ABSTRACT

MOTION COMPENSATION USING CORRELATION-FEEDBACK

by
Zebiba Shifa

Motion compensation is widely used for exploiting temporal redundancies in the coding of image sequences. Accurate estimation of motion information in image sequences is important in motion-compensated coding. Different approaches have been used to estimate motion to obtain the motion-compensated frame difference signal.

This work uses the correlation-feedback approach to estimate the velocity or the optic flow of the moving image pixel. After the motion of the pixel is estimated. the motion-compensated frame difference signal is found by subtracting the current frame from the predicted frame.

This correlation-feedback approach estimates the true motion vector of moving image accurately. Consequently, the reduced error in determining the optic flow of the moving image leads to a better motion-compensated frame difference signal. This work evaluates the performance of the correlation-feedback method by comparing it with the gradient-based approach and block method.

MOTION COMPENSATION
USING
CORRELATION-FEEDBACK

by
Zebiba Shifa

A Thesis
Submitted to the Faculty of
New Jersey Institute of Technology
in Partial Fulfillment of the Requirements for the Degree of
Master of Science in Electrical Engineering

Department of Electrical and Computer Engineering

May 1994

APPROVAL PAGE

MOTION COMPENSATION USING CORRELATION-FEEDBACK

Zebiba Shifa

Dr. Yun-Qing Shi, Thesis Advisor   Date
Associate Professor of Electrical and Computer Engineering,
NJIT

Dr. Nirwan Ansari, Committee Member   Date
Associate Professor of Electrical and Computer Engineering,
NJIT

Dr. Stanley Reisman, Committee Member   Date
Professor of Electrical and Computer Engineering,
NJIT

BIOGRAPHICAL SKETCH

Author: Zebiba Shifa

Degree: Master of Science in Electrical Engineering

Date: May 1994

Undergraduate and Graduate Education:

- Master of Science in Electrical Engineering,
  New Jersey Institute of Technology, Newark, NJ, 1994

- Bachelor of Science in Electrical Engineering,
  New Jersey Institute of Technology, Newark, NJ, 1992

I dedicate this work to my sister, Asma

All my life, every step I took
 was preceded by you.
Every new corner I turned,
 you had been there before.
Every hurdle I overcame,
 you had been there before.
But, for each of those trying times,
 when you leaped into the great unknown
 and found your ground.
You patiently waited for me,
 guiding me so that I would not get lost,
 supporting me so that I would always stand tall,
 keeping me in your ever loving sight,
  so that I would never ever fall

ACKNOWLEDGMENT

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

CHAPTER 1

INTRODUCTION

In recent years, motion estimation of moving images has received increasing attention in the areas of video conferencing and video phone, where the bit rate reduction is very important. The bit rate reduction translates into the compression, which can be evaluated by the amount of data sent over the total information. A higher compression ratio is achieved by cutting out overlapping information, termed redundancy, between one image frame and the next. Two types of redundancies, namely, temporal and spatial, have been studied extensively to achieve high compression ratios. This work addresses the field of motion compensation which studies the reduction of temporal pixel redundancies that exist in a sequence of image frames.

Motion compensation targets temporal redundancies in the frames of moving image sequences by accounting for the presence of motion. The process of determining the movement of objects within a sequence of image frames is called motion estimation.

In this work, two approaches to motion estimation will be discussed: namely, block matching and optic flow. Block matching method [2] [15] [23] estimates the displacement vector by comparing the grey values of successive frames in block by block basis. Different methods have been proposed for computation of the optic flow of an image. Among these methods we will briefly introduce the gradient-based and correlation-based approaches [10], [21].

A new approach, which is correlation-based and linked to the concept of feedback [17], will be used in motion estimation and compensation. Utilizing feedback reduces the error in determining the optic flow [17]. Therefore, the prediction error, i.e., the difference between the current frame and the predicted

1

frame, will be reduced. Consequently, the accuracy of motion compensation is improved significantly with the application of this optic-flow field estimator. Simulations in this study show that a correlation-feedback approach yields a prediction error lower than that of gradient-based and correlation-based approaches.

The scope of this thesis is to investigate techniques that estimate motion as accurately as possible for applications to motion-compensated coding. Using correlation-feedback, the true image vector can be estimated accurately.

Chapter 2 will discuss two region matching methods; namely, recursive and block matching. The recursive method implements an iterative procedure to estimate the displacement vectors. This method predicts the displacement of each pixel recursively. The block matching method estimates the displacement vector by comparing the gray values of successive frames on a block-by-block basis. The image is first segmented into blocks. Each block in the current frame is compared with all possible corresponding blocks within a search area in the previous frame. The best match is obtained by finding the minimum error, the displaced frame.

Chapter 3, discusses the estimation of optic flow. Three different approaches to optic flow are analyzed: the gradient-based, correlation-based, and spatio-temporal energy based approaches. The gradient-based approach computes the velocity from the spatio-temporal derivatives of the intensities. The correlation-based approach uses a matching technique, which compares each pixel in the first image with that in the second image. The difference forms the displacement vector for each pixel in the first image. This search involves finding the best match, which is obtained by finding the minimum distance measure.

Chapter 4 introduces past work correlation-feedback and also introduces the application of correlation-feedback to motion compensation. This technique incorporates both the correlation-based approach and the concept of feedback to compute the optic flow. After computing the optic flow using correlation-feedback, the

prediction of the current frame is made, and the prediction error, which is the difference between the predicted frame and the current frame, is computed. The prediction error together with the optic flow is sent to a receiver to reconstruct the current frame. The current frame is reconstructed at the receiver end by adding the predicted frame to the prediction error.

Chapter 5 contains the simulations using correlation-feedback and the block matching method. Lastly, Chapter 6 summarizes the obtained results and discusses the possible future work.

CHAPTER 2

MOTION ESTIMATION AND COMPENSATION

A motion picture is a sequence of still frames displayed in rapid succession. There is a great deal of temporal as well as spatial redundancy among adjacent frames. Motion compensation along the temporal dimension helps to reduce the temporal redundancy. Most of the variation in intensities from one frame to the next is due to object motion. After motion is estimated, the predicted frame is obtained. All approaches to motion compensation estimate a motion field at the at receiver end relating object locations between previous and current frames. This estimate is computed based on the pixel intensities of the current and previous frames. All motion compensation methods [4] [11] [16] operate under the assumption that this motion field estimate must be reconstructed at the decoder without the a-priori knowledge of the current frame pixel intensities. In the following will discuss motion estimation.

The process of determining the movement of objects within the sequence of image frames is known as motion estimation [13]. By estimating motion parameters, we can create a new frame between two adjacent existing frames. By predicting the intensity of the current frame from the previous frames, we can limit our coding to the difference in intensities between the actual and the predicted current frames. In this work, the motion estimation problem is only due to the translations of objects given by:

$$f(x, y, t_0) = f(x - dx, y - dy, t_{-1}), \tag{2.1}$$

where $f(x, y, t_{-1})$ is the previous frame, $f(x, y, t_0)$ is the current frame, and dx and dy are the horizontal and vertical displacement between t_{-1} and t_0, respectively.

4
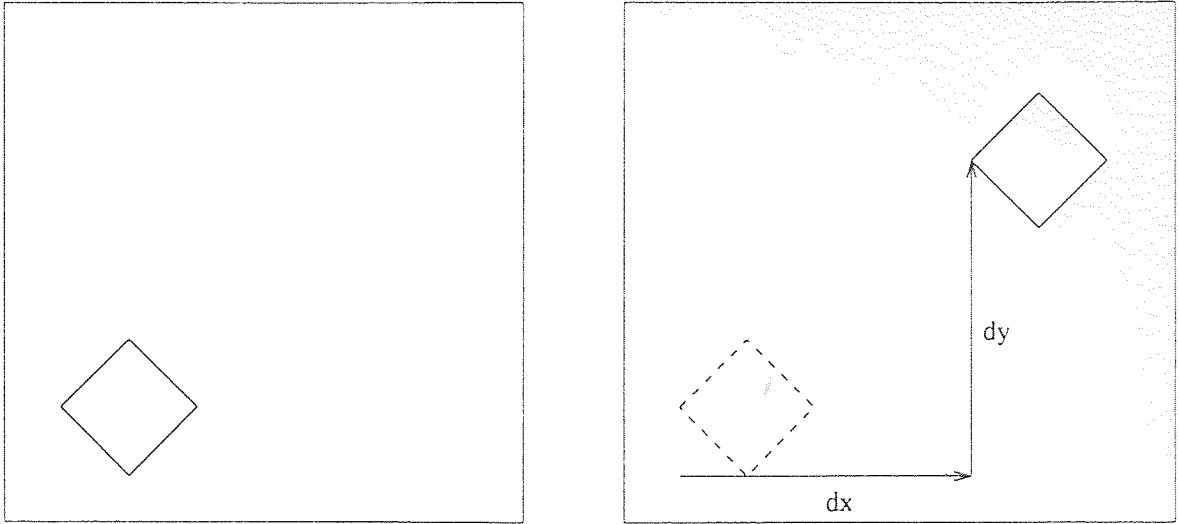
Figure 2.1 Image Translated with Displacement of (d_x, d_y)
a) $f(x, y, t_{-1})$ b)$f(x, y, t_0)$

Figure 2.1 shows an example which satisfies equation 2.1. Motion estimation [7] [13] is broadly divided into two groups: namely, the region matching method [13] and the spatio-temporal method. This paper will focus on the region matching method.

2.1 Region Matching

The region matching method considers a small region in an image frame and searches for the displacement which will produce the best match among possible regions in an adjacent frame. The displacement vector (d_x, d_y) is estimated by minimizing the following:

$$\text{Error} = \iint_{x,y \in R} |[f(x, y, t_0) - f(x - d_x, y - d_y, t_{-1})]| \, dx \, dy, \qquad (2.2)$$

where R is the local spatial region used to estimate (d_x, d_y).

Minimization of the above equation is a non-linear problem. To simplify this non-linear equation, two different methods can be used:

1. iterative method

2. block matching method.

In this study, the iterative method will be briefly introduced, whereas the block matching method will be reviewed.

2.1.1 Iterative Method

With this method, an iterative procedure [13] [14] is used to improve the estimate in each iteration. Let $(\hat{d}_x(k), \hat{d}_y(k))$ be the estimate of (d_x, d_y) after the kth iteration. In iterative methods, the estimate of (d_x, d_y) after the $(k+1)$th iteration, $(\hat{d}_x(k+1), \hat{d}_y(k+1))$, is obtained by:

$$
\begin{aligned}
\hat{d}_x(k+1) &= \hat{d}_x(k) + u_x(k) \\
\hat{d}_y(k+1) &= \hat{d}_y(k) + u_y(k),
\end{aligned}
\tag{2.3}
$$

where $u_x(k)$ and $u_y(k)$ are the update of correlation terms. The update terms vary depending on the descent method used. If we use the steepest descent method, Equation 2.4 becomes:

$$
\begin{aligned}
\hat{d}_x(k+1) &= \hat{d}_x(k) - \epsilon \left. \frac{\partial \mathrm{Error}(d_x, d_y)}{\partial d_x} \right|_{d_x = \hat{d}_x(k), d_y(k) = \hat{d}_y(k)} \\
\hat{d}_y(k+1) &= \hat{d}_y(k) - \epsilon \left. \frac{\partial \mathrm{Error}(d_x, d_y)}{\partial d_y} \right|_{d_x = \hat{d}_x(k), d_y(k) = \hat{d}_y(k)}
\end{aligned}
\tag{2.4}
$$

where ϵ is the step size that can be adjusted and $\mathrm{Error}(d_x, d_y)$ is the error in Equation 2.2 as a function of d_x and d_y for a given R. In the iterative methods, (d_x, d_y) is typically estimated at each pixel. Given $(\hat{d}_x(k), \hat{d}_y(k))$, we can use the recursion relation in Equation 2.4 only once for a pixel and then move on to the next pixel. We can also use the recursion relation more than once for a more accurate estimate of (d_x, d_y) before we move on to the next pixel.

2.1.2 Block matching

In the block matching approach [2] [5] [13] [15], [1] an image is divided into many fixed or variable size blocks, each block translated with a displacement vector (d_x, d_y).

The problem of minimizing the error in Equation 2.2 is to evaluate the error for every possible (d_x, d_y). First image is segmented in to several blocks and (d_x, d_y) estimated for each block. (d_x, d_y) is chosen such that the error is minimized. A block of pixel intensities in the current frame is compared with all possible blocks within a certain predefined search region in the previous frame. The predefined search area is fixed and has a size of $(b + 2d) \times (b + 2d)$, where b is the block size and d is the maximum displacement. Even though we generally choose the block size to be the same as R in Equation 2.2, it is not necessary to do so. It is assumed that the maximum displacement between two consecutive blocks is $\pm d$. This method segments an image into fixed size blocks and assumes that each block is undergoing independent uniform translation given by displacement vector $V = (d_x, d_Y)$ as shown in figure 2.2 [23]. To maintain the validity of the assumption, relatively small square block pixels (e.g., 8×8 or 16×16) are used in practice.

Block matching estimators rely on dividing the original image plane into blocks in which only the translation of motion is assumed. The algorithm's task is to find a translation vector for each block such that the error is minimized. Often the sum of the square differences or the sum of the absolute mean differences is adopted as the cost function, and minimization is made with respect to the restricted search area for the translation vector.

After the image is divided into several blocks, motion is detected for each block. The motion detector compares each pixel of a block at t_0 with the corresponding pixel in the block at t_{-1}. If the difference in intensities of a particular pixel in the current and in the previous frames is less than the threshold T_0, then the pixel in question is considered unchanged [1]. If the number of altered pixels within the block is greater than or equal to a certain threshold T_1, then the block is considered to be in motion. If motion is detected within the block, then motion is estimated. The block matching algorithm tries to find the best match for a block in motion in the
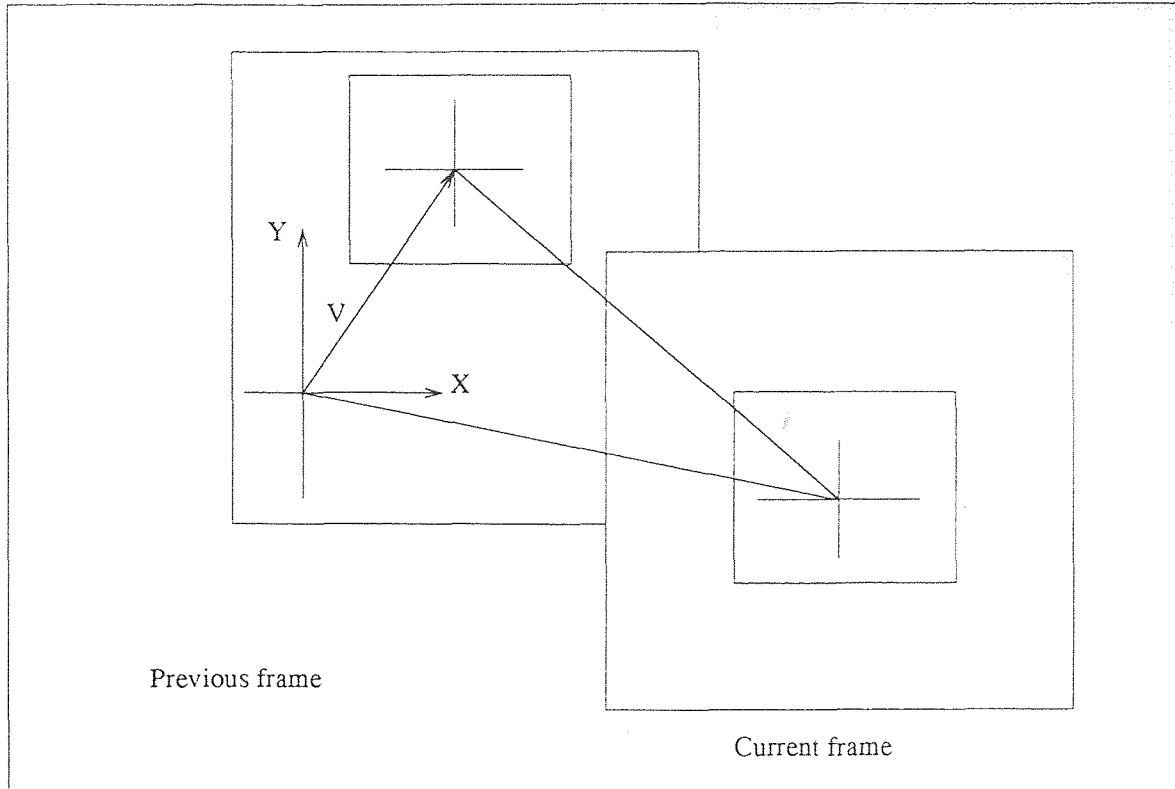
Figure 2.2 Block Matching Motion Estimation

current frame from the $b \times b$ blocks in a predefined search region in the previous frame.

The best match is found by minimizing the error in Equation 2.2. This process is repeated for all blocks, resulting in the prediction of the current frame. The difference between the original frame and the predicted frame is the prediction error, and it is termed the motion compensated frame difference (MCFD) signal. The prediction error is transmitted to the receiver along with the motion information to reconstruct the current frame. The MCFD signal and the motion information reconstruct the current frame. The above process can be equated as:

$$\mathrm{MCFD}(x,y) = f(x,y) - \hat{f}(x,y) \begin{cases} x & = 1,2,....M \\ y & = 1,2,....N. \end{cases} \tag{2.5}$$

CHAPTER 3

OPTIC FLOW ESTIMATION

Several techniques which are highly relevant to motion-compensated image coding have addressed the problem of motion estimation. Accuracy in motion information in image sequences is essential in video signal processing and computer vision applications. Two major approaches to motion estimation are block matching and the optic flow [12] [20]. Chapter 2 introduced the block matching approach. This chapter will address the optic flow approach.

Optic flow is the 2-D distribution of apparent velocities [6] [9] [14] that can be associated with the variation of brightness patterns on the image. The scene does not have to be in motion relative to the image for the optic flow to be non-zero. The image intensity is the power per unit area at a point (x,y) in the image. In Figure 3.1 the incident radiance arriving at the point $p = (x, y, z)$ on the surface is the power per unit area per unit solid angle reflected by a point in the scene, in a given direction. The direction of the incident radiance is given by the unit vector I. The direction (of the surface normal) at the point p is given by the unit vector N.

A change in the scene illumination [3] [8] [22] is one of the main effects that gives rise to an optic flow. Figure 3.2 shows that image of successive frames and the velocity field between two consecutive frames computed by an estimator.

Different approaches [12] [19] [22] optic flow estimation have been used in past works. Two of these approaches, listed below, will be briefly introduced in this work:

1. gradient-based approach,

2. correlation-based approach,

9

3.1 Gradient-Based Approach

Techniques based on the gradient-based approach conserve information about image intensity. These approaches assume that for a given scene point, the intensity I at the corresponding image point remains constant over time. In other words, if a scene point p, depicted in Figure 3.1, projects onto the image point (x, y) at time t and onto the image point $(x + \delta x, y + \delta y)$ at time $(t + \delta t)$, we can write

$$I(x, y, t) = I(x + \delta x, y + \delta y, t + \delta t). \tag{3.1}$$

Expanding the right-hand side in a Taylor series about (x, y, t) and ignoring the second higher order terms, we obtain:

$$I(x + \delta x, y + \delta y, t + \delta t) = I(x, y, t) + \delta x \frac{\partial I}{\partial x} + \delta y \frac{\partial I}{\partial y} + \delta t \frac{\partial I}{\partial t}. \tag{3.2}$$

Combining Equations 3.1 and 3.2 results in the following:

$$\delta x \frac{\partial I}{\partial x} + \delta y \frac{\partial I}{\partial y} + \delta t \frac{\partial I}{\partial t} = 0. \tag{3.3}$$

Dividing by δt and denoting the partial derivatives by I_x, I_y, and I_t and denoting the local velocity by (u, v), we obtain:

$$\frac{\delta x}{\delta t} \frac{\partial I}{\partial x} + \frac{\delta y}{\delta t} \frac{\partial I}{\partial y} + \frac{\partial I}{\partial t} = 0$$
$$u I_x + v I_v + I_t = 0. \tag{3.4}$$

Equation 3.4 is referred to as the motion constraint. The local constraint provides one linear equation in the variables u and v. Thus, the velocity vector (u, v) cannot be determined locally without applying additional constraints. The above equation can compute the optic flow with out applying additional constraint. To determine the velocity, the smoothness constraint as an auxiliary equation is added. Horn and Schunck [10] assumed conservation of intensity and smoothness of flow-field. The error comes from two major sources, the rate of change of image brightness and the

departure from smoothness in the image velocity, therefore Equation 3.4, not exactly zero. It is expressed as follows:

$$\epsilon_b = (uI_x + vI_y + I_t) \tag{3.5}$$

$$\epsilon_c^2 = \left(\frac{\delta u}{\delta x}\right)^2 + \left(\frac{\delta u}{\delta y}\right)^2 + \left(\frac{\delta v}{\delta x}\right)^2 + \left(\frac{\delta v}{\delta y}\right)^2 \tag{3.6}$$

where ϵ_b is the error for the rate of change of image brightness, and ϵ_c is the error from smoothness of the velocity flow. The problem of optic-flow estimation is to minimize the sum of the errors of the above equations, i.e., minimizing

$$\epsilon^2 = \int\int \left|\left[(\alpha^2\epsilon_c^2 + \epsilon_b^2)\right]\right| dx dy \tag{3.7}$$

where α is the weighting factor.

By using calculus of variation, we get:

$$\begin{aligned}
I_x^2 u + I_x I_y v &= \alpha^2 \nabla^2 u - I_x I_t \\
I_x I_y u + I_y^2 v &= \alpha^2 \nabla^2 v - I_y I_t
\end{aligned} \tag{3.8}$$

where

$$\nabla^2 u = \frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2}$$

$$\nabla^2 v = \frac{\partial^2 v}{\partial x^2} + \frac{\partial^2 v}{\partial y^2}$$

Using the approximation to the Laplacian [10],

$$\nabla^2 u = \bar{u} - u \tag{3.9}$$

$$\nabla^2 v = \bar{v} - v \tag{3.10}$$

equation 3.8 becomes

$$(\alpha^2 + I_x^2)u + I_x I_y v = (\alpha^2 \bar{u} - I_x I_t) \tag{3.11}$$

$$I_x I_y u + (\alpha^2 + I_y^2)v = (\alpha^2 \bar{v} - I_x I_t) \tag{3.12}$$

The determinant of the coefficient matrix equals $\alpha^2(\alpha^2 + I_x^2 + I_y^2)$. Solving for u and v, we get,

$$(\alpha^2 + I_x^2 + I_y^2)u = (\alpha^2 + I_y^2)\bar{u} - I_x I_y \bar{v} - I_x I_t \tag{3.13}$$

$$(\alpha^2 + I_x^2 + I_y^2)v = -I_x I_y \bar{u} + (\alpha^2 + I_x^2)\bar{v} - I_y I_t \tag{3.14}$$

We have a pair of equations for each point in the image. We can compute a new set of the velocity estimates $(u^{(n+1)}, v^{(n+1)})$ by

$$u^{(n+1)} = \bar{u}^n - I_x \frac{I_x \bar{u}^n + I_y \bar{v}^n + I_t}{(\alpha^2 + I_x^2 + I_y^2)} \tag{3.15}$$

$$v^{(n+1)} = \bar{v}^n - I_y \frac{I_x \bar{u}^n + I_y \bar{v}^n + I_t}{(\alpha^2 + I_x^2 + I_y^2)} \tag{3.16}$$

where $\bar{u}$ and $\bar{v}$ are the neighborhood average of image velocity.

3.2   Correlation-Based Approach

This approach uses two successive images of a time-varying scene. Singh's approach [21] uses region-based matching citeglazer [24]. This technique involves finding the best match for the pixel in the first frame from a group of pixels in the second frame.

Singh [21] splits the match technique into two steps

1. conservation information

2. neighborhood information.

Conservation information is the information that can be recovered by local measurements alone, by assuming conservation of some image property over time. The procedure of extracting the conservation information can be outlined as follows:

1. A correlation window w_c of size $(2m+1) \times (2m+1)$ is formed around the (x, y) pixel in the first image.

2. A search window w_s of size $(2n+1) \times (2n+1)$ is formed around the same location of (x, y) in the second image.

3. The match measure is computed by matching the correlation window w_c against a similar window around the candidate pixel that lies in the search window w_s.

4. The match measure $\epsilon_c(d_x, d_y)$ between w_c and a similar $(2n+1) \times (2n+1)$ window around each pixel in w_s, is displaced from (x, y) by an amount (d_x, d_y) and computed as the sum of the square of the differences as follows:

$$\epsilon_c(d_x, d_y) = \sum_{j=-n}^{n} \sum_{i=-n}^{n} w(i,j)[I_1(x+i, y+j) - I_2(x+d_x+i, y+d_y+j)]^2 \quad (3.17)$$

where $I_1(x, y)$ and $I_2(x, y)$ refer to the pixel intensities at the location (x, y) in the first and second images, respectively, and w denotes a 2-D window function.

5. The best match is found by minimizing the match measure or by minimizing the sum square difference (SSD). The pixel in w_s with the lowest match measure is selected as the best match.

The response distribution is given in terms of the exponential error-distribution as follows

$$R_c(u, v) = e^{k\epsilon_c(u,v)} \quad (3.18)$$

Each point in a search area is a candidate for the true match. Based in the weighted-least squares estimation, the estimate of of image velocity, denoted by $U_{cc} = (u_{cc}, v_{cc})$, can be computed as follows:

$$u_{cc} = \frac{\sum_u \sum_v R_c(u,v)u}{\sum_u \sum_v R_c(u,v)} \quad (3.19)$$

$$v_{cc} = \frac{\sum_u \sum_v R_c(u,v)v}{\sum_u \sum_v R_c(u,v)} \quad (3.20)$$

Neighborhood information is used to improve computation of image velocities. Neighborhood information is used to propagate velocity. Forming the neighborhood window of size $(2w+1) \times (2w+1)$, the velocities of these $(2w+1)^2$ pixels are mapped to the points (u_i, v_i) in uv space (where $1 \leq i \leq (2w+1)^2$), with the weight to the

point (u_i, v_i) as $R_n(u_i, v_i)$. Based on the weighted least square estimation, the image velocity of the central pixel denoted by $\bar{U} = (\bar{u}, \bar{v})$, is given by:

$$\bar{u} = \frac{\sum_u \sum_v R_c(u,v)u}{\sum_u \sum_v R_c(u,v)} \tag{3.21}$$

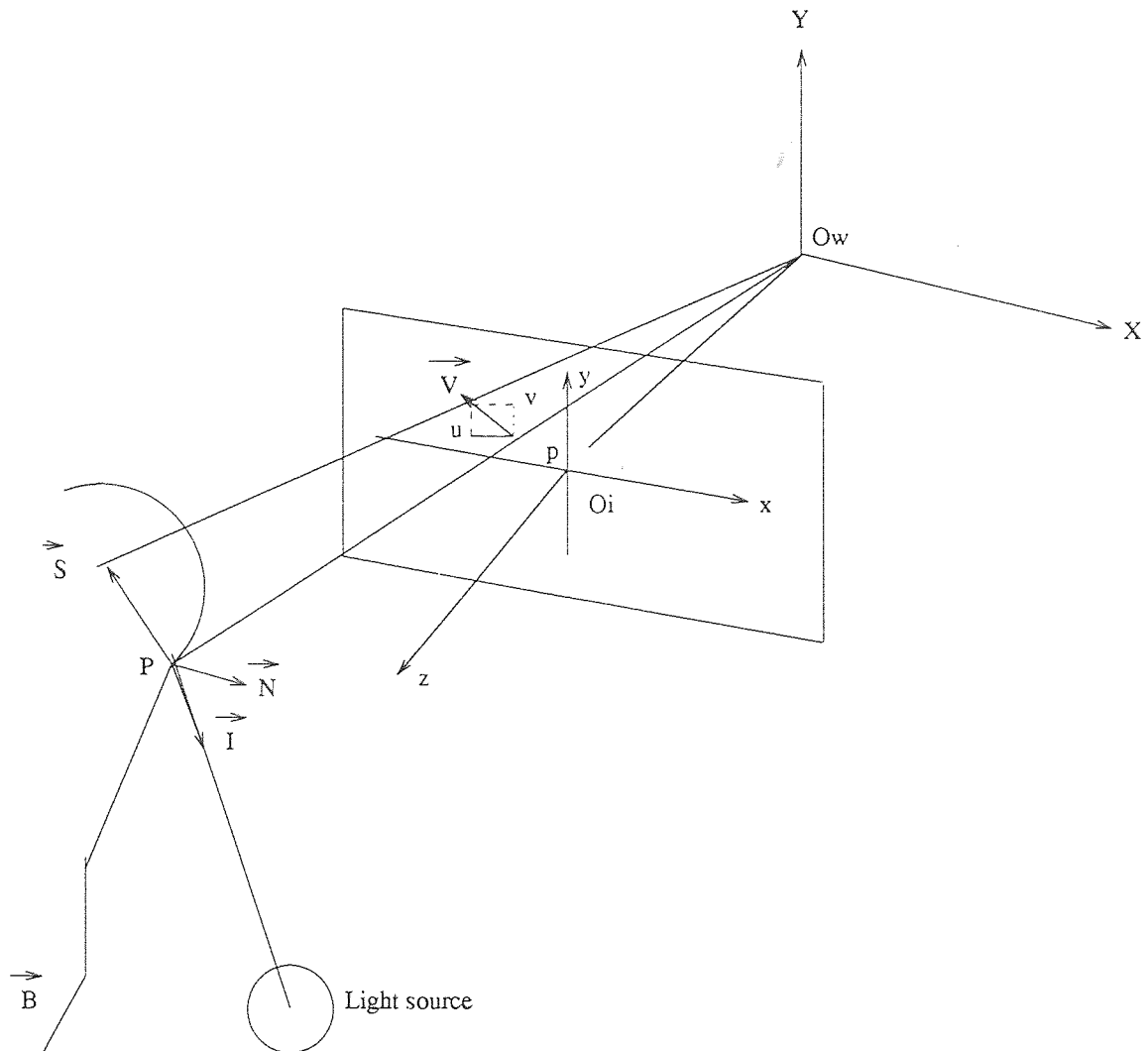$$\bar{v} = \frac{\sum_u \sum_v R_c(u,v)v}{\sum_u \sum_v R_c(u,v)} \tag{3.22}$$
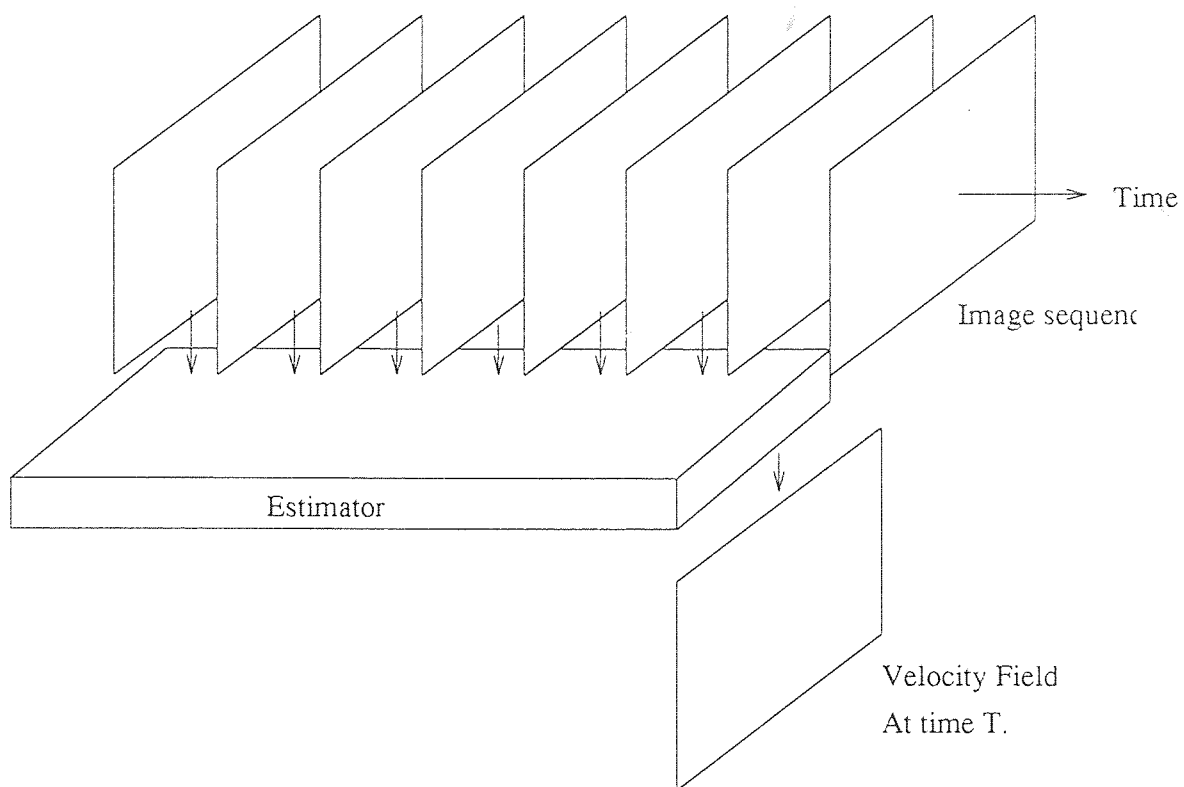
Figure 3.1 The Imaging Geometry

Figure 3.2 A Description of Optic Flow Estimation

CHAPTER 4

MOTION-COMPENSATION USING CORRELATION-FEEDBACK

4.1 Correlation-Feedback Approach

This is a new technique developed by J.N. Pan and Y.Q. Shi [17], which is based on the correlation-based approach and the concept of feedback. In this study, a virtual continuous image is obtained via a bilinear interpolation applied to a digital image. The concept of feedback is used to reduce error in calculating optical flow. It has been proved that this approach is more accurate than the gradient-based and the correlation-based approaches.

Correlation-feedback is characterized by a correlation step and a propagation step, depicted in Figure 4.1. Before correlation step, a continuous 2-D image function $f(i,j)$ is estimated from the first digital image $I(x,y)$ by using the bilinear interpolation technique [18]. In the bilinear interpolation technique, gray levels are assigned to the pixel (x,y) in the new image, given as follows:

Let the integer part of (x'',y'') be (x,y), so that the point (x'',y'') is surrounded by the four integer coordinate points:

$$(x,y+1) \qquad (x+1,y+1)$$
$$\left(x'',y''\right) \qquad\qquad (4.1)$$
$$(x,y) \qquad\qquad (x+1,y).$$

Let the fractional part of x'' and y'' be $\alpha = x'' - x$ and $\beta = y'' - y$, respectively, where $0 \le \alpha, \beta < 1$. Then the gray level that we assign to (x'',y'') is given by using bilinear interpolation [18] as follows:

$$[(1-\alpha)(1-\beta) \ (1-\alpha)\beta \ \alpha(1-\beta) \ \alpha\beta] \ [f(x,y) \ f(x,y+1) \ f(x+1,y) \ f(x+1,y+1)]^T.$$
$$(4.2)$$

Initially, image 1 is defined as follows:

$$f_1(i_1,j_1) = I_1(i_1,j_1) \qquad\qquad (4.3)$$

17

where i_1 and j_1 are indexes of I_1. Then, using the above technique, image 2 is defined as follows:

$$f_2\left(i_2 - u_n(i_2, j_2), j_2 - u_n(i_2, j_2)\right) =$$

$$(1 - a)[(1 - b)I_1(int(x), int(y)) + bI_1(int(x), int(y) + 1)]$$

$$+ \quad a[(1 - b)I_1(int(x) + 1, int(y)) + bI_1(int(x) + 1, int(y) + 1)] \qquad (4.4)$$

where

$$\begin{aligned}
x &= int(i_2 - u_n) \\
y &= int(i_2 - v_n) \\
a &= i_2 - u_n - int(x) \\
b &= j_2 - v_n - int(y)
\end{aligned} \qquad (4.5)$$

The correlation window w_c of size $(2n + 1) \times (2n + 1)$ is formed around the pixel at the location (x, y) in the first image I_1.

A search window w_s of size $(2N + 1) \times (2N + 1)$ is formed around the pixel at the location in the second image I_2. The error distribution is computed from the sum of the square of the difference as follows:

$$\epsilon_c(u, v) = \sum_{j=-n}^{n} \sum_{i=-n}^{n} [I_2(i_2 + x, j_2 + y) - f_2(i_2 + x - u, j_2 + y - v)]^2 \qquad (4.6)$$

Response distribution is computed as follows:

$$R_c(u, v) = \exp\left(-k\epsilon(u, v)\right) \qquad (4.7)$$

where k is an appropriately chosen parameter.

Based on the weighted-least squares estimation:

$$u_c(i_2, j_2) = \frac{\sum_u \sum_v R_c(u, v)u}{\sum_u \sum_v R_c(u, v)} \qquad (4.8)$$

$$v_c(i_2, j_2) = \frac{\sum_u \sum_v R_c(u, v)v}{\sum_u \sum_v R_c(u, v)} \qquad (4.9)$$

The propagation step is achieved by using neighborhood window of size $(2w + 1) \times (2w + 1)$. By mapping the velocity of this window $(2w + 1) \times (2w + 1)$ pixels to the

point (u_i, v_i) in the uv space and by choosing the weight $w(x, y)$ as a Gaussian mask, we get:

$$u_{n+1} = \sum_{x=-n}^{n} \sum_{y=-n}^{n} w(x, y) u_c(i_2 + x, j_2 + y) \qquad (4.10)$$

$$v_{n+1} = \sum_{x=-n}^{n} \sum_{y=-n}^{n} w(x, y) v_c(i_2 + x, j_2 + y) \qquad (4.11)$$
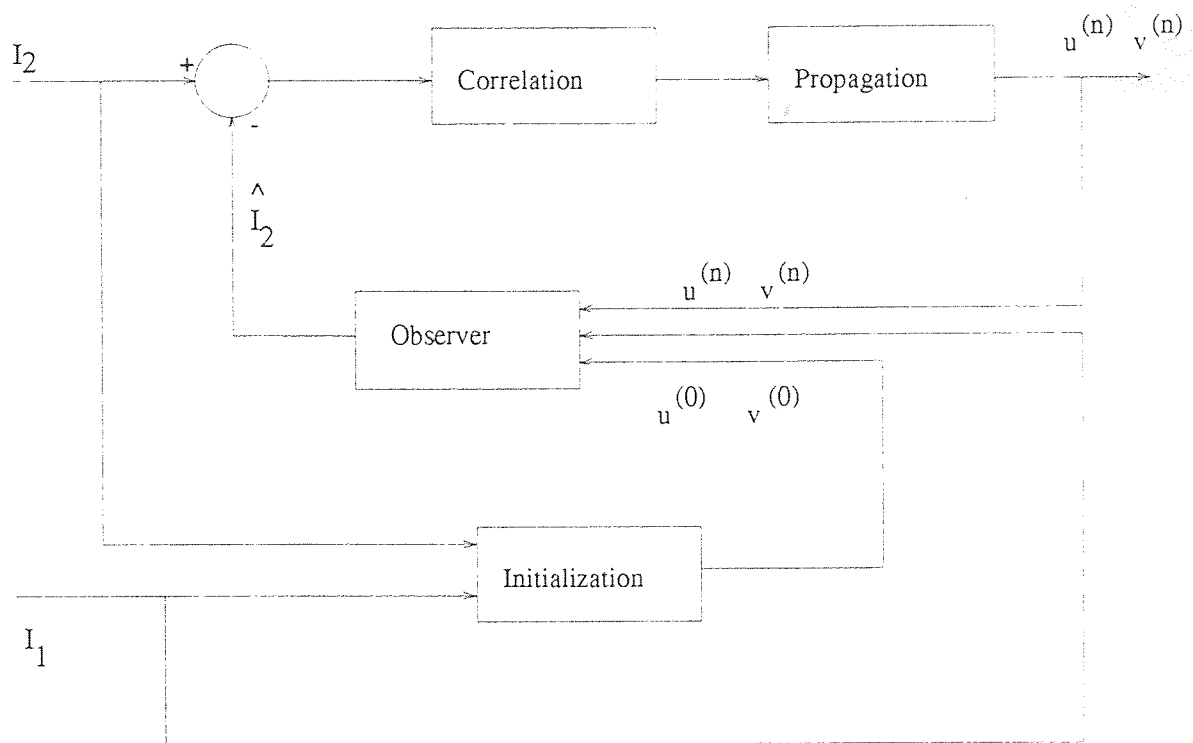
Figure 4.1 Block Diagram for Correlation-Feedback Approach

4.2 Analysis of Convergence

Correlation-feedback is convergent, if the normalized response distribution

$$u^{(n+1)} = \frac{R_c(u_i^n, v_j^n)}{\sum_{i=0}^{4} \sum_{j=0}^{4} R_c(u_i^n, v_j^n)} \qquad (4.12)$$

is symmetric and has only one peak which is assumed by the true image velocity and the image intensity is a linear function of coordinates. When $\left| u^{(n+1)} - u^n \right|$ and

$\left| v^{(n+1)} - v^n \right|$ are greater than predefined threshold, we update $\hat{I}_2$ in the observer stage according to $u^{(n+1)}(i, j)$ and $v^{(n+1)}(i, j)$ and I_1. Otherwise, the algorithm is convergent [17]. It is shown below that the (u^n, v^n) converges to the true optic-flow vector. If the propagation stage is neglected, Equation 4.8 can be written as

$$u^{(n+1)} = \frac{\sum_{i=0}^{4}\sum_{j=0}^{4} R_c(u_i^n, v_j^n) u_i^n}{\sum_{i=0}^{4}\sum_{j=0}^{4} R_c(u_i^n, v_j^n)} \tag{4.13}$$

$$(u_0^n, u_1^n, u_3^n, u_4^n) = \left(u^n - \frac{u^n}{2n}, u^n - \frac{u^n}{4n}, u^n, u^n + \frac{u^n}{4n}, u^n + \frac{u^n}{2n} \right) \tag{4.14}$$

Let the normalized response distribution be denoted by $g(u_i^n, v_i^n)$; then,

$$g(u_i^n, v_i^n) = \frac{R_c(u_i^n, v_i^n)}{\sum_{i=0}^{4}\sum_{j=0}^{4} R_c(u_i^n, v_j^n)}. \tag{4.15}$$

Consider the case where (u_i^n, v_j^n) is in the vicinity of true image vector (u^a, v^a). Assume $g(u_i^n, v_j^n)$ is a sample of a surface of revolution having only one extreme point (u_i^n, v_j^n).

Let

$$h(u_i^n) = \frac{\sum_{i=0}^{4} R_c(u_i^n, v_j^n)}{\sum_{i=0}^{4}\sum_{j=0}^{4} R_c(u_i^n, v_j^n)} \tag{4.16}$$

where $h(u_i^n)$ is a sample of an asymmetrical curve about axis $u_i^n = u^a$ with only one extreme value at $u_i^n = u^a$.

Pan [17] used the following Lemmas for the convergence proof:

Lemma 1: $h(u_0^n), h(u_1^n), h(u_2^n), h(u_3^n), h(u_4^n)$ satisfy the following equation.

$$\sum_{i=0}^{4} h(u_i^n) = 1. \tag{4.17}$$

Lemma 2: $u^{(n+1)}$ and u^n have the following relationships,

$$1. \ u^{(n+1)} = u^n \left(1 + \frac{h(u_4^n) - h(u_0^n)}{2n} + \frac{h(u_3^n) - h(u_1^n)}{4n} \right)$$

2. $u_3^n - u^{(n+1)} = u^n \left(\dfrac{3h(u_0^n)}{4n} + \dfrac{h(u_2^n)}{4n} + \dfrac{2h(u_1^n)}{4n} - \dfrac{h(u_4^n)}{4n} \right)$

3. $u_4^n - u^{(n+1)} \geq 0.$

Lemma 1 is apparent and its proof is therefore omitted. The proof of Lemma 2 is given below. From Equations 4.13 and 4.15, the proof of the first part of Lemma 2 is given as follows:

$$u^{(n+1)} = \sum_{i=0}^{4} u_i^n h(u_i^n) \tag{4.18}$$

$$= u^n \left(1 + \frac{h(u_4^n) - h(u_0^n)}{2n} + \frac{h(u_3^n) - h(u_1^n)}{4n} \right) \tag{4.19}$$

Similarly, the second part of Lemma 2 follows from Equations 4.13, 4.14, 4.16.

$$u_3^n - u^{(n+1)} = u^n \left(\frac{3h(u_0^n)}{4n} + \frac{h(u_2^n)}{4n} + \frac{2h(u_1^n)}{4n} - \frac{h(u_4^n)}{4n} \right) \tag{4.20}$$

With Equations 4.13, 4.14, 4.16, and some algebraic manipulations, the third part of Lemma 2 can proved as follows:

$$u_4^n - u^{(n+1)} = u^n \left(\frac{h(u_0^n)}{2n} + \frac{3h(u_1^n)}{4n} + \frac{h(u_3^n)}{4n} \right) \tag{4.21}$$

$u^4 - u^{(n+1)} > 0$, since $h(u_i^n) > 0$ $i = 0, \ldots, 4$.

Pan [17], gives proof of convergence of the algorithm:

$$\lim_{n \to \infty} |u^n - u^a| \to 0. \tag{4.22}$$

The gradient-based, correlation-based, and correlation-feedback have been used to estimate motion.

Motion compensation has been traditionally resolved using block matching methods. Pan's [17] correlation-feedback proved to converge faster. This work will apply correlation-feedback for motion compensation because the algorithm is convergent, (u_n, v_n) tends to the true optical-flow, and the concept of feedback reduces the error in determining the optic-flow in the iterative procedure. Therefore,

this true optical-flow creates a better predicted frame and because of this the difference between the predicted frame and the current frame which is motion compensated difference (MCFD) signal is reduced. In general, the smaller the difference, the smaller the prediction error.

Motion is compensated using the following algorithm:

After the optical flow vector $\vec{U}^n = (u^n, v^n)$ is obtained, we can define the prediction or estimate of the current frames as follows:

$$\hat{I}_2(i_2, j_2) = f\left(i_2 - u^n(i_2, j_2) \, , \, j_2 - v^n(i_2, j_2)\right) \qquad (4.23)$$

where $\hat{I}_2(i_2, j_2)$ is an estimate of $I_2(i_2, j_2)$. The prediction error, which is the difference between the original frame and the predicted frame, is calculated, and it is termed the motion-compensated frame difference signal, i.e.,

$$\text{MCFD}(i, j) \; = \; I_2(i, j) - \hat{I}_2(i, j). \qquad (4.24)$$

Both the prediction error MCFD and the optic flow are transmitted to the receiver in order to reconstruct the current frame.

CHAPTER 5

EXPERIMENTAL RESULTS

5.1 Implementation of Block Matching

The procedure of block matching method is given by:

1. The current frame is divided into $(B \times B)$ blocks.

2. A search array as shown in figure 5.1 is generated from the previous frame of size $(B + 2d) \times (B + 2d)$ where B is the block size and $d = (d_x, d_y)$ is the displacement vector. It is assumed that the maximum displacement between two consecutive $(B \times B)$ blocks in image frames is $\pm d$ pixels and there are $(2d + 1)^2$ locations to search for the best match for the present block.

3. Each block in the current frame is compared with all possible blocks of the search area in the previous frame.

4. A pixel is considered a moving pixel if the difference between two pixels of consecutive blocks is greater than the given threshold (in this experiment, the threshold is equal to three).

5. A block is considered a moving block, if the moving pixels in a block is greater than certain threshold (in this experiment the threshold is equal to 10).

6. The block matching algorithm tries to find the best match of $(B \times B)$ size blocks of the current frame.

7. The relative position of the two blocks define a motion vector associated with the current block.

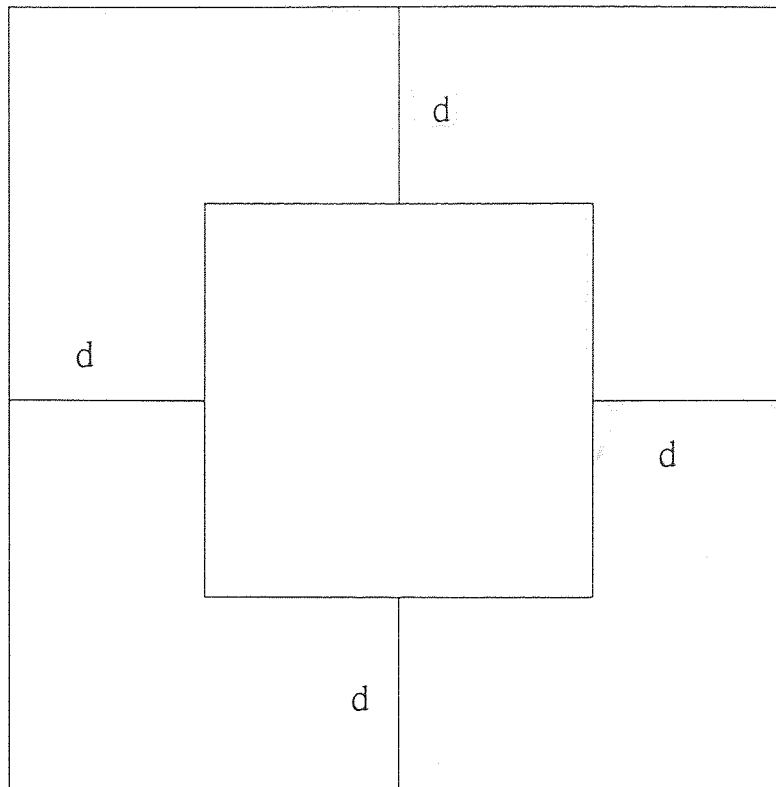8. The closest block is then used as a predictor for the current block.

Figure 5.1 Search Area

9. The collection of all the motion vectors define a motion field and is sent to receiver.

0. The process is repeated for all blocks and the prediction of the current frame is obtained.

1. The prediction error is transmitted to the receiver together with the motion information to reconstruct the the current frame. The prediction error is called the motion compensated frame difference (MCFD).

5.2 Implementation of Correlation-Feedback

Three images I_1, I_2, I_3 are used. I_1, I_2, and I_3 are the first, second and third frames of the image sequences. The time interval between I_1 and I_2 is assumed to be the same as the time interval between I_2 and I_3. It is also assumed that uniform motion occurs during these two intervals. The technique used for computing the optic flow using correlation-feedback is as follows:

1. The initial values $(u^{(o)}(i,j), v^{(0)}(i,j))$ are made available by using the Horn and Schunck's algorithm [10].

2. From $(u^{(0)}(i,j), v^{(0)}(i,j))$ and I_1 by using bilinear interpolation [18] the interpolated image $\hat{I}_2$ was found.

3. The search area is obtained from the interpolated image.

4. for each pixel in I_2, a correlation window W_c of size $(2m+1) \times (2m+1)$ is formed.

5. A correlation measure is selected to search for the best match for a given pixel of I_2 in a search-area. In this work the sum-of-square-differences (SSD) [21],[3] is used.

6. Response distribution $R_c(u^{(n)}, v^{(n)})$ can be calculated using Equations 4.6 and 4.7.

7. $(u^{(n)}(i,j), v^{(n)}(i,j))$ is obtained by using Equation 4.8.

8. The idea of feedback is used to update the estimate of I_2. The estimate of the second image $\hat{I}_2$ is only updated when $\left|u^{(n+1)} - u^n\right|$ and $\left|v^{(n+1)} - v^n\right|$ is greater than the predefined threshold.

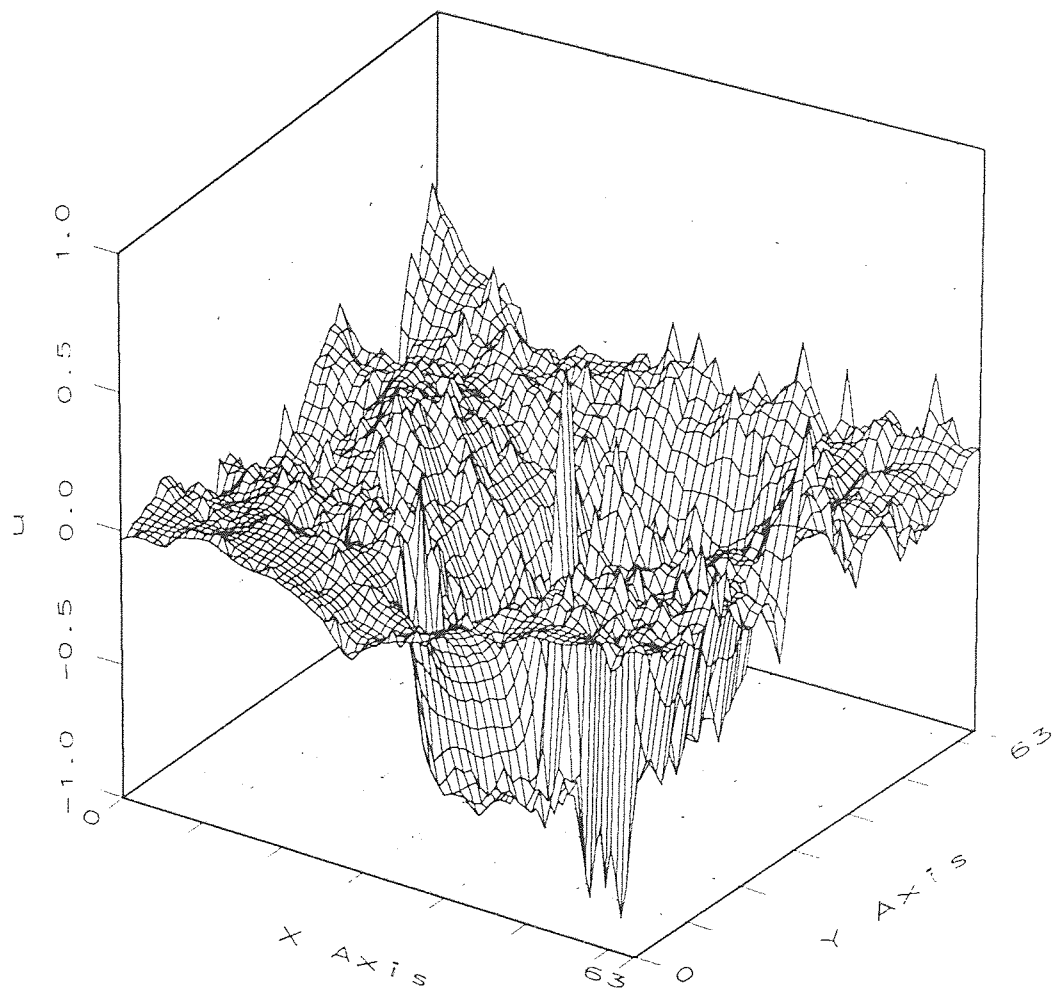9. The MCFD and the optic flow field send to receiver end to reconstruct the current image or frame.

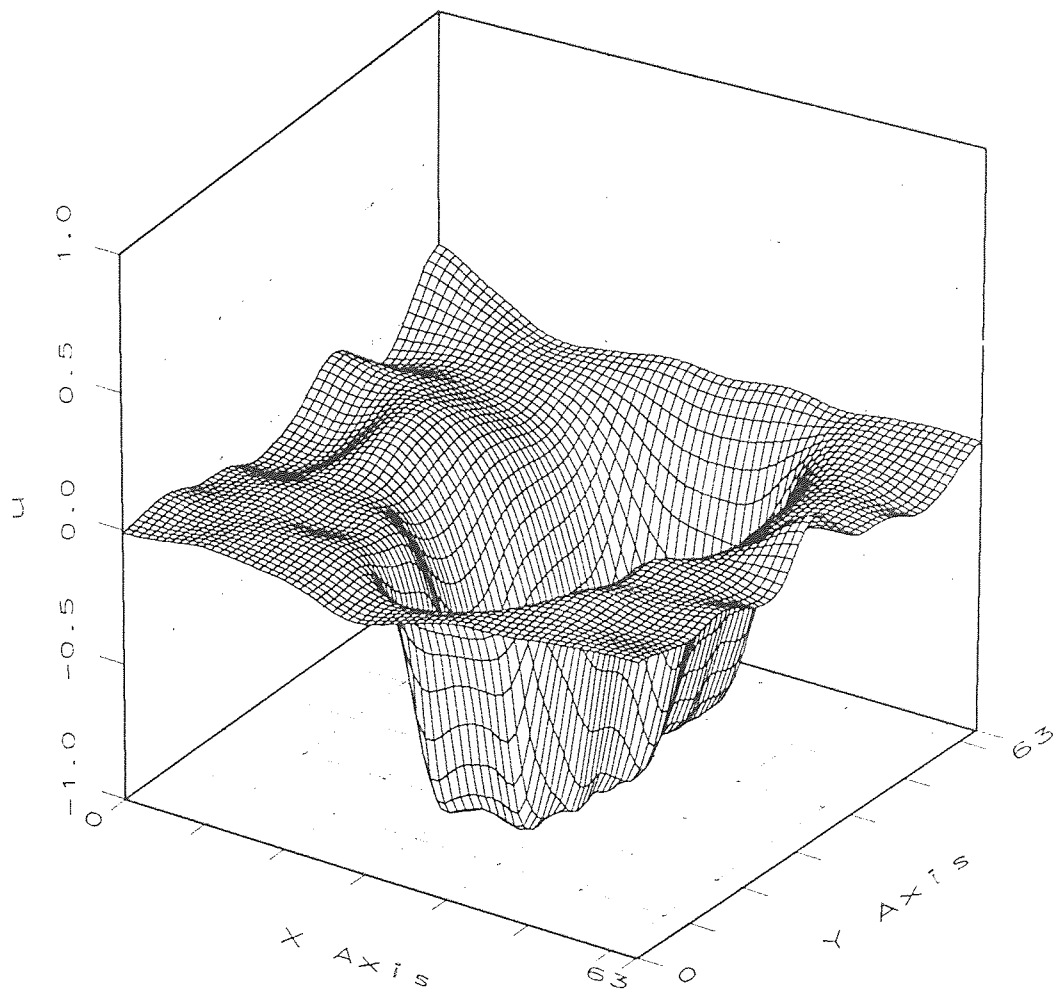**Figure 5.2** The Optic-Flow Field Calculated by the Gradient-Based Approach

Figure 5.3 The Optic-Flow Field Calculated by the Correlation-Feedback Approach
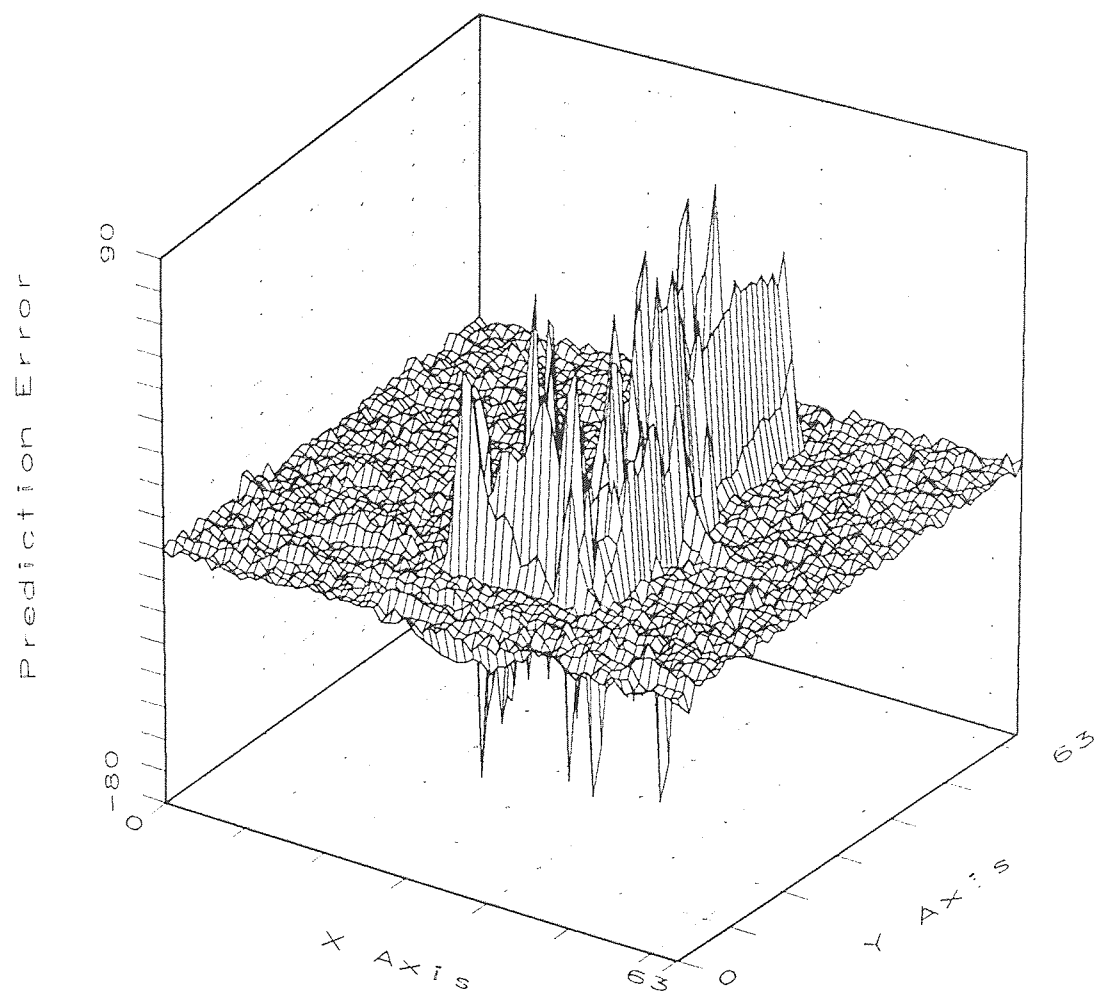
Figure 5.4 The Prediction Error Calculated by the Gradient-Based Approach
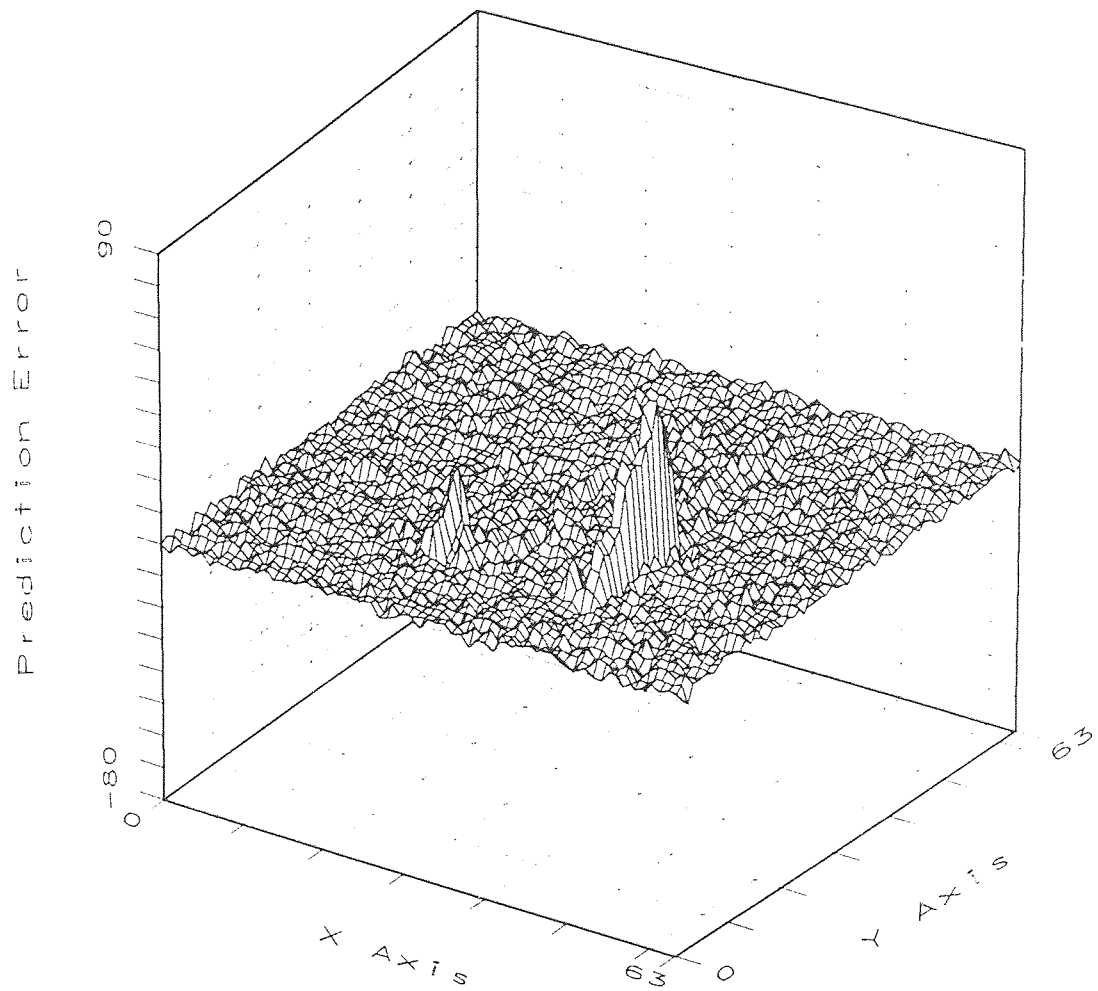
Figure 5.5 The Prediction Error Calculated by the Correlation-Feedback

5.3 Experimental Results

This section will cover the application of the gradient-based and correlation-feedback approaches on moving image sequences. Four different images, which have varying amounts of detail, will be used to evaluate the performance of these two methods. The optic-flow obtained by correlation-feedback is smoother than the gradient-based approach as evidenced in Figures 5.3 and 5.2, respectively. The plot of the prediction error of the above two approaches is shown in Figures 5.4 and 5.5. Less error is produced using correlation-feedback. Therefore, the correlation-feedback approach performs better than the gradient-based approach.

5.3.1 Experiment I

In this experiment 64 × 64 image sequences, depicted in Figure 5.6, 5.7, and 5.8 are used. The correlation-feedback yields MSE which is significantly smaller than that produced by the gradient-based approach, as illustrated in Figures 5.10 and 5.9, respectively. Table 5.1 gives the numerical results for the MSE of the MCFD. It shows that correlation-feedback produces 92% less error than the gradient-based approach for this image.

Table 5.1 The Comparison of Experiment I

	Gradient-Based Approach	Correlation-Feedback Approach
	$iteration = 120$ $alpha = 5$	$iteration = 15$ $horn - iteration = 30$ $sw = 5 \times 5$ $cw = 3 \times 3$
MSE	140.1	10.9

where

sw = search window

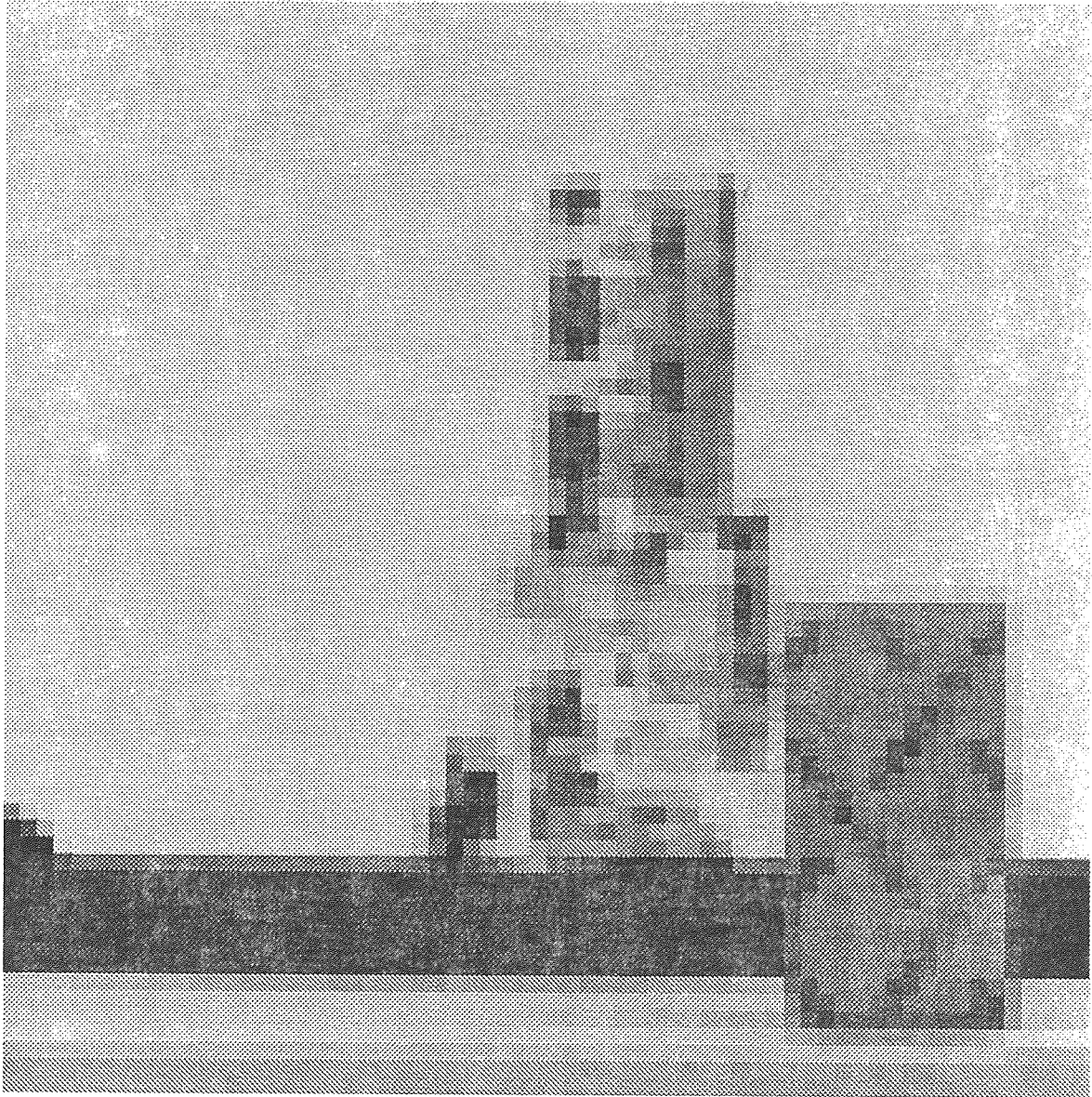cw = correlation window

$alpha$ = smoothing factor
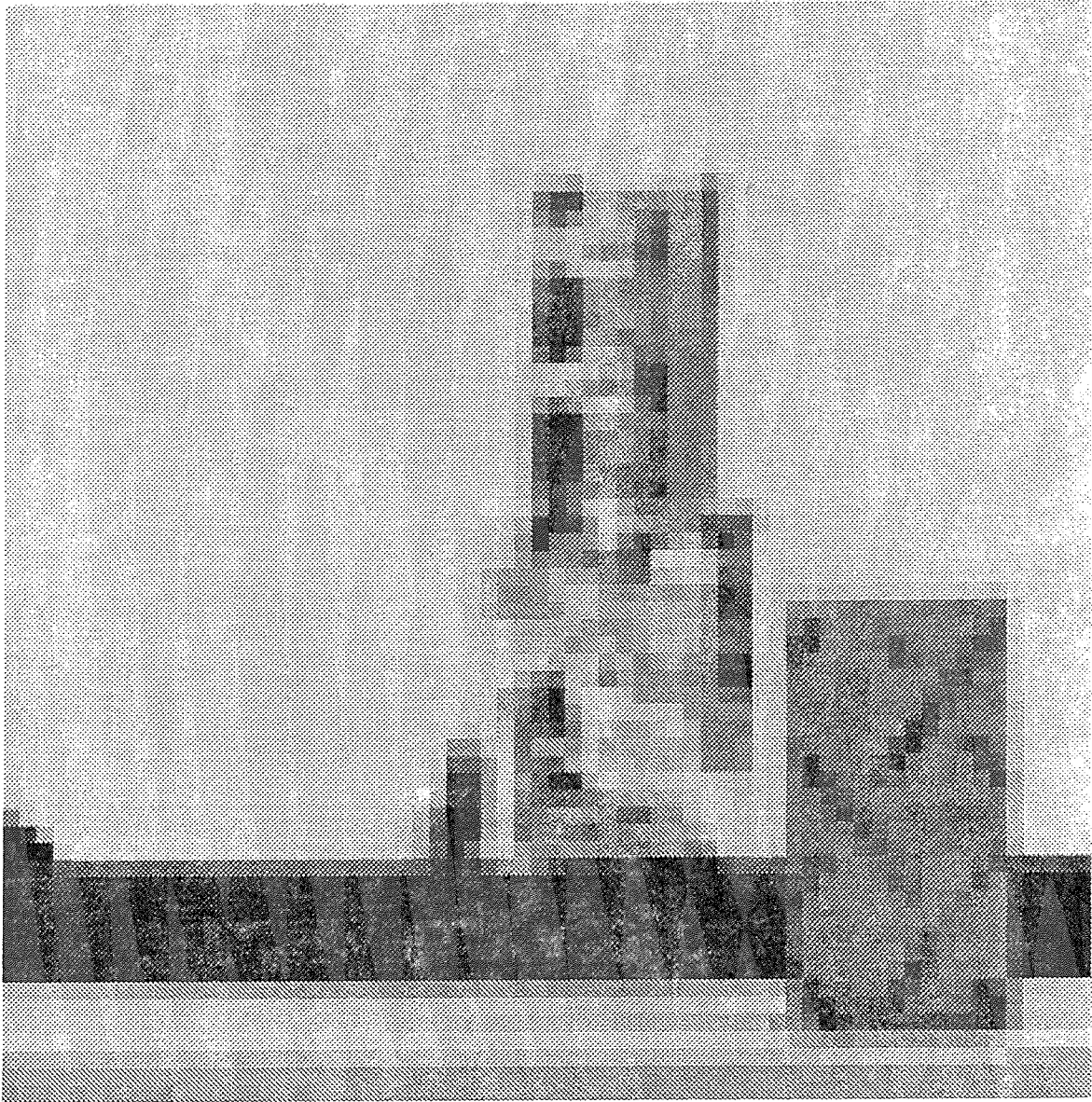
Figure 5.6 Image I Used for Experiment I
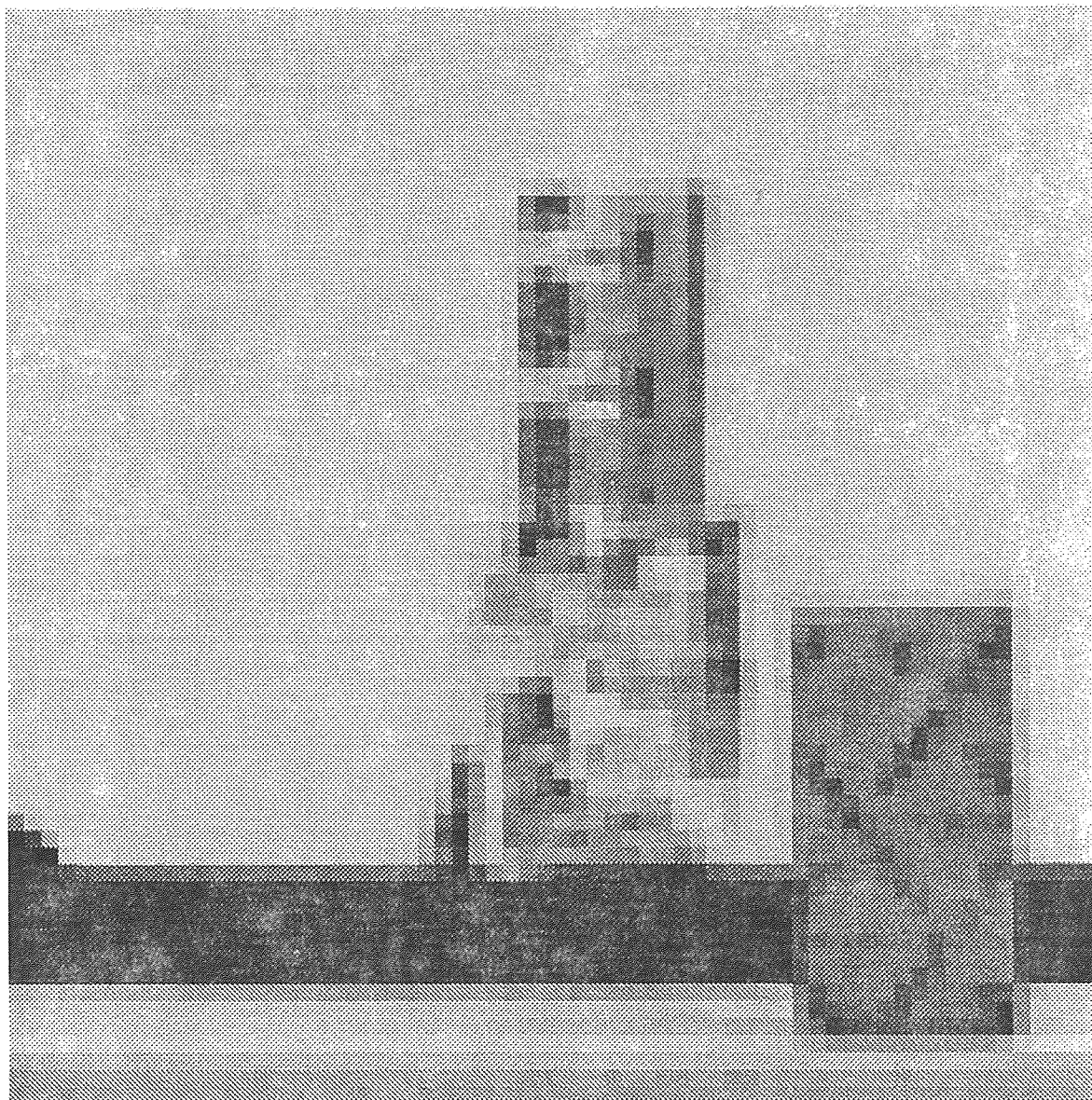
Figure 5.7 Image II Used for Experiment I

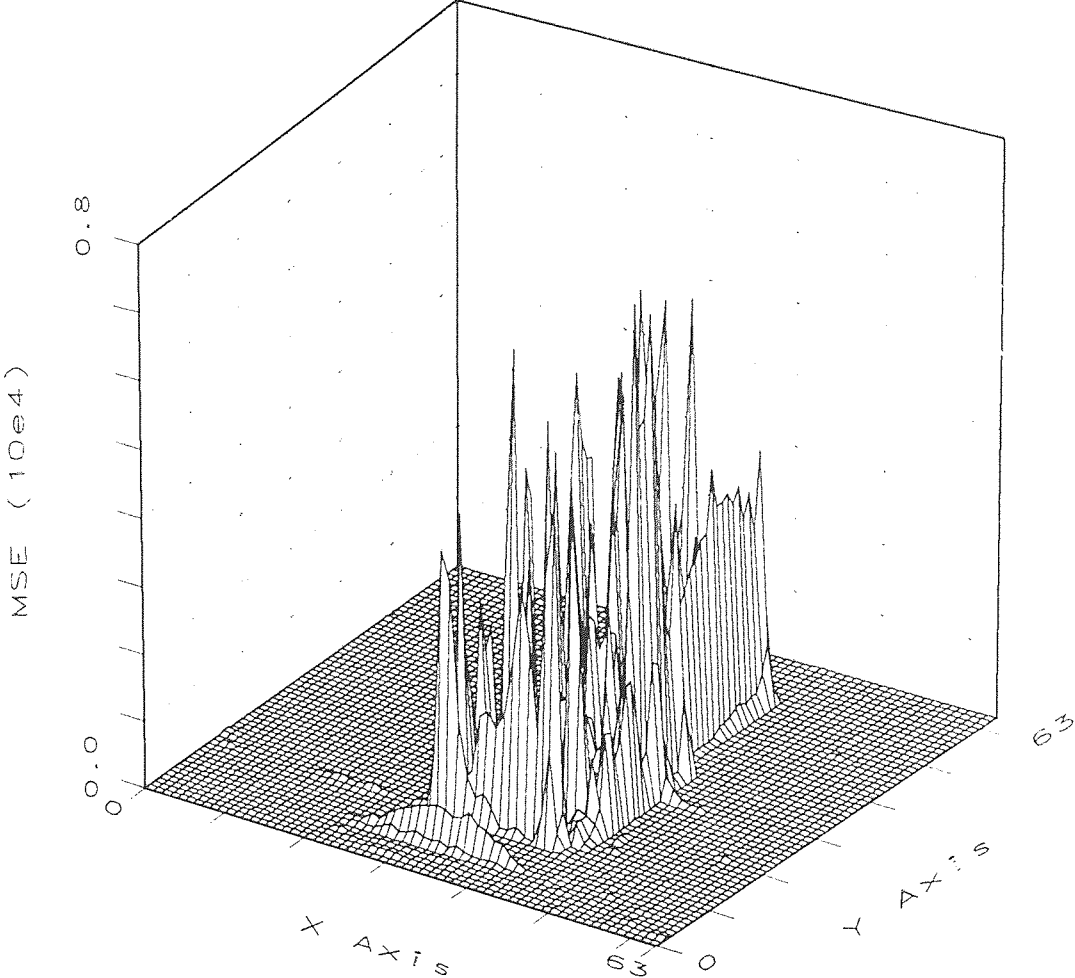Figure 5.8 Image III Used for Experiment I

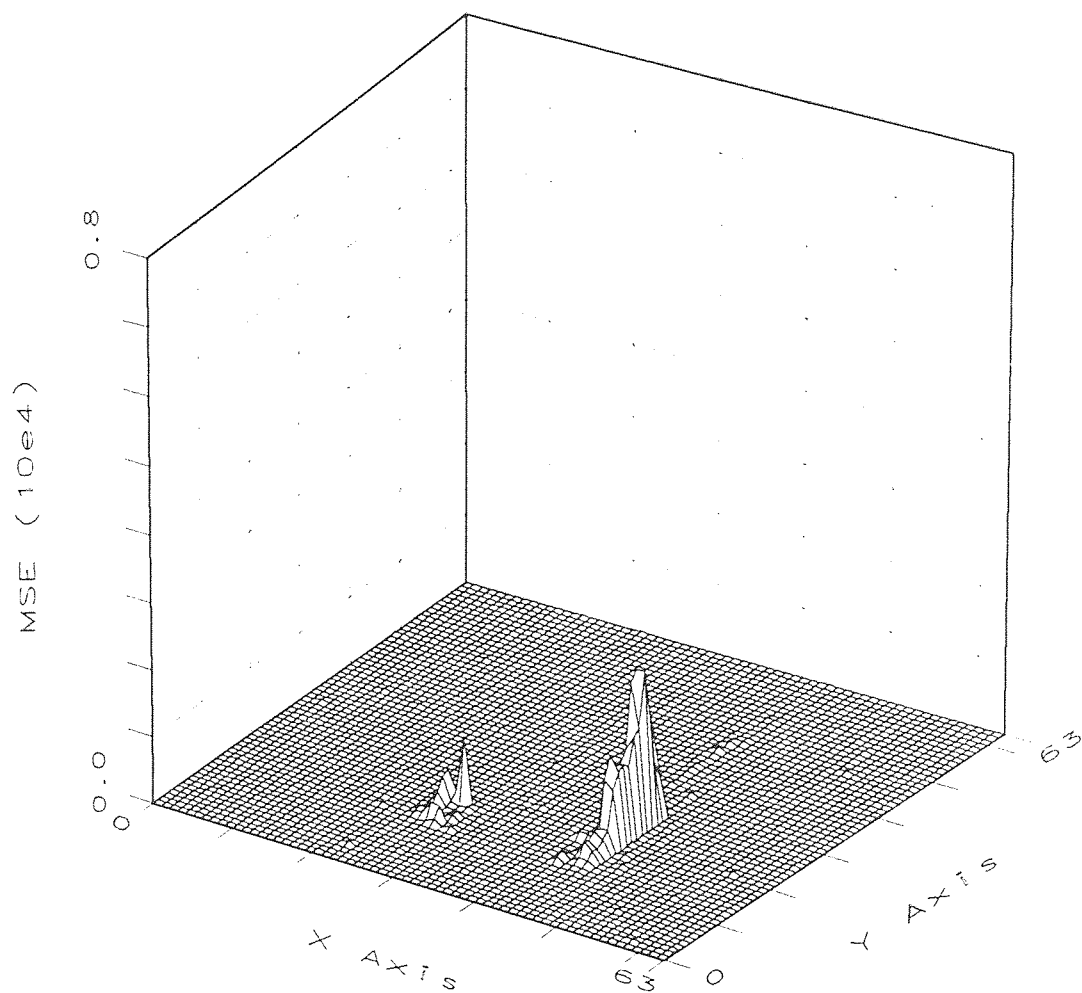**Figure 5.9** Error Squared of Experiment I Calculated by the Gradient-Based Approach

Figure 5.10 Error Squared of Experiment I Calculated by the Correlation-Feedback Approach

5.3.2 Experiment II

In this experiment, more detailed 75×75 image sequences as shown in Figurs 5.11, 5.12, and 5.13 are used. The MSE of the MCFD shows that the correlation-feedback produces 74.5% less error than the gradient-based approach for this image.

Table 5.2 The Comparison of Experiment II

	Gradient-Based Approach	Correlation-Feedback Approach
	$iteration = 120$ $alpha = 5$	$iteration = 15$ $horn - iteration = 30$ $sw = 5 \times 5$ $cw = 3 \times 3$
MSE	211.527	53.883

where

sw = search window

cw = correlation window

$alpha$ = smoothing factor

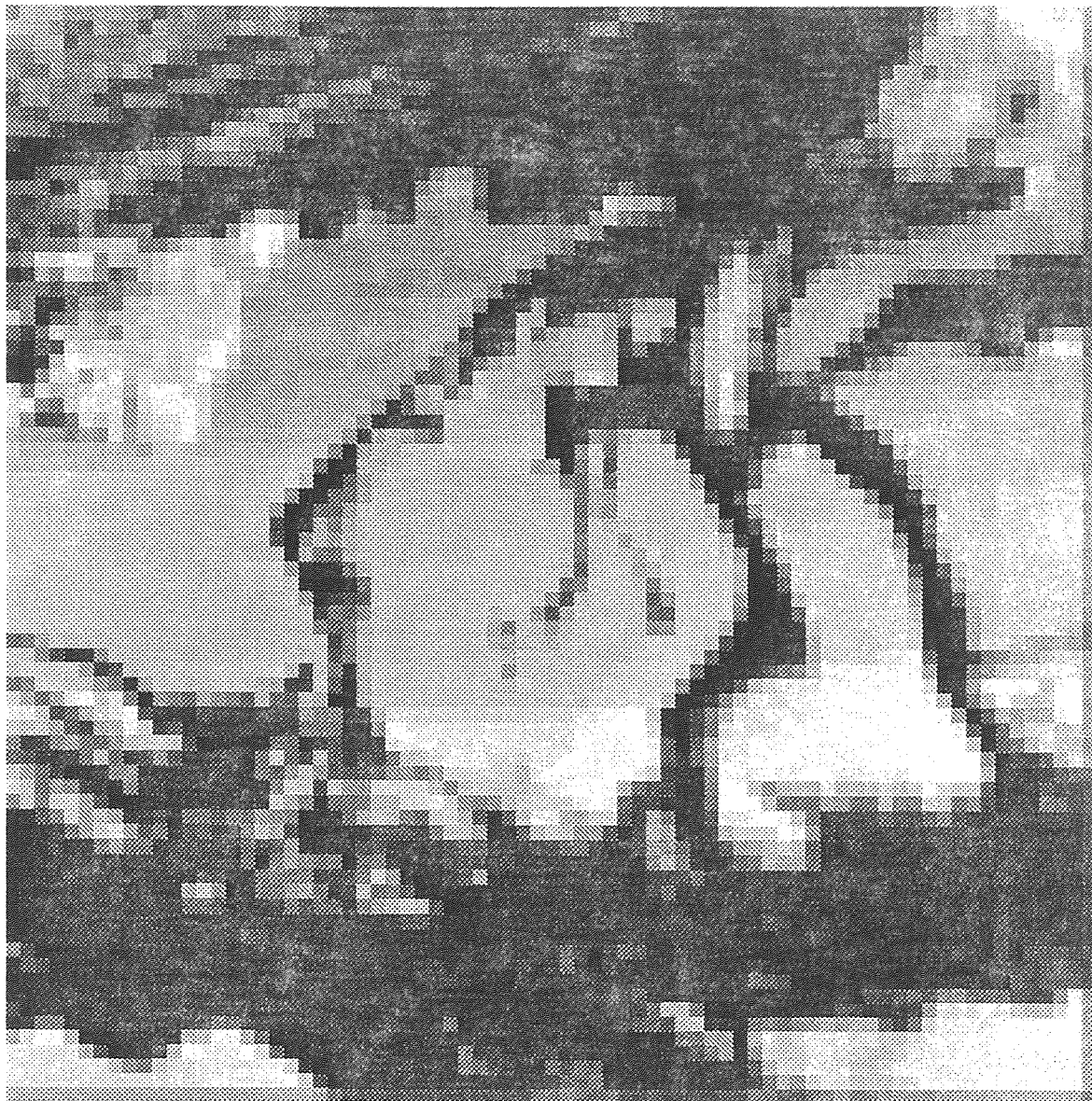Figure 5.11 Image I Used for Experiment II

Figure 5.12 Image II Used for Experiment II
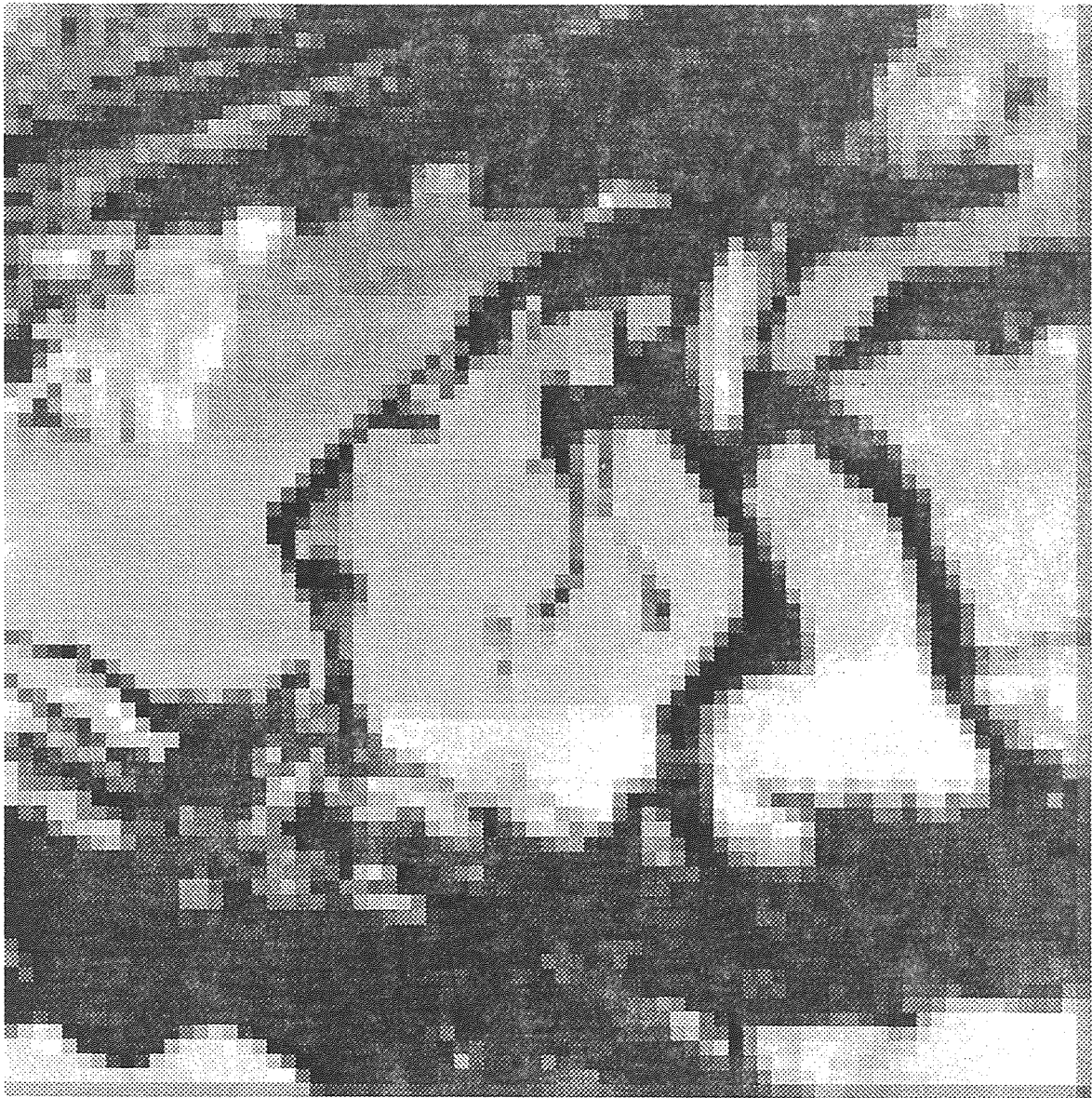
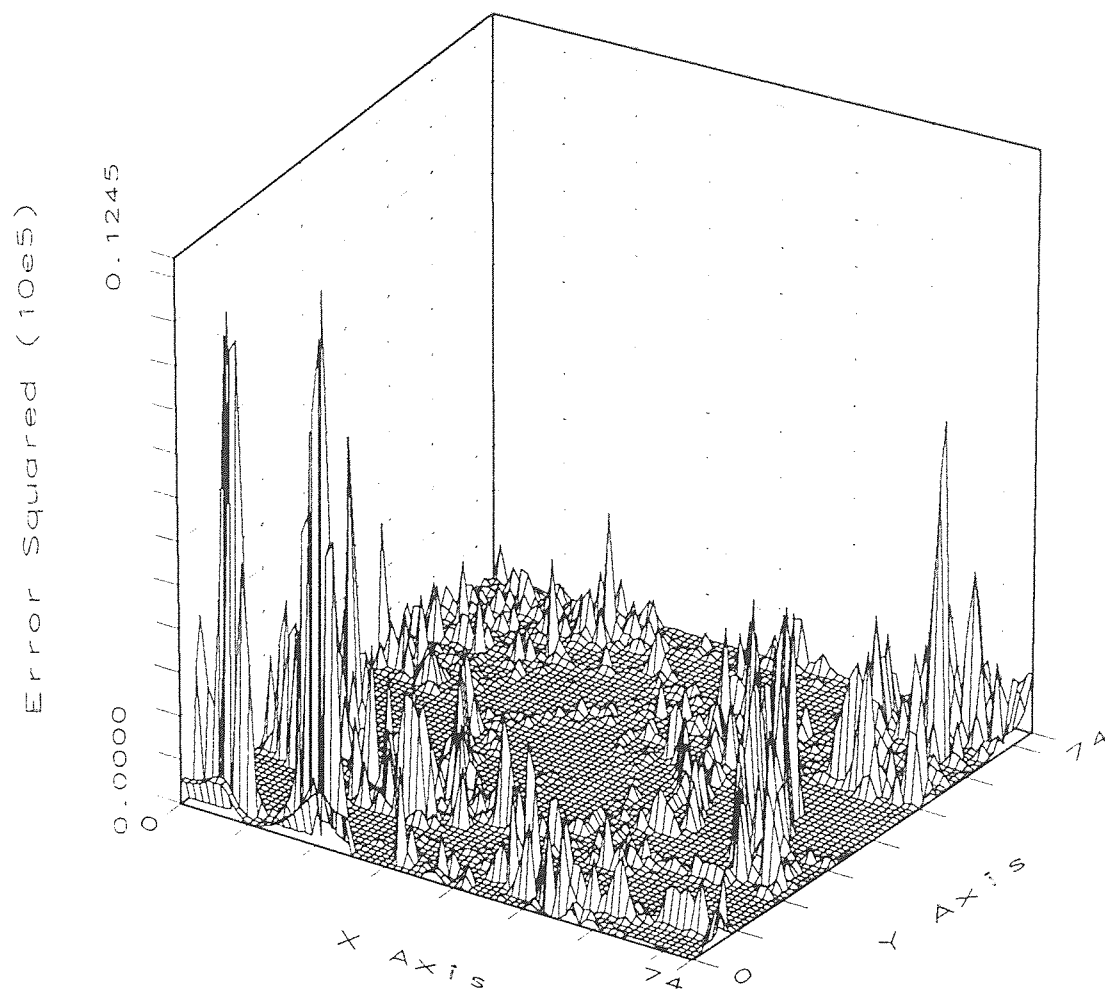Figure 5.13 Image III Used for Experiment II

Figure 5.14 Error Squared of Experiment II Calculated by the Gradient-Based Approach

Figure 5.15 Error Squared of Experiment II Calculated by the Correlation-Feedback Approach

5.3.3 Experiment III

In this experiment, 128 × 128 image sequences as shown in Figures 5.16, 5.17, and 5.18 are used. These image sequences have less detail than the image sequences in experiment II and more detail than the images sequences in experiment I. The MSE of the MCFD shows that the correlation-feedback produces 58.1% less error than the gradient-based approach.

Table 5.3 The Comparison of Experiment III

	Gradient-Based Approach	Correlation-Feedback Approach
	$iteration = 120$ $alpha = 5$	$iteration = 15$ $horn - iteration = 30$ $sw = 5 \times 5$ $cw = 3 \times 3$
MSE	917.74	384.13

where

$$sw \quad = \quad \text{search window}$$

$$cw \quad = \quad \text{correlation window}$$

$$alpha \quad = \quad \text{smoothing factor}$$

Figure 5.16 Image I Used for Experiment III

Figure 5.17 Image II Used for Experiment III

Figure 5.18 Image III Used for Experiment III

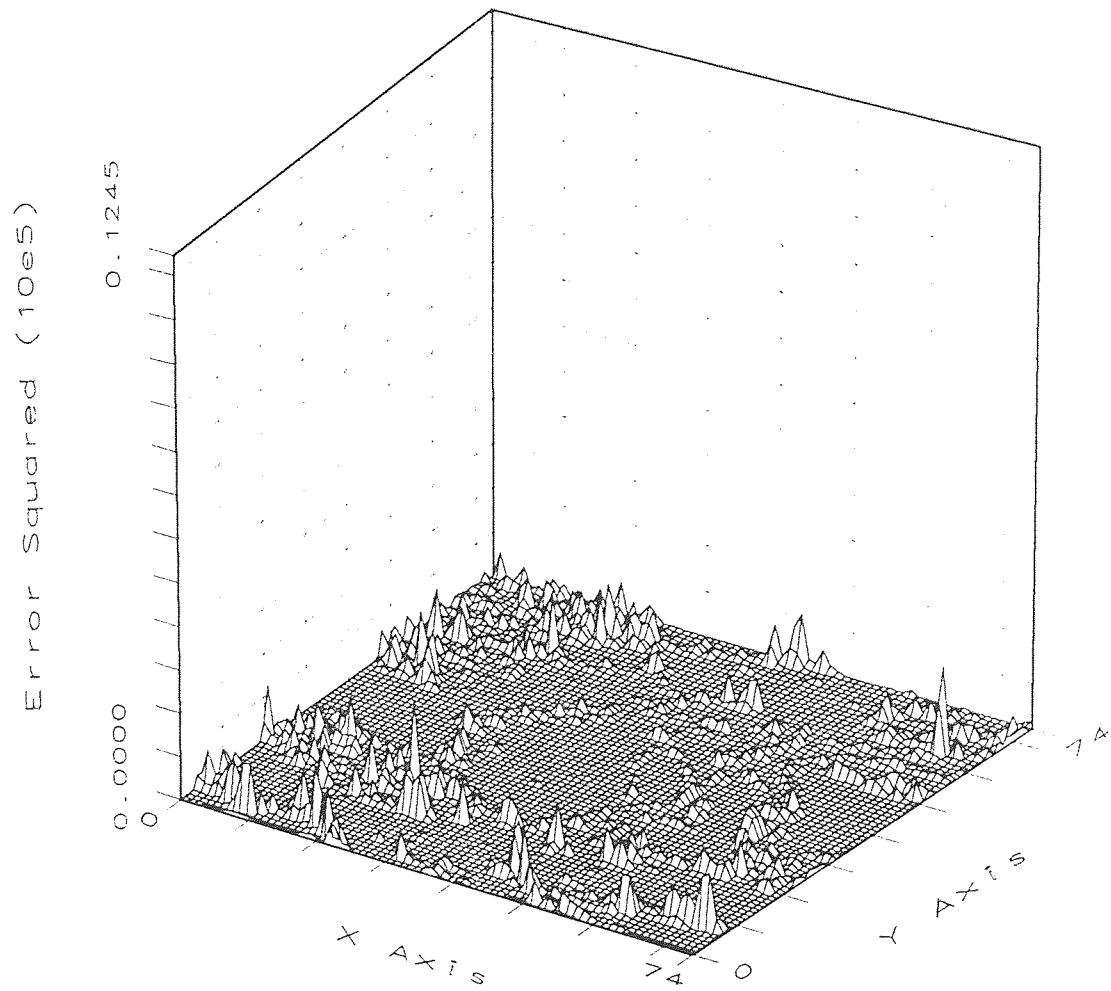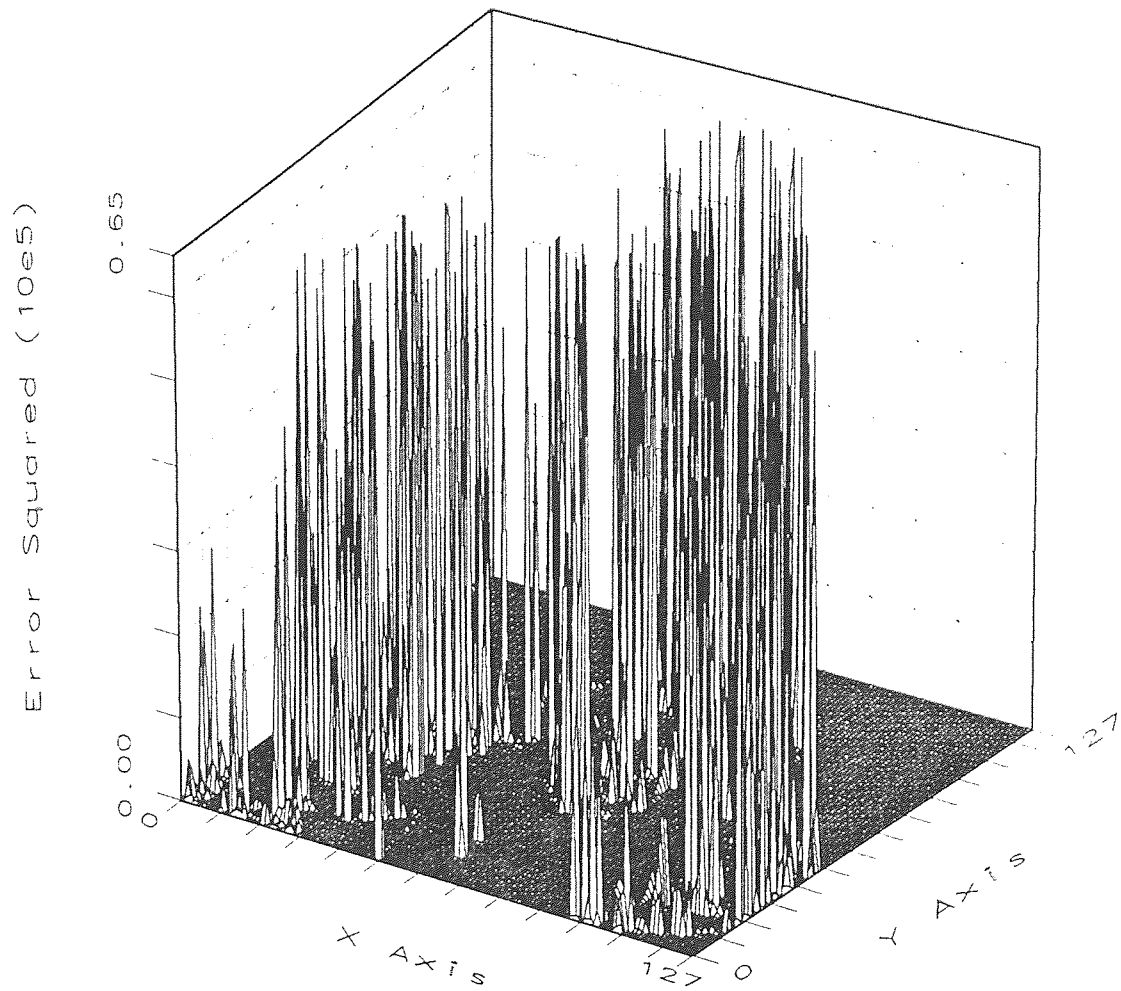**Figure 5.19** Error Squared of Experiment III Calculated by the Gradient-Based Approach

Figure 5.20 Error Squared of Experiment III Calculated by the Correlation-Feedback Approach

5.3.4 Experiment IV

In this experiment, 128 × 128 image sequences as shown in Figures 5.21, 5.22, and 5.23 are used. These image sequences have less detail than the image sequences in experiment II and more detail than the image sequences in experiment I. The MSE of the MCFD shows that the correlation-feedback produces 86% less error than the gradient-based approach.

Table 5.4 The Comparison of Experiment IV

	Gradient-Based Approach	Correlation-Feedback Approach
	$iteration = 120$ $alpha = 5$	$iteration = 15$ $horn - iteration = 30$ $sw = 5 \times 5$ $cw = 3 \times 3$
MSE	333.0	46.34

where

$$sw \quad = \quad \text{search window}$$

$$cw \quad = \quad \text{correlation window}$$

$$alpha \quad = \quad \text{smoothing factor}$$

Figure 5.21 Image I Used for Experiment IV

Figure 5.22 Image II Used for Experiment IV

Figure 5.23 Image III Used for Experiment IV

Figure 5.24 Error Squared for Experiment IV Calculated by the Gradient-Based Approach
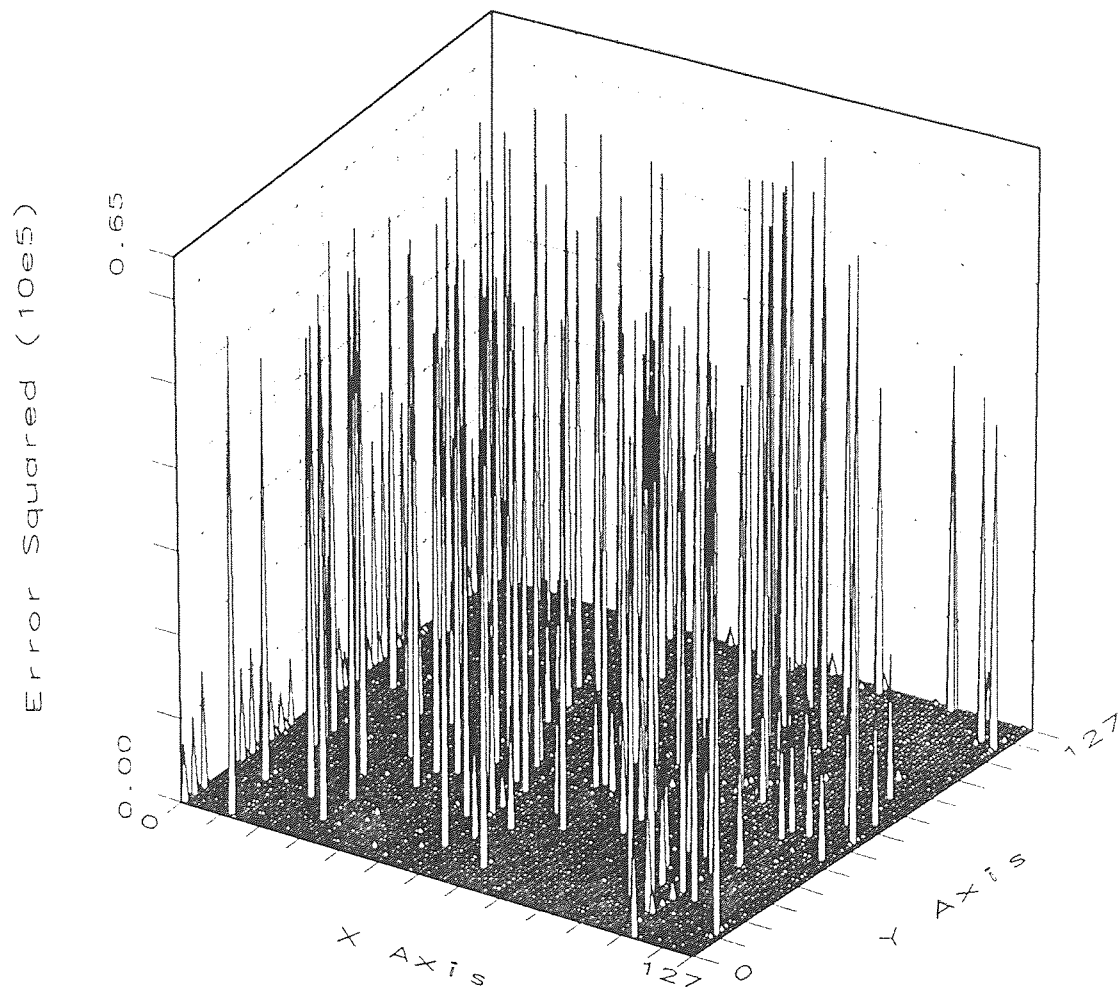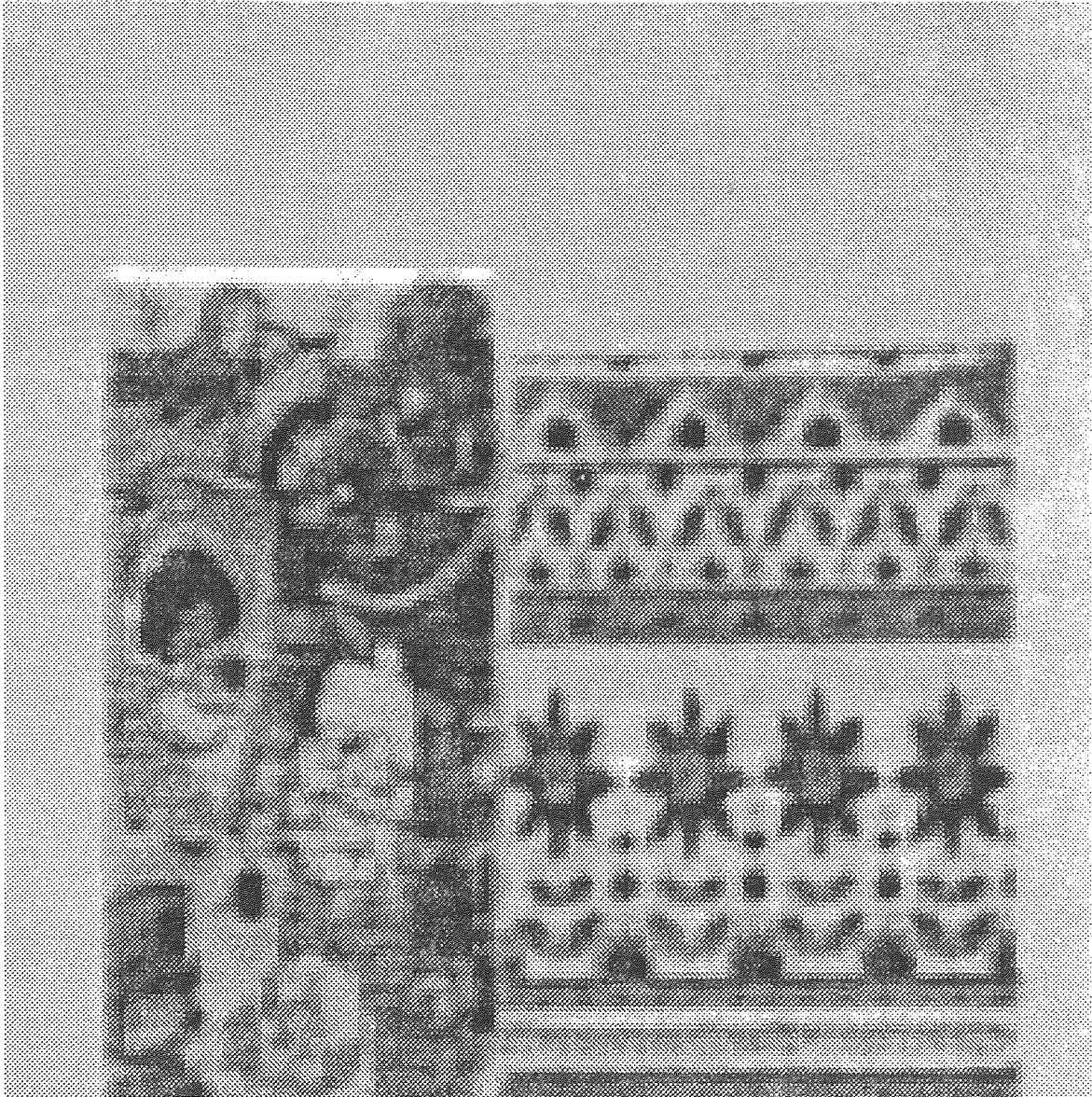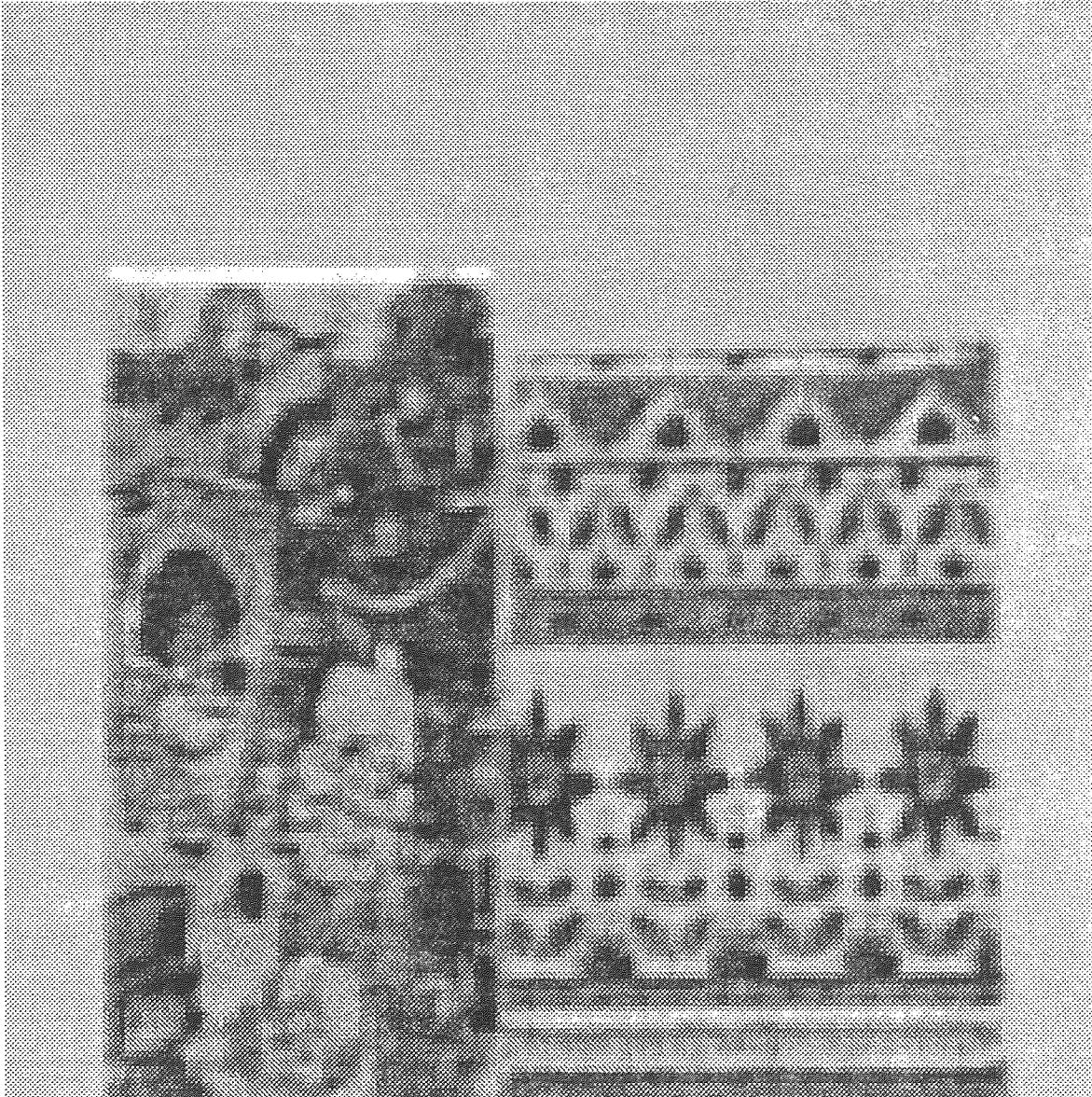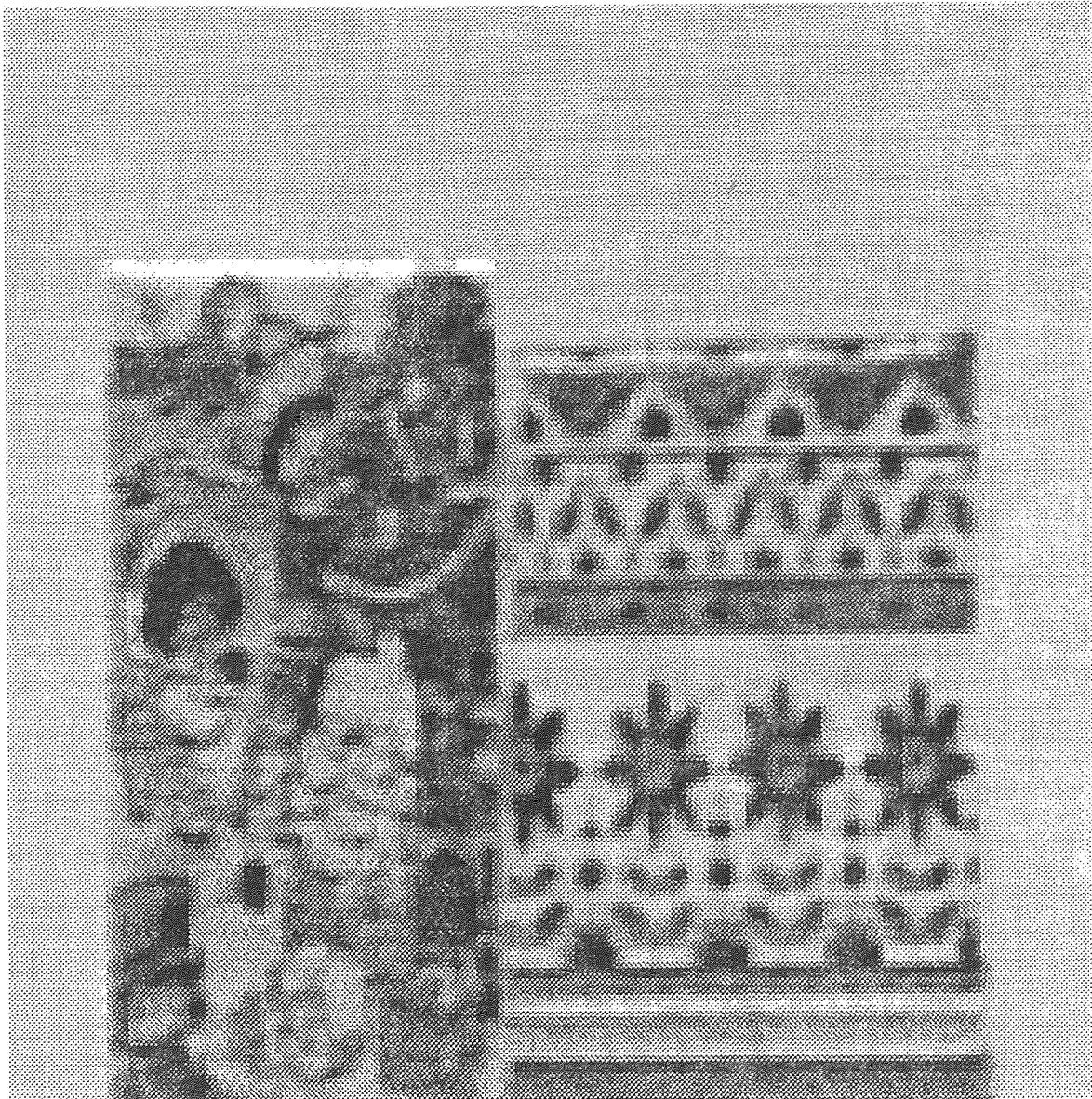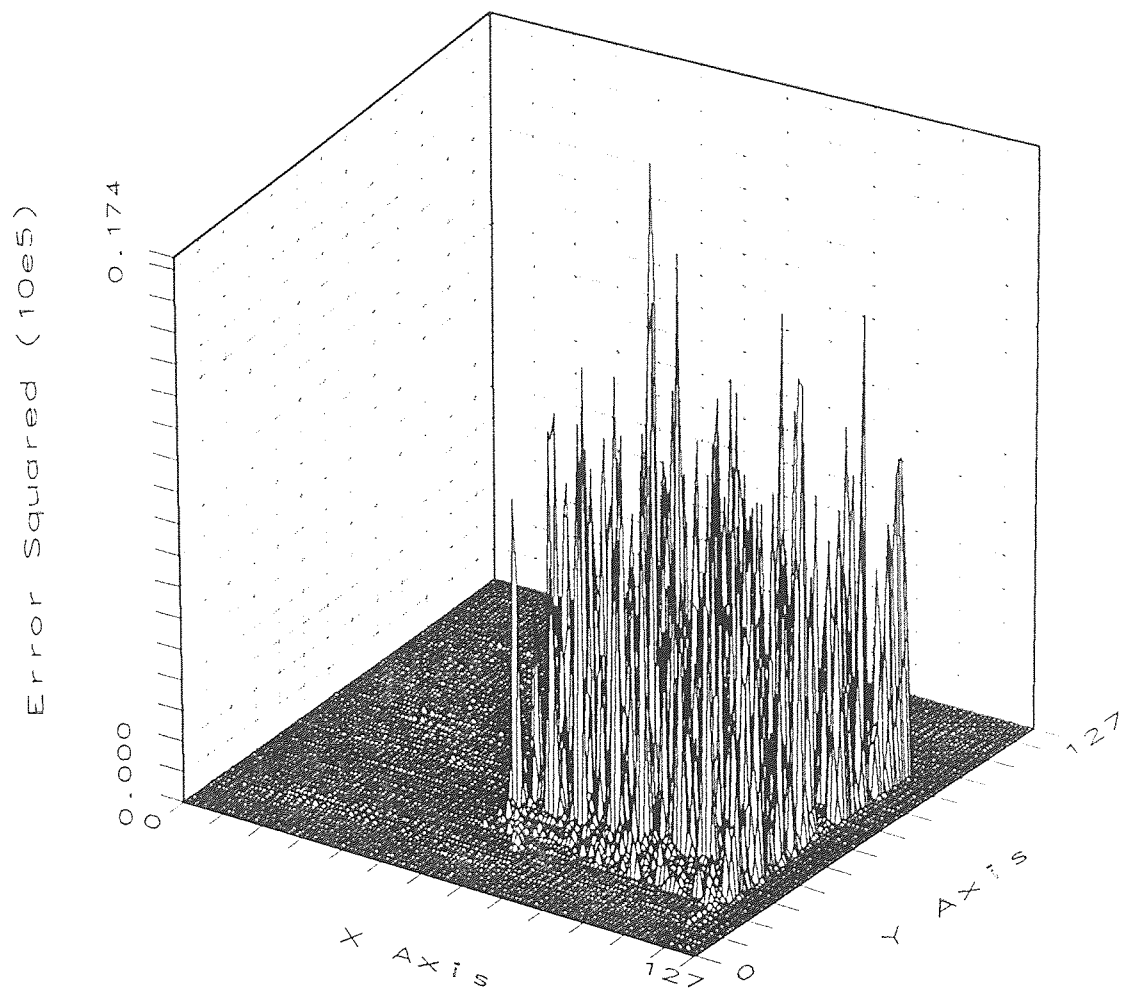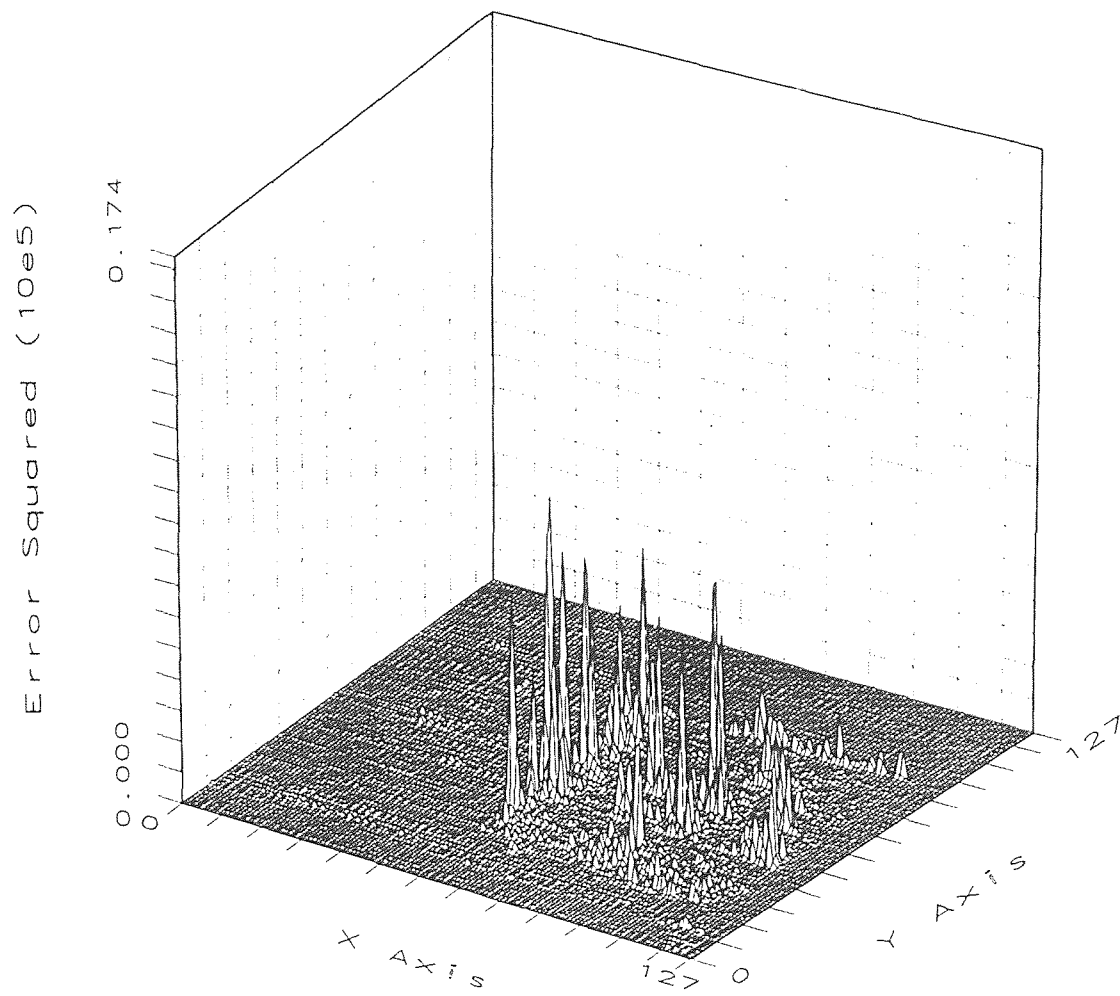
Figure 5.25 Error Squared for Experiment IV Calculated by the Correlation-Feedback Approach

5.3.5 Performance of the Correlation-Feedback

Image sequences varying amounts of detail and different sizes were used to evaluate the performance of the correlation-feedback in the mean square error (MSE). The following table contains the MSE for the correlation-feedback given varying parameters.

Table 5.5 The MSE of Correlation-Feedback

	64×64	128×128	200×200
Less detailed	6.9	46.4	129
More detailed	387.62	1784.35	1904.82

The table shows that the performance of the correlation-feedback for less detail and more detail. The ratio of the MSE of the less detailed images of size 128×128 over 64×64 is **6.7**. The ratio of the MSE of the less detailed images of size 200×200 over 128×128 is **2.8**. The ratio of the MSE of the more detailed images of size 128×128 over 64×64 is **4.6**. The ratio of the MSE of the more detailed images of size 200×200 over 128×128 is **1.1**. From the above, we can conclude that given an image with more detail, correlation-feedback performs well and it is not sensitive to an increase in the size of the image.

5.3.6 Comparison of Correlation-Feedback and Gradient-Based Approaches

To compare the correlation-feedback and gradient-based approaches different varying image sequences with different sizes were used. The following table evaluates the performances of the correlation-feedback and gradient-based approaches. It shows the *reduced* MSE in percentage produced by the correlation-feedback over the MSE produced by the gradient-based approach.

Table 5.6 Reduced MSE (%)

	64×64	128×128	200×200
Less detailed	98%	86%	70%
More detailed	82%	67.2%	58%

We previously noted that the correlation-feedback performs better given images with less detail and more detail. The difference in performance between the correlation-feedback and gradient-based approaches nearly similar over a variety of image sizes and detail. The above table shows that the correlation-feedback offers significantly superior performance ranging from $70\% - 98\%$ *less* MSE than the gradient-based approach given images with less detail. While images with more detail exihibited slightly less performance, $58\% - 82\%$, the correlation-feedback and the gradient-based approach is greatly reduced to a range of $58\% - 82\%$, given images of varying sizes with more detail. Nonetheless, the correlation-feedback consistently delivers superior performance in terms of reduced MSE over the gradient-based approach in a variety of image sizes and details.

5.3.7 Performance Analysis

In this section we compared block matching with correlation-feedback and gradient-based approaches. A less detailed 512×512 Cindy was used in this experiment. The following table shows the performance of the block matching, correlation-feedback, and gradient-based approaches.

Table 5.7 The MSE

	Block Matching	Gradient-Based Approach	Correlation-Feedback Approach
Cindy	15636.3	1351.16	995.19

From the above table we can see that correlation-feedback performs better than the block matching and gradient based approaches given images with less detail image in general.

CHAPTER 6

CONCLUSIONS

In this work we introduced a new method in motion compensation termed correlation-feedback. Correlation-feedback a technique originally developed for motion estimation in computer vision. It proved to yield a better performance demonstrated by reduced mean squared error (MSE) when compared against the gradient-based approach. We therefore introduced this technique in motion compensation because it estimates optic flow more accurately. Thus, the prediction error, which is the difference between the original frame and the predicted frame, is reduced.

When the concept of feedback is used, not only is the effect of noise reduced, but the boundary information is conserved as well. Our computer simulations show that correlation-feedback approach performs well for both less detailed as well as more detailed image sequences. Our approach yields less MSE when compared to the gradient-based approach in motion compensation. We compared the correlation-feedback approach to a traditional motion compensation method, block matching, and proved through simulations that our approach produces a lower MSE.

The correlation-feedback approach introduces less error. Therefore, for future work, we anticipate the further development of correlation-feedback in coding. Feedback technique can make systems robust against noise and improve performance drastically. The concept of feedback can also be applied to block matching for accurate prediction of the frames in motion.

REFERENCES

1. A. Akansu and S. Manjutha, "FSVQ of MCFD signal for video coding," submitted to *Proc. IEEE Int. Conf. Communications.*

2. A. Akansu and S. Manjutha, "Adaptive vector quantization of video signals with motion compensation and spatial masking," Unpublished manuscript, New Jersey Institute of Technology, 1993.

3. P. Anandan, *Measuring Visual Motion from Image Sequences,* PhD thesis, University of Massachusetts, Amherst, 1987.

4. R. Azriel, "Digital picture processing," *Academic Press, Edition,* vol. 2, pp. 33 36, 1982.

5. L. Bede and Z. Andre, "New fast algorithms for the estimation block vectors," *IEEE Trans. Circuits Systems Video Technology,* vol. 4, no. 2, April 1993.

6. B. Bir and B. Wilhelm, "A qualitative approach to dynamic scene understanding," *Image Understanding,* vol. 54, no. 2, pp. 184 205, September 1991.

7. J. Driessen, *Motion Estimation for Digital Video,* PhD thesis, Delft University of Technology, September 1992.

8. H. Edward and R. James, "Spatiotemporal energy models for the perception of motion," *J. Opt. Soc. Am.,* vol. 4, no. 2, pp. 284 298, February 1985.

9. H. Homer, "Determining motion and depth from binocular orthographic views," *Image Understanding,* vol. 54, no. 1, pp. 47 55, July 1991.

10. B. Horn and B. Schunck, "Determining optical flow," *Artificial Intelligence,* vol. 17, pp. 185 203, December 1981.

11. N. Jayant, *Digital Coding of Waveforms Principles and Applications to Speech and Video,* Prentice Hall, Englewood Cliff, New Jersey, 1984.

12. J. Kearney, B. Thompson, and D. Boley, "Optical flow estimation: Error analysis of gradient-based methods with local optimization," *IEEE Trans. Pattern Analysis and Machine Intelligence,* vol. 9, no. 2, March 1987.

13. J. Lim, *Two-Dimensional Signal and Image Processing,* Prentice Hall, Englewood Cliff, New Jersey, 1990.

14. J. Limb and J. Murphy, "Estimating the velocity of moving image in television signals," *Computer Graphics Image Processing,* vol. 4, pp. 311 327, February 1975.

15. T. Michael, "Predictive motion-field segmentation for image sequence coding," *IEEE Trans. Circuits Systems Video Technology*, vol. 3, no. 1, February 1993.

16. A. Netravali and J. Robbins, "Motion-compensated television coding: Part I," *Bell System Tech. J.*, vol. 58, pp. 631–670, March 1979.

17. J. Pan, Y. Shi, and C. Shu, "Correlation-feedback approach to computation of optic flow," Unpublished manuscript, New Jersey Institute of Technology, 1993.

18. A. Rosenfeld and C. Avinash, "Digital picture processing," *Academic Press, Edition*, vol. 2, pp. 33–36, 1982.

19. Y. Shi, C. Shu, and J. Pan, "Unified optical flow field approach to motion analysis," Unpublished manuscript, New Jersey Institute of Technology, 1993.

20. C. Shu and Y. Shi, "On unified optic flow field," *Pattern Recognition*, vol. 24, no. 6, pp. 579–586, June 1991.

21. A. Singh, "An estimation-theoretic frame work for image-flow computation," *Proc. 3rd Int. Conf. Computer Vision*, no. 5, pp. 100–109, December 1990.

22. A. Singh, *Optic Flow Computation a Unified Perspective*, IEEE Computer Society Press, Los Alamitos, California, 1991.

23. S. Vassilis and C. Mohammed, "Generalized block matching motion estimation," University of Essex, Department of Electronic Systems Engineering, 1993.

24. F. Wong, "Sequential hierarchical scene matching," *IEEE Trans. Computers*, vol. 27, no. 4, pp. 359–366, 1978.