

2007

Point correspondence for automatic image registration using affine moment invariants

Ramesh Narayan
University of Dayton

Follow this and additional works at: https://ecommons.udayton.edu/graduate_theses

Recommended Citation

Narayan, Ramesh, "Point correspondence for automatic image registration using affine moment invariants" (2007). *Graduate Theses and Dissertations*. 4625.
https://ecommons.udayton.edu/graduate_theses/4625

This Thesis is brought to you for free and open access by the Theses and Dissertations at eCommons. It has been accepted for inclusion in Graduate Theses and Dissertations by an authorized administrator of eCommons. For more information, please contact mschlangen1@udayton.edu, ecommons@udayton.edu.

Point Correspondence for Automatic Image Registration using Affine Moment Invariants

Thesis

Submitted to
the School of Engineering
University of Dayton

In partial fulfillment of the requirements for
the degree
Master of Science in Electrical Engineering

by
Ramesh Narayan

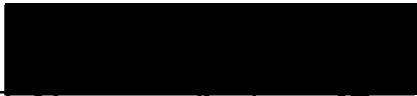
UNIVERSITY OF DAYTON

Dayton, Ohio

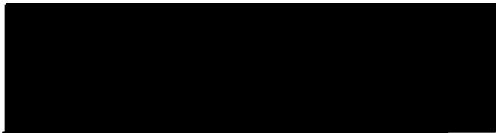
May 2007

Point Correspondence for Automatic Image Registration using Affine Moment Invariants

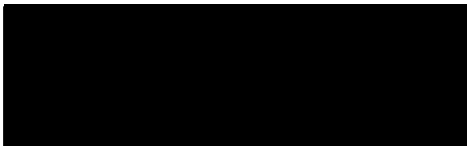
APPROVED BY:



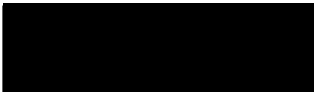
John S. Loomis, Ph.D.
Advisory Committee Chairman
Professor, Department of Electrical
and Computer Engineering



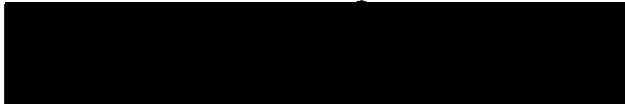
Raul Ordonez, Ph.D.
Committee Member
Associate Professor of Electrical
and Computer Engineering



Russel Hardie, Ph.D.
Committee Member
Professor, Department of Electrical
and Computer Engineering



Donald L. Moon, Ph.D.
Associate Dean
Graduate Engineering Programs &
Research
School of Engineering



Joseph E. Saliba, Ph.D., P.E.
Dean, School of Engineering

ABSTRACT

An approach towards determining point correspondence for the purpose of image registration from studying the local neighborhood of pixels around the detected control points has been presented in this thesis. A geometric technique for determination of the spatial mapping function parameters, which determines the angle of rotation from at least three corresponding points, has also been presented. The similarity measure is established by determining the standardized affine moment invariant of the overlapping circular segments of the region of interest, in stages, and eliminating an outlier if it falls below a pre-set threshold at any of these stages. As a final step in determining point correspondence, a similar procedure is followed with a range scaled rotation invariant texture description of the original region of interest using a symmetric convolution kernel, identifying that the moment invariant would lack the ability to identify where the variations in intensity function of an image are occurring. The aim of this thesis is to present a point correspondence technique which could be used for the purpose of automating the task of image registration with the presence of rotational transformation. The technique also takes into account the compensation of brightness and contrast differences between the two images and attempts to correct both linear and non-linear differences.

Keywords:

Image Registration; Similarity Measure; Affine Moment Invariant; Convex Hull;
Affine Transformation; Feature Extraction; Texture Description

TABLE OF CONTENTS

ABSTRACT.....	ii
LIST OF FIGURES.....	v
LIST OF TABLES.....	viii
LIST OF SYMBOLS.....	ix
Introduction	11
Chapter 1	15
Previous Research	15
Area Based Matching Techniques.....	15
Feature Extraction Based Techniques:	16
Chapter 2.....	18
Feature Extraction	18
Aperture Problem	19
Tensor-Based Technique.....	20
Corner Detection: Eigen-Values of a Structure Tensor.....	21
Noise Reduction Filter	22
Selection of Window Size	22
Selection of Gradient Filter	23
Results and Observations	23
Performance of the Corner Detector	26
Chapter 3.....	27
The Similarity Measure.....	27
Affine Moment Invariants: A Discussion	29
AMI Description of Regions of Interest	31
Similarity Measure: AMI of Gray-Scale Image and Rotation Invariant Texture.....	33
Similarity Measure: AMI and Range-Scaled Texture Description.....	36
Histogram Specification for Range Scaling.....	36
Exponential Affine Moment Invariant	40
Similarity Measure: Variations and Composition Based Similarity.....	47
Brightness and Contrast Compensation: Histogram Specification	57
Results	60
Performance Measurement.....	65
Linear Brightness and Contrast Adjustment	66
Results	72
Performance Measurement.....	78
Blur Variance of the Moment Invariant	78
Texture Description: Elimination Criteria	79
Results	81
Reliability Dependence	86
Uniqueness Measure	91
Results	92
Performance Measurement	93
Chapter 4.....	94
Consistency Checking	94
Affine Transformations and Convexity.....	95
Ordering the Point Set	99

Determining Applicable Equations.....	100
Consistency Checking Algorithm.....	112
Results and Performance Issues.....	112
Chapter 5.....	115
Performance Benchmark.....	115
Chapter 6.....	117
Conclusions.....	117
Chapter 7.....	119
Future Research.....	119
References.....	121
Appendix.....	123
A.1 Images used for testing the algorithm.....	123
A.2 Regions of interest extracted from the images in A.1	124
Data Set A.....	124
A.3 Regions of interest extracted from the images in A.1	126
Data Set B.....	126

LIST OF FIGURES

Figure 1: Demonstrating the aperture problem from the motion constraint equation	20
Figure 2: Demonstrates the aperture problem and occluded surfaces	20
Figure 3: Corner Detection with a 3x3 Gaussian pre-filter and	23
Figure 4: Corner Detection with a 3x3 Gaussian pre-filter and 3x3	25
Figure 5: Corner Detection with a 3x3 Gaussian filter and 3x3 Sobel edge.....	25
Figure 6: Cluster analysis based elimination to reduce the multiple responses at the corners of dayton1.tif.....	26
Figure 7: Cluster analysis based elimination to reduce the multiple responses at corners structures of dayton2.tif	26
Figure 8: Selected control points after uniqueness checking with correlation as the similarity measure threshold at 0.9	33
Figure 9: Selected control points after uniqueness checking with correlation as the similarity measure threshold at 0.9	34
Figure 10: Region of Interest 2 of image dayton1.tif.....	37
Figure 11: Histogram of the region roi2ofim1c.jpg	37
Figure 12: Histogram of the texture description of the image	37
Figure 13: Histogram specification quantizes the distribution of the intensity values.....	38
Figure 14: Region of Interest 7 of Image dayton1.tif.....	38
Figure 15: Histogram of the texture description of the image	38
Figure 16: Histogram specification quantizes the distribution of the intensity values.....	39
Figure 17: Histogram specification quantizes the distribution of the intensity values.....	39
Figure 18: Summation value at each pixel position of the moment invariant for the standard image	43
Figure 19: Standard Image for the blank circular disk.....	48
Figure 20: Standard Image after removing pixels from the center	48
Figure 21: Standard Image after removing pixels from the sides.....	48
Figure 22: Illustrates the difference in the summation at each pixel at the center with respect to the standard	49
Figure 23: Illustrates the difference in the summation at each pixel at the sides with respect to the standard	50
Figure 24: Plot showing the $\exp(-kxy^T)$ function.....	51
Figure 25: Region of Interest 21 of dayton1.tif	55
Figure 26: Region of Interest 10 of dayton2.tif	56
Figure 27: Region of Interest 2 of Dayton 3.....	57
Figure 28: Region of Interest 4 of Dayton 4.....	57
Figure 29: Histogram Specified Region of Interest 4 of Dayton 4	58
Figure 30: Region of Interest 5 of Dayton 3.....	59
Figure 31: Region of Interest 7 of Dayton 4.....	59
Figure 32: Histogram Matched Region of Interest 7 of Dayton 4	60
Figure 33: Histogram for Region of Interest 21 of image dayton1.tif	61

Figure 34: Histogram for Region of Interest 10 of image dayton2.tif	62
Figure 35: Histogram Specified image for Region of Interest 10 of image dayton2.tif	62
Figure 36: Result of Similarity Measure on the image dayton1.tif with dayton2.tif	63
Figure 37: Result of similarity measure on the image dayton2.tif with histogram specification	63
Figure 38: Result of Similarity Measure on the image dayton2.tif with dayton3.tif	64
Figure 39: Result of similarity measure on the image dayton3.tif with histogram specification	64
Figure 40: Result of similarity measure on the image dayton3.tif with dayton3.tif	65
Figure 41: Result of similarity measure on the image dayton4.tif with histogram specification	65
Figure 42: Region of Interest 2 of Dayton 3	70
Figure 43: Region of Interest 4 of Dayton 4	70
Figure 44: Linearly adjusted contrast and brightness	71
Figure 45: Linearly adjusted contrast and brightness	71
Figure 46: Result of Similarity Measure on the image dayton2.tif with dayton1.tif	72
Figure 47: Result of Similarity Measure on the image dayton1.tif with linear contrast and brightness correction	73
Figure 48: Result of Similarity Measure on the image dayton2.tif with dayton3.tif	74
Figure 49: Result of Similarity Measure on the image dayton3.tif with linear contrast and brightness	74
Figure 50: Result of Similarity Measure on the image dayton3.tif with dayton4.tif	75
Figure 51: Result of Similarity Measure on the image dayton4.tif compensated for linear contrast and brightness	75
Figure 52: Result of similarity measure for dayton1.tif	82
Figure 53: Result of similarity measure for dayton2.tif	82
Figure 54: Result of similarity measure for dayton2.tif	83
Figure 55: Result of similarity measure for dayton3.tif	83
Figure 56: Result of similarity measure for dayton3.tif	84
Figure 57: Result of similarity measure for dayton4.tif	84
Figure 58: Control points selected after running the similarity measure algorithm on the image dayton3.tif	85
Figure 59: Control points that are selected after the similarity measure algorithm on image dayton3.tif rotated by an angle of 20 degrees	85
Figure 60: Corner detection with uniqueness measure for dayton1.tif	92
Figure 61: Corner detection with uniqueness measure for dayton2.tif	93
Figure 62: Demonstrates the Graham's scan algorithm for convex hull on original set of points	98
Figure 63: Demonstrates the Graham's scan algorithm for convex hull on transformed set of points	99
Figure 64: Demonstrates the construction of convex hull layers of a point set	100
Figure 65: Vectorial technique demonstrating scaling before rotation and translation ..	102
Figure 66: Vectorial technique demonstrating scaling before translation and rotation ..	106
Figure 67: Vectorial technique demonstrating translation, scaling and rotation in that order	107
Figure 68: Vectorial technique demonstrating translation, rotation and scaling	109
Figure 69: Vectorial technique demonstrating rotation before translation and scaling ..	110
Figure 70: Vectorial technique demonstrating rotation before scaling and translation ..	111
Figure 71: Convex hull of the set of control points from dayton3.tif	112
Figure 72: Convex hull of the set of control points from dayton4.tif	113
Figure 73: Registered Images using the affine spatial mapping function in MATLAB	113
Figure 74: dayton1.tif	123

Figure 75: dayton2.tif	123
Figure 76: dayton3.tif	124
Figure 77: dayton4.tif	124

LIST OF TABLES

Table 1: Time of execution of the corner detection algorithm	26
Table 2: Similarity values obtained from moment invariant of original image and the texture descriptor	34
Table 3: Similarity measure between regions of interest from range scaled texture	39
Table 4: The similarity measure between dissimilar regions lying in the range of true matches	44
Table 5: The similarity measure between dissimilar regions lying in the range of true matches	45
Table 6: Normalized affine moment invariant for Figure 25, Figure 26 and Figure 27	48
Table 7: Similarity determined over the three iterations eliminates most false matches	53
Table 8: Similarity measure showing results for false matches of Set A	54
Table 9: Similarity measure result for region of interest 21 of dayton1.tif and region of interest 10 of dayton2.tif.....	56
Table 10: Performance comparison with conventional correlation based technique.....	115
Table 11: Performance comparison for uniqueness measure with conventional correlation based technique	115
Table 12: Time of execution of the three algorithms on MATLAB.....	116

LIST OF SYMBOLS

Symbol	Symbol Description
--------	--------------------

\vec{r}_1	Position vector of the initial points
\vec{r}_2	Position vector of the final points
x	x - component of the position vector
y	y - component of the position vector
t_x	x - component of the translation vector
t_y	y - component of the translation vector
\vec{r}_{s1x}	x - component of the vector representing the selected pixel position in the reference image
\vec{r}_{s2x}	x - component of the vector representing the selected pixel position at an intermediate scaled position as seen from the scaling frame of reference
s_x	scale factor along the x - direction about the scaling frame of reference as mentioned
\vec{r}_{s1y}	y - component of the vector representing the selected pixel position in the reference image
\vec{r}_{s2y}	y - component of the vector representing the selected pixel position at an intermediate scaled position as seen from the frame of the reference of scaling
s_y	scale factor along the y - direction about the scaling frame of reference as mentioned
v_{sx}	x - component of the position vector from the origin to the origin of the scaling frame of reference
v_{sy}	y - component of the position vector from the origin to the origin of the scaling frame of reference
\vec{v}_R	Vector to the origin of the rotation frame of reference
\vec{v}_S	Vector to the origin of the scaling frame of reference
S_p	Normalized sum of all intensity values in the region 1
S	Normalized sum of all intensity values in the region 2
S_{pMAX}	Maximum value of intensity in region 1
S_{pMIN}	Minimum value of intensity in region 1
S_{MAX}	Maximum value of intensity in region 2
S_{MIN}	Minimum value of intensity in region 2
a or c	Factor for contrast enhancement
b	Factor for brightness enhancement

M_{x1}	Moment invariant along only x -direction of the reference image region
M_{x2}	Moment invariant along only x -direction of the template image region
C	Moment summation with all pixel intensity replaced with 1's
x_c	x -component of the centroid
y_c	y -component of the centroid

Introduction

Image Registration is a fundamental task in image processing, which involves the analysis of one or more images, to extract information in the form of features, and determine an optimal geometric transformation between the corresponding image data. Image registration is used in variety of application areas like, computer vision, medical imaging, computer graphics and remote sensing. Due to extensive application in various important areas, there are a variety of techniques, adopted in theory, to solve the problem of image registration. Each technique varies with application area, with information about the nature of transformation and the accuracy of the desired result [1].

The problem arises due change in the position of the object that is being imaged or the change in position or perspective of the imaging device or change in illumination conditions, such that there is no direct spatial correspondence between the reference and template image.

Image Registration has an important application in the field of Computer Vision. There are several techniques in theory to register images, if the corresponding features are manually selected and supplied to the software. The common topic of research in this area is to automate this task, by extracting features from each pair of image and using an appropriate similarity measure to obtain point correspondence or by using a suitable correlation based image matching. The problem encountered is in the general nature of the deformations that an image undergoes. After the obtaining point correspondences, the transformation matrix and spatial mapping function type and parameters are estimated.

A set of well defined steps are followed in the methodology of registering the two images, in order to yield desired results,

1. **Pre-Processing:** This is preprocessing step where the noise reduction and other image restoration and enhancement filtering techniques are applied to the images.
2. **Feature Extraction:** This is the step, in the image registration methodology, where distinct and meaningful attributes are extracted from the image. This is commonly involves computing the gradient in each local neighborhood of the image. Corners, edges, lines, contours etc. are some common examples of these features.
3. **Similarity Measure:** This is a procedure that obtains feature correspondence, between two different set of features, by using some attribute from the neighborhood of each selected control point that remains invariant under spatial transformation and intensity function transformation. This is perhaps the most researched subject in the quest to automate the image registration methodology.
4. **Consistency Checking:** This step determines the spatial mapping function parameters and its type, from the set of point correspondences obtained as an output from establishing similarity measure.
5. **Image Transformation:** After the determination of the spatial mapping function parameters, the next step involves the application of the transformation matrix to the template image and interpolation techniques are used to determine the intensity at the destination pixel position.

Each of the steps mentioned above, has been followed carefully in this research to develop an appropriate technique to register two images, automatically. The similarity measure for this thesis would consider the affine moment invariant of the two regions of

interest. Affine moments are used as shape descriptors in object recognition. In this thesis, a signal processing approach, which identifies the variations in the two dimensional intensity function, is intended to be taken to determine the similarity between regions of interest. The thesis would aim to create a similarity measure which determines the similarity or the dissimilarity between two regions on the basis of the variations detected in the intensity function and their position relative to a given origin. The variations in intensity function and its position are the only means by which all the methods determine similarity or dissimilarity between two regions. Since, moments invariants combine these two pieces of information, it seems to be an appropriate choice for developing a similarity measure. In order to distinguish between regions, on the basis of the position of the variation in intensity function, texture description of the regions of interest could provide a suitable solution. Most of the techniques that use moment invariant to determine similarities between two regions use the shape description capability of the moment invariant to determine the solution. The thesis follows a feature extraction based approach and therefore, the AMI would be evaluated over a region of interest around the detected control point. Without the need for contrast and brightness correction, the technique is quite robust and leads to elimination of most outliers.

Consistency checking with spatial deformation model assumed to be an affine model, would determine the six parameters of the affine transformation matrix using three points at a time. If more than three points are present in the two sets, triangle meshes are generated and piece-wise affine transform is performed from triangle to triangle. In this thesis, however, the convex hull would be used to order the points and a simple vectorial technique would be used to determine the angle of rotation.

The next section summarizes, in brief, the research and ideas already integral to the similarity measure algorithms in image registration theory. Chapter 2 describes the feature extraction technique for corner detection using the structure tensor. Chapter 3

describes the similarity measure with the use of affine moment invariants and rotation invariant texture analysis. Chapter 4 describes the consistency checking algorithm developed which checks the consistency of the similarity measure result with that of the assumed deformation model (affine transformation).

Chapter 1

Previous Research

Considerable amount of work has already been done to determine correspondence between a pair of image and the spatial transformation parameters. The most common techniques are based on the cross-correlation and feature extraction followed by the feature correspondence using a similarity measure.

Area Based Matching Techniques

This is a region based matching technique and is based upon determining correspondences between the two images by determining the cross-correlation between regions of the image. The correlation function assumes a zero for totally dissimilar patterns and reaches maximum of one for similar features [1]. The technique is to determine similar regions in two images by considering one image and a sub-image and summation is taken over the region where the two images overlap.

$$c(x, y) = \sum_x \sum_t f(s, t) w(x + s, y + t) \quad (1.1)$$

For, $x = 0, 1, 2, \dots, M - 1$, $y = 0, 1, 2, \dots, N - 1$, f and w are the two images [2].

The sub-image w is moved around the image area of f , giving the value of the function $c(x, y)$. The overlap where the value of the function $c(x, y)$ yields the maximum is the point of best match and is used to establish the correspondence.

The correlation function is sensitive to the amplitude of f and w . The correlation result is sensitive to large DC term in the image. The correlation coefficient is used to overcome this difficulty [3],

$$\gamma(x, y) = \frac{\sum_s \sum_t [f(s, t) - \bar{f}(s, t)] [w(s + x, y + t) - \bar{w}]}{\sum_s \sum_t [f(s, t) - \bar{f}(s, t)]^2 \sum_s \sum_t [w(s + x, y + t) - \bar{w}]^2} \quad (1.2)$$

Another technique used, to yield better results, is to correlate in the frequency domain, using the phase component of the image only. This technique of determining the match is referred to as the binary phase-only matched filtering. The phase of the image consists of essential information about the distribution of the spatial structures in an image. Correlating the phase of the two images is not affected by the image energy, and yields the best results. Many other techniques exist in theory which uses extension of correlation to establish the similarity measure. These techniques include gradient descent search, snakes, wavelets and other multi-resolution schemes [14]. Each of these techniques

Feature Extraction Based Techniques:

This technique extracts certain features like lines, points, blobs etc, from the image using a suitable interest operator and determines correspondences between these features using image matching techniques. A set of matches between the two regions is established using a suitable similarity measure. The similarity measure establishes correspondence between a set of control points in the two images based upon intensity values lying in the region of interest around the selected points. When the correspondences are established a suitable consistency measure is derived to determine the spatial mapping function parameters and the type of interpolation algorithm to be

used to register the images. The technique depends highly upon the efficiency and accuracy of the feature extraction techniques and the robustness and reliability of the similarity measure.

A wide variety of techniques uses moment invariants for the purpose of registration. The journal paper [9] summarizes these techniques. Affine moment invariants derived by Flusser and Suk and were used to register SPOT and LANSAT images. Holm extracted closed boundary region and used moment invariants as a feature to represent them for registering, moments invariants along with chain codes have used for the purpose of registration, Flusser and Suk derived moment invariants that are invariant to blur and used it for registering SPOT and AVHRR satellite images, digital subtraction angiography images were registered by Bentoutou using these invariants and they further developed this technique to combine rotation invariance for registering 3-D images.

Chapter 2

Feature Extraction

The task of extracting interesting attributes from an image using the information provided by the structures in the image, as determined by the variations in intensity values in every local neighborhood of the image. These features could be in form of lines (gradient lines), points (corners or center of circular symmetric features), segments, blobs or could be numerical quantities derived from studying a specific property like the probability of the occurrence of a specific intensity value in the neighborhood of a pixel, which describes the differences between visually distinct regions and helps in classification of each region based upon these quantities. Feature extraction techniques are commonly used in object recognition tasks and are also extremely useful for image matching where these features could provide a set of corresponding regions as a basis for determining the match.

The features can be determined using several techniques, which involve the computation of gradient and then determining the positions by using techniques that are invariant with respect to spatial transformations. Feature extraction is a technique that uses information provided by the intensity function in a neighborhood (window) and therefore, the result varies with noise and gray level distortions. The accuracy of the technique is enhanced with the use of optimized linear filters and non-linear filters to diminish the effect of noise.

Aperture Problem

The aperture problem can be defined as the sampling of two dimensional motion of an object from local aperture like the receptive field [3]. The image motions of edge constraints its motion in the world to a line in the velocity space but does not narrow it down to a single velocity [3]. Therefore, the measurements of velocity obtained through observations from a local aperture are ambiguous

In determining the motion estimation, the assumption is that the brightness at a point can change with time, only due to motion. This results in the motion constraint equation or brightness change constraint equation, BCCE

$$(\nabla g)^T f + g_t = 0 \quad (2.1)$$

Where $f = [f_1, f_2]^T = \left[\frac{dx}{dt}, \frac{dy}{dt} \right]^T$ is the optical flow; ∇g defines the spatial gradient

and $\frac{\delta g}{\delta t}$ is g_t . The relation defines the aperture problem of motion estimation as shown

in the figure below.

From the equation above, only the flow perpendicular to constraint line can be estimated (normal flow as it points normal to the lines of constant image brightness, parallel to spatial gradient). In order to overcome the aperture problem, the size of the region of interest has to be enlarged, as the chances of distributed spatial structure in the image increases. But this may result in region extending over the motion boundaries. This is referred to as the generalized aperture problem [2].

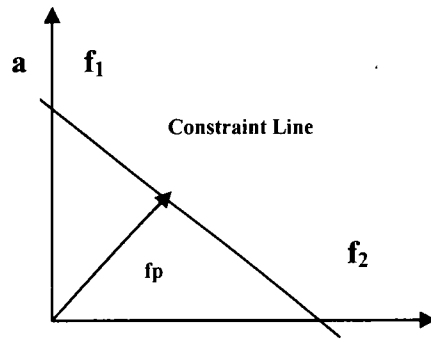


Figure 1: Demonstrating the aperture problem from the motion constraint equation

As can be seen from the Figure below, only the component of velocity that is normal to the edge line can be approximated unambiguously.

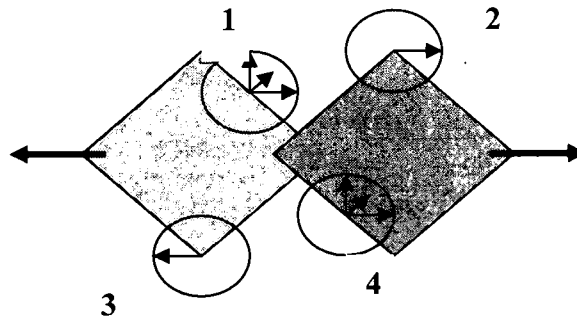


Figure 2: Demonstrates the aperture problem and occluded surfaces

Such features usually represent corner measures obtained at the intersection of more than one gradient line in the image. But, corners are also often resulting of occlusions.

Tensor-Based Technique

The corner measure is determined by considering points in the image, where, the intensity variations is two dimensional and ignoring the temporal variations. Hence, the determination of corner measure using optical flow technique is only followed for the purpose for providing stability to the interest operator.

The variations in gray level values are determined by the applying the gradient operator to the entire image.

The orientation of the gray level values in a simple neighborhood is determined as that value that shows least deviation from the directions of the gradient. The scalar product between the gradient vector and the orientation vector \vec{r} in a neighborhood window and the assumption of the two being perpendicular can be applied and the solution can be obtained by maximization of the least squares expression,

$$\int W(x-x', y-y') (\Delta g(x', y')^T \vec{r})^2 dx' dy' \quad (2.3)$$

Where, $W(x-x', y-y')$ is a Gaussian convolution kernel,

On maximization, the problem reduces to determining the eigenvalues of a structure tensor matrix obtained from the outer product of the 2-D gradient vector on solving

Equation 2.3

$$\Delta g(x', y') \Delta g(x', y')^T = \begin{bmatrix} g_x g_x & g_x g_y \\ g_x g_y & g_y g_y \end{bmatrix} \quad (2.4)$$

where the subscripts denote the direction in which the gradient is approximated.

Corner Detection: Eigen-Values of a Structure Tensor

On eigenvalue analysis of the structure tensor matrix, classification of the nature of intensity variations in local neighborhood of each pixel is done. If both eigenvalues are zero, this equates to what is known as the 'blank wall' problem. There are no spatial structures in the local neighborhood of the pixel and all the intensity values are approximately constant. If one eigenvalue is non-zero and the other zero, the intensity variations are in one direction, representing edges and as mentioned before aperture problem is present in this window of pixel values. If both eigenvalues are non-zero, the

intensity variations are two-dimensional and this represents the pixels that are free of aperture problem and represents corner measure. All such pixels are detected and classified as interest points for the purpose of determining corresponding control points for registration.

Since, the images considered in computer vision applications are known to be highly textured, the number of points detected in each image is usually very high, which needs to be reduced for robustness.

A possible solution to the problem mentioned above, is to select only those detected points, where the trace of the auto-correlation matrix is above the noise variance by a given factor, thus representing a corner that has resulted due to the curvature in gradient [5].

Noise Reduction Filter

A general linear convolution filter, used for reduction of noise in the image, has a property of suppressing the noise as well as useful information present in the image, to an equivalent level [2]. A filter used for only noise suppression, is non-linear in nature. In this thesis, an optimized linear Gaussian filter has been used for this purpose.

Selection of Window Size

The selection of the convolution window size would also significantly affect the results. In fact, [5] has illustration for the technique of the selection of an optimal window for interest operator based upon the desired features to be extracted and is a recommended reading. The size of the window chosen depends upon the content of the image. For most images, a window size of 3×3 or 5×5 would give reliable results.

Selection of Gradient Filter

The selection of the gradient filter also determines the accuracy of the result. As per [3], an optimized Sobel filter applied in conjunction with an optimized 5×5 Gaussian pre-filter in both vertical and horizontal directions would yield a good result. The result was however obtained by using the filter obtained from MATLAB image processing toolbox, which provided the matrix closest to the two-dimensional application of the optimized filter mentioned in [3]. The following result was obtained using a Sobel filter for edge detection with threshold at 0.17, along with a 5×5 or 3×3 Gaussian filter of width 0.75.



Figure 3: Corner Detection with a 3×3 Gaussian pre-filter and 3×3 Sobel edge on dayton1.tif

Results and Observations

It is observed that there aren't many common points detected between the two images and therefore, the accuracy of the similarity measure is desired. The points lying on the tree are not very stable since, they could change due to differences in physical condition of the scene (like wind) when they are photographed. Also observed, are the multiple responses of the corner detector around any given corner. The high number of corners detected with low number of correct pairs would unnecessarily increase the computational complexity of the algorithm.

One possible solution of this problem is to recognize the fact, that these points form spatial clusters, which could be separated based upon Euclidean distances of each point from the other. The average value of points lying within the same cluster is determined and a single point out of each cluster is selected, that lies closest to the average value and it is assumed that this point represents the true detected corner. Unfortunately, the first problem has no solution, since the detection of such a point would cause ambiguity, they must be treated as outliers. The result from the procedure just mentioned has been shown in **Figure 4**, **Figure 5**, **Figure 6** and. The number of control points that were selected in **Figure 4**, is 50 where as the number of points that were selected in **Figure 5**, is 8 and that in **Figure 6** is 250, whereas that in **Figure 7** is 53.

As the size of the Gaussian window increases, the response also becomes more prominent and the corners are detected over a larger area. It is a standard practice (mentioned in [5]) to choose the sizes of Gaussian kernel window and the gradient filter to be the same to ensure good results.

The elimination of the detected points that form a circular region around the true corner has been done by cluster analysis as mentioned above and yields the result shown in **Figure 5**.

After the detection of the interest points, the next step in determination of a corresponding set of control points from the both the images, is the selection of an appropriate similarity measure or more appropriately for our case, a dissimilarity measure.



Figure 4: Corner Detection with a 3x3 Gaussian pre-filter and 3x3 Sobel edge on dayton1.tif

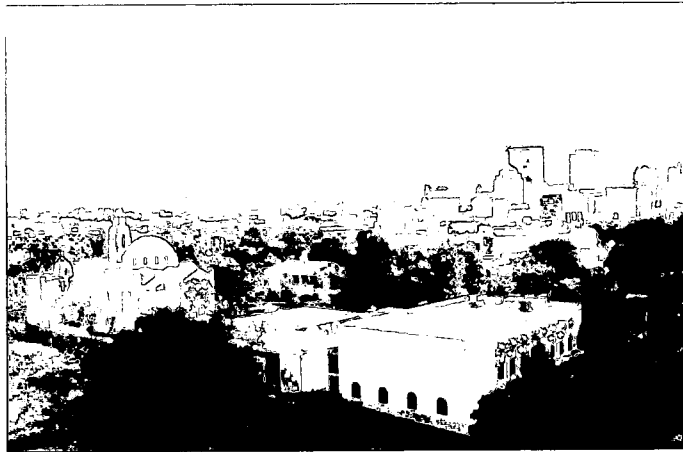


Figure 5: Corner Detection with a 3x3 Gaussian filter and 3x3 Sobel edge Detectors on dayton2.tif

The most common approach in the field of signal processing, is to consider a neighborhood around each detected point and use the intensity function in that neighborhood to establish the similarity between similar regions.



Figure 6: Cluster analysis based elimination to reduce the multiple responses at the corners of dayton1.tif



Figure 7: Cluster analysis based elimination to reduce the multiple responses at corners structures of dayton2.tif

The performance of the corner detection for the four images is almost equivalent. The computational complexity of the algorithm would increase with the size of the images.

Performance of the Corner Detector

Table 1: Time of execution of the corner detection algorithm

Image	Time of Execution (in seconds)
dayton1.tif	12.137070
dayton2.tif	11.473220
dayton3.tif	12.204665
dayton4.tif	11.343478

The table shown above summarizes the time of execution of the corner detection algorithm.

Chapter 3

The Similarity Measure

The most common approach for establishing similarity is to derive the correlation coefficient using the **Equation 1.1** mentioned in **Chapter 1**. The result obtained using this technique is a value that quantifies visual similarity between any two regions. Since, the measure depends entirely upon the values of intensity at each pixel in the image, it is quite sensitive to changes in intensity function. Another commonly used approach towards determining the similarity is the use of moment invariant of the region. Several different algorithms are present in the literature of image registration, which applies moment invariants of different kinds to determine the similarity. A comprehensive study of all these techniques have been summarized in the article [9].

In this thesis, I have taken a fresh approach towards establishing similarity between regions using affine moment invariants. Since, the texture of the region plays an important role in determining the results obtained from feature extraction, it would be meaningful to analyze the result of using rotation invariant texture description in conjunction with the moment invariant shape description, to establish the similarity between only true matches and develop the ability for the software to distinguish between dissimilar regions. Since, all the techniques that establish correspondences are based on some invariance property of the intensity function, suffer from sensitivity to modifications to the function, the two kinds of classification, invariant under rotation, for

a region and its corresponding match in the template image, enhances the probability of successful detection of outliers from among a given set of corresponding points.

Firstly, the application of the following filter kernel to the reference image, to determine a shape description that would remain invariant under rotation and doesn't depend upon the contrast and brightness conditions of the image. Since, the normalized differences eliminate both the multiplicative and additive terms that are present, if one assumes the change to be linear in nature. Secondly, the statistical measure, that moment invariant is, would remain ambiguous, since many regions of interest that have different structural property, would have the similar overall summation of intensity function over the image region and hence, in order to establish dissimilarity between these regions, the description in terms of spatial structure would be a prime distinguishing factor.

$$h = \begin{bmatrix} -\frac{1}{8} & -\frac{1}{8} & -\frac{1}{8} \\ -\frac{1}{8} & 1 & -\frac{1}{8} \\ -\frac{1}{8} & -\frac{1}{8} & -\frac{1}{8} \end{bmatrix}$$

When a region is convolved with the filter kernel h , the following is the result at the intensity value of each center pixel.

$$f(x, y) = g(x, y) - \frac{1}{8}(g(x+1, y) + g(x+1, y+1) + g(x+1, y-1) + g(x, y+1) + g(x, y-1) + g(x-1, y) + g(x-1, y+1) + g(x-1, y-1))$$

When a linear distortion in the intensity function of the region is assumed,

$$g(x, y) = a g(x, y) + b \quad (3.1)$$

Where, b is the brightness term and a is the contrast term. The application of the filter h at each pixel clearly eliminates the term b , modifying $f(x, y)$ by multiplicative factor of a .

$$f(x, y) = a f(x, y) \quad (3.2)$$

On normalization of $f(x, y)$, with its maximum value, the multiplicative factor a gets eliminated and for any two regions that differ in intensity function, only by a linear distortion, as given in **Equation 3.1**, this image would remain invariant, with respect to the original. This image would also be invariant to rotation, due to the symmetric nature of the convolution kernel used to obtain the image. A related texture description, known as the RISAR (Rotation Invariant Simultaneous Auto-Regressive model) [2] uses the same principle, but takes it one step further, and uses intensity values occurring on successive circles of increasing radii around the source center pixel to determine the value of the destination center pixel [2]. Scale invariance is not provided and to be able to incorporate scale invariance, scaling factor needs to be determined a priori. After determining the scaling parameter, the region of interest in the template image can be scaled to match the scale of the reference region and then filtered.

Affine Moment Invariants: A Discussion

Moment invariants are commonly used for shape description in digital image processing. The moments are functions that combine the product of intensity of each pixel in an image, with its spatial position along both dimensions, raised to varying powers. In a binary image, where all intensity values equal one, this is just the summation of the product of the powers of spatial positions, where the intensity function equals one. The summation result is an integer, which provides description of the shapes in the image. Affine moment invariants are a set of moment invariants that remain invariant under affine transformations.

Affine moment invariant was first derived by Flusser and Suk [8], for the purpose of object recognition. In a recent paper written by Jan Flusser [8], a systematic derivation of affine moment invariants of different orders has been published and provides the ability for research on this topic and develops variations that could help solve some challenging problems in the field of object recognition and image matching. A similar attempt has been made in this thesis to derive an affine moment invariant, which would assist in identifying similar regions in two images that differ by a spatial transformation.

The affine invariant moment derivation as described in [8], has been illustrated here. The cross product of two points and the cross product of their affine transformed counter-parts differ from each other, by a factor of a scalar, the determinant of the Jacobian matrix of the transformation matrix. The affine invariant moments are derived by eliminating this factor through proper normalization.

The following are the commonly derived affine moment invariants in theory, but knowing the technique of deriving it, moment invariants of any order or any desired behavior can be obtained,

$$I_1 = \frac{2(u_{02}u_{20} - u_{11}^2)}{u_{00}^4} \quad (3.3)$$

$$I_2 = \frac{u_{20}^2 u_{04} - 4u_{20}u_{11}u_{13} + 2u_{20}u_{02}u_{22} + 4u_{11}^2 u_{22} - 4u_{11}u_{02}u_{31} + u_{02}^2 u_{40}}{u_{00}^7} \quad (3.4)$$

In the theory of computer vision, the Gaussian filters have a several desirable characteristics like linearity and shift invariance. One of its properties is the circular shape of its convolution kernel, the effect of the application of such a filter on an image remains invariant to rotational transformation. This property makes it desirable for many applications in feature extraction techniques, used in this field. For instance, feature extraction using structure tensor technique, as mentioned in **Chapter 2** of this thesis, uses Gaussian smoothing filter of a selected width, to reduce the effect of noise. Scale space

techniques for feature extraction and localization, uses differential invariants, which are derived from Gaussian convolution kernel.

Hence, the idea behind using a circularly symmetric convolution kernel is to reduce the effect of rotational transformation, as well as, for determining results that contain information that is spread equally in all directions from the center pixel. Using a similar idea, a technique of establishing similarity between two regions in the reference and the template images, has been presented in this thesis, by using the affine moment invariant summation of the regions.

In order to determine similarity between any two regions, firstly, one needs to establish a certain standard against which the values of the similarity measure obtained for each region, is compared against. This is extremely important, since, the values obtained from any statistical measure has no global meaning, unless, the context in which it has been evaluated, is well defined. Hence, in order to use the moment invariant as a similarity measure, the value obtained from calculating the moment must be assigned a meaningful standard, relative to which all the regions could be compared. This would also provide reliability to the measure. In order to determine a relation that exists between any two arbitrary regions, one needs the complete information of the intensity function at each spatial position in the image. Contrast and brightness differences also play an important role. Next section discusses the observations that were made in using the affine moment invariant as a similarity measure and using the texture description as a way of adding vital information to enable rejection of dissimilar regions, observed from the values obtained from the moment evaluation.

AMI Description of Regions of Interest

As was discussed in previous sections, working with a circular window, in case of feature extraction and registration, yields results that are invariant with respect to rotational transformation. Hence, for this thesis, I used a circular window, to select the

intensity function around a detected control point. **Appendix A** contains the four test images, used in order to analyze the results obtained from developing the similarity measure. As is quite clear, just by looking at the images, there are differences in noise, brightness, contrast and blur relative to each other, which would distort the results in varied but predictable ways.

The first step that is followed after the determination of control points from corner detection is the rejection of those points in the same image that are very similar to each other and would cause difficulty in selecting the true matches that they correspond to. It is intuitive to think of these points as those, which lie on visually similar and repeating structures in the image. For example, a number of corners that were detected at the windows of the building have been shown in **Figure 8**. Therefore, it is logical to use correlation to detect these visually similar regions in the image, and remove them from further analysis. But, since, not all similarity measures detect similarity, with the same result as correlation, it would be better to use the similarity measure itself for checking the uniqueness of the points in an image. For the purpose of testing, however, correlation was used on the images **dayton1.tif** and **dayton2.tif**, to remove all the detected points that are similar above a certain predefined threshold. On analyzing the images, shown in **Figure 8** and **Figure 9**, it is evident that this would not eliminate all outliers. The figures show the control points that are left after rejecting all the points, the similarity of whose neighborhood, in terms of the correlation coefficient value, is greater than or equal to 0.9. The result obtained varies with the threshold. The subsequent figures (**Figure 10** and **Figure 11**) show the result obtained at a threshold of 0.95.

Appendix A contains the circular regions of interest cropped out of the image **dayton1.tif** and **dayton2.tif** on these selected points after determining the control points. There are 8 regions among the two sets that have true correspondences, in the other set. The goal is to determine high similarity values derived from moment invariant for the true

correspondences and suppress the similarity values for the moment invariants obtained for false correspondences.



Figure 8: Selected control points after uniqueness checking with correlation as the similarity measure threshold at 0.9

Similarity Measure: AMI of Gray-Scale Image and Rotation Invariant Texture

The first attempt in establishing this relation was to determine the similarity based upon the moment values obtained from the original regions and the image that describes the texture of the given image, formed from subtracting from the center pixel, the intensity values of the surrounding 8 pixels. The images corresponding to these regions are also shown in **Appendix A**. Similarity is defined as the sum of squared differences between the moment values obtained for any two regions being matched. The column and rows represent the number of the region corresponding to the order in which they are shown in **Appendix A**.

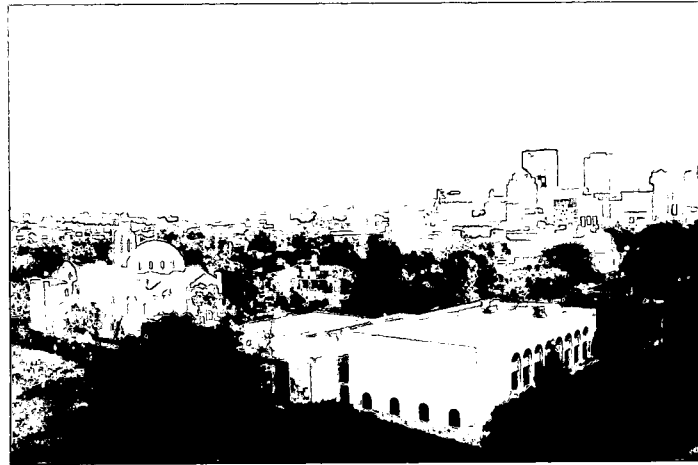


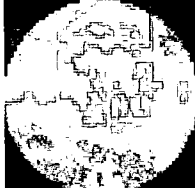


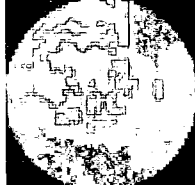


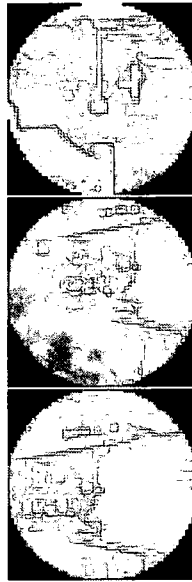
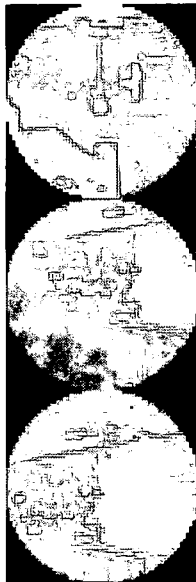
Figure 9: Selected control points after uniqueness checking with correlation as the similarity measure threshold at 0.9

The logarithm of moment values obtained was determined to increase the value of squared differences, for the two images. This was first suggested by Hsia (1981), who noted that the dynamic range of the normalized moment values can be quite large (5).

Tabulated below are the set of true matches obtained from the similarity measure described above,

Table 2: Similarity values obtained from moment invariant of original image and the texture descriptor

Regions (dayton1.tif)	Regions (dayton2.tif)	Similarity Between the Regions: $S = \sqrt{(M_{11} - M_{21})^2 + (M_{12} - M_{22})^2}$
		5.3588
		4.8563
		4.7639



5.8736

5.0700

4.3056

The results shown above are quite similar to each other and enable pairings. On analyzing the matrix obtained, it is found that there are several matches that lie in the same range as the true ones, but do not represent true matches. The threshold is not global, since there is no clear difference between the similarity of the true and false matches. Even a single outlier would fail the algorithm and hence modification needs to be made to this technique in order to eliminate this problem.

One of the problems in using the technique mentioned in the first attempt is that the description of the texture of the image yields an image, that has a distribution, resembling that of an exponential curve, and most of the intensity values lie in the range 0 to 0.25. The affine moment invariant of such an image would be an extremely small value. Similarity measure, which is obtained as a sum of squared differences between the moment values obtained from the texture descriptor and the original image, would also be diminished, when compared with the values obtained from the moment analysis of an invariant descriptor, which has the same range as that of the original intensity function. Therefore, scaling the range of the texture descriptor is necessary to enable setting of a global threshold.

Similarity Measure: AMI and Range-Scaled Texture Description

Histogram Specification for Range Scaling

Histogram specification, which is an image enhancement technique, is commonly used in object recognition to improve the properties of the intensity function of an image. This is done by transforming the probability distribution function or histogram of the image, by a suitable transformation, arbitrarily specified by the user. For the purpose of automating the similarity measure technique an approximation of desired transformation is needed for this purpose. Since, histogram for all the image regions, taken from the **Data Set A**, is quite similar to each other (**Figure 11, Figure 12, Figure 13 & Figure 14, Figure 15, Figure 16**) below, this can successfully be done.

The approach taken was to specify, for each image region, in **Data Set A**, a transformation function, which would transform the histogram of the regions, into a histogram that resembles a Gaussian distribution of a given mean and variance. The mean and variance is found, by iterating over different values and choosing the one pair that provides the best results, since, there is no deterministic relation between the transformation function of the image and the desired transformation function, which would yield an accurate result on moment invariant analysis. The histogram obtained from texture descriptor resembles an exponential distribution, on observation of a typical region, but in reality could be random. So, any attempt in determining a relation between the mean of the distribution of the image and the mean and variance of the Gaussian transformation function is likely to fail, not to mention the fact that, it would unnecessarily increase the computation load on the algorithm and hence, no such attempt was made to determine this relation. A value of 0.25 for mean and 0.25 for the variance of a Gaussian distribution as a global transformation function, works well for the purpose. However, the results obtained for others have also been summarized in **Table 2**.



Figure 10: Region of Interest 2 of image dayton1.tif

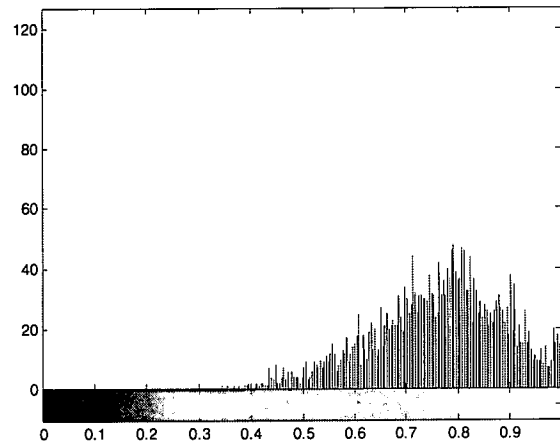


Figure 11: Histogram of the region roi2ofim1 c.jpg

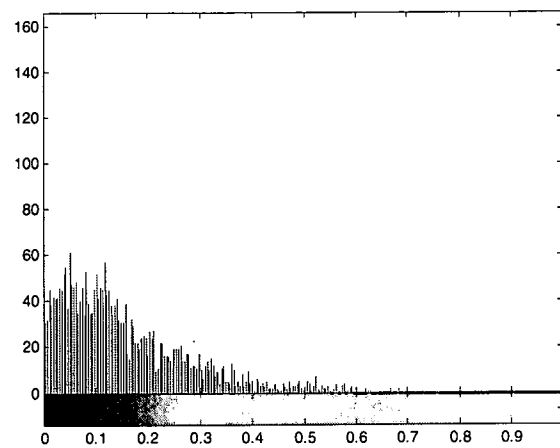


Figure 12: Histogram of the texture description of the image

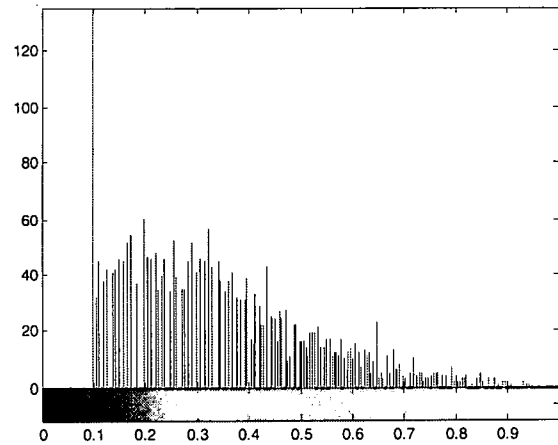


Figure 13: Histogram specification quantizes the distribution of the intensity values

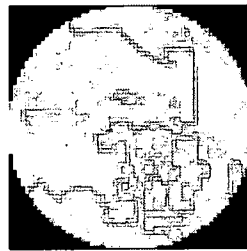


Figure 14: Region of Interest 7 of Image dayton1.tif

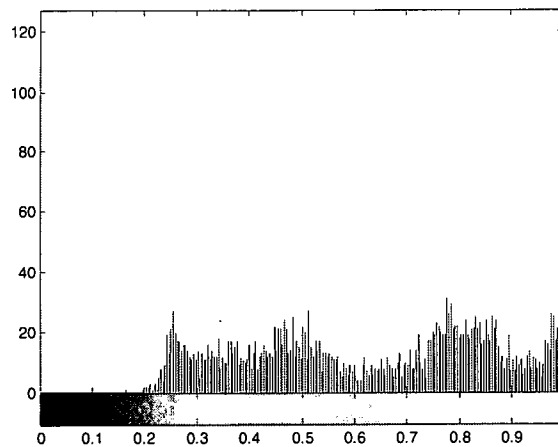


Figure 15: Histogram of the texture description of the image

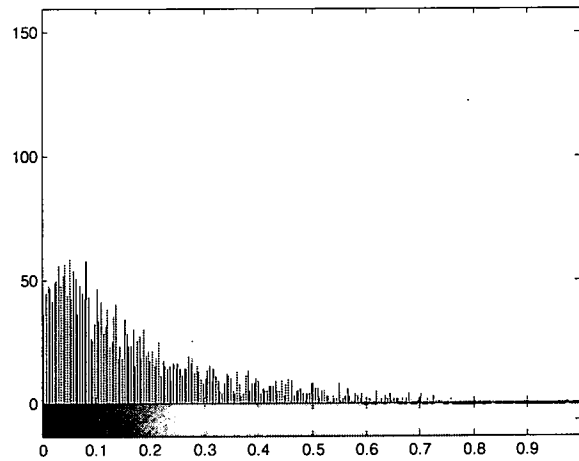


Figure 16: Histogram specification quantizes the distribution of the intensity values

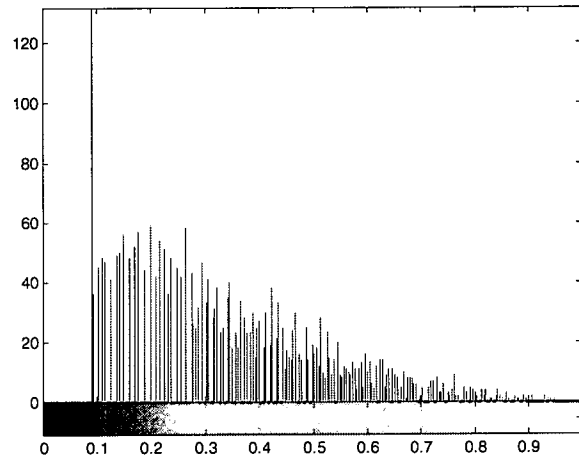


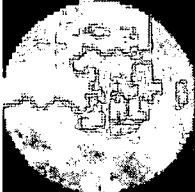
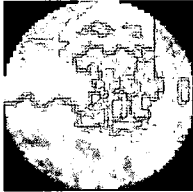
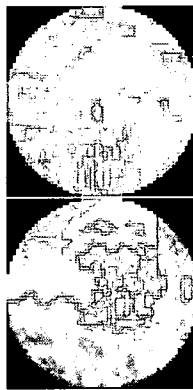
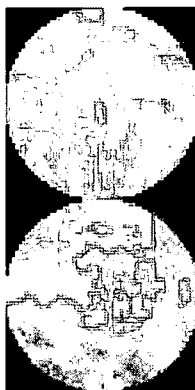


Figure 17: Histogram specification quantizes the distribution of the intensity values

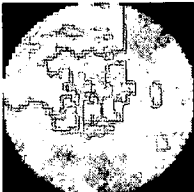
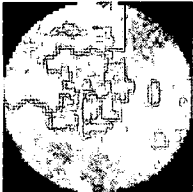
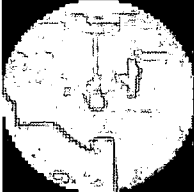
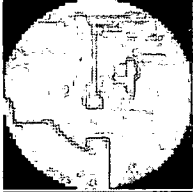
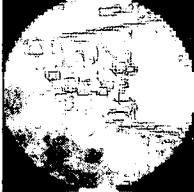

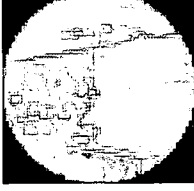
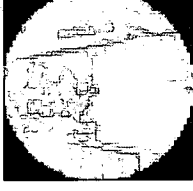
Table 3: Similarity measure between regions of interest from range scaled texture

Regions (dayton1.tif)	Regions (dayton2.tif)	Similarity Between the Regions: $S = \sqrt{(M_{11} - M_{21})^2 + (M_{12} - M_{22})^2}$
		5.2717
		4.4312



5.2717

4.4312

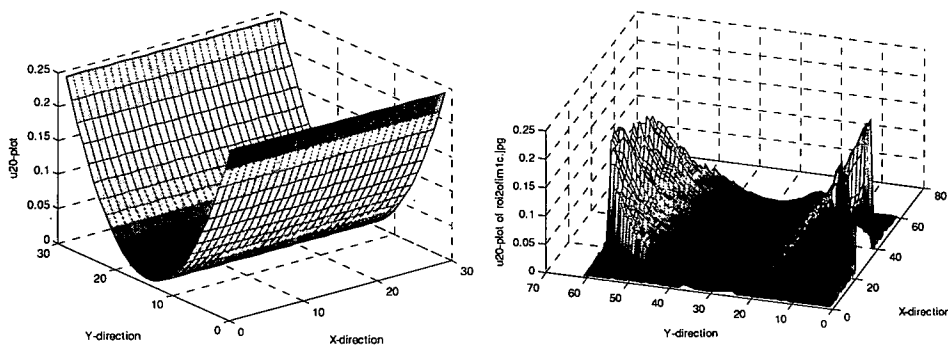
		4.5694
		5.0258
		4.5062
		4.0637

The results obtained above show, that for the selection of all correct matches, the joint similarity value obtained from the moment invariant analysis, of both the images, should have a global threshold at 4.0637. This suggests that all the outliers in the two image, must possess a similarity value of lower than 4.0637, with respect to all the regions lying in the other image. But there are several incorrect pairs still left that show a similarity value higher than the lowest from among the correct pairs.

Exponential Affine Moment Invariant

The next attempt was based on the following observation, made by studying in three dimensions, the weight (product of intensity values), that is multiplied, at each position to the product of the spatial positions in x -direction and y -direction, raised to some power, with centroid of the region as the origin. The simplest possible moment invariant was used in the previous discussion to determine a single value for similarity. The moment expression, given in **Equation 3.3**, is in continuous form and its discrete form is used for the

purpose of implementation. The plot below illustrates, that if the centroid of the region is also the center of the circular window, then the intensity pixels lying close to the center of the window, would not contribute highly to the summation, since the values of the squared distances in x -direction or y -direction, would rapidly increase to a large value relative to the values that lie near the center of the window.



So, moment summation determined using the invariant expression, is being dominated by the values of the distances in two directions, weighted by the intensity values, which lie along the border of the window. This is a frequently stated problem in this field, where all the values of the intensity function are not being represented equally with proper emphasis to the position where they lie in the image, with respect to position of the pixel in all other images that are candidates for a match. Hence, distinguishing between the images become dependent upon intensity values lying in some selected areas in the window, where the differences in the intensity values would reflect in the overall summation. This is intuitive, since, the moment invariant is derived by taking the cross product of two points, the squared area of the parallelogram would be significantly higher for the pixels lying away from the centroid.

In order to determine a solution to this problem, another moment invariant needs to be derived which doesn't suffer from this drawback. A moment invariant where the

cross-product is raised to the power of $\frac{3}{2}$ could be derived. But, since, the spatial position of the pixel could be negative, complex values would appear when the moment expression in **Equation (3.5)** is implemented.

$$I_{3/2} = \frac{\mu_{3/2,0}\mu_{0,3/2} - \frac{3}{2}\mu_{1/2,0}\mu_{0,1/2}\mu_{0,1}\mu_{1,0} + \frac{3}{8}\mu_{0,2}\mu_{2,0}\mu_{0,-1/2}\mu_{-1/2,0}}{\mu^{7/2}_{0,0}} \quad (3.5)$$

Since, the function e^{x^2} varies slower than x^2 , when the values of spatial positions represented by x , are normalized to lie in the range of $(-1, 1)$, an appropriate choice for the moment should be to find the exponential of the cross of product of the two points. Hence, a moment is derived which resembles the characteristic function of the first order affine moment invariant.

$$I_1 = \iiint e^{x_1 y_2} e^{-x_2 y_1} f(x_1, y_1) f(x_2, y_2) dx_1 dx_2 dy_1 dy_2 \quad (3.6)$$

The expression above doesn't have any normalization of the invariant expression to cancel the factor of the determinant of the Jacobian matrix, since, there is no way of doing so. The similarity would be established with an assumption of the presence of only rotational transformation between the two regions, in which case, the determinant of the Jacobian matrix factor equates to unity.

The moment invariant, however, does have a desirable property of weighing the pixel intensities in different regions with almost equal weight. The following figure (**Figure 18**) shows that for a standard circular region, the summation value at each pixel position in the image. Since, all intensity values are one, this plot shows just the summation value of the exponential of the cross product of two points weighted by ones in a circular region of the **Equation 3.6**.

The lower values present, on outer areas of the image in **Figure 18** is due to the circular shape of the image. The next attempt is made by implementing the moment

invariant expression from **Equation 3.6** and determining the similarity values obtained from both original as well as the rotation invariant texture description of the image. The expression for determining similarity remains the same, so does the values of the mean and variance of the global transformation function.

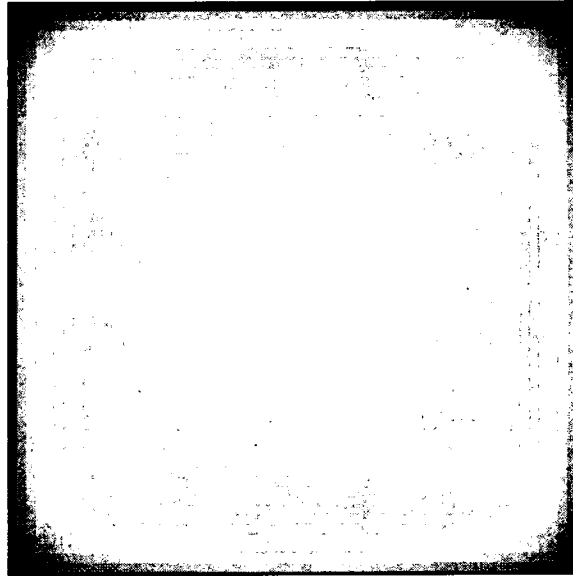



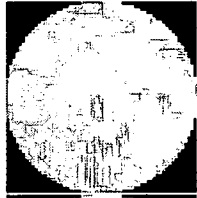
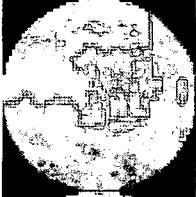
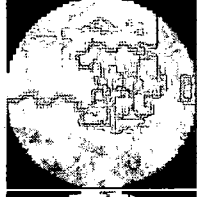
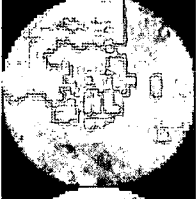
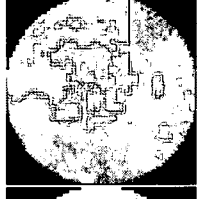
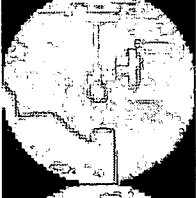
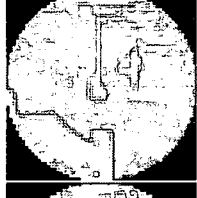
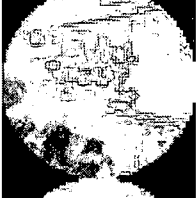
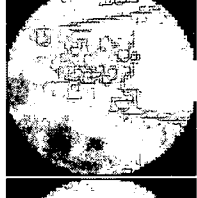
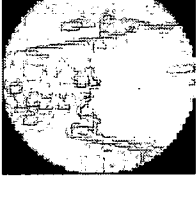
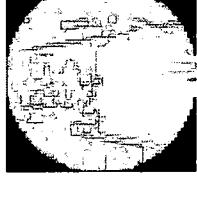
Figure 18: Summation value at each pixel position of the moment invariant for the standard image

In the result shown in **Table 4**, the values of the logarithm of the sum of squared differences of the moments, determined to establish similarity between the regions has been summarized. Global threshold is equal to the lowest value of the similarity measure attained observing the correct matches. On removing the Jacobian adjustment term from the analysis, the results were obtained that showed improvement, indicating that the even in case of a scaling transformation, the region first needs to be scaled down to meet the scale of the reference region. One more thing, as was discussed in the previously, the values of the similarity measure, even though are normalized with respect to the moment value of the standard image, have not been assigned a global meaning except the joint similarity value obtained from sum of squared difference of the individual moments of the original and textured images. The normalized moment values represent, how much the intensity composition of the region of interest, differs with respect to that

of the standard image. As would be seen later, this information could be used as a criterion for elimination of apparently incorrect matches, in order to reduce ambiguity as well as to reduce the computational load of the algorithm.

The next table (**Table 4**) summarizes the correct matches obtained after removing the Jacobian term from the analysis, recognizing the fact that for the selected window size, only the assumption of rotational transformation holds true.

Table 4: The similarity measure between dissimilar regions lying in the range of true matches

Region (dayton1.tif) Reference Image	Region (dayton2.tif) Template Image	Similarity Between the Regions: $S = \sqrt{(M_{11} - M_{21})^2 + (M_{12} - M_{22})^2}$ $\sigma = 0.25, \mu = 0.25$ $\sigma = 0.35, \mu = 0.25$	
		9.3469	9.3481
		9.3651	9.3680
		9.6930	9.6930
		10.8380	10.8331
		9.2572	9.2581
		8.3001	8.2996

The good information from the result is that the values obtained for similarity measure, show fewer variations and are almost all in the range of 9.0-10.0. This is desirable, since, it enhances the chances of obtaining a global threshold for all kinds of regions in question. If, on the other hand, the value of similarity varies widely with nature of the region, the chances of an incorrect pair of regions lying in the range of the correct pairs, increases.

The **Data Sets A** and **B** studied for the determination of similarity of the regions, is a good data set, since, on visual inspection, there is minute difference in the contrast, brightness and blur, which could affect the similarity value of the moment invariant for the correct matches. Next, all the incorrect matches have been summarized in **Table 5**, to see how good a similarity measure is the previous method.

There are several outliers in doing this as well, since the number of outlying matches exceeds that obtained from the previous result. But, on closer inspection it was observed that since, only one true match, the one between the region of interest 19 of the reference image and region of interest 13 of the template image, have shown a low similarity value of approximately 8.30, has lowered the global threshold for elimination of the incorrect matches

Table 5: The similarity measure between dissimilar regions lying in the range of true matches

Region (dayton2.tif) Template Image	Region (dayton1.tif) Reference Image	Similarity Between the Regions:
		$S = \sqrt{(M_{11} - M_{21})^2 + (M_{12} - M_{22})^2}$



9.2144

8.9679

11.2018



9.3222

10.1120

9.5214

10.9675

13.3287

10.7982

9.9608

10.5283

8.4840

9.7830

11.0640

11.1998

8.7334

10.1079



10.3512

9.8479

10.6464

8.3130

8.7671

8.2889

Also, since, the concept of choosing a circular window depends upon the Jacobian term, the method just mentioned is a more appropriate technique to approach establishing of the similarity. But, apparently this method also doesn't suffice and needs modification, in order to work properly.

Similarity Measure: Variations and Composition Based Similarity

So far, all the modifications to the original moment function, has been implemented based upon global assumptions, which attempts to establish similarity based upon a common property shared by all regions. The next attempt would be based upon the observations obtained from case studying similarity and dissimilarity between the particular regions that exist in the list above.

Firstly, the standard image (blank circular image), is considered and contribution of pixels in each region to the overall summation is studied.

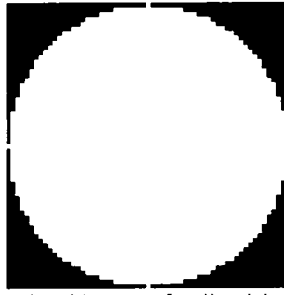


Figure 19: Standard Image for the blank circular disk

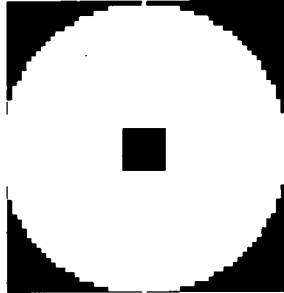


Figure 20: Standard Image after removing pixels from the center

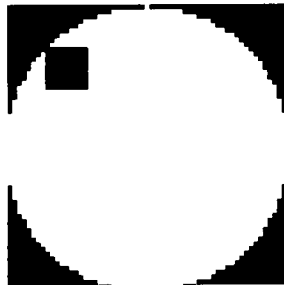


Figure 21: Standard Image after removing pixels from the sides

Table 6: Normalized affine moment invariant for Figure 25, Figure 26 and Figure 27

Figure 31	Figure 32	Figure 33
1.0039	1.0041	1.0038
1.0039	1.0043	1.0038

The observation above shows that a low intensity valued pixel, positioned close to the center of the image, (**Figure 22**) has a higher moment summation compared to that positioned on the sides (**Figure 23**). This observation infers, that darker regions present at or around the center of the image, would lower the moment summation lesser, relative to a similar region that is positioned on the sides of the image. The result summarized in **Table 6**, is normalized by square of the sum of all intensity values in the image.

The images shown in **Figure 22** and **Figure 23** are resulting of the difference of summation contribution at each pixel position of the standard image and the two reduced images in **Figure 20** and **Figure 21**. Apparently, **Figure 22** and **Figure 23** illustrate the positions in the image, where the differences would occur and quantifies the amount by which they would.

The percentage of the drop in the moment summation value is lower in the **Figure 22** than in **Figure 23**. Simply, changing the position of the low intensity value square area, would not change the moment summation, since, the change would manifest itself as rotation, which the moment summation value is invariant to. However, it is understood that when the pixels are removed from the center of the image, the contribution at each pixel position doesn't change significantly, since there are more pixels positioned along x and y columns at the center of the image. Hence, the summation contribution of the pixel is higher than the ones present away from the center of the image. The pixels that are positioned on the edge, contribute significantly to the moment summation value, since, the intensity values at that position is multiplied with the values of the exponential function of the outer product of the spatial positions ($\exp(-kxy^T)$), which is high at the edges.

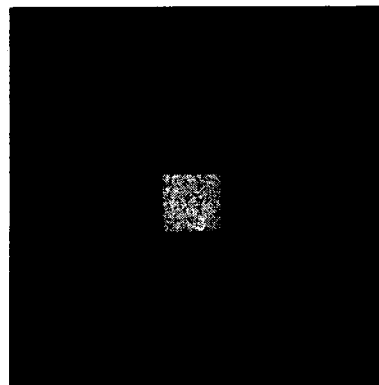


Figure 22: Illustrates the difference in the summation at each pixel at the center with respect to the standard



Figure 23: Illustrates the difference in the summation at each pixel at the sides with respect to the standard

When the pixels positioned are removed, the lowering of the moment value is more significant than when the pixels of same intensity are removed from the center of the image. The difference however is not very prominent as can be observed from the **Table 6**. The result for a larger square at the center and at the edge of the image has been given in the second column of **Table 6** and follows the same trend. One way of developing an understanding of this, is to see the removal of gray-level pixel values from the center as mostly the reduction of particular intensity values, which is used to normalize the invariant. The moment summation diminishes very little.

This attempt is based upon using moment invariant alone to determine the similarity between any two regions. Since, moment invariant is the summation of intensity values in an image, it doesn't have the capability of recognizing where in the image the variations are occurring. In order to determine an invariant, based upon the affine moment invariant, which has the capability of distinguishing between regions with distinct spatial structures, an invariant in the previous attempt was derived by determining the affine moment invariant of the rotation invariant texture of the image and combining that information with the moment invariant of the original region. To use the moment invariant alone, however, the moment invariant of segments in the image would provide a way of determining where the variations in intensity values occur and the intensity composition in those regions.

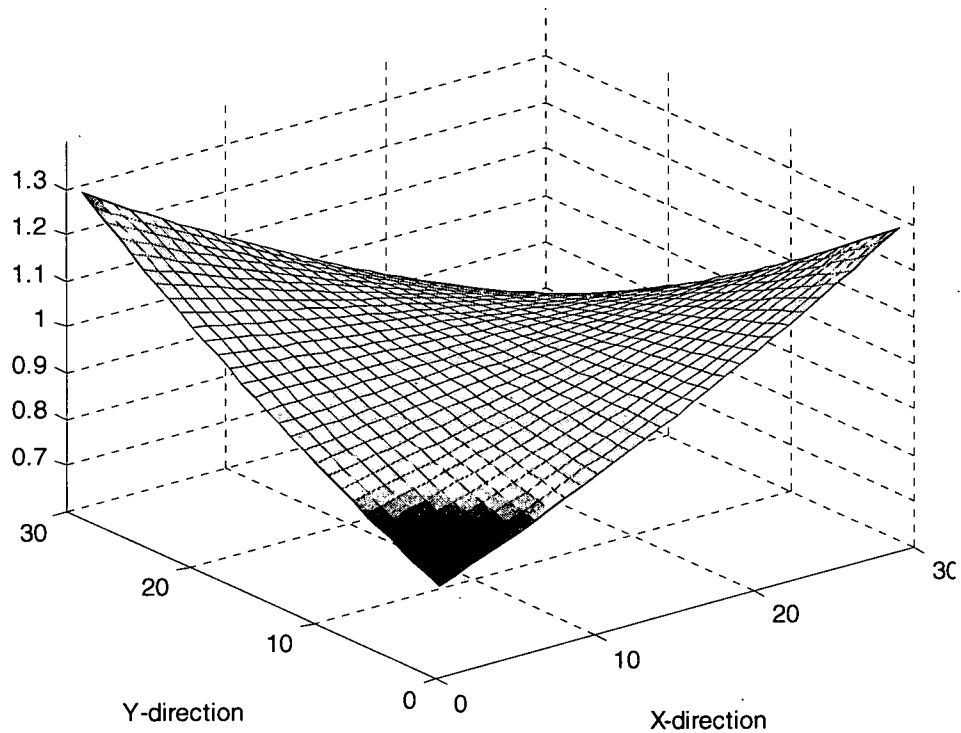


Figure 24: Plot showing the $\exp(-kxy^T)$ function

This would be enough information to enable distinguish-ability between regions that are truly similar and the ones where the overall gray level composition and variations only resembles that of the other. Higher order moments are shape descriptors and the moment derived in this thesis is also a shape descriptor but has a desirable property of weighing the intensity values in all parts of the image equally.

The normalization of the affine moment invariant of the region with the moment invariant of the standard image provides a measure of the composition of intensity values in the image, relative to the maximum, whereas, the variations do not change. An image region, which is dominated by the higher gray level values would yield a value in the range of **0.5-1.0** for the normalized moment invariant. On the other hand, the image region, dominated by lower intensity values would yield a value in the range of **0.0-0.5**. This defines a global characteristic which is relevant to all the regions and could be used

to distinguish between two regions that are completely dissimilar and hence would save the software from carrying out unnecessary moment calculations.

In order to determine the moment invariant analysis result, that identifies the differences in the positions of distinct intensity variations in the image, the image first needs to be segmented into different areas and the moment invariant values of these areas determined and matched. In order to implement this, however, the individual segments, in two corresponding regions, needs to be the same. If one assumes, the spatial transformation to be rotational, for simplicity, the simplest way to approach this problem is to consider successive circular regions around a given control point and proceed with an assumption that for a corresponding region, the circular regions would exhibit the same gray level composition.

The observation from the previous attempts could be used to better the resulting technique, by realizing that as long as the image region window is large enough to contain significant intensity variations, the ambiguities arise mainly due to the presence of small regions in the image, the intensity nature of which, is compensating for the differences in intensity composition at other positions. Hence, considering successive circular regions around a control point is not as good a technique, as to consider a large region of fixed size and then removing successive circular regions from its center. This would help in determining the regions that do not correspond with each other at some succession of the summation of the intensity variation.

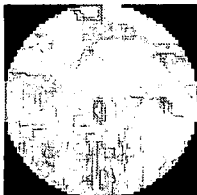
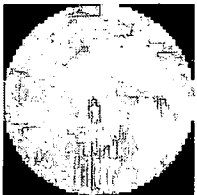
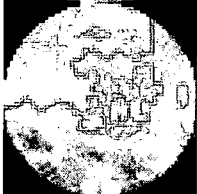
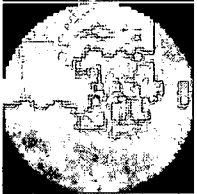
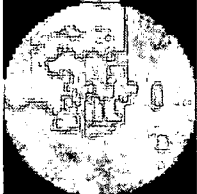
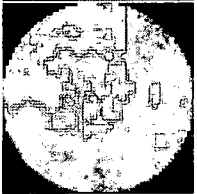
The algorithm for the observation was developed, with iteration of a given region of interest from a reference image matched with that of the template image, over a set of circular overlapping regions of increasing radii masked, from the center of the image. The false matches are eliminated at each iterative step, with the criteria of elimination being a high value for the difference in the gray-level composition of the region. This requires setting a definite threshold for the elimination at each iterative step. Since, the

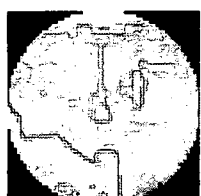
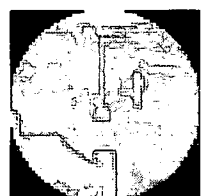
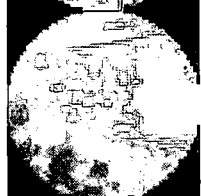
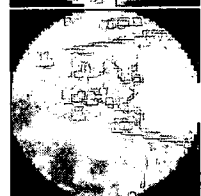
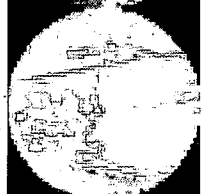
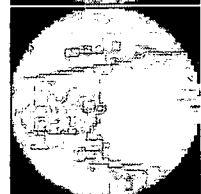
similarity between two dissimilar regions can never be above a given threshold, if high differences are observed, they are due to gray-level distortions. For the purpose of testing the feasibility of the technique, I used three iterations to eliminate the false matches. It is worth noting, that as long as the regions are homogenous, the moment values obtained would not differ significantly and would increase the chances of detecting incorrect similarities.

The following are results obtained when the technique was tested with **Data Set A**, given in **Appendix A**.

The results show that the true matches have shown consistently high results over the three iterations. Since, three iterations were used, three separate thresholds were used for the elimination. Keeping the threshold high enough to include the most controversial similarity and low enough to assure the elimination of all the false matches is the primary goal here. It was determined that the technique does eliminate most of the false matches. But, some false matches still remain and have been summarized below in **Table 8**.

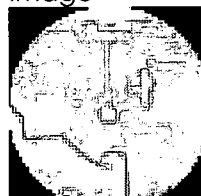
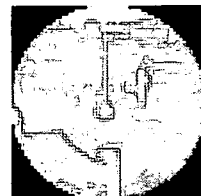
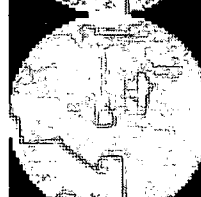
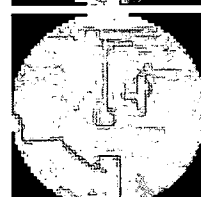
Table 7: Similarity determined over the three iterations eliminates most false matches

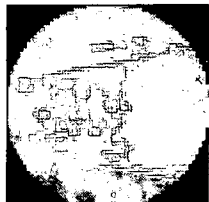
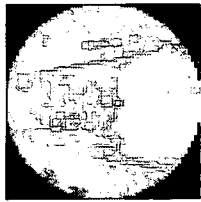
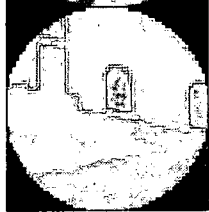
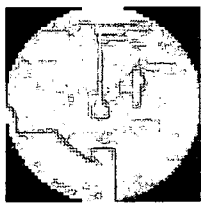
Region (dayton1.tif) Reference Image	Region (dayton2.tif) Template Image	Similarity Between the Regions:		
		$S = \sqrt{(M_{11} - M_{21})^2}$		
		Iteration 1	Iteration 2	Iteration 3
		9.4193	9.3516	11.8328
		9.6391	10.1727	10.0098
		10.0672	9.8019	10.2223

		11.2535	12.1227	9.9790
		9.5384	9.6943	9.7768
		8.5063	8.4692	9.0784

The result summarized in **Table 7** and **Table 8** show, that the visually similar regions are showing high similarity except the last among the outliers. This match has been analyzed in more details in a later part of the thesis. Even though the outliers were eliminated based upon visual similarity between two regions, the threshold was set manually based upon the least value of similarity obtained for this particular data set. In my opinion, in order to automate the process, a large number of such data sets should be used to set the threshold, by recording the lowest value of similarity that is observed among true matches, provided there is no contrast and brightness differences among the regions.

Table 8: Similarity measure showing results for false matches of Set A

Region (dayton1.tif) Reference Image	Region (dayton2.tif) Template Image	Similarity Between the Regions: $S = \sqrt{(M_{11} - M_{21})^2}$		
		Iteration 1	Iteration 2	Iteration 3
		10.0902	9.8465	10.3284
		11.3101	10.4689	7.6678

		7.7534	7.9298	8.3190
		11.9150	9.4076	8.5789

The dissimilarity result shown in **Table 9**, confirms that the region that has maximum visual similarity to the other has been selected over the three iterations. The values obtained for the second and third column of this Table shows that even though these regions have shown correspondence, the values of similarity obtained are near the threshold and could be eliminated if the threshold was set a little higher. The attaining of high-similarity between the region of interest 21 in image **dayton1.tif** and region of interest 10 in image **dayton2.tif**, shows that my initial assumption of using three iteration, (removing circular regions of radii 10, 15, 25), doesn't provide the elimination of all the outlying matches. Hence, there is either the need of a technique that involves more iteration or uses texture description of the region in order to eliminate the left incorrect pairings. In order to study the reason for this behavior, the following case studies were done.

Case Studies I:

Region of Interest 21 of dayton1.tif and Region of Interest 10 of dayton2.tif



Figure 25: Region of Interest 21 of dayton1.tif

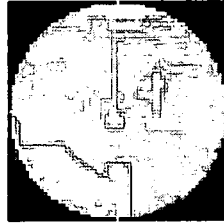

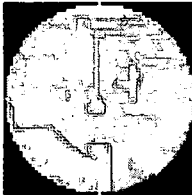
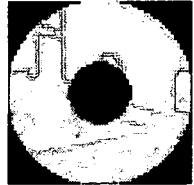
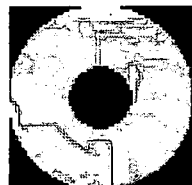


Figure 26: Region of Interest 10 of dayton2.tif

The two regions are visually dissimilar from **Data Set A** and **Data Set B** in **Appendix A**. The two regions have high similarity for all iterative steps in the algorithm, even though they are visually quite dissimilar. This similarity is due to the symmetric distribution of the intensity values about the center of the image and therefore, the change in gray level composition remains the same. There is, however, another way of creating the dissimilarity that we seek, by determining the similarity obtained from the texture description of the image, which would be discussed later. It was, however, the choice of a good data set that this flaw in the technique of determining the similarity was observed. This shows to how much extent the gray level values of an image region could become similar, without the ability to quantify differences based upon the differences in the positions of intensity variations in the two images. A larger region of interest could also be considered for the increasing the probability of the detection of the dissimilar regions.

Table 9: Similarity measure result for region of interest 21 of dayton1.tif and region of interest 10 of dayton2.tif

Region of Interest 21	Region of Interest 10	Similarity Measure
		11.915
		9.4076



8.5789

Before determining, if the incorrect match observed above could be eliminated using the texture analysis, the algorithm was implemented for another data set to ensure that the given technique wouldn't fail. The data set consists of 8 regions each in two images (**dayton3.tif** and **dayton4.tif**). The **Data Set B** has been shown in the **Appendix A** and would be used to demonstrate the difficulties encountered in using this technique.

The results obtained from this data set have several ambiguities in the determination of the correct pair of matches and have been studied among cases in the following sections.

Brightness and Contrast Compensation: Histogram Specification

Case Studies I

Investigating Dissimilarity between the Region of Interest 2 of Image **dayton3.tif and Region of Interest 4 of **dayton4.tif****



Figure 27: Region of Interest 2 of Dayton 3



Figure 28: Region of Interest 4 of Dayton 4

The moment values of the given regions, even though they seem to have a very high similarity with respect to each other are significantly different. On careful analysis, it is determined that one region is less bright than the other, especially at the areas where there is effect of shadow (not very clear in the images shown here) and it seems as if due to the changes in illumination of the scene, the shadows have become more prominent bringing down overall sum of the intensity values of the region by a significant amount. The following were the values obtained for affine moment invariant (See Table below).

In the previous data set, the threshold set for each stage of the iteration was 7.600 which correspond to a squared similarity measure of 0.00050. Hence, this pair of correct match would get eliminated at the first stage of the iteration. So, the solution of the problem was determined and once again histogram specification came to the rescue.

Moment Invariant of Roi2	Moment Invariant of Roi4	Similarity Measure
0.52713976	0.47024898	5.733243979
0.52713976	0.52815314	13.78892800

The change in contrast can be eliminated using the transformation function obtained from the region belonging to the reference image and can be specified to the region belonging to the template image.

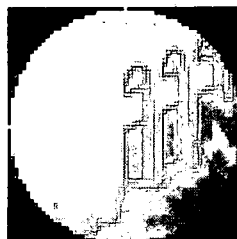


Figure 29: Histogram Specified Region of Interest 4 of Dayton 4

The shadow length of the tree in **Figure 28** is larger than the shadow of the tree in **Figure 27**. But due to histogram specification (specified to match the distribution of the reference region), the intensity distribution in both the images is equivalent to each other, and the conversion of a few lower intensity values, in the template region, due to higher length of the shadow has been compensated by brightening a few pixels which are

originally darker. Shadows are an issue in image processing, since their presence affects most of the statistical techniques from being used effectively as was seen in the example above. The effect of shadows cannot be completely eliminated. The method requires the calculation of the histogram specified values for every region in the reference with every region in the template image. 64 specifications for a data set of 8 regions each are required. For larger number of regions, which would normally be the case, the computational complexity would be quite high. For an image pair with 20 regions each, the amount of time it takes for the histogram specification of each region with the other corresponding region would result in 400 histogram specifications. Carrying out the operations on a 61×61 image would require a significant amount of time to process, whereas without the need for histogram specification, the algorithm would take only a few seconds to execute.

Taking another example from the same data set shows similar problem and the solution to the problem by using histogram specification for the reduction of contrast and brightness differences.

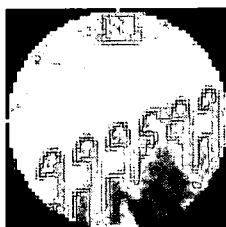


Figure 30: Region of Interest 5 of Dayton 3

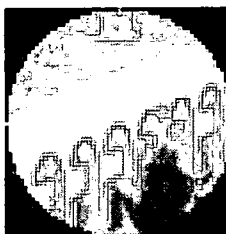


Figure 31: Region of Interest 7 of Dayton 4



Figure 32: Histogram Matched Region of Interest 7 of Dayton 4

On specification of the histogram from one region to the other, the results are much better. The following results show the similarity between the regions before histogram specification to improve the brightness and contrast of the second image by applying as the transformation function, the histogram of region 1. The areas in the image where the contrast and brightness changes are visible are the roof of the building and the sides of the windows in the first two images and third image shows the contrast and brightness corrected image.

Results

The application of the histogram specification of the reference region to the template region should not affect the results obtained from the similarity measure. Even though the presence of the regions of interest 21 of **dayton1.tif** and 10 of **dayton2.tif** as highly similar is a shortcoming of the technique, the correction of brightness and contrast could provide us with a condition to eliminate the incorrect pairings. This is due to the fact that similarity between any two regions determined from the difference of normalized moment invariant, may arise due to the presence of similar intensity characteristic of the image and doesn't necessarily require the presence of the pixel intensities of the original image. Thus, the histogram of the two regions may not match. On application of histogram specification with the transformation function obtained from the reference region, the pixel intensity probability distribution is transformed to match the distribution of the reference region. This changes the intensity values at individual pixel, thereby changing the overall intensity characteristic of the template region.

Hence, the similarity that was previously observed, for the two dissimilar regions, would vanish.

On the other hand, the contrast and brightness correction using histogram specification could also cause incorrect pairings. The problem with histogram specification is that, if the region is primarily bright (and does not have much variation), the transformation function of the darker region that it's being matched with, would reduce the value of all the pixels in this region to a lower value. There is a high probability that these two images would exhibit high similarity in all iterative steps, since each of the sub-regions in these images would not hold extra information which would significantly reduce the similarity between the regions. If the mean of the region before and after the specification is higher than 0.1, it is an indication that the regions aren't similar, and could be eliminated. This would reduce the computational complexity of the algorithm. The global threshold set for the elimination of the outliers is 7.600 and the tabulated values show the corrected result to be much higher than that.

Moment Invariant of Roi5	Moment Invariant of Roi7	Similarity Measure
0.52244342	0.46142077	5.59302755
0.52244342	0.52383686	13.1069143

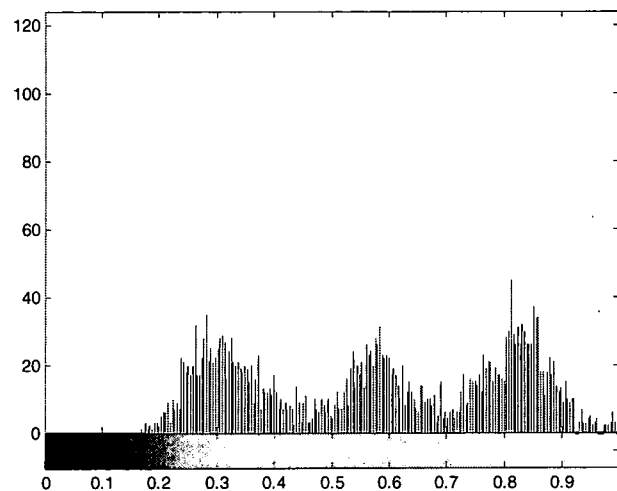


Figure 33: Histogram for Region of Interest 21 of image dayton1.tif

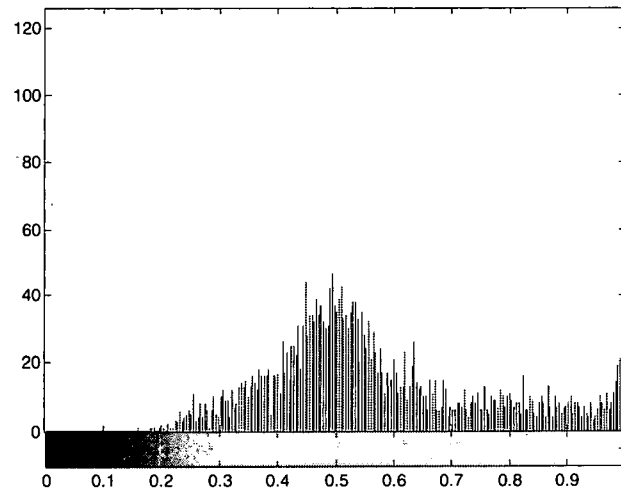


Figure 34: Histogram for Region of Interest 10 of image dayton2.tif

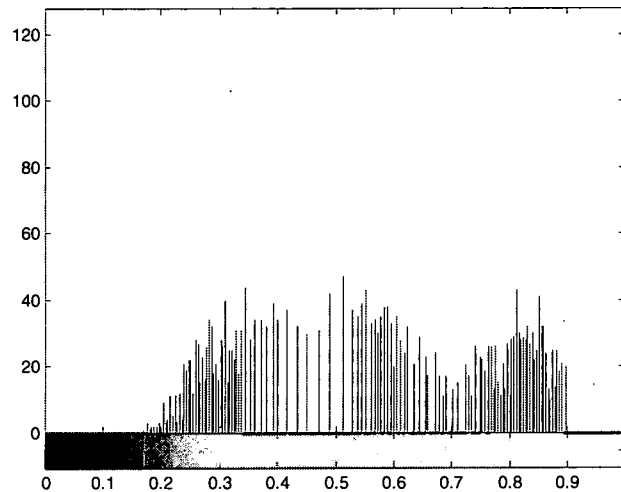


Figure 35: Histogram Specified image for Region of Interest 10 of image dayton2.tif

From the result obtained on using histogram specification for the correction of brightness and contrast, the similarity measure developed for the two images yields a good result in cases where there is no brightness and contrast difference (shown in **Figure 36** and **Figure 37**), where all the outliers were eliminated and no incorrect pair of control points were selected.

The threshold set for the elimination at each step was 0.0005, 0.0001, 0.0001, and 0.0001. The contrast and brightness adjustment using histogram specification has an added advantage over linear adjustment of being a more computationally efficient technique. Even though from the result it might seem that the technique works perfectly,

the technique yields outliers even in the cases where the contrast, brightness and blur differences are minimal indicating that the similarity measure needs improvement. **Figure 38** and **Figure 39** shows the result obtained on using **dayton2.tif** as the reference image and **dayton3.tif** as the template image and using histogram specification improve the contrast and brightness condition of the template regions.



Figure 36: Result of Similarity Measure on the image dayton1.tif with dayton2.tif

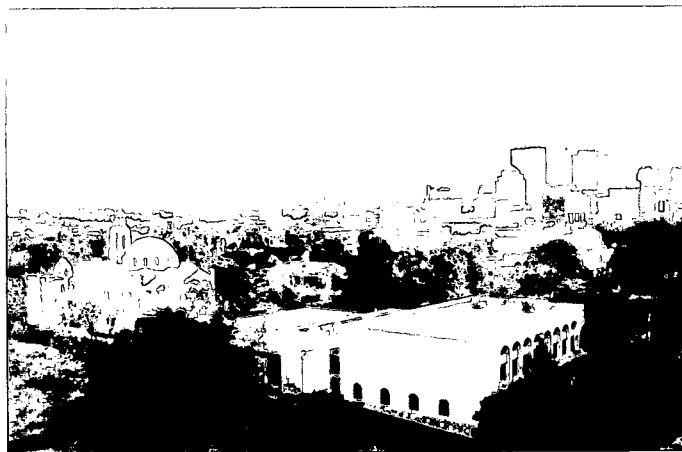


Figure 37: Result of similarity measure on the image dayton2.tif with histogram specification

There are 5-outliers detected from the similarity measure algorithm for the given pair of images (**dayton2.tif** and **dayton3.tif**). The contrast and brightness differences between these two images are slight and not visually detectable (See **Fig. 38** and **Fig. 39**). It would be important to analyze the result obtained on using the similarity measure algorithm for the detection of corresponding control point pairs for two images with detectable

contrast and brightness difference. The result obtained from application of the algorithm to the image pair **dayton3.tif** and **dayton4.tif** has been shown in **Fig. 38** and **Fig. 39**. The result obtained, shows the detection of at least 5-outliers. The result seems to be comparable to that determined from the image pairs where there is no difference in contrast and brightness between the two images.

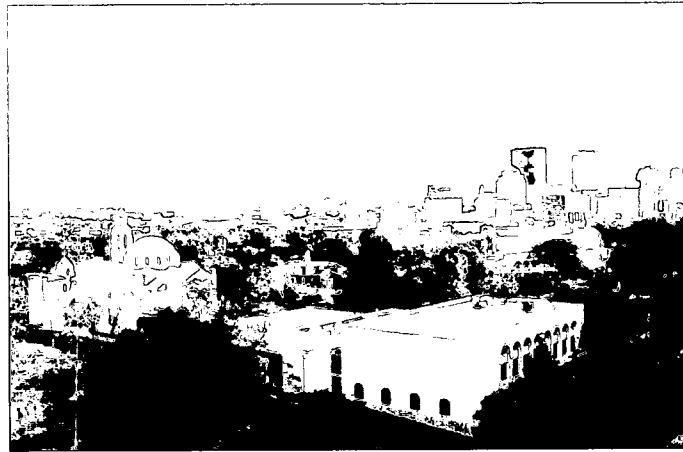


Figure 38: Result of Similarity Measure on the image dayton2.tif with dayton3.tif

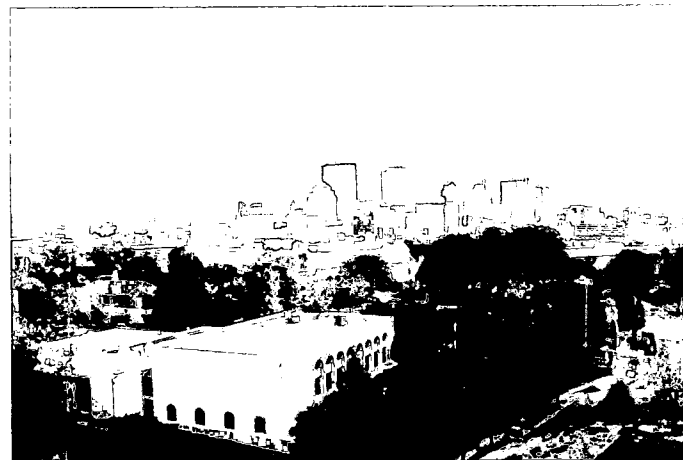


Figure 39: Result of similarity measure on the image dayton3.tif with histogram specification

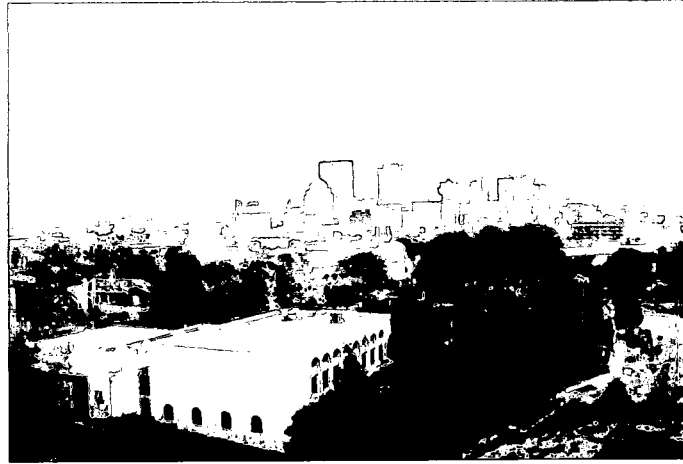


Figure 40: Result of similarity measure on the image dayton3.tif with dayton3.tif

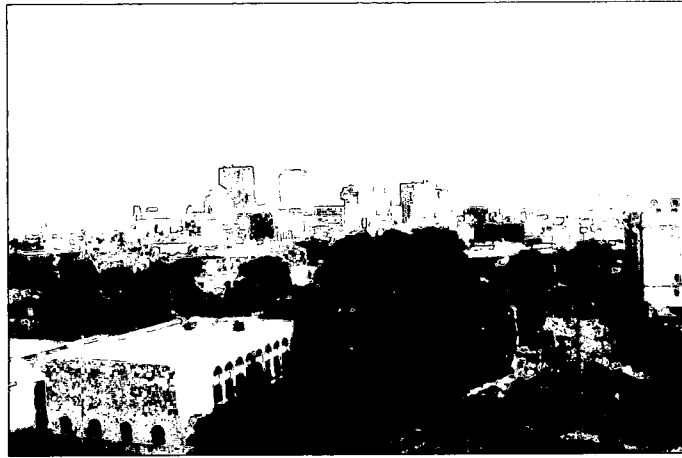


Figure 41: Result of similarity measure on the image dayton4.tif with histogram specification

Performance Measurement

The following Table contains the timing information for the execution of the algorithm on MATLAB for each of the three pairs of images. The time of execution of the similarity measure algorithm varies with each image pair and is the function of the number of distinct control points that are detected on the two images.

Reference Image	Template Image	Number of CP's in Reference	Number of CP's in Template	Time of Execution (in seconds)
Dayton1.tif	dayton2.tif	8	50	16.00
Dayton2.tif	dayton3.tif	50	38	82.76
Dayton3.tif	dayton4.tif	38	27	42.46

Linear Brightness and Contrast Adjustment

On detection of the outliers on automation of the task and from the observations determined from the results obtained from using the regions in **Data Set A** and **Data Set B** (See **Appendix A**), it was inferred that the technique mentioned in the previous section doesn't suffice and needs improvement in order to work appropriately. An attempt, therefore, was made to derive a technique of adjusting the contrast and brightness of the template region, assuming a linear distortion. As was mentioned before, the texture description of the region, which yields an image with high structural information and low intensity values, is invariant with respect to the contrast and brightness differences, iff the differences can be represented in a linear form.

Even though, the distorted image can't be recovered completely, the changes can be approximated in ways that are conducive for the entire task to continue.

Please note that we have ignored the factor of the determinant of Jacobian of the transformation matrix, since it has been assumed for the sake of simplicity that the following equations can only be used on purely rotated regions. The moment is the one measure that has so far been used as a basic tool to yield similarity between the regions. This leads to the second assumption that has been made in order to yield a solution to minimize the sensitivity of the moment invariant to these distortions, which is, that the two regions only differ by a change in contrast and brightness and the values thus obtained, would only be due to these changes.

$$\begin{aligned} M &= \iiint \int e^{x_2 y_1} e^{-x_1 y_2} (af(x_1, y_1) + b) dx_1 dx_2 dy_1 dy_2 \\ &= aM_x + bC \end{aligned} \tag{3.6}$$

Here, representation of the each variable used is the following,

$$M_x = \iiint \int e^{x_2 y_1} e^{-x_1 y_2} f(x_1, y_1) dx_1 dx_2 dy_1 dy_2$$

$$C = \iiint \int e^{x_2 y_1} e^{-x_1 y_2} dx_1 dx_2 dy_1 dy_2$$

This adjustment should yield the moment values that is as close to the true value of the moment M_p as possible

$$M_{x1} - aM_{x2} - bC = 0 \quad (3.7)$$

Another constraint equation can be obtained by the following. Let,

$$S_p - aS - b = 0 \quad (3.8)$$

Solving the equations the factor of a and b can be determined,

$$a = \left(\frac{M_{x1} - S_p C}{M_{x2} - SC} \right) \quad (3.9)$$

$$b = S_p - \left(\frac{M_{x1} - S_p C}{M_{x2} - SC} \right) S \quad (3.10)$$

Relations shown above, considers pixels from the entire region to determine the factors, however, from implementation point of view, this is quite complex. For two images with 20 regions each, there are 400 moment calculations required to determine the contrast and brightness constants only. Even though a better strategy is to determine the moment values before and then eliminate all the matches that are different by a given, more friendly, threshold, assuming that the values of contrast and brightness, in the two regions differ by a reasonable amount. This would reduce number of potential candidates that are similar to any given region by a specified amount.

Alternatively, another method could be followed, where the assumption of linearity holds true. For any two matching regions, the pixel with minimum and maximum value, inside a given region, would not change under a linear distortion of intensity values of

brightness and contrast. This would yield the required two equations to solve for **Equation 3.11** and **Equation 3.12**. This would also save the sought after time of computation from the computation of the moment, which is used in the previous solution. This would yield better solution, since, in the solution in **Equation 3.9** and **Equation 3.10**, constraints the moment values to equal 0 and then solves for the brightness and contrast correction factors that would meet the constraints and therefore, the initial and subsequent values of incorrect matches would yield a value of moment very close to 0. This defeats the purpose of the using moments to distinguish between the two dissimilar regions. However, observing values of brightness and contrast and setting a definite range that these values could fall into, would definitely help in eliminating the false matches as well

$$c = \left(\frac{(S_{pMAX} - S_{pMIN})}{(S_{MAX} - S_{MIN})} \right) \quad (3.11)$$

$$b = \left(\frac{S_{MAX}S_{pMIN} - S_{pMAX}S_{MIN}}{S_{MAX} - S_{pMAX}} \right) \quad (3.12)$$

The following were the results obtained when the equations were used before the similarity measure algorithms for the given data sets,

Even though, the brightness and contrast adjustment enhances the similarity measure, the degree to which such a difference is present in the two regions is not known. Therefore, when the enhanced image is obtained, the region can have the maximum value varying from higher than 1 to lower than 0. When the moment invariant is used to describe the region, then the presence of these pixels would bring down the value of moment invariant of a bright region and would force the value of the moment invariant of a dark region to a higher value. This would make my earlier assumption of being able to throw away the regions that significantly darker and significantly brighter with respect to any given reference region, irrelevant. Also the normalization of the

moment invariant of such a region with a blank circular image would not carry any useful information and hence, can't be used to standardize the threshold. Therefore, the detection of regions that are significantly different in terms of summation of its intensity function now lies in the value of contrast and brightness terms determined using the aforementioned equations. The maximum value of the corrected region, with another region which is a correct match, would only increase the value of the region of interest, enough to make the maximum lie close to 1 and the minimum lie close to 0, the range of the normalized intensity function. Any corrected region with the value of significantly greater or lower than this could be removed from further analysis with respect to the region it's being matched with. This means setting a definite threshold to the maximum and minimum that a region attains. Eliminating such matches would also improve the time of execution, since presumably a bulk of incorrect matches would be eliminated, saving the computational burden of going through iterations of trying to eliminate such a match, without a standardized threshold for selection.

The second method of correcting the brightness and contrast also suffers from a drawback. The minimum and maximum values in a region could lie on a pixel that is resulting of noise. Using more than one value to define the maximum and minimum (average of a few pixels which lie close to the maximum value) would improve the estimate. The linear correction was experimented on the regions of interest in **Data Set B**, since the regions in **Data Set A**, don't differ by any contrast and brightness distortions.

Case Studies I

Brightness and Contrast Adjustment for Region of Interest 2 of dayton3.tif and Region of Interest 4 of dayton4.tif



Figure 42: Region of Interest 2 of Dayton 3



Figure 43: Region of Interest 4 of Dayton 4

The region of interest shown in **Fig. 42** and **Fig. 43** has been studied in the previous **Case Studies**. In that case, histogram matching was the technique to eliminate the differences in contrast and brightness. The problem has been revisited, in order to examine the results, if a linear distortion is assumed. There are definite advantages in making this assumption, since, the variation in the value of the moment, as was determined to be high, in the case of histogram matching, is not desirable since, it doesn't allow the setting of a global threshold for the detection of incorrect pairs.

The values of the similarity measure that was observed for two corresponding regions with linear contrast and brightness adjustment and with histogram specified contrast and brightness adjustment were determined.

On using the linear adjustment equations (**Equation 3.9** and **Equation 3.10**), the following were the result obtained for the regions in this case study (**Figure 44** and **Figure 45**). **Figure 45** shows the result obtained from the **Equation 3.11** and **Equation 3.12**

Moment (Template Region)	Adjusted Moment (Template Region)	Similarity Measure (before adjustment)	Similarity Measure (after adjustment)
0.4225	0.4559	5.9332	7.7169
0.4225	0.4216	5.9332	5.7863



Figure 44: Linearly adjusted contrast and brightness



Figure 45: Linearly adjusted contrast and brightness

The adjustment using this technique helps in bringing the value of the similarity measure above the threshold used to eliminate outliers in cases, where there is no contrast and brightness difference between the two images. But, the similarity is still not satisfactory and the result is not as high as was in the previous case. The image obtained after the adjustment satisfies the constraint for equality of the moment invariant values more closely, but all the values of the intensity function doesn't lie in the range $[0, 1]$. After clipping the values that lie outside this range, the similarity is diminished. The second technique for determination of the factors, has also failed in bringing the similarity measure value higher than the threshold. This also suggests that the nature of distortion present between the images is non-linear and linear correction doesn't yield satisfactory result.

Results

From the result obtained on using linear adjustment for the correction of brightness and contrast, the similarity measure developed for the two images yields a good result in the case where there was no brightness and contrast difference (shown in **Figure 46** and **Figure 47**), where almost all the outliers were eliminated and only one incorrect pair of control points was selected.

The threshold set for the elimination at each step was 0.0005, 0.0002, 0.0002, and 0.0002. **Figure 46** and **Figure 47** shows the result obtained on using **dayton1.tif** as the reference image and **dayton2.tif** as the template image and using linear contrast and brightness adjustment factors to improve the similarity measure.



Figure 46: Result of Similarity Measure on the image dayton2.tif with dayton1.tif



Figure 47: Result of Similarity Measure on the image dayton1.tif with linear contrast and brightness correction

The result shows the presence of a single outlier. The result obtained from histogram specification is comparable to the one shown here. The result however, is better here since it results in selection of several more correct pairs than the one obtained from histogram specification. The technique, however, would still need extension, due to the presence of outliers.

The same threshold was used for images **dayton2.tif** and **dayton3.tif** to determine the correct pair of matches. 20 outliers were detected out of a total of 37 pairs, which is not a good result when compared with the result obtained from histogram specification to reduce the effect of such differences. The two images suffer from slight brightness and contrast difference and hence the similarity would be higher if the brightness and contrast enhancement is done.

As was mentioned before, the computational complexity of using linear adjustment is quite high and is also a drawback when compared with the time of execution of the previous algorithm. It takes approximately 458 seconds to execute the similarity measure algorithm.

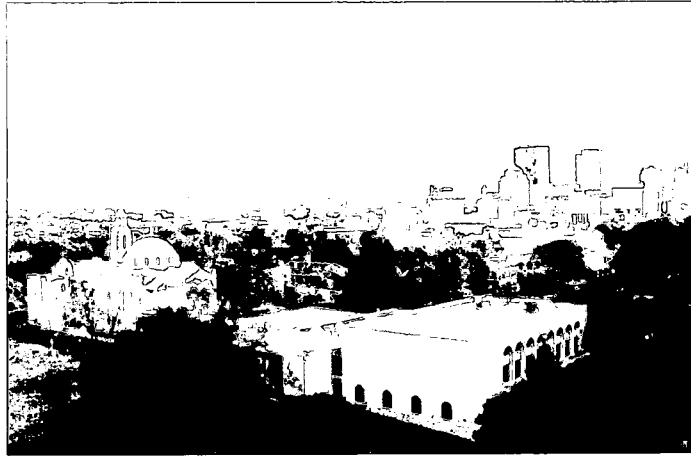


Figure 48: Result of Similarity Measure on the image dayton2.tif with dayton3.tif



Figure 49: Result of Similarity Measure on the image dayton3.tif with linear contrast and brightness

The images **dayton3.tif** and **dayton4.tif** suffer from significant brightness and contrast differences. The result obtained on automating the process with the linear adjustment is shown in **Figure 50** and **Figure 51** below.

The result shows the presence of 14 incorrect pairs out of a total of 17 pairings. This result is highly ambiguous and hence is not acceptable. The computational complexity of the algorithm remains quite high taking 275 seconds for executing the similarity measure algorithm. So, in cases where there are significant differences in the brightness and contrast and the moment invariant is sensitive to such changes, the given algorithm can't be used and hence technique for determination of a single set of linear contrast

and brightness terms using the two techniques mentioned in this chapter would not be considered any further in this research.

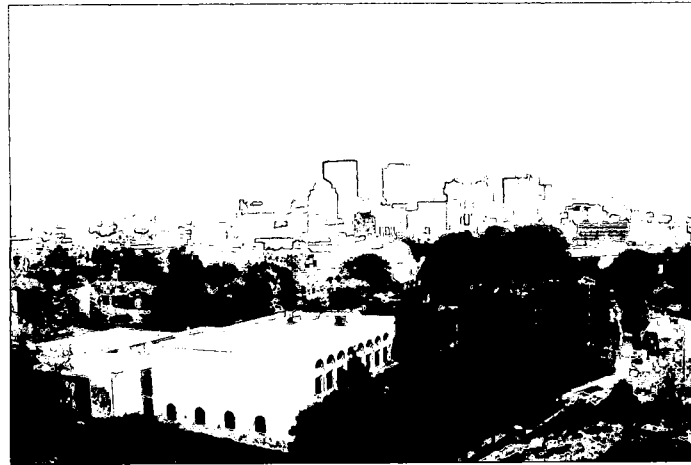


Figure 50: Result of Similarity Measure on the image dayton3.tif with dayton4.tif

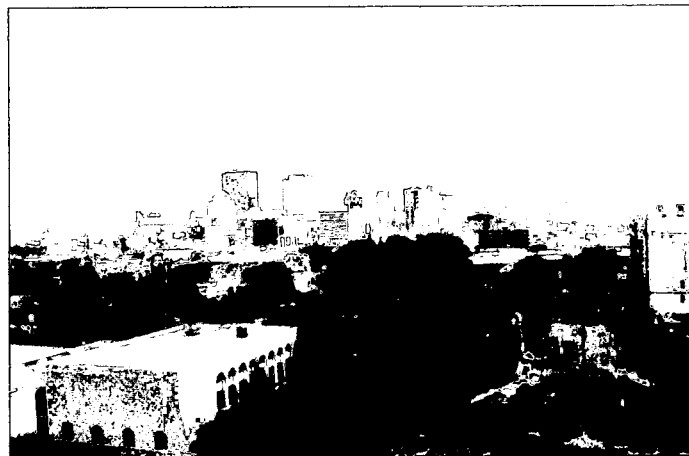


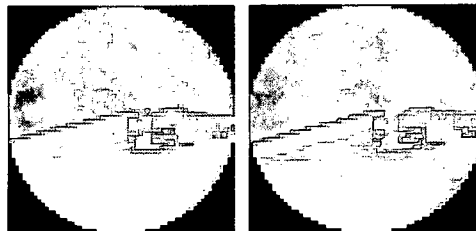
Figure 51: Result of Similarity Measure on the image dayton4.tif compensated for linear contrast and brightness

Another technique exploiting the isometry of rotational transformation was developed to determine the linear contrast and brightness adjustment factors for the any two regions. But this time, three factors were determined for three different regions, making it a piecewise linear adjustment method. The radius of the circular regions of interest was chosen to be 30 for the development of this algorithm. A circular ring of width one pixels is selected and it is assumed that the two regions of interest would consist of pixels of same intensity value. The average of all the pixels lying along this ring is

then taken and it is assumed that the two values differ from each other by linear factor of contrast and brightness. For a region of radius 10, 10 such rings are extracted and least squares method is used to determine the contrast and brightness factors. This procedure is followed for the region between the radius of 10-20 and the region between the radii of 20-30. Each of these regions is adjusted and joined together to attain the corrected form of the template image.

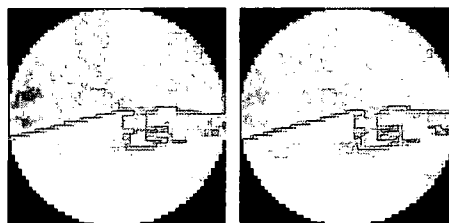
Case Studies II

Investigates the reason for the lower similarity determined using linear adjustment of contrast and brightness for regions of interest from **dayton3.tif** and **dayton4.tif**



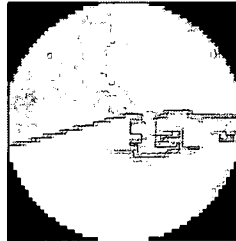
On linear adjustment of contrast and brightness, the following result was observed and the values of the moment invariant and the similarity measure tabulated.

The difference between the two moment values is not high enough for it to be selected even in the first iteration. The nature of the distortion, as was discussed previously,



is non-linear in nature and hence the technique just mentioned was used to determine the contrast and brightness factors for the three successive regions. Since, the images

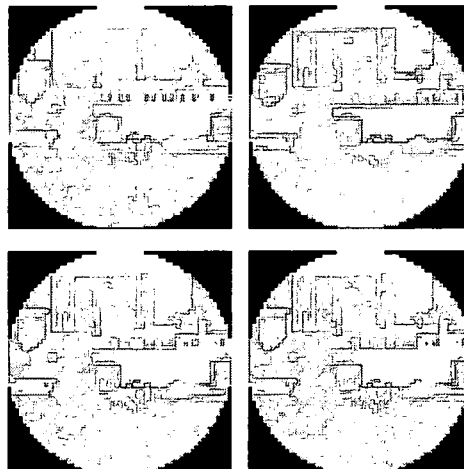
are digital in nature, the resolution of 10, yielding 10 equations to the least squares problem is the best that can be done for a region of size 61×61 .



Moment (Template Region)	Adjusted Moment (Template Region)	Similarity Measure (before adjustment)	Similarity Measure (after adjustment)
0.3579	0.4067	5.1461	7.1871
0.3579	0.4216	5.1461	9.3554

Clearly, there is an improvement in the result for the correct pairs obtained for the given pair of images using the least squares technique and it would also prove to be quite computationally efficient, if implemented carefully. If all the common data structures used for each pair of images, is constructed once outside the function and passed as an argument. But the technique suffers from one drawback and that is there is no simple way of incorporating the scaling transformation. The determination of the scaling factors would first need the images to be similar in terms of intensity values.

However, another data set was considered and it was determined if the result would suffice,



Moment (Template Region)	Adjusted Moment (Template Region)	Similarity Measure (before adjustment)	Similarity Measure (after adjustment)
0.38008	0.49278	3.770662	6.48444
0.38008	0.531730	3.770662	17.8798

The first column lists the result that was obtained using the linear correction technique just discussed. The second column lists the result obtained from histogram matching. The histogram matching technique has been observed to yield better results in all the cases discussed in this section and is the technique of choice for the purpose.

Performance Measurement

The following table lists the time of execution of the algorithm with linear adjustment in contrast and brightness using **Equation 3.9** and **Equation 3.10**. The linear adjustment in contrast and brightness of the image requires the computation of the reduced form of the moment invariant. The number of the multiplications for the computation of the moments for contrast and brightness is N^4 and $N^4 + N^2$ of additions.

Reference Image	Template Image	Number of CP's in Reference	Number of CP's in Template	Time of Execution (in seconds)
Dayton1.tif	dayton2.tif	8	50	16.00
Dayton2.tif	dayton3.tif	50	38	458.66
Dayton3.tif	dayton4.tif	38	27	274.01

Blur Variance of the Moment Invariant

Smoothing causes the value of affine moment invariant to vary for the same region as was inferred by the following experiment conducted for a region of interest. The image, that the region belongs to, is blurred by repeated convolution with a Gaussian convolution kernel of varying variance. At each subsequent stage, the same region is considered, normalized and the difference between the value of its affine moment invariant and that of the original region is observed.

Variance of Gaussian Convolution Kernel	Difference in Value of Blurred the Affine Moment Invariant Relative to the Original Region	Absolute Value of the Log of the Difference
1.0, [5, 5]	0.0131	4.3339
0.75, [5, 5]	0.0047	5.3695
0.50, [5, 5]	0.000802	7.1282
0.25, [5, 5]	0.0000000126	18.1914

As is evident from the values obtained the affine moment invariant is quite sensitive to blurring. The same region shows such high level of variations with increase in the variance of the Gaussian kernel (1.0, 0.75 and 0.5 respectively). Hence, the given moment invariant would not be able to identify the regions which are blurred form of the each other. The effect of smoothing on the result of the similarity measure would be significant. The original image suffering from this distortion can't be recovered unless the point spread function is known in advance.

Texture Description: Elimination Criteria

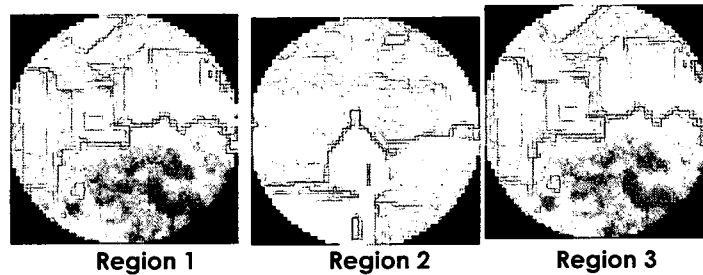
The rotation invariant texture description which was discussed previously would be used for the removal of the remaining outliers. This is because, it is understood that the moment invariant of the texture description, would not change under the effect of the rotational transformation. The only problem is that the range of the texture description image is quite low and hence needs to be scaled in order to yield results comparable to that obtained from the original images.

The same strategy is followed here, as was followed for the original gray level images. The values of the texture description of the image are scaled and the moment invariant determined for four overlapping circularly symmetric regions. The threshold used for elimination at each stage was determined to be 9.0, 9.0, 9.0 and 9.0 for obtaining good results for the data set in **Appendix A**. The following case studies show the moment invariant values that would be obtained for two corresponding regions using this

technique. The convolution kernel applied to the images is the same as that used for determining the joint moment similarity result.

Case Studies I

Two regions which show high similarity value for the moment invariant of all the segments in Region 1 and Region 2



The results were 9.8572, 10.3778, 10.6917, 12.0376, 12.6603, 10.9430, 10.3642, 12.1925 and 10.3946 for the segments of radii 0, 5, 7, 10, 13, 15, 17, 20 and 25 respectively. The result is higher than the threshold and indicates that the intensity composition of the overlapping regions is very similar to each other. This is a drawback in the technique, since, even for higher number of segments the differences in moments do not diminish at any level. In order to solve the problem, similarity determined from the texture description of the two regions was investigated.



Region 1	0.0425
Region 2	0.0578
Region 3	0.0419

Similarity between **Region 1** and **Region 2** is 8.3598 and the similarity between **Region 1** and **Region 3** is 14.8371. This result indicates that for any two regions, the similarity between two regions by taking the moment invariant of the rotation invariant texture description would add another dimension to it, based upon which two regions could be distinguished.

In order test for rotation invariance, an experiment was done to determine the similarity value on rotating the image by an angle of 20 degrees.



Since, rotation of the image involves interpolation, considerable smoothing of the image is observed. With the blur present in the original image, the moment value of the texture description of **Region 1** and that of the rotated image is expected to diminish and was observed to be 9.5105, which is higher than 8.3598 and above the global threshold. When the two corresponding regions were smoothed, the similarity diminished showing the moment invariant of is variant with respect to blurring of the image.

Results

The technique was used as a final step in determination of the outliers using the images shown in **Appendix A**. A threshold of 9.0, 9.0, 9.0 and 9.0 was used for this purpose. The symmetric convolution kernel described previously was used to form the images.



Figure 52: Result of similarity measure for dayton1.tif

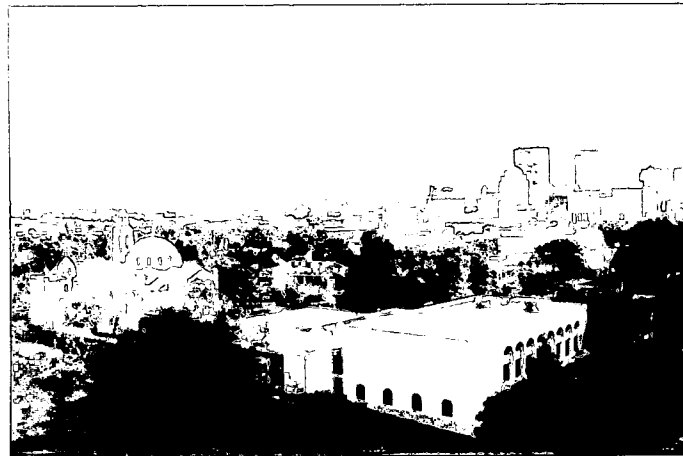


Figure 53: Result of similarity measure for dayton2.tif

The result shows the elimination of all outliers in the image. There are several correct pairs of regions, which have been eliminated and hence even though the technique has yielded the desired automation of registration of two images, it remains unreliable due to its inability to detect only the incorrect pair of regions.

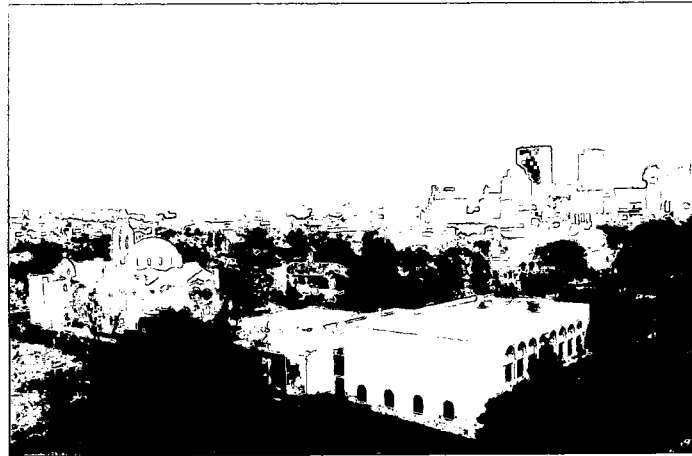


Figure 54: Result of similarity measure for dayton2.tif

The technique yields an outlier when the images **dayton2.tif** and **dayton3.tif** were registered. The outlier could be eliminated by varying the threshold or by considering a larger region of interest and increasing the number of iterations.

The technique was also tested with the final set of images, which suffer from significant differences in contrast and brightness. The results indicate that the effect of contrast and brightness differences can completely be eliminated by using the histogram matching technique.



Figure 55: Result of similarity measure for dayton3.tif

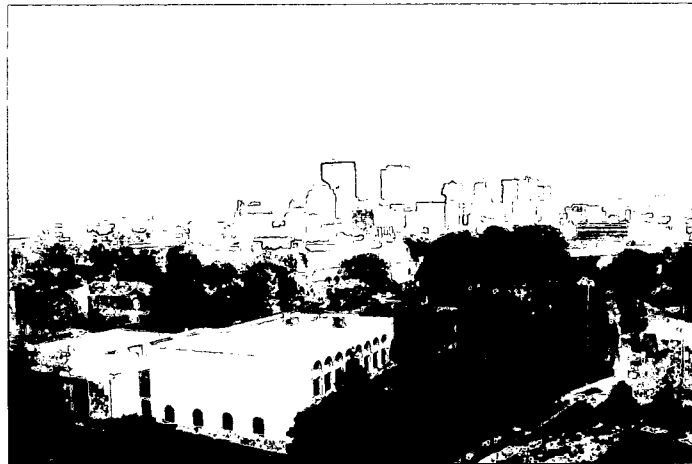


Figure 56: Result of similarity measure for dayton3.tif



Figure 57: Result of similarity measure for dayton4.tif

The result was also tested by rotating the image **dayton3.tif** by an angle of 20 degrees, and determining if the result varies. **Figure 58** and **Figure 59**, show the selected control points in the two images.



Figure 58: Control points selected after running the similarity measure algorithm on the image `dayton3.tif`

The results show that no outliers were detected among the control points selected from the two images. Many correct pairs of control points were eliminated. Once again, this behavior is due to the smoothing that occurs, when the image is rotated using the MATLAB `imrotate` routine.



Figure 59: Control points that are selected after the similarity measure algorithm on image `dayton3.tif` rotated by an angle of 20 degrees

But the results seem to vary with different images. The number of corresponding control points for the similarity measure from image `dayton1.tif` was determined to be none and for the images `dayton3.tif` and `dayton4.tif` two control points were selected, which is insufficient for the application of any spatial mapping function. The texture

description illustrated in this section is also sensitive to blurring. Therefore, the algorithm developed has a drawback of being highly sensitive to differences in amount of blur.

Reliability Dependence

The following section describes the reliability of using the similarity measure just developed with only affine moment invariant. The whole solution boils down to the analysis that was done previously, where the description of the moment in terms of the summation contribution at each pixel position of the image was done.

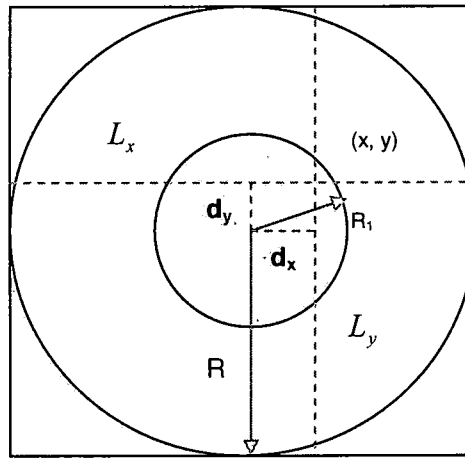
The goal of this section is to obtain, some measure of the changes that would occur to the moment invariant, when circular regions of varying radii are masked from the center of the image. When a circular region is masked from another circular region of a higher radius, a hollow disk shaped region is obtained. It is known that, at each pixel position in the image, the summation is computed by the following expression in the discrete space,

$$\begin{aligned}
 M &= \exp(-xy^T) \\
 K &= f(y)f^T(x) \\
 S_{x_p, y_p} &= \sum_i \sum_j M_{ij} K_{ij}
 \end{aligned} \tag{3.13}$$

Now, since the radius of the circular region being masked is known, the change in summation value at each pixel position can be expressed as a function of the distance in x -direction, distance in y -direction from the center and the radius of the inner circular region that is masked, keeping the outer circular radius at a fixed value. It would be important to note, that the matrix M , mentioned in the equations above, depends upon the centroid of the region. Since, the centroid of the region is a function of the intensity values present in the image, it is random in nature. Hence, this problem of determining the reliability of the technique, that was observed to have worked and

yielded desirable results, is an analytic problem and doesn't have a deterministic solution to it. The solution also demonstrates the symmetry about the center of each region.

Firstly, the positions are determined, where the pixel values are non-zero, as a function of the components of distance from the center and the inner radius of the circular region that was masked. The white areas in the image shown below represent the positions in the image where pixel values are present. The cyan colored areas represent the absence of pixels.



Case I: $d_x < R_1$, $d_y < R_1$

Suppose x_c and y_c are the centroid of the region. Then the positions on the columns and rows, where the pixels are non-zero and where the pixels are not present, are given by the following. The positions are the intersection of the lines L_x and L_y with the circular ring.

$$P_y(d_y, R_1, y_c) = \begin{bmatrix} 1 - y_c \\ \vdots \\ \left\{ \begin{array}{l} 1 - y_c - R + \sqrt{R^2 - d_x^2} \\ \vdots \\ 1 - y_c - R + \sqrt{R_1^2 - d_x^2} \end{array} \right. \\ \vdots \\ \left\{ \begin{array}{l} 1 - y_c - R - \sqrt{R_1^2 - d_x^2} \\ \vdots \\ 1 - y_c - R - \sqrt{R^2 - d_x^2} \end{array} \right. \\ \vdots \\ -y_c \end{bmatrix} \quad P_x(d_x, R_1, x_c) = \begin{bmatrix} 1 - x_c \\ \vdots \\ \left\{ \begin{array}{l} 1 - x_c - R + \sqrt{R^2 - d_y^2} \\ \vdots \\ 1 - x_c - R + \sqrt{R_1^2 - d_y^2} \end{array} \right. \\ \vdots \\ \left\{ \begin{array}{l} 1 - x_c - R - \sqrt{R_1^2 - d_y^2} \\ \vdots \\ 1 - x_c - R - \sqrt{R^2 - d_y^2} \end{array} \right. \\ \vdots \\ -x_c \end{bmatrix}$$

Here, the bracketed positions are the positions where the pixels have intensity corresponding to the image intensity and the resolution used in this thesis was $\frac{1}{60}$. The outer product yields a matrix that has symmetrically distributed rectangular regions, which are summed up weighted by the outer product of intensity values along x -column and y -column at that position.

The change in the moment values at each pixel is represented by determining the summation over the rectangular regions. The positions have been defined below. The difference at each pixel is given by the following expression.

$$I_{diff}(x_p, y_p) = \left(\int_{p_{y1}}^{p_{y4}} \int_{p_{x1}}^{p_{x4}} f_y(x) f_x(y) e^{-xy} dx dy + \int_{p_{y2}}^{p_{y3}} \int_{p_{x1}}^{p_{x4}} f_y(x) f_x(y) e^{-xy} dx dy - \int_{p_{y2}}^{p_{y3}} \int_{p_{x2}}^{p_{x3}} f_y(x) f_x(y) e^{-xy} dx dy \right) \times e^x \quad (3.14)$$

Where, (x_p, y_p) is the current pixel position

$$\begin{aligned}
p_{x1} &= 1 - \frac{m_{10}}{m_{00}} - R + \sqrt{R^2 - d_y^2} & p_{y1} &= 1 - \frac{m_{01}}{m_{00}} - R + \sqrt{R^2 - d_x^2} \\
p_{x2} &= 1 - \frac{m_{10}}{m_{00}} - R + \sqrt{R_1^2 - d_y^2} & p_{y2} &= 1 - \frac{m_{01}}{m_{00}} - R + \sqrt{R_1^2 - d_x^2} \\
p_{x3} &= 1 - \frac{m_{10}}{m_{00}} - R - \sqrt{R_1^2 - d_y^2} & p_{y3} &= 1 - \frac{m_{01}}{m_{00}} - R - \sqrt{R_1^2 - d_x^2} \\
p_{x4} &= 1 - \frac{m_{10}}{m_{00}} - R - \sqrt{R^2 - d_y^2} & p_{y4} &= 1 - \frac{m_{01}}{m_{00}} - R - \sqrt{R^2 - d_x^2}
\end{aligned}$$

The overall difference in the summation values for the moments for the region of the image that satisfies the condition

$$D = \iint I_{diff}(x, y) dx dy \quad (3.15)$$

For any two regions, that do not correspond with each other, but yields a high similarity value on application of the moment invariant over that region, would depend on the difference of the these difference values, to be higher than 0 and preferably, higher than the difference of the difference value of the corresponding match of the region, in the template image, if it exists. The technique depends upon the assumption that the similarity values of the true matches would show consistently high results and would not be eliminated when the circular regions from the center of the image are masked.

Case 2: $d_x < R_1$, $d_y > R_1$

This is the case, where, no differences in the moment value would occur along the y-columns. The differences in the x-direction would remain the same and is given by

$$I_{diff}(x_p, y_p) = \left(\int_{p_{y1}}^{p_{y4}} \int_{p_{x2}}^{p_{x3}} f_{y_p}(x) f_{x_p}(y) e^{-xy} dx dy \right) \times e^{x_p y_p} \quad (3.14)$$

and once again, the difference is given by the same expression in **Equation 3.15**.

Case 3: $d_y < R_1$, $d_x > R_1$

This is the case, where, no differences in the moment value would occur along the y -directions. The differences in the y -direction would remain the same and is given by

$$I_{diff}(x_p, y_p) = \left(\int_{p_{y1}}^{p_{y4}} \int_{p_{x2}}^{p_{x3}} f_{y_p}(x) f_{x_p}(y) e^{-xy} dx dy \right) \times e^{x_p y_p} \quad (3.16)$$

and once again, the difference is given by the same expression as in **Equation 3.15**.

Now, since the matrix M varies very slowly (range of $[0.7, 1.3]$), it can be thought of as a matrix of unity values. Then the overall summation of the outer product of the intensity pixels in the x -column and the y -column at each pixel, gives information about the variation of the 1-D intensity vector along the x -column, with respect to that along the y -column along with the total gray level composition of the two vectors. Overall summation of this matrix consists of this information over the entire image. So, the difference Equation represents the loss of this information over the entire region lying between the upper and lower limits of the position along the x -column and y -column for a given pixel, and is given by p_{x2} and p_{x3} , p_{y2} and p_{y3} . The original summation is more complicated since the matrix M varies with change in centroid of the region, which is arbitrary.

Therefore, if the centroid of the image is thought to be static, the technique just represents the change in this quantity (the difference function), with the increase in the

radius of the masked region for two dissimilar regions showing similarity. It is a reasonable assumption that these variations over a pair of dissimilar region would change in the two images. This is because for any two distinct regions showing similarity, the reason behind the similarity is that the variations in the intensity values along x -column and y -column of the image at that pixel position would sum to be the same for the entire image. But, when segmented into different regions would not vary the same way as the moment summation of the original image, since, the variations are specific to the position of the pixel relative to the center.

Hence, the first observation is that larger the area masked from the center of the image, the more likely it is, that the incorrect pair would get eliminated. Secondly, there is not much advantage in adding several iterations to eliminate the incorrect pairs, since the variations in the intensity values are not likely to change dramatically over an increase in radius of 1-2 pixels, for any two images. An iterative step between the radius of 15 and 25 could be inserted, in order to increase the probability of elimination. Alternatively, a larger circular region could be considered in the first place to engulf sufficiently higher amount of variations to change the composition of the standardized invariant for the two distinct regions.

This observation also indicates one drawback in using the texture description as the elimination criterion. Since, the range of the image obtained is quite low, there aren't much variations or change in composition of the image in different regions. Better results are obtained by scaling the range using either the histogram specification technique or by increasing it linearly.

Uniqueness Measure

As was mentioned in **Chapter 3**, the similar points that lie within the same image needs to be eliminated in order for the algorithm to determine point correspondence

without ambiguities. The reduction in the number of candidate pairs reduces the time of execution of the algorithm. A technique for doing so would be to accomplish this task by using the similarity measure. The difference is that the uniqueness measure would eliminate regions based upon high similarity instead of dissimilarity. Since, the regions under the matching procedure belong to the same image, the adjustment for contrast and brightness could be skipped.

Results

The similarity measure developed in the previous section was used for this algorithm to accomplish this task. The purpose of doing this is to eliminate only truly incorrect pair of regions the presence of which would only create ambiguities and also in order to reduce the computational load on the algorithm. Hence, a threshold is set, which would not eliminate any region which is distinct. A threshold of 9.6, 9.6, 9.6 and 9.6 was used for the purpose of documenting the result obtained from the similarity measure in this thesis. Any threshold, which doesn't allow the elimination of the distinct points in the image, could be used.

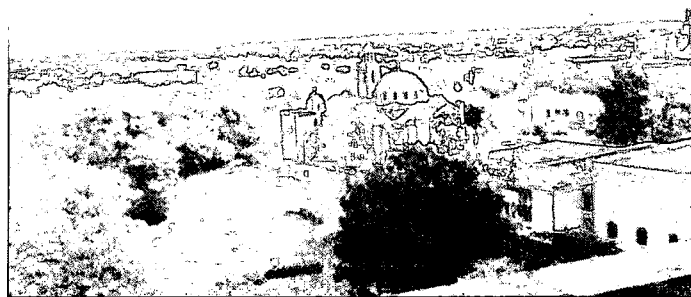


Figure 60: Corner detection with uniqueness measure for dayton1.tif

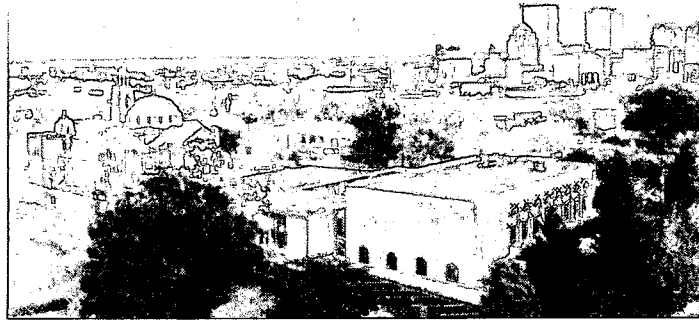


Figure 61: Corner detection with uniqueness measure for dayton2.tif

Performance Measurement

The following table lists the time of execution of the uniqueness measure algorithm. For the computation of the moments of region of interests in both the images $4 * N^4$ multiplications are required and $N^4 + N^2$ additions are required. The following are the reduction in the number of control points on using the uniqueness measure algorithm and the time of execution of the algorithm. The moments for the reference region derived here could be used at a later stage for similarity measure, thereby reducing its computational complexity.

Image Name	Reduction in CP's	Time of Execution (in seconds)
dayton1.tif	0	1.736095
dayton2.tif	2	8.508506
dayton3.tif	11	9.948987
dayton4.tif	7	5.449158

This is the purpose of the uniqueness measure, to eliminate the truly similar matches within the same image. This is the reason behind the conservative threshold. The time of execution of the algorithms of 1.736095 s and 8.508506 s is not too high.

Chapter 4

Consistency Checking

Spatial transformation on images can successfully be described in terms of affine transformations. Affine transformations commonly consist of translation, rotation, scaling (stretching or shrinking) and shear. The affine transformations differ from linear transformations by a translation and are therefore represented by the following equation

$$\bar{Y} = A \bar{X} + \bar{V} \quad (4.1)$$

Which can written in homogenous coordinates as,

$$\begin{bmatrix} x' \\ y' \\ 1 \end{bmatrix} = \begin{bmatrix} a_{11} & a_{12} & t_x \\ a_{21} & a_{22} & t_y \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} \quad (4.2)$$

Where A is a 3x3 non-singular matrix and \bar{Y} and \bar{X} are output and input vectors respectively and \bar{V} is the translation vector.

The origin for affine transformation frame of reference is not necessarily known, and transformations can be considered as linear transformations with the origin translated by a vector \bar{V} . The transformation on images can be approximated by an affine transformation. An affine transformation maps triangles onto triangles and rectangles into parallelograms [1]. Affine transformations can completely be determined using three non-collinear points, which would yield the required six equations to solve for six unknowns as shown in **Equation 4.2** above in 2-D space.

Even though affine transformations are restrictive, they are widely used to represent image transformations in the field of image processing. In the field of image mosaicking, simple transformations can be very well approximated as a combination of the aforementioned transformations. In order to register a pair of images, affine transformation requires the presence of at least three corresponding control points from the given images.

In this thesis, a technique has been proposed, which can determine point correspondence between the control points from each two-dimensional image, assuming an affine transformation of translation, scaling and rotation between the images. These control points are then given to the already existing functions to determine the transformation matrix A . These points are determined from feature extraction using feature detectors that are isotropic and invariant to scale.

Affine Transformations and Convexity

As mentioned above, after feature detection, the points obtained, are randomly placed and need to be in some sort of order, in order to determine point correspondence. Ordering of a set of points in a two dimensional space is important in context of applying an applicable set of equations and determining correspondences between two or more different set of points with the same ordering principle. This information can be used as the first step in determining the correspondence. A set of points are always defined with respect to a frame of reference. After establishing an appropriate frame of reference, the mutual relation between the given set of points (for example, distance measure and angles) can be used to provide specific ordering to it.

Affine transformation and convexity are very closely related topics in computational geometry. Affine transformations preserve convexity of a set. This theorem can be used to provide the set of points that is being considered, a specific ordering.

In my case, I am considering two sets of points in the 2-D standard Cartesian co-ordinate system. The solution to the problem lies in determining, for each point in one data set, a corresponding point in the transformed data set. The assumption is that, the two sets of points are mathematically related to each other, by a transformation that can be approximated by a single translation, rotation and scale in any combination.

Translations and rotations are Euclidean transformations and therefore are isometric (i.e. preserve the distance and angles measure). The shape of the geometric object would not change under these two transformations [13]. Scaling, on the other hand, shrinks or stretches the co-ordinate axis by a specified factor [13], which changes the length and angle measure and also changes the shape of the geometric object, if the scaling factor along the x -direction and the y -direction of the co-ordinate system, are different. Therefore, scaling transformation is not isometric but is an affine transformation and hence, a similarity transformation.

One technique that has widely been used, in the field of image processing and pattern recognition problems, is the concept of convex hull of a set of points S in 2-D space. Convex hull is the smallest convex set containing S . It is the smallest convex polygon that contains all the points in the set. Consecutive vertices of this polygon occur in sorted angular order about any interior point [11]. Therefore, it provides a definite ordering to an otherwise random set of points. Several algorithms for the determination of the convex hull are found in the theory of the computational geometry. These algorithms have different computational complexities and can vary from the worst of $O(N^4)$ to the best of $O(N \log N)$.

However, before considering the computational complexity requirements, one major consideration in use of convex hull algorithm in ordering of the point set, is its invariance with respect to the spatial transformations of translation, rotations and scaling. By invariance it is meant, that the set of points that define the vertices of a convex hull, do

not change when the set undergoes one or any combination of the spatial transformations mentioned above. It can be investigated by first considering one of the basic algorithms developed for the determination of the convex hull of a set of points, known as the Graham's scan, and then using the affine transformation theorems to determine, if the ordering of the vertices of the convex hull, would be changed under these transformations. **Figure 62** shows the Graham's scan showing an internal point of the hull p_1 and two vertices p_2 and p_3 of the hull.

The Graham's scan starts at the point **START**, which may be taken as the rightmost smallest ordinate point of the given set, which is definitely a vertex. The algorithm proceeds in counter-clockwise direction considering triples of consecutive points in order to determine whether or not they define a reflex angle (*i.e* $\geq \pi$). The angle $p_2p_1p_3$ is measured at each consecutive triples and the vertex p_1 rejected if the angle is found to be reflex and the scan advances only when a convex triple is found. This is due to the fact, that by definition when a convex hull is traversed, angle measured at each consecutive triple is convex. The scan terminates when it traverses all the way back to the point **START**, after selecting an ordered set of vertices.

For the transformed set of points, an arbitrary condition can be considered where the point p_1 is displaced relative to points p_2 and p_3 , such that the angle $p_2p_1p_3$ becomes convex. The Graham's scan for the determination of convex hull vertices would now include the point p_1 , among other ordered vertices. This is not desirable, as the convex hull of the transformed set contains points that are not present in the convex hull of the original set. This ordering of points would not be useful, since there is no way to detect these undesirable points and they would prevent the detection of the underlying correspondence, since, the equations describing these transformations, would not yield the correct result for the incorrect ordering of points in the two sets (convex hull sets). There is, therefore, a need to determine, if there is any probability of this scenario, if the

convex hull algorithm is to be used to order the point set for solving the problem of point correspondence. For the purpose of illustration, the **Figure 62** and **Figure 63** have been shown below [11].

To Prove:

Under an affine transformation the convex hull of a set of points in E^2 remains invariant.

Proof:

If T is an affine transformation of R^d onto itself, and if $A \subset R^d$ is convex, then $T(A)$ is convex [4]. Rotation and translation are isomeric transformations and therefore preserve the property of betweenness and non-betweenness. Hence, the non-extreme points of a convex hull maps to non-extreme points and extreme points map to extreme points. An affine mapping also preserves set inclusion property. $CH(A) \subset R^2$, in two-dimensional space is a smallest convex set that contains A . Hence, the convex sets of R^2 , which contain A , map to convex sets of R^2 which contain $T(A)$.

Hence, $T(CH(A)) = CH(T(A))$, which proves the invariance in case of isometric transformations [12]

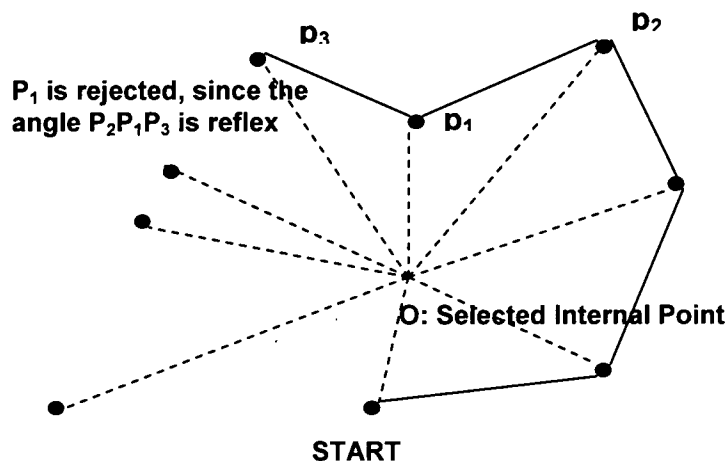


Figure 62: Demonstrates the Graham's scan algorithm for convex hull on original set of points

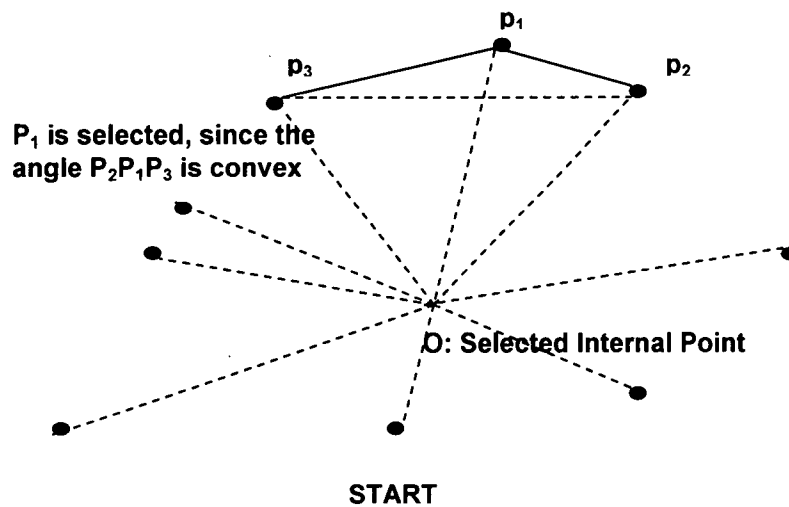


Figure 63: Demonstrates the Graham's scan algorithm for convex hull on transformed set of points

The transformation of scaling is however, not an isometry, but a similarity transformation. This means that it preserves the ratio of parallel lines and not the distance measure. The angles are not preserved in a similarity transformation. Scaling is dilation, and dilations are affine functions (Theorem 4, Sec. 5.2 of [1]). Therefore, scaling transformation preserves convexity.

This information is useful in ordering the two set of points. The ordering of the points in each set has been described in the next section.

Ordering the Point Set

The points in the two sets can be ordered by constructing a convex hull of the points using an efficient convex hull algorithm. The internal points of the convex hull can also be ordered by constructing successive convex hulls (called convex layers) until less than three points are left in the two sets, as shown in **Figure 64** below. The correspondence of each layer is determined separately. This method is used in the computational geometry for determination of the depth of each point in the set [10]. If only one point is left, the correspondence is established, but if two points are left then, at least one point can be extracted from the previous set (convex layer) and added to the two points left to

determine an additional layer. Therefore, correspondence for each point in the sets can be determined.

Determining Applicable Equations

The next step is the determination of the equations to use for the correspondence. The method used for this purpose is simply vectorial with three co-ordinate axes. The first one is the frame of reference about which all the points are defined and is assumed to be the center of the image, with origin at the bottom left corner.

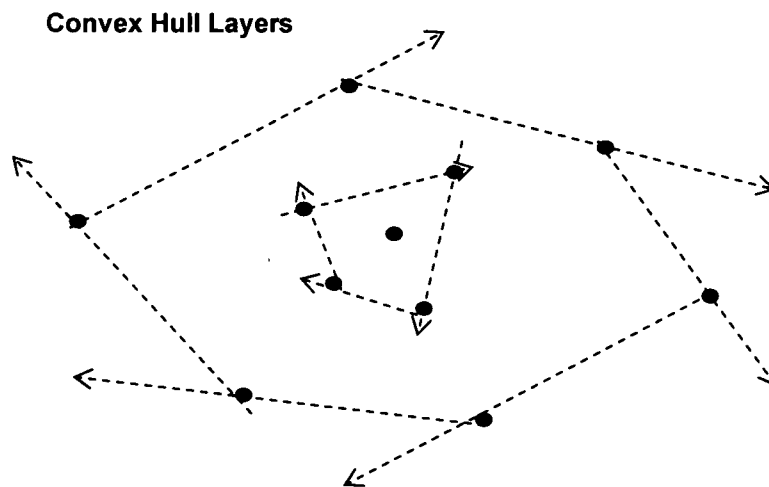


Figure 64: Demonstrates the construction of convex hull layers of a point set

The origin of the second co-ordinate axis lies at a translation of S and defines the origin about which the points are rotated. The origin of the third co-ordinate axis lies at a translation of S' and defines the origin of scale. The goal is to determine a common value of the angle of rotation, about the specified origin of rotation. The ordered set of points can then be used to find point correspondence using an iterative technique.

In affine transformation, the order in which the transformations are applied is very important and if a technique that approximates these transformations, must consider the assumption of any order, in which these transformations are applied. Geometric

techniques of solving the problem are most efficient in terms of computational complexity as well as the detection of the correct matches, since geometric techniques use the theorem's of mathematics as a basis and are therefore, less likely to fail than the other techniques that make learned assumptions about the nature of intensity values in the images, which are subject to several distortions, nature of which, depends upon the hardware, noise and illumination conditions, present during acquisition of the image. Therefore, in this thesis, I have tried to use a hand solved technique to determine the angle of rotation and then analyze the variance of this angle over several iterations. The iteration during which minimum variations in the value of this quantity is observed, represents the correct ordering of the control points in the two sets of convex hull of points. This strategy is rather simple and can be used as a consistency check to determine whether the outliers have been correctly rejected and may be used to identify the presence of such outliers. But this technique apparently cannot be used for outlier detection. For that, more advanced geometric techniques (graph theoretic techniques etc) or a determination of nature of the local variations of intensity values which are invariant under such transformations, can be used to distinguish between dissimilar selected regions, as has been demonstrated in the later part of the thesis. The following is the explanation of the technique that would be used later as a consistency check, to determine the correct matches and supply the corresponding set of points for registration and determination of the transformation matrix.

Case 1:

The diagram shown above, demonstrates the scaling transformation before rotation and translation. The following equation goes along with the **Figure 65** shown above,

$$\bar{S} = \begin{bmatrix} s_x & 0 & 0 \\ 0 & s_y & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \cos(\theta) & \sin(\theta) & 0 \\ -\sin(\theta) & \cos(\theta) & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & t_x \\ 0 & 1 & t_y \\ 0 & 0 & 1 \end{bmatrix} \quad (4.3)$$

Where, the juxtaposition of the matrices represents the product of the three matrices. Changing the order would obviously change the resulting position of the destination pixel.

The following shows the hand solved result obtained for determining the angle of rotation for the case shown below in **Figure 65**.

Considering the axes of scaling to be X' and Y' , the transformation that scales the vector whose position represents the position of the pixel in the reference image. The transformed vector is just the vector with its x and y components scaled by a constant factor.

$$\begin{aligned}\vec{r}_{s2x} &= s_x \vec{r}_{s1x} \\ \vec{r}_{s2y} &= s_y \vec{r}_{s1y}\end{aligned}\tag{4.4}$$

where the following changes take place in the vector that represents the original vector when the scaling transformation transforms the original pixel to an intermediate scaled position.

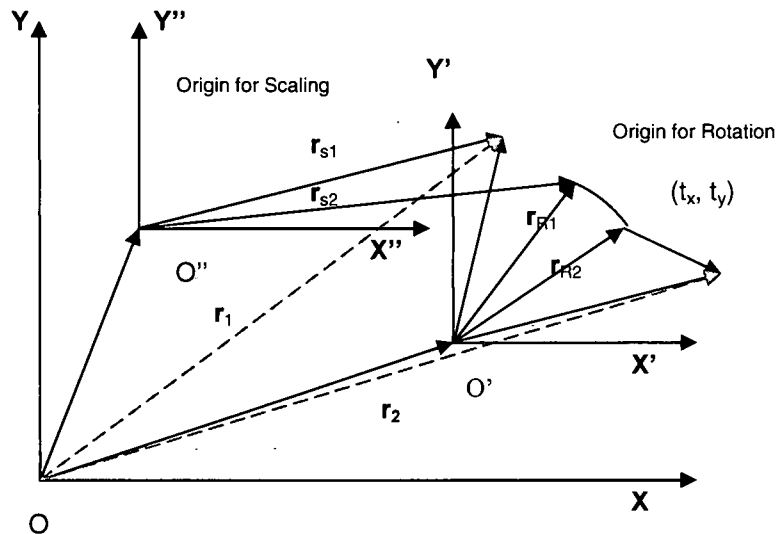


Figure 65: Vectorial technique demonstrating scaling before rotation and translation

Let \vec{v}_s represent the vector that represents the position of the origin of scale. The original vector \vec{v}_1 can be written in terms of the vector to the original position from the origin of scale \vec{r}_{s1} , represented as,

$$\vec{r}_1 = \vec{v}_s + \vec{r}_{s1} \quad (4.5)$$

$$\vec{r}_{s1} = \vec{r}_1 - \vec{v}_s$$

$$\vec{r}_{s1} = \vec{r}_{s1x} \hat{i} + \vec{r}_{s1y} \hat{j} \quad (4.6)$$

After transformation, the vectors become,

$$\vec{r}_{s2} = s_x \vec{r}_{s1x} \hat{i} + s_y \vec{r}_{s1y} \hat{j} \quad (4.7)$$

$$\vec{r}_{is1} = s_x [\vec{r}_1 - \vec{v}_s]_x \hat{i} + s_y [\vec{r}_1 - \vec{v}_s]_y \hat{j} + \vec{v}_s$$

$$\vec{r}_{is1} = s_x [x - v_{sx}] \hat{i} + s_y [y - v_{sy}] \hat{j} \quad (4.8)$$

Since, the next step is to determine the intermediate pixel position after rotation, it is necessary to express each vector in terms of the vector defined in the frame of reference for rotation. Defining the vector to the origin of the rotational frame of reference to be \vec{v}_R ,

$$\vec{r}_{R1} = \vec{r}_{is1} - \vec{v}_R \quad (4.9)$$

After the transformation,

$$\vec{r}_{R2x} = \vec{r}_{R1x} \cos(\theta) - \vec{r}_{R1y} \sin(\theta)$$

$$\vec{r}_{R2y} = \vec{r}_{R1x} \sin(\theta) + \vec{r}_{R1y} \cos(\theta)$$

$$\vec{r}_{ir1} = (\vec{r}_{R1x} \cos(\theta) - \vec{r}_{R1y} \sin(\theta) + \vec{v}_{Rx}) \hat{i} + (\vec{r}_{R1x} \sin(\theta) + \vec{r}_{R1y} \cos(\theta) + \vec{v}_{Ry}) \hat{j} \quad (4.10)$$

Substituting the value of the vector \vec{r}_{R1} in terms of the x -component and y -component of \vec{r}_{is1} ,

$$r_{R1x} = s_x [x - v_{sx}] - v_{Rx}$$

$$r_{R1y} = s_y [y - v_{sy}] - v_{Ry}$$

$$\vec{r}_{ir1} = [(s_x [x - v_{sx}] - v_{Rx}) \cos(\theta) + (s_y [y - v_{sy}] - v_{Ry}) \sin(\theta)] \hat{i} + [(-(s_x [x - v_{sx}] - v_{Rx})) \sin(\theta) + (s_y [y - v_{sy}] - v_{Ry}) \cos(\theta)] \hat{j} \quad (4.11)$$

Since, translation doesn't depend upon any defined co-ordinate system,

$$\begin{aligned} r_{2x} &= [(s_x [x - v_{sx}] - v_{Rx}) \cos(\theta) + (s_y [y - v_{sy}] - v_{Ry}) \sin(\theta)] + v_{Rx} + t_x \\ r_{2y} &= [(-(s_x [x - v_{sx}] - v_{Rx})) \sin(\theta) + (s_y [y - v_{sy}] - v_{Ry}) \cos(\theta)] + v_{Ry} + t_y \end{aligned}$$

Now consider two points and the x -component and y -component of \vec{r}_2 for each point is determined by Equation listed above. The difference in x -component and y -component is the translation, which remains the same for each point being considered from the set,

$$\begin{aligned} t_x &= x_f - [(s_x [x - v_{sx}] - v_{Rx}) \cos(\theta) + (s_y [y - v_{sy}] - v_{Ry}) \sin(\theta) + v_{Rx}] \\ t_y &= y_f - [(-(s_x [x - v_{sx}] - v_{Rx})) \sin(\theta) + (s_y [y - v_{sy}] - v_{Ry}) \cos(\theta) + v_{Ry}] \end{aligned} \quad (4.12)$$

Considering a set of three points from the set,

$$\begin{aligned} p_1 &= (x_{1i}, y_{1i}), (x_{1f}, y_{1f}) \\ p_2 &= (x_{2i}, y_{2i}), (x_{2f}, y_{2f}) \\ p_3 &= (x_{3i}, y_{3i}), (x_{3f}, y_{3f}) \end{aligned}$$

After some algebra, the following result is obtained,

The determination of the angle can be carried out by the determination of the roots of the quadratic equation,

$$a \tan(\theta)^2 + b \tan(\theta) + c = 0,$$

where a , b and c have been defined as follows,

$$\begin{aligned} R_{11} &= \frac{x_{1i} - x_{2i}}{y_{1i} - y_{2i}} \quad R_{12} = \frac{x_{1f} - x_{2f}}{y_{1f} - y_{2f}} \quad R_{21} = \frac{x_{2i} - x_{3i}}{y_{2i} - y_{3i}} \quad R_{22} = \frac{x_{2f} - x_{3f}}{y_{2f} - y_{3f}} \\ a &= R_{12} R_{11} - R_{21} R_{22} \\ b &= (R_{11} - R_{21})(1 - R_{12} R_{22}) \\ c &= R_{12} R_{21} - R_{22} R_{11} \end{aligned}$$

Minimum of three points are required for the determination of the transformation matrix under affine transformation for registration purpose. More than three control points would be needed to make the aforementioned solution work for the determination of the angle of rotation. The quadratic equation would yield two solutions one of which is incorrect and that one can be detected by comparing successive values of rotation obtained by applying the equation mentioned above, and constraining with the condition that the angle of rotation would not change from iteration to iteration.

Case 2:

The **Figure 66** shown below, determines θ , with an assumption of a different ordering of the transformation matrix as shown in **Equation (4.13)** below

$$\bar{S} = \begin{bmatrix} s_x & 0 & 0 \\ 0 & s_y & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & t_x \\ 0 & 1 & t_y \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \cos(\theta) & -\sin(\theta) & 0 \\ \sin(\theta) & \cos(\theta) & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad (4.13)$$

Then,

$$\begin{aligned} \bar{r}_{is1} &= s_x [\bar{r}_1 - \bar{v}_s]_x \hat{i} + s_y [\bar{r}_1 - \bar{v}_s]_y \hat{j} + \bar{v}_s \\ \bar{r}_{is1} &= (s_x [x - \bar{v}_{sx}] + v_{sx}) \hat{i} + (s_y [y - \bar{v}_{sy}] + v_{sy}) \hat{j} \end{aligned} \quad (4.14)$$

$$\begin{aligned} \bar{r}_{R2} &= (x_f - v_{Rx}) \hat{i} + (y_f - v_{Ry}) \hat{j} \\ \bar{r}_{R1} &= [(x_f - v_{Rx}) \cos(\theta) + (y_f - v_{Ry}) \sin(\theta)] \hat{i} + [-(x_f - v_{Rx}) \sin(\theta) + (y_f - v_{Ry}) \cos(\theta)] \hat{j} \\ \bar{r}_{ir1} &= [(x_f - v_{Rx}) \cos(\theta) + (y_f - v_{Ry}) \sin(\theta) + v_{Rx}] \hat{i} + [-(x_f - v_{Rx}) \sin(\theta) + (y_f - v_{Ry}) \cos(\theta) + v_{Ry}] \hat{j} \end{aligned} \quad (4.15)$$

are the equations representing the appropriate relations

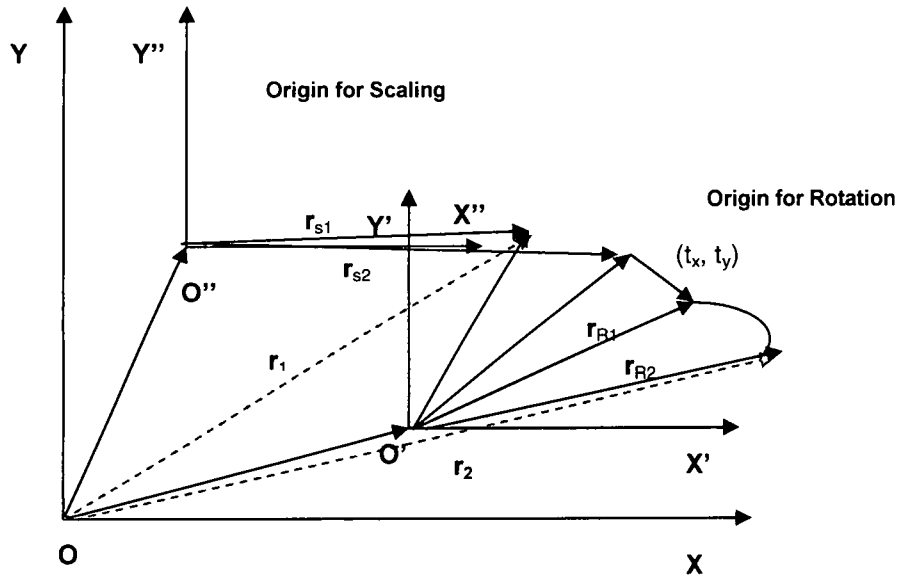


Figure 66: Vectorial technique demonstrating scaling before translation and rotation

Since, translation is assumed to be before rotation, considering the inverse transformation for the destination pixel position to the an intermediate position, the following equation is obtained,

Now,

$$\vec{r}_{ir1} - \vec{r}_{is1} = \vec{d} = t_x \hat{i} + t_y \hat{j}$$

Where the expressions for \vec{r}_{ir1} and \vec{r}_{is1} can be substituted from **Equations 4.14** and **Equation 4.15**, yielding equations for translations in both x and y directions. On doing so, it was determined that all the vectors that are common to all the vectors in the given set of points would be eliminated and a simple expression for rotation is observed.

$$R_{11} = \frac{x_{1i} - x_{2i}}{y_{1i} - y_{2i}} \quad R_{12} = \frac{x_{1f} - x_{2f}}{y_{1f} - y_{2f}} \quad R_{21} = \frac{x_{2i} - x_{3i}}{y_{2i} - y_{3i}} \quad R_{22} = \frac{x_{2f} - x_{3f}}{y_{2f} - y_{3f}}$$

$$a = R_{12}R_{11} - R_{21}R_{22}$$

$$b = (R_{21} - R_{11})(1 - R_{12}R_{22})$$

$$c = R_{22}R_{11} - R_{12}R_{21}$$

The following trigonometric identity was used in the solution,

$$\tan(\theta + \eta) = \frac{\tan(\theta) + \tan(\eta)}{1 - \tan(\theta) \tan(\eta)}$$

Case 3:

The **Figure 67** shown below determines θ , with an assumption of a different ordering of the transformation matrix as shown in **Equation 4.16** once again,

$$\bar{S} = \begin{bmatrix} 1 & 0 & t_x \\ 0 & 1 & t_y \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} s_x & 0 & 0 \\ 0 & s_y & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \cos(\theta) & -\sin(\theta) & 0 \\ \sin(\theta) & \cos(\theta) & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad (4.16)$$

The solution is determined in the same way as determined in Cases 1 and 2 illustrated above. Since, translation is assumed to be before scaling and rotation, the entire solution would be in terms of inverse relation, from destination pixel to source pixel, for rotation and scale. On determining the translation vector, the following result was observed,

$$t_x = x - \frac{1}{s_x} \left[(x_f - v_{Rx}) \cos(\theta) + (y_f - v_{Ry}) \sin(\theta) + (v_{Rx} - v_{Sx}) \right] + v_{Sx} \quad (4.17)$$

$$t_y = y - \frac{1}{s_y} \left[-(x_f - v_{Rx}) \sin(\theta) + (y_f - v_{Ry}) \cos(\theta) + (v_{Ry} - v_{Sy}) \right] + v_{Sy}$$

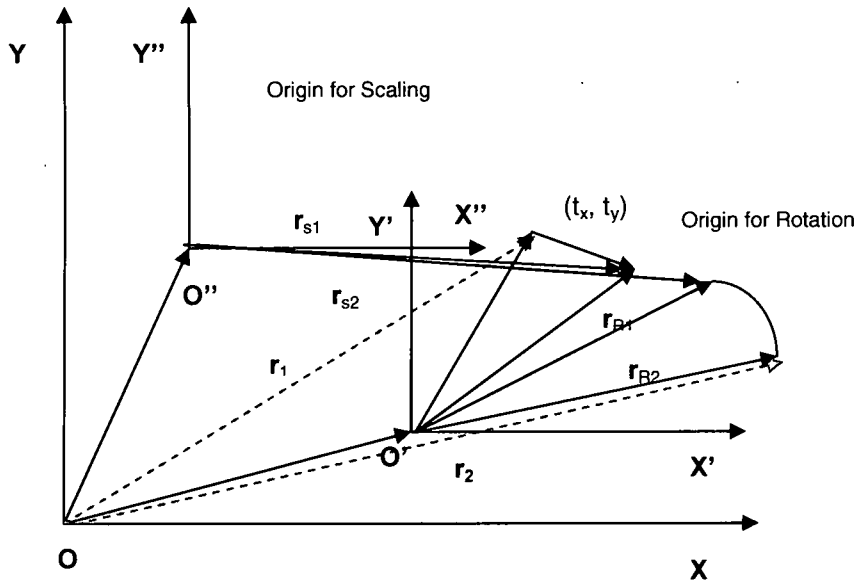


Figure 67: Vectorial technique demonstrating translation, scaling and rotation in that order

Once again, considering three distinct non-collinear points to solve for θ , the following result was observed,

$$R_{11} = \frac{x_{1i} - x_{2i}}{y_{1i} - y_{2i}} \quad R_{12} = \frac{x_{1f} - x_{2f}}{y_{1f} - y_{2f}} \quad R_{21} = \frac{x_{2i} - x_{3i}}{y_{2i} - y_{3i}} \quad R_{22} = \frac{x_{2f} - x_{3f}}{y_{2f} - y_{3f}}$$

$$a = R_{12}R_{11} - R_{21}R_{22}$$

$$b = (R_{21} - R_{11})(1 - R_{12}R_{22})$$

$$c = R_{12}R_{21} - R_{22}R_{11}$$

Case 4:

The **Figure 68** shown below determines θ , with an assumption of a different ordering of the transformation matrix as shown in **Equation 4.18** once again,

$$\bar{S} = \begin{bmatrix} 1 & 0 & t_x \\ 0 & 1 & t_y \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \cos(\theta) & -\sin(\theta) & 0 \\ \sin(\theta) & \cos(\theta) & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} s_x & 0 & 0 \\ 0 & s_y & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad (4.18)$$

The inverse relation for scaling and translation was, once again, used to determine the relation between the position of the destination pixel and the translated source pixel position. The **Equation 4.19**, shown below, illustrates the relation between components of the translation vectors and the rotation and scale parameters of the transformation function,

$$t_x = x - \left[\frac{1}{s_x} (x_f - v_{Sx}) + (v_{Sx} - v_{Rx}) \right] \cos(\theta) - \left[\frac{1}{s_y} (y_f - v_{Sy}) + (v_{Sy} - v_{Ry}) \right] \sin(\theta) + v_{Rx}$$

$$t_y = y - \left[\frac{1}{s_x} (x_f - v_{Sx}) + (v_{Sx} - v_{Rx}) \right] \sin(\theta) + \left[\frac{1}{s_y} (y_f - v_{Sy}) + (v_{Sy} - v_{Ry}) \right] \cos(\theta) + v_{Ry}$$

(4.19)

Since, translation vector remains unchanged for each point in the set, three non-collinear consecutive points from the convex hull of the set could be selected, for the reference and template regions, to solve for the angle of rotation which is determined as the solution of the quadratic equation, with a, b and c defined below,

$$R_{11} = \frac{x_{1i} - x_{2i}}{y_{1i} - y_{2i}} \quad R_{12} = \frac{x_{1f} - x_{2f}}{y_{1f} - y_{2f}} \quad R_{21} = \frac{x_{2i} - x_{3i}}{y_{2i} - y_{3i}} \quad R_{22} = \frac{x_{2f} - x_{3f}}{y_{2f} - y_{3f}}$$

$$a = R_{12}R_{11} - R_{21}R_{22}$$

$$b = (R_{12} - R_{22})(1 - R_{21}R_{11})$$

$$c = R_{11}R_{22} - R_{12}R_{21}$$

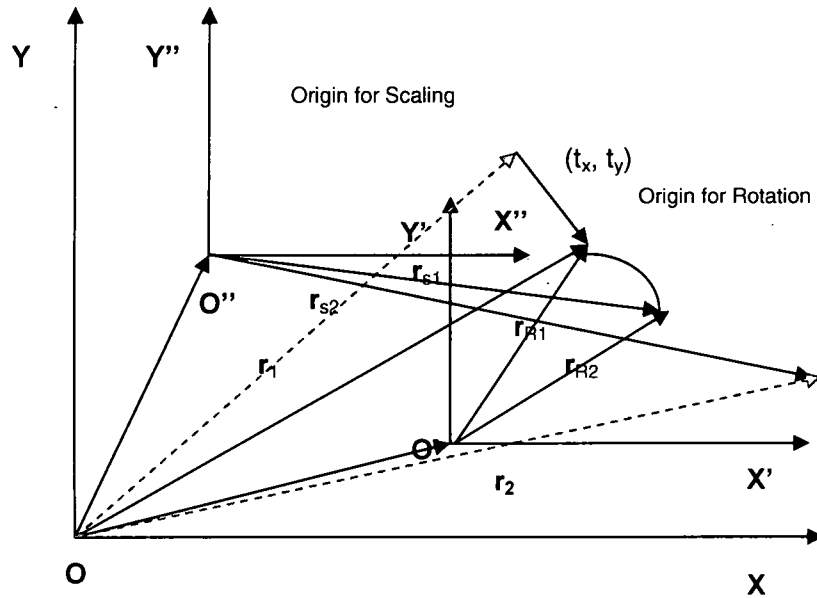


Figure 68: Vectorial technique demonstrating translation, rotation and scaling

Case 5:

The **Figure 69** shown below determines θ , with an assumption of a different ordering of the transformation matrix as shown in **Equation 4.20** once again,

$$\bar{S} = \begin{bmatrix} \cos(\theta) & -\sin(\theta) & 0 \\ \sin(\theta) & \cos(\theta) & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & t_x \\ 0 & 1 & t_y \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} s_x & 0 & 0 \\ 0 & s_y & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad (4.20)$$

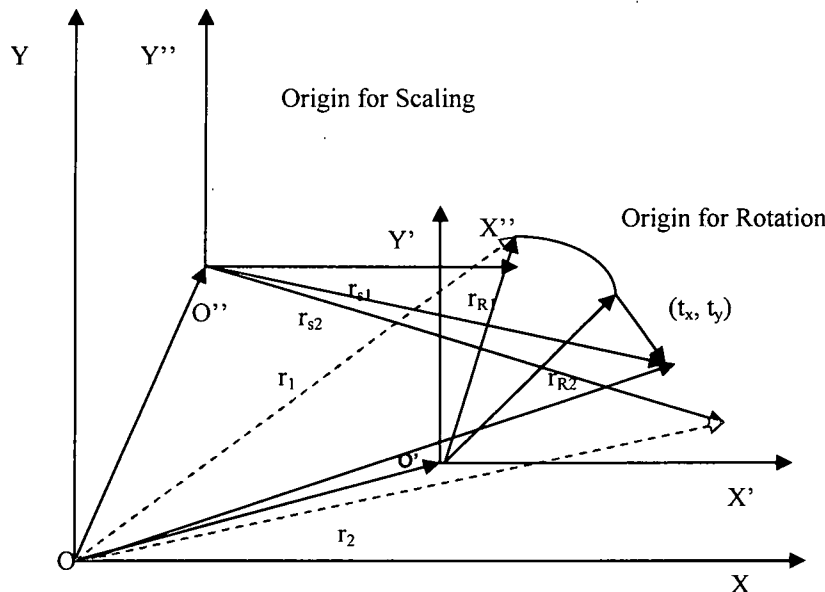
In determination of the angle of rotation below, the translation vector was determined and the components of this vector were used to solve a set of simultaneous equations. The angle of rotation is solved as a solution to the quadratic equation, with a, b and c defined below, for this particular ordering of the transformations as follows,

$$t_x = \left[\frac{1}{s_x} (x_f - v_{Sx}) + v_{Sx} \right] - (x - v_{Rx}) \cos(\theta) + (y - v_{Ry}) \sin(\theta) - v_{Rx} \quad (4.21)$$

$$R_{11} = \frac{x_{1i} - x_{2i}}{y_{1i} - y_{2i}} \quad R_{12} = \frac{x_{1f} - x_{2f}}{y_{1f} - y_{2f}} \quad R_{21} = \frac{x_{2i} - x_{3i}}{y_{2i} - y_{3i}} \quad R_{22} = \frac{x_{2f} - x_{3f}}{y_{2f} - y_{3f}}$$

$$b = (R_{22} - R_{12})(1 - R_{21}R_{11})$$

Case 6:



$$\begin{aligned}
t_x &= x_f - \left[s_x \left[(x - v_{Rx}) \cos(\theta) - (y - v_{Ry}) \sin(\theta) + (v_{Rx} - v_{Sx}) \right] + v_{Sx} \right] \\
t_y &= y_f - \left[s_y \left[(x - v_{Rx}) \sin(\theta) + (y - v_{Ry}) \cos(\theta) + (v_{Ry} - v_{Sy}) \right] + v_{Sy} \right]
\end{aligned}
\tag{4.23}$$

In determination of the angle of rotation above, the translation vector was determined and the components of this vector was used to solve a set of simultaneous equations. The angle of rotation is solved as a solution to the quadratic equation, with a, b and c defined below, for this particular ordering of the transformations. The steps of encountered in determining the solution have been shown in **Case I** and **II** and skipped for the rest of the cases, since the technique is very similar to each other, for all the cases described.

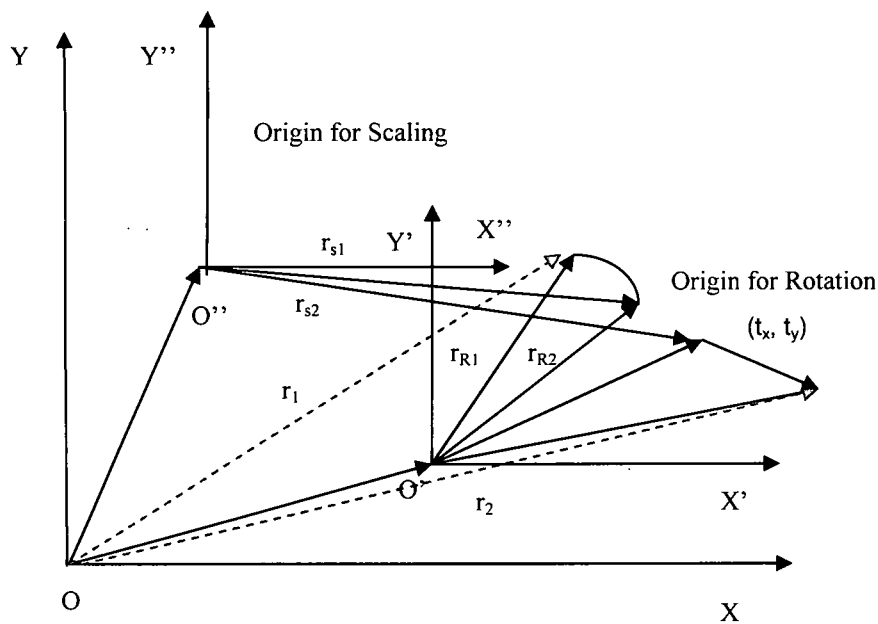


Figure 70: Vectorial technique demonstrating rotation before scaling and translation

$$\begin{aligned}
R_{11} &= \frac{x_{1i} - x_{2i}}{y_{1i} - y_{2i}} & R_{12} &= \frac{x_{1f} - x_{2f}}{y_{1f} - y_{2f}} & R_{21} &= \frac{x_{2i} - x_{3i}}{y_{2i} - y_{3i}} & R_{22} &= \frac{x_{2f} - x_{3f}}{y_{2f} - y_{3f}} \\
a &= R_{11} R_{12} - R_{22} R_{21} \\
b &= (R_{21} - R_{11})(1 - R_{22} R_{12}) \\
c &= R_{12} R_{21} - R_{22} R_{11}
\end{aligned}$$

Consistency Checking Algorithm

The solution for each of the cases above yields two solutions, one of which is the true value of the angle of rotation. This section discusses the result obtained from using the solution shown in six different cases above, by manually selecting a set of corresponding control points and then supplying these points as an input to an algorithm, which determines the correspondence.

The purpose of the consistency checking algorithm is to determine the parameters of the spatial mapping function. Since, the technique mentioned here determines the angle of rotation, the scaling factors as well as the translation in both directions, could easily be solved for, from this stage onwards. Hence, the technique is a sufficient one to serve the purpose of the consistency checking algorithm, even though it started out being a correspondence establishing algorithm, but failed due to its inability to detect the outliers.

Results and Performance Issues

The results obtained using the algorithm mentioned above, provided the following results for the registration of two images.

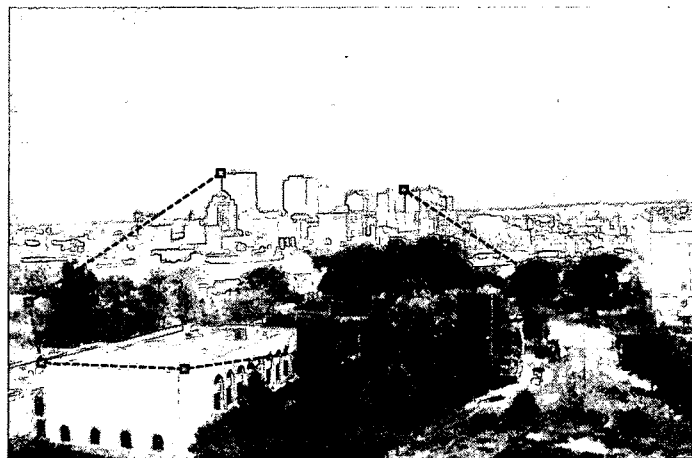


Figure 71: Convex hull of the set of control points from dayton3.tif

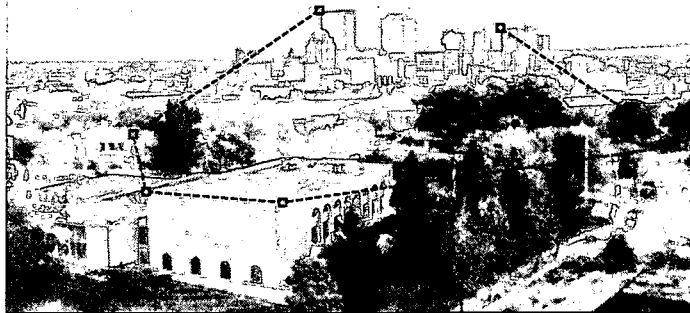


Figure 72: Convex hull of the set of control points from dayton4.tif

The correspondence and the results obtained for the angle of rotation are the following,

Angle of Rotation: -2.4591 degrees

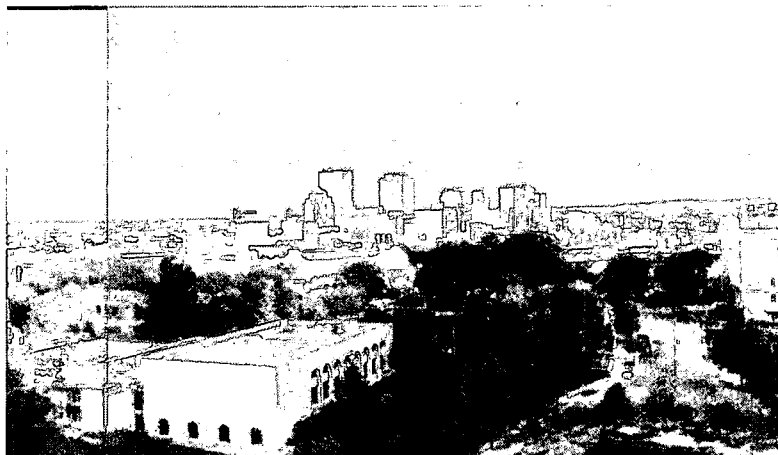


Figure 73: Registered Images using the affine spatial mapping function in MATLAB

Figure 75 shows that the images were properly registered and aligned and hence the point correspondence was achieved and the angle of rotation that was determined is correct. The good part of the result is evident from the overlapping images which show, that the correspondence was achieved on two images that have significant contrast and brightness differences.

The performance issue associated with this technique is its robustness, since, the outlier detection has not been achieved, and the determination of the spatial mapping function parameters should take a relatively small time compared to the similarity measure algorithm which needs to be highly accurate.

Chapter 5

Performance Benchmark

The performance of the algorithm is compared against a more robust correlation coefficient based similarity measure for the given set of images.

Table 10: Performance comparison with conventional correlation based technique

Reference Image	Template Image	No. of Outliers (Moment)	No. of Outliers (Correlation)	Time of Execution (Moment)	Time of Execution (Correlation)
dayton1.tif	dayton2.tif	0	0	29.613771	2.203616
dayton2.tif	dayton3.tif	4	0	147.37455	13.21489
dayton3.tif	dayton4.tif	1	0	81.065486	8.952308

The drawback of the correlation based technique is that it doesn't work when rotational transformation is present between the images. In absence of the need for contrast and brightness adjustment, the moment invariant based technique derived in this thesis works robustly for rotational transformation as well. The following compares the performance of the algorithms for the uniqueness measurement of each control point detected through feature extraction.

Table 11: Performance comparison for uniqueness measure with conventional correlation based technique

Reference Image	Time of Execution (Correlation)	Time of Execution (Moment)
dayton1.tif	0.49963	1.43257
dayton2.tif	1.58962	7.08208
dayton3.tif	2.12222	6.25429
dayton4.tif	0.87462	4.35234

The correlation based technique has an advantage at this stage over the technique developed in this thesis. The correlation based elimination would work robustly as well as eliminate more similar matches than the moment invariant based technique.

The correlation coefficient could be used to eliminate similar features in the algorithm developed in this thesis.

The following table compares the performance of the algorithms that were developed for similarity measure at subsequent stages of the development of the final algorithm. The time of execution and the number of outliers detected have improved at each stage of its development. All the timing measurements were made using the MATLAB time measurement facility.

Table 12: Time of execution of the three algorithms on MATLAB

	Time of Execution (in seconds)	No. of Outliers
Original and Texture Sum of Squared Difference	1.11146	Several
Exponential AMI with Range Scaled Texture	61.0134	23
Composition/Variation	40.0592	1

Chapter 6

Conclusions

A simple first order exponential affine moment invariant was derived to determine a single value of similarity between two regions of interest from a reference and a template image. The affine moment invariant is standardized by dividing its un-normalized value by the un-normalized affine moment invariant of the standard image. The moment invariant developed has the desirable property of weighing all the intensity values in the region with almost equal weights depending upon the position of the centroid, such that the contribution of intensity at each pixel is included in the summation. The distinguish-ability is based upon the signal content of each image. The ability of the similarity measure to detect the differences in spatial structures is fulfilled by filtering the image with a symmetric convolution kernel, which modifies the center pixel by subtracting from it, the average of the 8 neighboring pixels and taking the absolute value of the difference. The range of the resulting image is scaled either linearly or by using histogram specification. To provide the algorithm, the ability to identify where the changes are occurring, and the similarity is established iteratively. Circular regions of four different radii are used to mask the intensity values from the center of the image. The value of the affine moment invariant obtained is used to establish similarity at each iterative step and a threshold used to determine the presence of false matches. As a final step in determining similarity, a similar technique is followed with the rotation

invariant texture of the original region of interest. On developing the algorithm the elimination of all but one outlier was achieved using the images in **Appendix A**.

The drawback of the technique is that for a region with symmetrically identical distribution of intensity function, the similarity measure would remain high for distinct regions. The similarity obtained from the rotation invariant texture description could also fail due to the sparse information available even after range scaling. Secondly, the moment invariant derived is highly variant to changes in contrast, brightness and blur due to its dependence on fourth power of each intensity value, which enhances small differences. The elimination of the effect of these would lead to a higher computational complexity. Histogram specification is preferred method of choice in reducing the effect of contrast and brightness differences, but cannot be used if the sizes of the region of interest windows are different.

A geometric technique for the determination of the spatial mapping function parameters was also developed, which orders the control points by forming convex hull layers until only 1 or 2 points are left from the data set. The algorithm determines the angle of rotation requiring at least four points for reliable results. The angle of rotation could be used to determine the scale and translation.

The performance of the algorithm is the function of the window size chosen to determine the AMI. The computational complexity of the algorithm is $O(N^4)$ with $4 * N^4$ multiplication and $N^4 + N^2$ additions. A lot depends on the images that are being used for registration. If the feature extraction technique detects several control points in both images, the registration process would take a longer time. This implies the accuracy of the feature extraction technique determines the time it takes for the algorithm to execute.

Chapter 7

Future Research

- 1) Incorporation of scaling transformation into the similarity measure to determine the factor of the determinant of the Jacobian matrix.
- 2) Implementation of the affine moment invariant in the polar co-ordinate form to investigate the reduction in computational complexity of the algorithm. The expression for the affine moment invariant derived in **Section 4.1**, in the polar co-ordinate form is given by,

$$M = \iiint_{r_1, r_2} e^{r_1 r_2 \sin(\theta_2 - \theta_1) + r_2 R \sin(\tan^{-1}(\frac{y_c}{x_c}) - \theta_2) - r_1 R (\tan^{-1}(\frac{y_c}{x_c}) - \theta_1)} f(r_1 \cos(\theta_1) - x_c, r_1 \sin(\theta_1) - y_c) f(r_2 \cos(\theta_2) - x_c, r_2 \sin(\theta_2) - y_c) dr_1 dr_2 d\theta_1 d\theta_2$$

$$R = \sqrt{x_c^2 + y_c^2}$$

Due to the circularity of the polar co-ordinates, the implementation of the circular window for determining the moment invariant would be easier with this transformation. The co-ordinates $(r_1 \cos(\theta_1), r_1 \sin(\theta_1))$ and $(r_2 \cos(\theta_2), r_2 \sin(\theta_2))$ may lie on a non-integer position, in which case the intensity value at that co-ordinate would be determined via resampling from neighboring pixels. The position of the centroid still needs to be evaluated for every iteration.

- 3) The elimination of the outliers that are regions with symmetrically distributed intensity values and show high similarity in all iterations

- 4) Using the RISAR approach for an image which shows stronger invariance to higher amounts of rotation, for describing the texture instead of using the matrix described in **Chapter 3**, which causes changes in the intensity values of the center pixel for higher rotations.

References

- [1] Modersitzki, Jan. *Numerical Methods for Image Registration*. New York: Oxford University Press, 2004. ISBN 0 19 852841 8

- [2] Jähne, Bernd. Haußecker, Horst. Geißler, Peter. *Handbook of Computer Vision and Applications, Volume 2 Signal Processing & Pattern Recognition*. Academic Press, 1999. ISBN 0-12-379772-1

- [3] McDermott, J. & Adelson, E. H. (2004). *The geometry of the occluding contour and its effect on motion interpretation*. *Journal of Vision*, 4(10), 944-954

- [4] Gonzalez, Rafael C. Woods, Richard E. *Digital Image Processing, Second Edition*. Prentice-Hall, 2002. ISBN 0-201-18075-8

- [5] Haralick, Robert M. Shapiro, Linda G. *Computer and Robot Vision, Volume II*. Addison-Wesley, 1993.

- [6] Snapper, Ernst and Troyer, Robert J. *Metric Affine Geometry*. Academic Press, 1971

- [7] Ritter X., Gerhard, Wilson N., Joseph. *Handbook of Computer Vision Algorithm in Image Algebra*, CRC Press, ISBN 0-8493-2636-2

- [8] Flusser, Jan. *Moment Invariants in Image Analysis*, *Transaction on Engineering Computing and Technology* V11, February 2006 ISSN 1305-5313
- [9] Flusser, Jan., Zitova Barbara. *Image Registration Methods: A Survey*, *Image and Vision Computing* 21, 2003 (977-1000)
- [10] Preparata, Franco P., Shamos, Michael Ian, *Computational Geometry: An Introduction*, New York: Springer-Verlag, c1985
- [11] Grunbaum, Branko., Klee, Victor., Perles, M.A. and Shepard, G.C., *Convex Polytopes* Interscience [1967]
- [12] Toll, Bill. *Geometric Transformations*, *Computing and System Sciences*, Spring 1999
< <http://www.css.taylor.edu/~btoll/s99/424/res/mtu/Notes/geometry/geo-tran.htm>>
- [13] Cook, Diane., Swayne, Deborah F. *Interactive and Dynamic Graphics for Data Analysis with Examples using R and GGobi*,
<<http://www.public.iastate.edu/~dicook/ggobi-book>>, Aug 2006
- [14] Lewis, J.P. *Fast Normalized Cross-Correlation*. *Industrial Light and Magic*, 1995

Appendix

Appendix A

A.1 Images used for testing the algorithm



Figure 74: dayton1.tif



Figure 75: dayton2.tif

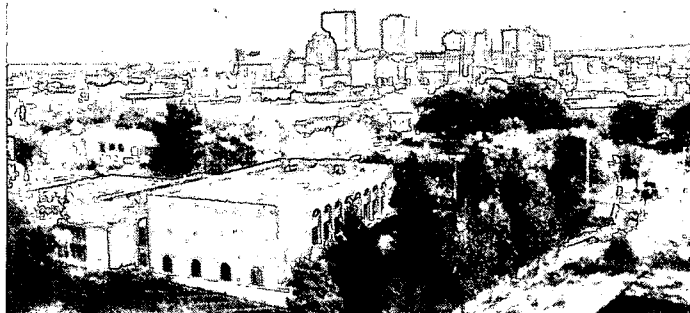


Figure 76: dayton3.tif

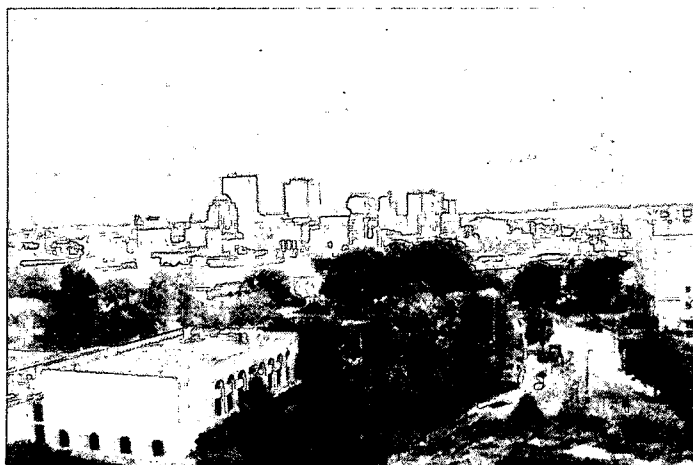
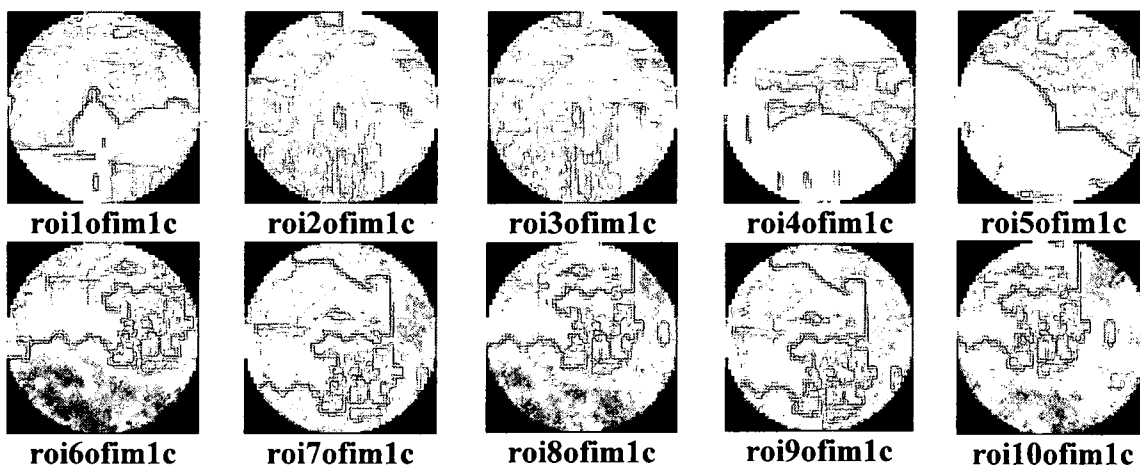
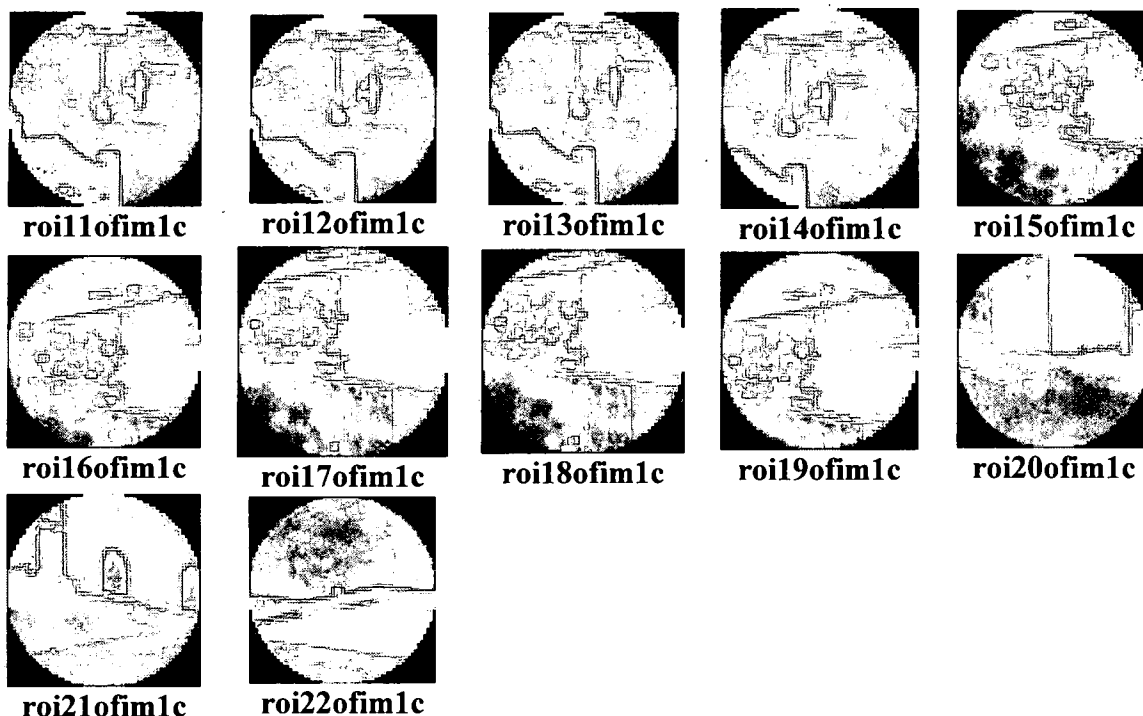


Figure 77: dayton4.tif

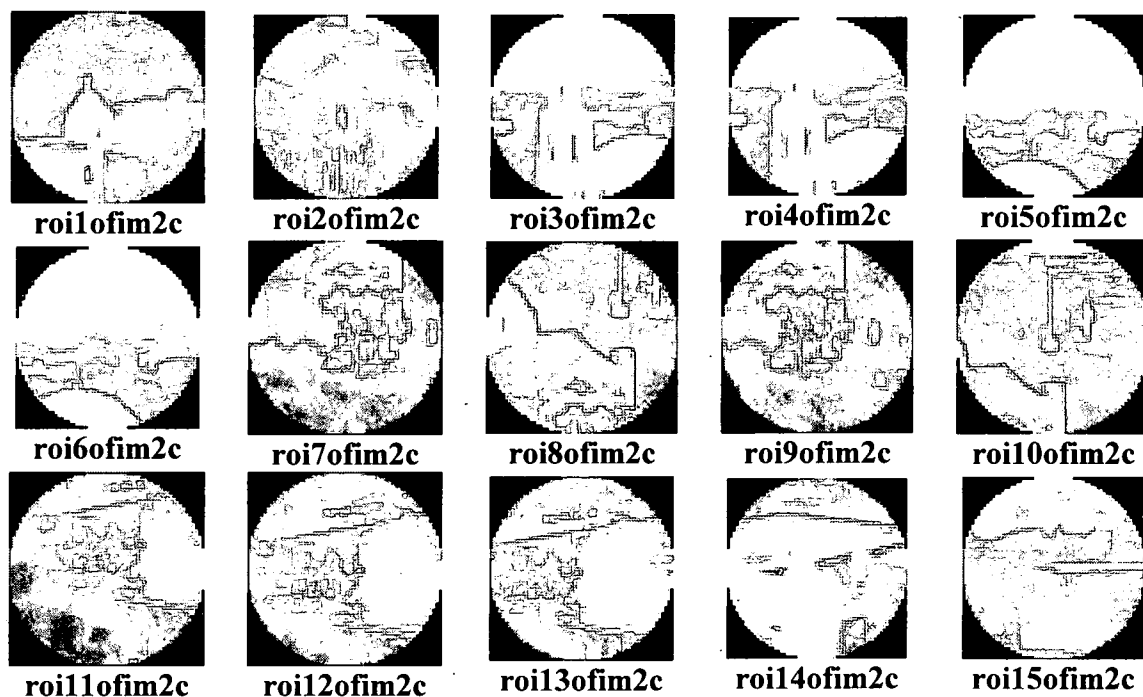
A.2 Regions of interest extracted from the images in A.1

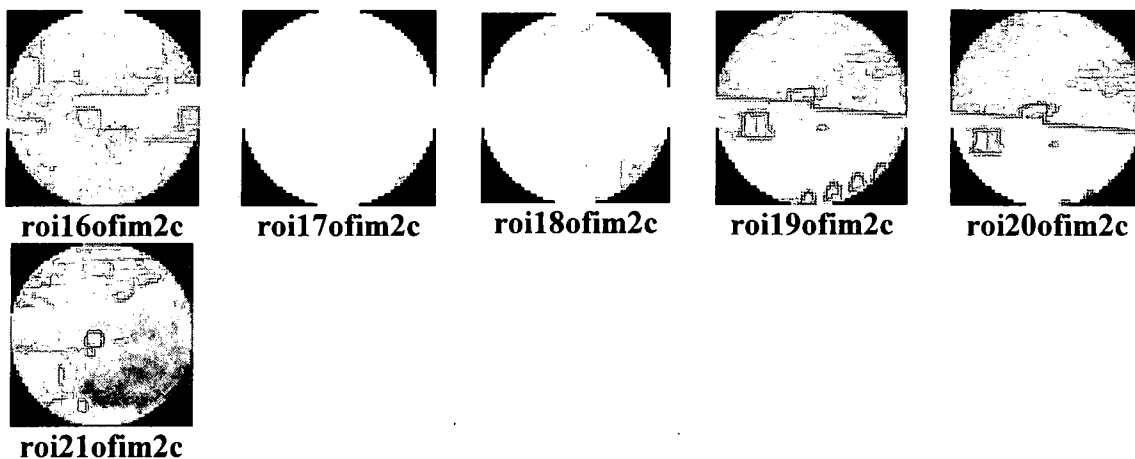
Data Set A





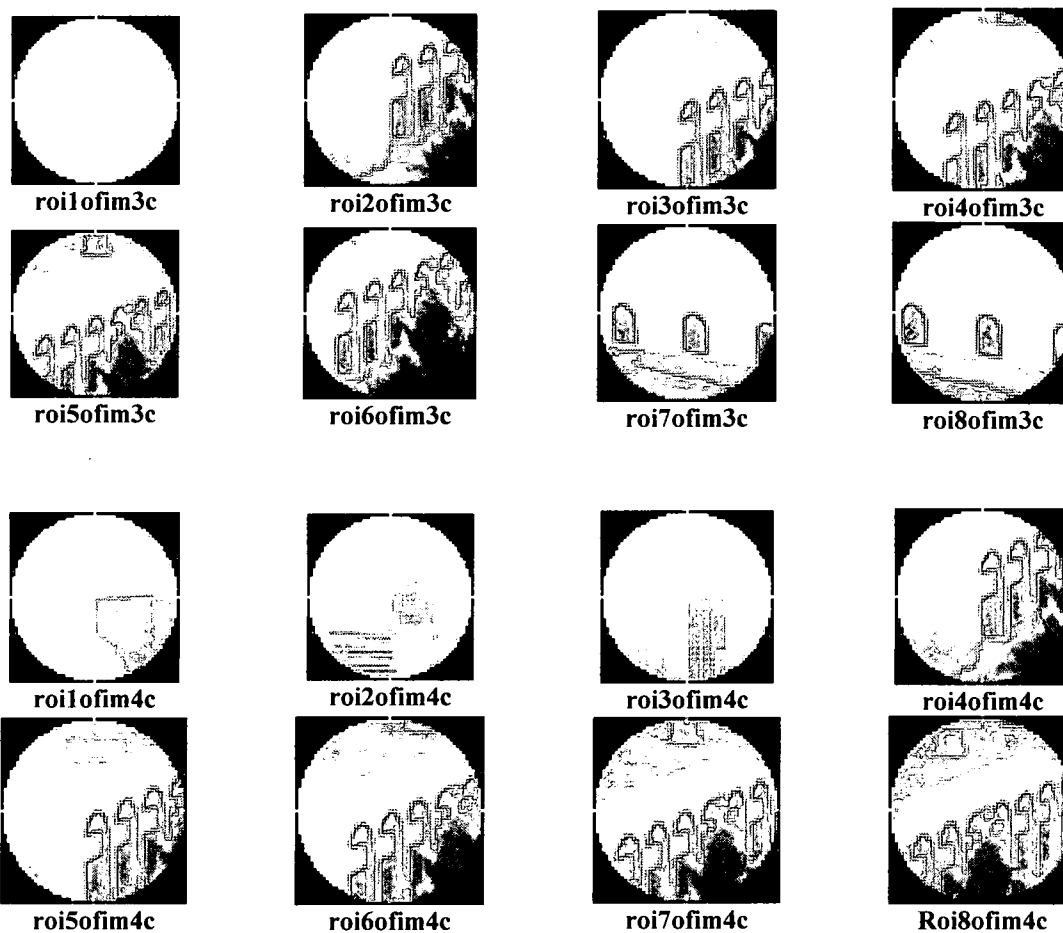
All the files were tested in the JPEG format, however, it would be better to use the tiff format to save the regions to prevent the loss of data each time the images are saved. All the images have been normalized with their ranges.





A.3 Regions of interest extracted from the images in A.1

Data Set B



R002593200