

IMAGE CORRELATION TECHNIQUE FOR RECOVERING  
DEFORMATION FIELDS FROM PICTURES

BY

ANDRZEJ K. MAZUR

THESIS

SUBMITTED TO THE FACULTY OF GRADUATE STUDIES  
IN PARTIAL FULFILLMENT OF THE REQUIREMENTS  
FOR THE DEGREE OF

DOCTOR OF PHILOSOPHY

DEPARTMENT OF ELECTRICAL AND COMPUTER ENGINEERING

UNIVERSITY OF MANITOBA

WINNIPEG, MANITOBA

COPYRIGHT © 1992 by ANDRZEJ K. MAZUR

**IMAGE CORRELATION TECHNIQUE FOR RECOVERING  
DEFORMATION FIELDS FROM PICTURES**

**BY**

**ANDRZEJ K. MAZUR**

A Thesis submitted to the Faculty of Graduate Studies of the University of Manitoba in partial fulfillment of the requirements for the degree of

**DOCTOR OF PHILOSOPHY**

© 1992

Permission has been granted to the LIBRARY OF THE UNIVERSITY OF MANITOBA to lend or sell copies of this thesis, to the NATIONAL LIBRARY OF CANADA to microfilm this thesis and to lend or sell copies of the film, and UNIVERSITY MICROFILMS to publish an abstract of this thesis.

The author reserves other publication rights, and neither the thesis nor extensive extracts from it may be printed or otherwise reproduced without the author's permission.

## ABSTRACT

It is a difficult task to determine strain distribution from images. In engineering, there have been some attempts to use images to determine strains in solids. Only relatively simple cases were considered. The problem, itself, is demanding and challenging and belongs to the most difficult in image processing. It has been solved in this thesis by introducing an original method, Image Correlation Technique, that matches two whole images of the same object, undistorted and distorted, pixel for pixel, in order to determine the deformation to which the object was subjected. The method makes use of simulated annealing optimization technique and works equally well for small and large deformations, and for both linear and non-linear cases. Shown here are the positive results of using the method on model objects subjected to simulated deformations.

There is no need for control points. The method is the first one that attempts, with success, to process and actually to relate images/pictures as a whole. It can be easily generalized to include three-dimension images.

The method opens new possibilities in the image recognition and in other connected fields of artificial intelligence, the most difficult problems.

The method can find an immediate application in radiology as the major constituent of a diagnostic procedure, that has been also suggested, Differential Radiography Technique, for the automatic detection of soft tissue abnormalities (growing masses or tumors).

## ACKNOWLEDGMENTS

To Professor Richard Gordon I owe gratitude for inspiring me with the idea of automation of the comparative diagnosis in radiology. He granted me total freedom during the formulation and the solution stage of this work. This left me alone and insecure, but finally it gave me success, and I thank him for his wisdom.

I have much to be thankful to the Dean of Graduate Studies, Dr. Kenneth Hughes, for his support and confidence in me.

I am grateful to my committee members, Drs. Edward A. Lyons and Al Wexler, for their suggestions and encouragement.

I have additional thanks to Dr. Edward A. Lyons for giving me the opportunity to use faster, Siemens LiteBox computer. It helped greatly in the final preparation of the thesis.

I would like to thank Frederick Séguin and Paul Bjorkholm for allowing us to use the Micro-CT scanner that was developed at American Science and Engineering, Inc., Boston, MA.

Thanks are also extended to Dr. Neil Crowson for providing mastectomy samples, to Nikola Boner and Donald Elcheshen for their help in providing and testing animal tissue.

The great majority of articles for Section 7 were collected with the active help of their authors: F. Bookstein, U. Grenander, and D. Terzopoulos. Based on their expertise, they called my attention to a particular set of publications. Moreover, they supplied me with physical copies and reprints using fax and fast delivery. I am grateful for their generous help.

I have special words for my wife, Elzbieta. During this long struggle it was she who lifted my spirit, many times. She provided an opportunity for discussion and helped me to focus on important aspects of the research. Her extensive experience in programming

helped me to shorten significantly my learning period. Finally, the work itself was challenging and demanding, introducing a lot of stress in our private life. At the same time she was also exposed to the same type of conditions doing her own Ph.D. I have to credit her for overcoming all of these.

This work was supported in part by a scholarship from the Natural Sciences and Engineering Research Council of Canada.

# CONTENTS

ABSTRACT .....	II
ACKNOWLEDGMENTS.....	III
CONTENTS .....	V
LIST OF FIGURES .....	VIII
LIST OF SYMBOLS .....	XIII
1. INTRODUCTION .....	1
2. IMAGE MATCHING	
2.1. Introduction.....	4
2.2. Correlation Function.....	5
2.3. Image Matching Using Location .....	9
2.4. Digital Correlation Technique.....	12
3. SIMULATED ANNEALING OPTIMIZATION TECHNIQUE	
3.1. Introduction.....	15
3.2. Formulation of Simulated Annealing .....	15
3.3. Solution and Convergence .....	18
3.4. Cooling Schedule .....	25
4. IMAGE CORRELATION TECHNIQUE: A NEW PROPOSAL FOR IMAGE MATCHING	
4.1. Introduction.....	29
4.2. Correlation Function.....	30
4.3. Movement Algorithm.....	35

4.4. Cooling Schedule .....	38
4.5. Discussion .....	40
4.5.1. Optimization .....	40
4.5.2. Image Grid Movement.....	41
4.5.3. Interpolation .....	58
<b>5. EXPERIMENTS</b>	
5.1. Experiment Descriptions .....	60
5.2. Computer implementation of ICT .....	63
5.3. Experiment 1.....	69
5.4. Experiment 2.....	84
5.5. Experiment 3.....	98
5.6. Experiment 4.....	110
5.7. Discussion .....	122
<b>6. DIGITAL DIFFERENTIAL RADIOGRAPHY</b>	
6.1. Introduction.....	126
6.2. Method Formulation .....	127
6.3. Model of Growing Mass .....	129
6.4. Composite Movement Algorithm.....	138
6.5. Experiment 5.....	147
6.6. Experiment 6.....	155
6.7. Discussion .....	167
<b>7. FINAL REMARKS</b>	
7.1. Conclusions	
7.1.1. Discussion on Convergence .....	168
7.1.2. Control Parameters.....	172
7.1.3. Noise .....	172

7.1.4. Computer Implementation.....	174
7.2. Image Correlation Technique Applications.....	176
7.3. Biological Atlases and Other Related Methods .....	177
8. BIBLIOGRAPHY .....	185

## LIST OF FIGURES

2.2-1. Correlation function: (a), calculated using (2.2-5) and (b), calculated using (2.2-7).....	7
2.3-1. Matching, registration and noise constraints .....	11
4.2-1. Correlation function (4.2-1) for matching images given by (4.2-2) with $I^*[1,1]=100$ . Region designated by horizontal arrows is a decreasing range of correlation on the right side to the point at $I^*[1,1]=100$ , where the match is perfect. The extension of the decreasing range is 20% of the value of the matching point. Beyond that range the correlation increases very slowly. ....	32
4.2-2. Correlation function (4.2-1) for matching images given by (4.2-2). Each curve is for a different value of parameter $I^*[1,1]$ .....	33
4.2-3. Correlation function for matching two images (2.2-8): (a), calculated using (4.2-3) with $A = 50$ , (b), calculated using (2.2-7). ....	34
4.5-1. The optimization process driven by the correlation function. An arrow in the right upper corner indicates the direction of the temperature-like parameter change. Probability (4.3-6) provides a means to overcome local minima with the depth $-\Delta C$ .....	41
4.5-2. A schematic movement for a particular pixel cell. Dashed line represents the direction chosen at random ( $\Delta i = 1, \Delta j = 1$ ), the solid line reflects the 'correction' imposed by means of parameters $R_x$ and $R_y$ . Note also that as long as $R_x$ and $R_y$ are real numbers from the open interval (0,1), all points within the quadrilateral are accessible. ....	42
4.5-3. The image movement induced by the movement algorithm. Pixels from the reference image are represented by the grid and pixel centers are in the intersections of the lines. The movement starts from the undisturbed configuration at the image location $i = 5, j = 5$ (seed pixel) in the direction $\Delta i = 1, \Delta j = 1$ from the center of the image towards the right side and down, with $R_x = 1, R_y = 0.5, F = 0$ and the flexibility parameter $R_f$ fixed at different levels. The greater the flexibility parameter the more rigid the grid becomes. In a real application run the variable $R_f$ and the factor $F$ that enter the denominator in Eq (4.3-3), together, ensure the flexibility of the optimized picture grid. Factor $F$ plays role as the flexibility limitation. It sets flexibility upper limit.....	45
4.5-4. (a) - (l), A sequence of movements applied to a 30 x 30 pixel image. Pixels from the reference image are represented by the grid and the deformed picture of a human breast CT sample is shown in the background. The reference grid from an undisturbed image is moved following the sequence of moves generated by the placement algorithm.....	46
4.5-5. The intensity interpolation. The image Correlation Technique adopts a simple way of the construction of the optimized image (temporary image) that is created at each iteration step. Cells of the optimized image grid that do not coincide with centers of the deformed pixel grid are left with the intensity zero (white cells). Others, that intersect one or more pixel centers acquire the average intensity of all interacting pixels (grey cells). The optimized image is then matched with the deformed one (correlation is calculated).....	58

E.1-1. The <i>hill</i> object under investigation. In the upper part is the undeformed image (reference configuration).....	72
E.1-2. Distorted by (5.3-1), the reference image grid is superimposed onto the distorted image. ....	73
E.1-3. The solution image. The reference image represented by the grid, matched with the deformed picture presented in gray levels.....	74
E.1-4a. The <i>hill</i> displacement field simulated by Eq. (5.3-1) in the <i>x</i> direction.....	75
E.1-4b. The <i>hill</i> displacement field simulated by Eq. (5.3-1) in the <i>y</i> direction. ....	76
E.1-5a. The displacement field recovered from Image Correlation Technique in the <i>x</i> direction. ....	77
E.1-5b. The displacement field recovered from Image Correlation Technique in the <i>y</i> direction.....	78
E.1-6a. The difference between the simulated displacement and that recovered from Image Correlation Technique, in the <i>x</i> direction.....	79
E.1-6b. The difference between the simulated displacement and that recovered from Image Correlation Technique, in the <i>y</i> direction.....	80
E.1-7. Image optimization for the hill model. The correlation and the difference versus the temperature-like parameter <i>T</i> .....	81
E.1-8. Image optimization for the hill model. The correlation and the difference versus iteration. ....	81
E.1-9. Image optimization for the hill model. The correlation and the difference versus the optimization time.....	82
E.1-10. Image optimization for the hill model. The acceptance ratio and the average computation time (elapsed between two consecutive , accepted trials) versus iteration .....	82
E.1-11. Image optimization for the hill model. Image optimization functions for different fixed numbers of accepted steps (CT ) that are required at each "temperature" level. The correlation versus the temperature-like parameter <i>T</i> . ....	83
E.1-12. Image optimization for the hill model. Image optimization functions for different fixed numbers of accepted steps (CT ) that are required at each "temperature" level. The temperature-like parameter <i>T</i> versus computing time. ....	83
E.2-1. The <i>face</i> under optimization. In the upper part is the undeformed image (reference configuration), in the lower part the distorted image using Eq. (5.3-1). (These are best seen viewed from a distance.).....	86
E.2-2. Distorted by (5.3-1), the reference image grid is superimposed onto the distorted image. ....	87
E.2-3a. The <i>face</i> displacement field simulated by Eq. (5.3-1) in the <i>x</i> direction. ....	88
E.2-3b. The <i>face</i> displacement field simulated by Eq. (5.3-1) in the <i>y</i> direction.....	89
E.2-4a. The displacement field recovered from Image Correlation Technique in the <i>x</i> direction. ....	90
E.2-4b. The displacement field recovered from Image Correlation Technique in the <i>y</i> direction.....	91
E.2-5a. The difference between the simulated displacement and that recovered from Image Correlation Technique, in the <i>x</i> direction.....	92

E.2-5b. The difference between the simulated displacement and that recovered from Image Correlation Technique, in the y direction.....	93
E.2-6. Image optimization for the face. The correlation and the difference versus the computing time.....	94
E.2-7. Image optimization for the <i>face</i> . The acceptance ratio and the average computation time (elapsed between two consecutive and accepted trials) versus the computing time.....	94
E.2-8. Image optimization for the <i>face</i> . The temperature-like parameter $T$ versus the computing time.....	95
E.2-9. The <i>face</i> on a 32 by 32 pixel matrix. In the upper part is the undistorted image (reference configuration), in the lower part the distorted image using Eq. (5.3-1).....	96
E.2-10. The <i>face</i> under investigation but on a 32 by 32 pixel matrix. The solution image: the reference image grid matched with the deformed picture.....	97
E.3-1. The <i>face</i> . In the upper part is the undeformed image (reference configuration), in the lower part the distorted image using Eq. (5.5-1).....	100
E.3-2. The reference image grid distorted by (5.5-1).....	101
E.3-3. The <i>face</i> . The reference image grid matched with the deformed picture.....	102
E.3-4a. The <i>face</i> displacement field simulated by Eq. (5.5-1) in the x direction.....	103
E.3-4b. The <i>face</i> displacement field simulated by Eq. (5.5-1) in the y direction.....	104
E.3-5a. The displacement field recovered from Image Correlation Technique in the x direction. ....	105
E.3-5b. The displacement field recovered from Image Correlation Technique in the y direction.....	106
E.3-6a. The difference between simulated displacement and that recovered from Image Correlation Technique, in the x direction.....	107
E.3-6b. The difference between simulated displacement and that recovered from Image Correlation Technique, in the y direction.....	108
E.3-7. Image optimization for the <i>face</i> . The correlation and the difference versus the computing time.....	109
E.3-8. Image optimization for the <i>face</i> . The acceptance ratio and the average computing time (elapsed between two consecutive, accepted trials) versus the computing time. ....	109
E.3-9. Image optimization for the <i>face</i> . The temperature-like parameter $T$ versus the computing time.....	110
E.4-1. Breast tissue sample from a CT scanner. In the upper part is the reference configuration, in the lower part the distorted image using Eq. (5.6-1).....	113
E.4-2. The reference image grid after distortion. Dots represent border pixels.....	114
E.4-3a. The <i>breast</i> displacement field simulated by Eq. (5.6-1) in the x direction. ....	115
E.4-3b. The <i>breast</i> displacement field simulated by Eq. (5.6-1) in the y direction.....	116

E.4-4a. The displacement field recovered from Image Correlation Technique in the $x$ direction. ....	117
E.4-4b. The displacement field recovered from Image Correlation Technique in the $y$ direction.....	118
E.4-5a. The difference between simulated displacement and that recovered from Image Correlation Technique, in the $x$ direction.....	119
E.4-5b. The difference between simulated displacement and that recovered from Image Correlation Technique, in the $y$ direction.....	120
E.4-6. Image optimization for the <i>breast</i> . The correlation and the difference versus the computing time.....	121
E.4-7. Image optimization for the <i>breast</i> . The acceptance ratio and the average computation time (elapsed between two consecutive and accepted trials) versus the computing time.....	121
6.2-1. Digital Differential Radiography flow chart. ....	128
6.3-1. A high resolution CT image of a mastectomy sample. ....	131
6.3-2. A CT image of a mastectomy sample obtained by shrinking the image from Fig. 6.3-1, by a factor of 10 to produce a reference picture of the size 30 x 30 pixels. ....	132
6.3-3. The reference image after the simulated distortion using Eq. (6.3-1). The pixels opened up in the middle due to the distortion have been assigned a unique gray level to represent a tumor.....	133
6.3-4. The appearance of the reference image pixel grid after the distortion (6.3-1).....	134
6.3-5a. The displacement field simulated by Eq. (6.3-1) in the $x$ direction. ....	135
6.3-5b. The displacement field simulated by Eq. (6.3-1) in the $y$ direction.....	136
6.3-6a. The divergence field from the Digital Differential Radiography technique obtained for the displacement fields simulated by Eq. (6.3-1). Wire frame chart.....	137
6.3-6b. The divergence field from the Digital Differential Radiography technique obtained for the displacement fields simulated by Eq. (6.3-1). Contour chart. ....	137
6.4-1. (a) - (c), A sequence of movements applied to a 30 x 30 pixel image. The reference grid from an undisturbed image is moved following the sequence of moves generated by the movement algorithm according to Eq. 6.4-1.....	140
6.4-2. (a) - (d), A sequence of movements applied to a 30 x 30 pixel image. The reference grid from an undisturbed image is moved following the sequence of moves generated by the Composite Movement Algorithm according to Eqs. 4.3-2 and 6.4-1 with $P_c = 0.05$ .....	143
E.5-1a. The displacement field recovered from Image Correlation Technique in the $x$ direction.....	149
E.5-1b. The displacement field recovered from Image Correlation Technique in the $y$ direction.....	150
E.5-2a. The difference between simulated displacement and that recovered from Image Correlation Technique, in the $x$ direction.....	151
E.5-2b. The difference between simulated and recovered from Image Correlation Technique displacement, in the $y$ direction.....	152

E.5-3a. The divergence field from the Digital Differential Radiography technique obtained for the displacement fields recovered by the Image Correlation Technique. Wire frame chart.....	153
E.5-3b. The divergence field from the Digital Differential Radiography technique obtained for the displacement fields recovered by the Image Correlation Technique. Contour chart.....	153
E.5-4. Image optimization. The correlation and the difference functions versus the computing time.....	154
E.5-5. Image optimization. The acceptance ratio and the average time (elapsed between two consecutive and accepted trials) versus the computing time.....	154
E.5-6. Image optimization. "Temperature" of the system as a function of the computing time.....	155
E.6-1a. The displacement field recovered from Image Correlation Technique in the $x$ direction.....	157
E.6-1b. The displacement field recovered from Image Correlation Technique in the $y$ direction.....	158
E.6-2a. The difference between simulated displacement and that recovered from Image Correlation Technique, in the $x$ direction.....	159
E.6-2b. The difference between simulated displacement and that recovered from Image Correlation Technique, in the $y$ direction.....	160
E.6-3a. The divergence field from the Digital Differential Radiography technique obtained for the displacement fields recovered by Image Correlation Technique. Wire frame chart.....	161
E.6-3b. The divergence field from the Digital Differential Radiography technique obtained for the displacement fields recovered by Image Correlation Technique. Contour chart.....	161
E.6-4. Image optimization. The correlation versus the temperature-like parameter $T$ .....	162
E.6-5. Image optimization. The acceptance ratio versus the temperature-like parameter $T$ .....	163
E.6-6. Image optimization. The averaged time (elapsed between two consecutive and accepted trials) versus the temperature-like parameter $T$ .....	164
E.6-7. Image optimization. 'Temperature' of the system as a function of the computing time.....	165
E.6-8. Experiment 6. The solution image.....	166
7.1-1. One dimensional curve matching.....	169
7.1-2. Correlation function (4.2-1) for matching images given by (4.2-2) with $I^*[1,1]=100$ . The enhanced dynamics of the correlation function occurs for the range that extends only to about 5% on both sides of the perfect match value ( $I^*[1,1]$ ).....	170
7.1-3. Floating point operations for some present and in development computers.....	175

## LIST OF SYMBOLS

$A_{ij}$	acceptance probability
$C$	correlation function
$c$	temperature like control parameter
$c_0$	initial value of the control parameter
$C_T$	iteration constant. Number of steps required at each temperature level
$D$	displacement field $D(u,v,w)$
$\delta$	delta function such as: $\delta(m,n) = \begin{cases} 1 & \text{if } m,n = 0 \\ 0 & \text{if otherwise} \end{cases}$
$F$	
$f$	cost function
$f_{opt}$	optimal cost
$G_{ij}$	generation probability
$I[m,n]$	image intensity function
$i,j,m,n$	image or Lagrange coordinates such as: $i,j,m,n \in \{1,\dots,M\}$
$k$	iteration step
$\chi$	characteristic function. Let $A$ and $A' \subset A$ be two sets. Then the function $\chi_{(A')} : A \rightarrow \{0,1\}$ of the set $A'$ is defined as $\chi_{(A')}(a) = \begin{cases} 1 & \text{if } a \in A' \\ 0 & \text{if } a \in A \setminus A' \end{cases}$
$k_b$	Boltzmann constant
$L$	Markov chain length

$\lambda$	acceptance ratio
$\lambda_0$	initial acceptance ratio
$M$	size of an image of $M \times M$ pixels
$P$	probability
$P\{A B\}$	probability of event <b>A</b> given event <b>B</b>
$P_c$	partition control
$P_{ij}$	transition probability
$q$	stationary distribution
$q^*$	stationary distribution at $c=0$
$S$	solution space
$\sigma$	standard deviation
$S_i$	set of neighbour solutions
$S_{opt}$	set of optimal solutions
$\theta$	neighborhood size
$u, v, w$	displacement functions of $x, y$ and $z$ directions,
$x, y, z$	Cartesian coordinates
$X_I[m, n], Y_I[m, n]$	Cartesian coordinates of a pixel with image coordinates $(m, n)$ and intensity $I$

## CHAPTER 1

### 1.1. INTRODUCTION

This work solves the problem of image matching for images that are connected by an unknown deformation. It shows that a successful match is possible by using only the gray level intensity distribution throughout the image plane.

A new method, the Image Correlation Technique (ICT), has been developed. It makes use of the simulated annealing optimization technique. ICT is capable of distinguishing and recognizing fields of large and non-linear deformations from the deformation recorded in a series of image frames.

There are many other methods that utilize optical means to measure strains. Most of them rely on the interferometry technique, holography or speckle photography. The interferometry methods work well for deformation cases where displacements are comparable to the optical wavelength or its low multiple.

The Digital Correlation Technique (DCT) uses correlation. It can be applied to investigate the surface strains of solids of order of 0.01 [Chu, 1985].

Among medical applications Digital Subtraction Mammography is an example of a method that correlates pictures of a human breast that are taken some time apart. This method exists only in the development stage with some preliminary results [Zhou, 1991].

It is intended to apply ICT in the field of medical radiography as a means to correlate two-dimensional (2D), or three-dimensional (3D) human body images. A far reaching goal could be to measure the mechanical properties of human tissues under the action of stress.

One of the possible applications could be breast screening. The most simple, effective, and highly recommended procedure to detect changes and abnormalities in a woman's breast is to examine the breast by hand. The entire population of women is encouraged to do so by themselves. This is a successful method, since Breast Self Examination (BSE) in most cases (in 90% of detected cases) is the first line of detection of breast cancer [Zhou, 1989]. It relies on feeling by hand the differences and changes of constitutive properties of tissues of the breast. However, no one has tried to employ the mechanical properties of breast abnormalities in quantitative diagnosis. By using medical imaging modalities, it is probably possible to map a strain field of the breast under a known external source of stress and by solving the reverse problem arrive at the mechanical properties of the breast. This may establish a new alternative to BSE, a diagnostic procedure that could be called "mechanical imaging". The problem, however, requires knowledge of the stress field, and it is an extremely difficult task to measure the stress distribution while deforming the breast.

The solution of the reverse problem adopts the finite element technique, designed mainly for direct problems in engineering that involves comparatively small linear strains and known, uniform mechanical properties. In addition, there is only an experimental approximation of the elasticity equation [Fung, 1972a], [Fung, 1972b].

There have been few attempts to reconstruct the distribution of mechanical properties for the living tissues. Ray [Ray, 1980], [Ray, 1979a], [Ray, 1979b] used ultrasound data to establish the position of heart valves and blood pressure to estimate the strength, distribution and changes of stress action. He found a unique and interesting relation between the disease progress and the mechanical properties of the heart valves tissue.

The ICT method this thesis proposes leaves behind all the concerns and limitations associated with the non-linear, non-elastic and viscous mechanics of living tissues.

Initially this thesis work was intended to follow the path outlined above and some of it (not reported here) was done. This includes writing a computer code for the finite element implementation.

However, as will be shown, the displacement field derived independently from using stress distribution data, will itself provide very interesting information on the deformed object. The new Digital Differential Radiography Technique, introduced in this thesis, is an example of a medical application that uses it.

The work presented here covers: (a), The basis of the Digital Correlation Technique. (b), The basis of model-based matching using locations. (c), Introduction of the Simulated Annealing Technique. (d), Formulation of the Image Correlation Technique. (e), Formulation of the Digital Differential Radiography Technique.

## CHAPTER 2

### IMAGE MATCHING

#### 2.1. Introduction

The importance of image matching can be seen in the large number of applications. The following are some of them: target search, character recognition, measurement of motion, cartographic stereoscopic measurements. The matching process can be viewed as a simple process of finding the location in a distorted image at which a given pattern of gray levels matches the original image. The pattern itself can be a step, an edge, a line or in more advanced applications, a template representing a known object. The last can be extended to a situation when a piece of one image is to be matched against another, the original one. Finally, the most challenging task can be to match the whole image of a given scene with its distorted version. The result of such a match would be a deformation field. The last listed problem, matching entire images, will draw our attention in the following sections and chapters.

In this section we will explore two interesting techniques in image matching. But first some remarks on the correlation function as a means of similarity measure.

## 2.2. Correlation Function

Among many possible ways of measuring the degree of similarity between images are so called distance measures, that are the most intuitive and natural:

$$\max |I - I^*|, \quad \int \int |I - I^*|, \quad \int \int (I - I^*)^2, \quad (2.2-1).$$

where the integration is performed over the image space where the match is made.

Expanding the last expression in the following way:

$$\int \int (I - I^*)^2 = \int \int I^2 + \int \int I^{*2} - 2 \int \int I I^* \quad (2.2-2),$$

it may be assumed that the first two terms on the right side of the Eq. (2.2-2) are constant and the whole change of the equation value is carried by the third one [Rosenfeld, 1976], [Rosenfeld, 1982]. In addition according to the Cauchy-Schwarz inequality the third integral is limited as follows:

$$\int \int I I^* \leq \left[ \int \int I^2 \int \int I^{*2} \right]^{1/2} \quad (2.2-3)$$

Thus the following definition of a correlation function as a measure of similarity or lack of similarity, has its justification:

$$C = \frac{\int \int I I^*}{\left[ \int \int I^2 \int \int I^{*2} \right]^{1/2}} \quad (2.2-4)$$

The correlation technique is often used in signal and image processing. In case of real images a discrete form of the correlation function is more appropriate to use:

$$C = \frac{\sum_i \sum_j I [i, j] I^* [i, j]}{\left[ \sum_i \sum_j I^2 [i, j] \sum_i \sum_j I^{*2} [i, j] \right]^{1/2}} \quad (2.2-5)$$

Moreover, when we intend to match two real images of the same scene, despite all precautions to capture and process both images in identical conditions, a slight shift in intensity  $s$  between images may occur or/and the images may differ in contrast. This may be a particular situation when images are separated in time and exact conditions that were present when the first picture was recorded are more difficult to recreate. However, if we are able to ensure that the intensity shift  $s$  and the contrast change  $c$  are independent of gray intensity, we may relate both images by a simple formula:

$$I^* = cI + s \quad (2.2-6)$$

This is probably feasible to achieve for every recording system. The only question is, whether the useful range where Eq. (2.2-6) holds is sufficient to accommodate conditions that are present at the time when each of the images is taken. If so, the correlation function calculated with respect to the mean intensities of both images will again accommodate the matching process [Bookstein, 1990]:

$$C = \frac{\sum_{i,j} (I [i,j] - \bar{I}) (I^*[i,j] - \bar{I}^*)}{\left[ \left( \sum_{i,j} (I [i,j] - \bar{I})^2 \right) \left( \sum_{i,j} (I^*[i,j] - \bar{I}^*)^2 \right) \right]^{1/2}} \quad (2.2-7)$$

It is essential that Eq. (2.2-7) be invariant under the transformation (2.2-6). At this point, it is instructive to show on a very simple example how matching conditions influence the correlation function. We use two four-pixel images to plot correlation functions, with only one entry  $I_{11}^*$  in the second image changing:

$$I = \begin{bmatrix} 50 & 60 \\ 80 & 70 \end{bmatrix} \quad I^* = 2I = \begin{bmatrix} I_{11}^* & 120 \\ 160 & 140 \end{bmatrix} \quad (2.2-8)$$

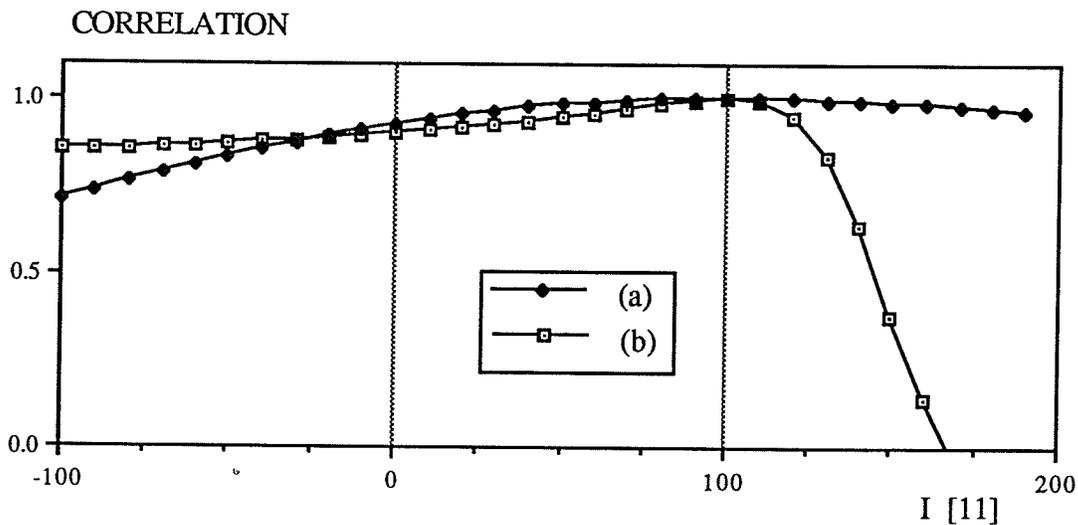


Fig. 2.2-1. Correlation function: (a), calculated using (2.2-5) and (b), calculated using (2.2-7).

In Fig. 2.2-1 both forms of correlation function, Eqs. (2.2-5) and (2.2-7), are displayed. When both images are identical, both correlation functions yield their maximum values that are equal to unity. However when any part of one of the images is distorted

(either its gray level scale or/and the geometry is changed), the decrease of the correlation function will indicate this mismatch. From Fig. 2.2-1, it is evident that the correlation function of Eq. (2.2-5) is flat and slightly out of symmetry, while the correlation calculated using Eq. (2.2-7), is highly non-symmetric. When the variable entry in the second matrix of Eq. 2.2-8 exceeds the value of 100 that makes a perfect match with the first image, the correlation function falls very rapidly. This may be a good reason why it is so difficult to verify the quality match using the human eye and correlation [Mazur, E., 1992]. It is however, a very good indicator of a perfect or almost perfect match.

If, somehow, someone is able to predict and possibly reconstruct the deformation of the second image, the correlation shows that by increasing its value. One way to do so is to search through the large number of images that actually constitute a library of possible distortions. The other approach is to use the so called *rubber mask* approach [Rosenfeld, 1976], [Rosenfeld, 1982], [Jacobson, 1976], that exploits the search throughout the space of permissible distortions (transformations). There is also an interesting proposal to subdivide the entire image into a number of subtemplates, connected by means of springs [Rosenfeld, 1982], [Jacobson, 1976] and finding the combination of partial matches that are as good as possible or require the least tension in the springs.

But generally, these methods are practical to use when the space of possibilities is relatively small, and when it is possible to make a good initial guess what kind of distortion should be used. In fact, the above two solutions have been limited to matching using small templates and to the situations when the deformations are limited to simple cases of rotation, scale change or other simple geometrical transformations.

### 2.3. Image Matching Using Location

An interesting approach to image matching has been given by Baird in his dissertation [Baird, 1985]. His interest was focused on the problem of the recognition of plane rigid shapes that have been subjected to an unknown distortion. The following definitions will help to present his method. Please note; The notation and definitions that are used below, are only for the purpose of this section and are not valid outside of it.

A single *pattern* is a set of local features that can be located independently from the others in a small region of the original image;

A *model pattern* is a set of patterns  $\mathbf{P}$  that form a model image;

The *instance image*  $\mathbf{Q}$  is expected to be a noisy and deformed copy of the model pattern.

The deformations considered affect the whole image as well as a particular pattern. However, in Baird's work, only pattern locations were used, leaving other properties unused. Patterns themselves were represented by single points.

A *matching*  $M$  of the size  $k$  is a one-to-one mapping from a subset of  $\mathbf{P}$  of the size  $k$  into  $\mathbf{Q}$ . From all possible matches, there is  $k!$  distinct possibilities.

The distortion that was investigated included only translation, rotation and scaling. A geometrical representation of matching is accomplished by a *registration* procedure  $\mathbf{R}$  that maps the Euclidean plane onto itself

$$\mathbf{R}(\mathbf{q}) \equiv \begin{pmatrix} r_x \\ r_y \end{pmatrix} + s \times \begin{pmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{pmatrix} \begin{pmatrix} q_x \\ q_y \end{pmatrix} \quad (2.3-1)$$

where  $r_x$  and  $r_y$  are the translational offsets,  $s$  is a scale factor, and  $\theta$  is a rotation angle

about the origin. By introducing additional variables  $r_1 = s \cos \theta$  and  $r_2 = s \sin \theta$ , we can rewrite Eq. (2.3-1) in the following form

$$\mathbf{R}(\mathbf{q}) \equiv \begin{pmatrix} r_x \\ r_y \end{pmatrix} + \begin{pmatrix} r_1 & -r_2 \\ r_2 & r_1 \end{pmatrix} \begin{pmatrix} q_x \\ q_y \end{pmatrix} \quad (2.3-2)$$

In other words, registration can be accomplished by specifying for all matching points  $\mathbf{q}$ , the set of parameters  $\{r_x, r_y, r_1, r_2\}$ .

Noise is introduced as the vector quantity

$$\boldsymbol{\varepsilon}_i \equiv \mathbf{R}(M(\mathbf{p}_i)) - \mathbf{p}_i \quad (2.2-3)$$

The last equation combines together the matching and the registration operations. In practical applications, noise is the result of an imprecision in feature extraction or in image acquisition. Because of that, it is conceivable to estimate  $\boldsymbol{\varepsilon}$  prior to the registration process. Noise constraints are such that the model point  $\mathbf{p}_i$  and the result of matching and registration  $\mathbf{R}(M(\mathbf{p}_i))$  must lie within the region in the plane bounded by straight sides. These sides, that are  $j$  in number, form a polygon. The following inequality holds for each side

$$\mathbf{u}_{ij} \cdot \boldsymbol{\varepsilon}_i \leq d_{ij} \quad (2.3-4)$$

for all  $j = 1, 2, \dots, l_i$  and all  $i = 1, 2, \dots, k$

where  $\mathbf{u}_{ij}$  is a unit vector perpendicular to the  $j$  side,  $d_{ij}$  is the closest distance of the point  $\mathbf{p}_i$  to the  $j$  side. Fig. 3.2-1 shows schematically the matching, the registration and noise constraints.

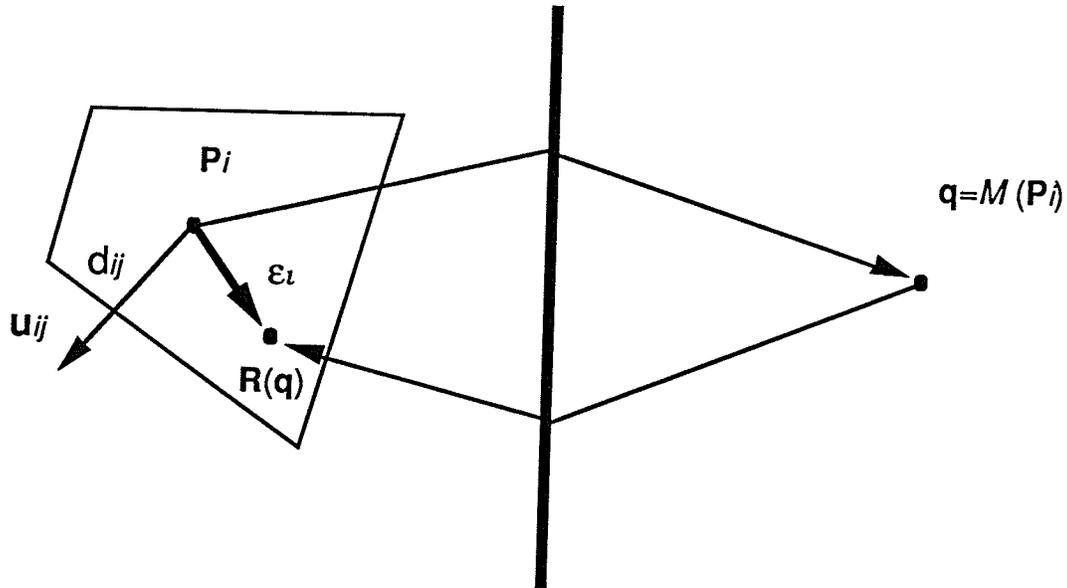


Fig. 2.3-1. Matching, registration and noise constraints.

Using (2.3-3), we can rewrite Eq. (2.3-4) to give the following form

$$\mathbf{u}_{ij} \cdot (\mathbf{R}(M(\mathbf{p}_i)) - \mathbf{p}_i) \leq d_{ij} \quad (2.3-5)$$

for all  $j = 1, 2, \dots, l_i$  and all  $i = 1, 2, \dots, k$

Assuming that noise constraints  $d_{ij}$ , model pattern  $\{\mathbf{p}_i\}$  and matching  $\{M(\mathbf{p}_i)\}$  are all given, the solution of Eq. (2.3-5), for each  $i$  and  $j$ , yields in four dimensional space  $\mathbf{R}^4[r_x, r_y, r_1, r_2]$ , a set of constraints or inequalities. These all constraints, when fulfilled simultaneously, indicate that match  $M$  is feasible.

All points in the model and the instance images are ordered and all possible matching permutations (altogether  $k!$ ), are checked using the feasibility test. However, if a partial match is not feasible a search branch is abandoned. By partial match we understand a match between the limited number of ordered points such as  $n < k$ . Depending on the feasibility criterion, the remaining branch, that is based on the given match of  $n$  ordered points is cut off or continued for  $n + 1$ . This original search algorithm is called the *pruned tree-search*.

The method is interesting with respect to how it incorporates noise. The Image Matching Using Location worked on models with simple deformations that included scaling, rotation and translation and was tested for the set of pattern points as large as  $k = 36$ . There is also requirement to transfer images from the gray level representation into the pattern representation prior to the matching process.

## 2.4. Digital Correlation Technique

Usually the surface of a body under investigation by DCT is coated, at random, with white and black regions to give a 'speckle appearance' in the scattered laser or white light. This ensures that each subset of an image is statistically different from others. Each image, as it is viewed by a TV camera, is a matrix of discrete samples of the light intensity scattered by a body surface. Each entry in the matrix that covers some surface of the object is referred to as a pixel. DCT determines the deformation through the numerical correlation of a selected subset of pixels from the reference image with its counterpart from the image of the body under deformation. The deformation itself is required to be a homogeneous one (displacements are linear functions of a pixel initial position, i.e., for a sufficiently small region straight lines remain straight after the deformation). If not, an image area under investigation must be broken into many smaller area subsets that can be processed separately. The procedure attempts to correlate the small subset  $\Delta M$  from the reference image (a square about 10 by 10 pixels in size) with all the same size subsets  $\{\Delta M^* [i]\}$  that cover the distorted image ( $i$  indexes these subsets). The coordinates of any point of the deformed body  $(x^*, y^*, z^*)$  can be expressed as a continuous function of the original coordinates  $(x, y, z)$  of the undeformed configuration:

$$x^* = x + u(x,y,z) \quad y^* = y + v(x,y,z) \quad z^* = z + w(x,y,z) \quad (2.4-1)$$

where  $u, v, w$  are displacement functions. Let  $I_{\Delta M}(x,y,z)$  and  $I_{\Delta M^*}^*(x^*,y^*,z^*)$  denote intensity functions at the centers of subsets of undeformed and deformed pictures, respectively. The essence of the DCT method is to arrive at the correct values for  $u, v$  and  $w$  of the subset  $\Delta M$ . In formal notation, for 2D, it is equivalent to minimizing in the least square sense the correlation coefficient  $C$  given by [Sutton, 1983] :

$$C_{\perp} \left( u, v, \frac{\partial u}{\partial x}, \frac{\partial v}{\partial y}, \frac{\partial u}{\partial y}, \frac{\partial v}{\partial x} \right) = \int_{\Delta M^* [i]} \left[ I_{\Delta M} - I_{\Delta M^* [i]}^* \right]^2 dM^* \quad (2.4-2)$$

The displacement functions  $u$  and  $v$  are the outcome variables. However, all derivatives help to reshape the  $\Delta M$  subset (a template) while matching it with all the deformed subsets  $\{\Delta M^* [i]\}$ . At the outset of the procedure, the template  $\Delta M$  with coordinates of its center  $(x,y,z)$  is chosen. Also ranges for all displacement functions and their derivatives are predetermined. (Usually the maximum values of displacement functions are limited by the dimensions of the distorted image, and accordingly the limitations on all derivatives are set.) At this stage, also, an increment for each variable is supplied. (Often it is equal to one tenth of the variable range.) The procedure starts with all derivatives set to zero. Eq. (2.4-2) estimates  $C$  for all possible combinations of  $u$  and  $v$  (roughly about a few hundred). The best values for  $u$  and  $v$  that minimize  $C$  are retained. In the next two steps, one at the time, all possible derivative combinations of the second and the third pair of variables are checked and set to their optimal values. After the estimation cycle of all six variables has been completed, the ranges of all the variables are decreased and centered about their recently estimated values. The procedure then enters the next iteration loop. The iteration process terminates as soon as the correlation function at the end of a cycle is

satisfactorily low. The presented sequence of steps ensures that the convergence of all variables is almost simultaneous regardless of the increment values of all variables that are used.

The above approach was recently presented by Sutton *et al.* [Sutton, 1983] in the form of a completely automatic computation procedure and employed successfully to determine displacements of a cantilever beam with a load at the end. In other works [Chu, 1985], [Peters, 1982], a different correlation function was used

$$C_I \left( u, v, \frac{\partial u}{\partial x}, \frac{\partial v}{\partial y}, \frac{\partial u}{\partial y}, \frac{\partial v}{\partial x} \right) = \int_{\Delta M^* [i]} I_{\Delta M} \cdot I_{\Delta M^* [i]}^* dM^* \quad (2.4-3)$$

and all effort was devoted to maximize the  $C$  function. This approach is similar to template matching in pattern recognition [Gonzalez, 1977]. DCT assures correct values for all parameters by picking them out from the ranges of values that are predetermined at the beginning of the optimization process.

The following are the limitations of the above described methods: All methods require, predetermined at the beginning, a set of value ranges for all variables. They correlate only a small portion of a reference image with the larger portion of the deformed image. As a final result, all six variables are estimated only for the central point of the subset  $\Delta M$ . The method is also limited to situations where the deformation is homogeneous as well as uniform, and were applied only in engineering to plane structural problems. Uniform translations as large as 0.14 inch were correlated [Peters, 1982], and elongations of 0.006 mm and strains of the order of 0.01, with accuracy less than 10%, were recovered [Chu, 1985].

## **CHAPTER 3**

### **SIMULATED ANNEALING OPTIMIZATION TECHNIQUE**

#### **3.1. Introduction**

We start with a descriptive presentation of the Simulated Annealing Technique (SAT).

SAT is a well known optimization technique that is a part of the whole family of Monte Carlo methods. The strong feature of this technique is that it finds a high quality solution that does not depend strongly on the choice of the initial solutions. It is applicable to the vast family of problems where the explicit mathematical solution does not exist.

To introduce SAT, the formalism of Markov chains is introduced that allows us to analyze the asymptotic convergence of the simulated annealing algorithm. Finally, some considerations are given to a cooling schedule.

#### **3.2. Formulation of Simulated Annealing**

SAT takes its name from the strong analogy to physical annealing. In the real process, close to perfect crystal structure emerges almost always at the end of a very slow and carefully mastered cooling process. It is essential that a system transform through a series of states of thermodynamic equilibrium.

In the liquid phase particles of the melted solid occupy energy states  $E_i$  with probability  $P_T$

$$P_T\{X=i\} = \frac{1}{Z(T)} \exp\left(\frac{-E_i}{k_b T}\right) \quad (3.2-1)$$

where  $X$  is the stochastic variable denoting the current state of the solid,  $k_b$  the *Boltzmann constant*,  $T$  the temperature of the melt and  $Z(T)$  is the canonical partition function [Gordon, 1980], [Hill, 1956], [Hill, 1960]

$$Z(T) = \sum_j \exp\left(\frac{-E_j}{k_b T}\right) \quad (3.2-2)$$

where the summation is extended over all possible states available for the solid to be found in. In the ground state that corresponds to a perfect crystal, the solid is in its lowest possible energy state. If the cooling process is not carefully done, the solid may end up in one of the number of meta-stable states. This is particularly true when instead of careful cooling, the solid was subjected to *quenching*.

Kirkpatric *et al.* [Kirkpatric, 1983], incorporated the concept of physical annealing into the optimization computation. The system under optimization is characterized by a cost function. Some authors call it a *goal* or a *state function*. Often the cost function has multiple global and local minima. In a real physical process, the cost function has its counterpart in the free energy function. The analogy goes even further: The method has the ability of a thermal system to locally increase its free energy. This helps it to *climb out* of local minima and consequently to lower its free energy level. The described feature is the most important factor that makes the method so powerful. The probability for such an increase in the energy (cost) level is governed by the Boltzmann formula:

$$P = \exp\left(\frac{-\Delta E}{k_b T}\right) \quad (3.2-3)$$

where  $\Delta E$  is the energy change and  $k_b T$  is a measure of the thermal energy. As the temperature  $T$  decreases, the probability  $P$  decreases also, and the number of accepted changes that increase the energy also reduces rapidly.

The basic difference that distinguishes SAT from Monte Carlo formulations is how the temperature is maintained during the optimization. The Monte Carlo method that was introduced by Metropolis *at al.* [Metropolis, 1953] and later modified by Gordon [Gordon, 1968, 1980], treats the temperature as a constant and the optimization as a process that tries to achieve and maintain equilibrium conditions (for the fixed temperature). As it is, it is rather a time development process. In some cases, the optimization is repeated for different values of the temperature. SAT takes all that and goes through many temperature levels (quite large in number), starting from a high value and going down till the system *freezes*. In another words, SAT treats the temperature as a variable (and not as a parameter). There is no doubt about advantages that this new approach has. One can imagine modeling of a real physical process that undergoes a phase transition instead of checking the equilibrium conditions at the particular temperature. However, there is a drawback: On each temperature level, to maintain the analogy to a physical process, there has to be an equilibrium in the system before the temperature is decreased. Because the temperature goes through a considerable number of steps, there has to be enough relaxation to keep the system at equilibrium. This, in turn, introduces some uncertainty as far as the system convergence is concerned. But, the traditional Monte Carlo method dealt with the same problem with the only difference that the scale was smaller.

Simulated annealing is a powerful and very flexible optimization method. To implement it, one has to design a unique goal function that describes the system under the investigation and its dynamism. It is also necessary to outline the basic principles the

system will be maintained by. In the case of image matching, we need a guideline that describes how to set different parts of a picture into the motion. We will call it the *movement algorithm* and we will explore it in the following chapters extensively. Another thing that also has to be prepared is the way or rate in which the system will approach the solution (*cooling schedule*).

### 3.3. Solution and Convergence

The following section gives the formal description of the simulated annealing. The notation as well as definitions are given after a book by Emile Aarts *et al.* [Aarts, 1989].

The *combinatorial optimization* problem is described by a set of problem instances.

An *instance* of the *combinatorial optimization* problem is defined as a pair  $(S, f)$  of the space solution  $S$  and the cost function  $f$  such as

$$f : S \rightarrow \mathbf{R} \quad (3.3-1)$$

and the task is to find the optimal solution  $i_{opt} \in S$  that satisfies

$$f(i_{opt}) \leq f(i) \quad \text{for all } i \in S \quad (3.3-2)$$

The search for the globally-optimal solution can be divided into two categories: minimization or maximization of the cost function. The only difference as far as the formal introduction of the method is concerned, is simply the reverse of the sign of the cost function. Consequently, in the case of the maximization

$$f(i_{opt}) \geq f(i) \quad \text{for all } i \in S \quad (3.3-3)$$

We also define a *neighborhood* of the solution  $i$ : For each solution  $i \in S$ , there is a set  $S_i \subset S$  of solutions that are *close* to  $i$  in some sense. It may be, for instance, a set of solutions that are obtained from  $i$  in a single transition.

Furthermore, we define the simulated annealing algorithm. The definition consists of three procedures.

(1) Generation mechanism. Let  $(S, f)$  be an instance of the combinatorial optimization problem. Among all solutions  $S$  we postulate (generate) a transition between two states  $i$  and  $j \in S$

(2) Acceptance criterion. The probability of accepting the transition from the state  $i$  to the state  $j$  is given by

$$P_c\{\text{accept } j\} = \begin{cases} 1 & \text{if } f(j) \leq f(i) \\ \exp\left(\frac{f(i) - f(j)}{c}\right) & \text{if } f(j) > f(i) \end{cases} \quad (3.3-4)$$

(3) Cooling schedule. A so called *control parameter*  $c_k$  is designed to play the role of the temperature. The decreasing of the control parameter is associated with an iteration step  $k$ .

Simulated annealing is the repetition of the Metropolis algorithm that is evaluated at decreasing values of the control parameter.

The mathematical formalism of Markov chains is a useful tool to model the simulated annealing. A Markov chain is a sequence of *trials* with the outcome probability of a given trial depending only on the outcome of the previous trial. Let  $X(k)$  be a stochastic variable denoting the outcome of the  $k^{\text{th}}$  trial, then the *transition probability* at the  $k^{\text{th}}$  trial for each pair  $i, j$  of outcomes is defined as

$$P_{ij}(k) = \mathbf{P} \{ \mathbf{X}(k) = j \mid \mathbf{X}(k-1) = i \} \quad (3.3-5)$$

The matrix  $P_{ij}(k)$  is referred to as the *transition matrix*. If the transition probability depends on the trial number  $k$ , then the Markov chain is called *inhomogeneous*. The Markov chain that does not depend on the trial number is called *homogeneous*.

The transition probabilities for simulated annealing algorithm are defined as follows

$$\forall i, j \in S : P_{ij}(k) = P_{ij}(c_k) = \begin{cases} G_{ij}(c_k) A_{ij}(c_k) & \text{if } i \neq j \\ 1 - \sum_{l \in S, l \neq i} P_{il}(c_k) & \text{if } i = j \end{cases} \quad (3.3-6)$$

Matrix  $G_{ij}(c_k)$  is the so called *generation probability*. It is the probability of generating a solution  $j$  from the solution  $i$  such as

$$\forall i, j \in S : G_{ij}(c_k) = G_{ij} = \frac{1}{\theta} \chi_{(S_i)}(j) \quad (3.3-7)$$

where  $\theta = |S_i|$ , for all  $i \in S$

$G_{ij}(c_k)$  are conditional probabilities that are independent of the control parameter  $c_k$  and chosen over the neighborhoods  $S_i$  (that are equal in size for all  $i$ ).

The second matrix in (3.7), matrix  $A_{ij}(c_k)$  is the so called *acceptance probability*

$$\forall i, j \in S : A_{ij}(c_k) = \exp\left(-\frac{[f(i) - f(j)]^+}{c_k}\right) \quad (3.3-8)$$

where, for all  $a \in \mathbf{R}$ ,  $a^+ = a$  if  $a > 0$ , and  $a^+ = 0$  otherwise

and is essentially the same as the Metropolis formula given in Eq. (3.3-4).

The definitions (3.3-6), (3.3-7) and (3.3-8) are more general ones of the simulated annealing algorithm. They are in an agreement and closely follow the analogy to the physical process.

The stationary distribution of a finite homogenous Markov chain with the transition matrix  $P$  is defined as a vector  $q$

$$q_i = \lim_{k \rightarrow \infty} \mathbf{P} \{ \mathbf{X}(k) = i \mid \mathbf{X}(0) = j \} \text{ for all } j \quad (3.3-9)$$

The stationary distribution can be regarded as the probability of the solution after an infinite number of transitions at a given  $T$ . Moreover it satisfies the following matrix equation.

$$\sum_j q_j P_{ji} = q_i \text{ for all } i \quad (3.3-10)$$

Clearly, it is an eigenvalue problem with solution  $\mathbf{1}$  for all  $q_i$ .

For the finite, irreducible, homogeneous and aperiodic Markov chain a given distribution  $q$  is stationary if the following relation is valid

$$q_i P_{ij} = q_j P_{ji} \quad (3.3-11)$$

The last equation is known as the *detailed balance equation*.

The following theorem (not proven here) is important as far as the simulated annealing algorithm convergence is concerned.

Theorem 1. Let  $(S, f)$  denote an instance of a combinatorial optimization problem and, let  $P(c)$  denote the transition matrix associated with the simulated annealing algorithm, defined by Eqs. (3.3-6), (3.3-7), (3.3-8). Furthermore, let the following condition be satisfied:

(1)

$$\begin{aligned} \forall i, j \in S \quad \exists p \geq 1, \quad \exists l_0, l_1, \dots, l_p \in S \\ \text{with } l_0 = i, l_p = j \quad \text{and} \\ G_{l_k l_{k+1}} > 0, \quad k = 0, 1, \dots, p-1 \end{aligned} \quad (3.3-12)$$

Then the Markov chain has a stationary distribution  $q(c)$ , whose components are given by

$$q_i(c) = \frac{1}{N_0(c)} \exp\left(-\frac{f(i)}{c}\right), \text{ for all } i \in S \quad (3.3-13)$$

where

$$N_0(c) = \sum_{j \in S} \exp\left(-\frac{f(j)}{c}\right) \quad (3.3-14)$$

It means that the stationary distribution  $q^*$ , at a zero value of a control parameter  $c$ , is attained after the infinite number of transitions.

$$\lim_{c \downarrow 0} q(c) = q^* \quad (3.3-15)$$

It also means (from Eqs. (3.3-13) and (3.3-14)) that each optimal solution is reached with equal probability

$$q_i^* = \frac{1}{|S_{opt}|} \chi_{(S_{opt})}(i) \quad (3.3-16)$$

or

$$\lim_{c \downarrow 0} \lim_{k \rightarrow \infty} \mathbf{P}_c \{ \mathbf{X}(k) = i \} = \lim_{c \downarrow 0} q(c) = q_i^* \quad (3.3-17)$$

And, finally, that the probability that the system is to be found in the optimal state is

$$\lim_{c \downarrow 0} \lim_{k \rightarrow \infty} \mathbf{P}_c \{ \mathbf{X}(k) \in S_{opt} \} = 1 \quad (3.3-18)$$

The last equation is a guarantee that for limit values of  $c$  and  $k$ , the simulated annealing algorithm asymptotically finds the optimal solution.

From the practical point of view, Theorem 1 implies that asymptotic convergence is attainable in the case of the infinite and homogeneous Markov chains, generated at descending values of the control parameter. This outcome is not very optimistic as far as the solution of real problems is concerned. For certain conditions, however, it is possible to expect asymptotic convergence with less restrictions. The answer lies in generating finite homogeneous Markov chains in descending values of the control parameter. Then the problem is reduced to the generation of a one single and inhomogeneous Markov chain.

Let  $c_k$  denote the value of the control parameter at the  $k^{\text{th}}$  trial,  $c_l^i$  be the value of the control parameter of the  $l^{\text{th}}$  homogeneous Markov chain and  $L$  stand for the length of the homogeneous and finite Markov chain.

$$c_k = c_l^i, \quad iL < k \leq (i+1)L \quad (3.3-19)$$

and

$$c_{l+1}^i \leq c_l^i, \quad i = 0, 1, \dots \quad (3.3-20)$$

such that

$$\lim_{l \rightarrow \infty} c_l^i = 0 \quad (3.3-21)$$

The following Theorem 2 lists conditions superimposed on the rate of the control parameter  $c_k$  that are necessary for Markov chain to converge to the distribution given by Eq. (3.3-16).

Theorem 2. Let  $(S, f)$  denote an instance of a combinatorial optimization problem and let  $P(k)$  denote the transition matrix associated with the simulated annealing algorithm defined by Eqs. (3.3-6), (3.3-7), (3.3-8). Furthermore, let the following condition be satisfied:

(1)

$$\begin{aligned} \forall i, j \in S \quad \exists p \geq 1, \quad \exists l_0, l_1, \dots, l_p \in S \\ \text{with } l_0 = i, l_p = j \quad \text{and} \\ G_{l_k l_{k+1}} > 0, \quad k = 0, 1, \dots, p-1 \end{aligned} \quad (3.3-22)$$

(2) The sequence  $\{c_l\}$  satisfies the following inequality

$$c_l \geq \frac{(L+1)\Delta}{\log(l+2)}, \quad l = 0, 1, \dots \quad (3.3-23)$$

where

$$\Delta = \max_{i, j \in S} \{f(j) - f(i) \mid j \in S_i\} \quad (3.3-24)$$

and  $L$  is chosen as the maximum of the minimum number of transitions required to reach an  $i_{opt}$  from  $j$ , for all  $j \in S$  such as  $L$  always exists as a result of condition (1).

Then Markov chain converges in distribution to the vector  $\mathbf{q}^*$ , with components

$$q_i^* = \frac{1}{|S_{opt}|} \chi_{(S_{opt})}(i), \quad \text{for all } i \in S \quad (3.3-25)$$

or, in other words

$$\lim_{k \rightarrow \infty} \mathbf{P}\{ \mathbf{X}(k) \in S_{opt} \} = 1 \quad (3.3-26)$$

The last result shows that the inhomogeneous Markov chain, built from the finite homogeneous chains, converges in distribution to a set of optimal solutions, provided that cooling is slow, such as in Eq. (3.3-23).

There is an improvement compared to the conditions listed in Theorem 1. Although the algorithm still finds an optimal solution after an infinite number of transitions, it is still impractical for any real application. A suitable approximation to the asymptotic convergence need to be worked out.

The simulated algorithm is an excellent general optimization algorithm, easily applicable to many problems. However, in many practical implementations, due to time constraints, the algorithm converges approximately to a global solution, and because of that, it has to be viewed rather as a reliable approximation approach. But on the positive side, the simulated annealing algorithm has the tremendous ability to return the solution arbitrarily close to the optimum in a wide variety of combinatorial optimization problems.

### 3.4. Cooling Schedule

In any highly randomized optimization process there is always a choice to be made: Either to follow the process that is aimed to get the optimal solution, at the risk of a very large, possibly impracticable, amount of computation time, or to limit the search to a quick solution at the risk of sub-optimality. Here, we present one of the possible finite-time implementations of the cooling schedule.

Let us define the *quasi-equilibrium*. The quasi-equilibrium is achieved if the probability distribution  $\mathbf{a}(L_k, c_k)$  of the solution after the  $L_k$  trials of the  $k^{\text{th}}$  Markov chain is

sufficiently close to  $\mathbf{q}(c_k)$ , the stationary distribution at  $c_k$ , defined by Eqs. (3.3-13) and (3.3-14), i.e.

$$\|\mathbf{a}(L_k, c_k) - \mathbf{q}(c_k)\| < \varepsilon \quad (3.4-1)$$

for a certain positive value  $\varepsilon$ .

The original cooling schedule [Kirkpatrick, 1983] consists of the following parameters: an initial value of the control parameter, a decrement of the control parameter and a final value of the control parameter.

The initial value of the control parameter is such that the initial acceptance ratio is close to one.

The decrement of the control parameter is given by a simple formula

$$c_{k+1} = \alpha c_k \quad k = 1, 2, \dots \quad (3.4-2)$$

The constant  $\alpha$  is usually chosen from an interval (0.8, 0.99). The final value of the control parameter is determined by the fluctuation of the cost function. If the cost function of the solution obtained in a particular Markov chain remains unchanged during the execution of a number of consecutive chains, the algorithm is terminated. In order to sustain the system in the state of quasi-equilibrium, the number of trials  $L_k$  is usually determined based on experience.

The other cooling schedule [Aarts, 1989], the so called the *polynomial cooling schedule*, is also based on the concept of quasi-equilibrium (3.4-1) modified in the following way

$$\forall k \geq 0: \|\mathbf{q}(c_k) - \mathbf{q}(c_{k+1})\| < \varepsilon \quad (3.4-3)$$

for some positive value of  $\varepsilon$ . The last equation postulates that quasi-equilibrium is maintained throughout the optimization process and only small steps are adopted so that the stationary distributions for the subsequent homogeneous chains will be close to each other. Let us assume that a certain number of transitions is generated at a certain level of  $c$ . The acceptance ratio  $\lambda$  for such a sequence can be expressed as

$$\lambda \approx \frac{m_1 + m_2 \exp\left(\frac{-\overline{\Delta f}^{(+)}}{c}\right)}{m_1 + m_2} \quad (3.4-4)$$

where  $m_1$  is the number of trials from the state  $i$  to the state  $j$  for which  $f(j) \leq f(i)$ ,  $m_2$ , the number of transitions such that  $f(j) > f(i)$ , and  $\overline{\Delta f}^{(+)}$  is the average difference in the cost over the  $m_2$ , the transitions that lead to an increase in the cost.

Then the expression on  $c$  can be calculated

$$c = \frac{-\overline{\Delta f}^{(+)}}{\ln\left(\frac{m_2}{m_2\lambda - m_1(1-\lambda)}\right)} \quad (3.4-5)$$

The last equation is useful to determine the initial value of the control parameter  $c_0$ . It is done by the convergence of  $c$  calculated repeatedly with the acceptance ratio set slightly below and close to one.

From Eq. (3.4-3) the closeness of the stationary distribution for any two successive values of the control parameter can be expressed

$$\forall i \in S : \quad \frac{1}{1+\delta} < \frac{q_i(c_k)}{q_i(c_{k+1})} < 1+\delta, \quad k = 0, 1, \dots \quad (3.4-6)$$

for some small positive number  $\delta$ . It can be proved that Eq. (3.4-6) is equivalent to the following

$$\forall i \in S : \frac{\exp\left(-\frac{\delta_i}{c_k}\right)}{\exp\left(-\frac{\delta_i}{c_{k+1}}\right)} < 1 + \delta, \quad k = 0, 1, \dots \quad (3.4-7)$$

That last can be rewritten to give an expression for the two consecutive values of the control parameter

$$\forall i \in S : c_{k+1} > \frac{c_k}{1 + \frac{c_k \ln(1 + \delta)}{f(i) - f_{opt}}}, \quad k = 0, 1, \dots \quad (3.4-8)$$

With some approximation the last equation gives the formula that can be used as a practical implementation of the cooling schedule

$$c_{k+1} = \frac{c_k}{1 + \frac{c_k \ln(1 + \delta)}{3\sigma_{c_k}}}, \quad k = 0, 1, \dots \quad (3.4-9)$$

where  $\sigma_{c_k}$  is a standard deviation of the cost function.

The algorithm introduced depends on two parameters: the initial acceptance ratio  $\lambda$  and the distance parameter  $\delta$ . There is always trade off between the decrement of the control parameter  $c_k$  and the length  $L_k$  of the homogeneous Markov chain. Intuitively, the larger decrement in  $c_k$  requires longer  $L_k$  to restore quasi-equilibrium at the next  $c_{k+1}$ .

## CHAPTER 4

# IMAGE CORRELATION TECHNIQUE: A NEW PROPOSAL FOR IMAGE MATCHING

### 4.1. Introduction

We introduce a new method, Image Correlation Technique (ICT), that estimates the transformation of deformations from pictures. The outcome of ICT is a displacement field. The search for the displacement field represents the inverse problem for image restoration. The geometric distortion that takes place between consecutive picture frames, is, in general, unknown. Using an original algorithm and employing simulated annealing, a well established optimization technique, by rearranging pixels, from a picture frame taken prior to the deformation, we arrive at the pixel arrangement represented by a picture frame taken in the course of, or after the deformation was completed. The method works equally well for linear and non-linear cases. We present examples of deformation estimation for pairs of two-dimensional images. However, the method can be readily applied to the three-dimensional object representation such as CT images of parts of a human body under the action of stress .

ICT finds the transformation of body coordinates that uniquely connects different stages of deformation recorded in a sequence (or at least two) of images. Under the term *image* we understand a set of data that can be viewed as a picture on a screen. The meaning can be extended to the set of data that represent a 3-dimensional (3D) structure,

such as a human body, reconstructed from projections by a means of computer tomography or other imaging modalities

ICT employs the powerful Boltzmann apparatus that forms the basis of simulated annealing. The specially crafted algorithm that we call a *movement algorithm*, a *goal function*, and a *cooling algorithm*, make Image Correlation Technique new, unique and distinct.

In the subsequent sections we present our concepts of the goal function, the placement algorithm, and the cooling schedule that integrated form Image Correlation Technique.

## 4.2. Correlation Function

In our application we introduce the following image similarity measure:

$$C = 1000 \frac{\sum_{ij} \frac{I [i,j] I^* [i,j]}{1 + \ln(|I [i,j] - I^* [i,j]| + 1)}}{\sum_{ij} (I^* [i,j])^2} \quad (4.2-1)$$

where  $I$  and  $I^*$  are intensity values from two different pictures. In the language of optimization techniques, we will refer to  $C$  as a *cost* or a *goal function* normalized with respect to the deformed image  $I^*$ . In case of a perfect match the value of the correlation (4.2-1) is normalized at 1000. From an algebraic point of view, the correlation function is a sensitive function of an exact match. To present some charts of the correlation (4.2-1), we introduce an example of two small test images as described by (4.2-2).

$$I = \begin{bmatrix} I_{11} & 60 \\ 80 & 70 \end{bmatrix} \quad I^* = \begin{bmatrix} I_{11}^* & 60 \\ 80 & 70 \end{bmatrix} \quad (4.2-2)$$

Fig. 4.2-1 shows the plot of correlation (4.2-1) for a match of the test images  $I$  and  $I^*$  (4.2-2) with  $I^* [1,1]=100$ . The element  $I [1,1]$  of  $I$  object is a variable (along the horizontal axis). The function has a sharp, delta-like maximum at the position  $I [1,1]=100$  where both images are identical. The correlation function is approximately symmetrical about the matching point. The region designated by two horizontal arrows shows the decreasing range that is on the right side of the matching point. Beyond this range, in the direction of higher values, the correlation increases very slowly and the symmetry is lost. That lack of symmetry is the result of using in Eq. (4.2-1) a denominator containing only the deformed image  $I^*$  that is unchanged during the entire optimization. This arrangement could be changed. We used it as it gave the convenience of calculating the denominator only once. This in turn made computations much faster. Lack of symmetry is compensated by the placement algorithm that we will introduce in the next paragraph. The feature of elastic movement of the picture with some restrictions on the rigidity of the movement ensures that, globally, the correlation function uniquely reflects the degree of similarity between images. To the extent of examples and deformations that have been explored in this work, correlation (4.2-1) is appropriate to use, giving positive results. It has to be stressed that the most important feature of the correlation function is its delta-like sharp shape at the point of the exact match. Using implementations of other known correlation functions gave negative results, although they were not checked extensively. In particular, Fig. 4.2-1 should be contrasted with the correlation functions of Fig. 2.2-1.

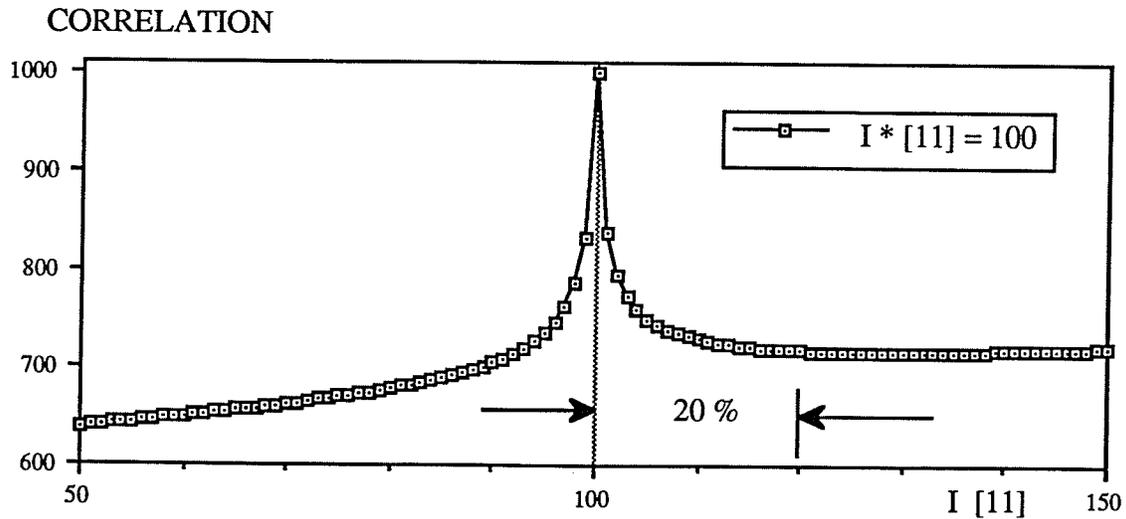


Fig. 4.2-1. Correlation function (4.2-1) for matching images given by (4.2-2) with  $I^*[1,1]=100$ . Region designated by horizontal arrows is a decreasing range of correlation on the right side to the point at  $I^*[1,1]=100$ , where the match is perfect. The extension of the decreasing range is 20% of the value of the matching point. Beyond that range the correlation increases very slowly.

Fig. 4.2-2 presents correlation function (4.2-1) with the element  $I^*[1,1]$  of the second image  $I^*$  as a parameter. Each curve resembles the one of Fig. 4.2-1. However, the dynamism (the magnitude of function changes, especially around the matching point) is quite different for different values of the parameter  $I^*[1,1]$ . As a matter of fact, the greater the parameter value the greater the function dynamism. This has important implications. Along the optimization process, at first, larger changes of the correlation function are restricted. This means that regions of relatively high grey level intensity values, are matched first. As optimization process proceeds, gradually, the regions with the lower intensity are matched. In other words, if we explore 3-D representation of an image (with the image grey level intensity on the third axis) we may expect that *tops* of the *hills* are being taken care of first while the deepest *valleys* are matched last.

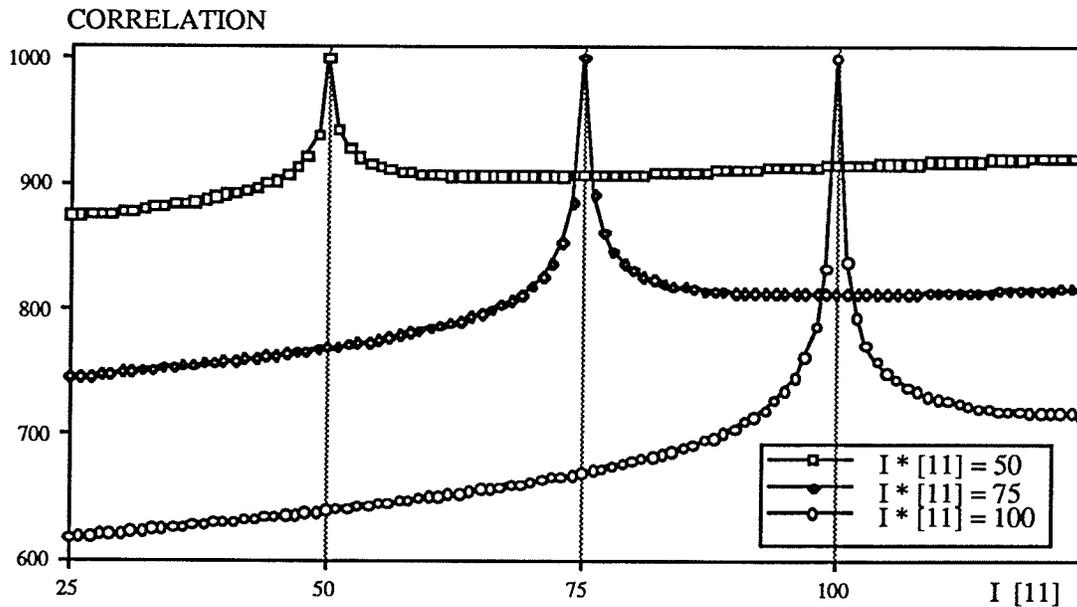


Fig. 4.2-2. Correlation function (4.2-1) for matching images given by (4.2-2). Each curve is for a different value of parameter  $I^* [1,1]$ .

For the purpose of the future work that may very well include matching real and noisy images, based on experience with correlation (4.2-1), the following form of the correlation may be more suitable to use:

$$C = \frac{\sum_{i,j} \frac{(I [i,j] - \bar{I}) (I^*[i,j] - \bar{I}^*)}{A \left| \frac{I [i,j] - \bar{I}}{\bar{I}} - \frac{I^*[i,j] - \bar{I}^*}{\bar{I}^*} \right| + 1}}{\left[ \left( \sum_{i,j} (I [i,j] - \bar{I})^2 \right) \left( \sum_{i,j} (I^*[i,j] - \bar{I}^*)^2 \right) \right]^{1/2}} \quad (4.2-3)$$

This may be a particular case when images are separated in time and exact conditions that were present when the first picture was recorded, are more difficult to recreate. Fig. 4.2-3 shows the correlation curves according to Eqs. (4.2-3) and (2.2-7). The second curve represents the correlation function that is commonly used. The proposed (4.2-3) has the sharp, delta-like shape, similar to the one of (4.2-1), with the possibility of controlling its

width by changing the parameter  $A$ . In addition by rearranging the normalization term in the denominator, it is possible to obtain similar dynamism as is shown in Fig. 4.2-2.

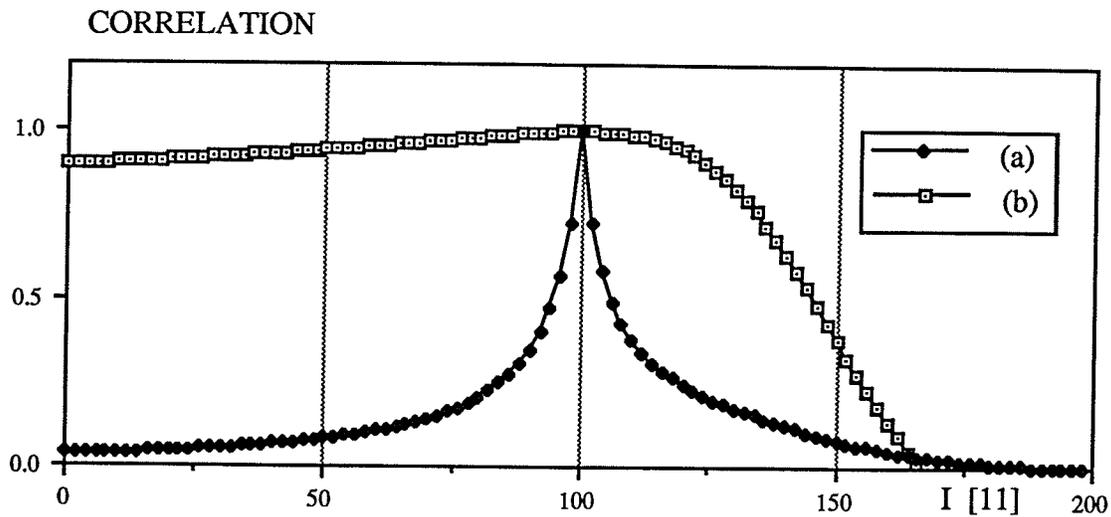


Fig. 4.2-3. Correlation function for matching two images (2.2-8): (a), calculated using (4.2-3) with  $A = 50$ , (b), calculated using (2.2-7).

Again, we may draw an analogy to a real physical system. We may designate, let us say, the distorted image to be a ground state. Thus the correlation function, already a match indicator, may well be a measure of an order in the system or a state function. The optimization process tries to bring the reference image as close as possible to the distorted one, that is, according to our convention the ground state. Because of this analogy to thermodynamics, the optimization process can be compared to the cooling itself. The proposed correlation function, together with the originally adopted *movement algorithm*, form the backbone of the new matching method and make it successful in matching images when large deformations and strains are present.

### 4.3. Movement Algorithm

The essence of ICT is the intentional deformation of the reference image. The way it is done, step by step, is prescribed by *the placement algorithm*. In this section we will present and discuss it at full length.

The ICT method is essentially an iteration method equipped with the Boltzmann decision apparatus. We start our process from an undeformed configuration (the reference image) and go through a series of not necessarily real deformations that, we hope, will lead us to the deformation that is close to or the same as the deformation captured by the second image (the deformation image). By doing this we are trying to emulate the real deformation that took place. Each intermediate deformation is generated at random and tested as to how far it falls from the destination deformation. The correlation function is used as a means of checking on the match and the decision part (to retain or reject the trial) is transferred to the Boltzmann decision algorithm.

One has to note that at each stage of the optimization of images, both the temporary and the final ones depict the same object but with different deformation histories. The similarity measure and the decision that are taken ensure that the simulated deformation will end up the same as, or close to the deformation recorded by the second picture (the deformed one). What is the most important, is the fact that throughout all this lengthy optimization process we retain the information on all pixel positions from the original (reference) picture configuration. Having this, we are in the position of knowing the deformation field.

The following three steps are repeated until the system arrives at the solution.

**Step 1.** At random, we select:

- a pixel  $I(i,j)$  with  $i, j \in \{2, \dots, M-1\}$
- a direction  $(\Delta i, \Delta j)$  such as  $\Delta i, \Delta j \in \{-1, 0, 1\}$   
and  $\Delta i = \Delta j \neq 0$
- three numbers  $R_x, R_y, P$ ,  
all from the open interval  $(0, 1)$ , and  
- a number  $R_f$ , from the open interval  $(0, M)$

Then, we move the canvas of pixels  $I(m,n)$  from its actually occupied position in the picture to a new position that is determined from formulas:

$$X_I^* [m, n] = X_I [m, n] + \Delta X_I [m, n] \quad (4.3-1)$$

$$Y_I^* [m, n] = Y_I [m, n] + \Delta Y_I [m, n]$$

where:

$$\Delta X_I [m, n] = \frac{X_I [m + \Delta i, n + \Delta j] - X_I [m, n]}{\exp [Z]} R_x \quad (4.3-2)$$

$$\Delta Y_I [m, n] = \frac{Y_I [m + \Delta i, n + \Delta j] - Y_I [m, n]}{\exp [Z]} R_y$$

and

$$Z = \frac{(X_I [m, n] - X_I [i, j])^2 + (Y_I [m, n] - Y_I [i, j])^2}{(F + R_f)^2} \quad (4.3-3)$$

At the same time, this new configuration changes the intensity distribution in the temporary image

$$w I^*[m,n] = \sum_{k,t=1}^M \delta \{ \text{round}(X_I[k,t] - m), \text{round}(Y_I[k,t] - n) \} I[k,t] \quad (4.3-4)$$

where  $w$  is the number of pixels whose cartesian coordinates overlap the pixel of picture coordinates  $(m, n)$ . If  $w$  is zero then  $I^*$  is also set to zero. Fig. 4.5-5 gives a graphical representation on how the interpolation works.

**Step 2.** Following the random displacement from step 1, we evaluate the cost function  $C$  (the matching that involves temporary and deformed image) and make a decision. We accept the change in the configuration, if  $C$  increases. If  $C$  decreases, we may accept it only if the probability  $P(C)$  is greater than the random number  $P$  chosen at random from a uniform distribution in step 1.

$$\left\{ \begin{array}{ll} \text{if } \Delta C > 0 & \text{step is accepted} \\ \text{if } \Delta C \leq 0 & \text{and } P(C) > P \quad \text{step is accepted} \\ \text{if } \Delta C \leq 0 & \text{and } P(C) \leq P \quad \text{step is not accepted} \end{array} \right\} \quad (4.3-5)$$

where

$$P(C) = \exp \left[ -\frac{|\Delta C|}{T} \right] \quad (4.3-6)$$

**Step 3.** The optimization terminates if  $C$  no longer increases or fluctuates with a certain small amplitude. This criterion helps to determine if the system has converged.

For a single value of the *temperature-like parameter*  $T$  in Eq. (4.3-6), all three steps are repeated many times.  $T$  itself is subject to change according to the cooling schedule that we introduce in the next section.

The conditional decision from step 2 is essential for the simulated annealing optimization. It provides means for escaping from local minima.

#### 4.4. Cooling Schedule

The optimization scheme will not converge to an optimal solution unless we prescribe an appropriate *cooling schedule*. The *temperature-like parameter*  $T$  in Eq. (4.3-6) assures the control of the process. As  $T$  decreases along the optimization process, the decision algorithm (4.3-5) accepts fewer movements that lead to the refinement of the solution. Traditionally the total number of trials at each  $T$  level is kept constant. The cooling schedule is of the form [Aarts, 1989]:

$$T^k = \frac{T^{k-1}}{1 + \frac{T^{k-1} \ln(1 + \Delta)}{\sigma}} \quad (4.4-1)$$

where  $\sigma$  and  $\Delta$  are means of additional control.

Initially  $T$  is set to a large value so that any change in the correlation function is acceptable.

It came out in the course of many experiments, that the most suitable cooling scheme that worked well together with the movement algorithm was as follows: The number of steps that were accepted on each  $T$  level (not the total number of steps), was kept constant. The result was that the number of trials on each  $T$  level was increasing as  $T$  was decreasing. The decrease of  $T$  is equivalent to an increase of the rejection number. This is the immediate difference from the traditional approach. This approach lengthens the process; however, it ensures that ICT converges with relative ease.

In experiments described in Chapter 6 another cooling schedule has been implemented: After completion of a predetermined number of iteration steps, if the correlation was less than a control value, then the correlation was assigned to the control value and  $T$  was reduced by a fraction of its actual value.

$$T^k = aT^{k-1} \quad (4.4-2)$$

where the fraction  $a$  is from an open interval (0-1). This procedure was repeated, each time, after completion of the predetermined number of steps. The meaning of this is that a system may stay unexpectedly long on a certain  $T$  level until the correlation lowers its value. Such a construction provides a certain level of feedback between a system in optimization and its cooling schedule. It makes convergence easier, but, on the other hand, it further increases the total optimization time.

## 4.5. Discussion

### 4.5.1. Optimization

Solving a combinatorial optimization problem is a process of finding the best or optimal solution among a finite or countably infinite number of available solutions [Foulds, 1981].

The search for the displacement is done by matching a picture taken prior to the deformation (reference image), with the picture taken in the course of or after the deformation. We will identify the process of image matching with changes in a hypothetical system. To find the optimal or the best match, we apply the optimization to the system with the incorporated concept of simulated annealing. The system under optimization is uniquely characterized by the cost function (4.2-1).

Fig. 4.5-1. shows schematically the behavior of the correlation function that drives the optimization process. An arrow in the upper right corner shows the direction of changes of the temperature-like parameter. At every optimization stage the system has the ability to lower its state function (the correlation function), and temporarily depart from the solution direction. The probability for such a movement, given by Eq. (4.3-6), depends on the  $C$  decrement and the temperature-like parameter  $T$ .

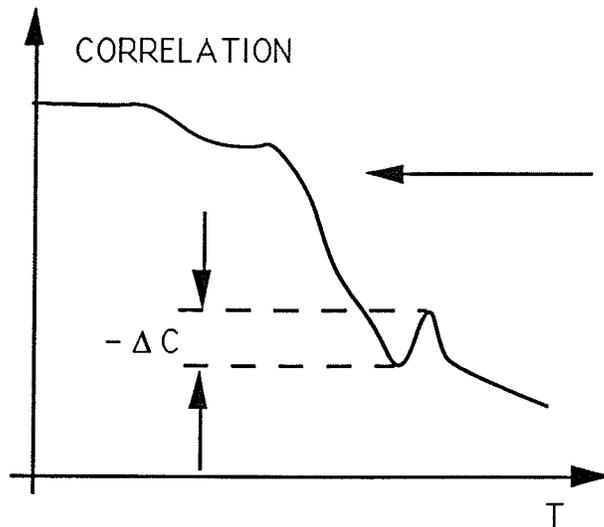


Fig. 4.5-1. The optimization process driven by the correlation function. An arrow in the right upper corner indicates the direction of the temperature-like parameter change. Probability (4.3-6) provides a means to overcome local minima with the depth  $-\Delta C$ .

#### 4.5.2. Image Grid Movement

When considering pixel movement, it is worth discussing steps 1 and 2 in more detail. There are eight directions to generate the picture movement, and all but the pixels that form the border of the picture can participate in the movement.

The distance that a particular pixel can move at is limited by the distance to its neighbor that lies in the direction  $(\Delta i, \Delta j)$ . By providing random numbers  $R_x, R_y$  (that are the same for all pixels in each trial) as a means of additional control, the whole cell, designated by  $(\Delta i, \Delta j)$ , is accessible is shown in Fig. 4.5-2.

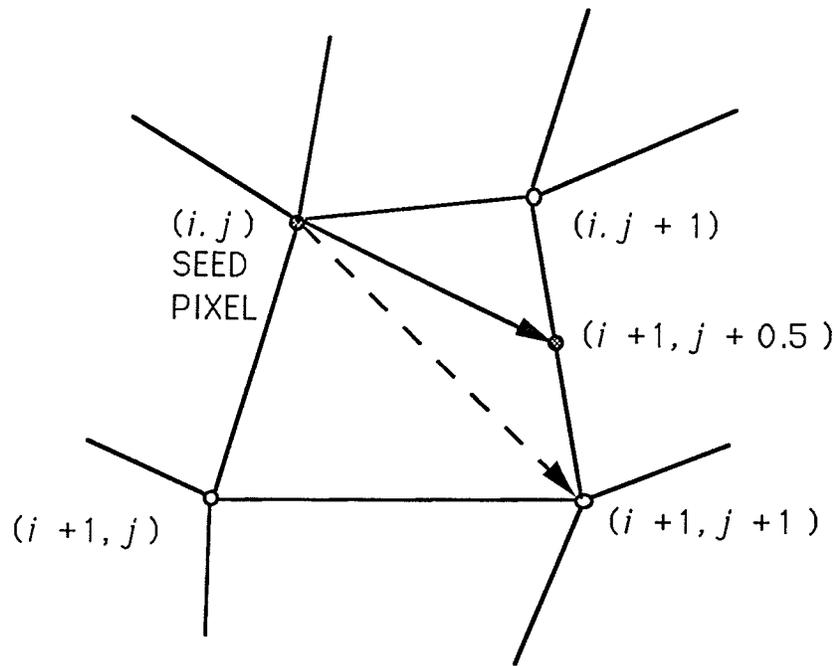


Fig. 4.5-2. A schematic movement for a particular pixel cell. Dashed line represents the direction chosen at random ( $\Delta i = 1, \Delta j = 1$ ), the solid line reflects the 'correction' imposed by means of parameters  $R_x$  and  $R_y$ . Note also that as long as  $R_x$  and  $R_y$  are real numbers from the open interval (0,1), all points within the quadrilateral are accessible.

In the sequence of images Fig. 4.5-4 from (a) through (l), several representative movements from consecutive trials are depicted. The CT image of a human breast sample was used to produce an example of moves that can be generated by the placement algorithm. The description of the image is provided in Chapter 6. The algorithm was set to accept all moves that were generated. This was done by assigning and keeping the temperature-like parameter  $T$  at a relatively high value so that Eq. (4.3-6) was very close to unity. In addition the flexibility constant  $F$  also was set to unity to give moderate and smooth deformations. The reference image, schematically represented by the grid, is superimposed onto the deformed image. In the course of movement generation, the grid takes different positions within the image frame. The deformed image is shown in the grey level representation and remains unchanged throughout the whole sequence. This way of displaying enables visual representation of the matching process by using images.

This convention is used in the following chapters to present results and the optimization process itself. The sequence starts before any move takes place and with reference image (a). The white dots placed close to the image edges mark positions of the reference border pixels that are excluded from any move. These dots are shown only in (a). Between each image in the sequel, there is a specific number of algorithm steps that were performed.

Fig. 4.5-4 b shows at least four moves of the reference grid that were already generated. We will list them but not necessary in the order they were generated. The first one, very rigid, is from right to the left. The second, also with high level of rigidity, is in the direction towards the bottom of the image. Both moves, were generated with a relatively high amplitude ( $R_x$  and  $R_y$  close to unity). The third move, from the upper right corner towards the lower left corner is with the small amplitude and mild rigidity. This move is really interesting. It produces five circular wave crests, centered on the center of the distortion, that result from the small displacement amplitude and rounding offs that occurred in the Quick Draw environment (Apple's Macintosh displaying software interface). And finally the fourth movement, in the central part of the image and to the left, has a mild amplitude and flexibility close to its higher limit. This one can be compared to the case presented previously in Fig. 4.5-3 with the flexibility  $R_f$  equal to unity. 4.5-4c presents the system some number of steps later. Obviously previous features are blurred by moves that were generated between frames (b) and (c). This is how moves with larger stiffness dissolve the ones with less rigidity, unless there is some other way to decide which of moves gives the desired configuration and should be sustained or rejected. An even more convincing example of the above can be discovered by comparing frames (d) and (e), following up the distortion in the left upper corner in (d). Most of the frames, from (d) to (j), show in a number of instances the reference image grid departing significantly from the frame common for both pictures. This surprising result has a simple explanation in the algorithm formulation. According to (4.3-2) the real displacement is determined by the distance of a particular pixel to its neighbor that lies in

the direction of the move and also, as we recall, all border pixels do not participate in any move. In addition, the displacement is weighted by factors  $R_x$ , and  $R_y$ . To explain it further, let us assume that in the series of moves the entire image grid was pushed away from one of the image borders (Fig. 4.5-4f). This explains wide spaces between image borders and the grid itself. If a move is generated that in fact tries to push the grid back, in the opposite direction to that we assumed, and in addition the rates  $R$ 's in the move direction are significantly high, the resulting image would be very much similar to the ones in the Fig. 4.5-4 f. or Fig. 4.5-4 i. An additional understanding of the above is also provided by Fig. 4.5-2. Finally, the beauty of images (k) and (l) proves high potentials of the algorithm.

As is apparent, the reference image movement is restricted only by the image borders and the image stiffness itself, limited by  $F$ . In the real optimization program run, an additional restriction is superimposed: lowering down the temperature-like parameter, makes the move rejection more and more frequent, until the system under the optimization freezes into the solution.

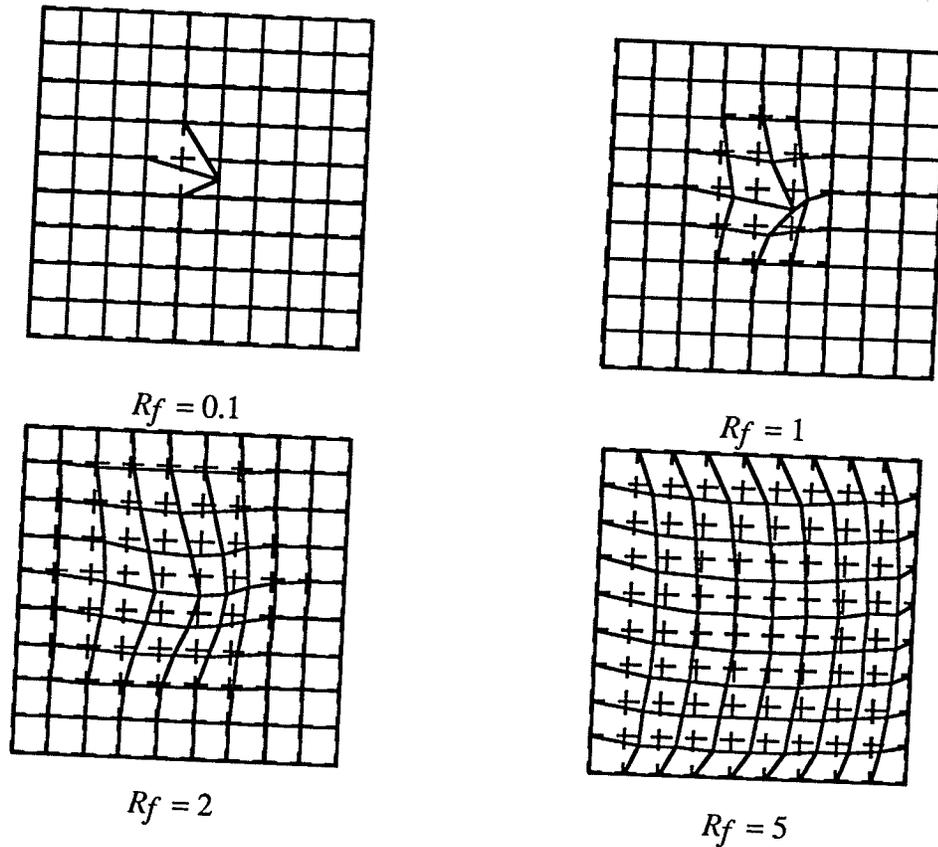
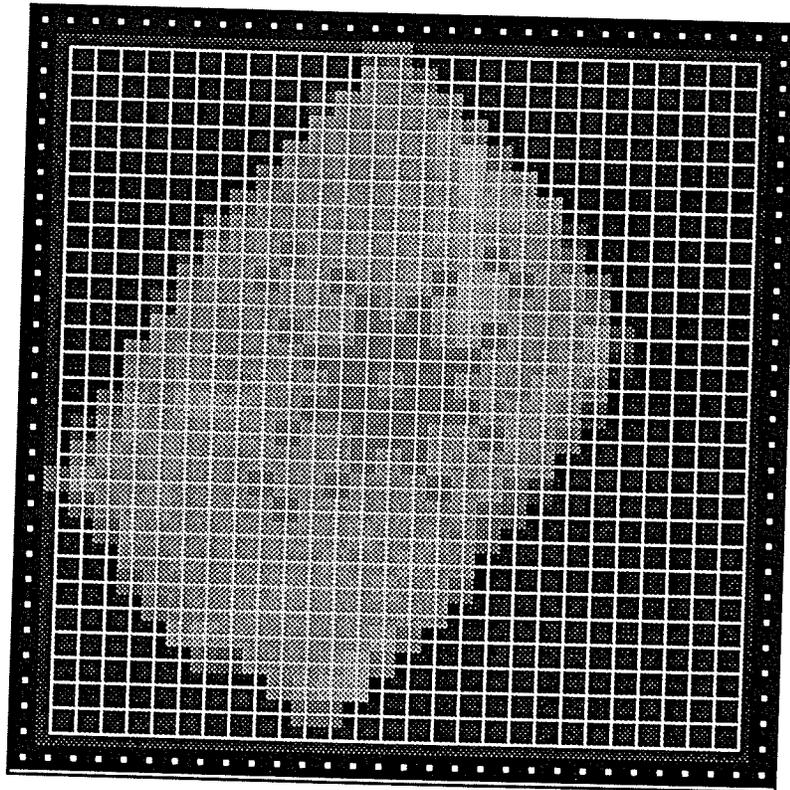


Fig. 4.5-3. The image movement induced by the movement algorithm. Pixels from the reference image are represented by the grid and pixel centers are in the intersections of the lines. The movement starts from the undisturbed configuration at the image location  $i = 5, j = 5$  (seed pixel) in the direction  $\Delta i = 1, \Delta j = 1$  from the center of the image towards the right side and down, with  $R_x = 1, R_y = 0.5, F = 0$  and the flexibility parameter  $R_f$  fixed at different levels. The greater the flexibility parameter the more rigid the grid becomes. In a real application run the variable  $R_f$  and the factor  $F$  that enter the denominator in Eq (4.3-3), together, ensure the flexibility of the optimized picture grid. Factor  $F$  plays role as the flexibility limitation. It sets flexibility upper limit.



(a)  
Fig. 4.5-4. (a) - (l), A sequence of movements applied to a 30 x 30 pixel image. Pixels from the reference image are represented by the grid and the deformed picture of a human breast CT sample is shown in the background. The reference grid from an undisturbed image is moved following the sequence of moves generated by the placement algorithm.

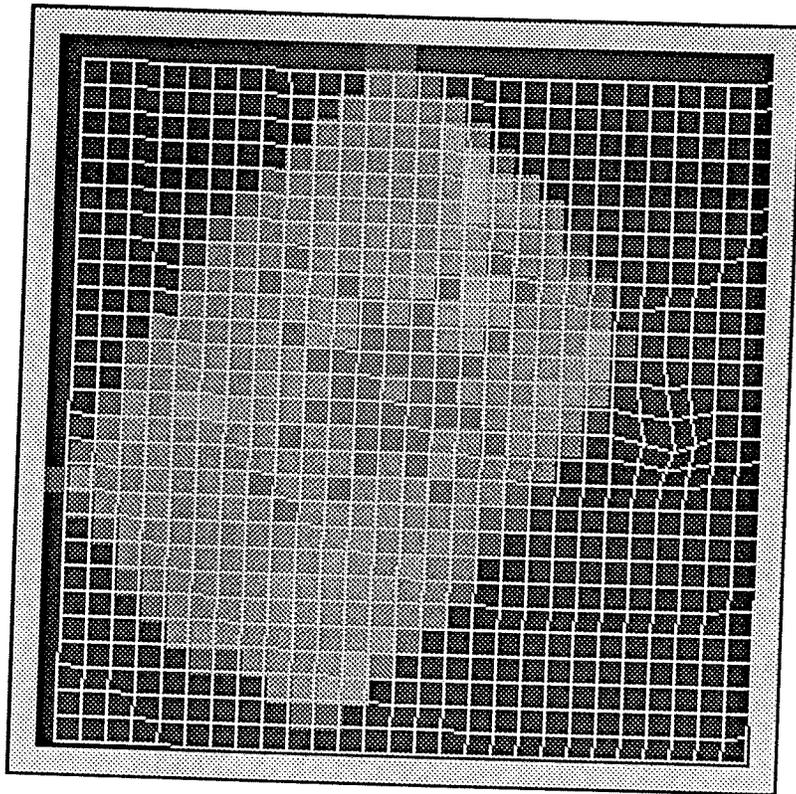


Fig. 4.5-4 contd.

(b)

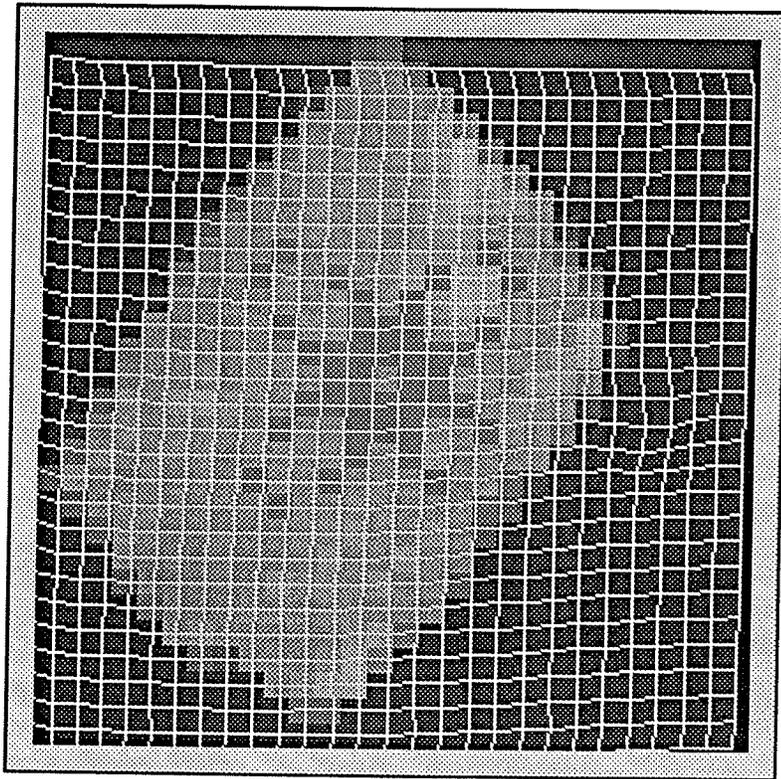


Fig. 4.5-4 contd.

(c)

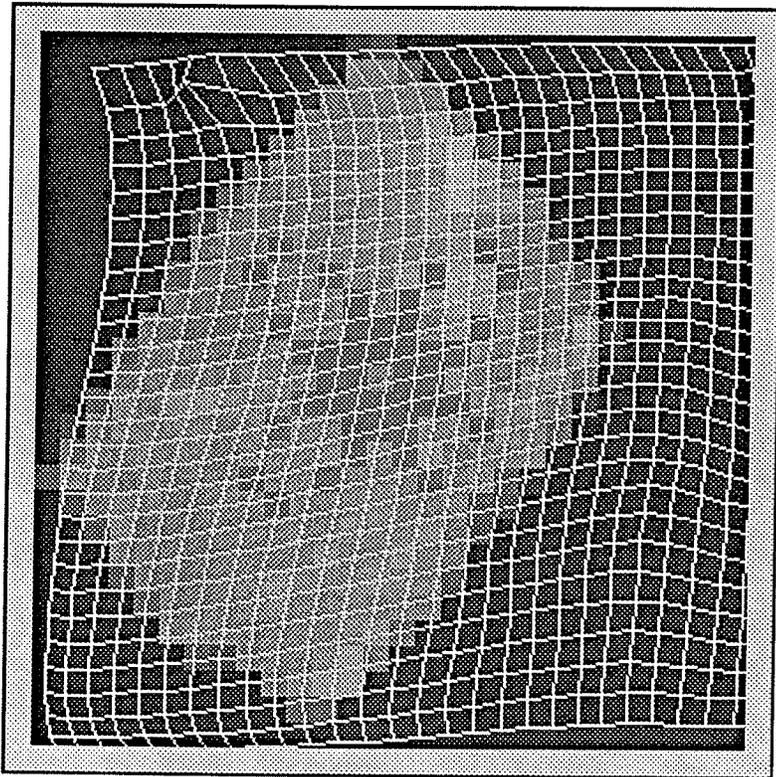


Fig. 4.5-4 contd.

(d)

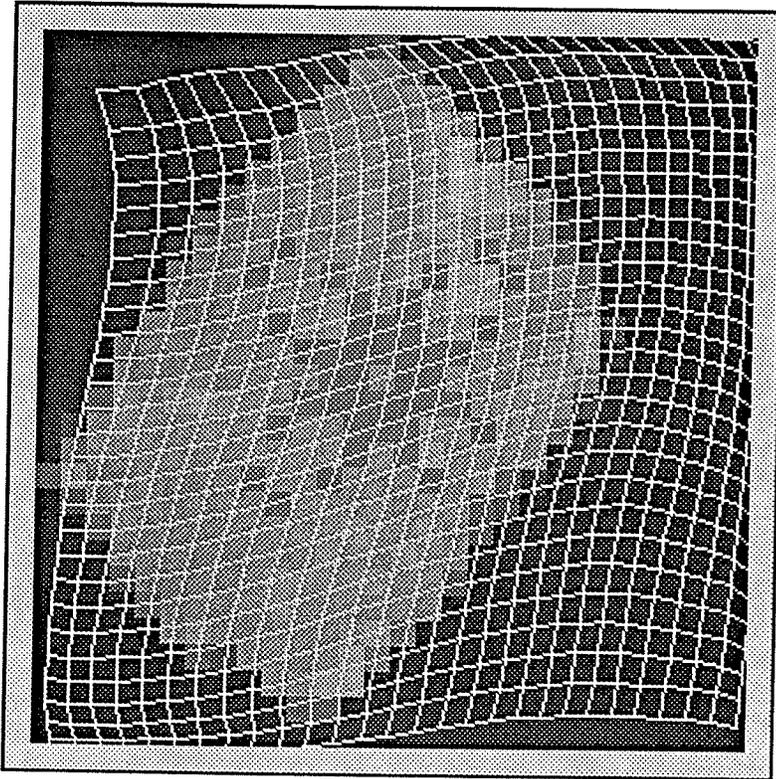


Fig. 4.5-4 contd.

(e)

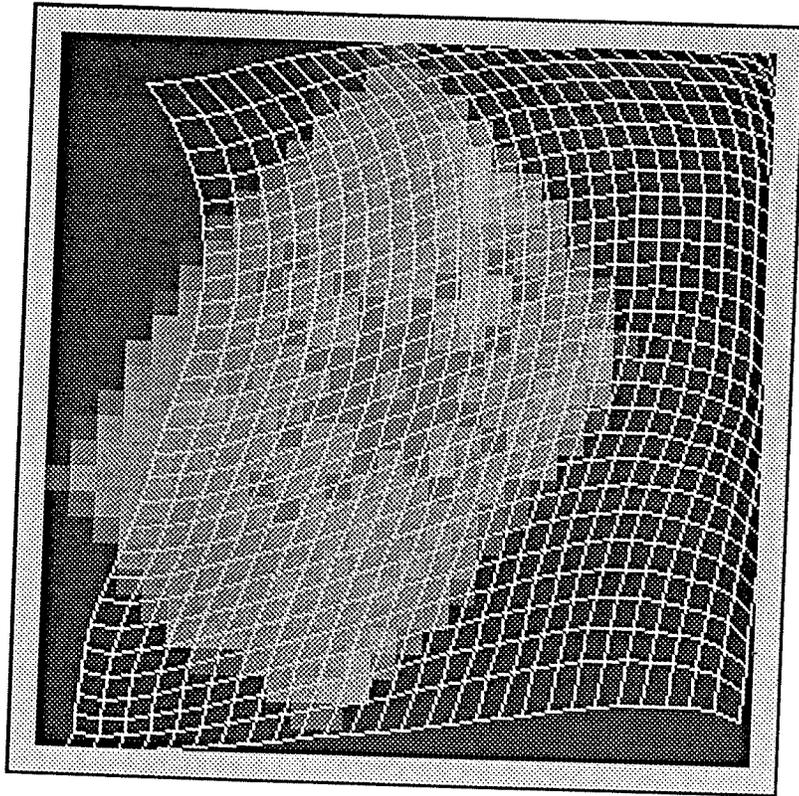


Fig. 4.5-4 contd.

(f)

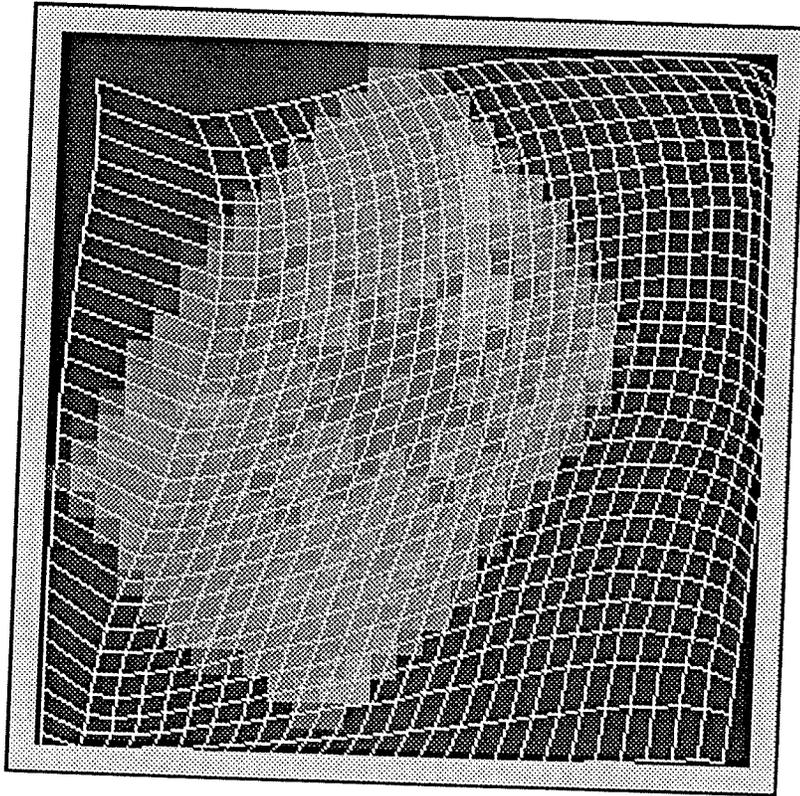


Fig. 4.5-4 contd.

(g)

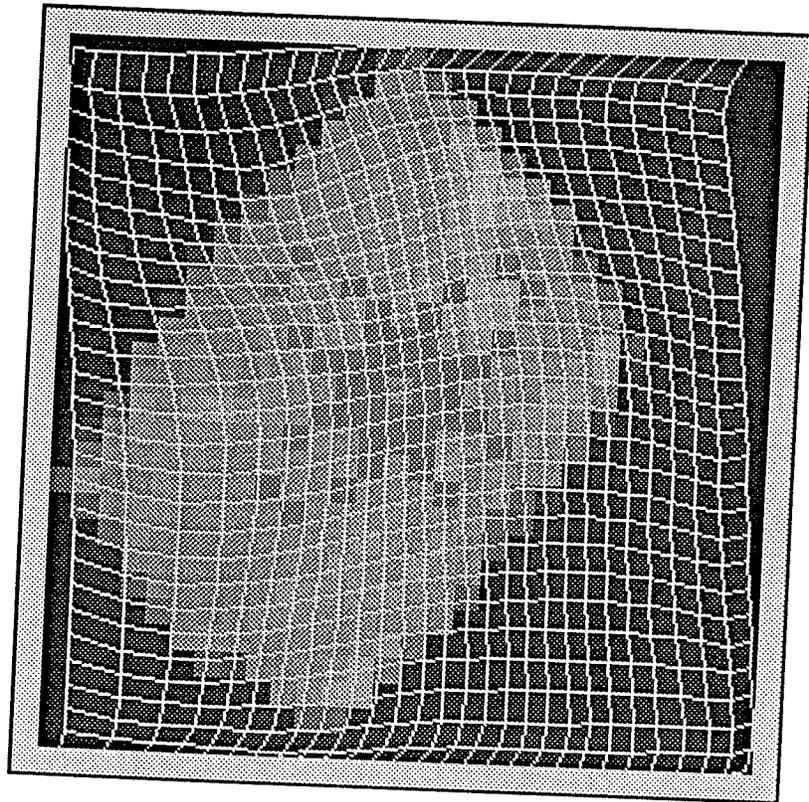


Fig. 4.5-4 contd.

(h)

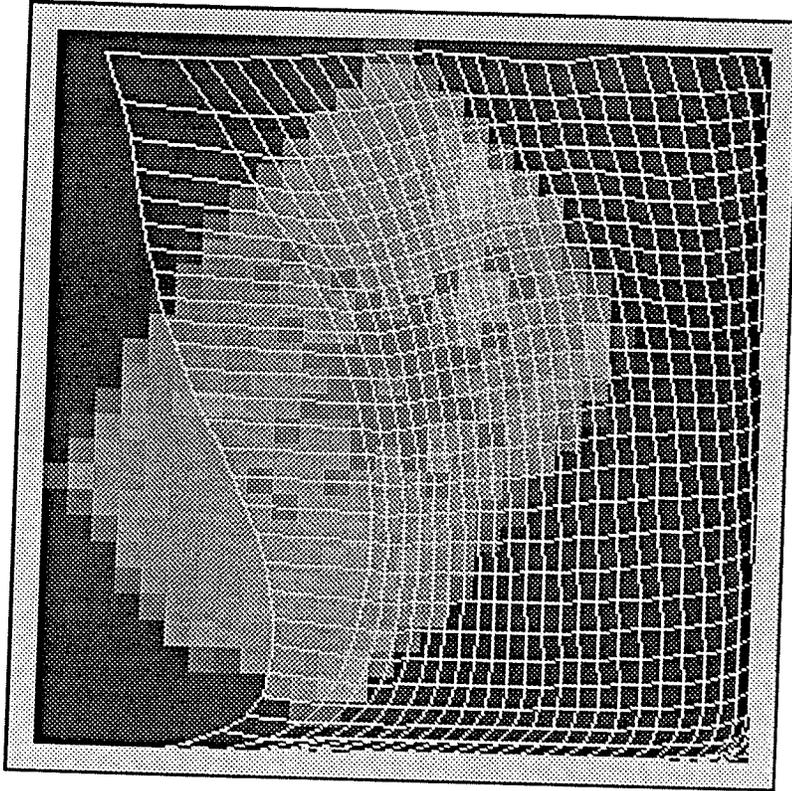


Fig. 4.5-4 contd.

(i)

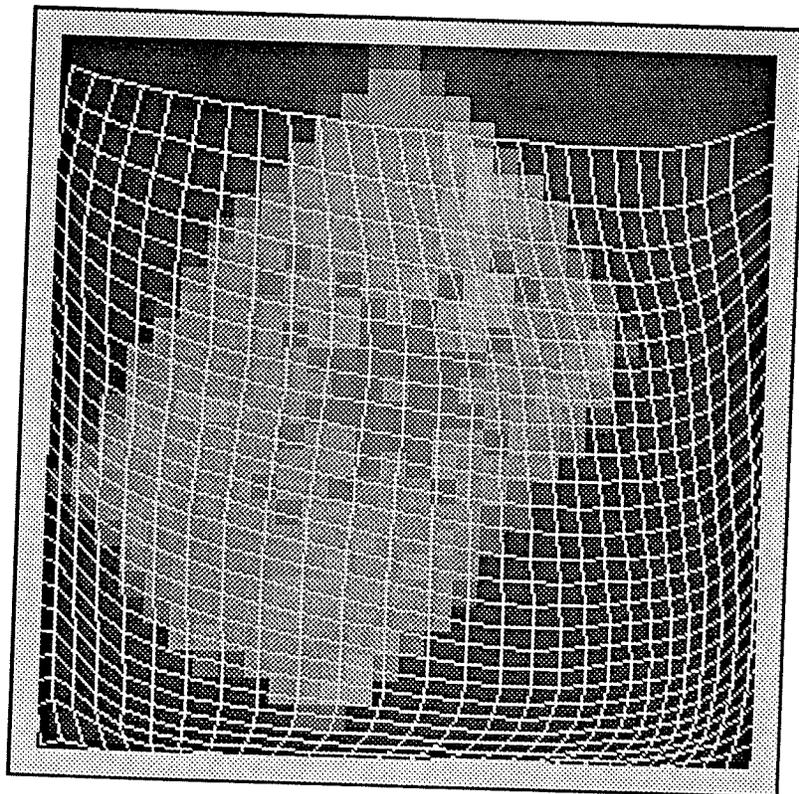


Fig. 4.5-4 contd.

(i)

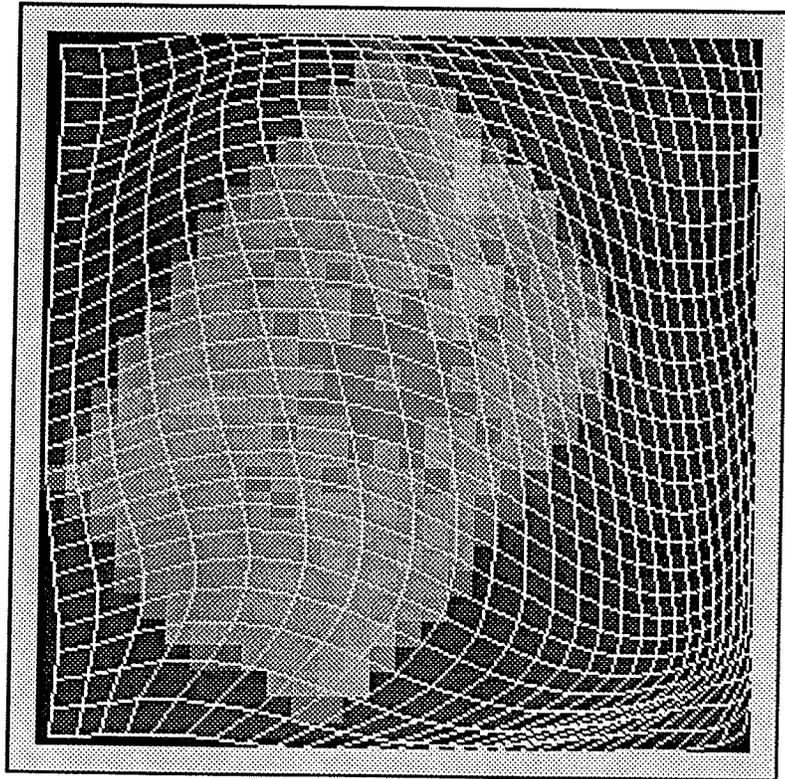


Fig. 4.5-4 contd.

(k)

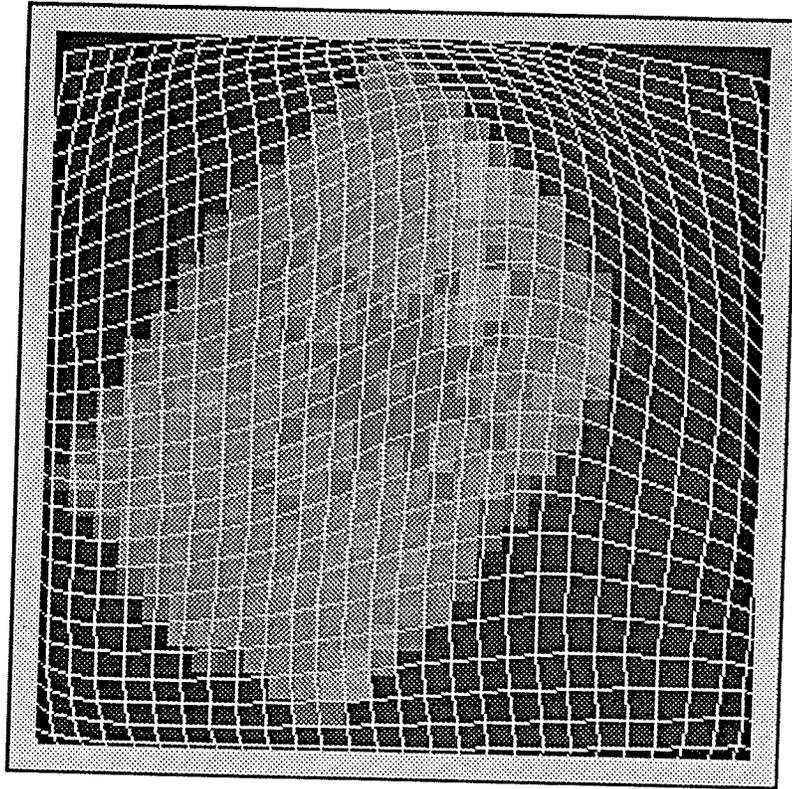


Fig. 4.5-4 contd.

(1)

### 4.5.3. Interpolation

In Eq. (4.3-4), we present a simple interpolation procedure that is applied to resample the image intensity (grey scale). Put simply, if, in the new configuration, a pixel is stretched over a surface of several pixels, its center designates a pixel from the temporary image that exclusively receives the intensity. However, when the centers of several pixels fall onto the surface of one pixel (from a temporary image), it takes on a new intensity value that is equal to the average intensity of all overlapping pixels. Fig. 4.5-5 shows schematically the interpolation.

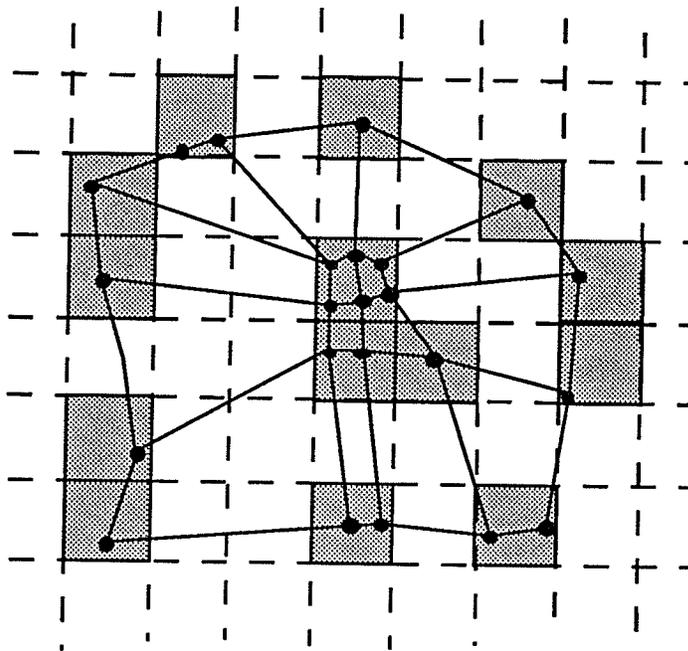


Fig. 4.5-5. The intensity interpolation. The image Correlation Technique adopts a simple way of the construction of the optimized image (temporary image) that is created at each iteration step. Cells of the optimized image grid that do not coincide with centers of the deformed pixel grid are left with the intensity zero (white cells). Others, that intersect one or more pixel centers acquire the average intensity of all interacting pixels (grey cells). The optimized image is then matched with the deformed one (correlation is calculated).

This kind of interpolation involves only the transformation of pixel coordinates and the averaging of the intensity. In other words, a center of each pixel that coincides with

the reference image grid is labeled by its initial coordinates, and may in general occupy different positions at different times. This approach is often called a *material description* or *Lagrangian formulation* [Little, 1973]. The intermediate image reflects this temporary spatial configuration of the reference image. One can argue that such an intensity interpolation is too primitive. It is, however, very computationally efficient. One has to remember that our procedure resamples each pixel of the temporary image thousands of times. Moreover, if incompressible structures are under investigation, there is no volume change and the error that arises from lack of the correction due to the change in pixel geometry is small. Of course there is a silent assumption that while a volume of matter represented by a pixel has been deformed its intensity remains unchanged (or radiological attenuation coefficients are the same). Bearing all that in mind, a compromise between speed and accuracy is not so drastic. This kind of interpolation also presents another advantage, i.e., it allows us to keep track of all pixel positions and pixel intensities for both pictures. As a result, at each stage these entries yield appropriate values for the correlation function  $C$  with little computation burden. Using the same interpolation there is still possible to improve it. As it is shown in Experiment 1, by increasing the sampling rate better accuracy can be achieved. However, if one chooses to use a more refined interpolation [Bookstein, 1990], [Hou, 1978], [Pratt, 1978], [Schoenberg, 1973], [Schumaker, 1981], [Stucki, 1979], [Yamaguchi, 1988] it should be just to display the final solution. Otherwise, the computation time would lengthen considerably.

# CHAPTER 5

## EXPERIMENTS

### 5.1. Experiment Descriptions

The ICT was successfully applied to estimate displacements in test images that had been deformed using exact mathematical formulas.

In the two following chapters, 5 and 6, optimization experiments are presented. Chapter 5 contains four of them: Exp. 1-4. These examples are experimental justification of the method.

In Chapter 6 a possible application to the field of radiology is given. Two experiments are presented there, Exp. 5 and 6. They are more complicated than the others and the displacement data resulting from the Image Correlation Technique are used for deriving a divergence field. The divergence fields, that provide diagnostic images, are presented there in a form of three-dimensional charts .

The definitions and assumptions given here are applicable to all experiments presented in both Chapters 5 and 6.

Figures and charts related to a particular experiment are marked by a capital letter E, followed by an experiment number and a current figure or a chart number.

We define the following functions and parameters.

The *difference function*, as the sum of absolute values of intensity differences of the deformed and optimized pictures at each pixel locations, .

The *acceptance ratio*, as the ratio of accepted number of steps and total number of trials for each temperature level;

The *average time*, as mean computing time between the two consecutive and accepted trials. (The mean time value is taken for each temperature level.).

From now on we will refer to the length of Markov chain  $L$  (defined by Eq. 3.3-19) as an *iteration constant* and will denote it  $C_T$ .

All experiments (Exp. 1-4) presented in this chapter, and one experiment in Chapter 6 (Exp. 5), were conducted using an Apple *Macintosh II* computer with co-processor and with 8 MB of RAM.

In Chapter 6, in Exp. 6, to correlate the image of a mastectomy sample, a faster, Apple *Macintosh II* based *Siemens LiteBox* computer was used.

We will present results in the following form: We start from a figure that displays the pair of reference and distorted configurations of the object (model) under consideration. In some instances the next figure will present a deformed pixel grid alone, in some other instances, the deformed reference grid of the reference image superimposed onto the deformed image. This way we are able to compare two images using one figure. The grid, an example of position representation, shows the positions of the centers of pixels of the reference image. Because an image that contains the grid is fairly transparent, we may superimpose it onto the deformed image displayed in gray scale. This is done without losing much detail or obscuring the deformed image. This way, it is convenient to observe and evaluate the optimization progress, when viewing it on a computer screen.

At the time of the convergence, in the same manner we may present a solution (the reference grid, after a considerable number of adjustments by the movement algorithm, positioned onto the deformed image).

We also provide three-dimensional charts that present displacements fields quantitatively in the  $x$  and  $y$  directions. The horizontal coordinates,  $x$  and  $y$ , are image coordinates and the third, vertical axis represents the displacement component for each image pixel. All units in these charts are pixels or fractions of pixels.

In addition, we present some charts of optimization functions to describe the process in a more exhaustive way. These are the correlation and the difference functions, the average computing time and the acceptance ratio. The most important is the correlation function (matching function) that in fact is a measure of similarity between the images. The correlation function is normalized and its maximum value (1000 in all experiments except Exp. 6 where 10000 is adopted) corresponds to a perfect image match (or convergence). It is also worth pointing out that, when the plateau of the correlation function is well established, the system is only a few percent or a fraction of percent from the convergence. The difference function is an additional measure of similarity. Due to the nonlinear character of the correlation function these two are not compatible. As a matter of fact, the difference function is itself incorporated into the correlation function (see Eq. (4.2-1)).

The acceptance ratio parameter and the average computing time are measures of the cooling rate. These are useful indicators in such cases as when the evolution of the system is halted at some 'strong' local maximum, or the optimization settings are too crude.

Each image is a discrete representation of a continuous intensity function that defines the object. This sampling of the continuous intensity function is a result of the imaging hardware. An original image, using the interpolation, can be enlarged or shrunk relative to an increase or a decrease of a sampling rate applied to the original image itself. Such sampling operations were done in the experiments 1, 5 and 6.

Each of the experiments represents the different stage of the algorithm development. For the clarity of presentation each experiment is separated in different sections. Starting from simple models the method was developed to accommodate more complicated

deformation instances. Altogether 144 experiments were conducted, lasting from several hours to days, and on one occasion several weeks.

The experiments that were chosen for this thesis are representative among many others. Taking into account the duration of each experiment, the overall effort to develop the method was considerable.

Before the experiment presentation, a Pascal procedure is listed that is the practical realization of the method.

## 5.2. Computer implementation of ICT

ICT was adopted to work in Apple's Macintosh environment.

The application was written in the *THINK Pascal* programming language designed by Symantec, Cupertino, CA, USA. Initially the 3.0 version was used that later was converted to version 4.0.

The coordinate matrices  $B_x$  and  $B_y$  have both double plane data structure (third dimension with two entries). These matrices are designed to store the coordinates of the reference image. Suppose that the first plane contains valid coordinates of the reference image, while the second hosts the coordinates from the present trial. If the trial is accepted, the second plane holds the valid reference coordinates. Only then is the first plane is discarded, to be available for the trial data from the next optimization trial.

The intensities (images) of the reference and distorted images are kept in matrices  $B_i$  and  $A_i$ , respectively.

The experimental matrix  $A_e$  is designed to hold the temporary image created at every iteration step. Matrixes  $A_e$  together with  $A_i$  are used to produce the value of the correlation.

Matrix  $A_n$  holds the interpolation constants  $w$  from Eq. 4.3-4.

All matrices described were implemented as dynamically allocated variables.

The following piece of the computer code, together with relevant declarations, is the implementation of ICT, that was used.

```

const
    MSX = 30;           { width of image in pixels }
    MSY = 30;           { height of image in pixels }
    CORRELATIONCONST = 1000;   { maximum value of correlation }
function }
    BACKGROUND = -1;   { intensity of border pixels }

type

    STN = (DISPLAY, INITIALIZATION, DOITERATION, PRINT, WAIT,
    CONVERGENCE);

    REALMATRIX = array[1..MSY, 1..MSX] of REAL;

    Real2Array = array[1..2] of REALMATRIX;

var
    Bi: REALMATRIX;     { Reference image, intensity matrix }
    Bx: Real2Array;     { Reference image, x coordinate matrix }
    By: Real2Array;     { Reference image, y coordinate matrix }

    Ai: REALMATRIX;     { Distorted image, intensity matrix }
    Ae: REALMATRIX;     { Temporary image, intensity matrix }
    An: REALMATRIX;     { Temporary matrix , coefficient 'w' see (4.3-4) }

    STAN: STN;          { Variable that shows state of the program }

    myPOINT: POINT;    { Auxiliary variable, to store pixel coordinates }
    FIXEDx, FIXEDy: REAL; { Auxiliary variable, to store pixel coordinates }

    XX, YY: REAL;      { Auxiliary variable }
    XXX, YYY: REAL;
    dXX, dYY: REAL;
    DrX, DrY: REAL;
    Rx, Ry: REAL;

```

DISTANCE: REAL;  
REVERSE: REAL;  
RADIUS: REAL;

FLEXIBILITY: REAL;        {Flexibility limit  $F$  }  
FLEXi: REAL;                {Flexibility variable}

R: REAL;                    {Probability variable}  
PT: REAL;                  {Probability variable  $P$ , see Eq. 4.3-6}

ROWS, COLUMNS: INTEGER;        {Image limits}  
ROWS\_1: INTEGER;                {ROWS\_1=ROWS-1;  
COLUMNS\_1=COLUMNS-1}  
ROWS\_1,: INTEGER;

COUNT: LONGINT;            { $k$ , total counter}  
TCOUNT: LONGINT;         {Temporary counter, on each 'temperature' level}  
TCONST: LONGINT;         { $C_T$ , predetermined number of steps on each  
'temperature' level}

TT: REAL;                  {Temperature variable}  
DELTA: REAL;                {Cooling control variable  $a$ , see Eq. 4.4-2}

I, J: INTEGER;              {Seed pixel coordinates}  
DI, DJ: INTEGER;            {Direction of a move}

K, KK: INTEGER;            {Auxiliary variables}  
TEMPEL1, TEMPEL2: REAL;  
M2: LONGINT;  
CCN: INTEGER;  
I\_2: REAL;  
DIFFEL: REAL;

CORRELATION: REAL;        {Correlation variable}  
LASTCORR: REAL;         {Correlation from previous 'temperature' level}  
MaxCorrelation: REAL;    {Normalization factor}  
OldCorrelation: REAL;    {Correlation from previous step}  
DCorrelation: REAL;      {Correlation increment}  
AVERAGECORR: REAL;     {Averaged correlation, on each 'temperature' level}

{ A pseudonumber generator with even distribution on the (0,1) interval. }

**procedure** RND (var RR: REAL);  
EXTERNAL;

{The control procedure to determine the state of the application}

**procedure** STATE (STAN: STN);  
EXTERNAL;

**procedure** ANNEA2;  
var

```

m, n: integer;                                {loop control variables}

begin
  repeat
    repeat                                     {selection of a seed pixel - x
                                             coordinate}
      RND(R);
      I := round(rows_1 * R) + 1;
    until (I > 1) and (I < ROWS);

    repeat                                     {selection of a seed pixel - y
                                             coordinate}
      RND(R);
      J := round(COLUMNS_1 * R) + 1;
    until (J > 1) and (J < COLUMNS);

    repeat                                     {direction to move in}
      RND(R);
      dI := round((0.5 - R) * 2);
      RND(R);
      dJ := round((0.5 - R) * 2);
    until (DI <> 0) or (DJ <> 0);

    RND(R);                                     {selection of constants  $R_x$  and  $R_y$ }
    Rx := R;
    RND(R);
    Ry := R;

    RND(R);                                     {flexibility selection}
    FLEXi := sqrt(FLEXIBILITY + ROWS * R);

    FIXEDx := Bx[KK, I, J];
    FIXEDy := By[KK, I, J];

  { setting temporary matrices}

  M := 1;                                       {UPER ROW}
  for N := 1 to COLUMNS do
    begin
      Ae[M, N] := BACKGROUND;
      An[M, N] := 1;
    end;
  M := ROWS;                                    {LOWER ROW}
  for N := 1 to COLUMNS do
    begin
      Ae[M, N] := BACKGROUND;
      An[M, N] := 1;
    end;

  N := 1;                                       {LEFT SIDE COLUMN}
  for M := 2 to ROWS_1 do
    begin

```

```

        Ae[M, N] := BACKGROUND;
        An[M, N] := 1;
    end;
N := COLUMNS;                                {RIGHT SIDE COLUMN}
for M := 2 to ROWS_1 do
    begin
        Ae[M, N] := BACKGROUND;
        An[M, N] := 1;
    end;
{ NEW CONFIGURATION FOR ALL EXCEPT BORDER PIXELS }
for M := 2 to ROWS_1 do
    for N := 2 to COLUMNS_1 do
        begin
            XXX := Bx[KK, M, N];
            YYY := By[KK, M, N];
            DrX := (XXX - FIXEDx);
            DrY := (YYY - FIXEDy);
            RADIUS := (SQR(DrX) + SQR(DrY)) / FLEXi;
            REVERSE := 1 / EXP(RADIUS);

            dXX := (Bx[KK, M + DI, N + DJ] - XXX) * Rx * REVERSE;
            dYY := (By[KK, M + DI, N + DJ] - YYY) * Ry * REVERSE;
            XX := XXX + dXX;
            YY := YYY + dYY;
            Bx[k, M, N] := XX;
            By[k, M, N] := YY;

            myPOINT.H := ROUND(XX);
            myPOINT.V := ROUND(YY);

            Ae[myPOINT.V, myPOINT.H] := Ae[myPOINT.V, myPOINT.H]
+ Bi[M, N];
            An[myPOINT.V, myPOINT.H] := An[myPOINT.V, myPOINT.H]
+ 1;

        end; {M,N}
    end;
CORRELATION := 0;
for M := 1 to ROWS do                            {THE NEW CORRELATION}
    for N := 1 to COLUMNS do
        if Ae[M, N] <> 0 then
            begin
                TEMPEL1 := Ai[M, N];
                TEMPEL2 := Ae[m, n] / An[M, N];
                Ae[m, n] := TEMPEL2;
                I_2 := TEMPEL1 * TEMPEL2;
                DIFFEL := ABS(TEMPEL1 - TEMPEL2);
                CORRELATION := CORRELATION + I_2 / (1 + ln(DIFFEL
+ 1));
                Ae[M, N] := 0;
            end;
        end;
    end;
end;

```

```

        An[M, N] := 0;
    end;

    CORRELATION := CORRELATION * MaxCorrelation;
    DCorrelation := OldCorrelation - CORRELATION;

    {BOLTZMANN DECISION ALGORITHM, SEE (4.3-4)}

    if DCorrelation > 0 then
        begin
            PT := EXP(-DCorrelation / TT);
            RND(R);
            M2 := M2 + 1;
            if (R < PT) then
                begin
                    COUNT := COUNT + 1;
                    TCOUNT := TCOUNT + 1;
                    CCN := KK;
                    kk := k;
                    K := CCN;
                    OldCorrelation := CORRELATION;

                    end; {IF (R / RM) < PT}
                end
            else
                begin
                    COUNT := COUNT + 1;
                    TCOUNT := TCOUNT + 1;
                    CCN := KK;
                    kk := k;
                    K := CCN;
                    OldCorrelation := CORRELATION;
                end;
            end;

    {COOLING SCHEDULE, SEE (4.4-2) }

    if TCOUNT = TCONST then
        begin
            AVERAGECORR := AVERAGECORR / TCOUNT;
            if AVERAGECORR > LASTCORR then
                TT := TT * (1 - DELTA);
                LASTCORR := AVERAGECORR;
                TCOUNT := 0;
                AVERAGECORR := 0;
                STATE(STAN);
            end;
        until STAN in [DISPLAY, PRINT, WAIT, CONVERGENCE];

    end; {ANNEA}

```

### 5.3. Experiment 1

An object that depicts a single hill pattern on a ten by ten pixel base (Fig. E.1-1), is deformed in the following way:

$$\begin{aligned} X^* &= X + \sin \left[ \pi \frac{X}{M/2} \right] \\ Y^* &= Y + \sin \left[ \pi \frac{Y}{M/2} \right] \end{aligned} \tag{5.3-1}$$

This is a non-linear deformation in both directions with the maximal amplitude equal to 1 pixel. The period of the sine functions is the same for both directions and is equal to  $M$  (image size). We may describe the deformation as a stress action from both right and left sides ( $x$  component), and at the same time, from the top and bottom sides ( $y$  component), forcing the image to collapse inside. Fig. E.1-2 shows the reference grid deformed by (5.3-1) and superimposed on the deformation image.

Actually, to increase the accuracy during the optimization process, the object was represented by a 20 by 20 pixel matrix i.e. (one original pixel was represented by a four pixels of the same gray level). This represents sampling rate twice of that of the original image. The double rim of dots (Fig. E.1-2) comes from the initial border pixels after the sampling rate increased. These dot pixels do not contribute to the correlation and the outer rim remains at its original position; however, the inner, dotted rim of border pixels is allowed to participate in the grid movement.

Fig. E.1-3 shows the solution configuration. The reference grid is matched with the deformed image.

The simulated deformation shows displacement in the  $x$  direction, Fig. E.1-4a, and in the  $y$  direction, Fig. E.1-4b. Here, the displacements for dotted border pixels are shown to

be zero. The recovered deformation in both  $x$  and  $y$  directions is shown in Fig. E.1-5a and Fig. E.1-5b, respectively. The recovery is smooth. The shape of the recovered deformation is surprisingly close to the original one. The error of the recovered displacements is presented in the next two charts E.1-6a and E.1-6b. The error range in the  $x$  direction is  $(-0.31,+0.2)$ , and  $(-0.36,+0.16)$  in the  $y$  direction. These values are less than the sampling range, that is, less than half of the pixel size (0.5).

In the course of the optimization the cooling schedule (4.4-1) was adopted with the following initial settings:  $\Delta = 0.1$  and  $\sigma = 5$ . The flexibility limit was set to  $F = 2$  and the iteration constant was set to  $CT = 50$ . For comparison, charts E.1-7, E.1-8 and E.1-9 show the correlation function together with the difference function. From these charts, it can be seen that the correlation function (similarity measure) is not compatible with the difference function, except in the case of the convergence. The optimization process converged when the temperature like parameter  $T$  was reduced to around 1.5, after approximately 2100 iteration steps and within about nine hours of computing time.

This experiment is characterized by the slow increase of the correlation function. At one point of the optimization, a steep increase of the correlation function was observed. This jump was from about the 40% level to a little below 100%. Not long after that, full convergence was recorded.

The acceptance ratio and computing time per accepted test are presented together in the chart E.1-10. Right from the beginning, the acceptance ratio falls quickly below 0.1 and for the most part of the process stays approximately at the 0.01 level. It stayed there even after convergence had occurred. The explanation to this lies in the image sampling; there is some room for the movement algorithm to move the reference grid without change in the correlation. By analogy to thermodynamics, the situation can be compared to energy states that are degenerated.

The same chart shows also that within the convergence range, on the average, an accepted step was recorded every 20 seconds.

The following two charts, E.1-11 and E.1-12, depict not one but three independent experiments performed using the same model that was described at the beginning of this section. They differ only in different settings for the iteration constant  $C_T$ . Experiments with  $C_T = 25$  and  $C_T = 50$  are almost indistinguishable as far as the correlation is concerned (E.1-11). The only difference is the different cooling rate (E.1-12). However, when  $C_T = 100$ , the correlation curve takes a different shape; it climbs to the convergence value in a more steady fashion and without any significant jumps. It also starts to converge earlier as far as the temperature-like factor is concerned. This means easier convergence. But, in terms of the optimization time, larger values of  $C_T$  lengthen the optimization process.

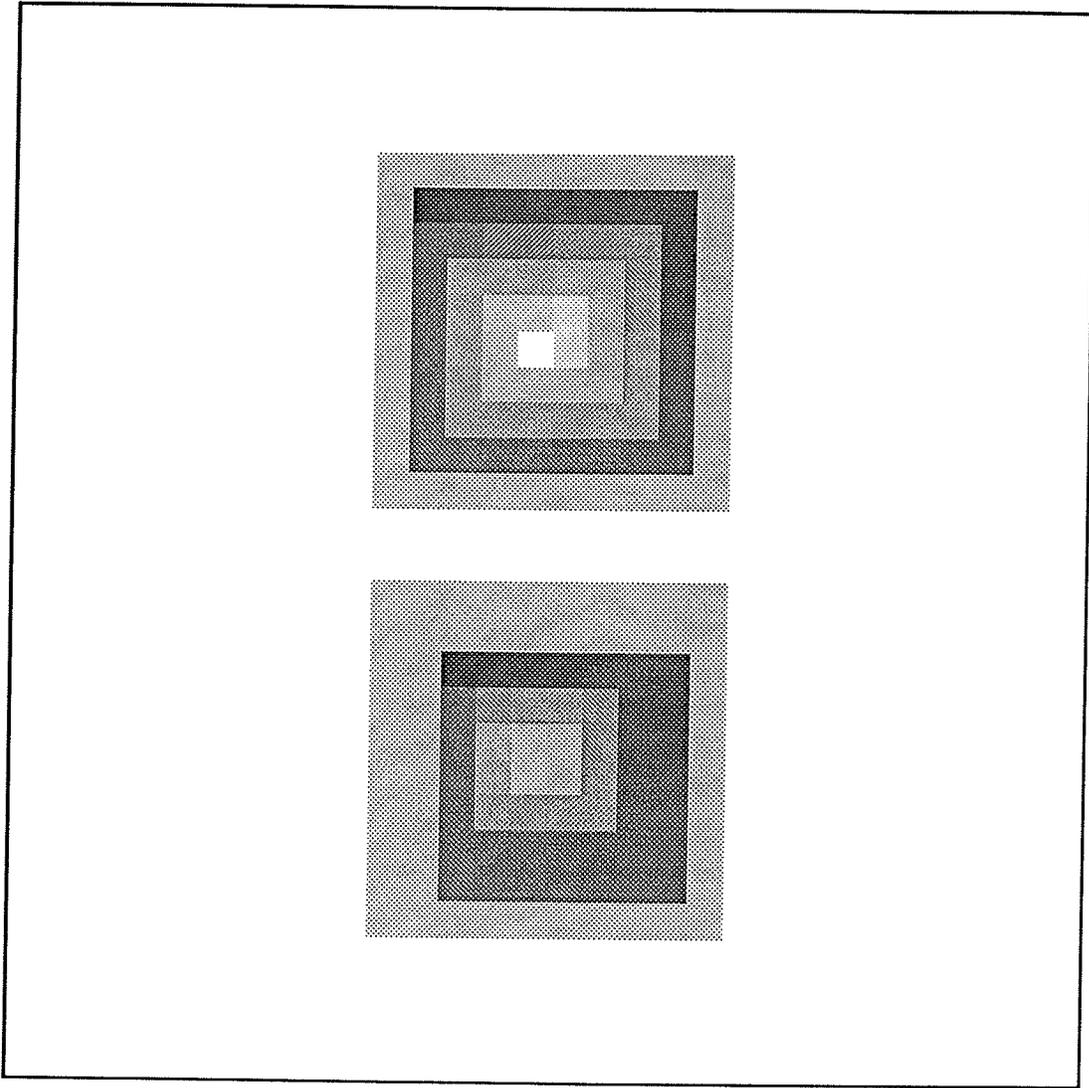


Fig. E.1-1. The hill object under investigation. In the upper part is the undeformed image (reference configuration); lower part shows the distorted image using Eq. (5.3-1).

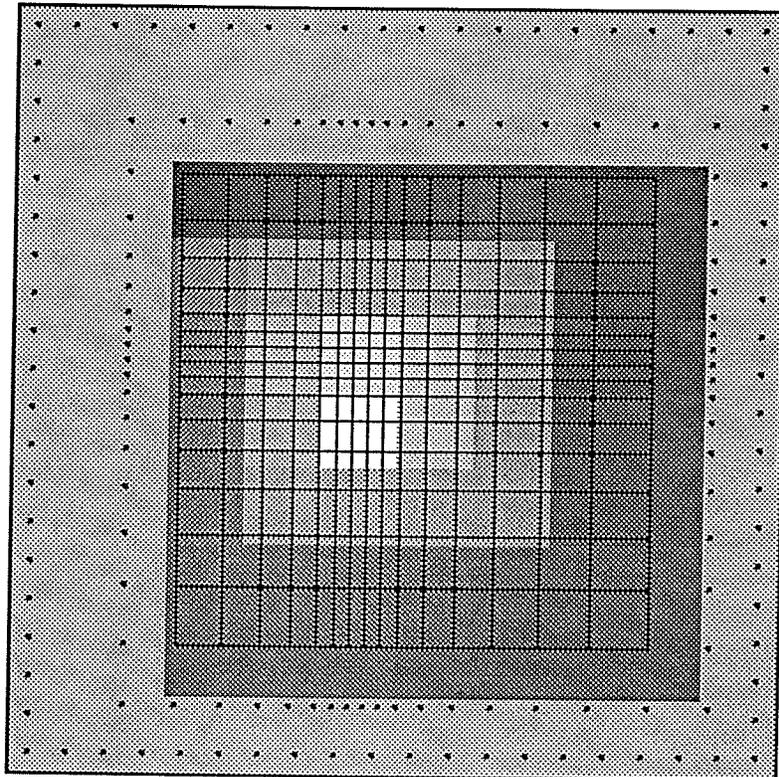


Fig. E.1-2. Distorted by (5.3-1), the reference image grid is superimposed onto the distorted image.

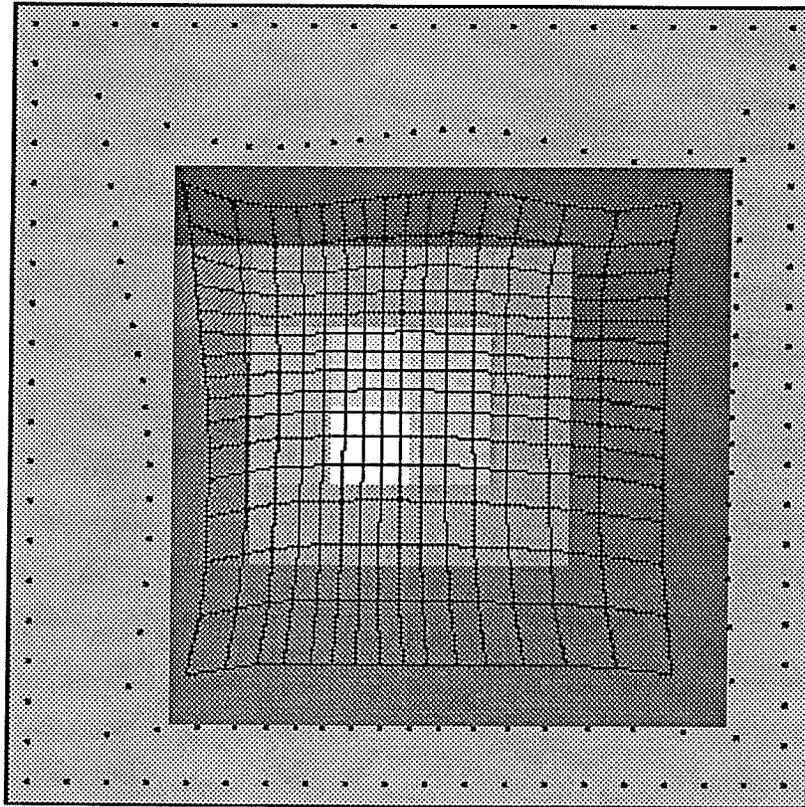


Fig. E.1-3. The solution image. The reference image represented by the grid, matched with the deformed picture presented in gray levels.

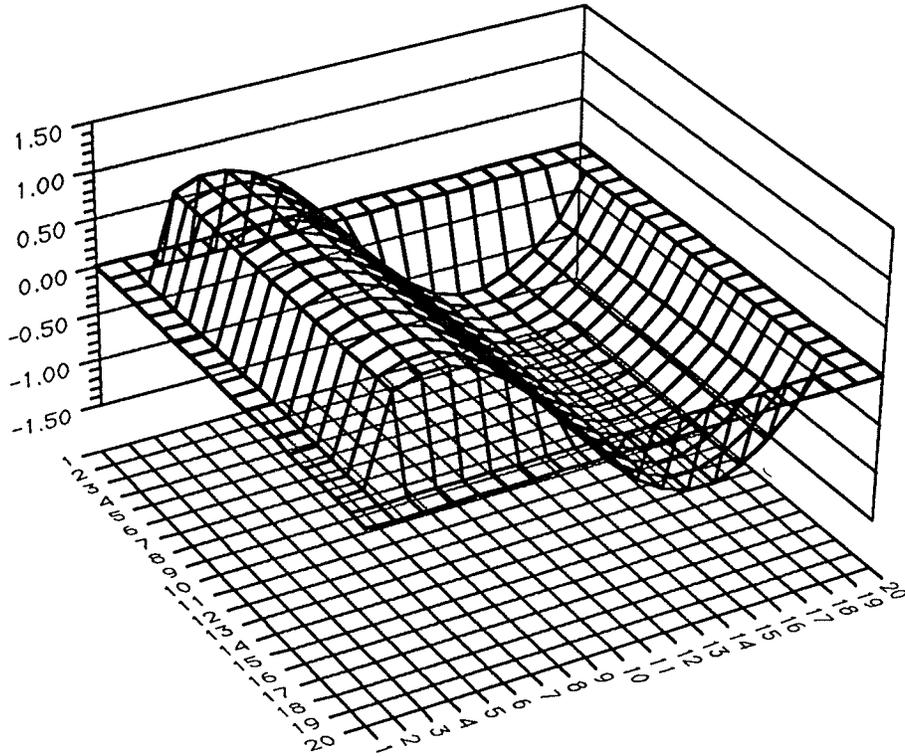


Fig. E.1-4a. The *hill* displacement field simulated by Eq. (5.3-1) in the  $x$  direction.

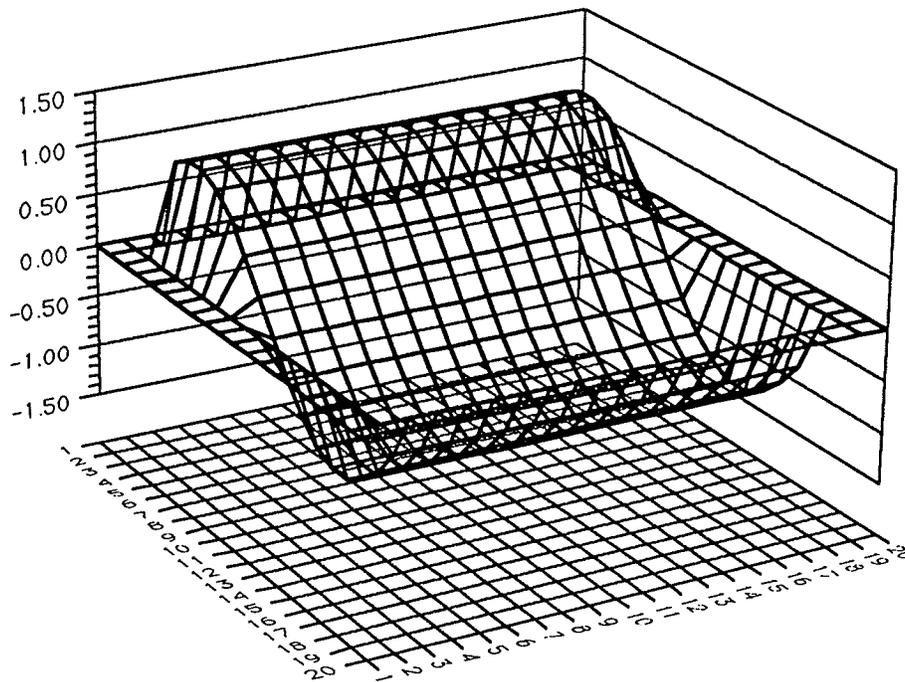


Fig. E.1-4b. The *hill* displacement field simulated by Eq. (5.3-1) in the y direction.

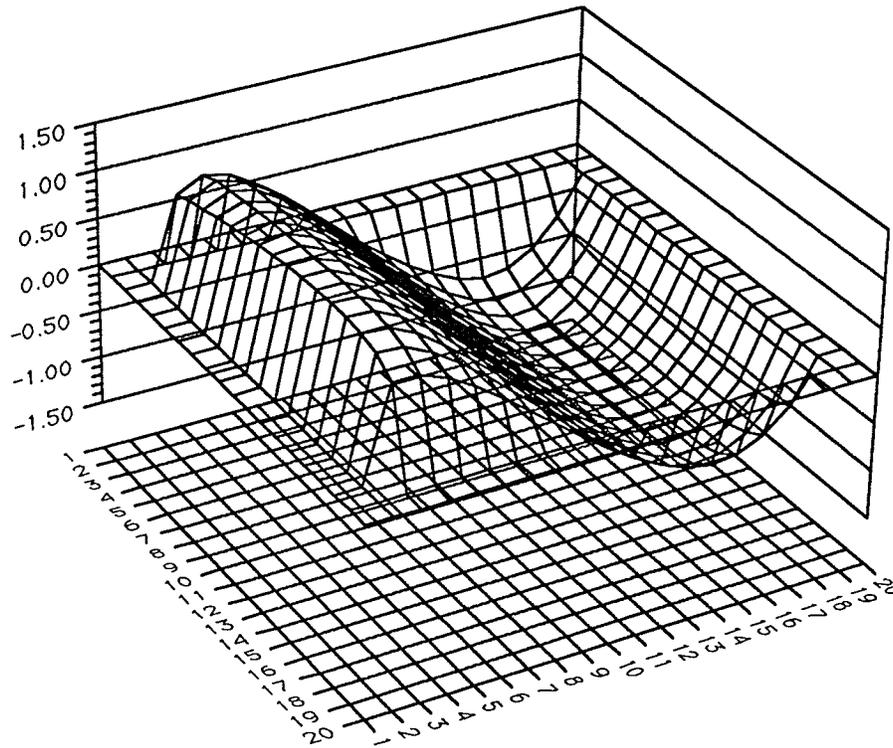


Fig. E.1-5a. The displacement field recovered from Image Correlation Technique in the  $x$  direction.

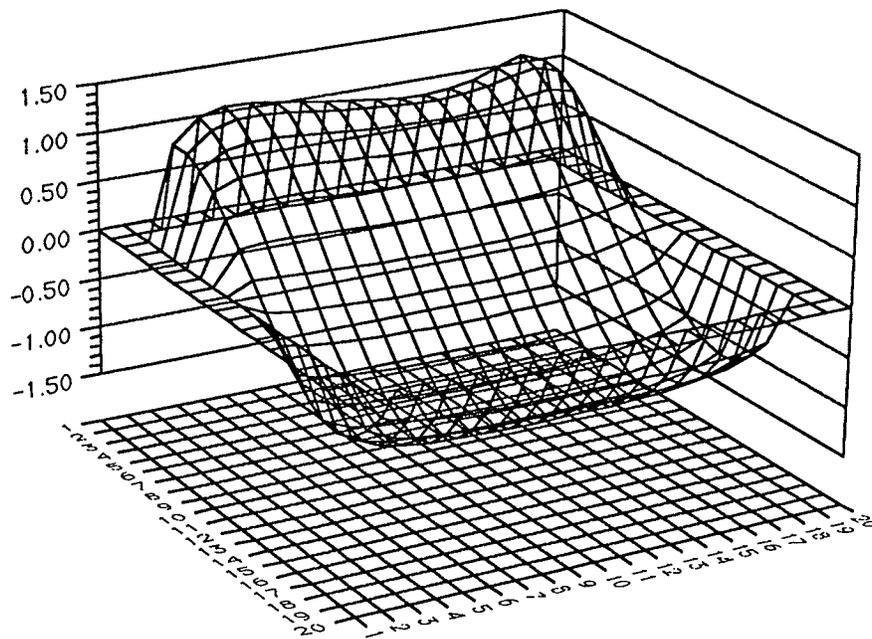


Fig. E.1-5b. The displacement field recovered from Image Correlation Technique in the y direction.

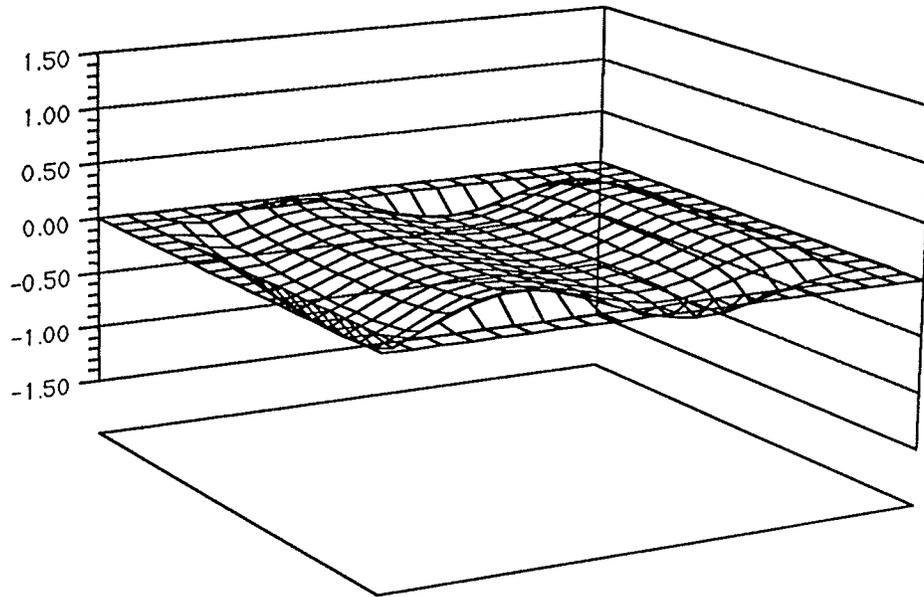


Fig. E.1-6a. The difference between the simulated displacement and that recovered from Image Correlation Technique, in the  $x$  direction.

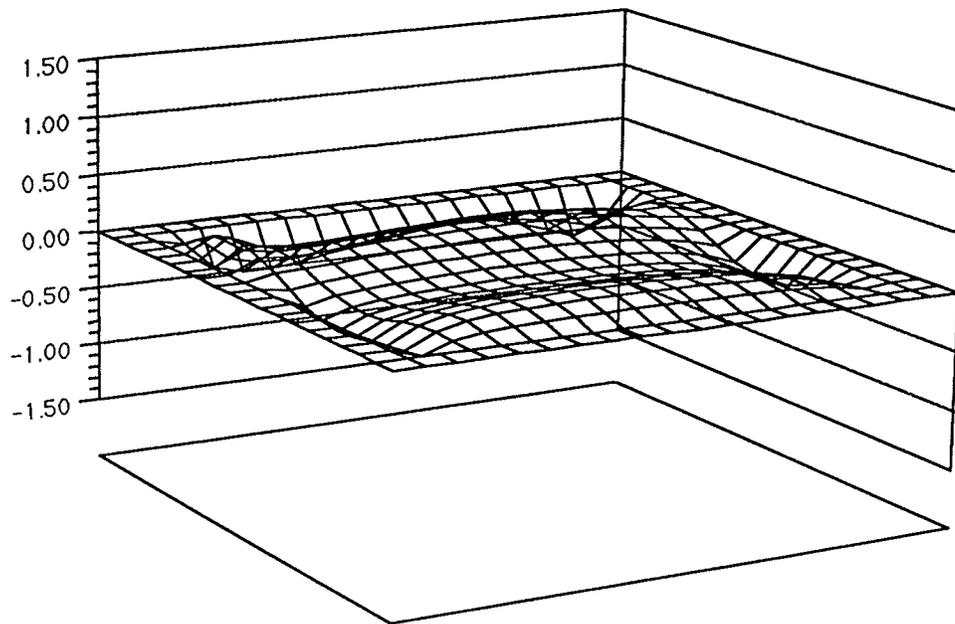


Fig. E.1-6b. The difference between the simulated displacement and that recovered from Image Correlation Technique, in the y direction.

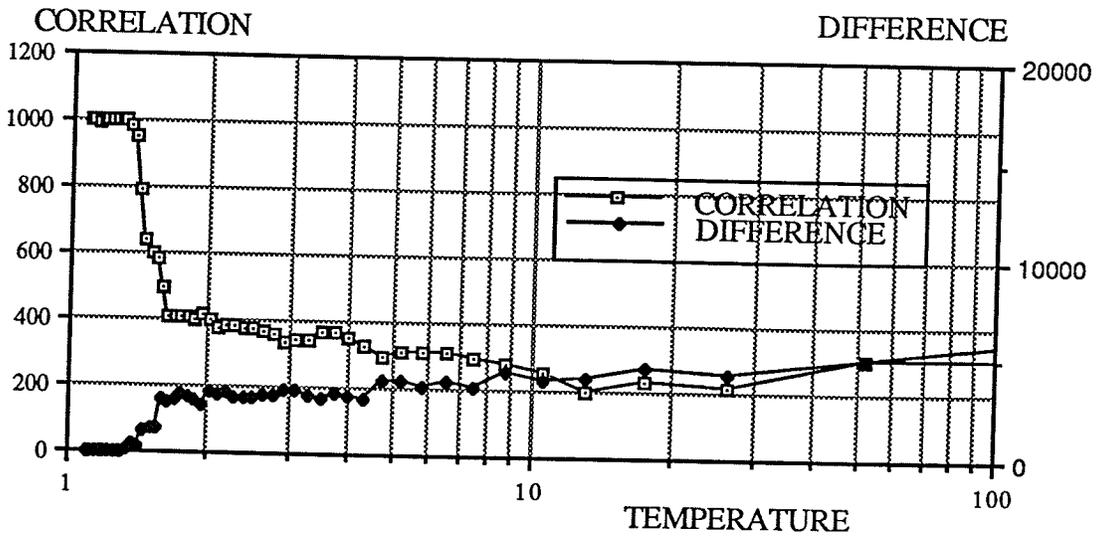


Fig. E.1-7. Image optimization for the *hill* model. The correlation and the difference versus the temperature-like parameter  $T$ .

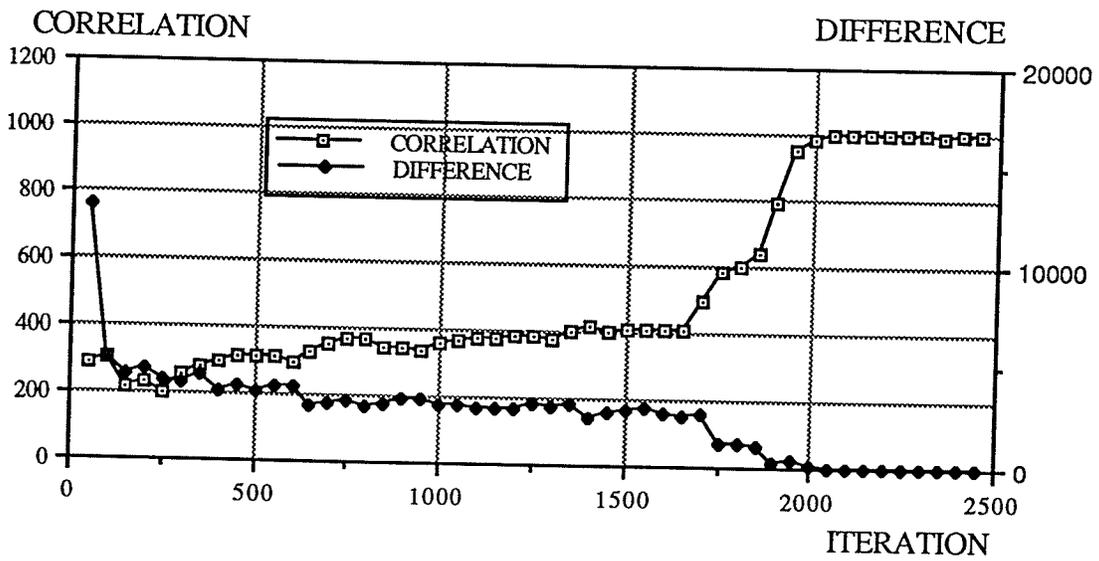


Fig. E.1-8. Image optimization for the *hill* model. The correlation and the difference versus iteration.

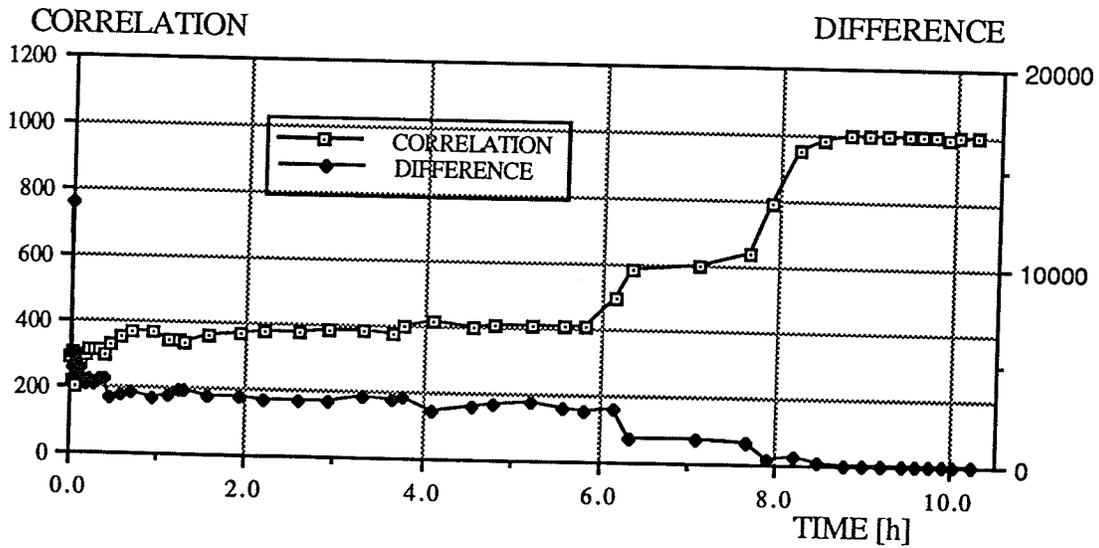


Fig. E.1-9. Image optimization for the *hill* model. The correlation and the difference versus the optimization time.

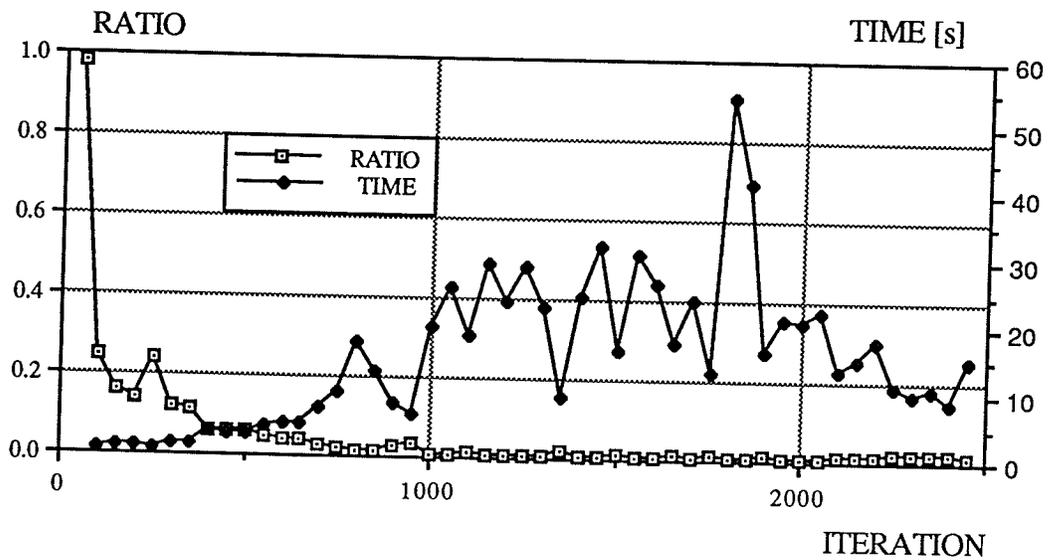


Fig. E.1-10. Image optimization for the *hill* model. The acceptance ratio and the average computation time (elapsed between two consecutive, accepted trials) versus iteration.

## CORRELATION

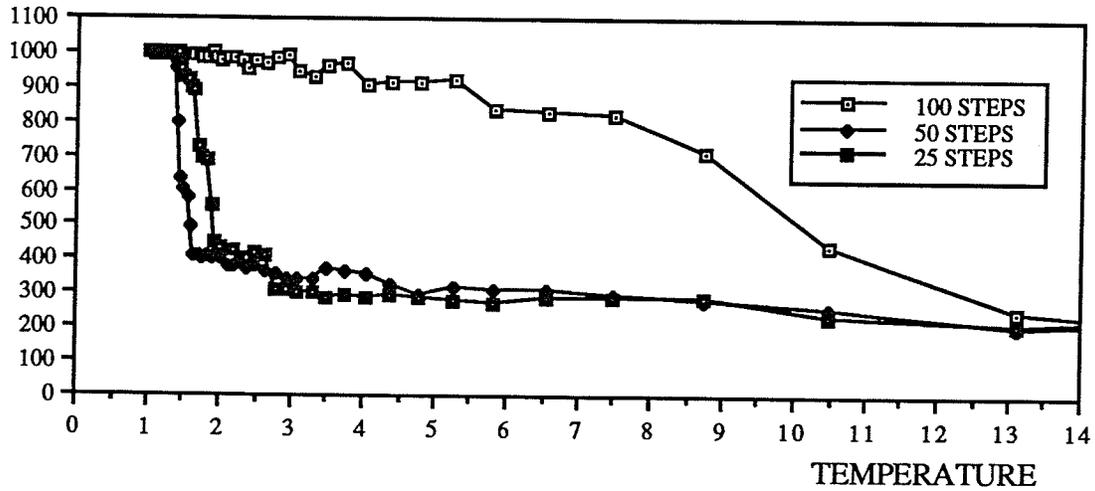


Fig. E.1-11. Image optimization for the *hill* model. Image optimization functions for different fixed numbers of accepted steps ( $C_T$ ) that are required at each "temperature" level. The correlation versus the *temperature-like parameter*  $T$ .

## TEMPERATURE

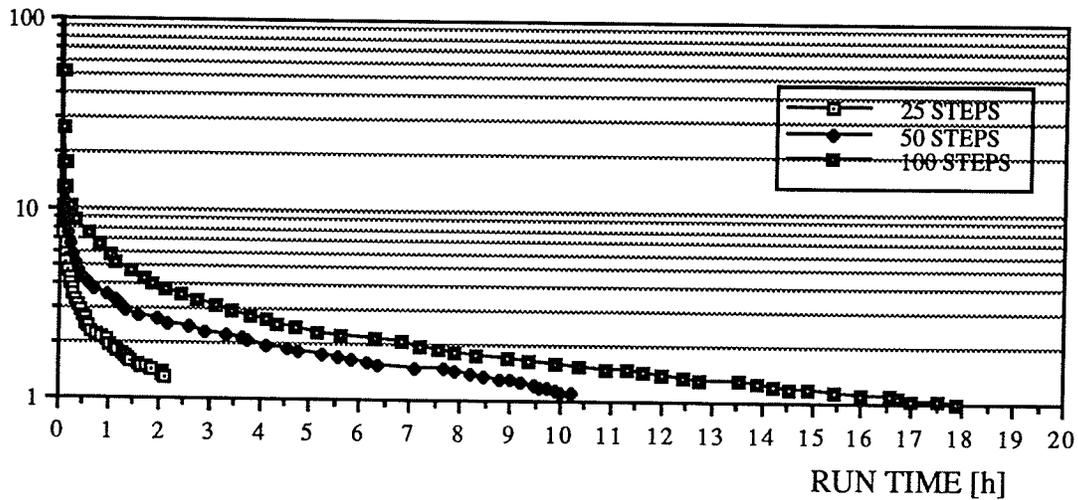


Fig. E.1-12. Image optimization for the *hill* model. Image optimization functions for different fixed numbers of accepted steps ( $C_T$ ) that are required at each "temperature" level. The *temperature-like parameter*  $T$  versus computing time.

## 5.4. Experiment 2

In this experiment a different object is considered. Fig. E.2-1 depicts a human face on a 16 by 16 pixel matrix. The deformation that was applied to the *face* is the same deformation (5.3-1) that was used in Exp. 1. Therefore, the deformation description from the previous experiment remains relevant. The only difference is in the applied sampling; in the present experiment sampling was the same as of the optimized images - equal to one.

Fig. E.2-1 shows the *face* in the reference and the deformed forms. The white stripes are zero value pixels that were left untouched by the interpolation (4.3-4), to produce the deformation. The sine function is within the interval  $(0, M)$  and the only reason why these stripes are not distributed symmetrically about the center of the image is because the first row and the first column of the image data are assigned coordinates equal to one, and as the border pixels by assumption, they are not considered in the deformation. (They are shown as the gray space in the deformed image.)

The displacement field resulting from the deformation is displayed in the images. E.2-3a, and E.2-3b for  $x$  and  $y$  direction, respectively.

The solution in the visual form, the superposition of the two images, is presented in image E.2-2.

The solution in the form of the displacement functions is depicted for both  $x$  and  $y$  directions in charts E.2-4a and E.2-4b. By comparing the simulated deformation with the recovered image, it is evident that the character of the deformation has been reconstructed. However, when the error charts E.2-5a and E.2-5a are compared with similar ones of Exp. 1, the advantage of using the optimization with the higher sampling rate is evident. The charts for experiments show greater variation of the error. This is also reflected in the error range itself:  $(-0.46, 0.27)$  and  $(-0.53, 0.38)$  in the  $x$  and  $y$  directions, respectively.

Optimization was carried out by applying the following set of controls:  $\Delta = 0.4$ ,  $\sigma = 1$ ,  $F = 2$ ,  $C_T = 10$ .

Figure E.2-6 shows the averaged correlation and the difference functions versus the computing time. The convergence occurred within the first hour of the application run. The process was characterized by the steep climbing of the correlation, right from beginning, that slowed down before it reached the convergence plateau, where it stayed.

Figure E.2-7 shows the acceptance ratio and the averaged time required to get an accepted move.

In the chart E.2-8 the cooling schedule is presented.

Figs. E.2-9 and E.2-10 display the *face* and the result of its optimization. The only difference, compared to Figs. E.2-1 and E.2-2, is that this time image is on a 32 by 32 pixel matrix.

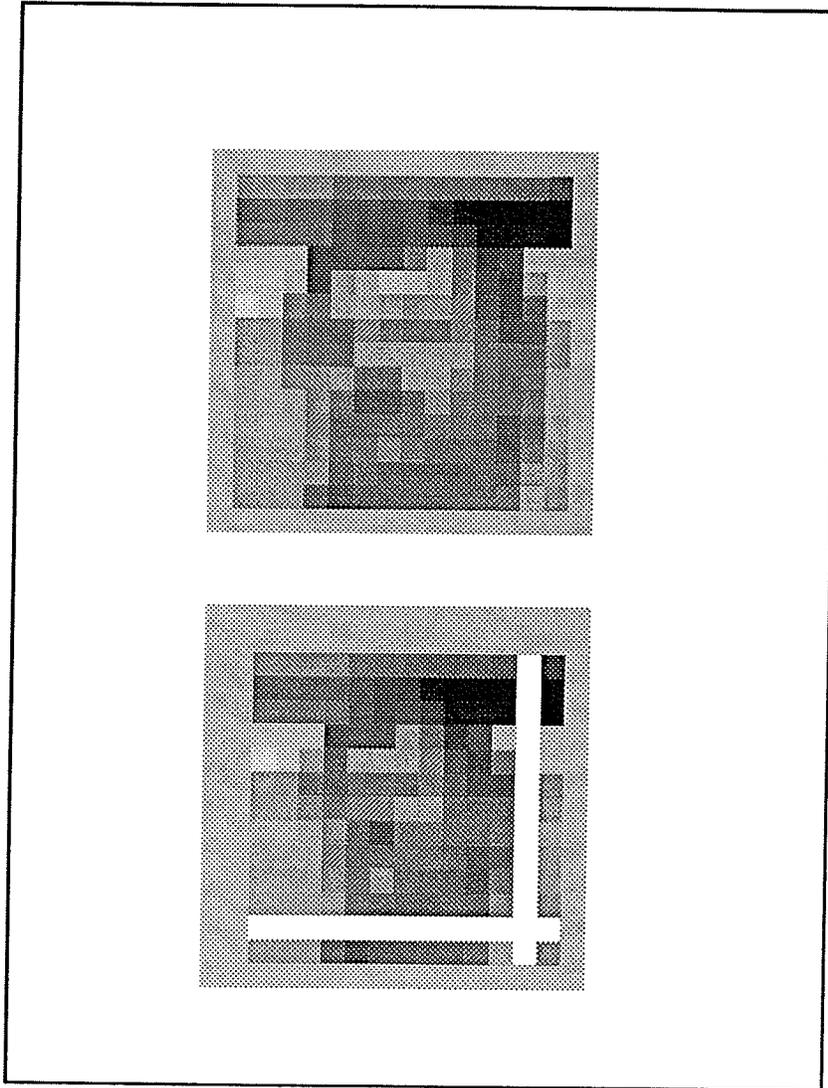


Fig. E.2-1. The *face* under optimization. In the upper part is the undeformed image (reference configuration), in the lower part the distorted image using Eq. (5.3-1). (These are best seen viewed from a distance.)

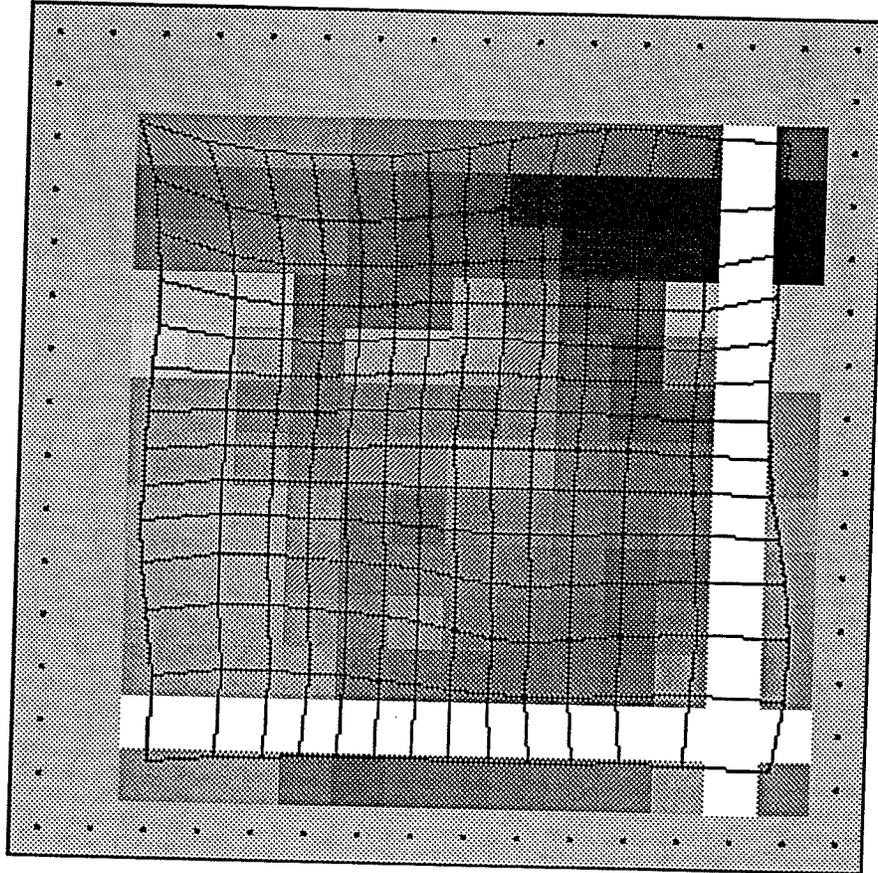


Fig. E.2-2. Distorted by (5.3-1), the reference image grid is superimposed onto the distorted image.

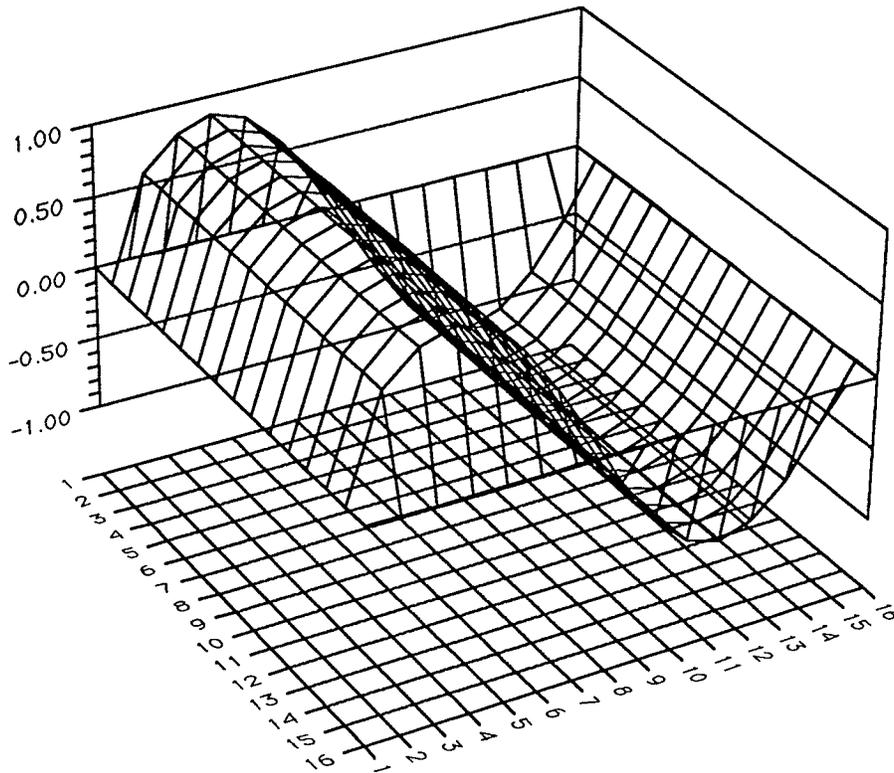


Fig. E.2-3a. The *face* displacement field simulated by Eq. (5.3-1) in the *x* direction.

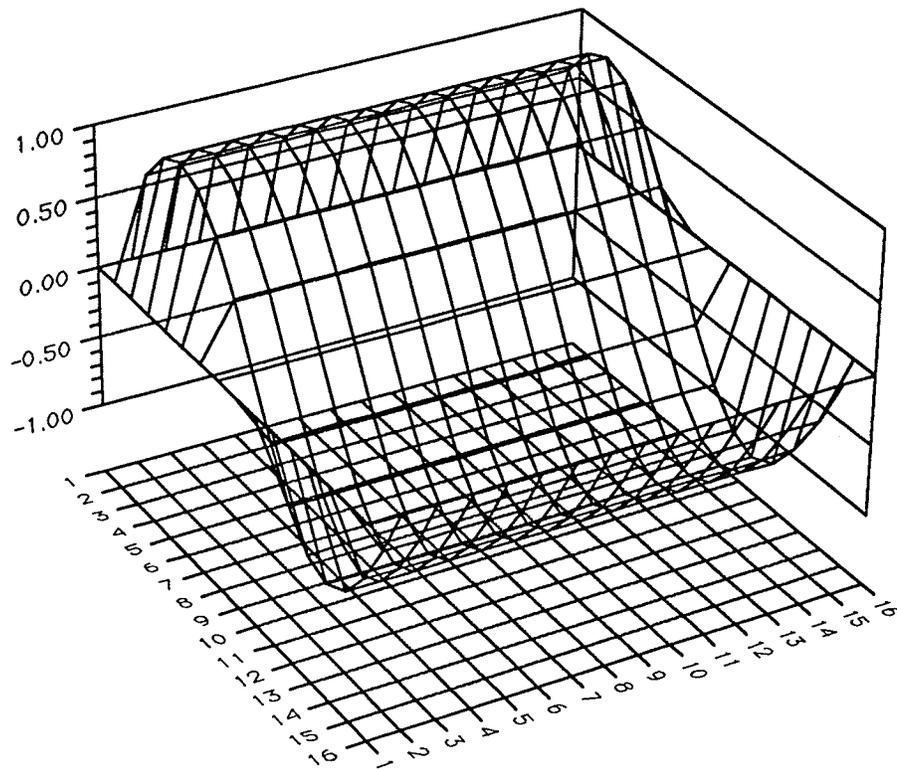


Fig. E.2-3b. The *face* displacement field simulated by Eq. (5.3-1) in the *y* direction.

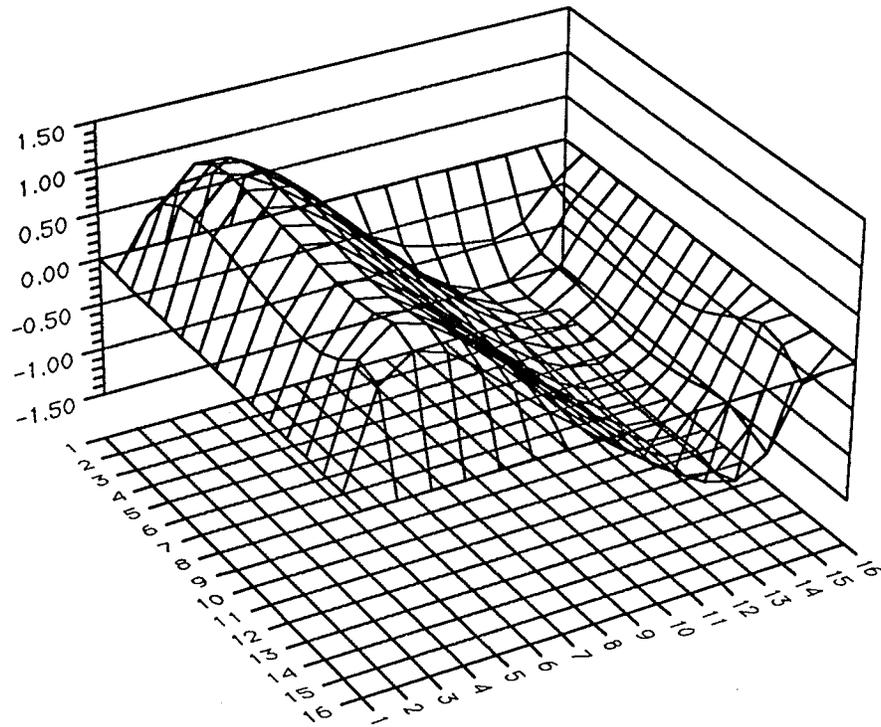


Fig. E.2-4a. The displacement field recovered from Image Correlation Technique in the  $x$  direction.

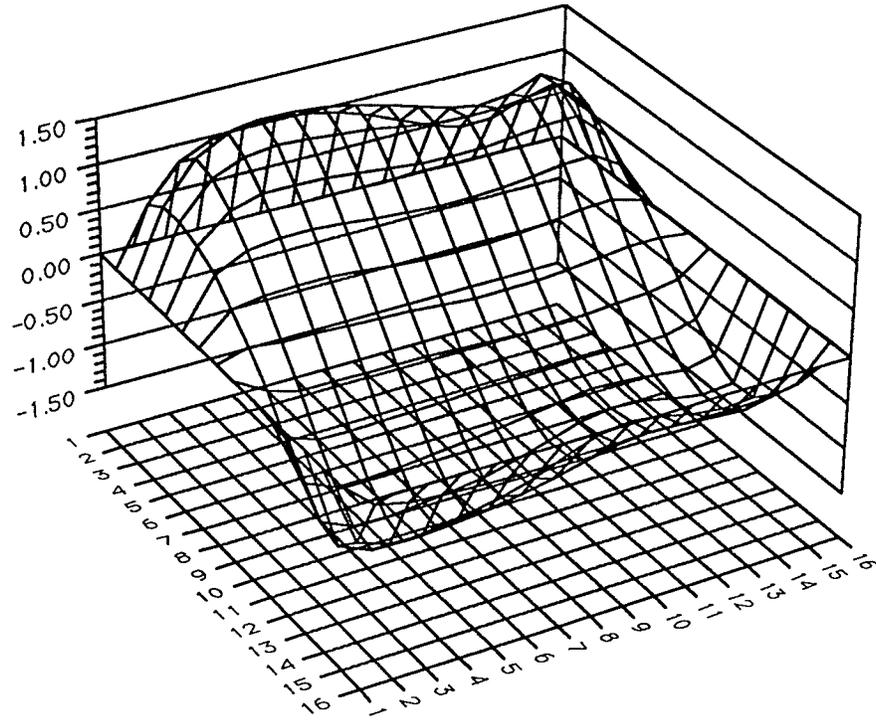


Fig. E.2-4b. The displacement field recovered from Image Correlation Technique in the y direction.

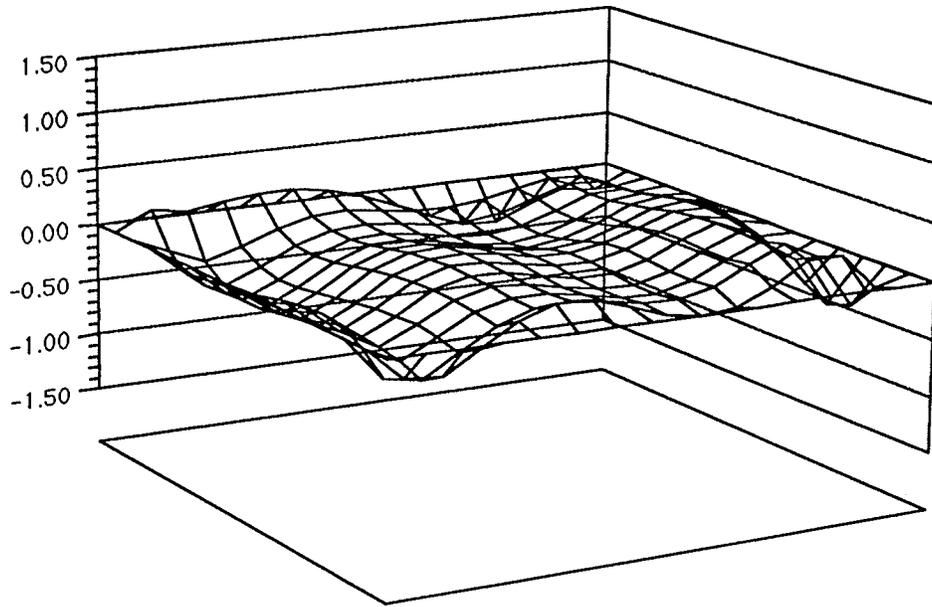


Fig. E.2-5a. The difference between the simulated displacement and that recovered from Image Correlation Technique, in the  $x$  direction.

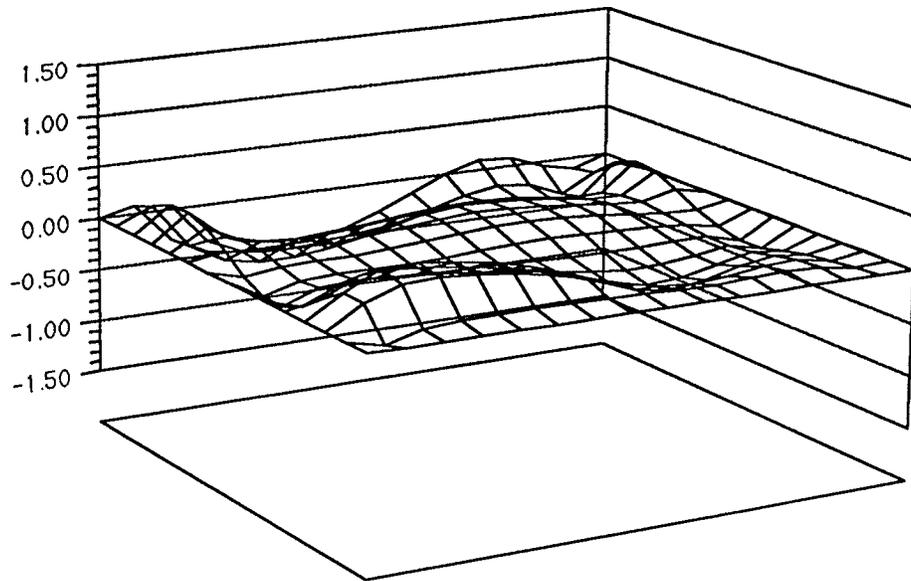


Fig. E.2-5b. The difference between the simulated displacement and that recovered from Image Correlation Technique, in the y direction.

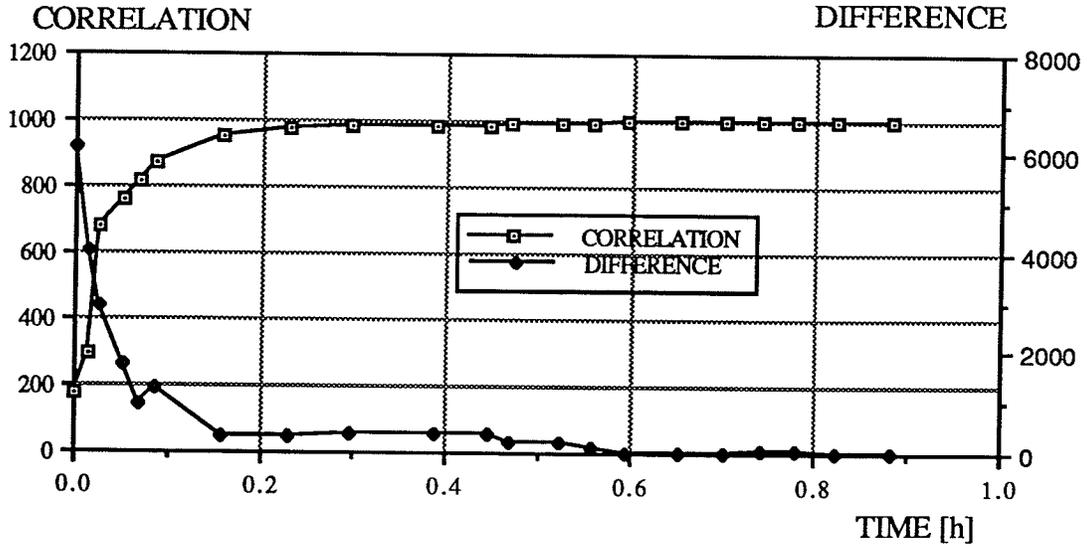


Fig. E.2-6. Image optimization for the *face*. The correlation and the difference versus the computing time.

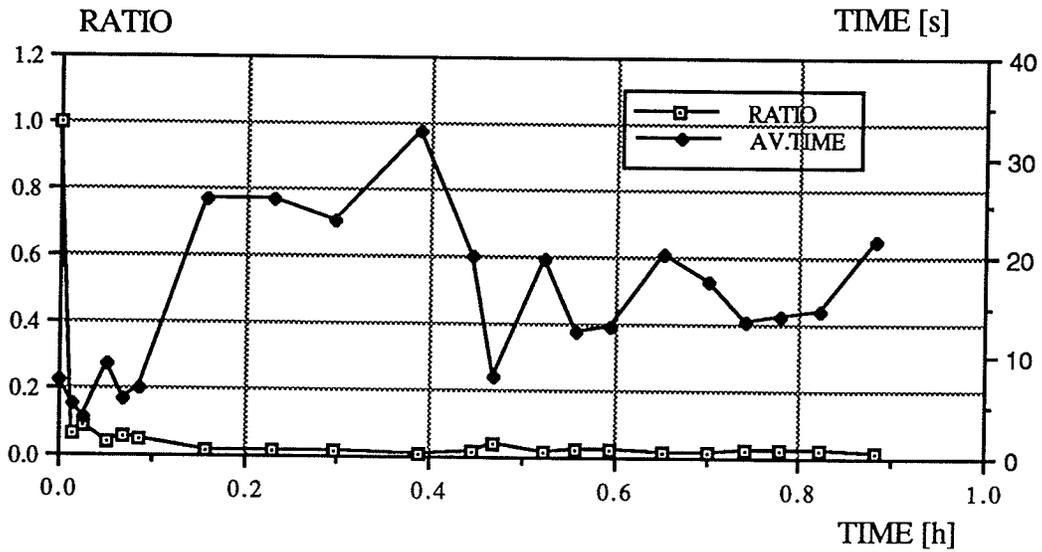


Fig. E.2-7. Image optimization for the *face*. The acceptance ratio and the average computation time (elapsed between two consecutive and accepted trials) versus the computing time.

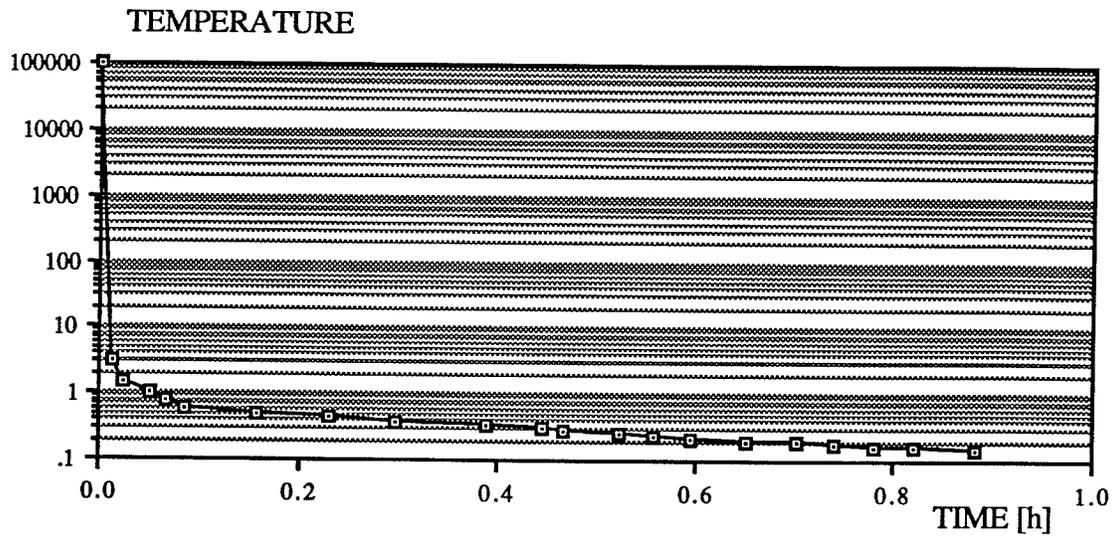


Fig. E.2-8. Image optimization for the *face*. The *temperature-like parameter*  $T$  versus the computing time.

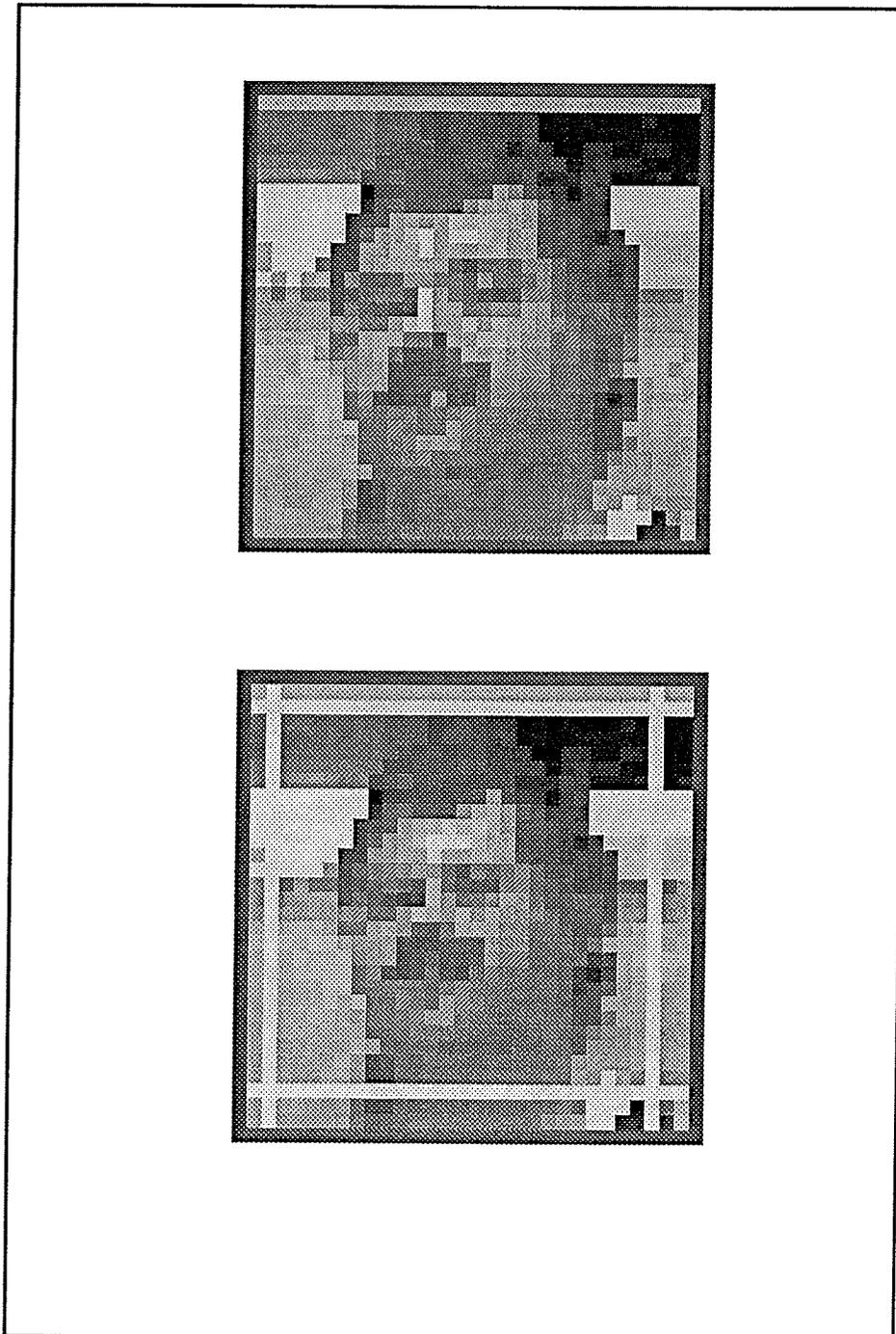


Fig. E.2-9. The *face* on a 32 by 32 pixel matrix. In the upper part is the undistorted image (reference configuration), in the lower part the distorted image using Eq. (5.3-1).

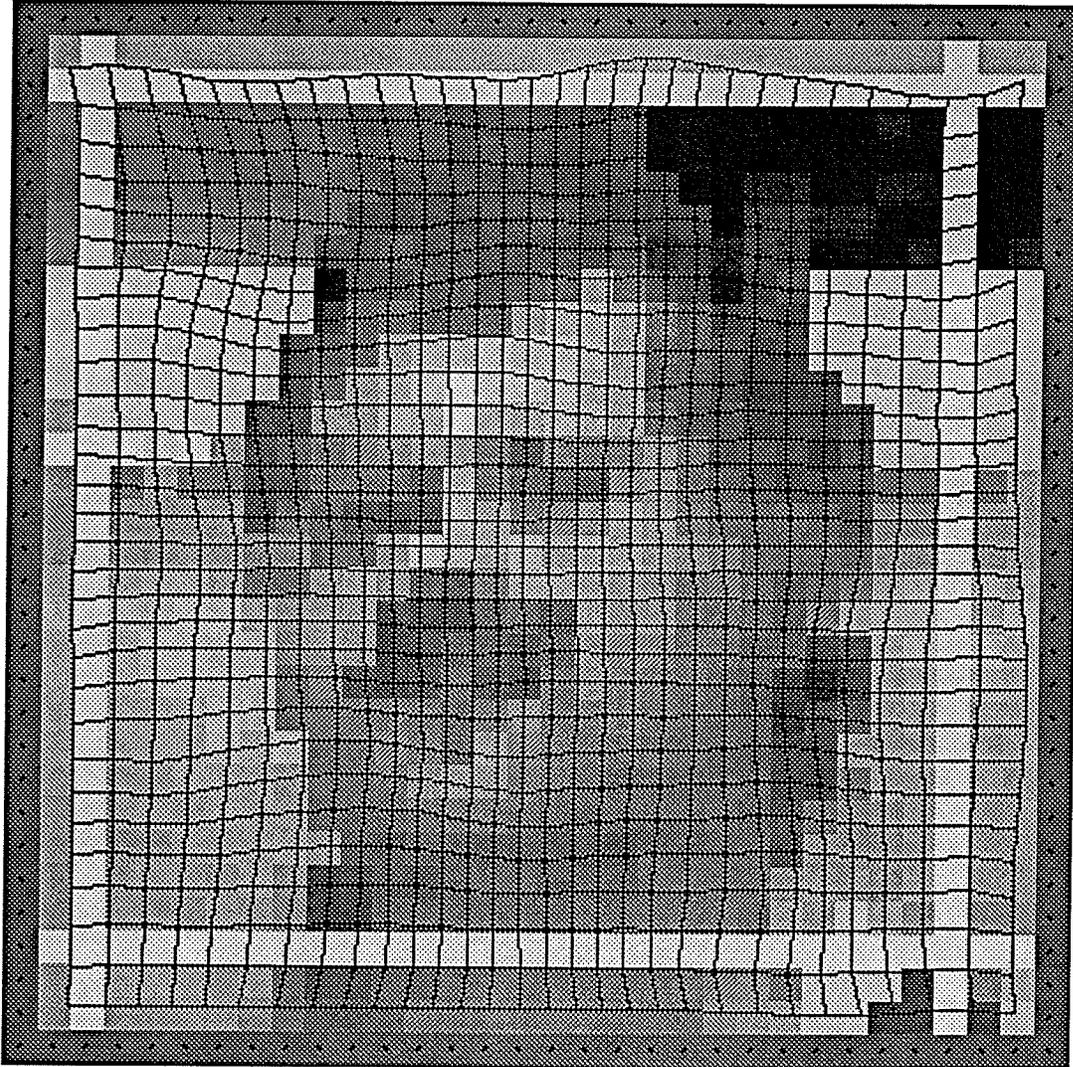


Fig. E.2-10. The *face* under investigation but on a 32 by 32 pixel matrix. The solution image: the reference image grid matched with the deformed picture.

### 5.5. Experiment 3

The same object as in Exp. 2, the human face, was subjected to a different and more violent deformation:

$$\begin{aligned} X^* &= X + \sin\left[\pi \frac{X}{M/2}\right] \sin\left[\pi \frac{Y}{M/2}\right] \\ Y^* &= Y + \sin\left[\pi \frac{Y}{M/2}\right] \end{aligned} \tag{5.5-1}$$

The image E.3-1 shows in the upper part the reference image, in its lower part the deformed configuration is presented. White fields are pixels with zero intensity, the result of using the interpolation function given by Eq. 4.3-4. This is still the deformation with the displacement amplitudes that can be compared to the ones from Exp. 1. However, present strains are more complicated and also shear strains are introduced. In the upper part of the image the horizontal stresses force the image to collapse inside, while in the lower part of the image the same horizontal forces work to stretch the image. The strength of the stresses varies from the top to the bottom of the image (E.3-4a), as the sine function. The vertical stresses are similar to those in the previous experiments, they try to push the image, with varying strength, from the top and the bottom, towards the image center (E.3-4b). The exact result of the deformation combined in both directions, is presented in the image E.3-2. The solution image, E.3-3 shows the superposition of the reference grid and the deformed image at the end of the optimization. The three dimensional charts E.3-5a and E.3-5b present the deformation field recovered from ICT. The shape of the recovered displacement, particularly in the  $x$  direction, is very good. The error charts E.3-6a and

E.3-6b show the error plane variations about the zero value to be within the error ranges: in the  $x$  direction  $(-0.36, 0.21)$ , and in the  $y$  direction,  $(-0.59, 0.32)$ .

Optimization was carried out by applying the following set of controls : $\Delta = 0.01$ ,  $\sigma = 1$ ,  $F = 2$ ,  $C_T = 50$ .

Figure E.3-7 shows the correlation and the difference (averaged over all values on each temperature-like parameter level) versus the computing time. Convergence occurred after more than six hours of the run. The process at the beginning was characterized by the steep climbing of the correlation that latter relaxed, before it reached the convergence plateau, where it stayed.

Figure E.3-8 shows the acceptance ratio and the averaged time required to get a single accepted move. For the most of the computing time, the averaged time oscillated around 30 seconds. The averaged time oscillation coincided with the fluctuation of the acceptance ratio.

In the chart E.3-9 the cooling process is presented.

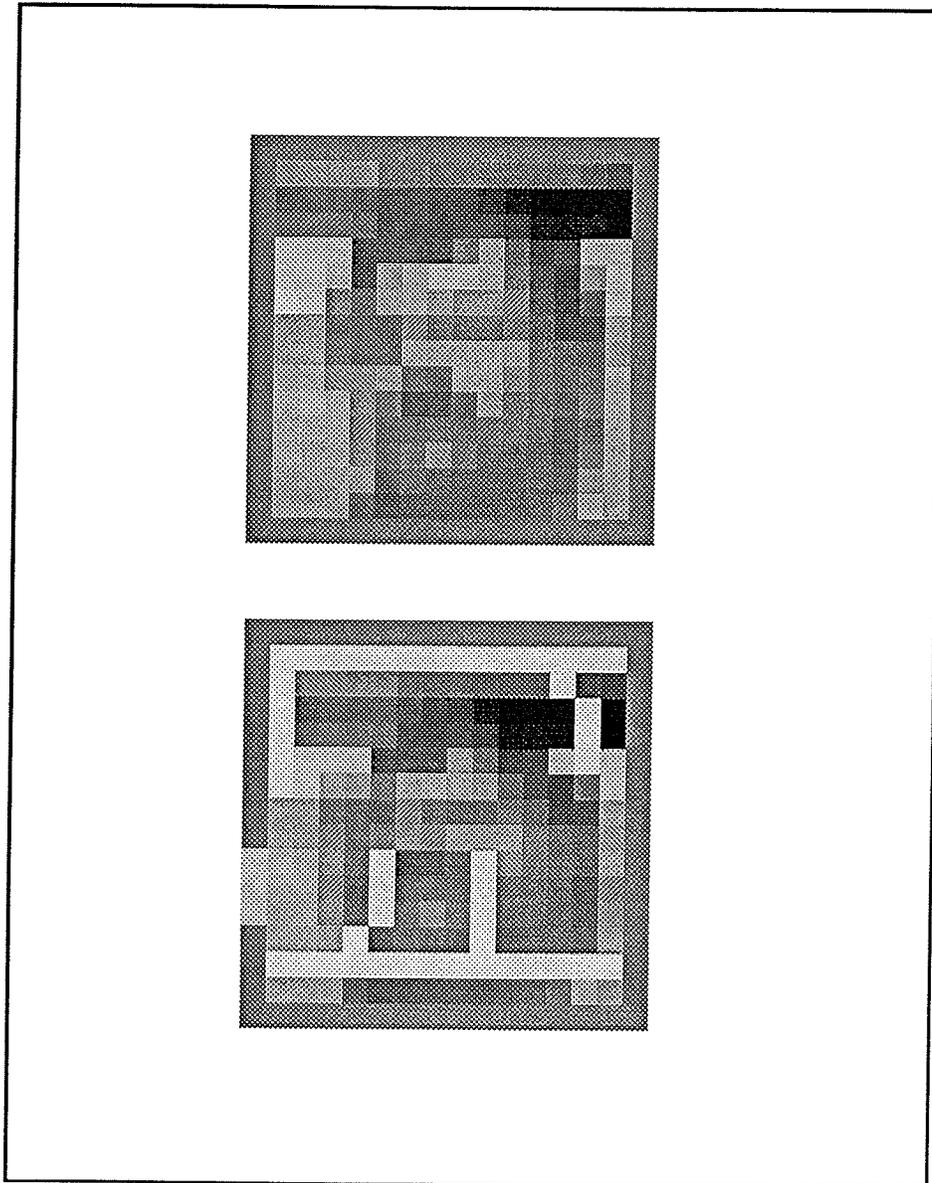


Fig. E.3-1. The *face*. In the upper part is the undeformed image (reference configuration), in the lower part the distorted image using Eq. (5.5-1).

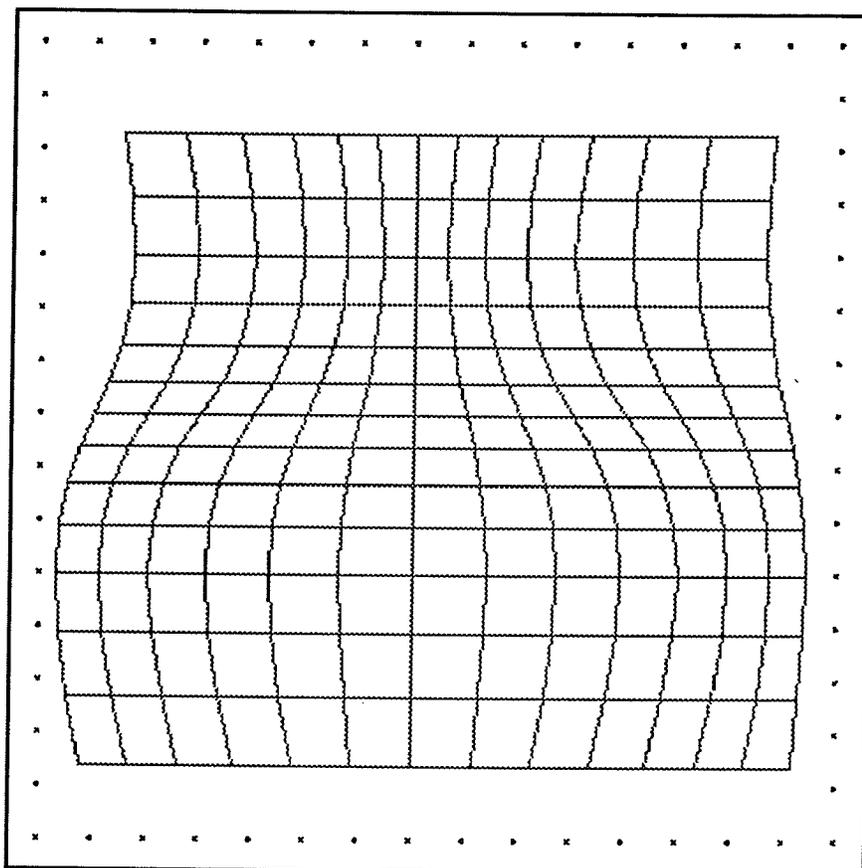


Fig. E.3-2. The reference image grid distorted by (5.5-1).

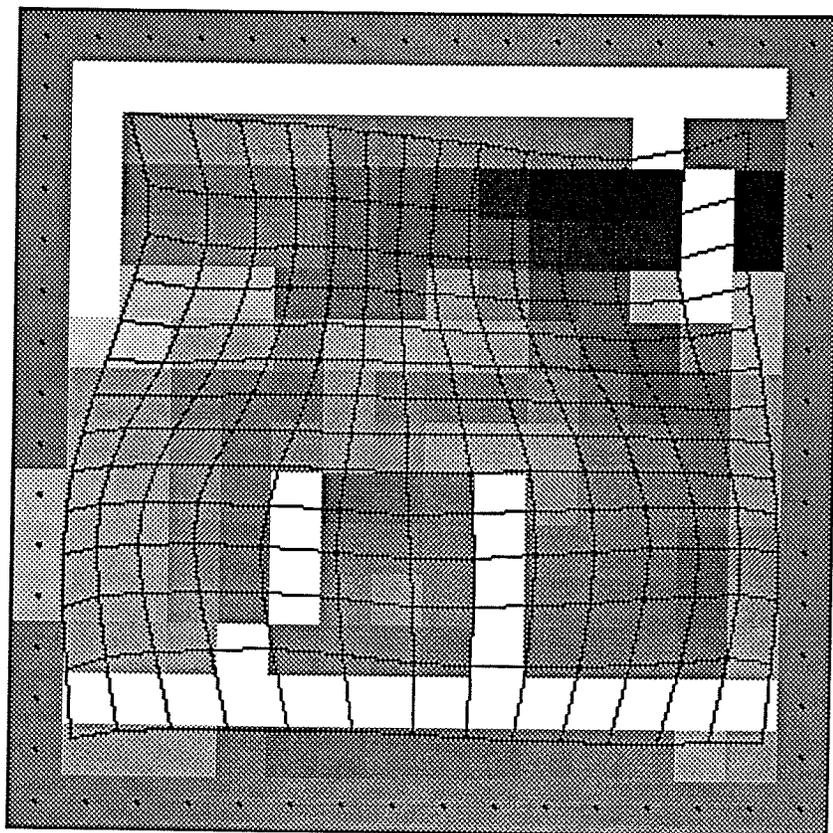


Fig. E.3-3. The *face*. The reference image grid matched with the deformed picture.

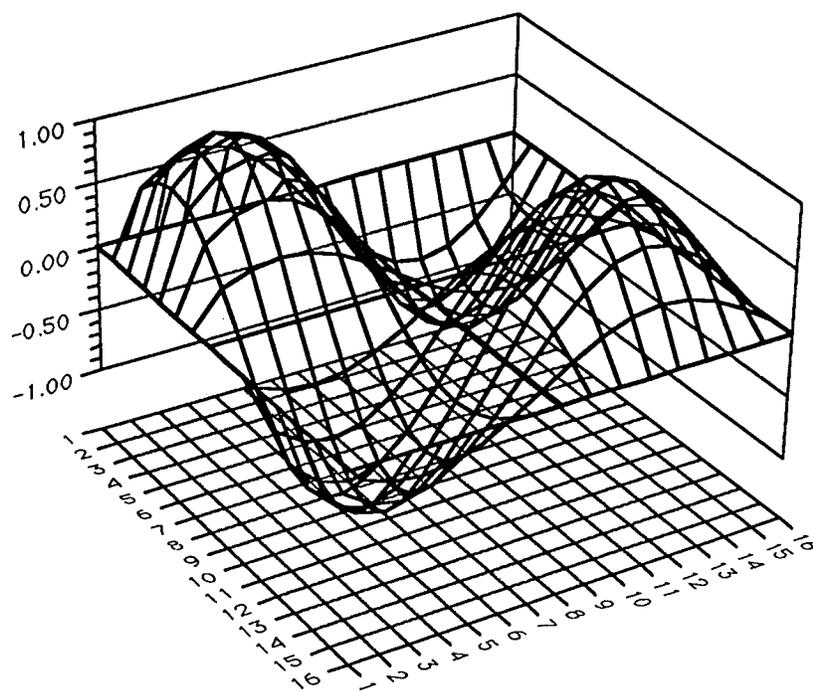


Fig. E.3-4a. The *face* displacement field simulated by Eq. (5.5-1) in the *x* direction.

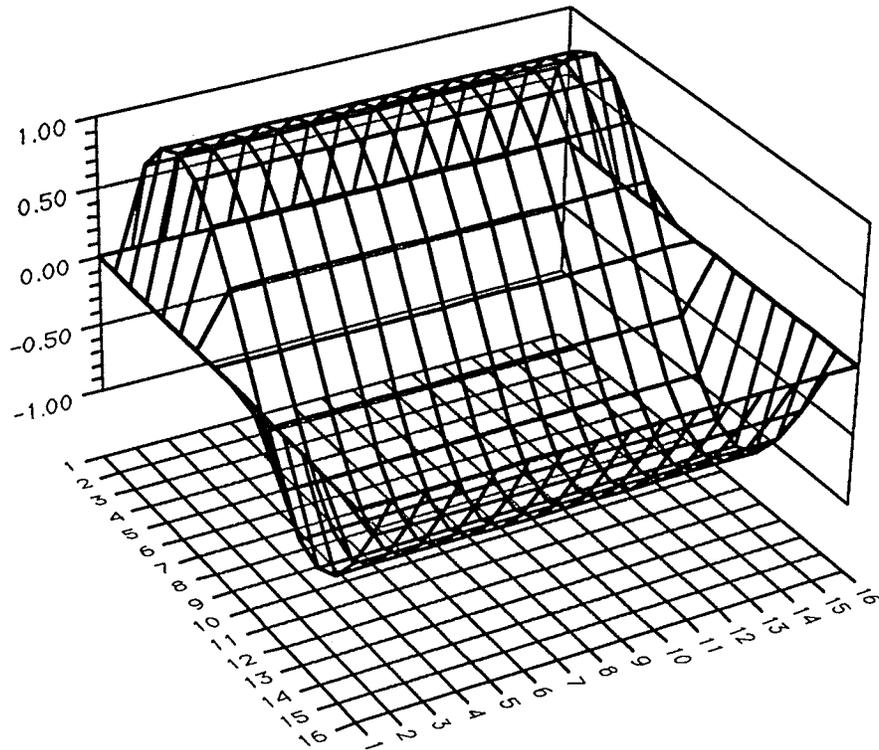


Fig. E.3-4b. The *face* displacement field simulated by Eq. (5.5-1) in the *y* direction.

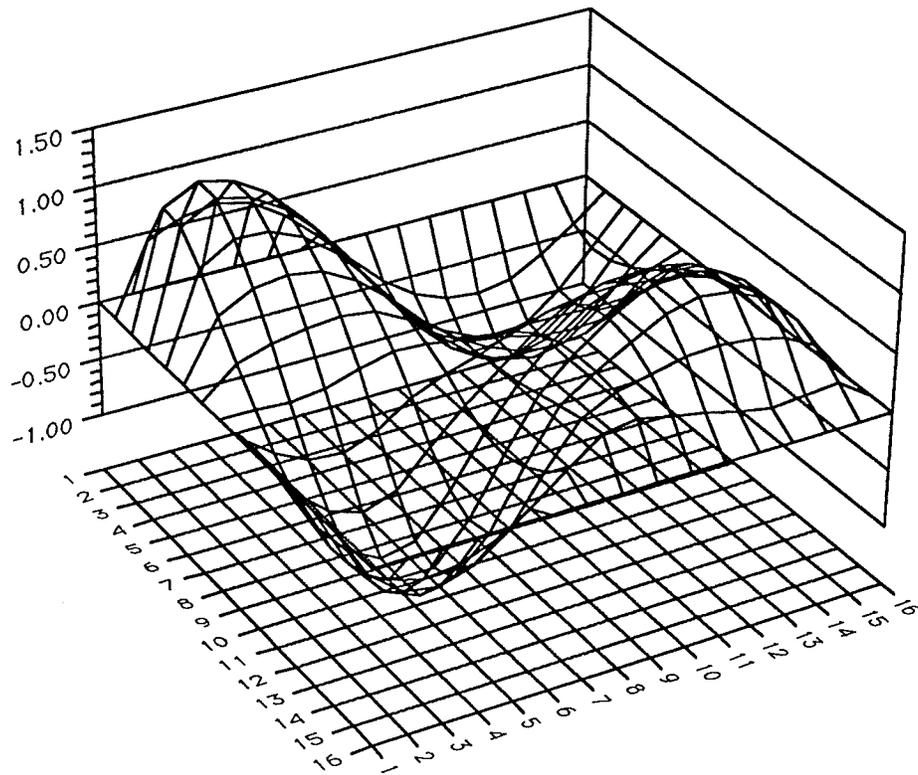


Fig. E.3-5a. The displacement field recovered from Image Correlation Technique in the  $x$  direction.

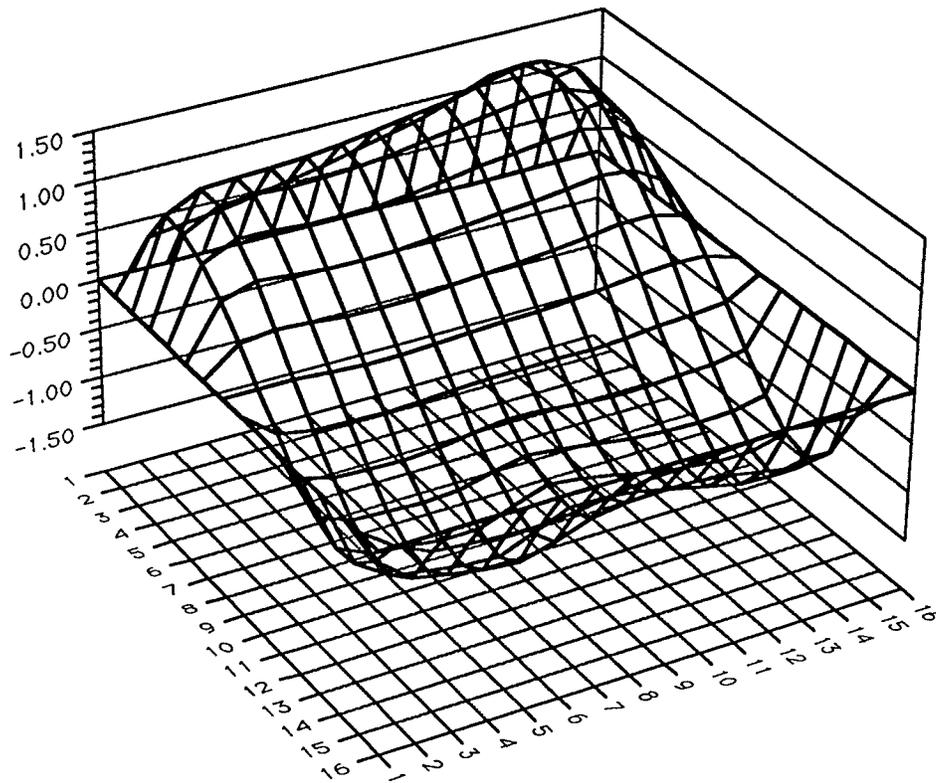


Fig. E.3-5b. The displacement field recovered from Image Correlation Technique in the y direction.

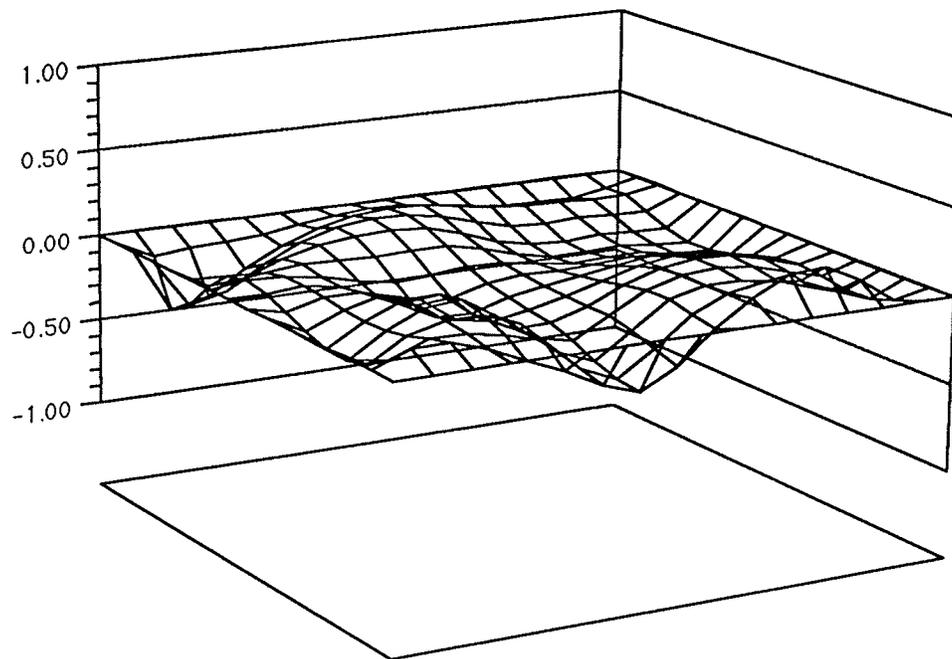


Fig. E.3-6a. The difference between simulated displacement and that recovered from Image Correlation Technique, in the  $x$  direction.

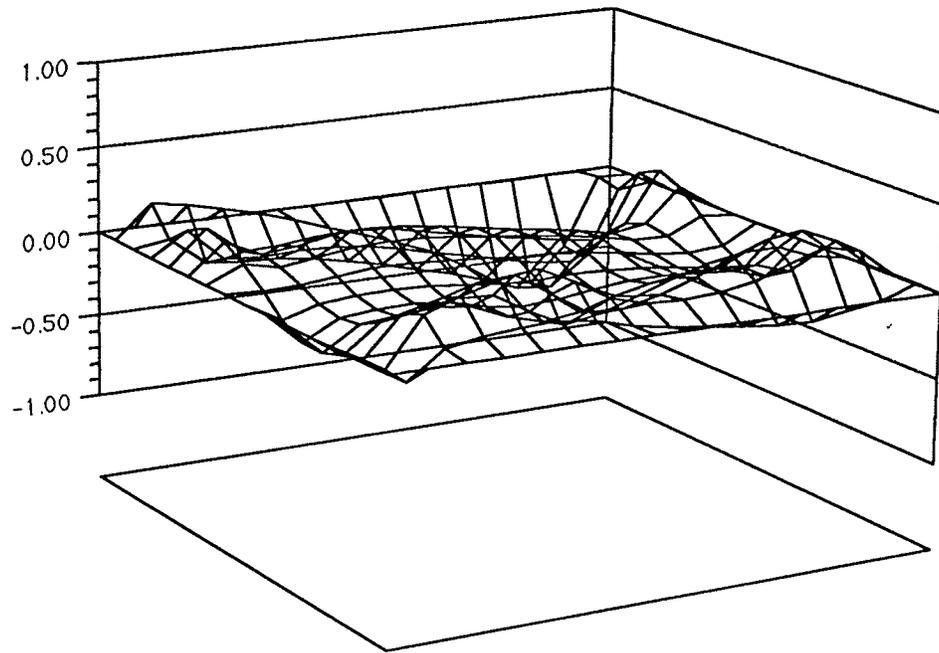


Fig. E.3-6b. The difference between simulated displacement and that recovered from Image Correlation Technique, in the y direction.

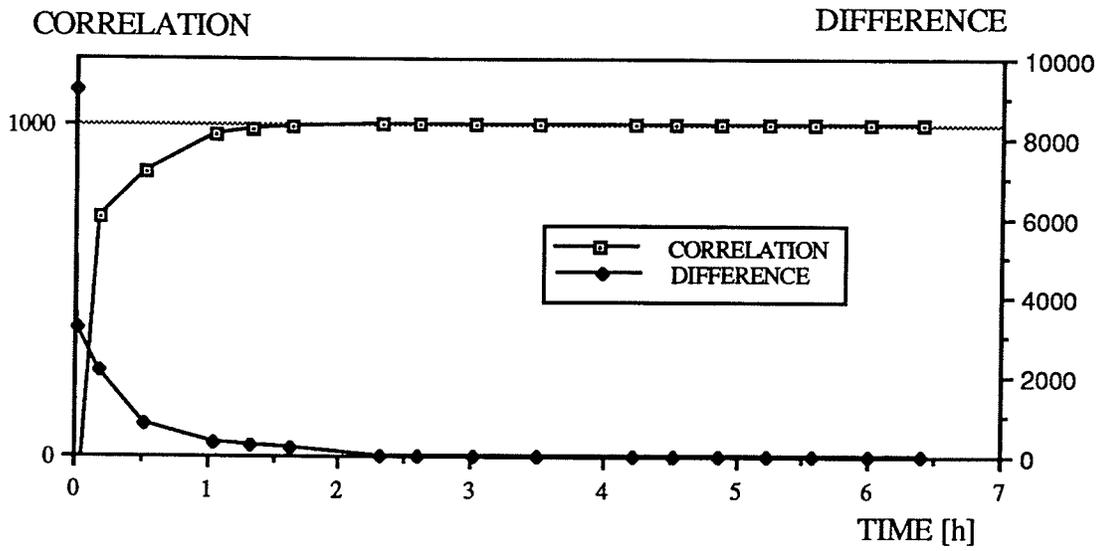


Fig. E.3-7. Image optimization for the *face*. The correlation and the difference versus the computing time.

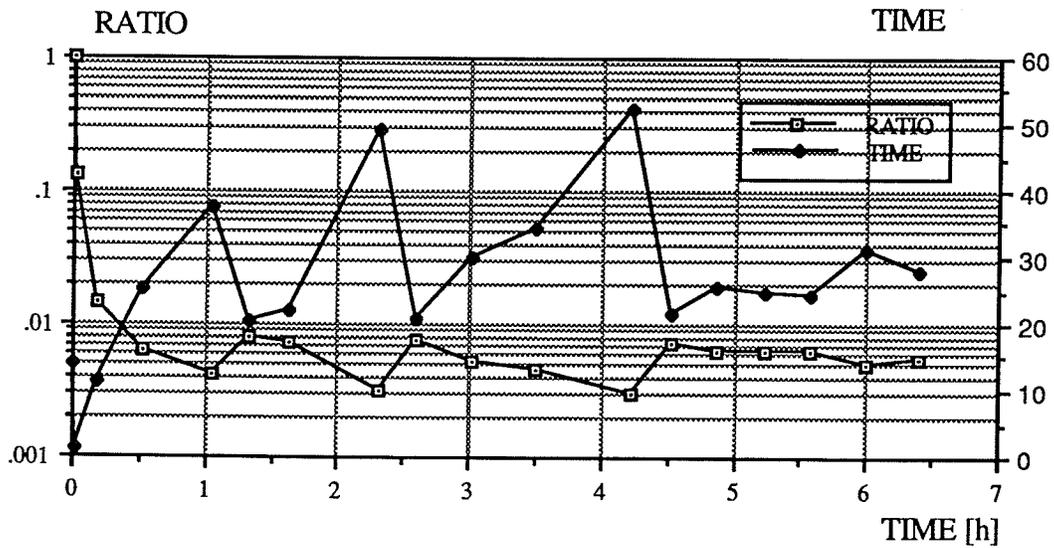


Fig. E.3-8. Image optimization for the *face*. The acceptance ratio and the average computing time (elapsed between two consecutive, accepted trials) versus the computing time.

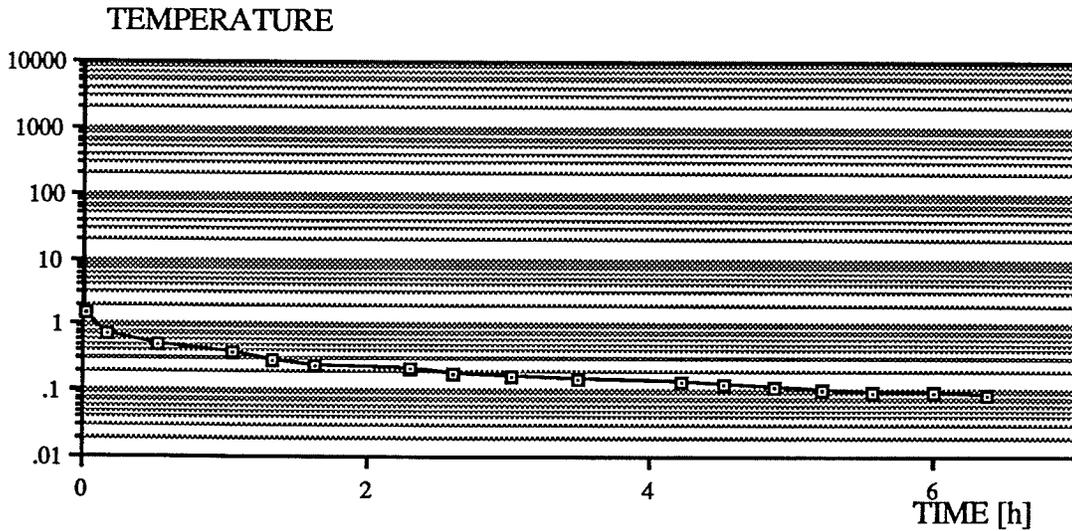


Fig. E.3-9. Image optimization for the *face*. The *temperature-like parameter T* versus the computing time.

## 5.6. Experiment 4

Here a real CT image of a mastectomy sample has been used. The image represents a cross-section obtained using high resolution scanner [Seguin, 1985]. In this case, we actually shrunk the data image by the factor of 10 to produce a small picture (on a 30 x 30 matrix). The compression was done to make the image smaller in size and allow for the method to converge in a reasonable span of the time. A highly nonlinear deformation was applied to the image:

$$X^* = X + 5 \sin \left[ \frac{\pi (M/2 - X)}{M} \right] \left| \sin \left[ \frac{\pi (M/2 - Y)}{M} \right] \right|$$

$$Y^* = Y + 5 \sin \left[ \frac{\pi (M/2 - Y)}{M} \right]$$
(5.6-1)

The image E.4-1 in its upper part displays the reference image and in its lower part, the deformed image. The deformation is a major one. The image E.4-2 shows the combined effect of the deformations in both the  $x$  and  $y$  directions that are applied to the reference image grid. The maximum displacement amplitude, in both directions, can be as great as 5. In the  $x$  direction (E.4-3a), the deformation is thrusting from both left and right sides, towards the center of the image. The strength of the deformation depends on the position in both directions. In the  $y$  direction (E.4-3b), the deformation is simply thrusting with varying strength towards the center of the image.

The deformation fields recovered, using ICT, are presented in Figs. E.4-4a and E.4-4b. The images E.4-5a and E.4-5b show differences between the recovered and exact deformations. The differences, which measure the interpolation error, are less than one pixel. The error range itself, in the  $x$  direction, is  $(-0.66, 0.45)$  and in the  $y$  direction  $(-0.57, 0.47)$ .

The optimization was carried out by applying the following set of controls:  $\Delta = 0.01$ ,  $\sigma = 1$ ,  $F = 1$ ,  $C_T = 25$ .

Figure E.4-6 shows the averaged correlation and the difference functions versus the computing time. Convergence never took place in this experiment. The process was characterized by the steep climbing of the correlation, right from the beginning, that relaxed before it reached a plateau at the 97.5% level. There was probably a chance to get the exact solution, however, there were no indications that this could happen rather sooner than later. Because of the time constraints, the experiment was terminated after fifty hours.

Despite incomplete convergence, the method, in such a case of highly non-linear and large strain deformation , reproduced the initial displacement field with fair accuracy.

Figure E.4-7 shows the acceptance ratio and the average time required to get a single accepted move. For most of the computing time, the average time was increasing its oscillations to around 70 seconds.

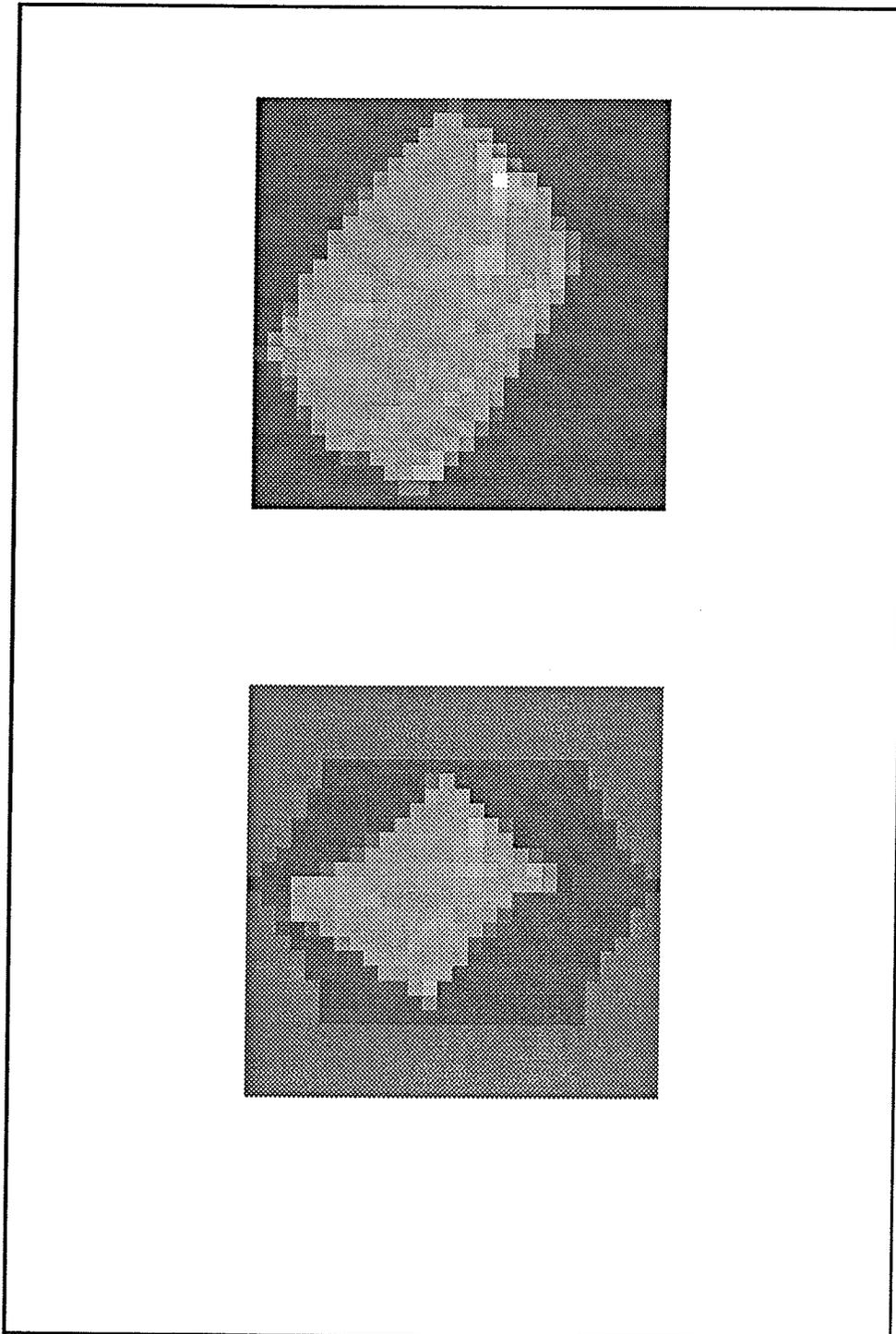


Fig. E.4-1. Breast tissue sample from a CT scanner. In the upper part is the reference configuration, in the lower part the distorted image using Eq. (5.6-1).

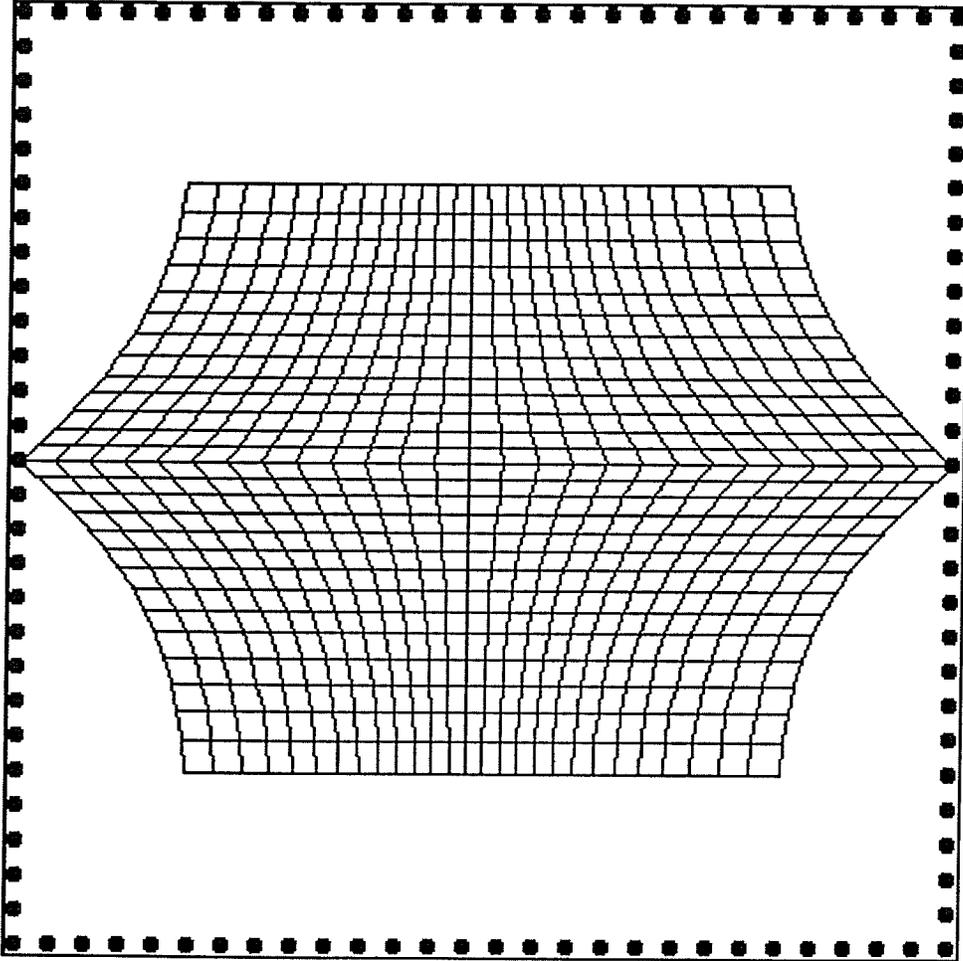


Fig. E.4-2. The reference image grid after distortion. Dots represent border pixels.

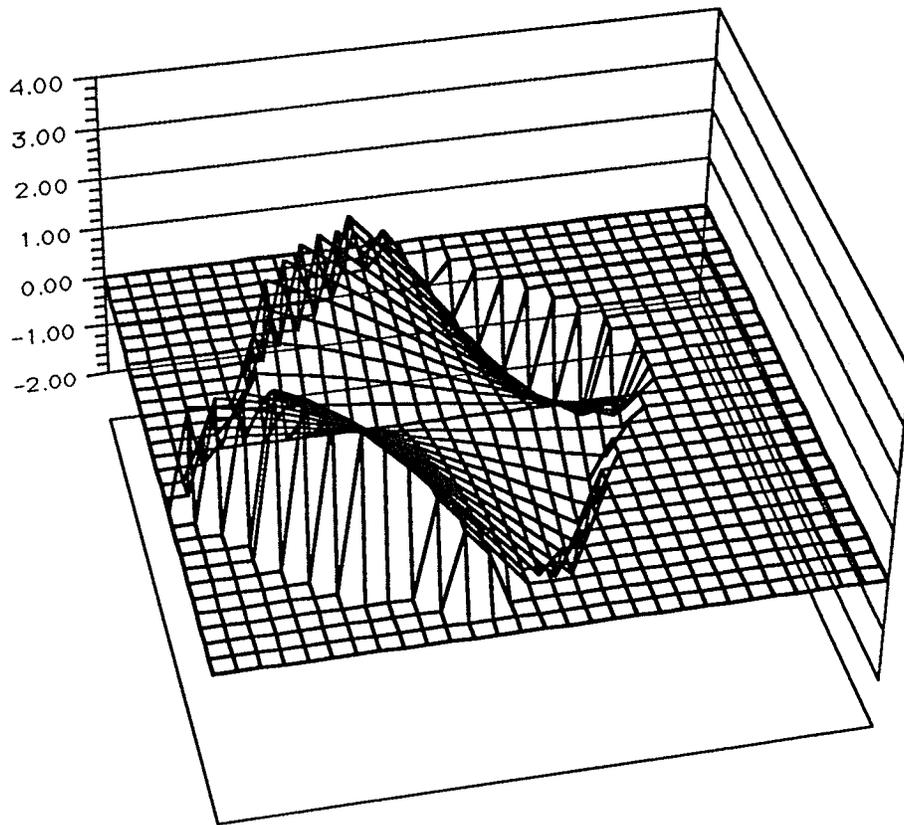


Fig. E.4-3a. The *breast* displacement field simulated by Eq. (5.6-1) in the *x* direction.

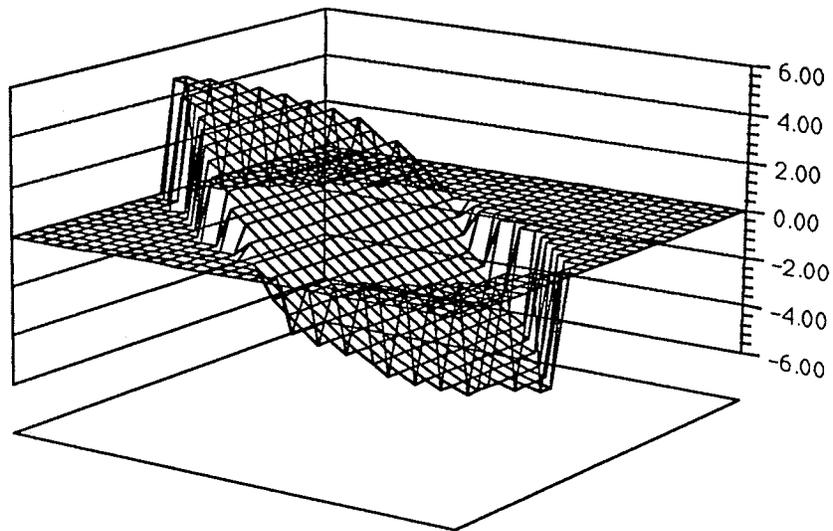


Fig. E.4-3b. The *breast* displacement field simulated by Eq. (5.6-1) in the *y* direction.

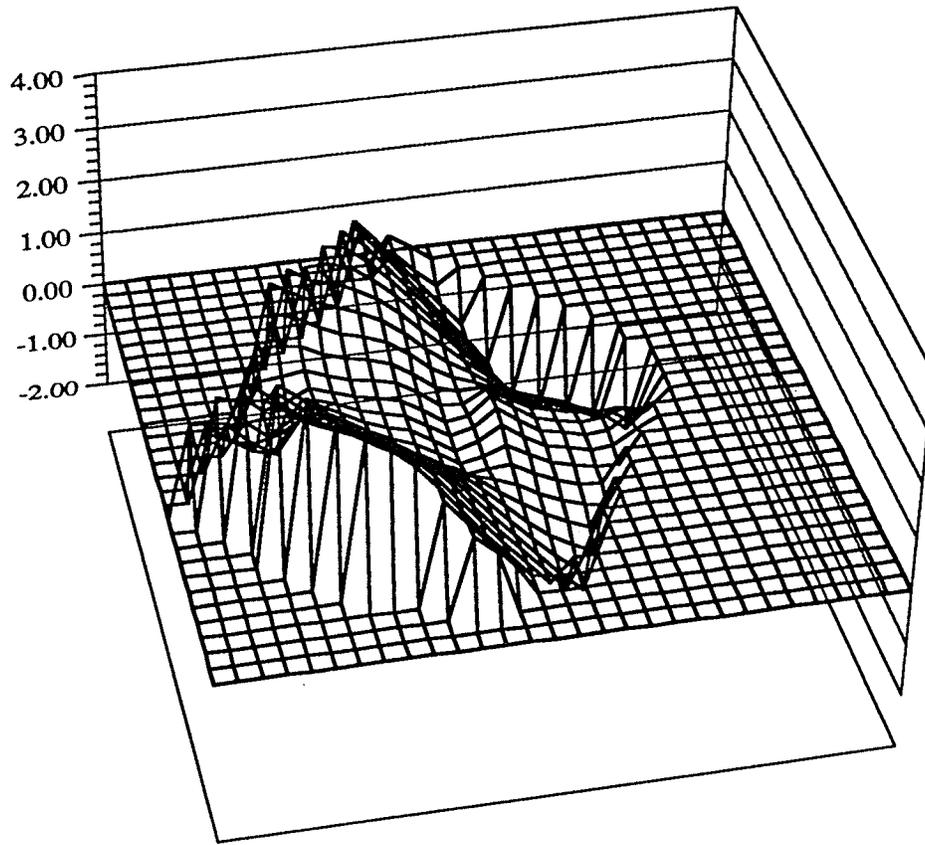


Fig. E.4-4a. The displacement field recovered from Image Correlation Technique in the  $x$  direction.

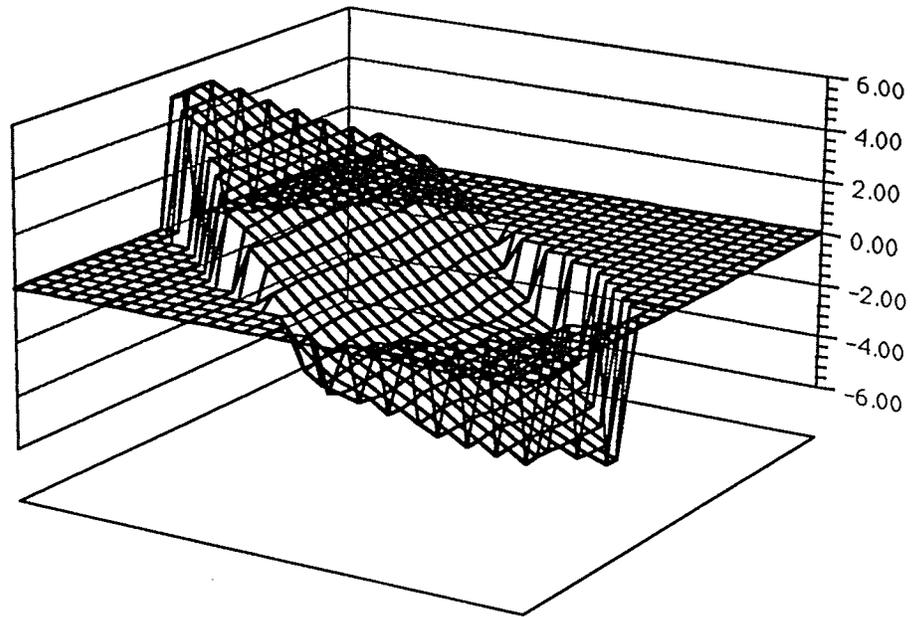


Fig. E.4-4b. The displacement field recovered from Image Correlation Technique in the y direction.

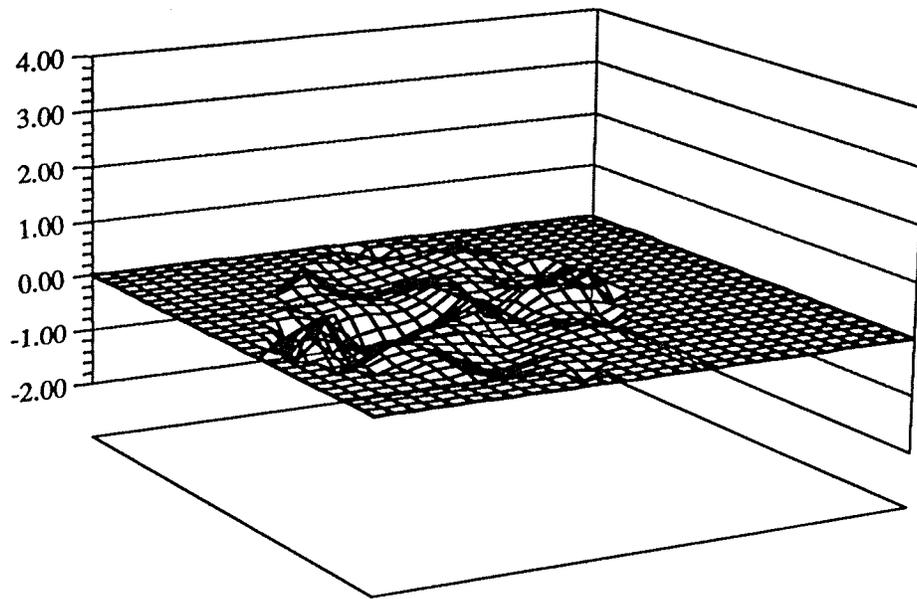


Fig. E.4-5a. The difference between simulated displacement and that recovered from Image Correlation Technique, in the  $x$  direction.

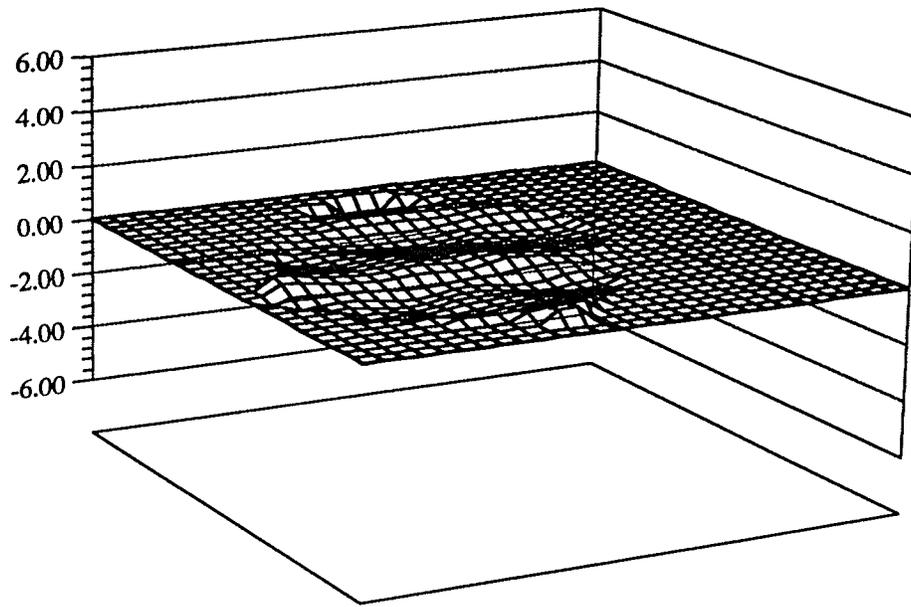


Fig. E.4-5b. The difference between simulated displacement and that recovered from Image Correlation Technique, in the y direction.

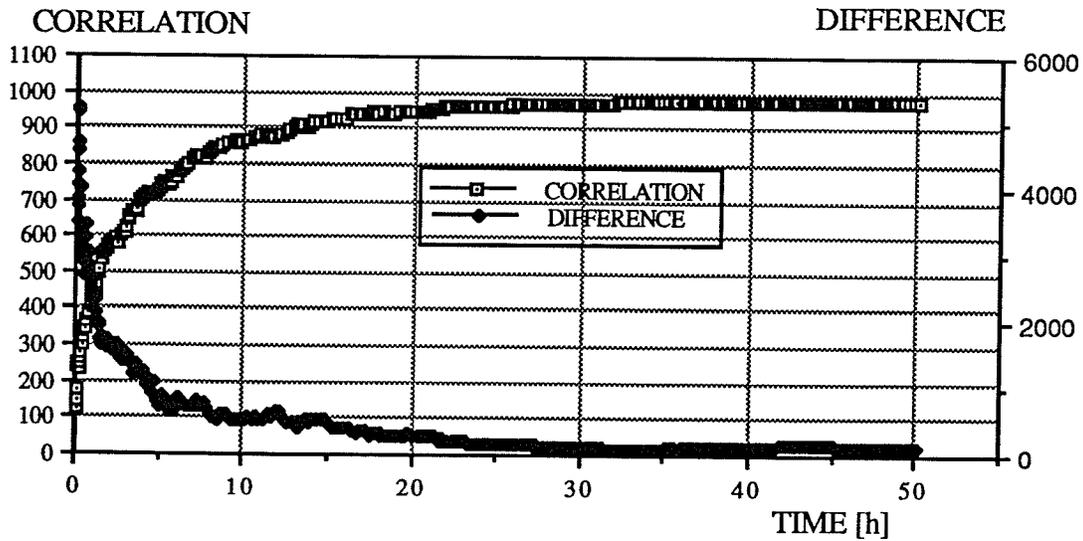


Fig. E.4-6. Image optimization for the *breast*. The correlation and the difference versus the computing time.

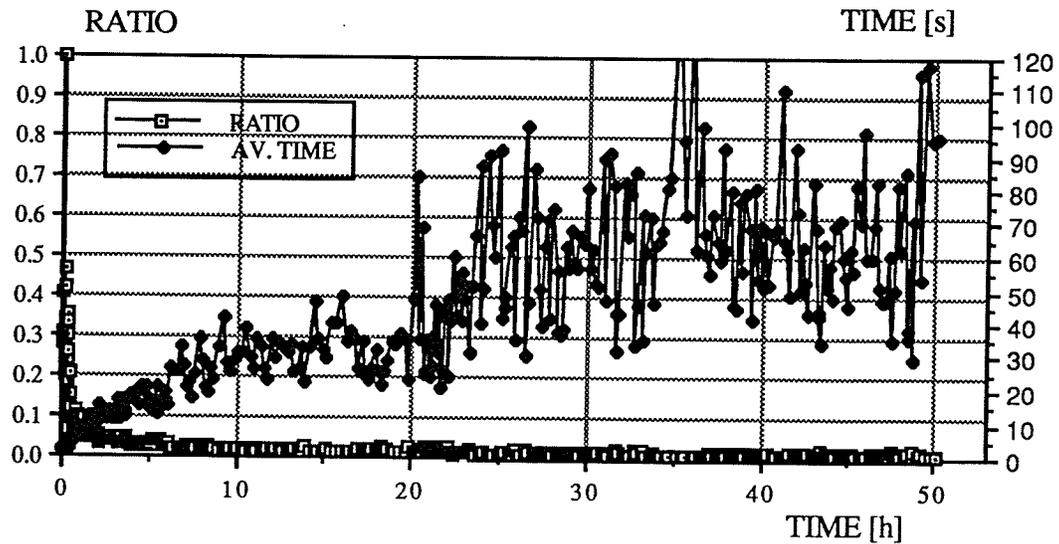


Fig. E.4-7. Image optimization for the *breast*. The acceptance ratio and the average computation time (elapsed between two consecutive and accepted trials) versus the computing time.

## 5.7. Discussion

All experiments show that the ICT method recovered the character of the deformation with surprising fidelity. This was true even for non-linear deformations and with an amplitude comparable to the sampling distance (pixel size).

From the results obtained in Experiments 1 and 2, it is clear that an increase in the spatial sampling rate leads to an increase in the accuracy of the final result.

In all experiments, the range of the error was always within the sampling distance (pixel width). There was also observed a tendency to underestimate the displacement field. The fluctuations of the error surface, on the three-dimensional charts, are rather small and the surface, with a small number of exceptions, is rather smooth. This explains why the shape recovery is so good, even in the experiments where the deformation amplitude was comparable with the sampling distance.

The results of the Experiment 4 confirmed that ICT is capable of reproducing the deformation field for strains that are complicated and non-linear.

The correlation function (4.2-1) that drove the optimization, in all the examples presented, has proven to be an effective measure of image similarity. Also the difference function played a useful role as a convergence indicator.

In every experiment there was always a certain number of accepted moves, even after the convergence occurred. The explanation lies in the image sampling rate, that left certain room for the movement algorithm to move the reference grid without change in the correlation. This type of move does not change the state of optimization, but does help lead to convergence. In this sense, this background movement noise is desirable.

The algorithm has a broad range of the iteration step parameter  $C_T$  that allows for convergence. If the iteration constant is small the convergence occurs later in the process and is accompanied by a steep increase in the correlation. If the iteration constant is sufficiently large and quasi-equilibrium of the system is achieved at each temperature like parameter level, the convergence is easy and the correlation changes are less dramatic. However, large values of the iteration constant usually mean longer runs. The appropriate values for the correlation constant are in general object dependent.

Experiment 4 revealed a very interesting feature of the ICT. The optimization had not to be carried out to the point where the correlation reached its highest possible value. As a matter of fact when the correlation reached the 95% limit value, the matched values were (on average) only approximately 3% apart. An additional confirmation of this fact is the detailed correlation function chart produced in Chapter 7 (Fig. 7.1-2).

The experiments presented showed that the movement algorithm given in Chapter 4 is capable of reproducing translation, nonlinear and shear strains. Because of the attachment of some pixels to the image border, the algorithm is probably incapable of handling a rotation, at least on a macro image scale. For this reason, it can be referred to as the translation algorithm.

As far as the intensity interpolation is concerned, the ICT uses a simple way of constructing of the optimized image (temporary image) that is created at each iteration step. Cells of the optimized image grid that do not coincide with centers of the deformed pixel grid are left with the zero intensity (white cells). Others, that intersect one or more pixel centers, acquire the average intensity of all interacting pixels (gray void cells). The optimized image is then matched with the deformed one (correlation is calculated).

If only the intensity differentiation across the object is unique, the interpolation should not yield a significant error that could jeopardize the optimization result. Possible empty spaces (zero intensity pixels) throughout the temporary image are not so important. Although, they may insult one's esthetics, they actually reproduce the deformation. One

has to remember that the outcome is the displacement field estimated for all pixels from the reference configuration. The possible loss of accuracy that may occur is compensated by the gain in the computation speed. There is no need to bother with filling in an empty space at each iteration step. Actually it is possible to treat these spaces as noisy pixels with specific value - zero. Moreover, Chapter 6 presents two experiments where that empty space, resulting from deformation, was replaced with non-zero pixels.

The Image Correlation Technique brings out the required displacement field. The method takes two images and by correlating them, gets the displacement field

These experiments show how the Image Correlation Technique works on test images under simulated deformations. Strains are large and deformations are non-linear.

At this stage, we do not address the question of noise. It is expected that noise will introduce spatial errors in the image matching.

One of the most inconvenient features of the method is the high requirement for computer time and speed. That is the main reason why the number of experiments is limited.

However, the goal was to demonstrate that the method works. The work was done using a relatively low power computer, the Macintosh II. But its excellent graphical interface gave an opportunity to interact at the each stage of development of the method. Having these results, the transfer to high computation power and fast super-computers will be straightforward and all outstanding questions can be addressed and solved in a reasonable time frame.

The ICT method that is introduced successfully reconstructs large strains and nonlinear deformations. There is no need for control points. It is the first method that attempts, with success, to process and actually to connect pictures as a whole. The method also opens new possibilities in the image recognition field, one of the most difficult tasks in image processing and in remote sensing.

On the other hand, it has to be mentioned that this procedure requires calculation power. It is also at its early stage of development and was only tested on simulated deformations.

## CHAPTER 6

# DIGITAL DIFFERENTIAL RADIOGRAPHY

### 6.1. Introduction

It has been proposed that three-dimensional mammograms taken some time apart could be subtracted to bring out changes which are due to growing breast tumors [Gordon, 1985], [Zhou, 1992], [Zhou, 1990]. Two different sources of misregistration can be encountered: 1), differences in patient positioning, breast compression, dose parameter settings, or change of equipment; 2), a growing mass, may change the second image itself. These two situations should be addressed separately and accordingly geometric transformation between these two mammograms should be performed before the digital mammogram subtraction takes place. In other words, in the first step, it is necessary to find the geometric so called *warping* transformation [Bookstein, 1990] that relates an earlier "reference" image to a current warped one. Digital Subtraction Mammography relies on a selection of features in both images that can be related to one another as corresponding control points. The selection of corresponding control points can be done by using template matching and texture classification techniques. Recent work emphasizes the usefulness of Zernike moments [Zhou, 1991]. But the most common method for the control point selection is manual selection of corresponding features in both images. Existing interpolation techniques help to extend the coordinate transformation from the control

points to all points that entirely cover both pictures. In the last step, the geometric unwarping is performed, the reverse of the transformation between an image and its warped version. Any differences between the reference and the unwarped image, that might occur, undergo careful inspection for growing abnormalities.

As it is, DSM is still at the development stage and the accuracy of its results very much depends on the number of control points and the judgment of a person who makes the initial correlation.

In the following sections a new approach has been proposed. Using ICT, a deformation field, for each pixel, from the reference image, is determined. By applying the differential operator, the deformation field is transformed into a strain field, which may be used as a diagnostic image.

## **6.2. Method Formulation**

The immediate application of ICT is in the field of radiology. It requires two radiological, CT or MRI images that are taken some time apart. Applying the divergence calculus to the displacement field (vector field) computed from ICT, produces spots that are regarded as sources of positive or negative displacements, sources of a growing mass or movements. Any sinking may, for instance, indicate drainage of body fluids or tissue necrosis. And by analogy, any source of displacement (movement) may be attributed to the gathering/deposition of physiological substances or growing tissues. Especially fast growing and concentrated abnormalities may be easy to detect. Another requirement is that both images must be radiological compatible, i.e., that there be a one to one correspondence between all tissues represented in both images.

This thesis postulates, that calculating the divergence field of the displacement field, yields an image that can be regarded as ready to use diagnosis.

The divergence of the vectorial displacement field  $\mathbf{D}$  can be expressed as

$$\text{div } \mathbf{D} = \frac{\partial u}{\partial x} + \frac{\partial v}{\partial y} \quad (6.2-1)$$

or its digital counterpart

$$\text{div } \mathbf{D} [i j] = \frac{u_I [i j + 1] - u_I [i j - 1]}{2} + \frac{v_I [i + 1 j] - v_I [i - 1 j]}{2} \quad (6.2-2)$$

Fig. 6.2-1 shows a flow chart of the method.

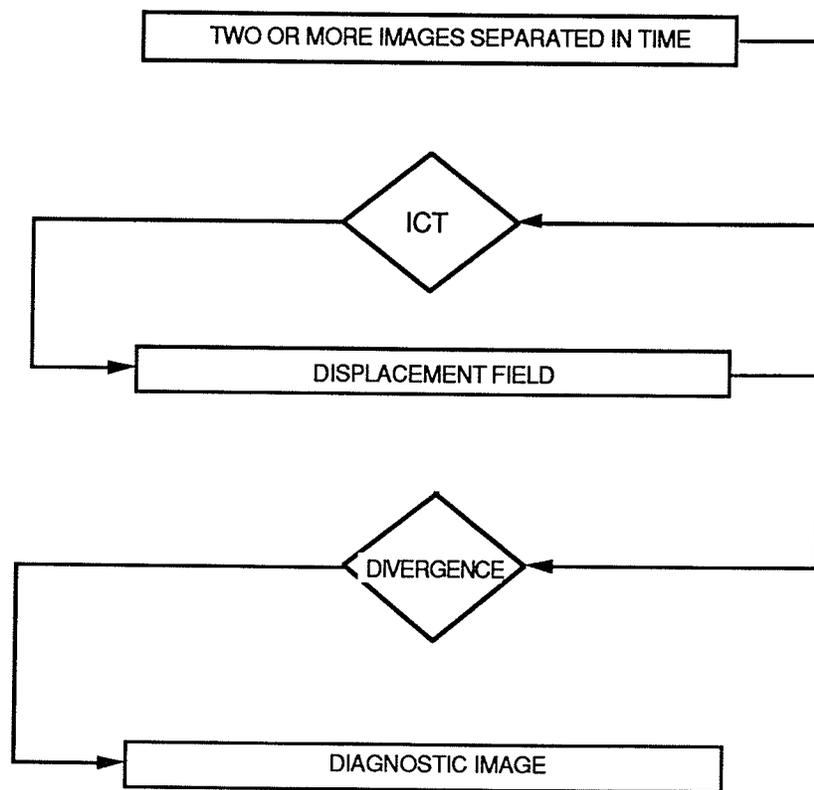


Fig. 6.2-1. Digital Differential Radiography flow chart.

### 6.3. Model of Growing Mass

In the experiments presented, a computed tomography (CT) image has been used of a mastectomy sample. Fig. 6.3-1 shows a cross-section that has been obtained using a high resolution scanner [Seguin, 1985]. The image was shrunk by averaging 10 x 10 blocks of pixels to produce a small reference picture on a 30 x 30 array. This compression was done to reduce computing time. The compressed version is shown in Fig. 6.3-2, and it is the reference image. Fig. 6.3-3 shows the distorted image of the mastectomy sample. The latter is the result of the application to the reference image of a simulated deformation that preserved the volume. From the center of the image, each pixel originally placed at the distance  $R$  from the position  $(i, j) = (15, 15)$  was moved aside, along a radial line, to a new position at the distance  $R^*$

$$R^* = \sqrt{R^2 + r_0^2} \quad (6.3-1)$$

where  $r_0 = 3$ .

The empty space left around the central pixel ( $i = 15, j = 15$ ) was filled with the unique density value to represent a growing mass. Fig. 6.3-4 shows how the reference image grid (based on the centers of image pixels) has changed due to the simulated distortion (6.3-1).

Fig. 6.3-5a and Fig. 6.3-5b show the displacement field resulting from the deformation (6.3-1), in  $x$  and  $y$  directions, respectively.

It is evident, that from the mathematical point of view, (6.3-1) has a discontinuity at  $R = 0$  and both displacements change sign when crossing it.

The displacement field images (in both directions, together) provide ready to use diagnostic documentation. The idea of presenting useful information using only one image

is much more attractive and convenient for medical staff. This can be done by applying Eq. 6.2-2 to the displacement fields in both directions. Fig. 6.3-6a presents the divergence field calculated from displacement fields, in the wire frame format. The same data are presented in Fig. 6.3-6b in the format of a two-dimensional contour chart. The shallow grow, that coincides with the position of the sample outline, represents the region with the negative displacement (contraction). This was an unexpected result of the decision that was made before the divergence calculation. Namely, the displacements of all pixels, that were not object pixels, were set to zero. This created the simulation of the tissue collapsing at its rim. The center of the simulated deformation (6.3-1) is reproduced in the form of a tall and sharp peak that is exactly positioned at the center of the deformation. In the contour representation this peak is equivalent to the number of concentric gray regions, exactly positioned at the center of the image.

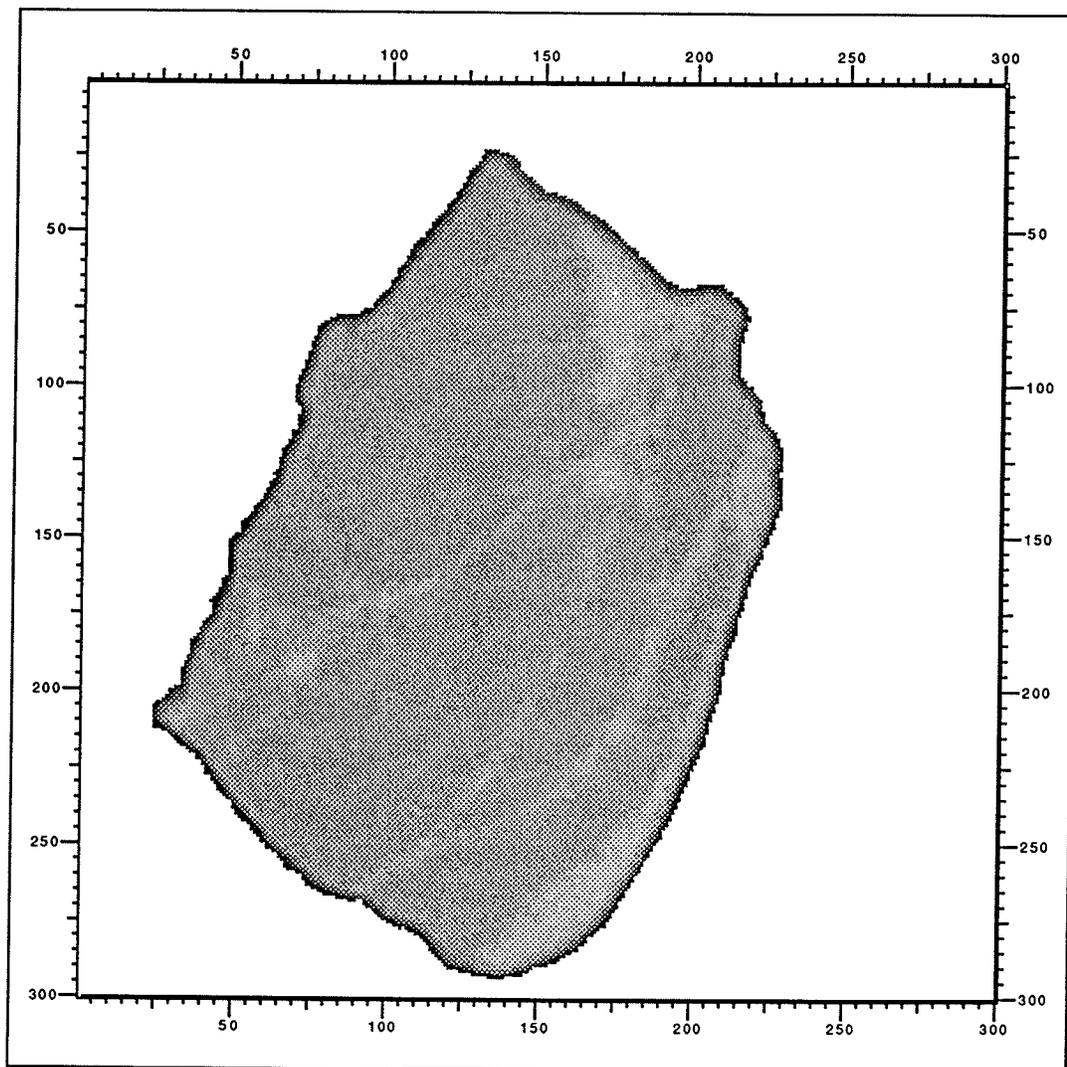


Fig. 6.3-1. A high resolution CT image of a mastectomy sample.

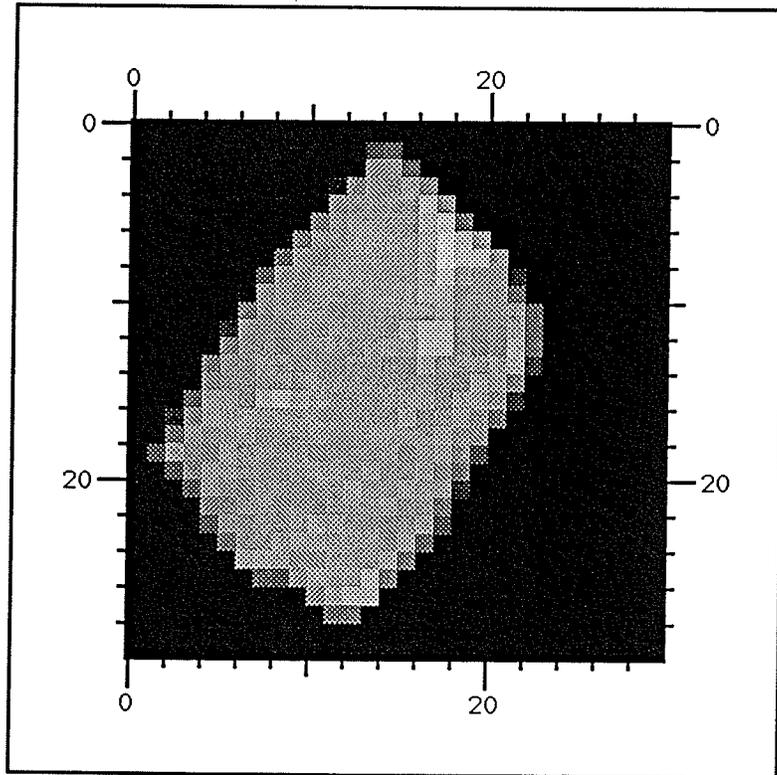


Fig. 6.3-2. A CT image of a mastectomy sample obtained by shrinking the image from Fig. 6.3-1, by a factor of 10 to produce a reference picture of the size 30 x 30 pixels.

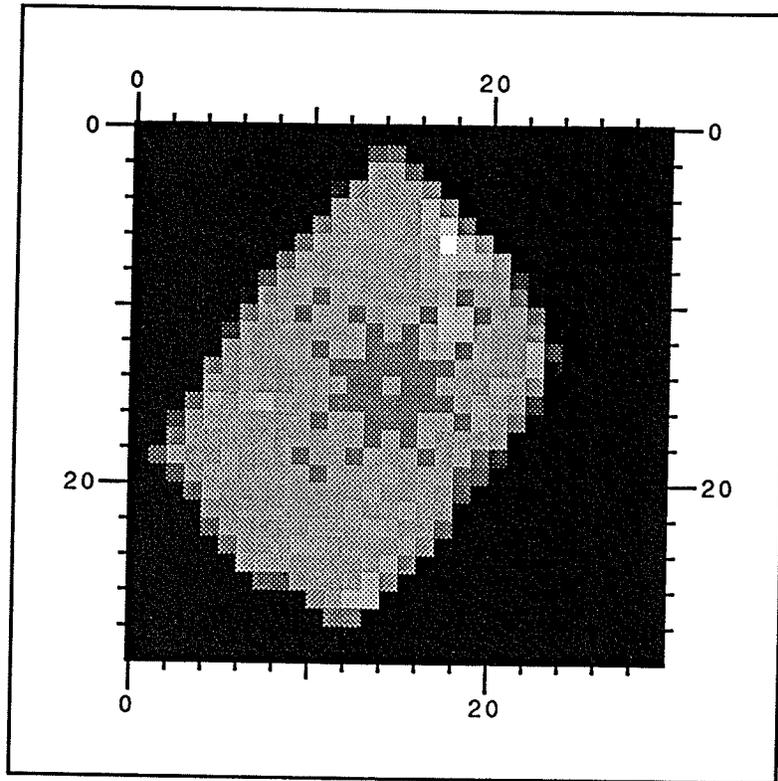


Fig. 6.3-3. The reference image after the simulated distortion using Eq. (6.3-1). The pixels opened up in the middle due to the distortion have been assigned a unique gray level to represent a tumor.

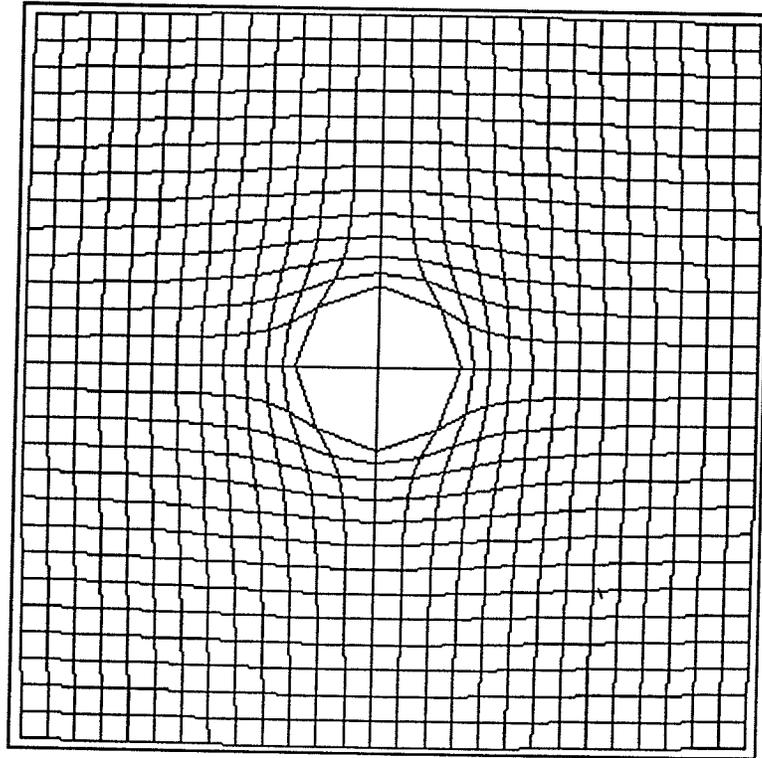


Fig. 6.3-4. The appearance of the reference image pixel grid after the distortion (6.3-1).

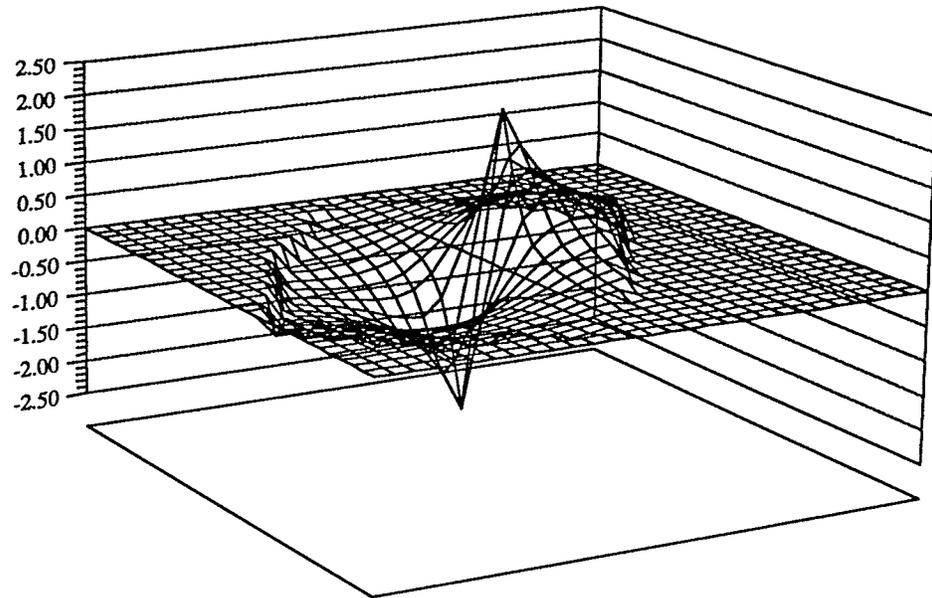


Fig. 6.3-5a. The displacement field simulated by Eq. (6.3-1) in the  $x$  direction.

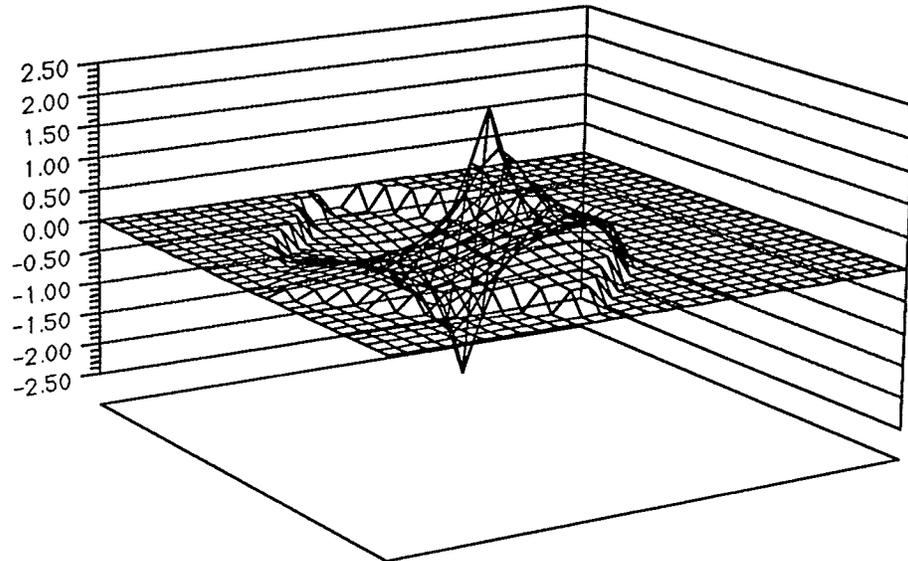


Fig. 6.3-5b. The displacement field simulated by Eq. (6.3-1) in the y direction.

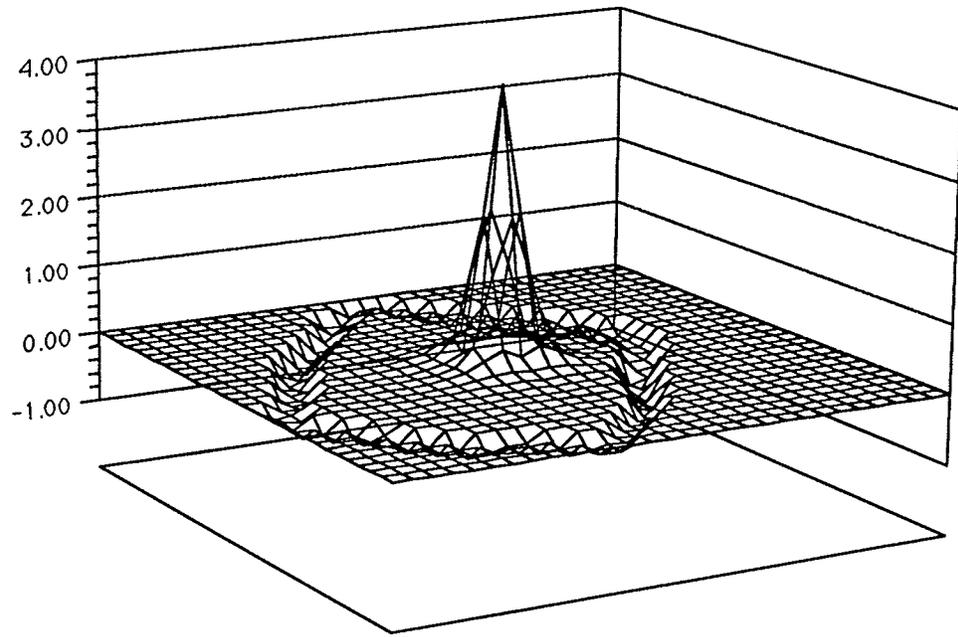


Fig. 6.3-6a. The divergence field from the Digital Differential Radiography technique obtained for the displacement fields simulated by Eq. (6.3-1). Wireframe chart.

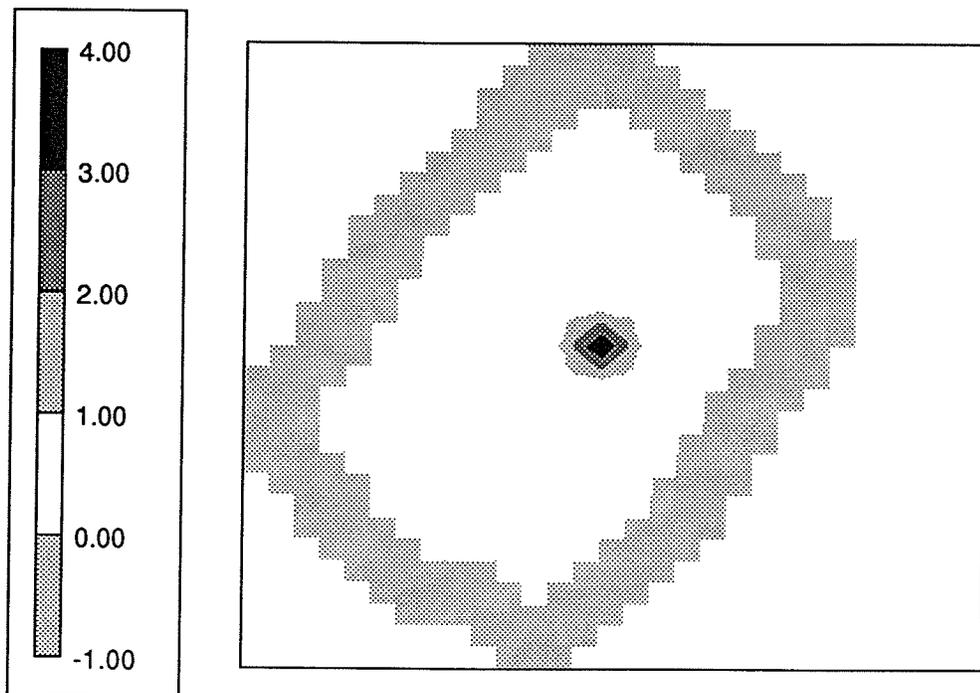


Fig. 6.3-6b. The divergence field from the Digital Differential Radiography technique obtained for the displacement fields simulated by Eq.(6.3-1). Contour chart.

## 6.4. Composite Movement Algorithm

If it is possible to predict the character and the scope of the deformation in the problem under investigation, it would be beneficial to seek for the solution by generating such a deformation. The generation, of course, should include random selection of the deformation range and the strength. The deformation (6.3-1) that was introduced earlier in this chapter and used to create the model of a growing tumor, may be a good candidate for animation by the placement algorithm. It can be done by the alteration of the transformation originally given by Eq. 4.3-2 . Because of the way the deformation (6.3-1) is produced, it can be called a *radial* algorithm.

$$\Delta i = (-1)^k \text{round} \left( \frac{m-i}{|m-i|} \right)$$

$$\Delta j = (-1)^k \text{round} \left( \frac{n-j}{|n-j|} \right)$$

(6.4-1)

$$\Delta X_I [m, n] = \frac{X_I [m + \Delta i, n + \Delta j] - X_I [m, n]}{Z} R_x$$

$$\Delta Y_I [m, n] = \frac{Y_I [m + \Delta i, n + \Delta j] - Y_I [m, n]}{Z} R_y$$

The variable  $k$  counts iterations and ensures that contraction and expansion of the deformation patterns are generated with equal probability. The meaning of all remaining variables and symbols are compatible with those introduced in Chapter 4.

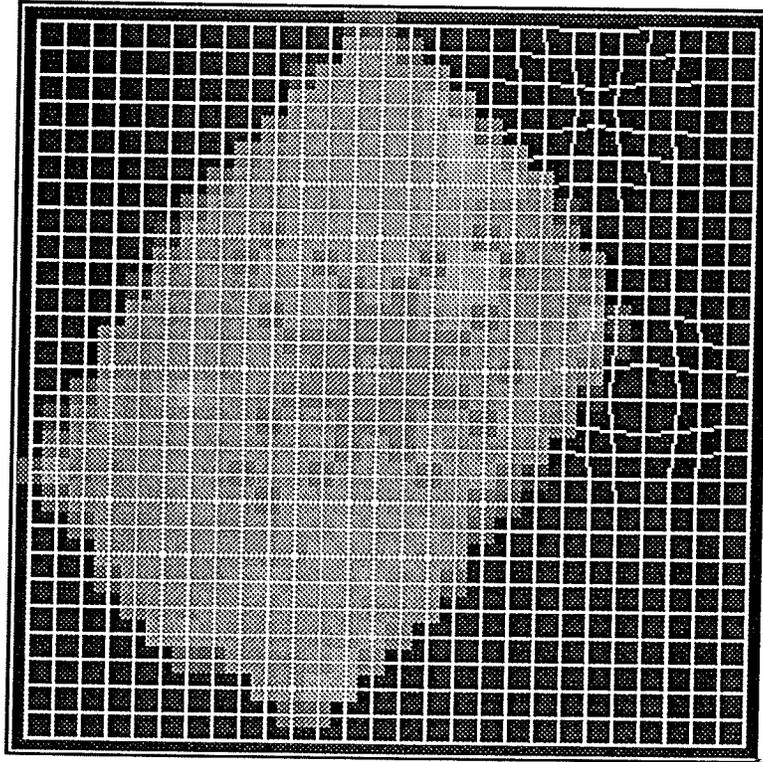
In the sequence of images in Fig. 6.4-1 from (a) through (c), several representative movements from the list of consecutive trials are depicted. The algorithm (6.4-1) was set to accept all moves that were generated. It was done by assigning and keeping the temperature-like parameter  $T$  at a relatively high value so that (4.3-5) was very close to unity. The reference image, schematically represented by the image grid, is superimposed onto the deformed image. In the course of movement generation, the grid takes different positions within the image frame. The deformed image is shown in the gray level representation and remains unchanged throughout the whole sequence. The sequence starts from the image (a), in which two moves were generated. In the right upper part of the image a contraction move was generated. In the central image part and close to its left border, in contrast, there is an example of expansion move. In images (b) and (c), further developments of the algorithm generation are shown. The reference grid as a whole stays in place with sharp variation of the mesh size.

The translation and radial versions of the deformation were generated randomly, but with probability  $P_C$  and  $1-P_C$ , respectively.  $P_C$  is a parameter that controls the proportion of these two different types of deformation. Fig. 6.4-2 shows some of the sequence generated by the algorithm with  $P_C = 0.05$ .

Despite such a low proportion of translation, these images show grid smoothness and translation.

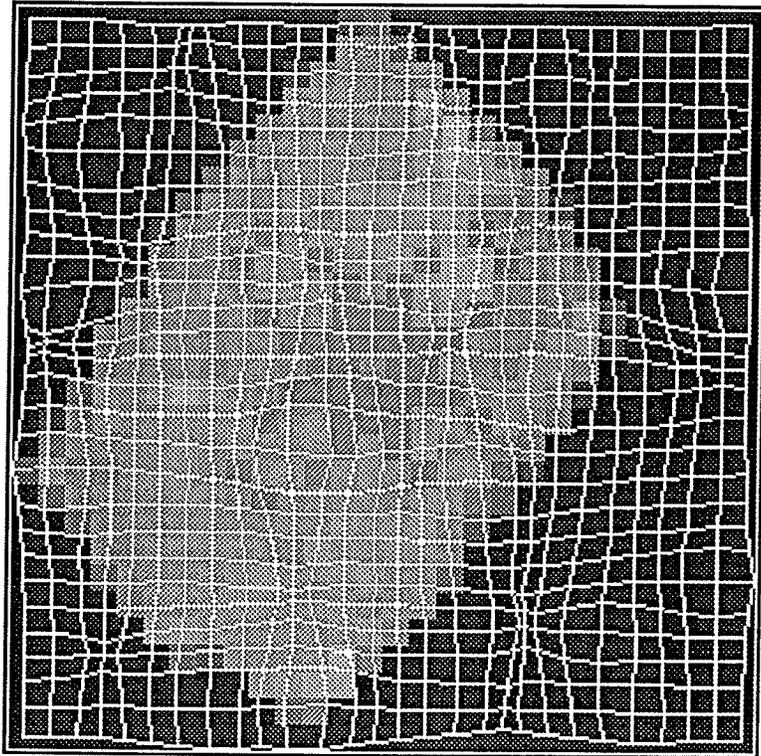
That idea of combining the translation algorithm (the most general way to generate moves) with a particular deformation pattern that is expected to occur in a particular application, forms the base of the Composite Movement Algorithm.

The concept of the Composite Movement Algorithm is explored in the experiment described at the end of this chapter.



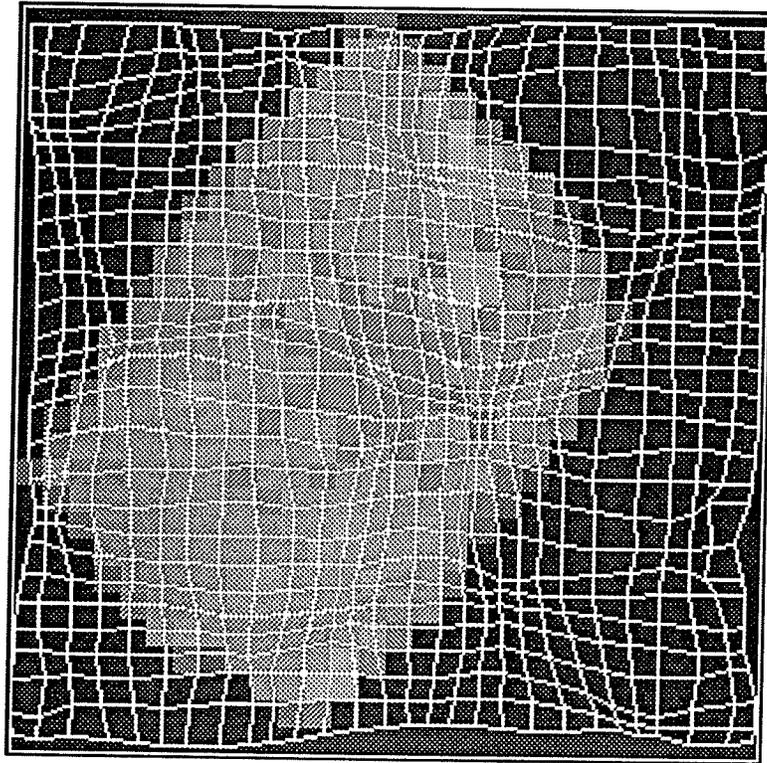
(a)

Fig. 6.4-1. (a) - (c), A sequence of movements applied to a 30 x 30 pixel image. The reference grid from an undisturbed image is moved following the sequence of moves generated by the movement algorithm according to Eq. 6.4-1.



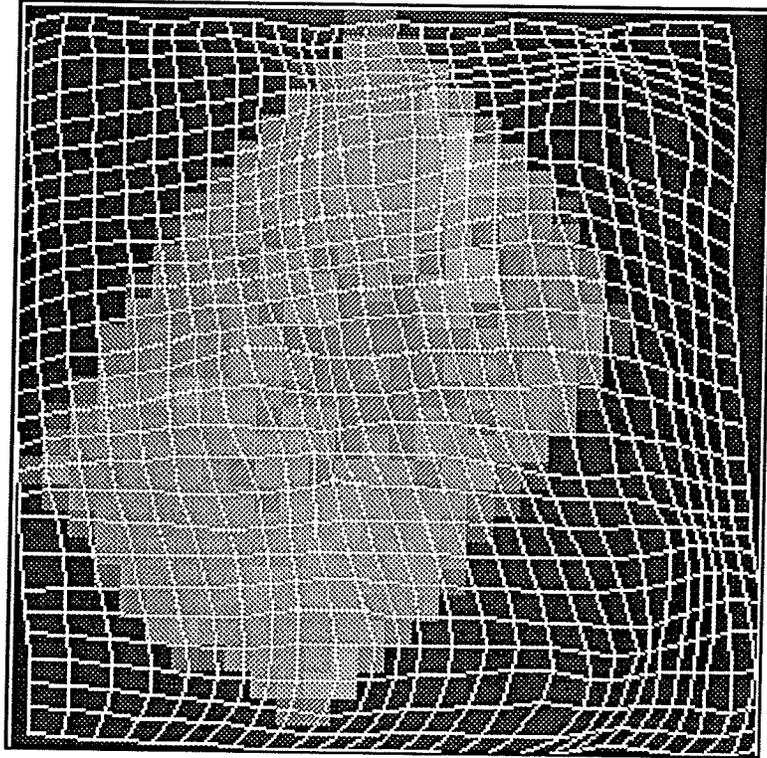
(b)

Fig. 6.4-1 contd.



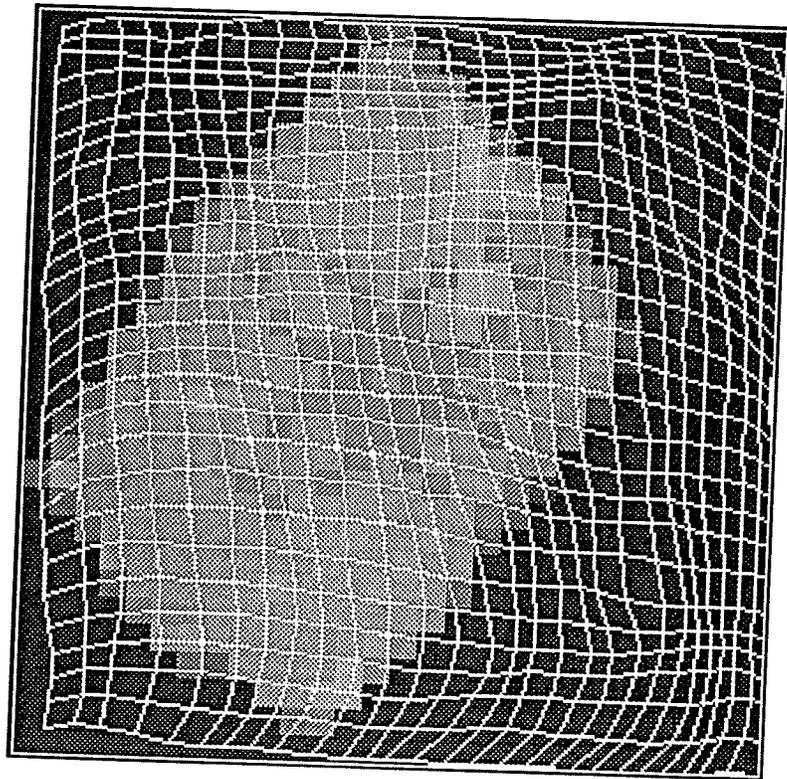
(c)

Fig. 6.4-1 contd.



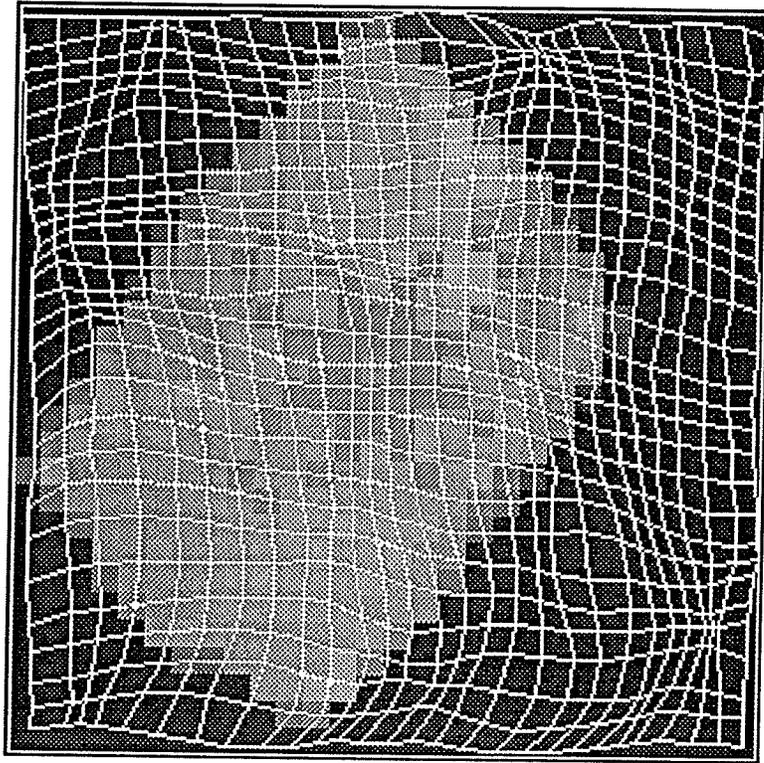
(a)

Fig. 6.4-2. (a) - (d), A sequence of movements applied to a 30 x 30 pixel image. The reference grid from an undisturbed image is moved following the sequence of moves generated by the Composite Movement Algorithm according to Eqs. 4.3-2 and 6.4-1 with  $P_c = 0.05$ .



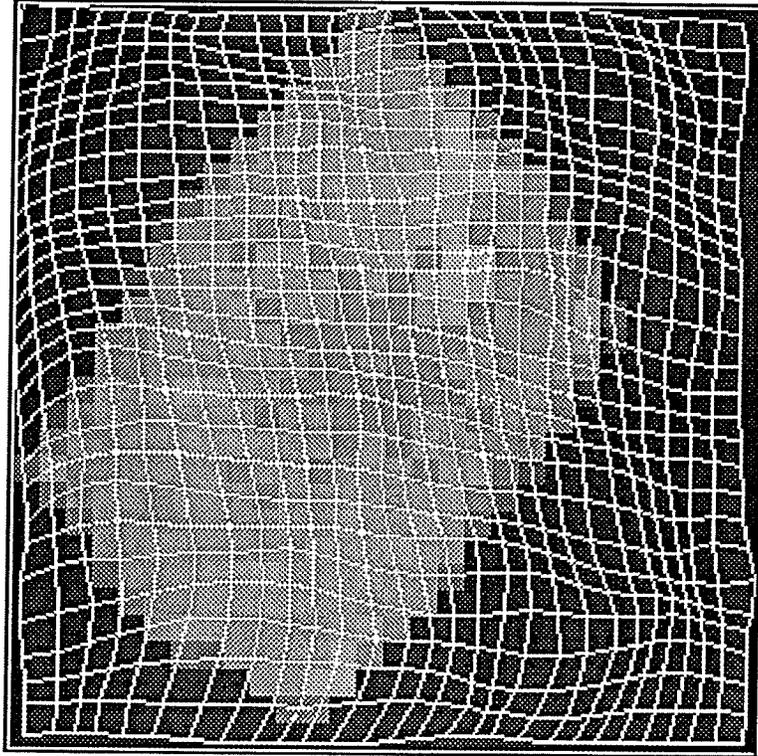
(b)

Fig. 6.4-2 contd.



(c)

Fig. 6.4-2 contd.



(d)

Fig. 6.4-2 contd.

## 6.5. Experiment 5

In this experiment the model of a growing mass, already discussed in Section 6.3, was subjected to the optimization. The displacements and the divergence field were to be recovered.

Figures E.5-1a and E.5-1b depict the recovered displacements in the  $x$  and  $y$  directions, respectively. There is some noise at those points that correspond to the tissue object. But the major feature of the applied deformation is recovered: The abrupt change of the displacement sign, simultaneously for both directions, is at the center of the deformation. It can be compared with Fig. 6.3-5.

Charts E.5-2a and E.5-2b show the displacement error (the subtraction of the simulated and recovered displacement, calculated at every point of the image plane). For the majority of points corresponding to the tissue, there is a considerable amount of fluctuation centered about the zero value. The amplitude of that variation is within the range of  $(-0.60, 1.36)$  in the  $x$  direction, and  $(-0.79, 0.75)$  in the  $y$  direction.

The divergence field (E.5-3a and b) based on the recovered displacement fields, despite some contamination with noise, reproduces the major feature of the simulated deformation, a sharp and distinguishable peak positioned at the center of the image. This coincides with the position of the simulated tumor growth at the position  $(15, 15)$ . There is a remarkable agreement between the positions of these maxima in both representations, wireform and contour charts.

The same peak in E.5-3a shows as two concentric regions in the center of the image E.5-3b. The shallow groove clearly visible in Fig. 6.3-6b is lost here. In the chart E.5-3b there is only an irregular region that approximately defines the tissue boundary.

This experiment was conducted using an Apple Macintosh II computer.

In the course of the optimization, cooling schedule (4.4-2) was adopted with the following setting:  $\alpha = 0.1$ . The flexibility limit was set to  $F = 1$  and the iteration constant was set to  $C_T = 50$ .

Fig. E.5-4 shows the correlation and the difference in the course of the optimization. The optimization was terminated after about 60 hours.

In Fig. E.5-5 the acceptance ratio and averaged time are presented.

The cooling schedule (Fig. E.5-6) shows the number of stays (stops) at a few particular temperature-like parameter levels. The last stop in the decline is particularly long. The decision to terminate the process was made because of time limitations.

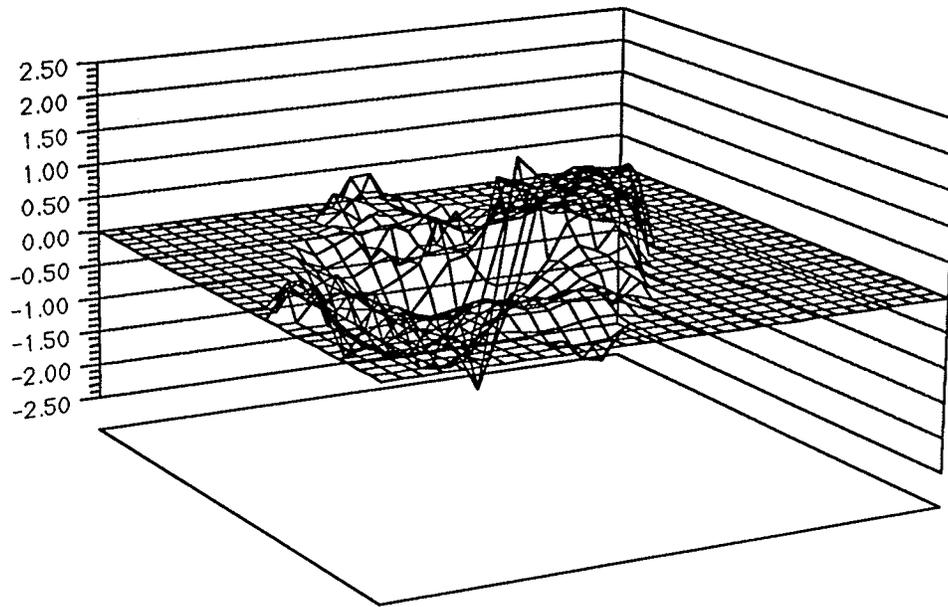


Fig. E.5-1a. The displacement field recovered from Image Correlation Technique in the  $x$  direction.

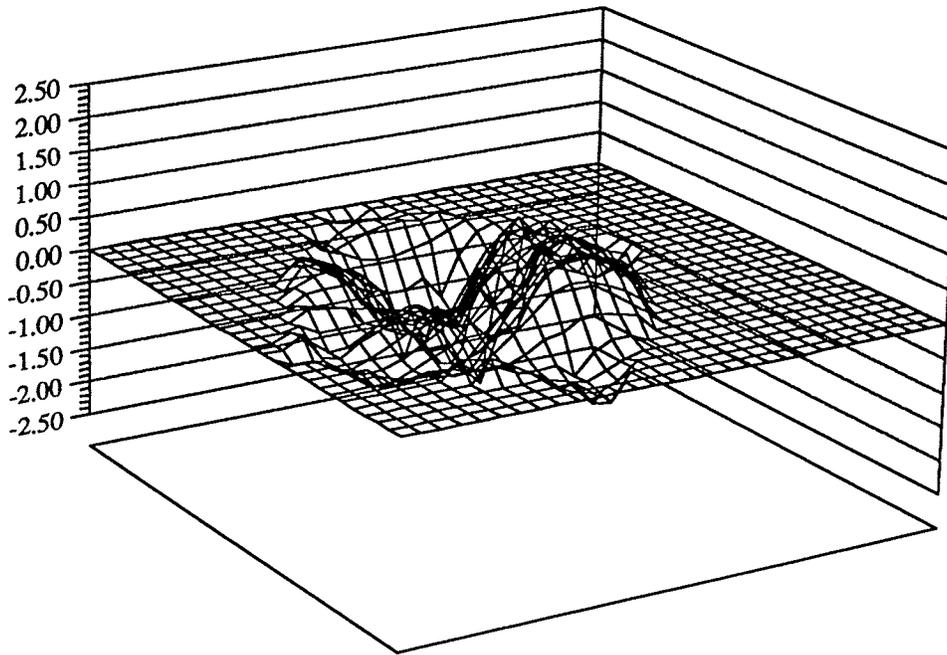


Fig. E.5-1b. The displacement field recovered from Image Correlation Technique in the y direction.

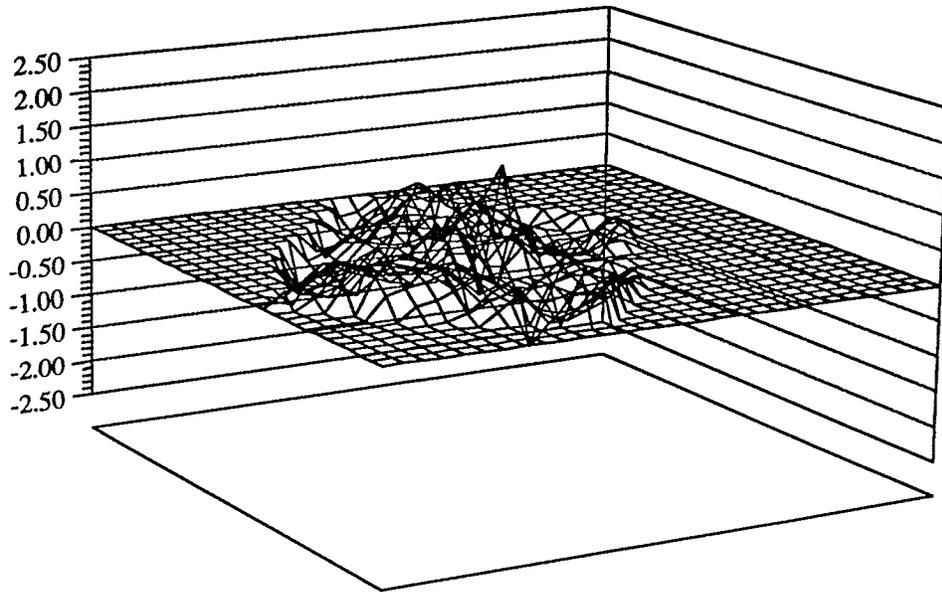


Fig. E.5-2a. The difference between simulated displacement and that recovered from Image Correlation Technique, in the  $x$  direction.

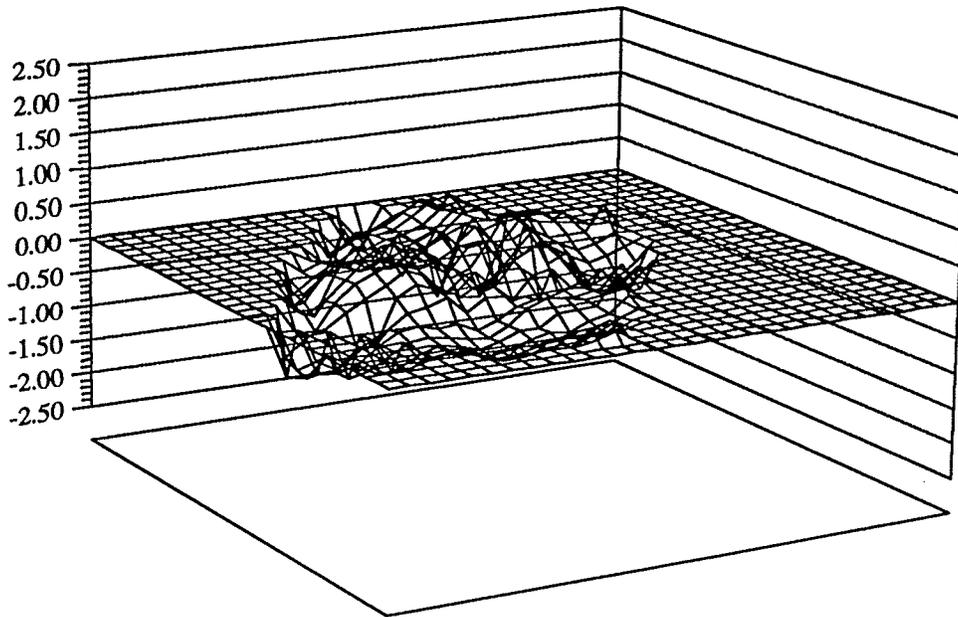


Fig. E.5-2b. The difference between simulated and recovered from Image Correlation Technique displacement, in the y direction.

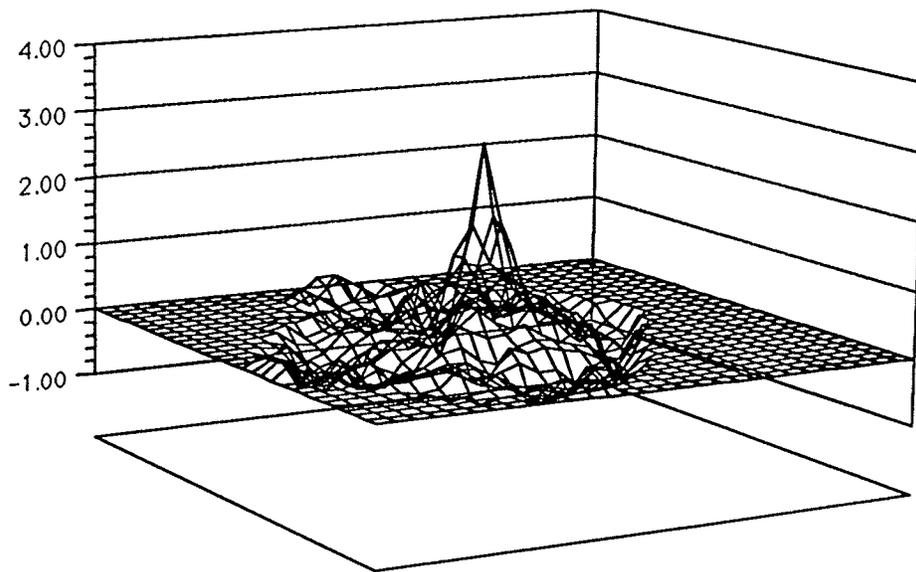


Fig. E5-3a. The divergence field from the Digital Differential Radiography technique obtained for the displacement fields recovered by the Image Correlation Technique. Wireframe chart.

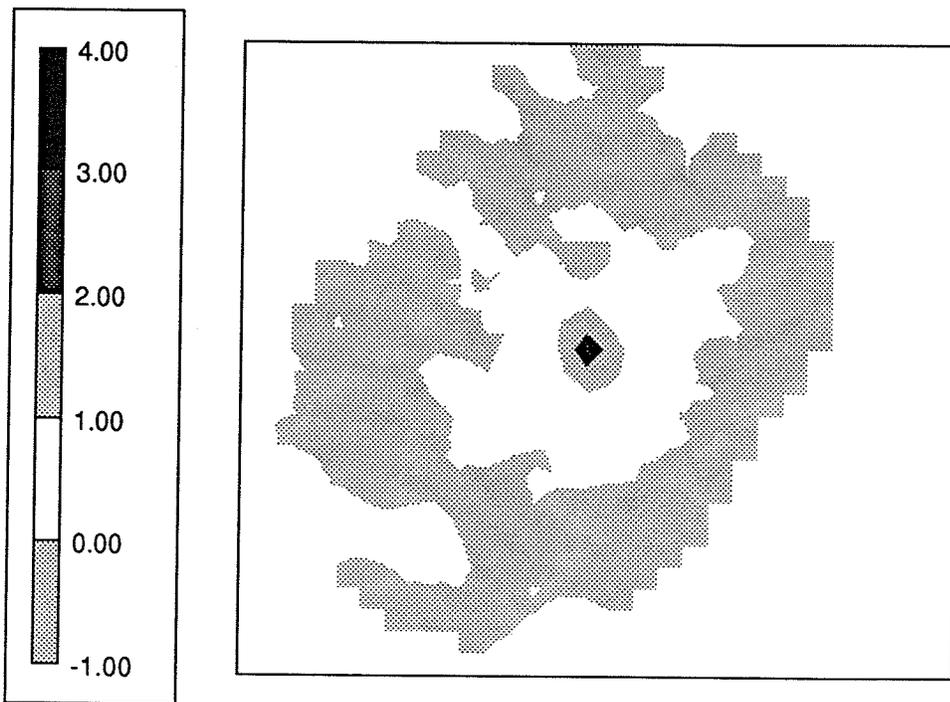


Fig. E5-3b. The divergence field from the Digital Differential Radiography technique obtained for the displacement fields recovered by the Image Correlation Technique. Contour chart.

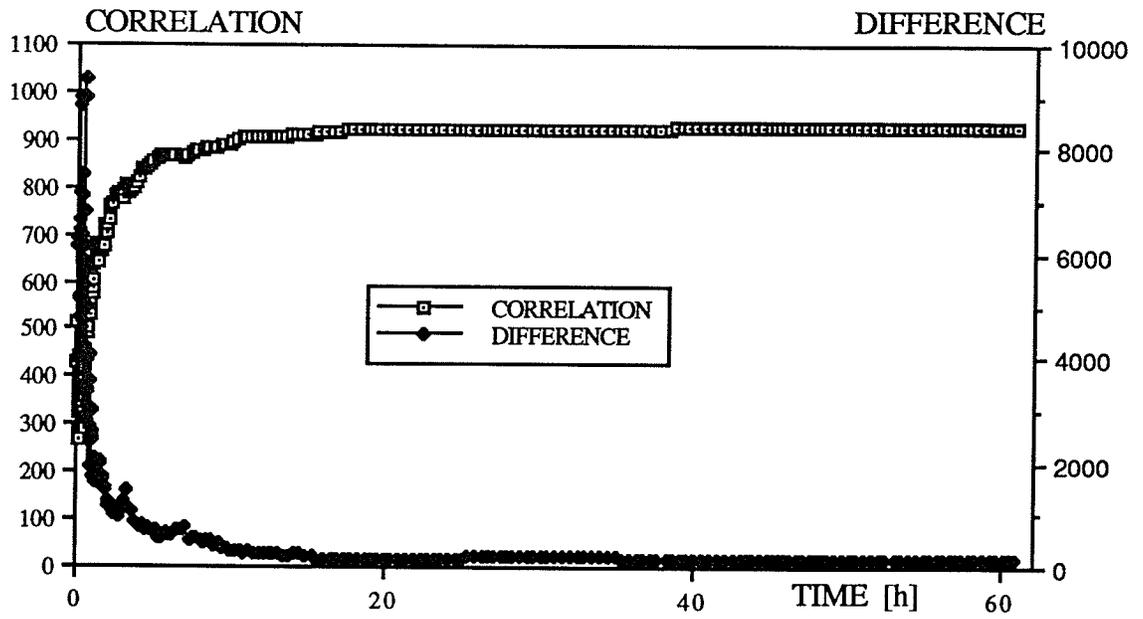


Fig. E.5-4. Image optimization. The correlation and the difference functions versus the computing time.

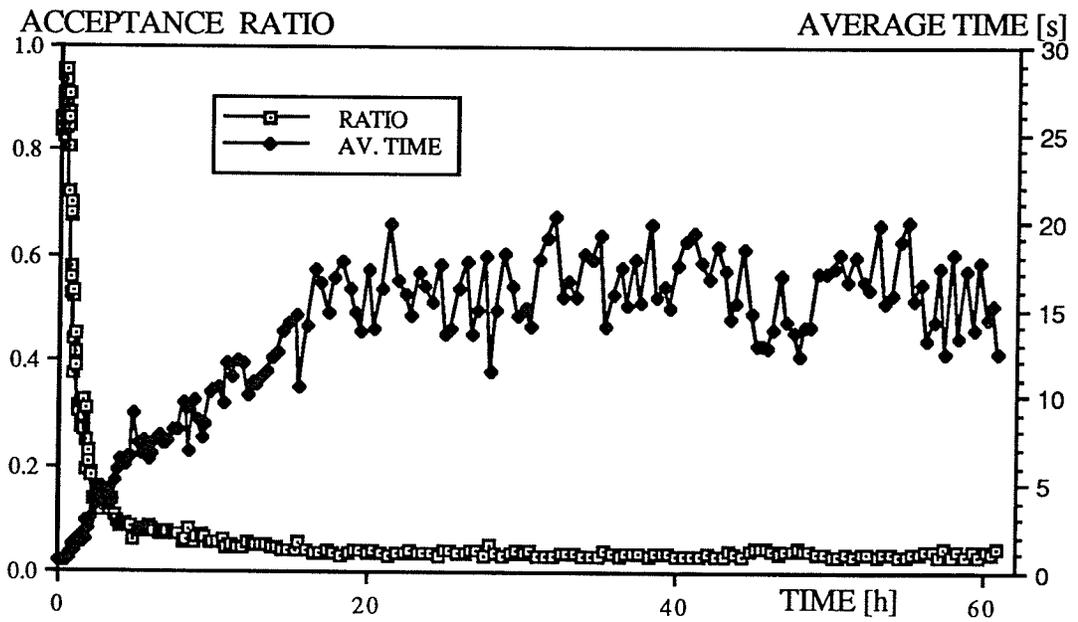


Fig. E.5-5. Image optimization. The acceptance ratio and the average time (elapsed between two consecutive and accepted trials) versus the computing time.

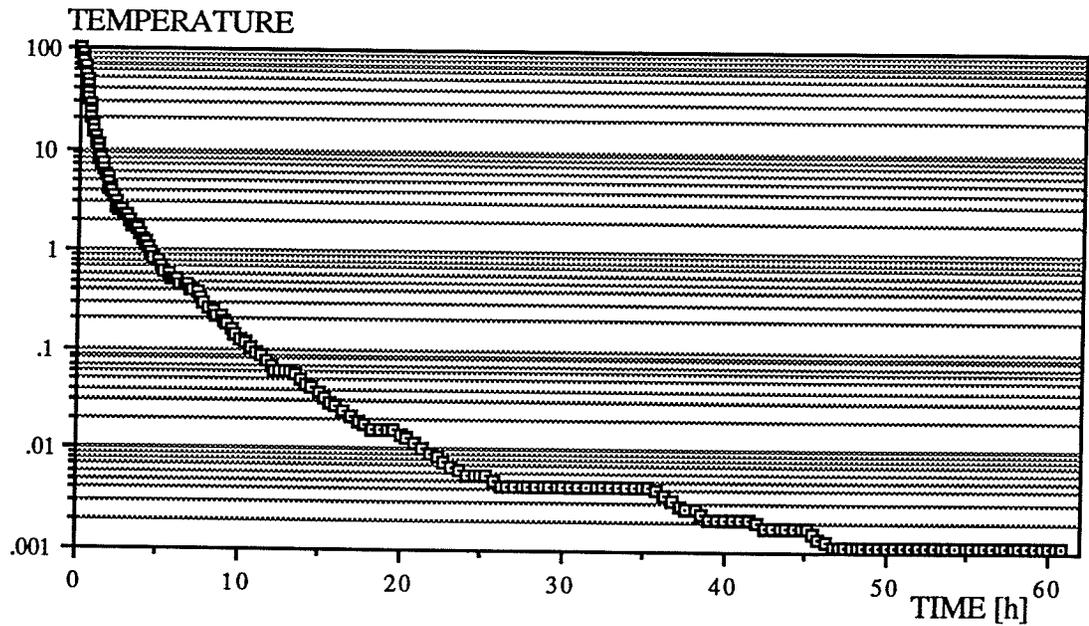


Fig. E.5-6. Image optimization. "Temperature" of the system as a function of the computing time.

## 6.6. Experiment 6

In this experiment the model of a growing mass already discussed in Section 6.3 was a subject of the optimization. As in Exp. 5, the displacements and the divergence field were to be recovered.

The differences between Exp. 5 and Exp. 6 relies on two aspects. First, Exp. 6 was conducted using a different computer, the Siemens LiteBox, and second, instead of the translation algorithm, the composite placement algorithm was used.

Figs E.6-1a and E.6-1b depict the recovered displacements in the  $x$  and  $y$  directions respectively. The major feature of the applied deformation was recovered. This result can be compared with Fig. 6.3-5.

Charts E.6-2a and E.6-2b show the displacement error (the simulated displacement minus the recovered one, calculated at each image point). For the majority of points

corresponding to the tissue, there is a certain amount of the fluctuation. The amplitude of that variation is within the range of  $(-0.54, 0.75)$  for the  $x$  direction and within  $(-0.73, 0.69)$  for the  $y$  direction. These results are better than those presented in Exp. 5.

The divergence field (E. 6-3a and b) based on the recovered displacement fields reproduces the major feature, a sharp and distinguishable peak positioned at the center of the image. Similar to results in the Exp. 5, there is a striking agreement between the positions of these maxima and position of the simulated tumor.

The peak in E. 6-3a shows as two concentric regions in the center of the E. 6-3b image. The shallow groove visible in Fig. 6.3-6b is lost here.

In the course of the optimization the cooling schedule (4.4-2) was adopted with the following setting:  $a = 0.1$ .

The partition control parameter was  $P_C = 0.3$  (30% translation moves and 70% radial moves).

The flexibility limit was set to  $F = 0.5$  and the iteration constant was set to  $C_T = 100$ .

The optimization was terminated after about 28 days. Because of memory limitations of the statistical software that was used to display the data, the following charts include only first 240 hours of the optimization. The optimization, however, did not change beyond that point.

Fig. E.6-4 shows the correlation change in the course of the optimization.

In Figs. E.6-5 and E.6-6 the difference and the acceptance ratio are presented, respectively.

The cooling schedule (Fig. E.6-7) shows the number of stays (stops) at a few particular "temperatures". The last stop that occurred at 0.00064 lasted till the decision was made to terminate the process .

Fig. E.6-8 shows the solution image after 28 days of the application run. By comparing it with Fig. 6.3-4 that depicts the simulated deformation of the reference image grid, it can be seen that the grid deformation in the center of the image was reproduced.

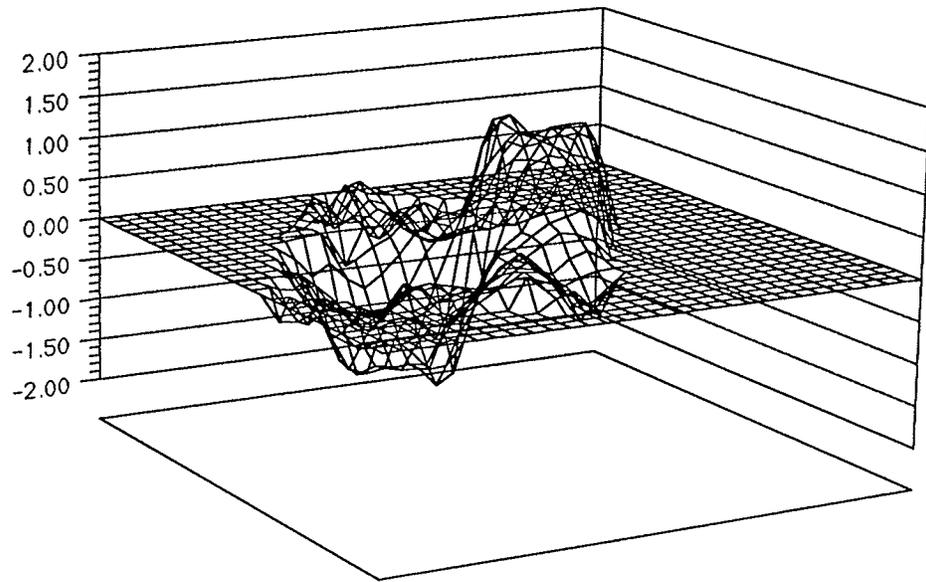


Fig. E.6-1a. The displacement field recovered from Image Correlation Technique in the  $x$  direction.

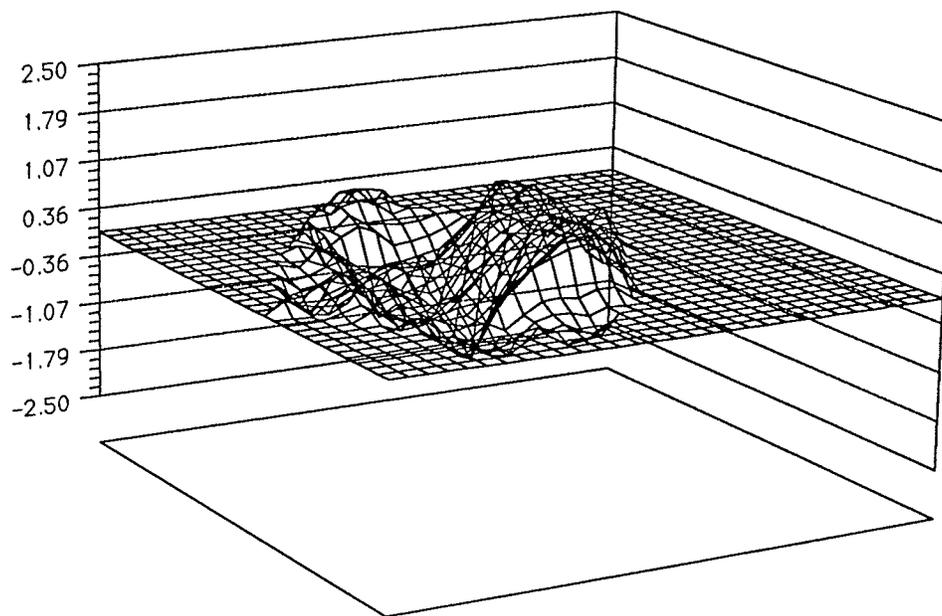


Fig. E.6-1b. The displacement field recovered from Image Correlation Technique in the y direction.

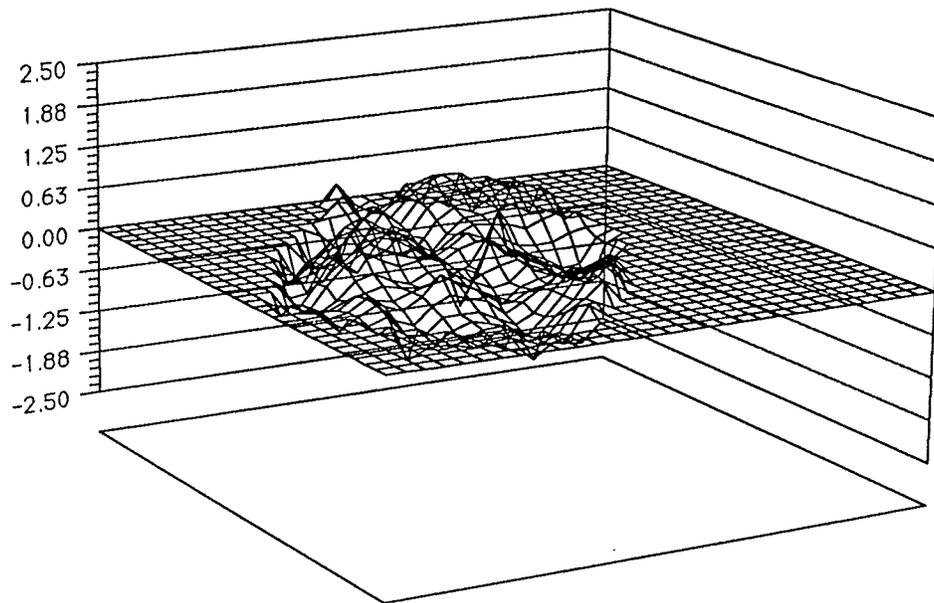


Fig. E.6-2a. The difference between simulated displacement and that recovered from Image Correlation Technique, in the  $x$  direction.

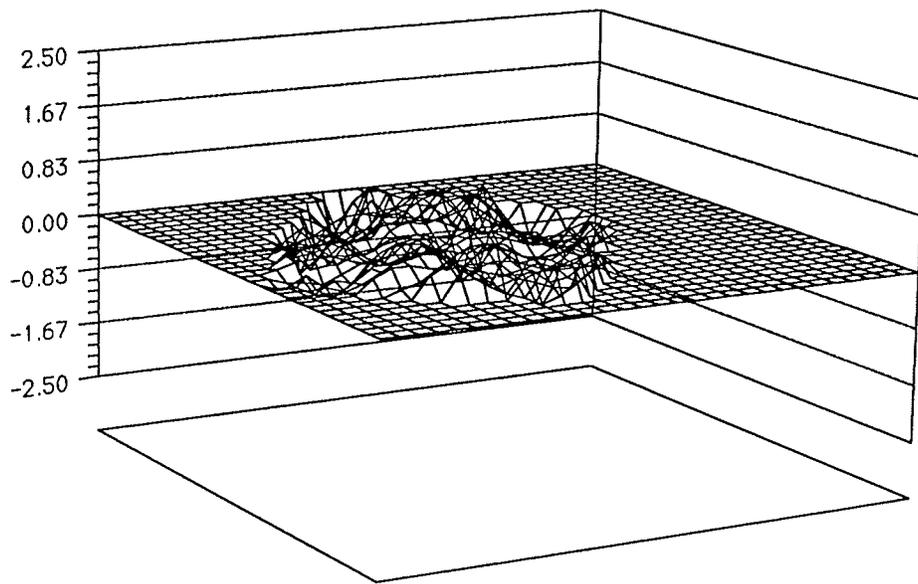


Fig. E.6-2b. The difference between simulated displacement and that recovered from Image Correlation Technique, in the y direction.

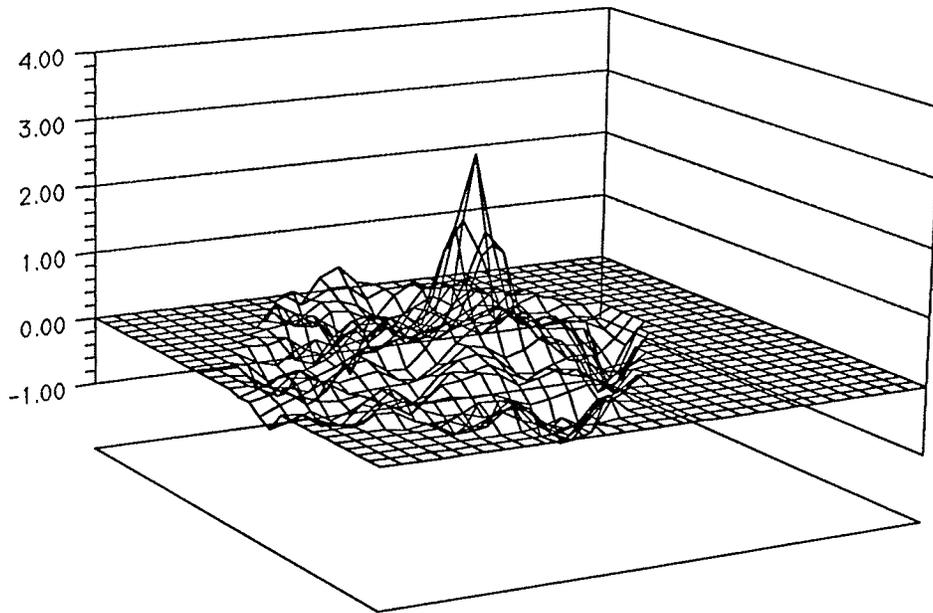


Fig. E.6-3a. The divergence field from the Digital Differential Radiography technique obtained for the displacement fields recovered by the Image Correlation Technique. Wireframe chart.

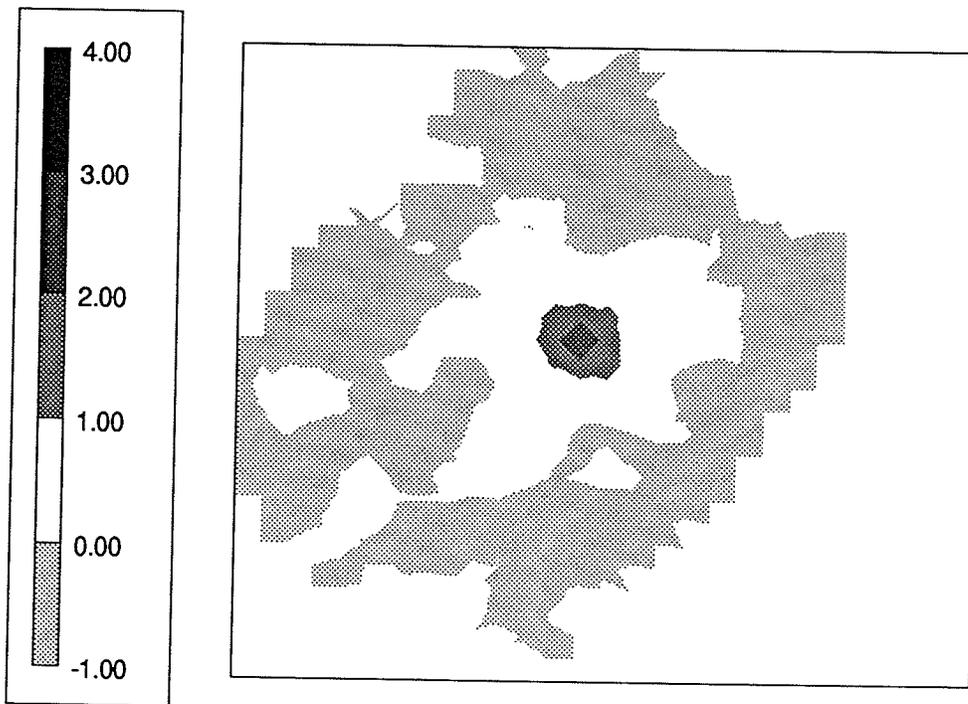


Fig. E.6-3b. The divergence field from the Digital Differential Radiography technique obtained for the displacement fields recovered by the Image Correlation Technique. Contour chart.

CORRELATION

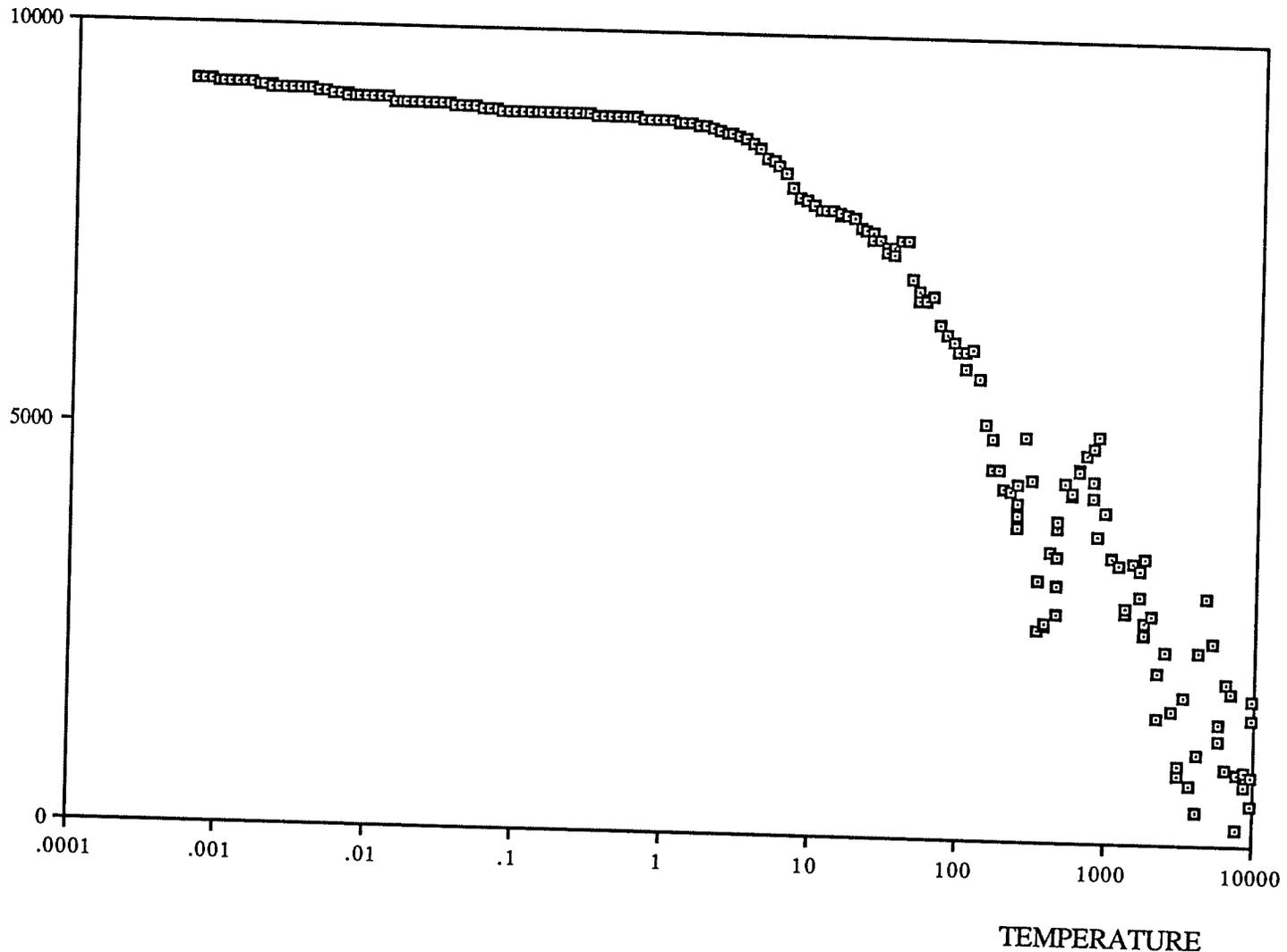


Fig. E.6-4. Image optimization. The correlation versus the *temperature-like parameter T*.

ACCEPTANCE RATIO

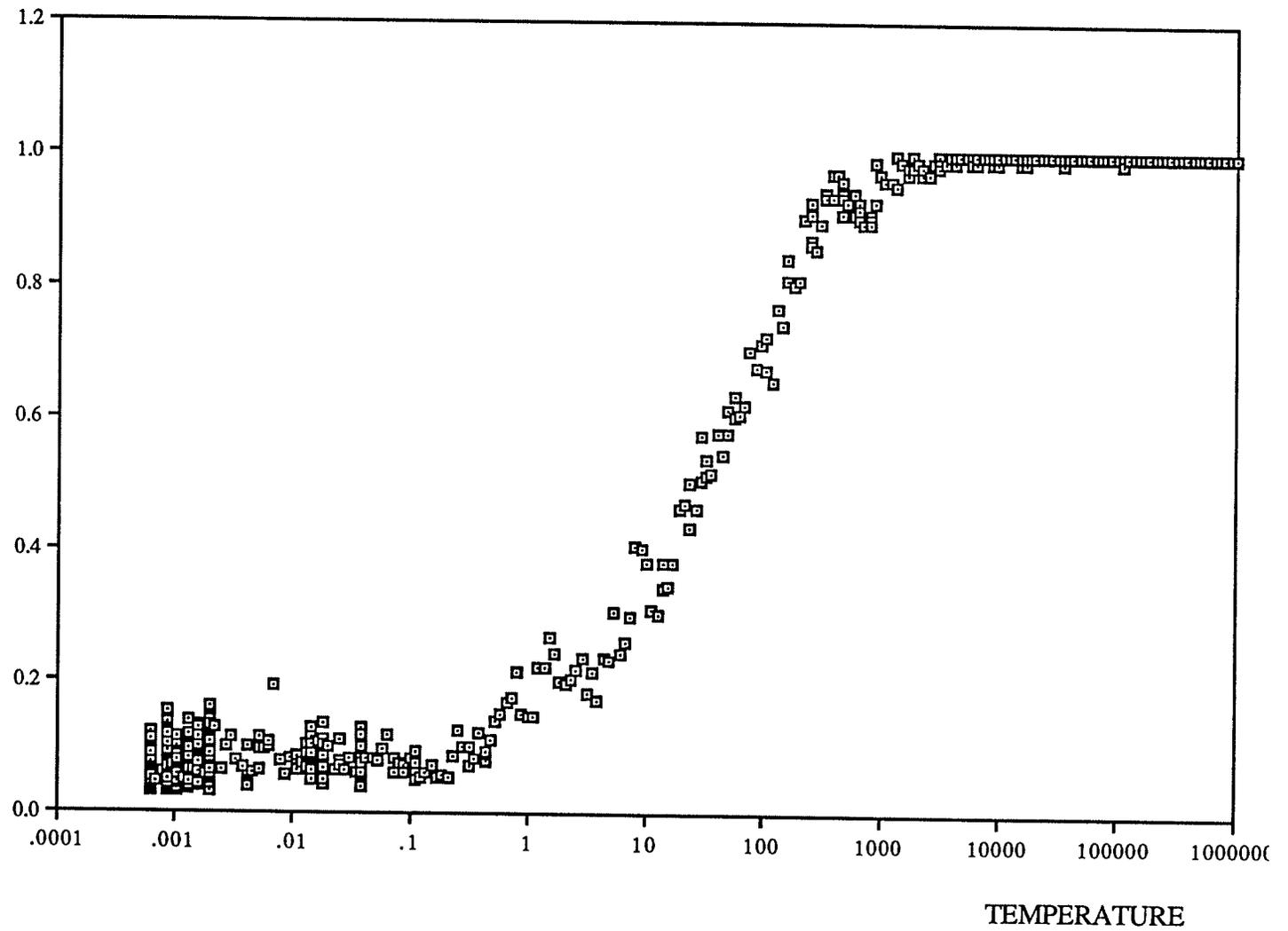


Fig. E.6-5. Image optimization. The acceptance ratio versus the *temperature-like parameter T*.

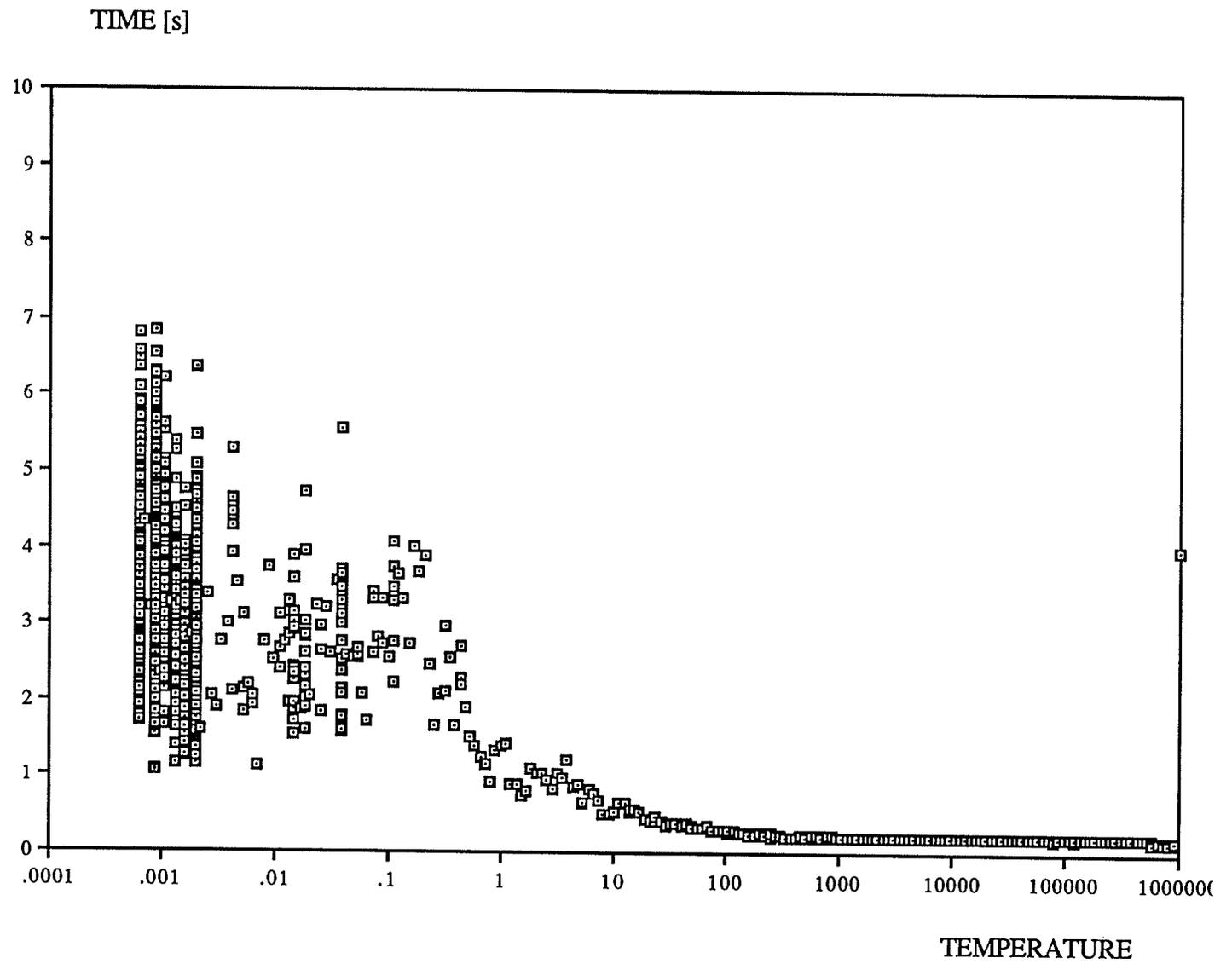


Fig. E.6-6. Image optimization. The averaged time (elapsed between two consecutive and accepted trials) versus the *temperature-like parameter T*.

# TEMPERATURE

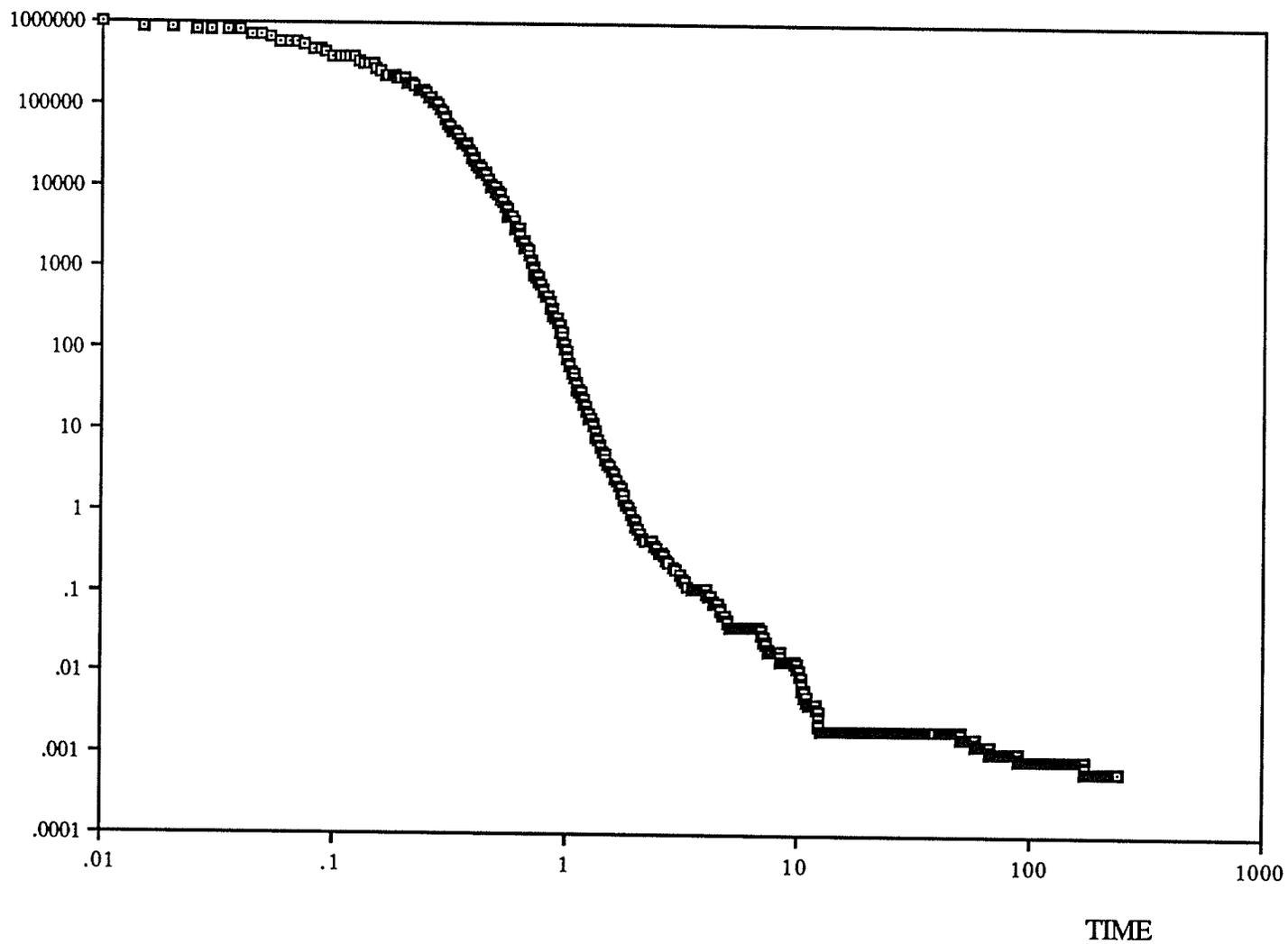


Fig. E.6-7. Image optimization. 'Temperature' of the system as a function of the computing time.

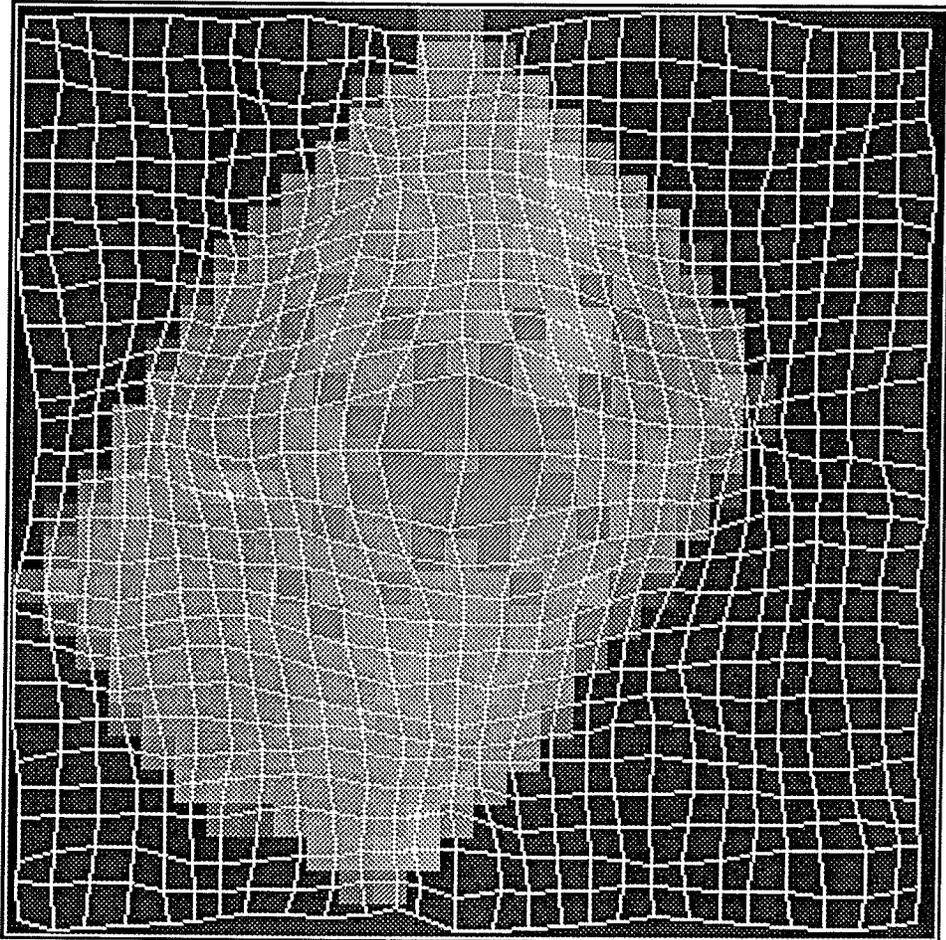


Fig. E.6-8. Experiment 6. The solution image.

## 6.7. Discussion

In both presented experiments, ICT reproduced the deformation (6.3-1) that simulated the model of a growing mass. This deformation can be analyzed in two separate stages: a fracture of the tissue structure and an introduction into the opening of a new structure. That new structure can also be viewed as introduction of noise to the deformed image. The simulation that has been demonstrated represents a difficult case as the center of the growth was chosen to be positioned within a tissue of a relatively uniform density.

In this difficult environment, in both experiments, ICT reproduced the deformation.

The recovery of the divergence maximum at the exact location of the deformation center, in both experiments, is evident. This result demonstrates that DDR could be a valuable diagnostic procedure.

The composite version of the placement algorithm proved its usefulness. The cooperative algorithm shows better accuracy of the solution as well as the narrower range of the displacement errors. The cooperative version in fact may be a very important factor in the optimization for the systems where specific type of deformation is expected to be present.

## CHAPTER 7

### FINAL REMARKS

#### 7.1. Conclusions

##### 7.1.1. Discussion on Convergence

At the base of the ICT method formulations was the following idea. In Fig. 7.1-1 a chain of data marked by solid triangles and named *before*, was subjected to a deformation. Each data point was restricted to a move only along the horizontal axis and within a displacement limit determined by its distance to the neighbor to the right or to the left. This type of deformation preserves the continuity and the order of all data points. Therefore it is conceivable to slide all data points along horizontal lines from their "before" position to fit a new one marked as *after*. It can be done in the finite number of steps, even by hand. Thus, it is possible to write an algorithm that would be capable to match both data sets of points. If we were only interpolating both sets we could have said that we were matching two functions sampled at the data points. The analogy to image matching is straightforward, we only have to extend our set of data points to the second dimension. However, in the case of an image matching, there is a complication. An image pixel, that corresponds to a data point, represents an average value of the intensity function calculated over the pixel surface. This may introduce an ambiguity in matching, especially in the case of violent deformations. The only way to overcome this is to increase the number of the interpolation curve samples.

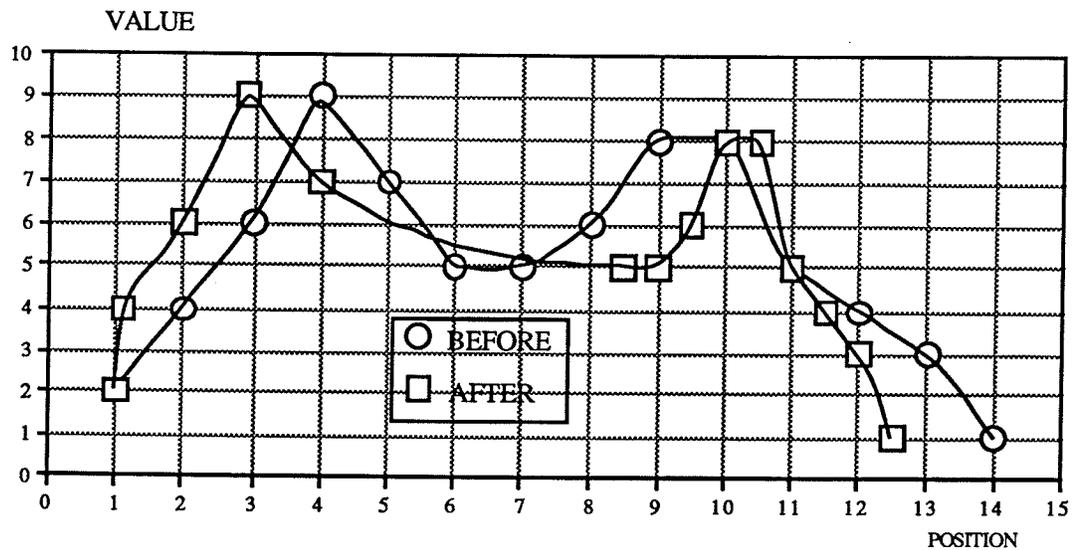


Fig. 7.1-1. One dimensional curve matching.

That simple intuition is the base for the method formulation.

ICT is successful because of the following features that occur jointly:

1. The movement algorithm that, as it is proposed in Step 1 of Chapter 4, ensures continuity of the temporary image at each stage of the optimization process.
2. The movement algorithm induces the move of the entire reference image grid at every optimization step.
3. The shape of the correlation function (4.2-1) has a strong response only to a perfect or almost perfect match.

As shown in Fig. 7.1-2, the enhanced dynamics of the correlation function occurs within a very narrow range of values that are not very much different from the value that gives the exact match. In fact, this range extends only to about 5% on both sides of the perfect match value. This means that in terms of the decision (4.3-4) the reference grid (image) "sticks" to its counterpart, the deformed image, only at the locations of the potential match (identical intensities in both images). In addition, as the temperature-like parameter  $T$

drops, these matched entries in both images that are of relatively high value settle down first. As temperature-like parameter drops further, gradually, matching takes place for image regions with the lesser and lesser intensities.

Moreover, by means of controlling the grid flexibility, and actually by limiting it from above, the grid continuity and order of the pixels are preserved.

But this is not the whole picture. We need to be aware of the fact that the conditions that have been described above, are valid and take place simultaneously for all points from the surface of the match. This becomes evident, if we recall how the movement algorithm sets into motion the entire reference image each time.

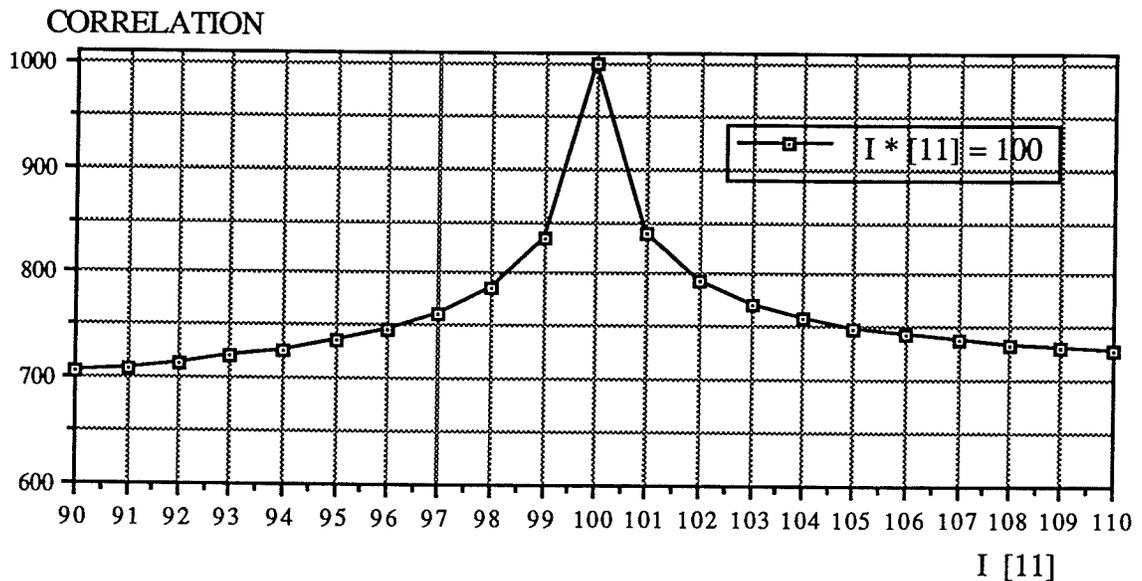


Fig. 7.1-2. Correlation function (4.2-1) for matching images given by (4.2-2) with  $I^*[1,1]=100$ . The enhanced dynamics of the correlation function occurs for the range that extends only to about 5% on both sides of the perfect match value ( $I^*[1,1]$ ).

It is not a surprise that there must be a region in a multi-dimensional space where the match of two given images is possible and the result is unique and accurate. The configuration of this hypothetical space could be spanned by coordinates such as:

-noise level,

-the compression properties of the imaged object,

-the deformation strain level,

However, the task was to demonstrate that the method works. It was done using a relatively low power computer, the Macintosh II. But its excellent graphical interface provided the opportunity to interact at each stage of development of the method. Based on the results and experience that were gained, transfer to high computation power and fast super computers will be straightforward and at the same time all pending questions could be addressed and solved in a reasonable time frame.

The Image Correlation Technique is a new method to determine the displacement field due to the deformation of a reference image. The method works equally well for deformations, including the large ones, and for non-linear cases.

The movement algorithm (Eqs. (4.3-1), (4.3-2), (4.3-3)) that was introduced, has proven to be flexible and capable of reconstruction of different types of deformation. The idea to complement the initial movement algorithm by a specific type of deformation, that is expected to occur, proposed under the name of "composite" algorithm, extends the useful range of possible applications. It also may shorten computations.

Usually in computer vision the following set of image processing operations is widely accepted: restoration, segmentation, registration and eventually matching on the top of a processing pyramid. ICT, as it is in the present form, bypasses all listed components and deals with the recorded image directly.

There is no need for control points. It is the first method that attempts, with success, to process and actually to relate images/pictures in one piece. The method opens new possibilities in image processing.

### 7.1.2. Control Parameters

For each application, there is probably a useful and an optimal range of the flexibility parameter  $F$ . It can vary with deformation, noise and the image itself. Some additional work that would help to understand the influence of the  $F$  parameter on the convergence and computation time, would be welcome. It is important because  $F$  is the second factor, apart from the cooling rate that has to be predetermined before the optimization method starts to approach the solution. For now, selection of  $F$  relies mostly on the operator's experience. Frequently,  $F$  as well as the cooling rate is adjusted and finally selected at the early stages of the optimization.

It is important to generate a sufficient number of moves at each "temperature" level, to let the system attain relative equilibrium. Almost everything in this method is interconnected. This is particularly true with respect to all controls. The iteration constant  $C_T$  and the set of cooling controls together ensure conditions for the system to approach the solution smoothly and without unnecessary stops. On the other hand, there are time constraints. Thus, for each group of applications an optimal range of parameters should be worked out. But for now, when there is no broad statistic data on convergence, a particular choice of the parameter set should be weighted by the operator's intuition and experience.

### 7.1.3. Noise

This thesis has shown the positive results of using the ICT method on test images under simulated deformations. Strains were large and deformations were non-linear.

At this stage, the question of noise was not addressed. Although, we expect that noise will introduce spatial errors in the image matching, providing there is enough time for convergence, we anticipate that the proposed algorithm will almost always give the

solution. Quantitative data on the improvement of the method's accuracy with respect to different noise filtration techniques would be most welcome.

The work presented included only matching between a real image and its simulated deformation. And it was possible to reach an exact match. For this purpose, formulation of the correlation function (4.3-2) was sufficient.

In Chapter 4 there has been proposed another formula for the correlation function (4.3-2) that may be applicable in matching of noisy images that obey Eq. (2.2-6).

It is true that introduction of noise will also introduce spatial error in the resulting displacement. But because of the shape of the proposed correlation function, for points that produce a mismatch the movement generated by the movement algorithm is not significantly restricted. That is true even for a relatively low value of "temperature" when the optimization is almost completed (relatively low dynamism for the correlation function and for a broad range of noise values: 0%-95% and 105%-120%). The correlation function, though, tends to play a considerable role in restricting pixels that are moving around, when two entries from both images are very close in value, within the range of 95%-105%. This is a particularly true when the temperature-like parameter is low. Two possible sources of errors may occur:

1. Noisy pixels from one picture may make a perfect match with the wrong and noise-free pixels from the second image, and vice versa,
2. Noisy pixels do not find their counterpart pixels in the second image and do not match.

The first case is worse: when a pixel makes a mismatch, it anchors in a wrong position and inserts undesired stresses in the whole image grid. If number of points that make such the mismatch (match with wrong counterpart due to noise) is high, the entire matching process may be in jeopardy or the resulting quality of the match may be low. For only a few mismatch points, the solution should degrade gracefully.

From the above reasoning the following two statements can be made. First, it is possible to make a successful image match, using the algorithm in its present form, providing that the noise amplitude is not greater than 5%. And second, image matching is still possible when noise exceeds the 5% limitation, but the number of noisy points has to be kept small.

One of the most inconvenient features of the method is the requirement for computer time and speed. That is the main reason why the number of experiments is limited, noise consideration is not addressed and no data was produced on the statistics of the convergence.

#### **7.1.4. Computer Implementation**

This work is the result of author's fascination for computer technique and programming in particular. It is true to say that development of the presented algorithms was accompanied by a testing process of the author's abilities. Despite of the fact that the computer implementation is complex and advanced, there is need and room for improvement and optimization of the code.

On the other hand, it has to be mentioned that the ICT method requires calculation power. Fig. 7.3-1 lists the computational speed of some present computers [Seiter, 1991]. The last data regarding new IBM instruments in development, is from the recent press release by S. Cheifet (*TV Journal of Computer Chronicles*). Speed is given in MFLOPS, an acronym for million floating-point operations per second.

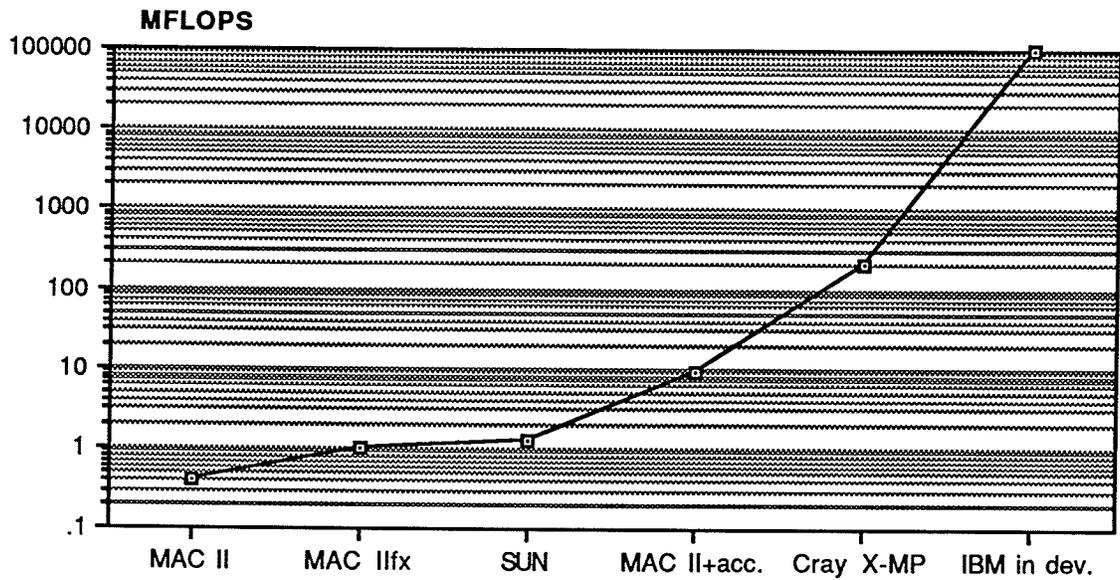


Fig. 7.1-3. Floating point operations for some present and in development computers.

The majority of the presented work was done on a computer Mac II which in fact is the slowest one among others in chart 7.1-3.

But clearly, fast computers and the technology to build even faster computers, are available. This is promising, and together with the improvements of the computer implementation of the algorithm, the method has an opportunity to be used in to the real life applications.

Although we have exploited only gray level intensity representation, instead, other image properties such as region structures, level representation, semantic divisions can be used separately or in the combination after certain algorithm modification.

## 7.2. Image Correlation Technique Applications

A new diagnostic procedure, Digital Differential Radiography for the automatic detection of a growing mass, has been presented in Chapter 6. It utilizes a novel Image Correlation Technique that brings out the displacement field. The procedure produces a diagnostic image using two images of the part of a body under investigation taken at different times.

It is very important that only the displacement be involved in the process and the solution of the problem becomes purely geometrical in its nature. There is no concern about the external forces necessary to develop the displacement. That significantly simplifies medical application and makes it feasible.

The DDR procedure requires two radiological images taken some time apart. It matches them to get the displacement field, and then calculates the digital divergence. The diagnosis can be made from a two-dimensional image of the divergence field. Presence of maxima in the diagnosis picture gives an indication of abnormal tissue growth.

In the computer experiment to test the procedure, I have used a CT image of a mastectomy sample and simulated the deformation due to the tumor growth. The simulation that has been used represented a difficult case the center of the growth was chosen to be positioned within a tissue of a relatively uniform density.

The procedure has high requirements for the computer time and speed and this is the main reason why presented results are limited to only two experiments. However, the goal was to demonstrate that the procedure works.

Digital Differential Radiography could be made more practical by making the algorithm more efficient and by using faster computers. As the first algorithm to automatically match an image to its distorted version, point for point, without hand generated control points, it represents the beginning of a new, promising diagnostic tool.

The new procedure is applicable to all imaging modalities, especially Computed Tomography and Magnetic Resonance Imaging, and if proven robust in the presence of noise could be also used with others.

DDR is easily generalized to include three-dimension images.

In general, ICT may be implemented in all these instances where matching by using templates were adopted. Fig. 7.1-1 may very well be an example of signal matching [Friedman, 1968]. Among others, voice recognition could be one of the possible applications.

ICT can form a basis of a new method of image reconstruction from projections. Based on the algorithm presented in Chapter 4, Mazur E. very recently proposed a new and promising method of reconstruction [E. Mazur, 1992].

### **7.3. Biological Atlases and Other Related Methods**

Up to this point references were made only to works that were pertinent to the development of the ICT method. In contrast, the following short review is the result of a search of recent literature. I refer to those positions that bear some kind of relevance to my work.

Most of this work is related to the brain. The common assumption made is that an atlas brain model and an individual brain image data are topologically equivalent, with internal structures mutually deformed. A research group from the Montreal Neurological Institute of McGill University used a multiresolution registration method designed for automatic identification of 3D MRI images of a human brain [Collins, preprint]. They introduced a volumetric brain structure model (VBSM) that contained volumetric and geometric data. The volumetric data consisted of a grey level intensity, the magnitude and the direction of a gradient. All were given for a number of different scales of mean MRI volume determined

on a certain set of volunteers. The geometric data formed a set of polyhedral objects that could be distinguished by anatomical names, tissue classifications and so on.

The proposed process of automatic segmentation consisted of finding a transformation function between the individual patient data and a predetermined volumetric brain structure model. This was done by calculating the 3D cross-correlation. The affine transformation was determined by optimization over translations, scaling and rotations, which represent a set of linear parameters. For local matching, grey-level gradients and the direction information were used to refine local non-linear deformations. For each scale space level a grid with spacing equal to the scale was identified. At each crossline point of the grid, within space limited by a radius equal to the actual scale, the cross-correlation with respect to the intensity, the gradient and the direction was evaluated. The best set of local deformations was then found at each grid node. The important feature of the algorithm is that the solution of each scale level is considered an input data set for the next, higher level in the resolution hierarchy.

Twenty MRI scans regarded as normal 3D brain scans were used to create the VBSM model. The volumetric data associated with different levels of the scale was arranged to form a pyramidal structure. The averaging of the volumetric data was done manually. Five data sets were prepared to accommodate the scale range (2 mm to 32 mm). However, non-linear matching was tested only on a model whose cross-section consisted of two concentric ellipsoids representing the brain and two smaller ellipsoids inside that represented ventricles. A simulated volumetric model was formed by distorting the original one using a thin plate spline warping deformation. Both the original model and the simulated one were on  $256 \times 256 \times 120$  matrices with voxel size equal to 1 cubic mm. The r.m.s. error for the recovered deformation was 9.2 mm for linear recovery and 6.4 mm for non-linear recovery (all at a scale equal to 4 mm).

A similar multiresolution approach with elements of elastic deformation was applied by Bajcsy and Kovacic [Bajcsy, 1989]. This method relied on preliminary extraction of a

solution at the coarsest level. Then an approximate solution from the first level was passed on to an intermediate (finer) level, and after a necessary refinement the resulting approximate entered the finest level. After the third level correction, the solution was used to produce an incremental deformation that was checked again throughout the 3-level matching extractor described above. The whole process was repeated until the desired matching was achieved. Each resolution level was obtained from the previous one using resampling by a factor two. Matching was performed in two stages. First, global matching was applied that involved scaling, translation (based on aligning the centers of "mass") and rotation (around the center of "mass"). This type of matching was performed only at the coarsest level. To perform elastic matching a certain preconditioning was done to both CT and atlas data. Also a suitable choice of landmark structures was selected:

"Unfortunately, edges of only a very limited number of structures can be detected in the CT brain. So far two matching structures have been used, the outer edge of the brain atlas was matched to the outer edge of CT brain and brain atlas ventricles were matched to the CT brain ventricles. Other anatomical structures in the brain atlas are deformed as a side-effect of ventricle and outer edge matching because deformations propagate through the elastic matching process. For matching purpose the atlas brain ventricles and the rest of the brain were colored using an average grey value of the corresponding regions in the real CT brain." [Bajcsy, 1989].

The elastic matching was performed on each multiresolution level.

As a similarity measure the correlation function projected onto a complete system of Hermite polynomials was evaluated at each grid site and its close neighborhood ( $3 \times 3 \times 3$ ). Locally, to maximize the correlation function (to make a better match), a force (proportional to a gradient of the correlation) was assumed to be exerted upon the local surrounding of each grid point. This force was used in the Navier's elastic equilibrium equation. The solution of that equation was a new value for the displacement that had to be applied to the local neighborhood in order to get better match. The solution

assumed an isotropic and homogeneous body and any stretch was considered not to produce perpendicular shrinkage. By iteration the final solution (matching) was obtained.

The method was applied to match brain images from two existing atlases (i.e. two different brains). Other examples matched real 3D CT data with brain atlas models. The method was tested for different sets of the elastic constraints.

Another approach based on the prior knowledge of landmark points was taken by Bookstein [Bookstein, 1989], [Bookstein, 1991]. Landmark points are points that have biological labels and usually their relative spatial locations are known as well. This method has previously been pursued and reviewed in our laboratory [Zhou, 1991].

A suitable general purpose interpolation function  $r^2 \log r$  was introduced to medical applications by Bookstein [Bookstein, 1991] to connect individual data to its counterpart in an atlas. The function satisfies a biharmonic equation and was found to model a thin steel plate tacked in the  $z$  direction at the position of landmark points. This deformation corresponds to a spatial arrangement of the sheet for minimal energy. The two-dimensional data are interpolated by a spline function formed as a linear superposition of an affine term with the sum of two-vector multiples of the function described above. The term  $r$  has the meaning of the Euclidean distance of a given point from each landmark point in turn. For the set of  $N$  landmark points there is a numerical solution that involves an  $(N+3) \times (N+3)$  matrix. The resulting transformation will map a set of landmark points from one object to a corresponding set of another object.

The application of the interpolation based on landmarks seems to be the most appropriate in production of atlases of different body parts, especially of the brain and parts that are adjacent to it. The soft tissues of the brain are in fact in a strict confinement sitting within the skull and supported by other tissues. Neglecting trauma cases and early development, the brain of a particular individual remains unchanged with respect to its shape, size and relative spacing between its constitutive parts.

An average biomedical atlas provides useful information about particular brain structures. It is commonly assumed that healthy brains to a certain degree have the same topology and differ only in shapes. Because of that, it may be possible to associate a brain atlas with all parameters of a geometrical, anatomical or even metabolic nature as well as mutual relations between them. There is a well established relationship between the spatial distribution of landmark points and the disease, age or, even in case of bone remains, intermediate stages of the development of species.

Bookstein suggests [Bookstein, 1991] using the thin-plate spline concept for the formulation of biomedical atlases, by utilizing the information carried by a set of landmark points. The morphometric technique includes:

"... the construction of a deformation relating any configuration of landmarks to any other via *thin-plate splines*, the extraction of statistically useful 'components' or 'factors' of the deformations, the application of the spline to deform individual clinical images into the geometry of the average shape..."

The deformation is viewed as composed of partial warps that are eigenfunctions of some quadratic form that is equivalent to an energy operator in plate bending. The idea is demonstrated using a configuration of 13 midsagittal MRI landmarks taken from nine student volunteers.

There are number of works that utilize Monte Carlo methods in the field of image restoration. In a recent paper by Grenander et al. [Grenander, 1991], the concept of a deformable template was used to remove noise from degraded images. The procedure has proven to be robust for different types of noise and degradation. Gaussian noise was generated with a variance up to 0.2 for the degradation of the pixel values that were considered from the [0,1] interval. Also degradation using uniformly distributed blocks of black (zero intensity) was introduced and successfully tested. The restoration relies on prior knowledge of the location and approximate orientation of of an object. It was suggested that finding these was to be dealt with at a separate initialization stage, before the

deformable templates method can be used. If both, the template and the data image, were located at different coordinates of a picture, the algorithm was able to adjust and produce a successful restoration for the rotation less than  $\pi/8$  (22.2 degrees) or when a translation of the entire object was less than 1% (0.64 pixels).

The method is sensitive to the influence of the image boundary. If an object is placed close to the boundary or fills the entire image, parts that are adjacent to the boundary are transformed very differently from the parts that are closer to the center.

The article presents restoration results of an X-ray image of a human hand. A variety of images degraded from that image were produced (data images) to test the restoration algorithm. The template consisted of a digitized picture of an X-ray of a different hand, basically in the same configuration. This hand image was initially smoothed and placed in the image center against zero background to form the template.

There is an apparent difference between the template hand and the reference hand that was used to produce the data images. (These are presumably two left hands of two different people.) However, the spatial arrangement of the anatomic constitutives as well as overall position of both hands were similar.

Deformation of the template and restoration of the reference object was achieved through the dynamic Monte Carlo approach. The underlying theoretical formalism is very sophisticated. The basic idea is that the restoration object is generated by a set of transformations applied to the template. The template itself focuses all important features on a given class (set) of objects:

"The transformations are set so as to preserve certain features of the template that are characteristic to the set of objects....In the spirit of the Bayesian statistics, a prior distribution is specified for the set of transformations rather than directly for the space of the images, reflecting an assumption as to which transformations are more likely, or in other words, which transformations preserve the characteristic features of the object set. The *true* object is then assumed to be a realization of the random transformation applied

to the template, and the *true* image is generated by the true object via the prescribed mapping."

The authors point difficulties, such as the transformation and the rotation of a template with respect to the data object influencing the restoration quality. And there is little said about the object deformation itself, as the object and the template are basically the same. Moreover they limited themselves to one example of the hand object and image restoration degraded by variation of noise rather than the geometrical deformation. The proposed restoration itself provides an insight into the structural understanding of pictures (Grenander, 1988).

Another example of the restoration of an object degraded by additive Gaussian noise and blurring its image is the work by S. Geman and D. Geman [Geman, 1984]. Pixel gray levels and the orientation of edges are viewed as energy states. By applying the annealing technique states that correspond to lower energies (representing an enhanced image) are refined.

An interesting approach can be found in work by Witkin et al. [Witkin, Terzopoulos, Kass, 1987]. It combines some aspects of a variational surface reconstruction and search through the scale space with a signal matching procedure. Practical implementation involves the minimization of energy (a function of the deformation) that is composed of two terms. One of them is responsible for smoothness of the reconstruction and the other serves as a similarity measure. The optimization process starts at a large scale where the signal is smooth and the solution relatively easy to extract. Then the solution is tracked down for declining values of the scale parameter.

The method recovers the deformation between two one-dimensional signals that have been deformed with respect to each other. An example of motion detection was given. In another example, by simple scanning of an image plane the method was applied to a pair of images to form a stereogram.

A similar approach to the stereo vision problem can be found in recent work by Barnard [Barnard, 1992]. By using a simulated annealing approach he was able to determine dense arrays of disparities resulting from two stereoscopic images as opposed to the other conventional approach such as feature matching. He assumed that matched points were of the same or similar intensity and that the resulting disparity image had to be smooth. Disparity itself was associated with an energy concept and was subject to minimization by simulated annealing. The energy function was defined locally around any point of a left image as the sum of absolute intensity differences of points from the left image and the right one. The position of the point from the right image resulted from the current disparity. To preserve smoothness conditions an additional term was added to the energy function. That term was a sum of the disparity differences taken from eight pixels surrounding the local one. The optimization was carried out by a random selection of a disparity parameter in all sites of the disparity (resulting) image.

## 8. BIBLIOGRAPHY

1. Aarts, E., Korst, J. Simulated annealing and Boltzmann machines. G. B.: John Wiley & Sons; 1989
2. Alan, R. Tissue engineering. USA: Alan R. Liss; 1988. (UCLA Symposia on molecular and cellular biology; v. 107).
3. Alexandrou, A. N. An inverse finite element method for directly formulated free boundary problems. Int. J. Num. Eng.. 1989; 28: 2383-2396.
4. Amamoto, D. Y., Kasturi, R., Mamourain, A. Tissue-type discrimination in magnetic resonance images. 10th International Conference on Pattern Recognition. Atlantic City, NJ; 1990: 603-607.
5. Amit, Y., Grenander, U., Piccioni, M. Structural image restoration through deformable templates. J. Am. Stat. Assoc.. 1991; 86(414): 376-387.
6. Baird, H. S. Model-based image matching using location. USA: The MIT Press; 1985; c1985. (ACM Distinguished Dissertations).
7. Bajcsy, R., Kovacic, S. Multiresolution elastic matching. Comput. Vision Graph. Image Process.. 1989; 46: 1-21.
8. Barnard, S. T. A stochastic approach to stereo vision. Proc. Natl. Conf. Artif. Intell. AAAI-86. Philadelphia, PA; 1976: 676-680.
9. Bartels H. Richard, Beatty C. John, Barsky A. Brian. Splines for use in computer graphics and geometric modeling. USA: Morgan Kaufmann Publisher; 1987
10. Bekey, G. A. System identification - an introduction and a survey. Simulation. 1970; 15(4): 151-166.
11. Bergan, P. G. Finite elements in non-linear mechanics. Norway: Tapir Publishers; 1978
12. Bernadou, M., Boisserie, J. M. The finite element method in thin shell theory: application to arch dam simulations. USA: Birkhauser; 1982
13. Bookstein, F. L. Coordinate systems and morphogenesis. Connelly, T. G., ed. Morphogenesis and Pattern Formation. New York: Raven Press; 1981: 265-282.
14. Bookstein, F. L. Distortion correction. Toga, A. E., ed. Three-Dimensional Neuroimaging. New York: Raven Press; 1990: 235-249.
15. Bookstein, F. L. The measurement of biological shape and shape change. Berlin: Springer-Verlag; 1978
16. Bookstein, F. L. Morphometric tools for landmark data. : Cambridge University Press; 1991
17. Bookstein, F. L. Principal warps: thin-plate splines and the decomposition of deformations. IEEE Trans. Pattern Anal. Machine Intelligence. 1989; 11(6): 567-585.
18. Bookstein, F. L. Random walk and the biometrics of morphological characters. Evol. Biol.. 1988; 23: 369-398.

19. Bookstein, F. L. Thin-plate splines and the atlas problem for biomedical images. author's personal copy
20. Borst, A., Egelhaaf, M. Principles of visual motion detection. *Tins*. 1989; 12(8).
21. Bounds, D. G. New optimization methods from physics and biology. *Nature*. Sept. 1987; v.329: 215.
22. Brodland, G. W. Large axisimetric deformation of thin shells of revolution. Canada: University of Manitoba; 1985
23. Carlbom, I., Terzopoulos, D., Harris, K.M. Reconstructing and Visualizing Models of Neuronal Dendrites. *Scientific Visualization of Physical Phenomena*. Patrikalakis, N. M. ed. Tokyo: Springer-Verlag; 1991: 623-638.
24. Ceperley, D., Alder, B. Quantum Monte Carlo. *Science*. Feb. 1986; vol. 231: p.555.
25. Chen, H. Y. L. Smooth muscle rheology: In search of specimen. Schmid-Schonenbein, G. W. *Frontiers in biomechanics*. USA: Springer-Verlag New York Berlin Heidelberg; 1985.
26. Cheung, Y. K., Yeo, M. F. A particular introduction to finite analysis. Great Britain: Cambridge University Press; 1979
27. Chew, P. H. Determination of material properties of biological tissues: Pericardium. Schmid-Schonenbein, G. W. *Frontiers in biomechanics*. USA: Springer-Verlag New York Berlin Heidelberg; 1985.
28. Chin-Shyurng Fachn. An adaptive reduction procedure for the piecewise linear approximation of digitized curves. *IEEE Trans. Pattern Anal. Machine Intell.* Sept. 1989; 11(no. 9): p.967.
29. Chow, Y., Grenander, U., Keenan, D. M. Hands: A Pattern Theoretical Study of Biological Shape. : preprint, Brown University, Division of Applied Mathematics. Authors' personal copy.
30. Chu, T. C., Ranson, M. A., Sutton, M. A. Applications of digital-image-correlation technique to experimental mechanics. *Exp. Mech.* 1985; 25(3): 232-244.
31. Collins, D. L., Peters, T. M., Dai, W., Evans, A. C. Model based segmentation of individual brain structures from MRI data. Authors' personal copy.
32. Cupradze, V. D. Three-dimensional problems of the mathematical theory of elasticity and thermoelasticity. Netherlands: North Holland; 1978
33. Davies, A. J. The finite element method. USA: Oxford University Press; 1980
34. Desai, C. S. Elementary finite element method. USA: Prentice-Hall; 1979
35. Dewdney, A. K. Leaping into Lyapunov Space. *Scientific American*. September 1991.
36. Donnelly, R. A. Geometry optimization by simulated annealing. *Chemical Physics Letters*. May 1987; v.136(no. 3, 4): 274.
37. Duncan, J. H., Chou, T. (University of Maryland). Computer Vision Laboratory. Center for Automation Research. Temporal edges: The detection of motion and the computation of optical flow . USA: University of Maryland; 1988.
38. Finite Elements in biomechanics; February 18-20, 1980; Tuscon, Arisona.

39. Foulds, L. R. Optimization techniques. USA: Springer-Verlag New York Berlin Heidelberg; 1981
40. Fraeijs de Veubeke, B. M. A course in elasticity. USA: Springer-Verlag Berlin Heidelberg; 1979
41. Friedman, D. H. Detection of signals by template matching. USA: The Hopkins Press Baltimore; 1968
42. Fung, Y. C. Biomechanics. Its foundations and objectives. USA: Prentice-Hall, Inc.; 1972
43. Fung, Y. C. Biomechanics. Mechanical properties of living tissues. USA: Springer-Verlag New York Berlin Heidelberg; 1981
44. Gallagher, R. H. Finite elements in biomechanics.
45. Gallagher, R. H. Finite elements in biomechanics. USA: John Wiley & Sons Ltd.; 1982
46. Gao, Y. Inverse variational principle in finite elasticity. Mechanics Research Communication. 1988; 15(3): 161-166.
47. Geman, S., Geman, D. Stochastic Relaxation, Gibbs Distributions, and the Bayesian restoration of images. IEEE Trans. Pattern Anal. Machine Intell.. Nov. 1984; vol. PAMI-6(no. 6): p.721.
48. Geman, S., Hwang, C. Diffusions for global optimization. SIAM J. Control and Optimization. 1986; 24(5): 1031-1043.
49. Gonzalez, R. C., Wintz, P. Digital image processing. USA: Addison-Wesley Publishing Company; 1977
50. Goodier, J. N., Hodge, P. G. Elasticity and plasticity. USA: John Wiley & Sons; 1958
51. Gordon, R. Geometric unwarping for detection of early breast cancer. 22nd National Conference on Breast Cancer, Boston, May 12-16. ; 1986.
52. Gordon, R. Monte Carlo methods for cooperative ISING models. Carreman, G. Cooperative Phenomena in Biology. USA: Pergamon Press Inc.; 1980; c1980: Gordon private.
53. Gordon, R. Steady-state Properties of Ising membranes. J. Chemical. Physics. 15 July 1968; 49(2).
54. Gordon, R. Toward robotic "x-ray vision": new directions for computed tomography. Applied Optics. 1985b; 24(23): 4124-4133.
55. Gould, P. L. Mathematical modeling of human aortic valve leaflets. Appl. Math. Modeling. 1976; 1: 33-36.
56. Herman, G. T., editor. Image reconstruction from projections. Implementation and applications. USA: Springer-Verlag Berlin-Heidelberg; 1979. (Topics in Applied Physics; v. 32).
57. Hill, T. L. An introduction to statistical thermodynamics. USA: Addison-Wesley Publ. Co., Inc.; 1960
58. Hill, T. L. Statistical mechanics. USA: McGraw-Hill Book Company; 1956
59. Hinton, E., Owen, D. R. J. An introduction to finite element computations. Great Britain: Redwood Burn; 1979

60. Hou, H. S., Andrews, H. C. Cubic spline for image interpretation and digital fitting. *IEEE Trans. Acoust. Speech Signal Process.* 1978; ASSP-26: 508-516.
61. Istfan, B. Inverse nonlinear finite element problems with local and global constrains. *IEEE Trans. Magnet.* 1978; 24(6): 2568-2572.
62. Jacobson, J. G., Gordon, R. Changes in the shape of the developing vertebrate nervous system analyzed experimentally, mathematically and computer simulation. *J. Exp. Zoology.* August, 1976; 197(2).
63. Jeffrey, W., Rosner, R. Optimization algorithms: simulated annealing and neural network processing. *The Astrophysical J.* Nov. 1986; 310: p.473.
64. Kirkpatrick, S., Gelatt, C. D. J., Vecchi, M. P. Optimization by simulated annealing. *Science.* May 1983; vol. 220(no. 4598): p.671.
65. Kupradze, V. D. Three-dimensional problems of the mathematical theory of elasticity and thermoelasticity. Netherlands: North Holland Publishing Company; 1979
66. Lancaster, P., Salkauskas, K. Curve and Surface Fitting. An Introduction. London, Great Britain: Academic Press; 1986
67. Lanir, Y. On the structural origin of the quasilinear viscoelastic behavior of tissues. Schmid-Schonnenbein, G. W. *Frontiers in biomechanics.* USA: Springer-Verlag New York Berlin Heidelberg; 1985.
68. Larrabee, W. F. A finite model of skin deformation. 1. Biomechanics of skin and soft tissue: a review. *Laryngoscope.* 1986; 96(April): 399-405.
69. Larrabee, W. F. A finite model of skin deformation. 2. An experimental model of skin deformation. *Laryngoscope.* 1986; 96(April): 406-412.
70. Larrabee, W. F. A finite model of skin deformation. 3. The finite element model. *Laryngoscope.* 1986; 96(April): 413-419.
71. Larrabee, W. F. A finite element model for the design of local skin flaps. *Otolaryngologic Clinics of N. Am.* 1986; 19(4): 807-824.
72. Lavrentev, M. M. Ill posed problems of mathematical physics and analysis. USA: American Mathematical Society; 1986
73. Little, R. W. /. *Elasticity.* : Prentice-Hall, Inc.; 1973
74. Lovasz, L., Plummer, M. D. Matching theory. North Holland : North Holland ; USE
75. Love, A. E. A Treatise on the mathematical theory of elasticity. Canada: General Publishing Company; Republication of the fourth edition from 1927
76. Maniatty, A. Finite element analysis of some inverse elasticity problems. *J. Engin. Mechanics.* 1989; 115(5): 1303-1317.
77. Martin, J. E. Breast imaging techniques. *Radiologic Clinics of North America.* March 1983; 21(1).
78. Mazur, A., Gordon, R. Digital Differential Radiography . in preparation.
79. Mazur, A., Gordon, R. Image Correlation Technique for recovering deformation fields from images. In preparation.

80. Mazur, E. Numerical simulations in low dose tomography. Manitoba, Canada: University of Manitoba; 1992; c1992.
81. Metropolis, N. Equation of state calculations by fast computing machines. The J. Chem. Physics. June 1953; vol. 21(no. 6): p.1087.
82. Michael, T. Y. Automatic recognition of characters by Fourier descriptors and boundary line encoding. Pattern Recog.. 1981; v.14(Nos. 1-6): 383.
83. Mikhailov, A. S. Graph identification by simulated annealing. J. Phys. A: Math. Gen. 1988; v.21: L487.
84. Moroz, Z. Mathematical models of inelastic material behavior. Canada: University of Waterloo; 1973
85. Moshfeghi, M. Elastic matching of multimodality medical images. CVGIP: Graphical Models and Image Processing. 1991; 53(3): 271-282.
86. Norrie, D. H., de Vries, G. An introduction to finite element analysis. USA: Academic Press; 1978
87. Oden, J. T. Computational mechanics. International conference on computational methods in nonlinear mechanics, Austin, Texas, 1974. Germany: Springer-Verlag Berlin Heidelberg; 1975
88. Osterby, O., Zlatev, Z. Direct methods for sparse matrices. Germany: Springer-Verlag Berlin Heidelberg; 1983
89. Otten, R. H. J., van Ginneken, L. P. P. The annealing algorithm. USA: Kluwer Academic Publisher
90. Peters, W. H., Ranson, W. F. Digital imaging in experimental stress analysis. Optical Eng.. May/June 1982; v.21(no. 3): p.427.
91. Pinto, J. G. The phenomenology of the heart muscle. Schmid-Schonlein, G. W. Frontiers in biomechanics. USA: Springer-Verlag New York Berlin Heidelberg; 1985.
92. Plevako, V. P. The two-dimensional problem of elasticity theory for inhomogeneous media in polar coordinates. PMM U.S.S.R. Appl. Math. Mech.. 1985; 49(5): 595-601.
93. Pratt, W. K. Digital Image Processing. New York, NY: John Wiley & Sons; 1978
94. Prescott, J. Applied Elasticity. USA; 1946
95. Ray, G., Ghista, D. N. A direct computational algorithm to estimate in vivo constitutive properties of some cardiovascular structures. Proc. Int. Conf. on Finite Elements in Biomechanics. ; 1980; 2: 623-637.
96. Ray, G., Ghista, D. N., Sandler, H. In vivo constitutive properties of the passive left ventricular myocardium. ASME paper #79-Bio-6. ; 1979a.
97. Ray, G., Ghista, D. N., Sandler, H. Mechanoventriculography: theory, evaluation and significance of the regional distributions of myocardial constitutive properties and blood pressure in left ventricular chamber. Cardiovascular Engineering. Baden-Baden, W. Germany: Verlag Gerhard Witzstrock; 1979b.
98. Ressel, S. S., Sutton, M. A. Strain-field analysis acquired through correlation of X-ray radiographs of fiber-reinforced composite laminate. Exp. Mech.. 1989; 29(2): 237-240.

99. Rosenfeld, A., Kak, A. C. Digital picture processing - second edition. : Academic Press, Inc.; 1982
100. Sabatier, P. C. Applied inverse problems. Germany: Springer-Verlag Berlin-Heilderberg; 1978
101. Sabochick, M. J. New global energy-minimization method. Physics Rev. B. June 1988; v.37(no. 18): 10846.
102. Sayre, D., Howells, M., Kirz, J., Rarback, H. X-Ray microscopy II. Germany: Springer-Verlag Berlin Heidelberg; 1988
103. Schoenberg, I. J. Cardinal Spline Interpolation. Philadelphia, PA: Society for Industrial and Applied Mathematics Press; 1973
104. Schumaker, L. L. Spline Functions: Basic Theory. New York, NY: John Wiley & Sons; 1981
105. Seguin, F. H., Burstein, P., Bjorkholm, P. J. X-ray computed tomography with 50- $\mu$ m resolution. Applied Optics. 1985; 24(23): 4117-4123.
106. Seiter, C. Science and the Mac. MacWorld. August, 1991.
107. Shoemaker, P. A. A constitutive model for two-dimensional soft tissues and its application to experimental data. J. Biomechanics. 1986; 19(9): 695-702.
108. Spath, H. Spline algorithms for curves and surface. Winnipeg, Canada: Utilitas Mathematica Publishing Incorporated; 1974
109. Stosker, J. J. Topics in non-linear elasticity. USA: Naw York University; 1964
110. Stucki, P. Image processing for document reproduction. Stucki, P. Advances in Digital Image Processing: Theory, Applications, Implementation . New York: Plenum Press; 1979: 177-218.
111. Sutton, M. A. Determination of displacements using an improved digital correlation method. Image Vision Comput.. Aug. 1983; v.1(no. 3): 133.
112. Sutton, M. A. Effect of subpixel image restoration on digital correlation error estimates. Optical Eng.. 1988; 27(10): 870-877.
113. Szu, H., Hartley, R. Fast simulated annealing. Physics Letters. June 1987; vol. 122(no. 3,4): p.157.
114. Tabor Laszlo, Dean Peter. Basic principles of mammographic diagnosis. Diagnostic Imaging in Clinical Medicine. 1985; 54.
115. Terzopoulos, D. The computation of visible-surface representations. IEEE Trans. Pattern Anal. Machine Intell.. 1988; 10(4): 417-439.
116. Terzopoulos, D. Multilevel computational processes for visual surface reconstruction. Comput. Vision Graph. Image Process.. 1983; 24: 52-96.
117. Torlin, V. N. Direct and inverse problems of the plane theory of elasticity of inhomogeneous body. Prikladnaja Mekhanika. 1976; 12(8): 49-52.
118. Truillo, D. M. Application of dynamic programming to the general inverse problem. Int. J. Num. Meth. Engin.. 1978; 12: 613-624.
119. Tzay Young, Y., King-Sun Fu. Handbook of pattern recognition and image processing. USA: Academic Press; 1986

120. Ural, O. Finite element method. USA: Intex Educational publishes; 1973
121. Van Ginneken. An inner loop criterion for simulated annealing. Physics Letters A. July 1988; v.130(no. 8, 9): 429.
122. Waldman, L. K. In-Vivo measurement of regional strains in myocardium. Schmid-Schonenbein, G. W. Frontiers in biomechanics. USA: Springer-Verlag New York Berlin Heidelberg; 1985.
123. Westlake, J. R. A handbook of numerical matrix inversion and solution of linear equations. USA: Robert E. Krieger Publishing Company; 1975
124. Williams, G. Computational linear algebra with models. USA: Allyn & Bacon; 1978
125. Witkin, A., Terzopoulos, D., Kass, M. Signal matching through scale space. Int. J. Comput. Vision. 1987: 133-144.
126. X-Ray Microscopy. Proceedings of the International Symposium, Gottingen, Fed. Rep of Germany, September, 14-16, 1983. Germany: Springer-Verlag Berlin Heidelberg; 1984
127. X-Ray microscopy, instrumentation and biological applications. Germany: Springer-Verlag Berlin Heidelberg; 1987
128. Yamaguchi Fujio. Curves and surfaces in computer aided geometric design. Germany: Springer-Verlag Berlin Heidelberg; 1988
129. Yamaguchi, F. Curves and Surfaces in Computer Aided Design. Berlin: Springer-Verlag Berlin-Heilderberg; 1988
130. Zane W. Bell. A bayesian Monte Carlo segmentation method for images dominant by Gaussian noise. IEEE Trans. Pattern Anal. Machine Intell.. Sept. 1989; vol. 11(no. 9): p.985.
131. Zhou, X. Digital subtraction mammography via geometric unwarping for detection of early breast cancer. Winnipeg: Department of Electrical & Computer Engineering, University of Manitoba; 1991.
132. Zhou, X., Gordon, R. 3D digital subtraction mammography. Canadian Med. and Biol. Eng. Conf. ; 1990; 16: in press.
133. Zhou, X., Gordon, R. Geometric unwarping for digital subtraction mammography. Proc. Vision Interface '88. Toronto: Canadian Image Processing and Pattern Recognition Society Press; 1988: 25-30.
134. Zhou, X., Gordon, R., Mazur, A. K., Mazur, E., Séguin, F. H., Bjorkholm, P. J. Theory of geometric unwarping with application to digital subtraction mammography. Image and Vision Computing: submitted.
135. Zhou, X., Gordon, R. Detection of early breast cancer. Review and future prospects. CCR Critical Reviews in Biomedical Engineering. 1989; 17(3): 203-255.
136. Zienkiewicz, O. C. The finite element method. Great Britain: McGraw-Hill Book Company; 1977