbol{P}_{\boldsymbol{eR}}^T\boldsymbol{Q}^T\right)^{-1} \tag{15}$$

where $\boldsymbol{P_e} = E\{\boldsymbol{ee}^T\} = \boldsymbol{GP_uG}^T, \boldsymbol{P_R} = E\{\boldsymbol{RR}^T\}$ and $\boldsymbol{P_{eR}} = E\{\boldsymbol{eR}^T\} = \boldsymbol{G}E\{\boldsymbol{uR}^T\}$. Because $\boldsymbol{Q}$ is an identity matrix, (15) can be written as

$$\boldsymbol{L} = (\boldsymbol{P_e} + \boldsymbol{P_{eR}})\left(\boldsymbol{P_e} + \boldsymbol{P_R} + \boldsymbol{P_{eR}} + \boldsymbol{P}_{\boldsymbol{eR}}^T\right)^{-1}. \tag{16}$$

Equation (16) is the Wiener-based estimation algorithm for the three-parameter motion model. In this algorithm, $\boldsymbol{G}$ is assumed to be a known constant. From (16), it can be seen that the functionality of the Wiener filter is to use the knowledge of $\boldsymbol{P_e}, \boldsymbol{P_R}$, and $\boldsymbol{P_{eR}}$ to estimate $\boldsymbol{Gu}$ form $\boldsymbol{DFD}$. Its performance is entirely dominated by the knowledge of these three matrices. Therefore, to discuss the performance of the Wiener filter, we need to first discuss the estimation of these three matrices.

B. The Estimation of $\boldsymbol{P_{eR}}$

As discussed in Section II-B, the characteristics of the expansion errors in the signal representation models for the two-parameter and the three-parameter cases are similar. In [6], for the two-parameter model, the assumption that u is uncorrelated with r has been made based on the observation that the expansion error is a random signal with zero mean. For the three-parameter model, as derived in Section II-B, the expansion error is still a random signal with zero mean when the estimation error is small. Therefore, the assumption that the random signal r is uncorrelated with the signal u is still made in this three-parameter model. With $E\{\boldsymbol{uR}^T\} = E\{\boldsymbol{Ru}\}^T = 0$, (16) will be

$$\boldsymbol{L} = \boldsymbol{P_e}(\boldsymbol{P_e} + \boldsymbol{P_R})^{-1}. \tag{17}$$

The matrix $\boldsymbol{P_R}$ represents the statistics of the high-order expansion terms. If the estimation process can gradually converge, the high-order expansion error will gradually decrease and the matrix $\boldsymbol{P_R}$ will approach to a zero matrix. The operation matrix $\boldsymbol{P_e}(\boldsymbol{P_e} + \boldsymbol{P_R})^{-1}$ will become an identity matrix and the process can be simplified to be a pseudo-inverse method. Therefore, the effectiveness of the Wiener filter is in the initial phase of the estimation process where the estimation error is still not very small.

After the filtering process, the estimated displaced frame difference $\boldsymbol{Gu}$ is then used to calculate the desired u with the knowledge of $\boldsymbol{G}$. That is

$$\boldsymbol{e} = \boldsymbol{Gu} = \boldsymbol{LD} = \boldsymbol{P_e}(\boldsymbol{P_e} + \boldsymbol{P_R})^{-1}\boldsymbol{D} \tag{18}$$

and

$$\begin{aligned}\boldsymbol{u} &= \left(\boldsymbol{G}^T\boldsymbol{G}\right)^{-1}\boldsymbol{G}^T\boldsymbol{P_e}(\boldsymbol{P_e} + \boldsymbol{P_R})^{-1}\boldsymbol{D} \\ &= \left(\boldsymbol{G}^T\boldsymbol{G}\right)^{-1}\boldsymbol{G}^T\boldsymbol{GP_uG}^T\left(\boldsymbol{GP_uG}^T + \boldsymbol{P_R}\right)^{-1}\boldsymbol{D} \\ &= \boldsymbol{P_uG}^T\left(\boldsymbol{GP_uG}^T + \boldsymbol{P_R}\right)^{-1}\boldsymbol{D}.\end{aligned} \tag{19}$$

The Wiener-based algorithm will become

$$\hat{\boldsymbol{A}}^{(p+1)} - \hat{\boldsymbol{A}}^{(p)} = \boldsymbol{u}^{(p)} = \left(\boldsymbol{G}^T\boldsymbol{P}_{\boldsymbol{R}}^{-1}\boldsymbol{G} + \boldsymbol{P}_{\boldsymbol{u}}^{-1}\right)^{-1}\boldsymbol{G}^T\boldsymbol{P}_{\boldsymbol{R}}^{-1}\boldsymbol{D}. \tag{20}$$

C. The Estimation of $\boldsymbol{P_u} = E\{\boldsymbol{uu}^T\}$

Actually, it is the correlation matrix $\boldsymbol{P_e}$ that is needed. By assuming that the gradient $\boldsymbol{G}$ is known, we then estimate the correlation matrix $\boldsymbol{P_u}$. From this, it can be seen that the Wiener filter itself is a function of the gradient. To evaluate the performance of the Wiener filter, both the knowledge of the gradient $\boldsymbol{G}$ and the correlation matrix $\boldsymbol{P_u}$ are important. For the two-parameter motion model in [5], the estimation errors u_x and u_y are assumed to be zero mean processes and are uncorrelated to each other. The corresponding correlation matrix $\boldsymbol{P_u}$ is assumed to be a diagonal identity matrix scaled by a constant variance σ_u^2. In [6], this assumption is also made but the variance σ_u^2 is made adaptive in each iteration. But in the three-parameter motion model, there exists a strong relationship among the parameters a_1, a_2, and a_3. The correlations among u_1, u_2, and u_3 are usually unknown and are dependent on the motion phenomenon. Thus, in this paper, the matrix $\boldsymbol{P_u}$ is estimated with

$$\boldsymbol{P}_{\boldsymbol{u}}^{(p+1)} = \frac{p}{p+1}\boldsymbol{P}_{\boldsymbol{u}}^{(p)} + \frac{1}{p+1}\boldsymbol{u}^{(p)^T}\boldsymbol{u}^{(p)} \tag{21}$$

where $\boldsymbol{u}^{(p)}$ is the estimated parameter error in the pth iteration and

$$\boldsymbol{P}_{\boldsymbol{u}}^{(1)} = \begin{bmatrix} 0.01 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}.$$

D. The Estimation of $\boldsymbol{P_R} = E\{\boldsymbol{RR}^T\}$

In [5], the high-order expansion error r in (12) for each image point is assumed to be an independent, identical, and zero mean process. In [6], a better assumption is made. The variance of the expansion error is assumed to be a function of the gradient and the estimation error u. From the second-order Taylor series expansion, the variance σ_r^2 of the expansion error for each image point is estimated as $\sigma_{ri}^2 = (G_{xyi}^2 + 1/2G_{xxi}G_{yyi})\sigma_u^4$. This error model can reasonably describe the variance of the expansion error when the parameter estimation error is small. When the parameter estimation error is large, the error term will be significant and the spatial correlation among a block of pixels do exist. In this case, the estimation of the matrix $\boldsymbol{P}_{\boldsymbol{R}}^{(p)}$ is of importance. Our approach is to use $\boldsymbol{u}^{(p)}$ as the initial estimate of $\boldsymbol{u}^{(p+1)}$. With this, the estimate of the current expansion error will be

$$\boldsymbol{R}^{(p+1)} = \boldsymbol{D}^{(p+1)} - \boldsymbol{G}^{(p+1)}\boldsymbol{u}^{(p)} \tag{22}$$

where $\boldsymbol{D}^{(p+1)}$ and $\boldsymbol{G}^{(p+1)}$ are from the new corresponding points obtained by $\boldsymbol{A}^{(p+1)} = \boldsymbol{A}^{(p)} + \boldsymbol{u}^{(p)}$. With this $\boldsymbol{R}^{(p+1)}, \boldsymbol{P}_{\boldsymbol{R}}^{(p+1)}$ is estimated as $\boldsymbol{R}^{(p+1)}\boldsymbol{R}^{(p+1)T}$. The new estimated $\boldsymbol{P}_{\boldsymbol{R}}^{(p+1)}$ is then used in the Wiener filter to estimate the new $\boldsymbol{u}^{(p+1)}$. The initial $\boldsymbol{P}_{\boldsymbol{R}}^{(1)}$ is assumed to be $\alpha\boldsymbol{I}_N$, with α proportional to the initial frame difference.

E. The Estimation Procedure of the Wiener-Based Algorithm

Based on the derivations in the above sections, the estimation procedure of the Wiener-based algorithm can be organized as in the following.

1) Set the initial values of $\boldsymbol{P_u}, \boldsymbol{P_R}$ and $\boldsymbol{A} = [1, 0, 0]^T$.
2) Identify the corresponding points with $(x_i', y_i') = (a_1 x_i + a_2, a_1 y_i + a_3)$.
3) Interpolate the pixel values of the new corresponding points with bilinear interpolation and estimate the gradients G_x and G_y.
4) Calculate the corresponding DFD. If DFD is less than a threshold, the process terminates.
5) Test the feasibility of the signal representation model for each new corresponding point.
   if $(G_x x_i + G_y y_i)^2 >$ threshold, then $c = 0$
   else $c = 1$
   and then let $G_x \Leftarrow cG_x$ and $G_y \Leftarrow cG_y$
6) Calculate the correlation matrices $\boldsymbol{P}_{\boldsymbol{u}}^{(p)}$ and $\boldsymbol{P}_{\boldsymbol{R}}^{(p)}$ according to (22) and (23).
7) Calculate the estimation error and the new motion parameter according to (21).
   go to step 2).

There are two purposes for the test measure in step 5). The first one is to test the feasibility of the signal representation model for a particular image point. This is because the signal representation model is good for small perturbation. According to our simulation, in area with large gradient the expansion error is usually large. Also, because of the multiplication effect of the coordinate as mentioned in Section II, the product term $G_x x_i + G_y y_i$ is considered as the test value. If this product term is large, the expansion error will not be a random signal with zero mean distribution. Usually, large gradient implies large variation in the image area. Therefore, the estimation of gradient can be poor. In this case, the accuracy of the parameter estimation may be affected. Although, with Wiener filter, the product term $\boldsymbol{Gu}$ can be recovered. To correctly derive $\boldsymbol{u}$, a good estimate of $\boldsymbol{G}$ is essential. For these two reasons, a test measure as shown in step 5) is proposed to exclude points that may affect the estimation accuracy of the motion parameters. Simulations have indicated that if these points are not included, the results will be more reliable. The simulation results are shown in the following section. Based on these seven steps, the Wiener-based algorithm is repeatedly applied until the DFD is less than a threshold or a predefined iteration number is reached.

IV. The Performance Evaluation

In Section III, the Wiener-based algorithm has been derived for the estimation of the three motion parameters. Due to the characteristics of the motion model, the coordinates of the new corresponding points are usually not integer and are not available. For this, the bilinear interpolation is used to calculate the luminance of these new corresponding points.

A. Performance Evaluation of the Wiener-Based Algorithm

The Wiener-based algorithm is derived from a signal representation model based on the first-order Taylor series expansion. The first-order Taylor series expansion is feasible only when the change of intensity is smooth. With Wiener algorithm to filter out the expansion error, the result is the product term $\boldsymbol{Gu}$. From the gradient $\boldsymbol{G}, \boldsymbol{u}$ is estimated to find the new corresponding points for subsequent iteration. If the knowledge of $\boldsymbol{G}$ is poor, $\boldsymbol{u}$ will not be correct. In this situation, the estimation will take longer time or even diverge. To show that the Wiener filter can indeed recover $\boldsymbol{Gu}$ from the noise corrupted DFD as well as the effect of the gradient $\boldsymbol{G}$ on the accuracy of $\boldsymbol{u}$, two synthetic images consisting of Gaussian distributions are used for demonstration. The first image contains three Gaussian distributions

$$S_{11}(x,y) = 255 \exp\left(\frac{-((x-50)^2 + (y+50)^2)}{200}\right)$$
$$S_{12}(x,y) = 255 \exp\left(\frac{-((x+50)^2 + (y-50)^2)}{200}\right)$$
$$S_{13}(x,y) = 255 \exp\left(\frac{-((x-50)^2 + (y-50)^2)}{200}\right).$$

The second image contains

$$S_{21}(x.y) = 255 \exp\left(\frac{-((x-51)^2 + (y+49)^2)}{200 \times (1.08)^2}\right)$$
$$S_{22}(x,y) = 255 \exp\left(\frac{-((x+47)^2 + (y-51)^2)}{200 \times (1.20)^2}\right)$$
$$S_{23}(x,y) = 255 \exp\left(\frac{-((x-55)^2 + (y-54)^2)}{200 \times (1.50)^2}\right).$$

From these two images, three motion parameters $\boldsymbol{A_1} = (1.08, 1, 1)^T, \boldsymbol{A_2} = (1.20, 3, 1)^T$ and $\boldsymbol{A_3} = (1.50, 5, 4)^T$ for the six Gaussian distributions are observed. In this simulation, the gradients are estimated with three different methods.

1) The optimum estimation:

$$G_x(x_i, y_i) = 255 \exp\left(\frac{-((x_i - j)^2 + (y_i - k)^2)}{200 \times r^2}\right) \times \frac{-2(x_i - j)}{200 \times r^2}$$
$$G_y(x_i, y_i) = 255 \exp\left(\frac{-((x_i - j)^2 + (y_i - k)^2)}{200 \times r^2}\right) \times \frac{-2(y_i - k)}{200 \times r^2}$$

   where (j, k) is the central point of the Gaussian distribution and r is the zooming factor. As indicated in step 2), the new corresponding points are usually not integer for most digital images. For gradient, the nearest integer points are used for estimation.
2) The six-point estimation:

$$\begin{aligned} G_x(x_i, y_i) = &\{S_k([x_i] + 1, [y_i] - 1) \\ &- S_k([x_i] - 1, [y_i] - 1)\}/4 \\ &+ \{S_k([x_i] + 1, [y_i]) - S_k([x_i] - 1, [y_i])\}/2 \\ &+ \{S_k([x_i] + 1, [y_i] + 1) \\ &- S_k([x_i] - 1, [y_i] + 1)\}/4 \end{aligned}$$
$$\begin{aligned} G_y(x_i, y_i) = &\{S_k([x_i] - 1, [y_i] + 1) \\ &- S_k([x_i] - 1, [y_i] - 1)\}/4 \\ &+ \{S_k([x_i], [y_i] + 1) - S_k([x_i], [y_i] - 1)\}/2 \\ &+ \{S_k([x_i] + 1, [y_i] + 1) \\ &- S_k([x_i] + 1, [y_i] - 1)\}/4 \end{aligned}$$

where $[x_i]$ and $[y_i]$ are the nearest integer points of x_i and y_i.

3) The two-point estimation:

$$G_x(x_i, y_i) = S_k([x_i] + 1, [y_i]) - S_k([x_i] - 1, [y_i])$$
$$G_y(x_i, y_i) = S_k([x_i], [y_i] + 1) - S_k([x_i], [y_i] - 1).$$

The simulation results for these six Gaussian distributions are shown in Table I and Fig. 4. Table I shows the estimated motion parameters with the three different gradient estimations. Fig. 4 shows the estimation of $\boldsymbol{A}$ corresponding to the movement of S_{11} to S_{21}. All cases show that the Wiener filter can indeed recover $\boldsymbol{Gu}$ from the signal representation model. As can be seen from Fig. 4, however, the effect of the gradient is very significant. This indicates the importance of the gradient estimation. For practical applications, gradient estimation is a major problem that deserves special attention. Whether or not the process will converge is largely dependent on the gradient. In the following, we show the applications for the real images. Two real images named as "Laboratory" and "Mickey" as shown in Fig. 1 are used for simulation. The zooming factor for image "Laboratory" is 1.065 or 0.94 depending on the choice of the reference image. The zooming factor for image "Mickey" is 1.03 or 0.97. Since the gradient is unknown, both the two-point estimation and the six-point estimation are used. The simulation with "Mickey" at coordinate (−4, 16) is shown in Fig. 5. The desired motion parameter is (1.03, 2, 1). The result with two-point gradient estimation is incorrect. With better six-point gradient estimation, however, the parameter can be correctly derived. In both cases, the test in step 5 is applied with c set at zero and the threshold set at 4 million. This means that these points not satisfying the test are not used in the algorithm. This is just to exclude these points that may have bad gradient estimation.

To further investigate the effect of the test measure, another simulation is shown in Fig. 6. The image used for simulation is the "Laboratory" and the desired motion parameter is (0.94, 2, 0). The coordinate of the test block is (0, −4). In Fig. 6, the six-point method is used for gradient estimation. If all points are used (c set at one), however, the result is incorrect. If these points not satisfying the measure are excluded, the algorithm converges to the correct value. Two other simulation results are shown in Tables II and III. Because the zooming parameter will dominate the estimation process, the accuracy of a1 is better than that of a_2 or a_3.

The test in step 5) is applied in each iteration to examine the appropriateness of the new corresponding points for use in deriving the parameter estimation error. By excluding these points not satisfying the test in step 5), the accuracy of the Wiener algorithm can be increased. Generally, the block of an image can be attributed to three different types.

Type 1: Area without zooming or translational phenomenon.

Type 2: Smooth changing area with zooming or translational phenomenon.

Type 3: Abrupt changing area with zooming or translational phenomenon.

TABLE I
THE ESTIMATED MOTION PARAMETERS OF THE THREE SETS OF GAUSSIAN DISTRIBUTIONS WITH DIFFERENT GRADIENT ESTIMATIONS. THE DESIRED MOTION PARAMETER IS (a) (1.08, 1, 1) (b) (1.20, 3, 1), AND (c) (1.50, 5, 4)

	estimated a_1	estimated a_2	estimated a_3
optimum	1.081	0.964	0.970
6-point	1.065	0.756	0.755
2-point	1.064	0.742	0.742

(a)

	estimated a_1	estimated a_2	estimated a_3
optimum	1.201	3.223	1.109
6-point	1.174	2.582	1.411
2-point	1.140	2.338	1.096

(b)

	estimated a_1	estimated a_2	estimated a_3
optimum	1.492	4.919	4.135
6-point	1.483	4.925	3.512
2-point	1.335	2.925	2.149

(c)

For block in type 1, the estimated motion parameters will be close to (1, 0, 0). For block in type 2, the estimated motion parameters can be close to the desired ones with our proposed Wiener-based method. Due to the truncation or interpolation errors, the estimated motion parameters may be slightly different from the desired ones. For block in type 3, the estimated parameters generate a corresponding pixel at the local minimum DFD, which may be away from the point with global minimum DFD. This is the problem encountered by most adaptive searching algorithms. To solve this problem, the concept of simulated annealing algorithm with gradually decreasing searching tap length may be useful. But this method is different from our proposed Wiener-based method and remains an interesting topic for further research.

B. The Combination of Gradient Search and Full Search with $a_1 = 1$

From the above discussions, it can be seen that for smooth regions that fit the signal model, the Wiener-based method can estimates the motion parameters with roughly three iterations. But, for the areas with edges and object boundaries, the gradient-based search will tend to diverge and result in higher bit rates than with the two-parameter model. For these regions, full search is used to find the motion parameters. Since a single zoom parameter is still not enough to describe the motion phenomenon, we choose to degenerate the three-parameter model to the two-parameter model and use a full search [12]–[14] with a_1 set at one to derive the other two parameters. This degenerate case will correspond to the conventional two-parameter block matching.

To see the performance of the combined method, the image sequence "Football" is used for simulation. The number of iteration of the gradient search is set at three and the search window of the full search is 16 × 16. The simulation results are shown in Tables IV–VII. Tables IV and VI show the mean

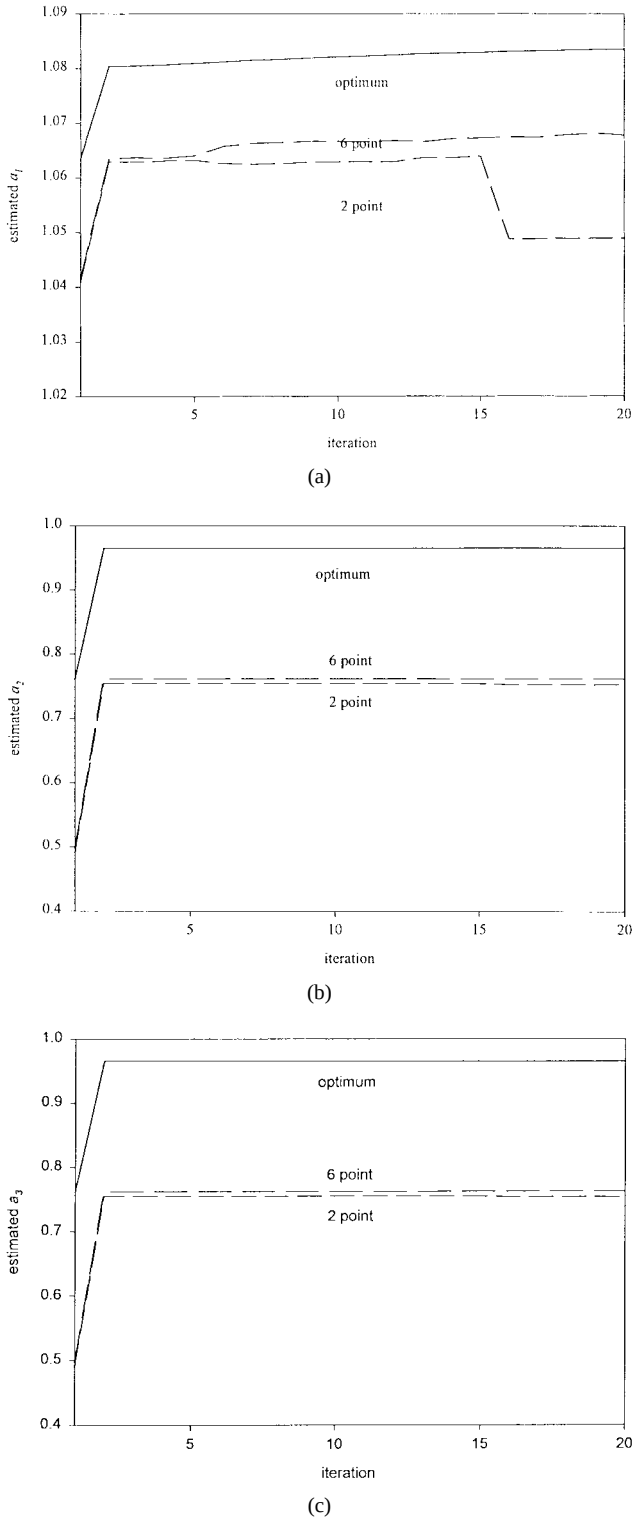


Fig. 4. The estimated motion parameter (a) a_1, (b) a_2, and (c) a_3 with the three different gradient estimation methods for S_{11} and S_{21} [the desired motion parameter is (1.08, 1, 1)].

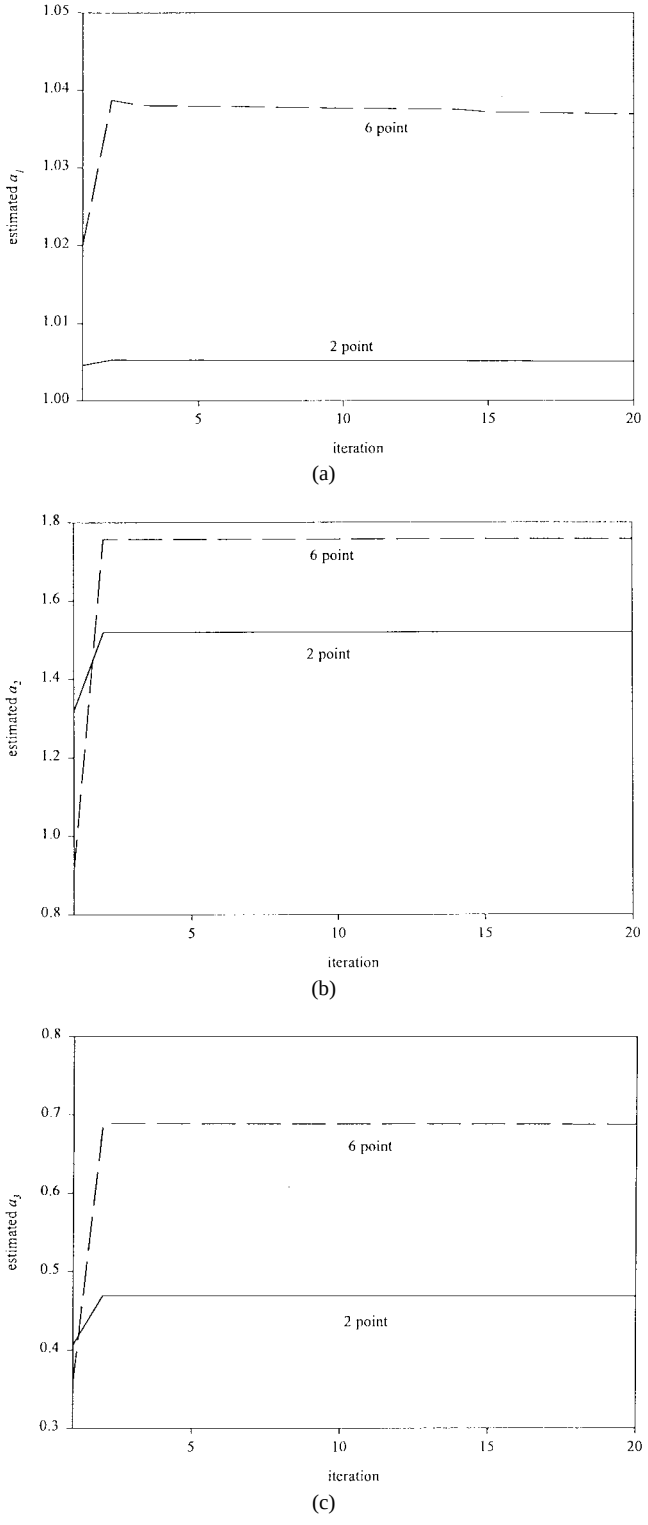
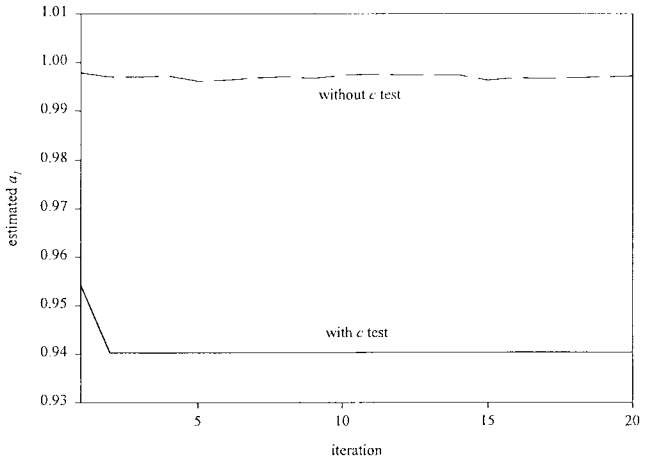


Fig. 5. The estimated motion parameter (a) a_1, (b) a_2, and (c) a_3 with the two different gradient estimation methods at block $(-4, 16)$ with the image "Mickey" [the desired motion parameter is (1.03, 2, 1)].

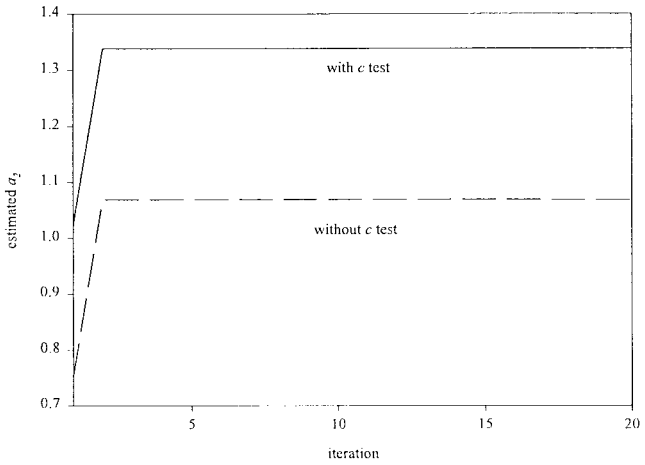
absolute error of DFD for these 11×3 simulated blocks with both the Wiener-based method and the block matching method. The corresponding estimated zooming parameter with the Wiener-based method for each block is presented in Tables V and VII. From these results, we can classify the image blocks into three categories.

A: Smooth area without zooming phenomenon.
B: Smooth area with zooming phenomenon.
C: Nonsmooth area or boundary area.

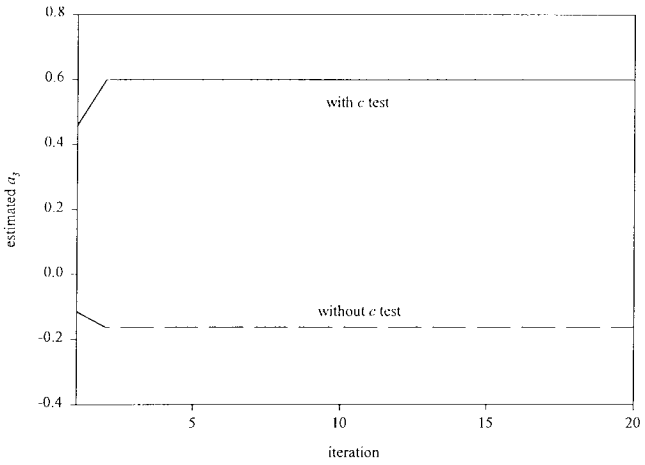
In case A, the two-parameter block matching search is sufficient. The full search block matching method can derive the best result. In Tables V and VII the blocks with zooming



(a)



(b)



(c)

Fig. 6. The effect of test step 5 (a) a_1, (b) a_2, and (c) a_3. The test is at block (0, −4) with the image "Laboratory" [the desired motion parameter is (0.94, 2, 0)].

parameter near one belong to this case. In case B, the use of zoom factor is shown to be useful. The performance of the Wiener-based gradient search is very stable. In case C, the gradient-based method is not suitable. It usually diverges or stays at a local minimum. In this case, point correspondence is hard to achieve.

TABLE II
THE EFFECT OF TEST IN STEP 5) FOR THE BLOCK IN (0, −4) FOR THE IMAGE "LABORATORY." THE DESIRED MOTION PARAMETER IS (0.94, 2, 0)

value of c	estimated a_1	estimated a_2	estimated a_3
1.0	1.00	1.07	-0.16
0.0	0.94	1.34	0.60

TABLE III
THE EFFECT OF TEST IN STEP 5) FOR THE BLOCK IN (68, −23) FOR THE IMAGE "LABORATORY." THE DESIRED MOTION PARAMETER IS (1.065, 1, 2)

value of c	estimated a_1	estimated a_2	estimated a_3
1.0	1.062	0.45	0.73
0.0	1.065	0.54	0.96

During the coding, one byte is used to code the extra motion parameter. Thus, the coding gain is obtained with significantly reduced DFD such that the reduction of bit rate of the subsequent run-length code can compensate for this. For this, with the combined method, only when the DFD with the full search is larger than 1.5 times of the DFD with the gradient-based method, will the zoom factor be considered. To see the performance of the combined method, comparison with the method using block matching only is made. In these simulations the image sequences "Football" and "Tennis" are used. With one extra byte for zooming factor, the value is from 0–2 with step size 1/128. The quantization step size of the DCT coefficients is chosen as 24 for luminance component and 16 for chrominance component. The average bit rate, mean MAD, and percentage of blocks coded with three parameters are listed in Table VIII. In Fig. 7, the bits/pixel for each frame of the sequence "Football" is shown. In Fig. 8, the percentage of blocks that Wiener-based method is effective in the combined method for the image sequence "Football" is shown. Similar results for the sequence "Tennis" are shown in Figs. 9 and 10. According to these results it could be found that bit-rate reduction with the combined method is quite significant. Also, it can be seen that the zoom factor can be most effectively derived in the smooth area. In the following, we discuss some methods to simplify the computation of the Wiener-based gradient search for these areas.

C. The Performance Comparison with Simplified Method

To see the effect of these correlation matrices on the performance of the Wiener-based method, in the following, three types of simplified Wiener-based methods are also studied.

1) The Type I Simplified Wiener-Based Algorithm: In this simplification, the correlation matrix $\boldsymbol{P_R}$ is assumed to be a diagonal matrix. From this simplification, the influence of the matrix $\boldsymbol{P_R}$ on the performance of the Wiener algorithm can be seen. The variance of the expansion error is assumed to be a function of the gradient and the estimation error u. From the second-order Taylor series expansion [6], the variance σ_{ri}^2 is estimated as

$$\sigma_{ri}^2 = (G_{xyi}^2 + 1/2 G_{xxi} G_{yyi})\sigma_u^4. \tag{23}$$

TABLE IV
THE MAD OF THE CORRESPONDING BLOCKS WITH WIENER-BASED/BLOCK-MATCHING METHOD. THESE 11 × 3 BLOCKS ARE WITHIN (−216, −40) IN THE x-DIRECTION AND (0, 48) IN THE y-DIRECTION. THE IMAGES ARE FRAME 1 AND FRAME 2 IN SEQUENCE "FOOTBALL"

36.10/5.08	15.97/7.56	16.12/23.97	22.92.29.90	8.01/5.03	12.55/10.94	30.77/30.03	18.86/12.38	12.90/21.33	22.79/24.16	5.20/5.39
7.25/5.82	9.85/12.37	7.89/17.98	18.85/27.07	13.13/11.98	4.51/4.91	4.24/4.27	11.62/12.93	18.12/18.91	15.19/17.40	8.58/7.08
7.15/7.08	8.45/8.08	4.80/10.58	5.53/8.79	13.94/8.54	9.46/5.51	8.84/7.01	10.97/17.37	13.52/13.65	8.26/8.14	9.12/7.23

TABLE V
THE CORRESPONDING ESTIMATED ZOOMING PARAMETER OF THE ABOVE 11×3 BLOCKS IN TABLE IV WITH WIENER-BASED METHOD

1.007	1.016	1.089	1.121	0.961	1.041	1.130	1.049	1.170	1.189	1.245
1.039	1.094	1.100	1.122	1.151	1.081	0.983	1.098	1.179	1.108	1.012
1.046	1.099	1.116	1.127	1.151	1.067	1.026	1.127	1.174	1.149	1.025

TABLE VI
THE MAD OF THE CORRESPONDING BLOCKS WITH WIENER-BASED/BLOCK-MATCHING METHOD. THESE 11× 3 BLOCKS ARE WITHIN (−216, −40)IN THE x-DIRECTION AND (0, 48) IN THE y-DIRECTION. THE IMAGES ARE FRAME 1 AND FRAME 5 IN SEQUENCE "FOOTBALL"

6.04/5.65	9.22/5.98	19.25/20.51	20.40/33.66	20.40/40.02	72.08/56.36	19.67/16.94	10.63/8.75	14.93/23.20	74.06/64.39	37.26/37.92
9.36/7.71	12.50/9.98	18.19/14.41	16.84/46.82	19.70/69.39	14.58/88.23	17.23/47.84	21.34/18.53	30.01/25.25	42.46/68.62	53.80/48.88
9.28/7.15	16.39/14.93	12.31/11.51	7.66/11.67	16.57/36.69	11.42/65.34	11.13/35.03	11.38/10.81	25.00/28.86	28.53/24.87	17.69/13.93

TABLE VII
THE CORRESPONDING ESTIMATED ZOOMING PARAMETER OF THE ABOVE 11 × 3 BLOCKS IN TABLE VI WITH WIENER-BASED METHOD

0.992	0.995	1.063	1.160	1.284	0.930	1.030	1.008	1.178	1.023	0.745
0.992	0.998	1.062	1.166	1.289	1.485	1.591	0.990	1.212	1.424	0.688
1.028	0.942	1.025	1.137	1.280	1.499	1.656	0.931	1.109	1.097	1.053

TABLE VIII
THE AVERAGE OF MAD, BITS/PIXEL, AND PERCENTAGE OF BLOCK IN WHICH WIENER BASED METHOD IS USED IN COMBINATIVE ALGORITHM FOR SIMULATED IMAGE SEQUENCE (a) "FOOTBALL" AND (b) "TENNIS"

Algorithm	Average MAD	Average bits/pixel	Percent of block with wiener
Block-matching method	14.55	1.4549	
Combinative method	13.21	1.3108	18

(a)

Algorithm	Average MAD	Average bits/pixel	Percent of block with wiener
Block-matching method	8.83	1.4709	
Combinative method	8.54	1.4214	6

(b)

2) The Type II Simplified Wiener-Based Algorithm: In this simplification, the correlation matrix $\boldsymbol{P_R}$ is assumed to be a diagonal identity matrix $\boldsymbol{I}_N$ scaled by a constant variance σ_r^2. With this simplification $\boldsymbol{P_R} = \sigma_r^2 \boldsymbol{I}_N$, the algorithm is

$$\hat{\boldsymbol{A}}^{(p+1)} = \hat{\boldsymbol{A}}^{(p)} + \left(\boldsymbol{G}^T\boldsymbol{G} + \sigma_r^2 \boldsymbol{P_u}^{-1}\right)^{-1} \boldsymbol{G}^T \boldsymbol{D}. \quad (24)$$

3) The Type III Simplified Wiener-Based Algorithm: In this type of filter, two simplifications are made. First, the correlation matrix $\boldsymbol{P_R}$ is assumed to be a diagonal identity matrix $\boldsymbol{I}_N$ scaled by a constant variance σ_r^2. Second, the correlation matrix $\boldsymbol{P_u}$ is also assumed to be a diagonal identity matrix scaled by the constant variance σ_u^2. Both of these two assumptions are the same as in [5].

The most simple one is the direct pseudo-inverse algorithm. With direct pseudo-inverse, both the expansion error and the correlation among the parameter errors are not considered. So, totally five motion estimation methods are compared. They are:

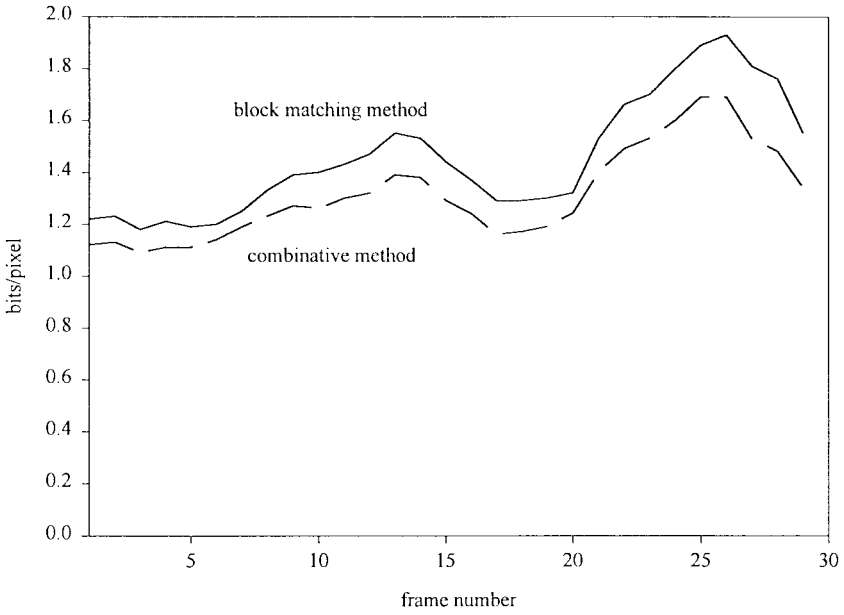


Fig. 7. The bits/pixel comparison for image sequence "Football."

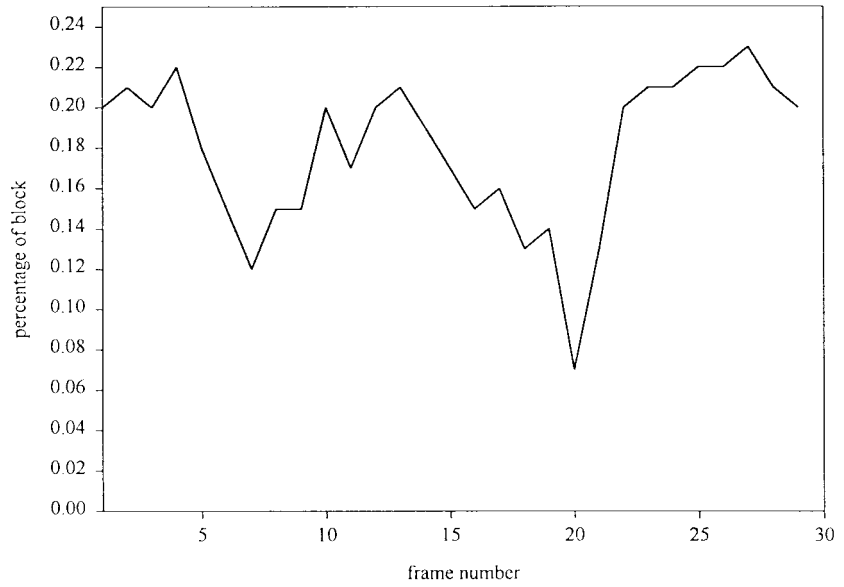


Fig. 8. Percentage of blocks that Wiener-based method is used in the combined estimation algorithm for image sequence "Football."

1) the original Wiener-based algorithm: named as method1;
2) the type I simplified Wiener-based algorithm: named as method2;
3) the type II simplified Wiener-based algorithm: named as method3;
4) the type III simplified Wiener-based algorithm: named as method4;
5) the direct pseudo-inverse algorithm as in [1]: named as method5.

The simulation results for these five different algorithms are shown in Tables IX, X and Fig. 11. In Tables IX and X, the results of the estimated motion parameters for different blocks with those five algorithms are shown. In Fig. 11, the convergence curve for the five different algorithms for the block in $(0, -4)$ is shown. The test image is "laboratory" with motion parameter (0.94, 2, 0). In all the simulations, the test in step 5) is applied and the threshold is set at 4 000 000. According to these results it could be found that to derive the correct motion parameters, the processing of the expansion error with Wiener filter is of use.

V. Conclusion

Three-parameter motion model is shown to be effective for bit rate reduction. To reduce the parameter estimation complexity, a gradient-based search algorithm is proposed. The Wiener filter is shown to be effective in suppressing the linearization error of the signal representation model on which

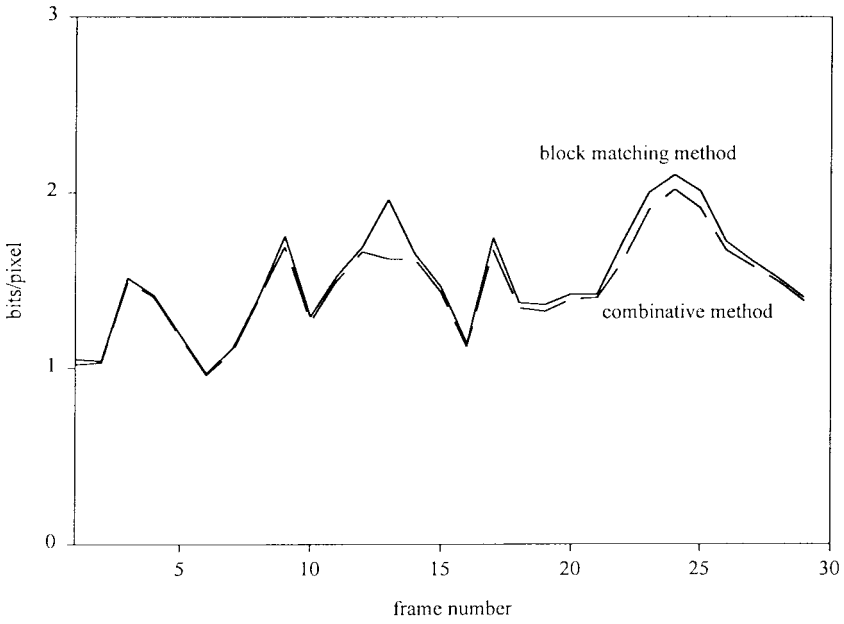


Fig. 9. The bits/pixel comparison for image sequence "Tennis."

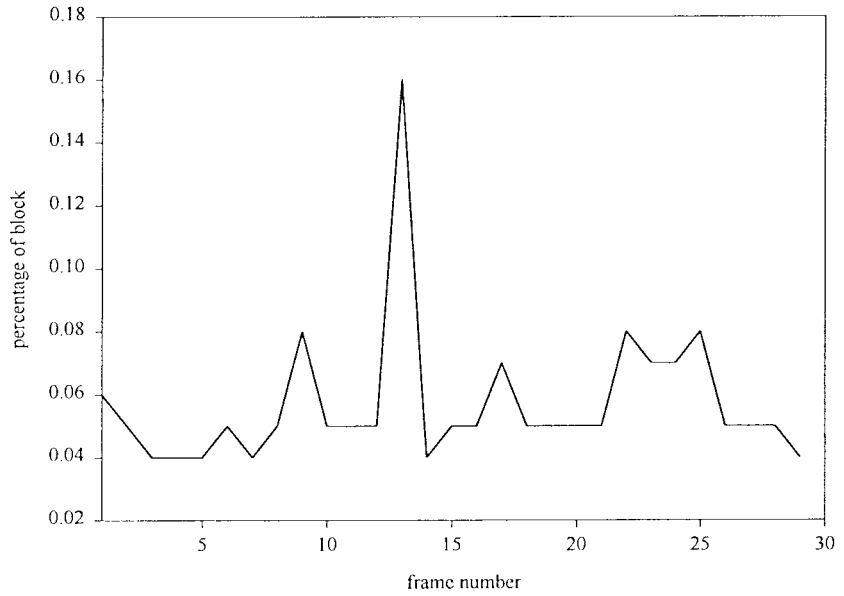


Fig. 10. Percentage of blocks that Wiener-based method is used in the combined estimation algorithm for image sequence "Tennis."

TABLE IX
THE COMPARISON OF FIVE DIFFERENT ALGORITHMS FOR THE BLOCK IN (0, −4) FOR THE IMAGE "LABORATORY." THE DESIRED MOTION PARAMETER IS (0.94, 2, 0)

algorithm	estimated a_1	estimated a_2	estimated a_3
method1	0.940	1.34	0.60
method2	0.966	1.02	-0.97
method3	0.965	0.58	-0.73
method4	1.706	1.29	5.43
method5	1.695	1.27	5.34

TABLE X
THE COMAPRISON OF 5 DIFFERENT ALGORITHMS FOR THE BLOCK IN (68, −32) FOR THE IMAGE "LABORATORY." THE DESIRED MOTION PARAMETER IS (1.065, 1, 2)

algorithm	estimated a_1	estimated a_2	estimated a_3
method1	1.065	0.54	0.96
method2	1.053	-1.22	0.86
method3	1.220	-12.45	5.99
method4	1.541	45.52	4.98
method5	6.00	-100	-28

the gradient algorithm is based. The improvement is seen to be very significant as compared with the direct pseudo-inverse. The effect of gradient accuracy is also shown to be significant. Therefore, for application in the real image, effective estimation of gradient is of importance. Similar to the most gradient-based algorithms, to overcome the situations with nonstationary data or boundary discontinuity is still a problem for further research.

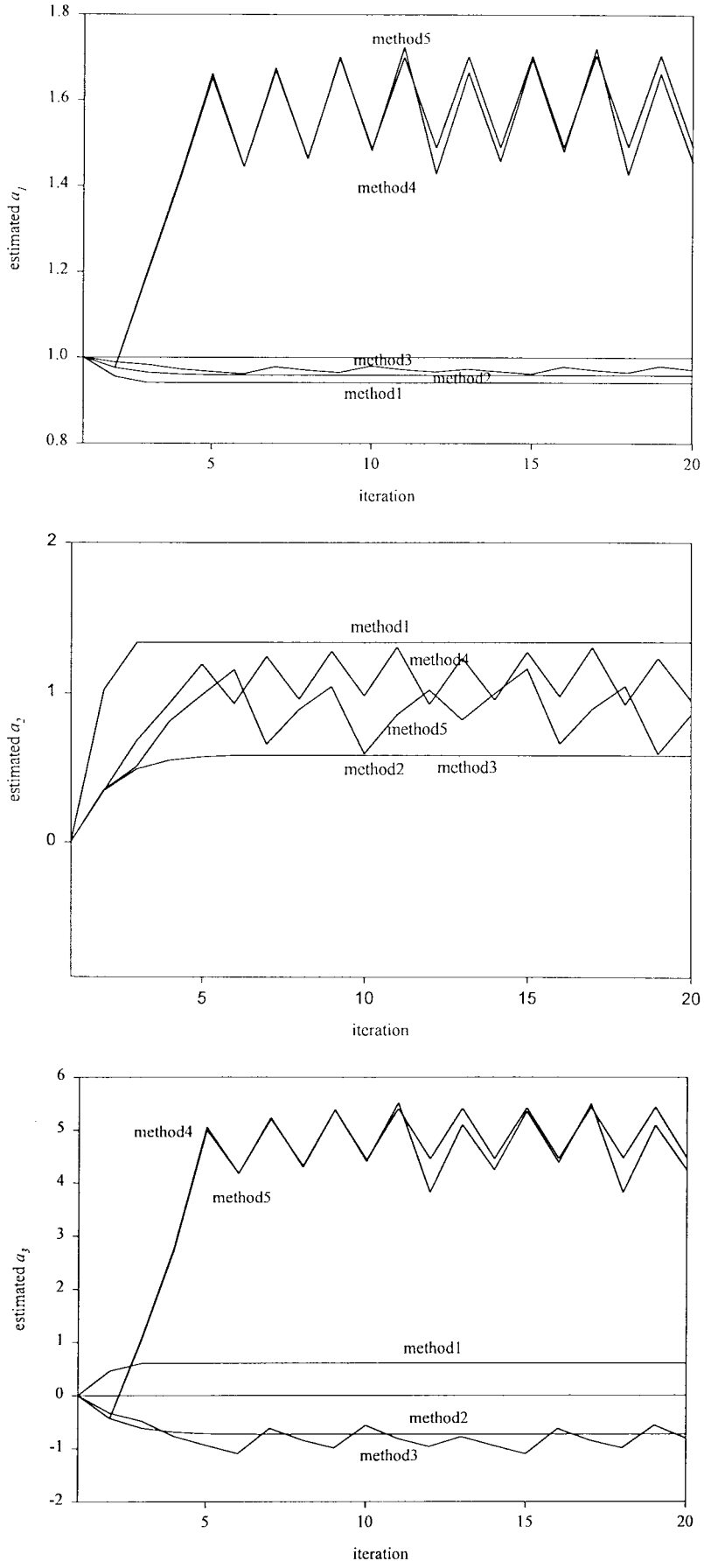


Fig. 11. The estimated motion parameter (a) a_1, (b) a_2, and (c) a_3 for five different methods for block in $(0, -4)$ with simulated image "Laboratory" [the desired motion parameter is (0.94, 2, 0)].

REFERENCES

[1] M. Hotter, "Differential estimation of the global motion parameters zoom and pan," *Signal Processing*, vol. 16, no. 3, pp. 249–265, Mar. 1989.

[2] D. Wang and L. Wang, "Global motion parameters estimation using a fast and robust algorithm," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 7, no. 5, pp. 823–826, Oct. 1997.

[3] D. R. Walker and K. R. Rao, "Improved pel-recursive motion compensation," *IEEE Trans. Commun.s*, vol. COM-32, no. 10, pp. 1128–1134, Dec. 1984.

[4] C. Cafforio and F. Rocca, "The differential method for motion estimation," in *Image Sequence Processing and Dynamic Scene Analysis* (NATO ASI Series, Vol. 12), T. S. Huang, Ed. New York: Springer-Verlag, 1983, pp. 104–124.

[5] J. Biemond, L. Looijenga, D. E. Boekee, and R. H. J. M. Plompen, "A pel-recursive Wiener-based displacement estimation algorithm," *Signal Processing*, vol. 13, pp. 399–412, Dec. 1987.

[6] L. Boroczky, J. N. Driessen, and J. Biemond, "Adaptive algorithms for pel-recursive displacement estimation," *SPIE Visual Comm. Image Proc.*, vol. 1360, pp. 1210–1221, 1990.

[7] J. Biemond, J. N. Driessen, A. M. Geurtz, and D. E. Boekee, "A pel-recursive Wiener-based algorithm for the simultaneous estimation of rotation and translation," *SPIE Visual Comm. Image Proc.*, vol. 1001, pp. 917–924, 1988.

[8] N. Baaziz and C. Labit, "Multiconstraint Wiener-based motion compensation using wavelet pyramids," *IEEE Trans. Image Processing*, vol. 3, no. 5, pp. 688–692, Sept. 1994.

[9] M. K. Ozkan, A. T. Erdem, M. I. Sezan, and A. M. Tekalp, "Efficient multiframe Wiener restoration of blurred and noisy image sequences," *IEEE Trans. Image Processing*, vol. 1, no. 4, pp. 453–476, Oct. 1992.

[10] R. C. Kim and S. U. Lee, "A VLSI architecture for pel-recursive motion estimation algorithm," *IEEE Trans. Circuits Syst.*, vol. 36, no. 10, pp. 1291–1300, Oct. 1989.

[11] E. D. Frimout, J. N. Driessen, and E. F. Deprettere, "Parallel architecture for pel-recursive motion estimation algorithm," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 2, no. 2, pp. 159–168, June 1992.

[12] A. N. Netravali and J. D. Robbins, "Motion compensated television coding: Part I," *Bell Syst. Tech. J.*, vol. BSTJ-58, no. 3, pp. 631–670, Mar. 1979.

[13] A. N. Netravali and B. G. Haskell, "Digital picture representation and compression," AT&T Bell Laboratories, New York, 1988.

[14] G. Musmann, P. Pirsch, and H.-j. Grallert, "Advances in picture coding," *Proc. IEEE*, vol. 73, no. 4, Apr. 1985.

[15] C.-T. Chen and T. R. Hsing, "Digital coding techniques for visual communications," *J. Visual Commun. Image Representation*, vol. 2, no. 1, pp. 1–16, Mar. 1991.

Pei-Chuan Liu was born in Taipei, Taiwan, R.O.C., in 1967. He received the B.S. and M.S. degrees in electrical engineering from National Chiao Tung University, Hsinchu, Taiwan, in 1990 and 1992.

He is currently a Ph.D. candidate in electrical engineering at National Chiao Tung University. His research interests include video compression, adaptive signal processing, DSP architecture design, video signal processor design, image processing, and video coding.

Wen-Thong Chang was born in Taiwan on November 15, 1956. He received the B.S. degree from National Chiao Tung University, Hsinchu, Taiwan in 1979 and the Ph.D. degree from Carnegie-Mellon University, Pittsburgh, PA, in 1985.

Since 1987 he has been with the Department of Communication Engineering, National Chiao Tung University, Hsinchu, Taiwan. His research interests include digital signal processing, digital image processing, computer vision and video compression.